

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

МАТЕРИАЛЫ
VI Международной молодежной
научной конференции
«МАТЕМАТИЧЕСКОЕ
И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
ИНФОРМАЦИОННЫХ,
ТЕХНИЧЕСКИХ
И ЭКОНОМИЧЕСКИХ СИСТЕМ»

Томск, 24–26 мая 2018 г.

*Под общей редакцией
кандидата технических наук И.С. Шмырина*

Томск
Издательский Дом Томского государственного университета
2018

ББК 22.17–22.19
УДК 519.2, 519.7, 519.8
Т78

**ЧЛЕНЫ КОЛЛЕГИИ, РУКОВОДИТЕЛИ НАУЧНЫХ РЕДАКЦИЙ
ПО НАПРАВЛЕНИЯМ:**

д-р техн. наук, проф. **А.А. Глазунов** – научная редакция «Механика, математика»; д-р техн. наук, проф. **Э.Р. Шрагер** – научная редакция «Механика, математика»; д-р техн. наук, проф. **А.М. Горцев** – научная редакция «Информатика и кибернетика»; д-р техн. наук, проф. **С.П. Сущенко** – научная редакция «Информатика и кибернетика»; д-р физ.-мат. наук, проф. **В.Г. Багров** – научная редакция «Физика»; д-р физ.-мат. наук, проф. **А.И. Потекаев** – научная редакция «Физика»; д-р биол. наук, проф. **С.П. Кулижский** – научная редакция «Биология»; д-р геол.-минер. наук, проф. **В.П. Парначев** – научная редакция «Науки о Земле, химия»; канд. хим. наук, доц. **Ю.Г. Слизов** – научная редакция «Науки о Земле, химия»; д-р филол. наук, проф. **Т.А. Демешкина** – научная редакция «История, филология»; д-р ист. наук, проф. **В.П. Зиновьев** – научная редакция «История, филология»; д-р экон. наук, проф. **В.И. Канов** – научная редакция «Юридические и экономические науки»; д-р юрид. наук, проф. **В.А. Уткин** – научная редакция «Юридические и экономические науки»; д-р ист. наук, проф. **Э.И. Черняк** – научная редакция «Философия, социология, психология, педагогика, искусствознание»; д-р психол. наук, проф. **Э.В. Галажинский** – научная редакция «Философия, социология, психология, педагогика, искусствознание»

НАУЧНАЯ РЕДАКЦИЯ ТОМА:

д-р техн. наук, проф. **А.М. Горцев**, д-р физ.-мат. наук, доц. **Ю.Г. Дмитриев**, д-р физ.-мат. наук, проф. **В.В. Конев**, д-р техн. наук, проф. **А.Ю. Матросова**, д-р техн. наук, проф. **А.А. Назаров**, д-р техн. наук, проф. **К.И. Лившиц**, канд. техн. наук **И.С. Шмырин**

Т78 Труды Томского государственного университета. – Т. 302. Серия физико-математическая: Математическое и программное обеспечение информационных, технических и экономических систем : материалы VI Международной молодежной научной конференции. Томск, 24–26 мая 2018 г. / под общ. ред. И.С. Шмырина. – Томск : Издательский Дом Томского государственного университета, 2018. – 378 с.

ISBN 978-5-94621-755-2

Сборник содержит материалы VI Всероссийской молодёжной научной конференции «Математическое и программное обеспечение информационных, технических и экономических систем», проводившейся 24–26 мая 2018 г. на базе Института прикладной математики и компьютерных наук Томского государственного университета. Материалы сгруппированы в соответствии с работавшими на конференции секциями.

Для научных работников, преподавателей, аспирантов, магистрантов и студентов.

УДК 539.3.004
ББК 22,25.22.251.22.62

ISBN 978-5-94621-755-2

© Томский государственный университет, 2018

СЕКЦИЯ I. МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ МАШИН И КОМПЬЮТЕРНЫХ СЕТЕЙ

ПОИСК ЛОЖНЫХ ПУТЕЙ В ПОСЛЕДОВАТЕЛЬНОСТНЫХ СХЕМАХ С ИСПОЛЬЗОВАНИЕМ ROBDD*

С.В. Чернышов, А.Ю. Матросова

*Томский государственный университет
semen.cher@mail.ru, mau11@yandex.ru*

Введение

Одним из важнейших параметров физических схем является тактовая частота, которая определяет их быстродействие. Разработчики стремятся к тому, чтобы у схем была максимально возможная частота при условии корректного функционирования. Определение тактовой частоты заключается в нахождении путей с максимальными задержками. Существуют пути, не оказывающие влияния на функционирование схемы, такие пути называются ложными, и их можно не учитывать при определении тактовой частоты. Исключая такие пути из рассмотрения, можно увеличить скорость функционирования схем. В данной работе рассматривается алгоритм поиска ложных путей в последовательностной схеме. Приводятся детали реализации этого алгоритма, основанного на поиске тестовых пар для не робастно тестируемых задержек путей с помощью операций над Reduced Ordered Binary Decision Diagrams (ROBDD-графами). Предлагаемый алгоритм имеет ограничение на длину последовательности, доставляющей тестовую пару из начального внутреннего состояния. Алгоритм является развитием подхода, предложенного в [1].

Определение 1. Под путём схемы понимают последовательность элементов в комбинационной схеме (комбинационном эквиваленте последовательностной схемы), в которой выход предыдущего элемента является входом последующего, причём вход первого элемента является входом схемы, а выход последнего является выходом схемы.

Определение 2. Путь называется ложным, если не существует набора входных сигналов, в котором изменение входного сигнала изменяет выходной сигнал.

Выделяют робастно и не робастно тестируемые задержки путей.

Определение 3. Неисправность пути проявляется как робастно тестируемая, если существует пара тестовых наборов, обнаруживающих неисправность этого пути независимо от исправности других путей схемы.

Определение 4. Неисправность пути проявляется как не робастно тестируемая, если существует пара тестовых наборов, обнаруживающих неисправность этого пути при отсутствии неисправностей других путей схемы.

В данной работе рассматриваются только не робастно тестируемые неисправности.

Алгоритм основан на нахождении всех тестовых наборов v_2 тестовых пар (v_1, v_2) для взаимно инверсных перепадов значений сигналов рассматриваемого пути (rising transition и falling transition) в комбинационном эквиваленте последовательностной схемы. Тестовые наборы представляются двумя ROBDD-графами (для rising transition и falling transition). Определяется существование установочных последовательностей длинны не больше l , доставляющих тестовые пары из начального внутреннего состояния схемы. Если нет тестовой пары, обнаруживающей задержку пути ни для rising transition ни для falling transition, то путь считается ложным.

* Исследование выполнено за счет гранта Российского научного фонда (проект №14-19-00218)

Значения переменной, соответствующей началу пути, должны быть взаимно инверсными в тестовых парах (v_1, v_2) , причём вектор v_1 считается предыдущим, а вектор v_2 – последующим.

Для нахождения всех векторов v_2 предлагается подход, основанный на вычислении булевых разностей, причём булевы разности также вычисляются с помощью операций над ROBDD-графами.

1. Вычисление булевой разности для пути

Рассмотрим одновыходную комбинационную схему (комбинационный эквивалент последовательностной схемы) C с входами x_1, x_2, \dots, x_n . Пусть в схеме имеется путь α с последовательностью символов $x_i, u_1, u_2, \dots, u_{r-1}, u_r$, где r – длина рассматриваемого пути, x_i – начало пути и вход схемы, $u_1, u_2, \dots, u_{r-1}, u_r$ – внутренние переменные, сопоставляемые выходам элементов пути α , u_r – выход последнего элемента пути и, соответственно, выход схемы.

Пусть u_i и u_{i-1} – выходы соседних элементов пути α . Рассмотрим подсхему C_{u_i} схемы C с выходом u_i и входами $x_1, x_2, \dots, x_n, u_{i-1}$. Вычислим булеву разность данной подсхемы по переменной u_{i-1} . Для вычисления булевой разности предлагается сделать следующие шаги.

1. Строим ROBDD $R(f_{u_i})$, представляющий функцию подсхемы C_{u_i} . Данный ROBDD зависит от переменных $x_1, x_2, \dots, x_n, u_{i-1}$. При получении ROBDD первой в разложении Шеннона участвует переменная u_{i-1} .
2. С помощью $R(f_{u_i})$, вычисляются два ROBDD-графа: $R(f_{u_i}^{u_{i-1}=0})$ и $R(f_{u_i}^{u_{i-1}=1})$, корнями которых являются дочерние вершины корня $R(f_{u_i})$. Эти ROBDD получаются из $R(f_{u_i})$ путем подстановки вместе переменной u_{i-1} , значений 0 и 1, соответственно.
3. Следующим шагом является перемножение $R(\bar{f}_{u_i}^{u_{i-1}=0})$, $R(f_{u_i}^{u_{i-1}=1})$, а также $R(f_{u_i}^{u_{i-1}=0})$, $R(\bar{f}_{u_i}^{u_{i-1}=1})$ и объединение полученных при перемножении результатов, представляемых также в виде ROBDD-графа:

$$R(D_{u_i} / D_{u_{i-1}}) = R(f_{u_i}^{u_{i-1}=0}) \& R(\bar{f}_{u_i}^{u_{i-1}=1}) \vee R(f_{u_i}^{u_{i-1}=1}) \& R(\bar{f}_{u_i}^{u_{i-1}=0}).$$

Отметим, что для получения представления в виде ROBDD инверсной функции достаточно поменять местами терминальные вершины.

В результате выполнения перемножения

$$R(D_{u_r} / D_{u_{r-1}}) \& R(D_{u_{r-1}} / D_{u_{r-2}}) \& \dots \& R(D_{u_2} / D_{u_1}) \& R(D_{u_1} / D_{x_j})$$

получаем булеву разность D_{path} для пути α , заданную в виде $R(D_{\text{path}})$.

Каждый путь в ROBDD $R(D_{\text{path}})$, соединяющий корень с 1-терминальной вершиной, представляет конъюнкцию (интервал) из переменных x_1, x_2, \dots, x_n . Булевы векторы, обращающие $R(D_{\text{path}})$ в 1 – это векторы v_2 для пути α . Если данный вектор обращает схему в 1, то это вектор для rising transition, если в 0, то это вектор для falling transition.

Если $R(D_{\text{path}})$ – пустой граф, то исследуемый путь является ложным, и в последовательностной схеме и в комбинационной. Если этот граф не пустой, то путь α не ложный в комбинационной схеме.

Чтобы определить, является ли путь ложным или нет для последовательностной схемы, необходимо доставить тестовую пару (v_1, v_2) из начального внутреннего состояния q_0 с помощью установочной последовательности длины не больше l . Если такой установочной последовательности не существует, то путь считается ложным в рамках рассматриваемых ограничений.

2. Разделение тестовых наборов из ROBDD $R(D_{\text{path}})$ на наборы v_2 для rising transition и falling transition

ROBDD $R(D_{\text{path}})$ представляет тестовые наборы для rising и falling transition вместе. Их можно разделить следующим образом:

$$R_{\text{rise}} = R(D_{\text{path}}) \& x_i \cdot (R(D_{\text{path}}) \& \bar{x}_i),$$

$$R_{\text{fall}} = R(D_{\text{path}}) \& \bar{x}_i \cdot (R(D_{\text{path}}) \& x_i).$$

Пример.

Пусть имеется комбинационная схема следующего вида:

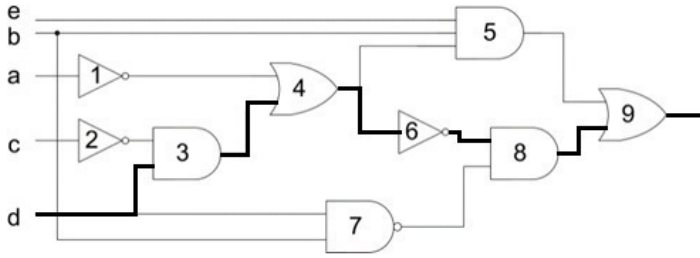


Рис. 1. Комбинационная схема С

Жирным на схеме выделен путь α , который можно представить последовательно элементом $d, u_3, u_4, u_6, u_8, u_9$.

D_{path} для α будет иметь следующий вид:

$$D_{u_9} / D_{u_8} = (u_5 \vee (u_8 = 1)) \oplus (u_5 \vee (u_8 = 0)) = \bar{u}_5 = \bar{b} \vee \bar{e} \vee ac \vee a\bar{d},$$

$$D_{u_8} / D_{u_6} = (u_6 = 1) \& u_7 \oplus (u_6 = 0) \& u_7 = u_7 = \bar{b} \vee \bar{d},$$

$$D_{u_4} / D_{u_3} = (\bar{a} \vee 1) \oplus (\bar{a} \vee 0) = a, \quad D_{u_3} / D_d = (\bar{c} \& 1) \oplus (\bar{c} \& 0) = \bar{c},$$

$$D_{\text{path}} = (\bar{b} \vee \bar{e} \vee ac \vee a\bar{d}) \& (\bar{b} \vee \bar{d}) \& a \& \bar{c} = a\bar{c}\bar{b} \vee a\bar{c}\bar{d}.$$

Отсюда:

$$D_{\text{rise}} = (a\bar{c}\bar{b} \vee a\bar{c}\bar{d}) \& \bar{d} = a\bar{c}\bar{d}, \quad D_{\text{fall}} = (a\bar{c}\bar{b} \vee a\bar{c}\bar{d}) \& d = a\bar{c}bd.$$

3. Определение существования установочной последовательности

Формируем ДНФ из внутренних состояний ROBDD R_{rise} и, используя её, получаем ROBDD R^{s_0} . Этот граф представляет все внутренние состояния, являющиеся частями полных состояний, представляемых векторами v_2 из тестовых пар (v_1, v_2) для rising transition пути α . Для смены значения, на выходе схемы необходимо, чтобы на вход схемы были доставлены последовательно два вектора, сначала v_1 , а потом v_2 из начального внутреннего состояния q_0 . Эти векторы обязательно отличаются значением переменной, соответствующей началу пути α .

Сначала приведем алгоритм определения существования установочной последовательности из [2] длины не больше l из начального состояния в некоторое состояние из множества внутренних состояний последовательностной схемы.

Пусть M^0 – некоторое множество внутренних состояний, представленных в виде ROBDD R^{s_0} , $R^{\bar{z}_i}$ – ROBDD-графы, представляющие функции переходов последовательностной схемы, соответствующие внутренним переменным z_i , причём эти ROBDD строятся путем выполнения разложений Шеннона с порядком переменных $z_1, z_2, \dots, z_p, x_1, x_2, \dots, x_n$.

Построим ROBDD R^{k_j} путём перемножения ROBDD графов, соответствующих переменным состояний конъюнкции k_j (с учётом инверсий над переменными). Конъюнкция k_j порождается путем ROBDD R^{s_0} , соединяющим корень с 1 терминальной вершиной. Заметим, что R^{k_j} представляет полные состояния.

Пусть $M(k_j)$ – множество внутренних состояний, представленных конъюнкцией k_j . Следовательно, ROBDD R^{k_j} задаёт все полные состояния, из которых возможен переход за один шаг в состояния $M(k_j)$.

Пусть $M(R^{k_j})$ – множество внутренних состояний, достижимых за один шаг перехода из полных состояний, представленных ROBDD R^{k_j} .

Утверждение 1. $M(k_j)$ содержит $M(R^{k_j})$.

Доказательство. Приведено в [2].

Утверждение 2. Начальный фрагмент δ пути, соединяющего корень ROBDD R^{k_j} с первой внутренней вершиной, помеченной символом входной переменной, представляет множество внутренних состояний, из которых некоторые состояния из $M(k_j)$ достижимы за один шаг перехода.

Доказательство. Приведено в [2].

В рамках данной задачи нас не интересует соответствие между предыдущим внутренним состоянием и следующим, достижимым за один шаг перехода состоянием, которое обеспечивается за счет предыдущего полного состояния. Поэтому мы можем рассматривать только предыдущие внутренние состояния. Следовательно, нас интересуют только δ -фрагменты путей ROBDD R^{k_j} . Эти фрагменты извлекаются следующим образом:

1. Удаляем из графа R^{k_j} все внутренние вершины, помеченные символами входных переменных.
2. Соединяем висячие рёбра с 1-терминальной вершиной;
3. Полученный граф упрощаем.

Упрощённый граф обозначаем $R^{k_j^*}$, он представляет все внутренние состояния, из которых можно перейти за один шаг во внутренние состояния из $M(k_j)$ за счёт подходящих входных состояний.

Для иллюстрации процедуры упрощения ROBDD-графа рассмотрим ДНФ следующего вида $\bar{x}_1 \bar{x}_2 \bar{z}_1 z_2 \vee x_1 \bar{x}_2 z_1 z_2 \vee x_1 x_2 \bar{z}_1 z_2 \vee x_1 x_2 z_1 \bar{z}_2 \vee \bar{x}_1 x_2 z_1 z_2$:

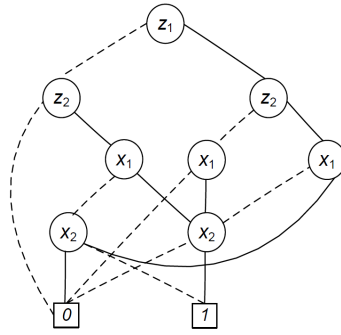


Рис. 2. ROBDD реализующая ДНФ

Выполняя шаги 1–3, получим последовательность графов (рис. 3, а–в).

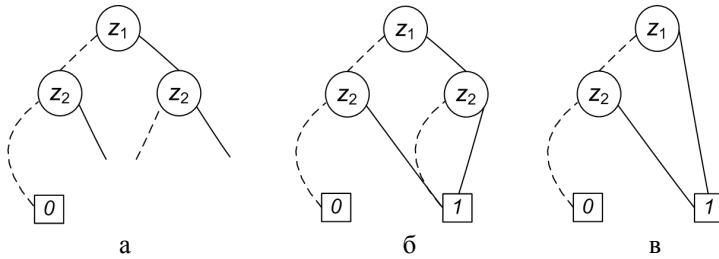


Рис. 3. а – удаление внутренних вершин, б – соединение висячих ребер с 1-терминальной вершиной, в – ROBDD $R^{k_j^*}$.

Получаем $R^{k_j^*}$ для всех путей, соединяющих корень ROBDD R^{s_0} с его 1-терминальной вершиной. Суммируем эти графы. В результате получим ROBDD R^{s_1} , который представляет все внутренние состояния, гарантирующие переход за один шаг в состояния множества M^0 .

Аналогичным образом строятся ROBDD $R^{s_2}, R^{s_3}, \dots, R^{s_l}$.

Утверждение 3. Если подстановка вектора q_0 в ROBDD R^{s_1} обращает его в 1, то существует переход за один шаг из состояния q_0 в некоторое внутреннее состояние из M^0 .

Доказательство. Приведено в [2].

На очередном шаге алгоритма выполняем подстановку вектора q_0 в ROBDD R^{s_i} . Если в результате этого R^{s_i} обращается в 1, тогда i – длина кратчайшей установочной последовательности в некоторое состояние из M^0 .

Утверждение 4. Если подстановка вектора q_0 в ROBDD R^{s_i} обращает граф в 1, то существует последовательность длины i , которая гарантирует достижение некоторого состояния из M^0 .

Доказательство. Приведено в [2].

Если $i = l$, но ROBDD R^{s_l} не обращается в 1 при подстановке вектора q_0 , то не существует установочной последовательности длины не больше l , которая достигает состояния из M^0 .

Для рассматриваемой задачи необходимо подать на вход схемы последовательно наборы v_1 и v_2 тестовой пары (v_1, v_2) . Для этого приходится корректировать первый шаг описанного алгоритма. Корректировка R^{s_i} выполняется следующим образом [1].

Каждое ребро переменной (с противоположным знаком инверсии), соответствующей началу пути α , заводят в 0-терминальную вершину ROBDD R^{s_i} . Это делается для

исключения перехода, при котором значение переменной, помечающей начало пути α , не изменяется. Скорректированный R^z пометим как $R^{z(1)}$.

Пример корректировки пути α с началом пути, помеченным x_1 , представлен на рис. 4.

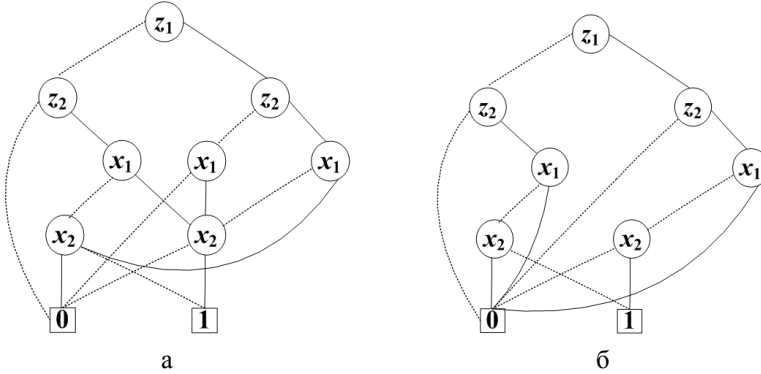


Рис. 4. а – R^z , б – $R^{z(1)}$.

ROBDD $R^{k_j(1)}$ получается перемножением скорректированных графов, соответствующих переменным конъюнкции k_j . Если рассматриваются установочные последовательности длины 2 и более, то для получения рекуррентных графов R^{k_j} используются не скорректированные ROBDD R^z .

Если не существует установочной последовательности для внутренних состояний R_{rise} и R_{fall} , то рассматриваемый путь является ложным.

4. Реализация алгоритма

Алгоритм определения существования установочной последовательности длины не больше l был реализован с использованием пакета CUDD 3.0.0, предназначенного для манипулирования ROBDD графами.

Процедура корректировки графов R^z выполнялась путём их умножения на инверсную переменную по отношению к переменной, отмечающей начало пути α .

На вход программа принимает файл с описанием схемы, начальное состояние схемы, ограничение на максимальную длину установочной последовательности, файл, в котором перечислены пути схемы, имеющие максимальную длину, а также набор файлов с описанием тестовых наборов для rising transition и falling transition для каждого из этих путей.

Результатом работы программы является файл, в котом приведены результаты для каждого пути, поданного на вход программы (ложный он или нет). Помимо этого формируется статистическая информация в целом для схемы, которая включает среднюю и максимальную длину установочной последовательности, число установочных последовательностей с длиной 1 (в процентах), и другое.

5. Экспериментальные результаты

Для экспериментов использовались бенчмарки MCNC с последовательностными схемами. Для каждой схемы рассматривалось сто путей, от входов схемы к выходам, причём если общее число путей в схеме меньше 100, то рассматривались все пути. Для каждого пути строились R_{rise} и R_{fall} . Для каждого примера проверялось существование установочных последовательностей длины не больше 100. Результаты приведены в таблице. Здесь **circuit** – название схемы, **num_i** – число входов, **num_o** – число выходов, **num_s_v** – число переменных состояний, **num_f_p** – число ложных путей в про-

центах среди всех рассматриваемых путей, **MAX_length** – максимальная длина установочной последовательности, **AVG_length** – средняя длина установочной последовательности, **num_t_s_1** – число установочных последовательностей с длиной 1 в процентах.

Судя по экспериментам, схемы имеют до 29% ложных путей. Средняя длина установочной последовательности среди всех схем 3,12. Число установочных последовательностей с длиной 1 может достигать 84,06%.

Таблица 1

Экспериментальные результаты

circuit	num_i	num_o	num_s_v	num_f_p (%)	MAX_length	AVG_length	num_t_s_1 (%)
S01494	8	19	8	1	20	10.82	12.4
bbara	4	2	4	3	6	2.5	30.8
bbsse	7	7	4	7	9	3.44	25.2
bbtas	2	2	3	2.63	5	2.09	40.3
beecount	3	4	3	6.5	3	1.3	74.8
cse	7	7	4	10	7	2.81	43.3
dk14	3	5	3	1	3	1.27	72.88
dk15	3	5	2	0	2	1.24	76.1
dk16	2	3	5	7	4	2.44	8.33
dk17	2	3	3	3	4	1.93	42.28
donfile	2	1	5	8	4	2.4	20
ex1	9	19	5	9	8	2.21	45.38
ex2	2	2	5	4	6	2.66	23.68
ex3	2	2	4	2	3	1.84	31.36
ex4	6	9	4	4	14	6.86	13.12
ex6	5	8	3	3	2	1.41	58.82
ex7	2	2	4	3	4	2.45	19.73
keyb	7	2	5	6	7	2.19	57.03
kirkman	12	6	4	17	9	4.03	14.29
lion	2	1	2	0	4	1.71	53.66
lion9	2	1	4	15.7	4	2.33	31.94
mark1	5	16	4	15	6	3.09	18.92
mc	3	5	2	3.7	3	1.54	58.43
opus	5	6	4	6	6	2.18	53.49
planet	7	19	6	29	22	10.51	14.44
s1	8	6	5	1	5	2.97	18.92
s1a	8	6	5	0	5	2.43	43.82
s8	4	1	3	0	3	1.82	29.58
sand	11	9	5	6	4	1.86	37.25
scf	27	56	7	15	15	10.43	4.28
sse	7	7	4	7	9	3.44	25.18
styr	9	10	5	11	7	3.13	39.16
tav	4	4	2	0	3	1.25	84.06
tbk	6	3	5	0	3	1.56	50.57

Заключение

Рассмотрен метод поиска ложных путей в последовательностной схеме, основанный на получении тестовых пар для не робастно тестируемых задержек путей, и реализованы основные алгоритмы, обеспечивающие реализацию метода. Разработанные алгоритмы позволили провести эксперименты на контрольных примерах.

ЛИТЕРАТУРА

1. Matrosova A.Y., Andreeva V.V., Chernyshov S.V., Rozhkova S.V., Kudin D.V. Finding False Paths in Sequential Circuits // Russian Physics Journal. – 2018. – V. 60. – № 10. P. 1837–1844.
2. Matrosova A., Andreeva V., Melnikov A. ROBDDs application for finding the shortest transfer sequence of sequential circuit or only revealing existence of this sequence without deriving the sequence itself // in Proc. of the 2016 IEEE East-West Design Test Symposium (EWDTS). – 2016. – P. 1–4.

О ПОВЫШЕНИИ ЗАЩИЩЁННОСТИ ЛОГИЧЕСКИХ СХЕМ ОТ ВНЕДРЕНИЯ ВРЕДОНОСНЫХ ПОДСХЕМ*

А.Ю. Матросова, В.А. Провкин
Томский государственный университет
maul1@yandex.ru, prowkan@mail.ru

Введение

При синтезе современных интегральных схем разработчики всё чаще прибегают к услугам сторонних фирм для реализации тех или иных компонент системы с целью снижения её стоимости. В компонентах, изготовленных сторонними фирмами, могут быть спрятаны вредоносные подсхемы (Trojan circuits) с целью разрушения системы или извлечения из неё конфиденциальной информации [1]. Вредоносные подсхемы обычно действуют в ситуациях, которые возникают в работающей системе чрезвычайно редко, поэтому они трудно обнаружимы как в процессе верификации схемы, так и в процессе её тестирования. Вредоносная подсхема (ТС) состоит из двух частей. Триггерная подсхема (Trojan trigger) включается при поступлении на её входы определенной комбинации значений сигналов. Вторая часть подсхемы (Trojan payload) является исполнительным устройством, включаемым триггерной подсхемой, которое может либо разрушить работу схемы, либо извлечь из неё секретную информацию. На рис. 1 внутренние полюсы IN_1 , IN_2 используются для включения триггерной подсхемы, и выход Trojan payload внедрён в линию между двумя элементами корректной схемы. При срабатывании ТС значение на выходе фрагмента (рис. 1) меняется с 0 на 1.

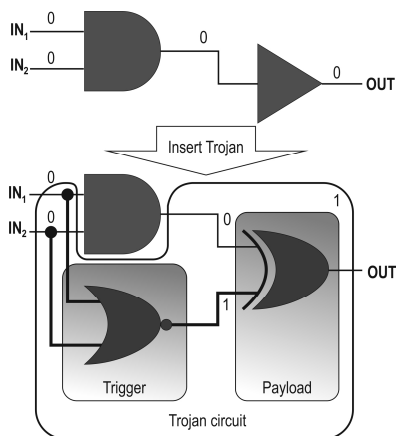


Рис. 1. Введение вредоносной подсхемы

Вредоносные подсхемы необходимо обнаруживать и, по возможности, нейтрализовать их действие.

1. Постановка задачи

Рассматривается проблема маскирования вредоносных подсхем в частично программируемых логических схемах, состоящих из вентилях и программируемых блоков памяти (Look-Up Table, LUT). В LUT могут быть свободные входы, которые можно использовать для коррекции схемы с целью маскирования в ней воздействия вредоносной подсхемы. Предполагается, в частности, что LUT может иметь один свободный вход. Исследуются возможности использования свободного входа в условиях перепрограммируемых LUT либо для маскирования вредоносной подсхемы в случае её обнаружения (в этой ситуации перепрограммируется один соответствующий LUT), либо для

* Исследование выполнено за счет гранта Российского научного фонда (проект №14-19-00218)

перепроектирования схемы таким образом, что включение в неё вредоносных подсхем оказывается затруднительным. Здесь речь идёт о превращении исходной логической схемы из вентилях в схему, защищённую от вредоносных подсхем. Это значит, что при сохранении спецификации схемы введение в неё вредоносных подсхем становится неэффективным – они обнаруживаются в процессе верификации схемы или в результате её тестирования.

Предполагается, что исходная схема состоит только из вентилях и реализует заданную спецификацию – систему полностью определённых булевых функций. Требуется покрыть некоторые её подсхемы программируемыми блоками памяти таким образом, чтобы сделать её, по возможности, максимально защищённой от вредоносных подсхем. Задача сводится к маскированию неисправных линий в схеме, константные неисправности которой трудно обнаружимы, т.е. множества тестовых наборов для таких неисправностей малы и не превышают некоторого порога [2]. Именно такие линии удобно использовать для подключения к ним вредоносных подсхем. В дальнейшем будем называть их «подозрительными» линиями.

Рассматривается комбинационная схема C (комбинационная составляющая последовательностной схемы), состоящая из вентилях. Используются способы маскирования одиночных константных неисправностей на линиях связи между логическими элементами с помощью LUTs, предложенные в работах [3,4]. Число входов программируемого блока (LUT) фиксировано. Один из входов может быть свободным, не используемым при покрытии вентилях схемы C . Для заданного множества «подозрительных» линий, к которым возможно подключение вредоносных подсхем, требуется выполнить покрытие фрагментов схемы C из вентилях программируемыми блоками так, чтобы при последующем их перепрограммировании замаскировать как можно большее количество этих линий.

2. Маскирование подозрительных линий

Пусть C является одно выходной комбинационной схемой. Рассмотрим способ маскирования, предложенный в [3] (рис. 2).

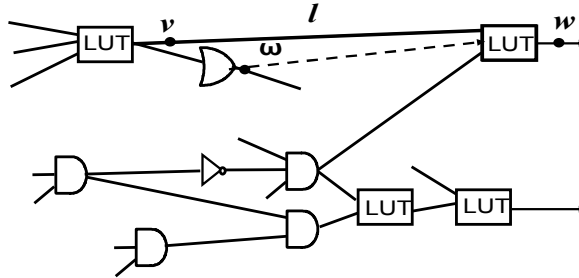


Рис. 2. Маскирование линии l , подключенной к входу LUT

Полнос v соединен линией l с входом u_i программируемого блока LUT, покрывающего подсхему C_{LUT} из вентилях. На рис. 2 линия l выделена жирным шрифтом. Входы подсхемы C_{LUT} являются либо входными, либо внутренними переменными схемы C . Число входных переменных LUT фиксировано, одна из переменных свободна, а функция от оставшихся переменных реализует подсхему из вентилях C_{LUT} , покрывающую соответствующий фрагмент из вентилях схемы C . Пусть в линию l включена вредоносная подсхема. Будем маскировать её, используя свободный вход LUT. В дальнейшем этот LUT будем называть корректирующим. Предполагаем, что в общем случае корректирующий LUT зависит от m переменных. Обозначим его выход символом w . Отметим, что позиция пунктирной линии на рис. 2 заранее не определена, она может соединять свободный вход LUT с выходом другого элемента схемы, при выполнении опре-

делённых условий. Предварительные исследования маскирования линии l способом, представленным на рис. 2, показали его редкую реализуемость. Поэтому был предложен иной способ маскирования, представленный на рис. 3.

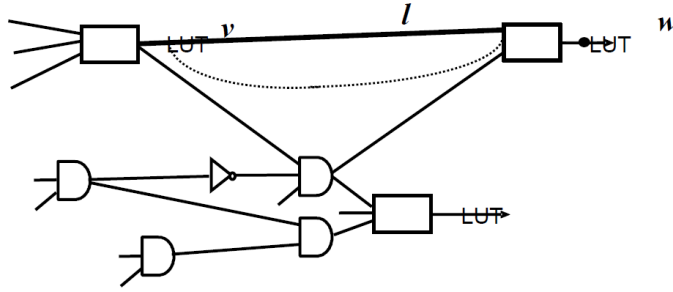


Рис. 3. Свободный вход LUT соединён с линией, дублирующей l

Для способа, предложенного на рис. 3, записываем в корректирующий программируемый блок единичные наборы полностью определенной функции, реализуемой подсхемой из вентилях, покрытой этим блоком и зависящей от его входных переменных, а именно от его $m - 1$ переменных. Каждому единичному набору сопоставляется два набора в пространстве m переменных: один с единичным значением переменной u_m , другой – с нулевым значением этой переменной.

При перепрограммировании LUT формируем функцию

$$f_{\text{LUT}}(u_i = 1) \wedge u_m \vee f_{\text{LUT}}(u_i = 0) \wedge \overline{u_m}$$

из единичных наборов функции корректирующего LUT. Эти наборы получены непосредственно по структуре подсхемы, покрытой корректирующим LUT, и теперь представляют функцию этого LUT в пространстве $u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_m$ его входных переменных. Она отличается от исходной функции LUT тем, что в ней переменная u_i заменена переменной u_m , и переменная u_m , (как прежде переменная u_i) теперь является существенной. С целью маскирования ТС делаем переменную u_i несущественной. Это значит, что каждому единичному набору функции

$$f_{\text{LUT}}(u_i = 1) \wedge u_m \vee f_{\text{LUT}}(u_i = 0) \wedge \overline{u_m}$$

необходимо сопоставить два набора в пространстве m переменных: один с единичным значением переменной u_i , а другой – с нулевым значением этой переменной. Тогда вредоносная подсхема, подключенная к входу u_i , не может изменить корректного поведения схемы C .

3. Покрытие линий, делающее их недоступными для включения вредоносных подсхем

Покрытие линий, исходящих из точек ветвления

Пусть дана комбинационная схема C , состоящая из вентилях, которые имеют только один выход (но при этом могут быть ветвления) и несколько входов. Рассмотрим задачу покрытия комбинационной схемы программируемыми блоками памяти, число входов которых на единицу больше максимального числа входов вентилях в исходной схеме. Предположим, что линии с низкой наблюдаемостью исходят из точек ветвления, причём из каждой точки ветвления выходит только одна «подозрительная» линия. Если это требование не выполняется, каждую из «подозрительных» линий маскируем способом, представленным на рис. 3, перепрограммируя его, как описано выше. На рис. 4 изображен фрагмент комбинационной схемы, в которой три элемента имеют ветвящиеся выходы. Соответственно нашим предположениям это три линии связи с низкой наблюдаемостью (на рис. они выделены жирными линиями), которые необходимо замаскировать.

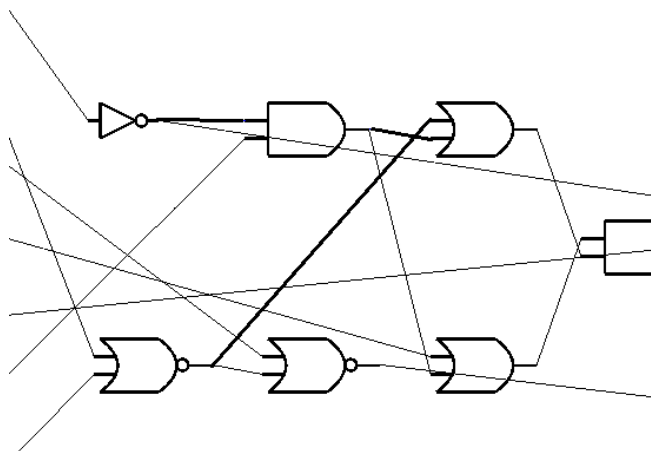


Рис. 4. Точки ветвления и «подозрительные» линии

В этом случае предлагается делать следующее: вместо элемента, в который ведёт «подозрительная» линия связи, ставится программируемый блок, сама линия связи удаляется, а к программируемому блоку также подводятся линии, которые получаются дублированием линий связи, идущих в элемент, из которого исходит «подозрительная» линия. При этом LUT программируется следующим образом: пусть элемент, в который ведёт «подозрительная» линия, реализует функцию $f(x_1, x_2, \dots, x_n)$, а элемент, из которого исходит «подозрительная» линия, реализует функцию $g(y_1, y_2, \dots, y_l)$. Пусть «подозрительная» линия соединена с входом x_i элемента, реализующего функцию f , тогда в LUT программируется следующая функция: $f(x_1, x_2, \dots, x_{i-1}, g(y_1, y_2, \dots, y_l), x_{i+1}, \dots, x_n)$. Новая функция зависит от $l + n - 1$ переменных. Если m – максимальное число входов вентиля в схеме, то для возможности покрытия элемента программируемым блоком должно выполняться следующее ограничение: $l + n - 1 \leq m + 1$. Заметим, что при таком способе покрытия «подозрительная» линия становится недоступной для использования.

Покрывание линий, не исходящих из точек ветвления

Некоторые линии могут быть покрыты таким образом, что входы и выходы линии содержатся в LUT. Такие LUT не перепрограммируются. Желательно покрывать таким способом как можно больше линий, поскольку они становятся недоступными для использования с целью введения вредоносных подсхем.

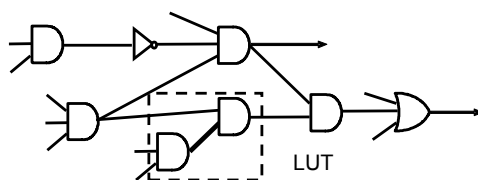


Рис. 5. Покрывание линии, не исходящей из точки ветвления

Предлагается сначала сделать как можно больше «подозрительных» линий недоступными для использования. После этого оставшиеся линии маскируются способом, представленным на рис. 3.

Заключение

Рассматриваются комбинационные схемы из вентилях. Для таких схем предложен алгоритм покрытия линий программируемыми блоками с целью снижения возможностей включения в эти линии вредоносных подсхем. Часть «подозрительных» линий

становятся недоступными для включения вредоносных подсхем, оставшиеся линии маскируются с помощью перепрограммируемых LUT. Разработана программа, реализующая описанные подходы к покрытию «подозрительных» линий. В дальнейшем предполагается проведение компьютерных экспериментов для контрольных примеров реальных логических схем.

ЛИТЕРАТУРА

1. *Матросова А.Ю., Останин С.А., Николаева Е.А.* Синтез частично программируемых схем, ориентированных на маскирование вредоносных подсхем (Trojan Circuits) // Труды Института системного программирования РАН. – 2017. – Т. 29. – № 5. – С. 61–74.
2. *Матросова А.Ю., Томков В.В.* Использование точных оценок управляемости и наблюдаемости для выявления вредоносных подсхем (Trojan Circuits) в последовательностных схемах // Известия вузов. Физика. – 2016. – Т. 59. – № 8/2. – С. 68–71.
3. *Yamashita S., Yoshida H., Fujita M.* Increasing yield using partially-programmable circuits // Workshop on Synthesis And System Integration of Mixed Information technologies (SASIMI). – 2010. – P. 237–242.
4. *Jo S., Matsumoto T., Fujita M.* SAT-based automatic rectification and debugging of combinational circuits with LUT insertions // Proc. Of IEEE Asian Test Symposium. – 2012. – P. 19–24.

ПОСТРОЕНИЕ ВСЕХ ТЕСТОВЫХ НАБОРОВ, ОБНАРУЖИВАЮЩИХ КОНСТАНТНЫЕ НЕИСПРАВНОСТИ ЛИТЕРАТУРЫ ЭНФ

М.И. Броч

Томский государственный университет
mishabroch@gmail.com

Введение

Для увеличения быстродействия схем высокого уровня интеграции необходимо обнаружение их ложных путей. Ложными путями в схеме являются пути, для которых изменение значения сигнала на входе пути не приводит к изменению значения сигнала на выходе схемы (комбинационной составляющей схемы с памятью). Ложные пути могут оказаться среди путей, характеризующихся самыми большими задержками. Эти задержки вычисляются по характеристикам элементов, составляющих путь. Самые большие задержки путей в схеме определяют её быстродействие: время между синхросигналами в схеме с памятью немного больше самой большой из задержек путей. Исключение ложных путей может привести к сокращению времени между синхросигналами и, следовательно, к повышению быстродействия схемы. В [1] проблема поиска ложных путей в схеме с памятью сведена к нахождению всех тестовых наборов, обнаруживающих константные неисправности литер эквивалентной нормальной формы (ЭНФ), сопоставляемых рассматриваемым путям. ЭНФ строится по комбинационной составляющей схемы с памятью. Далее находится последовательность, обеспечивающая доставку одного из тестовых наборов для константных неисправностей исследуемого пути. В данной работе ЭНФ компактно представляется И-ИЛИ деревом. По нему строится множество конъюнкций, порождающих тестовые наборы, которое затем представляется Reduced Ordered Binary Decision Diagram (ROBDD-графом). Затем выполняется перемножение полученного графа с ROBDD-графом схемы. Разработаны программы, реализующие алгоритм нахождения всех тестовых наборов для константной неисправности литеры ЭНФ.

1. Представление схемы размеченным И-ИЛИ деревом

Ранее было показано [2], что все пути схемы представляются размеченным И-ИЛИ деревом, причём каждой концевой вершине дерева сопоставляется единственный путь в схеме. Задана комбинационная схема (рис. 1):

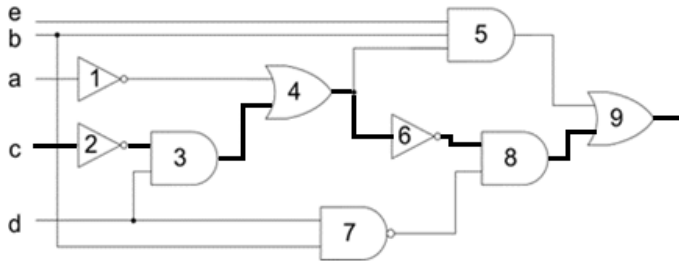


Рис. 1. Комбинационная схема

Представим её в виде BLIF (Berkeley Logic Interchange Format) – представление данных, целью которого является описание иерархической (комбинационной) схемы логического уровня в текстовой форме. Схема может рассматриваться как ориентированный граф комбинационных логических узлов и последовательных логических элементов. Автором написана программа, которая, используя такое представление данных, строит размеченное И-ИЛИ дерево. Далее в размеченном И-ИЛИ дереве находится конечная вершина, соответствующая заданному пути в схеме, с использованием алгоритма, представленного в [2].

На рис. 2 комбинационная схема рис. 1 представлена размеченным деревом с выделенной вершиной, сопоставляемой одному из путей схемы.

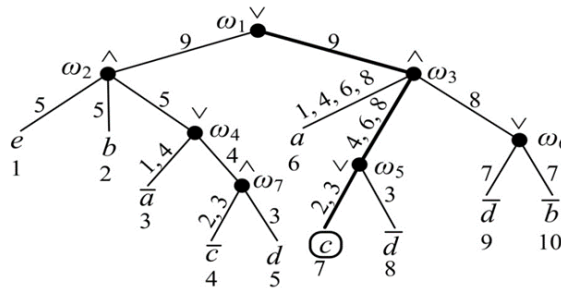


Рис. 2. Размеченное дерево для схемы рис. 1

2. Коррекция И,ИЛИ дерева относительно выделенной конечной вершины

После того, как в размеченном дереве найдена интересующая нас конечная вершина, переходим к построению всех тестовых наборов для b_p неисправности литеры, сопоставляемой рассматриваемому пути. Под b_p неисправностью понимается замена константой 1 каждого вхождения литеры ЭНФ, сопоставляемой рассматриваемому пути. Теперь нам нет необходимости использовать размеченное дерево, можно убрать разметки его рёбер и использовать неразмеченное И-ИЛИ дерево для дальнейших вычислений.

Далее неразмеченное (размеченное) дерево преобразуем таким образом, что выделенная вершина рассматриваемого пути оказывается крайней левой. Поясним необходимую последовательность действий на примере дерева рис. 2.

Вершина c является крайней левой относительно ближайшей к ней неконцевой вершине ω_5 . Переходим к соседней, более близкой к корню, неконцевой вершине ω_3 . Ребро, связывающее обе вершины, сделаем крайним левым вместе с инцидентным ему поддеревом. Наконец, ребро, связывающее вершины ω_3 , ω_1 вместе с инцидентным ему поддеревом сделаем крайним левым. В результате получаем дерево рис. 3.

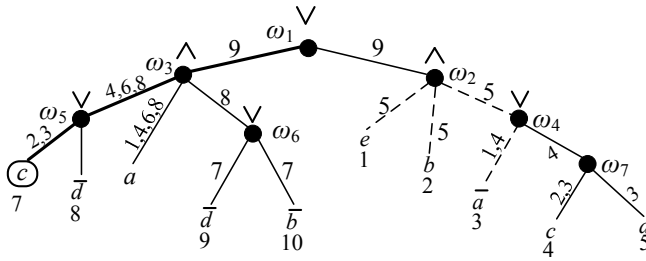


Рис. 3. Скорректированное размеченное дерево схемы рис. 1

Выделим существенное поддерево в И-ИЛИ дереве. Двигаемся от корня к конечным вершинам. Если очередная проходима вершина помечена символом \wedge , то включаем в существенное поддерево все исходящие из вершины рёбра, если символом ∇ – включаем одно исходящее ребро. На рис. 3 существенное поддерево выделено пунктиром. Жирной линией отмечен путь с вершиной c . Существенные поддеревья можно строить как для неразмеченных, так и размеченных И-ИЛИ деревьев. Нас интересуют конъюнкции, содержащие литеру, сопоставляемую рассматриваемому пути, и принадлежащие множеству конъюнкций, порождающих все тестовые наборы для b_p неисправности литеры.

3. Построение всех тестовых наборов

Тестовый набор, как известно, обеспечивает распространение смены значения сигнала в присутствии неисправности от места её возникновения до выхода схемы. В нашем случае – от входа схемы вдоль заданного пути.

Пусть K – найденная по дереву конъюнкция; она может быть пустой. Будем говорить, что K расширяема по литере x_i^α (здесь i – индекс входной переменной, α – путь, представляемый последовательностью номеров составляющих его элементов), если выбрасывание из неё этой литеры приводит к образованию непустой конъюнкции K^* , являющейся импликантой функции. Иначе она не расширяема по этой литере.

Будем рассматривать K^* как результат склеивания по литере x_i^α конъюнкций K и \underline{K} , в т.ч. и для пустой конъюнкции K . Здесь \underline{K} получается из K заменой литеры x_i^α на инверсную литеру. Будем называть \underline{K} дополнением K по литере x_i^α или просто дополнением K .

Построение всех тестовых наборов для b_p неисправности

Из неразмеченного И-ИЛИ дерева сокращённым перебором существенных поддеревьев, предложенным в [3], извлекаем все конъюнкции, порождающие тестовые наборы для b_p неисправности литеры. С этой целью корректируем процедуру отбрасывания частично построенных конъюнкций.

Не достраиваем частично построенные конъюнкции, содержащие повторяющиеся переменные, одна из которых отмечает начало рассматриваемого пути. Повторяющиеся считаются переменные, знаки инверсий которых совпадают.

Не достраиваем частично построенные конъюнкции, содержащие взаимно инверсные переменные, отличающиеся от переменной, отмечающей начало рассматриваемого пути.

Из множества конъюнкций, порождающих все тестовые наборы для b_p неисправности литеры формируем множество конъюнкций типа \underline{K} , в том числе и из подходящих пустых конъюнкций.

Каждая из конъюнкций типа \underline{K} приводится к нормальному виду исключением повторяющихся литер. По полученному множеству элементарных конъюнкций строится ROBDD-граф $R(\underline{K})$.

По И-ИЛИ дереву строится ROBDD-граф $R(C)$ исследуемой схемы.

Инверсная функция схемы C представляется графом $\overline{R(C)}$, который получается из $R(C)$ перестановкой терминальных вершин. Перемножив $R(\underline{K})$ и $\overline{R(C)}$, получаем ROBDD-граф $R(b_p)$, компактно представляющий все тестовые наборы для b_p неисправности.

Построение всех тестовых наборов для a_p неисправности

Построение всех тестовых наборов для a_p неисправности сводится к построению всех тестовых наборов для b_p неисправности инверсного И-ИЛИ дерева. Инверсное И-ИЛИ дерево получается из описанного ранее дерева заменой разметок внутренних вершин и знаков инверсий концевых вершин. Символ \vee заменяется символом \wedge , символ \wedge заменяется символом \vee .

Перемножив соответствующие $R(\underline{K})$ и $R(C)$ (вместо $\overline{R(C)}$, здесь используем $R(C)$), получаем ROBDD-граф $R(a_p)$, компактно представляющий все тестовые наборы для a_p неисправности.

Заключение

Реализованы программы, позволяющие представить комбинационную схему (Benchmarks), описанную в BLIF формате, в виде размеченного И-ИЛИ дерева, находить нужную концевую вершину в дереве, соответствующую заданному пути, отыскивать все конъюнкции, которые могут порождать b_p неисправности литеры. Далее, используя CUDD (пакет для осуществления операций над BDD-графами), строим ROBDD-графы и при их перемножении получаем множество всех тестовых наборов для константных неисправностей литеры.

ЛИТЕРАТУРА

1. Матросова А.Ю., Андреева В.В., Чернышов С.В., Рожкова С.В., Кудин Д.В. Обнаружение ложных путей в последовательностных схемах // Известия вузов. Физика. – 2017. – Т. 60. – № 10. – С. 170–178.
2. Матросова А.Ю., Останин С.А., Сингх В. Обнаружение несущественных путей логических схем на основе совместного анализа и-или деревьев и SSBDD-графов // Автоматика и телемеханика. – 2013. – № 9. – С. 126–142.
3. Матросова А.Ю. Алгоритмические методы синтеза тестов. – Томск, Издательство Томского университета, 1990. – 206 стр.

ИССЛЕДОВАНИЕ И РАЗРАБОТКА АЛГОРИТМОВ УПРОЩЕНИЯ КНФ ПРИ РЕШЕНИИ ПРОБЛЕМЫ ВЫПОЛНИМОСТИ

Н.Н. Пашкова, В.В. Андреева

Томский государственный университет
neldnelaps@gmail.com, avv.21@mail.ru

Введение

Сложность проектирования сверхбольших интегральных схем (СБИС) значительно возросла за последние годы. Большая часть времени теперь тратится на разработку тестов, которые обеспечивают быстрое обнаружение неисправностей, возникших в процессе производства схемы.

Методы тестирования разделяются на встроенные и внешние. Наиболее перспективным представителем встроенного метода является Built-In Self-Test (BIST) или

встроенное самотестирование. Однако при таком подходе к тестированию не все неисправности из рассматриваемого класса могут быть обнаружены, кроме того, для реализации этого подхода требуется дополнительная аппаратура. В ситуации, при которой требуется большое покрытие неисправностей тестами, используются методы внешнего тестирования. Метод внешнего тестирования основан на генерации наборов тестовой последовательности, обнаруживающей заданную неисправность. Рассматриваются синхронные логические схемы, начальное состояние которых известно. Множество тестовых наборов для заданной неисправности строится по комбинационной составляющей синхронной схемы. Обычно находятся все тестовые наборы. Тестовые наборы представляют собой полные состояния, содержащие входную составляющую и внутреннее состояние схемы. Последовательность входных векторов (установочная последовательность), переводящая схему из начального состояния в некоторое внутреннее состояние из заданного множества, представленного тестовыми наборами, является искомой тестовой последовательностью. Именно эта последовательность позволяет обнаружить заданную неисправность. В [1] поиск установочной последовательности сведен к операциям над ROBDD-графами, извлекаемыми из комбинационной составляющей синхронной схемы. В данной работе предлагается альтернативный подход, заключающийся в сведении к булевой выполнимости конъюнктивной нормальной формы (КНФ), строящейся для комбинационного эквивалента, состоящего из нескольких комбинационных составляющих. КНФ полезно предварительно упростить с целью сокращения вычислений при решении задачи выполнимости.

1. Построение КНФ разрешения

Рассматривается многовыходная комбинационная составляющая синхронной схемы. Логические элементы схемы (вентили) реализуют элементарные логические функции: НЕ, И, ИЛИ, И-НЕ, ИЛИ-НЕ и другие.

Функцией разрешения элемента, который реализует функцию $y = f(z_1, z_2, \dots, z_n)$, называется функция $\varphi(y, f) = y \sim f(z_1, z_2, \dots, z_n)$.

КНФ разрешения элемента – это представление функции $\varphi(y, f)$ в форме КНФ. КНФ разрешения для элементарных функций алгебры логики:

- $y = \bar{z} \rightarrow (z \vee y) \wedge (\bar{z} \vee \bar{y})$;
- $y = z_1 \wedge z_2 \wedge \dots \wedge z_n \rightarrow (z_1 \vee \bar{y}) \wedge (z_2 \vee \bar{y}) \wedge \dots \wedge (z_n \vee \bar{y}) \wedge (\bar{z}_1 \vee \bar{z}_2 \vee \dots \vee \bar{z}_n \vee y)$;
- $y = \bar{z}_1 \vee \bar{z}_2 \vee \dots \vee \bar{z}_n \rightarrow (\bar{z}_1 \vee y) \wedge (\bar{z}_2 \vee y) \wedge \dots \wedge (\bar{z}_n \vee y) \wedge (z_1 \vee z_2 \vee \dots \vee z_n \vee \bar{y})$;
- $y = z_1 \wedge z_2 \wedge \dots \wedge z_n \rightarrow (z_1 \vee y) \wedge (z_2 \vee y) \wedge \dots \wedge (z_n \vee y) \wedge (\bar{z}_1 \vee \bar{z}_2 \vee \dots \vee \bar{z}_n \vee \bar{y})$.

Примечание: КНФ разрешения инвертора $y = \bar{z}$ можно опустить с помощью замены всех вхождений переменной y на \bar{z} в КНФ разрешения схемы.

На рис. 1 представлен пример комбинационной составляющей синхронной схемы.

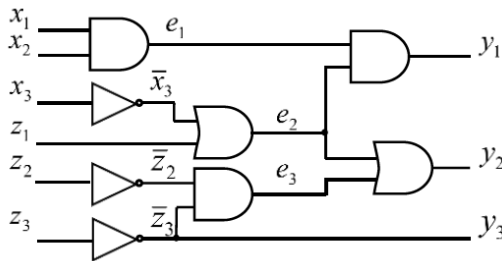


Рис. 1. Пример комбинационной составляющей

КНФ разрешения всех элементов комбинационной схемы рис. 1 имеют следующий вид:

- $e_1 = x_1 \wedge x_2 \rightarrow (x_1 \vee \bar{e}_1) \wedge (x_2 \vee \bar{e}_1) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee e_1)$;
- $e_2 = \bar{x}_3 \vee z_1 \rightarrow (x_3 \vee e_2) \wedge (\bar{z}_1 \vee e_2) \wedge (\bar{x}_3 \vee z_1 \vee \bar{e}_2)$;
- $e_3 = \bar{z}_2 \wedge \bar{z}_3 \rightarrow (\bar{z}_2 \vee \bar{e}_3) \wedge (\bar{z}_3 \vee \bar{e}_3) \wedge (z_2 \vee z_3 \vee e_3)$;
- $y_1 = e_1 \wedge e_2 \rightarrow (e_1 \vee \bar{y}_1) \wedge (e_2 \vee \bar{y}_1) \wedge (\bar{e}_1 \vee \bar{e}_2 \vee y_1)$;
- $y_2 = e_2 \vee e_3 \rightarrow (\bar{e}_2 \vee y_2) \wedge (\bar{e}_3 \vee y_2) \wedge (e_2 \vee e_3 \vee \bar{y}_2)$;
- $y_3 = \bar{z}_3$.

КНФ разрешения комбинационной составляющей схемы рис. 1 представляет собой логическое перемножение КНФ разрешения всех её элементов:

$$(x_1 \vee \bar{e}_1) \wedge (x_2 \vee \bar{e}_1) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee e_1) \wedge (x_3 \vee e_2) \wedge (\bar{z}_1 \vee e_2) \wedge (\bar{x}_3 \vee z_1 \vee \bar{e}_2) \wedge (\bar{z}_2 \vee \bar{e}_3) \wedge (\bar{z}_3 \vee \bar{e}_3) \wedge (z_2 \vee z_3 \vee e_3) \wedge (e_1 \vee \bar{y}_1) \wedge (e_2 \vee \bar{y}_1) \wedge (\bar{e}_1 \vee \bar{e}_2 \vee y_1) \wedge (\bar{e}_2 \vee y_2) \wedge (\bar{e}_3 \vee y_2) \wedge (e_2 \vee e_3 \vee \bar{y}_2) \wedge \bar{z}_3.$$

2. Методы упрощения КНФ

Проблема состоит в том, что исходные КНФ комбинационных схем имеют большие размерности для практических задач. Желательно упростить исходную КНФ, если это возможно. В дальнейшем будем рассматривать комбинационные эквиваленты синхронной схемы фиксированной длины и представлять их в виде КНФ. Блоки комбинационной логики комбинационного эквивалента представлены на рис. 2.

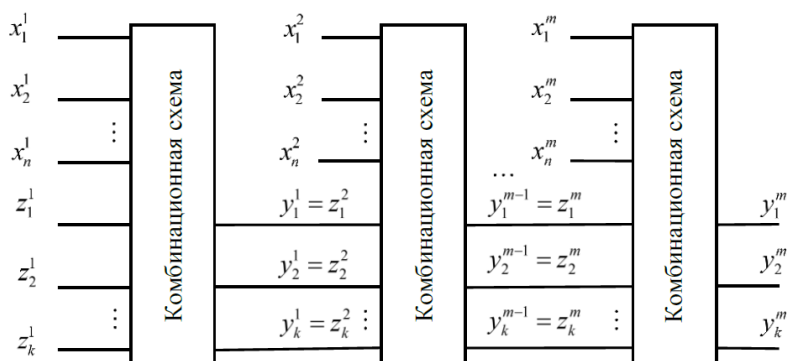


Рис. 2. Блоки комбинационной логики

КНФ комбинационного эквивалента является формулой, сложность которой растет с увеличением длины комбинационного эквивалента. Разным блокам сопоставляются аналогичные последовательности дизъюнктов, отличающиеся друг от друга символами переменных.

Известные методы упрощения КНФ делятся на методы, ориентированные на произвольные КНФ, и методы, учитывающие специфику КНФ. К первым методам относятся следующие.

Метод 1. Метод резолюции позволяет получить КНФ с меньшим количеством дизъюнктов и литералов, эквивалентную исходной КНФ [2]. Данный метод основан на правилах исчисления высказываний и включает в себя ряд методик:

1. Простая резолюция и разрешение уникальных переменных.
2. Резолюция по соседним дизъюнктам. Два дизъюнкта являются соседними, если они различаются знаком по единственному литералу. Резольвентой по соседним дизъюнктам называется дизъюнкт, являющийся максимально общим подмножеством соседних дизъюнктов. Если в КНФ присутствует конъюнкция соседних дизъюнктов, то конъюнкция заменяется резольвентой по соседним дизъюнктам.

3. Бинарная резолюция. Два дизъюнкта разрешимы относительно бинарной резолюции если они совпадают хотя бы по одной переменной, которая входит в один дизъюнкт с отрицанием, а в другой – без.

Метод 2. Норм минимизация путем итеративного разложения позволяет получить эквивалентную исходной КНФ, которая имеет минимально возможное число конъюнкций [3]. Общая идея метода состоит в следующем:

1. Построение импликационного графа заданной КНФ.
2. Выделение сильных компонент с дальнейшим переключением (например, замена x на \bar{x}), пока это возможно.
3. Удаление негативных конъюнкций из КНФ.

Метод 3. Алгоритм минимизация функции component-wise quadratic (CQ) позволяет получить КНФ с минимальным числом конъюнкций и минимальным числом литералов [4]. Основная идея алгоритма заключается в следующем:

1. Построение импликационного графа заданной КНФ;
2. Построение графа конъюнкций и поиск сильных компонент этого графа. Если граф не сильно связан, то его множество вершин можно разложить единственным образом в максимально сильно связанные подмножества, которые называются сильными компонентами;
3. Обработка сильных компонент в топологическом порядке.

К методам, ориентированным на специфику КНФ, относится метод, предложенный в работе [5] и предлагаемый нами метод.

Метод 4. Метод поиска кратчайшей установочной последовательности посредством сведения к выполнимости КНФ [5] основан на последовательном увеличении числа однотипных блоков комбинационной схемы, до тех пор, пока на некотором шаге не будет получена выполнимая КНФ разрешения, по которой определяется кратчайшая установочная последовательность, или не будет исчерпан лимит времени. Идея метода упрощения КНФ разрешения в данной работе заключается в следующем:

1. В первом блоке замена определенных значений переменных $z_1^1, z_2^1, \dots, z_k^1$ на неопределенные;
2. Проверка на выполнимость КНФ разрешения отдельных элементов комбинационной схемы. Если данная КНФ разрешения невыполнима, тогда исключаются все вхождения этого элемента в дизъюнктах КНФ разрешения комбинационной схемы;
3. Применение элементарных правил упрощения.

Метод 5. В работе предлагается метод, ориентированный на использование специфики КНФ, строящейся для нахождения последовательности, переводящей автомат из начального внутреннего состояния в одно из внутренних состояний множества. Множество представлено интервалом (конъюнкцией K) в пространстве внутренних переменных. Комбинационная составляющая строится из двухвыходных и одновыходных вентилей. Большинство из порождаемых ими дизъюнктов содержит по две литеры.

Операции метода:

1. Сначала фиксируются константами внутренние переменные первого блока в соответствие с заданным начальным состоянием синхронного автомата.
2. Далее в дизъюнктах последнего блока вычеркиваем переменные, инверсные присутствующим в конъюнкции K .

Операции 1, 2 приводят к образованию однолитерных дизъюнктов. Возникает возможность применения следующих правил упрощения:

- пусть A – одно литерный дизъюнкт, тогда $A \wedge (A \vee X) = A$;
- $A \wedge (\bar{A} \vee X) = A \wedge X$;
- $A \wedge A = A$.

Проиллюстрируем использование предложенного нами метода упрощения КНФ с последующим построением установочной последовательности на схеме рис. 1. Пусть начальное состояние представляется булевым вектором 111 в пространстве внутренних переменных. Конъюнкция $K = z_1 z_2 z_3$ в том же пространстве переменных.

КНФ разрешения для первого блока после применения правил упрощения имеет следующий вид:

$$C^1 = x_1^1 \wedge x_2^1 \wedge e_1^1 \wedge e_2^1 \wedge \bar{e}_3^1 \wedge y_1^1 \wedge y_2^1 \wedge 0.$$

Полученная КНФ разрешения C^1 невыполнима, т.е. установочной последовательности длины 1 для схемы рис. 1 не существует. Переходим к построению КНФ разрешения C^2 для следующего блока, пусть этот блок является последним.

Заметим, что в схеме рис. 1, являющейся комбинационной составляющей схемы с памятью, выходы y_1^1, y_2^1, y_3^1 отождествляются с входами z_1^2, z_2^2, z_3^2 , т.е. $z_1^2 = y_1^1$, $z_2^2 = y_2^1$, $z_3^2 = y_3^1$. КНФ разрешения C^2 принимает следующий вид:

$$C^2 = x_1^1 \wedge x_2^1 \wedge e_1^1 \wedge e_2^1 \wedge \bar{e}_3^1 \wedge z_1^2 \wedge z_2^2 \wedge (x_1^2 \vee \bar{e}_1^2) \wedge (x_2^2 \vee \bar{e}_1^2) \wedge (\bar{x}_1^2 \vee \bar{x}_2^2 \vee e_1^2) \wedge (x_3^2 \vee e_2^2) \wedge (\bar{z}_1^2 \vee e_2^2) \wedge (\bar{x}_3^2 \vee z_1^2 \vee \bar{e}_2^2) \wedge (\bar{z}_2^2 \vee \bar{e}_3^2) \wedge (\bar{z}_3^2 \vee \bar{e}_3^2) \wedge (z_2^2 \vee z_3^2 \vee e_3^2) \wedge (e_1^2 \vee \bar{y}_1^2) \wedge (e_2^2 \vee \bar{y}_1^2) \wedge (\bar{e}_1^2 \vee \bar{e}_2^2 \vee y_1^2) \wedge (\bar{e}_2^2 \vee y_2^2) \wedge (\bar{e}_3^2 \vee y_2^2) \wedge (e_2^2 \vee e_3^2 \vee \bar{y}_2^2) \wedge \bar{z}_3^2.$$

КНФ разрешения C^2 после упрощения:

$$C^2 = x_1^1 \wedge x_2^1 \wedge e_1^1 \wedge e_2^1 \wedge \bar{e}_3^1 \wedge z_1^2 \wedge z_2^2 \wedge \bar{z}_3^2 \wedge x_1^2 \wedge x_2^2 \wedge e_1^2 \wedge e_2^2 \wedge \bar{e}_3^2 \wedge y_1^2 \wedge y_2^2.$$

КНФ разрешения C^2 оказывается выполнимой, один из выполняющих наборов значений её переменных имеет следующий вид:

$$\begin{array}{cccccccccccccccc} z_1^1 & z_2^1 & z_3^1 & x_1^1 & x_2^1 & x_3^1 & e_1^1 & e_2^1 & e_3^1 & z_1^2 & z_2^2 & z_3^2 & x_1^2 & x_2^2 & x_3^2 & e_1^2 & e_2^2 & e_3^2 & y_1^2 & y_2^2 & y_3^2 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \end{array}$$

Он найден с помощью SAT решателя Riss [6] и представляет установочную последовательность длины 2: $X^1(110)$, $X^2(110)$, $Y^2(110)$.

Заключение

В работе рассмотрены известные методы упрощения КНФ и предложен оригинальный метод упрощения КНФ, ориентированный на специфику составляющих её дизъюнктов и ассоциированный с поиском установочной последовательности.

ЛИТЕРАТУРА

1. *Matrosova A., Andreeva V., Melnikov A.* ROBDDs Application for Finding the Shortest Transfer Sequence of Sequential Circuit or Only Revealing Existence of this Sequence without Deriving the Sequence itself // Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2016). – Kharkov: IEEE Computer Society. – 2016. – P. 513–516.
2. *Хныкин И.Г.* Эквивалентное преобразование КНФ, ассоциированных с задачами криптографического анализа, с помощью правил резолюции // Прикладная дискретная математика. – 2009. – № 1. – С. 91–93.
3. *Boros E.* [et al] A subclass of Horn CNFs optimally compressible in polynomial time // Annals of Mathematics and Artificial Intelligence. – 2009. – V. 57. – P. 249–291.
4. *Boros E., Sepek O., Kogan A.* Horn minimization by iterative decomposition // Annals of Mathematics and Artificial Intelligence. – 1998. – V. 23. – P. 321–343.
5. *Черемисинова Л.Д.* Поиск кратчайшей установочной последовательности схемы с памятью на D-триггерах // Весті Нацыянальнай акадэміі навук Беларусі. Серыя фізіка-матэматычных навук. – 2015. – С. 119–128.
6. Proceedings of SAT Competition 2017: Solver and Benchmark Descriptions, volume B-2017-1 of Department of Computer Science Series of Publications B, University of Helsinki 2017.

СТАТИЧЕСКОЕ ТЕСТИРОВАНИЕ ДИНАМИЧЕСКИХ СТРУКТУР ДАННЫХ

Е.А. Петренко, В.В. Андреева

Томский государственный университет
katrinka9655@yandex.ru, avv.21@mail.ru

Введение

При разработке программного обеспечения (далее – ПО) широко используются динамические структуры данных для решения задач, связанных с обработкой данных.

Динамические структуры данных – это такие структуры данных, память под которые выделяется и освобождается по мере необходимости. По определению, динамические структуры характеризуются отсутствием физической смежности элементов структуры в памяти, непостоянством и стохастичностью размера (числа элементов) структуры в процессе её обработки [1].

Поскольку элементы динамической структуры располагаются по непредсказуемым адресам памяти, адрес элемента такой структуры не может быть вычислен из адреса начального или предыдущего элемента. Для установления связи между элементами динамической структуры используются указатели, через которые устанавливаются явные связи между элементами. Такое представление данных в памяти называется связным. Отдельные элементы структуры (узлы) связываются между собой с помощью ссылок, где ссылки – это адреса в памяти других узлов того же типа, с которыми данный элемент логически связан. Каждый элемент состоит из двух областей памяти: поля данных и ссылок. В языке Си, С++ для организации ссылок используются указатели [2].

Таким образом, динамические связанные структуры данных характеризуются тем, что: не имеют имени, память для них выделяется в процессе выполнения программы, количество элементов структуры может не фиксироваться, размерность структуры может меняться в процессе выполнения программы, кроме того, может меняться характер взаимосвязи между элементами структуры [1].

При использовании динамических связанных структур важно соблюдать следующие правила:

- 1) если для структуры было выделено место в динамической памяти, то по завершении работы со структурой место, занимаемое ею, нужно освободить;
- 2) если логическая последовательность элементов структуры изменяется, то следует не перемещать данные в памяти, а корректировать указатели.

Нарушение этих правил может привести к утечке ресурсов, а именно, памяти.

Соблюдение вышеописанных правил необходимо отслеживать во время тестирования ПО. Тестирование можно выполнять как динамически, так и статически. Динамическое тестирование осуществляется путем запуска продукта и проверки его функционала. Проверка совершается с помощью ручного или автоматического выполнения заранее подготовленного набора тестов. Для динамического анализа кода запускаются готовые программы.

Большой интерес представляет статическое тестирование, которое производится без исполнения программного кода продукта. Тестирование осуществляется путем анализа программного или скомпилированного кода. Анализ может производиться как вручную, так и с помощью специальных инструментальных средств. Целью такого анализа является своевременное выявление ошибок и потенциальных проблем в продукте.

В настоящий момент существует несколько статических анализаторов динамических структур данных. Анализируя результаты их работы, можно сделать вывод, что они не ориентированы на проверку корректности операций над связями в динамических структурах. Данные анализаторы тестируют соблюдение лишь первого правила

использования связанных динамических структур данных. Следовательно, такие анализаторы не обнаруживают все возможные причины утечки памяти.

В данной работе предлагается алгоритм, позволяющий анализировать операции над связями между элементами динамических структур данных и, как следствие, выявлять причины утечки памяти.

1. Постановка задачи

Объектом исследования в представленной работе являются связанные динамические структуры, предметом исследования – статическое тестирование потери связей в динамических структурах. Рассматриваются такие динамические структуры, как линейный массив, однонаправленный список, двунаправленный список, двоичное дерево. Данные структуры характеризуются тем, что имеют поле данных и поля-указатели, кроме динамического массива. Поле данных не является динамической структурой, которая так же требует тестирования.

Для обнаружения операций, нарушающих связи между элементами динамической структуры, необходимо разработать алгоритм, ориентированный на тестирование программного кода, написанного на языке Си, С++, с целью обнаружения утечки памяти. Под утечкой памяти понимается обрыв связей между элементами структуры и, как следствие, невозможность доступа к ранее выделенным областям памяти.

2. Анализ операций над связями в динамических структурах

Любая динамическая структура характеризуется тем, что для неё выделяется именованный участок памяти, который хранит адрес на первый элемент структуры, в дальнейшем будем называть такой участок памяти *головой*. В языке C/C++ для хранения адреса участка памяти используются указатели соответствующего типа. В нашем случае переменную для сохранения начального адреса динамической структуры будем обозначать переменной *head*. Заметим, что при работе с динамическими структурами и выполнении различных операций над узлами, необходимо следить за переменной *head*, не допуская потерю начального адреса, так как в противном случае, потеря адреса приведет к потере доступа ко всей динамической структуре.

Анализируя рассматриваемые динамические структуры, можно выделить основные операции над связями этих структур, нарушение которых приводит к утечке ресурсов:

1) Перенаправление головы. Данная операция сопровождается изменением начального адреса, хранящегося в *head*, и приводит к потере доступа ко всей структуре (рис. 1). Рассмотрим на примере динамического массива. Как показано на рис. 1, перенаправление головы реализуется путем последовательного присвоения сначала одного адреса участка памяти, затем другого. Такое присвоение приводит к потере доступа к первоначальной структуре.



Рис. 1. Операция перенаправления head: а) начальная инициализация head, б) перенаправление head

Данная операция является общей для всех рассматриваемых динамических структур.

2) Добавление узла в динамическую структуру. Операция справедлива для структур, строящихся из узлов, таких как списки и деревья. В случае динамического массива данная операция вырождается в операцию перенаправление головы. В общем случае при реализации данной операции необходимо правильно установить связи между элементами, не нарушив целостность структуры, а в случае добавления на место

первого узла необходимо обновлять значение head. В качестве примера рассмотрим добавление узла после указанного элемента в однонаправленном списке (рис. 2). В приведённом ниже примере показана корректная последовательность действий при добавлении элементов, т.е. сначала необходимо установить связь a , затем b , при установке которой происходит разрушение связи l . На рис. 2 б) продемонстрирована потеря последнего узла при неправильном проведении связей, в данном случае – установка сначала связи b приводит к потере последнего элемента.

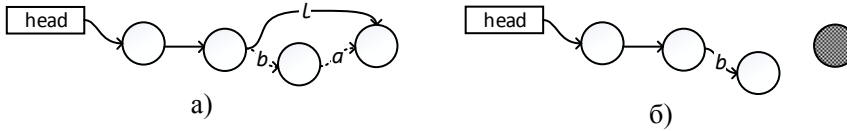


Рис. 2. Операция добавления в однонаправленный список: а) последовательность связей, для корректного добавления элемента, б) неверная последовательность установки связей между узлами

3) Удаление узла из динамической структуры. Данная операция справедлива для структур, строящихся из узлов, таких как: списки и деревья. В случае динамического массива данная операция вырождается в операцию удаления массива. В общем случае операция удаления является обратной операцией добавления, при которой также необходимо правильно и в нужном порядке установить связи. Данную операцию можно продемонстрировать на примере удаления элемента в двунаправленном списке. Двунаправленный список характеризуется тем, что каждый его узел имеет информационное поле и два поля-указателя, в которых хранятся адрес следующего такого же узла и предыдущего.



Рис. 3. Удаление узла u

Для выполнения корректной операции удаления узла, необходимо выполнить ряд действий, как показано на рис. 3 а), в частности: запомнить удаляемый элемент в tmp, провести связи a , b . На рис. 3 б) выполняется сразу удаление узла u , в таком случае происходит потеря связи с предыдущим элементом и последующим, в следствии чего установка связей a , b становится невозможной. В таком случае происходит нарушение целостности структуры.

4) Разделение динамической структуры на отдельные части. Данная операция справедлива для структур, строящихся из узлов, такие как списки и деревья. В общем случае эта операция сопровождается формированием для каждой i -ой части выделяемой структуры своей head _{i} , причем с сохранением целостности в каждой части динамической структуры. В качестве примера рассмотрим двоичное дерево. Оно характеризуется тем, что каждый узел имеет информационную часть и два поля-указателя, хранящих адрес левого и правого потомка – таких же узлов.

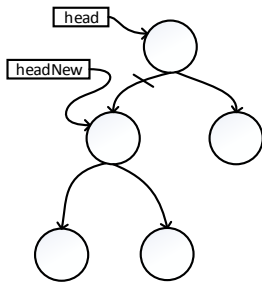


Рис. 4. Разделение дерева head на два дерева: head и headNew

б) Установка в NULL поля-указателя некоторого элемента. Данная операция справедлива для структур, строящихся из узлов, таких как списки и деревья. В общем случае эта операция приводит к нарушению целостности структуры, т.к. установка значения поля, хранящего адрес следующего элемента, приводит к потере части структуры, следующей за этим элементом.

Здесь мы рассмотрели общие операции, при выполнении которых либо при неправильном их выполнении происходит нарушение целостности в динамических структурах.

3. Методы статического тестирования

Статическое тестирование – это процесс, который обычно ассоциируют с анализом ПО. Статическим тестированием пользуются для верификации практически любого артефакта разработки: программного кода компонент, требований, системных спецификаций, функциональных спецификаций и т.д. Использование статических методов тестирования – один из наиболее эффективных способов обнаружения дефектов на ранних стадиях разработки ПО. Кроме того, статическое тестирование – это единственный способ тестирования без запуска программного кода приложения.

Ошибки, обнаруживаемые статическими анализаторами, весьма разнообразны. Например, возвращение из функции по ссылке адреса локальной переменной, выход за границу массива, неинициализированные или неиспользуемые переменные, переполнение буфера, опечатки, разыменовывание параметров без предварительной проверки, т.д. Данная работа ориентирована на статическое тестирование динамических структур, а именно – на обнаружение нарушения связей в динамических структурах без запуска исходного кода.

Авторы [3] осуществляют тестирование путем генерации тестовых данных, основываясь на системе, состоящей из трех частей: программы-анализатора, селектора пути и генератора данных для тестирования. Программа-анализатор принимает исходный код в качестве входных данных и обеспечивает поток или информацию о потоке данных, такую как график зависимости данных, график контрольного потока и т.д. На основе этих данных селектор выбирает пути, которые будут выполняться сгенерированными данными тестирования. Цель работы генератора данных для тестирования состоит в том, чтобы найти входные значения, которые выполняют каждый оператор пути. Авторы данной статьи сконцентрировались, прежде всего, на аспекте потраекторной генерации тестовых данных, провоцирующих необработанные исключительные ситуации. Однако, в работе не разработан метод, анализирующий результаты работы тестируемых функций на сгенерированных тестовых данных.

5) Извлечение элемента из списка. Эту операцию рассмотрим на примере однонаправленного списка. Данная операция справедлива для структур, строящихся из узлов, такие как списки и деревья. В общем случае эта операция подобна операции удаления элемента из динамической структуры, за исключением того, что сам элемент не подвергается удалению. При реализации этой операции также важно правильно проводить связи между элементами. Как показано на рис. 5, при извлечении элемента необходимо запомнить адрес самого элемента, а только потом провести связь l , позволяющую сохранить целостность структуры.

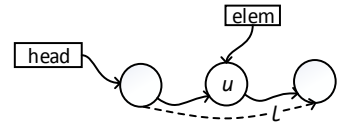


Рис. 5. Операция извлечения элемента

В [4] авторы также сосредоточились на генерации тестовых данных, ориентированных на путь динамического указателя. Эти данные уже отображают некоторые свойства узлов, что делает их более информативными при анализе. Тем не менее, анализ не производится, следовательно, результат работы предложенного метода не является наглядным.

Оба метода предполагают работу со структурами, информационное поле которых имеет либо целочисленный, либо вещественный тип данных. Алгоритмы, предложенные в [3,4] совмещают статическое и динамическое тестирование.

4. Обнаружение нарушение связей в динамических структурах статическими анализаторами

В настоящее время существует ряд статических анализаторов [5–7]. Наиболее популярными среди анализаторов, ориентированных также и на обнаружение утечки ресурсов данных, являются PVS-Studio, Cppcheck и Clang Static Analyzer.

PVS-Studio – это инструмент для выявления ошибок в исходном коде программ, написанных на языках C, C++ и C#. Работает в среде Windows и Linux. PVS-Studio выполняет статический анализ кода и генерирует отчёт, помогающий программисту находить и устранять ошибки [5]. PVS-Studio выполняет широкий спектр проверок кода, но наиболее силён в поисках опечаток и последствий неудачного Copy-Paste. PVS-Studio хорошо интегрируется в среду разработки Visual Studio 2010–2017.

Cppcheck – статический анализатор кода для языка C/C++, предназначенный для поиска ошибок, которые не обнаруживаются компиляторами. Главной целью этого проекта является сведение до минимума количества ложных срабатываний при поиске ошибок [6]. Cppcheck является бесплатным кроссплатформенным статическим анализатором с открытым исходным кодом. Он интегрируется со многими средами разработки. На данный момент Cppcheck является развивающимся проектом. Он хорош тем, что довольно быстро работает.

Clang Static Analyzer – средство статического анализа кода, которое находит ошибки в программах, написанных на C, C++ и Objective-C. Также является бесплатным кроссплатформенным приложением с открытым исходным кодом. Вызывается из командной строки и запускается во время сборки тестируемого проекта [7]. Clang Static Analyzer, в отличие от Cppcheck, работает существенно медленнее, но ошибки находит более серьезные.

Каждый из перечисленных анализаторов обнаруживает различные типы ошибок в программах: выход за границы массива, утечки памяти, связанные с захваченными, но не освобождёнными участками памяти, возможное разыменовывание NULL-указателя, неинициализированные переменные, проверяет обработку исключительных ситуаций на безопасность, находит устаревшие и неиспользуемые функции, предупреждает о неиспользуемом или бесполезном коде, находит подозрительные участки кода, которые могут содержать в себе ошибки.

С помощью данных анализаторов были протестированы программы, в которых были намерено введены в программный код ошибки, вызывающие обрыв связей в структурах и, как следствие, утечку памяти. Результаты тестирования представлены в табл. 1.

Таблица 1

Условия возникновения потери памяти	Запрограммированное число ошибок	Обнаруженное число ошибок с помощью PVS-Studio	Обнаруженное число ошибок с помощью Cppcheck	Обнаруженное число ошибок с помощью Clang Static Analyzer
Перенаправление head	3	0	0	0
Добавление в head однонаправленного списка	1	0	0	0

Условия возникновения потери памяти	Запрограммированное число ошибок	Обнаруженное число ошибок с помощью PVS-Studio	Обнаруженное число ошибок с помощью Cppcheck	Обнаруженное число ошибок с помощью Clang Static Analyzer
Добавление в head двунаправленного списка	1	0	0	0
Добавление в head дерева	1	0	0	0
Добавление элемента в однонаправленный список	1	0	0	0
Добавление элемента в двунаправленный список	1	0	0	0
Добавление элемента в дерево	2	0	0	0
Удаление head в однонаправленном списке	1	0	0	0
Удаление head в двунаправленном списке	1	0	0	0
Удаление элемента в однонаправленном списке	1	0	0	0
Удаление элемента двунаправленном списке	1	0	0	0
Разделение однонаправленного и двунаправленного списков и дерева на части	4	0	0	0
Извлечение элемента из однонаправленного и двунаправленного списков	2	0	0	0
Установка поля в значение NULL	2	0	0	0

Таким образом, из таблицы видно, что рассмотренные популярные статические анализаторы не отлавливают ситуации, связанные с операциями над связями элементов динамических структур, приводящие к утечке ресурсов.

5. Архитектура анализатора операций над связями в динамических структурах

Общая архитектура анализатора операций над связями в динамических структурах состоит из следующих блоков:

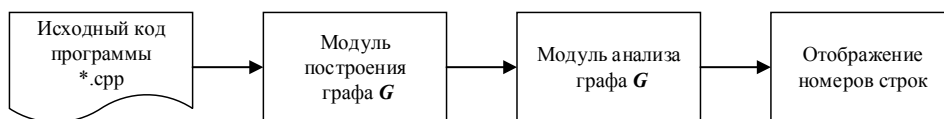


Рис. 6. Общая архитектура

На вход поступает исходный, скомпилированный без ошибок файл с программным кодом, написанном на языках Си, C++.

Исходный код преобразуется во вспомогательную структуру следующим образом: при каждом объявлении переменной-указателя создается корневая вершина для вспомогательного взвешенного графа G , где вершине графа соответствует узел тестируемой динамической структуры, а ребру – связи между элементами структуры. Вес ребра может быть равен нулю или единице, в зависимости от того, что хранится в соответствующем поле тестируемой структуры. Также вес ребра может измениться на ноль или единицу в ходе построения структуры: на ноль, если полю присваивается NULL, если добавляется новое исходящее ребро, на единицу, если вес ребра был ноль, и в соответствующее поле записывается некоторый адрес (не NULL).

При создании первой вершины графа в него добавляется еще две вершины с единичными исходящими (и входящими) ребрами, это делается с целью моделирования тестируемой структуры. Количество добавленных узлов объясняется тем, что оно покрывает все операции, приводящие к нарушению связей.

Корневые вершины графов, соответствующие локальным переменным, и их ребра удаляются при выходе из функции.

После завершения построения вспомогательного графа и моделирования операций над связями осуществляется его анализ: здесь необходимо исследовать каждую вершину графа, начиная с первой, и все её ребра:

На выходе анализатора получен набор строк, в которых возможен обрыв связей между элементами динамической структуры.

5.1. Построение графа G

Как было указано выше, вспомогательной структурой является взвешенный, ориентированный, связанный граф, позволяющий смоделировать тестируемую динамическую структуру. Каждый узел графа имеет вид рис. 7. Здесь под $line_i$ понимается обозначение узла, для которого справедливо три варианта:

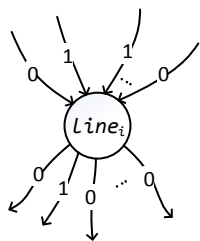


Рис. 7. Вид узла графа G

- а) new_i – строка кода под номером i с оператором new ;
- б) del_i – строка кода под номером i с оператором $delete$;
- в) i – строка кода под номером i , в которой происходит, обрыв связей.

Кроме того, каждый узел графа характеризуется тем, что может иметь входящие и исходящие дуги. Они могут быть помечены значениями 0 или 1, где 0 устанавливается при условии того, что в информационной части узла не записана конструкция new и номер, означает, переход по этой дуге невозможен, а 1 устанавливается в случае корректности связей между двумя элементами, соединяемыми данной дугой.

Напомним, что в рамках поставленной задачи рассматривается код с некоторыми ограничениями, а именно – код программы на языке C/C++ без синтаксических ошибок. Объектами тестирования являются только связанные динамические структуры. Предполагается, что каждый оператор кода находится на своей строке, и каждая строка пронумерована.

Для построения графа G анализируется файл *.cpp, удовлетворяющий описанным выше ограничениям.

Рассмотрим основные правила формирования узла графа G .

Когда в строке встречается оператор new , происходит формирование узла по следующим правилам:

1) Если тестируется **массив**, то узел помечается new_i , далее формируется входящая дуга, помеченная единичным весом. Вес дуги со значением 1 означает, что адрес создаваемого узла массива не утерян.

2) Если тестируется **однонаправленный список**, то узел помечается new_i , формируется входящая дуга, помеченная единичным весом, и исходящая дуга, помеченная значением 0. Вес дуги со значением 1 означает, что адрес создаваемого узла однонаправленного списка не утерян. Исходящее ребро со значением 0 означает, что узел один и переход по этому ребру невозможен.

3) Если тестируется **двунаправленный список**, то узел помечается new_i , формируется входящая дуга, помеченная единичным весом, и две исходящие дуги, помеченные значением 0. Вес дуги со значением 1 означает, что адрес создаваемого узла двунаправленного списка не утерян. Исходящие ребра с нулевыми значениями означают, что переход по эти ребрам невозможен, так как узел всего один.

4) Если тестируется **двоичное дерево**, то узел помечается new_i , формируется входящая дуга, помеченная единичным весом и две исходящие дуги, помеченные значением 0. Вес дуги со значением 1 означает, что адрес создаваемого узла дерева не утерян. Исходящие ребра с нулевыми значениями означают, что переход по этим ребрам невозможен, так как узел всего один.

Добавление узла в граф осуществляется по следующим правилам:

5) Если граф пустой, то в него добавляется сформированный ранее узел и для моделирования добавляется два таких же узла с изменением веса нулевых рёбер, по которым теперь возможен переход на соседний узел, на 1.

6) Если граф не пустой, то анализируются строки, в которых реализуются операции по перенаправлению связей. В этом случае после анализа операций над связями в текущем узле тестируемой структуры в соответствующем узле графа необходимо выполнить изменения. Эти изменения касаются содержимого узла, веса ребер, а также, возможно, формирования входящих/исходящих ребер.

Изменение информационной части узла графа и веса рёбер и добавление новых рёбер подчиняется следующим правилам:

7) Если в анализируемой строке вызывается оператор *delete*, то в информационную часть узла графа перезаписывается конструкция: *del_i*.

8) Если добавляется новое исходящее единичное ребро, то вес всех остальных исходящих рёбер обнуляется, т.к. добавление исходящего единичного ребра говорит о перенаправлении связей в тестируемой структуре.

9) Входящее единичное ребро добавляется, если в некоторую переменную-указатель записывается адрес элемента тестируемой структуры, соответствующего текущей вершине графа.

10) Исходящее единичное ребро добавляется в случае инициализации поля-указателя элемента тестируемой структуры новым значением (не NULL).

11) Если к вершине добавляется новое исходящее единичное ребро, то в её информационную часть записывается номер текущей анализируемой строки кода.

Так как граф G также является динамической структурой, для него необходимо создавать переменную для сохранения начального адреса первого узла графа. Для удобства будем называть эту переменную тем же именем, что и у переменной-*head* тестируемой структуры.

В табл. 2 приведены примеры графов G_i , отражающих процесс построения динамической структуры.

Таблица 2

Динамическая структура	Соответствующий взвешенный граф
Массив	
Однонаправленный список	
Двунаправленный список	
Двоичное дерево	

Пример 1. Рассмотрим код программы, в котором реализован случай некорректного добавления элемента в двунаправленном списке.

```

1 struct ListNode2
2 {
3     int _data;
4     ListNode2* _nx;
5     ListNode2* _pr;
6 };
7 void createList2(ListNode2** head,
8 int size)
9 {
10     ListNode2* newList = NULL;
11     ListNode2* ptr;
12     ptr = newList;
13     for (int i = 0; i < size; i++)
14     {
15         ListNode2* q;
16         q = new ListNode2;
17         q->_data = i+1;
18         q->_nx = NULL;
19         q->_pr = NULL;
20         if (newList != NULL) {
21             q->_pr = ptr;
22             ptr->_nx = q;
23             ptr = ptr->_nx;
24         }
25         else {
26             newList = q;
27             ptr = newList;
28         }
29     }
30     *head = newList;
31 }
32 void addBefore2(ListNode2** head,
33 int beforeKey, int key)
34 {
35     if (*head == NULL)
36         return;
37     ListNode2* ptr1 = NULL;
38     ptr1 = (*head);
39     if (ptr1->_data != beforeKey) {
40         ListNode2* q1;
41         q1 = new ListNode2;
42         q1->_data = key;
43         q1->_nx = NULL;
44         q1->_pr = NULL;
45         while (((ptr1->_nx) != NULL) &&
46             ((ptr1->_nx->_data) != beforeKey))
47             ptr1 = ptr1->_nx;
48         if ((ptr1->_nx->_data) == beforeKey) {
49             q1->_pr = ptr1;
50             ptr1->_nx = q1;
51         }
52         else {
53             q1->_data = 0;
54             ptr1->_nx = q1;
55             q1->_pr = ptr;
56         }
57     }
58     else {
59         addToHead2(head, key);
60     }
61 }
62 void main()
63 {
64     int size = 10;
65     ListNode2* head2 = NULL;
66     createList2(&head2, size);
67     addBefore2(&head2, 5, 888);
68     return;
69 }

```

Соответствующий граф G продемонстрирован на рис. 8.

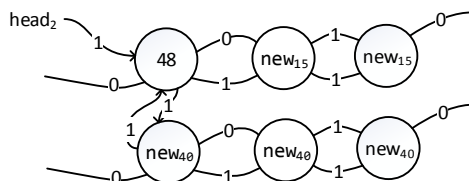


Рис 8. Граф G , соответствующий операции добавления элемента в двунаправленный список

В строке 66 объявляется переменная-указатель $head2$, для нее создается указатель на граф G с тем же именем (рис. 9).

head2

Рис. 9. Создание указателя на вспомогательный граф

Далее в строке 67 осуществляется вызов функции createList2, куда в качестве входного параметра head передаётся объявленная ранее переменная head2. В функции CreateList2 в строках 9, 10 и 14 объявляются переменные-указатели: newList, ptr и q соответственно. Следовательно, для них также создаются головы одноимённых графов (рис. 10).

head2
new List
ptr
q

Рис. 10. Создание указателей на вспомогательные графы

В строке 15 с помощью оператора new происходит выделение области памяти, адрес которой хранится в переменной q. Тогда, согласно правилу 3), формируется узел вида (рис. 11):

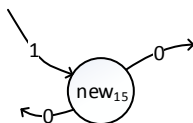


Рис. 11. Формирование узла графа для тестирования связей в двунаправленном списке

По правилу 5) в граф для переменной-указателя q добавляется сформированная вершина, тогда полученный граф будет иметь вид (рис. 12.):

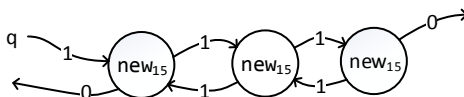


Рис. 12. Формирование графа для тестирования связей в двунаправленном списке

Далее, в строках 25 и 26 в переменную newList и ptr соответственно записывается адрес построенной тестируемой структуры, тогда в сформированном графе в первую вершину по правилу 9) добавляется два входящих единичных ребра (рис. 13):

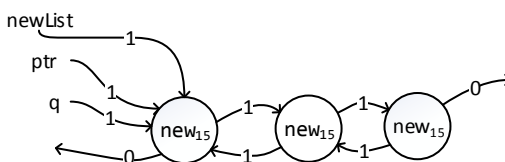


Рис. 13. Инициализация созданных ранее переменных-указателей

Цикл по построению двунаправленного списка, выполнение которого начинается со строки 12 и оканчивается строкой 28, достаточно выполнить один раз, так подход основан на моделировании тестируемой динамической структуры и точный её размер не имеет значения.

В строке 29 адрес, хранящийся в переменной newList, записывается в переменную head, следовательно, т.к. переменной head соответствует граф head2, необходимо выполнить правило 9) относительно первой вершины графа (рис. 14):

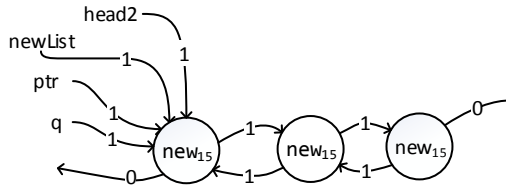


Рис. 14. Инициализация созданных ранее переменных-указателей

Далее осуществляется выход из функции createList2, поэтому головы графов, соответствующих локальным переменным, а именно – newList, ptr и q, удаляются. Также удаляются и соответствующие входящие в первую вершину графа единичные рёбра (рис. 15):

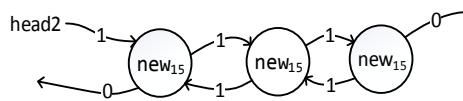


Рис. 15. Граф G, полученный в результате построения двунаправленного списка.

В строке 68 осуществляется вызов функции addBefore2, которая добавляет новый элемент в двунаправленном списке перед указанным. Переменная head2 передаётся в качестве входного параметра head. Далее в строке 36 объявляется переменная-указатель ptr1. Для неё создаётся голова одноимённого графа. Т.к. в строке 35 в переменную ptr1 записывается адрес, который хранится в переменной head, которая соответствует переменной head2, то голова графа ptr1 устанавливается на первую вершину в графе head2 (рис. 16):

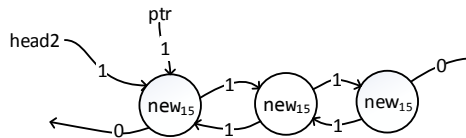


Рис. 16. Инициализация новой переменной-указателя относительно построенной структуры

Условие, описанное в строке 38, считается истинным, так как значение констант в данном случае не играет важной роли. Далее в строках 39 и 40 объявляется и инициализируется соответственно переменная q1. Для неё создаётся одноимённый указатель на граф, который строится при выполнении правил 3) и 5) (рис. 17):

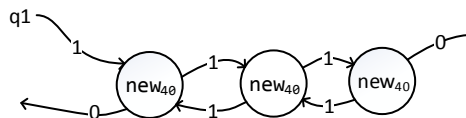


Рис. 17. Формирование нового графа

Цикл, описанный в строках 44–45, не представляет интереса для тестирования, т.к. в нём не осуществляются операции над связями в динамической структуре. Условие в

строке 44 снова считается истинным. Далее в строке 48 в поле `_pr` переменной `q1` записывается адрес, хранящийся в переменной `ptr1`. Голова графа `ptr1` указывает на первую вершину графа `head2`. Тогда к графу `q1` применяются правила 10) и 8), и новое исходящее единичное ребро направляется на тот узел, адрес которого хранится в переменной `ptr1` (рис. 18).

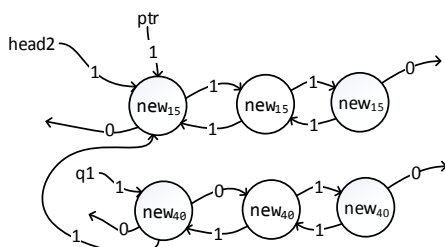


Рис. 18. Добавление единичного исходящего ребра

Строке 48 соответствует выполнение правил 10), 8) и 11) (рис. 19):

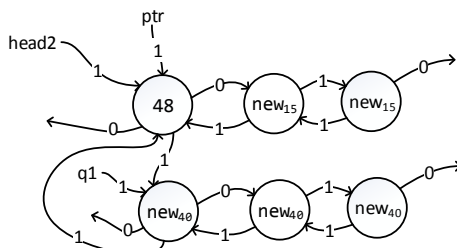


Рис. 19. Отражение перенаправления связей в тестируемой структуре

Далее выполняется выход из функции `addBefore2`, и головы графов, соответствующие локальным переменным указателям, а именно – `ptr1` и `q1`, удаляются вместе с соответствующими входящими единичными рёбрами. Соответствующий результирующий граф показан на рис. 8.

5.2. Анализ вспомогательной структуры

Анализ вспомогательной структуры осуществляется с первой вершины построенного графа и до последней. Выполняется проверка наличия следующих условий:

Если у вершины есть нулевые входящие или исходящие дуги, то: если в её информационное поле записано только число или узел помеченный del_i , то его (число) необходимо добавить в набор строк с потенциально «опасным» кодом.

Анализируя полученный в примере 1 граф, можно сделать вывод, что при добавлении нового элемента перед указанным происходит обрыв связей, на это указывает нулевое ребро, исходящее из первой вершины графа. Кроме того, информация, записанная в первой вершине, говорит о том, что ошибка происходит именно в 48 строке кода.

Заключение

В настоящей работе рассмотрены методы статического анализа динамических структур данных с помощью генерации тестовых данных. Также показано, что современные статические анализаторы кода не ориентированы на обнаружение операций, провоцирующих обрыв связей между элементами динамических структур данных. Представлен алгоритм, позволяющий обнаруживать утечку памяти, возникающую из-за обрыва связей в динамической структуре.

ЛИТЕРАТУРА

1. НОУ ИНТУИТ Лекция Динамические структуры данных URL: <https://www.intuit.ru/studies/courses/648/504/lecture/11455> (31.03.2018).
2. Язык Си учебник и практический курс для начинающих URL: <http://kpolyakov.spb.ru/download/devcpp6.pdf> (31.03.2018).
3. Sittisak Sai-ngern, Chidchanok Lursinsap, Peraphon Sophatsathit An address mapping approach for test data generation of dynamic linked structures // Information and Software Technology. – 2005. – № 47. – С. 199–214.
4. Zhao R., Li Q. Automatic test generation for dynamic data structures. // In: Proceedings SERA '07, IEEE Computer Society, Washington, DC. – 2007. – P. 545–549.
5. PVS-Studio статический анализатор кода для C, C++ и C# URL: <https://www.viva64.com/ru/pvs-studio/> (07.04.2018).
6. Cppcheck – Википедия URL: <https://ru.wikipedia.org/wiki/Cppcheck> (07.04.2018).
7. Clang Static Analyzer URL: <https://clang-analyzer.lvm.org/> (07.04.2018).

АДАПТИВНЫЙ НЕЙРО-НЕЧЕТКИЙ КЛАССИФИКАТОР ИНОВАЦИОННО-ТЕХНОЛОГИЧЕСКИХ РЕШЕНИЙ*

О.И. Мизюркина, М.Н. Головчинер

Томский государственный университет
olga.mizyurkina@mail.ru, golovchiner@mail.ru

Введение

В современной нейроинформатике актуальной задачей является разработка адаптивного нейро-нечеткого классификатора для анализа эффективности технологий, научно-технических решений, проектов, характеризующихся разнородными (количественными и качественными) данными. В большинстве случаев исходные данные представлены не в числовом виде, а в ранговой шкале, например, в виде последовательности упорядоченных лингвистических переменных (например: «низкий», «ниже среднего», «средний», «выше среднего», «высокий»). Один из методов решения данной задачи – это использование нечеткого дерева решений, построенного на основе обучающей выборки данных, с предварительной формализацией исходных параметров в виде вектора интервальных значений (нечеткого интервала) [1].

Основная характеристика объекта в теории нечетких множеств – функция принадлежности. Определение функции принадлежности – важная стадия, от которой в дальнейшем зависит сходимость метода классификации и точность оценок. В общем случае различают два подхода при построении функций принадлежности: прямые методы и косвенные. В случае косвенных методов экспертное мнение является лишь исходной информацией для дальнейшей работы, что уменьшает уровень субъективизма при работе системы. В работе представлен адаптивный метод фаззификации на основе построения треугольных функций, относящийся к категории косвенных методов.

1. Формализация исходных параметров

В случае, когда все входные параметры системы описываются лингвистическими переменными, этап построения функций принадлежности и дальнейшей фаззификации не применяется. Если же часть параметров задана в числовом виде, а часть – лингвистическими переменными, необходимо преобразовать входные параметры к однородному виду. В силу невозможности уточнения значений лингвистических переменных производится фаззификация точных значений и их перевод в соответствующие им ранги. Адаптивный алгоритм фаззификации на основе построения треугольных функций, описанный в данной работе, базируется на анализе распределения частот значений определенных лингвистических переменных [2].

* Исследования выполнены по программе повышения конкурентоспособности Национального исследовательского Томского государственного университета при финансовой поддержке РФФИ, грант № 16-29-12858.

Пусть значение x_i , соответствующее j -му рангу, встречается в выборке N_j раз. Тогда можно ввести в рассмотрение относительные частоты появления j -го ранга $f_j = \frac{N_j}{N}$ и кумулятивные частоты $F_1 = \frac{f_1}{2}$, $F_j = \frac{f_j}{2} + \sum_{i=1}^{j-1} f_i$, $j = \overline{2, m}$, при выполнении условия $\sum_{i=1}^m f_i = 1$.

Набор треугольных функций принадлежности, представленных на рис. 1, с центрами c_j в кумулятивных частотах F_j , задается на основе рекуррентного соотношения:

$$c_1 = \frac{f_1}{2}, \quad c_j = c_{j-1} + \frac{f_{j-1} + f_j}{2}, \quad j = \overline{2, m}.$$

$$\mu_1(x) = 1, \quad x \in [0, c_1],$$

$$\mu_j(x) = \begin{cases} \frac{x - c_{j-1}}{c_j - c_{j-1}}, & x \in [c_{j-1}, c_j], \\ \frac{c_{j+1} - x}{c_{j+1} - c_j}, & x \in [c_j, c_{j+1}], \\ 0, & x \notin [c_{j-1}, c_{j+1}], \end{cases}$$

$$\mu_m(x) = 1, \quad x \in [c_m, 1].$$

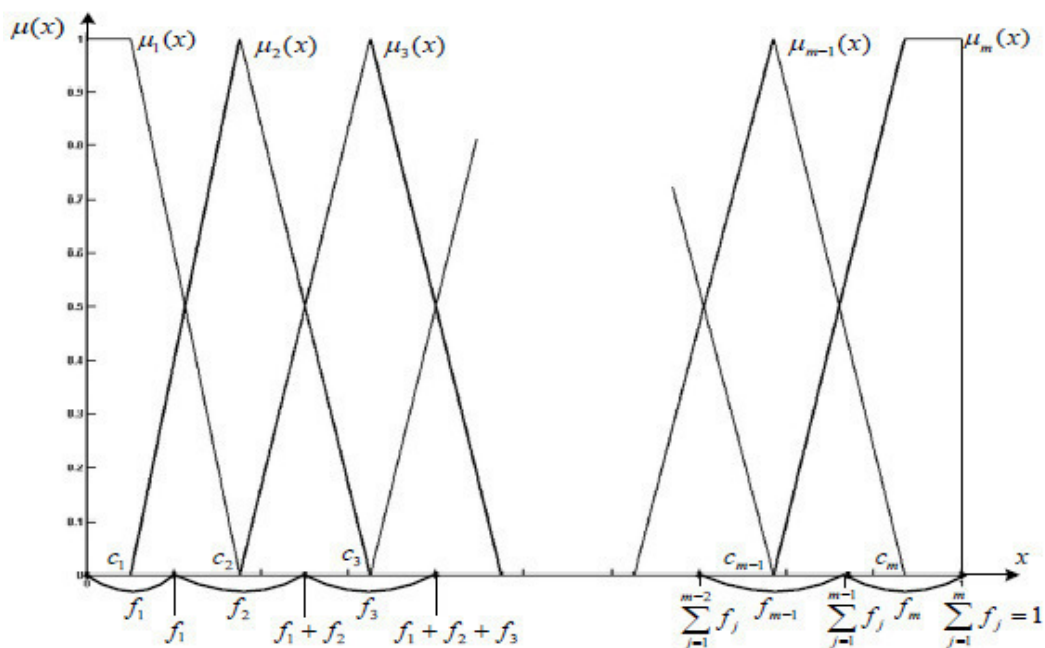


Рис. 1. Функции принадлежности для ранговых переменных

В результате исходный набор значений переменных x_1, \dots, x_N представляется множеством функций принадлежности $\mu_1(x), \dots, \mu_N(x)$, заданных на интервале $[0, 1]$, и проводится дальнейшая нечеткая классификация не исходных параметров, а соответствующих им значений лингвистических переменных.

2. Формирование нечеткого дерева

Реализация нечеткого дерева проводится по алгоритму, который был разработан Джоном Р. Квинланом – The Iterative Dichotomizer 3 (ID3). Исходные данные для работы представляются в виде таблицы, которая содержит в себе обучающее множество D мощности n . Каждый объект множества зависит от определенного набора признаков A_j мощности m , на основании значений (x_{ij}) которых объект может быть отнесен к одному из C_s классов, количество которых равно некоторому k (табл. 1).

Построение дерева решений, как и других объектов с древовидной структурой, выполняется рекурсивным алгоритмом. В корне дерева – всё обучающее множество, в листьях – множества данных, которые с некоторой долей ошибки относятся к одному и тому же результирующему классу.

Таблица 1

Общий формат представления данных

Атрибут ₁	Атрибут ₂	...	Атрибут _м	Класс
Значение ₁₁	Значение ₁₂	...	Значение _{1м}	Класс ₁
Значение ₂₁	Значение ₂₂	...	Значение _{2м}	Класс ₂
...
Значение _{н1}	Значение _{н2}	...	Значение _{нм}	Класс _к

Алгоритм на каждом шаге разбивает множество экземпляров по одному из признаков в соответствии с критерием оптимальности количества информации – информации

энтропии на текущем множестве: $Entropy(T) = -\sum_{i=0}^r \frac{|T_i|}{|T|} \log_2 \left(\frac{|T_i|}{|T|} \right)$, и на всех подмножествах, которые могут быть получены при разбиении данного узла:

$$Entropy_{A_j}(T) = \sum_{i=0}^r \frac{|T_i|}{|T|} Entropy(T_i).$$

Критерий информационного выигрыша представляет собой следующую разность: $Gain(A_j) = Entropy(T) - Entropy_{A_j}(T)$.

Каждый следующий атрибут (признак) для разбиения выбирается, если он максимизирует следующее выражение: $GainRatio(A_j) = \frac{Gain(A_j)}{SplitEntropy(A_j)}$.

Для некоторого узла разбиение прекращается и узел объявляется листом дерева в случае, если:

- относительная частота данных, значение текущей лексической переменной для которых равно c_r , больше некоторого порогового значения, т.е. $\frac{|D^{c_r}|}{D} \geq \theta_r$;
- количество объектов обучающей выборки в нечетком множестве D меньше некоторого заданного порогового значения: $|D| < \theta_r$;
- исчерпан список лексических переменных.

На рис. 2 приведена блок-схема одной итерации разбиения множества элементов.

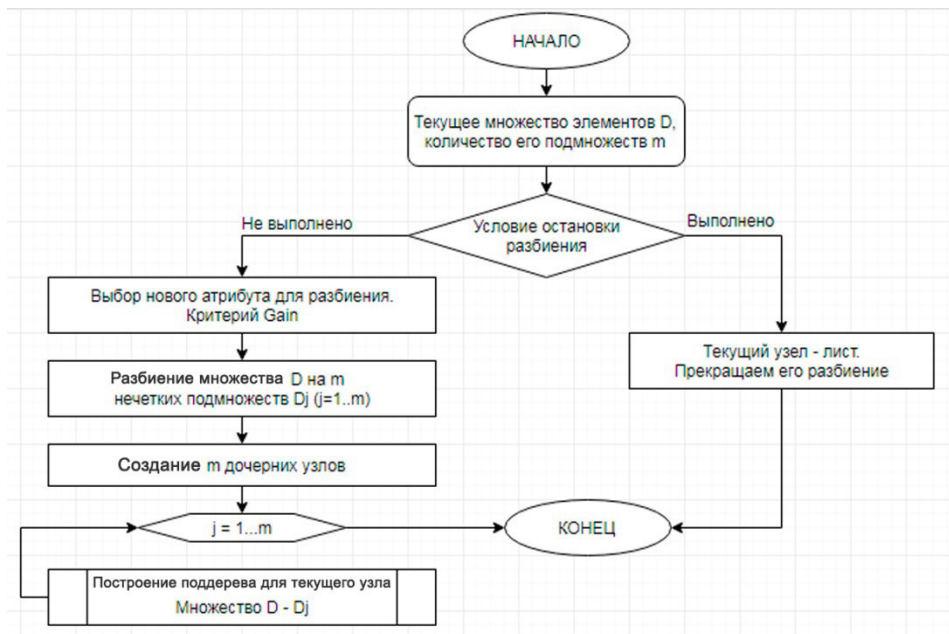


Рис. 2. Блок-схема рекурсивного алгоритма нечеткого ID3

На основе полученного в результате выполнения алгоритма дерева решений генерируется база нечетких правил следующего вида:

ЕСЛИ $x_1 = F_{1i}$ И ... И $x_m = F_{mi}$, ТО $y_1 = C_1(\beta_{i1})$,

где F_{mi} – лингвистический терм, которым оценивается переменная x_m , β_{i1} – действительные числа, которыми оцениваются выходные переменные y_1 .

Полученные правила служат в дальнейшем для построения нейро-нечеткого классификатора – интеллектуального ядра системы нечеткого вывода.

Классы разбиения, находящиеся в листах дерева, выступают в качестве правой части правила вывода, позволяющие отнести объект к конкретному классу. Для формирования же левых частей нечетких правил вывода осуществляется последовательное рекурсивное движение от листа к корню дерева. При таком движении на каждом следующем узле известна лингвистическая переменная, которая должна принять участие в формировании правила, а также значение терма, которое ей соответствует на данном пути. Переходя на алгебру логики, можно сказать, что каждый узел, который встречается на пути, представляет собой некоторую литеру, а весь путь от листа к корню – логическое выражение (конъюнкция).

Построение функций принадлежности и формирование нечеткого дерева реализовано в среде Microsoft Visual Studio 2015 на языке C# и платформе .Net. Программа позволяет пользователю работать непосредственно с файлами форматов .xls, .xlsx, в которых обычно хранятся исходные данные для классификации. Наглядное представление функций принадлежности выполнено с использованием библиотеки ZedGraph, которая позволяет визуализировать графику различного вида в приложениях с пользовательским интерфейсом Windows Forms Visual C#. Элемент управления TreeView позволяет достаточно наглядно демонстрировать результат работы – построенное дерево, с отображением классов и распределением объектов по ним.

Заключение

В работе описан метод построения функций принадлежности, алгоритм формирования нечеткого дерева решений, структура базы правил. При построении учитывается представление исходных данных в виде упорядоченной последовательности лингвистических

тических переменных, что усложняет процесс фаззификации данных, в отличие от стандартного представления входных параметров в числовом виде. Сформированное дерево используется для дальнейшего построения нейро-нечеткого классификатора как системы нечеткого вывода для решения аналитических, научно-технологических задач [3].

ЛИТЕРАТУРА

1. Горбачев С.В., Сырымкин В.И. Нейро-нечеткие методы в интеллектуальных системах обработки и анализа многомерной информации. – Томск: Изд-во Томского государственного университета, 2014. – 510 с.
2. Butkiewicz B.S. Robust fuzzy clustering with fuzzy data // Lecture Notes in Computer Science. – V. 3528. – Berlin – Heidelberg: Springer-Verlag, 2005. – P.76–82.
3. Горбачев С.В., Горбачева Н.Н. Логически прозрачный нейро-нечеткий классификатор для многокритериальной оценки эффективности инновационного развития // Материалы VII Всероссийской научно-практической конференции с международным участием «Региональные проблемы преобразования экономики: интеграционные процессы и социально-экономическая политика региона», Махачкала, 9–10 ноября 2016 г., с. 84–92.

РАЗРАБОТКА USE-ИНСТРУМЕНТА ДЛЯ СОЗДАНИЯ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

К.Н. Попеляева, В.В. Томков

Томский государственный университет
ksyufkas@gmail.com, isirbis@gmail.com

Введение

В настоящее время программисты стремятся упростить процесс разработки программных продуктов различными способами: техническими и управленческими. К техническим можно отнести фреймворки, наборы библиотек для решения часто возникающих задач и т.д. Среди управленческих – методики организации проведения работ на всех этапах создания программного обеспечения, например, Agile, RUP и др. [1].

Иногда часть работ по добавлению или изменению существующего функционала можно переложить на конечных пользователей программного продукта. Так для этого используются различные инструменты для визуального программирования. Такие инструменты позволяют разрабатывать программы без углубленных знаний языков программирования.

1. Визуальное программирование. Проблемы и минимизация рисков

С развитием технологий программные продукты всё больше стали применять для автоматизации бизнес-процессов, протекающих в различных предприятиях. Часть из этого программного обеспечения практически не изменяется в процессе жизни всего предприятия, а некоторая меняется очень часто.

В связи с увеличением потребностей у пользователей, приходится вкладывать всё больше ресурсов, чтобы программное обеспечение оставалось актуальным. Для этого используют технические и управленческие подходы. При использовании технических подходов очень много ресурсов уходит для содержания большого штата разработчиков. При управленческих возможно снизить расход ресурсов на разработку и поддержание программного обеспечения благодаря оптимизации того или иного рода.

Одной из таких оптимизаций является привлечение конечных пользователей программного продукта и небольшого повышения их квалификации. Для этого используются визуальные инструменты программирования, которые имеют ряд проблем.

Первая из них – это нерасширяемость, которая заключается в том, что пользователю предоставляется ограниченный набор методов для решения узкоспециализированных задач [2,3], которые успешно решаются, но при возникновении краевых случаев решение получается в лучшем случае крайне сложным, а в худшем – невозможным.

Вторая связана с генерацией кода, который будет выполняться. Как правило, он получается медленным в выполнении и сложным в оптимизации. Связано это с тем, что пользователям инструмента предлагают оперировать нечёткими абстракциями.

Третья заключается в бедности и запертости в них. Программное обеспечение, которое предоставляет бедный набор инструментов визуального программирования, например IDE для работы с GUI, ограничивает возможности разработчика при решении задач и буквально запирает его в себе.

Озвученные проблемы возникают перед разработчиками при создании инструментов, которые должны решать задачи без чётко определенных границ [4].

Универсального решения всех озвученных проблем нет, но существует возможность снизить фактор риска провала инструмента еще на стадии его проектирования. Для этого предлагается придерживаться следующего набора правил:

- ограничивать спектр задач минимально, не сводя инструмент к решению узкоспециализированных задач, например, формирование SQL запроса, создание схемы печатной платы и т.д.;
- наполнять инструмент возможностями для работы с задачами, которые возникают в предметной области инструмента;
- оперировать элементарными абстракциями и связями между ними.

Для проверки этих правил на практике было принято решение о создании инструмента для визуального программирования, с помощью которого возможно создание web-приложения с использованием обработчиков событий пользователя.

2. Концепция визуального инструмента программирования

Первоначально нужно определиться с набором элементарных абстракций, с помощью которых пользователь сможет формировать интерфейс. Для этого за основу взяты HTML-элементы, которые можно сгруппировать следующим образом:

- текстовые блоки;
- списки;
- изображения;
- таблицы;
- элементы формы.

Каждая группа содержит в себе набор элементарных абстракций, таких как поле ввода, элемент списка и т.д. Каждый элемент – это объект, имеющий набор специфических параметров – свойств, которые его описывают, а также обязательно содержащий методы сериализации и десериализации, использующиеся для его представления при хранении и получения в БД.

Для реализации композиции элементов могут использоваться параметры: перечень дочерних элементов и родительский элемент.

Предложенная концепция позволяет в дальнейшем расширять набор элементов, которые можно будет использовать при разработке интерфейса и покрывать больший спектр задач, которые будут стоять перед пользователем данного инструмента, что снижает риск возникновения проблемы расширяемости инструмента и позволяет охватывать смежные области.

Следующий этап – формирование концепции создания или изменения пользователем функционала приложения, без необходимости использования языка программирования. Для этого создаются абстракции: «функциональный блок» и «связь». С их помощью можно реализовать концепцию блочного программирования, с небольшими изменениями – «функциональный блок» может содержать в себе набор входных и выходных параметров, а также источник и тип события, на которое он должен быть выполнен.

Таким образом, можно запрограммировать элемент use-case'a реализуемого функционала, что, в общем, позволяет реализовать необходимое функциональное требование, предъявляемое к программному продукту.

Этот подход позволит в дальнейшем генерировать код, который будет выполняться, наименее избыточным и легким для изменений и оптимизации.

Последней задачей, которая возникает при создании инструмента – хранение и генерация исполняемого кода. Для хранения будет использоваться формат JSON, т.к. он позволяет проводить сериализацию и десериализацию как простых, так и сложных структур (рис. 1), что является важным фактором, даже не смотря на использование элементарных абстракций, т.к. даже самый элементарный элемент может в себе содержать множество свойств и их композицию.

```
"WidgetButton": {
  "id": "1222",
  "position": {
    "x": 100,
    "y": 100
  },
  "size": {
    "width": {
      "default": 100,
      "min": 50,
      "max": 150
    },
    "height": {
      "default": 50,
      "min": 30,
      "max": 75
    }
  },
  "pressed": false,
  "relations": {
    "parentId": "1210",
    "childrens": [
      "1223",
      "1224"
    ]
  }
}
```

Рис. 1. Структура элемента в формате JSON

Перед тем, как сохранить элемент в БД, у каждого элемента, а также у всех его дочерних элементов, вызывается метод его сериализации. При получении из БД используется обратная операция – десериализация; происходит она аналогично, т.е. после десериализации одного элемента находятся его дочерние элементы и происходит вызов соответствующего метода у них.

После этого этапа получаем их в том виде, которым удобно оперировать для формирования исполняемого кода. Эта часть этапа разбита на две части:

- формирование HTML-элементов, которые будут формировать HTML-документ;
- формирование исполняемого кода, для обработки событий.

Для формирования HTML-элементов можно воспользоваться следующим механизмом: сопоставить каждому абстрактному элементу HTML-элемент или набор HTML-элементов.

При формировании исполняемого кода стоит обратить внимание на концепцию блочного программирования, которая используется для возможности реализации функционала будущего программного продукта, а именно – на то, что каждый функциональный блок можно представить в виде лексемы или набор лексем для соответствующего языка программирования, в данном случае – JavaScript. Полученный набор лексем уже можно отправить интерпретатору или компилятору на следующий этап – этап синтаксического анализа [5,6].

Заключение

При создании инструментов визуального программирования возникают определённого рода проблемы, которые могут сделать созданный инструмент непригодным к использованию как пользователем без углубленных знаний программирования, так и самим программистом.

К сожалению, их полностью нельзя избежать, но имеется возможность снизить риск их возникновения. Для этого приходится проводить анализ смежных областей применения инструмента и оперировать, по возможности, элементарными абстракциями.

В ходе анализа проблем был предложен набор правил для минимизации возникновения проблем. На их основе предложена концепция инструмента и создан сам инструмент.

ЛИТЕРАТУРА

1. Мартин Р.С., Нью-кирк Д.В., Косс Р.С. Быстрая разработка программ. Принципы, примеры, практика. – М.: Вильямс, 2004. – 752 с.
2. Визуальный язык ДРАКОН. Официальный сайт языка ДРАКОН, 2017. – URL: <http://drakon.su> (24.03.2018).
3. VisSim – система динамического моделирования / Новые информационные тех-нологии и программы, 2016. – URL: <http://pro-spo.ru/winmat/480-vissim> (01.04.2018).
4. VAX – инструмент для визуального программирования, или как написать SQL мышкой / Хабрахабр / Издательство интернет-проектов для IT-специалистов, 2006–2018. – URL: <https://habrahabr.ru/post/333750/> (04.04.2018).
5. Хантер Р. Основные концепции компиляторов. – М.: Вильямс, 2002. – 256 с.
6. Ахо А.В., Лам М.С., Сети Р. Компиляторы: принципы, технологии и инструментарий. – М.: Вильямс, 2011. – 1184 с.

КЛЕТОЧНО-АВТОМАТНОЕ МОДЕЛИРОВАНИЕ РАСПРОСТРАНЕНИЯ ЗВУКА В ЗАДАЧЕ ПОИСКА ОПТИМАЛЬНОГО РАСПОЛОЖЕНИЯ АКУСТИЧЕСКИХ СИСТЕМ

Я.И. Шмаков, М.Н. Головчинер

Томский государственный университет
yaroslavshmakovrus@gmail.com, golovchiner@mail.ru

Введение

В процессе поиска оптимального расположения акустических систем решаются две задачи [1]:

1. «Вычисление оптимального положения громкоговорителей относительно фронтальной стены».
2. «Определение расстояния от фронтальных громкоговорителей от боковых стен».

Обе задачи решаются инженером-акустиком самостоятельно, либо согласно рекомендациям производителя акустических систем по размещению акустических систем (АС).

Существуют программные комплексы, позволяющие моделировать и визуализировать распространение звука. Проблема использования таких программ – платный функционал, невозможность расширения функционала приложения.

Поэтому было решено написать своё программное обеспечение для моделирования распространения звука.

1. Постановка задачи

Стандартным подходом к решению задачи моделирования распространения звука является проведение точных численных расчётов путей распространения звука с учётом эффектов отражения, преломления, рассеивания [2] (как реализовано, например, в

модуле «Акустика» программного комплекса COMSOL, основанного на методе трассировки лучей).

В разрабатываемом приложении моделирование основано на использовании клеточно-автоматной модели, что позволяет вместо проведения расчётов реализовать моделирование с помощью пошагового комбинаторного алгоритма, что существенно упрощает решение задачи.

Таким образом, основной задачей разрабатываемого программного комплекса явилось создание модели распространения звука, решающей задачу поиска оптимального расположения АС в воздушном пространстве непустого помещения с использованием аппарата клеточных автоматов.

2. Модель клеточного автомата

Модель задается двумерным клеточным автоматом. Имея двумерную плоскость, разделим её на квадратные ячейки, каждая из которых содержит суммарный уровень звука прошедшего через неё в одном из четырёх направлений и суммарный уровень звука по всем направлениям. Заданы правила, по которым каждая клетка будет вычислять свое будущее состояние. Новое состояние зависит от состояния соседних восьми ячеек.

В основе модели распространения звука лежат геометрические теории фронтового и лучевого распространения звука.

В предлагаемой модели клеточного автомата возможны 4 состояния его ячеек:

- 1) отражатель,
- 2) поглотитель,
- 3) диффузный отражатель,
- 4) свободная ячейка.

Отражатель – ячейка, полностью отражающая полученный с фронтом волны уровень звука.

Поглотитель – ячейка, полностью поглощающая полученный с фронтом волны уровень звука.

Диффузный отражатель – ячейка, частично поглощающая и отражающая полученный с фронтом волны уровень звука.

Любая свободная ячейка может содержать направления дальнейшего распространения звука вместе с передаваемой величиной уровня звука, при этом ячейка не принадлежит ни к одному из вышеперечисленных типов.

2.1 Правила перехода ячеек из одного состояния в другое

Далее описаны основные правила перехода.

Для примера будем рассматривать изменение состояния ячейки с номером 5 в зависимости от состояний соседних ячеек за один шаг моделирования.

Введем следующие обозначения.

Стрелки (\leftarrow \rightarrow \downarrow \uparrow) указывают направление распространения фронта звуковой волны, либо направление отражения.

«О» со стрелкой – ячейка-отражатель, отражает звук без потерь, стрелка указывает направление отражения. Направление распространения фронта звуковой волны противоположно направлению отражения.

«П» – ячейка-поглотитель, полностью поглощает входящий звук

«Д» – ячейка-диффузный отражатель, имеет направление отражения, обозначаемое стрелкой.

Если около номера ячейки нет буквенного обозначения и, возможно, стоит стрелка, то перед нами свободная ячейка. Направление стрелки в свободной ячейке соответствует направлению распространения фронта волны. Можно считать, что ячейка является поглотителем с коэффициентом поглощения, меньшим единицы.

Допустим также, что любая ячейка со стрелкой передаёт некоторую величину уровня звука.

Правила переходов представлены в виде таблиц, состоящих из 2 частей. Левая часть таблицы означает состояние автомата на некотором шаге n , а правая часть – на шаге $n + 1$.

Обратим внимание, что обновление состояния ячейки зависит от состояния соседних ячеек. В данной модели рассматриваются ближайшие 8 соседей.

Фронт звуковой волны (ФЗВ) представлен стрелками. В левой части таблицы ячейка со стрелками представляет источник распространения уровня звука, в правой – стрелки будут стоять в тех ячейках, куда распространился ФЗВ, который в общем случае распространяется Т-образно.

Примеры.

Таблица 1

Прямой переход слева направо. Случай 1

1	2	3	1	2	3→
4	5→	6	4	5	6→
7	8	9	7	8	9→

Из табл. 1 видно, что 5 ячейка передаёт фронт звуковой волны вправо в ячейки 3, 6, 9.

Таблица 2

Прямой переход слева направо. Случай 2

1→	2	3	1	2→	3
4	5	6	4	5→	6
7	8	9	7	8	9

В табл. 2 представлена ситуация, аналогичная табл. 1. Фронт волны, распространяемый вправо из 1 ячейки, попадёт в ячейки 2 и 5. В 8 ячейку ФЗВ не доходит, так как 8 ячейка не является соседней для 1.

Таблица 3

Правило отражённого перехода слева направо

1	2	3	1	2→	3
4«O» ←	5	6	4«O»	5→	6
7	8	9	7	8→	9

Из табл. 3 видно, что фронт звуковой волны, попавший в 4 ячейку-отражатель, отразится в обратном направлении и распространит фронт волны вправо в ячейки 2, 5, 8.

Таблица 4

Диффузный переход

1	2	3	1←	2	3→
4	5«Д» ←	6	4←	5«Д»	6→
7	8	9	7←	8	9→

По табл. 4 видно, что фронт волны, попавший в диффузный отражатель, распространится с сохранением направления в ячейки 1, 4, 7 и отразится в ячейки 3, 6, 9.

Таблица 5

Комбинированные правила перехода. Пример 1

1	2	3
4	5«O»↓ →	6
7	8	9

1←	2↑	3
4←	5«O»	6
7	8	9

В табл. 5 ячейка отражатель с номером 5 может отражать звук в двух направлениях – вверх и влево. Следовательно, отражённый фронт волны дойдёт до ячеек 1, 2, 4.

Таблица 6

Комбинированные правила перехода. Пример 2

1«O» ←↑→	2	3
4	5	6
7	8	9

1«O»	2	3
4	5↓	6
7	8	9

В табл. 6 представлен пример отражения звукового фронта по 3 направлениям. Из 1 ячейки в 5 попадает отражённый вниз фронт звуковой волны. Отражения слева направо (←) и справа налево (→) являются взаимоисключающими.

Таблица 7

Правила перехода в поглотитель. Пример 2

1	2«П»	3
4→	5	6
7	8«П»	9

1	2«П»	3
4	5→	6
7	8«П»	9

В табл. 7 фронт волны проходит из ячейки 4 только в 5 ячейку, в ячейках 2 и 8 фронт волны поглощается.

В табл. 1–7 представлены основные правила перехода клеточного автомата. Случаи с другими направлениями передачи фронта волны либо его отражением аналогичны описанным.

2.2 Пример моделирования распространения звука

Рассмотрим использование предлагаемой модели при моделировании распространения звука в квадратной комнате. Источник звука статичен, параметры комнаты не меняются в процессе моделирования.

Имеем квадратное помещение 2,5х2,5 м. На расстоянии 1 м от передней стены установлен отражатель шириной 0,4 м, отражающий вниз. Установим источник звука на расстоянии 1 м от нижней стены и на 1,1 м от правой стены. Источник звука испускает сигнал в 5 дБ во всех 4 направлениях (вверх, вниз, влево, вправо). Коэффициент затухания звукового уровня 0,0625.

Используемые обозначения: «П» – ячейка-поглотитель, «O» – ячейка-отражатель, «И» – ячейка-источника уровня звука.

Цифры означают суммарное количество уровня звука (по всем направлениям), содержащееся в ячейке в данный момент времени. Уровень звука – величина звукового давления, измеряемая в дБ [3].



Рис. 1. Моделирование. Первые три шага

На рис. 1 представлено распространение уровня звука из источника.

Пропустим несколько шагов моделирования. Перейдём к моделированию отражения.



Рис. 2. Моделирование отражения. Шаги 6–8

Из шагов 6–8 видно, что часть звукового фронта отразилась вниз, а часть обогнула отражатель и продолжила движение вверх.

Пропустим ещё пару шагов, перейдём к поглощению.



Рис. 3. Моделирование поглощения. Шаги 8, 9

На рис. 3 представлено поглощение звукового фронта нижней «стенкой» помещения, состоящей из ячеек-поглотителей.

Результаты моделирования

По рис. 4 видно, что в процессе моделирования была выявлена зона, куда звуковой фронт не доходит (выделенная овалом область). Уровень звука в этих ячейках равен нулю.

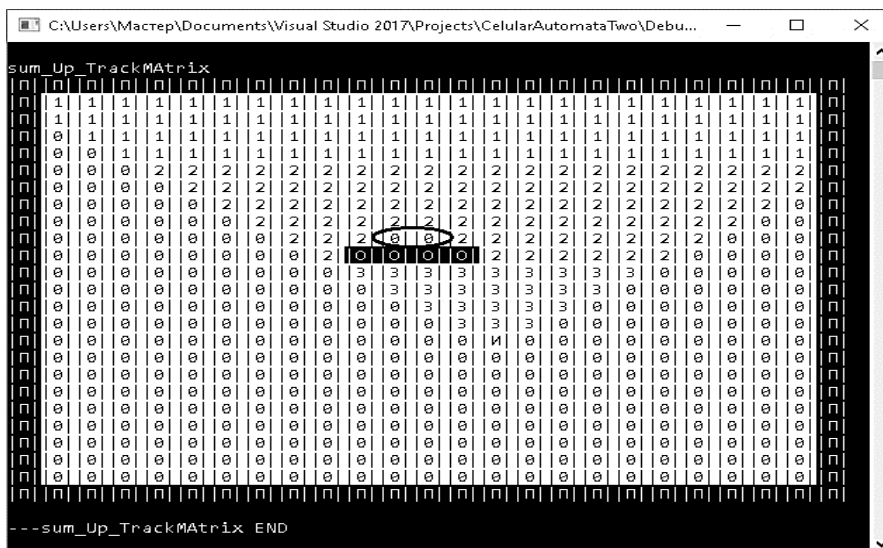


Рис. 4. Результат моделирования распространения звукового уровня вверх

На рис. 5 выделена треугольная область, где уровень звука максимален. Так произошло потому, что через выделенную область прошёл прямой и отражённый звуковой фронт.

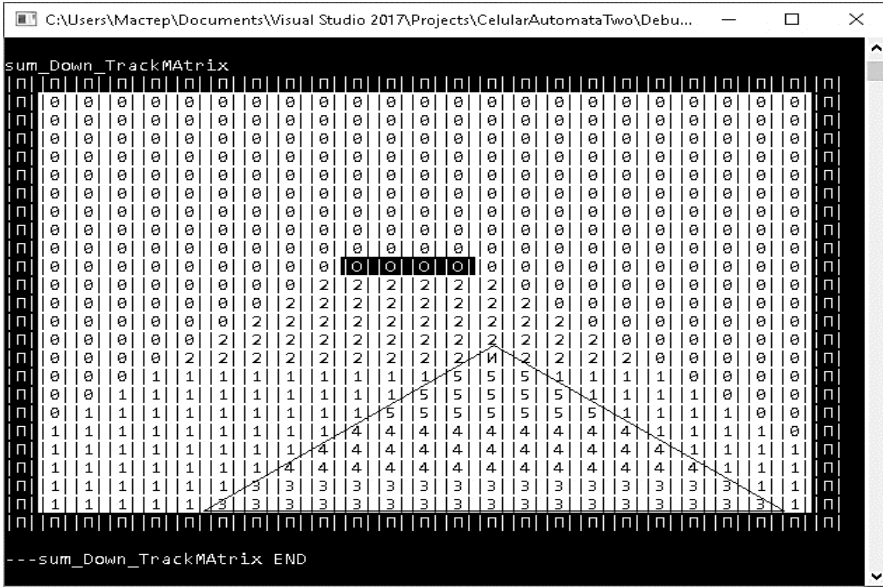


Рис. 5. Результат моделирования распространения звукового уровня вниз

Рис. 6 показывает общую картину моделирования распространения звукового фронта по всем направлениям.

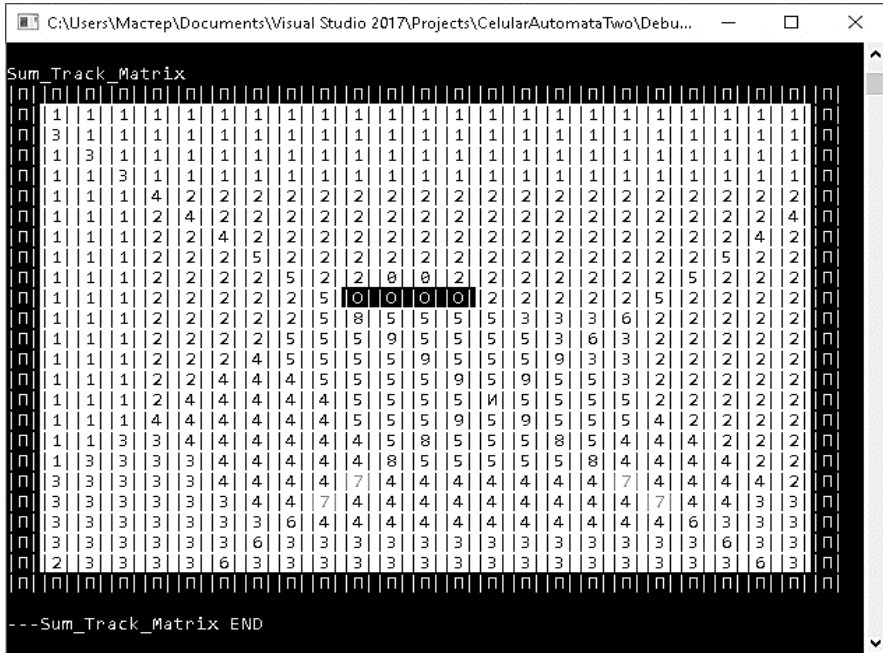


Рис. 6. Результат моделирования распространения звукового уровня по всем направлениям

В случае моделирования более сложной сцены (добавлен ещё один ряд отражателей, отражающий вверх, см. рис. 7) получаем следующий результат. В выделенных областях наблюдается повышенный уровень звука, что говорит о том, что в этих областях многократно проходил отражённый звук.

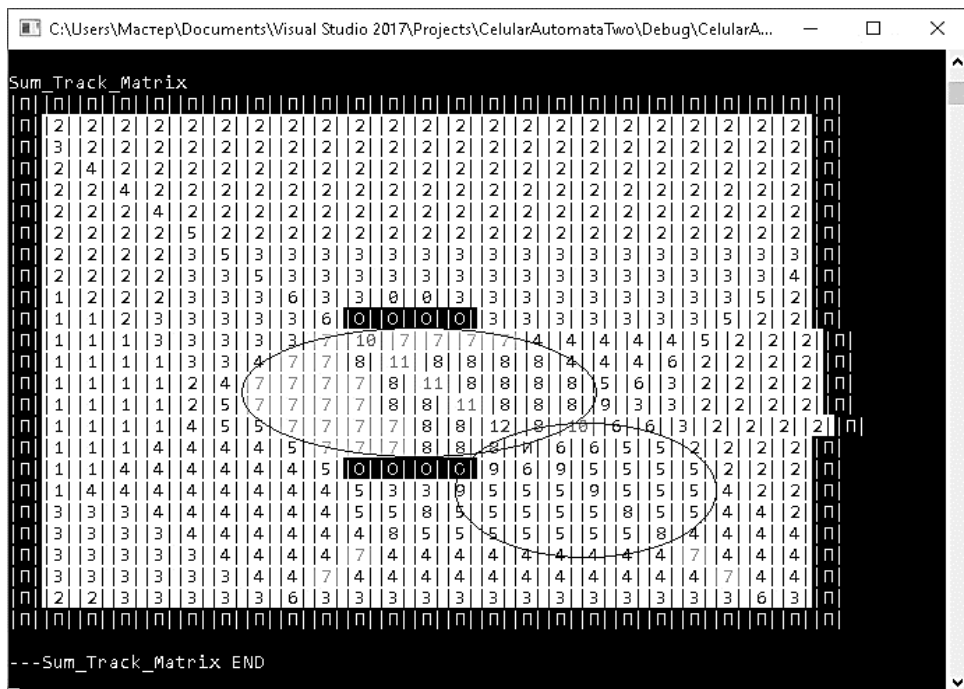


Рис. 7. Результат моделирования распространения звукового уровня по всем направлениям при наличии на сцене 2 рядов отражателей

Заключение

В настоящей работе предлагается клеточно-автоматная модель распространения уровня звука. Результаты моделирования в достаточной мере отображают акустические характеристики распространяемого уровня звука. С помощью полученных карт уровня звука можно вычислить оптимальное расположение акустических систем.

Клеточно-автоматная модель реализована в программе, с помощью которой можно получить двумерную карту уровня звука по предоставленным входным данным: размеры помещения, количество источников звука, начальный уровень звука источника и его направленность, коэффициенты поглощения/отражения элементов помещения, координаты элементов помещения и их количество.

ЛИТЕРАТУРА

1. Методология поиска оптимального расположения громкоговорителей – http://www.acoustic.ru/ref_book/articles/13/ (10.04.2018)
2. Елкин А.Б., Маслеева О.В. Акустическое загрязнение: Методические указания к выполнению практической работы по дисциплине «Экология». – Н. Новгород, НГТУ, 2014. – 11 с.
3. ГОСТ 12.1.050-86 Межгосударственный стандарт. Система стандартов безопасности труда. Методы измерения шума на рабочих местах.

РАЗРАБОТКА И РЕАЛИЗАЦИЯ СПЕЦИАЛИЗИРОВАННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КУРСА «МЕТОДЫ ОПТИМИЗАЦИИ»

В.З. Тычинский

*Томский государственный университет
tychinskiy.vyacheslav@gmail.com*

Введение

Задачей методов оптимизации в общем случае является выбор какого-либо действия x^* из множества допустимых действий D , при этом соблюдаются два условия:

- 1) каждому действию x из D взаимно однозначно соответствует величина $f(x)$, характеризующая его качество;
2. $f(x)$ достигает экстремума в точке x^* .

Целями лабораторных работ курса «Методы оптимизации» является формирование у студентов общего представления о специфике оптимизационных задач, а также наглядная демонстрация преимуществ и недостатков тех или иных методов решения таких задач. Для выполнения этих целей в 2001 году было разработано специализированное программное обеспечение, однако при его анализе был обнаружен ряд недостатков:

- проблемы совместимости с современными операционными системами;
- отсутствие файлов с исходным кодом (для внедрения улучшений и исправлений);
- отсутствие кроссплатформенности;
- ошибки в реализациях некоторых алгоритмов, связанные с недостаточной точностью вычислений;
- неудобный интерфейс.

Ввиду вышеперечисленных причин имеется необходимость в разработке нового программного обеспечения, обладающего тем же функционалом, но лишённого вышеперечисленных недостатков. При этом такое решение предпочтительнее использования готовых пакетов прикладных программ, поскольку

- 1) даёт возможность выполнения лабораторных работ вне университета;
- 2) выполнение лабораторных работ возможно студентами, не обладающими опытом работы с пакетами прикладных программ;
- 3) программное обеспечение проще адаптировать при изменении инфраструктуры, в то время как в случае использовании пакетов прикладных программ это может потребовать больших затрат.

Для разработки программного обеспечения был выбран язык C++, обеспечивающий хорошее быстродействие, что имеет существенное значение в тех лабораторных работах, где необходимо проводить большое количество вычислений. Кроме того, при разработке использованы библиотеки Qt, обеспечивающие кроссплатформенность и позволяющие уменьшить связность компонент программного обеспечения.

1. Постановка задачи

В лабораторных работах курса "Методы оптимизации" в настоящее время рассматриваются методы 5 разделов:

1. Минимизация функции градиентными методами
2. Минимизация функции овражными методами
3. Минимизация функции методами прямого поиска
4. Решение задачи линейного программирования симплекс-методом
5. Решение транспортной задачи методом потенциалов

В каждом из методов пользователем задаются различные исходные данные и параметры, необходимые для работы метода.

На основе этих входных данных требуется выполнить выбранный метод и вывести на экран результаты его работы (графически или в виде текстового отчёта). В некоторых методах требуется интерактивное взаимодействие с пользователем.

Для реализации этих методов программное обеспечение должно обладать следующими возможностями:

- работа с функциями многих переменных различного типа с высокой точностью (не менее $0.5 \cdot 10^{-5}$);
- изображение графика функции одной переменной и линий уровня функции двух переменных;
- включать в себя систему генерации различных видов функций в зависимости от введённого варианта;

Кроме того, для упрощения сопровождения и возможности последующего развития (например, добавление новых методов и алгоритмов при изменении программы курса), программное обеспечение должно быть легко модифицируемым и расширяемым. Этой цели можно достигнуть разделением программного обеспечения на несколько модулей, слабо связанных между собой.

Для осуществления обозначенной задачи служат следующие подзадачи:

1. Разработка общей архитектуры ПО.
2. Разработка и реализация базового модуля.
3. Разработка и реализация графического модуля.
4. Разработка и реализация модуля генерации отчётов.
5. Реализация всех разделов курса "Методы оптимизации".
 - 5.1. Разработка системы вариантов для генерации функций (необходимы для методов минимизации функций).
 - 5.2. Реализация всех 11 методов курса.

2. Анализ предметной области

С использованием [1–4] были изучены методы, рассматриваемые на лабораторных работах курса «Методы оптимизации» и проведён их анализ с целью определения необходимого функционала программного обеспечения и формулировки требований к его архитектуре.

2.1. Градиентные методы поиска экстремума функции

Градиентные методы являются одной из наиболее распространенных групп методов поиска безусловного экстремума. Обозначим через $\nabla f(\mathbf{X})$ вектор-градиент функции $f(\mathbf{X})$. Пусть поиск экстремума начинается с некоторой произвольной точки $\mathbf{X}^{(0)}$. Тогда движение по градиентному методу описывается следующим рекуррентным соотношением: $\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - t_k \cdot \nabla f(\mathbf{X}^{(k)})$, $t_k > 0$, $k = 0, 1, 2, \dots$, где t_k – величина шага на k -й итерации.

В различных вариантах градиентного метода используются различные способы выбора скаляра t_k .

Условием останова методов является ограниченность по модулю всех частных производных функции $f(\mathbf{X})$ в точке $\mathbf{X}^{(k)}$ (при заданной точности ε).

Исходя из задач, решаемых градиентными методами, можно сделать следующие заключения:

1. Разрабатываемое программное обеспечение должно обладать возможностью обработки функций многих переменных различных видов.
2. В тех алгоритмах, где скаляр t_k вычисляется на каждом шаге, необходимо решать задачу оптимизации функции одной переменной. Поскольку в зависимости от вида функций и выбора начальной точки значения градиента функции могут дос-

тигать больших значений ($> 10\,000$), возникает проблема точности вычисления скаляра t_k .

2.2. Овражные методы поиска экстремума функции

Градиентные методы показывают низкую эффективность при определении экстремума овражных функций (линии уровня таких функций вытянуты вдоль поверхности, называемой «дном оврага»). Овражные методы позволяют за небольшое количество итераций найти *окрестность* точки экстремума таких функций. Более точный поиск, если это необходимо, проводится с использованием других методов.

Основная идея овражных методов – осуществить спуск ко дну оврага, а затем двигаться вдоль него в направлении точки минимума.

Для реализации овражных методов используются те же структуры данных, что и для градиентных, поэтому их реализация не повлечёт за собой изменений структуры программного обеспечения.

2.3. Методы одномерного поиска экстремума функции

В рамках курса рассматриваются следующие методы.

I. Метод золотого сечения. В основе метода лежит принцип деления отрезка в пропорциях золотого сечения. На отрезке $[a, b]$, внутри которого находится искомый экстремум, выбираются две точки c и d ($c < d$), такие, что отношения длин отрезков $[a, b] : [c, b] = [a, b] : [a, d] = \frac{2}{\sqrt{5}-1}$ (пропорция золотого сечения). Затем в точках c и d сравниваются значения целевой функции: если $f(c) > f(d)$ (в случае поиска минимума), то $a = c$, в противном случае $b = d$.

II. Метод Дэвиса – Свенна – Кэмпи. Метод состоит из двух этапов:

1. Поиск отрезка, содержащего искомый экстремум: для заданных x_0 и Δx :

– $x_1 = x_0 + \Delta x$;

– если $f(x_1) \leq f(x_0)$, то $\Delta x = 2 \cdot \Delta x$, иначе $\Delta x = -\Delta x$;

– пока $f(x_{k+1}) < f(x_k)$, $\Delta x = 2 \cdot \Delta x$, $k = k + 1$;

– $x_{k+2} = x_{k+1} - \frac{\Delta x}{2}$.

2. Одна из точек x_{k-1} , x_k , x_{k+1} , x_{k+2} отбрасывается согласно методу золотого сечения, по оставшимся точкам проводится квадратичная интерполяция. Если заданная точность не достигнута, перейти к п. 1.

III. Метод Пауэлла. Метод использует схожий с методом ДСК принцип работы:

1. $x_1 = x_0 + \Delta x$.

2. Если $f(x_1) \leq f(x_0)$, то $x_2 = x_0 + 2 \cdot \Delta x$, иначе $x_2 = x_0 - \Delta x$.

3. Проводится квадратичная интерполяция по точкам x_0 , x_1 , x_2 .

4. Одна из четырёх точек отбрасывается по методу золотого сечения.

5. Если заданная точность не достигнута, перейти к п. 1.

Поскольку функции одной переменной являются подмножеством функций многих переменных, для реализации методов одномерного поиска возможно использование тех же структур данных, что и для градиентных методов.

2.4. Симплекс-метод решения задачи линейного программирования

Каноническая форма записи ЗЛП имеет следующий вид:

$$L(x_1, \dots, x_n) = \sum_{i=1}^n c_i \cdot x_i \rightarrow \min_{x_i, i=1, n}$$

$$\sum_{j=1}^n a_{ij}x_j = b_i, \quad i = \overline{1, m},$$

$$x_j \geq 0, \quad j = \overline{1, n}.$$

Множество D векторов $\mathbf{X} = (x_1, x_2, \dots, x_n)$, удовлетворяющих ограничениям и условиям неотрицательности переменных ЗЛП, называют областью определения задачи. Любой вектор \mathbf{X} из D называется допустимым решением или планом ЗЛП.

Идея симплекс-метода заключается в последовательном переборе точек, подозрительных на экстремум, причём при переходе от одной точки к другой значение линейной формы не должно увеличиваться.

Поскольку симплекс-метод работает с матричными структурами, очевидно, что при его реализации необходимо создание новой структуры данных (т.к. структуры данных, применяемые в рассмотренных ранее методах, неприменимы к матрицам).

2.5. Транспортная задача и её решение методом потенциалов

Пусть A_1, A_2, \dots, A_n – поставщики определённого однородного товара, a_i – запасы товара у поставщика A_i , $i = \overline{1, n}$; B_1, B_2, \dots, B_m – потребители этого товара, b_j – потребность в товаре у потребителя B_j , $j = \overline{1, m}$; c_{ij} – стоимость перевозки одной единицы товара от поставщика A_i потребителю B_j . Необходимо перевезти весь товар от поставщиков к потребителям так, чтобы исчерпать все его запасы поставщиков, удовлетворить потребности в товаре у потребителей и при этом минимизировать суммарную стоимость перевозок.

Для решения транспортной задачи методом потенциалов необходимо соблюдение двух условий:

1. Задача является сбалансированной, т.е. суммарные запасы товара всех поставщиков совпадают с суммарными потребностями в товаре всех потребителей.
2. Найден опорный план задачи, например, методом минимального тарифа, при котором на каждом шаге (до тех пор, пока весь товар не будет распределён) выбирается клетка с минимальным значением тарифа и для неё устанавливается максимально возможный объём перевозки.

Для определения потенциалов каждому поставщику A_i и каждому потребителю B_j сопоставляются величины u_i и v_j соответственно так, чтобы для всех базисных клеток плана было выполнено соотношение: $u_i + v_j = c_{ij}$, при этом u_1 полагается равным нулю.

Затем для каждой свободной клетки плана вычисляется величина $\Delta c_{ij} = c_{ij} - (u_i + v_j)$.

План является оптимальным, если все $\Delta c_{ij} \geq 0$. В противном случае для ячейки с отрицательной разностью Δc_{ij} можно построить цикл, в котором кроме этой клетки все остальные являются базисными. Такой цикл всегда существует и единственен. Затем среди чётных элементов цикла пересчёта определяется величина ρ , соответствующая объёму наименьшей из перевозок. После этого происходит переход к следующему опорному плану: объём перевозок нечётных элементов цикла увеличивается на величину ρ , а чётных – уменьшается.

Несмотря на то, что данный метод также работает с данными в виде таблицы (матрицы), использование структуры данных симплекс-метода не представляется возможным, поскольку в транспортной задаче ячейки таблицы имеют более сложную структуру, совмещая в себе величины c_{ij} , Δc_{ij} , x_{ij} (объём перевозки от i -го поставщика j -ому потребителю в опорном плане).

3. Общая архитектура программного обеспечения

В результате анализа предметной области можно сделать вывод, что программное обеспечение должно обладать возможностью обработки различных типов данных (в

случае методов минимизации функций – работа с функциями различных видов, в случае ЗЛП и транспортной задачи – обработка данных в виде таблиц). Поэтому оптимальным решением является разделение множества всех задач на несколько независимых компонент (модулей), каждая из которых содержит реализацию метода или класса схожих методов.

Помимо подобных «модулей методов» имеется потребность во вспомогательных модулях, которые не связаны с определённой предметной областью, но необходимы для обеспечения наглядности и удобства выполнения лабораторных работ (изображение графиков функций, отображение и сохранение текстовых отчётов и т.п.).

Общая структура программного обеспечения представлена следующей схемой.

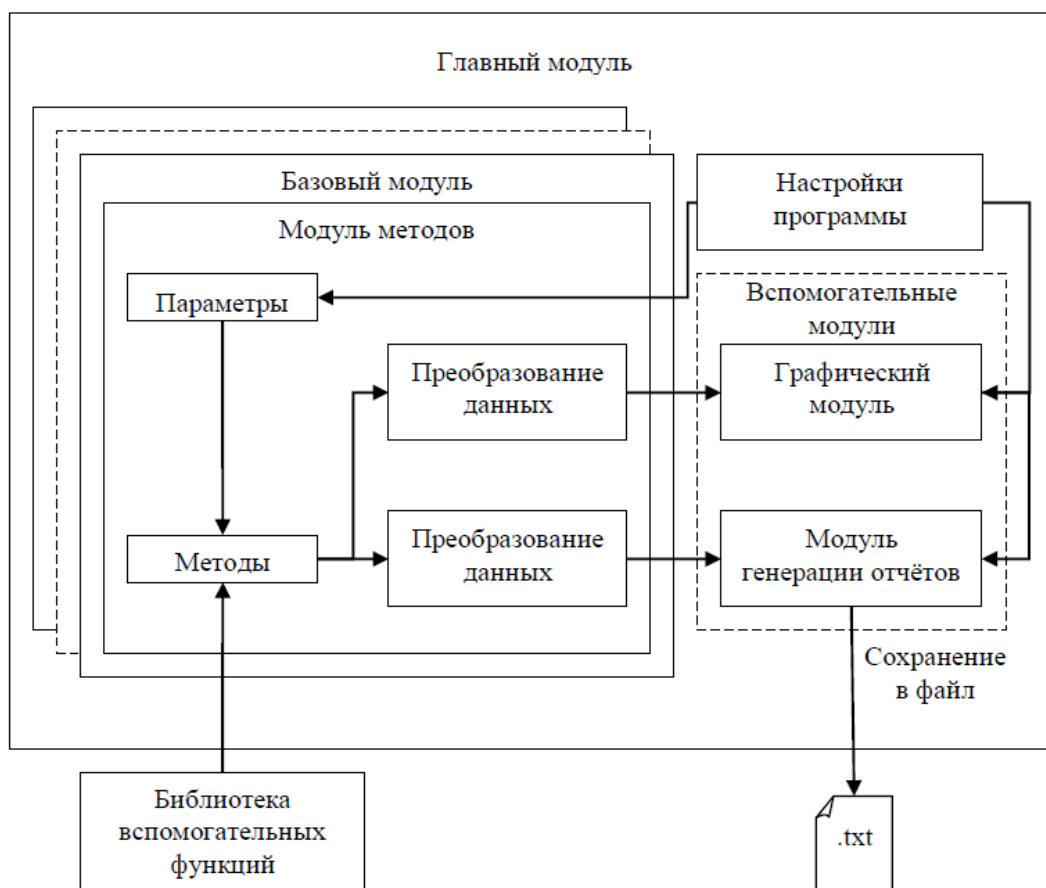


Рис. 1. Архитектура программного обеспечения

Рассмотрим назначение отдельных компонент программного обеспечения более подробно.

1. Основой для модулей методов является т.н. "Базовый модуль", в котором объявлены прототипы функций взаимодействия с модулями других групп. При реализации нового модуля методов он наследуется от базового модуля, и определяет необходимые ему функции. Такое решение позволит дополнительно уменьшить связность компонент между собой, что даёт ряд преимуществ:

- при изменении/добавлении модуля корректировке подвергается только он сам (поскольку интерфейсы взаимодействия при этом не меняются);

– подобное архитектурное решение позволяет с лёгкостью извлечь модуль из программного обеспечения и использовать его независимо/встроить в другое программное обеспечение.

2. Модуль методов содержит реализацию метода или нескольких схожих по решаемым задачам методов. После завершения работы метода модуль также отвечает за преобразование результатов его работы в типы данных, описанные в прототипах функций базового модуля для возможности их последующей обработки вспомогательными модулями.

3. Главный модуль устанавливает связи базовых модулей с вспомогательными модулями, содержит в себе настройки программного обеспечения и отвечает за главный интерфейс программного обеспечения.

4. Особенности реализации программного обеспечения

Связь модулей реализуется посредством системы сигналов и слотов, которую предоставляют библиотеки Qt. Данная система является крайне гибкой и позволяет устанавливать связи вида «один ко многим» и «многие к одному», не сообщая при этом объектам информацию о структуре объектов, с которыми они связаны посредством этой системы.

В целях повышения точности в ходе вычислений выражения для функции, её частных производных и точки минимума задаются аналитически. Для программного вычисления этих выражений используется польская инверсная запись, реализованная со следующим функционалом.

1. Допускаются целые, дробные и отрицательные числа.

2. Допускаются следующие операции: +, -, *, /, ^ (возведение в степень), % (остаток от деления).

3. Переменные-аргументы функции в её аналитическом выражении задаются в формате

«Имя аргументаИндекс аргумента»,

где именем аргумента является английская литера "x", индексом – цифра в диапазоне 1–9.

4. При создании нового варианта аналитический вид функции указывается с точностью до констант. Константы генерируются по формулам, которые задаёт преподаватель. Эти неопределённые константы в аналитическом выражении функции задаются в формате

«Имя константыИндекс константы»,

где именем константы могут являться любые строчные английские литеры (за исключением литеры "x"), индексом – цифра в диапазоне 1–9.

При задании формул возможно использование компонент составного ключа с целью генерации различных видов функций для студентов:

– номер студента в списке группы (имя ключа – "kl", возможный диапазон – 01–50);

– номер группы студента (имя ключа – "kg", возможный диапазон – 01–09);

– год поступления студента (имя ключа – "ky", возможный диапазон – 00–99).

Для работы с графикой используется библиотека QCustomPlot, обеспечивающая масштабирование и перемещение графика.

Для построения линий уровня функции использовался метод сканирования (построчной развертки). Пусть задан набор линий уровня функции: $C_1 < C_2 < \dots < C_N$. Исходная функция "просматривается" с достаточно малыми шагами h_x, h_y строка за строкой в пределах заданных границ $a_x \leq x \leq b_x, a_y \leq y \leq b_y$. При переходе значения функции $z = f(x, y)$ из одного интервала уровней $[C_{i-1}, C_i]$ в другой, на карте изображается точка. Сканирование в алгоритме производится вдоль "строк", параллельных оси Ox . В

связи с этим могут плохо воспроизводиться участки изолиний, близкие к горизонтальным, даже при малых шагах сканирования h_x, h_y . Алгоритм работает сравнительно медленно, поскольку для достаточно качественного "поточечного" воспроизведения изолиний желательно предусматривать мелкую сетку точек сканирования. Однако достоинством этого алгоритма является то, что он строит сразу всё семейство изолиний.

Заключение

Результатом данной работы является программное обеспечение, способное полностью заменить используемое в настоящее время программное обеспечение для лабораторных работ курса «Методы оптимизации». Оно не имеет проблем совместимости с современными ОС, а также может поддерживать несколько платформ. Кроме того, благодаря аналитическим выражениям для частных производных, данное программное обеспечение имеет возможность гораздо точнее вычислять их значения, что позволяет получать траектории методов, наиболее приближенные к оптимальным.

Ввиду наличия файлов с исходным кодом и модульной структуры, данное программное обеспечение имеет большой потенциал модернизации. Используемые архитектурные решения позволяют в короткие сроки расширить охват предметной области программным обеспечением или создать новое программное обеспечение, предназначенное для решения задач из иных предметных областей, (в первую очередь связанных с вычислениями).

ЛИТЕРАТУРА

1. Корицнов Ю.М. Математические основы кибернетики. – М.: Энергоатомиздат, 1972. – 376 с.
2. Калиткин Н.Н. Численные методы. – М.: Наука, 1978. – 512 с.
3. Ахмеров Р.Р. Методы оптимизации гладких функций. [Электронный ресурс] – URL: http://w.ict.nsc.ru/books/textbooks/akhmerov/mo_unicode/index.html (16.06.2018)
4. Метод градиентного спуска [Электронный ресурс]. – URL: http://www.machinelearning.ru/wiki/index.php?title=%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%B3%D1%80%D0%B0%D0%B4%D0%B8%D0%B5%D0%BD%D1%82%D0%BD%D0%BE%D0%B3%D0%BE_%D1%81%D0%BF%D1%83%D1%81%D0%BA%D0%B0 (16.06.2018)

ЧИСЛЕННЫЙ МЕТОД РЕШЕНИЯ ЗАДАЧИ ЭЛЕКТРОМАГНИТНОГО РАССЕЯНИЯ НА ТОНКИХ ПАРАЛЛЕЛЬНЫХ ИДЕАЛЬНО ПРОВОДЯЩЕМ И ДИЭЛЕКТРИЧЕСКОМ ЦИЛИНДРАХ

А.Г. Дмитренко, О.М. Балашова

Томский государственный университет

dmitr@fpmk.tsu.ru, jkz@pisem.net

Введение

Значительный интерес для исследователей представляет изучение рассеяния электромагнитных волн в резонансной частотной области на структурах, состоящих из одного или нескольких тонких цилиндров конечной длины. Этот интерес обусловлен необходимостью решения таких практически важных проблем, как проблемы радиолокационной заметности, идентификации объектов, оценки рассеяния диэлектрическими или металлическими цилиндрическими деталями различных геометрически сложных тел и др.

Под тонким цилиндром обычно понимается цилиндрическое тело, поперечные размеры которого много меньше его длины и длины падающей волны. Анализ имеющейся в распоряжении авторов литературы показывает, что известны работы, например, [1–3], в которых рассмотрено рассеяние электромагнитной волны на одиночном тонком прямолинейном идеально проводящем цилиндре, а также работы, например, [4–6], в которых рассмотрено рассеяние электромагнитной волны на одиночном тонком

однородном прямолинейном диэлектрическом цилиндре. Некоторый вклад в решение проблемы рассеяния электромагнитной волны на тонких цилиндрах внесён и работами одного из авторов данной статьи. Так, в [7] рассмотрено рассеяние на структурах, состоящих из нескольких тонких прямолинейных идеально проводящих цилиндров, в [8] предложен новый метод решения задачи электромагнитного рассеяния на тонком диэлектрическом цилиндре, а в [9] рассмотрено электромагнитное рассеяние на структурах, состоящих из нескольких тонких диэлектрических цилиндров конечной длины. Однако в известной литературе до сих пор отсутствуют работы, посвященные рассеянию на структурах, состоящих как из идеально проводящих, так и из диэлектрических тонких цилиндров.

В данной работе предложен численный метод решения задачи электромагнитного рассеяния для одной из таких структур – структуры, состоящей из двух параллельных тонких цилиндров, один из которых является диэлектрическим, а другой – идеально проводящим. Данный метод является объединением методов, предложенных ранее в [8] и [10] для решения задач электромагнитного рассеяния на тонких диэлектрических и идеально проводящих цилиндрах.

1. Постановка задачи

Геометрия задачи показана на рис. 1. Будем рассматривать стационарную (зависимость от времени выбрана в виде $e^{-i\omega t}$) задачу дифракции электромагнитного поля \vec{E}_0, \vec{H}_0 на структуре, состоящей из двух тонких прямолинейных параллельных цилиндров, один из которых является диэлектрическим, а другой – идеально проводящим. Диэлектрический цилиндр имеет длину l_d , радиус r_d и характеризуется электродинамическими параметрами ϵ_i, μ_i . Идеально проводящий цилиндр имеет длину l_p , радиус r_p . Для цилиндров выполняются условия $2r_d \ll \lambda$, $r_d \ll l_d$, $2r_p \ll \lambda$, $r_p \ll l_p$, где λ – длина падающей волны. Декартова система координат $Oxyz$ выбрана таким образом, что её начало O совпадает с серединой осевой линии диэлектрического цилиндра, а ось z направлена вдоль осевой линии. Структура размещена в однородной среде D_e с электродинамическими параметрами ϵ_e, μ_e . Требуется найти рассеянное поле $\{\vec{E}_e, \vec{H}_e\}$ в среде D_e .

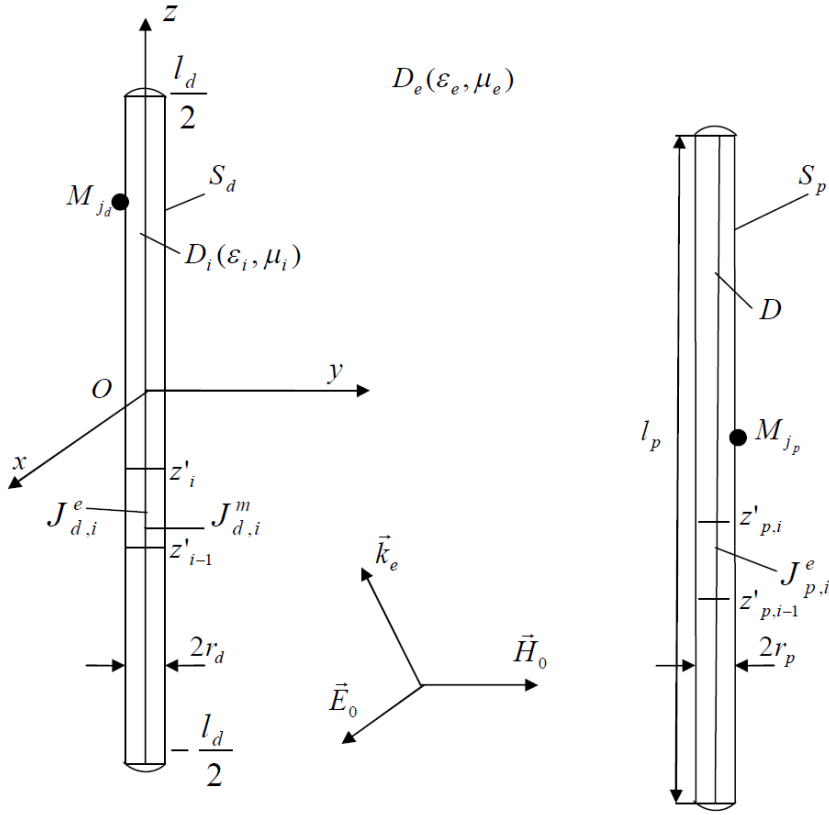


Рис. 1. Геометрия задачи

Кроме поля $\{\vec{E}_e, \vec{H}_e\}$ в D_e , существует поле $\{\vec{E}_i, \vec{H}_i\}$ внутри диэлектрического цилиндра (в области D_i).

Поля $\{\vec{E}_e, \vec{H}_e\}$ и $\{\vec{E}_i, \vec{H}_i\}$ должны удовлетворять уравнениям Максвелла

$$\text{rot } \vec{E}_e = i\omega\mu_e \vec{H}_e, \quad \text{rot } \vec{H}_e = -i\omega\varepsilon_e \vec{E}_e \quad \text{в } D_e, \quad (1)$$

$$\text{rot } \vec{E}_i = i\omega\mu_i \vec{H}_i, \quad \text{rot } \vec{H}_i = -i\omega\varepsilon_i \vec{E}_i \quad \text{в } D_i, \quad (2)$$

граничным условиям

$$[\vec{n}_i, (\vec{E}_i - \vec{E}_e)] = [\vec{n}_i, \vec{E}_0], \quad [\vec{n}_i, (\vec{H}_i - \vec{H}_e)] = [\vec{n}_i, \vec{H}_0] \quad (3)$$

на поверхности S_d диэлектрического цилиндра и граничным условиям

$$[\vec{n}, \vec{E}_e] = -[\vec{n}, \vec{E}_0] \quad (4)$$

на поверхности S_p идеально проводящего цилиндра. Кроме того, поле $\{\vec{E}_e, \vec{H}_e\}$ в D_e должно удовлетворять условиям излучения

$$[\sqrt{\varepsilon_e} \vec{E}_e, \vec{R}/R] + \sqrt{\mu_e} \vec{H}_e = O(R^{-1}), \quad [\mu_e \vec{H}_e, \vec{R}/R] - \sqrt{\varepsilon_e} \vec{E}_e = O(R^{-1}), \quad R \rightarrow \infty. \quad (5)$$

В выражениях (3)–(5) \vec{n}_i – единичный вектор нормали к поверхности S_d диэлектрического цилиндра; \vec{n} – единичный вектор нормали к поверхности S_p идеально проводящего цилиндра; $R = \sqrt{x^2 + y^2 + z^2}$; $[\vec{a}, \vec{b}]$ – векторное произведение. Отметим, что при записи граничных условий (3), (4) учтено, что поле во внешней среде D_e представлено в виде суммы рассеянного \vec{E}_e, \vec{H}_e и возбуждающего \vec{E}_0, \vec{H}_0 полей.

2. Метод решения задачи

Суть предлагаемого метода решения задачи заключается в следующем. Введём на оси диэлектрического цилиндра неизвестные непрерывно распределенные электрический \vec{J}_d^e и магнитный \vec{J}_d^m токи, а на оси идеально проводящего цилиндра – только непрерывно распределенный электрический ток \vec{J}_p^e . Представим неизвестное рассеянное поле в виде суммы полей введенных вспомогательных токов:

$$\begin{aligned}\vec{E}_e(M) &= \frac{i}{\omega\epsilon_e} \text{rot rot } \vec{\Pi}_d^e - \text{rot } \vec{\Pi}_d^m + \frac{i}{\omega\epsilon_e} \text{rot rot } \vec{\Pi}_p^e, \\ \vec{H}_e(M) &= \text{rot } \vec{\Pi}_d^e + \frac{i}{\omega\mu_e} \text{rot rot } \vec{\Pi}_d^m + \text{rot } \vec{\Pi}_p^e, \\ \vec{\Pi}_d^e &= \int_{-l_d/2}^{l_d/2} \frac{\vec{J}_d^e(z') e^{ik_e R_{MM'}}}{4\pi R_{MM'}} dz', \quad \vec{\Pi}_d^m = \int_{-l_d/2}^{l_d/2} \frac{\vec{J}_d^m(z') e^{ik_e R_{MM'}}}{4\pi R_{MM'}} dz', \\ \vec{\Pi}_p^e &= \int_{l_p} \frac{\vec{J}_p^e(z'_p) e^{ik_e \bar{R}_{MM'}}}{4\pi \bar{R}_{MM'}} dz'_p, \quad M \in D_e.\end{aligned}\tag{6}$$

В выражениях для $\vec{\Pi}_d^e$ и $\vec{\Pi}_d^m$ интегрирование проводится вдоль оси z диэлектрического цилиндра; $R_{MM'} = \sqrt{x^2 + y^2 + (z - z')^2}$ – расстояние от точки z' на оси диэлектрического цилиндра до точки M в D_e ; x, y, z – декартовы координаты точки M ; $k_e = \omega\sqrt{\epsilon_e\mu_e}$.

В выражении для $\vec{\Pi}_p^e$ интегрирование проводится вдоль оси идеально проводящего цилиндра; $\bar{R}_{MM'} = \sqrt{(x - x_0)^2 + (y - y_0)^2 + (z - z'_p)^2}$ – расстояние от точки z'_p на оси идеально проводящего цилиндра до точки M в D_e ; x_0 и y_0 – соответствующие координаты осевой линии в системе отсчета $Oxyz$.

Для представления электромагнитного поля \vec{E}_i, \vec{H}_i внутри диэлектрического цилиндра введём вспомогательную поверхность S_i (рис. 2), охватывающую цилиндр, аналогично тому, как это сделано в [11]. Поверхность S_i также представляет собой круговой цилиндр со сферически скругленными торцами; радиус цилиндра равен R_i , его длина равна длине рассеивающего цилиндра l_d . Выберем на вспомогательной поверхности S_i конечную совокупность точек $\{M_{n,i}\}_{n=1}^{N_i}$, в каждой из которых разместим пару независимых вспомогательных элементарных электрических диполей с моментами $\vec{p}_{\tau_1}^{n,i} = p_{\tau_1}^{n,i} \vec{e}_{\tau_1}^{n,i}$ и $\vec{p}_{\tau_2}^{n,i} = p_{\tau_2}^{n,i} \vec{e}_{\tau_2}^{n,i}$.

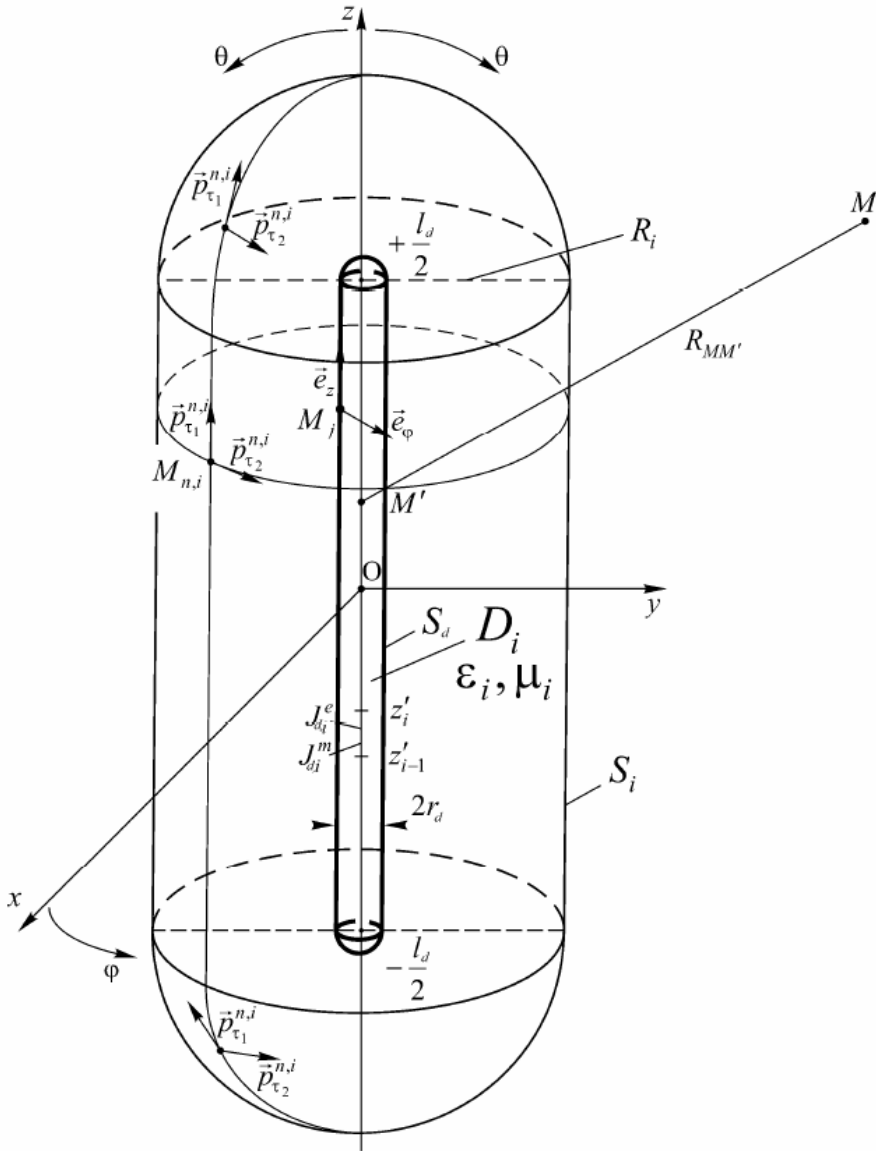


Рис. 2. Вспомогательная поверхность, охватывающая диэлектрический цилиндр

Единичные векторы $\vec{e}_{\tau_1}^{n,i}$ и $\vec{e}_{\tau_2}^{n,i}$ выбраны в плоскости, касательной к S_i в точке $M_{n,i}$; вектор $\vec{e}_{\tau_1}^{n,i}$ расположен в сечении $\varphi = \text{const}$, проходящем через точку $M_{n,i}$, а вектор $\vec{e}_{\tau_2}^{n,i}$ ортогонален вектору $\vec{e}_{\tau_1}^{n,i}$. Предполагается, что диполи, размещенные на S_i , излучают в однородную среду с проницаемостями ϵ_i и μ_i .

Представим неизвестное поле $\{\vec{E}_i, \vec{H}_i\}$ внутри диэлектрического цилиндра в виде суммы полей введенных вспомогательных диполей:

$$\vec{E}_i(M) = \frac{i}{\omega \epsilon_i} \sum_{n=1}^{N_i} \text{rot rot } \vec{\Pi}_{n,i}, \quad \vec{H}_i(M) = \sum_{n=1}^{N_i} \text{rot } \vec{\Pi}_{n,i}, \quad (7)$$

$$\vec{\Pi}_{n,i} = \Psi_i(M, M_{n,i}) \vec{p}_{\tau}^{n,i}, \quad \vec{p}_{\tau}^{n,i} = p_{\tau_1}^{n,i} \vec{e}_{\tau_1}^{n,i} + p_{\tau_2}^{n,i} \vec{e}_{\tau_2}^{n,i}, \quad M \in D_i.$$

В выражениях (7) функция $\Psi_i(M, M_{n,i}) = \exp(ik_i R_{MM_{n,i}}) / 4\pi R_{MM_{n,i}}$, $R_{MM_{n,i}}$ – расстояние от точки $M_{n,i}$ на вспомогательной поверхности S_i до точки M внутри диэлектрического цилиндра; $k_i = \omega\sqrt{\varepsilon_i\mu_i}$; N_i – число точек размещения на S_i ; $p_{\tau_1}^{n,i}$, $p_{\tau_2}^{n,i}$ ($n=1,2,\dots,N_i$) – неизвестные дипольные моменты.

Представления (6) удовлетворяют уравнениям Максвелла (1) для внешней области D_e и условиям излучения (5), а представления (7) – уравнениям Максвелла (2) для области D_i . Для того, чтобы удовлетворить граничным условиям (3)–(4), необходимо соответствующим образом выбрать дипольные моменты $p_{\tau_1}^{n,i}$, $p_{\tau_2}^{n,i}$ ($n=1,2,\dots,N_i$) и распределения осевых токов \vec{J}_d^e , \vec{J}_d^m и \vec{J}_p^e .

Предварительно введём кусочно-постоянную аппроксимацию вспомогательных осевых токов. Разобьём осевую линию диэлектрического цилиндра на N_d участков, в пределах каждого из которых величины токов \vec{J}_d^e и \vec{J}_d^m можно считать постоянными, а осевую линию идеально проводящего цилиндра разобьём на N_p участков, в пределах каждого из которых величину тока \vec{J}_p^e также можно считать постоянной. Тогда выражения для $\vec{\Pi}_d^e$, $\vec{\Pi}_d^m$ и $\vec{\Pi}_p^e$ в (6) можно представить в виде

$$\vec{\Pi}_d^e = \vec{e}_z \sum_{i=1}^{N_d} J_{d,i}^e \int_{z_{i-1}}^{z_i} \frac{\exp(ik_e R_{MM'})}{4\pi R_{MM'}} dz', \quad \vec{\Pi}_d^m = \vec{e}_z \sum_{i=1}^{N_d} J_{d,i}^m \int_{z_{i-1}}^{z_i} \frac{\exp(ik_e R_{MM'})}{4\pi R_{MM'}} dz',$$

$$\vec{\Pi}_p^e = \vec{e}_z \sum_{i=1}^{N_p} J_{p,i}^e \int_{z'_{p,i-1}}^{z'_{p,i}} \frac{\exp(ik_e \bar{R}_{MM'})}{4\pi \bar{R}_{MM'}} dz'_{p,i}$$

где $J_{d,i}^e$ и $J_{d,i}^m$ – элементы электрического и магнитного токов на i -м участке осевой линии диэлектрического цилиндра; $J_{p,i}^e$ – элементы электрического тока на i -м участке осевой линии идеально проводящего цилиндра; \vec{e}_z – единичный вектор, направленный вдоль оси z . При таком подходе нахождение неизвестных распределений осевых токов сводится к нахождению значений $(2N_d + N_p)$ элементов тока.

Для определения дипольных моментов и элементов токов используем граничные условия (3)–(4), удовлетворяя им в соответствии с методом коллокаций. Предварительно учтём, что для идеально проводящего тонкого цилиндра, в силу малости его диаметра по сравнению с длиной волны возбуждающего поля, вкладом в рассеянное поле азимутальной составляющей тока на поверхности цилиндра можно пренебречь.

Пусть M_{j_d} – точки коллокации на поверхности S_d диэлектрического цилиндра, а M_{j_p} – точки коллокации на поверхности S_p идеально проводящего цилиндра. Пусть число точек коллокации на поверхности S_d равно L_d , а число точек коллокации на поверхности S_p равно L_p . Тогда для нахождения неизвестных дипольных моментов $p_{\tau_1}^{n,i}$, $p_{\tau_2}^{n,i}$ ($n=1,2,\dots,N_i$) и элементов токов $J_{d,i}^e$, $J_{d,i}^m$ ($i=1,2,\dots,N_i$), $J_{p,i}^e$ ($i=1,2,\dots,N_p$) получим следующую систему линейных алгебраических уравнений:

$$\left[\vec{n}_i^{j_d}, (\vec{E}_i^{j_d} - \vec{E}_e^{j_d}) \right] = \left[\vec{n}_i^{j_d}, \vec{E}_0^{j_d} \right], \quad \left[\vec{n}_i^{j_d}, (\vec{H}_i^{j_d} - \vec{H}_e^{j_d}) \right] = \left[\vec{n}_i^{j_d}, \vec{H}_0^{j_d} \right], \quad j_d = 1, 2, \dots, L_d, \quad (8)$$

$$E_{e,z}^{j_p} = -E_{0,z}^{j_p}, \quad j_p = 1, 2, \dots, L_p$$

где $\vec{n}_i^{j_d}$ – единичный вектор нормали в точках коллокации M_{j_d} на поверхности диэлектрического цилиндра; $\vec{E}_e^{j_d}, \vec{H}_e^{j_d}, \vec{E}_i^{j_d}, \vec{H}_i^{j_d}$ и $\vec{E}_0^{j_d}, \vec{H}_0^{j_d}$ – компоненты внешнего, внутрен-

него и возбуждающего полей в этих же точках; $\vec{E}_{e,z}^{j_p}, \vec{E}_{0,z}^{j_p}$ – ориентированные вдоль оси z составляющие рассеянного и возбуждающего полей в точках коллокации M_{j_p} на поверхности идеально проводящего цилиндра.

После решения системы (8) – определения неизвестных дипольных моментов $p_{\tau_1}^{n,i}$, $p_{\tau_2}^{n,i}$ ($n=1,2,\dots,N_i$) и элементов токов $J_{d,i}^e, J_{d,i}^m$ ($i=1,2,\dots,N_i$), $J_{p,i}^e$ ($i=1,2,\dots,N_p$) – из представлений (6) получаем выражения для компонент рассеянного поля в дальней зоне, которые имеют следующий вид:

$$\begin{aligned} E_{e,\theta}(M) &= \sqrt{\frac{\mu_e}{\epsilon_e}} H_{e,\varphi}(M) = \frac{e^{ik_e R}}{R} D_\theta(\theta, \varphi) + O(R^{-2}), \\ E_{e,\varphi}(M) &= -\sqrt{\frac{\mu_e}{\epsilon_e}} H_{e,\theta}(M) = \frac{e^{ik_e R}}{R} D_\varphi(\theta) + O(R^{-2}), \end{aligned} \quad (9)$$

где компоненты диаграммы рассеяния $D_\theta(\theta, \varphi)$ и $D_\varphi(\theta)$ определяются выражениями

$$\begin{aligned} D_\theta(\theta, \varphi) &= -\frac{i\omega\mu_e}{4\pi} \sin\theta \left[\sum_{i=1}^{N_d} J_{d,i}^e \int_{z'_{i-1}}^{z'_i} e^{-ik_e z' \cos\theta} dz' + e^{-ik_e(x_0 \sin\theta \cos\varphi + y_0 \sin\theta \sin\varphi)} \sum_{i=1}^{N_p} J_{p,i}^e \int_{z'_{p,i-1}}^{z'_{p,i}} e^{-ik_e z' \cos\theta} dz' \right], \\ D_\varphi(\theta) &= \frac{ik_e}{4\pi} \sin\theta \sum_{i=1}^{N_d} J_{d,i}^m \int_{z'_{i-1}}^{z'_i} e^{-ik_e z' \cos\theta} dz'. \end{aligned} \quad (10)$$

Заключение

В настоящей работе получено решение задачи электромагнитного рассеяния на тонких параллельных идеально проводящем и диэлектрическом цилиндрах конечной длины. Оно определяется выражениями (9)–(10), в которых элементы токов $J_{d,i}^e, J_{d,i}^m$ ($i=1,2,\dots,N_i$) и $J_{p,i}^e$ ($i=1,2,\dots,N_p$) определяются путём решения системы линейных алгебраических уравнений (8).

ЛИТЕРАТУРА

1. Tavis M.T. Total radar cross section of thin wires at all angles of incidence // Journal of Applied Physics. – 1975. – V. 46. – № 7. – P. 3213–3215.
2. Hatanzadeh-Varmazyar S. New numerical method for determining the scattered electromagnetic fields from thin wires // Progress in Electromagnetic Research. – 2008. V. 3. – P.207–218.
3. Bogerd J.C., Tijhuis A.G., Klaasen J.J.A. Electromagnetic excitation of a thin wire: a travelling – wave approach // IEEE Transactions on Antennas and Propagation. – 1998. – V. 46. – № 8. – P. 1202–1211.
4. Yan W.Z., Du Y., Wu H., Liu D.W. EM scattering from a long dielectric circular cylinder // Progress in Electromagnetics Research, PIER 85. – 2008. – P. 39–67.
5. Santalla del Rio V., Abalde-Lima L., Christodoulou C.G. Electromagnetic scattering from vegetation cylindrical components // IEEE Geoscience and Remote Sensing Letters. – 2015. – V. 12. – № 4. – P. 751–755.
6. Sarabandi K., Sencior T.B.A. Low-frequency scattering from cylindrical structures at oblique incidence // IEEE Transactions on GRS. – 1990. – V. 28. – № 5. – P. 879–885.
7. Дмитренко А.Г., Келлер Ю.А. Компьютерное моделирование электромагнитного рассеяния на структурах из нескольких тонких проводников // Вестник Томского государственного университета. Серия «Математика. Кибернетика. Информатика». – 2006. – № 290. – С. 150–157.
8. Дмитренко А.Г., Гольцварт Е.П. Решение задачи электромагнитного рассеяния на тонком диэлектрическом цилиндре методом вспомогательных источников // Радиотехника и электроника. – 2011. Т. 56. – № 5. С. 600–607.
9. Дмитренко А.Г., Гольцварт Е.П. Рассеяние электромагнитной волны на структуре, состоящей из нескольких тонких диэлектрических цилиндров с конечными длинами // Известия вузов. Радиофизика. – 2014. Т. 57. – № 3. – С. 218–230.
10. Дмитренко А.Г., Колчин В.А. Численный метод исследования электромагнитного рассеяния структурами, содержащими тонкие проводники // Радиотехника и электроника. – 2003. – Т. 48. – № 5. – С. 545–551.
11. Дмитренко А.Г., Мукомолов А.И. Численный метод решения задач электромагнитного рассеяния на трехмерном магнитодиэлектрическом теле произвольной формы // Радиотехника и электроника. – 1995. – Т. 40. № 6. – С. 875–880.

О СИСТЕМЕ УПРАВЛЕНИЯ АУДИОМЕТРОМ С РЕАЛИЗАЦИЕЙ МЕТОДИКИ ТОНАЛЬНОЙ АУДИОМЕТРИИ И ОБЪЕКТИВНОЙ МЕТОДИКИ ASSR

Е.И. Зубова

*Томский государственный университет
vixert315@gmail.com*

Введение

Аудиометрией называется процедура проверки остроты слуха. Выполняется она при помощи специального электроакустического прибора – аудиометра. Этот прибор позволяет оценить пороги восприятия звука в диапазоне частот от 125 Гц до 8 кГц, построить аудиограмму и при необходимости настроить слуховой аппарат на каждой из октавных частот.

Актуальность разработки аудиометров обоснована высокой заболеваемостью слухового анализатора как у людей старшего поколения, так и детей. Так, согласно данным сайта Федеральной службы государственной статистики России в промежутке с 2000 по 2014 гг. среднее количество лиц, впервые признанных инвалидами с болезнями уха и сосцевидного отростка, ежегодно составляет 9.5 тысяч человек [1]. Диагностика – первый и самый ответственный этап процесса восстановления здоровья человека, поскольку на ранней стадии заболевания наиболее вероятен успех терапии.

На сегодняшний день аудиометрия широко используется для оценки состояния слуха пациентов в поликлиниках, амбулаториях, медпунктах, МСЧ, приемных отделениях больниц и госпиталей, сурдокабинетах и сурдоцентрах.

1. Постановка задачи

Все методы исследования слуха можно разделить на субъективные и объективные. Субъективные методы – это методы, требующие непосредственного осознанного участия пациента. Эта группа включает в себя тональную, речевую и надпороговую аудиометрию. Тональная пороговая аудиометрия позволяет оценить степень тугоухости и правильно настроить слуховой аппарат под каждую октавную частоту; с помощью речевой аудиометрии оценивают процент разборчивости слов; надпороговая аудиометрия помогает определить место поражения слухового анализатора.

Объективные методы применяются в случае, когда пациент не может давать четких ответов. Объективные методы исследования слуха – это, в основном, компьютерная аудиометрия (ASSR и КСВП тесты). Следует отметить, что объективные методы являются более точными.

В ходе работы были рассмотрены различные типы аудиометров, чтобы выяснить современный уровень реализации этих приборов. Основными производителями аудиометров являются компании Maico (Германия), Interacoustics (Дания), GSI (США), Нейрософт (Россия). Многие производители медицинской электроники идут по пути уменьшения массы и габаритов приборов, создавая их портативными. Хорошим примером являются аудиометры, которые производит компания MedRx (USA). Кроме того, большое значение в последние годы получила автоматизация процесса снятия аудиограммы, потому что она экономит время пациента и не требует дополнительных приспособлений. Также компании нацелены на производство функциональных приборов с большим количеством реализуемых тестов. Но необходимо учитывать, что стоимость такого прибора высока и не всегда требуется совмещения такого количества методик одновременно, выбор аудиометра всегда должен соответствовать конкретным поставленным задачам.

Таким образом, было принято решение разработать аудиометр, соответствующий основным тенденциям в проектировании аудиометров, но с ограниченным числом ба-

зовых методик, что позволит сохранить основное назначение прибора – диагностировать нарушение слуха, но при этом существенно сократится его стоимость. Для реализации была выбрана методика тональной аудиометрии, поскольку она является базовой методикой и используется во всех сложных приборах для исследования слухового анализатора.

Одним из способов уменьшения стоимости прибора является применение специально подобранных наушников. При проведении аудиометрических исследований необходимо использование профессиональных наушников TDH39 или HDA280. Данные наушники имеют равномерную амплитудно-частотную характеристику в широком диапазоне частот. Обычные наушники, как правило, имеют в этом диапазоне частот завалы из-за возникновения любого воздушного зазора между ухом и наушником. Подробный обзор наушников для аудиометрии и их характеристики представлен в [Ошибка! Источник ссылки не найден.]. В работе ставится задача повысить качество звукопроводения у обычных наушников за счет корректировки АЧХ. Она производится путем регулирования входного сигнала наушников. Таким образом, использование не профессиональных наушников позволяет сократить стоимость прибора, но при этом сохранить качество звукопроводения.

2. Методика тональной аудиометрии

2.1. Описание методики

Тональная аудиометрия служит для определения слуховых порогов на разных частотах. Как правило, частотный диапазон составляет 125 – 8000 Гц. Этот диапазон соответствует речевым частотам, поэтому он имеет первостепенное значение в исследовании слуха. Когда определяются пороги по воздушному звукопроводению, пациенту предъявляют чистый тон одной частоты (обычно начинают с частоты 1000 Гц) при каждом исследовании. Начальная интенсивность должна быть легко идентифицируемой для испытуемого.

Постепенно уровень интенсивности тона снижается с шагом в 10 дБ до исчезновения его восприятия. Затем уровень интенсивности стимула повышается с шагом в 5 дБ до возникновения слухового ощущения. Для более точного определения порогов эти операции повторяются.

В аудиометрии по костной проводимости тестовый сигнал воспроизводят воздействием костного осциллятора на сосцевидный отросток или на лобную кость испытуемого.

Значения порогов наносятся на бланк аудиограммы. По горизонтали фиксируется частота тона в герцах. По вертикали – интенсивность тона в децибелах относительно средних нормальных порогов слуха, принятых за ноль. Кривые воздушного звукопроводения изображают сплошной линией, кривые костного проведения – пунктиром. Пороги правого уха обычно обозначают точками, левого – крестиками.

Аудиограмма дает фундаментальное описание слуховой чувствительности. Аудиограмма позволяет определить степень тугоухости и тип патологии: сенсоневральный, кондуктивный или смешанный [3].

2.2. Структурная схема устройства

После анализа основной функциональности тонального аудиометра и обзора структурных схем, предлагаемых производителями этих устройств (например, структурная схема аудиометра-тимпанометра GSI 38) были выявлены основные блоки, входящие в их состав. Таковыми являются генератор синусоидальных сигналов, усилитель, аттенуатор, наушники, костный осциллятор, кнопка ответа пациента. Также на сегодняшний день актуально проектировать медицинские приборы, имеющие связь с ПК.

Для реализации была предложена структурная схема, показанная на рис. 1.

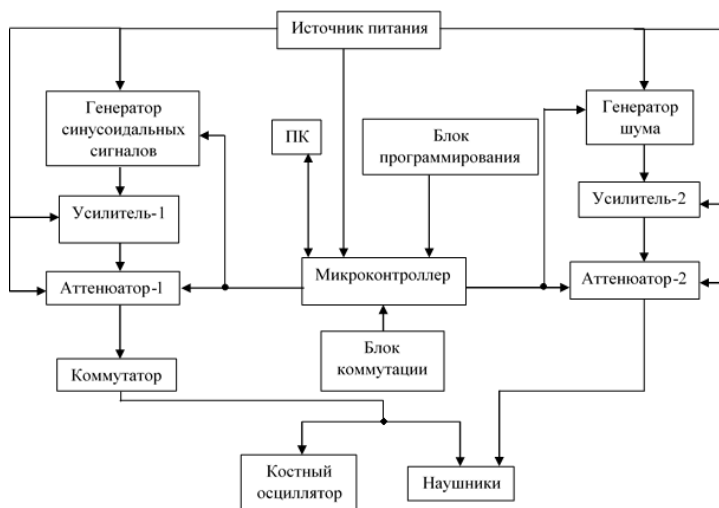


Рис. 1. Полная структурная схема разрабатываемого аудиометра

Назначение блоков заключается в следующем. Генератор синусоидальных сигналов формирует на выходе напряжение следующих частот: 125 Гц, 250 Гц, 500 Гц, 1000 Гц, 2000 Гц, 4000 Гц и 8000 Гц (чистые тоны). Управление генератора осуществляется микроконтроллером. Пройдя через усилитель-1, сигнал регулируется аттенуатором-1 и через коммутатор попадает либо на наушники, либо на костный осциллятор. Наушники применяются для проведения воздушной аудиометрии, костный осциллятор – для костной аудиометрии. Маскирующий белый шум, получаемый с помощью генератора шума, также усиливается и регулируется усилителем-2 и аттенуатором-2, но поступает только на наушники. Аттенуаторы управляются при помощи микроконтроллера. Кроме того, устройство предполагает двухстороннюю связь микроконтроллера с компьютером для визуализации аудиограммы. Для ответов пациента предусмотрена кнопка (блок коммутации). Присутствует блок программирования микроконтроллера. Питательное напряжение поступает на генераторы, усилители, аттенуаторы и микроконтроллер от источника питания. В качестве микроконтроллера была выбрана микросхема 32-битного ARM-микроконтроллера STM32F407VGT6 семейства Cortex-M4.

2.3. Результаты

В ходе работы был разработан макет тонального аудиометра. Для макета аудиометра была разведена и изготовлена печатная плата, выполнена пайка компонентов на плату. Управляющим элементом схемы является микроконтроллер STM32F407VGT6 в составе платы STM32F4-Discovery.

Программа микроконтроллера реализует: вывод в ЦАП коэффициентов, формирующих синусоидальный сигнал; двухстороннюю связь с компьютером через UART; управление цифровым резистором по SPI; коммутация управляющих входов аналогового ключа. Интерфейс (рис. 2) позволяет пользователю заполнять данные о пациенте, выполнять исследование (выбирать исследуемое ухо, задавать частоту и уровень звукового сигнала), получить на выходе аудиограммы и расчет степени тугоухости пациента. Интерфейс и программа для микроконтроллера были написаны на языках программирования C++ и C соответственно.

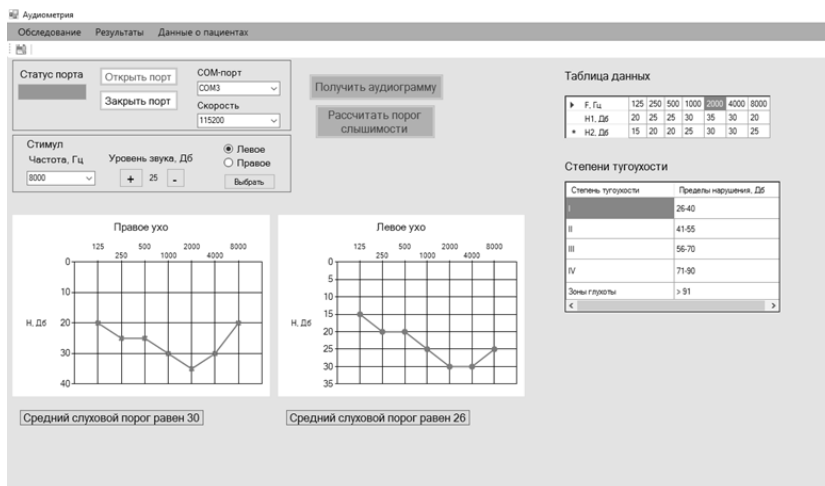


Рис. 2. Интерфейс программы для управления тональным аудиометром

В результате проведенных экспериментов были сняты осциллограммы, иллюстрирующие работу схемы. Было выявлено, что частоты и уровни сигнала изменяют корректно в соответствии с программой микроконтроллера. Однако уровни напряжения необходимо пронормировать, исходя из величины звукового давления, получаемого с наушников и выраженного в дБ.

Поскольку тональная аудиометрия – это субъективная методика, требующая непосредственного осознанного участия пациента, ее нельзя использовать для диагностирования слуха у пациентов, которые сами не могут давать четкие ответы, например, у новорожденных. К тому же, распространенной практикой является проведение разных аудиометрических тестов для подтверждения полученного результата. В связи с этим, было принято решение рассмотреть возможность реализации объективной методики аудиометрии в дополнение к тональной аудиометрии.

3. Рассмотрение способов реализации ASSR методики

Объективная методика AuditorySteady – StateResponse (ASSR) – слуховой ответ на стабильно модулированный тон. Отличительной особенностью этой методики является то, что пациенту подается модулированный сигнал. Модуляция может быть частотная (FM), амплитудная (AM), частотно-амплитудная (MM), а также по амплитуде, возрастающей по экспоненте (AM2).

Стимул, используемый в этом электрофизиологическом исследовании, имеет ряд преимуществ: модулированный тон в значительно меньшей степени подвержен спектральным искажениям, в отличие от акустического щелчка или короткой тональной посылки и имеет достаточную частотную специфичность [4,5]. Преимущество ASSR и в том, что этот метод позволяет обследовать пациентов с остаточным слухом, так как стимул можно предъявлять на уровнях вплоть до 120 дБ нПЧ [7].

Ответы, записанные с кожи головы на непрерывные или на постоянно модулированные (steady-state) тоны, также являются постоянными, т. е. существуют в течение всего времени предъявления стимула, и называются стационарными вызванными потенциалами (steady-state evoked potentials SSEP) или ответами, вызванными амплитудной модуляцией (amplitude-modulation following responses AMFR). Поскольку аббревиатура SSEP может быть перепутана с соматосенсорными вызванными потенциалами (somatosensory evoked potential), был принят термин «Auditory Steady-State Responses» (ASSR) [6].

На данный момент существуют два разных алгоритма анализа ASSR: поличастотный и моночастотные типы теста.

Проведение моночастотного ASSR-теста: во время действия постоянного модулированного тона регистрируется ЭЭГ. В методике используется алгоритм математического обнаружения порогов слуха по ЭЭГ: анализ Фурье на частоте модуляции стимула для получения информации об амплитуде и фазе ответа [8]. Проводится спектральный анализ гармоник и, когда достигается заранее установленная величина вероятности, алгоритм анализа автоматически останавливает пробу и представляет результат, либо останавливает пробу после 64 пробегов, когда статистически значимая величина не достигается.

Нужно отметить, что при проведении моночастотного ASSR-теста вызванный потенциал имеет низкую амплитуду вследствие чего его можно записать только при оптимальном соотношении сигнал/шум. Увеличение амплитуды ответа мы можем ожидать при расширении зоны возбуждения базилярной мембраны улитки, для чего был предложен метод стимуляции сигналом, включающий в себя звуковой спектр, состоящий из нескольких частот.

Проведение поличастотного ASSR-теста: для статистической обработки ответа используется не только первая гармоника, как в случае с моночастотной стимуляцией, но и нескольких более высоких, так как на них перераспределяется значительная часть силы ответа, в связи с тем, что возросла частота стимуляции и сам стимул уже носит несинусоидальный характер. Характер статистической обработки также отличается от моночастотного теста, в данном случае проводится дисперсионный анализ между силой предъявляемого сигнала и компонентами шума на ЭЭГ [9].

подавая стимул в виде поличастотного сигнала, анализ проводится одновременно, но для каждой частоты отдельно. Необходимо достичь 95% уровня устойчивости (дисперсионный анализ проводится компьютером в режиме реального времени) для того чтобы определить действительно ли присутствует ответ на отдельно взятой частоте.

Применяемый компьютерный алгоритм используется в данном виде диагностики с целью большей объективизации оценки результатов по сравнению с записью вызванных потенциалов мозга, при которых сам клиницист выделяет пики и межпиковые интервалы.

Подытожим преимущества ASSR-теста:

1. Определение значительных потерь слуха.
2. Получение точных аудиометрических данных, с дальнейшей возможностью использования их для какой-либо коррекции (настройки слухового аппарата, имплантации костной проводимости и т.д.)
3. Возможность определения потери слуха уже в первые месяцы после рождения.
4. Возможность проведения лечения и других методов коррекции слуха в первые 6 месяцев после рождения.

Неудобством в применении ASSR является необходимость низкой ЭЭГ-активности для достижения оптимального соотношения сигнал/шум. Это означает, что пациент должен находиться в состоянии естественного сна.

По результатам метода ASSR строится аудиограмма, как и при тональной аудиометрии. Слух считается нормальным, если ASSR регистрируются при уровне входного стимула 35 дБ (над порогом слышимости). Отсутствие ASSR на этот стимул является основанием для проведения диагностического обследования.

3.1. F-тест

В тесте исследуется частота, соответствующая частоте модуляции сигнала. По F-критерию Фишера оценивается соотношение между мощностью сигнала на частоте модуляции и мощностью фонового шума ЭЭГ. Мощность фонового шума рассчитывается, как средняя мощность сигнала для заданного количества смежных частот (как правило, 60), равномерно распределенных вокруг частоты сигнала:

$$F = \frac{|FFT(\text{сигнал})|^2}{\text{avg}|FFT(\text{шум})|^2},$$

где $FFT(\text{сигнал})$ – мощность сигнала на тестируемой частоте, $FFT(\text{шум})$ – мощность сигнала на смежных частотах [10].

Если тестируемый компонент не попадает в такое же распределение вероятности, как остальные компоненты, гипотеза о наличии сигнала отклоняется.

Существуют различные возможные протоколы обнаружения с различными процентными уровнями значимости F -теста (p) с использованием фиксированного или переменного времени тестирования.

При статистическом тестировании может возникать два типа ошибок: тип I – ложное обнаружение ответа (ложный положительный результат), тип II – не обнаруживает разницу, которая присутствует (ложный отрицательный результат). Вероятность возникновения ошибок типа I и типа II зависит от выбранного значения p . Ожидается, что при обнаружении ответа с использованием F -критерия для заданного значения $p < 0,05$ ложный положительный результат появится в 5% случаях [11]. Однако процент ложных срабатываний может увеличиваться при использовании переменных протоколов времени тестирования, которые прекращают тестирование после выполнения требуемого p -критерия. Повышенная частота ошибок возникает, поскольку переменные протоколы времени тестирования повторно проверяют данные ЭЭГ после каждого дополнительного последовательного сегмента времени записи (развертки). Повторное тестирование увеличивает вероятность ложного отклонения нулевой гипотезы.

Чтобы уменьшить вероятность ложных срабатываний, можно применять корректировку p -уровня в тесте. Но при этом необходимо учитывать, что слишком низкий p -уровень может предотвратить обнаружение истинного ответа.

3.2. Tcirc-тест

В тесте проверяется нулевая гипотеза об отсутствии сигнала путем сравнения распределения повторных измерений на частоте модуляции сигнала и теоретического распределения.

Обозначим среднее значение M -оценок компонентов Фурье z_1, z_2, \dots, z_M как $z_{\text{est}} = \sum \frac{z_i}{M}$. Предположим, что гипотетическое среднее значение равно ξ . ξ, z_{est}, z_j – комплексные величины, следовательно, существует две независимые оценки дисперсии совокупности V вещественной и мнимой частей с $2(M-1)$ степенями свободы:

$$V_{\text{indiv}} = \frac{1}{2(M-1)} \sum |z_j - z_{\text{est}}|^2, \quad V_{\text{group}} = \frac{M}{2} \sum |z_{\text{est}} - \xi|^2.$$

V_{group} и V_{indiv} являются оценками, полученными из независимых величин, при условии, что данные z_j являются образцами совокупности, среднее значение которой равно ξ . Поэтому отношение $V_{\text{group}}/V_{\text{indiv}}$ распределяется согласно распределению F . T_{circ}^2 оп-

ределяется как $T_{\text{circ}}^2 = \frac{1}{M} \frac{V_{\text{group}}}{V_{\text{indiv}}}$. MT_{circ}^2 распределяется в соответствии с распределением

$F_{[2, 2M-2]}$. Гипотеза об отсутствии сигнала отклоняется, если достигается заданный уровень значимости.

Преимуществом Tcirc теста является то, что в нем оценивается информация только из одного частотного канала. Так обеспечивается независимость от соседних каналов, которые могут содержать шум. Отсутствие информации из соседних каналов компенсируют последующие измерения. Следовательно, Tcirc тест первоначально показывает ложные положительные результаты. После одного или нескольких измерений, тестовые

ответы являются надежными [10]. Если тестовый сигнал содержит фазовые изменения, то тест Tsіgс показывает сильный разброс в уровнях значимости.

Таким образом, преимуществом F-теста является высокая надежность сразу после начала тестирования, тогда как Tsіgс-тест первоначально дает ложные положительные результаты. Также F-тест не чувствителен к фазовым изменениям, что положительно сказывается на итоговом результате. Однако Tsіgс-тест наименее чувствителен к шумам, так как использует информацию из одного частотного канала.

Заключение

Мы рассмотрели наиболее распространенные методики аудиометрии. Каждая методика занимает свою нишу: тональная пороговая аудиометрия позволяет оценить степень тугоухости и правильно настроить слуховой аппарат под каждую октавную частоту; с помощью речевой аудиометрии оценивают процент разборчивости слов; надпороговая аудиометрия помогает определить место поражения слухового анализатора; объективные методики являются наиболее точными, а также применяются в случае, когда пациент не может давать четких ответов. В разработке аудиометра реализована методика тональной аудиометрии, поскольку она является базовой и планируется реализация объективной методики ASSR.

ЛИТЕРАТУРА

1. Федеральная государственная служба статистики. Численность лиц, впервые признанных инвалидами. [Электронный ресурс]. – Режим доступа: http://www.gks.ru/wps/wcm/connect/rosstat_main/rosstat/ru/statistics/population/healthcare/, свободный (дата обращения: 09.10.2016).
2. Zubova E.I., Kapul A.A., Torgaev S.N. Pure-tone audiometer // Journal of Physics: Conference series. – 2017.
3. Альтман Я.А., Таварткиладзе Г.А. Руководство по аудиологии. – М.: ДМК Пресс, 2003. – 360 с.
4. G. Rance. et.al. Hearing Threshold Estimation in Infants Using Auditory Steady-State Responses // Journal of the American Academy of Audiology. – 2005. – V. 16. – № 5. – P. 291–300.
5. Rance G., Rickards F. Prediction of Hearing Threshold in Infants Using Auditory Steady State Evoked Potentials // Journal of the American Academy of Audiology. – 2002. – V. 13. – № 5. – P. 236–245.
6. Dimitrijevic A., John M.S., Picton T.R. Auditory Steady-State Responses in Normal-Hearing and Hearing-Impaired Adults // Ear and Hearing. – 2004. – V. 25. – № 1. – P. 68–84.
7. Levin S. Use Auditory Evoked Potentials in Modern Audiological Research. – St. Petersburg, 2009. – P. 21.
8. Korczak P., Smart J., Delgado R., Strobel T., Bradford C. Auditory Steady-State Responses // Journal of the American Academy of Audiology. – 2012. – 23. – P. 146–170.
9. Plourde G., Stapells D.R., Picton T.W. The human auditory steady-state evoked potentials // Acta Otolaryngologica (Stockh.). – 1991. – V. 491. – № 6. – P. 153–160.
10. Bahmer A., Baumann U. Recording and online analysis of auditory steady state responses (ASSR) in Matlab // Journal of Neuroscience Methods. – 2010. – V. 187. – P. 105–113.
11. Wilding T.S., McKay C.M., Baker R.J., Kluk K. Auditory steady state responses in normal-hearing and hearing-impaired adults: an analysis of between-session amplitude and latency repeatability, test time, and F-ratio detection paradigms // The Journal of the American Auditory Society Ear and Hearing. – 2012. – P. 267–278.

РАЗРАБОТКА ЭКСПЕРТНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ КОНТРОЛЯ ПАССАЖИРОПЕРЕВОЗОК

А.И. Павлова, Э.Р. Добров

*Новосибирский государственный университет экономики и управления
annstab@mail.ru, dobrov.eldar@gmail.com*

Введение

На ГПТ назрела необходимость реализации комплексной реформы, охватывающей все уровни управления, от федерального до муниципального, и все аспекты функционирования – от правового до технологического. Основные проблемы и противоречия в развитии ГПТ, требующие решения на общероссийском уровне, могут быть сконцентрированы в 4 основных блоках:

- правовые основы деятельности;

- организация и финансирование операционной деятельности;
- инвестиции в ГПТ;
- институциональные вопросы, организация управления.

Объектом исследования в данном проекте является городской пассажирский транспорт, а предметом выбрана безналичная система проезда на городском транспорте

Осуществить решение проблем и противоречия в развитии ГПТ можно, внедрив экспертную систему, которая позволит обеспечивать контроль объема предоставленных услуг, создать единый подход к формированию тарифов и мотивацию снижения затрат.

Таким образом, в данной работе ставится цель – разработать экспертную информационную систему для контроля пассажироперевозок.

Для выполнения целей проекта были поставлены следующие задачи:

- выбор входных и выходных переменных экспертной информационной системы для контроля пассажироперевозок, построение функции принадлежности;
- построение базы правил и системы нечеткого логического вывода;
- разработка программы экспертной информационной системы для контроля пассажироперевозок.

Средством реализации проекта – экспертной информационной системы для контроля пассажироперевозок – является язык C# в среде Microsoft Visual Studio 2010.

1. Выбор входных и выходных переменных, построение функции принадлежности

Описание входных данных с помощью векторов

лингвистических переменных

1. Данные о длине поездки

Название переменной: ω = “длина поездки”.

Терм-множество значений: T_1 = “низкая”, T_2 = ”средняя”, T_3 = ”высокая”.

Носитель: U = промежуток от 0% до 100%.

Синтаксическое правило: Уровень длины поездки.

Семантическое правило: определяется функциями принадлежности, для значения

$T_1 - \mu_1(U)$, для $T_2 - \mu_2(U)$, для $T_3 - \mu_3(U)$, они изображены на рис. 1, причем первая из них отвечает нечеткому подмножеству M_1 , вторая – M_2 , третья – M_3 .

2. Данные о числе поездок

Название переменной: ω = “число поездок”.

Терм-множество значений: T_1 = “мало”, T_2 = ”средне”, T_3 = ”много”.

Носитель: U = промежуток от 0 до 1000, целые числа.

Синтаксическое правило: уровень числа поездок.

Семантическое правило: определяется функциями принадлежности, для значения

$T_1 - \mu_1(U)$, для $T_2 - \mu_2(U)$, для $T_3 - \mu_3(U)$, они изображены на рис. 2, причем первая из них отвечает нечеткому подмножеству M_1 , вторая – M_2 , третья – M_3 .

Описание выходных данных с помощью векторов

лингвистических переменных

1. Размер скидки

Название переменной: ω = “размер скидки”.

Терм-множество значений: T_1 = “низкий”, T_2 = ”средний”, T_3 = ”высокий”.

Носитель: U = промежуток от 0 до 10, целые числа.

Синтаксическое правило: определение размера скидки на стоимость проезда.

Семантическое правило: определяется функциями принадлежности, для значения

$T_1 - \mu_1(U)$, для $T_2 - \mu_2(U)$, для $T_3 - \mu_3(U)$, они изображены на рис. 3, причем первая из них отвечает нечеткому подмножеству M_1 , вторая – M_2 , третья – M_3 .

Построение графиков для всех описанных лингвистических переменных

1. Данные о длине поездки



Рис. 1. График лингвистической переменной «Длина поездки»

2. Данные о числе поездок



Рис. 2. График лингвистической переменной «Число поездок»

3. Размер скидки

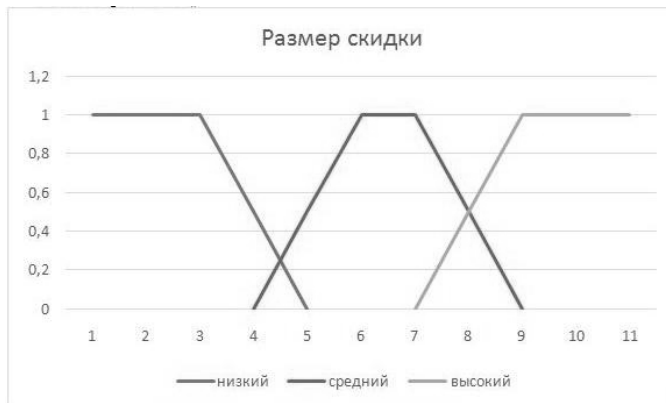


Рис. 3. График размера скидки на проезд

2. Построение базы правил и системы нечеткого логического вывода

Составление правила «Если-то»

1. Если длина поездки «менее 1 км» и число поездок «менее 5», то размер скидки «0%» (0,1).

2. Если длина поездки «менее 1 км» и число поездок «от 5 до 10», то размер скидки «0%» (0,2).
3. Если длина поездки «менее 1 км» и число поездок «больше 10», то размер скидки «0%» (0,3).
4. Если длина поездки «от 1 до 5 км» и число поездок «менее 5», то размер скидки «5%» (0,4).
5. Если длина поездки «от 1 до 5 км» и число поездок «от 1 до 10», то размер скидки «5%» (0,5).
6. Если длина поездки «от 1 до 5 км» и число поездок «больше 10», то размер скидки «5%» (0,6).
7. Если длина поездки «свыше 10 км» и число поездок «менее 5», то размер скидки «10%» (0,7).
8. Если длина поездки «свыше 10 км» и число поездок «от 1 до 10», то размер скидки «10%» (0,8).
9. Если длина поездки «свыше 10 км» и число поездок «больше 10», то размер скидки «10%» (0,9).

3. Разработка программы экспертной информационной системы для контроля пассажироперевозок.

В результате разработки, было создано приложение – экспертная информационная система для контроля пассажироперевозок на языке C# в среде Microsoft Visual Studio 2010. Программа имеет удобный интерфейс, который в полной мере понятен конечному абоненту. Масштаб разработки: 5 форм пользовательского интерфейса.

В режиме приобретения знаний общение с ЭС осуществляет (через посредничество инженера по знаниям) эксперт. В этом режиме эксперт, используя компонент приобретения знаний, наполняет систему знаниями, которые позволяют ЭС в режиме решения самостоятельно (без эксперта) решать задачи из проблемной области.

Для начала работы в основном интерфейсе программы необходимо обучить, внести утверждения и сохранить. В дальнейшем можно открывать (загружать) файл базы знаний. Программа готова к использованию.

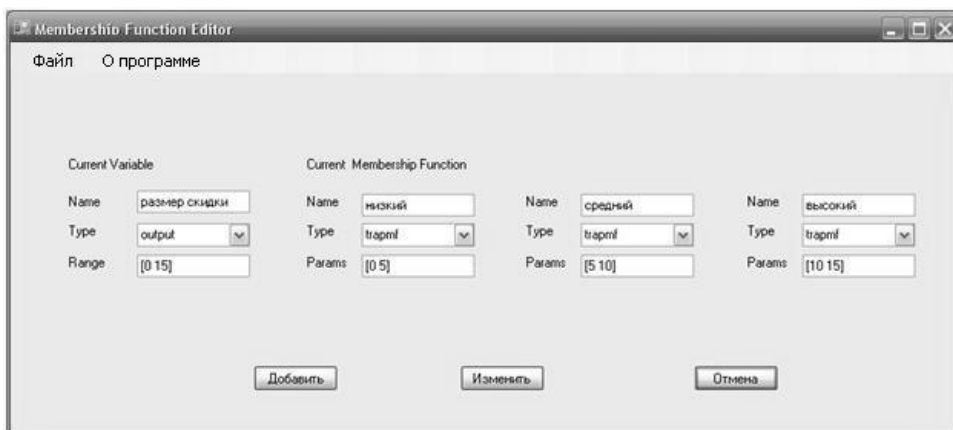


Рис. 4. Редактор функций принадлежности

Редактор функций принадлежности (рис. 4) – это инструмент, который позволяет отображать и редактировать все функции принадлежности, связанные со всеми входными и выходными переменными для всей системы нечеткого вывода. Редактор функций позволяет устанавливать функции принадлежности для переменной. Элементы управления, позволяют изменить имя, тип и параметры (форму) функции принадлежности после его выбора.

Ниже приведена информация о типе и имени текущей переменной. В этом регионе имеется текстовое поле, которое позволяет изменять пределы диапазона текущей переменной.

Теперь, когда переменные были названы, а функции принадлежности имеют соответствующие формы и имена, можно ввести правила.

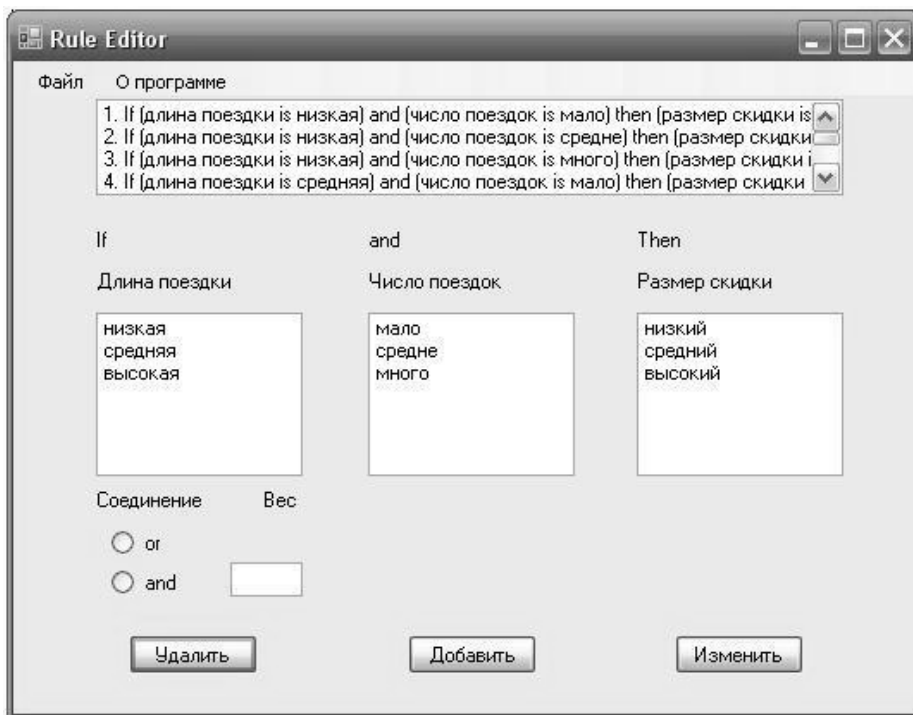


Рис. 5. Редактор правил

Редактор правил позволяет автоматически создавать инструкции правил, определяющих поведение системы.

Имеется возможность создать правила, выбрав элемент в каждом поле ввода и вывода, выбрав один элемент «Соединение» и нажав «Добавить». Нельзя выбрать ни одно из свойств переменной, чтобы исключить эту переменную из данного правила и не выбирать под каким-либо именем переменной, чтобы свести на нет соответствующее качество. Правило удаляется выбором правила и нажатием кнопки «Удалить». Редактирование правила осуществляется изменением выделения в поле переменной и нажатием кнопки «Изменить».

Вес правила указывается числом от 0 до 1. Если вес не указывается, предполагается, что он равен 1.

Результирующее правило: если (низкая длина поездки) и (число поездок мало), тогда (размер скидки низкий) (1). Цифры в круглых скобках обозначают веса.

Аналогичной процедуре необходимо следовать, чтобы вставить второй и последующие правила в редакторе правил.

Чтобы изменить правило, необходимо сначала нажать на правило, которое нужно изменить, затем внести необходимые изменения в это правило и нажать кнопку «Изменить».

В режиме консультации общение с ЭС осуществляет конечный пользователь, которого интересует результат и (или) способ его получения.

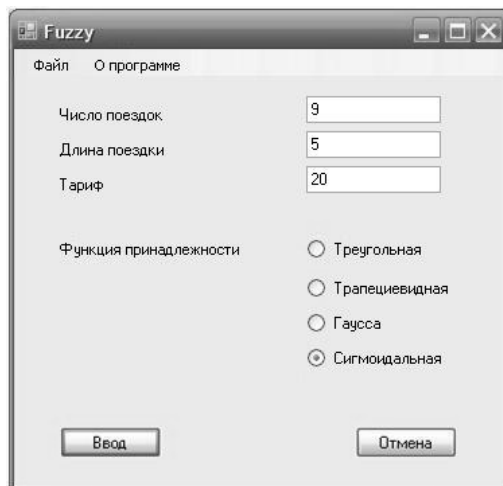


Рис. 6. Запрос пользователя

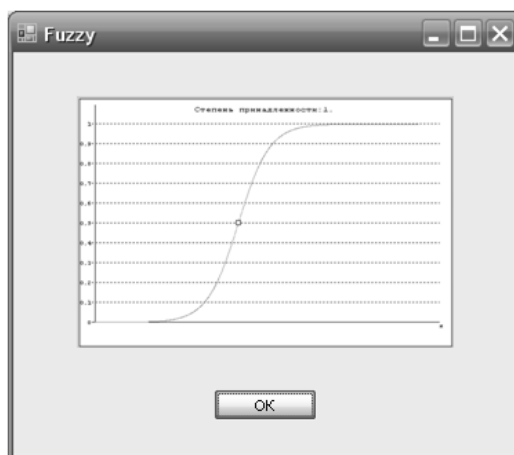


Рис. 7. Результат выполнения программы – сигмоидальная функция принадлежности

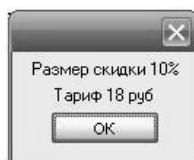


Рис. 8. Результат работы программы «размер скидки»

Заключение

В ходе выполнения проекта поставленные задачи были выполнены. Произведен выбор входных и выходных переменных экспертной информационной системы для контроля пассажироперевозок, построены функции принадлежности. Построена база правил и система нечеткого логического вывода. Разработана программа экспертная информационная система для контроля пассажироперевозок.

Данная работа дает пользователям предметной области ряд преимуществ: обеспечение контроля объема предоставленных услуг, создание единого подхода к формированию тарифов, создание мотивации снижения затрат.

ГЕНЕРАТОР ПСИХОДИАГНОСТИЧЕСКИХ ТЕСТОВ

Е.О. Скворцов, М.Н. Головчинер

Томский государственный университет

sk.zav@mail.ru, golovchiner@mail.ru

Введение

Современная психодиагностика всё шире используется в таких областях социальной практики, как расстановка кадров, профотбор и профориентация, прогнозирование социального поведения, оптимизация обучения и воспитания и т.д. Важной задачей психологов во всех перечисленных областях является выбор набора психологических тестов (методик), позволяющих провести соответствующие психодиагностические исследования изучаемого контингента респондентов.

В настоящей статье рассматривается структура и принципы функционирования разрабатываемой автоматизированной системы генерации совокупности тестов по описанию требований. Данная система может быть использована как для упрощения работы психологов, так и для обучения.

1. Постановка задачи

Основные задачи, решаемые системой – это предоставление возможных наборов методик, которые могут быть применены в соответствии с описанием требований, а также рекомендации по использованию оптимального по количеству набора методик для проведения исследования.

Очевидно, что описания требований, на основании которых производится подбор методик, включают необходимый набор личностных характеристик, которым должны соответствовать респонденты. В связи с этим разработка системы предполагает создание:

- информационной модели предметной области, в качестве которой выступает психологическая структура личности;
- банка психологических тестов;
- модуля генерации возможных наборов методик;
- совокупности интерфейсов.

2. Описание структуры

Общая структура системы и взаимодействие её компонентов представлены на рис. 1.

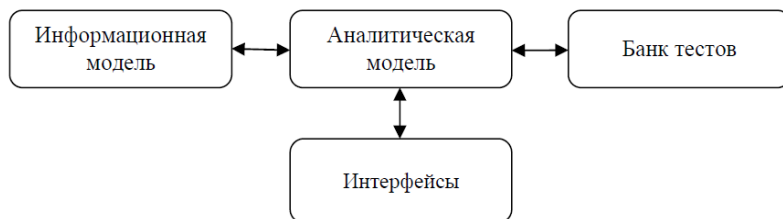


Рис. 1. Общая структурная схема генератора психодиагностических тестов

Банк тестов представляет собой комплекс методик, позволяющих диагностировать разнонаправленные свойства личности.

Информационная модель предметной области описывает основные формы проявления психики человека и их взаимосвязь.

Аналитическая модель выполняет следующие функции:

- генерирует возможные наборы тестов путём сопоставления связей информационный модели с методиками банка тестов в соответствии с полученным запросом;
- корректирует и расширяет информационную модель предметной области;
- пополняет банка тестов психодиагностическими методиками.

Пример запроса к системе

В организации для реализации государственного заказа необходимо определить руководителя проекта из числа кандидатов, который должен удовлетворять следующим характеристикам:

- требовательность;
- ответственность;
- принципиальность;
- ...

На основании полученного запроса система генерирует набор возможных методик, из которых психолог выбирает один или несколько тестов, которые и предлагаются кандидатам.

3. Информационная модель предметной области

Под психическими свойствами человека следует понимать устойчивые образования, обеспечивающие определённый качественно-количественный уровень деятельности и поведения, типичный для данного человека.

Свойства личности многообразны, и их нужно классифицировать в соответствии с группировкой психических процессов, на основе которых они формируются [1,3]. Психические свойства не сосуществуют вместе, они синтезируются и образуют сложные структурные образования личности, к которым необходимо отнести:

- жизненную позицию личности (систему потребностей, интересов, убеждений, идеалов, определяющую избирательность и уровень активности человека);
- темперамент (систему природных свойств личности – подвижность, уравновешенность поведения и тонус активности – характеризующую динамическую сторону поведения);
- способности (систему интеллектуально-волевых и эмоциональных свойств, определяющую творческие возможности личности);
- характер как систему отношений и способов поведения.

Фрагмент информационной модели психологической структуры личности представлен на рис. 2.

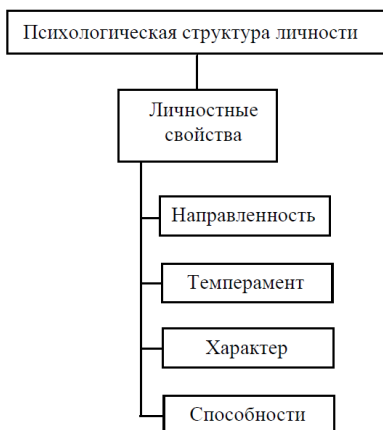


Рис. 2. Формы проявления личностных свойств человека

Информационная модель предметной области реализуется в виде семантической сети, в узлах которой располагаются компоненты психологической структуры личности, а связи (дуги) соответствуют возможным проявлениям черт (психологических, темперамента, характера и др.), связывающих эти компоненты. Например, в свойство направленности включаются следующие черты характера [1,2]:

Направленность на себя:

- самокритичность – некритичность;
- требовательность (к себе) – нетребовательность;
- скромность – хвастливость;
- другие.

Направленность на других:

- доброжелательность, дружелюбие – зависть;
- принципиальность — беспринципность;
- молчаливость – болтливость;
- другие.

Направленность на дело:

- трудолюбие — лень;
- ответственность – недобросовестность;
- усидчивость;
- другие.

На рис. 3. представлен фрагмент семантической сети, представляющий связь черт характера с видами направленности.

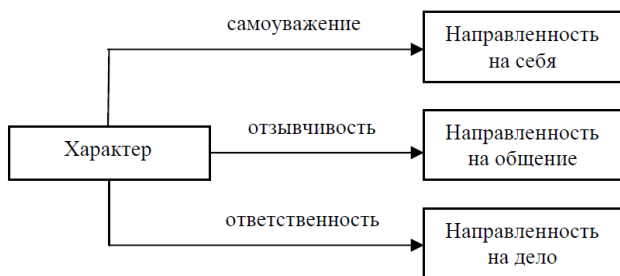


Рис. 3. Фрагмент семантической сети связи черт характера с видами направленности

4. Структура банка тестов

Банк тестов содержит совокупность тестов для каждой психологической характеристики, определённой в психологической модели личности. Ниже фрагмент содержимого банка тестов представлен в виде таблицы

Психологическая черта	Методики
Требовательность (к себе)	Оценка мотивации одобрения; СМОЛ; Лири;
Скромность	СМОЛ; Кэттел; Лири
Веселость	Кэттел; Характерологический опросник; Леонгард

Таким образом, банк тестов представляет собой перечень методик, структурированных в соответствии с психологическими чертами личности.

Заключение

В настоящей статье рассматривается структура и принципы функционирования автоматизированной системы генерации совокупности тестов по описанию требований пользователей, основу которой формируют информационная модель психотипа личности и банк психодиагностических методик. Данная система разрабатывается и будет реализована как WEB-приложение.

ЛИТЕРАТУРА

1. *Елисеев О.П.* Конструктивная типология и психодиагностика личности / Под ред. В.Н. Панферова. – Псков, 1994. – 280 с.
2. *Махлаков А.Г.* Общая психология. – СПб.: Питер, 2012. – 583 с.
3. *Столяренко Л.Д.* Основы психологии. – Ростов н/Д: Феникс, 2000. – 672 с.

РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ДЛЯ ПРОВЕДЕНИЯ ПСИХОДИАГНОСТИЧЕСКИХ ИССЛЕДОВАНИЙ

В.А. Осенчук, М.Н. Головчинер
Томский государственный университет
semdelion@gmail.com, golovchiner@mail.ru

Введение

Существует множество способов проведения психодиагностических исследований, начиная с вербального тестирования и заканчивая аппаратным. Данная работа посвящена компьютерному тестированию – автоматизированному тестированию в форме диалога испытуемого и ЭВМ.

Представленные в сети Интернет on-line системы для проведения психодиагностических исследований, по нашему мнению, обладают рядом недостатков, основным из которых является их закрытость. Это является причиной невозможности масштабирования – включения в систему новых тестов на основе специализированного пользовательского интерфейса [1]. В связи с этим, актуальной является разработка открытой системы для проведения психодиагностических исследований с возможностью расширения текущей базы тестов. Возможность расширения базы достигается за счёт наличия общего внутреннего способа (формата) представления тестов и способов их интерпретации.

Целью данной работы является создание:

- визуальной системы для проведения психодиагностического тестирования,
- единого внутреннего формата представления психодиагностических тестов,
- общей системы представления и обработки результатов тестирования,
- автоматическое формирование первичного диагноза на основании тестирования.

1. Общая архитектура системы

Общая архитектура системы (рис. 1) состоит из трёх основных блоков и двух баз данных (БД):

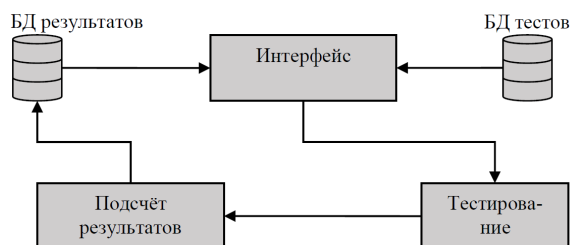


Рис. 1. Общая архитектура схемы

- «БД результатов» содержит все результаты тестирования в формате «*.txt»;
- «БД тестов» содержит все психодиагностические тесты, представленные во внутреннем формате;
- блок «Интерфейс» позволяет просматривать результаты тестирования и создавать новый набор психодиагностических тестов;
- блок «Тестирование» визуализирует каждый выбранный тест, представленный во внутреннем формате;
- блок «Подсчёт результатов» осуществляет подсчёт результатов и формирует первичный диагноз на основании результатов тестирования (результат тестирования записывается в «БД результатов»).

2. Способ представления психодиагностики тестов

Одна из основных проблем, возникающих при разработке системы, заключается в том, что тесты представляются множеством способов – в виде таблиц, изображений, списков, файлов различных форматов, и т.д. Способ формализации (перевода теста во внутренний формат системы) включает следующие шаги:

1. Изменение формата файлов

Все тесты преобразуются в формат «*.txt». В процессе смены формата некоторые тесты преобразуются для сохранения информации, хранящейся в таблицах и изображениях.

2. Разбиение теста на части

Несмотря на множественные способы представления тестов, тест, как правило, можно условно разделить на три части:

- список вопросов и ответов,
- представление ключей,
- интерпретация результатов тестирования.

Способ представления списка вопросов и ответов

Большинство тестов можно привести к следующему виду:

- i) Вопрос:
 - 1) Ответ
 - ...
 - j_i) Ответ

где i – номер вопроса, n – количество вопросов, $i \in (1, n)$, $(1, j_i)$ – список ответов на i -ый вопрос.

Способ представления ключей

Результатом тестирования, как правило, являются ответы на вопросы теста. Для проведения диагностики данные ответов необходимо правильно интерпретировать с использованием ключа интерпретации.

Ключи к тестам имеют множество способов представления: в виде таблиц списков, блоков и т.д. При преобразовании ключа интерпретации во внутренний формат каждая шкала оценивания представляется в виде отдельного блока. Каждый блок содержит список вопросов и ответов, каждому ответу соответствует определённое значение – балл. В некоторых случаях имеется математическое выражение, которое необходимо упростить и получить значение.

В примере 1 проиллюстрирован процесс преобразования ключа интерпретации во внутренний формат.

Пример 1

Часть ключа теста «Суицидальный риск» представлен в следующей таблице:

Часть ключа теста «Суицидальный риск»

Шкала	Номера суждений	Коэффициент	Максимальное значение фактора
1. Демонстративность	12,14,20,22,27	1,2	6
2. Аффективность	1,10,20,23,28,29	1,1	6,6

Порядок подсчета результатов, включенных в тест:

- 1) по каждой шкале подсчитывается сумма положительных ответов,
- 2) полученный балл умножается на коэффициент,
- 3) делается вывод о наличии/отсутствии факторов суицидального риска и степени их выраженности: чем ближе значение фактора к максимально возможной его величине, тем более он выражен.

Механизм реализации первых двух пунктов в рамках системы представляется следующей последовательностью шагов.

Шаг 1. Каждая из двух шкал оценивания помещается в соответствующий нумерованный дескриптор; номер вопроса записывается в – «[]», вариант ответа в – «()».

Шаг 2. Каждой паре вопрос-ответ присваивается балл.

После преобразования ключ во внутреннем формате принимает вид:

_SUM [0]<[12](1)=1,2+[14](1)=1,2+[20](1)=1,2+[22](1)=1,2+[27](1)=1,2>
_SUM [1]<1=1,1+[10](1)=1,1+[20](1)=1,1+[23](1)=1,1+[28](1)=1,1+[29](1)=1,1>.

Общее описание структуры дескрипторов приведено в п. 2.

3. Внутренний способ представления психодиагностических тестов

В процессе разработки приложения было принято решение разбить информацию теста на части и представить информацию как набор дескрипторов (тегов).

Общий вид дескриптора

Имя дескриптора [индекс] <Информационное поле>

“ ” – символ, обозначающий начало дескриптора, является обязательным.

“Имя дескриптора” – является обязательной частью; после имени дескриптора обязателен символ пробела.

“[индекс]” – индекс дескриптора, является необязательной частью дескриптора, необходим для нумерации дескрипторов.

“<Информационное поле>” – информационная часть дескриптора заключена в символы угловых скобок, содержит текст или информацию для работы дескриптора.

Основные виды дескрипторов

NAME <Имя теста> – дескриптор, содержащий имя теста.

DESCRIPTION <Инструкции> – дескриптор, содержащий дополнительную информацию, которая может понадобиться испытуемому, например:

- пояснения, необходимые для выбора ответа,
- донесение важной информации для испытуемого,
- напоминания.

QUESTION <Вопрос> – дескриптор, содержащий вопрос.

ANSWER <Вариант ответа> – дескриптор, содержащий в себе один из вариантов ответа на вопрос; один вопрос представляется как один дескриптор с вопросом и списком дескрипторов с вариантами ответов.

RESULT – дескриптор, являющийся разделителем между информационной частью теста и способом его оценки; данный дескриптор не содержит угловых скобок и располагается после заключительного ответа.

_HELP <Дополнительная информация> – дескриптор содержит дополнительную информацию, которая может понадобиться для правильной интерпретации диагноза.

_SUM [индекс] <[номер_вопроса](номер_ответа)=количество_баллов + ...+ [номер_вопроса](номер_ответа)=количество_баллов > – дескриптор для подсчёта баллов по заданному критерию. Данный дескриптор включает:

- **индекс** – необходим для обозначения необходимого критерия, нумерация индекса осуществляется от 0.
- **номер_вопроса** – нумерация начинается с 1.
- **номер_ответа** – нумерация начинается с 1.
- **количество_баллов** – количество полученных баллов.

В треугольных скобках находится список вида

<[номер_вопроса](номер_ответа)=количество_баллов+

...

+ [номер_вопроса](номер_ответа)=количество_баллов>

Замечание. Вместо знака ‘+’ могут быть применены знаки умножения и деления.

_SUMR [выражение]=>(левая_граница; правая_граница)<Текст> – дескриптор, проверяющий вхождения значения выражения в отрезок с указанными границами.

выражение имеет вид элементарного математического выражения с использованием таких знаков как:

- минус – ‘-’
- плюс – ‘+’
- умножение – ‘*’
- деление – ‘/’

– в круглых скобках “()” указывается индекс дескриптора (шкалы) **_SUM**

В случае истинности дескриптор возвращает <Текст>.

Данный подход имеет множество преимуществ, так как он позволяет сократить время обработки тестов, решить проблему классификации, а также является легко дополняемым, поскольку в случае невозможности обработки определённых частей теста можно представить данные части в виде дескриптора и дополнить базу дескрипторов новым дескриптором и способом его обработки в виде функции.

Пример 2

Тест состоит из трёх вопросов, каждый из которых имеет два варианта ответа. Ключ теста представлен в виде двух шкал:

1. Первая шкала состоит из трёх вопросов с вариантом ответа – «Верно», каждый ответ оценивается в один балл;
2. Вторая шкала состоит из двух вопросов с вариантом ответа – «Неверно», каждый ответ оценивается в три балла.

Сумма баллов, полученная по шкалам, сопоставляется с интервалами, и в случае принадлежности интервалу, выдается текст дескриптора.

Представление теста в виде последовательности дескрипторов

_NAME <Пробный тест>

_DESCRIPTION <Дайте один ответ на каждый вопрос>

_QUESTION <Вопрос 1>

_ANSWER <Верно>

_ANSWER <Неверно>

_QUESTION <Вопрос 2>

_ANSWER <Верно>

```

        _ANSWER <Неверно>
    _QUESTION <Вопрос 3>
        _ANSWER <Верно>
        _ANSWER <Неверно>
    _RESULT
    _HELP <Дополнительная информация>
    _SUM [0]<[1](1)=1+[2](1)=1+[3](1)=1>
    _SUM [1]<[1](2)=3+[2](2)=3>
    _SUMR [(0)]=>(0;3) <Вы ответили на 2 вопроса "Верно">
    _SUMR [(1)]=>(6;6) <Вы ответили на 1 и 2 вопрос "Неверно">
    _SUMR [1,3*(0)-0,4/(1)]=>(0;5) <Выражение в интервале>

```

4. Итоги проведения психодиагностического тестирования

После прохождения тестирования система формирует файл, в формате «*.txt» с результатом тестирования. Результат тестирования условно разделён на три части.

Пreamбула документа содержит информация о испытуемом: ФИО, дата рождения, дата тестирования и список пройденных тестов.

Главная часть файла включает результаты тестирования по каждому тесту в порядке их прохождения. Каждый тест представлен списком шкал и баллов.

Заключительная часть содержит информацию о тестировании: список вопросов и ответов, которые дал испытуемый. Формат записи – [номер вопроса] (номер ответа). Файл с результатом тестирования представлен в примере 3 [2].

Пример 3. Результат тестирования

Имя: Иванов Иван Иванович

Дата рождения: 8 3 1994

Дата тестирования: 07.04.2018

Тесты:

1) 13 Суицидальный Риск.txt

=====Диагноз=====

Опросник суицидального риска

(Баллов - 4) - 1) Демонстративность

- желание привлечь внимание окружающих к своим несчастьям, добиться сочувствия и понимания,
- оценивание из внешней позиции порой как «шантаж», «истероидное выпячивание трудностей»,
- демонстративное суицидальное поведение переживается изнутри как «крик о помощи».

Наиболее суицидоопасно сочетание с эмоциональной ригидностью, когда «диалог с миром» может зайти очень далеко.

(Баллов - 4) - 2) Аффективность

- доминирование эмоций над интеллектуальным контролем в оценке ситуации,
- готовность реагировать на психотравмирующую ситуацию непосредственно эмоционально.

В крайнем варианте (6 баллов) аффективная блокада интеллекта.

(Баллов - 4) - 3) Уникальность

- восприятие себя, ситуации и, возможно, собственной жизни в целом как явления исключительного, непохожего на другие и, следовательно, подразумевающего исключительные варианты выхода, в частности, суицид.

Тесно связан с феноменом «непроницаемости для опыта», т.е. с недостаточным умением использовать свой и чужой жизненный опыт.

(Баллов - 4) - 4) Несостоятельность

- отрицательная концепция собственной личности,
- представление о своей несостоятельности, некомпетентности, ненужности, «выключенности» из мира.

Данная субшкала может быть связана с представлениями о физической, интеллектуальной, моральной и прочей несостоятельности. Несостоятельность выражает интрапунитивный радикал. Формула внутреннего монолога: «Я плох». Возможно наличие «комплекса неполноценности».

(Баллов - 6) - 5) Социальный пессимизм тесно связан с экстрапунитивным стилем каузальной атрибутики

- отрицательная концепция принятия окружающего мира,
- восприятие мира как враждебного, не соответствующего представлениям о нормальных или удовлетворительных для человека отношениях с окружением.

(Баллов - 2) - 7) Максимализм

- инфантильный максимализм ценностных установок,
- максимальность ценностей,
- значимость малейшей потери с одновременной минимизацией ценностей значимости имеющихся достижений,
- распространение на все сферы жизни содержания локального конфликта в какой-то одной жизненной сфере,
- невозможность компенсации,
- аффективная фиксация на неудачах.

=====Ответы=====

Опросник суицидального риска

1[2](1)[3](2)[4](1)[5](2)[6](1)[7](1)[8](2)[9](1)[10](2)
[11](1)[12](1)[13](1)[14](1)[15](1)[16](1)[17](1)[18](2)[19](2)
[20](1)[21](1)[22](1)[23](1)[24](2)[25](1)[26](1)[27](2)[28](2)[29](1)

Заключение

В процессе выполнения работы была разработана открытая система для проведения психодиагностических исследований. Достигнуты следующие цели:

1. Разработана визуальная система для проведения психодиагностического тестирования. Система позволяет осуществлять следующие операции: выбор психодиагностических тестов, визуализация тестов, представленных во внутреннем формате, просмотр результатов тестирования.
2. На базе дескрипторов разработан единый внутренний формат представления психодиагностических тестов.
3. Разработана общая система представления и обработки результатов тестирования.
4. Разработана система, позволяющая автоматически формировать первичный диагноз на основании тестирования.

В систему на данный момент включено двадцать восемь психодиагностических тестов, используемых в одном из центра реабилитации нарко- и алкозависимости.

ЛИТЕРАТУРА

1. Немов Р.С. Психология : в 3 ч. – М.: Владос, 2002. – Ч.3: Психодиагностика. – 688 с.
2. Райгородский Д.Я. [и др.] Практическая психодиагностика: [учеб.-метод. пособие для вузов по специальности „Психодиагностика“]. – Самара: Изд-во Дом «Бахрах», 1998. – 672 с.

ПРОГРАММНЫЙ КОМПЛЕКС МОДЕЛИРОВАНИЯ ДВУХКОМПОНЕНТНОГО РАЗЛОЖЕНИЯ КАРДИОЛОГИЧЕСКОЙ КРИВОЙ

С.И. Самохина, М.М. Шишкин

Томский государственный университет
sv.sam.tsk@gmail.com, mikhael.shishkin@gmx.com

Введение

Важность проблем, связанных с исследованием кардиологических заболеваний, не подвергается сомнению. Сердечно-сосудистые заболевания – основная причина смертности среди населения. По данным Всемирной Организации Здравоохранения (ВОЗ) смертность от сердечно-сосудистых заболеваний составляет 31% и является наиболее частой причиной смертельных исходов во всем мире [1].

Одной из важнейших задач современной фармакологии является создание новых кардиотропных препаратов, с помощью которых можно было бы повысить эффективность лечения сердечно-сосудистых заболеваний и реализовать проблемы современной фармакотерапии.

В последнее время все большее внимание исследователей привлекает изучение компонентного сокращения миокарда, возникающего при действии определенных фармакологических соединений.

В научных трудах [2,3] исследована возможность использования двухкомпонентного сокращения миокарда с целью изучения механизма действия кардиотропных веществ. Для анализа механизма действия этих веществ была смоделирована математическая модель компонентной структуры сокращения методом регрессионного анализа и проведен статистический анализ модели.

Целью нашей работы является создание программного комплекса, предназначенного для двухкомпонентного разложения кардиологической кривой, полученной экспериментально и проведение статистического анализа исходных данных.

1. Описание программного комплекса моделирования двухкомпонентного разложения кардиологической кривой

Use-case – диаграмма вариантов использования разрабатываемой информационной системы изображена на рис. 1.

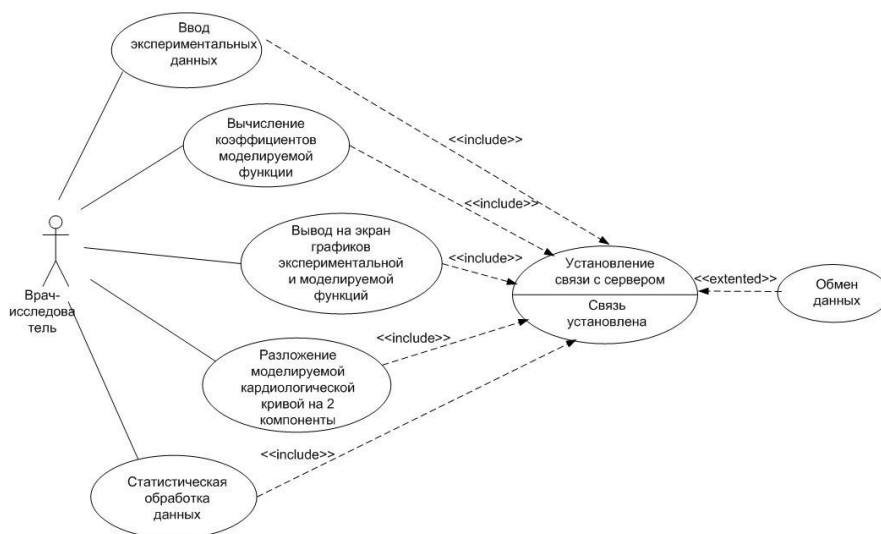


Рис. 1. Use-case схема вариантов использования

Пользователь (врач-исследователь) после запуска программы, нажав на кнопку «Загрузить данные из БД», устанавливает связь с сервером базы данных. При этом либо загружаются выбранные значения в программу, либо, если по какой-то причине не удается установить связь, возникает сообщение об ошибке.

При нажатии на кнопку «Расчёт коэффициентов», запускается модуль вычисления коэффициентов оптимизируемой функции. Вслед за этим, на основании полученных расчетных значений, автоматически строятся графики экспериментальной и двухкомпонентной моделируемой функций, а также осуществляется разложение моделируемой функции на две компоненты.

Пользователь может вычислить медицинские индексы, нажав на кнопку «Статобработка», после чего на основе найденных ранее коэффициентов и значений индексы вычисляются в автоматическом режиме.

Выбор средств разработки программного обеспечения

Программный комплекс реализован на языке программирования C# с использованием WinForms, а также СУБД MySQL.

C# – объектно-ориентированный язык программирования, базирующийся на программной платформе .NET Framework [4].

WindowsForms – интерфейс программирования приложений (API), отвечающий за графический интерфейс пользователя и являющийся частью Microsoft.NET Framework [5].

MySQL – свободная реляционная система управления базами данных [6].

Описание программного продукта

Программный комплекс включает все этапы от ввода данных, поступающих с аналого-цифрового преобразователя АЦП до получения конечных результатов обработки начальных данных и статистической обработки.

Исходными данными для программы являются экспериментальные значения, поступающие с АЦП и вводимые в базу данных. Устройство ввода данных представляет собой многоканальный аналоговый преобразователь, предназначенный для преобразования аналоговых сигналов в двенадцатиразрядный двоичный код. Эксперименты на установках проводятся сериями опытов. В каждом опыте находится по двадцать вводов. На основании введенных значений из базы данных программа осуществляет вычисление коэффициентов моделируемой функции методом градиентного спуска, а также расчёт максимумов аппроксимирующей функции для дальнейшего анализа сокращения миокарда. Рабочее окно программы представлено на рис. 2.

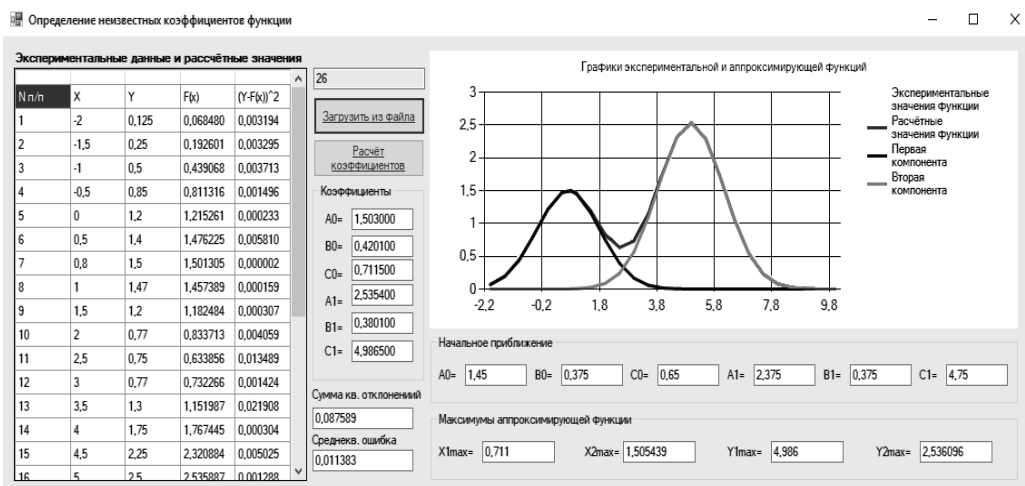


Рис. 2. Рабочее окно программы расчета моделируемой функции

В зависимости от действия введения определенного кардиотропного лекарственного препарата будет наблюдаться разложение сокращения сердечной мышцы на одну или две компоненты. После того, как будет проведено некоторое количество опытов и соберётся достаточно большой объём данных, можно провести статистический анализ – выяснить, имеется ли существенное влияние определенного лекарственного препарата на сердечную мышцу при помощи критерия Стьюдента.

2. Математическая модель компонентной структуры сокращения миокарда

Предложенная в [2–3] математическая модель компонентной структуры позволяет представить кривую сокращения миокарда в виде суммы двух отдельных модельных компонентов. Предполагается, что искомые компоненты сокращения принадлежат некоторому семейству функций $y(x, \vec{a})$, где x – время, \vec{a} – вектор параметров, определяющих конкретную форму кривой (функции), а моделью $f(x, \vec{a})$ наблюдаемой кривой силы сокращения $g(x)$ является сумма n компонентов, т.е. $f(x, \vec{a}) = \sum_{k=1}^n y(x, \vec{a}_k)$.

Поскольку на практике мы имеем дело со значениями функций в заданные моменты времени, то отсчётам наблюдаемой кривой $g(x_i)$ соответствуют отсчёты модели $f(x, \vec{a}) = \sum_{k=1}^n y(x_i, \vec{a}_k)$, $i = 1, N$, где N – количество моментов времени, в которые наблюдаются значения функций g и f .

Для оценки степени соответствия модели f и наблюдаемой кривой g выбран критерий F (среднее квадратичное отклонение), который определили как

$$F = (\vec{a}_1, \vec{a}_2, \dots, \vec{a}_n) = \frac{1}{N} \sum_{i=1}^N [g(x_i) - f(x_i, \vec{a})]^2 = \frac{1}{N} \sum_{i=1}^N \left[g(x_i) - \sum_{k=1}^n y(x_i, \vec{a}_k) \right]^2 \rightarrow \min.$$

При этом необходимо найти такие значения \vec{a} , при которых критерий F достигает наименьшей величины.

Были проанализированы периоды кривых сокращения сердечных мышц, которые в результате действий кардиотропных веществ, сравнивались однокомпонентными кривыми. Замечено, что кривые нарастания и спада такого одиночного компонента сокращения очень близки к экспоненциальной зависимости, при этом каждая из кривых явно отличается коэффициентами экспоненциальной функции, т.е. было выявлено, что этот период представляется в виде математической функции, задающейся в виде формулы $y = a_1 e^{-b_1(x-c_1)^2} + a_2 e^{-b_2(x-c_2)^2}$.

В случае, если регистрируемая кривая сокращения мышцы представляет собой сумму двух компонент, среднее квадратичное отклонение определится выражением

$$F = (x, a_1, b_1, c_1, a_2, b_2, c_2) = \frac{1}{N} \sum_{i=1}^N \left[g(x_i) - \left(a_1 e^{-b_1(x-c_1)^2} + a_2 e^{-b_2(x-c_2)^2} \right) \right]^2.$$

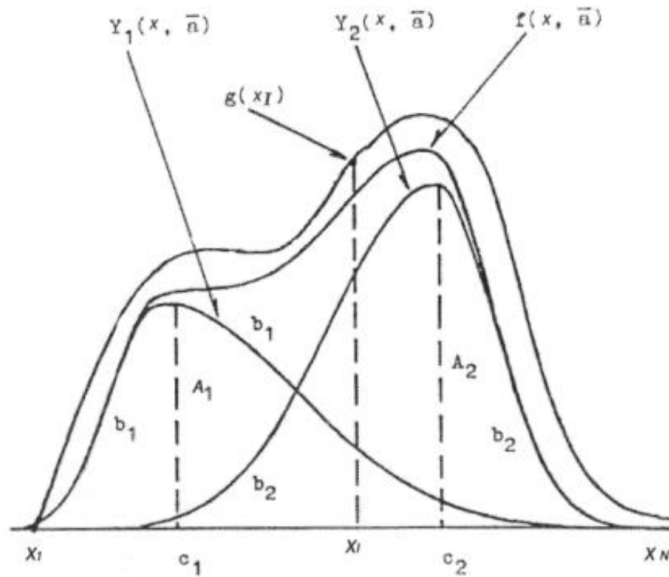


Рис. 2. Схема построения модели компонентной структуры сокращения миокарда: $g(x_i)$ – регистрируемая кривая сокращения, $f(x, \bar{a})$ – её математическая модель, $y_1(x, \bar{a})$ – первый модельный компонент, $y_2(x, \bar{a})$ – первый модельный компонент, A_1 и A_2 – амплитуды первого и второго модельных компонентов, b_1 и b_2 – коэффициенты, определяющие форму модельных кривых, c_1 и c_2 – время достижения максимумов модельных кривых, x_1 и x_N – время начала и конца наблюдения.

Если разница между коэффициентами c_1 и c_2 незначительна, то на графике отчетливо виден лишь один максимум, однако в общем случае кривая имеет два максимума (рис. 2).

В модели имеется шесть оптимизационных переменных, которые полностью характеризуют две искомые компоненты. В исследовательских и научно-практических целях необходим алгоритм, осуществляющий нахождение коэффициентов $a_1, a_2, b_1, b_2, c_1, c_2$ для того, чтобы в конечном итоге получить из функции, заданной табличным способом, разложение на отдельные компоненты.

3. Модуль вычисления коэффициентов оптимизируемой функции

Решая оптимизационным методом задачу нахождения минимальной величины критерия F , можно определить оптимальные значения этих переменных для каждой кривой сокращения мышцы. Для осуществления данной задачи воспользуемся методом градиентного спуска. Основная идея этого метода заключается в том, чтобы идти в направлении наискорейшего спуска, а это направление задается антиградиентом $-\nabla F$ [7].

Как известно, градиент функции в некоторой точке $x^{(k)}$ направлен в сторону наискорейшего локального возрастания функции. Вектор, противоположный градиенту $\nabla f(x^{(k)})$, называется антиградиентом, который направлен в сторону наискорейшего убывания функции $f(x)$.

Алгоритм метода поиска точки минимума состоит в следующем: на первом этапе берется произвольная точка $x^{(0)}$ и с помощью антиградиента $\nabla f(x^{(0)})$, вычисленного в этой точке, определяется направление, в котором функция $f(x)$ убывает с наибольшей скоростью, после этого, сделав небольшой шаг в найденном направлении, переходим в новую точку $x^{(1)}$. На следующем этапе вновь определяем наилучшее направление –

$\nabla f(x^{(1)})$, чтобы перейти в очередную точку $x^{(2)}$, и т.д. В результате проделанных действий получаем итерационный процесс следующего вида: $x^{(k+1)} = x^{(k)} - \alpha^{(k)} \nabla f(x^{(k)})$, $\alpha^{(k)} > 0$, $k = 0, 1, \dots, n$.

В данном методе величина шага $\alpha^{(k)}$ выбирается таким образом, чтобы было выполнено неравенство:

$$f(x^{(k)} - \alpha^{(k)} \nabla f(x^{(k)})) < f(x^{(k)}). \quad (1)$$

Значение функции должно убывать от итерации к итерации. Процесс выбора шага происходит следующим образом. Выбирается число, одно и то же для всех итераций. На k -й итерации проверяется выполнение неравенства (1) при шаге $\alpha^{(k)} = \alpha$. Если неравенство выполнено, полагаем шаг $\alpha^{(k)} = \alpha$ и переходим к следующей итерации. Если нет, то шаг $\alpha^{(k)}$ начинаем дробить, т.е. уменьшаем каждый раз, допустим в два раза, до тех пор, пока неравенство (8) не выполнится.

В качестве критерия останова итерационного процесса используется выполнение условия малости градиента: $\|\nabla f(x^{(k+1)})\| \leq \varepsilon$, где ε – заданная точность.

Блок-схема алгоритма расчёта коэффициентов оптимизируемой функции представлен на рис. 9.

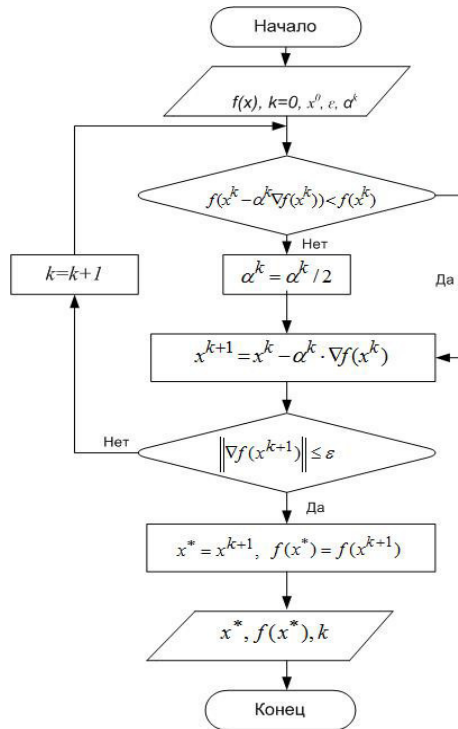


Рис. 9. Блок-схема алгоритма расчета коэффициентов оптимизируемой функции

Заключение

В ходе проделанной работы были реализованы поставленные цель и задачи:

1. Разработан программный комплекс, осуществляющий разложение кардиологической кривой на две компоненты и статистический анализ данных.
2. Реализован алгоритм вычисления коэффициентов оптимизируемой функции, описывающей разложения кардиологической кривой на две компоненты.

3. Исследована математическая модель двухкомпонентного разложения кардиологической кривой и её поведение в результате воздействия лекарственных кардиотропных препаратов.

4. Полученные результаты имеют прикладное значение, программа моделирования двухкомпонентного разложения кардиологической кривой используется для исследования сократительной активности миокарда в результате инотропного воздействия фармакологических агентов.

ЛИТЕРАТУРА

1. ВОЗ | 10 ведущих причин смерти в мире [Электронный ресурс] – URL: <http://www.who.int/mediacentre/factsheets/fs310/ru> (дата обращения 15.03.2018).
2. Богомаз С.А. Исследование действия кардиологических веществ на пулы кальция с помощью анализа компонентов структуры сокращения миокарда: дис. ... канд. мед.наук. – Томск, 1991. – 184 с.
3. Тарасенко В.Ф., Лантев Б.И. Способ компонентного анализа сокращения миокарда // Сборник трудов молодых ученых ТНЦ АМН. – 1991. – С. 44–47.
4. C Sharp [Электронный ресурс] – URL: https://ru.wikipedia.org/wiki/C_Sharp (дата обращения 03.04.2018).
5. Windows Forms [Электронный ресурс] – URL: https://ru.wikipedia.org/wiki/Windows_Forms (дата обращения 03.04.2018).
6. MySQL [Электронный ресурс] – URL: <https://ru.wikipedia.org/wiki/MySQL> (дата обращения 03.04.2018).
7. Метод градиентного спуска [Электронный ресурс] – URL: <http://bukvasha.ru/referat/410263> (дата обращения 05.04.2018).

ИНФОРМАЦИОННАЯ СИСТЕМА АНАЛИЗА СОСТОЯНИЯ ЗДОРОВЬЯ ЧЕЛОВЕКА «Н.TOGETHER»

В.А. Устинова, С.И. Самохина

Томский государственный университет
veronika.ustinova97@yandex.ru, svip@sibmail.com

Введение

Главной целью системы «Н.TOGETHER» является создание информационного пространства, в котором будут предоставлены инструменты для улучшения общего самочувствия пользователя. Актуальность данного проекта обусловлена потребностью о социальном здоровье как неотъемлемой части современной общественной жизни. Программа рекомендуется для использования пользователями, которые заинтересованы в улучшении своего здоровья, также она сможет помочь специалистам в области нутрициологии, физической культуры и спорта автоматизировать расчёты, которые ранее были сделаны вручную на листке бумаги.

Здоровый образ жизни – это совокупность факторов, таких как питание, физическая активность и благоприятная окружающая среда. Человеку необходимо равновесие между поступающей с пищей энергией и энергией, расходуемой во время жизнедеятельности, т.е. баланса энергии [1].

За последние 10 лет тема здорового образа жизни стала очень популярной. В этом можно убедиться по итогам опроса, проведенного ВЦИОМ 21 марта 2018 г. [2]. По данным опросов, каждый четвертый россиянин (25%) регулярно занимается спортом, 19% граждан соблюдают специальную диету, а 41% – старается есть здоровую пищу. Отсюда можно утверждать, что забота о здоровье – очевидный тренд на рынке технологий.

Без разработки качественной ИС сейчас не обходится ни один вид деятельности, как в науке, так и в производстве и в повседневной жизни [3–7]. Наша информационная система позволяет пользователю внимательнее следить за самочувствием и регулярно отслеживать свои результаты. В качестве среды разработки использована версия Microsoft Visual Studio Express 2013 [8].

Для общего понимания структуры данного проекта описана система на концептуальном уровне. При проектировании выделено 3 актера: нутрициолог, программист, пользователь, и 11 вариантов использования (рис. 1).

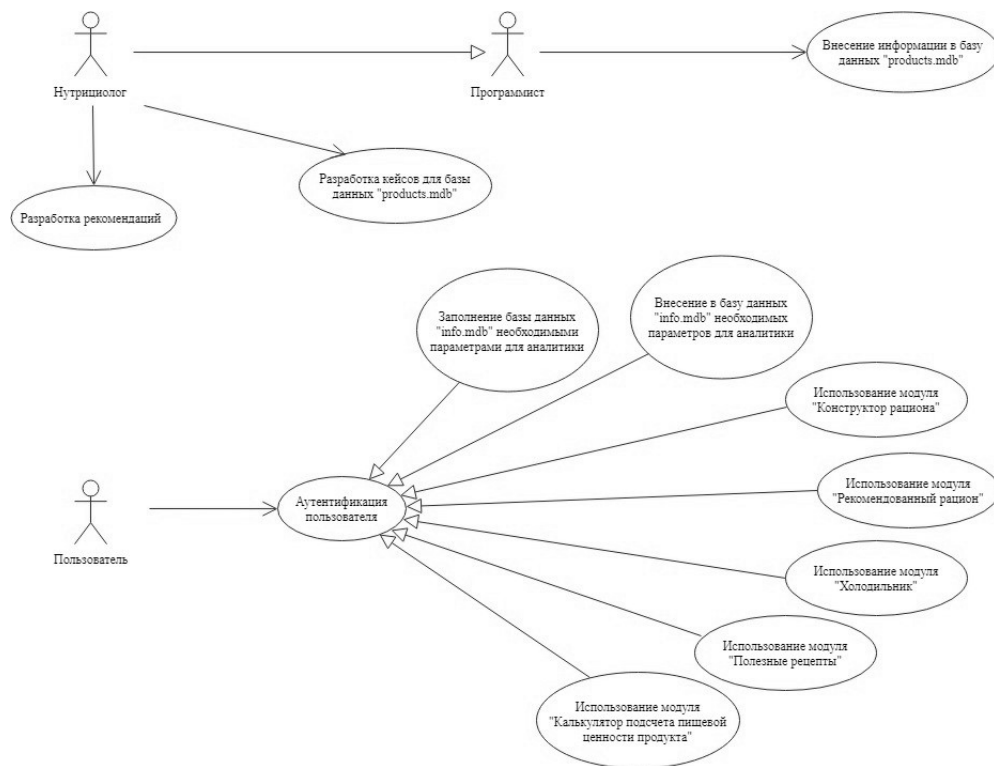


Рис. 1. Диаграмма вариантов использования

В [9] рассмотрено 5 блоков приложения, с помощью которых пользователь может зарегистрироваться в системе, посчитать свою суточную норму потребления калорий, заполнить антропометрические данные и произвести некоторую аналитику. На текущий момент информационную систему дополнили следующие пункты:

1. База данных «products.mdb».
2. Модуль «Полезные рецепты».
3. Модуль «Рекомендованный рацион».
4. Модуль «Конструктор рациона».

База данных «products.mdb» имеет структуру, представленную на рис. 2. Она состоит из 7 таблиц, по которым производится расчёты и сбор информации для модулей.

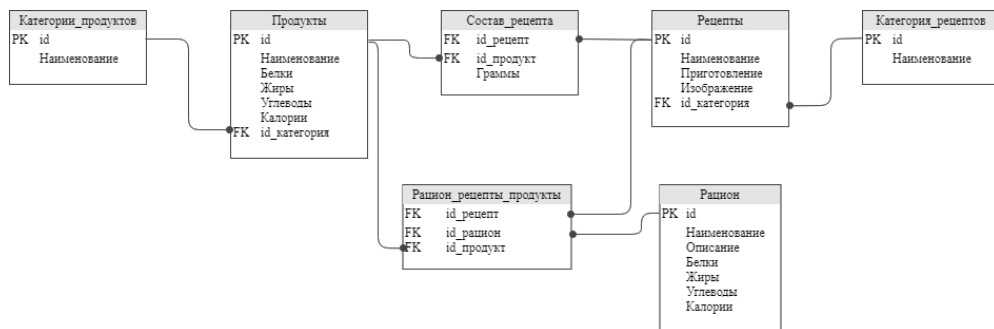


Рис. 2. База данных «products.mdb»

1. Модуль «Полезные рецепты»

Модуль «Полезные рецепты» (рис.3) имеет 7 категорий, по которым производится выборка: Завтрак, Обед, Ужин, Перекусы и выпечка, Соусы и заправки, Смузи и напитки, Салаты.

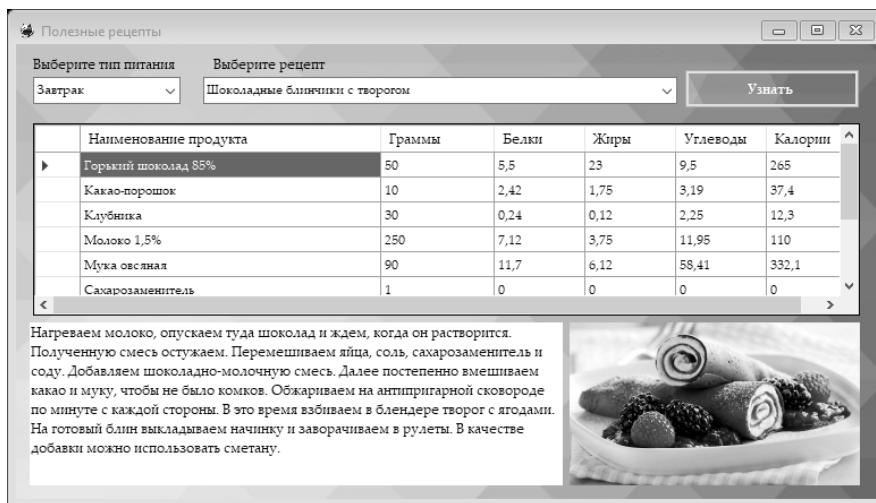


Рис. 3. Модуль «Полезные рецепты»

Далее пользователем выбирается наименование желаемого рецепта и получается информация о составе рецепта, а именно – граммы, пищевая ценность продуктов (белки, жиры, углеводы, калории), описание приготовления и демонстрация готового блюда в виде изображения. Для расчета итоговых показателей рецепта производится суммирование КБЖУ каждого из продуктов. Такой расчет осуществляется с помощью класса «CalculatorProducts.cs», принцип работы которого описан в [9].

Получение информации о приготовлении блюда и вставка изображения к выбранному рецепту осуществляются с помощью запроса к базе данных «products.mdb» по связанному таблицам.

2. Модуль «Конструктор рациона»

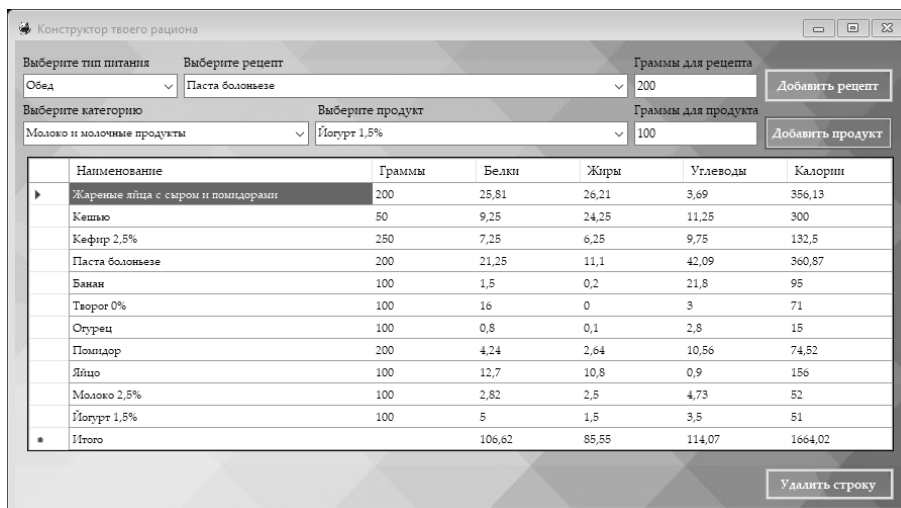


Рис. 4. Модуль "Конструктор рациона"

В модуле «Конструктор рациона» пользователь может самостоятельно выбрать из списка предлагаемых категорий любой рецепт или продукт. Далее, введя уточняющие граммы, он может спланировать рацион на любой день. Также это может быть применимо для расчёта собственного рецепта, для которого пользователь может выбрать нужные продукты из предлагаемого списка и вес каждого из ингредиентов. Программа выдаст итоговые значения готового блюда, что позволит пользователю узнать калорийность.

Заключение

С пищей мы получаем не только энергию, но и строительный материал для нашего организма. Рекомендуется следить за своим рационом, употреблять качественные продукты и наблюдать за балансом энергии для хорошего самочувствия. Данная система позволяет пользователю объединить эти процессы в одном информационном пространстве.

Пропаганда и мода на здоровье необходимы социуму, так как это позволяет побуждать человека к включению в его повседневную жизнь различных форм поведения, полезных для здоровья; помочь в отказе от вредных привычек; сформировывать убежденность в необходимости выполнения рекомендаций по устранению факторов риска заболеваний; предоставлять навыки и знания, оказывающие благоприятное влияние на психическое, психологическое, нравственное, социальное и физическое состояние индивида.

ЛИТЕРАТУРА

1. Трча С. Искусство вести здоровый образ жизни. – М.: Медицина, 1984. – 232 с.
2. Институт Общественного Мнения «Анкетолог» по материалам ВЦИОМ [Электронный ресурс] / – URL: <https://www.wciom.ru> (25.03.2018).
3. Самохина С.И., Петелин А.Е. Разработка программного комплекса для моделирования зоны сдвига в гцк-металлах // Вестник ТГУ. Приложение. – 2006. – № 18. – С. 141–145.
4. Самохина С.И., Петелин А.Е., Колупаева С.Н. Моделирование зоны кристаллографического сдвига в гцк металлах. Численное решение системы жестких дифференциальных уравнений // Вестник ТГУ. Приложение. – 2007. – № 23. – С. 333–338.
5. Petelina Y., Samokhina S., Petelin A., Polosukhin K., Vik K., Kolupaeva S. The dynamics of near-surface prismatic loops in lead // AIP Conference Proceedings 2. Сер. "Advanced Materials in Technology and Construction, AMTC 2015: Proceedings of the II All-Russian Scientific Conference of Young Scientists "Advanced Materials in Technology and Construction". – 2016. – С. 040004.
6. Просин Р.А., Просина С.Г., Самохина С.И. Пакет прикладных программ для исследования пространственных изменений сосудов // ИННОВАТИКА-2016: сб. материалов XII Международной школы-конференции студентов, аспирантов и молодых ученых. – Томск: SST, 2016. – С. 374–377.
7. Просин Р.А., Самохина С.И. Информационная система учета клиентов центра биологической медицины // ИННОВАТИКА-2013: сб. материалов IX Всероссийской школы-конференции студентов, аспирантов и молодых ученых с международным участием. Т. 2. – Томск: ТГУ, 2013. – С. 84–88.
8. Шилдт Г. Полный справочник по С#. – М.: Издательский дом «Вильямс», 2004. – 752 с.
9. Устинова В.А., Самохина С.И. Программа для анализа состояния здоровья человека // Информационно-измерительная техника и технологии: материалы VIII научно-практической конференции (22–25 ноября 2017 г.). – Национальный исследовательский Томский государственный университет, 2017. – С. 98–99.

РАЗРАБОТКА ПОЛЬЗОВАТЕЛЬСКОГО ПРИЛОЖЕНИЯ СРЕДСТВАМИ BORLAND DELPHI

Г.Я. Хусаинова, А.В. Камагаева

Стерлитамакский филиал Башкирского государственного университета
gkama@mail.ru

Введение

В современном мире быстрый и качественный доступ к нужной информации является одним из главных критериев эффективности управления организацией или производством, и поэтому современная жизнь немыслима без эффективного управления.

Важной категорией являются системы обработки информации, от которых во многом зависит эффективность работы любого предприятия или учреждения. Такая система должна [1]:

- обеспечивать получение общих и/или детализированных отчетов по итогам работы;
- позволять легко определять тенденции изменения важнейших показателей;
- обеспечивать получение информации критической по времени, без существенных задержек;
- выполнять точный и полный анализ данных.

Средства реализации

В данной работе рассмотрена реализация клиентского приложения для информационной системы «Биржа труда» средствами Borland Delphi.

Взаимодействие пользователя с системой осуществляется в диалоговом режиме. При запуске программы выводится следующее окно, предлагающее начать работу с приложением. Главная кнопочная форма программы имеет вид [2]:

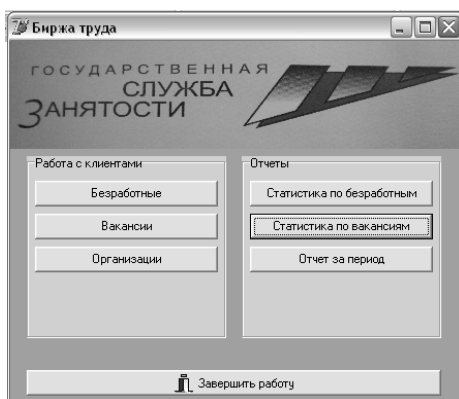


Рис. 1. Главная кнопочная форма

Кнопка «Безработные» (рис. 1) позволяет просмотреть и распечатать основную информацию о безработных, редактировать ее, добавить или удалить безработного и его последние места работы, отфильтровать безработных по одному из полей и отсортировать их по каждому полю; повышает скорость поиска безработных по номеру паспорта; дает возможность подобрать вакансию безработному (рис. 2).

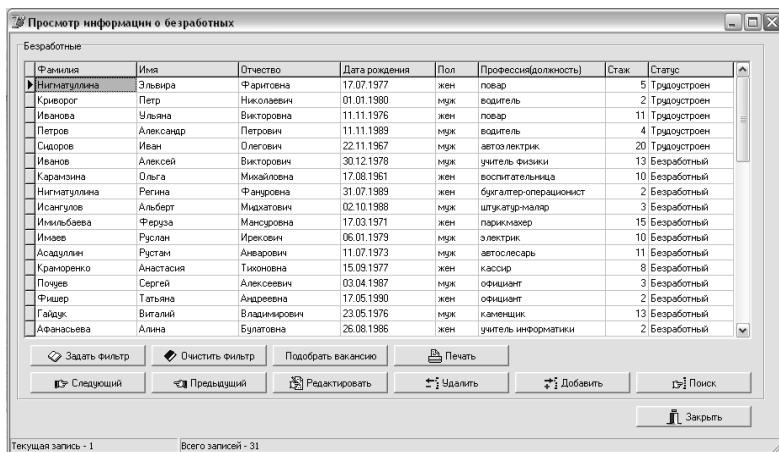


Рис. 2. Просмотр информации о безработных

Процедуры поиска безработного по номеру паспорта и печати информации реализованы в формах просмотра информации о вакансиях и организациях.

При нажатии кнопки «Вакансии» (рис. 1) на экране появляется форма, которая позволяет вести учет вакансий, редактировать, добавлять и удалять их; автоматизировано подбирать рабочего по каждой вакансии и производить поиск по номеру вакансии; реализовывать печать информации о вакансиях, их фильтрацию и сортировку по каждому полю (рис. 3).

Организация	Профессия	Стаж	Зарплата	Пол	Образование	Свободно	Состояние	Дата регистрации
ООО "Росавто"	автом лектрик	2	5500	муж	начальное профессиональное	0	открыта	01.01.2005
ООО Боулинг Клуб "Заря"	официант	0	4500	жен	основное среднее	3	открыта	01.01.2007
ОАО "Общепит"	водитель	0	6500	муж	начальное профессиональное	1	закрета	01.01.2010
ООО Боулинг Клуб "Заря"	повар	3	5000	жен	основное среднее	3	открыта	06.05.2011
ОАО "Общепит"	повар	1	10000	жен	основное среднее	1	открыта	11.05.2011
СГПА им. З.Бишиевой	ассистент	1	10000	муж	высшее профессиональное	0	открыта	11.05.2011
ОАО "Сода"	сантехник	0	5000	муж	основное среднее	1	открыта	11.05.2011
СГПА им. З.Бишиевой	водитель	0	6000	муж	среднее профессиональное	1	открыта	16.05.2011
МБОУ БЛИ №3	воспитательница	5	5000	жен	среднее профессиональное	2	открыта	16.05.2011
ООО Ночной клуб "Метро"	швея	2	6000	жен	среднее профессиональное	3	открыта	11.05.2011
ОАО "БашФотонур"	фотограф	3	7000	жен	среднее профессиональное	2	открыта	12.05.2011
МБОУ БЛИ №3	учитель информатики	0	8000	жен	высшее профессиональное	1	открыта	16.05.2011
СГПА им. З.Бишиевой	учитель физики	5	9000	муж	высшее профессиональное	1	открыта	10.05.2011
ЗАО "Кудесница"	кассир	1	7000	жен	высшее профессиональное	3	открыта	10.05.2011
ЗАО "Кудесница"	парикмахер	1	5000	жен	среднее профессиональное	5	открыта	10.05.2011
ЗАО "Кудесница"	бухгалтер-операционис	1	8000	жен	высшее профессиональное	2	открыта	18.05.2011
ЗАО "Кудесница"	продавец	1	6000	жен	высшее профессиональное	3	открыта	10.05.2011

Рис. 3. Просмотр информации о вакансиях

Кнопка «Организации» (рис. 1) позволяет просмотреть информацию о зарегистрированных организациях, редактировать ее, добавлять или удалять организации и отсортировать их по каждому полю; производить поиск по ИНН организации (рис. 4).

ИНН	Наименование	Отрасль	Адрес	Телефон	Руководитель	Регистрация
026745464	ООО "Росавто"	Транспорт и связь	Западная, 8	434281	Кадыршин М. З.	31.07.2007
026947465	ООО Боулинг Клуб "Заря"	Торговля и общественное питание	пр. Октября, 31	234545	Субботина Н. В.	14.03.2006
026898929	ОАО "Сода"	Промышленность	Бабюшкина, 2	289056	Тигов В. В.	02.05.2003
026754656	ЗАО "Каустик"	Промышленность	Техническая, 32	435674	Дмитриев И. А.	03.05.2005
026811111	СГПА им. З.Бишиевой	Народное образование	пр. Ленина, 47	253341	Кызыргулов	03.05.2008
026845364	ЗАО "СтройТехПром"	Строительство	пр. Ленина, 2	431221	Чернов И. Г.	04.05.2011
026846465	ЗАО "СтройЭнергоСервис"	Строительство	Элеваторная, 39	437865	Шапкин М. М.	11.02.2009
026846456	ООО Ночной клуб "Метро"	Торговля и общественное питание	Мира, 26	253750	Моисеева А. Н.	21.09.2007
026846564	МДОУ "Золотая рыбка"	Народное образование	Гоголя, 90	264533	Окунева О. М.	15.06.2005
026846575	ЗАО "Кудесница"	Сфера услуг	Худайбердина, 58	257769	Кудрявцева Л. Ф.	18.04.2009
026845453	ОАО "Общепит"	Торговля и общественное питание	пр. Ленина, 9	234456	Гаврилов Б. И.	06.05.2008
026844656	ЗАО АКБ ВТБ-24	Финансы, кредит, страхование	Коммунистическая, 85	267455	Прохоров А. Б.	17.05.2010
026746465	МБОУ БЛИ №3	Народное образование	Худайбердина, 96	223465	Ахмадуллина М. Ф.	11.09.2010
026806575	ЗАО "Чемпион"	Торговля и общественное питание	пр. Октября, 56	248900	Ковров С. П.	30.03.2010
026745465	ОАО "БашФотонур"	Сфера услуг	пр. Ленина, 6а	437645	Газизов М. С.	19.05.2010

Рис. 4. Просмотр информации об организациях

Кнопка «Статистика по безработным» (рис. 1) позволяет просмотреть статистические данные по зарегистрированным безработным: сводную информацию, распределение по половому признаку и по профессиям (рис. 5).

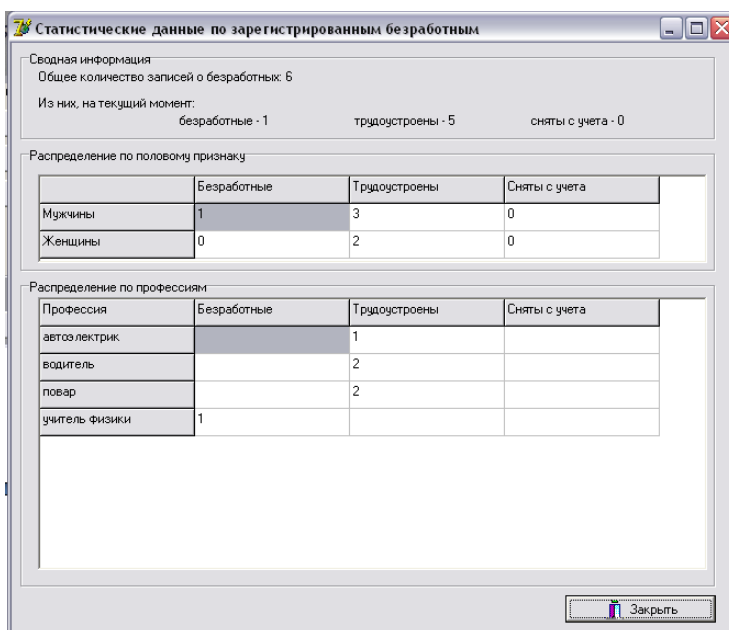


Рис. 5. Статистические данные по зарегистрированным безработным

Кнопка «Статистика по вакансиям» (рис. 1) также позволяет просмотреть статистические данные по имеющимся вакансиям: сводную информацию, распределение по организациям и по профессиям (рис. 6).

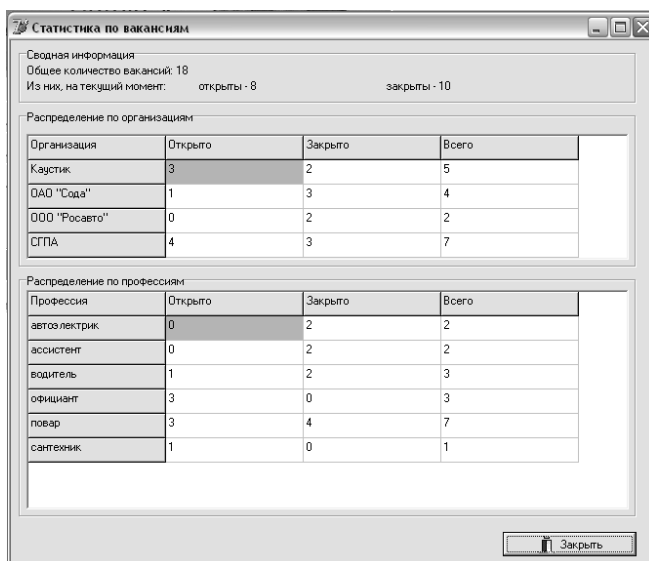


Рис. 6. Статистика по вакансиям

Кнопка «Отчет за период» (рис. 1) позволяет сформировать и просмотреть отчет о работе за выбранный период (рис. 7).

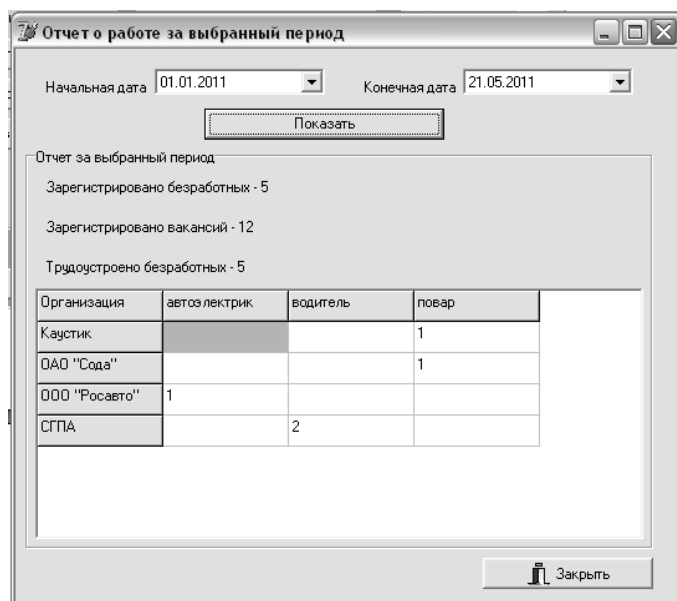


Рис. 7. Отчет о работе за выбранный период

Заключение

Приложение «Биржа труда» реализовано в среде разработки Delphi, связь которого с базой данных Access осуществляется с использованием компонент ADO. Подобный подход позволяет использовать данное приложение на большинстве компьютеров без установки дополнительного программного обеспечения [3]. Также было использовано CASE-средство проектирования информационных систем ERwin.

ЛИТЕРАТУРА

1. Хусаинова Г.Я., Хусаинов И.Г. Разработка автоматизированного рабочего места менеджера по продаже автомобилей // *Фундаментальные и прикладные исследования в современном мире*. – 2014. – Т. 1. – № 7. – С. 126–128.
2. Хусаинова Г.Я., Хусаинов И.Г. Информационные системы. Управление реляционными базами данных: Учеб.-метод. материалы. – Стерлитамак: Стерлитамак. гос. пед. академия, 2007. – 55 с.
3. Хусаинов И.Г. Тепловые процессы при акустическом воздействии на насыщенную жидкостью пористую среду // *Вестник Башкирского университета*. – 2013. – Т. 18. – № 2. – С. 350–353.

ИНФОЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ НА ПРИМЕРЕ БИРЖИ ТРУДА

Г.Я. Хусаинова, А.В. Федоров

*Стерлитамакский филиал Башкирского государственного университета
gkama@mail.ru*

Введение

В работе рассмотрена реализация инфологического этапа проектирования на примере биржи труда, оказывающей услуги населению по трудоустройству и помощь работодателям при закрытии вакансий на их предприятиях. На этом этапе основной целью является изучение предметной области [1].

Нестабильность экономики приводит к росту безработицы, поэтому технические разработки становятся актуальными, возникает необходимость автоматизации работы специалистов биржи труда с целью снижения временных затрат на выполнение стандартных рутинных операций. Практическая значимость работы определяется разработкой программного средства, служащего для автоматизации работы инспекторов центра

занятости населения. Это позволит увеличить производительность труда, освободить работника от выполнения менее значительных пунктов своей работы и предоставить часть времени на выполнение более важной работы. Автоматизированный способ ведения данных процессов позволяет наиболее достоверно, быстро и безошибочно собирать и производить различные операции с данными [2,3].

1. Концептуальный этап проектирования

Исходя из выше сказанного, можно выделить главные процессы в деятельности биржи труда:

1. Клиенты (безработные) оставляют информацию о себе на бирже труда в виде заполненной анкеты. Работник биржи труда (инспектор) регистрирует информацию о безработном.

2. Работодатели делают заявку на персонал, сообщая информацию о наличии вакансий, либо работник биржи труда самостоятельно по телефону выясняет данную информацию. Информация о вакансиях регистрируется.

3. Работник биржи труда анализирует информацию о вакансиях и безработных, выдает безработным информацию в виде перечня открытых вакансий, подходящих данному клиенту (безработному).

4. Безработные рассматривают предложения и сообщают о результатах (факт трудоустройства или отсутствие такового). Работник биржи фиксирует данную информацию.

После изучения предметной области, анализа и систематизации информации получены сущности и атрибуты.

Концептуальное проектирование – это сбор, анализ и редактирование требований к данным. Для этого осуществляются следующие мероприятия:

- обследование предметной области, изучение её информационной структуры;
- выявление всех фрагментов, каждый из которых характеризуется пользовательским представлением, информационными объектами и связями между ними, процессами над информационными объектами;
- моделирование и интеграция всех представлений.

По окончании данного этапа получаем концептуальную модель, инвариантную к структуре базы данных. Часто она представляется в виде модели «сущность-связь». Сущность – это существующие в действительности или воображаемые явления или объект, информацию о котором нужно сохранять или выяснять (обозначить существенным).

Каждая сущность должна иметь наименование, выраженное существительным в единственном числе. Примерами сущностей могут быть такие классы объектов, как «Безработные», «Вакансии», «Организации». Каждая сущность в модели изображается в виде прямоугольника с наименованием.

Атрибут (свойства) – именованный элемент информации, описывающий сущность. Атрибут может иметь только одно значение в каждый момент.

Наименование атрибута должно быть выражено существительным в единственном числе (возможно, с характеризующими прилагательными).

Атрибуты сущности «Безработные»:

Серия номер (первичный ключ)	Дата регистрации
Соц номер	Ведущий инспектор
Фамилия	Профессия(должность)
Имя	Стаж
Отчество	Пожелания требования
Дата рождения	Статус
Пол	

Атрибуты сущности «Вакансии»:

Номер_вакансии (первичный ключ)	Количество_мест
ИНН_организации (внешний ключ)	Количество_занято
Профессия	Образование
Стаж	Состояние_вакансии
Зарплата	Дата_регистрации
Пол	

Атрибуты сущности «Организации»:

ИНН_организации (первичный ключ)	Телефон
Наименование	Руководитель
Отрасль	Дата_регистрации_организации
Адрес	

Атрибуты сущности «ПоследниеМестаРаботы»

Идентификатор (первичный ключ)	Причина_увольнения
ИНН_организации (внешний ключ)	Общий_стаж_работы
Серия_номер (внешний ключ)	Средняя_зарплата
Должность	

Атрибуты характеристики «ПерсональныеДанные»:

Серия_номер (первичный ключ)	Семейное_положение
Страховой_номер	Количество_иждивенцев
Адрес	Образование
Телефон	Дополнительные_навыки
Гражданство	

Атрибуты обозначения «Инспектор»:

Номер_инспектора (первичный ключ)	ФИО
-----------------------------------	-----

Атрибуты обозначения «СтатусБезработного»:

Идентификатор (первичный ключ)	Статус
--------------------------------	--------

Исходя из имеющихся данных, становится возможным построение ER-диаграммы (рис. 1), необходимой для дальнейшего проектирования базы данных «Биржа труда».

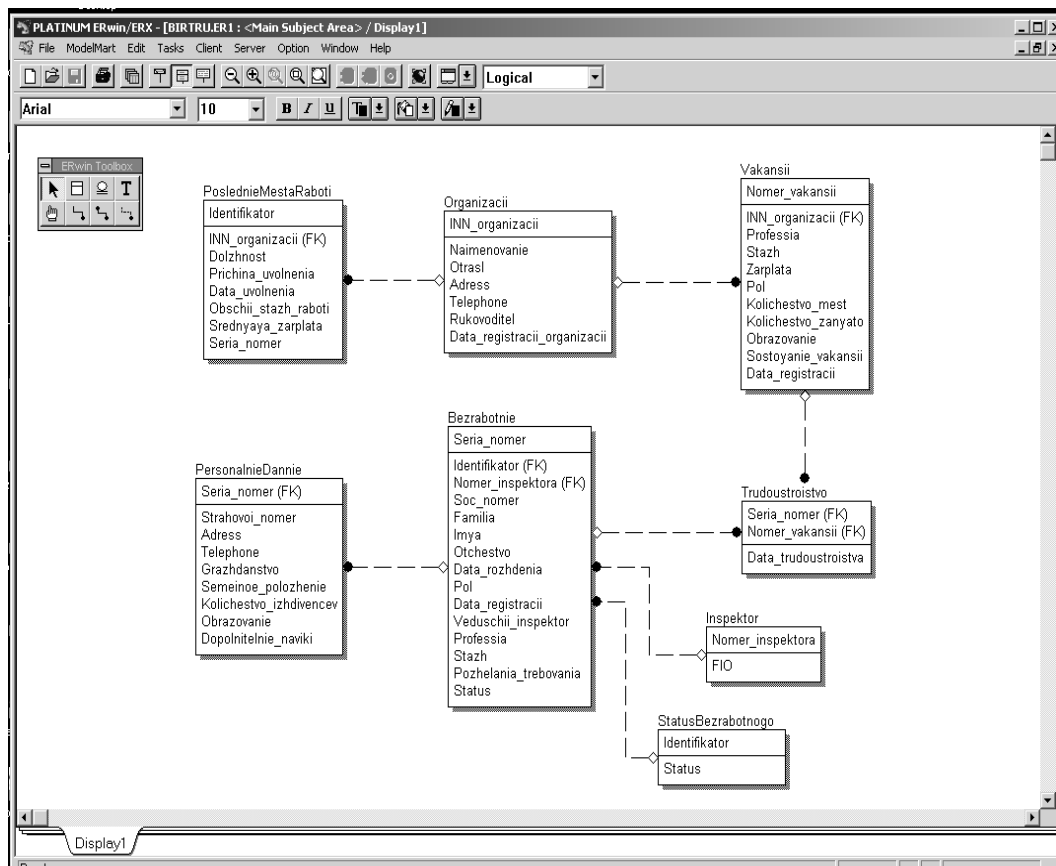


Рис. 1. Логическая схема

Заключение

Полученная логическая модель данных описывает понятия предметной области, их взаимосвязь, а также ограничения на данные, налагаемые предметной областью. Логическая модель данных является начальным прототипом будущей базы данных.

ЛИТЕРАТУРА

1. Хусаинова Г.Я., Хусаинов И.Г. Разработка автоматизированного рабочего места менеджера по продаже автомобилей // *Фундаментальные и прикладные исследования в современном мире*. – 2014. – Т. 1. – № 7. С. 126–128.
2. Хусаинова Г.Я., Хусаинов И.Г. Информационные системы. Управление реляционными базами данных: Учеб.-метод. материалы. – Стерлитамак: Стерлитамак, гос. пед. академия, 2007. – 55 с.
3. Хусаинов И.Г. Тепловые процессы при акустическом воздействии на насыщенную жидкостью пористую среду // *Вестник Башкирского университета*. – 2013. – Т. 18. – № 2. – С. 350–353.

СЕКЦИЯ II. ПРИКЛАДНОЙ СТАТИСТИЧЕСКИЙ АНАЛИЗ

ABC-XYZ АНАЛИЗ АССОРТИМЕНТА ПРЕДПРИЯТИЯ Г. ТОМСКА ПО СЛУЧАЙНО ЦЕНЗУРИРОВАННЫМ ДАННЫМ

Ж.Н. Зенкова, А.С. Охрименко

Томский государственный университет

zhanna.zenkova@mail.tsu.ru, aleksandraohrimenko97@mail.ru

Введение

Для анализа товарного ассортимента в зависимости от его прибыльности и стабильности спроса практически каждое производственное и торговое предприятие использует ABC-XYZ анализ, базирующийся на данных об осуществленных продажах за некоторый период [1–3]. При этом очевидно, что эти данные не отображают реального спроса на товар, т.к. не учитывают, например, ситуации возникновения дефицита, когда все имеющиеся в наличии изделия были проданы и некоторым покупателям товар просто не достался. В этом случае аналитики фактически работают с цензурированными справа данными, которые требуют особого подхода при их статистической обработке [4–8], обычные алгоритмы при этом приводят к серьёзным неточностям, а значит, к прямым убыткам.

В данной работе рассмотрена модификация ABC-XYZ анализа на случай присутствия случайного цензурирования справа [4,5] в исходных данных. При оценивании среднего уровня продаж и среднеквадратического отклонения применялся метод подстановки [9], в качестве оценки функции распределения использовалась непараметрическая оценка Каплана – Мейера [4,5].

Модифицированный метод апробировался на реальных данных о продажах томского производственного предприятия.

1. ABC-XYZ анализ

Метод ABC-анализа базируется на принципе Парето и позволяет классифицировать товарный ассортимент на три группы в зависимости от степени важности для предприятия, критерием классификации при этом может выступать прибыль или выручка от продажи M товаров R_i , $i = \overline{1, M}$, за рассматриваемый период [1].

Алгоритм классического ABC-анализа состоит из четырёх этапов:

1. Исходные данные сортируются в порядке убывания: $R_1 \geq R_2 \geq \dots \geq R_M$.

2. Для $i = \overline{1, M}$ вычисляется доля каждого показателя в общем итоге $d_i = \frac{R_i}{\sum_{i=1}^M R_i}$.

3. Для $i = \overline{1, M}$ рассчитывается кумулятивная доля $S_i = S_{i-1} + d_i$, $S_0 = 0$.

4. Принятие решения:

- если $S_i \leq 0.8$, то товар принадлежит группе A – наиболее значимой для предприятия, дефицит товаров этой группы ведёт к прямым убыткам, потому она требует к себе максимально пристального внимания;
- если $0.8 < S_i \leq 0.95$, то товар относят к менее значимой группе B , подходы к управлению которой менее затратны, чем к A ;
- если $S_i > 0.95$, то это товар наименее значимой группы C , который может быть даже исключён из ассортимента.

Методология XYZ- анализа основана на коэффициенте вариации:

$$CV = \frac{S}{\bar{X}} \cdot 100\%, \quad (1)$$

где

$$\bar{X} = \frac{1}{N} \cdot \sum_{i=1}^N X_i \quad (2)$$

– выборочное среднее, $X = (X_1, X_2, \dots, X_N)$ – выборка, состоящая из объёма продаж продукта, N – размер выборки,

$$S = \sqrt{\frac{1}{N} \cdot \sum_{i=1}^N (X_i - \bar{X})^2} \quad (3)$$

– выборочное среднеквадратическое отклонение.

Коэффициент вариации (1) рассчитывается для каждого товара, и после этого весь ассортимент группируется следующим образом:

- если $CV \leq 10\%$, то товар принадлежит группе X – группе со стабильным, практически постоянным спросом;
- если $10\% < CV \leq 30\%$, то товар принадлежит группе Y , где спрос нестабилен;
- если $CV > 30\%$, то товар принадлежит группе Z . Здесь часто оказывается спрос с ярко выраженной сезонностью.

Заметим, что границы групп Y и Z могут меняться, например, 30% могут быть изменены до 25% в зависимости от мнения экспертов.

2. Модификация ABC-XYZ анализа для случайно цензурированных данных

Рассмотрим случайную величину (с.в.) $\xi \geq 0$ – уровень продаж с функций распределения (ф.р.) $F(x) = P\{\xi < x\}$, которая является неизвестной. Её оценка по полной выборке – эмпирическая функция распределения (э.ф.р.) – позволяет путём метода подстановки [9] получить оценки (2) и (3).

Здесь рассмотрим случайно цензурированную справа выборку ц.в. $(X, I) = \{(X_1, I_1), \dots, (X_N, I_N)\}$ объёма N , где для $i = \overline{1, N}$

$$I_i = \begin{cases} 0, & X_i - \text{полное наблюдение;} \\ 1, & X_i - \text{наблюдение до цензурирования.} \end{cases}$$

Тогда в качестве оценки ф.р. $F(x)$ можно использовать $F_N^C(x) = 1 - S_N^C(x)$, где

$$S_N^C(x) = \prod_{j=2, \dots, N: x \in (X_{(j-1)}; X_{(j)})} \left(\frac{N - R_{(j)}}{N - R_{(j)} + 1} \right)^{1 - I_{(j)}} \quad - \text{оценка функции выживания Каплана -}$$

Мейера [4,5], $X_{(1)} \leq X_{(2)} \leq \dots \leq X_{(N)}$ – упорядоченные значения ц.в., $R_{(j)}$ – ранг $X_{(j)}$, $j = \overline{1, N}$, $S_N^C(x) = 1$ для $x \leq X_{(1)}$. Известно, что $S_N^C(x)$ – непараметрическая и состоятельная оценка функции выживания $S(x) = 1 - F(x)$ [4,5].

Методом подстановки [9] найдём оценки математического ожидания и среднеквадратического отклонения для случайно цензурированной выборки:

$$\bar{X}^C = \int_0^{\infty} x dF_N^C(x) = \sum_{i=1}^N X_i \bar{I}_i h_i, \quad S^C = \sqrt{\sum_{i=1}^N (X_i - \bar{X}^C)^2 \bar{I}_i h_i},$$

$$h_i = S_N^C(X_{(i-1)}) - S_N^C(X_{(i)}), \quad i = \overline{1, N}.$$

Оценка \bar{X}^C используется при перерасчёте итоговой выручки от каждого товара, для которого наблюдалось цензурирование, следующим образом: $R^C = N \cdot \bar{X}^C$, полу-

ченные значения выручки рассматриваются как базовые для повторной группировки согласно описанному выше алгоритму ABC-анализа.

Для модификации XYZ-анализа используется коэффициент вариации, рассчитанный для случайно цензурированных данных по формуле $CV = \frac{S^c}{\bar{X}^c} \cdot 100\%$. Принцип группировки при этом остается прежним.

3. Анализ товарного ассортимента производственного предприятия г. Томска

Предложенный метод применялся при анализе ассортимента крупного производственного предприятия г. Томска. Рассматривались данные о месячных продажах 35 товаров за 2017 г., при этом в разное время по каждой позиции наблюдался недостаток товара при заключении договоров на рынке B2B, в итоге региональные дистрибуторы скупали только тот объём, который имелся в наличии.

Традиционный подход, не учитывающий фактор дефицита, привёл к результатам, представленным в табл. 1.

Таблица 1

Результат ABC-XYZ анализа товарного ассортимента без учёта дефицита

	A	B	C
X	5	–	–
Y	23	–	–
Z	7	–	–

Общая выручка предприятия при этом составила 94 607 301.14 у.е./г. Учёт факта наличия дефицита поменял статус нескольких товаров (см. табл. 2). Более подробно расчеты рассмотрены на примере анализа продаж одного из товаров (см. рис. 1 и табл. 3).

Таблица 2

Результат ABC-XYZ анализа товарного ассортимента без учёта дефицита

	A	B	C
X	5	–	–
Y	24	–	–
Z	6	–	–

Таблица 3

Расчёты показателей для одного из товаров

Статус	Цена, у.е./ед.	Итого, ед./г.	Средняя продажа, ед./мес.	S	CV, %	Выручка, у.е./г.	d_i	ABC	XYZ
Без учета дефицита	42	37667,08	3 138,92	294,28	9.40	1582017	0,016720	A	X
С учетом дефицита	42	40378,08	3 364,84	511,27	15.19	1695880	0,015978	A	Y

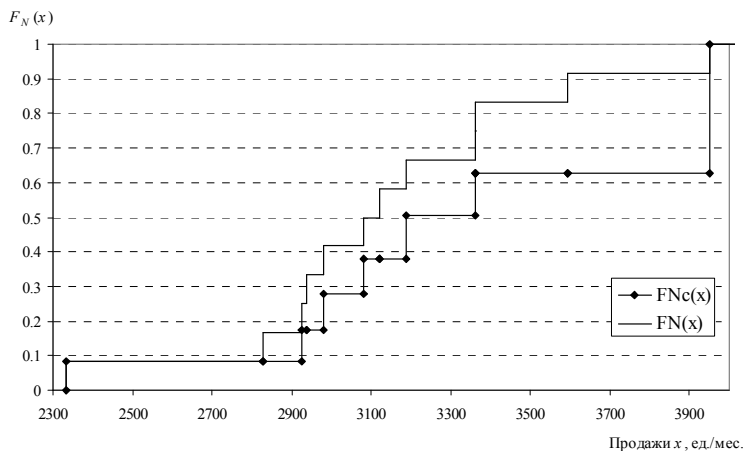


Рис. 1. Оценки функции распределения без учета цензурирования ($F_N^c(x)$) и с учётом цензурирования ($S_N^c(x)$) для одного товара

Общая выручка предприятия при условии, что спрос был бы удовлетворён полностью, составила бы 106 136 590.2 у.е./г. Таким образом, допустив дефицит, предприятие фактически потеряло около 12 млн. у.е. выручки за год.

Заключение

В работе предложен способ учёта дефицита продаж (фактически, случайного цензурирования спроса справа) при проведении ABC-XYZ анализа. Показано, что модифицированный метод даёт другие результаты группирования, которые нужно рассматривать как более адекватные, т.к. при модификации учитывалась природа реальных данных, что приводит к более точным результатам.

Таким образом, модифицированный метод ABC-XYZ анализа позволил показать, что дефицит привёл к существенным потерям в выручке (более 10%), изменениям статуса товаров в ABC-XYZ классификации, а значит, к пересмотру производственной и закупочной политики в отношении каждого наименования.

Предприятию даны рекомендации по системам управления товарными запасами.

ЛИТЕРАТУРА

1. Гаджинский А.М. Логистика: учеб. пос., 17-е изд., перераб. и доп. – М.: Дашков и К, 2008. – 484 с.
2. Зенкова Ж.Н. Логистический подход в управлении предприятием. Учебно-методический комплекс. – Томский государственный университет, 2012.
3. Зенкова Ж.Н., Стебловская А.С. ABC-, XYZ-анализ спроса на слабоалкогольные напитки // EDUCATIO. Новосибирск, 2016. – № 7(25)–8(26). С. 26–29.
4. Скрипник В.М., Назин А.Е., Приходько Ю.Г., Благовещенский Ю.Н. Анализ надежности технических систем по цензурированным выборкам. – М.: Радио и связь, 1988. – 184 с.
5. Тихов М.С., Бородин Т.С. Эконометрические модели с цензурированными данными: Учебно-методическое пособие – Нижний Новгород: Нижегородский госуниверситет, 2012. – 50 с.
6. Зенкова Ж.Н., Гаман М.И. Модификация методов ABC-, XYZ-анализа на случай цензурированных данных // Материалы VII Международной конференции «Логистические системы в глобальной экономике», 16–17 марта 2017 г., Красноярск. – Красноярск, 2017. – С. 150–153.
7. Зенкова Ж.Н., Краковецкая И.В. Моделирование по неполным данным в логистике и маркетинге // Логистические системы в глобальной экономике: материалы Междунар. науч.-практ. конф. (14–15 марта 2013 г., Красноярск): в 2 ч. Ч. 1. Научно-исследовательский сектор / Сиб. гос. аэрокосмич. ун-т. – Красноярск, 2013. – С. 98–105.
8. Зенкова Ж.Н., Краковецкая И.В. Непараметрическая оценка Тёрнбулла для интервально-цензурированных данных в маркетинговом исследовании спроса на биоэнергетические напитки // Вестник томского государственного университета. Управление, вычислительная техника и информатика. – 2013. № 3(24). – С. 64–69.
9. Боровков А.А. Математическая статистика – Новосибирск: Наука; Изд-во Института математики, 1997. – 772 с.

АВС-АНАЛИЗ ПРИ НАЛИЧИИ ДЕФИЦИТА

Ж.Н. Зенкова, А.Н. Козлов

Томский государственный университет
zhanna.zenkova@mail.tsu.ru, kozlov_alexey96@mail.ru

Введение

На сегодняшний день процедура АВС-анализа пользуется весьма большой популярностью, особенно в маркетинге и логистике, в частности, при анализе ассортимента и в управлении запасами [1–3].

АВС-анализ – это ранжирование ассортимента по разным параметрам. Он основан на правиле Парето (20/80), согласно которому 20% товарных позиций (группа А) обеспечивают предприятию 80% выручки или прибыли. Ранжировать таким образом можно и поставщиков, и складские запасы, и покупателей. Результатом АВС-анализа является группировка объектов по степени влияния на общий результат. Товары, клиенты, поставщики, попавшие в группу А, являются наиболее значимыми для организации.

Зачастую аналитики и менеджеры совершают ошибки, если в анализируемых данных имеются цензурированные значения. Такая проблема возникает, если имеется информация о дефиците какого-либо товара, объём продаж которого указывается в базах данных как константа, в то время как реальный спрос на него был заведомо больше – покупатели приобрели всё, что было на полках, и некоторым из них товара просто не хватило. Игнорируя факт наличия дефицита и проводя АВС-анализ так же, как и при полных данных, аналитики сталкиваются с недооценкой размера поставки нужного товара в определенной группе, а это в свою очередь влечёт к финансовым потерям в связи с неправильными подходами к закупкам и организации хранения ассортимента.

1. Метод АВС-анализа

Классический метод АВС-анализа применяется согласно следующему алгоритму: рассматривается прибыль или выручка от продажи M товаров R_i , $i = \overline{1, M}$, за некоторый период [1], далее:

1. Исходные данные сортируются в порядке убывания выручки: $R_1 \geq R_2 \geq \dots \geq R_M$.

2. Для $i = \overline{1, M}$ вычисляется доля каждого показателя в общем итоге $d_i = \frac{R_i}{\sum_{i=1}^M R_i}$.

3. Для $i = \overline{1, M}$ рассчитывается кумулятивная доля $S_i = S_{i-1} + d_i$, где $S_0 = 0$.

4. Если $S_i \leq 0.8$, то это товар из наиболее значимой группы А, дефицит в которой недопустим и ведёт к потере прибыли; если $0.8 < S_i \leq 0.95$, то товар относят к менее значимой группе В; если $S_i > 0.95$, то это товар наименее значимой группы С, вносящий минимальный вклад в общую выручку или прибыль, дефицит такого товара вполне допустим.

Пусть в течение исследуемого периода цены на товары p_j не менялась, при этом рассмотрим ξ_j , $j = \overline{1, M}$ – уровень спроса на j -й товар – как случайные величины (с.в.) с функциями распределения (ф.р.) $F_j(x) = P(\xi_j < x)$, $X = (X_{1j}, X_{2j}, \dots, X_{Nj})$ – выборка объёма N – значения продаж товара j -го за N дней (недель или месяцев). Тогда выручка от продажи j -го товара может быть представлена как

$$R_j = p_j N \overline{X_j}, \quad (1)$$

где $\bar{X}_j = \frac{1}{N} \sum_{i=1}^N X_{ij}$ – средний уровень продаж j -го товара, который фактически получается путём подстановки [4] эмпирической функции распределения (э.ф.р.) в функционал математического ожидания $M\xi_j = \int_{-\infty}^{\infty} x dF_j(x)$.

2. ABC-анализ по цензурированным данным без учета цензурирования

Пусть в имеющихся данных о продажах имеются цензурированные значения [5–8], т.е. наблюдался дефицит по некоторым позициям. Проиллюстрируем подход к расчетам на примере одного товара (опустим индекс j). При этом выборка принимает вид $(X, I) = \{(X_1, I_1), \dots, (X_N, I_N)\}$, где для $i = \overline{1, N}$

$$I_i = \begin{cases} 0, & X_i \text{ – полное наблюдение (дефицита не было),} \\ 1, & X_i \text{ – наблюдение до цензурирования, } X_i = T_1 \text{ (дефицит был),} \end{cases}$$

T_1 – значение цензуры.

Для простоты рассмотрим следующую схему цензурирования: количество неполных наблюдений в интервале $(T_1, +\infty)$ – с.в., численно равная доле g , $0 < g < 1$, от числа наблюдаемых объектов в конце интервала $[0, T_1]$, T_1 не случайно.

Классически фактор дефицита никак не учитывается при классификации, при этом фактически значение цензуры рассматриваются как полное наблюдение и в качестве оценки ф.р. используется

$$F_{N, T_1}(t) = \frac{1}{N} \sum_{i=1}^N C(x - X_i), \quad (2)$$

которая обладает свойствами [9, 10]:

$$\lim_{N \rightarrow \infty} MF_{N, T_1}(t) = \begin{cases} F(t), & t \leq T_1, \\ g + F(t)(1-g), & t > T_1, \end{cases}$$

$$\sigma_{N, T_1}^2(t) = \lim_{N \rightarrow \infty} NDF_{N, T_1}(t) = \begin{cases} F(t)(1-F(t)), & t \leq T_1, \\ \frac{(1-g)(1-F(t))}{1-p} (1-p(1-g))(F(t)-pg), & t > T_1. \end{cases}$$

Оценка математического ожидания (м.о.) (в данном случае – средней продажи) при этом определяется методом подстановки (2) в функционал м.о. следующим образом:

$$\bar{X}_{N, T_1} = \frac{1}{N} \sum_{i=1}^N X_i. \quad (3)$$

Исследуем м.о. оценки (3). Рассмотрим

$$\begin{aligned} M\bar{X}_{N, T_1} &= M\left(M\left(X_{N, T_1} | r\right)\right) = M\left(\frac{(N-r)(1-g)}{N} M\left(X_i I_{(X_i > T_1)}\right)\right) + \\ &+ M\left(\frac{r}{N} M\left(X_i I_{(X_i \leq T_1)}\right)\right) + M\left(\frac{(N-r)g}{N} T_1\right) = \\ &= M\left(\frac{(N-r)(1-g)}{N} \int_{-\infty}^{\infty} x I_{(x > T_1)} d\frac{F(x)-p}{1-p} + \frac{r}{N} \int_{-\infty}^{\infty} x I_{(x \leq T_1)} d\frac{F(x)}{p} + \frac{(N-r)g}{N} T_1\right), \end{aligned}$$

где r – количество полных наблюдений в промежутке $[0, T_1]$, $r \sim Bi(N, p)$, $p = F(T_1)$.

Зная, что $Mr = pN$, получаем:

$$\begin{aligned} M\bar{X}_{N,T_1} &= \frac{(N-Np)(1-g)}{N(1-p)} \int_{T_1}^{\infty} x dF(x) + \int_{-\infty}^{T_1} x dF(x) + (1-p)gT_1 = \\ &= M\xi - g \int_{T_1}^{\infty} x dF(x) + (1-p)gT_1. \end{aligned}$$

Таким образом, оценка (3) смещённая и даёт неточный результат при расчёте.

3. Модификация с помощью оценки Каплана – Мейера

Рассмотрим непараметрическую оценку ф.р. Каплана – Мейера, которая может быть представлена в виде [7]:

$$F_N^{\Pi}(t) = \begin{cases} \frac{1}{N} \sum_{i=1}^N I_{[0,t)}(X_i) \bar{I}_i, & 0 \leq t \leq T_1, \\ \frac{r}{N} + \frac{1}{(1-g)N} \sum_{i=1}^N I_{[T_1,t)}(X_i) \bar{I}_i, & (N-r)(1-g) > 0, T_1 < t, \\ \frac{r}{N}, & (N-r)(1-g) = 0, T_1 < t. \end{cases} \quad (4)$$

Оценка м.о., как и для (3) при этом определяется методом подстановки (4) в функционал м.о. по следующей формуле:

$$\bar{X}_N^{\Pi} = \frac{1}{N} \sum_{i=1}^N X_i \bar{I}_i I_{[0,T_1]}(X_i) + \frac{1}{(1-g)N} \sum_{i=1}^N X_i \bar{I}_i I_{(T_1,+\infty)}(X_i). \quad (5)$$

Исследуем м.о. оценки (5), получим:

$$\begin{aligned} M\bar{X}_N^{\Pi} &= M\left(M\left(X_N^{\Pi} \mid r\right)\right) = \\ &= M\left(\frac{r}{N} M\left(X_i I_{(X_i \leq T_1)}\right) \mid r = \overline{1, N-1}\right) + M\left(\frac{(N-r)(1-g)}{N(1-g)} M\left(X_i I_{(X_i > T_1)}\right) \mid r = \overline{1, N-1}\right) + \\ &+ M\left(\frac{(N-0)(1-g)}{N(1-g)} M\left(X_i I_{(X_i > T_1)}\right) \mid r = 0\right) + M\left(\frac{N}{N} M\left(X_i I_{(X_i \leq T_1)}\right) \mid r = N\right) = \\ &= M\left(\frac{r}{N} M\left(X_i I_{(X_i \leq T_1)}\right) \mid r = \overline{1, N}\right) + M\left(\frac{N-r}{N} M\left(X_i I_{(X_i > T_1)}\right) \mid r = \overline{0, N-1}\right). \end{aligned}$$

Зная, что $Mr = pN$, будем иметь:

$$M\bar{X}_N^{\Pi} = p \int_{-\infty}^{T_1} x \frac{dF(x)}{p} + (1-p) \int_{T_1}^{\infty} x \frac{d(F(x)-p)}{1-p} = \int_{-\infty}^{T_1} x dF(x) + \int_{T_1}^{\infty} x d(F(x)-p) = M\xi,$$

из чего следует, что оценка (5) несмещённая, а значит, точнее (3).

Таким образом, при расчёте выручки (1) в методе АВС при наличии дефицита рекомендуется использовать среднюю продажу, полученную по формуле (5).

4. Пример

Рассмотрим модифицированный метод АВС-анализа на конкретном примере. Пусть имеются модельные данные о продажах, представленные в табл. 1. Результаты классического АВС-анализа представлены в табл. 2.

Пусть по товару «Товар 7» за февраль и май наблюдался дефицит – имевшихся в наличии 281 и 190 шт. не хватило, спрос остался не удовлетворенным. Осуществим перерасчет результатов АВС-анализа, используя описанный выше метод расчета.

Для рассматриваемого случая $N = 6$, т.к. продажи наблюдались в течение 6 месяцев, $g = 0,4$ (5) при этом принимает вид:

Таблица 1

Модельные данные о продажах

Товар, наименование	Цена у.е./шт.	Продажи, шт./мес.					
		Январь	Февраль	Март	Апрель	Май	Июнь
Товар 1	107	396	258	260	336	218	290
Товар 2	97	79	225	153	80	422	200
Товар 3	57	180	271	216	421	132	245
Товар 4	42	205	296	399	358	333	370
Товар 5	28	447	461	295	463	454	460
Товар 6	27	499	494	297	402	471	315
Товар 7	33	178	281	291	432	190	456
Товар 8	19	292	430	491	310	204	338
Товар 9	25	312	317	284	255	215	158
Товар 10	67	99	91	95	91	101	90

Таблица 2

Результаты ABC-анализа

Товар, наименование	Средняя продажа, шт./мес.	Выручка, у.е.	Доля выручки в общем итоге	Накопленная доля выручки в общем итоге, %	Группа
Товар 1	292,9	187466,6	0,240066368	24,01 %	A
Товар 2	192,9	111891,7	0,143286412	38,34 %	A
Товар 3	244,0	83936,0	0,107486899	49,08 %	A
Товар 4	326,4	82061,2	0,105086001	59,59 %	A
Товар 5	429,6	72341,8	0,092639622	68,86 %	A
Товар 6	412,8	67002,7	0,085802618	77,44 %	A
Товар 7	304,6	60673,0	0,077696729	85,21 %	B
Товар 8	343,8	39050,0	0,050006712	90,21 %	B
Товар 9	256,7	38705,3	0,049565339	95,16 %	C
Товар 10	94,4	37766,7	0,048363299	100,00 %	C
ИТОГО:		780 895,2			

$$\bar{X}^{\text{II}} = \frac{1}{6} \cdot 178 + \frac{1}{(1-0,4) \cdot 6} \cdot (291 + 432 + 456) = 357,1667 \text{ шт./мес.},$$

т.е. учет цензурирования позволил оценить средний спрос на товар «Товар 7» большим значением, в отличие обычного среднего – 304,6 шт./мес. (см. табл. 2). Пересчитаем новое значение выручки согласно (1): $R = 33 \cdot 6 \cdot 357,1667 = 71147,6$ у.е.

В итоге, при пересчёте результатов ABC-анализа, мы получили несколько другое ранжирование товаров (см. табл. 3), а именно: «Товар 7» попал в группу «А», а «Товар 6» попал в группу «В». Из этого следует, что для обоих товаров требуется пересмотр закупочной политики и системы управления запасами.

Таблица 3

Результаты ABC-анализа с учетом цензурирования

Товар, наименование	Средняя продажа, шт./мес.	Выручка, у.е.	Доля выручки в общем итоге	Накопленная доля выручки в общем итоге, %	Группа
Товар 1	292,9	187466,7	0,236889	23,69 %	A
Товар 2	192,9	111891,7	0,14139	37,83 %	A
Товар 3	244,0	83936,0	0,106064	48,43 %	A
Товар 4	326,4	82061,2	0,103695	58,80 %	A
Товар 5	429,6	72341,8	0,091413	67,95 %	A
Товар 7	357,2	71147,6	0,089904	76,94 %	A
Товар 6	412,8	67002,9	0,084667	85,40 %	B
Товар 8	343,8	39050,0	0,049345	90,34 %	B
Товар 9	256,7	38705,3	0,048909	95,23 %	C
Товар 10	94,4	37766,7	0,047723	100,00 %	C
ИТОГО:		791 369,8			

Заключение

В данной работе был рассмотрен метод ABC-анализа, модифицированный на случай наличия дефицита. Факт дефицита рассматривался как цензурирование, для оценки функции распределения и среднего уровня продаж применялась непараметрическая оценка Каплана – Мейера, на основе модельных данных было установлено, что использование модифицированного метода даёт другие результаты ранжирования, которые нужно рассматривать как более адекватные, поскольку при модификации использовались более точные оценки статистических характеристик.

ЛИТЕРАТУРА

1. *Гаджинский А.М.* Логистика: учеб. 17-е изд., перераб. и доп. – М.: Дашков и К, 2008. – 484 с.
2. *Зенкова Ж.Н.* Логистический подход в управлении предприятием. Учебно-методический комплекс. – Томский государственный университет, 2012.
3. *Зенкова Ж.Н., Стебловская А.С.* ABC-, XYZ-анализ спроса на слабоалкогольные напитки // EDUCATIO, Новосибирск, 2016. – № 7(25)–8(26). – С. 26–29.
4. *Боровков А.А.* Математическая статистика. – Новосибирск: Наука, 1997. – 772 с.
5. *Зенкова Ж.Н., Гаман М.И.* Модификация методов ABC-, XYZ-анализа на случай цензурированных данных // Материалы VII Международной конференции «Логистические системы в глобальной экономике», 16–17 марта 2017 г., Красноярск. – Красноярск, 2017. – С. 150–153.
6. *Зенкова Ж.Н., Краковецкая И.В.* Моделирование по неполным данным в логистике и маркетинге // Логистические системы в глобальной экономике: материалы Международной научно-практической конференции, 2013. – С. 98–105.
7. *Скрипник В.М., Назин А.Е., Благовещенский Ю.Г.* Анализ надежности технических систем по цензурированным выборкам. – М.: Радио и связь, 1988. – 184 с.
8. *Зенкова Ж.Н., Краковецкая И.В.* Непараметрическая оценка Тёрнбулла для интервально-цензурированных данных в маркетинговом исследовании спроса на биоэнергетические напитки // Вестник томского государственного университета. Управление, вычислительная техника и информатика. – 2013. – № 3(24). – С. 64–69.
9. *Дмитриев Ю.Г., Зенкова Ж.Н.* Об одной оценке симметричной функции распределения по цензурированной выборке // Материалы IV Всеросс. науч.-практ. конф. "Информационные технологии и математическое моделирование", Анжержо-Судженск, 2005. – Томск: Изд-во Том. ун-та, 2005. – Ч.2. – С. 8–10.
10. *Зенкова Ж.Н.* Учет информации об S^a -равноплечной симметрии при обработке цензурированных данных // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – Томск: ТГУ, 2011. – № 2(15). – С. 32–45.

МЕТОД ЦЕНООБРАЗОВАНИЯ ШТОЦЕЛЯ ДЛЯ СЛУЧАЙНО ЦЕНЗУРИРОВАННЫХ ДАННЫХ

А.А. Андриевская, Ж.Н. Зенкова

Томский государственный университет
annaandrievskaya@mail.ru, zhanna.zenkova@mail.tsu.ru

Введение

Вопрос о назначении цены – стратегическая задача каждого коммерческого предприятия, т.к. цена является базовым фактором при принятии потребителем решения о покупке. От конечной цены продажи напрямую зависит прибыль предприятия, а также уровень спроса на товары и услуги. Обычно в процессе ценообразования предприятие ориентируется на затраты и целевой уровень прибыли, при этом рассчитанная цена может показаться потребителю несправедливой. Для решения этой проблемы используются методы ценообразования, ориентированные на спрос, которые позволяют рассчитать предпочитаемый потребителями ценовой диапазон.

В данной работе рассматривается метод ценообразования Штоцеля [1]. Он достаточно прост и относительно дешёв и нашел широкое применение в зарубежной маркетинговой практике, однако редко применяем российскими маркетологами. Статистические свойства метода практически не изучены.

В работе с помощью имитационного моделирования для случайно цензурированных данных исследуются свойства рассматриваемого метода, анализируются матема-

тическое ожидание и среднеквадратическое отклонение, а также функции распределения ценовых границ.

1. Метод Штоцеля на случай цензурированных данных

Рассматриваемый метод был предложен французским психологом и социологом Жаном Штоцелем в середине XX в. [1]. Этот метод очень прост и состоит в следующем. После того, как представители целевой аудитории пользовались товаром в течение некоторого периода времени, им предлагают ответить на следующие два вопроса:

1. Ниже какого уровня цены P_1 товар кажется вам настолько дешёвым, что возникает подозрение о его низком качестве и/или поддельности?
2. Начиная с какого уровня цены P_2 товар кажется настолько дорогим, что вопрос о покупке даже не стоит?

В результате получаем независимую двумерную выборку (P_{i1}, P_{i2}) , $i = \overline{1, N}$, где N – количество опрошенных потребителей. При этом цены P_j , $j = \overline{1, 2}$, рассматриваются как случайные величины (с.в.) с функциями распределения (ф.р.) $F^{(j)}(p) = P(P_j < p)$. Далее каждый элемент выборки обрабатывается отдельно, по каждому вектору $\{P_{ij}\}$, $j = \overline{1, 2}$, строится эмпирическая функция распределения (э.ф.р.) по формуле

$$F_N^{(j)}(p) = \frac{1}{N} \sum_{i=1}^N I_{(0, p]}(P_{ij}), \text{ где}$$

$$I_{(0, p]}(y) = \begin{cases} 0, & y \notin (0, p], \\ 1, & y \in (0, p] \end{cases}$$

– индикаторная функция. Э.ф.р. – кусочно-непрерывная функция [2], и в методе Ж. Штоцеля её сглаживают.

Искомые цены P_{\min} и P_{\max} находят из уравнения $d(p) = F_N^{(1)}(p) - F_N^{(2)}(p) = 0.5$.

Пример реализации метода Штоцеля приведен на рис. 1.

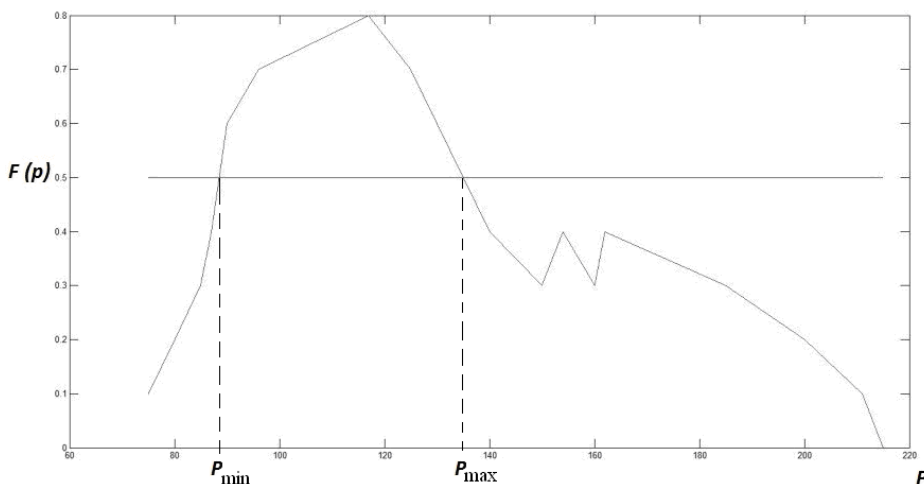


Рис 1. Пример реализации метода Штоцеля

Заметим, что назначать цену ниже P_{\min} и больше P_{\max} нецелесообразно – при слишком низкой цене основная масса потенциальных покупателей откажется от покупки, подозревая низкое качество, а при слишком высокой отказ от покупки произойдет в связи с неплатежеспособностью целевой аудитории. Если предприятия нацелено на

широкий охват рынка и завоевание большей рыночной доли, то цену лучше выбрать ближе к P_{\min} , если же цель предприятия – завоевание репутации статусного товара, то цена должна быть высокой, ближе к P_{\max} .

2. Модификация метода Штоцеля для случайно цензурированных данных

На практике маркетолог часто сталкивается с ситуацией, когда потребитель затрудняется дать точные ответы о ценовых предпочтениях в анкете и вместо этого дает некий ценовой интервал. В итоге возникает необходимость в модификации метода ценообразования на случай неполных, цензурированных данных [3–12]. При этом выборка принимает вид $(P, I) = \{(P_1, I_1), (P_2, I_2), \dots, (P_N, I_N)\}$, где

$$I_i = \begin{cases} 0, & P_i \text{ цензурировано,} \\ 1, & P_i \text{ не цензурировано,} \end{cases} \quad i = \overline{1, N},$$

оценки ф.р. строятся по формуле $F_{N,C}^{(j)}(p) = 1 - S_N^{(j)}(p)$, где

$$S_N^{(j)}(p) = \prod_{i=2, N: p \leq P_{(i)}} \left(\frac{N - R_{(i)}}{N - R_{(i)} + 1} \right)^{1 - I_{(i)}} - \text{непараметрическая асимптотически несмещенная}$$

оценка функции выживания Каплана-Мейера [13], $P_{(i)}$ – i -я порядковая статистика, $i = \overline{1, N}$, $R_{(i)}$ – ранг $P_{(i)}$.

Метод Штоцеля в результате модифицируется следующим образом: итоговые ценовые значения P_{\min}^C и P_{\max}^C определяют из уравнения $d^C(p) = F_{N,C}^{(1)}(p) - F_{N,C}^{(2)}(p) = \frac{1}{2}$.

Для исследования качества модифицированного метода Штоцеля использовалось имитационное моделирование. Моделировались выборки из равномерных распределений $F_1(p) = R_{(0,100)}(p)$, $F_2(p) = R_{(110,210)}(p)$, параметр моделирования $M = 10\,000$, объём выборки N изменялся от 20 до 100. Каждая имеющаяся выборка подвергалась случайному цензурированию с заданной долей цензурирования Q в случайном «окне» цензурирования $[L, +\infty)$. Примеры реализации метода приведены на рис. 2–4.

Исследовалось поведение математического ожидания оценок границ ценовых диапазонов, а также их среднеквадратическая ошибка (СКО) в зависимости от разных параметров. Заметим, что истинные значения ценовых границ равны $P_{\min} = 50$, $P_{\max} = 160$.

По результатам моделирования можно сделать вывод о том, что с ростом доли цензурирования смещение и СКО растут, при сужении ширины окна цензурирования L усредненные ценовые значения сходятся к истине значительно быстрее, СКО при этом снижается (рис. 5–10).

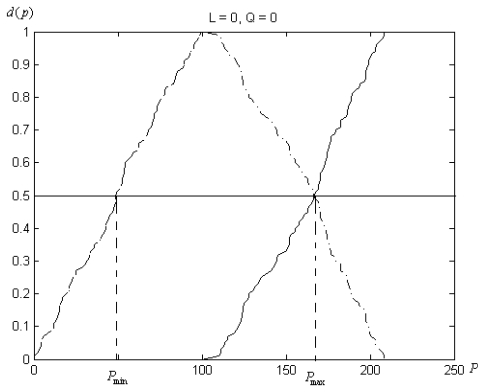


Рис 2. Пример реализации метода Штоцеля без цензурирования, $N = 100$

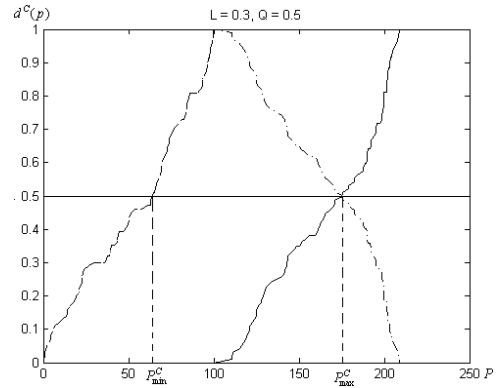


Рис 4. Пример реализации метода Штоцеля в случае цензурирования левой и правой границ при $L = 0.3$, $Q = 0.5$, $N = 100$

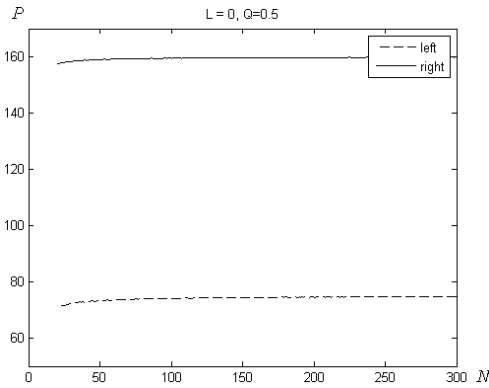


Рис 5. Усредненные цены в случае цензурирования левой ценовой границы в зависимости от N

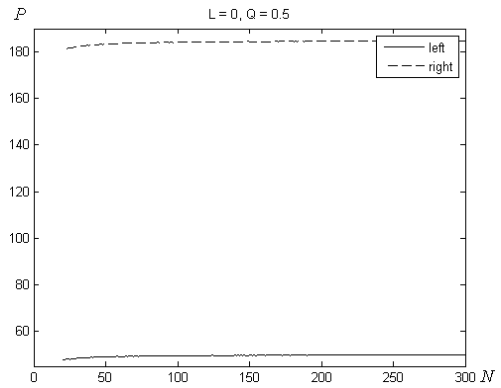


Рис 6. Усредненные цены в случае цензурирования правой ценовой границы в зависимости от N

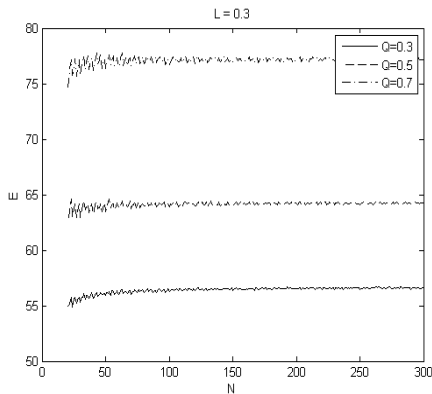


Рис 7. Усредненные цены цензурированной левой границы в зависимости от N для разных долей Q при $L = 0.3$

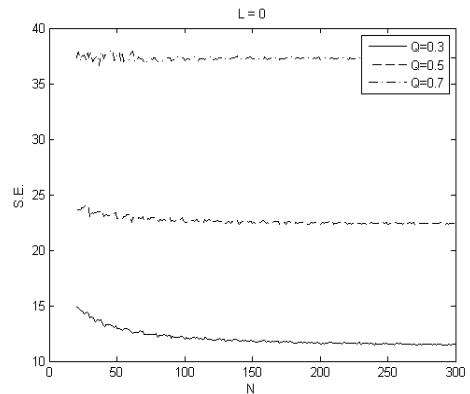


Рис 8. СКО цензурированной левой границы в зависимости от N для разных долей Q при $L = 0$

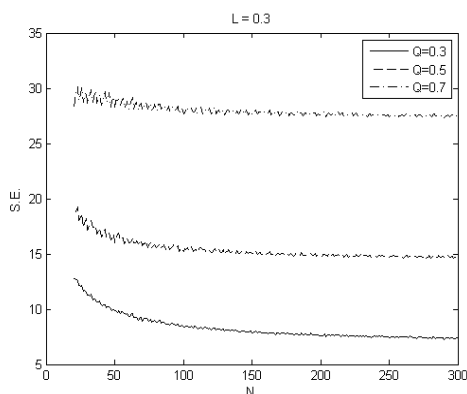


Рис 9. СКО цензурированной левой границы в зависимости от N для разных долей Q при $L = 0.3$

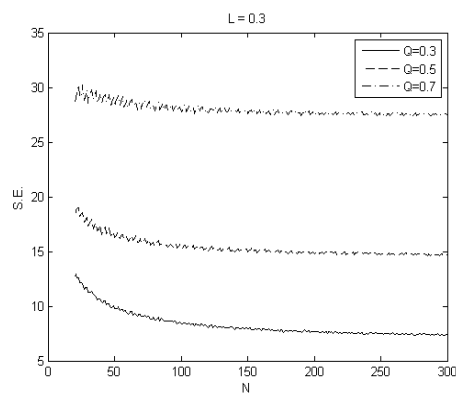


Рис 10. СКО цензурированной правой границы в зависимости от N для разных долей Q при $L = 0.3$

Заключение

В работе с помощью имитационного моделирования исследовались статистические свойства метода ценообразования Штоцеля для случайно цензурированных данных. Выявлено, что цензурирование негативно сказывается на качестве метода, приводя к возникновению смещения и росту среднеквадратического отклонения. Маркетологам рекомендуется по возможности избегать появления цензурирования.

ЛИТЕРАТУРА

1. *Stoetzel J. Psychological / Sociological Aspects of Price. // Pricing Strategy. Bernard Taylor and Gordon Wills (eds.). – Brandon Systems Press, Princeton, NJ, 1970. – P. 70–74.*
2. *Боровков А.А. Математическая статистика. – Новосибирск: Наука, 1997. – 772 с.*
3. *Зенкова Ж.Н., Краковецкая И.В. Исследование ожидаемой цены спроса на биоэнергетический напиток по интервально-цензурированным данным // Материалы Международной молодежной конференции «Биопротранство» в рамках фестиваля науки / отв. ред. А.М. Данченко. – Томск: Томский государственный университет, 2012. – С. 60–65.*
4. *Зенкова Ж.Н., Краковецкая И.В. Непараметрическая оценка Тёрнбулла для интервально-цензурированных данных в маркетинговом исследовании спроса на биоэнергетические напитки // Вестник томского государственного университета. Управление, вычислительная техника и информатика. – 2013. – № 3 (24). – С. 64–69.*
5. *Зенкова Ж.Н., Краковецкая И.В. Моделирование по неполным данным в логистике и маркетинге // Логистические системы в глобальной экономике: материалы Междунар. науч.-практ. конф. (14–15 марта 2013 г., Красноярск): в 2 ч. Ч. 1. Научно-исслед. сектор / Сиб. гос. аэрокосмич. ун-т. Красноярск, 2013. – С. 98–105.*
6. *Журко Е.С., Зенкова Ж.Н. Метод ценообразования PSM для случая цензурированных выборок малого объема // Труды Томского государственного университета. – Т. 299. Серия физико-математическая: Математическое и программное обеспечение информационных, технических и экономических систем : материалы IV Всероссийской молодежной научной конференции. Томск, 20–21 мая 2016 г. / под общ. ред. И.С. Шмырина. – Томск : Издательский Дом Томского государственного университета, 2016. – С. 47–51.*
7. *Зенкова Ж.Н., Цыбулькикова Н.Р. Исследование цены спроса на продукцию ЗАО «Натуральное мыло» методом Штоцеля по цензурированным данным // Труды Томского государственного университета. – Т. 299. Серия физико-математическая: Математическое и программное обеспечение информационных, технических и экономических систем : материалы IV Всероссийской молодежной научной конференции. Томск, 20–21 мая 2016 г. / под общ. ред. И.С. Шмырина. – Томск : Издательский Дом Томского государственного университета, 2016. – С. 62–65.*
8. *Журко Е.С., Зенкова Ж.Н. Моделирование модификации метода ценообразования PSM по цензурированным данным // Сборник трудов XIV Международной научно-практической конференции студентов, аспирантов и молодых учёных «Молодёжь и современные информационные технологии». Национальный исследовательский Томский политехнический университет, Институт кибернетики. 2017. – С. 128–130.*
9. *Журко Е.С., Зенкова Ж.Н. Метод ценообразования PSM для случая цензурированных выборок малого объема // Материалы VI Международной конференции «Логистические системы в глобальной экономике», 14–15 марта 2016 г., Красноярск: электрон. сб. / Сиб. гос. аэрокосмич. ун-т. – Красноярск, 2016. – С. 152–157.*
10. *Журко Е.С., Зенкова Ж.Н. Метод ценообразования PSM для цензурированных данных с учетом квантиля // СПб., Международный союз ученых «Наука. Технологии». – 2015. – № 9 (13). – С. 13–16.*
11. *Журко Е.С., Зенкова Ж.Н. Метод ценообразования PSM на основе цензурированных справа и слева данных // Труды Томского государственного университета. – Т. 297. Серия физико-математическая: Математическое и программное обеспечение информационных, технических и экономических систем : материалы III Всерос-*

сийской молодежной научной конференции. Томск, 22–23 мая 2015 г. / под общ. ред. И.С. Шмырина. – Томск : Издательский Дом Томского государственного университета, 2015. – С. 206–210.

12. *Зенкова Ж.Н., Копнова Е.Е., Бараксанов Д.Н.* Применение метода PSM и алгоритма Тёрнбулла при определении цены нового программного продукта // Высокие технологии, фундаментальные исследования, инновации: сб. статей XVII Междунар. научно-практ. конф. "Фундаментальные и прикладные исследования, разработка и применение высоких технологий в промышленности и экономике", 22–23 мая 2014 г., г. Санкт-Петербург, Россия. – СПб.: Изд-во Политехн. ун-та, 2014. – С. 109–112.

13. *Скрипник В.М., Назин А.Е., Приходько Ю.Г., Благоевский Ю.Н.* Анализ надежности технических систем по цензурированным выборкам. – М.: Радио и связь, 1988. – 184 с.

РАСЧЕТ ЧИСТОЙ НЕТТО-ПРЕМИИ ПО ОДНОКРАТНО СПРАВА ЦЕНЗУРИРОВАННОЙ ВЫБОРКЕ

Ж.Н. Зенкова, К.А. Серебренникова

Томский государственный университет
zhanna.zenkova@mail.tsu.ru, krisert@mail.ru

Введение

Страховые услуги являются одним из наиболее значимых элементов рыночной инфраструктуры. Страхование обеспечивает застрахованным надёжную защиту их интересов от финансовых рисков, неблагоприятных последствий аварий, различного рода стихийных бедствий и пр. [1].

В данной статье рассматривались два подхода к расчёту нетто-премии, а именно: оценивание среднего по полной выборке с использованием метода подстановки эмпирической функции распределения (э.ф.р.) в интеграл для среднего, по цензурированной выборке (прогрессивное цензурирование I типа) с использованием оценки Каплана – Мейера.

Рассматриваемые методики были апробированы на реальных данных о выплатах по автомобильному страхованию КАСКО некоторой страховой компании.

1. Оценивание чистой нетто-премии по цензурированной выборке

Чистая нетто-премия является базовой величиной при расчёте стоимости страхового полиса и определяется формулой [1–3]: $p = z \cdot E X$, где X – размер выплаты – случайная величина (с.в.) с функцией распределения (ф.р.) $F(t) = P(X < t)$, $E X$ – её математическое ожидание (м.о.), z – вероятность наступления страхового случая. На практике z и $E X$ точно не известны, их приходится оценивать по накопленным наблюдениям.

В итоге оценка чистой премии имеет вид $\hat{p} = \hat{z} \bar{X}$, где $\bar{X} = \frac{1}{N} \sum_{i=1}^N X_i$ – выборочное среднее, \hat{z} – оценка вероятности наступления страхового случая, равная соотношению числа страховых случаев к числу заключённых за период договоров.

Заметим, что выборочное среднее \bar{X} получено методом подстановки [4] э.ф.р.

$F_N(t) = \frac{1}{N} \sum_{i=1}^N C(t - X_i)$ в интеграл для математического ожидания:

$$\bar{X} = \int_{-\infty}^{+\infty} t dF_N(t) = \frac{1}{N} \sum_{i=1}^N X_i.$$

Известно [4], что э.ф.р. является несмещённой оценкой ф.р., её нормированная на N дисперсия $\sigma^2(t) = N dF_N(t) = F(t)(1 - F(t))$.

При обработке статистической информации часто приходится сталкиваться с данными, которые представляются в виде цензурированных выборок (ц.в.), когда имеет место неопределённость относительно знания результатов наблюдений, и известно, что

некоторые значения попали в определенные интервалы. В страховании такие ситуации возможны, когда оценщик указывает приблизительную стоимость ущерба или все последствия страхового случая не выявлены, а значит, не оплачены к моменту исследования.

При этом выборка приобретает вид $(X, I) = \{(X_1, I_1), \dots, (X_N, I_N)\}$, где

$$I_i = \begin{cases} 0, & X_i - \text{полное наблюдение,} \\ 1, & X_i - \text{цензурированное наблюдение,} \end{cases}$$

и актуарию фактически приходится работать с выборкой, цензурированной справа.

Обработка цензурированных данных требует особых подходов, достаточно трудоёмка, при этом возникает потеря информации, которая сказывается на качестве статистических процедур [5–13].

Рассмотрим наиболее простую для анализа ситуацию, когда цензура T_1 не случайна, количество неполных наблюдений в интервале $(T_1, +\infty)$ – с.в., численно равная доле g , $0 < g < 1$, от числа полных наблюдений, попавших в интервал $(T_1, +\infty)$. Тогда оценка ф.р. определяется по формуле Каплана – Мейера [5], которая может быть представлена в виде [5,6]

$$F_N^u(t) = \begin{cases} \frac{1}{N} \sum_{i=1}^N I_{[0,t)}(X_i) \bar{I}_i, & 0 \leq t \leq T_1, \\ \frac{r}{N} + \frac{1}{(1-g)N} \sum_{i=1}^N I_{[T_1,t)}(X_i) \bar{I}_i, & (N-r)(1-g) > 0, \\ r/N, & (N-r)(1-g) = 0, \end{cases} \quad T_1 < t < +\infty, \quad (1)$$

где для $i = \overline{1, N}$ $\bar{I}_i = 1 - I_i$, r – число полных наблюдений в интервале $[0, T_1]$, $I_A(x) = \{0 : x \notin A, 1 : x \in A\}$. Данная оценка является асимптотически несмещённой, её нормированная дисперсия определяется формулой [6]

$$\sigma_u^2(t) = \lim_{N \rightarrow \infty} NDF_N^u(t) = F(t)(1-F(t)) + \frac{g(F(t)-p)(1-F(t))}{(1-p)(1-g)} I_{[T_1, +\infty)}(t),$$

где $p = F(T_1) \in (0, 1)$, $DF_N^u(t)$ – дисперсия оценки $F_N^u(t)$. Заметим, что $\sigma_u^2(t) \geq \sigma^2(t)$.

На рис. 1 приведены графики $\sigma^2(t)$ и $\sigma_u^2(t)$ для случая $F(x) = R_{(0,1)}(x)$ – равномерного в $(0, 1)$ распределения с параметрами $p = 0.3$, $g = 0.6$.

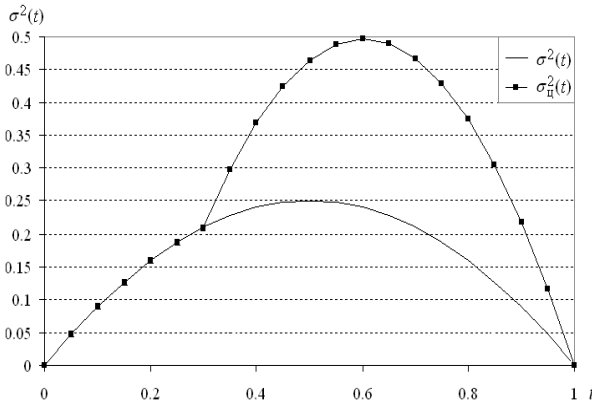


Рис. 1. Графики $\sigma^2(t)$ и $\sigma_u^2(t)$ для случая $F(x) = R_{(0,1)}(x)$ с $p = 0.3$, $g = 0.6$

Оценку среднего значения с.в. X можно найти с помощью метода подстановки [4]

$$\bar{X}^u = \int_0^T t dF_N^u(t) = \frac{1}{N} \sum_{i=1}^N X_i \cdot \bar{I}_i \cdot I_{[0, T_i)}(X_i) + \frac{1}{(1-g)N} \sum_{i=1}^N X_i \cdot \bar{I}_i \cdot I_{[T_i, +\infty)}(X_i), \quad (2)$$

где $F_N^u(t)$ – оценка (1).

Заметим, что при $g = 0$, т.е. при отсутствии цензурирования, получаем классическое выборочное среднее \bar{X} .

Оценка (2) является несмещённой, нетрудно получить, что её нормированная дисперсия в асимптотике определяется формулой $\varepsilon(g) = \lim_{N \rightarrow \infty} ND\bar{X}^u = \frac{DX \cdot (1-0.5g)}{1-g}$. Т.к.

$g \in (0,1)$, то $\frac{1-0.5g}{1-g} = \frac{2-g}{2(1-g)} \geq 1$, следовательно, $\varepsilon(g) \geq DX$.

На рис. 2 приведен график зависимости $\varepsilon(g)$ от g для $F(x) = R_{(0,1)}(x)$, когда $DX = \frac{1}{12}$. Заметим, что с ростом g дисперсия оценки (2) стремительно возрастает.

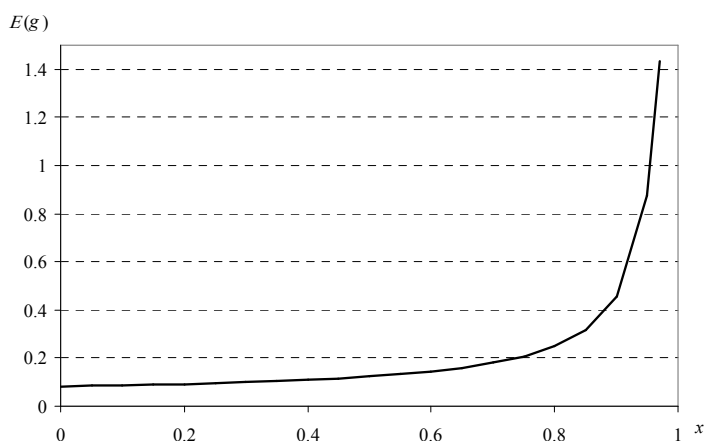


Рис. 2. График зависимости $\varepsilon(g)$ от g для $F(x) = R_{(0,1)}(x)$

Таким образом, при наличии цензурирования чистая нетто-премия может быть рассчитана по формуле $\hat{p}^u = \hat{z}\bar{X}^u$, при этом будет наблюдаться существенная потеря точности оценивания.

2. Расчет чистой нетто-премии в комплексном автомобильном страховании КАСКО по цензурированной выборке

Предлагаемая методика применялась для расчёта чистой нетто-премии в комплексном автомобильном страховании КАСКО для некоторой страховой компании. Рассматривались данные об ущербе для $N = 820$ страховых случаев, при этом было получено, что $\hat{z} = 0.183$.

При работе с полными выборками средний ущерб $\bar{X} = 79\,230.55$ у.е./ед., чистая нетто-премия составила $p = 14\,499.19$.

Затем данные были искусственно процензурированы, цензура $T_1 = 50\,000$ у.е./ед., доля $g = 0.2046$ (были процензурированы 71 из 347 значений, превышающих T_1). На

рис. 3 приведены графики э.ф.р. и оценки Каплана – Мейера (КМ), построенные на основе рассматриваемых выборок.

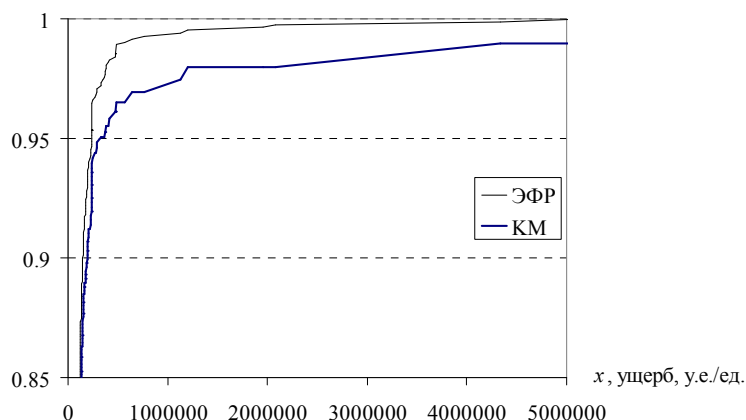


Рис. 3. Графики э.ф.р. и оценки Каплана – Мейера

В результате расчетов получили, что $\bar{X}^u = 171\,905.62$ у.е./ед., чистая нетто премия с учетом цензурирования $p^u = 31\,458.73$. Заметим, что наличие цензурирования привело к существенной переоценке нетто-премии, а значит, и к повышению цены страхового полиса.

Заключение

В данной работе показано, что наличие цензурирования существенно снижает точность оценивания чистой нетто-премии, которая является базой для расчёта стоимости страхового полиса. На примере реальных данных об убытках по комплексному автомобильному страхованию КАСКО проиллюстрировано, что цензурирование привело более чем к двукратному увеличению чистой нетто-премии.

Актуариям страховой компании рекомендовано по возможности избегать цензурирования и использовать в расчётах максимально полные данные.

ЛИТЕРАТУРА

1. Миронкина Ю.Н., Сорокин А.С. Основы актуарных расчетов. – М.: Изд. Центр ЕАОИ, 2011. – 284 с.
2. Фалин Г.И., Фалин А.И. Теория риска для актуариев в задачах. – М.: Мир, Научный мир, 2004.
3. Зенкова Ж.Н., Крайнова Е.А. Оценка чистой нетто-премии с использованием дополнительной информации о квантиле функции распределения // Бизнес-информатика. – 2017. – № 4 (42). – С. 55–63.
4. Боровков А.А. Математическая статистика. – Новосибирск: Наука; Институт математики, 1997. – 772 с.
5. Скрипник В.М., Назин А.Е., Благовещенский Ю.Г. Анализ надежности технических систем по цензурированным выборкам. – М.: Радио и связь, 1988. – 184 с.
6. Зенкова Ж.Н., Гаман М.И. Модификация методов ABC-, XYZ-анализа на случай цензурированных данных // Материалы VII Международной конференции «Логистические системы в глобальной экономике», 16–17 марта 2017 г., Красноярск. – Красноярск, 2017. С. 150–153.
7. Зенкова Ж.Н., Краковецкая И.В. Моделирование по неполным данным в логистике и маркетинге // Логистические системы в глобальной экономике: материалы Междунар. науч.-практ. конф. (14–15 марта 2013 г., Красноярск): в 2 ч. Ч. 1. Научно-исследовательский сектор / Сиб. гос. аэрокосмич. ун-т. – Красноярск, 2013. – С. 98–105.
8. Зенкова Ж.Н., Краковецкая И.В. Непараметрическая оценка Тёрнбулла для интервально-цензурированных данных в маркетинговом исследовании спроса на биоэнергетические напитки // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2013. – № 3 (24). – С. 64–69.
9. Зенкова Ж.Н. Оценка математического ожидания по дважды справа цензурированным данным и ее применение в задаче ценообразования // Математическое моделирование в экономике, страховании и управлении рисками: Сборник материалов Международной молодежной научно-практической конференции. Саратов, 5–8 ноября 2013 г. – Саратов: Изд-во Саратов. Ун-та, 2013. – С. 72–76.

10. *Зенкова Ж.Н., Муравлева М.А.* Расчет стоимости оборотных средств предприятия по цензурированным интервалом данным // Логистические системы в глобальной экономике [Электронный ресурс] : материалы Междунар. науч.-практ. конф. (3–4 марта 2014 г., Красноярск). Вып. 1. – Электрон. сб. – Сиб. гос. аэрокосмич. ун-т. – Красноярск, 2014. С. 103–107.

11. *Зенкова Ж.Н., Макеева О.Б.* Применение методов обработки цензурированных данных при анализе оборачиваемости // Вестник науки Казахского агротехнического университета им. С. Сейфуллина. Астана. – 2014. – № 3 (82). – С. 21–30.

12. *Зенкова Ж.Н., Муравлева М.А.* Асимптотическая несмещенность оценки функции распределения по однократно интервалом цензурированным данным с привлечением информации о симметрии // Новые информационные технологии в исследовании сложных структур: материалы X российской конференции с международным участием. – Томск, Издательский Дом Томского государственного университета, 2014. – С. 114–115.

13. *Дмитриев Ю.Г., Зенкова Ж.Н.* Об одной оценке симметричной функции распределения по цензурированной выборке // Материалы IV Всероссийской научно-практической конференции "Информационные технологии и математическое моделирование". Анжиро-Судженск. 2005. – Томск: Изд-во Том. ун-та, 2005. – Ч. 2. – С. 8–10.

ПРОГНОЗИРОВАНИЕ ТЕНДЕНЦИИ МАКСИМАЛЬНОЙ ЦЕНЫ АКЦИИ

Ю.Г. Дмитриев, Ю.Е. Пальянова

Томский государственный университет

dmit@mail.tsu.ru, pje0360@gmail.com

Введение

В работе рассматривается применение математической модели растущих систем [1–3] к моделированию и прогнозированию цен акций. Под растущей системой понимается любая совокупность объектов одной природы, которые вместе участвуют в процессах возникновения, создания и развития системы, подчиняющихся законам, одинаковым для всех объектов. Описание растущих систем возможно с помощью вероятностно-статистических моделей [1,2]. Системы подобного типа используются в геологии для изучения совокупности месторождений полезных ископаемых [4,5], для выявления закономерностей численности населения различных территорий [6], определения экстремального расхода воды в реках [7], расчета финансовых показателей и в других сферах практической деятельности.

В данной работе, исходя из предположения, что совокупность цен акций за определенный период времени является растущей системой, проводится подбор математической модели, которая отражает зависимость максимальных цен акций за разные периоды времени. Полученная модель приводит к специфической зависимости между ценами акций, позволяющей предсказывать тенденцию изменения их максимальных значений. Результаты применения модели иллюстрируются на модельных данных, а также данных стоимостей акций публичного акционерного общества «Газпром».

1. Описание модели для анализа цен акций

Пусть имеется N временных периодов с одинаковым количеством n дней (в рамках растущей системы это означает, что имеется N ячеек накопления). Наблюдается максимальное значение цены акции в каждый из n дней. Максимальные цены между днями будем считать независимыми и одинаково распределенными случайными величинами с функцией распределения $F(x)$. Обозначим максимальную цену акции за n дней в k -м периоде через $X_k^{(n)}$ и рассмотрим последовательность $X_1^{(n)}, \dots, X_N^{(n)}$, отвечающую N периодам. При этом функция распределения $G_{(n)}(x) = P\{X_k^{(n)} < x\} = F^n(x)$.

Выберем в качестве $F(x)$ двойной экспоненциальный закон $F(x; \mu, \sigma) = \exp\{-\exp\{-(x - \mu) / \sigma\}\}$, $-\infty < x < \infty$, $-\infty < \mu < \infty$, $\sigma > 0$, предложенный в [7] для описания максимальных значений наблюдаемой величины. Упорядочим на-

блюдения по возрастанию: $X_{(1)}^{(n)} < X_{(2)}^{(n)} < \dots < X_{(N)}^{(n)}$. Порядковые статистики $X_{(k)}^{(n)}$ можно рассматривать как оценки квантилей закона распределения $G_{(n)}(x)$ уровней $p_k = k / (N + 1)$, $k = \overline{1, N}$. На основании этого, следуя [8], можно записать, $F_0^n \left(\left(X_{(k)}^{(n)} - \mu \right) / \sigma \right) = p_k + \varepsilon_{(k)}$, где $\varepsilon_{(k)} = u_{(k)} - p_k$, $u_{(k)}$ – k -я порядковая статистика выборки из равномерного в $(0, 1)$ закона. Отсюда

$$\frac{X_{(k)}^{(n)} - \mu}{\sigma} = F_0^{-1} \left(\left(p_k + \varepsilon_{(k)} \right)^{1/n} \right), \quad k = \overline{1, N}, \quad \frac{X_{(k)}^{(n)} - \mu}{\sigma} = F_0^{-1} \left(p_k^{1/n} \right) + \varepsilon_{(k)}, \quad k = \overline{1, N}, \quad (1)$$

$$X_{(k)}^{(n)} \approx \mu + \sigma \cdot Y_{(k)}^{(n)} + \varepsilon_{(k)}, \quad k = \overline{1, N}, \quad (2)$$

где $Y_{(k)}^{(n)} = F_0^{-1} \left(p_k^{1/n} \right) = \left(-\ln \left(-n^{-1} \ln \left(p_k \right) \right) \right)$, $k = \overline{1, N}$, $F_0(x) = F(x; 0, 1)$ с $M\varepsilon_{(k)} = 0$ и матрицей ковариации $V = \text{cov}(\varepsilon_{(k)}, \varepsilon_{(k')}) = \frac{k(N - k' + 1)}{(N + 1)^2 (N + 2)}$, $k, k' = \overline{1, N}$ для порядковых статистик из равномерного закона в $(0, 1)$. Модель (2) отражает линейную регрессионную зависимость значений максимальных цен акций от квантилей закона распределения $F_0(x)$. Параметры μ и σ находятся по обобщенному методу наименьших квадратов (ОМНК) [8].

Рассмотрим уравнение (1), разложив функцию F_0^{-1} по формуле Тейлора:

$$\frac{X_{(k)}^{(n)} - \mu}{\sigma} = F_0^{-1} \left(p_k^{1/n} \right) + \left[F_0^{-1} \left(p_k^{1/n} \right) \right]' \varepsilon_{(k)} + \frac{1}{2} \left[F_0^{-1} \left(p_k^{1/n} \right) \right]'' \varepsilon_{(k)}^2 + O(\varepsilon_{(k)}^3),$$

$$X_{(k)}^{(n)} = \mu + \sigma \left(F_0^{-1} \left(p_k^{1/n} \right) + \left[F_0^{-1} \left(p_k^{1/n} \right) \right]' \varepsilon_{(k)} + \frac{1}{2} \left[F_0^{-1} \left(p_k^{1/n} \right) \right]'' \varepsilon_{(k)}^2 + O(\varepsilon_{(k)}^3) \right),$$

$$EX_{(k)}^{(n)} = \mu + \sigma \left(F_0^{-1} \left(p_k^{1/n} \right) + \frac{1}{2} \left[F_0^{-1} \left(p_k^{1/n} \right) \right]'' E\varepsilon_{(k)}^2 + O(E\varepsilon_{(k)}^3) \right) + \sigma \left[F_0^{-1} \left(p_k^{1/n} \right) \right]' E\varepsilon_{(k)}.$$

Так как $\varepsilon_{(k)}$ – это порядковые статистики из равномерного распределения, то согласно [8] при $k = k'$ получим $E\varepsilon_{(k)}^2 = \frac{k(N - k + 1)}{(N + 1)^2 (N + 2)}$.

$$EX_{(k)}^{(n)} = \mu + \sigma \left(F_0^{-1} \left(p_k^{1/n} \right) + \frac{1}{2} \left[F_0^{-1} \left(p_k^{1/n} \right) \right]'' \cdot \frac{k(N - k + 1)}{(N + 1)^2 (N + 2)} + O(E\varepsilon_{(k)}^3) \right) + \sigma \left[F_0^{-1} \left(p_k^{1/n} \right) \right]' E\varepsilon_{(k)}.$$

Если пренебречь величиной $O(E\varepsilon_{(k)}^3)$, то

$$X_{(k)}^{(n)} = \mu + \sigma \left(F_0^{-1} \left(p_k^{1/n} \right) + \frac{1}{2} \left[F_0^{-1} \left(p_k^{1/n} \right) \right]'' \cdot \frac{k(N - k + 1)}{(N + 1)^2 (N + 2)} \right) + \sigma \left[F_0^{-1} \left(p_k^{1/n} \right) \right]' E\varepsilon_{(k)},$$

$$EX_{(k)}^{(n)} = \mu + \sigma \left(F_0^{-1} \left(p_k^{1/n} \right) + \frac{1}{2} \left[F_0^{-1} \left(p_k^{1/n} \right) \right]'' \cdot \frac{k(N - k + 1)}{(N + 1)^2 (N + 2)} \right). \quad (3)$$

Обозначим $\alpha_{(k)}^{(n)} = F_0^{-1}(p_k^{1/n}) + \frac{1}{2} [F_0^{-1}(p_k^{1/n})]'' \cdot \frac{k(N-k+1)}{(N+1)^2(N+2)}$. Подставив $\alpha_{(k)}^{(n)}$ в (3),

получим $EX_{(k)}^{(n)} = \mu + \sigma\alpha_{(k)}^{(n)}$. Таким образом, имеем уточненную, по сравнению с (2), модель:

$$X_{(k)}^{(n)} = \mu + \sigma \cdot \alpha_{(k)}^{(n)} + \varepsilon_{(k)}, \quad k = \overline{1, N}. \quad (4)$$

2. Алгоритм прогноза

Задача заключается в прогнозировании тенденции (увеличении или уменьшении) максимальной цены акции в $N+1$ периоде по сравнению с предыдущим. Прогнозирование тенденции максимальной цены акции в $N+1$ периоде предлагается проводить следующим образом [1]. Пусть известны максимальные значения цен акций за N периодов, и вычислены квантили $Y_{(1)}^{(n)} < \dots < Y_{(N+1)}^{(n)}$ за $N+1$ периодов. Поставим в соответствие N упорядоченным значениям цен $X_{(k)}^{(n)}$ N квантилей $Y_{(k)}^{(n)}$ из $N+1$ возможных. Наилучшим будем считать такое соответствие, при котором погрешность

$$SKO = \sqrt{\sum_{k=1}^N (X_{(k)}^{(n)} - \hat{X}_{(k)}^{(n)})^2}, \quad (6)$$

где $\hat{X}_{(k)}^{(n)} \approx \hat{\mu} + \hat{\sigma}Y_{(k)}^{(n)} + \varepsilon_{(k)}$, $k = \overline{1, N}$, $(\hat{X}_{(k)}^{(n)} = \hat{\mu} + \hat{\sigma}\alpha_{(k)}^{(n)} + \varepsilon_{(k)}, k = \overline{1, N})$ будет наименьшей по всем возможным переборам соответствий с учетом порядка, а также $\hat{\mu}$ и $\hat{\sigma}$, вычисляемых по ОМНК для каждого соответствия. Когда такое соответствие найдено, то становится известен «свободный» квантиль $Y_{(k)}^{(n)}$, которому соответствия не нашлось. Подставляя этот квантиль в оцененное уравнение регрессии при наилучшем соответствии, находим прогнозное значение неизвестной цены акции на $N+1$ период и, соответственно, тенденцию цены.

3. Прогнозирование модельных данных

Возьмем в качестве $F(x)$ двойной экспоненциальный закон распределения $F(x; \mu, \sigma) = \exp\{-\exp\{-(x-\mu)/\sigma\}\}$. Сгенерируем 100 выборок для каждого набора значений параметров из этого закона с заданными μ , σ , N и n . Следовательно, можно увидеть растёт или убывает значение $N+1$ периода. При этом качество прогноза будем отслеживать по количеству верных прогнозов. Для сравнения будем использовать две модели, определенными формулами (2) и (4).

В табл. 1 приведены результаты по прогнозированию значения на $N+1$ период при фиксированном μ и разных N , n и σ . Также показано, сколько раз тенденция этого значения была верно определена. По результатам прогноза оказалось, что обе модели (2) и (4) улавливают тенденции одинаково.

Таблица 1

Результаты прогноза на $N+1$ период

Значения параметров при $n = 1$	Процент верных прогнозов тенденции, %	Значения параметров при $n = 5$	Процент верных прогнозов тенденции, %
$\mu = 2, \sigma = 5, N = 4$	85	$\mu = 2, \sigma = 5, N = 4$	65
$\mu = 2, \sigma = 5, N = 9$	80	$\mu = 2, \sigma = 5, N = 9$	60
$\mu = 2, \sigma = 0.1, N = 4$	90	$\mu = 2, \sigma = 0.1, N = 4$	70
$\mu = 2, \sigma = 0.1, N = 9$	90	$\mu = 2, \sigma = 0.1, N = 9$	70

Табл. 1 показывает, что наибольшее влияние на точность прогноза тенденции оказывает масштабный параметр σ : чем больше σ при фиксированном μ , тем хуже модель оценивает все параметры, и тем сложнее определить значение на $N + 1$ период. При этом количество элементов N в выборке не так сильно влияет на качество прогноза.

В табл. 2 приведены результаты по прогнозированию значения на $N + 1$ период при фиксированном σ и разных N , n и μ . Также показано, сколько раз тенденция этого значения была верно определена.

Таблица 2

Результаты прогноза на $N + 1$ период

Значения параметров при $n = 1$	Процент верных прогнозов тенденции, %	Значения параметров при $n = 5$	Процент верных прогнозов тенденции, %
$\mu = 2, \sigma = 2, N = 4$	80	$\mu = 2, \sigma = 2, N = 4$	65
$\mu = 2, \sigma = 2, N = 9$	76	$\mu = 2, \sigma = 2, N = 9$	61
$\mu = 1000, \sigma = 2, N = 4$	90	$\mu = 1000, \sigma = 2, N = 4$	75
$\mu = 1000, \sigma = 2, N = 9$	85	$\mu = 1000, \sigma = 2, N = 9$	70

По результатам из табл. 2 можно сделать вывод о том, что разные значения параметра положения μ несильно влияют на качество прогноза.

Таким образом, наиболее точный результат прогнозирования тенденции для значения в $N + 1$ период достигается при $n = 1$ и любом объёме выборки N . При этом оказалось, что обе модели (2) и (4) улавливают тенденции одинаково. Следовательно, данные результаты целесообразно использовать для дальнейшего прогноза реальных цен акций.

4. Прогноз тенденции по реальным данным

Рассмотрим реальные данные: цены акций ПАО «Газпром» на 15 временных интервалах за 2006–2018 гг. Поскольку торги на бирже открыты с понедельника по пятницу, то за неделю возьмем пять дней, т.е. $N = 4$, $N + 1 = 5$. $n = 1$. Точность подгонки будем определять по наименьшему среднеквадратическому отклонению из (6). Сравнивать полученные тенденции цен будем по моделям из (2) и (4). В табл. 3–4 знаком «+» показано, что данный способ отразил тенденцию цены на пятый день, т.е. если реальная цена росла, то и прогнозное значение этой цены также росло и наоборот, если реальная цена падала, то и прогноз падал. Соответственно, знак «-» отражает тот факт, что выбранный способ не показывает реальную ситуацию. Приведем результаты прогноза.

Таблица 3

Результаты прогнозирования

№ п/п	СКО для модели (2)	Прогнозируемая цена для модели (2), руб.	СКО для модели (4)	Прогнозируемая цена для модели (4), руб.
1	0.709	139.064 (+)	0.552	139.138 (+)
2	0.676	134.188 (+)	0.608	133.77 (+)
3	0.592	140.899 (+)	0.496	140.97 (+)
4	0.091	130.147 (+)	0.164	130.191 (+)
5	0.435	131.294 (+)	0.438	132.086 (+)
6	0.019	129.329 (+)	0.045	129.703 (+)
7	0.589	129.743 (+)	0.531	129.858 (+)
8	0.502	126.484 (-)	0.283	126.678 (-)
9	0.384	122.635 (+)	0.291	122.673 (+)
10	0.24	129.394 (-)	0.29	140.052 (+)
11	0.138	143.196 (+)	0.211	143.261 (+)
12	0.218	142.407 (+)	0.209	142.388 (+)
13	0.228	131.705 (+)	0.167	133.455 (-)
14	0.368	122.902 (-)	0.292	123.028 (-)
15	1.55	121.504 (+)	1.351	121.634 (+)

За представленный период реальная ситуация отражена в 12 случаях из 15, что соответствует 80%. При оценивании тенденции значения цены на пятый день разница между моделями (2) и (4) незначительна.

Рассмотрим следующую ситуацию. Предположим, что известны цены акций за четыре предыдущих периода (недели). Требуется спрогнозировать тенденцию цены акции на пятую неделю. Для данного случая рассмотрим 15 наблюдений в период с 2006 по 2018 год, т.е. $N = 4$, $N + 1 = 5$, $n = 5$. Приведем результаты прогноза.

Таблица 4

Результаты прогнозирования

№ п/п	СКО для модели (2)	Прогнозируемая цена для модели (2), руб.	СКО для модели (4)	Прогнозируемая цена для модели (4), руб.
1	1.824	146.331 (+)	1.669	146.003 (+)
2	0.856	134.892 (+)	1.114	133.921 (+)
3	3.379	130.143 (-)	4.684	128.644 (-)
4	1.019	125.703 (+)	0.888	125.759 (+)
5	0.248	122.157 (+)	0.183	122.098 (+)
6	0.846	118.526 (+)	0.611	118.797 (+)
7	0.679	122.686 (+)	0.484	122.447 (+)
8	2.331	128.79 (-)	2.796	129.615 (-)
9	1.697	144.605 (-)	1.491	144.63 (-)
10	3.474	131.205 (+)	3.409	131.495 (+)
11	0.49	146.015 (-)	0.624	145.863 (-)
12	2.244	143.408 (-)	3.179	142.419 (-)
13	1.209	157.72 (+)	0.961	157.345 (+)
14	1.503	145.883 (-)	2.153	145.255 (-)
15	0.614	138.679 (+)	0.549	138.629 (+)

За рассмотренный период реальная ситуация проявилась в 9 случаях из 15, что соответствует 60%. При оценивании значения тенденции цены на пятую неделю разница между моделями (2) и (4) незначительна.

Заключение

В работе рассмотрено применение математических моделей растущих систем для анализа максимальных цен акций с помощью линейной регрессии, основанной на квантилях равноотстоящих уровней двойного экспоненциального закона распределения. На примерах показаны прогностические свойства моделей (2) и (4), как на модельных, так и на реальных данных. Выявлено, что разница в прогностических свойствах между этими моделями незначительна. Полученные результаты целесообразно использовать для дальнейшего изучения прогнозных тенденций и реальных цен акций.

ЛИТЕРАТУРА

1. Дмитриев Ю.Г., Устинов Ю.К. Математические модели растущих систем // Журнал «Вычислительные технологии». – 2007. – Т. 12. – №1. – С. 68–75.
2. Dmitriev Yu.G., Tarasenko F.P. Towards a Modeling of Natural Systems // Cybernetics and Systems: An International Journal. – 2009. – № 40. – P. 236–248.
3. Дмитриев Ю.Г., Тарасенко П.Ф. Интерпретация геологических данных с использованием компьютерных технологий // Журнал «Известия ТПУ». – 2008. – Т. 313. – № 5. – С. 26–32.
4. Дмитриев Ю.Г., Тарасенко П.Ф. Автоматизированная система «Октава» для геологического прогнозирования СО РАН // Новосибирск: Вычислительные технологии, 2003. – Вып. 8.
5. Дмитриев Ю.Г., Тарасенко П.Ф. Об одной природной закономерности // Материалы Всероссийской конференции с международным участием (ЗОНТ-09), Новосибирск, 22–24 октября 2009. – Т.1. – С. 4–13.
6. Талейко А.В. Закономерности формирования численности населения по территориям // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2009. – № 3. – С. 80–86.
7. Смирнов Н.В., Дунин-Барковский И.В. Курс теории вероятностей и математической статистики для технических приложений. – М.: Наука, 1969. – 512 с.
8. Введение в теорию порядковых статистик: пер. с англ. / под редакцией А.Я. Боярского – М.: Статистика, 1970. – 413 с.

АНАЛИЗ ТОВАРНОГО АССОРТИМЕНТА ТОРГОВОГО ПРЕДПРИЯТИЯ Г. ПРОКОПЬЕВСКА

Н.А. Зенкова, Е.В. Чаусова

Томский государственный университет
zenkova.natasha@mail.ru, chauev@mail.ru

Введение

Проблема оптимизации товарного ассортимента является одной из наиболее важных в управлении коммерческим предприятием. Для её решения используется ABC-XYZ-анализ, играющий значимую роль в логистике и маркетинге [1,2]. ABC-XYZ анализ позволяет оценить товарный ассортимент и понять, как можно управлять разными группами товаров, какие товары следует вывести из ассортимента, а для каких товаров увеличить страховой запас. Данный инструмент активно используется практиками и рассматривается как стратегический при принятии решений о структуре товарного ассортимента. Это подтверждается большим количеством работ, связанных с применением ABC-XYZ-анализа на практике [3–6]. Однако необходимо учитывать особенности современной бизнес-ситуации и стратегические цели предприятия, поэтому очень важно уметь правильно применять ABC-XYZ-анализ при работе с конкретными данными.

В представленной работе на примере реальных данных торгового предприятия г. Прокопьевска, которое занимается продажей безалкогольных и слабоалкогольных напитков, проведен ABC-XYZ-анализ товарного ассортимента. Наряду с классическим ABC-анализом рассматриваются альтернативные подходы: метод касательных и метод петли. Полученные результаты приведены в виде таблиц и графиков.

1. Классический ABC-анализ

ABC-анализ – это метод, позволяющий изучить товарный ассортимент, определить наиболее значимые товары для компании с точки зрения определенного критерия, который выбирается исходя из цели проведения анализа [7]. В данной работе в качестве критерия используется выручка от продаж. ABC-анализ позволяет выявить нерентабельные или низко-рентабельные группы товаров, улучшить и оптимизировать ассортиментный портфель.

В основе классического ABC-анализа лежит принцип Парето, согласно которому 20% всех товаров дают 80% оборота. Основная цель метода ABC – классификация товаров на группы *A*, *B* и *C*. В результате анализа все товары делятся на три категории:

- *A* – наиболее важные товары, которые всегда должны присутствовать в ассортименте, это лидеры продаж. Это те 20% ассортимента, на которые приходится 80% всей выручки, их дефицит приводит к существенным убыткам;
- *B* – товары средней степени важности (30% ассортимента – 15% продаж);
- *C* – наименее важные товары, это претенденты на исключение из ассортимента и товары-новинки (50% ассортимента – 5% продаж).

Алгоритм проведения анализа

1. Упорядочиваем все товары по убыванию выручки от продаж.
2. Для каждого товара вычисляем его долю в общей сумме выручки:

$$d_i = \frac{R_i}{\sum_{i=1}^N R_i} \cdot 100\%, \quad (1)$$

где R_i – выручка от реализации i -го товара, $i = \overline{1, N}$, N – количество товаров.

3. Находим накопленную долю в процентах:

$$S_i = S_{i-1} + d_i, \quad (2)$$

где $S_0 = 0$, $i = \overline{1, N}$.

4. Распределяем товары на группы A , B и C по следующему правилу:

- если $S_i \leq 80\%$, то i -й товар относят к группе A ;
- если $80\% < S_i \leq 95\%$, то i -й товар относят к группе B ;
- если $S_i > 95\%$, то i -й товар относят к группе C .

Распределение товаров на классификационные группы принято вести с использованием таблицы. При этом в качестве объекта анализа могут выступать не только отдельные товары, но и товарные группы или подгруппы.

На основе реальных данных торгового предприятия г. Прокопьевска были проведены соответствующие расчеты. Были использованы данные о ежемесячной выручке за один год по девяти товарным группам, всего 50 видов товаров.

Динамика общей выручки предприятия от продажи напитков в течение одного календарного года представлена на рис. 1. По графику видно, что рост продаж приходится на третий месяц (март), летние месяцы и декабрь. Мартовский и декабрьский пики связаны с праздничными выходными днями.

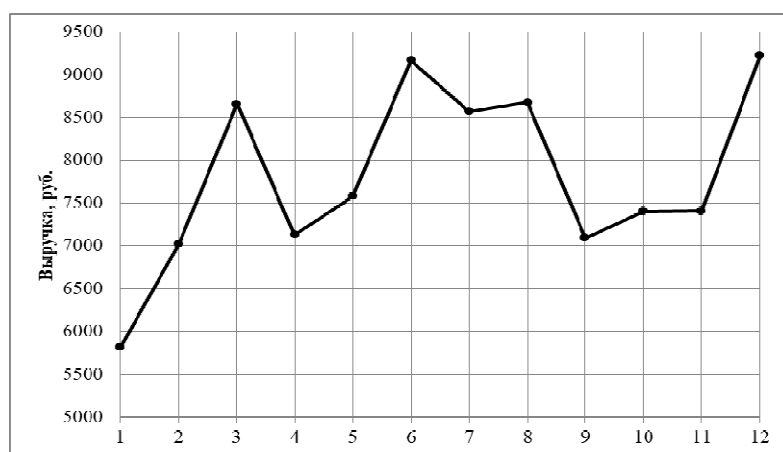


Рис. 1. Выручка торгового предприятия по месяцам за год

Результаты классического ABC-анализа, выполненного на товарных группах, приведены в табл. 1.

Таблица 1

Классический метод ABC-анализа на товарных группах

№	Товарная группа	R_i , руб.	d_i , %	S_i , %	Группа
1	ГОСТ	43186,82	46,05	46,05	A
2	Напиток слабоалкогольный 1	25342,44	27,02	73,07	A
3	Напиток безалкогольный 1	8294,04	8,84	81,92	B
4	Напиток слабоалкогольный 6	6127,09	6,53	88,45	B
5	Напиток слабоалкогольный 9	4612,46	4,92	93,37	B
6	Фирменный	2903,78	3,10	96,46	C
7	Минеральная вода	1809,21	1,93	98,39	C
8	Напиток слабоалкогольный 5	1456,26	1,55	99,95	C
9	Газированная вода	50,95	0,05	100,00	C
	Итого:	93783,04	100	–	–

Заметим, что первые две товарные группы попали в группу A , товарные группы 3, 4 и 5 – в группу B , а товарные группы 6–9 попали в группу C из-за низкой выручки, что наглядно представлено на рис. 2.

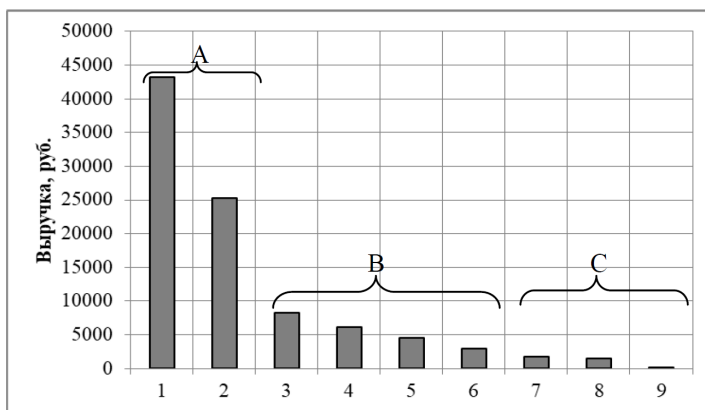


Рис. 2. Выручка торгового предприятия по товарным группам

В каждой товарной группе также был проведен классический ABC-анализ. Подробно рассмотрим товарную группу «ГОСТ», как группу с самой высокой выручкой. В табл. 2 приведены результаты распределения товаров по классификационным группам.

Таблица 2

Классический метод ABC-анализа внутри товарной группы «ГОСТ»

№	ГОСТ	R_i , руб.	d_i , %	S_i , %	Группа
1	Напиток слабоалкогольный 3 (ПЭТ 3.0)	24305,94	56,28	56,28	A
2	Напиток слабоалкогольный 3 (ПЭТ 1.5)	7787,46	18,03	74,31	A
3	Напиток слабоалкогольный 3 (бут.)	4326,60	10,02	84,33	B
4	Напиток слабоалкогольный 2 (ПЭТ 1.5)	1811,21	4,19	88,53	B
5	Напиток слабоалкогольный 2 (ПЭТ 3.0)	1376,22	3,19	91,71	B
6	Напиток слабоалкогольный 2 (бут.)	1193,63	2,76	94,48	B
7	Напиток слабоалкогольный 4 (ПЭТ 1.5)	871,17	2,02	96,49	C
8	Напиток слабоалкогольный 4 (ПЭТ 3.0)	833,27	1,93	98,42	C
9	Напиток слабоалкогольный 4 (бут.)	681,32	1,58	100	C
	Итого:	43186,82	100	–	–

Несмотря на очевидные преимущества классического метода ABC-анализа (простоту, прозрачность, универсальность), практика его применения показала, что он не всегда позволяет получить оптимальный результат. Классический метод ABC-анализа, разделяющий номенклатуру товаров на три группы, не учитывает некоторые особенности продаж из-за строгости границ деления на группы, классифицирует товары только по одному критерию, а современная бизнес-ситуация часто не может быть описана единственным критерием. Поэтому в практике применяются методы ABC-анализа, позволяющие классифицировать товары на группы по двум и более критериям и решать вопрос о количестве и границах групп.

2. ABC-анализ. Метод касательных (графический метод)

Метод касательных, предложенный В.С. Лукинским [8], относится к классу методов ABC-анализа с нефиксированными границами. Этот метод также называют методом построения кумулятивной кривой. Суть метода заключается в разделении товаров на группы при помощи касательных к ABC-кривой (её также называют кривой Парето, кривой Лоренца, кумулятивной кривой).

Алгоритм

1. Строим кривую Парето, описывающую зависимость накопленной доли от номенклатурных позиций, предварительно упорядоченных по убыванию выручки.

2. Соединяем её крайние точки (значения накопленной доли для первой и последней номенклатурной позиции) отрезком.

3. Параллельно этому отрезку проводим касательную к кривой Парето, точка касания определяет границу группы *A*. Все товары, лежащие левее этой точки, включаются в группу *A*.

4. Далее соединяем начальную и конечную точки оставшейся части кривой и проводим параллельно полученному отрезку касательную к кривой Парето, получаем вторую точку – границу группы *B*, которая разделяет оставшиеся товары на группы *B* и *C*. В зависимости от целей анализа можно выделить произвольное количество групп, но чаще всего – это три, реже 4–5 групп.

Рис. 3 демонстрирует, что товары под номерами 1, 2 и 3 попадают в классификационную группу *A*, товары 4, 5 и 6 попадают в группу *B*, товары 7, 8 и 9 относятся к группе *C*. В отличие от классического метода, группа *A* расширилась до 4-х товаров за счёт уменьшения группы *B*.

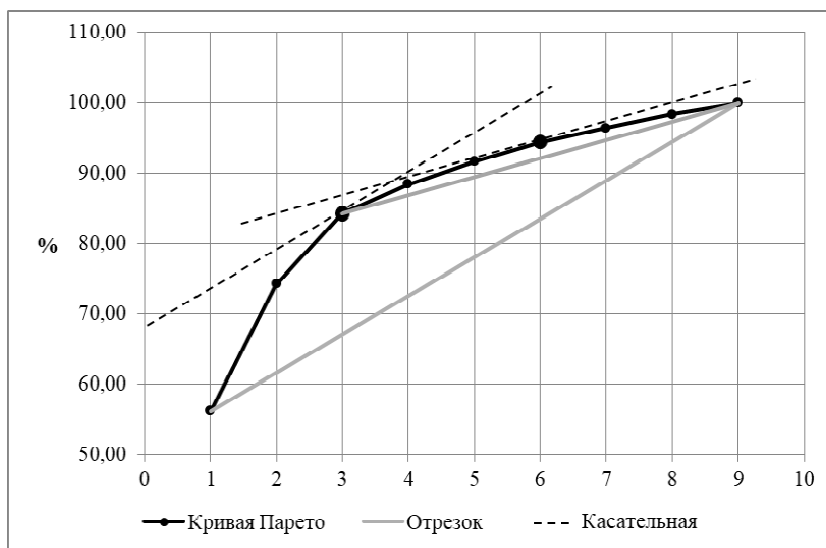


Рис. 3. Применение графического метода к товарной группе «ГОСТ»

3. ABC-анализ. Метод определения границ с помощью петли

Метод петли разработан А.М. Гаджинским [1]. В данном методе в качестве границ выступают участки резкого изменения кривизны ABC-кривой. Для построения петли проводим касательную к кривой ABC и восстанавливаем нормаль вправо от кривой. Длина нормали подбирается так, чтобы она не доставала до множества центров кривизны, которые соответствуют начальному и конечному участкам графика. Касательная начинает движение к противоположной стороне кривой, при этом нормаль тоже начинает движение в сторону движения касательной «вправо и вверх». Как только касательная достигает середины, направление движения нормали меняется «влево и вниз». Точка на кривой, которая соответствует моменту изменения касательной, указывает на границу между группами *A* и *B*. Движение продолжается до момента входа касательной на конечный участок графика, здесь снова меняется направление движения нормали и образуется граничная точка между группами *B* и *C*.

Требуется определить следующие вспомогательные параметры:

$$E_i = \frac{i \cdot 100\%}{N}, \quad i = \overline{1, N},$$

где E_i – количество строк таблицы ABC-анализа нарастающим итогом в % к общему количеству строк, i – номер товара, N – общее количество номенклатур,

$$G_i = \arctg\left(\frac{E_i}{d_i}\right),$$

где G_i – рабочий параметр, d_i определяются по формуле (1),

$$H_i = E_i + J_i \cdot \cos(G_i), \quad I_i = S_i - J_i \cdot \sin(G_i),$$

где H_i – абсцисса конца нормали, I_i – ордината конца нормали, S_i вычисляется по формуле (2), J_i – длина нормали задается в %, $J_i \in [20, 200]$. Длина нормали J_i определяется путём нескольких итераций, начиная с 20% и постепенно меняя на 5–10% до тех пор, пока границы товарных групп не перестанут меняться.

Определим границы классификационных групп с помощью метода петли на примере товарной группы «ГОСТ». Значения вспомогательных параметров приведены в табл. 3. Длина нормали приняла значение 70%.

Таблица 3

Расчет вспомогательных параметров для товарной группы «ГОСТ»

i	d_i	E_i	S_i	G_i	H_i	I_i	J_i
1	56,28	11,11	56,28	0,19	79,79	42,72	70
2	18,03	22,22	74,31	0,55	81,82	37,59	70
3	10,02	33,33	84,33	0,84	80,21	32,34	70
4	4,19	44,44	88,53	1,21	69,16	23,04	70
5	3,19	55,56	91,71	1,29	74,85	24,42	70
6	2,76	66,67	94,48	1,33	83,56	26,55	70
7	2,02	77,78	96,49	1,39	90,28	27,62	70
8	1,93	88,89	98,42	1,40	100,87	29,45	70
9	1,58	100,00	100,00	1,43	109,84	30,70	70

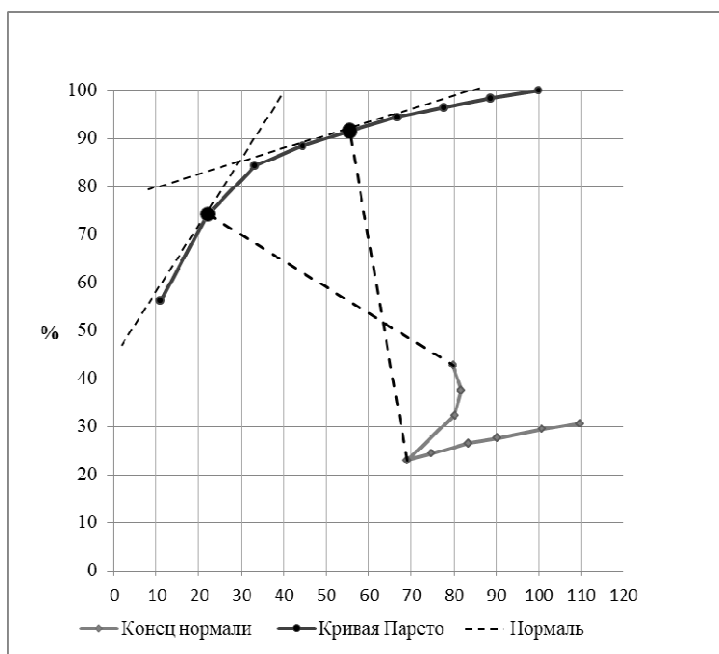


Рис. 4. Пример применения графического метода петли к товарной группе «ГОСТ»

Результаты построения петли представлены на рис. 4. Из графика видно, что товары 1 и 2 из табл. 3 относятся к группе A , товары 3, 4 и 5 – к группе B , а к группе C относятся товары 6, 7, 8 и 9.

Сведём результаты классификации товаров, полученные тремя методами, в табл. 4. Заметим, что товар 3 при классификации методом касательных относится к группе A , а при классификации классическим методом и методом петли – к группе B . Товар 6 при

классификации методом петли попал в группу *C*, а в двух других случаях – в группу *B*. При формировании ассортимента эти различия следует принять во внимание.

Таблица 4

Сводная таблица по трем методам ABC-анализа для товарной группы «ГОСТ»

№	ГОСТ	Классический	Касательных	Метод петли
1	Напиток слабоалкогольный 3 (ПЭТ 3.0)	<i>A</i>	<i>A</i>	<i>A</i>
2	Напиток слабоалкогольный 3 (ПЭТ 1.5)	<i>A</i>	<i>A</i>	<i>A</i>
3	Напиток слабоалкогольный 3 (бут.)	<i>B</i>	<i>A</i>	<i>B</i>
4	Напиток слабоалкогольный 2 (ПЭТ 1.5)	<i>B</i>	<i>B</i>	<i>B</i>
5	Напиток слабоалкогольный 2 (ПЭТ 3.0)	<i>B</i>	<i>B</i>	<i>B</i>
6	Напиток слабоалкогольный 2 (бут.)	<i>B</i>	<i>B</i>	<i>C</i>
7	Напиток слабоалкогольный 4 (ПЭТ 1.5)	<i>C</i>	<i>C</i>	<i>C</i>
8	Напиток слабоалкогольный 4 (ПЭТ 3.0)	<i>C</i>	<i>C</i>	<i>C</i>
9	Напиток слабоалкогольный 4 (бут.)	<i>C</i>	<i>C</i>	<i>C</i>

Существенным недостатком ABC-анализа является то, что он не позволяет оценить сезонные колебания спроса на товары. В большинстве случаев напитки являются сезонным товаром, что важно учитывать при формировании товарного ассортимента, при планировании запасов и расчёте страхового запаса.

4. XYZ-анализ

XYZ-анализ – это инструмент, позволяющий разделить продукцию по степени стабильности продаж и колебания уровня их потребления. Как и ABC-анализ, он применяется для разделения всех товаров на три группы. Основная цель XYZ-анализа – классификация товаров на группы *X*, *Y* и *Z* в зависимости от стабильности их продаж.

- *X* – товары со стабильной величиной потребления и высокой степенью прогнозирования;
- *Y* – товары с сезонными колебаниями продаж и средними возможностями их прогнозирования;
- *Z* – товары с нерегулярным потреблением и непредсказуемыми колебаниями, поэтому спрогнозировать их спрос достаточно сложно.

Для реализации XYZ-анализа необходимы данные о продажах товаров за несколько отчетных периодов. Мы располагали ежемесячными данными о выручке от продаж за один год. Основа анализа – коэффициент вариации:

$$CV = \frac{\sigma}{\bar{X}} \cdot 100\%, \quad (3)$$

где σ – среднеквадратическое отклонение (СКО), \bar{X} – это средняя выручка за анализируемый период времени, в нашем случае за один год. СКО и средняя выручка вычисляются по формулам:

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i, \quad \sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2},$$

где X_i – выручка от продаж товаров в i -м месяце, всего $n = 12$ месяцев.

Коэффициент вариации (3) рассчитывается для каждого товара, после чего товары классифицируются по группам следующим образом:

- если $CV \leq 10\%$, то соответствующий товар относится к группе *X*;
- если $10\% < CV \leq 30\%$, то соответствующий товар относится к группе *Y*;
- если $CV > 30\%$, то соответствующий товар относится к группе *Z*.

Результаты расчетов представлены в следующей таблице.

XYZ-анализ для товарной группы «ГОСТ»

№	ГОСТ	CV	Классический XYZ
1	Напиток слабоалкогольный 3 (ПЭТ 3.0)	13,24	Y
2	Напиток слабоалкогольный 3 (ПЭТ 1.5)	15,49	Y
3	Напиток слабоалкогольный 3 (бут.)	18,95	Y
4	Напиток слабоалкогольный 2 (ПЭТ 1.5)	17,37	Y
5	Напиток слабоалкогольный 2 (ПЭТ 3.0)	17,27	Y
6	Напиток слабоалкогольный 2 (бут.)	19,53	Y
7	Напиток слабоалкогольный 4 (ПЭТ 1.5)	19,72	Y
8	Напиток слабоалкогольный 4 (ПЭТ 3.0)	12,80	Y
9	Напиток слабоалкогольный 4 (бут.)	22,08	Y

Группы X не оказалось в представленных товарных номенклатурах. Это объясняется тем, что рассматриваемые напитки являются сезонным товаром.

Проведем совмещенный ABC-XYZ-анализ. Оба метода хорошо дополняют друг друга. Если ABC-анализ позволяет оценить вклад каждого продукта в структуру сбыта, то XYZ-анализ позволяет оценить скачки сбыта и его нестабильность. Составим матрицу ABC-XYZ. Заполнение этой матрицы происходит на основе двух классификаций ABC и XYZ. Таким образом, получаем 9 групп товаров, товары группы AX являются безусловными лидерами, а товары группы CZ – аутсайдеры. При этом возможна ситуация, когда не все группы заполнены.

Формирование групп с использованием матрицы ABC-XYZ-анализ позволяет в полной мере оценить весь ассортимент компании, пересмотреть отношение к ассортиментной политике, выработать рекомендации по прогнозированию продаж и эффективному управлению запасами товаров. По табл. 6 можно сделать следующие выводы:

1. Группы AY и BY (товары под № 1–6) при высоком товарообороте имеют недостаточную стабильность продаж, т.е. необходимо поддерживать страховой запас, чтобы обеспечить их наличие.

2. В группе CY (товары под № 7, 8, 9) страховой запас формируется, исходя из имеющихся у компании финансовых возможностей.

Таблица 6

Итоговая матрица ABC-XYZ-анализа для товаров группы «ГОСТ»

ABC/XYZ	X	Y	Z
A	–	Товары № 1, 2	–
B	–	Товары № 3–6	–
C	–	Товары № 7, 8, 9	–

Заключение

В работе рассмотрено применение ABC-XYZ методов для анализа ассортимента торгового предприятия г. Прокопьевска. На основе реальных данных о продажах слабоалкогольных и безалкогольных напитков был проведен ABC-XYZ-анализ ассортимента товаров с применением классического ABC-анализа, метода касательной и метода «петли». Полученные результаты позволяют выработать рекомендации по оптимизации товарного ассортимента на предприятии и повысить эффективность бизнеса.

ЛИТЕРАТУРА

1. Гаджинский А.М. Логистика: учеб. 17-е изд., перераб. и доп. – М.: Дашков и К., 2012. – 484 с.

2. *Зенкова Ж.Н.* Логистический подход в управлении предприятием : учебно-методический комплекс / Ж.Н. Зенкова ; Том. гос. ун-т, [Ин-т дистанционного образования]. – Томск: [ИДО ТГУ], 2012. URL: <http://vital.lib.tsu.ru/vital/access/manager/Repository/vtls:000460937>.
3. *Зенкова Ж.Н., Гаман М.И.* Модификация методов ABC-, XYZ-анализа на случай цензурированных данных / Материалы VII Международной конференции «Логистические системы в глобальной экономике», Красноярск, 2017. С. 150–153.
4. *Гаман М.И., Зенкова Ж.Н.* ABC-XYZ-анализ спроса на напитки с учетом дефицита // Сборник научных статей Международный молодежный симпозиум по управлению, экономике и финансам. – Казань: КГУ, 2017. – С. 405–408.
5. *Жарков Д.Н., Зенкова Ж.Н.* ABC-XYZ-анализ ассортимента табачной продукции торгового предприятия г. Томска // Сборник научных статей Международный молодежный симпозиум по управлению, экономике и финансам. – Казань, КГУ, 2017. – С. 408–411.
6. *Zenkova Z.N. , Kabanova T.V.* The ABC-XYZ Analysis Modified for Data with Outliers // Proceeding GOL'2018 The 4th IEEE International Conference on Logistics Operations Management. April 10–12, 2018, Le Havre, France. P. 63–68.
7. *Ксенофонтова О.Л., Новосельска Н.А.* ABC-XYZ-анализ как средство управления товарным ассортиментом торгового предприятия // Современные наукоемкие технологии. Региональное приложение. – 2013. – № 2. – С. 70–76.
8. *Лукинский В.С., Цвиринько И.А.* Прикладные методы и модели в логистике // Логистика сегодня. – 2004. – № 6. – С. 32–40.

СЕКЦИЯ III. ПРИКЛАДНОЙ ВЕРОЯТНОСТНЫЙ АНАЛИЗ

ОЦЕНИВАНИЕ ПАРАМЕТРОВ ПРОСТЕЙШЕГО ПОТОКА СОБЫТИЙ И МЁРТВОГО ВРЕМЕНИ, РАСПРЕДЕЛЕННОГО ПО ЗАКОНУ РЭЛЕЯ

М.Е. Завгородняя, Ю.Е. Табольжина

Томский государственный университет
mari.zavgor@mail.ru, tabolzhina@mail.ru

Введение

Потоки однородных событий являются распространенными математическими моделями многих реальных физических процессов. Часто на практике возникают ситуации, когда наступившее событие может повлечь за собой ненаблюдаемость последующих событий. Одним из искажающих факторов при оценке параметров потока событий выступает мёртвое время регистрирующих приборов, которое порождается зарегистрированным событием. Другие же события, наступившие в течение периода мёртвого времени, недоступны наблюдению. Все устройства регистрации делятся на две группы. Первую группу составляют устройства с непродлевающимся мёртвым временем, вторую – устройства с продлевающимся мёртвым временем. Отметим, что подобные задачи по оценке параметров случайных потоков событий, функционирующих в условиях непродлевающегося мёртвого времени, рассматриваются в работах [1–10].

В настоящей статье исследуется простейший поток событий, функционирующий в условиях непродлевающегося мёртвого времени, имеющего распределение Рэлея.

1. Постановка задачи

Рассматривается простейший поток событий интенсивности λ . После каждого события, зарегистрированного в момент t_k , наступает период мёртвого времени, в течении которого остальные события недоступны наблюдению. По окончании периода мёртвого времени первое наступившее событие снова создает период мёртвого времени случайной длительности и т.д. Длительность мёртвого времени распределена по закону распределения Рэлея с параметром σ , т.е. плотность распределения вероятностей имеет

вид $p(t) = \frac{t}{\sigma^2} e^{-\frac{t^2}{2\sigma^2}}$. Вариант возникновения возможной ситуации приведен на рис. 1,

где прямоугольниками обозначены периоды мёртвого времени длительности T_1, T_2, \dots ; t_1, t_2, \dots – моменты наступления событий в наблюдаемом потоке.

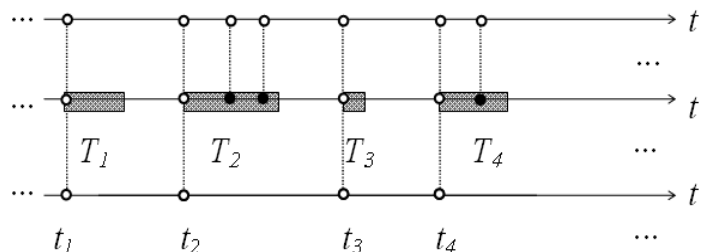


Рис. 1. Формирование наблюдаемого потока событий

Рассматривается установившийся (стационарный) режим функционирования наблюдаемого потока событий, поэтому переходными процессами на полуинтервале на-

блюдения $(t_0, t]$, где t_0 – начало наблюдений, t – окончание наблюдений, пренебрегаем. Тогда без потери общности можно положить $t_0 = 0$.

Необходимо в момент окончания наблюдений (в момент времени t) на основании выборки t_1, \dots, t_k наблюденных моментов наступления событий потока осуществить методом моментов построение оценки $\hat{\lambda}$ при известном значении параметра σ и оценки $\hat{\sigma}$ при известном значении параметра λ .

2. ММ – оценки параметров λ и σ

Обозначим через $\tau_k = t_{k+1} - t_k$, $k = 1, 2, \dots$ значение длительности k -го интервала между соседними событиями наблюдаемого потока ($\tau_k \geq 0$). Так как рассматривается стационарный режим, плотность вероятности значений длительности k -го интервала есть $\rho_{\tau_k}(x) = \rho_{\tau}(x)$, $x \geq 0$, для любого k в наблюдаемом потоке событий. В силу этого момент времени t_k без потери общности можно положить равным нулю, т.е. момент наступления события есть $\tau = 0$. В [1] получено выражение для плотности вероятности $\rho(x|t)$, когда длительность мёртвого времени является детерминированной величиной:

$$\rho(x|t) = 0, \quad x < t; \quad \rho(x|t) = \lambda e^{-\lambda(x-t)}, \quad x \geq t. \quad (1)$$

Тогда плотность вероятности $\rho_{\tau}(x)$ для наблюдаемого потока выпишется в виде:

$$p_{\tau}(x) = \int_0^x p(t) \rho(x|t) dt. \quad (2)$$

Подставляя в (2) выражение (1) и учитывая, что $p(t) = \frac{t}{\sigma^2} e^{-\frac{t^2}{2\sigma^2}}$, запишем (2) в следующем виде:

$$p_{\tau}(x) = \int_0^x \frac{t}{\sigma^2} e^{-\frac{t^2}{2\sigma^2}} \lambda e^{-\lambda(x-t)} dt. \quad (3)$$

Методом моментов получим оценку параметра λ при известном σ и оценку параметра σ при известном λ . Суть метода моментов заключается в приравнивании теоретических моментов распределения к соответственным эмпирическим моментам, найденных по выборке:

$$M\{\tau^k\} = \frac{1}{n} \sum_{i=1}^n \tau_i^k. \quad (4)$$

Теоретическое среднее значение длительности интервала между соседними моментами наступления событий наблюдаемого потока (начальный момент первого порядка) определяется по формуле:

$$M\{\tau\} = \int_0^{\infty} x p_{\tau}(x) dx. \quad (5)$$

Подставляя в (5) выражение (3), получим:

$$M\{\tau\} = \int_0^{\infty} \int_0^x x \frac{t}{\sigma^2} e^{-\frac{t^2}{2\sigma^2}} \lambda e^{-\lambda(x-t)} dt dx. \quad (6)$$

Поменяем порядок интегрирования в (6) и перепишем его в виде:

$$M\{\tau\} = \int_0^{\infty} \frac{t}{\sigma^2} e^{-\frac{t^2}{2\sigma^2}} e^{\lambda t} dt \int_t^{\infty} x \lambda e^{-\lambda x} dx. \quad (7)$$

Отдельно рассмотрим интеграл $\int_t^{\infty} x\lambda e^{-\lambda x} dx$:

$$\int_t^{\infty} x\lambda e^{-\lambda x} dx = te^{-\lambda t} + \frac{1}{\lambda} e^{-\lambda t}. \quad (8)$$

С учётом (8) перепишем (7) в виде:

$$M\{\tau\} = \int_0^{\infty} \frac{t}{\sigma^2} e^{-\frac{t^2}{2\sigma^2}} e^{\lambda t} e^{-\lambda t} \left(t + \frac{1}{\lambda} \right) dt. \quad (9)$$

В (9) сделаем замену $\frac{t}{\sigma} = z$ и запишем его в виде:

$$M\{\tau\} = \int_0^{\infty} ze^{-\frac{z^2}{2}} \left(z\sigma + \frac{1}{\lambda} \right) dz = I_1 + I_2 \quad (10)$$

Учитывая, что $I_1 = \sigma \int_0^{\infty} z^2 e^{-\frac{z^2}{2}} dz = \sigma \frac{\sqrt{2\pi}}{2}$, $I_2 = \frac{1}{\lambda} \int_0^{\infty} ze^{-\frac{z^2}{2}} dz = \frac{1}{\lambda}$, выражение (10) при-

мет вид: $M\{\tau\} = \sigma \frac{\sqrt{2\pi}}{2} + \frac{1}{\lambda}$.

Перепишем (4) для уравнения моментов первого порядка следующим образом:

$\sigma \frac{\sqrt{2\pi}}{2} + \frac{1}{\lambda} = \tau_{cp}$, $\tau_{cp} = \frac{1}{n} \sum_{i=1}^n \tau_i$. Рассмотрим два случая оценивания параметра λ простейшего потока событий и параметра σ мёртвого времени, имеющего распределение Рэлея.

1. Пусть известно значение параметра λ простейшего потока событий, а значение параметра σ необходимо оценить на основании выборки t_1, \dots, t_k . Составим уравнение

моментов и получим оценку $\hat{\sigma}$ в виде $\hat{\sigma} = \frac{2 \left(\tau_{cp} - \frac{1}{\lambda} \right)}{\sqrt{2\pi}}$.

2. Пусть известно значение параметра σ мёртвого времени, имеющего распределение Рэлея, а значение параметра λ необходимо оценить на основании выборки t_1, \dots, t_k .

Составим уравнение моментов и получим оценку $\hat{\lambda}$ в виде $\hat{\lambda} = \frac{1}{\tau_{cp} - \sigma \frac{\sqrt{2\pi}}{2}}$.

3. Результаты численных расчетов

Для получения численных результатов разработан алгоритм вычисления по формулам, представленным выше. Программа реализована в среде Microsoft Visual Studio 2015 на языке программирования C#. Первый этап расчёта заключается в нахождении численных значений ММ-оценок рассматриваемого потока. Второй этап расчёта предполагает имитационное моделирование исходного простейшего потока событий, схемы создания мёртвого времени, распределённого по закону Рэлея и наблюдаемого потока событий. Результатом работы имитационной модели является получение величин τ_i ($\tau_i \geq 0$) – значение длительности i -го интервала между соседними событиями наблюдаемого потока.

С целью установления свойств получаемых оценок проведены статистические эксперименты. Результаты приведены в табл. 1–4. В первой строке таблиц указано время моделирования T и соответствующая величина τ_{cp} , причем T изменяется от 50 до 200 с

шагом 50 ед. времени. Во второй строке табл. 1–4 для соответствующих T и τ_{cp} приведены численные значения ММ-оценок.

Таблица 1

$T = 50, \tau_{cp} = 4,6$	$T = 100, \tau_{cp} = 5,18$	$T = 150, \tau_{cp} = 5,58$	$T = 200, \tau_{cp} = 5,81$
$\hat{\sigma} = 2,87$	$\hat{\sigma} = 3,34$	$\hat{\sigma} = 3,65$	$\hat{\sigma} = 3,84$

В табл. 1 приведены значения оценки $\hat{\sigma}$ при заданных параметрах $\lambda = 1, \sigma = 4$.

Таблица 2

$T = 50, \tau_{cp} = 4,24$	$T = 100, \tau_{cp} = 4,75$	$T = 150, \tau_{cp} = 4,85$	$T = 200, \tau_{cp} = 4,87$
$\hat{\sigma} = 3,22$	$\hat{\sigma} = 3,63$	$\hat{\sigma} = 3,71$	$\hat{\sigma} = 3,73$

В табл. 2 приведены значения оценки $\hat{\sigma}$ при заданных параметрах $\lambda = 5, \sigma = 4$.

Таблица 3

$T = 50, \tau_{cp} = 1,43$	$T = 100, \tau_{cp} = 1,44$	$T = 150, \tau_{cp} = 1,49$	$T = 200, \tau_{cp} = 1,5$
$\hat{\lambda} = 5,66$	$\hat{\lambda} = 5,35$	$\hat{\lambda} = 4,23$	$\hat{\lambda} = 4,05$

В табл. 3 приведены значения оценки $\hat{\lambda}$ при заданных параметрах $\lambda = 4, \sigma = 1$.

Таблица 4

$T = 50, \tau_{cp} = 7,68$	$T = 100, \tau_{cp} = 6,34$	$T = 150, \tau_{cp} = 6,25$	$T = 200, \tau_{cp} = 6,22$
$\hat{\lambda} = 0,37$	$\hat{\lambda} = 0,75$	$\hat{\lambda} = 0,8$	$\hat{\lambda} = 0,83$

В табл. 4 приведены значения оценки $\hat{\lambda}$ при заданных параметрах $\lambda = 1, \sigma = 4$.

По результатам статистического эксперимента можно сделать вывод о том, что при увеличении времени моделирования T оценка $\hat{\lambda}$ при известном значении параметра σ и оценка $\hat{\sigma}$ при известном значении параметра λ улучшаются, приближаясь к истинным значениям.

Заключение

Полученные результаты показывают возможность оценивания параметра λ при известном значении параметра σ и параметра σ при известном значении параметра λ по результатам текущих наблюдений за потоком событий в условиях мёртвого времени, имеющего распределение Рэлея (оценки $\hat{\lambda}$ и $\hat{\sigma}$ при больших T достаточно близко лежат от истинных значений λ и σ).

ЛИТЕРАТУРА

1. Горцев А.М., Климов И.С. Оценка интенсивности пуассоновского потока событий в условиях частичной его ненаблюдаемости // Радиотехника. – 1991. – № 12. – С. 3–7.
2. Горцев А.М., Завгородняя М.Е. Оценка параметров альтернирующего потока событий при условии его частичной наблюдаемости // Оптика атмосферы и океана. – 1997. – Т. 10. – № 3. – С. 273–280.
3. Горцев А.М., Куснатдинов Р.Т. Оценивание состояний МС-потока событий при его частичной наблюдаемости // Известия высших учебных заведений. Физика. – 1998. – № 4. – С. 22–30.
4. Горцев А.М., Паришина М.Е. Оценивание параметров альтернирующего потока событий в условиях «мёртвого времени» // Известия высших учебных заведений. Физика. – 1999. – № 4. – С. 8–13.
5. Vasileva L.A., Gortsev A.M. Estimation of parameters of a double-stochastic flow of events under conditions of its incomplete observability // Automation and Remote Control. – 2002. – V. 63. – No. 3. – P. 511–515.
6. Vasileva L.A., Gortsev A.M. Estimation of the dead time of an asynchronous double stochastic flow of events under incomplete observability // Automation and Remote Control. – 2003. – V. 63. – No. 12. – P. 1890–1898.
7. Горцев А.М., Нежелская Л.А. Оценивание длительности мёртвого времени и параметров синхронного альтернирующего потока событий // Вестник Томского государственного университета. – 2003. – № 6. – С. 232–239.
8. Горцев А.М., Нежелская Л.А. Оценивание длительности «мёртвого времени» и интенсивностей синхронного дважды стохастического потока событий // Радиотехника. – 2004. – № 10. – С. 8–16.

9. Горцев А.М., Ниссенбаум О.В. Оценивание длительности мёртвого времени и параметров асинхронного альтернирующего потока событий при непродлеваемом мёртвом времени // Известия высших учебных заведений. Физика. – 2005. – Т. 10. – № 3. – С. 35–49.

10. Горцев А.М., Завгородняя М.Е. Оценивание параметра непродлеваемого мёртвого времени случайной длительности в пуассоновском потоке событий // Вестник Томского государственного университета. – 2017. – № 40. – С. 1–8.

СТАТИСТИЧЕСКИЕ ЭКСПЕРИМЕНТЫ НА ИМИТАЦИОННОЙ МОДЕЛИ ПУАССОНОВСКОГО ПОТОКА СОБЫТИЙ ПРИ ПРОДЛЕВАЮЩЕМСЯ СЛУЧАЙНОМ МЁРТВОМ ВРЕМЕНИ

А.А. Шитина

Томский государственный университет

ann.shitina@gmail.com

Введение

Распространёнными математическими моделями физических явлений и процессов являются потоки событий. В частности, такие модели применяются при исследовании информационных потоков сообщений, в телекоммуникационных системах, в спутниковых сетях связи т.п. [1,2]. В большинстве публикаций авторы рассматривают математические модели потоков событий, когда события потока доступны наблюдению. Однако на практике возникают ситуации, когда наступившее событие влечет за собой ненаблюдаемость последующих событий. Причиной ненаблюдаемости, как правило, выступает мёртвое время регистрирующих приборов [3], в течение которого зарегистрированное событие обрабатывается; другие же события, поступившие в этот период, теряются.

Регистрирующие приборы при этом делятся на два вида: с непродлевающимся мёртвым временем и продлевающимся [3]. При этом длительность мёртвого времени может быть как детерминированной величиной, так и случайной. Задачи по оценке параметров и состояний потока событий в условиях наличия мёртвого времени фиксированной длительности, в частности, рассматривались в работах [4–10]. При этом в [4–6] получены результаты для непродлеваемого мёртвого времени, в [7–10] – для продлеваемого.

Однако достаточно открытым остается вопрос изучения потоков событий, когда мёртвое время является случайной величиной с тем или иным законом распределения. Здесь отметим работу [11], в которой решается задача оценки параметра распределения непродлеваемого мёртвого времени случайной длительности в пуассоновском потоке событий, и работу [12], в которой находятся формулы для начальных моментов общего периода ненаблюдаемости в пуассоновском потоке событий при наличии продлеваемого случайного мёртвого времени. В настоящей статье рассматривается пуассоновский поток событий, функционирующий в условиях продлеваемого случайного мёртвого времени, приводятся результаты статистических экспериментов, реализованных на имитационной модели изучаемого потока.

1. Математическая модель наблюдаемого потока

Рассматривается пуассоновский поток событий интенсивности λ . После каждого зарегистрированного в момент времени t_k события наступает период мёртвого времени случайной длительности, который порождается этим событием, так что другие события пуассоновского потока, наступившие в течение этого периода мёртвого времени (периода ненаблюдаемости), недоступны наблюдению. В то же время, хотя события и не наблюдаются в течение периода мёртвого времени случайной длительности, они вызывают продление периода ненаблюдаемости на некоторую величину случайной длительности, так что наблюдаться будет лишь то событие, которое наступило после оконча-

ния последнего периода мёртвого времени. В результате формируется общий период ненаблюдаемости событий потока случайной длительности. Общий период ненаблюдаемости также является ненаблюдаемой случайной величиной. По окончании общего периода ненаблюдаемости первое наступившее событие снова создает период мёртвого времени случайной длительности и т.д. Принимается, что случайная длительность мёртвого времени, порождаемая каждым событием пуассоновского потока, распределена по экспоненциальному закону с параметром α ($\alpha > 0$). Вариант возникающей ситуации приведен на рис. 1, где на первой временной оси белыми кружками обозначены события исходного пуассоновского потока; на второй временной оси приведена схема создания общего периода ненаблюдаемости, где серым цветом обозначены значения периодов мёртвого времени, порождаемые каждым событием исходного пуассоновского потока, черными кружками обозначены ненаблюдаемые события; на третьей временной оси приведены моменты t_1, t_2, \dots наступления событий наблюдаемого потока, обозначенные белыми кружками, и обозначенные штриховкой значения ξ_1, ξ_2, \dots случайной величины ξ (ξ – общий период ненаблюдаемости событий исходного пуассоновского потока).

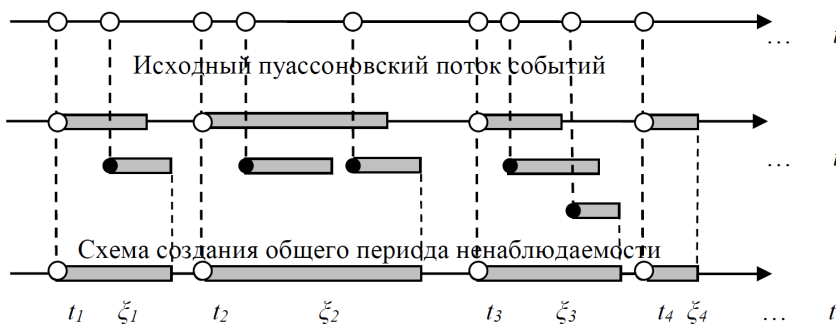


Рис. 1. Пуассоновский поток событий в условиях продлевающегося случайного мёртвого времени

Настоящая статья посвящена вопросу построения имитационной модели пуассоновского потока событий, функционирующего в условиях продлевающегося случайного мёртвого времени, с целью проведения статистических экспериментов на этой модели для изучения содержательной информации о статистических характеристиках общего периода ненаблюдаемости ξ .

2. Имитационная модель пуассоновского потока событий при продлевающемся случайном мёртвом времени

Метод имитационного моделирования широко применяется на этапе проектирования сложных систем. Основным средством реализации имитационного моделирования служит компьютер, позволяющий осуществлять цифровое моделирование систем и сигналов. При имитационном моделировании используемая математическая модель воспроизводит алгоритм функционирования исследуемой системы во времени при различных сочетаниях значений параметров системы. При исследовании процесса методом имитационного моделирования должно быть реализовано наблюдение за ним с течением времени.

Основные преимущества имитационного моделирования по сравнению с аналитическими методами заключаются в следующем [13]: 1) большая адекватность между сущностью физического явления (объекта) и математической моделью физического явления (объекта); 2) более широкий охват физических процессов и систем по сравнению с аналитическими методами; 3) имитационное моделирование функционирования процессов и систем при разнообразных законах распределения случайных величин как в установившихся, так и в переходных режимах; 4) получение в результате имитацион-

ного моделирования самых разнообразных статистических характеристик исследуемых процессов и систем.

На рис. 2 представлена блок-схема алгоритма имитационного моделирования пуассоновского потока событий в условиях продлевающегося случайного мёртвого времени.

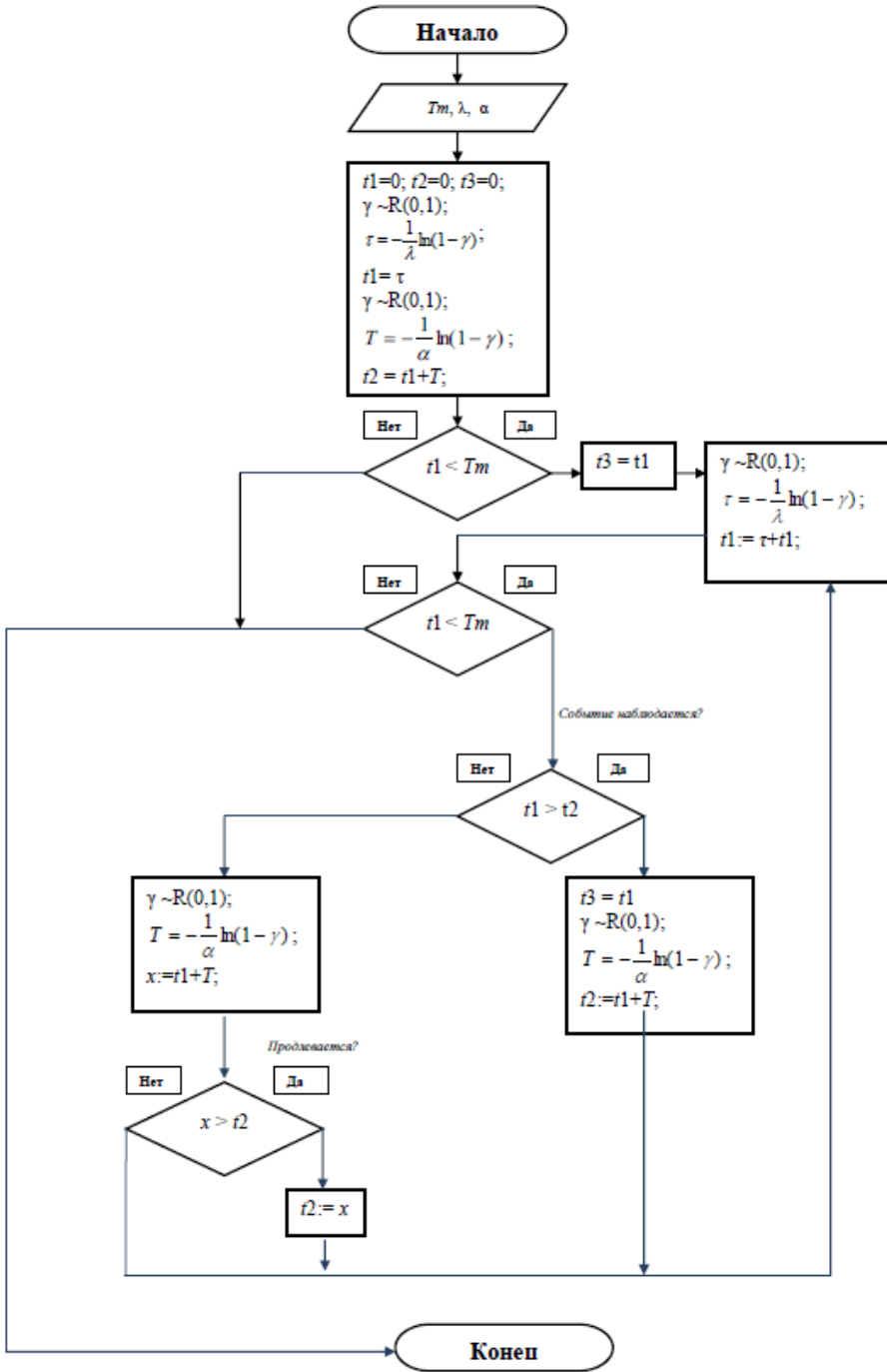


Рис. 2. Блок-схема алгоритма имитационного моделирования пуассоновского потока событий в условиях продлевающегося случайного мёртвого времени

На рис. 2 введены следующие обозначения: T_m – время моделирования; t_1 – момент наступления события в пуассоновском потоке; t_2 – момент окончания длительности мёртвого времени; t_3 – момент наступления события в наблюдаемом потоке; T – длительность мёртвого времени; τ – интервал времени между соседними событиями; x – вспомогательная переменная.

Для получения численных результатов разработана программа. Она реализована на языке программирования C# в среде Microsoft Visual Studio C#. Первый этап расчёта предполагает имитационное моделирование исходного потока событий, схемы создания продлевающегося случайного мёртвого времени и наблюдаемого потока событий.

Результатом работы имитационной модели является последовательность значений длительности общих периодов ненаблюдаемости событий исходного пуассоновского потока $\{\xi_1, \dots, \xi_n\}$, где n – последнее наблюдаемое событие на интервале моделирования T_m .

3. Статистические эксперименты и численные результаты

С целью получения численных значений общего периода ненаблюдаемости представлены статистические эксперименты.

Первый статистический эксперимент: 1) при заданных значениях параметров λ , α и заданном времени моделирования T_m осуществляется имитационное моделирование наблюдаемого потока; выходом имитационной модели являются последовательности значений $\xi_i^{(1)}$, $i = \overline{1, N_1}$ – общий период ненаблюдаемости, порождённый одним (наблюдаемым) событием исходного потока; $\xi_i^{(2)}$, $i = \overline{1, N_2}$ – общий период ненаблюдаемости, порождённый двумя событиями исходного потока (одним наблюдаемым и одним ненаблюдаемым); ... ; $\xi_i^{(k)}$, $i = \overline{1, N_k}$ – общий период ненаблюдаемости, порождённый k событиями исходного потока (одним наблюдаемым и $k-1$ ненаблюдаемыми), $k = 1, 2, \dots$; значения N_1, \dots, N_k зависят от заданных значений λ , α , T_m и конкретной реализации наблюдаемого потока; 2) в рамках конкретной реализации наблюдаемого потока вычисляются следующие статистические характеристики [14]: $\hat{M}_k = \frac{1}{N_k} \sum_{i=1}^{N_k} \xi_i^{(k)}$ – оценка условного математического ожидания общего периода ненаблюдаемости $M(\xi | k)$, $k = 1, 2, \dots$; $\hat{D}_k = \frac{1}{N_k} \sum_{i=1}^{N_k} [\xi_i^{(k)} - \hat{M}_k]^2$ – оценка условной дисперсии общего периода ненаблюдаемости $D(\xi | k)$, $k = 1, 2, \dots$; $D^{(+,-)} = \hat{M}_k \pm 1,96 \sqrt{\hat{D}_k} / \sqrt{N_k - 1}$ – верхняя (+) и нижняя (–) границы доверительного интервала общего периода ненаблюдаемости, порождённого k событиями исходного потока (доверительная вероятность $\gamma_k = 0,95$), $k = 1, 2, \dots$; 3) после чего время моделирования T_m увеличивается и шаги 1, 2 повторяются.

В табл. 1–12 приведены результаты первого статистического эксперимента.

Таблица 1

Оценка условного математического ожидания общего периода ненаблюдаемости при $\lambda = 1$, $\alpha = 90$

T_m	1 000	2 000	3 000	4 000	5 000
\hat{M}_1	0,01130	0,01113	0,01066	0,01071	0,01102
\hat{M}_2	0,03333	0,02555	0,02750	0,02801	0,02621

Таблица 2

Оценка условной дисперсии общего периода ненаблюдаемости при $\lambda = 1, \alpha = 90$

T_m	1 000	2 000	3 000	4 000	5 000
\hat{D}_1	0,00014	0,00013	0,00012	0,00011	0,00012
\hat{D}_2	0,00049	0,00029	0,00017	0,00017	0,00025

Таблица 3

Нижняя граница доверительного интервала общего периода ненаблюдаемости при $\lambda = 1, \alpha = 90$

T_m	1 000	2 000	3 000	4 000	5 000
$D_1^{(-)}$	0,01056	0,01063	0,01027	0,01038	0,01072
$D_2^{(-)}$	0,01797	0,01888	0,02264	0,02428	0,02175

Таблица 4

Верхняя граница доверительного интервала общего периода ненаблюдаемости при $\lambda = 1, \alpha = 90$

T_m	1 000	2 000	3 000	4 000	5 000
$D_1^{(+)}$	0,01204	0,01163	0,01105	0,01104	0,01132
$D_2^{(+)}$	0,04870	0,03222	0,03236	0,03173	0,03068

Таблица 5

Оценка условного математического ожидания общего периода ненаблюдаемости при $\lambda = 1, \alpha = 90$

T_m	10 000	20 000	30 000	40 000	50 000
\hat{M}_1	0,01101	0,01094	0,01101	0,01105	0,01099
\hat{M}_2	0,02556	0,02711	0,02555	0,02844	0,02783

Таблица 6

Оценка условной дисперсии общего периода ненаблюдаемости при $\lambda = 1, \alpha = 90$

T_m	10 000	20 000	30 000	40 000	50 000
\hat{D}_1	0,00012	0,00012	0,00012	0,00012	0,00012
\hat{D}_2	0,00024	0,00027	0,00022	0,00032	0,00029

Таблица 7

Нижняя граница доверительного интервала общего периода ненаблюдаемости при $\lambda = 1, \alpha = 90$

T_m	10 000	20 000	30 000	40 000	50 000
$D_1^{(-)}$	0,01080	0,01079	0,01088	0,01094	0,01090
$D_2^{(-)}$	0,02271	0,02493	0,02392	0,02676	0,02641

Таблица 8

Верхняя граница доверительного интервала общего периода ненаблюдаемости при $\lambda = 1, \alpha = 90$

T_m	10 000	20 000	30 000	40 000	50 000
$D_1^{(+)}$	0,01123	0,01110	0,01114	0,01116	0,01109
$D_2^{(+)}$	0,02841	0,02929	0,02717	0,03012	0,02925

Таблица 9

Оценка условного математического ожидания общего периода ненаблюдаемости при $\lambda = 1, \alpha = 90$

T_m	60 000	70 000	80 000	90 000	100 000
\hat{M}_1	0,01101	0,01104	0,01101	0,01099	0,01099
\hat{M}_2	0,02796	0,02748	0,02762	0,02729	0,02758
\hat{M}_3	0,04255	0,03981	0,03789	0,04090	0,03522

Оценка условной дисперсии общего периода ненаблюдаемости при $\lambda = 1, \alpha = 90$

T_m	60 000	70 000	80 000	90 000	100 000
\hat{D}_1	0,00012	0,00012	0,00012	0,00012	0,00012
\hat{D}_2	0,00025	0,00028	0,00026	0,00026	0,00030
\hat{D}_3	0,00040	0,00058	0,00097	0,00041	0,00019

Таблица 11

Нижняя граница доверительного интервала общего периода ненаблюдаемости при $\lambda = 1, \alpha = 90$

T_m	60 000	70 000	80 000	90 000	100 000
$D_1^{(-)}$	0,01092	0,01096	0,01094	0,01092	0,01092
$D_2^{(-)}$	0,02669	0,02629	0,02651	0,02626	0,02655
$D_3^{(-)}$	0,03129	0,02404	0,02158	0,03027	0,02839

Таблица 12

Верхняя граница доверительного интервала общего периода ненаблюдаемости при $\lambda = 1, \alpha = 90$

T_m	60 000	70 000	80 000	90 000	100 000
$D_1^{(+)}$	0,01110	0,01112	0,01109	0,01107	0,01105
$D_2^{(+)}$	0,02922	0,02868	0,02872	0,02832	0,02860
$D_3^{(+)}$	0,05381	0,05558	0,05420	0,05154	0,04204

Из анализа табл. 1–12 следует, что для условного математического ожидания \hat{M}_1 стационарный режим устанавливается (\hat{M}_1 стремится к константе с увеличением T_m) для $T_m > 1000$, для \hat{M}_2 стационарный режим устанавливается (\hat{M}_2 стремится к константе с увеличением T_m) для $T_m > 10\,000$. Общий период ненаблюдаемости, порождённый тремя событиями, появляется только при $T_m > 60\,000$.

Таким образом, при заданных параметрах $\lambda = 1, \alpha = 90$ и $T_m > 60\,000$ стационарный режим устанавливается только для общего периода ненаблюдаемости, порождённого одним либо двумя событиями исходного потока.

Второй статистический эксперимент: 1) при заданных значениях параметров λ, α и заданном времени моделирования T_m осуществляется имитационное моделирование наблюдаемого потока: выходом имитационной модели является последовательность значений $\xi_i, i = \overline{1, N}$ – общий период ненаблюдаемости, порождённый одним наблюдаемым событием исходного потока, либо двумя событиями исходного потока (одним наблюдаемым и одним ненаблюдаемым), либо тремя событиями исходного потока (одним наблюдаемыми и двумя ненаблюдаемыми), ... ; значения N зависят от заданных значений λ, α, T_m и конкретной реализации наблюдаемого потока; 2) в рамках конкретной реализации наблюдаемого потока вычисляются следующие статистические характеристики [14]: $\hat{M} = \frac{1}{N} \sum_{i=1}^N \xi_i$ – оценка безусловного математического ожидания общего

периода ненаблюдаемости $M(\xi)$; $\hat{D} = \frac{1}{N} \sum_{i=1}^N [\xi_i - \hat{M}]^2$ – оценка безусловной дисперсии

общего периода ненаблюдаемости $D(\xi)$; $D^{(+,-)} = \hat{M} \pm 1,96\sqrt{\hat{D}}/\sqrt{N-1}$ – верхняя (+) и нижняя (–) границы доверительного интервала общего периода ненаблюдаемости (до-

верительная вероятность $\gamma = 0,95$); 3) после чего время моделирования T_m увеличивается и шаги 1, 2 повторяются.

В табл. 13–15 приведены результаты второго статистического эксперимента.

Таблица 13

Второй статистический эксперимент при $\lambda = 1, \alpha = 90$

T_m	1 000	2 000	3 000	4 000	5 000
\hat{M}	0,01116	0,01095	0,01120	0,01107	0,01108
\hat{D}	0,00016	0,00013	0,00023	0,00019	0,00019
$D^{(-)}$	0,01049	0,01045	0,01080	0,01072	0,01078
$D^{(+)}$	0,01183	0,01145	0,01161	0,01142	0,01139

Таблица 14

Второй статистический эксперимент при $\lambda = 1, \alpha = 90$

T_m	10 000	20 000	30 000	40 000	50 000
\hat{M}	0,01126	0,01106	0,01115	0,01115	0,01124
\hat{D}	0,00012	0,00026	0,00026	0,00018	0,00024
$D^{(-)}$	0,01104	0,01091	0,01103	0,01104	0,01114
$D^{(+)}$	0,01148	0,01122	0,01128	0,01126	0,01134

Таблица 15

Второй статистический эксперимент при $\lambda = 1, \alpha = 90$

T_m	60 000	70 000	80 000	90 000	100 000
\hat{M}	0,01117	0,01120	0,01117	0,01119	0,01116
\hat{D}	0,00016	0,00016	0,00021	0,00019	0,00032
$D^{(-)}$	0,01108	0,01112	0,01109	0,01111	0,01109
$D^{(+)}$	0,01126	0,01128	0,01125	0,01126	0,01123

Анализируя численные результаты, представленные в табл. 13–15, можно сделать следующий вывод: стационарный режим для \hat{M} устанавливается (\hat{M} стремится к константе с увеличением T_m) для $T_m > 1000$.

Заключение

В настоящей статье разработана имитационная модель пуассоновского потока событий, функционирующего в условиях продлевающегося мёртвого времени, которая является инструментом для задач оценивания параметра исходного пуассоновского потока λ и параметра экспоненциального распределения длительности мёртвого времени α . Проведённые статистические эксперименты, а также другие многочисленные испытания имитационной модели подтверждают её работоспособность.

ЛИТЕРАТУРА

1. *Basharin G.P., Gaidamaka Y.V., Samouylov K.E.* Mathematical theory of teletraffic and its application to the analysis of multiservice communication of next generation networks // Automatic Control and Computer Sciences. – 2013. – Vol. 47. – No. 2. – P. 62–69.
2. *Дудин А.Н., Клименок В.Н.* Системы массового обслуживания с коррелированными потоками. – Минск: Изд-во БГУ, 2000. – 175 с.
3. *Апанасович В.В., Коляда А.А., Чернявский А.Ф.* Статистический анализ случайных потоков в физическом эксперименте. – Минск: Университетское, 1988. – 256 с.
4. *Горцев А.М., Ниссенбаум О.В.* Оценивание длительности мёртвого времени и параметров асинхронного альтернирующего потока событий с инициированием лишнего события // Вестник Томского государственного университета. – 2004. – № 284. – С. 137–145.
5. *Горцев А.М., Ниссенбаум О.В.* Оценивание длительности мёртвого времени и параметров асинхронного альтернирующего потока событий при непродлеваемом мёртвом времени // Известия высших учебных заведений. Физика. – 2005. – № 10. – С. 35–49.

6. Горцев А.М., Калягин А.А., Нежелская Л.А. Оценка максимального правдоподобия длительности мёртвого времени в обобщённом полусинхронном потоке // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2015. – № 1. – С. 27–37.
7. Горцев А.М., Климов И.С. Оценивание периода ненаблюдаемости и интенсивности пуассоновского потока событий // Радиотехника. – 1996. – № 2. – С. 8–11.
8. Горцев А.М., Паршина М.Е. Оценивание параметров альтернирующего потока событий в условиях «мёртвого времени» // Известия высших учебных заведений. Физика. – 1999. – № 4. – С. 8–13.
9. Горцев А.М., Нежелская Л.А. Полусинхронный дважды стохастический поток событий при продлеваемом мёртвом времени // Вычислительные технологии. – 2008. – Т. 13. – № 1. – С. 31–41.
10. Бушланов И.В., Горцев А.М., Нежелская Л.А. Оценка параметров синхронного дважды стохастического потока событий // Автоматика и телемеханика. – 2008. – № 9. – С. 76–93.
11. Горцев А.М., Завгородняя М.Е. Оценивание параметра продлевающегося мёртвого времени случайной длительности в пуассоновском потоке событий // Вестник Томского государственного университета. Управление, вычислительная техника и информатика – 2017. – № 40. – С. 32–40.
12. Глухова Е.В., Терпугов А.Ф. Оценка интенсивности пуассоновского потока событий при наличии продлевающегося мёртвого времени // Известия высших учебных заведений. Физика. – 1995. – Т. 38. – № 3. – С. 22–31.
13. Лифшиц А.Л., Мальц Э.А. Статистическое моделирование систем массового обслуживания. – М.: Сов. Радио, 1978. – 248 с.
14. Шулепин В.П. Математическая статистика. Ч.1. Параметрическая статистика. – Томск: Изд-во НТЛ, 2012. – 540 с.

СТАТИСТИЧЕСКИЕ ЭКСПЕРИМЕНТЫ НА ИМИТАЦИОННОЙ МОДЕЛИ ОБОБЩЁННОГО СИНХРОННОГО ПОТОКА СОБЫТИЙ ВТОРОГО ПОРЯДКА С НЕПРОДЛЕВАЮЩИМСЯ МЁРТВЫМ ВРЕМЕНЕМ

Л.А. Нежелская, Е.Ф. Сидорова

*Томский государственный университет
ludne@mail.ru, katusha_sidorova@mail.ru*

Введение

Вследствие интенсивного развития компьютерной техники и применения инноваций в сфере информационных технологий возникли новые задачи, подобные тем, что были выдвинуты и исследованы А.К. Эрлангом в начале XX века. В то время как работы датского ученого были направлены на решение задач оптимизации обслуживания заявок, поступающих на телефонную станцию, задачи, которые остро ставятся на современном этапе, имеют место во многих сферах научных исследований: экономических и технических системах, системах связи и управления, процессах функционирования автоматизированных систем управления (АСУ) и т.д. При решении задач данного типа пользуются математическим аппаратом теории массового обслуживания (ТМО).

Наиболее сложные исследования случайных входящих потоков событий – основных элементов систем массового обслуживания (СМО) – характерны для АСУ и сетей связи [1,2]. Усложнение структуры телекоммуникационных систем, глобальных компьютерных сетей, спутниковых сетей связи и т.п., цифровых систем интегрального обслуживания (ЦСИО), разнообразие программного и аппаратного обеспечения, интеграция различных систем связи выявили необходимость построения новых математических моделей входящих потоков событий, достаточно адекватно описывающих информационные потоки, функционирующие в современных ЦСИО. Одними из первых работ в этом направлении были статьи [3–5]. В реальных ситуациях параметры, определяющие входящий поток событий, изменяются с течением времени, более того, изменения обычно носят случайный характер. Последнее приводит к рассмотрению дважды стохастических потоков событий [6–9]. Случайные входящие потоки с зависящей от времени интенсивностью, являющейся случайным процессом, можно разделить на два класса. Первый класс составляют потоки, интенсивность которых является непрерывным случайным процессом, второй – потоки с интенсивностью, представляющей собой кусочно-постоянный случайный процесс с конечным числом состояний. Потоки второго класса также носят название МС-потоков (Markov chain) либо MAP-потоков

(Markovian Arrival Process) [3,5] и являются наиболее характерными для реальных телекоммуникационных сетей. Обобщение МАР-потока событий приведено в [10], связь МС-потоков и МАР-потоков установлена в [11].

Основная часть литературы по СМО посвящена нахождению вероятностных характеристик систем, функционирующих в стационарном режиме, в условиях доступности наблюдению всех событий входящего потока. Однако на практике наступившее событие может создать период мёртвого времени, в течение которого другие события потока становятся недоступными для наблюдения или, другими словами, теряются. В этом случае мёртвое время, порождаемое зарегистрированным событием, можно рассматривать как искажающий фактор [12], т.к. его эффект влечёт за собой потери событий потока, что отрицательно сказывается на оценивании как состояний, так и параметров потока. Чтобы оценить потери событий потока необходимо оценить значение длительности мёртвого времени, при этом период ненаблюдаемости (недоступности наблюдению) потока событий может продолжаться некоторое фиксированное время, но также может быть и случайным. Отметим, что работы [13–15] являются одними из первых по оценке параметров случайных потоков событий, функционирующих в условиях неполной наблюдаемости.

В настоящей статье проводится исследование обобщённого синхронного потока событий второго порядка, которое является непосредственным развитием работ [16,17]. Для рассматриваемого потока построена имитационная модель, учитывающая наличие неполной его наблюдаемости. Кроме того, проведен ряд экспериментов с целью получения корректных и непротиворечивых численных результатов, подтверждающих работоспособность разработанной модели.

1. Постановка задачи

Рассматривается обобщённый синхронный поток событий второго порядка (поток), сопровождающий случайный процесс которого $\lambda(t)$ является кусочно-постоянным с двумя состояниями S_1 и S_2 . Далее по тексту под i -м состоянием процесса $\lambda(t)$ понимается состояние S_i , $i = 1, 2$.

Длительность интервала между событиями потока в i -м состоянии определяется случайной величиной $\eta_i = \min(\xi_i^{(1)}, \xi_i^{(2)})$, где случайная величина $\xi_i^{(1)}$ распределена по закону $F_i^{(1)}(t) = 1 - e^{-\lambda_i t}$, случайная величина $\xi_i^{(2)}$ распределена по закону $F_i^{(2)}(t) = 1 - e^{-\alpha_i t}$; $\xi_i^{(1)}$ и $\xi_i^{(2)}$ – независимые случайные величины, $i = 1, 2$. В момент наступления события потока процесс $\lambda(t)$ переходит из i -го состояния в j -е либо с вероятностью $P_1^{(1)}(\lambda_j | \lambda_i)$, либо с вероятностью $P_1^{(2)}(\lambda_j | \lambda_i)$ в зависимости от того, какое значение приняла случайная величина η_i , $i, j = 1, 2$, $i \neq j$. В момент наступления события потока процесс $\lambda(t)$ остается в i -м состоянии либо с вероятностью $P_1^{(1)}(\lambda_i | \lambda_i)$, либо с вероятностью $P_1^{(2)}(\lambda_i | \lambda_i)$ в зависимости от того, какое значение приняла случайная величина η_i , $i = 1, 2$. При этом $P_1^{(1)}(\lambda_j | \lambda_i) + P_1^{(1)}(\lambda_i | \lambda_i) = 1$, $P_1^{(2)}(\lambda_j | \lambda_i) + P_1^{(2)}(\lambda_i | \lambda_i) = 1$, $i, j = 1, 2$, $i \neq j$. Таким образом, длительность интервала между событиями потока в i -м состоянии процесса $\lambda(t)$ есть случайная величина с функцией распределения $F_i(t) = 1 - e^{-(\lambda_i + \alpha_i)t}$, $i = 1, 2$.

В дальнейшем изложении принимается, что имеет место состояние S_1 (первое состояние) процесса $\lambda(t)$, если $\lambda(t) = \lambda_1$, и состояние S_2 (второе состояние) процесса $\lambda(t)$, если $\lambda(t) = \lambda_2$ ($\lambda_1 > \lambda_2 \geq 0$).

Очевидно, что в сделанных предположениях для рассматриваемого потока событий кусочно-постоянный случайный процесс $\lambda(t)$ является марковским.

Матрицы инфинитезимальных характеристик процесса $\lambda(t)$ имеют вид

$$D_0 = \begin{vmatrix} -(\lambda_1 + \alpha_1) & 0 \\ 0 & -(\lambda_2 + \alpha_2) \end{vmatrix},$$

$$D_1 = \begin{vmatrix} \lambda_1 P_1^{(1)}(\lambda_1 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_1 | \lambda_1) & \lambda_1 P_1^{(1)}(\lambda_2 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_2 | \lambda_1) \\ \lambda_2 P_1^{(1)}(\lambda_1 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_1 | \lambda_2) & \lambda_2 P_1^{(1)}(\lambda_2 | \lambda_2) + \alpha_2 P_1^{(2)}(\lambda_2 | \lambda_2) \end{vmatrix}.$$

Диагональными элементами матрицы D_0 являются интенсивности выхода процесса $\lambda(t)$ из своих состояний, взятые с противоположным знаком; элементами, лежащими вне главной диагонали – интенсивности переходов из состояния в состояние без наступления события. Элементы матрицы D_1 – это интенсивности переходов из состояния в состояние процесса $\lambda(t)$ при наступлении события потока.

Рассматривается ситуация, при которой не все события доступны наблюдению. После каждого зарегистрированного в момент времени t_k события наступает время фиксированной длительности T (мёртвое время), в течение которого другие события исследуемого потока теряются. При этом события, наступившие в течение мёртвого времени, не вызывают продления его периода, т.е. имеет место непродлевающееся мёртвое время. По окончании мёртвого времени первое наступившее событие снова порождает период ненаблюдаемости длительности T и т.д.

Пример описанной ситуации приведен на рис. 1, где S_1, S_2 – состояния ненаблюдаемого случайного процесса $\lambda(t)$, t_1, t_2, \dots – моменты наступления событий в наблюдаемом потоке, штриховкой обозначены периоды мёртвого времени фиксированной длительности T ; белыми кружками обозначены наблюдаемые события, черными – события, недоступные наблюдению.

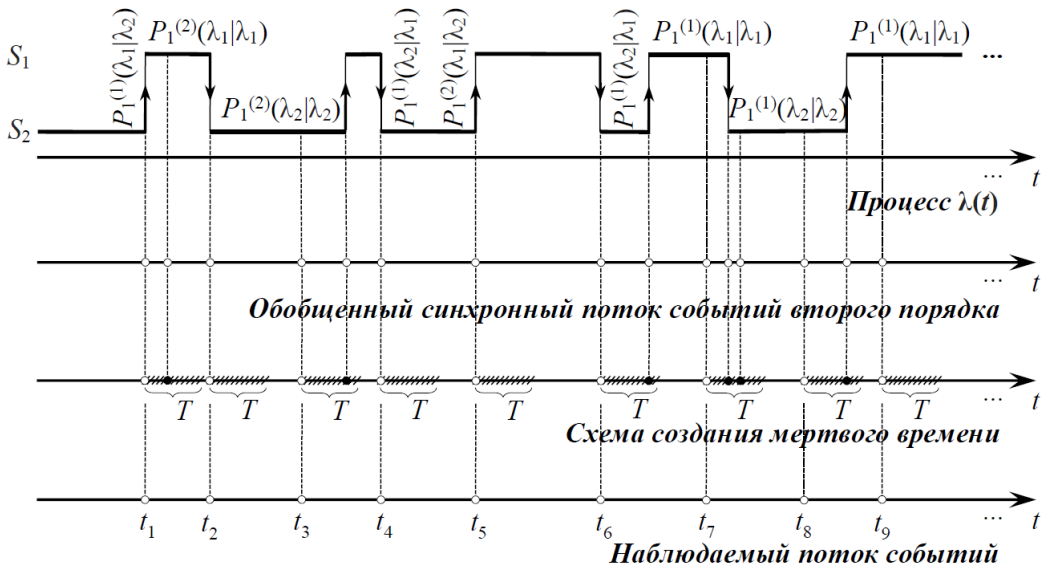


Рис. 1. Формирование наблюдаемого потока событий

В данной работе ставится цель построения имитационной модели обобщённого синхронного потока событий второго порядка, являющегося дважды стохастическим потоком событий и представляющим собой актуальную математическую модель реального информационного потока в телекоммуникационных системах, при этом функционирование потока рассматривается в условиях непродлевающегося мёртвого времени.

2. Имитационная модель потока

Имитационная модель обобщённого синхронного потока второго порядка с непродлевающимся мертвым временем разработана в виде программного кода на языке программирования C# в среде Microsoft Visual Studio 2013 по формулам, полученным методом обратных функций [18], согласно которому формула моделирования длительностей интервалов между соседними событиями в исследуемом потоке имеет вид:

$$\tau = -\frac{1}{\beta_i} \ln(1 - \gamma), \quad \beta_i \in \{\lambda_i, \alpha_i\},$$
 где γ – равномерно распределенная на $(0,1)$ случайная величина, λ_i – параметр экспоненциального распределения случайной величины $\xi_i^{(1)}$, α_i – параметр экспоненциального распределения случайной величины $\xi_i^{(2)}$, $i = 1, 2$.

Применение математического аппарата имитационного моделирования для исследования сложных систем, изучаемых в рамках ТМО, позволяет проводить желаемое число экспериментов и испытаний с различными входными параметрами и оценивать характеристики системы в определённые моменты времени. При этом для имитационного моделирования как вида компьютерного моделирования характерно воспроизведение (имитация) на ЭВМ процесса функционирования исследуемых систем с наглядным (визуальным) представлением результатов.

3. Статистические эксперименты на имитационной модели

Первый эксперимент заключается в проверке адекватности результатов, получаемых посредством работы имитационной модели в условиях полной наблюдаемости исследуемого потока событий (отсутствия мёртвого времени), т.е. $T = 0$.

Фиксируются значение времени моделирования (времени наблюдения за потоком) $T_m = 100$ и исходные данные, приведенные в табл. 1. Результаты первого эксперимента приведены в табл. 2.

Исходные данные для эксперимента

Таблица 1

$\lambda_1 = 3, \lambda_2 = 2$	$P_1^{(1)}(\lambda_1 \lambda_1) = 0,2$	$P_1^{(1)}(\lambda_2 \lambda_2) = 0,4$	$P_1^{(2)}(\lambda_1 \lambda_1) = 0,7$	$P_1^{(2)}(\lambda_2 \lambda_2) = 0,9$
$\alpha_1 = 2, \alpha_2 = 1$	$P_1^{(1)}(\lambda_2 \lambda_1) = 0,8$	$P_1^{(1)}(\lambda_1 \lambda_2) = 0,6$	$P_1^{(2)}(\lambda_2 \lambda_1) = 0,3$	$P_1^{(2)}(\lambda_1 \lambda_2) = 0,1$

Результаты первого эксперимента

Таблица 2

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	...	t_{327}
t_{flow}	0,0587	0,1506	0,3252	0,7289	1,0930	1,2034	1,2867	1,7695	1,8594	...	99,6587
S	S_1	S_2	S_2	S_1	S_2	S_2	S_1	S_1	S_2	...	S_1
	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	...	t_{327}
t	0,0587	0,1506	0,3252	0,7289	1,0930	1,2034	1,2867	1,7695	1,8594	...	99,6587

Здесь и далее t_{flow} – значения моментов времени наступления событий в обобщённом синхронном потоке второго порядка, S – состояние случайного процесса $\lambda(t)$, в которое процесс переходит (или в котором остается) в момент наступления очередного события потока, t – значения моментов времени наступления событий в наблюдаемом потоке событий.

В данном эксперименте моменты времени наступления событий в наблюдаемом потоке полностью совпадают с моментами наступления событий в рассматриваемом обобщённом синхронном потоке событий второго порядка, т.е. ни одно из событий исходного потока не теряется, что не противоречит результатам, полученным в [16].

Приведенные в табл. 3 результаты *второго эксперимента*, проделанного для исходных данных табл. 1, $T_m = 100$, $T = 0,5$, позволяют отметить существенное сокращение числа событий в наблюдаемом потоке (в сравнении с первым экспериментом) за счёт введения мёртвого времени. В табл. 3 и далее в табл. 5 тёмной диагональной штриховкой отмечены события исходного потока, потерянные в соответствующие моменты времени. Табл. 5 содержит результаты, полученные для исходных данных, представленных в табл. 4. В совокупности данные табл. 3 и 5 отражают зависимость между параметрами потока ($\lambda_1, \lambda_2, \alpha_1, \alpha_2$ и вероятностями $P_1^{(l)}(\lambda_j | \lambda_i)$, $i, j, l = 1, 2$) и количеством событий в исходном потоке, наступающих в процессе реализации испытания модели, а также количеством событий в наблюдаемом потоке событий.

Таблица 3

Результаты второго эксперимента

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	...	t_{329}
t_{flow}	0,1192	0,2187	0,5474	0,7168	0,9811	1,5483	1,7438	2,3463	2,4658	...	99,5303
S	S_2	S_2	S_2	S_2	S_1	S_2	S_1	S_1	S_1	...	S_2
	t_1			t_2		t_3		t_4		...	t_{128}
t	0,1192			0,7168		1,5483		2,3463		...	99,5303

Таблица 4

Исходные данные для эксперимента

$\lambda_1 = 5, \lambda_2 = 3$	$P_1^{(1)}(\lambda_1 \lambda_1) = 0,2$	$P_1^{(1)}(\lambda_2 \lambda_2) = 0,4$	$P_1^{(2)}(\lambda_1 \lambda_1) = 0,3$	$P_1^{(2)}(\lambda_2 \lambda_2) = 0,1$
$\alpha_1 = 4, \alpha_2 = 2$	$P_1^{(1)}(\lambda_2 \lambda_1) = 0,8$	$P_1^{(1)}(\lambda_1 \lambda_2) = 0,6$	$P_1^{(2)}(\lambda_2 \lambda_1) = 0,7$	$P_1^{(2)}(\lambda_1 \lambda_2) = 0,9$

Таблица 5

Результаты второго эксперимента

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	...	t_{511}
t_{flow}	0,1259	0,6695	0,7668	0,9543	1,1119	1,2198	1,7522	1,8425	1,9982	...	99,1119
S	S_2	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	...	S_1
	t_1	t_2				t_3	t_4			...	t_{150}
t	0,1259	0,6695				1,2198	1,7522			...	98,8273

Полученные в табл. 3 и 5 численные результаты свидетельствуют о том, что в связи с увеличением значений параметров потока λ_i и α_i , $i = 1, 2$, отмечается увеличение количества наступивших в потоке событий, однако в наблюдаемом потоке количество событий в относительном отношении сократилось ($128 / 329 > 150 / 511$) ввиду наличия мёртвого времени.

Оценка значения длительности интервала между событиями в потоке при наличии мёртвого времени определяется в *третьем эксперименте* следующим образом. При фиксированных значениях параметров потока, времени моделирования $T_m = 100$ и количестве повторений эксперимента $N = 100$ для k -й реализации определяется оценка (выборочное среднее) значения длительности интервала между соседними событиями в наблюдаемом потоке $\hat{\tau}_k = \frac{1}{n_k} \sum_{r=1}^{n_k} \tau_r^{(k)}$, где $k = \overline{1, N}$, $\tau_r^{(k)}$ – значение длительности r -го интервала между событиями в k -й реализации потока, n_k – количество интервалов за

время T_m . На основании полученного набора $\hat{\tau}_1, \hat{\tau}_2, \dots, \hat{\tau}_N$ вычисляется оценка среднего значения длительности интервала между событиями в обобщённом синхронном потоке второго порядка при неполной его наблюдаемости $\hat{\tau} = \frac{1}{N} \sum_{r=1}^N \hat{\tau}_r$.

Результаты эксперимента, полученные для заданных в табл. 6 значений параметров потока, приведены в табл. 7. На рис. 2 представлен график зависимости отношения $\hat{\tau} / T$ от значения мёртвого времени T , построенный по данным табл. 7.

Таблица 6

Исходные данные для эксперимента

$\lambda_1 = 5, \lambda_2 = 3$	$P_1^{(1)}(\lambda_1 \lambda_1) = 0,5$	$P_1^{(1)}(\lambda_2 \lambda_2) = 0,4$	$P_1^{(2)}(\lambda_1 \lambda_1) = 0,6$	$P_1^{(2)}(\lambda_2 \lambda_2) = 0,5$
$\alpha_1 = 4, \alpha_2 = 2$	$P_1^{(1)}(\lambda_2 \lambda_1) = 0,5$	$P_1^{(1)}(\lambda_1 \lambda_2) = 0,6$	$P_1^{(2)}(\lambda_2 \lambda_1) = 0,4$	$P_1^{(2)}(\lambda_1 \lambda_2) = 0,5$

Таблица 7

Результаты третьего эксперимента

T	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1,0	1,1	1,2	1,3
$\hat{\tau}$	0,2577	0,3630	0,4641	0,5656	0,6687	0,7689	0,8663	0,9683	1,0646	1,1645	1,2634	1,3614	1,4627
$\hat{\tau} / T$	2,5770	1,8150	1,5470	1,4140	1,3374	1,2815	1,2376	1,2104	1,1829	1,1645	1,1485	1,1345	1,1252

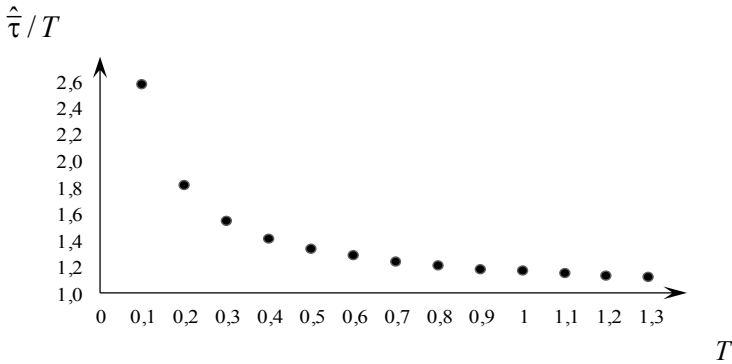


Рис. 2. График зависимости $\hat{\tau} / T$ от значения T

Оценка среднего значения длительности интервала между событиями в наблюдаемом потоке $\hat{\tau}$ при заданном наборе параметров прямо пропорциональна значению мёртвого времени T , чему полностью соответствуют результаты работы имитационной модели (табл. 7). Отметим, что при увеличении T значение отношения $\hat{\tau} / T$ уменьшается и стремится к некоторой константе (рис. 2).

Предметом исследования следующих двух экспериментов являются оценки значений длительностей пребывания процесса $\lambda(t)$ в i -м состоянии, $i = 1, 2$. Для k -й реализации обобщённого синхронного потока второго порядка находятся $\hat{T}_i^{(k)}$, $i = 1, 2$, $k = \overline{1, N}$, – суммы значений длительностей всех интервалов, в течение которых имеют место состояния S_i процесса $\lambda(t)$, $i = 1, 2$. С использованием полученной выборки $\hat{T}_i^{(1)}, \hat{T}_i^{(2)}, \dots, \hat{T}_i^{(N)}$ вышеуказанные оценки определяются как $\hat{T}_i = \frac{1}{N} \sum_{r=1}^N \hat{T}_i^{(r)}$, $i = 1, 2$. В четвёртом и пятом экспериментах устанавливается зависимость оценок \hat{T}_i , $i = 1, 2$, от количества реализаций N и значений вероятностей $P_1^{(l)}(\lambda_j | \lambda_i)$, $i, j, l = 1, 2$, соответственно; расчёты производились при значении времени моделирования $T_m = 100$.

В качестве исходных данных для проведения четвертого эксперимента взяты данные табл. 8; отметим, что в данном случае рассматривается частный случай задания вероятностей $P_1^{(l)}(\lambda_j | \lambda_i)$, $i, j, l = 1, 2$, согласно которому имеют место переходы процесса $\lambda(t)$ из первого состояния во второе, либо наоборот, в каждый момент наступления события потока. Результаты имитационного моделирования потока в рамках описанного эксперимента представлены в табл. 9.

Таблица 8

Исходные данные для эксперимента

$\lambda_1 = 2, \lambda_2 = 1$	$P_1^{(1)}(\lambda_1 \lambda_1) = 0$	$P_1^{(1)}(\lambda_2 \lambda_2) = 0$	$P_1^{(2)}(\lambda_1 \lambda_1) = 0$	$P_1^{(2)}(\lambda_2 \lambda_2) = 0$
$\alpha_1 = 0,8, \alpha_2 = 0,6$	$P_1^{(1)}(\lambda_2 \lambda_1) = 1$	$P_1^{(1)}(\lambda_1 \lambda_2) = 1$	$P_1^{(2)}(\lambda_2 \lambda_1) = 1$	$P_1^{(2)}(\lambda_1 \lambda_2) = 1$

Таблица 9

Результаты четвертого эксперимента

N	100	200	300	400	500	600	700	800	900	1000	1100
\hat{T}_1	37,2548	36,5535	36,9937	36,9760	37,1792	37,2124	36,9136	37,1121	37,0087	37,0662	37,0407
\hat{T}_2	62,2352	62,9068	62,4192	62,4939	62,2719	62,2993	62,5643	62,5034	62,5825	62,4177	62,5272

Рис. 3 и 4 иллюстрируют графики исследуемых зависимостей, построенные по данным табл. 9.

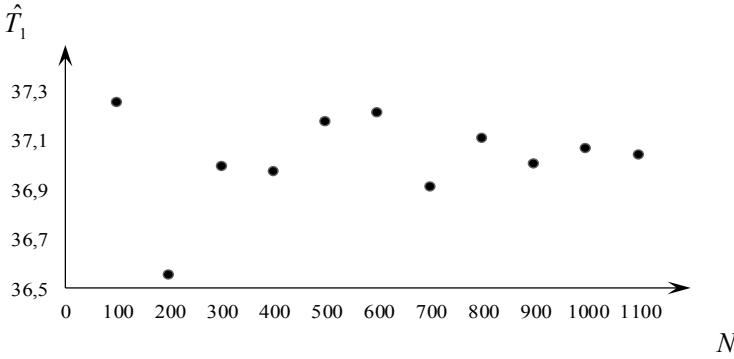


Рис. 3. График зависимости \hat{T}_1 от значения N

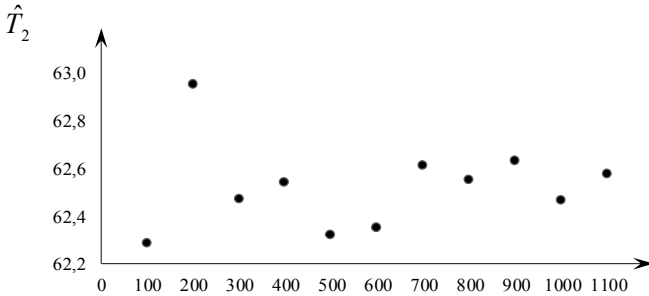


Рис. 4. График зависимости \hat{T}_2 от значения N

Анализируя численные результаты, отражённые в табл. 9, и их графическое представление (рис. 3 и 4), отмечаем колебательное поведение оценок \hat{T}_1 и \hat{T}_2 , однако, при существенном увеличении количества реализаций каждая из оценок устанавливается в

пределах своего стационарного значения, соответствующего входным параметрам потока (табл. 8).

Полученные при исходных данных табл. 10 результаты пятого эксперимента приведены в табл. 11, в первой строке которой указаны значения вероятностей $P_1^{(1)}(\lambda_1 | \lambda_1) = P_1^{(2)}(\lambda_1 | \lambda_1) = P$, $0 \leq P \leq 1$. В настоящем эксперименте количество реализаций фиксировано, $N = 100$.

Таблица 10

Исходные данные для эксперимента

$\lambda_1 = 2$	$P_1^{(1)}(\lambda_1 \lambda_1) = 0, 0, 1, \dots, 1$	$P_1^{(1)}(\lambda_2 \lambda_2) = 0$
$\lambda_2 = 1$	$P_1^{(1)}(\lambda_2 \lambda_1) = 1 - P_1^{(1)}(\lambda_1 \lambda_1)$	$P_1^{(1)}(\lambda_1 \lambda_2) = 1$
$\alpha_1 = 0,8$	$P_1^{(2)}(\lambda_1 \lambda_1) = 0, 0, 1, \dots, 1$	$P_1^{(2)}(\lambda_2 \lambda_2) = 0$
$\alpha_2 = 0,6$	$P_1^{(2)}(\lambda_2 \lambda_1) = 1 - P_1^{(2)}(\lambda_1 \lambda_1)$	$P_1^{(2)}(\lambda_1 \lambda_2) = 1$

Таблица 11

Результаты пятого эксперимента

P	0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1,0
\hat{T}_1	37,0357	39,8866	43,0757	46,6970	49,5492	54,2339	60,1916	67,2245	74,4207	85,8957	99,3231
\hat{T}_2	62,4241	60,6946	57,4286	53,8189	51,0073	46,2506	40,3793	33,2874	26,1034	14,7441	0,3131

Построенный по данным табл. 11 график зависимостей оценок \hat{T}_1 и \hat{T}_2 от значений вероятностей P представлен на рис. 5.

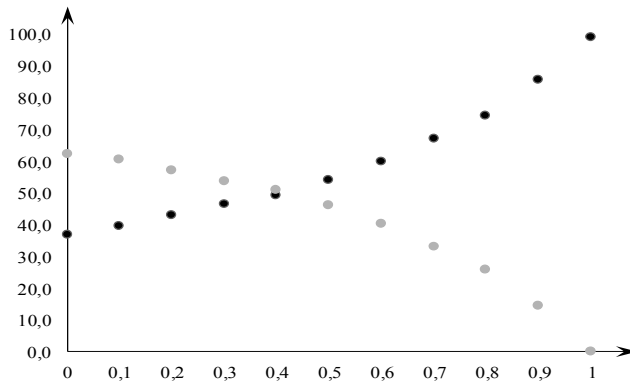


Рис. 5. График зависимости \hat{T}_1 от значений P – тёмным маркером, \hat{T}_2 от значений P – светлым маркером

Полученные численные результаты пятого статистического эксперимента на имитационной модели обобщённого синхронного потока событий второго порядка свидетельствуют о сильной чувствительности оценок значений длительностей пребывания процесса $\lambda(t)$ в i -м состоянии, $i=1,2$, к изменениям значений вероятностей $P_1^{(l)}(\lambda_2 | \lambda_1)$ и $P_1^{(l)}(\lambda_1 | \lambda_1)$, $l=1,2$, согласно которым процесс $\lambda(t)$ либо переходит из первого состояния во второе, либо остается в первом состоянии (в зависимости от значений случайных величин $\xi_1^{(l)}$, $l=1,2$). Рассмотренный эксперимент может быть проделан в условиях, аналогичных приведенным, при измененных значениях параметров

потока λ_i , α_i , $i = 1, 2$, и $P_1^{(1)}(\lambda_2 | \lambda_2) = P_1^{(2)}(\lambda_2 | \lambda_2) = P$, $0 \leq P \leq 1$, в результате чего будет иметь место аналогичная ситуация, а именно сильная зависимость оценок \hat{T}_1 и \hat{T}_2 от значений вероятностей $P_1^{(l)}(\lambda_1 | \lambda_2)$ и $P_1^{(l)}(\lambda_2 | \lambda_2)$, $l = 1, 2$. Этот факт позволяет обобщить результат и сделать вывод о сильной чувствительности оценок \hat{T}_1 и \hat{T}_2 к вариациям значений вероятностей $P_1^{(l)}(\lambda_j | \lambda_i)$, $i, j, l = 1, 2$.

Заключение

В данной работе рассмотрен обобщённый синхронный поток событий второго порядка с непродлевающимся мёртвым временем, для которого построена имитационная модель в виде проекта приложения Windows Forms средствами объектно-ориентированного языка программирования C#. Анализ корректности и непротиворечивости результатов экспериментов, проведённых на имитационной модели потока, позволяет утверждать, что получена работоспособная модель, непротиворечащая входным данным и отвечающая им.

ЛИТЕРАТУРА

1. Дудин А.Н., Клименок В.И. Системы массового обслуживания с коррелированными потоками. – Минск: Изд-во Белорусского государственного университета, 2000. – 175 с.
2. Basharin G. P., Gaidamaka Yu. V., Samouylov K. E. Mathematical theory of teletraffic and its application to the analysis of multiservice communication of next generation networks // Automatic Control and Computer Sciences. – 2013. – Vol. 47. – No. 2. – P. 62–69.
3. Башарин Г.П., Кокотушкин В.А., Наумов В.А. О методе эквивалентных замен расчета фрагментов сетей связи // Изв. АН СССР. Техн. кибернетика. – 1979. – № 6. – С. 92–99.
4. Башарин Г.П., Кокотушкин В.А., Наумов В.А. О методе эквивалентных замен расчета фрагментов сетей связи // Изв. АН СССР. Техн. кибернетика. – 1980. – № 1. – С. 55–61.
5. Neuts M.F. A versatile Markov point process // Journal of Applied Probability. – 1979. – Vol. 16. – P. 764–779.
6. Горцев А.М., Леонова М.А., Нежелская Л.А. Совместная плотность вероятностей длительности интервалов обобщённого асинхронного потока событий при непродлеваемом мёртвом времени // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2012. – № 4 (21). – С. 14–25.
7. Горцев А.М., Калягин А.А., Нежелская Л.А. Совместная плотность вероятностей длительности интервалов обобщённого полусинхронного потока событий при непродлеваемом мёртвом времени // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2014. – № 2 (27). – С. 19–29.
8. Gortsev A.M., Nezhel'skaya L.A., Shevchenko T.I. Estimation of the states of an MC-stream of events in the presence of measurement errors // Russian Physics Journal. – 1993. – Vol. 36. – Is. 12. – P. 1153–1167.
9. Горцев А.М., Леонова М.А., Нежелская Л.А. Сравнение МП- и ММ-оценок длительности мёртвого времени в обобщённом асинхронном потоке событий // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2013. – № 4 (25). – С. 32–42.
10. Нежелская Л.А. Совместная плотность вероятностей длительности интервалов модулированного МАР-потока событий и условия рекуррентности потока // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2015. – № 1. – С. 57–67.
11. Горцев А.М., Нежелская Л.А. О связи МС-потоков и МАР-потоков событий // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2011. – № 1 (14). – С. 13–21.
12. Апанасович В.В., Коляда А.А., Чернявский А.Ф. Статистический анализ случайных потоков в физическом эксперименте. – Минск: Изд-во «Университетское», 1988. – 254 с.
13. Горцев А.М., Климов И.С. Оценка интенсивности пуассоновского потока событий в условиях частичной его ненаблюдаемости // Радиотехника. – 1991. – № 12. – С. 3–7.
14. Горцев А.М., Климов И.С. Оценивание параметров знакопеременного пуассоновского потока событий // Радиотехника. – 1994. – № 8. – С. 3–9.
15. Горцев А.М., Климов И.С. Оценивание периода ненаблюдаемости и интенсивности пуассоновского потока событий // Радиотехника. – 1996. – № 2. – С. 8–11.
16. Нежелская Л.А., Сидорова Е.Ф. Имитационное моделирование обобщённого синхронного потока второго порядка // Труды Томского государственного университета. Серия физико-математическая. Томск: Издательский Дом ТГУ, 2016. – Т. 299. – С. 104–109.
17. Нежелская Л.А., Сидорова Е.Ф. Алгоритм оптимального оценивания состояний обобщённого синхронного потока событий второго порядка // Труды Томского государственного университета. Серия физико-математическая. Томск: Издательский Дом ТГУ, 2017. – Т. 301. – С. 105–113.
18. Соболев И.М. Численные методы Монте-Карло. – М.: Наука. Гл. ред. физ.-мат. лит., 1973. – 312 с.

СТАТИСТИЧЕСКИЕ ЭКСПЕРИМЕНТЫ НА ИМИТАЦИОННОЙ МОДЕЛИ ПОЛУСИНХРОННОГО ПОТОКА СОБЫТИЙ ВТОРОГО ПОРЯДКА В УСЛОВИЯХ ЕГО НЕПОЛНОЙ НАБЛЮДАЕМОСТИ

Л.А. Нежелская, Д.А. Тумашкина

Томский государственный университет

ludne@mail.ru, diana1323@mail.ru

Введение

При описании и анализе реальных экономических, технических, физических и других процессов довольно часто возникает необходимость применять математические модели теории массового обслуживания. Основной задачей теории массового обслуживания является нахождение вероятностных характеристик, определяющих функциональные возможности систем массового обслуживания и эффективность их функционирования. В настоящее время в связи с бурным развитием информационных технологий важнейшими сферами приложений теории массового обслуживания являются проектирование и создание телекоммуникационных сетей, спутниковых сетей связи и глобальных компьютерных сетей, объединенных термином – цифровые сети интегрального обслуживания (ЦСИО). В связи с тем, что на практике параметры, определяющие случайный поток событий, как правило, случайным образом изменяются со временем, адекватными математическими моделями информационных потоков сообщений, функционирующих в ЦСИО, являются дважды стохастические потоки событий [1,2]. В данных потоках событий случайны не только моменты наступления событий, но и интенсивность потока представляет собой случайный процесс, т.е. имеет место двойная стохастика.

В данной работе рассматривается дважды стохастический поток событий с числом состояний, равным двум. В зависимости от того, каким образом происходит переход из состояния в состояние, данные потоки событий можно разделить на три типа: 1) синхронные потоки, переход из состояния в состояние в которых зависит непосредственно от наступления события [3]; 2) асинхронные потоки, переход из состояния в состояние в которых не зависит от того, наступило событие или нет [4]; 3) полусинхронные потоки, у которых для одного состояния справедливо определение первого типа, а для второго состояния – второго типа [5]. Объектом изучения настоящей работы является полусинхронный поток событий второго порядка.

В большинстве случаев рассматриваются модели входящих потоков событий, когда события потока полностью наблюдаемы. Однако на практике любое регистрирующее устройство затрачивает определенное время на регистрацию события, в течение которого устройство не способно обработать следующее событие, т.е. событие, поступившее на прибор, порождает период так называемого мёртвого времени [6], в течение которого другие наступившие события потока являются недоступными для наблюдения (теряются). Будем считать, что этот период продолжается некоторое фиксированное время (непродлевающееся мёртвое время).

В данной статье приводятся численные результаты ряда экспериментов, поставленных на разработанной имитационной модели потока в условиях его неполной наблюдаемости, и их анализ.

Отметим, что настоящая статья является непосредственным развитием работ [7,8].

1. Постановка задачи

Рассматривается стационарный режим функционирования полусинхронного дважды стохастического потока событий второго порядка (поток), сопровождающий случайный процесс которого $\lambda(t)$ является кусочно-постоянным с двумя состояниями S_1 и S_2 . Далее по тексту под i -м состоянием процесса понимается состояние S_i , $i = 1, 2$. Дли-

тельность интервала между событиями потока в первом состоянии определяется случайной величиной $\eta = \min(\xi^{(1)}, \xi^{(2)})$, где случайная величина $\xi^{(1)}$ имеет функцию распределения $F_1^{(1)}(t) = 1 - e^{-\lambda_1 t}$, случайная величина $\xi^{(2)}$ имеет функцию распределения $F_1^{(2)}(t) = 1 - e^{-\alpha_1 t}$; $\xi^{(1)}$ и $\xi^{(2)}$ – независимые случайные величины.

В момент наступления события потока процесс $\lambda(t)$ переходит из первого состояния во второе либо с вероятностью $P_1^{(1)}(\lambda_2 | \lambda_1)$, либо с вероятностью $P_1^{(2)}(\lambda_2 | \lambda_1)$ в зависимости от того, какое значение приняла случайная величина η . В момент наступления события потока процесс $\lambda(t)$ остается в первом состоянии либо с вероятностью $P_1^{(1)}(\lambda_1 | \lambda_1)$, либо с вероятностью $P_1^{(2)}(\lambda_1 | \lambda_1)$ в зависимости от значения случайной величины η . Здесь $P_1^{(1)}(\lambda_2 | \lambda_1) + P_1^{(1)}(\lambda_1 | \lambda_1) = 1$, $P_1^{(2)}(\lambda_2 | \lambda_1) + P_1^{(2)}(\lambda_1 | \lambda_1) = 1$. Длительность интервала между событиями потока в первом состоянии процесса $\lambda(t)$ является случайной величиной с функцией распределения $F(t) = 1 - e^{-(\lambda_1 + \alpha_1)t}$.

Длительность пребывания процесса $\lambda(t)$ во втором состоянии есть случайная величина с функцией распределения $F_2(t) = 1 - e^{-\alpha_2 t}$. В течение времени пребывания процесса $\lambda(t)$ во втором состоянии имеет место пуассоновский поток событий с параметром λ_2 .

В последующем изложении полагается, что имеет место состояние S_i (i -е состояние) процесса $\lambda(t)$, если $\lambda(t) = \lambda_i$, $i = 1, 2$; $\lambda_1 > \lambda_2 \geq 0$.

Утверждение. Процесс $\lambda(t)$ является марковским.

Доказательство. Длительность пребывания процесса $\lambda(t)$ в первом состоянии есть случайная величина, распределённая по экспоненциальному закону, с функцией распределения $F_1(t) = 1 - e^{-[\lambda_1 P_1^{(1)}(\lambda_2 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_2 | \lambda_1)]t}$. В свою очередь, длительность пребывания процесса $\lambda(t)$ во втором состоянии есть случайная величина, распределённая по экспоненциальному закону $F_2(t) = 1 - e^{-\alpha_2 t}$. Отсюда следует формулировка утверждения.

Поскольку процесс $\lambda(t)$ является принципиально ненаблюдаемым, наблюдаются только моменты времени наступления событий t_1, t_2, \dots , тогда $\lambda(t)$ – скрытый марковский процесс или ненаблюдаемый сопровождающий марковский процесс.

После каждого зарегистрированного в момент времени t_k события наступает период мёртвого времени фиксированной длительности T , в течение которого другие события рассматриваемого потока являются недоступными наблюдению. По окончании периода мёртвого времени первое наступившее событие снова создаёт период мёртвого времени длительности T и т.д. (непродлевающееся мёртвое время).

Матрицы инфинитезимальных характеристик процесса $\lambda(t)$ имеют вид

$$D_0 = \begin{vmatrix} -(\lambda_1 + \alpha_1) & 0 \\ \alpha_2 & -(\lambda_2 + \alpha_2) \end{vmatrix},$$

$$D_1 = \begin{vmatrix} \lambda_1 P_1^{(1)}(\lambda_1 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_1 | \lambda_1) & \lambda_1 P_1^{(1)}(\lambda_2 | \lambda_1) + \alpha_1 P_1^{(2)}(\lambda_2 | \lambda_1) \\ 0 & \lambda_2 \end{vmatrix}.$$

Элементами матрицы D_1 являются интенсивности переходов процесса $\lambda(t)$ из состояния в состояние с наступлением события. Недиагональные элементы матрицы D_0 – интенсивности переходов из состояния в состояние без наступления события; диаго-

нальные элементы матрицы D_0 – интенсивности выхода процесса $\lambda(t)$ из своих состояний, взятые с противоположным знаком.

Для наглядности на рис. 1 приведён пример возникающей ситуации, где $t_1, t_2, \dots, t_n, \dots$ – моменты наступления событий в наблюдаемом потоке; периоды мёртвого времени длительности T обозначены пунктиром; чёрными кружками обозначены события полусинхронного потока событий второго порядка, недоступные наблюдению.

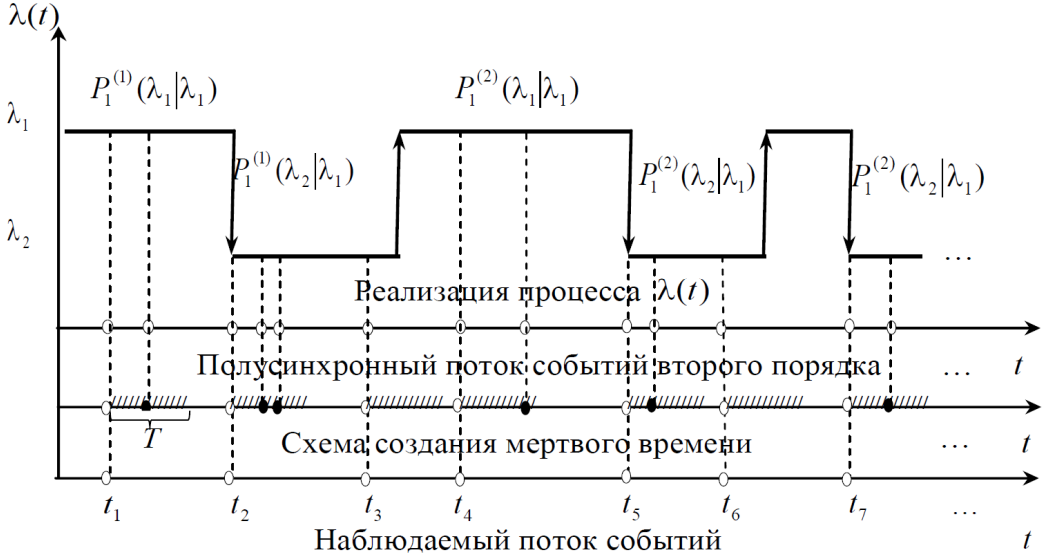


Рис. 1. Формирование наблюдаемого потока событий

Задача настоящей работы заключается в построении имитационной модели полусинхронного потока событий второго порядка в условиях его неполной наблюдаемости, а также проведении статистических экспериментов с целью проверки работоспособности построенной модели.

2. Результаты имитационного моделирования

Моделирование полусинхронного потока второго порядка в условиях непродлевающегося мёртвого времени осуществлено по формулам, которые получены методом обратных функций из экспоненциального распределения.

Пусть длительность пребывания процесса $\lambda(t)$ в i -м состоянии имеет плотность распределения вероятностей $p_i(t) = \lambda e^{-\theta t}$ и функцию распределения $F_i(t) = 1 - e^{-\theta t}$, $i = 1, 2$. В соответствии с методом обратных функций [9] обозначим $F_i(t) = \gamma$. Тогда имеем $\gamma = 1 - e^{-\theta t}$, откуда находим формулу моделирования длительности интервалов между соседними событиями в полусинхронном потоке второго порядка:

$$t = -\frac{1}{\theta_i} \ln(1 - \gamma), \quad \theta_i \in \{\lambda_i, \alpha_i\},$$

где γ – равномерно распределенная на интервале $(0, 1)$ случайная величина.

Ниже приведены результаты экспериментов, заключающихся в изучении поведения полусинхронного потока событий второго порядка в условиях его неполной наблюдаемости при различных наборах вероятностей перехода и интенсивностей.

Эксперимент 1. Иллюстрируется поведение процесса $\lambda(t)$ ($T = 0$).

Пусть $\lambda_1 = 2$, $\lambda_2 = 0,8$, $\alpha_1 = 2$, $\alpha_2 = 0,9$, время наблюдения за потоком (время моделирования) $T_m = 10$. Примем крайние значения вероятностей перехода:

- 1) $P_1^{(1)}(\lambda_1|\lambda_1) = P_1^{(2)}(\lambda_1|\lambda_1) = 1$ и $P_1^{(1)}(\lambda_2|\lambda_1) = P_1^{(2)}(\lambda_2|\lambda_1) = 0$ (первый случай);
- 2) $P_1^{(1)}(\lambda_2|\lambda_1) = P_1^{(2)}(\lambda_2|\lambda_1) = 1$ и $P_1^{(1)}(\lambda_1|\lambda_1) = P_1^{(2)}(\lambda_1|\lambda_1) = 0$ (второй случай).

Результаты данного эксперимента приведены в табл. 1, 2. В строках приведенных таблиц указаны моменты наступления событий.

Таблица 1

Результаты работы имитационной модели в первом случае эксперимента 1

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
0,3569	1,3563	2,0658	3,5841	4,7961	4,9855	7,1097	8,4376

Таблица 2

Результаты работы имитационной модели во втором случае эксперимента 1

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}	t_{12}	t_{13}
1,4782	2,1383	2,9171	3,8475	4,2816	4,3958	4,7915	5,0731	6,2391	8,0174	8,5242	8,9982	9,4485

На рис. 2, 3 приведены реализации случайного процесса $\lambda(t)$ для эксперимента 1.

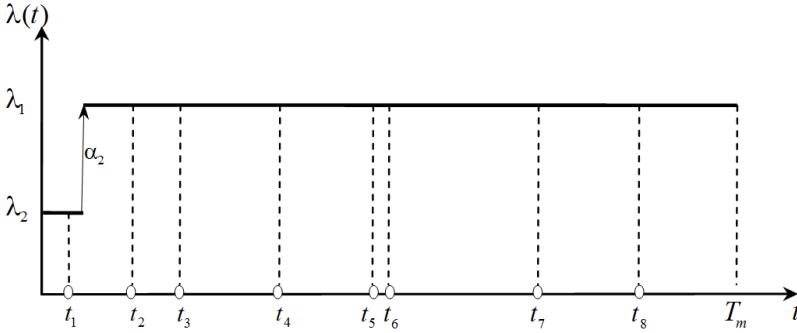


Рис. 2. Реализация потока в первом случае

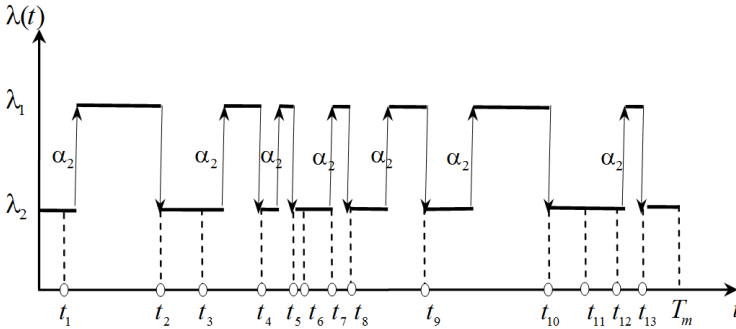


Рис. 3. Реализация потока во втором случае

В первом случае, т.е. при $P_1^{(1)}(\lambda_1|\lambda_1) = P_1^{(2)}(\lambda_1|\lambda_1) = 1$, процесс $\lambda(t)$, попав в первое состояние, в нём и остается. Во втором случае процесс $\lambda(t)$, находясь в первом состоянии, при наступлении события переходит во второе состояние с вероятностью единица.

Эксперимент 2. Пусть $\alpha_1 = 1$, $\alpha_2 = 0,8$, $P_1^{(1)}(\lambda_1|\lambda_1) = P_1^{(2)}(\lambda_1|\lambda_1) = 0,6$, $P_1^{(1)}(\lambda_2|\lambda_1) = P_1^{(2)}(\lambda_2|\lambda_1) = 0,4$, $T_m = 10$, $T = 2$. Рассмотрим результаты моделирования в двух случаях:

- 1) $\lambda_1 = 2$, $\lambda_2 = 0,5$ (первый случай);
- 2) $\lambda_1 = 4$, $\lambda_2 = 1$ (второй случай).

Результаты данного эксперимента приведены в табл. 3, 4. В строках приведённых таблиц указаны моменты наступления событий.

Таблица 3

Результаты работы имитационной модели в первом случае эксперимента 2

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
t_{initial}	0,3652	2,4587	3,5693	3,9556	4,7505	6,8275	7,8650	9,9108
	t_1	t_2			t_5	t_6		t_8
t_{observed}	0,3652	2,4587			4,7505	6,8275		9,9108

Здесь t_{initial} – моменты наступления событий полусинхронного потока второго порядка, t_{observed} – моменты наступления событий в наблюдаемом потоке; в табл. 3 и далее в табл. 4 штриховкой отмечены события исходного потока, потерянные в соответствующие моменты времени.

Таблица 4

Результаты работы имитационной модели во втором случае эксперимента 2

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	...	t_{16}
t_{initial}	0,4679	0,9563	1,3675	2,4810	2,6585	3,1915	3,7698	4,6512	...	9,8677
	t_1			t_4				t_8	...	t_{16}
t_{observed}	0,4679			2,4810				4,6512	...	9,8677

Отметим, что с увеличением интенсивностей потока (табл. 4) увеличилось количество событий, наступивших в полусинхронном потоке событий второго порядка.

На имитационной модели поставлены два статистических эксперимента, демонстрирующих ее работоспособность.

1. *Первый статистический эксперимент* состоит из следующих этапов:

- 1) для фиксированного набора параметров, вероятностей переходов и длительности мёртвого времени моделируется полусинхронный поток событий второго порядка на отрезке времени длиной T_m (j -й эксперимент);
- 2) осуществляется расчет $\hat{T}_i^{(j)}$, $i = 1, 2$ (для j -го эксперимента) – оценок длительности пребывания процесса $\lambda(t)$ в i -м состоянии;
- 3) осуществляется повторение шагов 1, 2 N раз ($j = \overline{1, N}$).

Результатом выполнения приведенного выше алгоритма являются выборки оценок $\hat{T}_i^{(j)}$, $i = 1, 2$, в N опытах, $j = \overline{1, N}$, по которым находятся выборочные средние длительностей пребывания процесса $\lambda(t)$ в i -м состоянии $\hat{T}_i = \frac{1}{N} \sum_{j=1}^N \hat{T}_i^{(j)}$, $i = 1, 2$.

Результаты первого статического эксперимента приведены в табл. 5, 6, а также отражены на рис. 4–7.

В данном эксперименте при $T_m = 100$, $T = 1$, $\lambda_1 = 4$, $\lambda_2 = 2$, $\alpha_1 = 4$, $P_1^{(1)}(\lambda_1|\lambda_1) = P_1^{(2)}(\lambda_1|\lambda_1) = 0,6$, $P_1^{(1)}(\lambda_2|\lambda_1) = P_1^{(2)}(\lambda_2|\lambda_1) = 0,4$ рассматривается зависимость изменения \hat{T}_1 , \hat{T}_2 при значениях $N = 50, \dots, 500$ с шагом 50. Рассмотрим результаты моделирования в двух случаях:

- 1) $\alpha_2 = 3$ (первый случай);
- 2) $\alpha_2 = 0,5$ (второй случай).

Таблица 5

Зависимость \hat{T}_1, \hat{T}_2 от значения N при $\alpha_2 = 3$

N	50	100	150	200	250	300	350	400	450	500
\hat{T}_1	65,02	63,19	64,22	62,77	63,28	64,2	64,35	64,32	64,62	64,33
\hat{T}_2	34,98	36,81	35,78	37,23	36,72	35,8	35,65	35,68	35,38	35,67

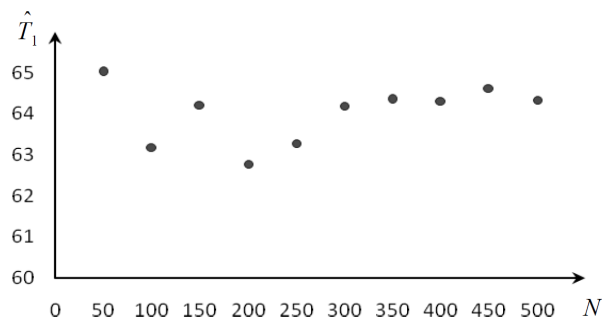


Рис. 4. График зависимости \hat{T}_1 от значения N при $\alpha_2 = 3$

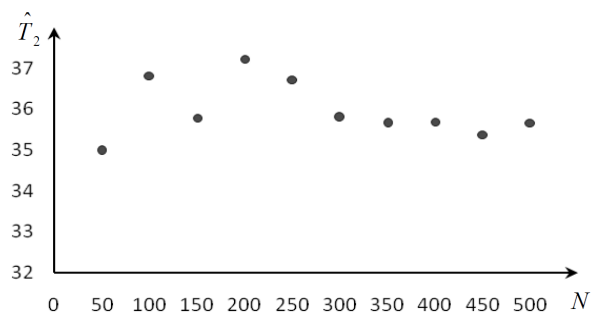


Рис. 5. График зависимости \hat{T}_2 от значения N при $\alpha_2 = 3$

Таблица 6

Зависимость \hat{T}_1, \hat{T}_2 от значения N при $\alpha_2 = 0,5$

N	50	100	150	200	250	300	350	400	450	500
\hat{T}_1	25,11	26,01	24,72	23,61	25,44	24,37	24,56	24,42	24,61	24,63
\hat{T}_2	74,89	73,99	75,28	76,39	74,56	75,63	75,44	75,58	75,39	75,37

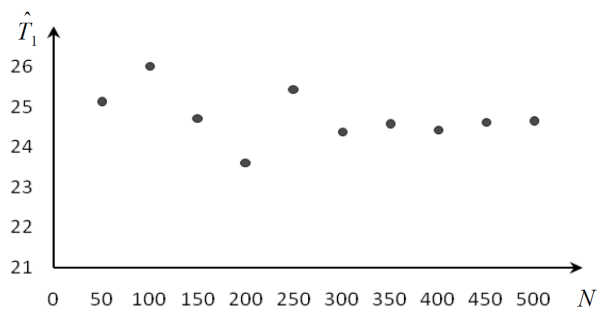


Рис. 6. График зависимости \hat{T}_1 от значения N при $\alpha_2 = 0,5$

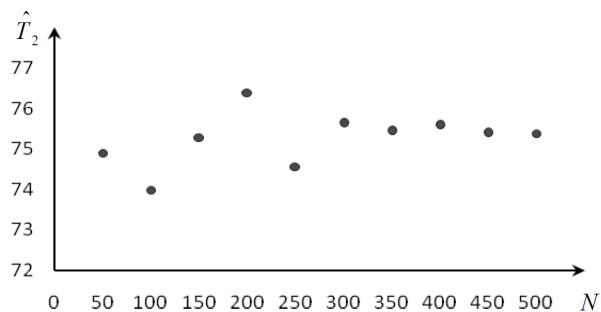


Рис. 7. График зависимости \hat{T}_2 от значения N при $\alpha_2 = 0,5$

Анализ численных результатов, приведенных в табл. 5, 6, показывает, что разброс значений оценок \hat{T}_1 , \hat{T}_2 уменьшается с увеличением N для обоих рассматриваемых случаев, т.е. \hat{T}_1 , \hat{T}_2 с ростом N ведут себя стабильнее, а также оценка \hat{T}_2 увеличивается с уменьшением параметра α_2 (табл. 6), что является естественным в силу того, что при малых α_2 процесс $\lambda(t)$ преимущественно находится во втором состоянии для значений параметров, заданных выше.

II. Во *втором статистическом эксперименте* оценивается значение длительности интервала между наблюдаемыми событиями в рассматриваемом потоке в зависимости от значения длительности мёртвого времени. Данный эксперимент состоит из трёх этапов:

1) для фиксированного набора параметров, вероятностей переходов и длительности мёртвого времени моделируется полусинхронный поток событий второго порядка на отрезке времени длиной T_m (j -й эксперимент);

2) осуществляется расчёт $\hat{\tau}_j = \frac{1}{k_j} \sum_{i=1}^{k_j} \tau_i^{(j)}$, (для j -го эксперимента) – выборочных

средних значений длительности интервала между событиями в наблюдаемом потоке, где $\tau_i^{(j)}$ – длительность i -го интервала между событиями в наблюдаемом потоке, k_j – количество интервалов между событиями в наблюдаемом потоке в j -ой реализации;

3) осуществляется повторение шагов 1,2 N раз ($j = \overline{1, N}$).

Результатом выполнения приведенного выше алгоритма является выборка средних

$\hat{\tau}_j$ в N опытах, $j = \overline{1, N}$, по которой находится оценка среднего $\hat{\bar{\tau}} = \frac{1}{N} \sum_{j=1}^N \hat{\tau}_j$.

В табл. 7 и рис. 8 приведены результаты второго статистического эксперимента. В данном эксперименте при $T_m = 100$, $\lambda_1 = 4$, $\lambda_2 = 2$, $\alpha_1 = 4$, $\alpha_2 = 2$, $P_1^{(1)}(\lambda_1|\lambda_1) = P_1^{(2)}(\lambda_1|\lambda_1) = 0,4$, $P_1^{(1)}(\lambda_2|\lambda_1) = P_1^{(2)}(\lambda_2|\lambda_1) = 0,6$ рассматривается зависимость изменения $\hat{\bar{\tau}}$ при значениях $T = 0,5, \dots, 1,5$ с шагом 0,1.

Таблица 7

Зависимость $\hat{\bar{\tau}}$ от значения T

T	0,5	0,6	0,7	0,8	0,9	1	1,1	1,2	1,3	1,4	1,5
$\hat{\bar{\tau}}$	0,36	0,44	0,52	0,57	0,65	0,72	0,79	0,86	0,94	1,02	1,11

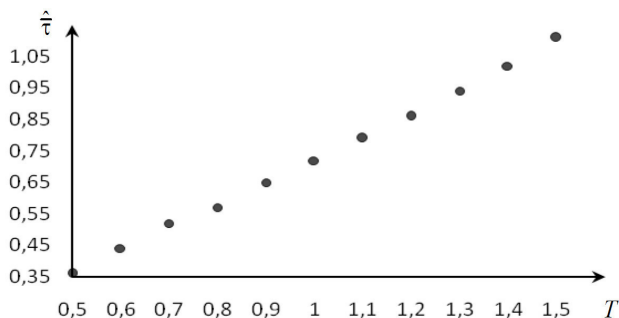


Рис. 8. График зависимости $\hat{\tau}$ от значения T

Анализируя численные результаты, приведенные в табл. 7, а также графическое представление зависимости $\hat{\tau}$ от значения T , отраженное на рис. 8, можно сделать следующий вывод: оценка $\hat{\tau}$ увеличивается с увеличением T .

Заключение

В настоящей работе рассмотрен полусинхронный дважды стохастический поток событий второго порядка в условиях его неполной наблюдаемости. Построена имитационная модель данного потока в условиях непродлевающегося мёртвого времени на языке программирования С# в среде Visual Studio 2013, а также приведены численные результаты статистических экспериментов, поставленных на имитационной модели, которые не противоречат физической интерпретации.

ЛИТЕРАТУРА

1. Башарин Г.П., Кокотушкин В.А., Наумов В.А. О методе эквивалентных замен расчета фрагментов сетей связи // Изв. АН СССР. Техн. кибернетика. – 1979. – № 6. – С 92–99.
2. Neuts M.F. A versatile Markov point process // Journal of Applied Probability. – 1979. – Vol. 16. – P. 764–779.
3. Горцев А.М., Нежелская Л.А. Оценивание параметров синхронного дважды стохастического потока событий методом моментов // Вестник Томского государственного университета. – 2002. – № 1. – С. 24–29.
4. Горцев А.М., Леонова М.А., Нежелская Л.А. Совместная плотность вероятностей длительности интервалов обобщённого асинхронного потока событий при непродлевающемся мёртвом времени // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2012. – № 4 (21). – С. 14–25.
5. Горцев А.М., Калягин А.А., Нежелская Л.А. Совместная плотность вероятностей длительности интервалов обобщенного полусинхронного потока событий при непродлевающемся мёртвом времени // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2014. – № 2 (27). – С. 19–29.
6. Апанасович В.В., Коляда А.А., Чернявский А.Ф. Статистический анализ случайных потоков в физическом эксперименте. – Минск: Университетское, 1988. – 254 с.
7. Нежелская Л.А., Тумашкина Д.А. Имитационная модель полусинхронного потока второго порядка // Труды Томского государственного университета. Серия физико-математическая. Томск: Издательский Дом ТГУ, 2016. – Т. 299. – С. 109–114.
8. Нежелская Л.А., Тумашкина Д.А. Оптимальное оценивание состояний полусинхронного потока событий второго порядка // Труды Томского государственного университета. Серия физико-математическая. – Томск: Издательский Дом ТГУ, 2017. – Т. 301. – С. 97–105.
9. Соболев И.М. Численные методы Монте-Карло. – М.: Наука, 1973. – 311 с.

СТАТИСТИЧЕСКИЕ ЭКСПЕРИМЕНТЫ НА ИМИТАЦИОННОЙ МОДЕЛИ ОБОБЩЁННОГО МАР-ПОТОКА СОБЫТИЙ С ПРОИЗВОЛЬНЫМ ЧИСЛОМ СОСТОЯНИЙ

А.В. Кеба, Л.А. Нежелская
Томский государственный университет
mir.na.mig7@mail.ru, ludne@mail.ru

Введение

В настоящее время теория массового обслуживания (ТМО), формально являясь частью теории случайных процессов и занимаясь системами массового обслуживания (СМО), интенсивно развивается и имеет широкую сферу применения: в экономике и финансах, в планировании и организации процессов производства и обслуживания, в военном деле, медицине, авиации, в транспортной сфере и других отраслях человеческой деятельности [1,2].

В большинстве работ по ТМО до 60-х годов XX столетия входящие потоки событий преимущественно рассматривались как пуассоновские потоки. Но в результате стремительного развития техники, телекоммуникационных, спутниковых и компьютерных сетей связи модель простейшего потока перестала соответствовать информационным потокам сообщений в режиме реального времени. Тем самым, требования практики стали причиной рассмотрения дважды стохастических потоков [3–5] в виде математической модели реальных потоков событий. Двойная случайность (стохастика) подразумевает, во-первых, случайность моментов времени наступления событий потока, во-вторых, представление интенсивности потока как случайного процесса.

В дважды стохастических потоках выделяются два класса: 1) интенсивность потоков есть непрерывный случайный процесс; 2) интенсивность потоков представляет собой кучно-постоянный случайный процесс с конечным числом состояний.

Потоки второго класса в работах [6,7] получили название МС-потоков (Markov Chain). Именно к этому классу относится обобщённый МАР-поток событий с произвольным числом состояний и представляет собой адекватную математическую модель, соответствующую реальным потокам случайных событий.

1. Постановка задачи

Рассматривается обобщённый МАР-поток событий с произвольным числом состояний, сопровождающий процесс $\lambda(t)$ которого является случайным кусочно-постоянным ненаблюдаемым процессом с n состояниями: S_1, S_2, \dots, S_n . Далее под i -м состоянием понимается S_i , $i = \overline{1, n}$.

Длительность пребывания процесса $\lambda(t)$ в i -м состоянии ($i = \overline{1, n}$) определяется случайной величиной, распределённой по экспоненциальному закону, с функцией распределения $F_i(t) = 1 - e^{-\lambda_i t}$, $i = \overline{1, n}$. Если процесс $\lambda(t)$ в момент времени t находится в i -м состоянии, т.е. $\lambda(t) = \lambda_i$, то на полуинтервале $[t; t + \Delta t)$, где Δt – достаточно малая величина, в момент окончания i -го состояния происходит розыгрыш состояния – включается вероятностный механизм:

- 1) наступает событие потока, и процесс $\lambda(t)$ переходит из i -го состояния в j -е состояние ($i, j = \overline{1, n}$), вероятность чего есть $P_1(\lambda_j | \lambda_i)$;
- 2) не наступает событие потока, и процесс $\lambda(t)$ переходит из i -го состояния в j -е состояние ($i, j = \overline{1, n}$), вероятность чего есть $P_0(\lambda_j | \lambda_i)$.

При этом выполняется условие нормировки $\sum_{j=1}^n P_0(\lambda_j | \lambda_i) + \sum_{j=1}^n P_1(\lambda_j | \lambda_i) = 1, i = \overline{1, n}$.

Если $\lambda(t) = \lambda_i (i = \overline{1, n})$, то будем говорить, что имеет место i -е состояние процесса $\lambda(t)$. Будем полагать, что $\lambda_1 > \lambda_2 > \dots > \lambda_n > 0$.

На рис. 1 приведён пример одной из возможных реализаций процесса $\lambda(t)$, где t_1, t_2, \dots – моменты наступления событий в потоке.

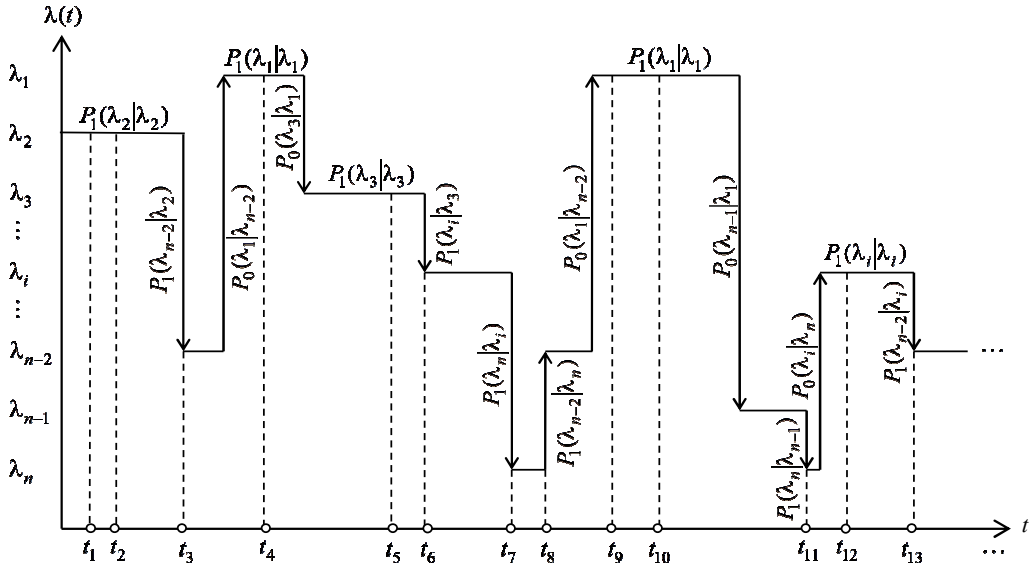


Рис. 1. Реализация обобщённого МАР-потока событий с произвольным числом состояний

Блочная матрица инфинитезимальных характеристик процесса $\lambda(t)$ имеет вид

$D = \|D_0 | D_1\|$, где

$$D_0 = \begin{pmatrix} -\lambda_1 & \lambda_1 P_0(\lambda_2 | \lambda_1) & \dots & \lambda_1 P_0(\lambda_n | \lambda_1) & \lambda_1 P_0(\lambda_1 | \lambda_1) \\ \lambda_2 P_0(\lambda_1 | \lambda_2) & -\lambda_2 & \dots & \lambda_2 P_0(\lambda_n | \lambda_2) & \lambda_2 P_0(\lambda_2 | \lambda_2) \\ \lambda_3 P_0(\lambda_1 | \lambda_3) & \lambda_3 P_0(\lambda_2 | \lambda_3) & \dots & \lambda_3 P_0(\lambda_n | \lambda_3) & \lambda_3 P_0(\lambda_3 | \lambda_3) \\ \dots & \dots & \dots & \dots & \dots \\ \lambda_n P_0(\lambda_1 | \lambda_n) & \lambda_n P_0(\lambda_2 | \lambda_n) & \dots & -\lambda_n & \lambda_n P_0(\lambda_n | \lambda_n) \end{pmatrix}_{n \times (n+1)},$$

$$D_1 = \begin{pmatrix} \lambda_1 P_1(\lambda_1 | \lambda_1) & \lambda_1 P_1(\lambda_2 | \lambda_1) & \dots & \lambda_1 P_1(\lambda_{n-1} | \lambda_1) & \lambda_1 P_1(\lambda_n | \lambda_1) \\ \lambda_2 P_1(\lambda_1 | \lambda_2) & \lambda_2 P_1(\lambda_2 | \lambda_2) & \dots & \lambda_2 P_1(\lambda_{n-1} | \lambda_2) & \lambda_2 P_1(\lambda_n | \lambda_2) \\ \lambda_3 P_1(\lambda_1 | \lambda_3) & \lambda_3 P_1(\lambda_2 | \lambda_3) & \dots & \lambda_3 P_1(\lambda_{n-1} | \lambda_3) & \lambda_3 P_1(\lambda_n | \lambda_3) \\ \dots & \dots & \dots & \dots & \dots \\ \lambda_n P_1(\lambda_1 | \lambda_n) & \lambda_n P_1(\lambda_2 | \lambda_n) & \dots & \lambda_n P_1(\lambda_{n-1} | \lambda_n) & \lambda_n P_1(\lambda_n | \lambda_n) \end{pmatrix}_{n \times n}.$$

Элементы $d_{ii}^{(0)}, i = \overline{1, n}$ прямоугольной матрицы D_0 есть интенсивности выхода процесса $\lambda(t)$ из своих состояний, взятые с противоположным знаком; остальные элементы – интенсивности переходов из состояния i в состояние j без наступления события, $i, j = \overline{1, n}$. Элементами матрицы D_1 являются интенсивности переходов процесса $\lambda(t)$ из состояния i в состояние j с наступлением события, $i, j = \overline{1, n}$.

Подчеркнём, что обобщённый МАР-поток событий с двумя состояниями впервые введён в работе [8]. Отличие обобщённого МАР-потока событий от МАР-потока состоит в том, что вводится вероятность $P_0(\lambda_j|\lambda_i) \neq 0$, $i = \overline{1, n}$, перехода процесса $\lambda(t)$ из i -го состояния в i -е без наступления события потока.

Отметим, что задачи по оценке состояний и параметров дважды стохастических потоков событий с произвольным числом состояний рассматривались в [9] (синхронный поток) и в [10] (асинхронный поток).

Задача заключается в построении имитационной модели обобщённого МАР-потока событий с произвольным числом состояний.

2. Имитационное моделирование

Имитационное моделирование – это метод исследования, при котором строится модель изучаемой системы, описывающая её с достаточной степенью точности, и на базе имитирующей модели проводится серия экспериментов в течение заданного периода времени с целью получения информации об исходной системе.

Сущность метода имитационного моделирования заключается в разработке на ЭВМ программ, имитирующих поведение изучаемой системы, её свойства и характеристики в необходимом объёме. Реализуемый на ЭВМ алгоритм позволяет по заданным значениям параметров и состоянию системы в начальный момент времени получить характеристики системы, необходимые для дальнейшего применения в практических задачах. Проведение серий экспериментов позволяет достаточно полно исследовать исходную систему, выяснить зависимость между требуемыми параметрами.

Модель исследуемого обобщённого МАР-потока событий с произвольным числом состояний построена с помощью метода обратных функций. Пусть $\tau \geq 0$ – длительность пребывания процесса $\lambda(t)$ в i -м состоянии. Случайная величина τ имеет функцию распределения $F_i(t) = P(\tau < t) = 1 - e^{-\lambda_i t}$ и плотность распределения вероятностей $p_i(t) = F_i'(t) = \lambda_i e^{-\lambda_i t}$, $i = \overline{1, n}$. В соответствии с методом обратных функций обозначим $F_i(t) = \gamma$. Имеем $\gamma = 1 - e^{-\lambda_i t}$, тогда $-\lambda_i t = \ln(1 - \gamma)$, откуда находим формулу моделирования значений длительностей интервалов пребывания процесса $\lambda(t)$ в i -м состоянии $t = -\frac{1}{\lambda_i} \ln(1 - \gamma)$, где γ – равномерно распределённая на интервале $(0, 1)$ случайная величина, λ_i – параметр потока, $i = \overline{1, n}$.

3. Результаты статистических экспериментов

Построена имитационная модель обобщённого МАР-потока событий с произвольным числом состояний в виде программного кода на языке программирования C# в среде Visual Studio 2015 и поставлен ряд статистических экспериментов с целью изучения качества построенной модели.

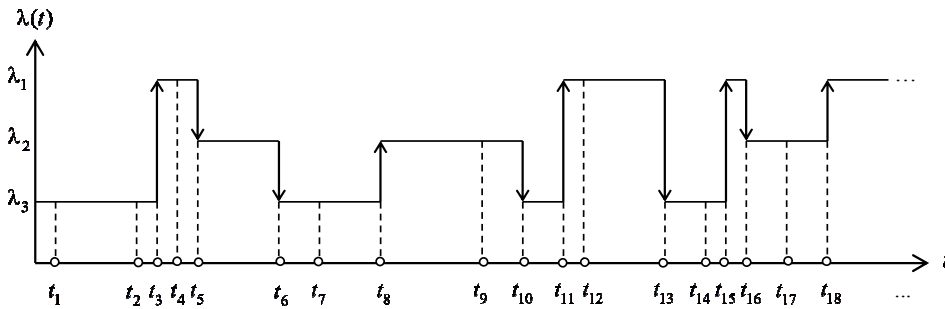
Эксперимент 1. Фиксируются значение времени моделирования $T_m = 100$, число состояний $n = 3$, количество опытов $N = 100$. Рассмотрим результаты моделирования при крайних значениях вероятностей перехода, когда:

$$a) \sum_{j=1}^3 P_0(\lambda_j|\lambda_i) = 0, \sum_{j=1}^3 P_1(\lambda_j|\lambda_i) = 1, i = \overline{1, 3}.$$

Исходные данные для испытания а) имитационной модели

$\lambda_1 = 3$	$\lambda_2 = 2$	$\lambda_3 = 1$
$P_0(\lambda_1 \lambda_1) = 0$	$P_0(\lambda_1 \lambda_2) = 0$	$P_0(\lambda_1 \lambda_3) = 0$
$P_0(\lambda_2 \lambda_1) = 0$	$P_0(\lambda_2 \lambda_2) = 0$	$P_0(\lambda_2 \lambda_3) = 0$
$P_0(\lambda_3 \lambda_1) = 0$	$P_0(\lambda_3 \lambda_2) = 0$	$P_0(\lambda_3 \lambda_3) = 0$
$P_1(\lambda_1 \lambda_1) = 0,4$	$P_1(\lambda_1 \lambda_2) = 0,3$	$P_1(\lambda_1 \lambda_3) = 0,3$
$P_1(\lambda_2 \lambda_1) = 0,3$	$P_1(\lambda_2 \lambda_2) = 0,4$	$P_1(\lambda_2 \lambda_3) = 0,3$
$P_1(\lambda_3 \lambda_1) = 0,3$	$P_1(\lambda_3 \lambda_2) = 0,3$	$P_1(\lambda_3 \lambda_3) = 0,4$

В результате имитации потока при исходных данных, соответствующих табл. 1, процесс $\lambda(t)$ переходит из состояния i в состояние j только с наступлением события потока, $i, j = \overline{1,3}$. Реализация ситуации проиллюстрирована на рис. 2.

Рис. 2. Реализация обобщенного МАР-потока событий при $n = 3$

Рассмотрим результаты моделирования при других крайних значениях вероятностей перехода, когда события потока отсутствуют.

$$\text{б) } \sum_{j=1}^3 P_0(\lambda_j|\lambda_i) = 1, \quad \sum_{j=1}^3 P_1(\lambda_j|\lambda_i) = 0, \quad i = \overline{1,3}.$$

Таблица 2

Исходные данные для испытания б) имитационной модели

$\lambda_1 = 3$	$\lambda_2 = 2$	$\lambda_3 = 1$
$P_0(\lambda_1 \lambda_1) = 0,4$	$P_0(\lambda_1 \lambda_2) = 0,3$	$P_0(\lambda_1 \lambda_3) = 0,3$
$P_0(\lambda_2 \lambda_1) = 0,3$	$P_0(\lambda_2 \lambda_2) = 0,4$	$P_0(\lambda_2 \lambda_3) = 0,3$
$P_0(\lambda_3 \lambda_1) = 0,3$	$P_0(\lambda_3 \lambda_2) = 0,3$	$P_0(\lambda_3 \lambda_3) = 0,4$
$P_1(\lambda_1 \lambda_1) = 0$	$P_1(\lambda_1 \lambda_2) = 0$	$P_1(\lambda_1 \lambda_3) = 0$
$P_1(\lambda_2 \lambda_1) = 0$	$P_1(\lambda_2 \lambda_2) = 0$	$P_1(\lambda_2 \lambda_3) = 0$
$P_1(\lambda_3 \lambda_1) = 0$	$P_1(\lambda_3 \lambda_2) = 0$	$P_1(\lambda_3 \lambda_3) = 0$

При имитации потока с исходными данными, соответствующими табл. 2, процесс $\lambda(t)$ переходит из состояния i в состояние j без наступления события потока, $i, j = \overline{1,3}$. Реализация ситуации представлена на рис. 3.

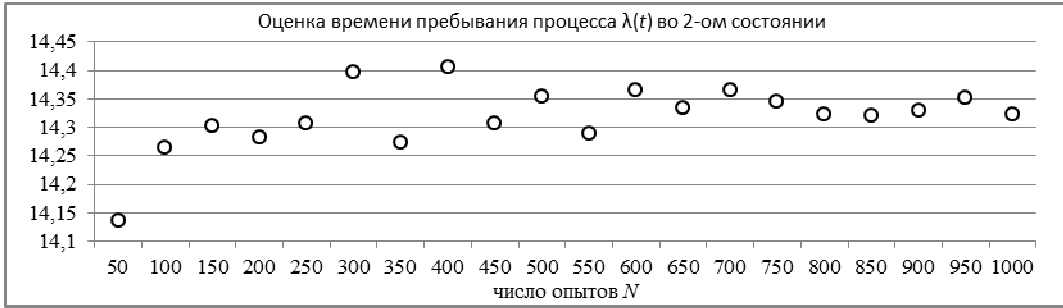


Рис. 6

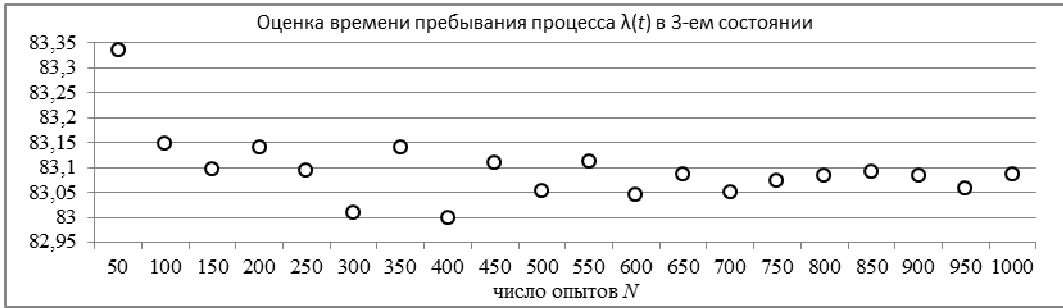


Рис. 7

Из анализа представленных результатов на рис. 5–7 можно заключить, что стационарный режим устанавливается при количестве опытов $N \geq 600$.

Эксперимент 3. Фиксируются значение числа состояний $n = 3$, время моделирования $T_m = 100$, число опытов $N \in \{50, 100, 150, \dots, 1000\}$; вероятности перехода приведены в табл. 4.

Таблица 4

Исходные данные для эксперимента 3

$\lambda_1 = 3$	$\lambda_2 = 2$	$\lambda_3 = 1$
$P_0(\lambda_1 \lambda_1) = 0,12$	$P_0(\lambda_1 \lambda_2) = 0,14$	$P_0(\lambda_1 \lambda_3) = 0,23$
$P_0(\lambda_2 \lambda_1) = 0,25$	$P_0(\lambda_2 \lambda_2) = 0,23$	$P_0(\lambda_2 \lambda_3) = 0,11$
$P_0(\lambda_3 \lambda_1) = 0,15$	$P_0(\lambda_3 \lambda_2) = 0,14$	$P_0(\lambda_3 \lambda_3) = 0,09$
$P_1(\lambda_1 \lambda_1) = 0,18$	$P_1(\lambda_1 \lambda_2) = 0,06$	$P_1(\lambda_1 \lambda_3) = 0,26$
$P_1(\lambda_2 \lambda_1) = 0,19$	$P_1(\lambda_2 \lambda_2) = 0,25$	$P_1(\lambda_2 \lambda_3) = 0,17$
$P_1(\lambda_3 \lambda_1) = 0,11$	$P_1(\lambda_3 \lambda_2) = 0,18$	$P_1(\lambda_3 \lambda_3) = 0,14$

При заданных параметрах эксперимента на рис. 8–10 показана зависимость выборочного среднего значения длительности пребывания процесса $\lambda(t)$ в i -м состоянии. Обозначения в табл. 5 аналогичны обозначениям табл. 3.

Таблица 5

Результаты эксперимента 3

N	оценка времени пребывания (\hat{T}_i)			оценка числа событий (\hat{K})
	в 1-м состоянии (\hat{T}_1)	во 2-м состоянии (\hat{T}_2)	в 3-м состоянии (\hat{T}_3)	
$N = 50$	6,7000	24,5694	68,7305	162

$N = 100$	6,8295	24,5549	68,6154	162
$N = 150$	6,7933	24,4009	68,8056	161
$N = 200$	6,7327	24,4420	68,8252	161
...
$N = 900$	6,7626	24,3676	68,8697	161
$N = 950$	6,7333	24,4712	68,7954	161
$N = 1000$	6,7407	24,472	68,7872	161

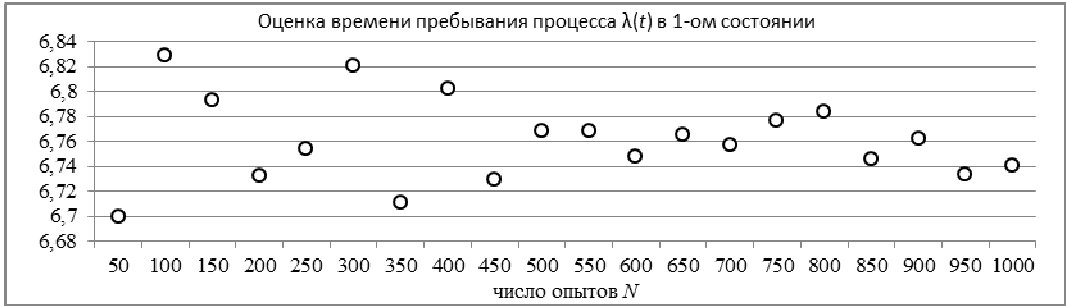


Рис. 8

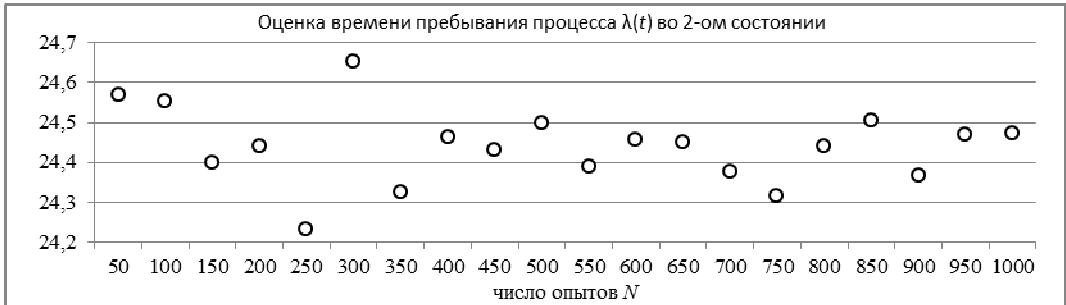


Рис. 9



Рис. 10

На основании полученных результатов на рис. 8–10 можно заключить, что стационарный режим устанавливается при количестве опытов $N \geq 450$.

Заключение

В данной статье рассмотрен обобщённый МАР-поток событий с произвольным числом состояний и получена матрица инфинитезимальных характеристик процесса $\lambda(t)$. Построена имитационная модель исследуемого потока в виде программного кода на языке программирования C# в среде Visual Studio 2015. Анализ результатов прове-

дённных статистических экспериментов позволяет утверждать, что реализована работоспособная модель, непротиворечащая соответствующим входным данным.

ЛИТЕРАТУРА

1. Вишневский В.М. Теоретические основы проектирования компьютерных сетей. – М.: Техносфера, 2003. – 512 с.
2. Фархадов М.П., Петухова Н.В., Ефросинин Д.В., Семёнова О.В. Двухфазная модель с неограниченными очередями для расчёта характеристик и оптимизации речевых порталов самообслуживания // Проблемы управления. – 2010. – № 6. – С. 53–57.
3. Горцев А.М., Нежелская Л.А. Оценивание параметров полусинхронного дважды стохастического потока событий методом моментов // Вестник Томского государственного университета. – 2002. – № 1 (1). – С. 18–23.
4. Горцев А.М., Леонова М.А., Нежелская Л.А. Сравнение МП- и ММ-оценок длительности мёртвого времени в обобщённом асинхронном потоке событий // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2013. – № 4 (25). – С. 32–42.
5. Горцев А.М., Калягин А.А., Нежелская Л.А. Совместная плотность вероятностей длительности интервалов обобщённого полусинхронного потока событий при непродлеваемом мёртвом времени // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2014. – № 2 (27). – С. 19–29.
6. Башарин Г.П., Кокотушкин В.А., Наумов В.А. О методе эквивалентных замен расчёта фрагментов сетей связи. Ч. 1 // Изв. АН СССР. Техн. кибернетика. – 1979. – № 6. – С. 92–99.
7. Башарин Г.П., Кокотушкин В.А., Наумов В.А. О методе эквивалентных замен расчёта фрагментов сетей связи. Ч. 2 // Изв. АН СССР. Техн. кибернетика. – 1980. – № 1. – С. 55–61.
8. Березин Д.В., Нежелская Л.А. Оптимальное оценивание состояний обобщённого МАР-потока событий в условиях непродлеваемого мёртвого времени // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2017. – № 41. – С. 12–23.
9. Бушланов И.В., Горцев А.М., Нежелская Л.А. Оценка параметров синхронного дважды стохастического потока событий // Автоматика и телемеханика. – 2008. – № 9. – С. 76–93.
10. Горцев А.М., Зуевич В.Л. Оптимальная оценка состояний асинхронного дважды стохастического потока событий с произвольным числом состояний // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2010. – № 2. – С. 44–65.

СТАТИСТИЧЕСКИЕ ЭКСПЕРИМЕНТЫ НА ИМИТАЦИОННОЙ МОДЕЛИ МАР-ПОТОКА СОБЫТИЙ С ДВУМЯ СОСТОЯНИЯМИ В УСЛОВИЯХ ПРОДЛЕВАЮЩЕГОСЯ МЁРТВОГО ВРЕМЕНИ

Б.О. Лопсанчап, Л.А. Нежелская

Томский государственный университет

buyan050@mail.ru, ludne@mail.ru

Введение

С появлением новых информационных технологий возникают новые задачи, как в теории массового обслуживания (ТМО), так и в её приложениях, касающихся различных сфер деятельности: связи и телекоммуникаций, экономики и финансов, военного дела и гражданской авиации, логистико-транспортного дела и многих других отраслей, в которых находят применение системы массового обслуживания (СМО).

Особенно широко СМО применялись при решении задач, возникающих при интеграции телефонных систем и станций: уменьшение загруженности операторов телефонных станций, оптимизация эффективности работы и минимизация потери информации [1].

Преимущественно до 70–80-х годов 20-го столетия в большинстве моделей СМО входящим потоком являлся пуассоновский или простейший поток событий. В дальнейшем, с усложнением телекоммуникационных сетей связи, модели с входящим простейшим потоком событий перестали адекватно описывать реальные системы, т.к. при моделировании компьютерных сетей возникла необходимость учитывать тот факт, что интенсивность наступления событий меняется во времени. Это обстоятельство привело к появлению новых математических моделей потоков событий, которые относятся к классу дважды стохастических потоков [2–5]. Стохастика (двойная случайность) пони-

мается в том смысле, что моменты времени наступления событий потока есть случайные величины, а интенсивность потока – случайный процесс.

В дважды стохастических потоках событий рассматриваются два класса: 1) интенсивность потоков есть непрерывный случайный процесс [6]; 2) интенсивность потоков представляет собой кусочно-постоянный случайный процесс с конечным числом состояний [7].

В настоящей статье рассматривается МАР-поток событий с двумя состояниями, относящийся ко второму классу дважды стохастических потоков. Его функционирование происходит в условиях продлевающегося мёртвого времени. МАР-потоки событий представляют собой адекватную математическую модель, соответствующую реальным информационным потокам в телекоммуникационных системах.

1. Математическая модель потока

Рассматривается МАР-поток событий с двумя состояниями в условиях продлевающегося мёртвого времени, сопровождающий случайный процесс которого $\lambda(t)$ является кусочно-постоянным ненаблюдаемым процессом с двумя состояниями: S_1 и S_2 . Далее под i -м состоянием понимается состояние S_i , $i = 1, 2$.

Длительность пребывания процесса $\lambda(t)$ в i -м состоянии ($i = 1, 2$) есть случайная величина, имеющая экспоненциальный закон распределения с функцией $F_i(t) = 1 - e^{-\lambda_i t}$, $i = 1, 2$. Если процесс $\lambda(t)$ в момент времени t находится в i -м состоянии ($\lambda(t) = \lambda_i$), то на полуинтервале $[t; t + \Delta t)$, где Δt – достаточно малая величина, в момент окончания i -го состояния происходит розыгрыш пар, при котором возможны 3 ситуации:

- 1) не наступает событие потока и процесс $\lambda(t)$ переходит из i -го состояния в j -е состояние ($i, j = 1, 2, i \neq j$), вероятность чего есть $P_0(\lambda_j | \lambda_i)$;
- 2) наступает событие потока и процесс $\lambda(t)$ остается в i -м состоянии ($i = 1, 2$), вероятность чего есть $P_1(\lambda_i | \lambda_i)$;
- 3) наступает событие потока и процесс $\lambda(t)$ переходит из i -го состояния в j -е состояние ($i, j = 1, 2, i \neq j$), вероятность чего есть $P_1(\lambda_j | \lambda_i)$.

При этом выполняется условие:

$$P_0(\lambda_j | \lambda_i) + P_1(\lambda_i | \lambda_i) + P_1(\lambda_j | \lambda_i) = 1, \quad i, j = 1, 2, \quad i \neq j.$$

Если $\lambda(t) = \lambda_i$ ($i = 1, 2$), то будем говорить, что имеет место i -е состояние процесса $\lambda(t)$. Полагается, что $\lambda_1 > \lambda_2 > 0$.

Блочная матрица инфинитезимальных характеристик процесса $\lambda(t)$ имеет вид $D = \| \| D_0 \mid D_1 \| \|$, где

$$D_0 = \left\| \begin{array}{cc} -\lambda_1 & \lambda_1 P_0(\lambda_2 | \lambda_1) \\ \lambda_2 P_0(\lambda_1 | \lambda_2) & -\lambda_2 \end{array} \right\|, \quad D_1 = \left\| \begin{array}{cc} \lambda_1 P_1(\lambda_1 | \lambda_1) & \lambda_1 P_1(\lambda_2 | \lambda_1) \\ \lambda_2 P_1(\lambda_1 | \lambda_2) & \lambda_2 P_1(\lambda_2 | \lambda_2) \end{array} \right\|.$$

Элементы на главной диагонали матрицы D_0 представляют собой интенсивности выхода процесса $\lambda(t)$ из своих состояний, взятые с противоположным знаком; элементы на побочной диагонали есть интенсивности переходов из состояния i в состояние j без наступления события, $i, j = 1, 2, i \neq j$. Матрица D_1 содержит интенсивности переходов процесса $\lambda(t)$ из состояния i в состояние j с наступлением событий, $i, j = 1, 2, i \neq j$.

После каждого зарегистрированного события МАР-потока наступает период ненаблюдаемости событий фиксированной длительности T (мёртвое время), которой создается этим событием, так что другие события МАР-потока, наступившие в течение времени T , недоступны наблюдению. Вместе с тем, хотя события и не наблюдаются в течение мёртвого времени, каждое из них вызывает продление периода ненаблюдаемости на ту же величину T , так что наблюдаться будет лишь то событие, которое наступило после окончания последнего периода мёртвого времени. По окончании общего периода ненаблюдаемости первое наступившее событие снова создает период мёртвого времени длительности T и т.д. Таким образом, общий период ненаблюдаемости есть случайная величина.

На рис. 1. приведён пример одной из возможных реализаций процесса $\lambda(t)$ и наблюдаемого потока, где t_1, t_2, \dots – моменты наступления событий в наблюдаемом потоке; черными кружками обозначены потерянные события исходного потока; штриховкой – длительности мёртвого времени; 1 и 2 – состояния случайного процесса.

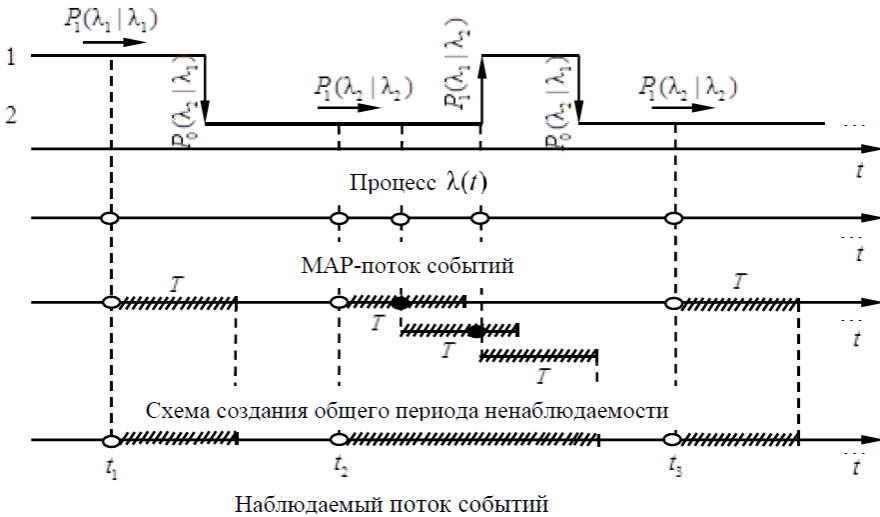


Рис. 1. Формирование наблюдаемого потока событий

Подчеркнём, что дважды стохастические потоки событий с двумя состояниями в условиях продлевающегося мёртвого времени рассмотрены в статьях [8–10]: в [8] – полусинхронный поток, в [9] – синхронный поток, в [10] – альтернирующий асинхронный поток событий.

Задача состоит в построении имитационной модели МАР-потока событий с двумя состояниями в условиях продлевающегося мёртвого времени и проверке работоспособности модели путём проведения ряда статистических экспериментов.

2. Имитационное моделирование

Одним из эффективных методов исследования, при котором строится модель рассматриваемой системы, является имитационное моделирование. Имитационная модель описывает с достаточной степенью точности изучаемую систему и на данной основе проводятся опыты в течение задаваемого периода времени моделирования с целью получения информации об исходной системе.

Смысловая нагрузка метода имитационного моделирования заключается в разработке на базе ЭВМ программ, реализующих имитацию различных поведений изучаемой системы, её свойств и характеристик в необходимом объёме. Программы позволяют по заданным значениям параметров и состоянию системы в исходный момент вре-

мени получить характеристики системы, необходимые для дальнейшего применения в решении практических задач. Проведение статистических опытов позволяет исследовать исходную систему и выяснить зависимость получаемых статистических характеристик от исходных параметров.

Имитационная модель исследуемого МАР-потока событий с двумя состояниями в условиях продлевающегося мёртвого времени построена с помощью метода обратных функций [11]. Пусть $\tau \geq 0$ – длительность пребывания процесса $\lambda(t)$ в i -м состоянии. Случайная величина τ имеет функцию распределения $F_i(t) = P(\tau < t) = 1 - e^{-\lambda_i t}$, $i = 1, 2$. В соответствии с методом обратных функций обозначим $F_i(t) = \gamma$. Имеем $\gamma = 1 - e^{-\lambda_i t}$, тогда $-\lambda_i t = \ln(1 - \gamma)$, откуда находим формулу моделирования значений длительностей интервалов пребывания процесса $\lambda(t)$ в i -м состоянии $t = -\frac{1}{\lambda_i} \ln(1 - \gamma)$, где γ – равномерно распределенная на интервале $(0,1)$ случайная величина, λ_i – параметр потока, $i = 1, 2$.

В результате построена имитационная модель МАР-потока событий с двумя состояниями в условиях продлевающегося мёртвого времени в виде программного кода на языке программирования C# в среде Visual Studio 2013 и поставлен ряд статистических экспериментов с целью установления работоспособности имитационной модели.

3. Результаты статистических экспериментов

Цель экспериментов. Оценить суммарный общий период ненаблюдаемости и длительности пребывания процесса $\lambda(t)$ в i -м состоянии, $i = 1, 2$, при фиксируемом значении времени моделирования $T_m = 1000$ и значении мёртвого времени $T = 0,1$.

Эксперимент 1. Рассматривается крайний случай, когда значения вероятностей переходов подобраны таким образом, что процесс переходит из состояния i в состояние j только с наступлением событий, $i, j = 1, 2$, $i \neq j$. Значения исходных данных для имитационной модели (значения вероятностей перехода и интенсивностей) представлены в табл. 1.

Таблица 1

Исходные данные имитационной модели

$\lambda_1 = 10$	$\lambda_2 = 5$
$P_0(\lambda_2 \lambda_1) = 0$	$P_0(\lambda_1 \lambda_2) = 0$
$P_1(\lambda_1 \lambda_1) = 0,5$	$P_1(\lambda_2 \lambda_2) = 0,5$
$P_1(\lambda_2 \lambda_1) = 0,5$	$P_1(\lambda_1 \lambda_2) = 0,5$

Итогом имитации потока при исходных данных, приведенных в табл. 1, является реализация ситуации, проиллюстрированной на рис. 2.

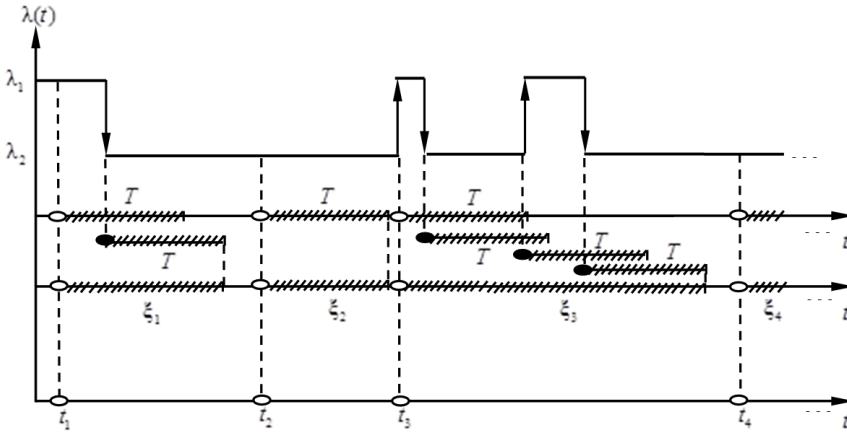


Рис. 2. Реализация MAP-потока событий в условиях продлевающегося мёртвого времени

В табл. 2 приведены численные результаты эксперимента 1, где N – число опытов (реализаций потока), $\hat{\xi}$ – оценка значения суммарного общего периода ненаблюдаемости потока, \hat{T}_i – оценка значения длительности пребывания процесса $\lambda(t)$ в i -м состоянии, $i = 1, 2$. Данные статистические характеристики рассчитываются по формулам:

$$\hat{\xi} = \frac{1}{N} \sum_{j=1}^N \hat{\xi}^{(j)}, \quad \hat{\xi}^{(j)} = \sum_{k=1}^l \xi_k^{(j)}, \quad (1)$$

где $\xi_k^{(j)}$ – k -й общий период ненаблюдаемости в j -й реализации; l – количество общих периодов ненаблюдаемости в j -й реализации,

$$\hat{T}_i = \frac{1}{N} \sum_{j=1}^N \hat{T}_i^{(j)}, \quad \hat{T}_i^{(j)} = \sum_{k=1}^{m_i} T_{ik}^{(j)}, \quad (2)$$

где $T_{ik}^{(j)}$ – k -е значение длительности пребывания процесса $\lambda(t)$ в i -м состоянии в j -й реализации; m_i – количество временных участков, когда имеет место состояние S_i процесса $\lambda(t)$, $i = 1, 2$, в j -й реализации.

Таблица 2

Результаты эксперимента 1

N	Оценка суммарного общего периода ненаблюдаемости и длительности пребывания процесса $\lambda(t)$ в i -м состоянии, $i = 1, 2$		
	$\hat{\xi}$	\hat{T}_1	\hat{T}_2
50	473,01	334,01	665,94
100	473,32	333,50	666,42
150	473,11	333,73	666,21
200	472,91	333,81	667,10
...
900	472,97	333,56	666,43
950	473,14	333,45	666,54
1000	473,09	333,22	666,76

Графики, представленные на рис. 3–5 и построенные на основе данных табл. 2, демонстрируют сходимость оценок $\hat{\xi}$, \hat{T}_1 , \hat{T}_2 к некоторым постоянным величинам.

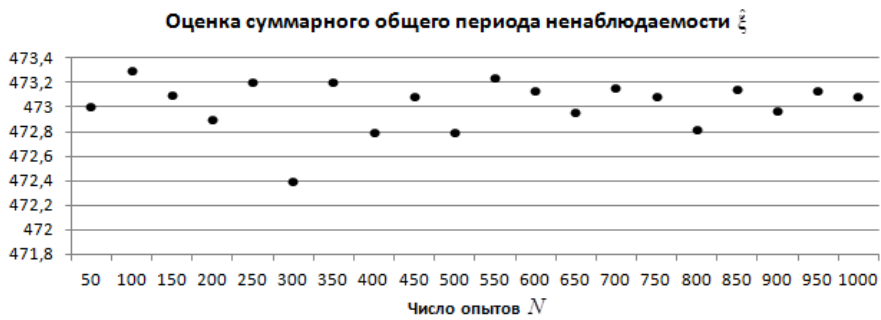


Рис. 3. График зависимости $\hat{\xi}$ от значения N

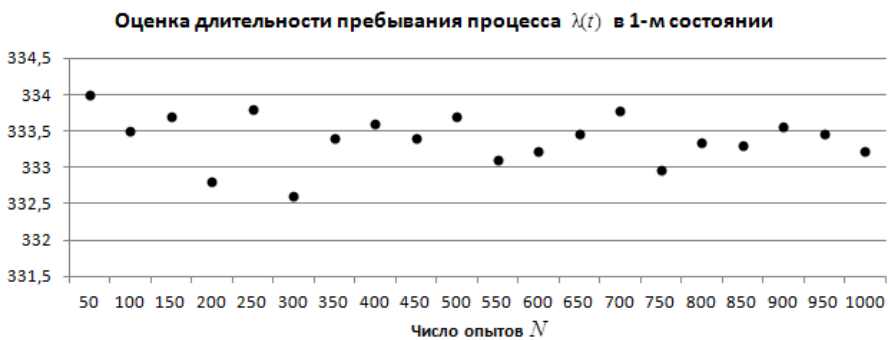


Рис. 4. График зависимости \hat{T}_1 от значения N

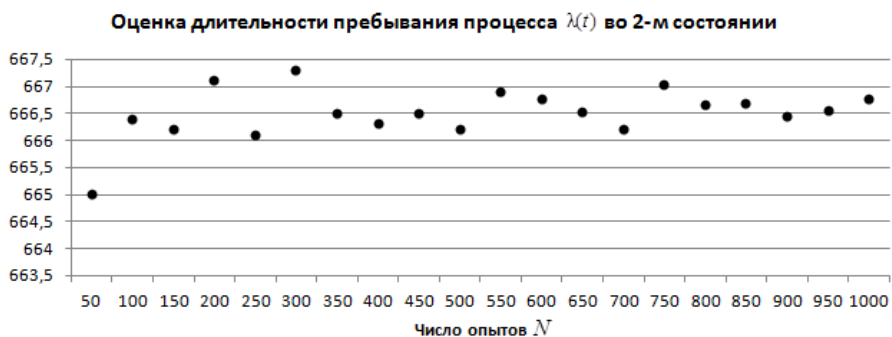


Рис. 5. График зависимости \hat{T}_2 от значения N

Анализируя результаты, приведенные на рис. 3–5, можно заключить, что стационарный режим устанавливается при количестве опытов $N \geq 800$.

Эксперимент 2. Рассматривается случай, когда значения вероятностей переходов выбраны произвольно, т.е. процесс $\lambda(t)$ может переходить из состояния i в состояние j и с наступлением событий, и без наступления событий, $i, j = 1, 2, i \neq j$. Значения исходных данных для имитационной модели (значения вероятностей перехода и интенсивностей) представлены в табл. 3.

Исходные данные имитационной модели

$\lambda_1 = 5$	$\lambda_2 = 1$
$P_0(\lambda_2 \lambda_1) = 0,1$	$P_0(\lambda_1 \lambda_2) = 0,3$
$P_1(\lambda_1 \lambda_1) = 0,6$	$P_1(\lambda_2 \lambda_2) = 0,2$
$P_1(\lambda_2 \lambda_1) = 0,3$	$P_1(\lambda_1 \lambda_2) = 0,5$

Итогом имитации потока при исходных данных, приведенных в табл. 3, является реализация ситуации, проиллюстрированной на рис. 6.

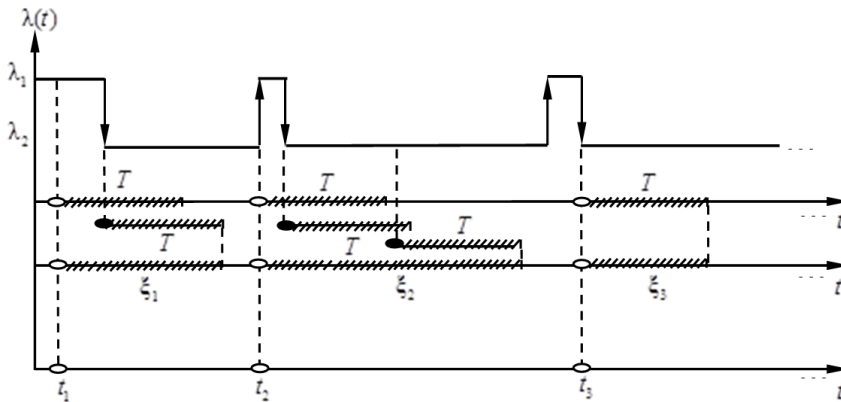


Рис. 6. Реализация MAP-потока событий в условиях продлевающегося мёртвого времени

В табл. 4 представлены численные результаты эксперимента 2 по вычислению величин $\hat{\xi}$, \hat{T}_1 , \hat{T}_2 , определённых формулами (1), (2).

Таблица 4

Результаты эксперимента 2

N	Оценка суммарного общего периода ненаблюдаемости и длительности пребывания процесса $\lambda(t)$ в i -м состоянии, $i = 1, 2$		
	$\hat{\xi}$	\hat{T}_1	\hat{T}_2
50	153,81	286,41	713,50
100	152,52	284,72	715,21
150	153,16	283,81	713,16
200	152,90	284,53	715,42
...
900	153,45	285,95	714,04
950	153,38	285,98	714,01
1000	153,05	285,45	714,54

Графики, представленные на рис. 7–9 и построенные на основе данных табл. 4, демонстрируют сходимость оценок к некоторым постоянным величинам.

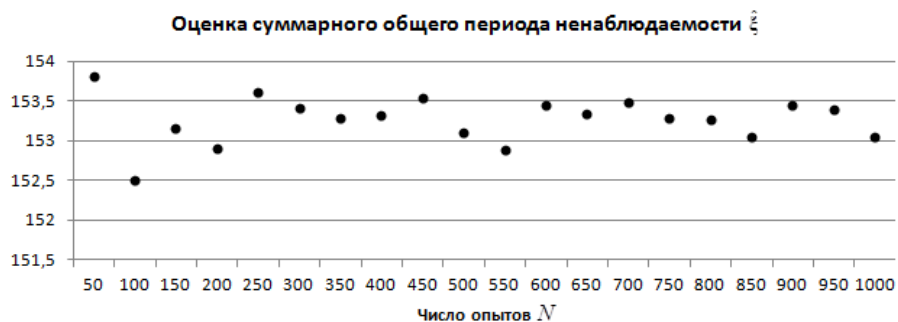


Рис. 7. График зависимости $\hat{\xi}$ от значения N

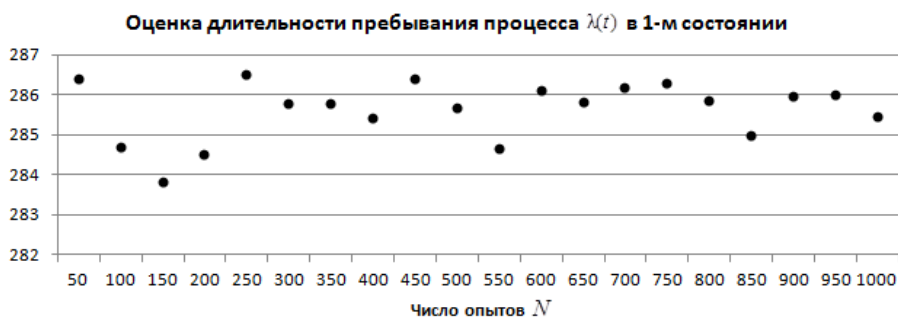


Рис. 8. График зависимости \hat{T}_1 от значения N

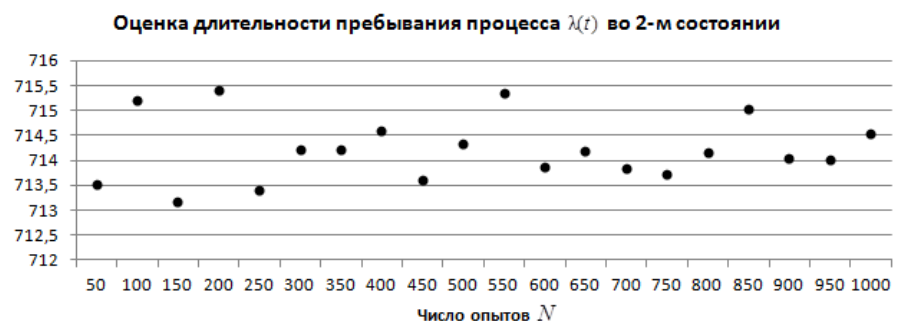


Рис. 9. График зависимости \hat{T}_2 от значения N

Анализируя результаты, представленные на рис. 7–9, можно заключить, что стационарный режим устанавливается при количестве опытов $N \geq 650$.

Сравнивая между собой результаты двух поставленных экспериментов, замечаем, что суммарный общий период ненаблюдаемости в эксперименте 1 больше, чем в эксперименте 2, что является естественным, т.к. МАР-поток в первом эксперименте более интенсивен, чем во втором. Последнее подтверждает работоспособность имитационной модели.

Заключение

В данной статье рассмотрен МАР-поток событий с двумя состояниями в условиях продлевающегося мёртвого времени. Построена имитационная модель исследуемого потока в виде программного кода на языке программирования C# в среде Visual Studio

2013. Анализ результатов проведённых статистических экспериментов позволяет утверждать, что реализована работоспособная модель, непротиворечащая соответствующим исходным данным.

ЛИТЕРАТУРА

1. Erlang A.K. The theory of probabilities and telephone conversations // *Nyt Jidsskrift for Matematik. Seria B.* – 1909. – Vol. 20. – Is. B. – P. 33–39.
2. Башарин Г.П., Кокотушкин В.А., Наумов В.А. О методе эквивалентных замен расчета фрагментов сетей связи. Ч. 1 // *Изв. АН СССР. Техн. кибернетика.* – 1979. – № 6. – С. 92–99.
3. Neuts M.F. A versatile Markov point process // *Journal of Applied Probability.* – 1979. – Vol. 16. – P. 764–779.
4. Gortsev A.M., Nezhel'skaya L.A., Shevchenko T.I. Estimation of the states of an MC-stream of events in the presence of measurement errors // *Russian Physics Journal.* – 1993. – Vol. 36. – Is. 12. – P. 1153–1167.
5. Горцев А.М., Леонова М.А., Нежелская Л.А. Сравнение МП- и ММ-оценок длительности мёртвого времени в обобщенном асинхронном потоке событий // *Вестник Томского государственного университета. Управление, вычислительная техника и информатика.* – 2013. – № 4 (25). – С. 32–42.
6. Cox D.R. The analysis of non-Markovian stochastic processes // *Proc. Cambr. Phil. Soc.* – 1955. – Vol. 51. – № 3. – P. 433–441.
7. Lucantoni D.M. New results on single server with a bath Markovian arrival process // *Stochastic Models.* – 1991. – Vol. 7. – P. 1–46.
8. Горцев А.М., Нежелская Л.А. Полусинхронный дважды стохастический поток событий при продлеваемом мёртвом времени // *Вычислительные технологии.* – 2008. – Т. 13. – №1. – С. 31–41.
9. Бушланов И.В., Горцев А.М., Нежелская Л.А. Оценка параметров синхронного дважды стохастического потока событий // *Автоматика и телемеханика.* – 2008. – № 9. – С. 76–93.
10. Горцев А.М., Паршина М.Е. Оценивание параметров альтернирующего потока событий в условиях «мёртвого времени» // *Известия высших учебных заведений. Физика.* – 1999. – № 4. – С. 8–13.
11. Соболев И.М. Численные методы Монте-Карло. – М.: Наука. Гл. ред. физ.-мат. лит. – 1973. – 331 с.

СТОХАСТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ТЕРРИТОРИИ ПАРКОВКИ АВТОМОБИЛЕЙ

Е.Д. Петров, А.Н. Моисеев

Томский государственный университет

petrov.evgeny.d@gmail.com, moiseev.tsu@gmail.com

Введение

За последние десятилетия количество автомобилей в городах существенно увеличилось, при этом их транспортные системы не успевают расширяться с такой же скоростью. Возникает не только проблема с пробками, но и проблема с паркингом.

Потребность в парковочных местах определяется в зависимости от характеристики объекта, рядом с которым будет находиться парковка (количества одновременных посещений, количества сотрудников, торговой площади и т.д.). Как следствие, неправильно организованная парковка в «час пик» может привести к возникновению заторов или к потере посетителей, которые не смогли найти место для стоянки.

Для решения данной проблемы следует проектировать парковку исходя из характеристик объекта и интенсивности движения на прилегающих дорогах. Расчёт характеристик функционирования позволил бы заранее определить наиболее подходящий размер территории парковки, количество въездов и выездов, их наилучшее расположение. В этом могут помочь методы математического моделирования и теории транспортных потоков. Корректно построенная модель позволит многократно моделировать состояние парковки при различных значениях параметров.

В настоящей работе выполнены разработка и реализация стохастической модели парковочной территории. Модель представлена в виде многофазной системы массового обслуживания с входящим нестационарным пуассоновским потоком [1,2], а сам процесс моделирования выполняется имитационно с применением разработанного приложения в среде Salabim [3].

1. Постановка задачи

Парковка предназначена для стоянки автомобилей, которые могут въехать на неё с некоего участка транспортной сети. Типичная схема парковки представлена на рис. 1.

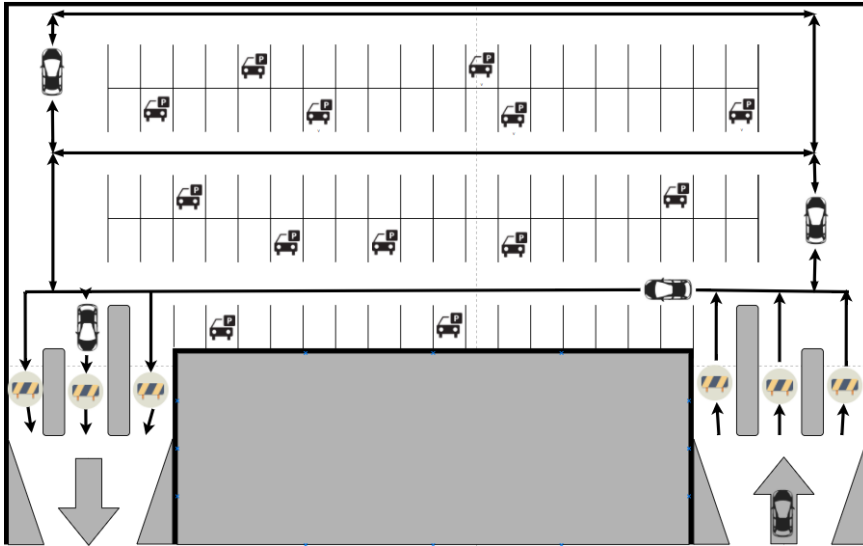


Рис. 1. Схема парковки

Въезд на парковку начинается с прилегающей дороги. При большой интенсивности транспортного потока очередь на въезд может начинаться непосредственно с данного участка транспортной сети. Здесь водители могут отказаться от въезда на парковку и покинуть очередь. Далее дорога разделяется на несколько въездов. Для каждого въезда выделяется своя полоса движения. Въезды могут быть оборудованы шлагбаумами или иными искусственными препятствиями, предназначенными для регулирования транспортного потока. На данном участке покинуть очередь уже нельзя.

После въезда автомобиль попадает на самую территорию парковки и переходит к следующему этапу – стоянке. Места для стоянки занимают большую часть территории парковки, здесь также имеются полосы для движения автомобилей. Количество стояночных мест на парковке всегда ограничено. Если спустя какое-то время автомобиль не смог найти свободное парковочное место, он покидает парковку.

Для выезда автомобилей также выделяется несколько полос движения. Обычно количество въездов и выездов одинаково, но в некоторых случаях их количество может различаться. Как и въезд, выезд может быть оборудован шлагбаумами или иными искусственными препятствиями. После выезда автомобиль выезжает на прилегающий участок транспортной сети.

Цель настоящей работы – разработать стохастическую модель транспортного потока на парковке, реализовать и выполнить имитационное моделирование на основе разработанной модели.

2. Описание разработанной модели

Разрабатываемую стохастическую модель парковки предлагается представить в виде многофазной системы массового обслуживания, в которой каждой фазе соответствует своя роль – въезд, стоянка и выезд.

Для моделирования транспортного потока будем использовать нестационарный пуассоновский поток. Данный поток позволяет реализовать основное свойство транспортного потока при въезде на парковку – количество автомобилей, въезжающих на парковку, изменяется с течением времени. Исходными данными для модели является

информация о количестве автомобилей, которые въезжают на парковку в различное время суток. В настоящей работе для моделирования входящего потока будем использовать нестационарный пуассоновский поток, заданный интенсивностью, определяемой из почасового графика [4] (рис. 2).

Первой фазой, на которую поступает поток заявок, является въезд на парковку. Данная фаза представлена в виде системы массового обслуживания $G/G/n/\infty$. Для данной системы массового обслуживания используется комбинированная очередь, состоящая из двух частей. Первая часть – полнодоступная очередь, с нетерпеливыми клиентами, куда попадают все заявки без исключения. Размер данной очереди не ограничен. Нетерпеливость клиентов определяется либо параметром, задающим интенсивность оставления очереди, либо функцией распределения вероятностей времени пребывания в очереди до момента ее оставления. Вторая часть комбинированной очереди на въезд – это n неполнодоступных очередей, где n – количество обслуживающих приборов (въездов), каждая очередь закреплена за конкретным обслуживающим прибором. Клиенты в очереди – терпеливые и не покидают её. Размер очередей задаётся при помощи отдельного параметра. Въезд осуществляется посредством обслуживания на приборах. Время обслуживания определяется некоторой заданной функцией распределения. После обслуживания все заявки поступают на вход фазы 2.

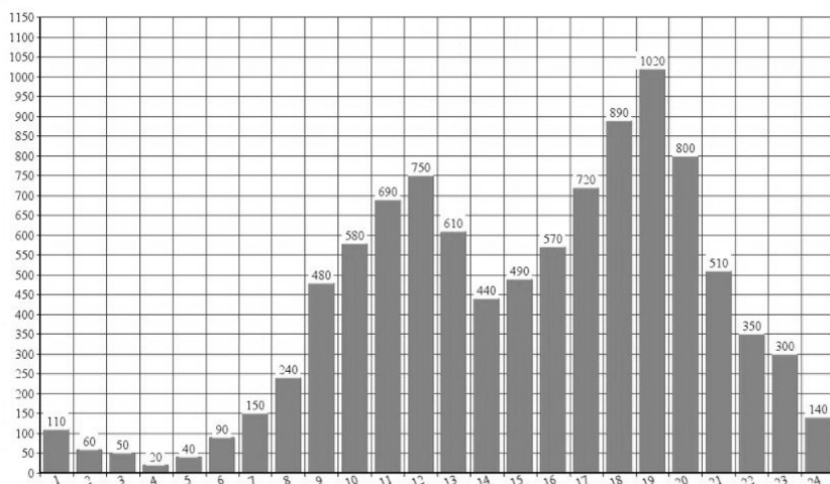


Рис. 2. Изменение интенсивности автомобилей, въезжающих на парковку, в течение дня

Основным этапом описываемой модели является фаза стоянки (фаза 2). Данная фаза сама представлена как двухфазная система массового обслуживания $\cdot/G/\infty \rightarrow \cdot/G/n/0$. Здесь предусмотрено две фазы обслуживания – поиск свободного места и непосредственно стоянка. Изначально заявка, поступившая в эту систему, попадает для обслуживания на фазу поиска, тем самым моделируя поведение, когда автомобиль, въехавший на парковку, выбирает (ищет) место для стоянки. После того, как место выбрано, он становится на стоянку, т.е. заявка переходит на обслуживание во второй фазе $\cdot/G/n/0$. Здесь очередь отсутствует, а число обслуживающих приборов n – это количество мест для стоянки автомобилей. Если же по истечении времени поиска (пребывания на первой фазе), клиент не смог найти свободный прибор для стоянки (на второй фазе), он направляется к выезду, т.е. заявка отправляется на вход фазы 3 СМО парковки. Также заявка попадает на вход фазы 3 СМО парковки и после завершения стоянки.

Заключительным этапом является фаза выезда (фаза 3). Все заявки, прошедшие через фазу 1 и фазу 2 (успешно или нет – не имеет значения), попадают на данный этап. Данная фаза представлена, как система массового обслуживания $\cdot/G/n/\infty$ и является полным аналогом фазы 1 за одним исключением – на данной фазе невозможны потери и

клиенты не покидают очередь. После того, как заявка закончила обслуживание на фазе выезда, она покидает всю систему.

3. Реализация имитационной модели

Для реализации описанной выше модели и расчёта характеристик функционирования парковки был использован python-фреймворк Salabim [3]. Данный фреймворк реализует дискретно-событийный подход к моделированию и имеет ряд преимуществ, таких как симуляция в реальном времени, анимация процесса моделирования, стохастическое моделирование случайных величин различных видов, базовая статистическая обработка результатов и другое.

Для моделирования входящего нестационарного пуассоновского потока в работе был применен метод прореживания [1,5]. На рис. 3 представлен его алгоритм. Входными данными является информация об ожидаемой интенсивности входящего транспортного потока паркинга, которая определяется значениями, представленными на рис. 2.

```
a0 = 0.0;
n = 0;
while (an < τ) {
  s = an;
  do {
    s = s + Exponential(1 / λmax);
    u = Uniform(0, λmax);
  } while (u > λ(s));
  an+1 = s;
  n++;
}
return a1, a2, ..., an-1;
```

Рис. 3. Метод прореживания

Представленный алгоритм, а также все необходимые элементы разработанной многофазной модели обслуживания, были реализованы средствами фреймворка Salabim. Длительности обслуживания на каждой из фаз моделировались как экспоненциально распределённые случайные величины с соответствующими заданными параметрами. Список задаваемых параметров (входных данных) представлен на рис. 4.

Параметры входящего потока:

- длительность моделирования;
- среднее количество автомобилей за 1 день;
- интенсивность автомобилей по часам.

Фаза 1. Въезд:

- размер полностью доступной очереди;
- интенсивность оставления полностью доступной очереди;
- размер недоступных очередей;

Фаза 2. Стоянка:

- время обслуживания при поиске;
- количество приборов обслуживания;
- время обслуживания при стоянке.

Фаза 3. Выезд:

- количество приборов обслуживания;
- время обслуживания.

Рис. 4. Настраиваемые параметры при моделировании

После выбора параметров и запуска моделирования на экран выводится лог событий, отображающий процесс моделирования системы и её текущее состояние (рис. 5). При помощи данного лога можно определить, сколько заявок обработано на данный момент, сколько приборов используется и т.п.

После того, как моделирование завершилось, на экран выводится информация о производительности на всех фазах. Данная статистика показывает среднюю загрузженность системы (число занятых приборов) за всё время моделирования, её дисперсию и другие вероятностные характеристики. Например, на рис. 6 показана загрузженность фазы 2, т.е. среднее число автомобилей, находившихся непосредственно на стоянке, за 1 день моделирования, при 100 приборах обслуживания и суммарным числом входящих заявок, равным 2000.

На основе результатов моделирования вычисляются различные вероятностные характеристики, в том числе и эмпирические распределения вероятностей загруженности системы (числа занятых приборов). Например, на рис. 7 представлены распределения вероятностей числа автомобилей на стоянке. На данном рисунке показано, как может изменяться это распределение при неизменном входящем потоке и различном количестве обслуживающих приборов (числе парковочных мест).

Line#	time	current component	action	information
			line numbers refers to	main.py
			default environment initialize	
105			main create	
105	0.000	main	current create	
106			generator create	
106			generator activate	scheduled for 0.000 @ 74 process=process
108			parking create	capacity=100
109			entry create	capacity=2
110			departure create	capacity=2
113			main run	scheduled for 86399.000 @ 113+
74	0.000	generator	current create	
78			car.0 create	
78			car.0 activate	scheduled for 0.000 @ 84 process=process
79			generator hold	scheduled for 0.000 @ 79+
84	0.000	car.0	current create	
85			car.0 request honor	request for 1 from entry
85			car.0 request honor	scheduled for 0.000 @ 85+
79+	0.000	generator	current create	
78			car.1 create	
78			car.1 activate	scheduled for 0.000 @ 84 process=process
79			generator hold	scheduled for 111.250 @ 79+
85+	0.000	car.0	current create	
86			car.0 hold	scheduled for 2.852 @ 86+
84	0.000	car.1	current create	
85			car.1 request honor	request for 1 from entry
85			car.1 request honor	scheduled for 0.000 @ 85+
85+	0.000	car.1	current create	
86			car.1 hold	scheduled for 61.990 @ 86+

Рис. 5. Лог моделирования

Statistics of Available quantity of parking at 86399			
	all	excl.zero	zero
duration	86399	75974.958	10424.042
mean	36.486	41.492	
std.deviation	37.165	36.919	
minimum	0	1	
median	14	30	
90% percentile	91	92	
95% percentile	93	93.087	
maximum	100	100	

Рис. 6. Загруженность фазы 2 за 1 день моделирования

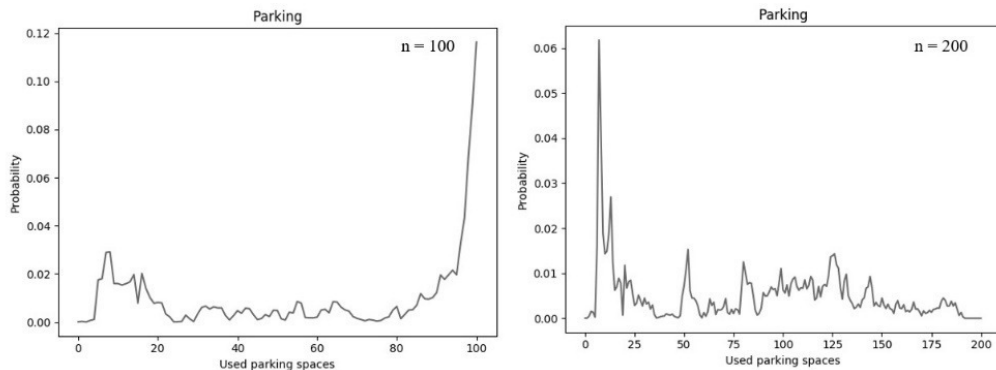


Рис. 7. Эмпирические распределения вероятностей числа занятых парковочных мест при различных значениях размера парковки n : слева – $n = 100$, справа – $n = 200$.

Заключение

В настоящей работе построена и реализована стохастическая модель парковочной территории в виде многофазной системы массового обслуживания. Данная модель разработана с целью решить одну из проблем проектирования транспортных узлов – это

парковка автомобилей. Имитационная модель реализована в виде приложения, функционирующего на фреймворке Salabim. Результатом работы приложения являются статистические вероятностные характеристики моделируемой парковочной территории.

ЛИТЕРАТУРА

1. Лабскер Л.Г. Вероятностное моделирование в финансово-экономической области. – М.: Альпина Паблшер, 2002. – 224 с.
2. Frank A. Haight Mathematical theories of traffic flow. – New York London: Academic press, 1963. – 286 с.
3. Salabim website [Электронный ресурс]. – 2017. – URL: <http://www.salabim.org/manual/> (дата обращения: 10.04.2018).
4. Блюдин А.А., Романкова А.А., Буркина В.А., Власов А.А. Моделирование работы парковок // Современная техника и технологии. 2014. № 12 [Электронный ресурс]. – URL: <http://technology.snauka.ru/2014/12/5097> (дата обращения: 01.02.2017).
5. Моделирование пуассоновского процесса [Электронный ресурс]. – 2014. – URL: <https://habrahabr.ru/post/210432/> (дата обращения: 01.02.2018).

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ОБРАБОТКИ ТЕКСТОВОЙ ИНФОРМАЦИИ В СИСТЕМАХ ТЕСТИРОВАНИЯ

А.А. Погода, А.А. Назаров

Томский государственный университет
aapoguda@gmail.com, anazarov@fpmk.tsu.ru

Введение

Проблема анализа текстовой информации связана с выявлением в его тематико-содержательной природе собственно структуры, содержания, правил построения и языковых особенностях. Всё чаще в образовательных организациях возникает потребность проводить проверку остаточных знаний обучающихся, аннотирования и реферирования текстов, создания на их основе новых знаний, справок, отчётов и др.

Как правило, проверку оценочных знаний проводят с помощью компьютерного тестирования или бланковых тестов. Для систем тестирования существует большое количество моделей и алгоритмов, обрабатывающих ответы тестируемых, но многие из них имеют существенные недостатки [1]. В [2] предлагается использовать алгоритм NeuroLD для анализа ответов на вопросы открытой формы, поскольку данный алгоритм в большей степени лишен основных недостатков. Данный алгоритм удовлетворяет предъявляемым требованиям к современным системам тестирования [4], поскольку использует семантический анализ текста, поэтому был выбран для задачи обработки входящих строк.

Идея алгоритма NeuroLD заключается в применении алфавитно-взвешенного редакционного расстояния, абстрактных уровней, разбиении предложений на узкие по смыслу кластеры, использовании словарей для нормализации слов и применении нейронных сетей Кохонена. Данный подход позволяет алгоритму обучаться на существующих ответах, позволяет приводить нормализацию слов, а подключенный словарь русского языка – учитывать синонимы и антонимы и анализировать толкование того или иного слова.

В [1] произведены результаты эксперимента, в котором сравнивались результаты программного модуля, реализованного на разработанном алгоритме NeuroLD, и алгоритмов, используемых в системах антиплагиата: Антиплагиат, Miratools и Advego Plagiatus. Результат показал, что разработанный алгоритм NeuroLD хорошо справился со всеми видами тестов и показал результаты, не уступающие, а иногда и превосходящие результаты работы существующих систем.

Применение алгоритма NeuroLD в системе антиплагиата будет более эффективным по сравнению с распространенным методом «шинглов», значительным недостатком которого является отсутствие возможности обработки синонимов. Использование в предложенном алгоритме NeuroLD фильтрации текста, стемминга и преобразования

символов позволяет находить заимствованные тексты даже при их незначительной модификации, а подключенный словарь – учитывать синонимы, сокращения, замены русских букв на латинские и другие уловки, которыми активно пользуются при написании работ.

В данной статье проводится исследование системы массового обслуживания с неограниченным числом обслуживающих приборов и нестационарным пуассоновским потоком, на вход которой поступают ответы обучающихся в виде строк.

1. Постановка задачи

Для реализации математической модели обработки входящих потоков строк в алгоритм NeuroLD с учётом исходящего потока используем теорию марковских потоков [3]. Входящими строками будут являться ответы тестируемых на вопросы закрытой и открытой формы. Интенсивность входящего потока $\lambda(t)$ зависит от времени, соответственно, это система массового обслуживания с нестационарным пуассоновским потоком.

На рис. 1 $\lambda(t)$ – поток строк, поступающих в алгоритм обработки ответов NeuroLD (система массового обслуживания), $B(x)$ – произвольное распределение, $m(t)$ – число событий выходящего потока, наступивших за время T на интервале $0 \leq t \leq T$.

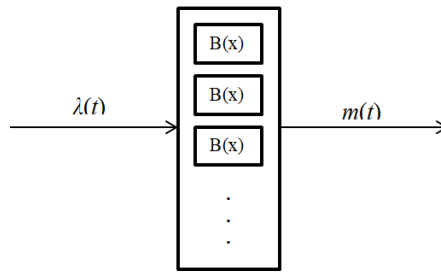


Рис. 1. Система массового обслуживания

Для анализа потока, применим метод марковского суммирования. Он заключается в том, рассматривается ось времени t на оси времени выделяется точка $t = 0$, точка $t = T$ и $t = t_0$ при $t_0 \rightarrow -\infty$. На оси времени отметим моменты наступления входящего потока и построим ещё одну ось. Для $t < 0$ обозначим $S(t) = B(T - t) - B(-t)$, т.е. вероятность того, что заявка входящего потока, поступившая в некоторый момент времени t , завершит свое обслуживание на выделенном интервале (наступит событие выходящего потока).

Рассмотрим ось T и отметим события входящего потока (рис. 2).

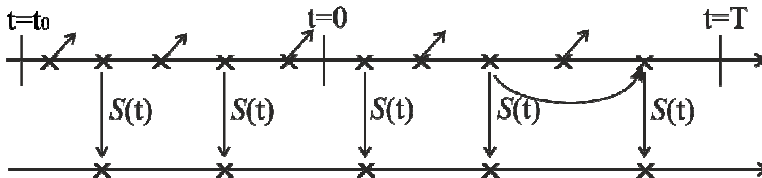


Рис. 2. События входящего потока

Необходимо найти распределение вероятностей $m(T)$. Рассмотрим $m(t)$ – число событий выходящего потока, наступивших за время T на интервале $[0, T]$. Поскольку входящий поток марковский нестационарный, то процесс $m(t)$ тоже марковский.

2. Система дифференциальных уравнений Колмогорова

Обозначим $P\{m(t) = m\} = P\{m, T\}$, тогда для распределения вероятностей получаем систему дифференциальных уравнений Колмогорова

$$P(m, t + \Delta t) = P(m, t)(1 - \lambda(t)\Delta t) + \lambda(t)\Delta t [P(m-1, t)S(t) + P(m, t)(1 - S(t))] + o(\Delta t),$$

$$\frac{\partial P(m, t)}{\partial t} = -\lambda(t)P(m, t) + \lambda(t)S(t)P(m-1, t) + \lambda(t)(1 - S(t))P(m, t).$$

Получили бесконечную систему дифференциальных уравнений первых порядков с переменными коэффициентами.

Перейдём к характеристическим функциям с переменными уравнениями.

$$\frac{\partial H(u, t)}{\partial t} = \lambda(t)S(t)P(m-1, t) - \lambda(t)S(t)P(m, t),$$

$$\begin{cases} \frac{\partial H(u, t)}{\partial t} = \lambda(t)S(t)(e^{ju} - 1)H(u, t), \\ H(u, t_0) = 1. \end{cases}$$

$$H(u, t) = \exp\left\{ (e^{ju} - 1) \int_{t_0}^t \lambda(x)S(x)dx \right\}.$$

При $t_0 \rightarrow -\infty, t = T$

$$H(u, T) = \exp\left\{ (e^{ju} - 1) \int_{-\infty}^T \lambda(x)S(x)dx \right\}.$$

Число событий выходящего потока, наступивших за время T на интервале $[0, T]$ при нестационарном входящем потоке с параметром $\lambda(t)$ и при произвольном обслуживании, является пуассоновским с параметром $a(T)$:

$$a(T) = \int_{-\infty}^T \lambda(x)S(x)dx.$$

$$\begin{aligned} a(T) &= \int_{-\infty}^0 \lambda(t)S(t)dt + \int_0^T \lambda(t)S(t)dt = \int_{-\infty}^0 \lambda(t)[B(T-t) - B(-t)]dt + \int_0^T \lambda(t)B(T-t)dt = \\ &= \int_{-\infty}^0 \lambda(t)[B(T-t) - 1 + 1 - B(-t)]dt + \int_0^T \lambda(t)[B(T-t) - 1 + 1]dt = \\ &= \int_{-\infty}^0 \lambda(t)[B(T-t) - 1]dt + \int_{-\infty}^0 \lambda(t)[1 - B(-t)]dt + \int_0^T \lambda(t)[B(T-t) - 1]dt + \\ &+ \int_0^T \lambda(t)dt = \int_0^T \lambda(t)dt + \int_{-\infty}^T \lambda(t)[B(T-t) - 1]dt + \int_{-\infty}^0 \lambda(t)(1 - B(-t))dt = \\ &= \int_0^T \lambda(t)dt + \int_0^T \lambda(t)(T-x)(B(x) - 1)dx + \int_0^{\infty} \lambda(-x)(1 - B(x))dx = \end{aligned}$$

$$= \int_0^T \lambda(t) dt + \int_0^{\infty} [\lambda(-x) - \lambda(T-x)](1-B(x)) dx.$$

Если $\lambda(t) \equiv \lambda$, то $a(T) = \lambda T$ и не зависит от $B(x)$. Если $\lambda(t) \neq \lambda$, то $a(T)$ слабо зависит от $B(x)$, так как $1 - B(x) \xrightarrow{x \rightarrow \infty} 0$ и второй интеграл ограничен при $\forall T$.

Заключение

В настоящей работе рассмотрена математическая модель обработки входящих потоков строк в алгоритм NeuroLD с учётом исходящего потока. В рамках исследования рассматривалась модель при экспоненциальном обслуживании и при нестационарном входящем потоке [5].

Результаты исследования показали, что выходящий поток системы в условии растущего времени обслуживания является асимптотически простейшим и интенсивность выходящего потока совпадает с интенсивностью входящего.

Рассматриваемая система не теряет поступающие ответы тестируемых, поскольку все ответы обрабатываются.

ЛИТЕРАТУРА

1. *Погуда А.А.* Модели и алгоритмы контроля знаний по гуманитарным дисциплинам: дис. канд. тех. наук. Томский государственный университет систем управления и радиоэлектроники, Томск, 2016. – 174 с.
2. Handbook of Research on Estimation and Control Techniques in E-Learning Systems / Vardan Mkrttchian; Alexander Bershadsky; Alexander Bozhday; Mikhail Kataev; Sergey Kataev / Chapter 10. Mitsel A.A. and Poguda A.A. An Integrated Approach to Automated Testing Knowledge. – Published in the United States of America by Information Science Reference (an imprint of IGI Global) 701 E. Chocolate Avenue Hershey PA, USA 17033, 2016. – P. 128–140.
3. *Назаров А.А., Терпугов А.Ф.* Теория массового обслуживания. – Томск: Изд-во НТЛ, 2005. – 228 с.
4. *Чельщикова М.Б.* Теория и практика конструирования педагогических тестов: Учебное пособие. – М.: Логос, 2002. – 432 с.
5. *Лапатин И.Л., Назаров А.А.* Асимптотическое свойство выходящих потоков систем массового обслуживания с неограниченным числом приборов и входящим MAP-потокм // Автоматика и телемеханика. – 2012. – Вып. 5. – С. 57–70.

СИСТЕМА УПРАВЛЕНИЯ РЕСУРСАМИ С КУСОЧНО-ПОСТОЯННОЙ СКОРОСТЬЮ ПОСТУПЛЕНИЯ РЕСУРСОВ И ПОСТОЯННОЙ ИНТЕНСИВНОСТЬЮ МОМЕНТОВ ПОТРЕБЛЕНИЯ РЕСУРСОВ

Н.М. Ревенко

*Томский государственный университет
rammixtsu@gmail.com*

Введение

Управление запасами отвечает на простой вопрос: "Что, когда и сколько надо закупить, чтобы хватило, и не осталось лишнего?" Вообще, управление запасами предприятия и их оптимизация – очень интересная тема, т.к. позволяет заниматься ей как узкоспециально – не входя в противоречия с другими бизнес-процессами и оптимизируя только запасы, так и очень широко – смотря через призму управления товарными запасами на предприятии на работу всего предприятия. Из-за бурного развития логистики в последнее время, многие специалисты говорят о проблеме эффективного управления запасами, где она рассматривается с точки зрения минимизации совокупных издержек; т.е. оптимизация товарных запасов рассматривается в первую очередь как снижение затрат на их создание и обслуживание. Это, действительно, очень важный момент мероприятий по оптимизации запасов на предприятии, но очевидно, что управление товарно-материальными запасами, кроме логистического, имеют – как минимум – ещё финансовый и маркетинговый аспект. Соответственно эффективность

управления запасами не может оцениваться только издержками на их создание и обслуживание, а стратегия управления запасами не может строиться только на снижении этих издержек, и оптимальное управление запасами подразумевает такую политику управления запасами, которая гармонизирует все аспекты, связанные с этой задачей, а не только оптимизацию запасов на складе [2].

Это так же применимо для товаров типа газов, воды в хранилищах, например, для поддержания необходимого запаса объёма товара, и в случае его избытка соответствующим образом менять политику накопления запасов, чтобы вернуть уровень к нормальному значению.

Проводится исследование математической модели системы управления запасами с релейным управлением объёмом накопленных запасов. Рассмотрен случай гиперэкспоненциального распределения объёмов потребления ресурсов для двух видов систем. Найдено явное выражение для стационарной плотности распределения значения запаса в системе. Приводятся результаты численного эксперимента.

В [4–6] рассматриваются математические модели деятельности фонда социального страхования с релейным управлением капиталом фонда. В [3] также рассмотрено релейно-гистерезисное управление капиталом такого фонда, когда выплаты по страховым случаям и выплаты на финансирование социальных программ образуют пуассоновские потоки событий с постоянной и переменной интенсивностями соответственно, а величины выплат являются одинаково распределёнными независимыми случайными величинами с экспоненциальной функцией распределения. В [4] исследуются основные характеристики деятельности фонда социального страхования в случае непрерывной скорости поступления денежных средств и экспоненциально распределённых страховых выплат. В [5] построена и исследована математическая модель деятельности некоммерческого фонда в случае пуассоновского потока поступающих платежей постоянной интенсивности при экспоненциальном распределении страховых премий и релейного управления капиталом.

В данной работе исследуется модель, аналогичная [1,3], в случае, когда объёмы запаса на расходование имеют гиперэкспоненциальное распределение.

1. Постановка задачи

Рассмотрим систему, где v и λ меняются при достижении системой уровня объёма запаса $s(t) = S$, и обозначим $v = v(s)$, $\lambda = \lambda(s)$.

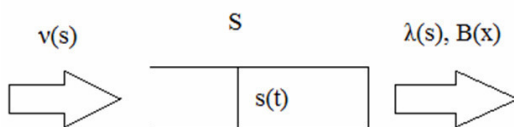


Рис. Система управления запасами

$$v(s) = \begin{cases} v_1, & s < S, \\ v_2, & s > S, \end{cases} \quad \lambda(s) = \begin{cases} \lambda_1, & s < S, \\ \lambda_2, & s > S. \end{cases}$$

Условие существования стационарного режима в рассматриваемой системе имеет вид $v_1 > \lambda_1 b$, $\lambda_2 b > v_2$. Здесь b – математическое ожидания функции распределения $B(x)$.

Таким образом, если $v_1 > \lambda_1 b$, $\lambda_2 b > v_2$ и $s(t) < S$, то объём ресурса в системе будет увеличиваться в среднем, а при достижении уровня S и его превышении, т.е. $s(t) > S$, в связи с уменьшением скорости поступления ресурсов и увеличением интен-

сивности моментов потребления, объём ресурсов будет уменьшаться; S – некоторое пороговое значение уровня запасов $s(t)$.

Заметим, что процесс $s(t)$ может принимать отрицательные значения $s(t) < 0$ и система продолжает функционировать, откладывая исполнение заявки на потребление ресурсов до момента накопления необходимого количества.

Из описания математической модели следует, что случайный процесс $s(t)$ является марковским с непрерывным временем t и непрерывным множеством значений $-\infty < s < +\infty$.

2. Уравнение Колмогорова

Определим, корректна ли построена математическая модель и выполним её исследование. Обозначим плотность распределения $P(s) = \frac{\partial P\{s(t) < s\}}{\partial s}$ и запишем следующее равенство:

$P(s + \Delta t) = P(s)(1 - \lambda(s)\Delta t) + \lambda(s)\Delta t \int_0^\infty P(s+x, t) dB(x) + o(\Delta t)$, из которого

получим уравнение $v(s)P'(s) + \lambda(s)P(s) = \lambda(s) \int_0^\infty P(s+x) dB(x)$. Обозначим

$P(s) = \begin{cases} P_1(s), & s < S, \\ P_2(s), & s \geq S. \end{cases}$ В предположении, что плотность распределения имеет разрыв

при объёме, равном S , будем считать, что $P_2(s)$ непрерывна справа.

Рассмотрим случай, когда

$$B(x) = \sum_{i=1}^n b_i (1 - e^{-\mu_i x}). \quad (1)$$

Тогда $b = \sum_{i=1}^n \frac{b_i}{\mu_i}$. Получим два интегро-дифференциальных уравнения

$$\begin{aligned} v_2 P_2'(s) + \lambda_2 P_2(s) &= \lambda_2 \int_0^\infty P_2(s+x) dB(x), \quad s \geq S, \\ v_1 P_1'(s) + \lambda_1 P_1(s) &= \lambda_1 \int_0^{S-s} P_1(s+x) dB(x) + \lambda_2 \int_{S-s}^\infty P_2(s+x) dB(x), \quad s < S. \end{aligned} \quad (2)$$

Подставим в уравнения (2) выражение (1):

$$\begin{aligned} v_2 P_2'(s) + \lambda_2 P_2(s) &= \lambda_2 \sum_{i=1}^n \left(b_i \mu_i \int_0^\infty P_2(s+x) e^{-\mu_i x} dx \right), \quad s \geq S, \\ v_1 P_1'(s) + \lambda_1 P_1(s) &= \sum_{i=1}^n \left(\lambda_1 b_i \mu_i \int_0^{S-s} P_1(s+x) e^{-\mu_i x} dx + \lambda_2 b_i \mu_i \int_{S-s}^\infty P_2(s+x) e^{-\mu_i x} dx \right), \quad s < S. \end{aligned} \quad (3)$$

Поочередно для $i = \overline{1, n}$ обе части уравнения умножим на $e^{-\mu_i s}$, вычислим $\frac{\partial}{\partial s}$, умножим на $e^{\mu_i s}$. В итоге получим однородное дифференциальное уравнение $n+1$ -го порядка, решая которое, получим $P_2(s) = \sum_{i=1}^{n+1} C_i e^{z_i s}$. Здесь $z_{1,2..n+1}$ – корни полученного дифференциального уравнения, $z_{1,2..n} \geq 0$, $z_{n+1} < 0$. Из условия $P_2(+\infty) = 0$ получим

$$P_2(s) = C_{n+1} e^{z_{n+1}s}. \quad (4)$$

Подставив $P_2(s)$ из (4) во второе уравнение из (3) и проделав аналогичные действия, получим дифференциальное уравнение $n+1$ -го порядка, из которого найдём $P_1(s)$: $P_1(s) = \sum_{i=1}^{n+1} C_i e^{z_i s}$. Здесь $C_{1,2,..,n+1}$ – другие константы, никак не связанные с предыдущими, $z_{1,2,..,n} > 0$, $z_{n+1} = 0$, $z_{1,2,..,n+1}$ – корни уравнения полученного дифференциального уравнения. Из условия $P_1(-\infty) = 0$ получим

$$P_1(s) = \sum_{i=1}^n C_i e^{z_i s}. \quad (5)$$

Покажем, что будет действительно n положительных корней. Подставим (4) и (5) во второе уравнение (3). Прделав простые операции, и группируя слагаемые, получим:

$$C_i e^{z_i s} \left(v_1 z_i + \lambda_1 + \lambda_1 \frac{b_1 \mu_1}{z_i - \mu_1} + \dots + \lambda_1 \frac{b_n \mu_n}{z_i - \mu_n} \right) = 0, \quad i = \overline{1, n}. \quad (6)$$

Обозначим $f(z) = \lambda_1 \frac{b_1 \mu_1}{\mu_1 - z} + \dots + \lambda_1 \frac{b_n \mu_n}{\mu_n - z}$. Покажем, что (6) имеет ровно n различных положительных корней. Будем считать, что $\mu_1 < \mu_2 < \dots < \mu_n$. Имеем: $f'(z) = \lambda_1 \left(\frac{b_1 \mu_1}{(\mu_1 - z)^2} + \dots + \frac{b_n \mu_n}{(\mu_n - z)^2} \right) > 0$. Функция $f(z)$ монотонно возрастает. На интервале $0 < z < \mu_1$ функция $f(z)$ непрерывна, монотонно возрастает и принимает значения $\lambda_1 < f(z) < \infty$. При выполнении условия $v_1 > \lambda_1 b$ уравнение (6) имеет по крайней мере один корень $z_1 > 0$.

Далее рассмотрим функцию $f(z)$ на интервале $\mu_{k-1} < z < \mu_k$, где $f(z)$ непрерывна, монотонно возрастает и принимает значения $-\infty < f(z) < +\infty$, поэтому уравнение (6) на интервале $\mu_{k-1} < z < \mu_k$ также имеет по крайней мере один корень $z_k > 0$.

Количество рассматриваемых интервалов равно n , совпадающее с числом положительных корней $z = z_k$, $k = \overline{1, n}$.

3. Нахождение констант

Подставляя (4) и (5) во второе уравнение (3), получаем уравнение

$$\begin{aligned} & C_1 \lambda_1 \left(\frac{b_1 \mu_1}{z_1 - \mu_1} e^{z_1 s} e^{-\mu_1(S-s)} + \dots + \frac{b_n \mu_n}{z_1 - \mu_n} e^{z_1 s} e^{-\mu_n(S-s)} \right) + \dots + \\ & + C_n \lambda_1 \left(\frac{b_1 \mu_1}{z_n - \mu_1} e^{z_n s} e^{-\mu_1(S-s)} + \frac{b_n \mu_n}{z_n - \mu_n} e^{z_n s} e^{-\mu_n(S-s)} \right) - \\ & - C_{n+1} \lambda_2 \left(\frac{b_1 \mu_1}{z_{n+1} - \mu_1} e^{z_{n+1} s} e^{-\mu_1(S-s)} + \dots + \frac{b_n \mu_n}{z_{n+1} - \mu_n} e^{z_{n+1} s} e^{-\mu_n(S-s)} \right) = 0. \end{aligned} \quad (7)$$

Группируя в (7) слагаемые по $e^{-\mu_i(S-s)}$, $i = \overline{1, n}$, получим n независимых уравнений

$$\left(C_1 \lambda_1 \frac{b_1 \mu_1}{z_1 - \mu_1} e^{\bar{z}_1 S} + \dots + C_n \lambda_1 \frac{b_n \mu_n}{z_n - \mu_n} e^{\bar{z}_n S} - C_{n+1} \lambda_2 \frac{b_1 \mu_1}{z_{n+1} - \mu_1} e^{\bar{z}_n S} \right) e^{-\mu_1(S-s)} = 0,$$

...

$$\left(C_1 \lambda_1 \frac{b_n \mu_n}{z_1 - \mu_n} e^{\bar{z}_1 S} + \dots + C_n \lambda_1 \frac{b_n \mu_n}{z_n - \mu_n} e^{\bar{z}_n S} - C_{n+1} \lambda_2 \frac{b_n \mu_n}{z_{n+1} - \mu_n} e^{\bar{z}_n S} \right) e^{-\mu_n(S-s)} = 0.$$

Составим квадратную матрицу A , где $A_{i,j} = \lambda_1 \frac{b_i \mu_i}{z_j - \mu_i} e^{\bar{z}_i S}$, $i, j = \overline{1, n}$,

$$A_{i,n+1} = \lambda_2 \frac{b_i \mu_i}{z_{n+1} - \mu_i} e^{\bar{z}_i S}, \quad i = \overline{1, n}, \quad A_{n+1,j} = \frac{e^{\bar{z}_j S}}{z_j}, \quad j = \overline{1, n}, \quad A_{n+1,n+1} = -\frac{e^{\bar{z}_{n+1} S}}{z_{n+1}},$$

где $n+1$ -я строка – условие нормировки для $P(s)$. Обозначим вектор $F = (0, 0, \dots, 0, 1)^T$ размерности $n+1$.

Обозначим вектор $C = (C_1, C_2, \dots, C_n, C_{n+1})^T$. Решая матричное уравнение $AC = F$, однозначно найдем $C_1, C_2, \dots, C_n, C_{n+1}$. В итоге имеем:

$$P(s) = \begin{cases} \sum_{i=1}^n C_i e^{\bar{z}_i s}, & s < S, \\ C_{n+1} e^{\bar{z}_{n+1} s}, & s \geq S. \end{cases}$$

Заключение

В данной работе построена математическая модель системы управления запасами. Получено аналитическое выражение для стационарной плотности распределения значений объёма запасов при гиперэкспоненциальном распределении объёмов потребления объёмом запасов. Предложенный подход может быть применен к аналогичным задачам при различных распределениях объёмов расходования ресурсов.

ЛИТЕРАТУРА

1. Назаров А.А., Бронер В.И. Система управления запасами с гиперэкспоненциальным распределением объёмов потребления ресурсов // Вестник Томского государственного университета. – 2016. – № 1. – С. 43–48.
2. <http://upravlenie-zapasami.ru>
3. Вальц О.В., Змеев О.А. Математическая модель деятельности фонда социального страхования при экспоненциальных страховых выплатах и со случайными расходами на социальные программы // Вестник Томского государственного университета. – 2004. – № 284. – С. 37–41.
4. Змеев О.А. Математическая модель деятельности фонда социального страхования при экспоненциальных страховых выплатах // Вестник Томского государственного университета. – 2003. – № 280. – С. 130–135.
5. Лившиц К.И., Шифердекер И.Ю. Математическая модель деятельности некоммерческого фонда при релейном управлении капиталом // Вестник Томского государственного университета. Приложение. – 2006. – № 18. – С. 302–308.
6. Лившиц К.И., Сухотина Л.Ю., Шифердекер И.Ю. Пуассоновская модель деятельности некоммерческого фонда при релейном управлении капиталом // Вестник Томского государственного университета. Приложение. – 2006. – № 19. – С. 302–312.

СЕКЦИЯ IV. ИДЕНТИФИКАЦИЯ СТОХАСТИЧЕСКИХ СИСТЕМ

О ДОВЕРИТЕЛЬНОМ ОЦЕНИВАНИИ ПАРАМЕТРОВ AR(1) ПРОЦЕССА С НЕИЗВЕСТНЫМ СРЕДНИМ ПО ЗАШУМЛЁННЫМ НАБЛЮДЕНИЯМ

В.В. Конев, Б.Н. Назаренко

Томский государственный университет
vvkonev@mail.tsu.ru, nazarenkoB@sibmail.com

Введение

В работе авторов [1] предложена последовательная процедура оценивания параметров неустойчивого процесса AR(1) с неизвестным средним, установлено неасимптотическое нормальное распределение вектора неизвестных параметров в случае гауссовских шумов.

В данной работе рассматривается задача оценивания параметров процесса в предположении, что процесс авторегрессии недоступен прямому наблюдению и оценивание осуществляется по наблюдениям с аддитивным шумом.

Задача последовательного оценивания параметров авторегрессии по зашумлённым наблюдениям изучалась в [2], где была построена двухэтапная процедура оценивания неизвестных параметров с гарантированной среднеквадратической точностью.

Основная цель данной работы – построить доверительную область фиксированных размеров для неизвестных параметров с заданным коэффициентом доверия. Доверительная область требует двух моментов остановки, используемых для последовательной модификации базовых выборочных оценок Юла – Уокера.

1. Постановка задачи

Пусть $\{x_k\}_{k \geq 0}$ – процесс авторегрессии первого порядка с неизвестным средним, удовлетворяющий уравнению

$$x_k = \theta_1 x_{k-1} + \theta_2 + \varepsilon_k, \quad k \geq 1, \quad (1)$$

где $(\varepsilon_k)_{k \geq 1}$ – ненаблюдаемый процесс (шум), θ_1, θ_2 – неизвестные параметры. Требуется оценить вектор параметров $\theta = (\theta_1, \theta_2)'$ по наблюдениям процесса

$$y_k = x_k + \eta_k, \quad k \geq 1. \quad (2)$$

Предположим, что шумы $\{\varepsilon_k\}, \{\eta_k\}$ являются независимыми последовательностями независимых одинаково распределённых гауссовских случайных величин с нулевыми средними и дисперсиями 1 и Δ соответственно, т.е. $\varepsilon_k \sim \mathcal{N}(0,1)$ и $\eta_k \sim \mathcal{N}(0,\Delta)$. Начальное значение x_0 является случайной величиной, не зависящей от процессов $\{\varepsilon_k\}$ и $\{\eta_k\}$.

Подставляя (2) в (1), получим уравнение для $\{y_k\}$:

$$y_k = \theta_1 y_{k-1} + \theta_2 + \xi_k, \quad \xi_k = \eta_k - \theta_1 \eta_{k-1} + \varepsilon_k. \quad (3)$$

Умножим уравнение (3) по отдельности на y_{k-2} и y_{k-3} и возьмем математическое ожидание от обеих частей уравнения. Получим систему Юла – Уокера

$$\begin{cases} \mathbb{E}y_k y_{k-2} = \theta_1 \mathbb{E}y_{k-1} y_{k-2} + \theta_2 \mathbb{E}y_{k-2}, \\ \mathbb{E}y_k y_{k-3} = \theta_1 \mathbb{E}y_{k-1} y_{k-3} + \theta_2 \mathbb{E}y_{k-3}. \end{cases}$$

Переходя к выборочным средним и разрешая систему, получаем оценку Юла – Уокера для вектора неизвестных параметров: $\hat{\theta}_n = L_n^{-1}V_n$, где

$$L_n = \begin{bmatrix} \sum_{k=1}^n y_{k-1}y_{k-2} & \sum_{k=1}^n y_{k-2} \\ \sum_{k=1}^n y_{k-1}y_{k-3} & \sum_{k=1}^n y_{k-3} \end{bmatrix}, \quad V_n = \begin{bmatrix} \sum_{k=1}^n y_k y_{k-2} \\ \sum_{k=1}^n y_k y_{k-3} \end{bmatrix}.$$

2. Построение последовательной оценки Юла – Уокера. Основной результат

Для каждого числа $h > 0$ введём два момента останковки

$$\tau_1 = \tau_1(h) = \inf \left\{ n \geq 1 : \sum_{l=1}^n y_{2l-2}^2 \geq h \right\}, \quad (4)$$

$$\tau_2 = \tau_2(h) = \inf \left\{ n > \tau_1(h) : \sum_{l=\tau_1(h)+1}^n y_{2l-3}^2 \geq h \right\}, \quad (5)$$

Введём корректирующие факторы $\alpha_1(h)$, $\alpha_2(h)$, $0 < \alpha_i(h) \leq 1$, $i = 1, 2$, исходя из уравнений

$$\sum_{l=1}^{\tau_1(h)-1} y_{2l-2}^2 + \alpha_1(h)y_{2\tau_1(h)-2}^2 = h, \quad \sum_{l=\tau_1(h)+1}^{\tau_2(h)-1} y_{2l-3}^2 + \alpha_2(h)y_{2\tau_2(h)-3}^2 = h.$$

Определим последовательную оценку Юла – Уокера по формуле

$$\theta^* = L(h)^{-1}v(h), \quad (6)$$

где

$$L(h) = \begin{bmatrix} \sum_{l=1}^{\tau_1(h)} \sqrt{\beta_{2l}^1(h)} y_{2l-1} y_{2l-2} & \sum_{l=1}^{\tau_1(h)} \sqrt{\beta_{2l}^1(h)} y_{2l-2} \\ \sum_{l=\tau_1(h)+1}^{\tau_2(h)} \sqrt{\beta_{2l}^2(h)} y_{2l-1} y_{2l-3} & \sum_{l=\tau_1(h)+1}^{\tau_2(h)} \sqrt{\beta_{2l}^2(h)} y_{2l-3} \end{bmatrix}, \quad v(h) = \begin{bmatrix} \sum_{l=1}^{\tau_1(h)} \sqrt{\beta_{2l}^1(h)} y_{2l} y_{2l-2} \\ \sum_{l=\tau_1(h)+1}^{\tau_2(h)} \sqrt{\beta_{2l}^2(h)} y_{2l} y_{2l-3} \end{bmatrix},$$

$$\beta_k^i = \begin{cases} 1, & \text{если } i = 1, 1 \leq k < \tau_1(h), \\ \alpha_1(h), & \text{если } i = 1, k = \tau_1(h), \\ 1, & \text{если } i = 2, \tau_1(h) + 1 \leq k < \tau_2(h), \\ \alpha_2(h), & \text{если } i = 2, k = \tau_2(h), \end{cases}$$

Для получения свойств этой оценки нам потребуется следующий результат.

Теорема 1. Пусть на вероятностном пространстве (Ω, \mathcal{F}, P) с фильтрацией $(\mathcal{F}_k)_{k \geq 0}$ задана пара квадратично интегрируемых мартингалов $(M_k^1, \mathcal{F}_k)_{k \geq 0}$ и $(M_k^2, \mathcal{F}_k)_{k \geq 0}$ с квадратическими характеристиками $\{\langle M^{(1)} \rangle_n\}_{n \geq 1}$, $\{\langle M^{(2)} \rangle_n\}_{n \geq 1}$, удовлетворяющими условиям:

$$(a) \quad P(\langle M^{(1)} \rangle_\infty = +\infty) = 1 \text{ и } P(\langle M^{(2)} \rangle_\infty = +\infty) = 1.$$

$$(b) \quad \text{Law}(\Delta M_k^{(1)} | \mathcal{F}_{k-1}) = \mathcal{N}(0, \sigma_1^2(k-1)), \quad k = 1, 2, \dots,$$

$$\text{Law}(\Delta M_k^{(2)} | \mathcal{F}_{k-1}) = \mathcal{N}(0, \sigma_2^2(k-1)), \quad k = 1, 2, \dots,$$

т.е. с \mathcal{F}_{k-1} -условно гауссовскими приращениями $\Delta M_k^l = M_k^{(l)} - M_{k-1}^{(l)}$ с параметрами 0 и $\sigma_i^2(k-1) = \mathbb{E}((\Delta M_k^{(i)})^2 | \mathcal{F}_{k-1})$.

Для $h > 0$ определим два момента остановки

$$\tau_1 = \tau_1(h) = \inf \left\{ n > 0 : \sum_{k=1}^n \sigma_1^2(j-1) \geq h \right\}, \quad \tau_2 = \tau_2(h) = \inf \left\{ n > \tau_1(h) : \sum_{k=\tau_1(h)+1}^n \sigma_2^2(j-1) \geq h \right\},$$

где $\tau_0 = \tau_0(h) = 0$, $\inf\{\emptyset\} = +\infty$, и пару случайных величин

$$m_1(h) = \frac{1}{\sqrt{h}} \sum_{k=1}^{\tau_1} \sqrt{\beta_k(h,1)} \Delta M_k^{(1)}, \quad m_2(h) = \frac{1}{\sqrt{h}} \sum_{k=\tau_1+1}^{\tau_2} \sqrt{\beta_k(h,2)} \Delta M_k^{(2)},$$

где

$$\beta_k(h,l) = \begin{cases} 1, & \text{если } \tau_{l-1}(h) < k < \tau_l(h), \\ \alpha_l(h), & \text{если } k = \tau_l(h); \end{cases}$$

и корректирующие факторы $\alpha_j(h)$, $0 < \alpha_j(h) \leq 1$, определяемые из уравнений

$$\sum_{k=\tau_{j-1}+1}^{\tau_j-1} \sigma^2(k-1) + \alpha_j(h) \sigma_l^2(\tau_l(h)-1) = h.$$

Тогда для любого $h > 0$ вектор $m(h) = (m_1(h), m_2(h))'$ имеет 2-мерное стандартное гауссовское распределение, т.е. $m(h) \sim \mathcal{N}(0, I_2)$.

Сформулируем основной результат работы.

Теорема 2. Для каждого $h > 0$ $\text{Law} \left(\frac{L(h)}{\sqrt{dh}} (\theta^* - \theta) \right) = \mathcal{N}(0, I_2)$.

Доказательство.

Подставляя (3) в (6), получаем

$$\theta^* = \theta + L(h)^{-1} \begin{bmatrix} \sum_{l=1}^{\tau_1(h)} \sqrt{\beta_{2l}^1(h)} y_{2l-2} \xi_{2l} \\ \sum_{l=\tau_1(h)+1}^{\tau_2(h)} \sqrt{\beta_{2l}^2(h)} y_{2l-3} \xi_{2l} \end{bmatrix}. \quad (7)$$

Перепишем это уравнение в виде $\theta^* = \theta + \sqrt{dh} L(h)^{-1} m(h)$, где $d = \Delta(1 + \theta_1^2) + 1$,

$$m(h) = \frac{1}{\sqrt{h}} \begin{bmatrix} \sum_{l=1}^{\tau_1(h)} \sqrt{\beta_{2l}^1(h)} y_{2l-2} \tilde{\xi}_{2l} \\ \sum_{l=\tau_1(h)+1}^{\tau_2(h)} \sqrt{\beta_{2l}^2(h)} y_{2l-3} \tilde{\xi}_{2l} \end{bmatrix}, \quad (8)$$

$$\tilde{\xi}_k = \frac{\xi_k}{\sqrt{d}} \sim \mathcal{N}(0, 1).$$

Далее нам потребуется естественная фильтрация $\{\mathcal{F}_k\}_{k \geq 0}$, порождённая процессами ε_k и η_k : $\mathcal{F}_0 = \sigma(x_0)$, $\mathcal{F}_k = \sigma(x_0, \varepsilon_1, \dots, \varepsilon_{2k-1}, \varepsilon_{2k}, \eta_1, \dots, \eta_{2k-1}, \eta_{2k})$, $k \geq 1$.

Введём пару мартингалов относительно этой фильтрации

$$M_0^1 = 0, \quad M_n^1 = \sum_{l=1}^n y_{2l-2} \tilde{\xi}_{2l}, \quad n \geq 1; \quad M_0^2 = 0, \quad M_n^2 = \sum_{l=1}^n y_{2l-3} \tilde{\xi}_{2l}, \quad n \geq 1.$$

Эта пара мартингалов удовлетворяет условиям Теоремы 1, причём

$$\langle M^1 \rangle_n = \sum_{l=1}^n y_{2l-2}^2, \quad \langle M^2 \rangle_n = \sum_{l=1}^n y_{2l-3}^2,$$

$$\text{Law}(\Delta M_l^1 | \mathcal{F}_{k-1}) = \mathcal{N}(0, y_{2l-2}^2), \quad \text{Law}(\Delta M_l^2 | \mathcal{F}_{k-1}) = \mathcal{N}(0, y_{2l-3}^2).$$

Применяя Теорему 1 к вектору (8) и учитывая равенство (7), приходим к утверждению Теоремы 2. ◀

Используя Теорему 2, можно построить доверительную область для параметров $\theta = (\theta_1, \theta_2)'$. Будем предполагать, что $|\theta_1| \leq L_0$, где L_0 – известная константа.

Пусть $0 < \gamma < 1$ – доверительная вероятность и a_γ – радиус круга, внутрь которого стандартная двумерная гауссовская случайная величина попадает с вероятностью γ .

По Теореме 2, $P\left\{\left\|\frac{L(h)}{\sqrt{dh}}(\theta^* - \theta)\right\| \leq a_\gamma\right\} = \gamma$, где $\|\cdot\|$ означает евклидову длину двумерного вектора. Положим $\delta_\gamma = a_\gamma \sqrt{\Delta(1 + L_0^2) + 1}$. Так как

$$\left\{\left\|\frac{L(h)}{\sqrt{h}}(\theta^* - \theta)\right\| \leq a_\gamma \sqrt{d}\right\} \subset \left\{\left\|\frac{L(h)}{\sqrt{h}}(\theta^* - \theta)\right\| \leq \delta_\gamma\right\},$$

тогда

$$P\left\{\frac{L(h)}{\sqrt{h}}(\theta^* - \theta) \in \bar{B}_{\delta_\gamma}\right\} \geq \gamma, \quad (9)$$

где \bar{B}_{δ_γ} – замкнутый круг на плоскости с центром в начале координат радиуса δ_γ .

3. Исследование асимптотического поведения оценки

Теорема 3. Для стационарного устойчивого процесса авторегрессии (1), т.е. при $|\theta_1| < 1$ верны асимптотически равенства

$$a) \frac{\sum_{k=1}^n y_{2k}^2}{n} \rightarrow \frac{\theta_2^2 + \theta_1^2 \theta_2^2 (1 + 2\theta_1) + \Delta + 1 - \theta_1^4 \Delta + \theta_1 \theta_2^2 (\theta_1 + 1) + \theta_1^2}{1 - \theta_1^4} \stackrel{\text{def}}{=} \alpha \text{ п.н. при } n \rightarrow \infty,$$

$$b) \frac{\tau_1(h)}{h} \rightarrow \frac{1}{\alpha} \text{ п.н. при } h \rightarrow \infty,$$

$$c) \frac{\tau_2(h)}{h} \rightarrow \frac{2}{\alpha} \text{ п.н. при } h \rightarrow \infty,$$

$$d) \frac{L(h)}{h} \rightarrow \begin{bmatrix} \theta_1 \alpha + \frac{\theta_2^2}{1 - \theta_1} - \theta_1 \Delta & \frac{\theta_2}{(1 - \theta_1) \alpha} \\ \theta_1^2 \alpha + \frac{\theta_2^2 (1 + \theta_2)}{1 - \theta_1} - \theta_1^2 \Delta & \frac{\theta_2}{(1 - \theta_1) \alpha} \end{bmatrix} \text{ п.н. при } h \rightarrow \infty.$$

Доказательство.

Умножая выражение (3) для y_k на y_k и повторно используя выражение (3), получим следующее уравнение для $\sum_{k=1}^n y_{2k}^2$:

$$\begin{aligned} (1 - \theta_1^4) \sum_{k=1}^n y_{2k}^2 &= \theta_1^4 (y_{2n}^2 - y_0^2) + \sum_{k=1}^n \left\{ \theta_1^2 y_{2k-2} [2\theta_1 \theta_2 + \theta_2] + \theta_2 y_{2k} \right\} + \\ &+ \sum_{k=1}^n \left\{ \theta_1^2 y_{2k-2} [2\theta_1 \xi_{2k-1} + \xi_{2k}] + \xi_{2k} y_{2k} \right\} + \sum_{k=1}^n \left\{ \theta_1 (\xi_{2k-1} + \theta_2) (\theta_1 \theta_2 + \theta_2 + \theta_1 \xi_{2k-1} + \xi_{2k}) \right\}. \end{aligned}$$

Принимая во внимание следующие предельные соотношения, которые следуют из классического усиленного закона больших и мартингалного усиленного закона больших чисел (см., например, Следствие 2 к неравенству Буркхольдера в [3])

$$\frac{\sum_{k=1}^n y_{2k}}{n} \xrightarrow[n \rightarrow \infty]{\text{п.н.}} \frac{\theta_2}{1 - \theta_1}, \quad \frac{\sum_{k=1}^n y_{2k-2} \xi_{2k-1}}{n} \xrightarrow[n \rightarrow \infty]{\text{п.н.}} -\Delta \theta_1, \quad \frac{\sum_{k=1}^n y_{2k-2} \xi_{2k}}{n} \xrightarrow[n \rightarrow \infty]{\text{п.н.}} 0,$$

$$\frac{\sum_{k=1}^n y_{2k} \xi_{2k}}{n} \xrightarrow[n \rightarrow \infty]{\text{п.н.}} \Delta + 1, \quad \frac{\sum_{k=1}^n \xi_{2k}^2}{n} \xrightarrow[n \rightarrow \infty]{\text{п.н.}} \Delta$$

и выполняя арифметические преобразования, приходим к результату а) теоремы.

Результаты б) и с) следуют непосредственно из результата а).

Для получения д) используются дополнительно соотношения

$$\frac{\sum_{k=1}^n y_{2k-1} y_{2k-2}}{n} = \theta_1 \frac{\sum_{k=1}^n y_{2k-2}^2}{n} + \theta_2 \frac{\sum_{k=1}^n y_{2k-2}}{n} + \frac{\sum_{k=1}^n y_{2k-2} \xi_{2k-1}}{n} \xrightarrow[n \rightarrow \infty]{\text{п.н.}} \theta_1 \alpha + \frac{\theta_2^2}{1 - \theta_1} - \theta_1 \Delta,$$

$$\frac{\sum_{k=1}^n y_{2k-1} y_{2k-3}}{n} = \frac{\theta_1^2 \sum_{k=1}^n y_{2k-3}^2}{n} + \frac{(\theta_1 \theta_2 + \theta_2) \sum_{k=1}^n y_{2k-3}}{n} +$$

$$+ \frac{\sum_{k=1}^n y_{2k-3} (\xi_{2k-1} + \theta_1 \xi_{2k-2})}{n} \xrightarrow[n \rightarrow \infty]{\text{п.н.}} \theta_1^2 \alpha + \frac{\theta_2^2 (1 + \theta_2)}{1 - \theta_1} - \theta_1^2 \Delta \blacktriangleleft$$

4. Результаты численного моделирования

Для проверки построенной доверительной области (9) моделировалась последовательная процедура (4), (5), (6) и вычислялась случайная величина $L(h)(\theta^* - \theta)$ при различных значениях параметров θ_1 и θ_2 при $h = 200$, $\Delta = 0.01$. Далее вычислялась выборочная вероятность попадания случайной величины $L(h)(\theta^* - \theta)$ в круг \bar{B}_{δ_γ} по 10000 реализациям процедуры. Предполагалось, что $|\theta_1| \leq 0.9$, уровень доверительной вероятности $\gamma = 0.95$.

Результаты моделирования представлены в табл. 1. Полученные данные подтверждают вывод о работоспособности метода построения доверительной области.

Таблица 1

Выборочные вероятности попадания случайной величины $L(h)(\theta^* - \theta)$ в круг \bar{B}_{δ_γ} по 10000 реализациям процедуры при $h = 200$ и $\Delta = 0.01$

$\theta_1 \setminus \theta_2$	-3	-2	-1	-0.6	0	0.6	1	2	3
-0.95	0.9512	0.9538	0.9505	0.9525	0.9562	0.9526	0.9560	0.9489	0.9531
-0.9	0.9509	0.9516	0.9522	0.9544	0.9508	0.9505	0.9538	0.9504	0.9520
-0.7	0.9483	0.9522	0.9505	0.9517	0.9532	0.9494	0.9536	0.9509	0.9534
-0.3	0.9533	0.9533	0.9502	0.9531	0.9508	0.9525	0.9533	0.9542	0.9541
0	0.9509	0.9508	0.9514	0.9511	0.9518	0.9542	0.9508	0.9521	0.9534
0.3	0.9480	0.9514	0.9538	0.9527	0.9550	0.9509	0.9507	0.9493	0.9534
0.7	0.9559	0.9518	0.9541	0.9528	0.9528	0.9534	0.9526	0.9553	0.9518
0.9	0.9511	0.9508	0.9531	0.9522	0.9529	0.9517	0.9552	0.9534	0.9494
0.95	0.9534	0.9526	0.9512	0.9542	0.9518	0.9490	0.9499	0.9533	0.9510

В табл. 2 приведены выборочные 10%-усеченные средние (т.е. из упорядоченной исходной выборки были удалены по 10% наблюдений с концов для получения устой-

чивого среднего) длины осей ($\hat{a}_{20\%}$ – наибольшая, $\hat{b}_{20\%}$ – наименьшая ось) доверительного эллипса для $(\theta^* - \theta)$ по 10000 реализациям. Из полученных результатов видно, что с ростом параметра h размеры эллипса уменьшаются.

В табл. 3 показаны результаты вычисления выборочных средних ($\bar{\theta}_i^*$) и дисперсий ($\text{var}(\theta_i^*)$) оценки θ^* при значениях параметров $\theta_1 = -0.7$, $\theta_2 = 2.8$ по 400 наблюдениям. Из приведенных данных можно заключить, что с ростом h выборочные средние приближаются к точным значениям параметров, а выборочные дисперсии уменьшаются.

Таблица 2

Выборочные 10%-усеченные средние длины осей доверительного эллипса для $(\theta^* - \theta)$ по 10000 реализациям

$\theta_1 = 0.5, \theta_2 = -0.9$			$\theta_1 = 0.5, \theta_2 = 2.8$		
h	$\hat{a}_{20\%}$	$\hat{b}_{20\%}$	h	$\hat{a}_{20\%}$	$\hat{b}_{20\%}$
20	4.395	0.01352	20	37.368	0.00765
100	1.460	0.00144	100	10.943	0.00118
200	0.863	0.00052	200	6.636	0.00041
400	0.477	0.00019	400	4.330	0.00015
$\theta_1 = -0.7, \theta_2 = -0.9$			$\theta_1 = -0.7, \theta_2 = 2.8$		
h	$\hat{a}_{20\%}$	$\hat{b}_{20\%}$	h	$\hat{a}_{20\%}$	$\hat{b}_{20\%}$
20	0.3661	0.03538	20	0.8125	0.01959
100	0.0235	0.00418	100	0.0721	0.00242
200	0.0079	0.00154	200	0.0259	0.00090
400	0.0027	0.00055	400	0.0091	0.00032

Таблица 3

Выборочные средние и дисперсии для θ^* по 400 реализациям при $\theta_1 = -0.7, \theta_2 = 2.8$

h	$\bar{\theta}_1^*$	$\text{var}(\theta_1^*)$	$\bar{\theta}_2^*$	$\text{var}(\theta_2^*)$
50	-0.859	11.934	3.021	20.985
100	-0.765	1.636	2.903	3.376
200	-0.683	0.059	2.769	0.145
300	-0.689	0.041	2.790	0.107
400	-0.696	0.027	2.795	0.070

Заключение

В данной работе была предложена последовательная процедура оценивания параметров процесса авторегрессии с неизвестным средним по зашумлённым наблюдениям.

Для нормализованного отклонения построенной оценки доказана двумерная неасимптотическая нормальность, а также приведены асимптотические результаты для длительности процедуры и нормирующего множителя. Представлены результаты численного моделирования.

ЛИТЕРАТУРА

1. Konev V.V., Nazarenko B.N. Non-asymptotic distribution of the sequential estimates of parameters in a first-order unstable autoregression with unknown mean // Международная научная конференция "Робастная статистика и финансовая математика – 2017" (03–05 июля 2017 г.) : сборник статей. Томск, 2017. С. 17–23. URL: <http://vital.lib.tsu.ru/vital/access/manager/Repository/vtls:000620488>.
2. Васильев В.А., Конеv В.В. Последовательное оценивание параметров динамических систем при наличии мультипликативной и аддитивной помех в наблюдениях / В.А. Васильев, // Автоматика и телемеханика. – 1985. – № 6. – С. 33–43.

СКОРЕЙШЕЕ ОБНАРУЖЕНИЕ РАЗЛАДКИ АВТОРЕГРЕССИОННЫХ ПРОЦЕССОВ ПРИ НЕИЗВЕСТНОЙ КОНЕЧНОЙ МОДЕЛИ

А.В. Пупков

Томский государственный университет

andrewpupkov@gmail.com

Введение

В теории стохастических процессов особую роль занимает задача обнаружения разладки. Разладкой называют любое событие, при котором происходит резкое изменение поведения случайного процесса. Например, в качестве такого изменения могут выступать смена математического ожидания случайного процесса, его дисперсии, корреляционной зависимости элементов, смена распределения шума и т.д. В реальной жизни таким изменениям может соответствовать обвал на рынке ценных бумаг, авария на производстве, несанкционированное вторжение в компьютерную сеть, обнаружение космического объекта и многое другое.

Целью задачи обнаружения разладки является наискорейшая реакция на изменение поведения случайного процесса. Для решения этой проблемы создаются статистические процедуры, одной из таких процедур является алгоритм кумулятивных сумм (CUSUM), разработанный Е.С. Пэйджем в 1954 г. в работе [1]. Оптимальность данного алгоритма была доказана Г. Лорденом в 1971 г. [2] для последовательности независимых наблюдений. Для случая зависимых случайных величин доказательство оптимальности привел Т.Л. Лай в 1998 г. [3]. Алгоритм рассматривался в параметрической постановке и требовал знания распределения случайного процесса до и после момента разладки для построения статистик логарифмического правдоподобия. Поскольку алгоритм кумулятивных сумм, в классической постановке, требует большого априорного знания о изучаемом процессе, возникает желание уменьшить это знание без потери эффективности алгоритма. Решение данной проблемы предложили В.В. Конев и С.Э. Воробейчиков в статье [4] в 2017 г. Основная идея предложенных авторами модификаций алгоритма заключается в замене статистик логарифмического правдоподобия на другую систему статистик, которая не зависит от распределения случайного процесса до и после момента разладки и обладает аналогичными статистическими свойствами, что и статистики логарифмического правдоподобия. В работе предполагалось, что в некоторый момент времени происходит резкое изменение значений параметров модели стохастической регрессии, значения которых известны до и после момента разладки. Данное допущение крайне редко выполняется на практике, поскольку до разладки идентификация параметров модели может быть произведена, а после разладки это не представляется возможным. В данной статье рассматривается модификация алгоритма, предложенного в [4], для обнаружения разладки процесса авторегрессии первого порядка AR(1) при неизвестном значении параметра процесса после момента отклонения.

1. Случай известного параметра модели после разладки

Рассмотрим процесс авторегрессии первого порядка AR(1), в котором происходит смена значения корреляционного коэффициента в момент времени v

$$\begin{aligned}x_n &= \theta_0 x_{n-1} + \varepsilon_n, \quad n = 1, 2, \dots, v-1, \\x_n &= \theta_1 x_{n-1} + \varepsilon_n, \quad n \geq v,\end{aligned}\tag{1}$$

где θ_0, θ_1 ($\theta_0 \neq \theta_1$) – значения параметров процесса до и после момента разладки соответственно, $\{\varepsilon_n\}_{n \geq 1}$ – белый шум, распределение которого предполагается неизвестным. Начальное значение x_0 и процесс $\{\varepsilon_n\}_{n \geq 1}$ независимы. Предполагается, что процесс $\{x_n\}_{n \geq 0}$ измерим относительно фильтрации $\{F_n\}_{n \geq 0}$ такой, что $F_0 = \sigma(x_0)$, $F_n = \sigma(x_0, \varepsilon_1, \dots, \varepsilon_n)$, $n \geq 1$. Заметим также, что $E(\varepsilon_n | F_{n-1}) = 0$, $E(\varepsilon_n^2 | F_{n-1}) = 1$.

Необходимо по наблюдениям процесса наискорейшим образом среагировать на изменение параметра. Рассмотрим упрощенный вариант процедуры, предложенной в [4]. Для решения поставленной задачи рассматривается функционал вида

$$J(m) = 2 \sum_{n=1}^{m-1} x_{n-1} (\theta_1 - \theta_0) \left(x_n - \frac{x_{n-1}(\theta_1 + \theta_0)}{2} \right) + C_N,$$

где

$$C_N = \sum_{n=1}^N (x_n - \theta_1 x_{n-1})^2.$$

Приращения функционала имеют вид

$$\Delta J(m) = J(m+1) - J(m) = 2x_{m-1} (\theta_1 - \theta_0) \left(x_m - \frac{x_{m-1}(\theta_1 + \theta_0)}{2} \right). \quad (2)$$

Статистики вида (2) обладает свойствами, аналогичными свойствам статистик логарифмического правдоподобия. В частности, математическое ожидание данных статистик до момента разладки имеет значение меньше нуля, а после – больше. Именно этот факт позволяет использовать данные статистики в процедуре CUSUM, вместо статистик логарифмического правдоподобия. Данные статистики является более предпочтительными, поскольку для их использования не нужно знать распределение случайного процесса до и после момента отклонения.

Процедура CUSUM имеет следующий вид

$$N = \inf \left(n \geq 1 : \max_{1 \leq k \leq n} \sum_{i=k}^n \Delta J(i) \geq C \right), \quad (3)$$

где C – некоторый порог.

Процедура хорошо реагирует на изменение коэффициента корреляции, но требует знание значения этого параметра после разладки, что на практике не всегда выполнимо. Следовательно, возникает естественное желание модифицировать процедуру таким образом, чтобы она позволяла отслеживать разладку, не зная значение параметра θ_1 . Предложенная модификация рассматривается в следующем разделе.

2. Случай неизвестного параметра после разладки

Рассмотрим модель типа (1) при условии, что параметр авторегрессии θ_1 после момента разладки неизвестен и переобозначим его как θ . Значение неизвестного параметра принадлежит некоторому множеству $\theta \in \Theta \subset (-\infty, +\infty)$. Рассмотрим функцию от параметра θ следующего вида

$$\Delta J_m(\theta) = J_{m+1}(\theta) - J_m(\theta) = 2x_{m-1} (\theta - \theta_0) \left(x_m - \frac{x_{m-1}(\theta + \theta_0)}{2} \right). \quad (4)$$

Для модификации процедуры используем метод, аналогичный тому, который применяется в обобщенном методе кумулятивных сумм (Generalized CUSUM) [5]. Основная идея этого метода заключается в максимизации статистик логарифмического правдоподобия по неизвестному параметру. Используем аналогичный подход для статистик (4). Модифицированные статистики примут вид

$$\Lambda_n^{k+1} = \sup_{\theta \in \Theta} \sum_{i=k+1}^n \Delta J_i(\theta).$$

Модифицированная процедура кумулятивных сумм имеет вид

$$N^* = \inf \left(n \geq 1 : \max_{1 \leq k \leq n} \Lambda_n^k \geq C \right), \quad (5)$$

где C – порог.

Данная модификация является привлекательной с практической точки зрения, поскольку при её использовании требуется только знание функционального вида модели, используемой для аппроксимации случайного процесса, и знание значения параметра модели до разладки.

В следующем разделе рассмотрим реализацию предложенной процедуры.

3. Численное моделирование

Рассмотрим процесс AR(1) вида

$$x_n = 0.4x_{n-1} + \varepsilon_n, \quad n = 1, 2, \dots, v-1,$$

$$x_n = -0.5x_{n-1} + \varepsilon_n, \quad n \geq v,$$

где $v=100$, $\varepsilon_n \sim N(0,1)$, т.е. в качестве функции распределения шума выступает нормальное распределение с нулевым средним и единичной дисперсией.

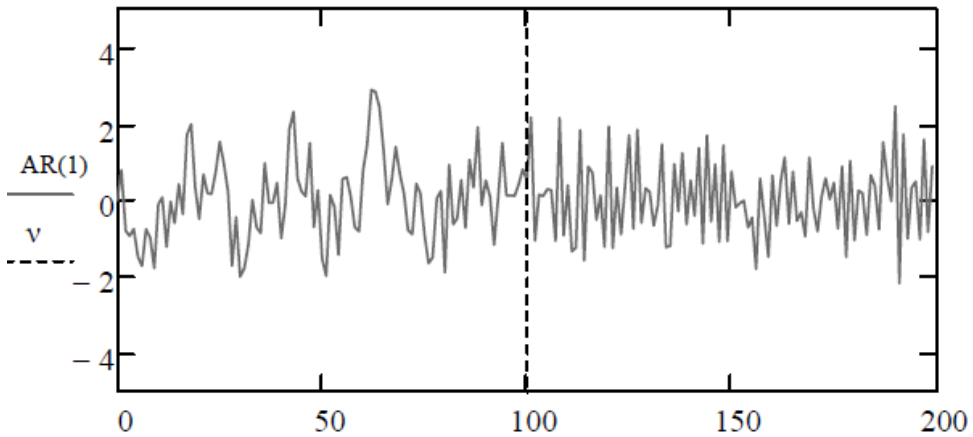


Рис. 1. Процесс AR(1) с разладкой ($\theta_0 = 0.4$, $\theta_1 = -0.5$, $v = 100$)

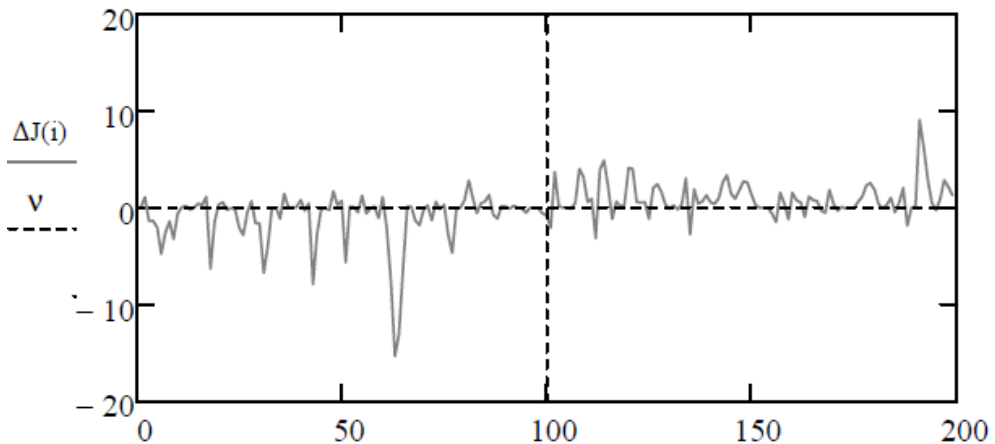


Рис. 2. Последовательность статистик $\Delta J(m)$

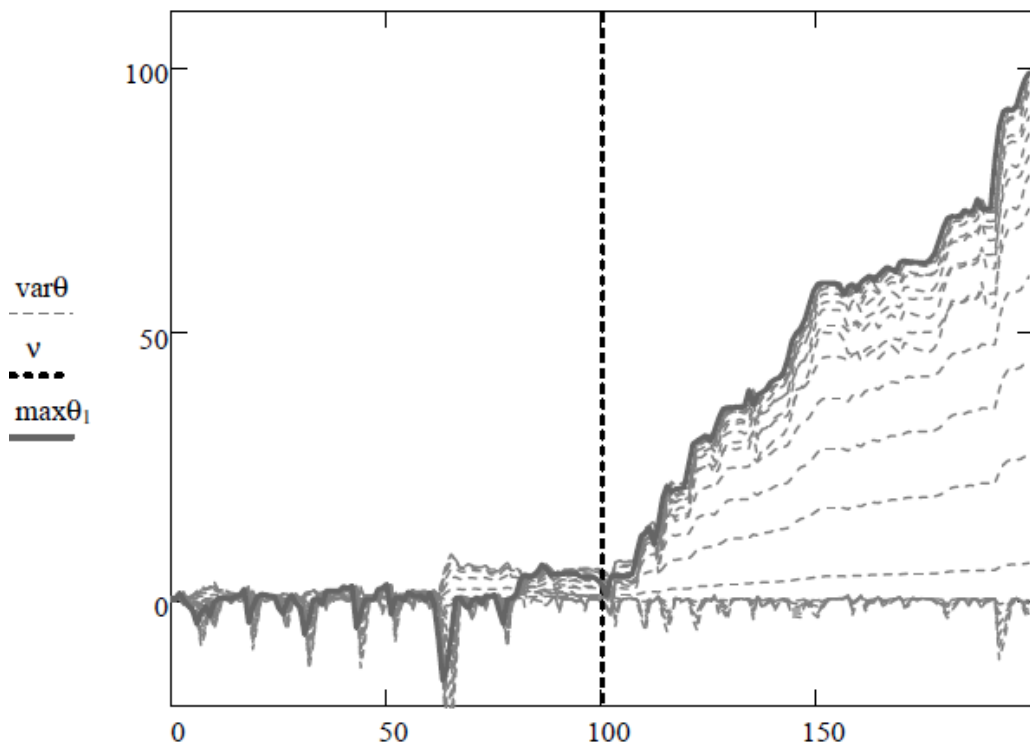


Рис. 3. Реализация процедуры при вариации неизвестного параметра θ

На рис. 1 представлена реализация процесса AR(1) с разладкой в момент времени $v = 100$, в который происходит смена значения параметра процесса. На рис. 2 изображена реализация последовательности статистик (2). Видно, что среднее статистик до момента разладки меньше нуля, а после разладки – больше нуля. На рис. 3 изображена реализация процедуры (5). В частности, сплошной темной линией ($\max \theta_i$) изображена последовательность максимумов сумм статистик до момента n , при условии, что известно значение параметра процесса после момента отклонения. Последовательность представлена статистиками вида

$$\left\{ \max_{1 \leq k \leq n} \sum_{i=k}^n \Delta J(i) \right\}_{n \geq 1}.$$

Тонкими пунктирными линиями ($\text{var } \theta$) изображены последовательности максимумов сумм статистик до момента n при вариации неизвестного параметра, т.е.

$$\left\{ \max_{1 \leq k \leq n} \sum_{i=k}^n \Delta J_i(\theta) \right\}_{n \geq 1}, \quad \theta \in \{-1, -1 + \Delta x, \dots, 1\},$$

где Δx – шаг дискретизации. В данной реализации $\Delta x = 0.1$. Иными словами, производится параллельный запуск непараметрических процедур вида (3) для всех предполагаемых значений неизвестного параметра θ и на каждом шаге n максимизируется значение по всем полученным реализациям; т.е. результирующая последовательность представляет из себя верхнюю границу всех реализаций при вариации неизвестного параметра. Видно, что последовательность $\max \theta_i$ (реализация процедуры при известном значении параметра после разладки) приближается к верхней границе реализаций процедуры при вариации неизвестного параметра. Можно сделать предварительный вывод, что предложенная процедура по эффективности близка к процедуре, рассмотренной в [4].

Частотные характеристики процедуры

C	δFA	ADD	C	δFA	ADD
10	0,13	12,264	20	0	22,28
15	0,01	16,202	25	0	26,84

где C – пороговое значение процедуры, δFA – доля ложных тревог (алгоритм сигнализирует о разладке, когда её нет) в последовательности реализаций процедуры (количество реализаций $m = 100$), ADD – среднее время запаздывания процедуры

$$ADD = \frac{\sum_{t=1}^m (N_t^* - v + 1) \chi_{\{N_t^* \geq v\}}}{\sum_{t=1}^m \chi_{\{N_t^* \geq v\}}},$$

где N_t^* – момент остановки процедуры при реализации на данных из набора t , χ – индикаторная функция.

Заключение

В данной статье предложена модификация непараметрической процедуры CUSUM, позволяющая отслеживать разладку при неизвестном параметре процесса авторегрессии первого порядка AR(1) после момента отклонения. Представлена реализация процедуры при гауссовости шума и приведены частотные характеристики, демонстрирующие работоспособность алгоритма.

ЛИТЕРАТУРА

1. Page E.S. Continuous inspection scheme // *Biometrika*. – 1954. – Vol. 41. – P. 100–115.
2. Lorden G. Procedures for reacting to a change in distribution // *The annals of mathematical statistics*. – 1971. – Vol. 42. – № 6. – P. 1897–1908.
3. Lai T.L. Information bounds and quick detection of parameter changes in stochastic system // *IEEE transaction on information theory*. – 1998. – Vol. 44. – № 7. – P. 2917–2929.
4. Konev V., Vorobeychikov S. Quickest detection of parameter changes in stochastic regression: nonparametric CUSUM // *IEEE transaction on information theory*. – 2017. – Vol. 63. – № 9 – P. 5588–5602.
5. Tartakovsky A., Nikiforov I., Basseville M. *Sequential analysis: hypothesis testing and changpoint detection*. – Boca Raton: A CHAPMAN & HALL BOOK, 2015. – 584 p.

ОБРАБОТКА МЕДИЦИНСКИХ ДАННЫХ, СОДЕРЖАЩИХ ИНФОРМАЦИЮ О ФАКТОРАХ СЕРДЕЧНО- СОСУДИСТЫХ ЗАБОЛЕВАНИЙ

Т.Е. Малахова¹, Т.В. Кабанова¹, А.А. Горлова²

¹Томский государственный университет

²Научно-исследовательский институт кардиологии

malahova.tanny@yandex.ru, tvk@bk.ru

Введение

В последние годы очень востребованными являются статистические методы обработки медицинских данных, поскольку математическая статистика разрабатывает методы статистической обработки и анализа данных, занимается обоснованием и проверкой их достоверности, эффективности, условий применения, устойчивости к нарушению условий применения и т.п. [1,2].

В данной работе изучаются факторы эмболического инсульта, характеризующегося закупоркой сосудов головного мозга или других внутренних органов тромбом либо эмболом, который формируется в сердце, а при определённых условиях, с током крови попадает в сосуды.

1. Первичная обработка данных

Имеется информация о 335 пациентах, у которых наблюдался один из 4 видов инфарктов, а именно – острое нарушение мозгового кровообращения, инфаркт тонкого кишечника, инфаркт селезёнки и инфаркт почки. Также было 12 факторов, представляющих признаки и анализы пациента (пол, возраст, масса сердца и т.д.).

Поскольку многие факторы содержали пропущенные наблюдения, ошибки и опечатки, был применён метод удаления пропусков, т.к. другие методы в силу специфичности предметной области могли отрицательно сказаться на корректности результатов.

Большое количество статистических методов исходит из предположения нормальности распределения изучаемых данных, поэтому первым этапом при анализе данных является проверка на нормальность [3]. Применение критерия Шапиро – Уилка к имеющимся величинам показало, что ни один признак не удовлетворяет нормальному закону.

2. Сравнение групп

Проверка статистических гипотез о равенстве средних в группах пациентов, перенесших и не перенесших инфаркт, проводилась с помощью непараметрического критерия Манна – Уитни, т.к. вид распределения выборки неизвестен и наблюдения независимы.

В табл. 1 приводятся факторы со статистически значимыми различиями в средних, выделенные в процессе данного анализа.

Таблица 1

Название заболевания	Факторы со статистически значимыми различиями	p-level	Нормальная статистика Z
ОНМК	толщина пжк	0,005843	2,739428
	масса сердца	0,000009	4,427880
Инфаркт селезёнки	возраст	0,000582	3,439018
	масса тела	0,022163	-2,28556
	масса сердца	0,001489	-3,17560
Инфаркт почки	возраст	0,000002	4,747452
	масса сердца	0,019152	-2,34171
Инфаркт тонкого кишечника	возраст	0,022942	-2,27368

На рис. 1, 2 приводятся диаграммы размаха по некоторым факторам со статистически значимыми отличиями, приведенным в табл. 1, для большей наглядности.

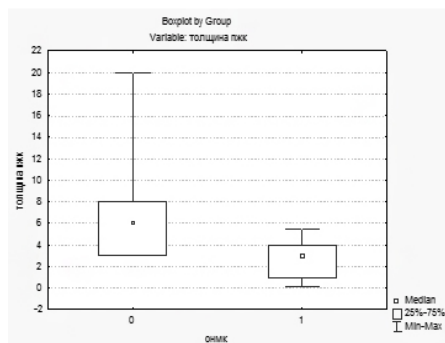


Рис. 1. Диаграмма размаха фактора «толщина пжк» по группам пациентов с ОНМК и без

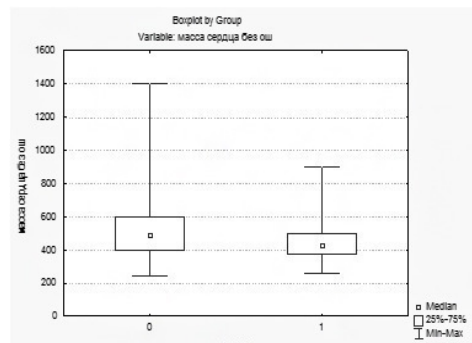


Рис. 2. Диаграмма размаха фактора «масса сердца» по группам пациентов с ОНМК и без

Для проверки статистически значимой корреляционной связи между признаками, представленными качественными переменными, были использованы таблицы сопря-

жённости. Данный анализ показал, что статистически значимой связи между ожирением и видами инфарктов в данном случае нет.

3. Регрессионный анализ

Регрессионный анализ – это один из самых распространённых методов обработки данных, часто являющийся основой таких методов, как многомерный статистический анализ, планирование эксперимента, дисперсионный анализ и др. [4,5].

Однако линейная регрессия не всегда может качественно предсказывать значения зависимой переменной. Например, в случае, когда зависимая переменная принимает значения только 0 и 1, использовать множественную регрессию не предоставляется возможным, поскольку она не накладывает ограничения на значения зависимой переменной, и, таким образом, предсказанные значения окажутся выходящими за диапазон значений зависимой переменной, что невозможно интерпретировать в рамках поставленной задачи.

Логистическая регрессия – это разновидность множественной регрессии, которая применяется для решения задач предсказания зависимой переменной Y , принимающей значения на интервале $[0; 1]$. Также её можно использовать для решения задач с бинарным откликом, когда зависимая переменная может принимать только два значения. При этом используется следующее уравнение регрессии:

$$y = \frac{\exp\{\beta_0 + \beta_1 x_1 + \dots + \beta_n x_n\}}{1 + \exp\{\beta_0 + \beta_1 x_1 + \dots + \beta_n x_n\}},$$

которое можно преобразовать к модели множественной регрессии с помощью логистического преобразования [6].

Поскольку в задаче несколько зависимых переменных, для каждой из них строилось отдельное уравнение логистической регрессии. Рассмотрим данный анализ на примере одной из них, а именно инфаркте селезёнки.

На рис. 3 представлен результат логистической регрессии.

Модель: Логистическая регрессия Число 0: 131 1: 65 (Нормированные данные)									
Зав. пер.: инф селезенки Потери: Максимум правдоподобия (Масштаб С									
Итоговые потери: 113,92071770 $\chi^2(8)=21,207$ $p=.00663$									
N=196	B0	Возраст	Рост	Масса тела	ИМТ	ожир	масса сердца	толщ лж	толщ пж
Оценка	-0,74674	-0,38211	0,39057	-1,1963	1,337613E+	-0,020544	0,50766	-0,28755	-0,02046
Станд. ошибка	0,1622	0,17216	0,56339	1,2009	1,193969E+	0,26078	0,19600	0,17437	0,18331
t(187)	-4,6016	-2,2194	0,69324	-0,99613	1,120308E+	-0,078775	2,5900	-1,6490	-0,11162
p-знач.	0,00000771E	0,02765E	0,48901	0,32047	2,640193E	0,93729	0,010353	0,10082	0,91124
-95%CL	-1,0668	-0,7217E	-0,7208	-3,5654	-1,017766E+	-0,53499	0,12099	-0,6315	-0,38209
+95%CL	-0,42661	-0,04248	1,5020	1,1728	3,692992E+	0,49390	0,89433	0,05644E	0,34116
Хи-квадрат Вальде	21,175	4,9260	0,48059	0,99229	1,255090E+	0,006206	6,7081	2,7192	0,012459
p-знач.	0,00000420E	0,026461	0,4881	0,31919	2,625909E	0,93720	0,009601E	0,09915	0,91112
Отн.Шансов(ед. из	0,47390	0,6824	1,4778	0,30230	3,809937E+	0,97966	1,661	0,75009	0,97974
-95%CL	0,34408	0,48590	0,48633	0,028284	3,614012E	0,58567	1,1286	0,5317	0,68243
+95%CL	0,65271	0,95840	4,4906	3,2311	4,016484E+	1,6387	2,4456	1,058	1,406
Отн.Шансов(разме	0,16493	6,7126	0,0001014	5,141056E+	0,9566	37,173	0,14490	0,84081	
-95%CL	0,03324	0,02977E	0,0000000000012	2,602803E	0,31057	2,3672	0,01437	0,039251	

Рис. 3. Результат логистической регрессии

В этом случае гипотеза о незначимости модели отклоняется, поскольку значение статистики $\chi^2 = 21,207$ при уровне значимости $p = 0,00663$.

Из рис. 3 видно, что модель содержит незначимые факторы, поэтому имеет смысл удалить их из анализа. Однако предварительно необходимо проверить наличие корреляционной связи между зависимыми факторами.

Корреляционный анализ – статистический метод, позволяющий проверить наличие статистически значимой связи между двумя и более переменными. Для проверки корреляционной связи использовался ранговый коэффициент корреляции Спирмена, яв-

ляющийся непараметрическим методом и не требующий нормального распределения сопоставляемых переменных [7].

На рис. 4 представлена таблица ранговых корреляций Спирмена.

Ранговые корреляции Спирмена (Нормированные данные) ПД попарно удалены Отмеченные корреляции значимы на уровне $p < 0,05000$								
Перем.	Возраст	Рост	Масса тела	ИМТ	ожир	масса сердца	толщ лж	толщ пж
Возраст	1,000000	-0,321856	-0,144562	0,014494	-0,025000	-0,114973	0,142332	-0,192426
Рост	-0,321856	1,000000	0,418978	-0,015855	-0,119879	0,230669	0,017502	0,120070
Масса тела	-0,144562	0,418978	1,000000	0,870042	0,663009	0,371767	0,179483	-0,053673
ИМТ	0,014494	-0,015855	0,870042	1,000000	0,790970	0,265661	0,207838	-0,096979
ожир	-0,025000	-0,119879	0,663009	0,790970	1,000000	0,156046	0,130110	-0,125173
масса сердца	-0,114973	0,230669	0,371767	0,265661	0,156046	1,000000	0,161406	0,237186
толщ лж	0,142332	0,017502	0,179483	0,207838	0,130110	0,161406	1,000000	0,326402
толщ пж	-0,192426	0,120070	-0,053673	-0,096979	-0,125173	0,237186	0,326402	1,000000

Рис. 4. Ранговые корреляции Спирмена

Из рис. 4 видно, что статистически значимой связи между значимыми факторами логистической регрессии – возрастом и массой сердца – нет. Следовательно, можно провести регрессионный анализ, включая лишь значимые факторы в модель.

На рис. 5 представлен результат логистической регрессии после удаления незначимых факторов из модели.

Модель: Логистическая регрессия Число 0: 131 1: 65 (Нормированные данн Зав. пер.: инф селезенки Потери: Максимум правдоподобия (Масштаб С Итоговые потери: 117,10280372 $\chi^2(2)=14,843$ $p=,00060$			
N=196	ВО	Возраст	масса сердца
Оценка	-0,739950	-0,354799	0,466945
Станд. ошибка	0,158906	0,156418	0,170883
t(193)	-4,65653	-2,26826	2,73253
p-знач.	0,00000597333	0,0244200	0,00686841
-95%CL	-1,05336	-0,663308	0,129906
+95%CL	-0,426535	-0,0462899	0,803984
Хи-квадрат Вальда	21,6832	5,1450	7,46677
p-знач.	0,0000032287	0,0233195	0,00628826
Отн.Шансов(ед. изм.)	0,477137	0,701314	1,59511
-95%CL	0,348761	0,515144	1,13872
+95%CL	0,652766	0,954765	2,23442
Отн.Шансов(размах)		0,187619	27,8165
-95%CL		0,0437893	2,52239

Рис. 5. Результат логистической регрессии после удаления незначимых факторов

В этом случае, как и в предыдущем, гипотеза о незначимости модели отклоняется, поскольку значение статистики $\chi^2 = 14,843$ при уровне значимости $p = 0,0006$. Таким образом, уравнение логистической регрессии примет вид:

$$y = \frac{\exp\{-0,7399508 - 0,3547991x_1 + 0,4669458x_2\}}{1 + \exp\{-0,7399508 - 0,3547991x_1 + 0,4669458x_2\}}$$

где x_1 – возраст, x_2 – масса сердца.

На рис. 6 представлен график полученной модели.

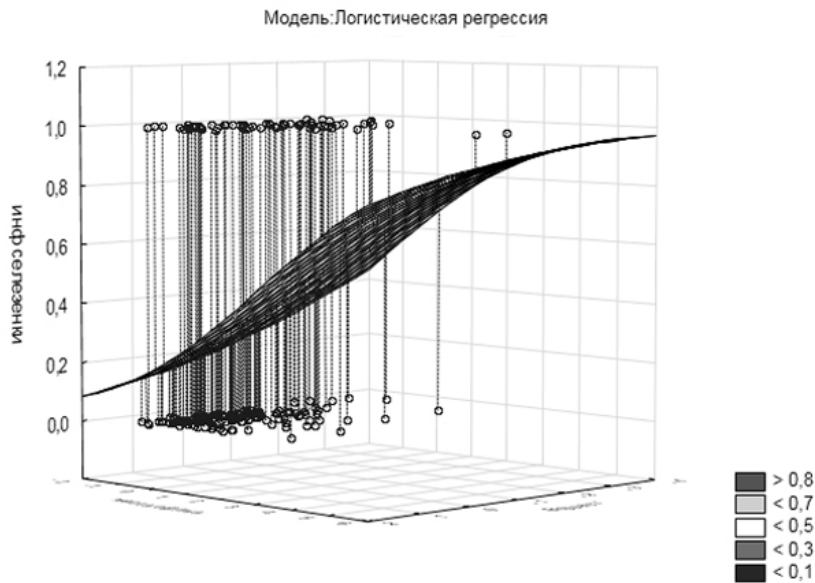


Рис. 6. График логистической регрессии

Заключение

В данной работе были рассмотрены методы предварительной обработки и анализа медицинских данных, проведён корреляционный анализ и построены модели логистической регрессии по данным, которые были предоставлены научно-исследовательским институтом кардиологии.

ЛИТЕРАТУРА

1. Медик В.А., Токмачев М.С. Математическая статистика в медицине. М.: Финансы и статистика, 2007. – 800 с.
2. Кочетов А.Г., Лянз О.В., Масенко В.П. [и др.]. Метод статистической обработки медицинских данных. – М.: РКНПК, 2012. – 42 с.
3. Плющ О.Б., Северин Г.М., Безъязычная Т.В. Информационные технологии анализа и обработки данных. – Мн.: Акад. упр. при Президенте Респ. Беларусь, 2010. – 112 с.
4. Магнус Я.Р., Катышев П.К., Пересецкий А.А. Эконометрика. Начальный курс. М.: Дело, 2005. – 504 с.
5. Айвазян С.А. Методы эконометрики: учебник. – М.: Магистр: ИНФРА-М, 2010. – 512 с.
6. Портал знаний: Логистическая регрессия [Электронный ресурс]. URL: <http://statistica.ru/theory/logisticheskaya-regressiya/> (дата обращения: 30.07.2018).
7. Медицинская статистика [Электронный ресурс]. URL: <http://medstatistic.ru/index.php/> (дата обращения: 30.07.2018).

СЕКЦИЯ V. МАТЕМАТИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ СОЦИАЛЬНО-ЭКОНОМИЧЕСКОЙ ДЕЯТЕЛЬНОСТИ

РОЗНИЧНАЯ ПРОДАЖА ПРОДУКЦИИ С ОГРАНИЧЕННЫМ СРОКОМ ГОДНОСТИ ПРИ ММР-ПОТОКЕ МОМЕНТОВ ПРОДАЖ

К.И. Лившиц, Т.А. Поволоцкая, Е.С. Ульянова
Томский государственный университет
kim47@mail.ru, tpovolt@yandex.ru, katerina_tomsk@sibmail.com

Введение

Одной из классических задач теории управления запасами является проблема управления запасами, имеющими ограниченный срок годности (the newsvendor problem), которая интенсивно изучается в последнее время [1,2]. Первоначально исследовалась задача реализации единственного товара, имеющего фиксированную цену, затем аналогичные задачи начали рассматриваться для много продуктовых моделей [3,4]. Целью данной работы является определение оптимального размера однородной партии товара, имеющего ограниченный срок реализации, в предположении, что моменты продажи товара образуют ММР-поток с двумя состояниями. Такое предположение представляется довольно естественным, так как интенсивность потока покупателей в любой торговой точке изменяется с течением времени случайным образом.

1. Постановка задачи

Продавец приобретает партию товара размера ξ по оптовой цене d и перепродает её по розничной цене c . Считается, что время реализации ограничено. По истечении времени T товар не может быть реализован, а продавец несёт дополнительные затраты b , связанные с утилизацией непроданной части товара. Необходимо определить оптимальный размер партии товара ξ , который обеспечивает продавцу максимальную среднюю прибыль.

В работе предполагается, что моменты покупок образуют ММР-поток с интенсивностью $\lambda(t)$. Интенсивность $\lambda(t)$ является однородной цепью Маркова с непрерывным временем и двумя состояниями $\lambda(t) = \lambda_i$, $i = 1, 2$. Переход из состояния в состояние задаётся матрицей инфинитезимальных характеристик $Q = [q_{ij}]$, где $q_{ij} < 0$ при $i \neq j$ и $q_{j1} + q_{j2} = 1$. Таким образом, переход из состояния i в состояние j за малое время Δt имеет вероятность $P_{ij} = q_{ij}\Delta t + o(\Delta t)$, $i \neq j$, $P_{ii} = 1 + q_{ii}\Delta t + o(\Delta t)$, $i = j$.

Финальные вероятности состояний λ_j будут равны соответственно

$$\pi_1 = \frac{q_{22}}{q_{11} + q_{22}}, \quad \pi_2 = \frac{q_{11}}{q_{11} + q_{22}}.$$

Далее предполагается, что покупатели покупают товар независимо друг от друга. Объем покупки – случайная величина z с плотностью распределения $\psi(z)$, средним значением $M\{z\} = a_1$ и вторым моментом $M\{z^2\} = a_2$.

2. Асимптотическая плотность вероятностей спроса

Пусть $S(t)$ – количество товара, проданного к моменту времени t , $\lambda(t)$ – значение интенсивности потока покупателей в момент времени t . Обозначим

$$P(S, k, t) ds = P\{S < S(t) \leq S + ds, \lambda(t) = \lambda_k\}.$$

Пусть в момент времени t $S(t) = S$, $\lambda(t) = \lambda_k$. Пусть прошло время Δt . За время Δt могли произойти следующие события:

1. Покупки не производились, интенсивность потока покупателей не изменилась. Вероятность этого события равна $(1 + q_{kk}\Delta t)(1 - \lambda_k\Delta t) + o(\Delta t)$.
2. Произведена покупка величины x , интенсивность потока покупателей не изменилась. Вероятность этого события равна $(1 + q_{kk}\Delta t)\lambda_k\Delta t\psi(x)dx + o(\Delta t)$.
3. Покупки не происходили, интенсивность потока покупателей изменилась с λ_k на λ_j . Вероятность этого события равна $(1 - \lambda_k\Delta t)q_{kj}\Delta t$.

Остальные события имеют вероятность $o(\Delta t)$.

Используя формулу полной вероятности, получим

$$P(S, k, t + \Delta t) = (1 - \lambda_k\Delta t + q_{kk}\Delta t)P(S, k, t) + \sum_{j \neq k} q_{jk}P(S, j, t) + \lambda_k\Delta t \int_0^S P(S - x, k, t)\psi(x)dx.$$

Переходя к пределу при $\Delta t \rightarrow 0$, будем иметь

$$\frac{\partial P(S, k, t)}{\partial t} = -\lambda_k P(S, k, t) + \sum_{j=1}^2 q_{jk} P(S, j, t) + \lambda_k \int_0^S P(S - x, k, t)\psi(x)dx. \quad (1)$$

Если начальные вероятности состояний λ_k совпадают с финальными вероятностями π_k , то имеют место начальные условия

$$P(S, k, 0) = \pi_k \delta(S). \quad (2)$$

Плотность распределения спроса к моменту времени t $P(S, t) = \sum_{k=1}^2 P(S, k, t)$.

Обозначим $g_k(w, t) = \int_0^\infty e^{-wS} P(S, k, t) dS$ – частичные производящие функции процесса $S(t)$. Производящая функция процесса $S(t)$ будет, очевидно, равна $g(w, t) = g_1(w, t) + g_2(w, t)$.

Применяя преобразование Лапласа к уравнениям (1), получим, что функции $g_k(w, t)$ удовлетворяют системе уравнений

$$\frac{\partial g_k(w, t)}{\partial t} = -\lambda_k g_k(w, t) + \sum_{j=1}^2 q_{jk} g_j(w, t) + \lambda_k g_k(w, t) Y(w), \quad (3)$$

где $Y(w) = \int_0^\infty \psi(S) e^{-wS} dS$ и $g_k(0, t) = \int_0^\infty P(S, k, t) dS = P\{\lambda(t) = \lambda_k\} = \pi_k$.

Обозначим $\mu_{j,k}(t) = M\{S(t)^k | \lambda(t) = \lambda_j\}$, $k = 1, 2$ – частичные моменты процесса $S(t)$ первого и второго порядка. Дважды дифференцируя соотношения (3) по w и положив $w = 0$, получим системы уравнений на моменты $\mu_{kj}(t)$

$$\dot{\mu}_{n,1}(t) = \lambda_n \pi_n a_1 + \sum_{i=1}^2 q_{in} \mu_{i,1}(t), \quad \mu_{n,1}(0) = 0, \quad (4)$$

$$\dot{\mu}_{n,2}(t) = \lambda_n \pi_n a_2 + 2\lambda_n a_1 \mu_{n,1}(t) + \sum_{i=1}^2 q_{in} \mu_{i,2}(t), \quad \mu_{n,2}(0) = 0. \quad (5)$$

Безусловные моменты процесса $S(t)$ будут равны $\mu_i(t) = \sum_{n=1}^2 \mu_{n,i}(t)$. Решая систему

уравнений (4), получим
$$\mu_{11}(t) = \lambda_0 \pi_1 a_1 t + \frac{(\lambda_0 - \lambda_1) \pi_1 a_1}{q_{11} + q_{22}} (1 - e^{-(q_{11} + q_{22})t}),$$

$$\mu_{21}(t) = \lambda_0 \pi_2 a_1 t + \frac{(\lambda_0 - \lambda_2) \pi_2 a_1}{q_{11} + q_{22}} (1 - e^{-(q_{11} + q_{22})t}),$$
 откуда

$$\mu_1(t) = \lambda_0 a_1 t, \quad (6)$$

где $\lambda_0 = \lambda_1 \pi_1 + \lambda_2 \pi_2$.

С учётом свойств матрицы $[q_{ij}]$, складывая уравнения (5), будем иметь

$$\dot{\mu}_2(t) = \lambda_0 a_2 + 2a_1 (\lambda_1 \mu_{11}(t) + \lambda_2 \mu_{21}(t)). \quad (7)$$

Проинтегрировав уравнение (7), получим

$$\mu_2(t) = \lambda_0 a_2 t + \lambda_0^2 a_1^2 t^2 - 2 \frac{a_1^2 \sigma_\lambda^2}{q_{11} + q_{22}} t + 2 \frac{a_1^2 \sigma_\lambda^2}{(q_{11} + q_{22})^2} (e^{-(q_{11} + q_{22})t} - 1), \quad (8)$$

где $\sigma_\lambda^2 = \sum_{i=1}^2 (\lambda_i - \lambda_0)^2 \pi_i$.

Обозначим $V_k(w, u) = \int_0^\infty e^{-ut} g_k(w, t) dt$. Применяя к уравнению (3) преобразование

Лапласа по t , получим, учитывая (2),

$$\sum_{j=1}^2 (q_{jk} - (u + \lambda_k w Y(w) \delta_{ik})) V_i(w, u) = -\pi_k, \quad k = 1, 2, \quad (9)$$

откуда функция $V(w, u) = V_1(w, u) + V_2(w, u) = \int_0^\infty g(w, t) e^{-ut} dt$ будет равна

$$V(w, u) = -\frac{q_{11} + q_{22} - u - (\lambda_2 \pi_1 + \lambda_1 \pi_2) w Y(w)}{\det(w, u)},$$

где определитель матрицы системы уравнений (9) $[q_{jk} - (u + \lambda_k w Y(w) \delta_{ik})]$

$$\det(w, u) = u^2 - u(q_{11} + q_{22} - (\lambda_1 + \lambda_2) w Y(w)) - \lambda_0 (q_{11} + q_{22}) w Y(w) + \lambda_1 \lambda_2 w^2 Y(w)^2.$$

Функция $g(w, t)$ определится как обратное преобразование Лапласа

$$g(w, t) = \frac{1}{2\pi j} \int_{\sigma - j\infty}^{\sigma + j\infty} V(w, u) e^{ut} du.$$

Уравнение $\det(w, u) = 0$ имеет корни

$$u_{1,2}(w) = \frac{1}{2} \left(q_{11} + q_{22} - (\lambda_1 + \lambda_2) w Y(w) \pm \sqrt{(\lambda_1 - \lambda_2) w Y(w) + (q_{22} - q_{11})^2 + 4q_{11}q_{22}} \right), \quad (10)$$

поэтому по теореме о вычетах [5] $g(w, t) = A_1(w) e^{u_1(w)t} + A_2(w) e^{u_2(w)t}$, где

$$A_1(w) = -\frac{q_{11} + q_{22} - u_1(w) - (\lambda_2 \pi_1 + \lambda_1 \pi_2) w Y(w)}{u_1(w) - u_2(w)},$$

$$A_2(w) = \frac{q_{11} + q_{22} - u_2(w) - (\lambda_2 \pi_1 + \lambda_1 \pi_2) w Y(w)}{u_1(w) - u_2(w)}.$$

Рассмотрим поведение функций $u_i(w)$ и $A_i(w)$ при $w \ll 1$. Раскладывая выражения (10) в ряд по степеням w , можно показать, что при $w \ll 1$

$$u_1(w) = -\lambda_0 a_1 w + \left(\lambda_0 a_2 - \frac{2\sigma_\lambda^2 a_1^2}{q_{11} + q_{22}} \right) \frac{w^2}{2} + o(w^2),$$

$$u_2(w) = (q_{11} + q_{22}) - (\lambda_1 \pi_2 + \lambda_2 \pi_1) a_1 w + \left((\lambda_1 \pi_2 + \lambda_2 \pi_1) a_2 + \frac{2\sigma_\lambda^2 a_1^2}{q_{11} + q_{22}} \right) \frac{w^2}{2} + o(w^2).$$

Так как $Y(0) = 1$, то $A_1(0) = 1$ и $A_2(0) = 0$.

Введём случайную величину $z = \frac{S(t) - \mu_1(t)}{\sigma(t)}$, где

$$\sigma^2(t) = \mu_2(t) - \mu_1^2(t) \quad (11)$$

и величины $\mu_1(t)$ и $\mu_2(t)$ определяются соотношениями (6) и (8) соответственно. Её производящая функция

$$g_z(w, t) = e^{\frac{w \mu_1(t)}{\sigma(t)}} g\left(\frac{w}{\sigma(t)}, t\right). \quad (12)$$

Подставляя в (12) все необходимые выражения, заменив w на jw и переходя к пределу при $t \rightarrow \infty$, получим, что $\lim_{t \rightarrow \infty} g_z(jw, t) = e^{-\frac{w^2}{2}}$.

Таким образом, случайная величина z при $t \gg 1$ имеет асимптотически стандартное нормальное распределение и, следовательно, при $t \gg 1$ количество товара $S(t)$, проданного к моменту времени t , имеет асимптотически нормальное распределение со средним значением $\mu_1(t)$ (6) и дисперсией $\sigma^2(t)$ (11).

Пример. Для оценки скорости сходимости рассмотрим случай, когда $\psi(S) = \frac{1}{a} e^{-\frac{S}{a}}$.

На рис. 1 приведены зависимости $f(w) = \ln g_z(jw, t)$, вычисленные по формуле (12), при $t = 50$ и $t = 500$. Параметры $q_{11} = -1$, $q_{22} = -0.5$, $\lambda_1 = 1.1$, $\lambda_2 = 0.6$, $a = 2$. Как следует из приведенных графиков, с ростом t наблюдается сходимость к нормальному распределению.

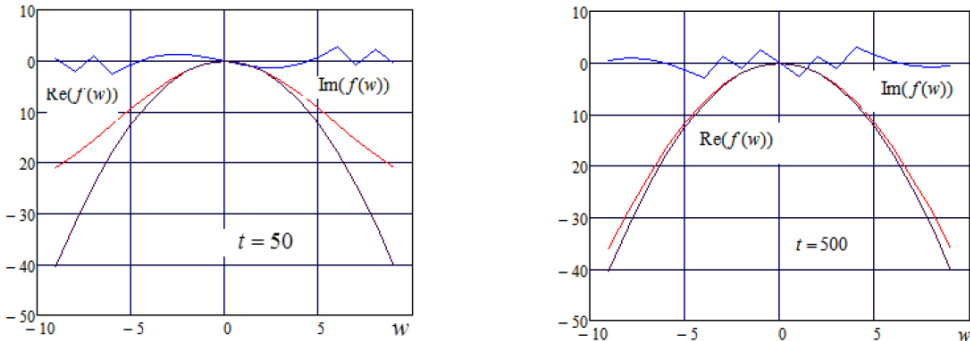


Рис. 1. Зависимость $f(w) = \ln g_z(jw, t)$ от длительности t торговой сессии

3. Определение средней прибыли

Обозначим через W среднюю прибыль продавца за время торговой сессии T . Так как на приобретение ξ единиц товара затрачивается $d\xi$ средств, а выручка составляет $c\xi$, если была продана вся партия товара, и $cx - b(\xi - x)$, если была продана партия товара размера x , то средняя прибыль

$$W = -d\xi + c\xi \int_{\xi}^{\infty} P(x, T) dx + \int_0^{\xi} (cx - b(\xi - x))P(x, T) dx, \quad (13)$$

где $P(x, T)$ – плотность распределения спроса на товар за время T .

При фиксированных ценах c и d и платах за утилизацию b можно определить оптимальный объём партии товара ξ , максимизирующий среднюю прибыль. Дифференцируя (13) по ξ , получим

$$\int_0^{\xi} P(S, T) dS = 1 - \frac{b+d}{b+c}. \quad (14)$$

Учитывая, что $d < c$, а интеграл монотонно возрастает по ξ в интервале от 0 до 1, получаем, что уравнение (14) имеет единственное решение. Если $\chi(z)$ – функция, обратная $\int_0^z P(S, T) dS$, то $\xi_{\text{оп}} = \chi\left(1 - \frac{b+d}{b+c}\right)$.

При $T \gg 1$ распределение $P(S, T)$ может быть аппроксимировано нормальным распределением. Поэтому при $T \gg 1$ оптимальное значение партии товара $\xi_{\text{оп}}$ определяется соотношением $\xi_{\text{оп}} = \mu_1(T) + \sigma(T)X\left(1 - \frac{b+d}{b+c}\right)$, где $\mu_1(T)$ и $\sigma(T)$ определяются (6) и (11), а $X(\cdot)$ – функция, обратная к функции стандартного нормального распределения.

Заключение

В работе рассмотрена задача розничной продажи продукции, имеющей ограниченный срок годности при условии, что моменты реализации продукции образуют ММР-поток с двумя состояниями. Найдена асимптотическая плотность распределения спроса на продукцию, определено оптимальное значение количества товара, предназначенного для перепродажи. Отметим, что тем же путём могут быть исследованы и более сложные модели, например, случай продажи многономенклатурной партии товара.

ЛИТЕРАТУРА

1. *Petruzzi N.C., Dada M.* Pricing and the newsvendor problem: A review with extensions // *Operations Research*. – 1999. – V. 47 (2). – P. 183–194.
2. *Chen X., Simchi-Levi D.* Joint Pricing and Inventory Management. // *The Oxford handbook of pricing management* – London: Oxford University Press. – 2012.
3. *Turken N., Tan Y., Vakharia A.J., Wang L. et al.* The Multi-Product Newsvendor Problem: Review, Extension, and Directions for Future Research // *Handbook of Newsvendor Problems: Models, Extensions and Applications*. – 2012. – V. 176. – P. 3–39.
4. *Kitaeva A., Livshits K., Ulyanova E.* The multiproduct Newboy Problem with Price- Depended Demand and Fast Moving Items // *Information technologies and mathematical modeling: Proceeding of the 16 th International Scientific Conference, ITMM 2017 named after A.F. Terpugov*. – CCIS. – V. 800. – Springer, 2017. – P. 297–311.
5. *Лаврентьев М.А., Шабат Б.В.* Методы теории функций комплексного переменного. – М.: Наука, 1973. – 736 с.

ПРОСТАЯ АППРОКСИМАЦИЯ ВЕРОЯТНОСТИ РАЗОРЕНИЯ СТРАХОВОЙ КОМПАНИИ ДЛЯ МОДЕЛИ КРАМЕРА – ЛУНДБЕРГА С ММР-ПОТОКОМ СТРАХОВЫХ ВЫПЛАТ

М.Г. Воробьева, К.И. Лившиц
Томский государственный университет
mashe4ka96-96@mail.ru, kim47@mail.ru

Введение

Стандартной задачей актуарной математики является задача вычисления вероятности разорения страховой компании, т.е. вероятности ситуации, когда страховая компания не может исполнять свои финансовые обязательства ввиду отсутствия денежных средств при различных предположениях о потоках, поступающих в компанию страховых премий и страховых выплат, производимых страховой компанией [1–3]. Сложность состоит, как правило, в нахождении явных решений соответствующих интегро-дифференциальных уравнений, определяющих вероятность разорения. Поэтому представляет интерес нахождение оценок для вероятности разорения, для построения которых не нужно решать соответствующие уравнения. В настоящей работе предлагается и исследуется оценка вероятности разорения страховой компании для модели Крамера-Лундберга с ММР-потоком страховых выплат [4–6].

1. Модель Крамера – Лундберга с ММР-потоком страховых выплат

Модель Крамера-Лундберга с ММР-потоком страховых выплат строится при следующих предположениях [4–6]: процесс поступления страховых премий в компанию считается детерминированным, за время t приращение капитала равно Ct , где C – количество средств, поступивших в компанию в единицу времени, страховые выплаты – независимые, одинаково распределенные случайные величины с плотностью распределения $\Psi(x)$ и моментами $M\{x^k\} = a_k$, $k = 1, 2, \dots$, моменты наступления страховых выплат образуют ММР-поток с интенсивностью $\lambda(t)$. Интенсивность $\lambda(t)$ является одно-родной цепью Маркова с непрерывным временем и m состояниями $\lambda(t) = \lambda_i$. Переход из состояния в состояние задаётся матрицей инфинитезимальных характеристик $Q = [q_{ij}]$ ранга $n - 1$. Таким образом, переход из состояния i в состояние j за малое время Δt имеет вероятность

$$P_{ij} = q_{ij}\Delta t + o(\Delta t), \quad i \neq j,$$
$$P_{ii} = 1 + q_{ii}\Delta t + o(\Delta t), \quad i = j,$$

где $q_{ij} > 0$ при $i \neq j$ и $\sum_{i=1}^m q_{ij} = 1$. Если все ненулевые собственные значения матрицы $[q_{ij}]$ имеют отрицательные действительные части, то существуют финальные вероятности π_i состояний λ_i , которые являются решением системы уравнений

$$\sum_{j=1}^m q_{ji}\pi_j = 0, \tag{1}$$
$$\pi_1 + \pi_2 + \dots + \pi_m = 1.$$

Предполагается также, что с начала деятельности страховой компании прошло достаточно длительное время, так что поток страховых выплат не зависит от потока страховых премий.

Пусть $S(t)$ – капитал компании в момент времени t . При сделанных выше предположениях

$$S(t) = S(0) + Ct - y(t), \quad (2)$$

где

$$y(t) = \sum_{j=1}^{n(t)} y_j \quad (3)$$

– процесс исков (страховых выплат), $S(0)$ – начальный капитал, $n(t)$ – число страховых выплат к моменту времени t , y_j – величина j -й страховой выплаты. Можно показать [4,5], что при $t \gg 1$ средний капитал страховой компании $\bar{S}(t) = S(0) + (C - \lambda_0 a_1)$, где $\lambda_0 = \sum_{i=1}^m \lambda_i \pi_i$ – средняя интенсивность потока исков. Таким образом, при $C = (1 + \theta)\lambda_0 a_1$, где $\theta > 0$, капитал компании в среднем растет. При невыполнении этого условия компания разоряется. Параметр θ – нагрузка страховой премии.

2. Уравнения для вероятностей разорения и выживания

Пусть в момент времени t капитал компании равен $S(t)$. Определим $T = \inf\{t: S(t) < 0\}$ и $T = \infty$, если $S(t) \geq 0 \quad \forall t$. Случайная величина T – момент разорения. Тогда $P_i(S) = \Pr\{T < \infty | S(0) = S, \lambda(0) = \lambda_i\}$ – вероятность разорения страховой компании на бесконечном временном интервале при условии, что в начальный момент времени её капитал равен S и значение интенсивности $\lambda(0) = \lambda_i$. По смыслу задачи функции $P_i(S)$ являются непрерывными монотонно возрастающими функциями. Наконец, пусть $P(S) = \sum_{i=1}^m \pi_i P_i(S)$ – вероятность разорения компании в стационарном режиме при условии, что её начальный капитал равен S .

Можно показать, что вероятности разорения $P_i(S)$ определяются системой уравнений [4,5]

$$C\dot{P}(S) = \lambda_i P_i(S) - \sum_{i=1}^m q_{ij} P_j(S) - \lambda_i \int_0^S P_i(S-x) \psi(x) dx - \lambda_i \int_S^\infty \psi(x) dx \quad (4)$$

с граничными условиями $P_i(S) \rightarrow 0$ при $S \rightarrow \infty$. Далее можно показать, что при любом распределении страховых выплат $\psi(S)$ вероятность разорения при нулевом начальном капитале $P(0) = \frac{1}{1 + \theta}$ и, таким образом, зависит только от нагрузки страховой премии.

При произвольном распределении страховых выплат $\psi(S)$ получить точное решение системы уравнений (4) не удастся. Из литературы известно точное решение задачи при экспоненциальном распределении страховых выплат [4] при произвольном m и при гиперэкспоненциальном распределении страховых выплат при $m = 2$ [7].

Известна оценка вероятности разорения $P(S)$ [5]. При $\theta \ll 1$

$$P(S) = e^{-\frac{A_2}{A_1} \theta S} + o(\theta), \quad (5)$$

где $A_1 = \frac{\lambda_0 a_2}{2} - a^2 \sum_{i=1}^{n-1} (\lambda_i - \lambda_0) \pi_i \sum_{j=1}^{n-1} R_{ij} (\lambda_j - \lambda_0)$, $A_2 = \lambda_0 a$ и матрица

$$R = [R_{ij}] = \begin{bmatrix} q_{11} & q_{12} & \dots & q_{1,n-1} \\ q_{21} & q_{22} & \dots & q_{2,n-1} \\ \dots & \dots & \dots & \dots \\ q_{n-1,1} & q_{n-1,2} & \dots & q_{n-1,n-1} \end{bmatrix}^{-1}.$$

Точность оценки (5) тем выше, чем меньше значение параметра θ .

3. Моменты процесса страховых выплат при $m = 2$

Для дальнейшего исследования нам понадобятся моменты процесса страховых выплат $y(t)$ (3). Обозначим $P(n, x, t)dx = P\{n(t) = n, x < y(t) \leq x + dx\}$. Пусть в момент времени t $n(t) = n$ и $y(t) = x$. Пусть прошло время Δt . За время Δt могли произойти следующие события:

1. Страховые выплаты не производились, интенсивность потока не изменилась. Вероятность этого события равна $(1 - \lambda_n \Delta t)(1 + q_{nn} \Delta t) + o(\Delta t)$.

2. Произведена страховая выплата размера x , интенсивность потока страховых выплат не изменилась. Вероятность этого события равна $\lambda_n \Delta t (1 + q_{nn} \Delta t) \psi(x) dx + o(\Delta t)$.

3. Страховые выплаты не производились, интенсивность потока изменилась со значения λ_n на значение λ_k . Вероятность этого события равна $(1 - \lambda_n \Delta t) q_{nk} \Delta t + o(\Delta t)$.

Остальные события имеют вероятность $o(\Delta t)$.

Используя формулу полной вероятности и переходя к пределу при $\Delta t \rightarrow 0$, получим систему уравнений

$$\frac{\partial P(n, x, t)}{\partial t} = -\lambda_n P(n, x, t) + \sum_{i=1}^m q_{in} P(i, x, t) + \lambda_n \int_0^x P(n, x - z, t) \psi(z) dz \quad (6)$$

с начальными условиями $P(n, x, 0) = \pi_n \delta(x)$.

Безусловная плотность распределения процесса страховых выплат $y(t)$:

$$P(x, t) = \sum_{n=1}^m P(n, x, t).$$

Обозначим $Y(\omega) = \int_0^\infty e^{-\omega x} \psi(x) dx$, $g(n, \omega, t) = \int_0^\infty P(n, x, t) e^{-\omega x} dx$. Применяя к уравнениям (6) преобразование Лапласа, получим систему уравнений на условные производящие функции $g(n, \omega, t)$

$$\frac{\partial g(n, \omega, t)}{\partial t} = -\lambda_n g(n, \omega, t) + \sum_{i=1}^m q_{in} g(i, \omega, t) + \lambda_n g(n, \omega, t) Y(\omega). \quad (7)$$

Обозначим $\mu_{n,k}(t) = \int_0^\infty x^k P(n, x, t) dx = (-1)^k \frac{\partial^k g(n, \omega, t)}{\partial \omega^k} \Big|_{\omega=0}$, $k = \overline{1, 3}$, $n = \overline{1, m}$ – частичные моменты процесса страховых выплат. Моменты процесса страховых выплат $\mu_k(t)$ будут, очевидно, равны $\mu_k(t) = \sum_{n=1}^m \mu_{k,n}(t)$.

Трижды дифференцируя соотношения (7) по ω и положив $\omega = 0$, получим системы соотношений на определение частичных моментов $\mu_{n,k}(t)$

$$\dot{\mu}_{n,1}(t) = \lambda_n \pi_n a_1 + \sum_{i=1}^m q_{in} \mu_{i,1}(t), \quad \mu_{n,1}(0) = 0, \quad (8)$$

$$\dot{\mu}_{n,2}(t) = \lambda_n \pi_n a_2 + 2\lambda_n a_1 \mu_{n,1}(t) + \sum_{i=1}^m q_{in} \mu_{i,2}(t), \quad \mu_{n,2}(0) = 0, \quad (9)$$

$$\mu_{n,3}(t) = \lambda_n \pi_n a_3 + 3\lambda_n a_2 \mu_{n,1}(t) + 3\lambda_n a_1 \mu_{n,2}(t) + \sum_{i=1}^m q_{i,n} \mu_{i,3}(t), \quad \mu_{n,i}(0) = 0. \quad (10)$$

Суммируя уравнения системы (8) и учитывая (1), можно сразу получить [8]

$$\mu_1(t) = \lambda_0 a_1 t. \quad (11)$$

Однако для дальнейшего нам понадобится явный вид моментов $\mu_{n,1}(t)$. Решая систему уравнений (8) при $m = 2$, получим

$$\mu_{1,1}(t) = \lambda_0 \pi_1 a_1 t + \frac{(\lambda_0 - \lambda_1) \pi_1 a_1}{q_{11} + q_{22}} \left(1 - e^{(q_{11} + q_{22})t}\right),$$

$$\mu_{2,1}(t) = \lambda_0 \pi_2 a_1 t + \frac{(\lambda_0 - \lambda_2) \pi_2 a_1}{q_{11} + q_{22}} \left(1 - e^{(q_{11} + q_{22})t}\right).$$

Из уравнений (9) с учётом (1) следует, что второй момент $\mu_2(t)$ удовлетворяет уравнению $\dot{\mu}_2(t) = \lambda_0 a_2 + 2a_1(\lambda_1 \mu_{1,1}(t) + \lambda_2 \mu_{2,1}(t))$, интегрируя которое, получим, что

$$\mu_2(t) = \lambda_0 a_2 t + \lambda_0^2 a_1^2 t^2 - 2 \frac{a_1^2 \sigma_\lambda^2}{q_{11} + q_{22}} t + 2 \frac{a_1^2 \sigma_\lambda^2}{(q_{11} + q_{22})^2} \left(e^{(q_{11} + q_{22})t} - 1\right), \quad (12)$$

где $\sigma_\lambda^2 = \sum_{i=1}^2 (\lambda_i - \lambda_0)^2 \pi_i$.

Наконец, третий момент $\mu_3(t)$, как следует из (10), удовлетворяет уравнению

$$\dot{\mu}_3(t) = \lambda_0 a_3 + 3a_2 \sum_{n=1}^2 \lambda_n \mu_{n,1}(t) + 3a_1 \sum_{n=1}^2 \lambda_n \mu_{n,2}(t), \text{ решение которого имеет вид}$$

$$\mu_3(t) = \lambda_0 a_3 t + \lambda_0^3 a_1^3 t^3 + 3 \left(a_1 a_2 \lambda_0^2 - 2 \frac{\lambda_0 a_1^3}{q_{11} + q_{22}} \sigma_\lambda^2 \right) t^2 - 6 a_1 a_2 \frac{\sigma_\lambda^2}{q_{11} + q_{22}} t +$$

$$+ 6 \frac{a_1^3 (m_3 - \lambda_0^3)}{(q_{11} + q_{22})^2} t e^{(q_{11} + q_{22})t} + 6 \left(\frac{a_1 a_2 \sigma_\lambda^2}{(q_{11} + q_{22})^2} + 2 \frac{a_1^3 (\lambda_0 m_2 - m_3)}{(q_{11} + q_{22})^3} \right) \left(e^{(q_{11} + q_{22})t} - 1 \right), \quad (13)$$

где $m_2 = \lambda_1^2 \pi_1 + \lambda_2^2 \pi_2$, $m_3 = \lambda_1^3 \pi_1 + \lambda_2^3 \pi_2$.

При $t \gg 1$ соотношения (12) и (13) принимают вид

$$\mu_2(t) = \lambda_0 a_2 t + \lambda_0^2 a_1^2 t^2 - 2 \frac{a_1^2 \sigma_\lambda^2}{q_{11} + q_{22}} t,$$

$$\mu_3(t) = \lambda_0 a_3 t + \lambda_0^3 a_1^3 t^3 + 3 \left(a_1 a_2 \lambda_0^2 - 2 \frac{\lambda_0 a_1^3}{q_{11} + q_{22}} \sigma_\lambda^2 \right) t^2 - 6 a_1 a_2 \frac{\sigma_\lambda^2}{q_{11} + q_{22}} t.$$

4. Простая оценка вероятности разорения

Следуя [9], назовем «простой» оценкой вероятности разорения, такую её аппроксимацию, для построения которой используются только моменты a_k распределения страховых выплат и не требуется строить оценку решения уравнения (4). Для классической модели Крамера – Лундберга известна оценка вероятности разорения De Vylder [9,10], которая основана на идее замены истинного процесса риска с произвольным распределением страховых выплат на процесс риска, в котором страховые выплаты имеют экспоненциальное распределение, а параметры аппроксимирующего процесса $\hat{\lambda}$, \hat{C} подобраны так, чтобы три первых момента у истинного и аппроксимирующего процессов риска совпадали. За оценку вероятности разорения при этом принимается вероятность разорения, соответствующая аппроксимирующему процессу. Воспользуемся этой идеей.

Пусть страховые выплаты имеет экспоненциальное распределение

$$\psi(S) = \frac{1}{a} e^{-\frac{S}{a}}, \quad (14)$$

поток страховых выплат является пуассоновским с интенсивностью $\hat{\lambda}$, скорость поступления страховых премий равна $\hat{C} = (1 + \hat{\theta})\hat{\lambda}a$. Тогда, как хорошо известно [1–3], вероятность разорения страховой компании

$$P(S) = \frac{1}{1 + \hat{\theta}} e^{-\frac{\hat{\theta}}{(1 + \hat{\theta})a} S}. \quad (15)$$

Соотношение (15) будем принимать за оценку $\hat{P}(S)$ вероятности разорения страховой компании. Так как оценка (15) зависит от трёх неизвестных параметров a , $\hat{\lambda}$, $\hat{\theta}$, то потребуем, чтобы у истинного процесса риска (2) и у аппроксимирующего процесса риска $\hat{S}(t) = S(0) + \hat{C}t - \hat{y}(t)$, где $\hat{y}(t) = \sum_{j=1}^{\hat{n}(t)} \hat{y}_j$, $\hat{n}(t)$ – количество событий пуассоновского потока с интенсивностью $\hat{\lambda}$ за время t и случайные величины \hat{y}_j имеют распределение (14), совпадали три первых момента.

Из соотношений (11), (12) и (13) вытекает, что моменты процесса $\hat{y}(t)$ будут равны $\hat{\mu}_1(t) = \hat{\lambda}at$, $\hat{\mu}_2(t) = 2\hat{\lambda}a^2t + \hat{\lambda}^2a^2t^2$, $\hat{\mu}_3(t) = \hat{\lambda}a_3t + \hat{\lambda}^3a^3t^3 + 6a^3\hat{\lambda}^2t^2$.

Равенства $M\{S(t)^k\} = M\{\hat{S}(t)^k\}$, $k = 1, 2, 3$, приводят к соотношениям

$$M(\hat{c}t - \hat{y}(t)) = M(ct - y(t)), \quad (16)$$

$$M(\hat{c}t - \hat{y}(t))^2 = M(ct - y(t))^2, \quad (17)$$

$$M(\hat{c}t - \hat{y}(t))^3 = M(ct - y(t))^3. \quad (18)$$

Из соотношения (16) получим

$$z_1 = C - \lambda_0 a_1 = \hat{C} - \hat{\lambda}a. \quad (19)$$

Из соотношения (17) при $t \gg 1$, учитывая (19), будем иметь

$$z_2 = \lambda_0 a_2 - 2 \frac{a_1^2 \sigma_\lambda^2}{q_{11} + q_{22}} = 2\hat{\lambda}a^2. \quad \text{Наконец, соотношение (18) при } t \gg 1 \text{ даёт}$$

$$z_3 = \lambda_0 a_3 - 6a_1 a_2 \frac{\sigma_\lambda^2}{q_{11} + q_{22}} = 6\hat{\lambda}a^3, \quad \text{откуда окончательно получаем, что параметры аппроксимирующего потока } \hat{S}(t) \text{ равны}$$

$$a = \frac{z_3}{3z_2}, \quad (20)$$

$$\hat{\lambda} = \frac{9z_3^3}{2z_2^2}, \quad (21)$$

$$\hat{\theta} = \frac{\hat{C} - \hat{\lambda}a}{\hat{\lambda}a} = \frac{2z_1 z_3}{3z_2^2}. \quad (22)$$

5. Примеры

1. Экспоненциальное распределение страховых выплат $\psi(S) = \frac{1}{a} e^{-\frac{S}{a}}$

Вероятность разорения для этого случая при $m = 2$ получена в [3]. На рис. 1 приведены графики вероятности разорения страховой компании $P(S)$, вычисленные по формулам из [3], и её оценки $\hat{P}(S)$, определяемой соотношениями (15), (20)–(22). Для сравнения приведён график вероятности разорения страховой компании $P_1(S)$, полученный в предположении, что поток страховых премий является пуассоновским потоком, интенсивность которого равняется λ_0 , параметры C и a сохранены. Значения параметров $a = 1$, $\theta = 0.5$, $\lambda_1 = 1$, $\lambda_2 = 20$, $q_{11} = -5$, $q_{22} = -10$.

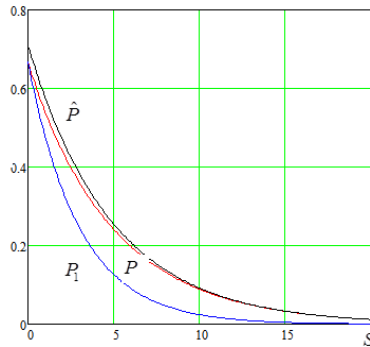


Рис. 1. Оценка вероятности разорения при экспоненциальном распределении страховых выплат

2. Гиперэкспоненциальное распределение страховых выплат

Пусть $\psi(S) = A_1 \alpha_1 e^{-\alpha_1 S} + A_2 \alpha_2 e^{-\alpha_2 S}$, где $A_1, A_2 > 0$, $A_1 + A_2 = 1$. Точные формулы для вычисления вероятности разорения для этого случая при $m = 2$ получены в [7]. На рис. 2 приведены графики вероятности разорения страховой компании $P(S)$, вычисленные по формулам из [7], и её оценки $\hat{P}(S)$, определяемой соотношениями (15), (20)–(22). Параметры $A_1 = 0.5$, $A_2 = 0.5$, $\alpha_1 = 1$, $\alpha_2 = 2$, $q_{11} = -3$, $q_{22} = -5$, $\lambda_1 = 2$, $\lambda_2 = 4$, $\theta = 0.2$.

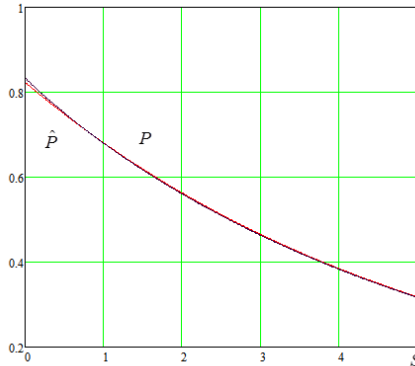


Рис. 2. Оценка вероятности разорения при гиперэкспоненциальном распределении страховых выплат

Заключение

В работе получены расчетные формулы, позволяющие вычислить оценку вероятности разорения страховой компании при произвольном распределении страховых выплат для модели Крамера – Лундберга с ММР-потоком страховых выплат. Как следует из рассмотренных примеров, во всех рассмотренных случаях точность предлагаемой оценки является удовлетворительной. В принципе тот же подход может быть использован для вычисления вероятностей разорения и для более сложных моделей, например для случая, когда и моменты поступления страховых премий образуют ММР-поток [11].

ЛИТЕРАТУРА

1. Panjer H.Y., Wilmot G.E. Insurance Risk Models. – Society of Actuaries, 1992. – 442 p.
2. Байэрс Н., Гербер Х., Джонс Д., Несбит С., Хикман Д. Актуарная математика. – М.: Янус-К, 2001. – 656 с.
3. Глухова Е.В., Змеев О.А., Лившиц К.И. Математические модели страхования. – Томск: Изд-во Том. ун-та, 2004. – 180 с.
4. Lu Y., Li S. On the Probability of Ruin in a Markov-modulated Risk Model // Insurance: Mathematics and Economics. – 2005. – V. 37 (2). – P. 522–532.
5. Лившиц К.И., Бублик Я.С. Вероятность разорения страховой компании при дважды стохастическом потоке страховых выплат // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2010. – № 1 (5). – С. 66–77.
6. Лившиц К.И., Бублик Я.С. Распределение условного времени до разорения страховой компании при дважды стохастическом потоке страховых выплат // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2010. – № 4 (13). – С. 15–23.
7. Лившиц К.И., Сухотина Л.Ю. Вероятность разорения страховой компании при гиперэкспоненциальных распределениях страховых премий и страховых выплат для различных моделей страхования // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2016. – № 2 (18). – С.37–45.
8. Назаров А.А., Моисеева С.П. Метод асимптотического анализа в теории массового обслуживания. – Томск: Изд-во НТЛ, 2006. – 112 с.
9. Grandell J. Simple approximation of ruin probabilities // Insurance: Mathematics and Economics. – 2000. – V. 26. – P. 157–173.
10. De Vylder F.E. A practical solution to the problem of ultimate ruin probability // Scandinavian Actuarial Journal. – 1978. – P. 114–119.
11. Лившиц К.И., Бублик Я.С. Вероятность разорения страховой компании при дважды стохастических потоках страховых премий и страховых выплат // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2011. – № 4 (10). – С. 64–73.

РОБАСТНАЯ ЭКСТРАПОЛЯЦИЯ В ДИСКРЕТНЫХ СИСТЕМАХ СО СЛУЧАЙНЫМИ СКАЧКАМИ С ДВУМЯ СОСТОЯНИЯМИ*

К.С. Ким, В.И. Смагин

Томский государственный университет
kks93@rambler.ru, vsm@mail.tsu.ru

Введение

Задачи оптимального управления и оценивания для объектов с непрерывным временем и скачкообразными параметрами рассматривались в [1–5]. Аналогичные задачи для дискретных систем изучались в [6–9]. В настоящей работе рассмотрена задача синтеза оптимального экстраполятора для дискретного объекта со случайными скачкообразными параметрами с двумя состояниями. Предложено решение задачи синтеза робастного экстраполятора с определением матрицы коэффициентов передачи, не зависящими от состояния скачкообразного процесса.

*Работа выполнена при финансовой поддержке РФФИ, проект № 17-08-00920.

1. Модель объекта. Постановка задачи

Пусть модель объекта описывается дискретным уравнением $x(k+1) = A_\gamma x(k) + q_\gamma(k)$, $x(0) = x_0$, где $x(k) \in \mathbb{R}^n$ – вектор состояния, x_0 – случайный вектор $\bar{x}_0 = M\{x_0\}$, $N_0 = M\{(x_0 - \bar{x}_0)(x_0 - \bar{x}_0)^T\}$; A_γ – заданная постоянная матрица; $\gamma = \gamma(k)$ – марковская цепь с двумя дискретными состояниями γ_1, γ_2 ; $q_\gamma(k)$ – вектор случайных возмущений со следующими характеристиками $M\{q_\gamma(k)\} = 0$, $M\{q_\gamma(k)q_\gamma^T(k)\} = Q_\gamma \delta_{kj}$ (Здесь $M\{\cdot\}$ – математическое ожидание)

Канал наблюдений имеет вид $y = S_\gamma x(k) + v_\gamma(k)$, где $v_\gamma(k)$ – независимые гауссовские случайные последовательности с характеристиками $M\{v_\gamma(k)\} = 0$, $M\{v_\gamma(k)v_\gamma^T(k)\} = V_\gamma \delta_{kj}$.

Вектор вероятностей $p(k) = (p_1(k), p_2(k))^T$, где $p_i(k) = P\{\gamma(k) = i\}$, $i = 1, 2$, удовлетворяет уравнению $p(k+1) = Pp(k)$, $p(0) = p_0$, где $P = \{p_{ij}\}$ – матрица вероятностей перехода системы из состояния i в состояние j за один такт времени, p_0 – вектор начальных значений вероятностей.

Требуется найти оценку экстраполяции $\hat{x}(k)$ на основе минимизации следующего критерия:

$$J[0; T_f] = M \left\{ \left[\sum_{k=0}^{T_f} \sum_{i=1}^2 p_i(k) e^T(k) R_i e(k) + \sum_{i=1}^2 p_i((T_f)) e^T((T_f)) L_i(T_f) e(T_f) \right] / \gamma(0) = \gamma_0 \right\}, \quad (1)$$

где $e(k) = x(k) - \hat{x}(k)$, $R_i > 0$ и $L_i(T_f) > 0$ – заданные весовые матрицы, T_f – конечный момент времени, γ_0 начальное значение случайной переменной γ .

2. Оптимизация критерия (1)

Построим оценку $\hat{x}(k)$ с использованием экстраполятора Калмана $\hat{x}(k+1) = A_\gamma \hat{x}(k) + K(k)(y(k) - S_\gamma \hat{x}(k))$, $\hat{x}(0) = \bar{x}$. Также $\bar{x} = M\{x_0\}$, кроме того предполагаются известными начальные дисперсии $N_{0,i} = M\{(x_0 - \bar{x}_0)(x_0 - \bar{x}_0)^T / \gamma = \gamma_i\}$, $i = \overline{1, 2}$.

Введём обозначения для матриц $Q_\gamma, V_\gamma, N_\gamma, A_\gamma, S_\gamma$, при $\gamma = \gamma_i$: Q_i, V_i, N_i, A_i, S_i соответственно. Дисперсионные матрицы $N_i(k)$ удовлетворяют следующим разностным матричным уравнениям (ниже в формулах для простоты зависимость K от переменной k опускается):

$$N_i(k+1) = (A_i - KS_i)(p_{i,1}N_1(k) + p_{i,2}N_2(k))(A_i - KS_i)^T + Q_i + KV_iK^T, \quad N_i(0) = N_0. \quad (2)$$

Представим критерий (1) в виде суммы

$$J[0; T_f] = \sum_{i=1}^2 J_i[k; T_f], \quad k = \overline{0, T_f}. \quad (3)$$

В (3) значения $J_i[k; T_f]$ можно представить в виде

$$J_i[k; T_f] = \sum_{\xi=k}^{T_f-1} \text{tr } p_i(\xi) N_i(\xi) R_i(\xi) + \text{tr } p_i(T_f) N_i(T_f) L_i(T_f), \quad (4)$$

где tr – след квадратичной матрицы.

Введём функцию:

$$W(k, N_i(k)) = \text{tr } p_i(k) N_i R_i(k) + \text{tr} \sum_{t=k}^{T_f} p_i(t) [Q_i + K V_i K^T + \Psi_i(t)] L_i(t), \quad (5)$$

где $\Psi_i(t) > 0$ – некоторая матрица, $L_i(t)$ удовлетворяет следующему разностному уравнению:

$$L_i(k) = (A_i - K S_i)^T (p_{i,1} L_1(k+1) + p_{i,2} L_2(k+1)) (A_i - K S_i) + R_i, \quad L_i(T_f) = L_{T_f}. \quad (6)$$

Обозначим $\bar{\Psi}_i(t) = Q_i + K V_i K^T + \Psi_i(t)$. Тогда функция (5) примет вид:

$$W(k, N_i(k)) = \text{tr } p_i(k) N_i R_i(k) + \text{tr} \sum_{t=k}^{T_f} p_i(t) \bar{\Psi}_i(t) L_i(t).$$

Просуммируем по $k = \overline{t, T_f - 1}$ конечные разности функции $W(k, N_i(k))$, учитывая формулу (2):

$$\begin{aligned} \sum_{k=t}^{T_f-1} \Delta W(k, N_i(k)) &= \sum_{k=t}^{T_f-1} [W(k+1, N_i(k+1)) - W(k, N_i(k))] = \\ &= \sum_{k=t}^{T_f-1} \text{tr} [p_i(k+1) N_i(k+1) L_i(k+1) - p_i(k) N_i(k) L_i(k) - p_i(k) \bar{\Psi}_i(k) L_i(k)]. \end{aligned} \quad (7)$$

С другой стороны, данное выражение можно представить как:

$$\begin{aligned} \sum_{k=t}^{T_f-1} \Delta W(k, N_i(k)) &= W(t+1, N_i(t+1)) - W(t, N_i(t)) + \dots + \\ &+ W(T_f, N_i(T_f)) - W(T_f-1, N_i(T_f-1)) = \\ &= \text{tr } p_i(T_f) N_i(T_f) L_i(T_f) - \text{tr } p_i(t) N_i(t) L_i(t) - \text{tr} \sum_{\xi=t}^{T_f-1} p_i(\xi) \bar{\Psi}_i(\xi) L_i(\xi). \end{aligned} \quad (8)$$

Подставим в формулу (4) разность (7) и (8):

$$\begin{aligned} J_i[k; T_f] &= \sum_{\xi=k}^{T_f-1} \text{tr } p_i(\xi) N_i(\xi) R_i(\xi) + \text{tr } p_i(T_f) N_i(T_f) L_i(T_f) + \\ &+ \sum_{\xi=k}^{T_f-1} \text{tr} [p_i(\xi+1) N_i(\xi+1) L_i(\xi+1) - p_i(\xi) N_i(\xi) L_i(\xi) - p_i(\xi) \bar{\Psi}_i(\xi) L_i(\xi)] - \\ &- \left[\text{tr } p_i(T_f) N_i(T_f) L_i(T_f) - \text{tr } p_i(k) N_i(k) L_i(k) - \text{tr} \sum_{\xi=k}^{T_f-1} p_i(\xi) \bar{\Psi}_i(\xi) L_i(\xi) \right] = \\ &= \sum_{\xi=k}^{T_f-1} \text{tr } p_i(\xi) N_i(\xi) R_i(\xi) + \sum_{\xi=k}^{T_f-1} \text{tr } p_i(\xi+1) N_i(\xi+1) L_i(\xi+1) - \sum_{\xi=k+1}^{T_f-1} \text{tr } p_i(\xi) N_i(\xi) L_i(\xi). \end{aligned} \quad (9)$$

Распишем критерий (3), подставив (9) и (2):

$$\begin{aligned} J[0; T_f] &= \sum_{i=1}^2 \left\{ \sum_{\xi=k}^{T_f-1} \text{tr } p_i(\xi) N_i(\xi) R_i(\xi) - \sum_{\xi=k+1}^{T_f-1} \text{tr } p_i(\xi) N_i(\xi) L_i(\xi) + \right. \\ &+ \left. \sum_{\xi=k}^{T_f-1} \text{tr } p_i(\xi+1) [(A_i - K S_i) (p_{i,1} N_1(\xi) + p_{i,2} N_2(\xi)) (A_i - K S_i)^T + Q_i + K V_i K^T] L_i(\xi+1) \right\}. \end{aligned}$$

Используя правила дифференцирования функции след tr от произведения матриц, найдем производную по K .

$$\begin{aligned} \frac{\partial J}{\partial K} = & \sum_{\xi=k}^{T_j-1} \sum_{i=1}^2 \left[-L_i(\xi+1)p_i(\xi+1)A_i(p_{i,1}N_1(\xi) + p_{i,2}N_2(\xi))S_i^T - \right. \\ & - p_i(\xi+1)L_i(\xi+1)A_i \left(\sum_{j=1}^2 p_j N_j(\xi) \right) S_i^T + p_i(\xi+1)L_i(\xi+1)KS_i(p_{i,1}N_1(\xi) + p_{i,2}N_2(\xi))S_i^T + \\ & + L_i(\xi+1)p_i(\xi+1)KS_i(p_{i,1}N_1(\xi) + p_{i,2}N_2(\xi))S_i^T + p_i(\xi+1)L_i(\xi+1)KV_i + \\ & \left. + L_i(\xi+1)p_i(\xi+1)KV_i \right]. \end{aligned}$$

Приравняв эту производную нулю и полагая, что каждое слагаемое по равно нулю, получим уравнения для определения элементов матрицы K

$$\begin{aligned} & p_i(\xi+1)L_i(\xi+1)KS_i(p_{i,1}N_1(\xi) + p_{i,2}N_2(\xi))S_i^T + p_i(\xi+1)L_i(\xi+1)KV_i + \\ & + L_i(\xi+1)p_i(\xi+1)KS_i(p_{i,1}N_1(\xi) + p_{i,2}N_2(\xi))S_i^T + L_i(\xi+1)p_i(\xi+1)KV_i = \\ & = L_i(\xi+1)p_i(\xi+1)A_i(p_{i,1}N_1(\xi) + p_{i,2}N_2(\xi))S_i^T + \\ & + p_i(\xi+1)L_i(\xi+1)A_i(p_{i,1}N_1(\xi) + p_{i,2}N_2(\xi))S_i^T. \end{aligned} \quad (10)$$

Запишем аналитическое решение линейного матричного уравнения (10) для вектора $\text{ct}(K)$ с использованием операции Кронекеровского произведения [10]:

$$\begin{aligned} \text{ct}(K) = & \left(\sum_{i=1}^2 p_i(k+1) \left[L_i(k+1) \otimes S_i(p_{i,1}N_1(k) + p_{i,2}N_2(k))S_i^T + L_i(k+1) \otimes V_i \right] \right)^{-1} \times \\ & \times \text{ct} \left(\sum_{i=1}^2 p_i(k+1)L_i(k+1)A_i(p_{i,1}N_1(k) + p_{i,2}N_2(k))S_i^T \right), \end{aligned} \quad (11)$$

где $\text{ct}(K)$ – вектор-столбец, составленный из элементов строк матрицы K .

Для нахождения матрицы K необходимо решить с учётом уравнения (11) двухточечную краевую задачу (2) и (6). В этом случае матрица K будет зависеть от времени.

Представляет интерес синтез стационарного экстраполятора. В этом случае матрица коэффициентов передачи будет постоянной, а критерий примет вид:

$$J[0; \infty] = \sum_{i=1}^2 \sum_{\xi=k}^{\infty} \text{tr} \bar{p}_i N_i R_i.$$

Двухточечная краевая задача преобразуется в следующую систему матричных уравнений

$$N_i = (A_i - KS_i)(p_{i,1}N_1 + p_{i,2}N_2)(A_i - KS_i)^T + Q_i + KV_i K^T, \quad (12)$$

$$L_i = (A_i - KS_i)^T (p_{i,1}L_1 + p_{i,2}L_2)(A_i - KS_i) + R_i, \quad (13)$$

$$\begin{aligned} \text{ct}(K) = & \left(\sum_{i=1}^2 \bar{p}_i \left[L_i \otimes S_i(p_{i,1}N_1 + p_{i,2}N_2)S_i^T + L_i \otimes V_i \right] \right)^{-1} \times \\ & \times \text{ct} \left(\sum_{i=1}^2 \bar{p}_i L_i A_i (p_{i,1}N_1 + p_{i,2}N_2)S_i^T \right), \end{aligned} \quad (14)$$

где \bar{p}_i – установившиеся вероятности в системе (решение системы (1) при $k \rightarrow \infty$).

Таким образом, для синтеза стационарного экстраполятора необходимо решить систему матричных уравнений (12)–(14).

Отметим, что из уравнения (13) и условия $R_i > 0$ следует справедливость теоремы 1.6 [11], а это означает выполнение условия стохастической устойчивости стационарного экстраполятора со скачкообразными параметрами.

3. Пример синтеза экстраполятора

Рассмотрим задачу синтеза экстраполятора для следующих исходных данных:

$$A_1 = \begin{pmatrix} 1,075 & 0,1 \\ -0,05 & 0,94 \end{pmatrix}, A_2 = \begin{pmatrix} 1,15 & 0,75 \\ -0,02 & 0,725 \end{pmatrix}, Q_1 = \begin{pmatrix} 0,3 & 0 \\ 0 & 0,4 \end{pmatrix}, Q_2 = \begin{pmatrix} 0,5 & 0 \\ 0 & 0,3 \end{pmatrix},$$

$$S_1 = S_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, V_1 = V_2 = \begin{pmatrix} 0,06 & 0,01 \\ 0,01 & 0,02 \end{pmatrix}, R_1 = R_2 = \begin{pmatrix} 0,1 & 0 \\ 0 & 0,15 \end{pmatrix}.$$

Для приведённых данных решение системы матричных уравнений (12)–(14) даёт следующие значения:

$$K = \begin{pmatrix} 0,997 & 0,349 \\ -0,042 & 0,806 \end{pmatrix}.$$

В данном примере матрицы A_1 и A_2 неустойчивые их собственные значения приведены в табл. 1.

Таблица 1

Собственные числа матриц A_1 и A_2

A_1	A_2
$\begin{pmatrix} 1,008+0,021i \\ 1,008-0,021i \end{pmatrix}$	$\begin{pmatrix} 1,111 \\ 0,764 \end{pmatrix}$

В тоже время динамика экстраполятора остается устойчивой для возможных значений скачкообразного процесса γ . В табл. 2 приведены собственные числа матриц $A_1 - KS_1$ и $A_2 - KS_2$.

Таблица 2

Собственные числа матриц $A_1 - KS_1$ и $A_2 - KS_2$

$A_1 - KS_1$	$A_2 - KS_2$
$\begin{pmatrix} 0,053 \\ 0,159 \end{pmatrix}$	$\begin{pmatrix} 0,186 \\ -0,114 \end{pmatrix}$

Как видно из таблицы, матрицы динамики экстраполятора для обоих состояний устойчивы, что гарантирует стохастическую устойчивость.

Заключение

Получено решение задачи синтеза нестационарного и стационарного экстраполятора для линейного дискретного объекта со случайными скачкообразными параметрами с двумя состояниями. В примере показана возможность решения задачи синтеза робастного экстраполятора с определением матрицы коэффициентов передачи, не зависящей от состояния скачкообразного процесса.

ЛИТЕРАТУРА

1. Wonham W.M. Random differential equation in control theory / ed. A.T. Bharucha-Reid. // Probabilistic methods in applied mathematics. Academ. Press. N.Y. – 1970. – P. 131–213.
2. Shi P., Boukas E.K., Agarwal R.K. Kalman filtering for continuous-time uncertain systems with Markovian jumping parameters // IEEE Transactions on Automatic Control. – 1999. – V. 44. – Is. 8. – P. 1592–1597.
3. Смагин В.И., Поползухина Е.В. Синтез следящих систем управления для объектов со случайными скачкообразными параметрами и мультипликативными возмущениями // Вестник Томского государственного университета. – 2000. – № 271. – С. 171–174.
4. Ломакина С.С., Смагин В.И. Робастная фильтрация в непрерывных системах со случайными скачкообразными параметрами // Вестник Томского государственного университета. – 2003. – № 280. – С. 201–203.
5. Ломакина С.С., Смагин В.И. Робастная фильтрация для непрерывных систем со случайными скачкообразными параметрами и вырожденными шумами в наблюдениях // Автметрия. – 2005. – № 2. – С. 36–43.

6. Liu W. State estimation for discrete-time Markov jump linear systems with time-correlated measurement noise // *Automatica*. – 2017. – V. 76. – P. 266–276.
7. Costa E.F., De Saporta B. Linear Minimum Mean Square Filters for Markov Jump Linear Systems // *IEEE Trans. On Automatic Control*. – 2017. – V. 62. – Is. 7. – P. 3567–3572.
8. Беляевский Г.И., Мисюра И.В. Фильтрация сигналов со скачками, возникающими в дискретном времени и с конечным горизонтом // *Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Физико-математические науки*. – 2014. – № 2 (194). – С. 137–144.
9. Mariton M. Jump linear systems in automatic control. – New York: Marcel Dekker, 1990.
10. Ланкастер П. Теория матриц. – М.: Наука, 1973. – 280 с.
11. Li F., Shi P., Wu L. Control and filtering for semi-markovian jump systems. – *Studies in Systems, Decision and Control*. Springer International Publishing, 2017. – 200 p.

ЗАДАЧА УПРАВЛЕНИЯ ЗАПАСАМИ С УЧЁТОМ ТРАНСПОРТНЫХ ОГРАНИЧЕНИЙ

С.С. Смородинова, В.И. Смагин

Томский государственный университет
ssmorodinsvetlana@gmail.com, vsm@mail.tsu.ru

Введение

В настоящее время логистика играет важную роль в процессе принятия управленческих решений в экономике. Необходимость применения логистики на предприятии объясняется многими причинами. В частности, стремление к обеспечению рентабельности и повышению прибыли вынуждает руководителей предприятий и собственников фирм к повышению надежности и качества поставок и к снижению издержек на хранение товара. Также бурное развитие компьютерных технологий позволяет использовать готовые программные продукты и разрабатывать новые для управления поставками и складскими запасами с целью повышения эффективности функционирования логистических систем.

Решение задач управления запасами при различных условиях и ограничениях рассматривались в работах [1–5]. В настоящей работе рассматривается задача формирования объёма поставок, обеспечивающих желаемый темп роста прибыли склада с учетом затрат на транспортировку товара и издержек на хранение.

1. Описание математической модели

Для описания математической модели склада используем вариант модели [6], записанный в виде разностных уравнений:

$$\begin{aligned} z(k+1) &= (1+k_1)z(k) + u(k) - s(k), \quad z(0) = z_0, \\ v(k+1) &= (1-k_2)v(k) + s(k), \quad v(0) = v_0, \end{aligned} \quad (1)$$

$$w(k+1) = w(k) + cs(k) - c_0z(k) - c_1u(k), \quad w(0) = w_0,$$

$$s(k) = n_0 \exp(-c) \left(1 - \frac{v(k)}{Y} \right) z(k), \quad (2)$$

где $z(k)$ – количество товара на складе, $v(k)$ – количество товара у потребителя, $w(k)$ – прибыль, $u(k)$ – объём поставок, $s(k)$ – объём продаж, k_1 – коэффициент потерь товаров на складе (это доля потерь в один такт), k_2 – коэффициент потребления, c_0 – издержки на хранение единицы товаров в единицу времени, c – стоимость товара на рынке, n_0 – коэффициент продаж, c_1 – транспортные издержки на поставку единицы товара в единицу времени, Y – потенциальный спрос.

Представим модель (1), (2) в векторно-матричной форме с учётом дополнительного аддитивного возмущения $q(k)$:

$$x(k+1) = A\varphi(x(k)) + Bu(k) + q(k), \quad x(0) = x_0, \quad (3)$$

где $x(k) = [z(k), v(k), w(k)]^T$, вектор $\varphi(x(k))$ задаётся в виде вектора $\varphi(k) = [z(k), v(k), w(k), z(k)v(k)]^T$, $q(k)$ – вектор возмущений, описывающий действие случайных факторов и возможных ошибок параметров модели. Для модели (3) будем предполагать, что $q(k)$ – гауссовская последовательность с характеристиками $M\{q(k)\} = 0$, $M\{q(k)q^T(j)\} = Q\delta_{k,j}$, где $M\{\cdot\}$ – операция математического ожидания, $\delta_{k,j}$ – символ Кронекера. В (3) матрицы A и B имеют вид:

$$A = \begin{bmatrix} 1 - (n_0 e^{-c}) - k_1 & 0 & 0 & \frac{n_0 e^{-c}}{Y} \\ n_0 e^{-c} & 1 - k_2 & 0 & -\frac{n_0 e^{-c}}{Y} \\ c(n_0 e^{-c}) - c_0 & 0 & 1 & -c \frac{n_0 e^{-c}}{Y} \end{bmatrix}, B = \begin{bmatrix} -1 \\ 0 \\ c_1 \end{bmatrix}.$$

2. Алгоритм управления

Введём темп роста чистой прибыли r . Тогда желаемая прибыль будет определяться разностным уравнением: $\bar{w}(t+1) = (1+r)\bar{w}(t)$, $\bar{w}(0) = w(0)$.

Сформулируем задачу как задачу слежения за желаемой прибылью, для этого в качестве критерия выберем следующий локальный критерий:

$$J(k) = M\left\{\left(w(k+1) - \bar{w}(k+1)\right)^2 C_v + u(k)^2 D\right\}, \quad (4)$$

где $C_v \geq 0$, $D > 0$ – весовые матрицы, $w(k) = Fx(k)$ – вектор наблюдаемого выхода ($F = [0 \ 0 \ 1]$).

Вычислив критерий (4), получим:

$$J(t) = u^T(t) \left(B^T F^T C_v F B + D \right) u(t) + u^T(t) \left(B^T F^T C_v \left(F A \varphi(x(t)) - \bar{w}(t) \right) + \left(\bar{F} A \varphi(x(t)) - \bar{w}(t) \right)^T C_v \bar{F} B u(t) + \beta \right), \quad (5)$$

где β – слагаемые, не зависящие от управления. Тогда, выполнив оптимизацию критерия (5), получим закон локально-оптимального управления поставками

$$u^*(k) = - \left(B^T F^T C_v F B + D \right)^{-1} B^T F^T C_v \left(F A \varphi(x(k)) - \bar{w}(k) \right).$$

При управлении поставками используют информацию о количестве товара на складе, прибыль и др. Но зачастую получить полную информацию достаточно затруднительно в связи с дороговизной или невозможностью сбора полной информации. В данной работе рассматривается случай, когда отсутствует информация о потребителе (переменная $v(k)$ недоступна наблюдению). В работе предлагается применить фильтр Калмана для оценки ненаблюдаемой компоненты вектора состояния. Для этого вводится математическую модель сбора информации об объекте $y(k) = Hx(k) + \eta(k)$, где $y(k)$ – вектор, содержащий информацию о состоянии рассматриваемого процесса, H – матрица (если компонента вектора состояния ненаблюдаема, то соответствующий столбец матрицы принимается нулевым), $\eta(k)$ – гауссовская случайная последовательность, не зависящая от $q(k)$, с характеристиками $M\{\eta(k)\} = 0$, $M\{\eta(k)\eta^T(j)\} = V\delta_{k,j}$.

Так как модель нелинейна по состоянию, то для построения оценки $\hat{x}(k)$ будем использовать линеаризованный фильтр Калмана:

$$\hat{x}(k+1) = A\varphi(\hat{x}(k)) + Bu(k) + K_f(k) \left[y(k+1) - H(A\varphi(\hat{x}(k)) + Bu(k)) \right], \hat{x}(0) = \bar{x}(0),$$

$$P_f(k+1/k) = \bar{A}P_f(k)\bar{A}^T + Q,$$

$$K_f(k) = P_f(k+1/k)H^T \left[HP_f(k+1/k)H^T + V \right]^{-1},$$

$$P_f(k+1) = \left[E_2 - K_f(k)H \right] P_f(k+1/k), P_f(0) = P_{f_0}.$$

Предполагается, что априорное распределение вектора f_0 является гауссовским: $M\{f_0\} = \bar{f}_0$, $M\left\{ (f_0 - \bar{f}_0)(f_0 - \bar{f}_0)^T \right\} = P_{f_0}$, где P_{f_0} – дисперсионная матрица ошибок оценивания начального состояния, матрица \bar{A} определяется по формуле $\bar{A} = A \frac{\partial \varphi(x(k))}{\partial x(k)} \Big|_{\hat{x}(k)}$.

3. Учёт транспортных ограничений

Для поставки продукции на склад предприятием используется одно транспортное средство. Грузоподъёмность данного транспортного средства обозначим как u_{\max} .

Поставки определяются с учётом транспортных ограничений:

$$u(k) = \begin{cases} 0, & u(k) < Kg \cdot u_{\max}, \\ u(k), & Kg \cdot u_{\max} < u(k) < u_{\max}, \\ u_{\max}, & u(k) > u_{\max}, \end{cases}$$

где значения Kg – коэффициент использования грузоподъёмности транспортного средства (в логистике Kg обычно выбирается в пределах от 0,8 до 1,0).

4. Численные результаты

В данной работе используются данные для однопродуктового склада для следующих исходных данных

$$k_1 = 0,0001; k_2 = 0,2; k_3 = 0,05;$$

$$c_0 = 1,15; c = 3,55; n_0 = 1,15; r = 0,002;$$

$$C = 0,1; D = 0,001; u_{\max} = 50; Kg = 0,9.$$

Поступление новых поставок имеет фиксированную величину и наступает в момент, когда предыдущая поставка распродана.

В случае, когда отсутствует информация о потребителе, матрица H примет вид:

$$H = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Результаты моделирования представлены на рис. 1–4.

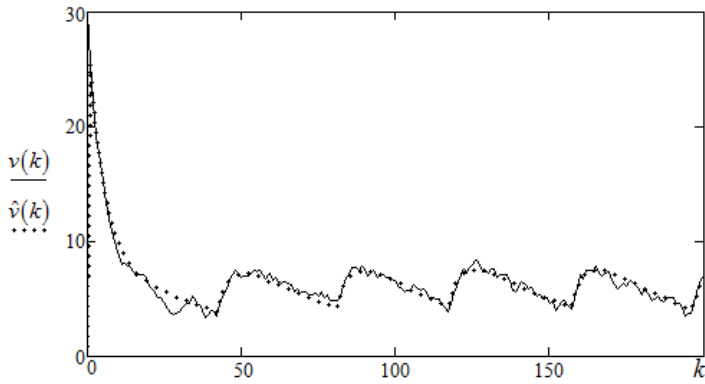


Рис. 1. Количество товара на складе

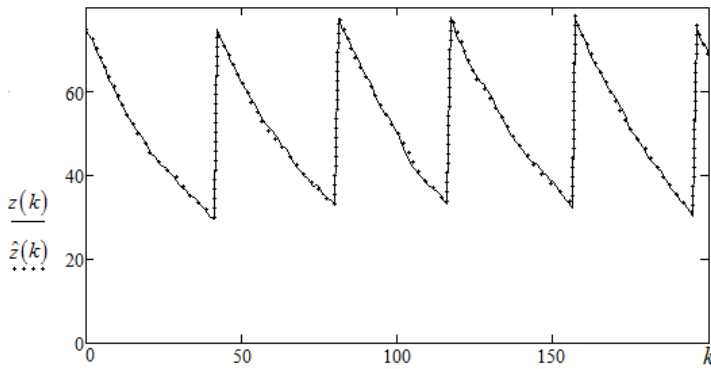


Рис. 2. Объем товара у потребителя и его оценка

На рис. 3 представлены желаемая прибыль $\bar{w}(k)$, реализованная прибыль $w(k)$, а также её оценка $\hat{w}(k)$.

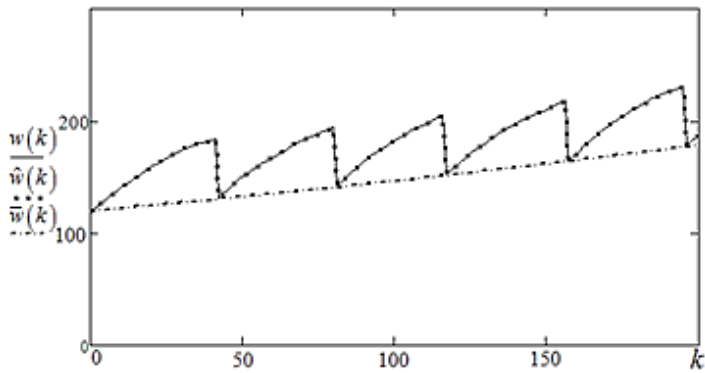


Рис. 3. График изменения прибыли

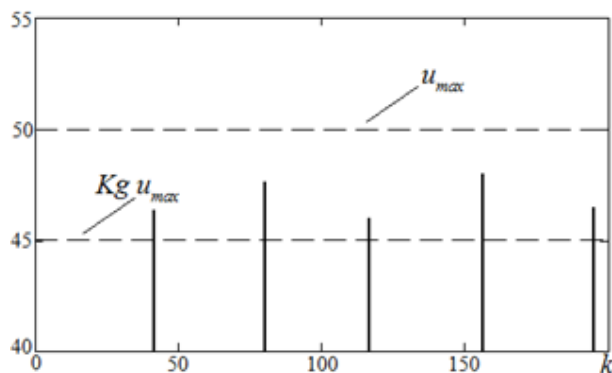


Рис. 4. Диаграмма загруженности транспортного средства при поставке товара на склад

Рис. 4 показывает объёмы поставок, а также моменты этих поставок. Как видно из диаграммы, объёмы поставок не превышают грузоподъемность транспортного средства, обеспечивая при этом коэффициент использования грузоподъемности не ниже заданной величины Kg .

Заключение

В данной работе построена дискретная модель экономической системы, описывающая процесс поставок, хранения и сбыта продукции. Задача управления запасами рассмотрена с учётом транспортных ограничений, учитывающих грузоподъемность и установленный коэффициент использования грузоподъемности. Управление формируется при отсутствии информации о потребителе.

ЛИТЕРАТУРА

1. Лотоцкий В.А. Управление запасами при частично наблюдаемом спросе // Статистические методы теории управления. – М.: Наука, 1978. – С. 222–224.
2. Лотоцкий В.А., Мандель А.С. Модели и методы управления запасами. – М.: Наука, 1991. – 189 с.
3. Беляков А.Г., Латин А.В., Мандель А.С. Управление запасами товаров ажиотажного спроса // Проблемы управления. – 2005. – № 6. – С. 40–45.
4. Смагин В.И., Смагин С.В. Управление запасами по двум критериям с учетом ограничений // Вестник Томского государственного университета. – 2006. – № 290. – С. 244–246.
5. Киселева М.Ю., Смагин В.И. Управление производством, хранением и поставками товаров на основе прогнозирующей модели выхода системы // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2009. – № 2 (5). – С. 24–31.
6. Горский А.А., Колпакова И.Г., Локишин Б.Я. Динамическая модель процесса производства, хранения и сбыта товара повседневного спроса // Изв. РАН. Теория и системы управления. – 1998. – № 1. – С. 144–148.

УПРАВЛЕНИЕ МОДЕЛЬЮ СРЕДНЕЙ ФИРМЫ

Д.А. Попырин, С.А. Цветницкая
Томский государственный университет
svetasa@sibmail.com

Введение

Рассматривается фирма, обладающая средними показателями: величиной капитала и числом сотрудников. В данной статье анализируются две модели средней фирмы: линейная двумерная и нелинейная трёхмерная. Управление моделью строится с использованием метода АКАР.

1. Двумерная модель

В [1] были рассмотрены две модели средней фирмы: линейная двумерная и нелинейная трёхмерная. В качестве фазовых координат двумерной модели были выбраны число сотрудников и капитал фирмы. Введём обозначения: x_1 – число сотрудников, x_2 –

капитал фирмы. При построении модели используется метод пропорций: изменение величины пропорционально «приросту» этой величины, минус её «потери». Изменение числа сотрудников пропорционально разности количества новых и уволившихся сотрудников. Количество новых сотрудников зависит от капитала x_2 . Количество уволившихся не может превысить число нанятых работников в настоящий момент, т.е. это некоторая часть x_1 . Увеличение капитала пропорционально доходу от вложения капитала, минус оплата труда. Учитывая названные пропорции, можно записать систему уравнений

$$\begin{aligned} \dot{x}_1 &= f_1(x) = \alpha x_2 - \gamma x_1, \\ \dot{x}_2 &= f_2(x) = \mu x_2 - \beta x_1. \end{aligned} \quad (1)$$

В (1), в зависимости от параметров α , β , γ , μ , возможны следующие переходные процессы. Рассмотрим случаи а) $\mu > \gamma$, б) $\mu < \gamma$, в) $\mu = \gamma$ (в первом случае коэффициент эффективности капитала больше коэффициента увольнения, для второго случая коэффициент эффективности капитала меньше коэффициента увольнения, в третьем случае коэффициенты равны). Переходные процессы приведены в рис. 1.

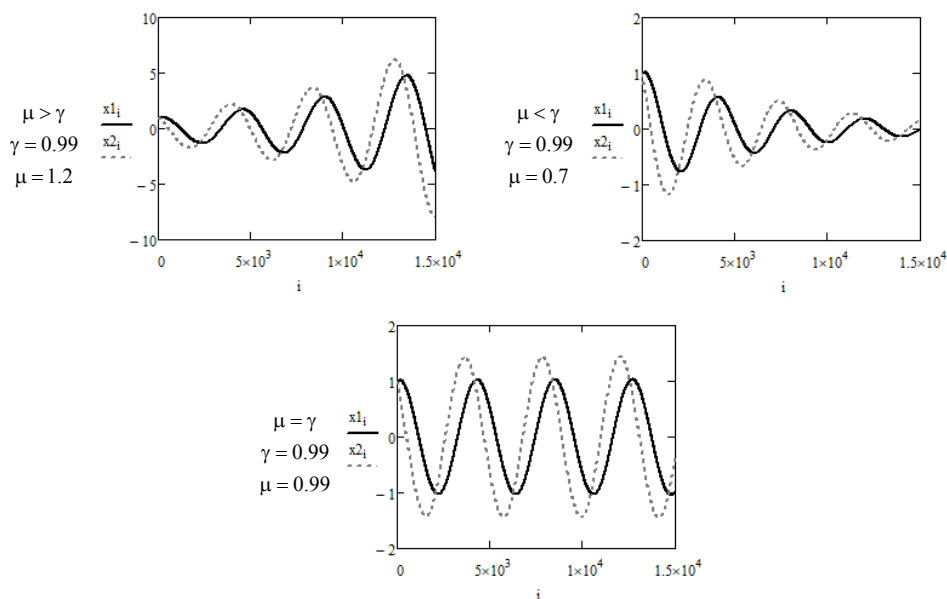


Рис. 1.

В [1] показано, что в случае а) происходит увеличение капитала и числа сотрудников, и наступает момент, когда фирма не может достойно оплачивать труд наёмных рабочих, увеличивается число уволившихся, γ увеличивается и система переходит в состояние б). В состоянии б) происходит уменьшение числа сотрудников, и со временем фирма опять сможет увеличить заработную плату, число уволившихся уменьшится, и система перейдет в случай а). Фазовые портреты приведены в рис. 2. Управляющие параметры для трёх случаев: а) $\mu = 1.2$, $\gamma = 0.99$; б) $\mu = 0.7$, $\gamma = 0.99$; в) $\mu = 0.99$, $\gamma = 0.99$.

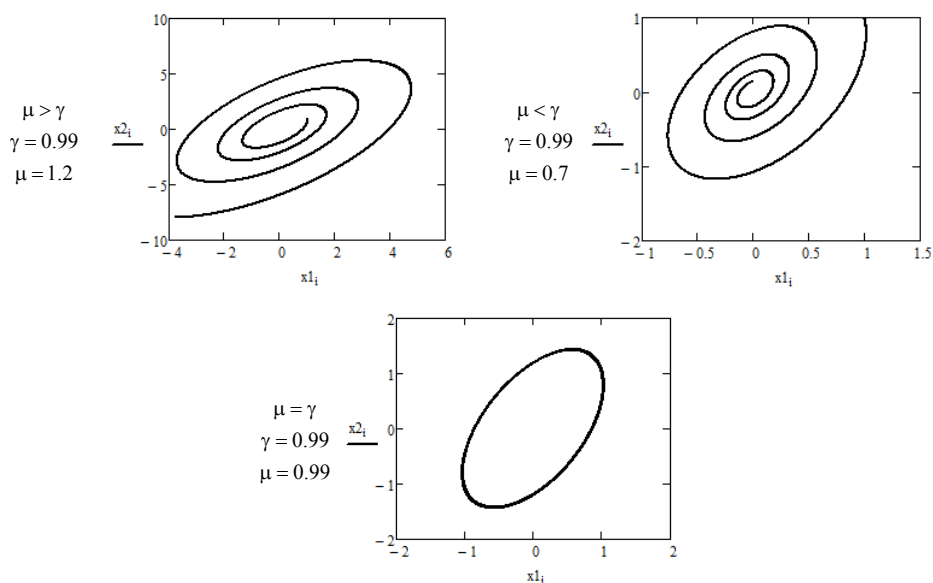


Рис. 2. Фазовые портреты системы (1)

2. Построение управления для двумерной модели

Добавим в систему (1) управления u_1, u_2 , тогда система примет вид

$$\begin{aligned} \dot{x}_1 &= \alpha x_2 - \gamma x_1 + u_1, \\ \dot{x}_2 &= \mu x_2 - \beta x_1 + u_2. \end{aligned} \quad (2)$$

Поставим задачу найти u_1, u_2 такие, чтобы система перешла в состояние $x_1 = \rho x_2, x_2 = x_2^*, \rho, x_2^*$ – заданные константы. Первая цель управления – добиться заданного соотношения между капиталом фирмы и числом сотрудников. Вторая цель – достичь заданного уровня капитала фирмы. Для решения задачи воспользуемся методом АКАР [2].

Идея АКАР состоит в введении макропеременных и записи для них функциональных уравнений, из которых находят искомое управление. Для поставленной цели управления $x_1 = \rho x_2, x_2 = x_2^*, \rho, x_2^*$ – заданные константы. Введём две макропеременные: $\psi_1(x) = x_1 - \rho x_2, \psi_2(x) = x_2 - x_2^*$. Каждая переменная удовлетворяет функциональному уравнению [2]

$$T_i \dot{\psi}_i + \psi_i = 0, \quad (3)$$

где T_i – параметры алгоритма. Для выбранных макропеременных уравнение (3) примет вид

$$T_i \dot{\psi}_i + \psi_i = T_i \left(\frac{\partial \psi_i}{\partial x_1} \dot{x}_1 + \frac{\partial \psi_i}{\partial x_2} \dot{x}_2 \right) + \psi_i = 0, \quad i = 1, 2. \quad (4)$$

Запишем систему (4) в виде

$$\begin{aligned} T_1 (f_1 + u_1 - \rho (f_2 + u_2)) + \psi_1 &= 0, \\ T_2 (f_2 + u_2) + \psi_2 &= 0. \end{aligned} \quad (5)$$

Из (5) находим управления u_1, u_2 ; $u_1 = -\frac{\psi_1}{T_1} - f_1 + \rho \left(-\frac{\psi_2}{T_2} \right), u_2 = -\frac{\psi_2}{T_2} - f_2$.

В [2] показано, что построенное управление доставляет минимум функционалу

$$I = \int_0^{\infty} \sum_i (\psi_i^2 + T_i^2 \dot{\psi}_i^2) dt.$$

На рис. 3 приведены переходные процессы и управление системы (2). Решение системы было получено при следующих параметрах: $\alpha = 1.293$, $\beta = 1.668$, $\gamma = 0.99$, $\mu = 1.2$, $T_1 = T_2 = 1$, $\rho = 0.7$, $x_2^* = 2$.

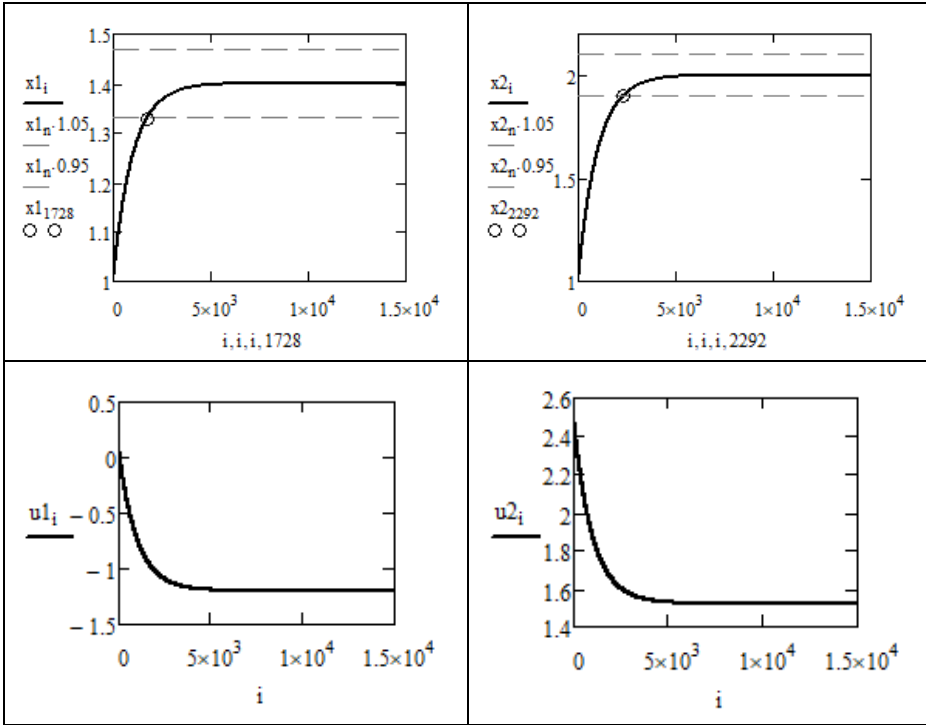


Рис. 3. Переходные процессы и управление системы (2)

Таблица

Численные характеристики переходных процессов и управления системы (2)

	x_1	x_2
Верхнее значение пятипроцентного интервала	1.47	2.1
Нижнее значение пятипроцентного интервала	1.33	1.9
Номер итерации попадания в пятипроцентный интервал	1728	2292
Полные затраты на управление переходным процессом	139.033	195.341
Затраты на управление для попадания в пятипроцентный интервал	27.276	36.551

3. Управление для трехмерной модели

В [2] в качестве модели средней фирмы используют систему Лоренца. В трёхмерной модели в качестве третьей компоненты вводится кредит. Увеличение затрат на сотрудников пропорционально величине капитала и взятому кредиту, минус средства, освободившиеся в связи с увольнением. Изменение капитала пропорционально доходу от эффективного вложения средств капитала и кредита. Из этой величины вычтем оплату деятельности наёмных работников и затраты на погашение долговых обязательств. Прирост величины кредита зависит от капитала фирмы, минус затраты на обслуживаемые кредиты. Трёхмерная модель имеет вид

$$\begin{aligned} \dot{x}_1 &= f_1(x) = \alpha x_2 x_3 - \gamma x_1, \\ \dot{x}_2 &= f_2(x) = \mu(x_2 + x_3) - \beta x_1 x_3, \\ \dot{x}_3 &= f_3(x) = \delta x_2 - \lambda x_3, \end{aligned} \quad (6)$$

δ, λ – коэффициенты, отражающие моменты взятия кредита и моменты, затрудняющие взятие кредита соответственно.

Найдём стационарные точки системы (6), для этого решим систему уравнений

$$\begin{aligned} f_1(x) &= \alpha x_2 x_3 - \gamma x_1 = 0, \\ f_2(x) &= \mu(x_2 + x_3) - \beta x_1 x_3 = 0, \\ f_3(x) &= \delta x_2 - \lambda x_3 = 0. \end{aligned} \tag{7}$$

Эти уравнения имеют тривиальное решение $x_1 = x_2 = x_3 = 0$. Из третьего уравнения (7) имеем $x_2 = \frac{\lambda}{\delta} x_3$ и, после подстановки x_2 в первое и второе уравнения (7), получим:

$$\frac{\alpha\lambda}{\delta} x_3^2 - \gamma x_1 = 0, \quad \mu \left(\frac{\lambda}{\delta} x_3 + x_3 \right) - \beta x_1 x_3 = 0, \quad x_1 = \mu \frac{\lambda + \delta}{\beta\delta}, \quad x_3 = \pm \sqrt{\frac{\delta\gamma}{\alpha\lambda}} x_1.$$

Вторая стационарная точка: $x_1 = \mu \frac{\lambda + \delta}{\beta\delta}, \quad x_3 = \pm \sqrt{\frac{\delta\gamma}{\alpha\lambda}} x_1, \quad x_2 = \frac{\lambda}{\delta} x_3.$

Поведение системы в стационарных точках будет определяться собственными числами матрицы $A(X) = \begin{bmatrix} \frac{\partial f_i}{\partial x_j} \end{bmatrix}$. Матрица $A(X)$ имеет вид

$$A(X) = \begin{bmatrix} -\gamma & \alpha X_3 & \alpha X_2 \\ -\beta X_3 & \mu & \mu - \beta X_1 \\ 0 & \delta & -\lambda \end{bmatrix}. \tag{8}$$

Подставив в (8) стационарные точки и вычислив собственные значения матрицы, определим её устойчивость или неустойчивость по знаку действительной части собственных значений. Для параметров $\alpha = 5, \beta = 8, \gamma = 0.9, \delta = 2.7, \lambda = 3.9, \mu = 2.1$ собственные числа матрицы (8) равны $(0.361 \pm 2.676i, -3.422)$. Переходные процессы и фазовые портреты системы (6) представлены на рис. 4 и 5 соответственно.

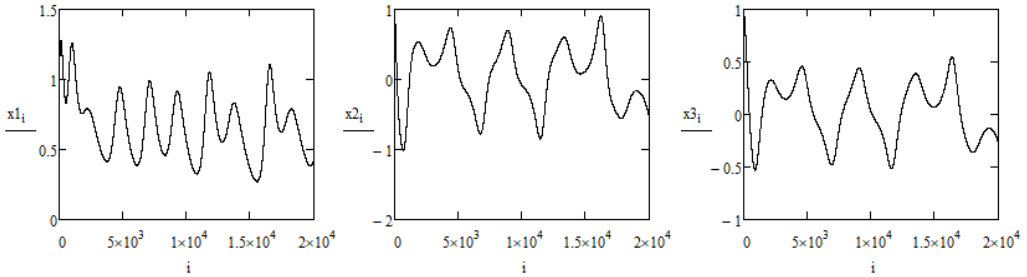


Рис. 4. Переходные процессы системы (6)

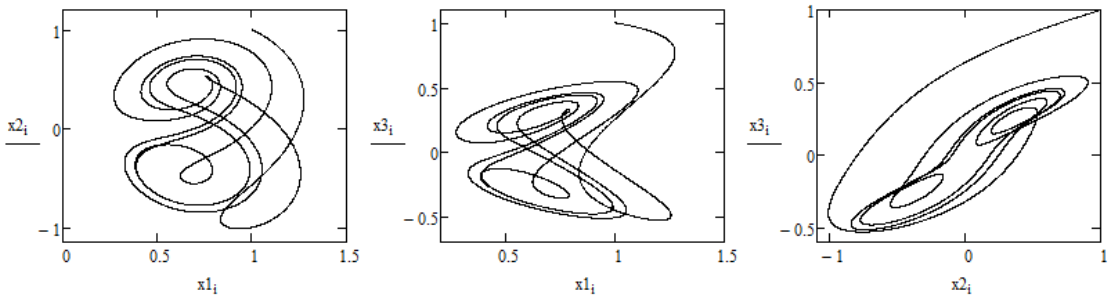


Рис. 5. Фазовые портреты системы (6)

Поведение системы при выбранных параметрах является странным аттрактором. Добавим в систему (6) управления

$$\begin{aligned} \dot{x}_1 &= f_1(x) = \alpha x_2 x_3 - \gamma x_1 + u_1, \\ \dot{x}_2 &= f_2(x) = \mu(x_2 + x_3) - \beta x_1 x_3, \\ \dot{x}_3 &= f_3(x) = \delta x_2 - \lambda x_3 + u_2. \end{aligned} \quad (9)$$

Поставим две цели управления: $x_1 = \rho x_2$, $x_3 = x_3^*$. Первая цель – добиться заданного соотношения между капиталом и затратами на зарплату, вторая – поддерживать уровень кредита. Введём две макропеременные $\psi_1(x) = x_1 - \rho x_2$, $\psi_2(x) = x_3 - x_3^*$. Для каждой макропеременной запишем функциональное уравнение (3).

$$\begin{aligned} T_1 \dot{\psi}_1 + \psi_1 &= T_1 \left(\frac{\partial \psi_1}{\partial x_1} \dot{x}_1 + \frac{\partial \psi_1}{\partial x_2} \dot{x}_2 + \frac{\partial \psi_1}{\partial x_3} \dot{x}_3 \right) + \psi_1 = \\ &= T_1 (\alpha x_2 x_3 - \gamma x_1 + u_1 + \rho(\mu(x_2 + x_3) - \beta x_1 x_3)) + \psi_1 = 0, \end{aligned} \quad (10)$$

$$T_2 \dot{\psi}_2 + \psi_2 = T_2 \left(\frac{\partial \psi_2}{\partial x_1} \dot{x}_1 + \frac{\partial \psi_2}{\partial x_2} \dot{x}_2 + \frac{\partial \psi_2}{\partial x_3} \dot{x}_3 \right) = T_2 (\delta x_2 - \lambda x_3 + u_2) + \psi_2 = 0. \quad (11)$$

Из (10) и (11) найдем управления u_1 , u_2 :

$$u_1 = -\frac{x_1 - \rho x_2}{T_1} - \alpha x_2 x_3 + \gamma x_1 + \rho(\mu(x_2 + x_3) - \beta x_1 x_3), \quad u_2 = -\frac{x_3 - x_3^*}{T_2} - \delta x_2 + \lambda x_3.$$

На рис. 6 приведены переходные процессы, на рис. 7 – управления для системы (9). Параметры системы и алгоритма следующие: $\alpha = 5$, $\beta = 8$, $\gamma = 0.9$, $\delta = 2.7$, $\lambda = 3.9$, $\mu = 2.1$, $\rho = 0.8$, $x_3^* = 0.7$, $T_1 = 0.1$, $T_2 = 0.01$.

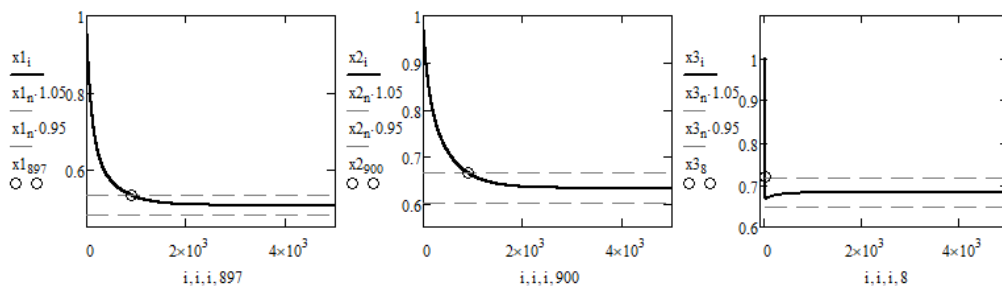


Рис. 6. Переходные процессы системы (9)

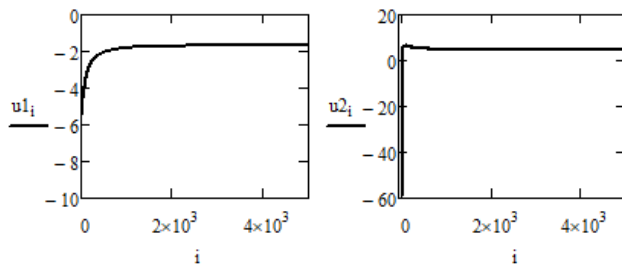


Рис. 7. Управления системы (9)

Значения параметров алгоритма T_1 , T_2 , влияют на достижение целей управления (при $T_1 = 0.1$, $T_2 = 0.01$ $x_3 = 0.663$ в конечный момент времени, при $T_1 = 0.1$, $T_2 = 0.005$,

$x_3 = 0.687$). Вторая цель управления – отношение первой компоненты ко второй, равное 0.8, достигается в обоих случаях.

Таблица

Численные характеристики переходных процессов и управления системы (9)

	x_1	x_2	x_3
Верхнее значение пятипроцентного интервала	0.531	0.644	0.716
Нижнее значение пятипроцентного интервала	0.481	0.601	0.648
Номер итерации попадания в пятипроцентный интервал	897	900	8
Полные затраты на управление переходным процессом	136.039	–	344.489
Затраты на управление для попадания в пятипроцентный интервал	80.059	–	89.676

Заключение

В работе получено управление для двумерной и трёхмерной модели средней фирмы. Управление для двумерной модели позволяет поддерживать капитал фирмы на заданном уровне и заданном соотношении между затратами на персонал и капиталом фирмы. В трёхмерной модели полученное управление позволяет поддерживать заданный уровень кредита и заданное соотношение между капиталом и затратами на персонал.

ЛИТЕРАТУРА

1. Шаповалов В.И. Синергетическая модель устойчивости средней фирмы / Синергетика и проблемы теории управления. Под ред. А.А. Колесникова – М.: ФИЗМАТЛИТ, 2004. – 504 с.
2. Колесников А.А. Проблемы теории аналитического конструирования нелинейных регуляторов и синергетический подход / Синергетика и проблемы теории управления. Под ред. А.А. Колесникова – М.: ФИЗМАТЛИТ, 2004. – 504 с.

ИССЛЕДОВАНИЕ НЕСКОЛЬКИХ ИНВЕСТИЦИОННЫХ СТРАТЕГИЙ С ИСПОЛЬЗОВАНИЕМ КОРОТКОЙ И ДЛИННОЙ ПОЗИЦИИ

Е.В. Зувич

Томский государственный университет
z_kote_v@mail.ru

Введение

Построение оптимального портфеля ценных бумаг является одной из основных задач управления финансовой системой, в которую отдельные активы входят лишь составляющей частью. Однако каждый актив тоже требует отдельного изучения. Инвестиционные инерционные стратегии основаны на предположении о том, что тенденция наиболее доходных акций в прошлом будет сохраняться в будущем, аналогичное предположение и о наиболее убыточных акциях. Поэтому в инерционных стратегиях инвестор часть акций наиболее доходных стремится купить, а менее доходные продать. Инерционные стратегии привлекают своей простотой. В начале развития таких стратегий в качестве критерия ранжирования использовали среднюю доходность. В [1] в качестве критериев ранжирования используются различные отношения, учитывающие доходность и риск.

В данной работе применялись для ранжирования три критерия. Целью работы было определение наиболее эффективного критерия для краткосрочного периода владения, равного 30 дням.

1. Постановка задачи

Рассматриваемая инвестиционная стратегия состоит в следующем:

1. Ранжирование активов по выбранному критерию.
2. Разделение активов на «хорошие» и «плохие» согласно значению критерия.

3. Одновременная продажа (короткая позиция) «плохих» активов на определённую сумму и покупка (длинная позиция) на эту же сумму «хороших» активов.
4. Формирование двух портфелей «хорошего» и «плохого» из соответствующих активов.
5. Определение дохода инвестиционной стратегии в конце периода владения, как разность стоимостей «хорошего» и «плохого» портфелей.

На рис. 1 показаны моменты времени и используемые термины рассматриваемого алгоритма.

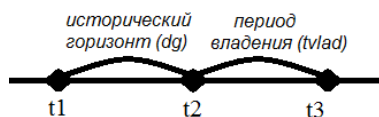


Рис. 1

Интервал $[t_1, t_2]$ будем называть историческим горизонтом, $[t_2, t_3]$ – периодом владения.

2. Ранжирование активов

Обозначим через $r_{i,j}$ доходность j -го актива в i -й момент времени. Рассмотрим массив доходностей на выбранном историческом горизонте. Для каждого актива вычислим значение выбранного критерия. Ранжирование активов проводится по трём критериям:

Критерий среднего. Для j -го актива среднее значение дневной доходности на историческом горизонте обозначим через Sr_j :
$$Sr_j = \frac{1}{t_2 - t_1} \sum_{i=t_1}^{t_2} r_{i,j}.$$

Критерий Шарпа. Для j -го актива критерий Шарпа обозначим через $Sharp_j$ $Sharp_j = \frac{Sr_j}{\sigma_j}$, где σ_j – среднее квадратичное отклонение j -го актива на историческом горизонте.

Критерий Farinelli – Tibiletti. Массив доходностей j -го актива делится на два массива r_j^+ , r_j^- . Доходность $r_{i,j} \in r_j^+$, если $r_{i,j} \geq 0$, и $r_{i,j} \in r_j^-$, если $r_{i,j} \leq 0$. Критерий Farinelli – Tibiletti для j -го актива есть отношение средних значений модулей двух массивов r_j^+ , r_j^- : $FT_j = \frac{M\{r_j^+\}}{M\{r_j^-\}}$, где M – символ математического усреднения.

3. Формирование двух портфелей

В описываемой стратегии используем десять активов. Динамика цен приведена на рис. 2.

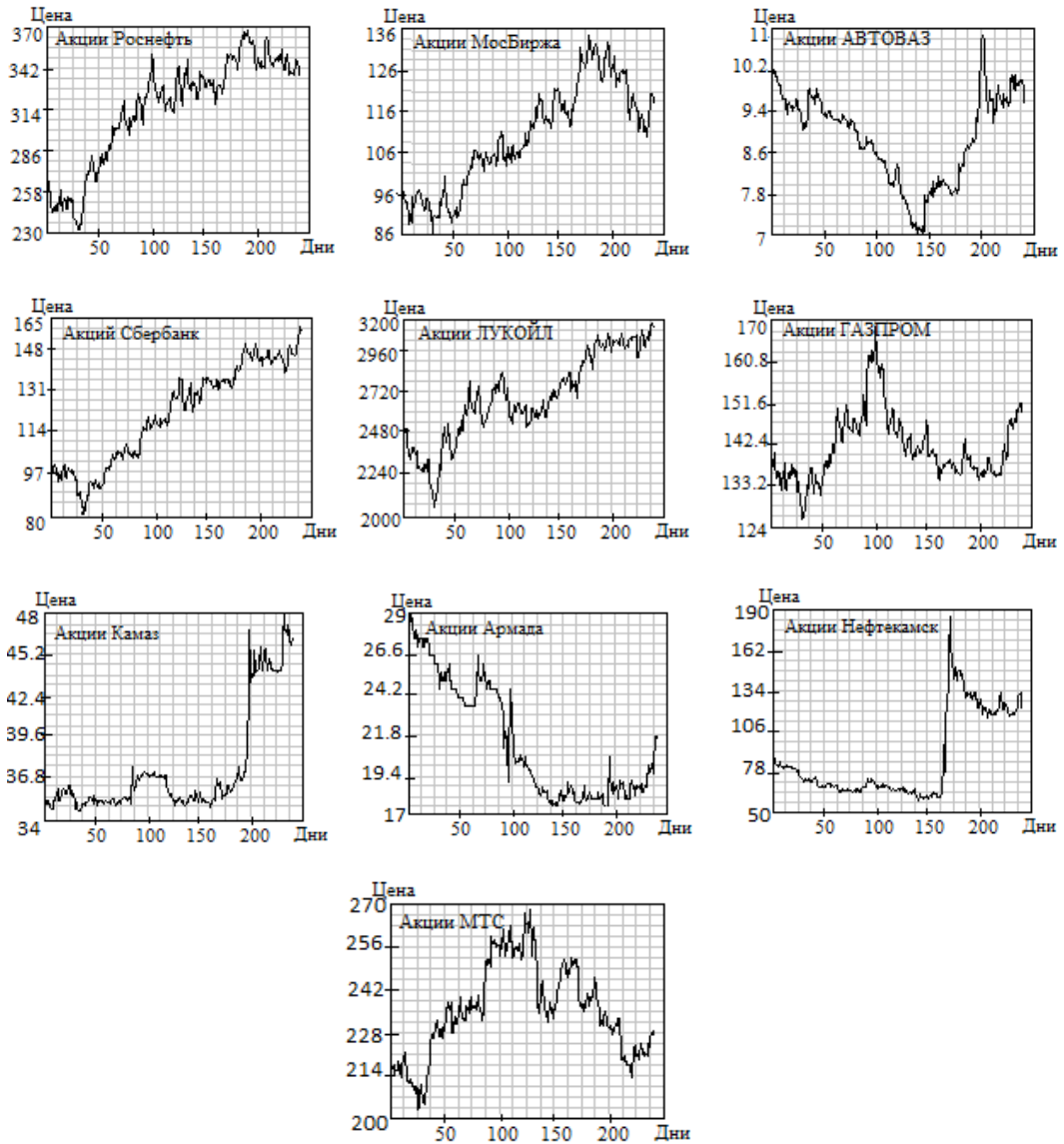


Рис. 2. Динамика цен активов

В момент t_2 начала инвестирования для выбранного критерия имеем ранжированный на историческом горизонте массив активов. Ранжирование проводили в порядке убывания значений критерия. Первые три актива в ранжированном массиве назовем «хорошими», последние три – «плохими». Построим из «хороших» и «плохих» активов равновзвешенные портфели со структурой $x = \begin{pmatrix} 1 & 1 & 1 \\ 3 & 3 & 3 \end{pmatrix}^T$. Обозначим через w_i^g , w_i^b капитал в i -й момент времени портфеля из «хороших» и «плохих» активов соответственно. В начале периода владения $w_0^g = w_0^b$. Для получения динамики двух портфелей необходимо сформировать матрицу доходностей «хороших» и «плохих» активов на интервале владения $[t_2, t_3]$, обозначим их через r^g , r^b соответственно, а доходность портфеля из «хороших» активов и доходность портфеля из «плохих» активов получим по

формулам $rp^g = r^g x$, $rp^b = r^b x$. Капитал двух портфелей на интервале владения вычислим по формулам: $w_{i+1}^g = w_i^g (1 + rp_i^g)$, $w_0^g = w_0$, $w_{i+1}^b = w_i^b (1 + rp_i^b)$, $w_0^b = w_0$.

На рис. 3. приведены изменения капитала портфеля из «хороших» активов и «плохих».



Рис. 3: Капитал портфеля

Оптимальный портфель получим по следующему алгоритму. После ранжирования активов и разделения их на «хорошие» и «плохие» были решены две задачи оптимизации. Находили структуру портфеля из «хороших» активов, которая максимизировала выбранный критерий, а из плохих активов находили структуру портфеля, которая минимизировала критерий. Задачи оптимизации решали с помощью решающих блоков в MathCAD. На рис. 4 приведены доходы инвестиционной стратегии для одного периода владения равновзвешенным и оптимальным портфелями.

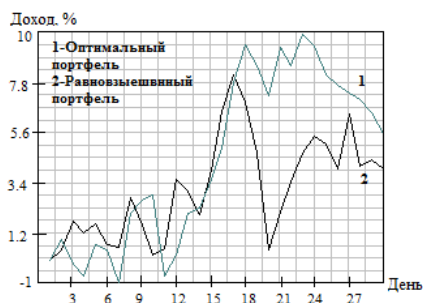


Рис. 4. Доход портфеля

4. Переформирование портфелей

В рассматриваемом алгоритме период владения портфеля равен 30 дням. В течение этого периода структура портфелей сохраняется. В конце периода владения отдаём долг (стоимость портфеля из «плохих» активов), а доход будет равен разности стоимостей портфелей из «хороших» и «плохих» активов. Затем процедура повторяется: ранжирование активов на новом историческом горизонте, разделение активов на «хорошие» и «плохие», построение двух портфелей. На рис. 5 показано изменение дохода стратегии.

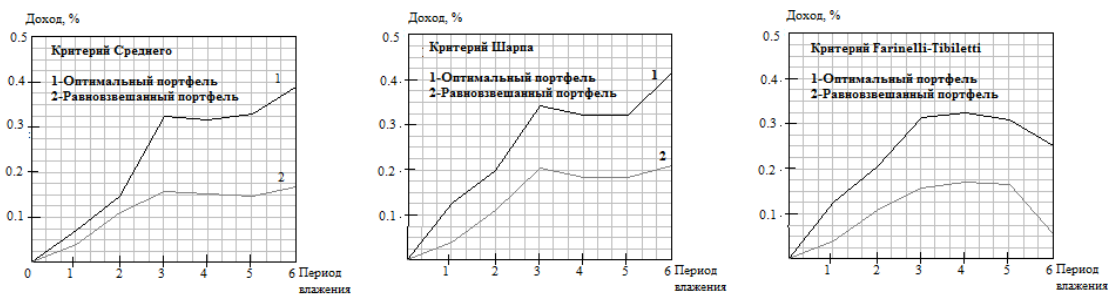


Рис. 5

5. Сравнение численных результатов

Результаты моделирования представлены в табл. 1.

Таблица 1

Доход инвестиционной стратегии						
Начальный момент	Равновзвешанный			Оптимальный		
	Mean	Шарпа	FT	Mean	Шарпа	FT
50	0,04	0,04	0,040	0,071	0,125	0,125
80	0,109	0,109	0,108	0,145	0,199	0,204
110	0,157	0,204	0,156	0,323	0,340	0,313
140	0,15	0,183	0,168	0,316	0,321	0,323
170	0,146	0,183	0,165	0,327	0,321	0,306
200	0,166	0,208	0,054	0,387	0,413	0,249

Из табл. 1 видно, что за шесть периодов владения и пять перестроек структур портфелей доход равновзвешенного портфеля, построенного по критерию Среднего, равен 16.6%, по критерию Шарпа – 20.8% и по критерию Farinelli – Tibiletti 5.4%. Доход оптимального портфеля, построенного по тем же самым критериям, равен для критерия Среднего 38.7%, для критерия Шарпа – 41.3%, для Farinelli – Tibiletti 24.9%.

Заключение

В результате проведённого моделирования оказалось, что наиболее эффективной инвестиционной стратегией является стратегия, использующая критерий ранжирования Шарпа. Этот результат справедлив как для равновзвешенного портфеля, так и для оптимального.

ЛИТЕРАТУРА

1. Biglova A., Jasic T., Rachev S., Fabozzi F. Profitability of Momentum Strategies: Application of Novel Risk // Investment Management and Financial Innovations. – 2004. – V. 4. – P. 47–62.

АНАЛИЗ ЭКОНОМИЧЕСКОГО РОСТА В РАСШИРЕННОЙ МОДЕЛИ СОЛОУ С ЧЕЛОВЕЧЕСКИМ КАПИТАЛОМ

Д.У. Эшмурадов

Томский государственный университет

daur.eshmuradov@mail.ru

Введение

Вопросы анализа экономического роста относятся к актуальным задачам макроэкономики. Для анализа экономического роста используются различные модели. Базовой является модель Солоу [2]. В классической постановке модель Солоу не объясняет причин различий экономического роста разных стран. Поэтому в последнее время экономисты используют расширенные модели Солоу для анализа проблем экономического роста.

В данной работе рассматривается задача анализа экономического роста в расширенной модели Солоу с человеческим капиталом. Цель работы – показать существование стационарной траектории в экономике и выявить влияние различных факторов на экономический рост.

1. Постановка задачи

Рассмотрим следующую модель закрытой экономики с физическим и человеческим капиталом [1], в которой K – объём основных фондов, H – человеческий капитал, L – трудовые ресурсы, A – объём знаний. Данная модель является двухсекторной: первый сектор – производственный, который обеспечивает выпуск благ, и второй – образовательный.

В этой модели предполагается, что человеческий капитал производится в своем собственном секторе со своей собственной производственной функцией. Люди (L) важны только в качестве субъектов обучения, а не в качестве прямых факторов производства товаров. Начальные значения факторов производства заданы.

Модель строится в непрерывном времени. Предполагается, что выпуск Y в момент времени t определяется производственной функцией Кобба – Дугласа [3], которая имеет вид:

$$Y(t) = [(1 - a_K)K(t)]^\alpha [(1 - a_H)H(t)]^{1-\alpha}, \quad (1)$$
$$0 < \alpha < 1, 0 < a_K < 1, 0 < a_H < 1,$$

где a_K и a_H – доли физического и человеческого капитала, задействованные в образовательной отрасли, α – эластичность выпуска по капиталу.

Динамика капитала определяется уравнением $\dot{K}(t) = sY(t) - \delta_K K(t)$, где s – норма сбережений, а δ_K – амортизация капитала.

Динамика человеческого капитала определяется производственной функцией нейтрального по Харроду научно-технического прогресса с учётом амортизации с нормой δ_H :

$$\dot{H}(t) = B[a_K K(t)]^\gamma [a_H H(t)]^\varphi [A(t)L(t)]^{1-\gamma-\varphi} - \delta_H H(t), \quad \gamma > 0, \varphi > 0, \gamma + \varphi < 1.$$

Знания и труд растут с постоянными темпами: $\dot{L}(t) = nL(t)$, $\dot{A}(t) = gA(t)$, где n и g заданы экзогенно.

2. Определение стационарной траектории

Поставленную задачу будем решать в удельных переменных: $k = \frac{K}{AL}$, $h = \frac{H}{AL}$, где k и h – объём физического и человеческого капитала, приходящегося на единицу эффективного труда (AL) [1] соответственно. Отсюда следует, что $K = kAL$, $H = hAL$.

Выведем уравнение для \dot{k} :

$$\begin{aligned}\dot{k} &= \left(\frac{K}{AL} \right)' = \frac{K'AL - K(AL)'}{(AL)^2} = \frac{K'AL}{(AL)^2} - \frac{K(A'L + AL')}{(AL)^2} = \frac{K'}{AL} - \frac{KA'L + KAL'}{(AL)^2} = \\ &= \frac{K'}{AL} - \frac{K}{AL} \frac{A'L}{AL} - \frac{K}{AL} \frac{AL'}{AL} = \frac{K'}{AL} - kg - kn = \frac{sY(t) - \delta_K K(t)}{AL} - kg - kn = \\ &= s \frac{Y(t)}{AL} - \delta_K k - kg - kn = s \frac{Y(t)}{AL} - (\delta_K + g + n)k = \\ &= s \frac{[(1-a_K)K(t)]^\alpha [(1-a_H)H(t)]^{1-\alpha}}{AL} - (\delta_K + g + n)k = \\ &= s \frac{[(1-a_K)kAL]^\alpha [(1-a_H)hAL]^{1-\alpha}}{AL} - (\delta_K + g + n)k = \\ &= s \frac{[(1-a_K)k]^\alpha [(1-a_H)h]^{1-\alpha} AL^{\alpha+1-\alpha}}{AL} - (\delta_K + g + n)k = \\ &= s [(1-a_K)k]^\alpha [(1-a_H)h]^{1-\alpha} - (\delta_K + g + n)k.\end{aligned}$$

Таким образом, $\dot{k} = s(1-a_K)^\alpha (1-a_H)^{1-\alpha} k^\alpha h^{1-\alpha} - (\delta_K + g + n)k$.

Введём обозначение: $c_K = (1-a_K)^\alpha (1-a_H)^{1-\alpha}$. В результате получаем уравнение:

$$\dot{k} = sc_K k^\alpha h^{1-\alpha} - (\delta_K + g + n)k. \quad (2)$$

Получим уравнение для \dot{h} .

$$\begin{aligned}\dot{h} &= \left(\frac{H}{AL} \right)' = \frac{H'AL - H(AL)'}{(AL)^2} = \frac{H'AL}{(AL)^2} - \frac{H(A'L + AL')}{(AL)^2} = \frac{H'}{AL} - \frac{HA'L + HAL'}{(AL)^2} = \\ &= \frac{H'}{AL} - \frac{H}{AL} \frac{A'L}{AL} - \frac{H}{AL} \frac{AL'}{AL} = \frac{H'}{AL} - hg - hn = \\ &= \frac{B[a_K K(t)]^\gamma [a_H H(t)]^\varphi [A(t)L(t)]^{1-\gamma-\varphi} - \delta_H H(t)}{AL} - kg - kn = \\ &= \frac{B[a_K k]^\gamma [a_H h]^\varphi [A(t)L(t)]^{1-\gamma-\varphi+\gamma+\varphi}}{AL} - (\delta_H + g + n)h = \\ &= B[a_K k]^\gamma [a_H h]^\varphi - (\delta_H + g + n)h.\end{aligned}$$

Таким образом, $\dot{h} = Ba_K^\gamma a_H^\varphi k^\gamma h^\varphi - (\delta_H + g + n)h$.

Введём обозначение: $c_H = Ba_K^\gamma a_H^\varphi$. В результате получаем уравнение:

$$\dot{h} = c_H k^\gamma h^\varphi - (\delta_H + g + n)h. \quad (3)$$

Для определения существования стационарной траектории получим уравнения, описывающие множество комбинаций k и h таких, что $\dot{k} = 0$ и $\dot{h} = 0$.

Из уравнения (2) при $\dot{k} = 0$ получим $sc_K k^\alpha h^{1-\alpha} - (\delta_K + g + n)k = 0$, отсюда $k^{1-\alpha} = \frac{sc_K h^{1-\alpha}}{(\delta_K + g + n)}$, или

$$k = \left[\frac{sc_K}{(\delta_K + g + n)} \right]^{\frac{1}{1-\alpha}} h. \quad (4)$$

Для анализа вида зависимости k от h найдём первую и вторую производную функции $k(h)$ (4): $k' = \frac{1}{1-\alpha} \left[\frac{sc_K}{(\delta_K + g + n)} \right] > 0$, $0 < \alpha < 1$. Первая производная больше нуля, т.к. $0 < \alpha < 1$, а все остальные переменные положительные, следовательно, $k(h)$ – функция возрастающая.

Вторая производная $k'' = 0$ равна нулю, следовательно, $k(h)$ – линейная функция, которая в пространстве (k, h) выглядит следующим образом:

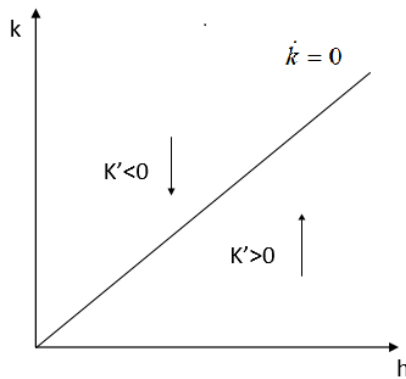


Рис 1.

На рис. 1 стрелками указаны направления возможности выхода на стационарную траекторию $\dot{k} = 0$.

Получим уравнение, описывающее множество комбинаций k и h таких, что $\dot{h} = 0$. Из уравнения (3) получим $c_H k^\gamma h^\varphi - (\delta_H + g + n)h = 0$, отсюда $k^\gamma = \frac{\delta_H + g + n}{c_H} h^{1-\varphi} = 0$, следовательно,

$$k = \left[\frac{\delta_H + g + n}{c_H} \right]^{\frac{1}{\gamma}} h^{\frac{1-\varphi}{\gamma}} = 0. \quad (5)$$

Для анализа вида зависимости k от h (5) найдём первую и вторую производную функции $k(h)$.

$$k' = \frac{1-\varphi}{\gamma} \left[\frac{\delta_H + g + n}{c_H} \right]^{\frac{1}{\gamma}} h^{\frac{1-\varphi-\gamma}{\gamma}} > 0.$$

Производная больше нуля, так как $0 < \alpha < 1$, $\varphi > 0$, $\gamma + \varphi < 1$, следовательно, функция $k(h)$ – возрастающая.

Найдём вторую производную, чтобы определить выпуклость функции.

$$k'' = \frac{1-\varphi-\gamma}{\gamma} \frac{1-\varphi}{\gamma} \left[\frac{\delta_H + g + n}{c_H} \right]^{\frac{1}{\gamma}} h^{\frac{1-\varphi-2\gamma}{\gamma}} > 0.$$

Вторая производная больше нуля, следовательно, функция является выпуклой вниз.

В пространстве (k, h) график функции $k(h)$ выглядит следующим образом:

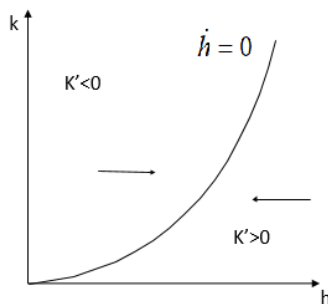


Рис. 2

На рис. 2 стрелками указаны направления возможности выхода на стационарную траекторию $\dot{h} = 0$.

Совместим графики рис. 1 и рис. 2 на одном рисунке:

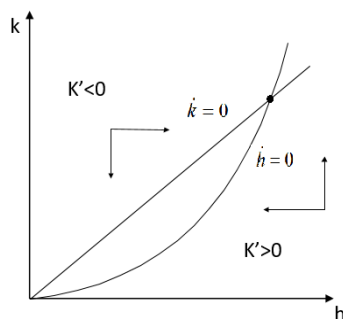


Рис. 3

Графики функций $k(h)$ (4) и (5) пересекаются в единственной точке, следовательно, стационарная траектория существует и является единственной.

3. Анализ влияния различных факторов на выпуск

Поскольку аналитически оценить влияние факторов на выпуск (1) затруднительно, проведём численное исследование влияния долей физического и человеческого капитала a_K и a_H , задействованных в образовательной отрасли, на изменение объёма выпуска для следующего иллюстративного примера.

Пусть $\alpha = 0.3$, $\gamma = 0.3$, $\varphi = 0.3$, $B = 1$, $g = 0.03$, $n = 0.03$, $\delta_K = 0.12$, $\delta_H = 0.12$.

Норму сбережений s , а также a_K и a_H зададим экзогенно: $s = 0.6$, $a_K = 0.1$, $a_H = 0.1$.

Запишем производственную функцию (1) в интенсивной форме:

$$y = f(k, h) = F\left(\frac{K}{AL}, \frac{H}{AL}\right) = \frac{[(1-a_K)K(t)]^\alpha [(1-a_H)H(t)]^{1-\alpha}}{AL} =$$

$$= [(1-a_K)k]^\alpha [(1-a_H)h]^{1-\alpha}, \quad (6)$$

где y – выпуск на единицу эффективного труда.

Для анализа зависимости выпуска y от a_K и a_H необходимо получить зависимость $y(k, h)$ во времени, для этого получим решение системы дифференциальных уравнений:

$$\begin{cases} \dot{k} = sc_K k^\alpha h^{1-\alpha} - (\delta_K + g + n)k, \\ \dot{h} = c_H k^\gamma h^\varphi - (\delta_H + g + n)h. \end{cases} \quad (7)$$

В предлагаемой работе решение системы (7) определяется с использованием встроенной функции *ode45* пакета MATLAB, которая реализует метод Рунге – Кутты четвёртого порядка.

Найденное решение системы (7), используем для построения графика функции $y(t)$ (6). С помощью данного графика будем анализировать, как влияют доли a_K и a_H на объём выпуска. Построим график $y(t)$ при $a_K = 0.1$, $a_H = 0.1$. На рис. 4 это изображено непрерывной линией. Предположим, что в момент времени $t = 20$ произошло изменение доли физического капитала в образовательном секторе $a_K = 0.15$ при $a_H = 0.1$ (результат – пунктирная линия). Аналогично при $a_K = 0.1$, $a_H = 0.15$ получаем график, изображённый штрихпунктирной линией. Результаты совместим на одном рис. 4.

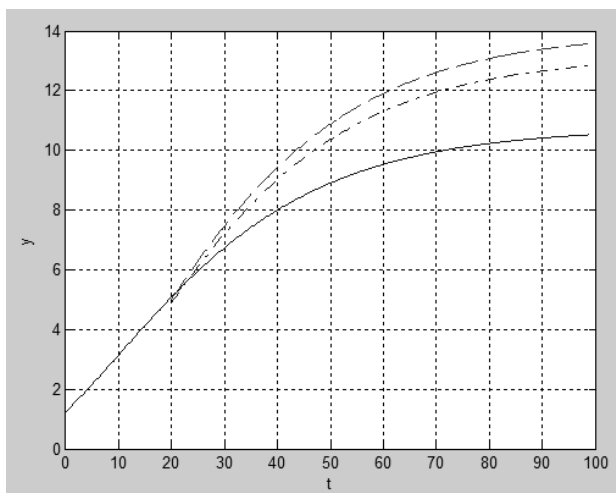


Рис. 4

Из графиков видно, что увеличение долей физического и человеческого капитала a_K и a_H в образовательном секторе увеличивает выпуск (ВВП). При этом выпуск более чувствителен к изменению a_K , т.к. при изменении на одну и ту же величину, доля a_K обеспечивает больший прирост выпуска, чем увеличение a_H . Таким образом, доля физического капитала в образовательном секторе влияет на выпуск больше, чем доля человеческого капитала в образовательном секторе.

Проведём анализ влияния нормы амортизации физического капитала δ_K и человеческого капитала δ_H на изменение объёма выпуска.

Построим $y(t)$ при $\delta_K = 0.12$, $\delta_H = 0.12$. Предположим, что в момент времени $t = 20$ произошло изменение либо δ_K , либо δ_H . Для этих двух ситуаций построим гра-

фики $y(t)$ и сравним с результатом, когда δ_K и δ_H не менялись. Для того, чтобы посмотреть, что влияет на выпуск сильнее, допустим, что δ_K и δ_H изменились на одинаковую величину. Пусть $\delta_K = 0.15$ в первом случае и $\delta_H = 0.15$ во втором. Результаты совместим на одном рисунке.

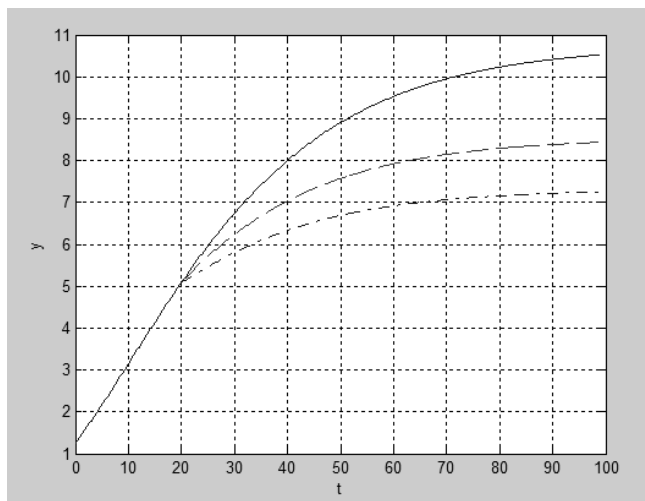


Рис. 5

Результат моделирования при $\delta_K = 0.12$, $\delta_H = 0.12$ изображён графиком с непрерывной линией, при $\delta_K = 0.15$, $\delta_H = 0.12$ – пунктирной, а $\delta_K = 0.12$, $\delta_H = 0.15$ – штрихпунктирной.

Оказывается, что амортизация человеческого капитала сильнее влияет на выпуск чем амортизация физического капитала.

Заключение

В данной работе была рассмотрена задача анализа экономического роста в расширенной модели Солоу с человеческим капиталом. Было доказано существование стационарной траектории для данной модели, и на стационарной траектории проведён анализ влияния долей физического и человеческого капитала, а также влияния нормы амортизации физического капитала и человеческого капитала на изменение объёма выпуска.

Было установлено, что доля физического капитала в образовательном секторе влияет на выпуск больше, чем доля человеческого. Это может означать, что страны, в которых образовательная отрасль финансируется больше, в итоге получают больший прирост к выпуску. С другой стороны, было выявлено, что норма амортизации человеческого капитала влияет сильнее на выпуск, чем норма амортизации физического, следовательно, можно сделать вывод, что человеческий капитал следует обновлять быстрее, чем физический. Этим можно объяснить различие в доходах между странами с развитым и слабым образовательным сектором. Там, где образовательная отрасль хорошо развита, как правило, имеется более высокий уровень доходов, чем там, где она недостаточно развита.

ЛИТЕРАТУРА

1. *Romer P.* Advanced Macroeconomics. 2-nd ed. – McGraw-Hill, NewYork, 2001.
2. *Solow R.* Growth Theory. – Oxford University Press, 2000.
3. *Шапаев Ю.В.* Теория экономического роста: учеб. Пособие для вузов. – М.: Изд. дом ГУ ВШЭ, 2006.

ПРОЕКТ КОРПОРАТИВНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ПОЛЁТНЫМ РАСПИСАНИЕМ АВИАКОМПАНИИ

Н.Т. Дарибаева

Оренбургский государственный университет
nurz_dar7@mail.ru

Введение

Авиационная индустрия – область, которая не стоит на месте и прорывы в обработке корпоративной информации могут дать колоссальные конкурентные преимущества. Большое внимание уделяется процессу автоматизации составления расписания и планирования. Расписание представляет собой один из основных продуктов авиакомпании и, безусловно, ведущий фактор в выборе пассажиром конкретного перевозчика. Работа специалиста по управлению расписанием может быть одной из самых сложных в любой авиакомпании. Планирование – одна из самых важных функций в бизнесе, такое же важное, как прогнозирование, ценообразование, назначение флота или финансирование. В данной работе представлен проект корпоративной информационной системы управления полетным расписанием авиакомпании.

1. Входные и выходные данные информационной системы

Миссия планирования столь же широка, как и миссия самой авиакомпании, потому что авиакомпания несёт ответственность за предоставление надлежащего обслуживания городам, которым она служит. Авиакомпаниям также должны действовать эффективно и экономично. Помимо общественного сервиса и экономических аспектов, планирование должно быть сбалансировано с другими факторами, такими как:

1. *Техническое обслуживание оборудования.* Отдельный план техобслуживания должен быть составлен для каждого типа воздушного судна. Все планы маршрутизации должны быть скоординированы для обеспечения лучшего сервиса. Обслуживание самолетов требует, чтобы определенные станции были обеспечены оборудованием и персоналом для периодических механических проверок. Концентрация обслуживания желательна только на нескольких станциях, и желательно также использование в полной мере выделенных средств, обеспечивающих равномерное техобслуживание.

2. *Экипажи.* Необходимо наличие равномерного графика работы экипажа (пилотов, инженеров и бортпроводников).

3. *Услуги.* Оборудование в аэропорту должно отвечать всем требованиям. Мощность терминала, включая счетчики билетов, зоны обработки багажа и залы ожидания должны отвечать растущим требованиям рынка. Мощность аэропорта, включая взлетно-посадочные полосы, рулежные дорожки и навигационные средства, устанавливают верхний предел операций.

4. *Маркетинговые факторы.* Факторы маркетинга многочисленны, включая такие характеристики, как размер рынка, продолжительность полета, часовые пояса и близость аэропорта.

5. *Другие факторы.* Сезонные изменения ветра требуют разницы в летнее и зимнее время на определенных маршрутах (обычно с востока на запад); однако некоторые авиакомпании используют постоянные круглогодичные полеты на маршрутах, где изменения в компонентах ветра являются незначительными (обычно маршруты с севера на юг). Кроме того, во многих сегментах изменение графика используется, чтобы в некоторой степени разрешить ожидаемые задержки в периоды загруженного трафика.

В предлагаемой корпоративной информационной системе должны быть учтены все эти факторы. На рис. 1 представлены входные и выходные данные системы.



Рис. 1. Входные и выходные данные информационной системы

Данные, изображённые на схеме являются входными данными системы. Изначально составляется расписание с грубыми предположениями, затем учитываются внутренние и внешние факторы. Внутренние факторы были описаны выше. Нельзя назначать несколько рейсов на посадку в одно и то же время, если аэропорт обладает небольшой взлетно-посадочной полосой, в противоположность этому, если же это большой столичный аэропорт, нельзя допускать, чтобы он часто пустовал. Наличие базы технического обслуживания берётся в учет, а также время, когда воздушные судна могут пройти этапы от рутинной проверки вплоть до капитального ремонта. Внешние факторы также должны приниматься во внимание отделом планирования. Почтовые службы и логистические компании имеют собственные предпочтения в расписании, руководство аэропорта старается найти плавный поток трафика для оптимизации использования объектов и препятствует достижению часа-пик. На практике крупные аэропорты назначают слоты – время, в течение которого оно должно совершить взлет или посадку. Слот выдается диспетчерской службой аэропорта или государственным органом, время вылета указывается либо фиксировано, либо в определённом интервале. Слоты распределяются в соответствии с правилами, установленными службой планирования полетов Международной ассоциации воздушного транспорта, ИАТА (International Air Transport Association). Гостиницы и отели предпочитают, чтобы гости заселялись и выселялись между 11:00 утра и 01:00 ночи.

2. Состав корпоративной информационной системы (КИС)

В разрабатываемой корпоративной системе должны тесно взаимодействовать процессы прогнозирования спроса, планирования расписания, управления доходностью (ERP-система) и сбора информации о предпочтениях и потребностях своих клиентов (CRM-система).

В общем виде задачу составления расписания можно представить в виде схемы (рис. 2). Как видно из рисунка, в предлагаемом проекте корпоративной информационной системы представлены три подсистемы: подсистема планирования расписания, управления доходами и подсистема управления взаимоотношениями с клиентами.

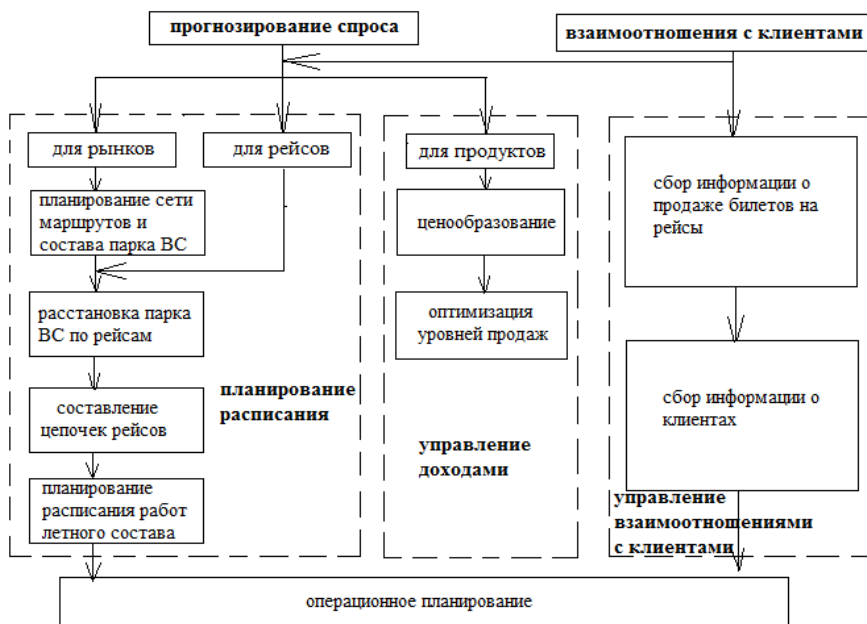


Рис. 2. Структура задач КИС управления расписанием авиаперевозок

Подсистема планирования расписания (для рейсов и рынков) содержит четыре модуля:

1. Модуль планирования сети маршрутов и состава парка воздушных судов предназначен для составления сети маршрутов и структуры парка ВС, которые обеспечили бы производство оптимального объема авиатранспортной продукции. Производится оптимизация парка воздушных судов с учетом комплекса по обслуживанию определенного типа воздушного судна, маршрутов следования, степени загрузки, производственных затрат. При этом должна быть достигнута максимальная эксплуатационная прибыль при условиях несовершенной конкуренции и существующих ограничениях ресурсов. Для решения такой сложной системной задачи используются теория оптимизации сетей, методы линейного и динамического программирования, статистико-экономический подход.

2. Модуль расстановки парка ВС с учетом данных, полученных в первом модуле. Производят выбор по экономическим показателям и определяют, на какой рейс лучше всего поставить тот или иной тип воздушного судна. Эту задачу решают одновременно с задачей расстановки самих рейсов по времени. На время вылета и прилета существенно влияют ограничения в виде слотов, а качество расписания измеряется суммарной долей на обслуживаемых рынках.

3. Модуль составления цепочек рейсов предназначен для определения последовательности выполнения рейсов для каждого типа воздушного судна, разрабатывается график оборота ВС. Главным критерием при составлении цепочки рейсов является условие эффективности коммерческой эксплуатации ВС при выполнении всех видов технического обслуживания.

4. Модуль планирования расписания работ летного состава. На основе результатов второго и третьего модуля разрабатывается график работы летного состава по критерию минимизации оплачиваемого времени, которое не используется для выполнения полетов.

Задачей подсистемы управления доходами является определение оптимального объема авиатранспортной продукции на каждой линии и тарифа. Эта задача сводится к задаче оптимального управления продажами. Путем сегментации спроса максимизиру-

ется доход путём разделения покупателей на некоторые классы и назначения цены для каждого из них. Модули ценообразования и планирования сети маршрутов и состава парка ВС тесно взаимодействуют между собой: результаты модуля ценообразования влияют на процесс определения оптимальной сети маршрутов, и наоборот.

Подсистема управления взаимоотношениями с клиентами (CRM-система) служит связующим звеном между внутренней и внешней средой, т.е. между предлагаемыми и оказываемыми услугами авиакомпании и их конечными потребителями. Данная подсистема включает процессы сбора, хранения и анализа информации о клиентах, партнерах. Управление продажами осуществляется на сайте авиакомпании, где любой пассажир может купить билет «онлайн» и зарегистрироваться на рейс; также система включает фронтальную часть, обеспечивающую обслуживание клиентов на точках продаж. В состав подсистемы включены специализированные маркетинговые блоки программ для ведения электронной коммерции, позволяющие отслеживать поведение клиента на сайте в режиме реального времени. Также необходимо включить управление клиентским обслуживанием и колл-центрами, т.е. системы по обработке обращений клиентов.

Заключение

В настоящей работе представлен проект корпоративной информационной системы управления полетным расписанием. Описаны входные и выходные данные, а также модули, входящие в состав системы. Разбиение системы на модули позволяет покупателям данного продукта приобретать и внедрять только те части, которые необходимы. Это не только позволяет избежать излишних затрат, но еще и помогает не перегружать систему.

СЕКЦИЯ VI. ГЕОИНФОРМАТИКА. ВЕБ ПРИЛОЖЕНИЯ

ИССЛЕДОВАНИЕ И РЕАЛИЗАЦИЯ ЭФФЕКТИВНЫХ СРЕДСТВ УПРАВЛЕНИЯ ВЕБ-КОНТЕНТОМ

Р.А. Аношкин

Томский государственный университет

reverse.wvcore@gmail.com

Введение

Одной из главных составляющих сетевых технологий являются web-серверы, главной задачей которых является предоставить контент (разметку HTML) для отображения информации в браузерах. Второстепенной задачей считается использование web-сервера в виде сервиса для обслуживания различного рода приложений.

Большинство web-серверов работают на базе стека LAMP (Linux, Apache, MySQL/MariaDB[1], PHP). При обработке запроса пользователя, в большинстве реализаций, наблюдается существенная просадка производительности на этапе работы PHP.

PHP [2] — это интерпретируемый язык программирования. Весь процесс, начиная от трансляции php-кода в опкоды (operation codes), представляющие собой набор последовательных операций библиотеки php, достаточно трудоемкий. Большинство современных систем еще сильнее замедляют его работу наличием большого количества объектов глубокого уровня зависимостей, выполнением множества неоптимальных, порой даже лишних запросов в базу данных, лишними обращениями к файловой системе.

Исходя из данных проблем следуют три основные задачи:

1. Оптимизация работы с базой данных.
2. Реализация системы кеширования данных.
3. Управляемый шаблонизатор.

1. База данных

Для реализации оптимального способа общения с базой данных стоит определить с типами запросов и возможными вариантами их оптимизации [3]. Опытным путём было определено три типа запросов:

1. Запросы, зависимые от действий пользователя. Такие запросы включают в себя динамические и статические результаты выполнения. К динамическим относятся запросы наподобие «Выбрать 10 сообщений, начиная с 30 позиции», что могло бы определяться как четвёртая страница новостей. Статические запросы – это запросы состояния пользователя, например, запросы на получение данных его учётной записи, которые либо не изменяются совсем, либо изменяются крайне редко.

Для оптимизации выбран метод объединения динамических запросов средствами оператора UNION MySQL и JSON сериализация статических запросов и их последующее хранение на жёстком диске, либо в памяти приложения.

2. Статические запросы, независимые от действий пользователя. К таким запросам относятся запросы на получение свойств приложения, настройки ресурса, настраиваемые права доступа, данные, которые изначально удобнее хранить в базе данных.

Для их оптимизации использован ранее описанный способ хранения результатов в виде JSON объектов.

3. Часто выполняемые динамические запросы, независимые от действий пользователя. К таким запросам относятся запросы на получения статистики, выборка состоя-

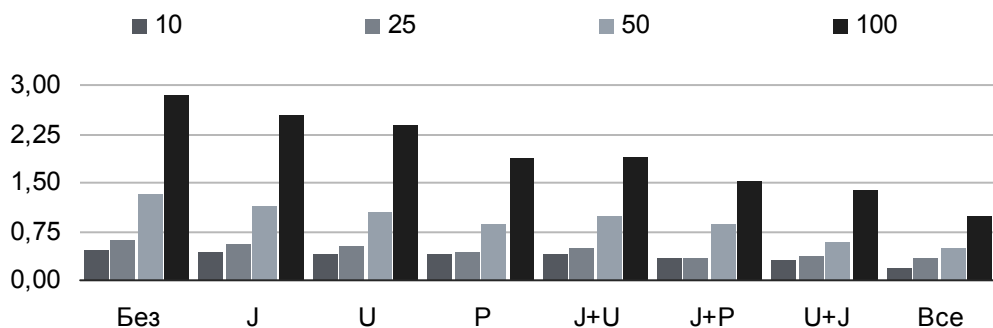
ний других пользователей, выбор списка доступных в данный момент товаров и прочее.

Для такого вида запросов единственным способом оптимизации является сокращение времени разбора выражений базой данных, откуда следует решение в виде использования Prepared Statements (Подготавливаемые выражения), что и является их основной задачей.

Ниже приведена статистика времени обработки данных в зависимости от количества запросов. Задача поставлена следующим образом: каждый третий запрос является зависимым от пользователя, каждый пятый запрос подходит под критерий кеширования, все остальные запросы можно классифицировать как подготавливаемые.

	10	25	50	100
Без модификаций	0,452	0,624	1,328	2,845
JSON	0,415	0,562	1,128	2,526
UNION	0,410	0,530	1,054	2,385
PREPARED	0,387	0,424	0,860	1,876
JSON + UNION	0,392	0,497	0,989	1,893
JSON + PREPARED	0,342	0,351	0,864	1,527
UNION + PREPARED	0,319	0,363	0,589	1,372
Все модификации	0,137	0,324	0,495	0,985

При тестировании использована база данных MariaDB (GNU GPL версия баз данных семейства MySQL) с выключенным кэшированием и десятью таблицами, заполненными случайными данными на 20 атрибутов и 100 кортежей, на одном и том же компьютере на базе ОС Debian с отключенными сторонними сервисами и низкой общей нагрузкой системы. Для каждого показателя взято среднее значение трёх тестирований.



Программной реализацией данного решения служит подготовка результата запросов перед выполнением логических скриптов приложения.

В процессе разработки, параллельно с разработкой скрипта выполнения модуля или контроллера, готовится отдельный скрипт с описанием запросов к базе данных. Описание этого скрипта сводится к функциональной реализации:

```
db()->query('name1', 'SELECT * FROM `t1` WHERE ...', JSON);
db()->query('name2', 'SELECT * FROM `t2` WHERE ...', UNION);
```

```
db()->query('name3', 'SELECT * FROM `t3` WHERE ...', UNION);
```

что в результате дает массив полученных данных: первый элемент массива получен из JSON кэша, оставшиеся два получены из базы данных.

```
[  
  'name1' => ['col1', 'col2'],  
  'name2' => ['data1', 'data2'],  
  'name3' => ['r1', 'r2']  
]
```

Изнутри данные запросы будут разделены на два блока: первый блок отправляется в обработку JSON базы и если файл кэша не найден, возвращается пустой результат в объект управления базой данных, указатель которого доступен из функции db(), после чего запрос добавляется в блок UNION. Блок UNION компилирует все полученные запросы в единый запрос, с использованием MySQL оператора – UNION:

```
(SELECT * FROM `t1` WHERE ...)  
UNION  
(SELECT * FROM `t2` WHERE ...)
```

При этом стоит заметить, что псевдоним * в реальной системе запрещён, т.к. для оператора UNION необходимо, чтобы количество колонок совпадало во всех перечисленных таблицах. Для решения этой проблемы система реализована таким образом, что она автоматически подставляет NULL в аргументы выбора колонок таблиц так, чтобы общее количество атрибутов каждой из таблиц соответствовало максимальному среди всех запросов.

Prepared Statement запросы объявляются заранее и представляют собой словарь данных, ставящих соответствие между псевдо-именами (кодами) запросов и самими запросами. Например, это может выглядеть так:

```
[  
  'query1' => 'SELECT * FROM `t1` WHERE _order < ? AND _order > ?',  
  'name2' => 'SELECT * FROM `t2` WHERE name = ?',  
  'name3' => 'SELECT * FROM `t3` WHERE id >= ?'  
]
```

Таким образом, реализован метод:

```
db()->prepared('name3', $id);
```

где как \$id значится минимальный ключ, необходимый для вхождения выборки.

Грамотно оперируя данным функционалом, можно добиться максимальной скорости работы с базой данных.

2. Кэширование данных

Помимо оптимизации работы с базой данных, можно оптимизировать работу с файловой системой, используя кэширование, для избежания выполнения особенно «тяжелых» алгоритмов при высокой нагрузке системы пользователями [4]. Стоит заметить, что в случае атак на сервер данный метод может нести негативный характер.

Кэширование подразумевает под собой сохранение копий страниц, блоков для отображения одной и той же информации нескольким пользователям. Например, вывод

списка пользователей онлайн может быть достаточно «тяжелым» для применения данного метода. Можно было бы использовать кэширование результата базы данных, но вместо этого можно сохранить временную копию всего блока отображения списка с готовой разметкой, избегая цикла вывода информации. Исходя из этой идеи было разработано две системы кэширования страниц:

1. Автоматическая – реализует необходимый функционал для отслеживания периодичности обращения к конкретной странице пользователями, ведения статистики нагрузки этой страницы и частотой её обновления. Таким образом, автоматическая система позволяет показывать информацию в реальном времени при обращении пользователя в условиях низкой нагрузки, а в случае высокой нагрузки – выполнять весь исходный код лишь единожды за целый период нагрузки, изменения страницы. Слабым местом данной реализации может являться то, что первые недели работы системы (в зависимости от настройки) в автоматическом режиме всегда будет выполняться полная загрузка страницы для выполнения анализа. Из настроек данной системы есть два варианта – «кэшировать всё, кроме» и «кэшировать только».

2. Принудительная – в коде явно сообщается, какой блок необходимо кэшировать, с указанием времени кэширования после очередной загрузки страницы.

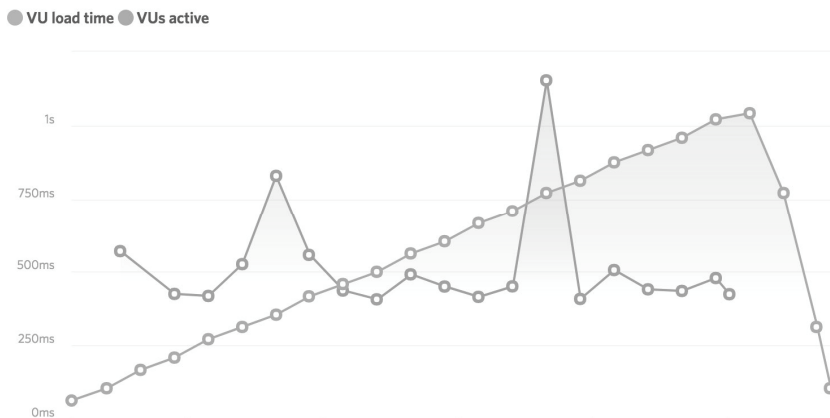
```
if(!cache()->load('cache_name', 30))
{
// предполагается нагрузки
$data = 'Hello, World!';
cache()->save('cache_name', $data, 30);
}
else
cache()->put('cache_name', $data);
```

С использованием данных способов кэширования достигается минимальное время ожидания отклика сервера.

Ниже приведена статистика времени отклика страницы с нагрузкой, создаваемой сервисом <https://loadimpact.com/> как без, так и с использованием кэширования страницы, при отключенном кэшировании запросов базы данных. Сервером нагрузки выбран сервер в Германии.

Активные пользователи	10	25	50
С кэшированием	0,57	0,407	0,423
Без кэширования	0,58	0,840	0,950

На графике ниже наглядно видны моменты времени, когда происходило обновление файлов кэша (пики времени загрузки страницы).



Важно заметить, что пересчёт кэша происходит только для одного пользователя, первого, который обратился в новый момент времени, когда старый кэш считается устаревшим.

3. Шаблонизатор

Шаблонизатор – это средство для упрощения построения HTML страницы, необходимое для разделения бизнес-логики от отображения. В первую очередь, шаблонизаторы используют для того, чтобы работа программиста и верстальщика были независимы друг от друга.

Существует множество различных шаблонизаторов, но большинство невозможно интегрировать с системой кэширования, для которой необходима полная интеграция. На данный момент создан простой шаблонизатор, предоставляющий программисту полный интерфейс для выбора шаблонов и вставку данных. Синтаксис со стороны верстальщика частично скопирован из шаблонизатора {mustache} [5]. Запланирована полная интеграция шаблонизатора {mustache} в ядро приложения.

Заключение

При правильном использовании созданных методов оптимизации базы данных, кэширования и шаблонизатора достигнуто ускорение обработки сервером запросов в два и более раз в условиях высокой нагрузки. Но стоит помнить о том, что при низких нагрузках и в случае атак данные методы могут показать обратный результат.

ЛИТЕРАТУРА

1. MariaDB [Электронный ресурс] – URL: <https://mariadb.com/>
2. PHP [Электронный ресурс] – URL: <http://php.net/>
3. Ускоряем базу данных веб-сайта [Электронный ресурс] – URL: <https://habrahabr.ru/post/116142/>
4. 11 видов кэширования для современного сайта [Электронный ресурс] – URL: <https://habrahabr.ru/company/zerotech/blog/316316/>
5. Logic-less templates [Электронный ресурс] – URL: <https://mustache.github.io/>

СОЗДАНИЕ МАСШТАБИРУЕМОЙ И ОТКАЗОУСТОЙЧИВОЙ СИСТЕМЫ ПЕРЕДАЧИ ПАКЕТОВ ДАННЫХ

С.О. Бойко

Томский государственный университет
 sergey.boiko0791@gmail.com

Введение

Основной целью создания масштабируемой и отказоустойчивой системы передачи пакетов данных является гарантированная доставка пакетов данных по надёжному каналу связи. Такие системы используются в различных областях и, как правило, пред-

ставляют собой распределённую сетевую структуру «клиент – сервер», расположенную на разных компьютерах.

Обеспечение отказоустойчивости является важной, если не главной задачей проектирования и создания серверного приложения, т.к. каждая минута неработоспособности сетевого приложения ведёт за собой большие потери для проекта, будь то количество отказавшихся от услуг пользователей или денежных средств, которые могли быть заработаны в эти минуты неработоспособности системы. Причина, по которой эта задача была, есть и будет актуальна, является человеческий фактор, присущий всем программистам, сетевым инженерам, работникам центров обработки данных («дата-центры»). Под человеческим фактором подразумевается допущение ошибок:

- 1) программисты совершают ошибки при создании программного обеспечения;
- 2) сетевые инженеры совершают ошибки при проектировании архитектуры и поддержании работоспособности сети;
- 3) работники центров обработки данных, например, электрики, могут не предусмотреть перебои с электроснабжением.

Такие упущения могут повлечь за собой выход из строя одного или нескольких элементов системы доставки пакетов данных.

Также при проектировании системы нужно предусмотреть и обеспечить способы масштабирования, т.к. при увеличении рабочей нагрузки на сервер нужно предоставить для пользования дополнительные ресурсы.

В настоящий момент основная часть разработчиков серверного программного обеспечения жертвует отказоустойчивостью и масштабируемостью в пользу сокращения времени разработки, либо выбирают что-то одно. Такой подход приемлем для небольших и простых программных комплексов, используемых в качестве временных решений, но никак не приемлем для сложных и перспективных проектов, нагрузки на которые будут со временем расти.

В данной работе представляются средства обеспечения отказоустойчивости и масштабируемости и реализованный программный комплекс – веб-чат.

Для реализации серверного комплекса использованы язык C++, POSIX-стандарты (POSIX-socket для межпроцессорного взаимодействия), библиотеки:

- «boost» – включает в себя API для разбора программных опций, подаваемых приложению при запуске, асинхронные операции над каналами связи: чтения, записи, подключения;
- «protobuf» – содержит API для компактного и удобного представления пакетов данных, отправляемых по сети;
- «MySQL Connector» – подключение и работа с базой данных MySQL;
- «jsoncpp» – разбор данных, записанных в формате JSON (текстовый формат записи данных, основанный на JavaScript).

1. Отказоустойчивость системы

Под отказоустойчивостью понимается свойство системы сохранять работоспособность при выходе из строя одного или нескольких элементов. Работоспособность в данном случае означает сохранение функционала целиком или частично, но с возможностью его полного восстановления.

Обеспечить отказоустойчивость системы можно множеством способов. Основные из них:

- 1) использование средств для автоматического запуска программного обеспечения;
- 2) использование сетевого протокола, гарантирующего доставку пакетов данных и позволяющего вовремя определить вышедший из строя элемент системы;
- 3) разделение серверной бизнес-логики на несколько частей, каждая из которых будет выполнять свой класс задач, и применение горизонтального масштабирования компонент.

1.1. Средства автоматического запуска

Средства автоматического запуска применяются в тех случаях, в которых необходимо вовремя обнаружить остановку программы (сервера), случившуюся из-за критической ошибки работы программы или из-за нехватки оперативной памяти, предпринять меры по устранению неисправности, подкорректировать конфигурации запускаемой программы, системные настройки (в случае необходимости) и запустить программу (сервер).

Утилиты, выполняющие автоматический запуск, принято называть «лаунчерами». Задача лаунчера – определить в кратчайший срок остановку подконтрольной программы и запустить программу (возможно, с заранее измененными конфигурациями), после чего (по возможности) оповестить системных администраторов или руководителей проекта.

Для реализации такого лаунчера использованы механизмы межпроцессорного взаимодействия.

В UNIX-подобных операционных системах все процессы создаются порождением от другого процесса. Порождение осуществляется созданием точной копии исходного процесса, который замещается новым процессом. В UNIX-подобных системах создать новый процесс, являющийся практически точной копией исходного, можно системным вызовом «fork», который копирует всю выделенную память родительского процесса, включая его виртуальное адресное пространство. Весь код после вызова «fork» выполняется параллельно и в родительском, и в дочернем процессе, благодаря чему родительский процесс может иметь несколько порожденных процессов.

Метод создания лаунчера на основе межпроцессорного взаимодействия заключается в том, чтобы написать программу, которая будет порождать и следить за дочерним процессом, являющимся программой-сервером.

В языке программирования С имеется реализация функции «ropen», создающая дочерний процесс и устанавливающая односторонний канал связи, но основной недостаток её использования заключается в том, что родительский процесс после вызова «ropen» останавливает свое выполнение, пока дочерний процесс не завершится. Это значит, что лаунчер может следить и запускать только один процесс. Поскольку реализуется масштабируемая система, несколько экземпляров одной компоненты могут быть запущены на одном компьютере, и использовать для каждого экземпляра программы свой лаунчер — нерационально. Поэтому была написана оригинальная реализация автоматического лаунчера, который может запускать несколько экземпляров подконтрольной программы.

Наблюдение за состоянием порожденного процесса будет происходить через односторонний канал связи (pipe), т.к. предполагается, что никакие данные в него не будут записываться. Необходим он для того, чтобы, в случае завершения порожденного процесса (программы-сервера), ядро ОС сообщило родительскому процессу (лаунчеру) о том, что был закрыт файловый дескриптор записи, открытый у порожденного процесса. Алгоритм создания такого канала связи:

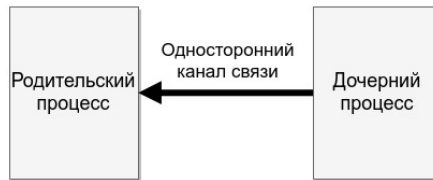
- родительский процесс создаёт канал, т.е. устанавливает два файловых дескриптора и связывает их через ядро ОС. Первый файловый дескриптор используется для чтения из этого канала, второй для записи в этот канал;



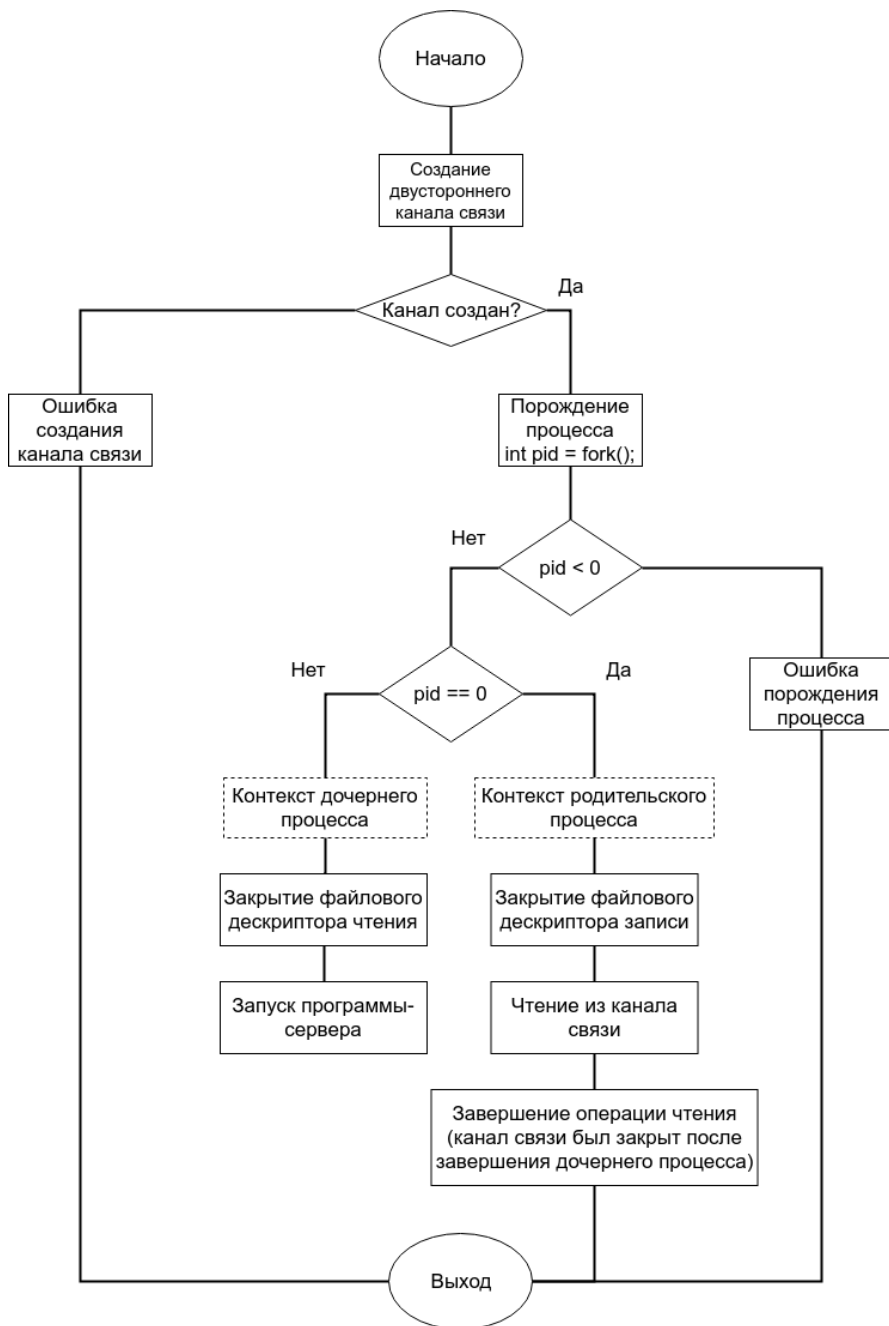
- родительский процесс создаёт собственную точную копию вызовом `fork`. В случае успеха родительский и порождённый процессы будут иметь по два открытых файловых дескриптора, которые связаны через ядро ОС. Такая связь называется двусторонним каналом связи;



- из двустороннего канала связи можно получить односторонний, если в родительском закрыть один файловый дескриптор и в порождённом процессе закрыть противоположный файловый дескриптор. Например, в порождённом закрывается дескриптор на чтение, а в родительском – на запись.



Алгоритм запуска дочернего приложения:



Алгоритм работы лаунчера:

- считывается конфигурационный файл, содержащий в себе путь до запускаемого файла (программы) и аргументы, подаваемые на вход. Файл записан в формате JSON. Пример конфигурационного файла:

```

{
  "exe": "executable_file",
  "count": 1,
  "argv":
  [
    "--config=config_file.json"
  ]
}
  
```

Здесь «exe» – переменная-строка, содержащая путь до файла-программы, «count» – количество создаваемых экземпляров программы, «argv» – массив переменных-строк, содержащих аргументы запускаемой программы, например, путь до конфигурационного файла;

- запускается одно дочернее приложение-сервер;
- при завершении порождённого процесса алгоритм повторяется сначала.

Предполагается, что порождённый процесс не будет ничего писать в канал связи. Он (канал связи) нужен для того, чтобы определить, когда дочерний процесс завершится (при завершении процесса ядро ОС закрывает все его каналы связи и сообщает связанным с ним процессам о закрытии). В логике такого лаунчера предусмотрено выявление факта завершения процесса, но не причины, т.к. важен сам факт выхода из строя.

Такой подход нередко используется для создания автоматических лаунчеров. Но в данной реализации используются механизмы асинхронного чтения из каналов связи, что позволяет следить за большим числом подконтрольных процессов. Асинхронность достигается использованием библиотеки Boost.Asio.

1.2. Надёжный сетевой протокол

Одной из важнейших задач проектирования масштабируемой и отказоустойчивой системы является выбор надёжного протокола сетевого взаимодействия элементов системы. Под надёжным протоколом понимается протокол, обеспечивающий гарантированную доставку пакетов данных, их целостность и правильный порядок.

Выбор протокола передачи сообщений начинается с транспортного уровня, т.к. в зависимости от него выбирается соответствующий протокол более высокого уровня. На транспортном уровне существует два наиболее популярных протокола: TCP и UDP.

TCP (Transmission Control Protocol) – протокол передачи потока данных с предварительной установкой соединения, контролирующей доставку пакетов данных, их целостность и упорядоченность. Реализация протокола присутствует в ядре ОС Linux и во многих других системах.

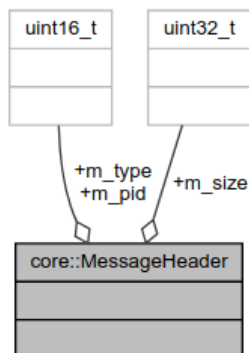
UDP (User Datagram Protocol) – протокол, позволяющий передавать пакеты данных без предварительной установки соединения. В задачах этого протокола нет обеспечения надёжности, упорядочивания или целостности данных, поэтому эти задачи решает протокол прикладного уровня. Реализация протокола присутствует в ядре ОС Linux и во многих других системах.

При использовании TCP, в отличие от UDP, на прикладном уровне не нужно уделять внимание проверке целостности пакетов и их упорядоченности при составлении собственного протокола сетевого взаимодействия. Поэтому для создания масштабируемой и отказоустойчивой системы использован именно TCP протокол.

На прикладном уровне существует большое множество протоколов, работающих как поверх TCP (HTTP, FTP, SSH, WebSocket), так и поверх UDP (RIP, SNMP, RTP).

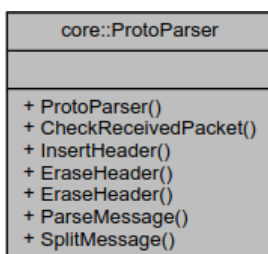
HTTP (HyperText Transfer Protocol) – протокол коммуникации компонент клиент-серверной архитектуры, работающий поверх установленного TCP соединения. Изначально предназначен для передачи гипертекстовых документов, но его область применения была расширена до передачи любых данных. Одной из главных особенностей протокола HTTP 1.0 (первоначальная версия) является то, что после обработки запроса сервером и получением ответа клиентом последний должен закрыть соединение. Но после стандартизации работы протокола HTTP версии 1.1 появилась возможность использовать новый режим работы соединения – keep-alive. Этот режим позволяет отправлять по установленному соединению несколько запросов на сервер. Использование режима keep-alive протокола HTTP имеет свои недостатки:

- каждое сообщение формируется в HTTP запрос, заголовок которого может занимать очень много места, в зависимости от количества атрибутов;



Для подготовки сериализованных данных к отправке по сети был написан класс «ProtoParser», который добавляет заголовок к отправляемым данным и производит обратную операцию – удаляет заголовок из полученного сообщения, чтобы получить данные пакета.

Диаграмма класса «ProtoParser»:



1.3. Разделение логики сервера на независимые части и масштабирование компонент

Идея метода разделения логики сервера заключается в том, чтобы выделить и объединить в группы задачи схожего характера и определить компоненты системы, которые будут решать эти задачи независимо друг от друга. Такой подход позволяет избежать критических ошибок работы всей системы при выходе из строя одной из компонент, потеряв на время часть функционала.

Достоинства разбиения бизнес-логики по нескольким компонентам:

- благодаря тому, что каждая компонента должна выполнять задачи одного соответствующего типа, программы получаются очень простыми;
- при выходе из строя одной компоненты теряется часть функционала системы, а не весь целиком;
- разработку таких компонент можно разбить на несколько независимых команд программистов, согласовав протоколы и интерфейсы взаимодействия частей.

Для реализации чат-сервера было реализовано разбиение серверной бизнес-логики по нескольким компонентам и реализована возможность масштабирования некоторых из них, что позволило создать отказоустойчивую систему доставки пакетов данных, рабочую нагрузку которой можно регулировать добавлением или удалением узлов.

Будем рассматривать систему, в которой присутствует компонента, выполняющая роль главного сервера. Под главным сервером понимается программа, принимающая заявки на вход в систему и распределяющая поступающие задачи между подчинёнными серверами.

Этапы проектирования и создания отказоустойчивой и масштабируемой системы:

- 1) определить основные задачи системы;
- 2) объединить задачи в группы для определения компонент, которые будут решать их;

3) разработать интерфейс взаимодействия между компонентами, позволяющий масштабировать систему.

Для того, чтобы правильно разбить бизнес-логику сетевого приложения, необходимо наиболее объективно определить список задач, которые будет решать сервер.

В качестве отказоустойчивой и масштабируемой системы передачи пакетов данных был выбран многопользовательский веб-чат. Основная идея: общение между пользователями происходит в виртуальных «комнатах». Таким образом, можно выделить две фигурирующие сущности: виртуальные «комнаты» и пользователи.

Благодаря тому, что в дальнейшем будут выделены независимые компоненты, решающие определённый набор задач, функциональное наполнение можно будет легко и быстро изменить, поэтому было решено на начальном этапе реализовать минимальный функционал:

- каждый пользователь должен быть зарегистрирован и авторизован в системе;
- виртуальные «комнаты» могут быть созданы любым пользователем;
- любой пользователь может зайти в любую «комнату» и выйти из неё. Список всех «комнат» доступен каждому пользователю;
- любой пользователь, находящийся в «комнате», может отправить сообщение, которое будет доставлено всем участникам «комнаты»;
- после закрытия «комнаты» все данные о ней стираются, т.е. имеется только один тип виртуальных «комнат» – временные;
- у каждой «комнаты» есть название (не обязательно уникальное), описание, имя владельца (создателя) «комнаты».

С учётом выбранного функционала сетевого приложения были выделены три основных группы задач:

- задачи, связанные с обработкой запросов неавторизованных пользователей;
- задачи, связанные с обработкой запросов, связанных с виртуальными «комнатами»;
- задачи, связанные с приемом запросов авторизованных пользователей и передачей сообщений между ними.

Задачи второй и третьей групп являются частными случаями группы задач, связанных с обработкой запросов авторизованных пользователей, но для того, чтобы как можно более равномерно распределить задачи по группам, были выделены три, перечисленные выше. Каждой группе задач будет соответствовать определенная компонента:

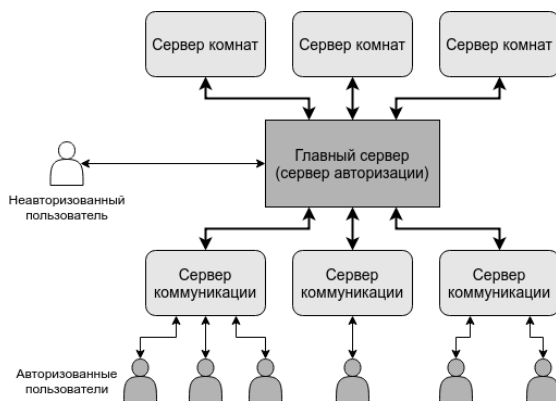
- сервер авторизации – компонента системы, предназначенная для обработки запросов авторизации и регистрации пользователей;
- сервер комнат – компонента системы, предназначенная для обработки запросов на вход в «комнату», выход из неё, хранение информации о «комнатах», об их участниках и владельцах;
- сервер коммуникации – компонента системы, предназначенная для поддержания постоянного соединения с авторизованными пользователями, приёма запросов и передачи сообщений между пользователями.

Для того, чтобы отправить поступающий от пользователя запрос на нужную компоненту системы, необходимо разработать интерфейс взаимодействия между пользователем и системой и между компонентами.

Необходимо также определить, на какие компоненты будет приходиться наибольшая нагрузка, и спроектировать архитектуру таким образом, чтобы можно было увеличить пропускную способность системы. Такими компонентами являются: сервер коммуникации, поддерживающий одновременно соединения с большим числом авторизованных пользователей, и сервер комнат, отвечающий за хранение виртуальных «комнат», обработку связанных с ними запросов.

Для взаимодействия компонент друг с другом было решено выделить главный сервер, принимающий заявки на вход в систему, и распределяющий поступающие задачи между компонентами системы. Таким образом, роль главного сервера будет выполнять сервер авторизации.

Архитектура сетевого приложения, получившаяся в результате проектирования, представлена в виде диаграммы.



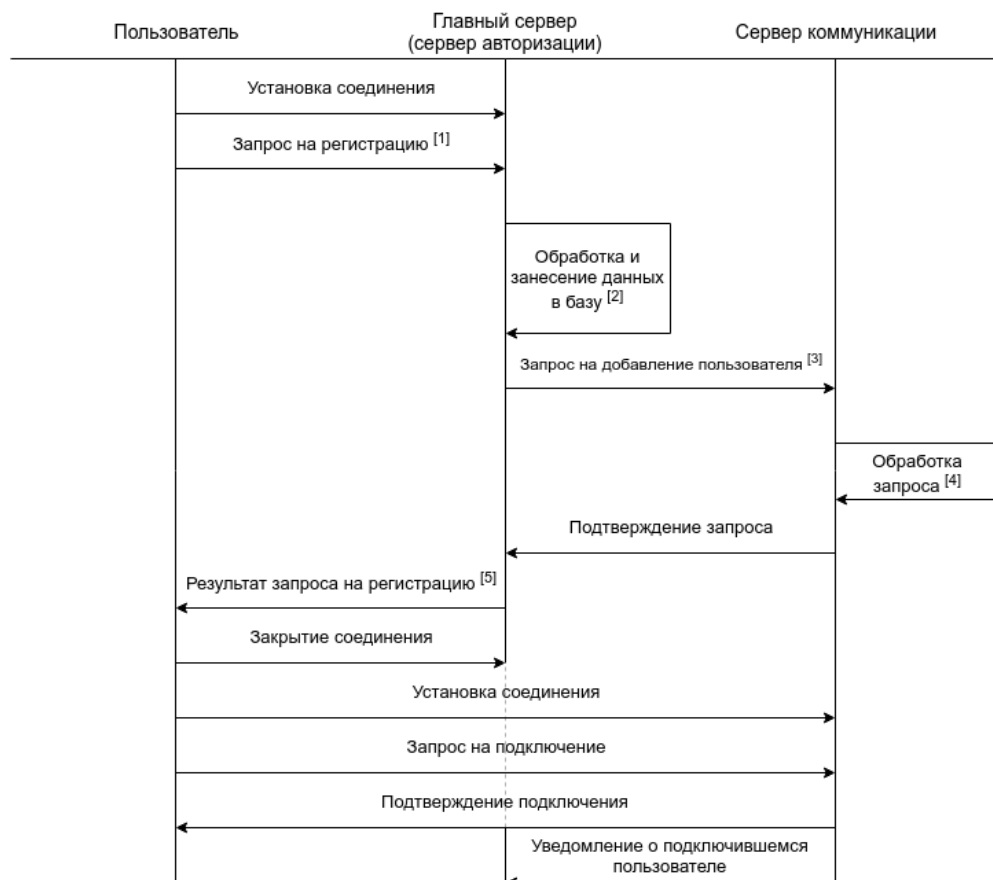
Для того, чтобы главный сервер мог распределять задачи по компонентам, он должен поддерживать с ними связь. Существует два способа сообщить главному серверу о подключившемся узле:

- напрямую сообщить каким-то образом главному серверу адрес нового подчиненного сервера (IP адрес и port). В таком случае каждый подчиненный сервер ожидает, когда главный сервер инициирует соединение;
- все подчиненные сервера знают адрес главного сервера, используя этот адрес, устанавливают соединение с ним и делают запрос на подключение к системе.

Таким образом, главный сервер не знает и не запоминает адреса других узлов системы, а постоянно находится в состоянии ожидания их подключения.

На этапе разработки архитектуры были описаны алгоритмы решения всех основных задач, описываемых минимальным функционалом, который в итоге должен реализовать создаваемый веб-чат. Основные алгоритмы:

1. Авторизация пользователя.



[1] - запрос на регистрацию включает в себя "Имя пользователя" и "Пароль".

[2] - проверка уникальности "Имени пользователя", корректности "Пароля" и занесение этих данных в БД.

[3] - случайным образом из списка выбирается сервер коммуникации, на который отправляется запрос, включающий в себя уникальный идентификатор пользователя, который должен подключиться к этому серверу.

[4] - занесение полученных данных о пользователе в контейнер.

[5] - ответ на запрос на регистрацию содержит в себе адрес (IP и port) сервера коммуникации, на который следует подключиться пользователю в случае успешной регистрации.

2. Создание виртуальной «комнаты».



[1] - запрос на создание комнаты представляет собой основные конфигурации создаваемой комнаты, например, название, ограничение на количество пользователей в комнате.

[2] - случайным образом выбирается сервер комнат, на который будет перенаправлен запрос.

3. Отправка сообщения всем пользователям «комнаты».



[1] - запрос на передачу сообщения представляет собой само сообщение и идентификатор комнаты, в которую оно было отправлено.

[2] - по идентификатору комнаты определяется список пользователей, которым адресовано сообщение. Если эти пользователи подключены к разным серверам коммуникации, то это сообщение рассылается на остальные сервера комнат

Заключение

В настоящей работе рассмотрены основные средства обеспечения отказоустойчивости и масштабируемости системы передачи пакетов данных и представлен оригинальный программный комплекс – веб-чат.

СЕКЦИЯ VII. ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ И ВИЗУАЛИЗАЦИЯ ДАННЫХ

ОЦЕНКА ЭФФЕКТИВНОСТИ МЕТОДОВ ПРОГНОЗИРОВАНИЯ В ЗАДАЧЕ ЗАПОЛНЕНИЯ ПРОПУСКОВ

А.И. Гальцева¹, А.А. Орлов², М.Н. Головчинер¹

¹Томский государственный университет,

²Томский государственный университет систем управления и радиоэлектроники
alex.galtseva@gmail.com, d1scnc@gmail.com, golovchiner@mail.ru

Введение

В настоящее время повсеместное внедрение автоматизации технических систем и процессов привело к применению широкой номенклатуры средств фиксации физических и технических величин и параметров (датчики и пр. измерительное оборудование). Таким образом, значительное количество данных о системах, объектах представлены временными рядами – это числовые значения, характеризующие состояние системы, объекта или процесса, последовательно полученные через равные промежутки времени [1]. Однако из-за некорректной работы датчика, отказа оборудования, ошибок пользователя или по иным причинам во временном ряду могут не фиксироваться как отдельные значения, так и данные в течении длительных промежутков времени. Если в первом случае можно воспользоваться методами интерполяции или аппроксимации, то во втором нужно использовать средства машинного обучения, чтобы правдоподобно восполнить ряд и сделать данные доступными для решения задач в области здравоохранения, экономики, промышленности и др.

1. Постановка задачи и исходные данные

Имеется набор показаний ЭЭГ [2], содержащий 30 файлов, каждый из которых содержит измерения от 64 электродов при частоте измерений 256 Гц (эпоха 3,9 мсек) в течение 1 секунды. Таким образом, каждое испытание – это матрица 64x256.

Данные взяты из исследования для изучения корреляций ЭЭГ с генетической предрасположенностью к алкоголизму. В данном наборе данных не содержится пропусков, поэтому они были добавлены «искусственно», чтобы стало возможным провести исследование.

Требуется оценить эффективность восстановления временных рядов, сравнить полученное восстановление с восстановлением методом кластеризации “k-means”, критерием «близости» рядов выбрать среднее квадратическое отклонение.

2. Данные. Источник и способ преобразования

Пропуски добавлялись потоком случайных чисел «mt19937ar» (среда разработки «Matlab») во временные ряды с номерами: 2, 5, 6, 9, 12–19, 24, 27, 31–42, 46–49, 53, 55, 58–62. Номера рядов выбраны произвольно авторами. Из каждого ряда удалялось от 10% до 40% значений с шагом 5%.

3. Обзор методов прогнозирования временных рядов

Задачу восстановления пропусков можно свести как к задаче кластеризации, так и к задаче прогнозирования. Рассмотрим существующие методы прогнозирования временных рядов и дадим им краткое описание [3].

Простая скользящая средняя (SMA) – метод прогнозирования, суть которого заключается в суммировании последних n значений. Затем нужно разделить сумму на n .

Скользящая средняя позволяет распознать тенденции и тренды и сгладить неровности. Часто используется для прогнозирования исторических данных.

Экспоненциальное сглаживание (ES) – метод математического преобразования временных рядов с использованием экспоненциального окна. В то время, как в простой скользящей средней прошлые наблюдения взвешиваются одинаково, экспоненциальное сглаживание присваивает значениям экспоненциально уменьшающиеся веса с течением времени. Существуют следующие типы экспоненциального сглаживания: 1) простое; 2) сезонное; 3) экспоненциальное сглаживание Холта. Метод используется для краткосрочного прогнозирования.

Авторегрессионная интеграционная скользящая средняя (ARIMA) – метод, который использует для прогнозирования временного ряда его метакarakterистики: сезонность, тенденции, циклы ошибки и нестационарность. Метод используется для краткосрочного прогнозирования. При этом, чтобы спрогнозировать одну точку, необходимо иметь 40 предшествующих ей значений. Лучше всего работает при стабильных и последовательных данных [3].

Нейронная сеть (NN) – вычислительный алгоритм, моделирующий работу человеческого мозга, а именно – его способность обучаться на примерах, уменьшая ошибку в предсказании [4]. Таким образом, чем большее количество примеров используется для обучения сети, тем точнее будет её работа. Нейронные сети являются алгоритмом искусственного интеллекта [4]. Часто используются для решения задач в областях экономики и бизнеса (прогнозирование временных рядов), медицины и здравоохранения (обработка медицинских изображений, очистка показаний приборов от шумов, прогнозирование результатов применения разных методов лечения, анализ эффективности проведённого лечения), связи (сжатие видеoinформации), интернета (ассоциативный поиск информации), робототехники (распознавание сцены, объектов и препятствий перед роботом, прокладка маршрута движения), геологоразведки (анализ сейсмических данных, ассоциативные методики поиска полезных ископаемых, оценка ресурсов месторождений) и других [Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.9]. Метод позволяет обнаруживать крайне нелинейные зависимости в данных. Нейронные сети часто используются для прогнозирования и моделирования экономических систем и процессов [6–9].

4. Заполнение пропусков во временных рядах нейронными сетями

Существует масса библиотек для программной реализации искусственных нейронных сетей (TensorFlow, PyTorch, Theano, Caffe, NaiveBase). Интерактивная среда разработки Matlab также содержит библиотеку Neural Network Toolbox, которым и воспользовались авторы как средством решения поставленной задачи.

Для заполнения данных использовался встроенный метод `newff` [7], который реализует однонаправленную сеть, обучаемую с применением алгоритма обратного распространения. Метод `newff` создаёт и возвращает однонаправленную сеть, состоящую из N слоёв. Он имеет следующий интерфейс:

```
net = newff(PR,[S1 S2 ... SN],{TF1 TF2 ... TFN},BTF,BLF,PF),
```

где

- PR – $R \times Q$ -матрица минимальных и максимальных значений строк входной матрицы с размерностью $R \times Q$. Для получения матрицы PR можно использовать функцию «`minmax`»;
- S_i – количество нейронов в i -м слое, N_1 – количество слоёв;
- TF_i – функция активации i -го слоя, по умолчанию = «`tansig`»;
- BTF – обучающая функция обратного распространения, по умолчанию = «`trainlm`»;
- BLF – алгоритм подстройки весов и смещений (обучающий алгоритм), по умолчанию = «`learngdm`»;

• PF – функция оценки функционирования сети, по умолчанию = «mse».

В рассматриваемой системе $N = 2$; $S1 = n$ (число входных значений), $S2 = 1$; $TF1 = \text{«logsig»}$, $TF2 = \text{«purelin»}$, ВТФ установлен по умолчанию.

Авторами были реализованы 2 способа восстановления рядов динамики нейронными сетями: последовательный и итерационный (интегрированный). Рассмотрим каждый из них и оценим результаты.

4.1. Последовательный способ

Данный подход основан на том, что подготовка данных производится до создания и тренировки сети.

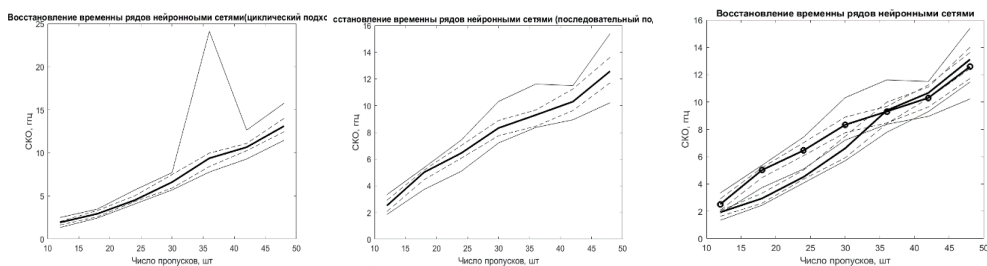
Для создания сети представим данные в виде матрицы входных значений и вектора выходных значений. Для этого, начиная с первого элемента ряда, n значений сохраним в строку матрицы, а $n + 1$ -е значение обозначим выходным значением. Продолжаем данную процедуру со сдвигом в один элемент до тех пор, пока $n + 1$ -е значение не окажется последним элементом ряда. Из матрицы удаляем строки с пропусками, а из вектора выходных значений соответствующие им элементы.

Создается и обучается сеть на основе подготовленных данных. Далее на вход обучающей нейронной сети подается входной вектор, соответствующий очередному пропуску, а на выходе сети получается предсказание пропущенного значения. Таким образом, заполняем ряд, например, ряд с 10% пропусков 10 раз. Для каждого заполнения вычисляем максимум и минимум и квантили уровня 25, 50 и 75.

При вышеописанной процедуре заполнения от 10% до 40% с шагом 5% рассчитывалось 1 час 50 минут на вычислительной машине с процессором Intel Core i5, оперативной памятью 8 Гб.

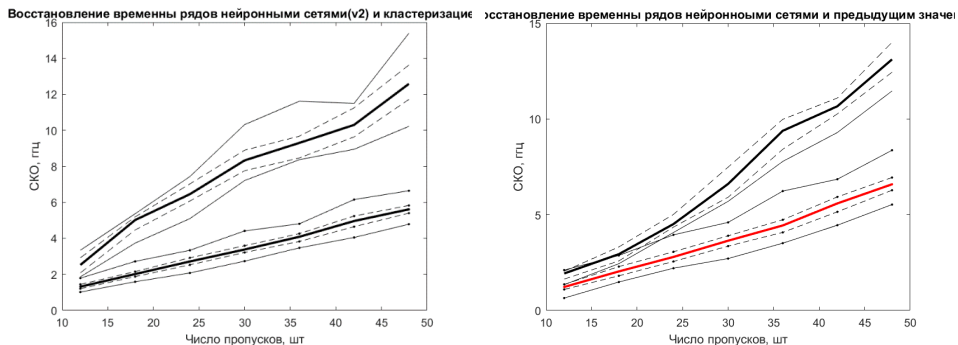
4.2. Итерационный (интегрированный или циклический) способ

Данный метод заполнения отличается тем, что подготовка данных и обучение сети повторяются итерационно для каждого пропуска, т.е. производится предобработка данных до очередного пропуска (а не до последнего элемента ряда), сеть создается и обучается по этой матрице. Пропуск определяем предсказанием функции обучения. При данном способе заполнения пропусков время проведения расчетов увеличилось до 6 часов 15 минут при прочих равных.



5. Заполнение предыдущими значениями и сравнение с кластеризацией

По сравнению с заполнением кластеризацией нейросеть показала плохие результаты, поэтому восполним пропуски предыдущими значениями, чтобы проверить гипотезу о том, что сведение к задаче прогнозирования возможно. Как видно на графике, заполнение предыдущими значениями даёт меньшее отклонение от исходных данных.



Рассмотрим СКО заполнений предыдущими значениями и нейронными сетями. Сравним поэлементно векторы отклонений, чтобы понять, всегда ли у нейронных сетей СКО больше. И так, для каждого процента пропусков (от 10% до 40% с шагом 5%) проделываем 10 заполнений. После каждого заполнения считаем количество случаев, когда элемент в векторе СКО заполнения, например, нейронными сетями, больше, чем при заполнении предыдущими. Далее суммируем эти показатели для всех итераций каждого значения пропусков и имеем следующие результаты: 2096/8470 (24,74%) – заполнения нейронными сетями давали меньшее отклонение, 6372/8470 (75,2%) – заполнения предыдущими значениями оказывались лучше, 2/8470 – СКО заполнений были равны.

Таким образом, на 25% процентах данных нейронная сеть показала себя лучше.

Приведём сравнительный анализ полученных результатов заполнения временных рядов методами, описанными в данной статье, с результатами из статьи «Оценка эффективности метода кластеризации «k-means» в задаче заполнения пропусков.». Анализ приведен в таблице ниже.

Название метода (способ заполнения)	Количество заполнений	Время заполнения
Предыдущее значение $[x(t+1) = x(t)]$	1000	18 минут
Среднее значение	1000	1 секунда
Медианное значение	1000	1 секунда
«k-means»	100	2 минуты
Нейронные сети: последовательный подход	10	1 час 50 минут
Нейронные сети: итерационный(циклический) подход	10	6 часов 15 минут

Заключение

В данном исследовании рассмотрены прогнозирующие методы восстановления пропусков в рядах динамики. Также проведен сравнительный анализ качества заполнения нейронными сетями, методом кластеризации и простыми (классическими) статистическими методами, из которого следует, что методы ИИ – это ресурсозатратные, сложные в понимании и эксплуатации методы, эффективность которых не всегда пропорциональна затраченным средствам. На исследуемых данных и при текущей постановке задачи метода прогнозирования показали худшие результаты.

В дальнейшем авторы намереваются разработать систему, которая выбирала бы алгоритм и параметры восстановления рядов динамики в зависимости от их характеристик (природа, размер, количество пропусков, статистические характеристики и тд).

ЛИТЕРАТУРА

1. Time series [Электронный ресурс] // URL: https://en.wikipedia.org/wiki/Time_series (дата обращения: 04.04.2018).
2. EEG database data set [Электронный ресурс] // URL: <https://archive.ics.uci.edu/ml/datasets/EEG+Database> (дата обращения: 01.11.2017).
3. 5 Statistical Methods For Forecasting Quantitative Time Series [Электронный ресурс] // URL: <https://www.bistasolutions.com/resources/blogs/5-statistical-methods-for-forecasting-quantitative-time-series/> (дата обращения: 07.03.2018).

4. Ежов А.А., Шумский С.А. Нейрокомпьютинг и его применение в экономике и бизнесе. – М.: МИФИ, 1998. – 224 с.
5. Geektimes Прогнозирование фондового рынка с помощью нейронных сетей [Электронный ресурс] // URL: <https://geektimes.com/post/279170/> (дата обращения: 07.04.2018).
6. Лесик И.А. Решение задачи прогнозирования с использованием нейронных сетей прямого распространения на примере построения прогноза роста курса акций // Программные продукты и системы. – 2015. – Вып. 2. – С. 70–74.
7. MATLAB.Exponenta MATLAB&Toolboxes [Электронный ресурс] // URL: <http://matlab.exponenta.ru/neuralnetwork/book2/11/newff.php> (дата обращения: 15.02.2018).
8. Artificial neural network [Электронный ресурс] // URL: https://en.wikipedia.org/wiki/Artificial_neural_network (дата обращения: 10.04.2018).
9. NeuroPro Области применения искусственных нейронных сетей [Электронный ресурс] // URL: <http://neuropro.ru/neu7.shtml> (дата обращения: 15.02.2018).

ПОСТРОЕНИЕ РЕАЛИСТИЧНЫХ ИЗОБРАЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ GLSL-ШЕЙДЕРОВ OPENGL

Л.В. Чупрасова

Томский государственный университет

l.chuprasova@gmail.com

Введение

Компьютерная графика – это достаточно сложная, основательно проработанная дисциплина, предметом изучения которой является создание, хранение и обработка моделей и их изображений с помощью ЭВМ.

3D-графика применяется во множестве различных областей: в науке, для наглядного изображения каких-то не до конца изученных процессов, в видеоиграх, которые давно не ограничиваются компьютерной платформой, в кинематографе для создания сложнейших спецэффектов, в медицине, строительстве, рекламе, и т.д. Имея качественную модель, можно увидеть объект, который не существует, посмотреть его со всех сторон, окунуться в виртуальный мир компьютерных игр, сидя дома перед экраном компьютера.

Следуя правилам создания качественной 3D-модели, можно добиться эффекта присутствия, эффекта реальности, т.е. такие объекты отличаются от других правдоподобностью. В наше время реалистичные 3D-изображения являются пиком совершенства в игровой, рекламной, строительной индустрии.

Моделирование 3D-изображений – это достаточно ресурсоемкая задача, ставящаяся перед современными вычислительными системами, и, несмотря на постоянное совершенствование техники, стремительное развитие технологий, требования к задачам остаются неизменно высокими, т.к. с ростом возможностей растут и требования, т.е. помимо оптимизации технической базы требуется оптимизация на уровне прикладного программирования.

Одним из таких методов оптимизации является грамотное использование графической библиотеки OpenGL, а именно – программирование на шейдерах. Оно дает возможность переложить часть сложных вычислений, связанных с отрисовкой сцены, с центрального процессора на специально предназначенный для этого графический процессор, являющийся аппаратной частью видеокарты.

1. Библиотека OpenGL и шейдеры

OpenGL является одним из самых распространенных прикладных программных интерфейсов (API – Application Programming Interface) к графической аппаратуре для разработки приложений в области двумерной и трёхмерной графики.

Библиотека содержит в себе около 120 команд, которые может использовать программист для задания объектов и операций, необходимых для написания интерактивного графического приложения.

Основными особенностями этой графической библиотеки являются:

- Стабильность – все нововведения создаются таким образом, чтобы сохранялась совместимость с предыдущим обеспечением.
- Кроссплатформенность – независимо от используемой операционной системы OpenGL гарантирует одинаковый визуальный результат. Кроме того, эти приложения работают и на рабочих станциях, и на суперкомпьютерах.
- Интуитивно понятный интерфейс и продуманная структура, что обеспечивает легкость применения [1].

Вместе с выходом версии OpenGL 3.0 стала доступна шейдерная программа, именно она лежит в основе решения задач реализации реалистичного 3D-изображения.

Шейдерная программа – это небольшая программа, состоящая из программных модулей – шейдеров (вершинного и фрагментного, возможны и другие). Отличительной особенностью шейдеров является то, что они выполняются на аппаратном уровне оборудования, используя видеопамять и графический процессор (GPU) [2]. Программа заменяет часть графического конвейера видеокарты, без её использования мы не получим даже самую простую картинку на экране.

Программы, работающие с трёхмерной графикой, используют шейдеры для определения параметров геометрических объектов или изображения, для изменения изображения (для создания эффектов сдвига, отражения, преломления, затемнения с учётом заданных параметров поглощения и рассеяния света, для наложения текстур на геометрические объекты и др.)

Шейдеры делят на три типа, в зависимости от того, какой процессор будет их исполнять:

- Вершинный шейдер (vertex shader). Заменяет часть графического конвейера, выполняющего преобразование, связанные с данными вершин геометрических примитивов, такие, как умножение вершин и нормалей на матрицу проекции и моделирования, установка цветов вершин, установка материалов освещения. Он работает для каждой отрисованной вершины.
- Фрагментный или пиксельный шейдер (pixel shader). Заменяет часть графического конвейера, обрабатывает каждый полученный на предыдущих стадиях фрагмент, а именно – данные, связанные с пикселями. Например, получение данных из текстуры, просчёт освещения, просчёт смешивания. Пиксельный шейдер используется на последней стадии графического конвейера для формирования фрагмента изображения.
- Геометрический шейдер (geometry shader). В отличие от вершинного, геометрический шейдер способен обработать не только одну вершину, но и целый примитив. Такой тип шейдера используется редко [2].

Шейдеры для OpenGL пишутся на специальном языке GLSL. Шейдером называется независимо компилируемая единица, написанная на этом языке. GLSL-шейдеры принято хранить в виде исходных кодов. Исходные коды компилируются драйвером лишь после создания действующего контекста OpenGL. Шейдерная программа – это набор скомпилированных шейдеров, связанных вместе.

GLSL (OpenGL Shading Language) – язык высокого уровня для программирования шейдеров. Синтаксис языка базируется на языке программирования ANSI C, однако, из-за его специфической направленности для упрощения и повышения производительности из него были исключены многие возможности. В язык включены дополнительные функции и типы данных, например, для работы с векторами и матрицами [3].

2. Реалистичное изображение

Для создания реалистичных изображений необходимо определить как свойства самого объекта, так и свойства внешней среды.

Первая группа включает в себя параметры материала, из которого сделан объект, степень его прозрачности, а также текстура. Ко второй группе относятся количество и свойства источников света, модель освещения и отрисовка теней.

Каждое свойство задаётся последовательно в исходном коде программы, с помощью вызовов специальных команд OpenGL.

Для снятия нагрузки с центрального процессора была написана шейдерная программа, состоящая из двух вершинных и двух фрагментных шейдеров, в которой обрабатываются все данные, необходимые при построении качественных 3D-моделей.

Чтобы нарисовать какие-либо модели, создан класс Mesh.

2.1. Источники света

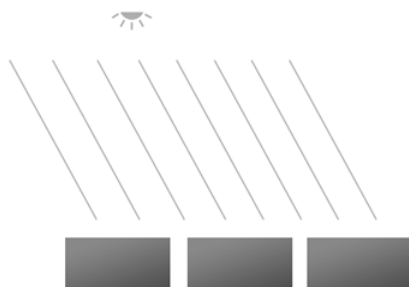
В компьютерной графике чаще всего работают с тремя типами источников света [4]:

- Точечный источник света (point light).
- Направленный источник света (directional light).
- Прожектор (spot light).

В работе настроен и реализован направленный источник света.

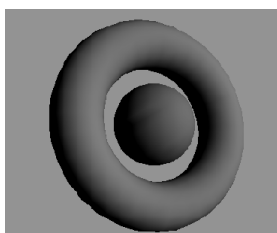
Направленный источник света находится в бесконечности, и свет распространяет в заданном направлении. Если источник света достаточно удалён от объекта, то все его приходящие лучи параллельны друг другу, поэтому создается впечатление, что свет направлен одинаково и не зависит ни от расположения источника света, ни от положения наблюдателя. Поэтому и расчёт освещения для всех объектов будет одинаковым. Такой свет идеально подходит для равномерного освещения.

Хорошей аналогией направленного источника света является солнце. Хоть оно и не бесконечно удалено от нас, этого расстояния достаточно, чтобы считать его таковым при расчётах [5].



Свет настраивается в исходном коде программы, но его свойства требуются при расчётах в построении модели освещения, которые производятся в шейдерной программе. В OpenGL нет встроенных функций передачи конкретных данных в шейдер, поэтому для работы с источником света создан класс Light.

Когда мы настраиваем все компоненты источника света, становится видно сцену с объектами:



2.2. Модель освещения

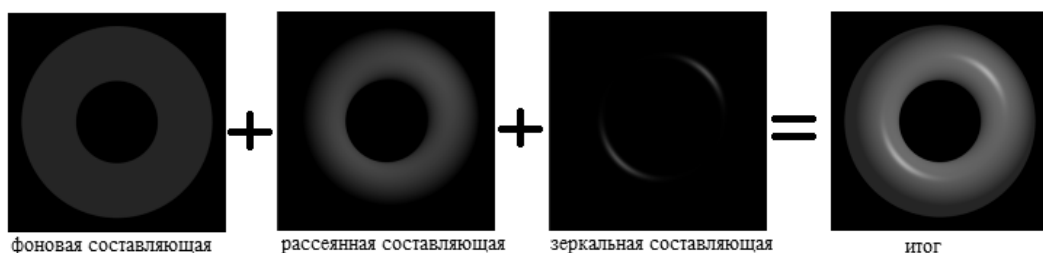
В любом трехмерном изображении использование какой-либо модели освещения придает реалистичность. Любая модель освещения включает в себя закон, по которому

рассчитывается освещённость точки в пространстве и метод закраски многоугольника. В OpenGL нет готовой реализации модели освещения.

В работе использована закраска методом Фонга, потому что она не требует больших вычислительных затрат, однако позволяет решить многие проблемы других методов.

Суть метода Фонга заключается в определении нормали к поверхности в каждой вершине полигона и дальнейшей интерполяции вектора по всему полигону. Далее вычисляется значение яркости для каждого пикселя, основываясь на значениях вектора нормали. При этом достигается лучшая аппроксимация кривизны поверхности и, следовательно, получается более реалистичное изображение.

Освещённость каждой точки складывается из трёх составляющих: фоновое освещение, рассеянный свет и бликовая составляющая. Свойства источника света определяют мощность излучения для каждой из этих компонент, а свойства материала поверхности определяют её способность воспринимать каждый вид освещения.



1. Фоновое освещение (ambient) – это постоянная в каждой точке величина надбавки к освещению. Теоретически, в сцене оно имитирует рассеянный свет, не имеющий конкретного источника. Поскольку центра и направления у него нет, оно совершенно одинаково для всех точек поверхности.

2. Рассеянный свет или диффузное отражение (diffuse) соответствует отражению света от неровной поверхности. При попадании на подобную поверхность свет рассеивается практически равномерно по всем направлениям

3. Зеркальный свет (specular) учитывает не только направление на источник света, но и местоположение наблюдателя. При попадании на поверхность подчиняется следующему закону: «падающий и отражённый лучи лежат в одной плоскости с нормалью к отражающей поверхности в точке падения, и эта нормаль делит угол между лучами на две равные части». Таким образом, отражённая составляющая освещённости в точке зависит от того, насколько близко направления на наблюдателя и отражённого луча [6].

2.3. Материал объекта

Также нужно определить материал объекта для того, чтобы знать, как этот объект будет реагировать на освещение. Свойства материала определяются следующими параметрами:

1. Восприятие фонового освещения (ambient). Фоновое освещение на практике используется, чтобы подсветить излишне тёмные участки.

2. Восприятие рассеянного, диффузного освещения (diffuse). Диффузное отражение придает объекту его естественный цвет.

3. Восприятие отражённого освещения (specular). Зеркальная компонента определяет отражающие свойства поверхности.

4. Самостоятельное свечение (emission). Самостоятельное свечение материала определяет интенсивность излучаемого света материалом.

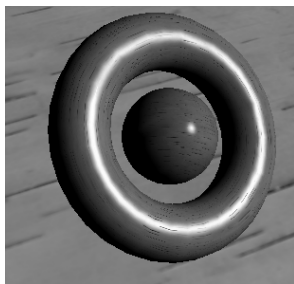
5. Коэффициент блеска (shiniess). Коэффициент блеска определяет степень зеркального отражения материала. Значение задается от 0 до 128 [7].

6. Текстурирование. Под текстурой понимается некоторое изображение, которое надо определённым образом нанести на объект, например, для придания иллюзии релье-

ефности поверхности. Изображение может быть любым, но чаще всего это шаблоны кирпича, дерева, листьев и другие, с помощью которых сцене добавляется дополнительный реализм.

Все свойства материалов также определены в исходном коде программы. При рендере сцены данные передаются в шейдер, в котором выполняется расчёт цвета для каждого пикселя с учётом всех обозначенных свойств. Для работы с материалом создан класс Material.

После добавления модели освещения и текстурирования получено следующее изображение:



2.4. Тени

Одной из главных задач при синтезе реалистичных изображений является построение теней. С помощью теней реализм виртуальной сцены значительно увеличивается, а также они помогают лучше понять взаимное расположение объектов и источника света.

К сожалению, в OpenGL нет функции построения теней, но мы можем реализовать сами один из нескольких известных алгоритмов с помощью этой библиотеки.

С физической точки зрения, тени можно разделить на две составляющие: полная (абсолютная) тень и полутень. Полная тень – это та часть затеняемого объекта, которая должна быть невидима совсем – тёмная, резко очерченная часть. При реализации алгоритма построения теней используется только полная тень.

Существует несколько алгоритмов при визуализации сцены с тенями. В настоящее время наиболее распространены три:

1. Проективные тени.
2. Метод теневых объёмов.
3. Карта теней.

Был использован алгоритм карты теней (Shadow maps). Это наиболее распространенный метод. Пожалуй, его суть является самой простой из всех. Метод базируется на том, что точки, находящиеся в тени – это те точки, которые “спрятаны” от источника света. Метод использует буфер глубины (Z-буфер).

Для того, чтобы определить видимые и невидимые точки, выполняем следующие действия:

1. Сцену выводим в Z-буфер из положения источника света, после чего создаем пустую текстуру для хранения буфера глубины (CreateDepthTexture). Содержимое буфера копируем в эту текстуру, которая и называется картой тени. Она хранит значения глубины ближайших к источнику света элементов сцены.

Так как при заполнении буфера глубины нет необходимости обрабатывать цвет пикселя, а нужно знать только его глубину, написан отдельный вершинный шейдер, в котором отключены все лишние расчёты, чтобы как можно больше ускорить работу программы, а т.к. цвет пикселя нас не интересует, фрагментный шейдер может быть пустым.

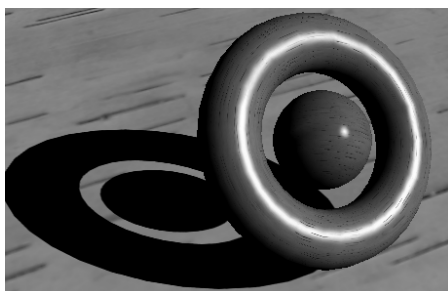
Также здесь важно отметить, что для разных типов источников света все происходит по-разному. Направленные источники света не имеют позиции в пространстве, к ним можно при определённом приближении отнести солнечный свет. Но для формиро-

вания карты теней эту позицию приходится выбирать, обычно её привязывают к положению наблюдателя, чтобы в карту попадали объекты, находящиеся в поле зрения наблюдателя. При рендеринге используют ортографическую проекцию.

2. Рисуем сцену из основной камеры, с точки зрения наблюдателя. Для того, чтобы понять, в тени точка или освещена, достаточно перевести координаты этой точки в пространство карты теней и сравнить. Переводим координаты точки из мировой системы координат в текстурные координаты, которые лежат в диапазоне [0;1]. Если полученные координаты не попали в диапазон [0;1], то точка не находится в карте теней, значит, она освещена. Сделав эту выборку, сравниваем глубину точки, сохранённой в карту теней и глубину точки, переведенной в систему координат источника. Точка оказывается в тени, если значение в карте теней меньше, т.е., если значение из карты теней меньше, то в этой точке есть какой-то объект, который находится ближе к источнику света, и мы находимся в его тени.

Скорость выполнения данной реализации метода не зависит ни от сложности затеняющих объектов, ни от сложности сцены, что обуславливает высокую эффективность [8].

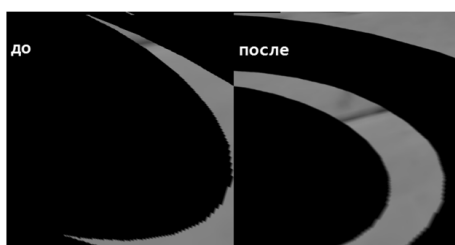
Реализация тени выглядит таким образом:



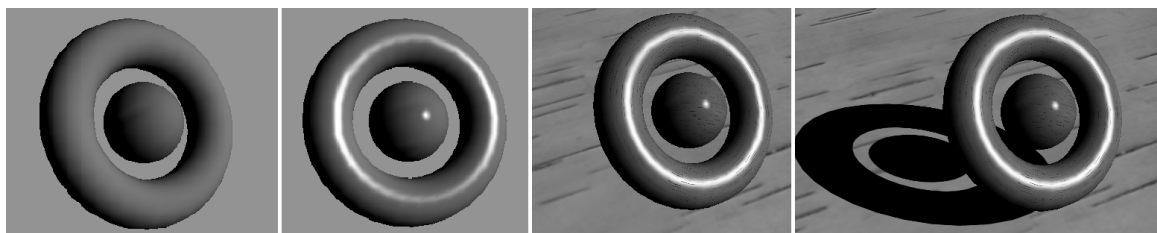
2.5. Сглаживание краев тени

Для сглаживания краёв тени использована техника Percentage Closer Filtering (PCF). Данная техника заключается в том, что производится несколько тестов на затенение в окрестностях искомого фрагмента [9], т.е. для каждого пикселя берётся среднее значение его соседних пикселей в карте теней. Таким образом, получается плавный переход из тёмной области в светлую область, граница теней будет более «мягкой».

Расчёт средних значений произведен также в основном шейдере.



Демонстрация:



Плоская закраска + свет + модель освещения Фонга + текстурирование + тень

Были проведены эксперименты, связывающие сложность сцены и время выполнения программы (ЦП: AMD E-350 Processor 1.60 GHz, GPU: Radeon HD 6330M, эффективная тактовая частота памяти 1800 МГц, память 1024 МБ).

Количество объектов сцены	Время в секундах
1	0.00145484
3	0.00274012
6	0.00380818
12	0.00420133
100	0.00551013

Так как все расчеты и прорисовка выполняются на графическом процессоре, изменение сложности сцены несущественно влияет на время выполнения программы.

Заключение

Перед нами была поставлена задача спроектировать реалистичную сцену, освещаемую методом Фонга и затенённую по алгоритму карты теней. Для этого были изучены некоторые методы стандарта OpenGL. Благодаря OpenGL и его шейдерной программе процесс построения этой модели был сведён к прикладному математическому решению и использованию небольшого количества ресурсов, что сводит математические расчёты к минимуму.

Задача была выполнена, разработана и реализована архитектура классов, способных отрисовывать трёхмерные сцены с использованием основных методов OpenGL.

Создано демонстрационное приложение с использованием этой библиотеки. Реализованы алгоритмы 3D-графики: освещение методом Фонга с текстурированием объекта и отрисовка теней алгоритмом карты теней с использованием техники сглаживания Percentage Closer Filtering (PCF).

ЛИТЕРАТУРА

1. Графическая библиотека OpenGL. Учебно-методическое пособие [Электронный ресурс]. / Ю.М. Баяковский, А.В. Игнатенко, А.И. Фролов – Факультет вычислительной математики и кибернетики МГУ им. Ломоносова, 2003. – URL: <https://tsdn.org/article/opengl/oglut2.xml> (дата обращения: 20.05.2018).
2. Шейдер [Электронный ресурс]. – URL: <https://ru.wikipedia.org/wiki?curid=107896> (дата обращения: 20.05.2018).
3. OpenGL Shading Language [Электронный ресурс]. – URL: https://ru.wikipedia.org/wiki/OpenGL_Shading_Language (дата обращения: 20.05.2018).
4. Компьютерная графика / теория, алгоритмы, примеры на C++ и OpenGL / Источники света [Электронный ресурс]. – URL: http://compgraphics.info/OpenGL/lighting/light_sources.php (дата обращения: 20.05.2018).
5. Learn OpenGL. Урок 2.5 – Источники света [Электронный ресурс]. – URL: <https://habrahabr.ru/post/337642/> (дата обращения: 20.05.2018).
6. Компьютерная графика / теория, алгоритмы, примеры на C++ и OpenGL / Закон Ламберта. Модель отражения Фонга. Модель отражения Блинна – Фонга [Электронный ресурс]. – URL: http://compgraphics.info/3D/lighting/phong_reflection_model.php (дата обращения: 20.05.2018).
7. Свойства материала [Электронный ресурс]. – URL: http://metodpro.ru/index.php?type_page&katalog&id=999&met6 (дата обращения: 20.05.2018).
8. Реализация теней с помощью библиотеки OpenGL / Журавлев С.В., Михайлюк М.В., Торгашев М.А. [Электронный ресурс]. – URL: http://swsys.ru/print/article_print.php?id=863 (дата обращения: 20.05.2018).
9. Компьютерная графика лекция GLSL-тени от объектов [Электронный ресурс]. – URL: <http://docplayer.ru/52035159-Компьютерная-графика-лекция-gsl-teni-ot-obektov.html> (дата обращения: 20.05.2018).

РАЗРАБОТКА И АНАЛИЗ МОДЕЛИ С ПОМОЩЬЮ ТЕХНОЛОГИИ PROCESS MINING

В.В. Никитина

Российский университет дружбы народов
nikitinavalerie@mail.ru

Введение

В современном мире особым ресурсом для каждого предприятия является информация. Все больше и больше происходящих событий автоматически фиксируется информационной системой и хранится в журнале событий. Это дает возможность улучшать и поддерживать бизнес-процессы в быстро изменяющейся конкурентной окружающей среде. В данной статье помощью интеллектуального анализа данных исследуется, как данные из фиксируемого журнала событий преобразуются в модель, рассматривается техника обнаружения бизнес-процессов, а также все этапы от начального регистрирования процесса до анализа функционирования предприятия

1. Постановка задачи

Для анализа процессов была взята статистика системы по осуществлению заказа, действия которой фиксируются в журнале событий (рис. 1).

Case ID	Event ID	Timestamp	Activity	Resource
1	Ch=6680555556-1	2012-07-30 11:14	Проверить наличие на складе	SYS1
1	Re=597222222-1	2012-07-30 14:20	Получить продукт со склада	Рик
1	Co=6319444444-1	2012-07-30 15:10	Подтвердить заказ	Чак
1	Ge=6402777778-1	2012-07-30 15:22	Получить адрес доставки	SYS2
1	Em=6555555556-1	2012-07-30 15:44	Выставить счет	SYS2
1	Re=4180555556-1	2012-08-04 10:02	Получить оплату	SYS2
1	Sh=665972222-1	2012-08-05 11:11	Отправить продукт	Сьюзи
1	Ar=3833333333-1	2012-08-06 09:12	Занести заказ в архив	DMS
2	Ch=4055555556-2	2012-08-01 09:44	Проверить наличие на складе	SYS1
2	Ch=4208333333-2	2012-08-01 10:06	Проверить наличие материалов	SYS1
2	Re=6666666667-2	2012-08-01 11:12	Запросить исходные материалы	Ринго
2	Ob=3263888889-2	2012-08-03 07:50	Получить исходные материалы	Олаф
2	Ma=6131944444-2	2012-08-04 14:43	Изготовить продукт	SYS1
2	Co=6187615741-2	2012-08-04 14:51	Подтвердить заказ	Конни
2	Em=6388888889-2	2012-08-04 15:20	Выставить счет	SYS2
2	Ge=6439814815-2	2012-08-04 15:27	Получить адрес доставки	SYS2
2	Sh=7277777778-2	2012-08-04 17:28	Отправить продукт	Сара
2	Re=3613111111-2	2012-08-07 08:40	Получить оплату	SYS2
2	Ar=3680555556-2	2012-08-07 08:50	Занести заказ в архив	DMS
3	Ch=4208333333-3	2012-08-02 10:06	Проверить наличие на складе	SYS1
3	Ch=4243055556-3	2012-08-02 10:11	Проверить наличие материалов	SYS1
3	Ma=6694444444-3	2012-08-02 16:04	Изготовить продукт	SYS1
3	Co=6751157407-3	2012-08-02 16:12	Подтвердить заказ	Чак
3	Em=6896833333-3	2012-08-02 16:33	Выставить счет	SYS2
3	Sh=7013888889-3	2012-08-02 16:50	Получить адрес доставки	SYS2
3	Ge=7069444444-3	2012-08-02 16:58	Отправить продукт	Эмиль
3	Re=4305555556-3	2012-08-06 10:20	Получить оплату	SYS2
3	Ar=4340277778-3	2012-08-06 10:25	Занести заказ в архив	DMS
4	Ch=3409722222-4	2012-08-04 08:11	Проверить наличие на складе	SYS1
4	Re=5000115741-4	2012-08-04 12:00	Получить продукт со склада	Ханс
4	Co=5041898148-4	2012-08-04 12:06	Подтвердить заказ	Ханс
4	Ge=5223148148-4	2012-08-04 12:32	Получить адрес доставки	SYS2
4	Em=4034837963-4	2012-08-08 09:41	Выставить счет	SYS2
4	Re=4180555556-4	2012-08-08 10:02	Получить оплату	SYS2
4	Sh=5715277778-4	2012-08-08 13:43	Отправить продукт	Сьюзи
4	Ar=5888888889-4	2012-08-08 14:08	Занести заказ в архив	DMS

Рис. 1. Журнал событий «Осуществление заказа»

Были поставлены 3 задачи:

- 1) рассмотреть этап Process Mining «Обнаружение» на основе простейшего примера журнала событий;
- 2) составить отношения альфа-алгоритма для журнала событий;
- 3) построить бизнес-процессы из исходного журнала событий и сделать представление в сети Петри.

2. Интеллектуальный анализ данных и журнал событий

Интеллектуальный анализ данных определяется как «анализ (часто больших) наборов данных для поиска непредвиденных отношений и обобщение данных по новым способам, которые являются понятными и полезными для владельца данных [1].

Входные данные обычно приводятся в виде таблицы, которую называют журналом событий, а результатом могут быть правила, кластеры, древовидные структуры, графики, уравнения и шаблоны.

Интеллектуальный анализ данных включает в себя три этапа: обнаружение, проверка соответствия и улучшение. На первом этапе модель бизнес-процесса восстанавливается при помощи анализа журнала информационной системы, а также специальных алгоритмов. Далее происходит проверка соответствия модели бизнес-процесса экземплярам бизнес-процесса, которые были получены из событийного журнала информационной системы. Последний этап способствует улучшению исходных процессов с помощью получения информации из журнала. Одним из видов улучшения является ремонт, т.е. изменение модели, чтобы лучше отразить реальность. Например, если два действия моделируются последовательно, но на самом деле происходят в любом порядке, то модель может быть исправлена, чтобы отразить это. Другим типом улучшения является расширение, т.е. добавление новой перспективы в модель процесса путём взаимной корреляции с журналом.

Журнал событий представляется в форме списка в табличном формате (рис. 1). Можно заметить, что каждое событие имеет уникальный идентификатор. Помимо этого, он относится к одному отдельному случаю, имеет метку времени и показывает, какие ресурсы были использованы. Это могут быть участники (как например, Конни и Рик) или программные системы (SYS1, SYS2, DMS).

Для того, чтобы было легче осуществлять альфа-алгоритм, присвоим букву каждому процессу в журнале событий. Таким образом, создадим журнал рабочих процессов (рис. 2).

Case ID	Event ID	Timestamp	Activity	Letter	Activities
1	Cn=6680555556-1	2012-07-30 11:14	Проверить наличие на складе	a	Проверить наличие на складе
1	Re=5972222222-1	2012-07-30 14:20	Получить продукт со склада	b	Получить продукт со склада
1	Co=6319444444-1	2012-07-30 15:10	Подтвердить заказ	c	Проверить наличие материалов
1	Ge=6402777778-1	2012-07-30 15:22	Получить адрес доставки	d	Запросить исходные материалы
1	Em=6555555556-1	2012-07-30 15:44	Выставить счет	e	Получить исходные материалы
1	Re=4180555556-1	2012-08-04 10:02	Получить оплату	f	Изготовить продукт
1	Sh=4659722222-1	2012-08-05 11:11	Отправить продукт	g	Подтвердить заказ
1	Ar=3833333333-1	2012-08-06 09:12	Занести заказ в архив	h	Получить адрес доставки
2	Cn=4055555556-2	2012-08-01 09:44	Проверить наличие на складе	i	Отправить продукт
2	Ch=4208333333-2	2012-08-01 10:06	Проверить наличие материалов	j	Выставить счет
2	Re=4666666667-2	2012-08-01 11:12	Запросить исходные материалы	k	Получить оплату
2	Ob=3263888889-2	2012-08-03 07:50	Получить исходные материалы	l	Занести заказ в архив
2	Ma=6131944444-2	2012-08-04 14:43	Изготовить продукт		
2	Co=6187415741-2	2012-08-04 14:51	Подтвердить заказ		
2	Em=6388888889-2	2012-08-04 15:20	Выставить счет		
2	Ge=6439814815-2	2012-08-04 15:27	Получить адрес доставки		
2	Sh=7277777778-2	2012-08-04 17:28	Отправить продукт		
2	Re=3611111111-2	2012-08-07 08:40	Получить оплату		
2	Ar=3680555556-2	2012-08-07 08:50	Занести заказ в архив		


	Workflow Log a,b,g,h,j,k,i,l a,c,d,e,f,g,j,h,i,k,l ...
---	--

Рис. 2. Журнал рабочих процессов

3. Альфа-алгоритм и отношения

Альфа-алгоритм является наиболее простым алгоритмом для автоматического обнаружения процесса, который берёт за основу журнал событий L и его α -отношения (рис. 3).

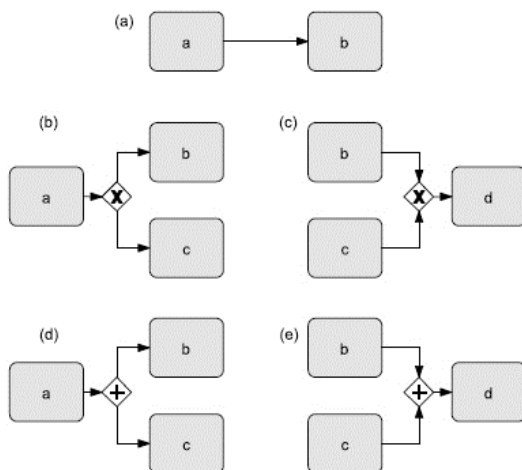


Рис. 3. Альфа-отношения

Главная идея алгоритма состоит в том, что задачи, которые непосредственно следуют друг за другом в журнале, должны быть непосредственно связаны в модели процесса. Кроме того, если есть не менее чем одна задача, которая может следовать за другой, необходимо определить, является ли набор последующих задач частично исключочающим или параллельным. Можно выделить следующие шаги α -алгоритма.

1. Определить набор всех процессов в журнале как T_L .
2. Определить первый процесс, как T_1 .
3. Определить конечный процесс, как T_0 .
4. Определить набор всех соединений, которые потенциально могут быть представлены в процессе, как набор X_L . Добавить в X_L следующие элементы:
 - а. Шаблон (а): все пары, для которых выполняется следование $a \rightarrow b$.
 - б. Шаблон (б): все тройки, для которых выполняется $a \rightarrow (b\#c)$.
 - с. Шаблон (с): все тройки, для которых выполняется $(b\#c) \rightarrow d$.

Обратите внимание, что тройки, для которых шаблон (д) $a \rightarrow (b\|c)$ или шаблон (е) $(b\|c) \rightarrow d$, не включены в X_L .

5. Построить множество Y_L в качестве подмножества X_L следующим образом:
 - а. Исключаем $a \rightarrow b$ и $a \rightarrow c$, если существует некоторое $a \rightarrow (b\#c)$.
 - б. Исключаем $b \rightarrow c$ и $b \rightarrow d$, если существует $(b\#c) \rightarrow d$.
6. Соединить начальное и конечное события следующим образом:
 - а. Если в наборе T_L несколько задач, то соединить начальное событие с логическим оператором (XOR или AND). В противном случае, соединить начальное событие только с одной первой задачей.
 - б. Для каждой последней задачи в наборе T_0 добавить конечное событие и соединить с последней из задач.
7. Построить дуги потока следующим образом:
 - а. Шаблон (а): для каждого $a \rightarrow b$ в Y_L нарисовать дугу из a в b .
 - б. Шаблон (б): Для каждого $a \rightarrow (b\#c)$ в Y_L нарисовать дугу от a до логического оператора XOR, и оттуда из b в c .
 - с. Шаблон (с): Для каждого $(b\#c) \rightarrow d$ в Y_L нарисовать дугу из b и c в логический оператор XOR, а из него – в d .

d. Шаблон (d) и (e): если задача в построенной таким образом модели процесса имеет несколько входящих или исходящих дуг, соединить эти дуги с помощью логического разделителя AND или объединяющего оператора AND соответственно.

8. Вернуть вновь созданную модель процесса.

На рис. 4 показано, какие порядковые отношения можно получить на основе журнала рабочих процессов.

	a	b	c	d	e	f	g	h	i	j	k	l
a	#	→	→	#	#	#	#	#	#	#	#	#
b	←	#	#	#	#	#	→	#	#	#	#	#
c	←	#	#	→	#	#	#	#	#	#	#	#
d	#	#	←	#	→	#	#	#	#	#	#	#
e	#	#	#	←	#	→	#	#	#	#	#	#
f	#	#	#	#	←	#	→	#	#	#	#	#
g	#	←	#	#	#	←	#	→	#	→	#	#
h	#	#	#	#	#	#	←	#	→		#	#
i	#	#	#	#	#	#	#	←	#	#		→
j	#	#	#	#	#	#	←		#	#	→	#
k	#	#	#	#	#	#	#	#		←	#	→
l	#	#	#	#	#	#	#	#	←	#	←	#

Рис. 4. Представление матрицы рабочих процессов $L = \langle [a,b,g,h,j,k,i], \langle a,c,d,e,f,g,j,h,i,k,l \rangle \rangle$

4. Представление модели в сети Петри

Сети Петри – это самый давний и наиболее изученный язык моделирования процессов, который определяется через $\langle P,T,I,O \rangle$, где

- P, T – конечные множества позиций и переходов;
- I – дуги, имеющие направление от позиций к переходам;
- O – дуги от переходов к позициям.

Множество позиций и множество переходов не пересекаются.

Хотя графическая нотация является интуитивно понятной и простой, сети Петри являются используются для анализа во многих методы и системах. Состояние сети Петри определяется распределением точек по позициям и обозначается как его маркировка.

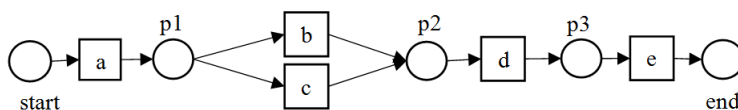


Рис. 5. Пример сети Петри, где a, b, c, d, e – переходы, $p1, p2, p3$ – позиции

Они составляют набор следующих отношений для формализации в сетях Петри:

1. Основные порядковые отношения следования (рис. 6).

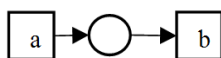


Рис. 6. Шаблон следования $a \rightarrow b$

2. Отношение параллелизма. Возникает, когда $a \rightarrow b, a \rightarrow c, b || c$ или $b \rightarrow a, c \rightarrow a, b || c$ (рис. 7, 8).

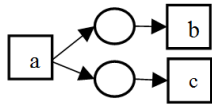


Рис. 7. AND-split

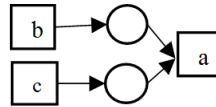


Рис. 8. AND-join

3. Отношение выбора или итерации. Справедливо при $a \rightarrow b$, $a \rightarrow c$, $b \parallel c$ или $b \rightarrow a$, $c \rightarrow a$, $b \parallel c$ (рис. 9, 10).

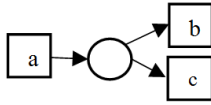


Рис. 9. XOR-split

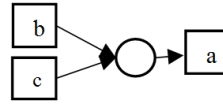


Рис. 9. XOR-join

Для того, чтобы создать наше представление, мы используем матрицу рабочих процессов и шаблоны отношений. Таким образом, мы создаем наше представление модели (рис. 5).

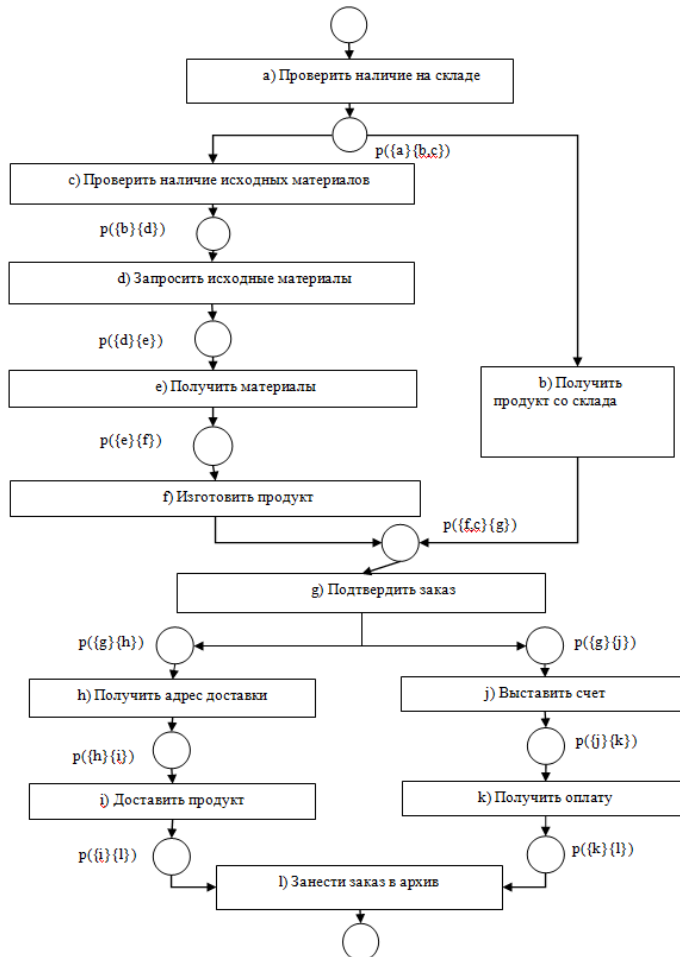


Рис. 5. «Осуществление заказа». Сети Петри

Заключение

В настоящей работе была решена задача интеллектуальной обработки данных, которая проявляется в целом через данные о выполнении процесса, а в частности – через журналы событий. Эти данные служат для контроля и мониторинга процессов предприятия.

На основе выполнения процесса «осуществление заказа» была рассмотрена структура журнала событий, основная идея Process Mining, последовательные шаги альфа-алгоритма и его отношения. Был изложен один из основных языков моделирования – сети Петри, и реализовано представление модели из журнала событий.

ЛИТЕРАТУРА

1. Hand D., Mannila H., Smyth P. Principles of Data Mining. – MIT Press, Cambridge, MA, 2001.

АНАЛИЗ ЖУРНАЛА СОБЫТИЙ МЕТОДАМИ PROCESS MINING

Д.М. Хуракай

Российский университет дружбы народов
homelinkin@mail.ru

Введение

Process Mining – это интеллектуальный анализ процессов. Представляет собой особую технику анализа бизнес-процессов. Подход, нацеленный на решение трёх задач: выявление или обнаружение процессов, проверка соответствия и расширение модели. Начало анализа идёт не от моделирования к реализации, а процесс исследуется сначала с внутренней стороны. При этом речь идёт не только о автоматическом построении модели процесса в виде диаграммы, но и об определении его количественных характеристик.

Свое начало Process Mining берёт с журнала событий, который предоставляет данные для анализа, на основе которых создаётся будущая модель через альфа-алгоритм. После разработки модели начинается интеллектуальный анализ, где на первом этапе происходит восстановление модели бизнес-процесса при помощи анализа журнала информационной системы, на втором этапе производится проверка соответствия модели бизнес-процесса экземплярам бизнес-процесса, полученным из событийного журнала информационной системы, и в конце проводится последующее улучшение (расширение) исходных процессов посредством получения информации из журнала.

В данной статье приводится исследование анализа журнала событий «Возврат билетов» с помощью методов Process Mining: обнаружение, проверка на соответствие.

1. Обнаружение

Журнал событий – это данные, где владелец процесса может просмотреть, как участник процесса выполняет задачи, и, опираясь на информацию, делать выводы по ним. Журнал событий является инструментом, который направлен на грамотный анализ процесса, его методы направлены на обнаружение слабых мест или ненужных вовсе.

Существуют алгоритмы, которые преобразовывают данные из журнала событий в некоторую структуру, например, в BPMN-диаграмму. Одним из таких инструментов является альфа-алгоритм, который является одним из самых первых и простых.

Альфа-алгоритм является основным методом для автоматического обнаружения процесса, т.е. он является внутренним инструментом именно для одного из методов Process Mining. Идея алгоритма состоит в том, что задачи, которые следуют друг за другом в журнале, должны быть непосредственно связаны в модели процесса.

Рассмотрим этап «Обнаружение» на основе простейшего примера журнала событий (рис. 1).

Case ID	Event ID	dd-MM-yyyy:HH.mm	Activity	Resource	Costs	
1	35654423	30-12-2010:11.02	register request	Pete	50 a	
2	1	35654424	31-12-2010:10.06	examine thoroughly	Sue	400 b
3	1	35654425	05-01-2011:15.12	check ticket	Mike	100 c
4	1	35654426	06-01-2011:11.18	decide	Sara	200 d
5	1	35654427	07-01-2011:14.24	reject request	Pete	200 f
6	2	35654483	30-12-2010:11.32	register request	Mike	50 a
7	2	35654485	30-12-2010:12.12	check ticket	Mike	100 c
8	2	35654487	30-12-2010:14.16	examine casually	Sean	400 g
9	2	35654488	05-01-2011:11.22	decide	Sara	200 d
10	2	35654489	08-01-2011:12.05	pay compensation	Ellen	200 e
11	3	35654521	30-12-2010:14.32	register request	Pete	50 a
12	3	35654522	30-12-2010:15.06	examine casually	Mike	400 g
13	3	35654524	30-12-2010:16.34	check ticket	Ellen	100 c
14	3	35654525	06-01-2011:09.18	decide	Sara	200 d
15	3	35654526	06-01-2011:12.18	reinitiate request	Sara	200 k
16	3	35654527	06-01-2011:13.06	examine thoroughly	Sean	400 b
17	3	35654530	08-01-2011:11.43	check ticket	Pete	100 c
18	3	35654531	09-01-2011:09.55	decide	Sara	200 d
19	3	35654533	15-01-2011:10.45	pay compensation	Ellen	200 e
20	4	35654641	06-01-2011:15.02	register request	Pete	50 a
21	4	35654643	07-01-2011:12.06	check ticket	Mike	100 c
22	4	35654644	08-01-2011:14.43	examine thoroughly	Sean	400 b
23	4	35654645	09-01-2011:12.02	decide	Sara	200 d

Рис. 1. Отрывок из журнала событий «Возврат билетов»

Дан журнал событий «Возврат билетов», в котором собраны данные за некоторый промежуток времени одной туристической компании. Всего в журнале 6 событий, все они имеют разный сценарий действий. Для работы с журналом событий можно применить альфа-алгоритм.

«Case ID» является сводом, в котором хранится информация про один случай, туда записываются все события, происшедшие в рамках одного лица, в данном случае, с билетами. Также каждому событию присваивается номер – «Event ID», фиксируется время, когда событие произошло – «dd-MM-yyuu:HH.mm». Под «Activity» записываются сами события, в данной задаче, «Resource» – это сотрудник турфирмы, совершивший действие, «Cost» – это цена происшедшего события.

Для удобства применение алгоритмов и методов каждому событию было присвоено буквенное значение.

На первом этапе альфа-алгоритма нам необходимо выделить все порядковые отношения. Под порядковыми отношениями подразумевается последовательность событий, следующая друг за другом. Для обозначения порядкового отношения используется символ «>», например, после событие a идёт событие b , $a > b$ (см. табл. 1).

Таблица 1

Порядковые отношения журнала событий «Возврат билетов»

$a > b$	$b > c$	$c > d$	$d > f$	$c > g$	$g > d$
$d > e$	$a > g$	$g > c$	$k > c$	$d > k$	$k > b$
$k > g$	$a > c$	$c > b$	$b > d$		

На втором этапе выделяют причинную связь, она выполняется, если $a > b$, но не $a \succ b$. Это справедливо для всех пар, кроме (b,c) и (g,c) , получается:

Таблица 2

Причинные отношения журнала событий «Возврат билетов»

$a \rightarrow b$	$a \rightarrow c$	$a \rightarrow g$	$c \rightarrow d$	$d \rightarrow f$	$g \rightarrow d$
$d \rightarrow e$	$k \rightarrow c$	$d \rightarrow k$	$k \rightarrow b$	$k \rightarrow g$	$b \rightarrow d$

Также существуют потенциальные параллельные отношения справедливы для $b \parallel c$ и $g \parallel c$.

Далее, используем журнал событий «Возврат билетов» и создаём таблицу порядковых отношений. Эти идентифицированные отношения относятся к задачам, которые непосредственно следуют друг за другом в журнале.

Таблица симметрична относительно диагонали, содержит три отношения: причинные отношения (\rightarrow или \leftarrow), параллельные отношения (\parallel), отношения отсутствия наследия ($\#$). Получаем:

Таблица 3

Представление матрицы рабочих процессов «Возврат билетов»

	a	b	c	d	e	f	g	k
a	#	\rightarrow	\rightarrow	#	#	#	\rightarrow	#
b	\leftarrow	#	\parallel	\rightarrow	#	#	#	\leftarrow
c	\leftarrow	\parallel	#	\rightarrow	#	#	\parallel	\leftarrow
d	#	\leftarrow	\leftarrow	#	\rightarrow	\rightarrow	\leftarrow	\rightarrow
e	#	#	#	\leftarrow	#	#	#	#
f	#	#	#	\leftarrow	#	#	#	#
g	\leftarrow	#	\parallel	\rightarrow	#	#	#	\leftarrow
k	#	\rightarrow	\rightarrow	\leftarrow	#	#	\rightarrow	#

На основе полученной таблицы строится BPMN-диаграмма.

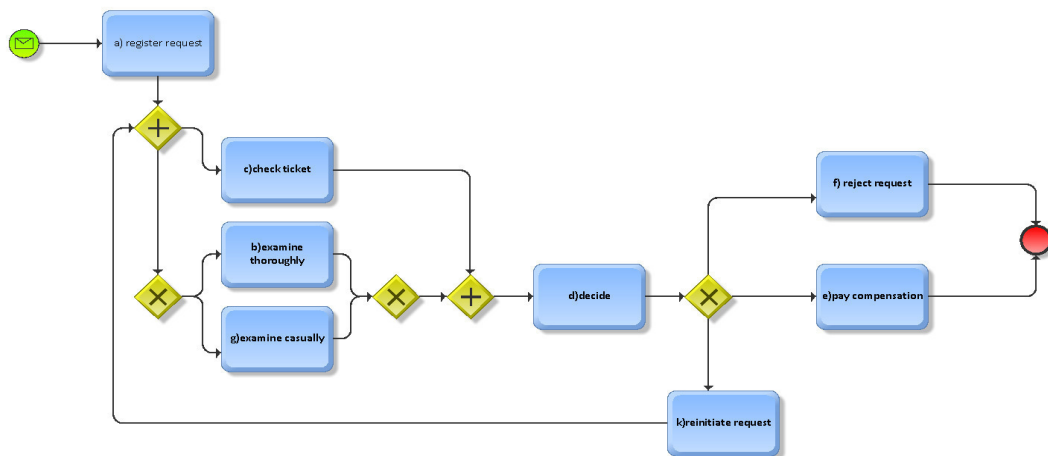


Рис. 2. Модель процесса «Возврат билетов», построенная по альфа-алгоритму

Провести проверку BPMN-диаграммы можно через простой способ: из журнала событий рассматривается случай. Для примера рассмотрим одно из событий (Case ID = 3). Выписываются последовательности действий из события: *agcdkcgdkgcdf*. Далее данная последовательность проходит «прогон» по модели. Если не возникло никаких ошибок, то BPMN-диаграмма считается верной при условии, что другие события были пройдены без ошибок.

Модель процесса можно построить в программном обеспечении ProM, которое реализует альфа-алгоритм и распечатывает граф, построенный через сети Петри.

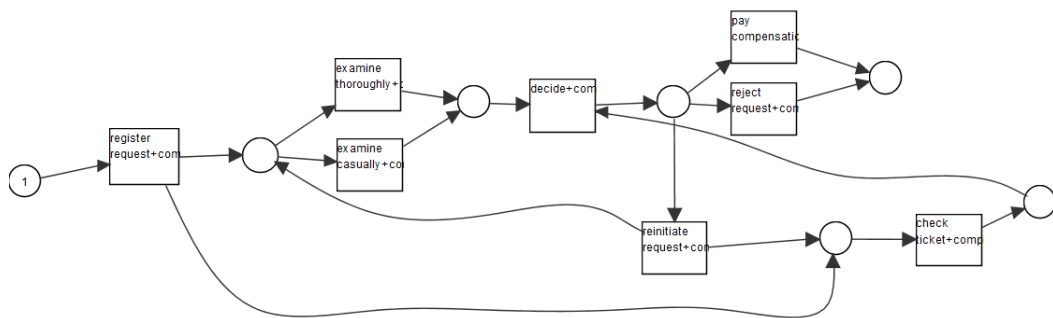


Рис. 3. Модель процесса «Возврат билетов», построенная в ProM

После того, как модель реализовалась, необходимо провести проверку самого альфа-алгоритма. Дело в том, что могут возникнуть последующие проблемы с альфа-алгоритмом, а именно – с незавершённостью и шумом. Существует несколько подходов для решения этих проблем. В общем, они направлены на то, чтобы сбалансировать четыре основных критерия качества: гибкость, простота, точность и обобщение. Гибкость относится к поведению журнала, где модель процесса можно проанализировать. Простота означает, что полученные модели процессов должны быть максимально легкими и понятными. Она может быть измерена с помощью различных методов определения сложности моделей процессов, таких как модель размера или степени структурированности. Точность означает степень поведения, которая позволяет в самой модели, но при этом остается не замеченной в логах. Обобщение относится к способности процесса абстрагироваться от поведения, которое описано в журналах. В контексте решаемой задачи удовлетворяются только простота, обобщение, для остальных двух факторов недостаточно информации из журнала событий.

Понятие полноты, принимаемое для альфа-алгоритма относится к связи, которая является для остальных отношений производной. Количество рассматриваемых случаев увеличивается в факториале числа, равному количеству параллельных задач. Поэтому желательно использовать алгоритмы, которые помогут различать правдоподобные и неправдоподобные поведенческие журналы событий для того, чтобы обобщить или сократить количество процессов, пока процессы из журнала не завершатся. Здесь также хорошо применимы методы решения проблем с шумом. В журнал событий часто включаются случаи с отсутствующей головой, хвостом, или отсутствует промежуточный эпизод. Кроме того, могут быть ошибки в записях журнала событий, они могут быть обменены или записаны дважды. Такие записи не должны искажать процессы нахождения результата.

В данной задаче, где всего случаев шесть, легко проверить это вручную простой сверкой данных из журнала и диаграммы. Диаграммы построены корректно, каждый случай начинается с регистрации – *a*, заканчивается одним из двух конечных событий – *e, f*, отсутствуют промежуточные дыры, все события записаны корректно (см. рис. 1).

2. Проверка на соответствие

С помощью проверки на соответствие выявляются нарушения и делаются выводы об этих нарушениях. Когда правила не соблюдаются, следует предпринять корректирующие действия. Нарушения могут быть связаны с одной из трёх процессов перспективы потока управления, данных и ресурсов в отдельности или в комбинации.

Соответствие потока управления можно изучать двумя способами: либо на основе явных ограничений, либо на основании нормативной модели процесса. В рамках решаемой задачи информации недостаточно, соответственно, соответствие потока управления не может быть проведено тщательно. Но если рассматривать, что все случаи на-

чинаются и заканчиваются определёнными событиями, то можно сделать вывод о том, что нормативная деятельность модели выполняется.

Помимо ограничений на управление потоком существуют дополнительные ограничения на данные и на ресурсы. В задаче ресурсами являются сотрудники турфирмы. Если рассмотреть весь журнал событий, то можно заметить, что Сара является единственным работником, который рассматривает одобрение того или иного случая, скорее всего, это связано с тем, что она занимает руководящую должность (см. рис. 4).

Такие ограничения могут быть проверены в журнале, где определённое поле данных принимает конкретное значение. Пример дополнительной необходимости какой-то подписи уже указывает на соответствие данных и ограниченности ресурсов. Например, необходимость согласовать с Сарой принятие решений, т.е. необходимо утвердить дан-ные.

1	4E+07	05-01-2011-15.12	check ticket	Mike	100	c
1	4E+07	06-01-2011-11.18	decide	Sara	200	d
1	4E+07	07-01-2011-14.24	reject request	Pete	200	f
2	4E+07	30-12-2010-11.32	register request	Mike	50	a
2	4E+07	30-12-2010-12.12	check ticket	Mike	100	c
2	4E+07	30-12-2010-14.16	examine casually	Sean	400	g
2	4E+07	05-01-2011-11.22	decide	Sara	200	d
2	4E+07	08-01-2011-12.05	pay compensation	Ellen	200	e
3	4E+07	30-12-2010-14.32	register request	Pete	50	a
3	4E+07	30-12-2010-15.06	examine casually	Mike	400	g
3	4E+07	30-12-2010-16.34	check ticket	Ellen	100	c
3	4E+07	06-01-2011-09.18	decide	Sara	200	d
3	4E+07	06-01-2011-12.18	reinitiate request	Sara	200	k
3	4E+07	06-01-2011-13.06	examine thoroughly	Sean	400	b
3	4E+07	08-01-2011-11.43	check ticket	Pete	100	c
3	4E+07	09-01-2011-09.55	decide	Sara	200	d
3	4E+07	15-01-2011-10.45	pay compensation	Ellen	200	e
4	4E+07	06-01-2011-15.02	register request	Pete	50	a
4	4E+07	07-01-2011-12.06	check ticket	Mike	100	c
4	4E+07	08-01-2011-14.43	examine thoroughly	Sean	400	b
4	4E+07	09-01-2011-12.02	decide	Sara	200	d
4	4E+07	12-01-2011-15.44	reject request	Ellen	200	f
5	4E+07	06-01-2011-09.02	register request	Ellen	50	a
5	4E+07	07-01-2011-10.16	examine casually	Mike	400	g
5	4E+07	08-01-2011-11.22	check ticket	Pete	100	c
5	4E+07	10-01-2011-13.28	decide	Sara	200	d
5	4E+07	11-01-2011-16.18	reinitiate request	Sara	200	k
5	4E+07	14-01-2011-14.33	check ticket	Ellen	100	c
5	4E+07	16-01-2011-15.50	examine casually	Mike	400	g
5	4E+07	19-01-2011-11.18	decide	Sara	200	d
5	4E+07	20-01-2011-12.48	reinitiate request	Sara	200	k
5	4E+07	21-01-2011-09.06	examine casually	Sue	400	g
5	4E+07	21-01-2011-11.34	check ticket	Pete	100	c
5	4E+07	23-01-2011-13.12	decide	Sara	200	d
5	4E+07	24-01-2011-14.56	reject request	Mike	200	f
6	4E+07	06-01-2011-15.02	register request	Mike	50	a
6	4E+07	06-01-2011-16.06	examine casually	Ellen	400	g
6	4E+07	07-01-2011-16.22	check ticket	Mike	100	c
6	4E+07	07-01-2011-16.52	decide	Sara	200	d
6	4E+07	16-01-2011-11.47	pay compensation	Mike	200	e

Рис. 4. Журнал событий «Возврат билетов»

Заключение

В настоящей работе рассмотрен интеллектуальный анализ журнала событий «Возврат билетов», с помощью альфа-алгоритма была получена модель, представленная в виде диаграмм, были использованы процесс обнаружения и проверка на соответствие, представлены теоритические выводы на их основе.

МЕТОДИКА РАЗРАБОТКИ АЛГОРИТМОВ ДЛЯ СЖАТИЯ ГИПЕРСПЕКТРАЛЬНЫХ АЭРОКОСМИЧЕСКИХ ИЗОБРАЖЕНИЙ, ПРИМЕНИМЫХ НА БОРТУ КОСМИЧЕСКИХ АППАРАТОВ

А.Ж. Сарина, А.В. Замятин

Томский государственный университет
sarinova@mail.tsu.ru, zamyatin@mail.tsu.ru

Введение

Работа посвящена описанию методики построения разработки алгоритмов сжатия гиперспектральных аэрокосмических изображений (АИ) аппаратной реализации с целью последующего применения в системах дистанционного зондирования Земли (ДЗЗ). В качестве алгоритмов сжатия, необходимого для сокращения объёма передаваемой информации, предлагается использование разработанных методов сжатия на основе разностных и дискретных преобразований.

В работе рассмотрена методика разработки алгоритмов сжатия, с учётом которой разработан адаптивный алгоритм для сжатия гиперспектральных АИ с использованием программируемых устройств. Проведённые исследования показали, что предложенные алгоритмы имеют эффективность, достаточную для использования, и могут быть применены на борту космических аппаратов (КА) при передаче гиперспектральных данных ДЗЗ в условиях ограниченной ёмкости буферной памяти и пропускной способности канала связи.

1. Постановка задачи

Гиперспектральные АИ ДЗЗ важны для наблюдения и исследований изменений поверхности Земли, мониторинга природных ресурсов и последствий чрезвычайных ситуаций и т.д. В настоящее время разработка программных систем передачи таких данных является актуальной задачей. В решении данной задачи существуют два направления исследования: разработка алгоритмов сжатия, используемых в наземных центрах приёма и обработки данных ДЗЗ; применяемых на борту КА. Активно ведутся исследования в области разработки алгоритмов сжатия первого направления, для которых имеются множество публикаций [1–10]. Во втором направлении исследования имеется потенциал для разработки алгоритмов, диктуемый необходимым перечнем проблем для решения задачи сжатия.

Предлагается сформулировать некоторые проблемы разработки алгоритмов сжатия гиперспектральных АИ, применимых на борту.

- *Объём поступающих данных.* Гиперспектральные АИ имеют диапазон в несколько сотен и тысяч спектральных каналов. При этом размер каналов достигает сотни тысяч пикселей.

- *Ресурсы памяти.* Технические характеристики аппаратного обеспечения, которое принимает, хранит и передаёт такие данные, ограничены в объёме памяти и вычислительной мощности.

- *Передача данных.* Огромные объёмы формируются на борту КА с определённой скоростью, превышающей возможности передаваемого канала связи, имеющего фиксированную пропускную способность, недостаточную для передачи несжатых гиперспектральных АИ.

- *Качество данных.* Требования к качеству данных очень высокие, т.к. поступающая информация уникальна.

Применение сжатия в таких условиях невозможно. Перечисленные выше проблемы, возникающие при разработке программных систем, применимых на борту КА, порождают следующий спектр требований к алгоритму сжатия:

- скорость формирования потока сжатых данных;

- низкая вычислительная сложность;
- высокий коэффициент сжатия;
- контроль ошибок при сжатии с потерями.

В настоящий момент существуют и используются для сжатия гиперспектральных АИ известные алгоритмы на основе усечённого блочного кодирования [1], дифференциально-импульсной модуляции [1], дискретно-косинусного преобразования [2], дискретного вейвлет-преобразования [2]. Данные алгоритмы на боту КА, требующие больших вычислительных ресурсов, не всегда удовлетворяют вышеперечисленным требованиям к сжатию. Поэтому требуется разработка новых методов сжатия гиперспектральных АИ, не уступающих по своей эффективности известным методам, но требующих меньших вычислительных ресурсов.

Таким образом, в данной работе предлагается рассмотреть методику разработки алгоритмов сжатия гиперспектральных АИ, удовлетворяющих указанным требованиям.

2. Описание методики разработки алгоритмов сжатия

Методику построения разработки алгоритма аппаратной реализации сжатия следует разделить на три этапа.

Первым этапом является выбор и обоснование аппаратного и программного обеспечения для сжатия гиперспектральных АИ.

Вторым этапом является разработка алгоритмов сжатия без потерь и с потерями на программируемых интегральных схемах.

Третий этап – тестирование и проведение экспериментов разработанных алгоритмов в сравнении с известными алгоритмами по производительности скорости и эффективности в степени сжатия.

Этап 1. Для проверки разработанных алгоритмов на первом этапе в качестве тестируемой программируемой интегральной схемы была выбрана 32-разрядная микросхема фирмы STM32 фирмы Electronics и программное обеспечение CodoxIDE со встроенным языком C++.

Одними из последних процессоров для встраиваемых систем являются процессоры, основанные на архитектуре *ARM Cortex-M3*. Эти процессоры предназначены для использования в цифровой обработке сигналов (Digital Signal Processing, DSP).

Общее описание архитектуры ARM и 32-разрядных микроконтроллеров STM. Процессоры ARM являются ключевым компонентом для большого количества успешных 32-битных встраиваемых систем. Процессоры ARM широко используются в мобильных телефонах, планшетах и других портативных устройствах. ARM основаны на RISC-архитектуре, что позволяет уменьшить потребление энергии процессором и, таким образом, делает их идеальным выбором для встраиваемых систем.

Преимущества использования:

1. Переменное количество циклов выполнения для простых инструкций. Простые инструкции ARM требуют на выполнение более одного цикла. Например, выполнение инструкций Load и Save зависит от количества регистров, которые им переданы.

2. Возможность соединять команды сдвига и вращения с командами обработки информации.

3. Условное выполнение – инструкция выполняется только в том случае, если выполняется конкретное условие. Это увеличивает производительность и позволяет избавиться от операторов ветвления.

4. Улучшенные инструкции – процессоры ARM поддерживают улучшенные DSP-инструкции для операций с цифровыми сигналами.

Имеются два программных продукта, которые реализуют функциональность для выполнения прошивки без использования среды разработки: *STM32 ST-LINK Utility* и *ST Visual Programmer*.

Программа *STM32 ST-LINK Utility* предназначена для работы с 32-разрядными контроллерами через интерфейс *ST-LINK*. Общая характеристика ядра микроконтроллеров STM32 представлены в табл. 1.

Таблица 1

Основные характеристики ядра микроконтроллеров STM32

Характеристика	Значение
Ширина слов для данных, разряд	32
Архитектура	Гарвардская
Конвейер	3-ступенчатый
Набор инструкций	RISC
Организация памяти программ, разряд	32
Буфер предвыборки, разряд	2x64
Средний размер инструкции, байт	2
Тип прерываний	Векторизованные
Задержка реагирования на прерывания	12 циклов
Режимы управления энергопотреблением	Сон, сон по выходу, глубокий сон
Отладочный интерфейс	ST-LINK, JTAG

Микроконтроллеры данного типа построены на гарвардской архитектуре и имеют 3-ступенчатый конвейер, который минимизирует время выполнения команд. Они разработаны для построения систем с максимальной энергоэффективностью и имеют несколько режимов управления энергопотреблением. В них используются внутренние интерфейсы памяти шириной больше, чем средняя длина инструкции. Это минимизирует число доступов к шине памяти, следовательно, и потребление электроэнергии, связанное с операциями по шине и чтением энергонезависимой памяти. Технология непрерывной обработки прерываний с исключением внутренних операций над стеком сокращает время реакции на прерывания и исключает лишние операции со стеком.

Плата *STM32F4 Discovery* оснащена:

- микроконтроллером STM32F407VGT6 с ядром Cortex-M3 тактовой частотой 168 МГц, 1 Мб Flash-памяти, 192 Кб RAM;
- отладчиком ST-Link для отладки и программирования;
- питанием платы через USB или от внешнего источника питания 5 В;
- датчиком движения ST MEMS LIS302DL и выходами цифрового акселерометра;
- датчиком звука ST MEMS MP45DT02;
- звуковым ЦАП CS43L22;
- восемью светодиодами;
- двумя кнопками (для программирования пользователем и для перезапуска).

Таким образом, отладочная плата оснащена большим количеством периферии, что позволяет реализовывать на ней алгоритмы различной сложности.

Этап 2. Разработан оригинальный и эффективный алгоритм сжатия без потерь гиперспектральных АИ методом регрессионного преобразования. Алгоритм сжатия гиперспектральных аэрокосмических изображений без потерь, отличающийся применением по-канально разностного линейного регрессионного преобразования, существенно уменьшающего диапазон изменения данных и увеличивающего за счёт этого степень сжатия. Основная идея преобразования заключается в алгоритме с нахождением пар коррелированных каналов с последующим созданием преобразованных блоков без потерь с помощью регрессионного анализа, позволяющий уменьшить размеры каналов аэрокосмического изображения и преобразовать их перед сжатием модифицированного алгоритма Хаффмана.

Разработаны несколько ортогональных преобразований, адаптированных для сжатия гиперспектральных АИ с потерями. Разработан адаптивный алгоритм дискретно-косинусного преобразования с последующем квантованием с уровнем потерь и сжатием кодированием Хаффмана.

Адаптированный алгоритм Хаффмана. Известная стандартная таблица кодов Хаффмана рекомендована для JPEG-изображений, не для гиперспектральных АИ, поэтому для кодирования предлагается сгенерированная. Модификация таблицы заключается в следующем:

1) каждой возможной паре RZ присваиваем натуральное число intMerge , начиная с двойки;

2) рассмотрим двоичную запись числа intMerge и будем обращать внимание на его длину (lenMerge). К примеру, для $(R;Z) = (4;7)$ $\text{intMerge} = 117$, двоичная запись – 1110101, $\text{lenMerge} = 7$;

3) построим код следующим образом: возьмём единицу ($\text{lenMerge} - 1$) раз, приписываем 0 и двоичную запись числа intMerge без лидирующей единицы. Для наглядности после нуля припишем апостроф: 1111110'110101.

Для примера из таблицы: $(R;Z) = (14;1)$ $\text{intMerge} = 17$, двоичная запись – 10001, $\text{lenMerge} = 5$. Код Хаффмана: 11110'0001.

Число Z не превышает 14, так как последовательность из 15-ти нулей мы заменяем кодом 111111110'00000000 (как если бы intMerge равнялось 256). Код для конца блока – 00. Пара $(R;Z) = (15;0)$ кодируется как 01. Если R превышает 16, то присваиваем intMerge , начиная с 257, аналогично.

Данная модификация позволила повысить степень сжатия гиперспектральных АИ за счёт проведённого анализа частот пар R и Z, т.к. более встречаемым величинам при кодировании выделяются коды меньшей длины.

На втором этапе для повышения вычислительной эффективности сжатия был разработан и организован алгоритм многопоточной обработки сжатия.

Алгоритм нескольких потоков кодирования гиперспектральных АИ

1) определение количества кодируемых пар в множестве упорядоченных пар (МУП);

2) первый поток получает и кодирует главный порождающий файл;

3) второй и последующие потоки кодируют последующие пары в МУП;

4) по освобождению все последующие потоки получают следующую пару каналов, пока МУП не пусто;

5) по завершению кодирования освобождаем все потоки.

Алгоритм одного потока при декодировании

1) рассматриваем следующую пару в МУП. Пусть это будет k_1 и k_2 , где k_1 – порождающий файл, k_2 – регрессируемый;

2) если файл k_1 является главным порождающим, или он уже декодирован, то декодируем пару ($k_1 \rightarrow k_2$);

3) если файл k_1 еще не декодирован (возможно, находится в процессе), то рассматриваем следующую пару после k_1, k_2 в МУП (не отбрасывая её);

4) переход к шагу 2 – если найдена такая пара, переходим в режим ожидания – если нет.

5) по завершении декодирования оповещаем остальные потоки в ожидании, переходим к шагу 1 (если МУП не пусто), либо завершаем исполнение потока. При этом отбрасываем закодированную пару ($k_1 \rightarrow k_2$) из МУП.

3. Эксперименты разработанных алгоритмов аппаратной реализации

Для определения эффективности предлагаемого адаптивного алгоритма с точки зрения степени сжатия и вычислительной эффективности, а также пределов его применимости проведен ряд экспериментов на гиперспектральных АИ (система ДЗЗ *Aviris*) (рис. 1).

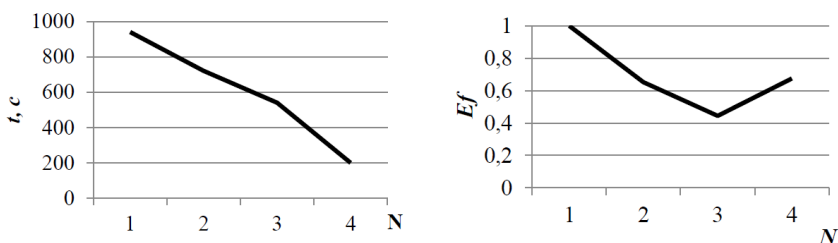


Рис. 1. Адаптированный алгоритм сжатия гиперспектральных АИ: слева – время вычисления, справа – эффективность выполнения

Эксперименты выполнены на ПЭВМ с процессором IntelCore i5 2,5 ГГц и объёмом оперативной памяти 4 Гб под управлением операционной системы Windows 8.1.

В качестве аналогов взяты алгоритмы сжатия JPEG Lossless/ Lossy, архиватор 7Z, Winrar. Результаты экспериментов сжатия предложенной методикой представлены в табл. 2, 3. Для сравнения взяты усредненные по всем экспериментам показатели степени сжатия и скорость обработки.

Таблица 2

Сравнение алгоритмов сжатия без потерь

Алгоритм сжатия	Коэффициент сжатия	Время вычисления, с
JPEG Lossless	3,5	221452
7z	3,8	205146
Winrar	3,9	220738
Адаптивный алгоритм	5,9	74816

Эксперименты проводились на ПЭВМ с процессором IntelCore i5 2,5 ГГц и объёмом оперативной памяти 4 Гб под управлением операционной системы Windows 8.1.

Таблица 3

Сравнение алгоритмов сжатия с потерями

Алгоритм сжатия	Коэффициент сжатия	Время вычисления, с
JPEG Lossy	5,7	225672
Адаптивный алгоритм	9,2	78235

Следует отметить, что в данных экспериментах адаптивный алгоритм показывает результаты по степени сжатия выше по сравнению с известными решениями более чем на 55%, в скорости обработки существенно превосходит в 3 раза.

Заключение

По результатам исследования показано, что предложенная методика разработки алгоритмов сжатия без потерь и с потерями на основе аппаратной реализации обладает высокой эффективностью в степени сжатия и скорости обработки. Это объясняется тем, что для такой реализации алгоритмы были адаптированы с целью возможности применения на борту КА. Данная методика построения алгоритмов обладает высоким быстродействием за счёт перехода на оборудование, оперирующее битами, обеспечивая увеличение быстродействия в разы. Адаптивные алгоритмы могут быть использованы как в системах ДЗЗ, так и в любых других приложениях обработки и сжатия гиперспектральных АИ

ЛИТЕРАТУРА

1. Гонсалес Р., Вудс Р. Цифровая обработка изображений. – М.: Техносфера, 2012. – С. 55–67.
2. Каикин В.Б., Сухинин А.И. Цифровая обработка аэрокосмических изображений. – Красноярск: СФУ, 2008. – 278 с.

3. Гашиников М.В., Глумов Н.И. Бортовая обработка гиперспектральных данных в системах дистанционного зондирования Земли на основе иерархической компрессии // Компьютерная оптика. – 2016. – Т. 40. – № 4. – С. 543–551.
4. Петров Е.П., Харина Н.Л., Сухих П.Н. Метод сжатия многозрядных спутниковых снимков без потерь // Современные проблемы дистанционного зондирования Земли из космоса. – 2016. – Т. 13. – № 2. – С. 203–210.
5. Fernando García-Vilchez, Jordi Muñoz-Mari, Maciel Zorteza, Ian Blanes, Vicente González-Ruiz, Gustavo Camps-Valls, Antonio Plaza. On the Impact of Lossy Compression on Hyperspectral Image Classification and Unmixing // IEEE Geoscience and remote sensing letters. – 2011. – V. 8. – №. 2.
6. Meena B. Vallakati, Dr. R. R. Sedamkar. Low Complexity DCT-based DSC approach for Hyperspectral Image Compression with Arithmetic Code. // International Journal of Computer Science Issues. – September 2012. – V. 9. – Is. 5. – № 1.
7. Yongjian Nian, Mi He, Jianwei Wan. Low-Complexity Compression Algorithm for Hyperspectral Images Based on Distributed Source Coding // Hindawi Publishing Corporation. Mathematical Problems in Engineering. – V. 2013. – Article ID 825673. – 7 p.
8. Ganeshraj P., Sivasankar A. Scalable Compression Method for Hyperspectral Images // Research Journal of Engineering Science. – 2013. – V. 2(3). – P. 1–5.
9. Diego Valsesia, Enrico Magli. A Novel Rate Control Algorithm for Onboard Predictive Coding of Multispectral and Hyperspectral Images. 2014.
10. Dr. S.M. Ramesh, P. Bharat, J. Anand, J. Anbu Selvan. Analysis of Lossy Hyperspectral Image Compression Techniques // International Journal of Computer Science and Mobile Computing. – V. 3. – Is. 2. – P. 302–307.

СЕКЦИЯ VIII. ТЕОРИЯ МАССОВОГО ОБСЛУЖИВАНИЯ И ТЕЛЕТРАФИКА

ИССЛЕДОВАНИЕ СИСТЕМЫ $M|GI|_{\infty}$ С НЕОРДИНАРНЫМ ВХОДЯЩИМ ПОТОКОМ И БЕСКОНЕЧНЫМ ЗНАЧЕНИЕМ СРЕДНЕГО ВРЕМЕНИ ОБСЛУЖИВАНИЯ

Е.Е. Худяшова, А.А. Назаров

Томский государственный университет
kopnova.e@gmail.com, nazarov.tsu@gmail.com

Введение

Системы массового обслуживания в настоящее время являются востребованными во многих областях науки и техники. Применение моделей теории массового обслуживания необходимо для анализа систем и решения большого перечня задач в самых различных областях: телекоммуникационные системы [1,2], социально-экономические системы [3], производственные системы [4,5], системы управления транспортными потоками [6–7], вычислительные системы [8].

Особое внимания требуется уделить моделям систем массового обслуживания, в которых имеется бесконечное число приборов, т.к. именно они позволяют адекватно описать сложные технические системы, для которых число приборов может быть достаточно велико. Например, Brown L., Gans N., Mandelbaum A., Sakov A. применяют такие системы для моделирования работы колл-центра, в которой агенты предоставляют телефонные услуги [9,10]. Обычно число работающих в такой компании операторов достаточно велико. И обслуживание клиента должно начинаться незамедлительно. Также бесконечнолинейные системы используют и в качестве аппроксимации для многолинейных систем в таких случаях, когда, например, вероятность отказа в обслуживании пренебрежимо мала [11–13].

На начальном этапе большинство исследований теории массового обслуживания проводилось в предположении того, что входящий поток заявок является простейшим [14,15]. Однако развитие компьютерных и мобильных систем привело к необходимости создания новых математических моделей потоков данных, которые являются непуассоновскими или неординарными потоками. Это и послужило причиной для увеличения интереса к исследованию систем с более сложными входящими потоками. Системы с непуассоновскими потоками изучали российские и зарубежные авторы, такие как П.П. Бочаров, А.В. Печинкин, А.А. Назаров, С.П. Моисеева, А.Н. Моисеев, D. Baum, E.A. Doorn, A.A. Jages и другие [16–20]. Исследованием неординарных потоков занимались Ушаков и Матвеев, именно в их книге [21] была получена производящая функция числа заявок в системе. Далее такой же результат продублирован у Климова [22]. Авторы книг [21,22] используют в качестве доказательства представление исходного потока в виде суперпозиции независимых потоков с одинаковыми объёмами пачек заявок, и изучение исходной системы сведено к такой, что требования поступают группами фиксированного объёма.

Результаты, полученные в этой статье, согласуются с [21,22], но получены с помощью другого метода – метода динамического просеивания. Этот метод ещё достаточно новый, и поэтому его рассмотрение является актуальным. Таким образом, данная статья посвящена получению нестационарного распределения вероятностей числа заявок в системе с неординарным входящим потоком, произвольной функцией распределения обслуживания приборов и бесконечным значением среднего времени обслуживания с

помощью метода динамического просеивания, предложенными на кафедре ТВиМС НИТГУ.

1. Постановка задачи

Рассмотрим (рис. 1) систему массового обслуживания с неограниченным числом приборов.

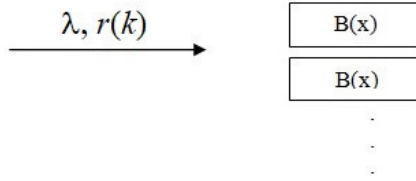


Рис 1. Система массового обслуживания с неограниченным числом приборов, функцией распределения времени обслуживания $B(x)$, входящим неординарным потоком интенсивности λ и распределением $r(k)$ числа заявок в пачке

На вход этой системы поступает пуассоновский неординарный поток заявок, длины интервалов между моментами наступления событий в котором имеют экспоненциальное распределение с параметром λ , а количество k заявок в пачке, поступивших в момент наступления события, имеет дискретное распределение $r(k)$.

Продолжительности обслуживания заявок являются независимыми случайными величинами с функцией распределения $B(x)$ и бесконечным первым моментом, т.е. выполняется равенство

$$\int_0^{\infty} (1 - B(x)) dx = \infty. \tag{1}$$

Обозначим $i(t)$ число заявок (число занятых приборов) в системе в момент времени t . Для рассматриваемой системы, в силу условия (1) не существует стационарного распределения вероятностей значений процесса $i(t)$, поэтому его нестационарное распределение обозначим $P(i, t) = P\{i(t) = i\}$, полагая при этом, что в момент времени $t = 0$ система свободна, и в ней нет обслуживаемых заявок.

Задачей исследования в данной работе является нахождение распределения вероятностей числа заявок в системе, а также построение двух аппроксимаций этого распределения с целью оценки их точности.

Для решения поставленной задачи воспользуемся методом динамического просеивания [17].

2. Метод динамического просеивания для нахождения распределения вероятностей числа заявок в системе

Рассмотрим (рис. 2) две оси времени t . На первой оси отметим моменты наступления событий входящего рекуррентного потока, а также моменты времени $t = 0$ и $t = T$.

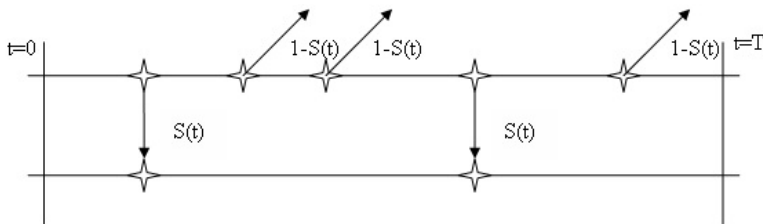


Рис. 2. Метод динамического просеивания заявок, поступающих в систему на обслуживание

Обозначим $S(t) = 1 - B(T - t)$, $0 \leq t \leq T$ – вероятность того, что заявка входящего потока, поступившая в момент времени $0 \leq t \leq T$, будет находиться в системе в момент времени $t = T$, занимая один из её приборов. Каждое событие входящего потока, наступившее в момент времени t , с вероятностью $S(t)$ просеивается на вторую ось времени, а с вероятностью $1 - S(t)$ не рассматривается.

По построению на второй оси времени генерируется нестационарный просеянный поток.

Обозначим $n(t)$ – число событий просеянного потока, наступивших за время t на интервале $[0; t]$. Рассмотрим $P\{n(t) = n\} = P(n, t)$.

$$P(n, t + \Delta t) = P(n, t)(1 - \lambda \Delta t) + \lambda \Delta t \sum_{v=0}^n P(n - v, t) \sum_{k=v}^{\infty} r(k) C_k^v (S(t))^v (1 - S(t))^{k-v} + o(\Delta t),$$

$$\frac{\partial P(n, t)}{\partial t} = -\lambda P(n, t) + \lambda \sum_{v=0}^n P(n - v, t) \sum_{k=v}^{\infty} r(k) C_k^v (S(t))^v (1 - S(t))^{k-v}.$$

Введём характеристическую функцию $H(u, t) = \sum_{n=0}^{\infty} e^{jun} P(n, t)$.

$$\frac{\partial H(u, t)}{\partial t} = -\lambda H(u, t) + \lambda \sum_{n=0}^{\infty} e^{jun} \sum_{v=0}^n P(n - v, t) \sum_{k=v}^{\infty} r(k) C_k^v (S(t))^v (1 - S(t))^{k-v}.$$

После несложных преобразований получим:

$$\frac{\partial H(u, t)}{\partial t} = \lambda H(u, t) \left\{ \sum_{k=0}^{\infty} r(k) \left[1 - (1 - e^{ju}) S(t) \right]^k - 1 \right\}. \quad (2)$$

Обозначим производящую функцию дискретного распределения числа заявок в пачке $R(z)$:

$$R(z) = \sum_{k=0}^{\infty} r(k) z^k, \quad (3)$$

где значение аргумента этой функции в (2) принимает вид $z = 1 - (1 - e^{ju}) S(t)$. Тогда уравнение (2) для характеристической функции числа занятых приборов принимает вид:

$$\frac{\partial H(u, t)}{\partial t} = \lambda H(u, t) \left\{ R(1 - (1 - e^{ju}) S(t)) - 1 \right\},$$

решение которого можно записать в виде

$$H(u, t) = \exp \left\{ \lambda \int_0^t \left[R(1 - (1 - e^{ju}) S(x)) - 1 \right] dx \right\},$$

т.е. для процесса $i(t)$ в момент времени $t = T$, выполнив обратное преобразование Фурье характеристической функции, получим распределение вероятности числа занятых приборов в системе:

$$P(i, T) = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{-jun} H(u, T) du. \quad (4)$$

3. Распределение $r(k)$ числа заявок в пачке

В качестве дискретного распределения $r(k)$ рассмотрим такое, для которого производящая функция, определяемая формулой (3) имеет вид:

$$R(z) = \left(\frac{1-\beta}{1-\beta z} \right)^\alpha, \quad 0 < \beta < 1, \quad \alpha > 0. \quad (5)$$

Такое распределение называется распределением Паскаля (отрицательным биномиальным). Его легко модифицировать в геометрическое, пуассоновское или биномиальное. Очевидно, чтобы из (5) получить производящую функцию геометрического распределения, нужно в (5) положить $\beta = p$ и $\alpha = 1$.

Для получения производящей функции биномиального распределения положим в (5) $\alpha = -n$ и $\beta = -\frac{p}{1-p}$:

$$\left(\frac{1-\beta}{1-\beta z} \right)^{-n} = \left(\frac{1-\beta z}{1-\beta} \right)^n = \left(\frac{1}{1-\beta} - \frac{\beta}{1-\beta} z \right)^n = ((1-p) + pz)^n.$$

Для получения производящей функции пуассоновского распределения устремим β к нулю и положим $\alpha = \frac{a}{\beta}$. Несложно показать, что $\lim_{\beta \rightarrow 0} R(z) = e^{a(z-1)}$, а это является производящей функцией пуассоновского распределения.

4. Дискретная гауссовская аппроксимация дискретного распределения вероятностей $P(i, T)$ числа заявок в системе

Дадим следующее определение. Дискретной гауссовской аппроксимацией $P_1(i, T)$ нестационарного распределения вероятностей $P(i, T)$ числа $i(T)$ заявок в момент времени $t = T$ в системе $M|GI|_\infty$ с неординарным входящим потоком будем называть распределение вероятностей $P_1(i, T)$, определяемое равенством

$$P_1(i, T) = (G(i+0.5, T) - G(i-0.5, T)) [1 - G(-0.5, T)]^{-1},$$

где $G(x, T)$ – функция гауссовского распределения с параметрами, определяемыми математическим ожиданием κ_1 и дисперсией κ_2 числа заявок в системе.

Для построения гауссовской аппроксимации введём кумулятивную функцию $K(u) = \ln(H(u)) = \lambda \int_0^T [R(1 - (1 - e^{ju})S(t)) - 1] dx$, проинтегрировав которую соответствующее количество раз и положив $u = 0$, получим семиинварианты числа заявок в системе $\kappa_1 = \lambda r_1 \int_0^T (1 - B(x)) dx$, $\kappa_2 = \kappa_1 + \lambda r_2 \int_0^T (1 - B(x))^2 dx$, где r_k – k -факториальный момент, полученный по формуле $r^{(k)}(z)|_{z=1} = r_k$, $k = 1, 2$.

Несложно показать, что третий семиинвариант числа заявок в системе имеет вид $\kappa_3 = \kappa_1 + 3\kappa_2 + \lambda r_3 \int_0^T (1 - B(x))^3 dx$, где r_3 – третий факториальный момент.

Рассмотрим функцию $H_3(u) = \exp \left\{ ju\kappa_1 + \frac{(ju)^2}{2} \kappa_2 + \frac{(ju)^3}{6} \kappa_3 \right\}$. Найдём действительную часть преобразования Фурье для этой функции: $P_3(i) = \text{Re} \left(\frac{1}{2\pi} \int_{-\pi}^{\pi} e^{-ju} H_3(u) du \right)$. Из последовательности чисел $P_3(i)$ оставим только не-

отрицательные $P_4(i) = \frac{1}{2}(P_3(i) + |P_3(i)|)$. Далее нормируем числа $P_4(i)$:

$P_2(i) = P_4(i) / \sum_{i=0}^{\infty} P_4(i)$. Полученное распределение $P_2(i)$ назовём модифицированной

аппроксимацией и сравним его с истинным распределением $P(i, T)$ для оценки его

точности, находить которую будем с помощью расстояния Колмогорова:

$\Delta = \max_{0 \leq i < \infty} \left| \sum_{n=0}^{\infty} (P(n, T) - P_m(n, T)) \right|$, где $m = 1, 2$, а нестационарное распределение вероятностей $P(n, T)$ определяется по формуле (4).

Моделирование системы $M|GI|_{\infty}$ с неординарным входящим потоком выполнялось при заданных значениях параметров $\lambda = 5$, $\alpha = 0,5$, $\beta = 0,8$ и заданной функцией распределения $B(x)$ времени обслуживания заявок. Функцию распределения $B(x)$, удовлетворяющую свойству (1), возьмём в виде $B(x) = \frac{x^{\gamma}}{1 + x^{\gamma}}$, $0 < \gamma \leq 1$, задав определённое

значение параметра $0 < \gamma \leq 1$.

На рис. 3, 4 приведены графики распределения вероятностей числа занятых приборов в системе и его аппроксимаций, полученных с помощью моделирования с заданными параметрами.

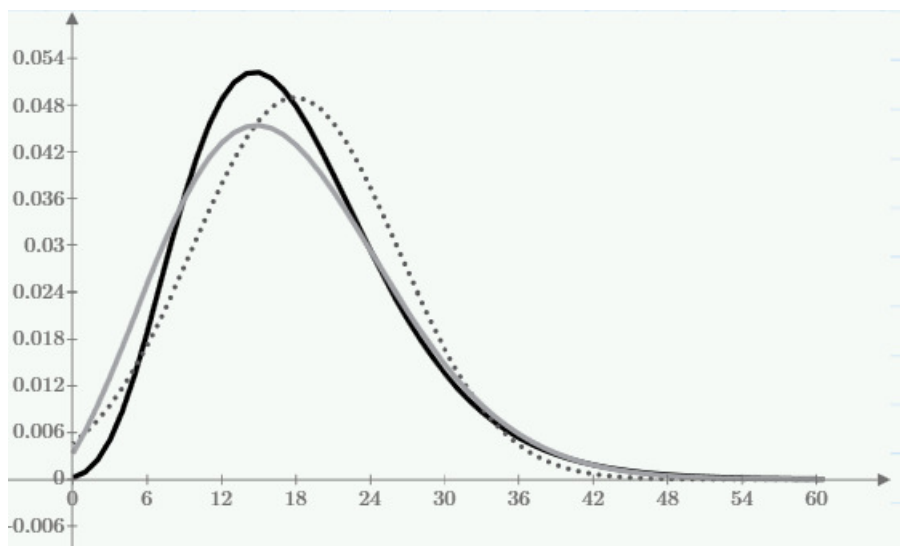


Рис. 3. Распределение вероятностей $P(i, T)$ (чёрная сплошная линия), стандартная гауссовская аппроксимация $P_1(i)$ (пунктирная линия) и предложенная модифицированная аппроксимация $P_2(i)$ (серая сплошная линия); $\alpha = 0,5$, $\beta = 0,8$, $\lambda = 5$, $T = 5$, $\gamma = 1$

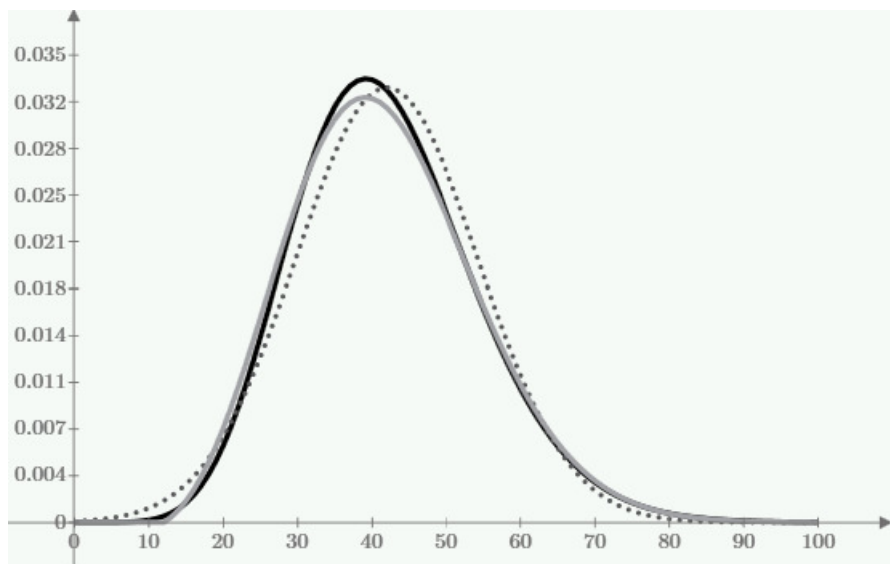


Рис. 4. Распределение вероятностей $P(i, T)$ (чёрная сплошная линия), стандартная гауссовская аппроксимация $P_1(i)$ (пунктирная линия) и предложенная модифицированная аппроксимация $P_2(i)$ (серая сплошная линия); $\alpha = 0,5$, $\beta = 0,8$, $\lambda = 5$, $T = 10$, $\gamma = 0,25$

Таблица 1

Расстояние Колмагорова для истинного распределения вероятностей числа занятых приборов в системе и его аппроксимации 2-го и 3-го порядков при значениях параметров $\alpha = 0,5$, $\beta = 0,8$, $\lambda = 5$, и указанных γ и T .

$\gamma \backslash T$		T						
		1	5	8	10	15	25	100
1	$P_1(i)$	0,169	0,061	0,050	0,047	0,042	0,038	0,031
	$P_2(i)$	0,100	0,052	0,043	0,040	0,035	0,031	0,025
0,5	$P_1(i)$	0,167	0,049	0,038	0,034	0,028	0,021	0,013
	$P_2(i)$	0,099	0,036	0,024	0,020	0,014	0,012	0,005
0,25	$P_1(i)$	0,167	0,046	0,034	0,031	0,024	0,020	0,010
	$P_2(i)$	0,097	0,030	0,019	0,015	0,011	0,006	0,002

Будем считать, что нас удовлетворяет оценка точности $\Delta < 0,005$. Тогда, как видно из результатов численного моделирования, областью применимости гауссовской дискретной аппроксимации является значение параметра времени $T \geq 8$, при котором расстояние Колмагорова Δ даёт значение, меньше 0,05 при любых значениях параметра γ . Однако при уменьшении интенсивности входящего потока λ , очевидно, стоит увеличивать нижнюю границу для параметра T , и, наоборот, при увеличении интенсивности λ диапазон значений для времени T можно расширять, т.е. уменьшать нижнюю границу параметра T . Так, численное моделирование уже при значении $\lambda = 22$ показывает приемлемые результаты при любом значении $T \geq 1$.

Также стоит отметить, что модифицированная аппроксимация $P_2(i)$ демонстрирует более высокую точность по сравнению со стандартной гауссовской аппроксимацией $P_1(i)$.

Заключение

В работе получено аналитическое выражение для распределения вероятностей числа заявок в системе массового обслуживания с неограниченным числом приборов и неординарным входящим потоком, длины интервалов между моментами наступления событий в котором имеют экспоненциальную функцию распределения, а число заявок в пачке определяется заданных дискретным распределением $r(k)$. Предложены 2 аппроксимации распределения вероятностей, с помощью компьютерных экспериментов

показана достаточно высокая точность предложенных аппроксимаций распределений вероятностей, также установлена область их применимости.

Целью дальнейших исследований является рассмотрение системы с неограниченным числом приборов и произвольным входящим потоком $GI|GI|_{\infty}$, а также нахождение предельного распределения вероятностей числа занятых приборов системы в предельном условии $T \rightarrow \infty$.

ЛИТЕРАТУРА

1. Lee W.C.Y. Mobile Cellular Telecommunications. – Analog and Digital Systems: 2nd ed., McGraw-Hill, 1995. – 664 p.
2. Гайдамака Ю.В., Зарипова Э.Р., Самуйлов К.Е. Модели обслуживания вызовов в сети сотовой подвижной связи. – М.: Изд-во РУДН, 2008. – 72 с.
3. Жидкова Л.А., Моисеева С.П. Математическая модель потоков покупателей двухпродуктовой торговой компании в виде системы массового обслуживания с повторными обращениями к блокам // Известия Томского политехнического университета. – 2013. – Т. 322. – № 6. – С. 5–9.
4. Королькова Л.И., Переверзев П.П. Оптимизация процессов предприятия на основе новой методики расчета характеристик многофазной системы массового обслуживания с непрерывной загрузкой без промежуточных накопителей // Современные проблемы науки и образования. – 2012. – № 3.
5. Balsamo S., De Nitto Personè V., Inverardi P. A review on queueing network models with finite capacity queues for software architecture performance prediction // Performance Evaluation. – 2003. – V. 51. – Iss. 2. – P. 269–288.
6. Рачинская М.А., Федоткин М.А. Построение и исследование вероятностной модели циклического управления потоками малой интенсивности // Вестник Нижегородского университета им. Н.И. Лобачевского. – 2014. – № 4 (1). С. 370–376.
7. Клейнрок Л. Вычислительные системы с очередями. – М.: Мир, 1979. – 600 с.
8. Афанасьева Л.Г., Булинская Е.В. Математические модели транспортных систем, основанные на теории очередей // Труды МФТИ. – 2010. – Т. 2. – № 4. – С. 6–10.
9. Грачев В.В., Моисеев А.Н., Назаров А.А., Ямольский В.З. Многофазная модель массового обслуживания системы распределенной обработки данных // Доклады ТУСУРа. – 2012. – № 2 (26). – Ч. 2. – С. 248–251.
10. Mandelbaum A., Pats G. State-dependent queues: approximations and applications // Stochastic Networks, IMA Volumes in Mathematics, F.P. Kelly and R.J. Williams, eds. – Springer, 1995. – P. 239–282.
11. Mandelbaum A., Zeltyn S. The impact of customers patience on delay and abandonment: some empirically-driven experiments with the M/M/n + G queue // Operations Research. – 2004. – V. 26. – P. 377–411.
12. Reed J.E. The G/GI/N queue in the Halfin-Whitt regime I: infinite-server queue system equations // The Stern School, NYU. – 2007.
13. Puhalskii A.A., Reed J.E. On many-server queues in heavy traffic // Annals of Applied Probability. – 2008. – V. 20. – P. 129–195.
14. Saaty T.L. Elements of queueing theory. – McGraw Hill book company, New York 1961. – 505 p.
15. D'Auria B. Stochastic decomposition of the M/G/∞ queue in a random environment // Oper. Res.Lett. – 2007. – 35. – P. 805–812.
16. Бочаров П. П., Печинкин А.В. Теория массового обслуживания. – М.: Изд-во РУДН, 1995. – 520 с.
17. Моисеев А.Н., Назаров А.А. Бесконечнолинейные системы и сети массового обслуживания. – Томск: Изд-во НТЛ, 2015. – 240 с.
18. Моисеева С.П. Разработка методов исследования математических моделей немарковских систем обслуживания с неограниченным числом приборов и негуассоновскими входящими потоками: дис. доктора физ.-мат. наук: Томск: НИ ТГУ, 2014. 260 с.
19. Baum D. The infinite server queue with Markov additive arrivals in space // Proceedings of the international conference “Probabilistic analysis of rare events”. Riga, Latvia, 1999. – P. 136–142.
20. Doorn E.A., Jagers A.A. Note on the GI/GI/∞ system with identical service and interarrival-time distributions // Journal of queueing systems. – 2004. – № 47. – P. 45–52.
21. Матвеев В.Ф., Ушаков В.Г. Системы массового обслуживания. – М.: Изд-во МГУ, 1984. – 242 с.
22. Климов Г.П. Теория массового обслуживания. – М.: Изд-во МГУ, 2011. – 312 с.

АСИМПТОТИЧЕСКИЙ АНАЛИЗ ДВУХЛИНЕЙНОЙ RQ-СИСТЕМЫ ВИДА ММРР|M|2 С НЕТЕРПЕЛИВЫМИ ЗАЯВКАМИ В УСЛОВИИ ДОЛГОЙ ТЕРПЕЛИВОСТИ

О.А. Выговская, С.П. Моисеева
 Томский государственный университет
 o.vygovskaya@bk.ru, smoiseeva@mail.ru

Введение

Системы массового обслуживания с повторными вызовами или RQ-системы (Retrial Queueing System) – математические модели, широко применяемые для анализа и оптимизации различных телекоммуникационных систем, сетей мобильной связи, call-центров и др. [4,5]. Характерной чертой таких систем является наличие повторных обращений заявок к обслуживающему прибору спустя некоторое случайное время после неудачной попытки обслуживания. Такие ситуации могут быть вызваны не только отсутствием свободных серверов в моменты поступления заявок, но некоторыми техническими причинами. Детальное описание RQ-систем и результатов их исследований представлены в монографиях Г.И. Фалина, Дж. Арталехо [2,3].

Существует достаточно большое число работ, посвященных RQ-системам с нетерпеливыми заявками [1,7]. Однако в большинстве из них нетерпеливость понимается в том смысле, что заявка с определённой вероятностью уходит из ИПВ после неудачной попытки обслуживания [3,6]. В настоящей работе предполагается, что терпеливость заявки описывается по экспоненциальному закону.

1. Описание модели

Рассмотрим RQ-систему, т.е. систему массового обслуживания с повторными вызовами с двумя обслуживающими приборами, на вход которой поступает ММРР-поток заявок с матрицей условных интенсивностей Λ и матрицей инфинитезимальных характеристик Q . Время обслуживания каждой заявки распределено по экспоненциальному закону с параметром μ . Если поступившая заявка застаёт хотя бы один из двух приборов свободным, то она занимает его для обслуживания. Если оба прибора заняты, то заявка переходит в источник повторных вызовов (ИПВ или на орбиту), где осуществляет случайную задержку, продолжительность которой имеет экспоненциальное распределение с параметром σ . Из ИПВ после случайной задержки заявка вновь обращается к приборам с повторной попыткой обслуживания. Заявки в ИПВ являются нетерпеливыми, т.е. после случайного времени, распределённого по экспоненциальному закону с параметром α , заявка покидает систему.

$$\Lambda = \text{diag}[\lambda_1, \dots, \lambda_n], \quad Q = \|q_{ij}\|,$$

$$q_{ij} = \lim_{\Delta t \rightarrow 0} \frac{P_{ij}(\Delta t)}{\Delta t}, \quad i, j = \overline{1, n}, \quad i \neq j, \quad q_{ii} = \lim_{\Delta t \rightarrow 0} \frac{P_{ii}(\Delta t) - 1}{\Delta t}, \quad i = \overline{1, n}.$$

Схематичное представление модели показано на рис. 1.

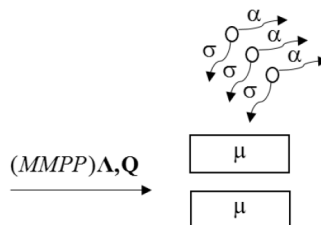


Рис. 1. RQ-система ММРР|M|2 с нетерпеливыми заявками

Пусть $i(t)$ – случайный процесс, характеризующий число заявок в ИПВ, $s(t)$ – цепь Маркова, управляющая ММРР-поток, а $k(t)$ определяет состояние прибора следующим образом:

$$k(t) = \begin{cases} 0, & \text{если прибор свободен,} \\ 1, & \text{если один прибор занят,} \\ 2, & \text{если оба прибора заняты.} \end{cases}$$

Обозначим $P\{k(t) = k, i(t) = i, s(t) = s\} = P(k, i, s, t)$ – вероятность того, что прибор в момент времени t находится в состоянии k , управляющая ММРР-поток цепь Маркова – в состоянии s и в источнике повторных вызовов находится i заявок. Тогда процесс $\{k(t), i(t), s(t)\}$ изменения состояний данной системы во времени является марковским. Ставится задача нахождения распределения вероятностей числа заявок в источнике повторных вызовов такой системы.

2. Исследование RQ-системы вида ММРР|M|2 в условии долгой терпеливости

Для получения распределения вероятностей $P(k, i, s, t)$ состояний рассматриваемой RQ-системы составим систему уравнений Колмогорова.

$$\left\{ \begin{array}{l} \frac{\partial P_0(s, i, t)}{\partial t} = -(\lambda_s + i\sigma + i\alpha - q_{ss})P_0(s, i, t) + \mu P_1(s, i, t) + \\ \quad + \alpha(i+1)P_0(s, i+1, t) + \sum_{v \neq s} P_0(v, i, t)q_{vs}, \quad s = 1, S, \\ \frac{\partial P_1(s, i, t)}{\partial t} = -(\lambda_s + \mu + i\alpha + i\sigma - q_{ss})P_1(s, i, t) + \lambda_s P_0(s, i, t) + P_2(s, i, t)2\mu + \\ \quad + (i+1)\sigma \cdot P_0(s, i+1, t) + \alpha(i+1)P_1(s, i+1, t) + \sum_{v \neq s} P_1(v, i, t)q_{vs}, \quad s = 1, S, \\ \frac{\partial P_2(s, i, t)}{\partial t} = -(\lambda_s + i\alpha + 2\mu - q_{ss})P_2(s, i, t) + \lambda_s P_1(s, i, t) + \sigma(i+1)P_1(s, i+1, t) + \\ \quad + \alpha(i+1)P_2(s, i+1, t) + \lambda_s P_2(s, i-1, t) + \sum_{v \neq s} P_2(v, i, t)q_{vs}, \quad s = 1, S. \end{array} \right. \quad (1)$$

Обозначим \mathbf{R} – стационарное распределение вероятностей значения цепи Маркова, управляющей входящим ММРР-поток, которое определяется из следующей системы:

$$\begin{cases} \mathbf{RQ} = 0, \\ \mathbf{RE} = 1, \end{cases}$$

где \mathbf{E} – единичный вектор-столбец, $\mathbf{0}$ – вектор-столбец с нулевыми элементами.

Введём вектор-строки $\mathbf{\Pi}(k, i) = \{P(k, 1, t) P(k, 2, t) \dots P(k, S, t)\}$, где в стационарном режиме $\lim_{\Delta t \rightarrow 0} P(k, i, s, t) = \mathbf{\Pi}(k, i, s)$. Тогда в стационарном режиме система (1) в матричном виде примет вид:

$$\left\{ \begin{array}{l} 0 = \mathbf{\Pi}_0(i)(-\Lambda - i(\sigma + \alpha)\mathbf{I} + \mathbf{Q}) + \mu\mathbf{\Pi}_1(i) + \alpha(i+1)\mathbf{\Pi}_0(i+1), \\ 0 = \mathbf{\Pi}_1(i)(-\Lambda - i(\sigma + \alpha)\mathbf{I} + \mathbf{Q} - \mu\mathbf{I}) + \\ \quad + \mathbf{\Pi}_0(i)\Lambda + 2\mu\mathbf{\Pi}_2(i) + \sigma(i+1)\mathbf{\Pi}_0(i+1) + \alpha(i+1)\mathbf{\Pi}_1(i+1), \\ 0 = \mathbf{\Pi}_2(i)(-\Lambda - i\alpha\mathbf{I} + \mathbf{Q} - 2\mu\mathbf{I}) + \\ \quad + \mathbf{\Pi}_1(i)\Lambda + \sigma(i+1)\mathbf{\Pi}_1(i+1) + \alpha(i+1)\mathbf{\Pi}_2(i+1) + \mathbf{\Pi}_2(i-1)\Lambda, \end{array} \right. \quad (2)$$

где \mathbf{I} – единичная матрица.

Перейдём в системе (2) к частичным характеристическим функциям $\mathbf{H}_k(s, u) = \sum_i^{\infty} e^{ju_i} \mathbf{\Pi}_k(s, i)$, где в векторном обозначении $\mathbf{H}_k(u) = [\mathbf{H}_k(1, u), \dots, \mathbf{H}_k(s, u)]$.

Учитывая, что $\frac{\partial \mathbf{H}_k(s, u)}{\partial u} = j \sum_i^{\infty} i e^{ju_i} \mathbf{\Pi}_k(s, i)$, $j = \sqrt{-1}$ – мнимая единица, система уравнений (2) переписывается в виде:

$$\left\{ \begin{array}{l} 0 = j(\sigma + \alpha(1 - e^{-ju})) \frac{\partial \mathbf{H}_0(u)}{\partial u} + \mu \mathbf{H}_1(u) + \mathbf{H}_0(u)(\mathbf{Q} - \Lambda), \\ 0 = j(\sigma + \alpha(1 - e^{-ju})) \frac{\partial \mathbf{H}_1(u)}{\partial u} + \mathbf{H}_1(u)(\mathbf{Q} - \Lambda) - \mu \mathbf{H}_1(u) + \mathbf{H}_0(u) \Lambda - \\ - j\sigma e^{-ju} \frac{\partial \mathbf{H}_0(u)}{\partial u} + 2\mu \mathbf{H}_2(u), \\ 0 = j\alpha(1 - e^{-ju}) \frac{\partial \mathbf{H}_2(u)}{\partial u} + \mathbf{H}_2(u)(-2\mu \mathbf{I} + \mathbf{Q} - \Lambda(1 - e^{ju})) + \mathbf{H}_1(u) \Lambda - j\sigma e^{-ju} \frac{\partial \mathbf{H}_1(u)}{\partial u}. \end{array} \right. \quad (3)$$

Преобразуем систему (3) путём прибавления к третьему уравнению первых двух, получим:

$$\left\{ \begin{array}{l} 0 = j(\sigma + \alpha(1 - e^{-ju})) \frac{\partial \mathbf{H}_0(u)}{\partial u} + \mu \mathbf{H}_1(u) + \mathbf{H}_0(u)(\mathbf{Q} - \Lambda), \\ 0 = j(\sigma + \alpha(1 - e^{-ju})) \frac{\partial \mathbf{H}_1(u)}{\partial u} + \mathbf{H}_1(u)(\mathbf{Q} - \Lambda) - \mu \mathbf{H}_1(u) + \mathbf{H}_0(u) \Lambda - \\ - j\sigma e^{-ju} \frac{\partial \mathbf{H}_0(u)}{\partial u} + 2\mu \mathbf{H}_2(u), \\ 0 = j\alpha(1 - e^{-ju}) \sum_{k=0}^2 \frac{\partial \mathbf{H}_k(u)}{\partial u} + j\sigma \sum_{k=0}^1 \frac{\partial \mathbf{H}_k(u)}{\partial u} - j\sigma e^{-ju} \sum_{k=0}^1 \frac{\partial \mathbf{H}_k(u)}{\partial u} + \sum_{k=0}^2 \mathbf{H}_k(u) \mathbf{Q} - \mathbf{H}_2(u) \Lambda(1 - e^{ju}). \end{array} \right.$$

3. Метод асимптотического анализа первого порядка

Введём обозначения $\alpha = \varepsilon$, $u = \varepsilon w$, $\mathbf{H}_0(u) = \varepsilon^2 \mathbf{F}_0(w, \varepsilon)$, $\mathbf{H}_1(u) = \varepsilon \mathbf{F}_1(w, \varepsilon)$, $\mathbf{H}_2(u) = \mathbf{F}_2(w, \varepsilon)$, где $\varepsilon \downarrow 0$ – бесконечно малая величина. Учитывая, что

$$\mathbf{H}'_0(u) = \varepsilon \frac{\partial}{\partial w} \mathbf{F}_0(w, \varepsilon), \quad \mathbf{H}'_1(u) = \frac{\partial}{\partial w} \mathbf{F}_1(w, \varepsilon), \quad \mathbf{H}'_2(u) = \frac{1}{\varepsilon} \frac{\partial}{\partial w} \mathbf{F}_0(w, \varepsilon),$$

$$\left\{ \begin{array}{l} 0 = j(\sigma + \varepsilon(1 - e^{-j\varepsilon w})) \frac{\partial}{\partial w} \mathbf{F}_0(w, \varepsilon) + \mu \mathbf{F}_1(w, \varepsilon) + \varepsilon \mathbf{F}_0(w, \varepsilon)(\mathbf{Q} - \Lambda), \\ 0 = j(\sigma + \varepsilon(1 - e^{-j\varepsilon w})) \frac{\partial}{\partial w} \mathbf{F}_1(w, \varepsilon) + \varepsilon \mathbf{F}_1(w, \varepsilon)(\mathbf{Q} - \Lambda) - \mu \varepsilon \mathbf{F}_1(w, \varepsilon) + \varepsilon^2 \mathbf{F}_0(w, \varepsilon) \Lambda - \\ - j\sigma e^{-j\varepsilon w} \varepsilon \frac{\partial}{\partial w} \mathbf{F}_0(w, \varepsilon) + 2\mu \mathbf{F}_2(w, \varepsilon), \\ 0 = j\varepsilon(1 - e^{-j\varepsilon w}) \sum_{k=0}^2 \varepsilon^{2-k-1} \frac{\partial}{\partial w} \mathbf{F}_k(w, \varepsilon) + j\sigma(1 - e^{-j\varepsilon w}) \sum_{k=0}^1 \varepsilon^{2-k-1} \frac{\partial}{\partial w} \mathbf{F}_k(w, \varepsilon) + \\ + \sum_{k=0}^2 \varepsilon^{2-k} \mathbf{F}_k(w, \varepsilon) \mathbf{Q} - \mathbf{F}_2(w, \varepsilon) \Lambda(1 - e^{j\varepsilon w}). \end{array} \right. \quad (4)$$

Совершив предельный переход при $\varepsilon \downarrow 0$ в первых двух уравнениях системы (4) и принимая во внимание, что в третьем уравнении $e^{-j\varepsilon w} = 1 - j\varepsilon w$, получим:

$$\left\{ \begin{array}{l} j\sigma \frac{\partial}{\partial w} \mathbf{F}_0(w, \varepsilon) = -\mu \mathbf{F}_1(w, \varepsilon), \\ j\sigma \frac{\partial}{\partial w} \mathbf{F}_1(w, \varepsilon) = -2\mu \mathbf{F}_2(w, \varepsilon), \\ 0 = j^2 \varepsilon w \sum_{k=0}^2 \varepsilon^{2-k} \frac{\partial}{\partial w} \mathbf{F}_k(w, \varepsilon) + j^2 \sigma \varepsilon w \sum_{k=0}^1 \varepsilon^{2-k-1} \frac{\partial}{\partial w} \mathbf{F}_k(w, \varepsilon) + \\ + \sum_{k=0}^2 \varepsilon^{2-k} \mathbf{F}_k(w, \varepsilon) \mathbf{Q} + \mathbf{F}_2(w, \varepsilon) \Lambda j \varepsilon w. \end{array} \right. \quad (5)$$

Преобразуем последнее уравнение системы (5) путем домножения на \mathbf{e} – единичный вектор столбец. Учитывая, что $\mathbf{Qe} = 0$, справедливо следующее равенство:

$$0 = j \sum_{k=0}^2 \varepsilon^{2-k} \frac{\partial}{\partial w} \mathbf{F}_k(w, \varepsilon) \mathbf{e} + j\sigma \sum_{k=0}^1 \varepsilon^{2-k-1} \frac{\partial}{\partial w} \mathbf{F}_k(w, \varepsilon) \mathbf{e} + \mathbf{F}_2(w, \varepsilon) \Lambda \mathbf{e}$$

или

$$0 = j \frac{\partial}{\partial w} \mathbf{F}_2(w, \varepsilon) \mathbf{e} + j\sigma \frac{\partial}{\partial w} \mathbf{F}_1(w, \varepsilon) \mathbf{e} + \mathbf{F}_2(w, \varepsilon) \Lambda \mathbf{e}.$$

Принимая во внимание, что $\mathbf{F}_2(w, \varepsilon) = R\Phi(w)$, $R\mathbf{e} = 1$, справедливо равенство:

$$j\Phi'(w) = \Phi(w)(2\mu - \Lambda), \quad \Phi(w) = e^{-j(2\mu - \Lambda)w},$$

откуда $\mathbf{F}_2(w, \varepsilon) = R e^{-j(2\mu - \Lambda)w}$.

Так как допредельная характеристическая функция приближенно равна

$$h(u) = \sum_{k=0}^2 \mathbf{H}_k(u) = \mathbf{F}_2\left(\frac{u}{\varepsilon}\right) + O(\varepsilon) \approx \mathbf{F}_1\left(\frac{u}{\varepsilon}\right),$$

получили асимптотическое приближение характеристической функции первого порядка

$$h^{(1)}(u) = R e^{\frac{j u (\Lambda - 2\mu)}{\alpha}}, \quad R = h^{(1)}(0).$$

Таким образом, был получен вид асимптотической характеристической функции первого порядка с математическим ожиданием $\frac{(\Lambda - 2\mu)}{\alpha}$.

Заключение

В настоящей работе была исследована математическая модель RQ-системы M|M|2 с нетерпеливыми заявками и был найден вид асимптотики первого порядка характеристической функции числа заявок в источнике повторных вызовов в условии долгой терпеливости.

ЛИТЕРАТУРА

1. Назаров А.А., Терпугов А.Ф. Теория массового обслуживания. – Томск: Изд-во НТЛ, 2005. – 228 с.
2. Artalejo J.R., Gomez-Corral A. Retrial Queueing Systems. A Computational Approach. – Springer, 2008.
3. Falin G.I., Templeton J.G.C. Retrial queues. – Chapman & Hall, London, 1997.
4. Roszik J., Sztrik J., Kim C. Retrial queues in the performance modelling of cellular mobile networks using MOSEL // International Journal of Simulation. – 2005. – №6. P. 38–47.
5. Kuznetsov D., Nazarov A. Analysis of non-Markovian models of communication networks with adaptive protocols of multiple random access // Automation and Remote Control. – 2001. – №5. – P. 124–146.

СТАЦИОНАРНОЕ АСИМПТОТИЧЕСКОЕ РАСПРЕДЕЛЕНИЕ ВЕРОЯТНОСТЕЙ ЧИСЛА ЗАЯВОК И ОБЪЁМОВ ЗАНЯТЫХ РЕСУРСОВ В НЕОДНОРОДНОЙ РЕСУРСНОЙ БЕСКОНЕЧНОЛИНЕЙНОЙ СМО С ВХОДЯЩИМ ММРР-ПОТОКОМ

А.А. Галилейская, Е.Ю. Лисовская

Томский государственный университет
lusta.nastya@mail.ru, ekaterina_lisovs@mail.ru

Введение

Системы и сети массового обслуживания уже давно применяются в качестве математических моделей различных информационно-компьютерных систем и сетей [1,2]. Чаще аналитические результаты для исследования таких систем массового обслуживания (СМО) удаётся получить лишь в предположении, что входящие потоки пуассоновские. Однако марковские или рекуррентные потоки событий наиболее адекватно описывают современные потоки данных компьютерных и телекоммуникационных сетей.

Кроме того, реальные потоки содержат в себе разнотипные данные, которые передают текстовую или голосовую информацию, в связи с чем возникает необходимость учитывать объём передаваемой информации во избежание потери из-за нехватки предоставляемого ресурса [12–14], а также разнотипность заявок и разнородность обслуживания [3–6].

В данной работе рассматривается ресурсная система с неограниченным числом неоднородных приборов, которая является одной из модификаций систем массового обслуживания (СМО). Требования приходят независимо друг от друга, по одному, они имеют различную природу, требуют разное количество ресурса и по-разному обслуживаются.

Так как для систем массового обслуживания с двумя и более числом приборов, непуассоновскими входящими потоками и неэкспоненциальным временем обслуживания получение аналитических результатов является затруднительным, для исследования применяется метод асимптотического анализа [7,8].

1. Постановка задачи

Рассмотрим систему массового обслуживания $\text{MMPP}^{(v)}|GI^{(n)}|_{\infty}$ с n типами неоднородных (в смысле скорости обслуживания) [9] обслуживающих приборов, на вход которой поступает ММРР-поток разнотипных заявок, управляемый цепью Маркова $k(t) = 1, 2, \dots, K$, задаётся матрицей инфинитезимальных характеристик $\mathbf{Q} = \|q_{vk}\|$, $v, k = \overline{1, K}$ и диагональной матрицей условных интенсивностей $\mathbf{\Lambda} = \text{diag}\{\lambda_1, \dots, \lambda_K\}$ (рис. 1).

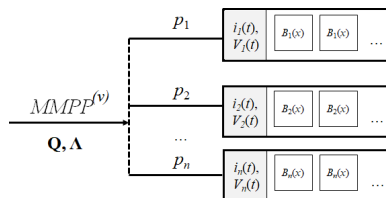


Рис. 1. Бесконечнолинейная ресурсная СМО с n типами неоднородных приборов и с входящим ММРР-потоком

Дисциплина обслуживания определяется следующим образом. Заявка, поступающая в систему, с вероятностью p_i ($i = \overline{1, n}$) требует для обслуживания случайное количество некоторого ресурса соответствующего типа с функцией распределения $G_i(v)$ и обслуживается в течение случайного времени, имеющего функцию распределения $B_i(x)$. Вероятности p_i удовлетворяют условию нормировки $\sum_{i=1}^n p_i = 1$.

Пусть $l_i(t)$ – число заявок i -го типа в системе в момент времени t , $V_i(t)$ – суммарный объём занятого ресурса i -го типа в системе в момент времени t , $i = \overline{1, n}$. Поставим задачу нахождения характеристик многомерного случайного процесса $\{\mathbf{I}(t), \mathbf{V}(t)\} = \{l_1(t), \dots, l_n(t), V_1(t), \dots, V_n(t)\}$. Отметим, что исследуемый процесс не является марковским. Для его исследования применим метод многомерного динамического просеивания [10].

Изобразим n параллельных осей времени, пронумерованных от 0 до n (рис. 2). Ось под номером 0 будет отображать события входящего потока, ось под номером 1 будет соответствовать первому просеянному потоку, ось под номером 2 – второму и т.д., ось под номером n соответствует n -му просеянному потоку.

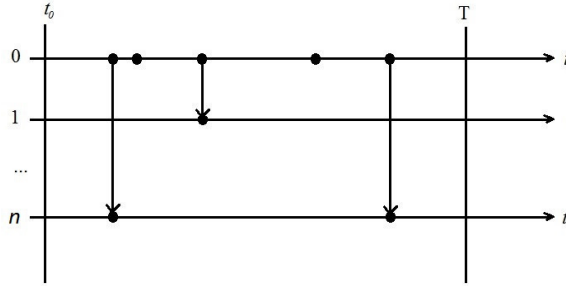


Рис. 2. Просеивание заявок входящего потока

Пусть имеется набор функций $S_i(t)$, $i = \overline{1, n}$, значения которых лежат в диапазоне $[0, 1]$ и обладают свойством $\sum_{i=1}^n S_i(t) \leq 1$ для любых t . Событие входящего потока может просеяться только на одну из осей, либо не просеется ни на одну. Вероятность того, что заявка входящего потока, поступившая в систему в момент времени $t > t_0$, сформирует событие потока на i -й оси, т.е. к моменту времени T будет находиться на обслуживании, равна $S_i(t) = 1 - B_i(T - t)$. Вероятность того, что заявка не сформирует событие ни на одной из осей, равна $S_0(t) = 1 - \sum_{i=1}^n p_i S_i(t)$, т.е. к моменту времени T заявка закончит обслуживание и покинет систему.

Обозначим $n_i(t)$ – число событий, наступивших на i -й оси просеянного потока до момента t , $W_i(t)$ – суммарный объём занятого ресурса просеянными заявками на i -й оси. Как показано в [10], многомерное распределение вероятностей числа заявок на фазах системы в момент времени T совпадает с многомерным распределением вероятностей числа просеянных заявок на соответствующие оси: $P\{\mathbf{I}(T) = \mathbf{m}\} = P\{\mathbf{n}(T) = \mathbf{m}\}$ для любых $\mathbf{m} = [m_1, \dots, m_n]$. Нетрудно показать, что для исследуемого процесса $\{\mathbf{i}(t), \mathbf{V}(t)\}$ справедливо

$$P\{\mathbf{I}(T) = \mathbf{m}, \mathbf{V}(T) < \mathbf{z}\} = P\{\mathbf{n}(T) = \mathbf{m}, \mathbf{W}(T) < \mathbf{z}\} \quad (1)$$

для любых $\mathbf{m} = [m_1, \dots, m_n]$ и любых $\mathbf{z} = [z_1, \dots, z_n]$. Следует отметить, что неравенства $\mathbf{V}(T) < \mathbf{z}$, $\mathbf{W}(T) < \mathbf{z}$ подразумевают поэлементное сравнение векторов, т.е. $V_1(T) < z_1$, $W_1(T) < z_1$ и т.д. Будем использовать равенство (1) для исследования процесса $\{\mathbf{I}(t), \mathbf{V}(t)\}$ с помощью исследования процесса $\{\mathbf{n}(t), \mathbf{W}(t)\}$.

2. Система дифференциальных уравнений Колмогорова

Добавим компоненту $k(t)$ – состояние управляющей цепи Маркова в момент времени t – к процессу $\{\mathbf{n}(t), \mathbf{W}(t)\}$, тогда полученный многомерный процесс будет являться марковским. Введём обозначение для его распределения вероятностей

$$P(k, \mathbf{n}, \mathbf{w}, t) = P\{k(t) = k, \mathbf{n}(t) = \mathbf{n}, \mathbf{W}(t) < \mathbf{w}\}.$$

Для этого распределения составим Δt -методом прямую систему дифференциальных уравнений Колмогорова. По формуле полной вероятности запишем:

$$\begin{aligned} P(k, \mathbf{n}, \mathbf{w}, t + \Delta t) = & P(k, \mathbf{n}, \mathbf{w}, t)(1 - \lambda_k \Delta t)(1 + q_{kk} \Delta t) + P(k, \mathbf{n}, \mathbf{w}, t) \lambda_k \Delta t S_0(t) + \\ & + \lambda_k \Delta t \sum_{i=1}^n p_i S_i(t) \int_0^{w_i} P(k, \mathbf{n} - \mathbf{e}_i, \mathbf{w} - \mathbf{y}_i, t) dG_i(y) + \sum_{v \neq k} q_{vk} \Delta t P(v, \mathbf{n}, \mathbf{w}, t) + o(\Delta t), \end{aligned} \quad (2)$$

где \mathbf{y}_i – вектор, все элементы которого равны 0 за исключением i -го, который равен y , \mathbf{e}_i – вектор, все элементы которого равны 0 за исключением i -го, который равен 1.

Из (2) получаем систему дифференциальных уравнений Колмогорова

$$\begin{aligned} \frac{\partial P(k, \mathbf{n}, \mathbf{w}, t)}{\partial t} = & -\lambda_k \sum_{i=1}^n p_i S_i(t) P(k, \mathbf{n}, \mathbf{w}, t) + \sum_{v \neq k} q_{vk} \Delta t P(v, \mathbf{n}, \mathbf{w}, t) + \\ & + \lambda_k \sum_{i=1}^n p_i S_i(t) \int_0^{w_i} P(k, \mathbf{n} - \mathbf{e}_m, \mathbf{w} - \mathbf{y}_m, t) dG_i(y), \quad k = \overline{1, K}, \quad w_i > 0, \quad n_i = 0, 1, \dots, \quad i = \overline{1, n}. \end{aligned}$$

Начальное условие для решения $P(k, \mathbf{n}, \mathbf{w}, t)$ в момент времени t_0 определим в виде

$$P(k, \mathbf{n}, \mathbf{w}, t_0) = \begin{cases} r(k), & \mathbf{n} = \mathbf{w} = \mathbf{0}, \\ 0, & \text{иначе,} \end{cases}$$

где $r(k)$ – стационарное распределение вероятностей состояний цепи Маркова $k(t)$.

Введём характеристические функции вида

$$h(k, \mathbf{u}, \mathbf{v}, t) = \sum_{n_1=0}^{\infty} e^{ju_1 n_1} \int_0^{\infty} e^{jv_1 w_1} \dots \sum_{n_n=0}^{\infty} e^{ju_n n_n} \int_0^{\infty} e^{jv_n w_n} P(k, \mathbf{n}, d\mathbf{w}, t),$$

где $j = \sqrt{-1}$ – мнимая единица. Тогда можем записать следующие дифференциальные уравнения:

$$\begin{aligned} \frac{\partial h(k, \mathbf{u}, \mathbf{v}, t)}{\partial t} = & \sum_v q_{vk} h(k, \mathbf{u}, \mathbf{v}, t) + \lambda_k \sum_{i=1}^n p_i S_i(t) h(k, \mathbf{u}, \mathbf{v}, t) \left(e^{ju_i G_i^*(v_i)} - 1 \right), \quad k = \overline{1, K}, \\ & G_i^*(v) = \int_0^{\infty} e^{jvy} dG_i(y). \end{aligned}$$

Перепишем эту систему в виде матричного уравнения

$$\frac{\partial \mathbf{H}(\mathbf{u}, \mathbf{v}, t)}{\partial t} = \mathbf{H}(\mathbf{u}, \mathbf{v}, t) \left[\Lambda \sum_{i=1}^n p_i S_i(t) \left(e^{ju_i G_i^*(v_i)} - 1 \right) + \mathbf{Q} \right] \quad (3)$$

с начальным условием

$$\mathbf{H}(\mathbf{u}, \mathbf{v}, t_0) = \mathbf{r}, \quad (4)$$

где $\mathbf{H}(\mathbf{u}, \mathbf{v}, t) = [h(1, \mathbf{u}, \mathbf{v}, t), \dots, h(K, \mathbf{u}, \mathbf{v}, t)]$, $\mathbf{r} = [r(1), \dots, r(K)]$ – вектор стационарного распределения вероятностей состояний управляющей цепи Маркова $k(t)$, удовлетворяющей системе

$$\begin{cases} \mathbf{r}\mathbf{Q} = \mathbf{0}, \\ \mathbf{r}\mathbf{e} = 1, \end{cases}$$

и \mathbf{e} – единичный вектор-столбец.

3. Метод асимптотического анализа

Так как прямое решение уравнения (3) не представляется возможным, для решения задачи (3)–(4) воспользуемся методом асимптотического анализа [10] в условии неограниченно растущей интенсивности входящего потока [11].

Применим метод асимптотического анализа в условии высокой интенсивности входящего потока и предельно частых изменений состояний цепи Маркова [12–14].

Подставим в уравнение (3) $\Lambda = N\bar{\Lambda}$ и $\mathbf{Q} = N\bar{\mathbf{Q}}$, где $N \rightarrow \infty$ – некоторый параметр, который используется для асимптотического анализа. Тогда можно записать:

$$\frac{1}{N} \frac{\partial \mathbf{H}(\mathbf{u}, \mathbf{v}, t)}{\partial t} = \mathbf{H}(\mathbf{u}, \mathbf{v}, t) \left[\bar{\Lambda} \sum_{i=1}^n p_i S_i(t) \left(e^{ju_i} G_i^*(v_i) - 1 \right) + \bar{\mathbf{Q}} \right], \quad (5)$$

с начальным условием (4).

Теорема 1. Асимптотическая характеристическая функция первого порядка многомерного случайного процесса $\{k(t), \mathbf{n}(t), \mathbf{W}(t)\}$ имеет вид:

$$\mathbf{H}(\mathbf{u}, \mathbf{v}, t) = \mathbf{r} \exp \left\{ N\lambda \sum_{i=1}^n \left(ju_i + jv_i a_i^{(i)} \right) p_i \int_{t_0}^t S_i(\tau) d\tau \right\},$$

где $\lambda = \bar{\mathbf{r}}\bar{\Lambda}\mathbf{e}$ – средняя интенсивность входящего потока, $a_i^{(i)}$ – математическое ожидание занимаемого ресурса i -го типа.

Доказательство.

Обозначим

$$\frac{1}{N} = \varepsilon, \quad \mathbf{u} = \varepsilon \mathbf{x}, \quad \mathbf{v} = \varepsilon \mathbf{y}, \quad \mathbf{H}(\mathbf{u}, \mathbf{v}, t) = \mathbf{F}_1(\mathbf{x}, \mathbf{y}, t, \varepsilon). \quad (6)$$

Перепишем задачу (4)–(5) с учётом введенных обозначений в виде

$$\varepsilon \frac{\partial \mathbf{F}_1(\mathbf{x}, \mathbf{y}, t, \varepsilon)}{\partial t} = \mathbf{F}_1(\mathbf{x}, \mathbf{y}, t, \varepsilon) \left[\bar{\Lambda} \sum_{i=1}^n p_i S_i(t) \left(e^{j\varepsilon x_i} G_i^*(\varepsilon y_i) - 1 \right) + \bar{\mathbf{Q}} \right] \quad (7)$$

с начальным условием

$$\mathbf{F}_1(\mathbf{x}, \mathbf{y}, t_0, \varepsilon) = \mathbf{r}. \quad (8)$$

Найдём асимптотическое решение задачи (7)–(8) в два этапа.

Этап 1. Подставляя в (7) $\varepsilon = 0$, получим $\mathbf{F}_1(\mathbf{x}, \mathbf{y}, t) \bar{\mathbf{Q}} = \mathbf{0}$. Сравнивая это уравнение с первым в системе для \mathbf{r} , можно сделать вывод, что его решение можно записать в виде:

$$\mathbf{F}_1(\mathbf{x}, \mathbf{y}, t) = \mathbf{r} \Phi_1(\mathbf{x}, \mathbf{y}, t), \quad (9)$$

где $\Phi_1(\mathbf{x}, \mathbf{y}, t)$ – некоторая скалярная функция, которая удовлетворяет условию:

$$\Phi_1(\mathbf{x}, \mathbf{y}, t_0) = 1. \quad (10)$$

Этап 2. Умножим (7) на вектор \mathbf{e} , подставим (9), учитывая разложение

$$e^{j\epsilon x} = 1 + j\epsilon x + O(\epsilon^2), \quad (11)$$

разделим результаты на ϵ и произведём асимптотический переход $\epsilon \rightarrow 0$. Тогда, учитывая, что $\bar{\mathbf{Q}}\mathbf{e} = 0$ и $\mathbf{r}\mathbf{e} = 1$, для функции $\Phi_1(\mathbf{x}, \mathbf{y}, t)$ получим следующее дифференциальное уравнение

$$\frac{\partial \Phi_1(\mathbf{x}, \mathbf{y}, t)}{\partial t} = \Phi_1(\mathbf{x}, \mathbf{y}, t) \lambda \sum_{i=1}^n p_i S_i(t) (jx_i + jy_i a_1^{(i)}). \quad (12)$$

Проинтегрировав уравнение (12) от t_0 до t , учитывая начальное условие (10), получим

$$\Phi_1(\mathbf{x}, \mathbf{y}, t) = \exp \left\{ \lambda \sum_{i=1}^n (jx_i + jy_i a_1^{(i)}) p_i \int_{t_0}^t S_i(\tau) d\tau \right\}.$$

Подставляя это выражение в (9) и выполняя замены, обратные к (6), получим

$$\begin{aligned} \mathbf{H}(\mathbf{u}, \mathbf{v}, t) &= \mathbf{F}_1(\mathbf{x}, \mathbf{y}, t, \epsilon) \approx \mathbf{F}_1(\mathbf{x}, \mathbf{y}, t) = \\ &= \mathbf{r} \exp \left\{ \lambda \sum_{i=1}^n (jx_i + jy_i a_1^{(i)}) p_i \int_{t_0}^t S_i(\tau) d\tau \right\} = \mathbf{r} \exp \left\{ N\lambda \sum_{i=1}^n (ju_i + jv_i a_1^{(i)}) p_i \int_{t_0}^t S_i(\tau) d\tau \right\}. \end{aligned}$$

Теорема 1 доказана.

Теорема 2. Асимптотическая характеристическая функция второго порядка многомерного случайного процесса $\{k(t), \mathbf{n}(t), \mathbf{W}(t)\}$ имеет вид:

$$\begin{aligned} \mathbf{H}(\mathbf{u}, \mathbf{v}, t) &\approx \mathbf{r} \exp \left\{ N\lambda \sum_{i=1}^n (ju_i + jv_i a_1^{(i)}) p_i \int_{t_0}^t S_i(\tau) d\tau + \right. \\ &+ N \sum_{i=1}^n \frac{(ju_i)^2}{2} \left(\lambda p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \\ &+ N \sum_{i=1}^n \frac{(jv_i)^2}{2} \left(\lambda a_1^{(i)} p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa (a_1^{(i)})^2 p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \\ &+ N \sum_{i=1}^n ju_i jv_i \left(\lambda a_1^{(i)} p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa a_1^{(i)} p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \\ &+ N \sum_{i=1}^n \sum_{l=1, l \neq i}^n ju_i ju_l \kappa p_i p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau + N \sum_{i=1}^n \sum_{l=1, l \neq i}^n jv_i jv_l \kappa a_1^{(i)} p_i a_1^{(l)} p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau + \\ &\left. + N \sum_{i=1}^n \sum_{l=1, l \neq i}^n ju_i jv_l \kappa a_1^{(l)} p_i p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau \right\}, \end{aligned}$$

где $\kappa = 2\mathbf{g}(\bar{\Lambda} - \lambda\mathbf{I})\mathbf{e}$, $a_1^{(i)}$ и $a_2^{(i)}$ – первый и второй начальные моменты случайных величин с функцией распределения вероятностей $G_i(y)$.

Доказательство.

Перейдём к построению гауссовской аппроксимации суммарных объёмов занятых ресурсов каждого типа. Представим функцию $\mathbf{H}(\mathbf{u}, \mathbf{v}, t)$ в виде

$$\mathbf{H}(\mathbf{u}, \mathbf{v}, t) = \mathbf{H}_2(\mathbf{u}, \mathbf{v}, t) \exp \left\{ N\lambda \sum_{i=1}^n p_i (ju_i + jv_i a_1^{(i)}) \int_{t_0}^t S_i(\tau) d\tau \right\}, \quad (13)$$

получим уравнение относительно функции $\mathbf{H}_2(\mathbf{u}, \mathbf{v}, t)$:

$$\begin{aligned} \frac{1}{N} \frac{\partial \mathbf{H}_2(\mathbf{u}, \mathbf{v}, t)}{\partial t} + \mathbf{H}_2(\mathbf{u}, \mathbf{v}, t) \lambda \sum_{i=1}^n p_i S_i(t) (ju_i + jv_i a_1^{(i)}) = \\ = \mathbf{H}_2(\mathbf{u}, \mathbf{v}, t) \left[\bar{\Lambda} \sum_{i=1}^n p_i S_i(t) (e^{ju_i} G_i^*(v_i) - 1) + \bar{\mathbf{Q}} \right]. \end{aligned} \quad (14)$$

Выполним здесь следующие замены:

$$\varepsilon^2 = \frac{1}{N}, \quad \mathbf{u} = \varepsilon \mathbf{x}, \quad \mathbf{v} = \varepsilon \mathbf{y}, \quad \mathbf{H}_2(\mathbf{u}, \mathbf{v}, t) = \mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon). \quad (15)$$

С использованием обозначений (15) уравнение (14) переписывается в виде:

$$\begin{aligned} \varepsilon^2 \frac{\partial \mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon)}{\partial t} + \mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon) \lambda \sum_{i=1}^n p_i S_i(t) (j\varepsilon x_i + j\varepsilon y_i a_1^{(i)}) = \\ = \mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon) \left[\bar{\Lambda} \sum_{i=1}^n p_i S_i(t) (e^{j\varepsilon x_i} G_i^*(\varepsilon y_i) - 1) + \bar{\mathbf{Q}} \right]. \end{aligned} \quad (16)$$

Найдём асимптотическое, при $\varepsilon \rightarrow 0$, решение этой задачи, т.е. $\mathbf{F}_2(\mathbf{x}, \mathbf{y}, t) = \lim_{\varepsilon \rightarrow 0} \mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon)$.

Этап 1. Выполним предельный переход при $\varepsilon \rightarrow 0$ в (16), получим $\mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon) \bar{\mathbf{Q}} = 0$. Представим $\mathbf{F}_2(\mathbf{x}, \mathbf{y}, t)$ в виде

$$\mathbf{F}_2(\mathbf{x}, \mathbf{y}, t) = \mathbf{r} \Phi_2(\mathbf{x}, \mathbf{y}, t), \quad (17)$$

где $\Phi_2(\mathbf{x}, \mathbf{y}, t)$ – некоторая скалярная дифференцируемая функция, удовлетворяющая условию $\Phi_2(\mathbf{x}, \mathbf{y}, t_0) = 1$.

Этап 2. Решение уравнения (16) запишем в виде разложения

$$\mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon) = \Phi_2(\mathbf{x}, \mathbf{y}, t) \left\{ \mathbf{r} + \mathbf{g} \sum_{i=1}^n p_i S_i(t) (j\varepsilon x_i + j\varepsilon y_i a_1^{(i)}) \right\} + O(\varepsilon^2), \quad (18)$$

где \mathbf{g} – некоторая вектор-строка. Подставим разложение (18) в (16), используя разложение (11), получим матричное уравнение для вектора \mathbf{g} : $\mathbf{g} \bar{\mathbf{Q}} = \mathbf{r} (\lambda \mathbf{I} - \bar{\Lambda})$.

Этап 3. Домножим (16) на вектор \mathbf{e} , используя (18) и разложение $e^{j\varepsilon x} = 1 + j\varepsilon x + \frac{(j\varepsilon x)^2}{2} + O(\varepsilon^3)$, в результате несложных преобразований, получаем

$$\begin{aligned} \frac{\partial \Phi_2(\mathbf{x}, \mathbf{y}, t)}{\partial t} = \Phi_2(\mathbf{x}, \mathbf{y}, t) \left[\sum_{i=1}^n \frac{(jx_i)^2}{2} (\lambda p_i S_i(t) + \kappa p_i^2 S_i^2(t)) + \right. \\ \left. + \sum_{i=1}^n \frac{(jy_i)^2}{2} (\lambda a_2^{(i)} p_i S_i(t) + \kappa (a_1^{(i)})^2 p_i^2 S_i^2(t)) + \sum_{i=1}^n jx_i jy_i (\lambda a_1^{(i)} p_i S_i(t) + \kappa a_1^{(i)} p_i^2 S_i^2(t)) + \right. \\ \left. + \sum_{i=1}^n \sum_{l=1}^n jx_i jx_l \kappa p_i S_i(t) p_l S_l(t) + \sum_{i=1}^n \sum_{l=1}^n jy_i jy_l \kappa a_1^{(i)} p_i S_i(t) a_1^{(l)} p_l S_l(t) + \right. \\ \left. + \sum_{i=1}^n \sum_{l=1}^n jx_i jy_l \kappa a_1^{(l)} p_i S_i(t) p_l S_l(t) \right], \end{aligned}$$

где $\kappa = 2\mathbf{g}(\bar{\Lambda} - \lambda \mathbf{I})\mathbf{e}$. Решение этого уравнения с учетом начального условия, имеет вид:

$$\begin{aligned}
\Phi_2(\mathbf{x}, \mathbf{y}, t) = & \exp \left\{ \sum_{i=1}^n \frac{(jx_i)^2}{2} \left(\lambda p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \right. \\
& + \sum_{i=1}^n \frac{(jy_i)^2}{2} \left(\lambda a_2^{(i)} p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa (a_1^{(i)})^2 p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \\
& + \sum_{i=1}^n jx_i jy_i \left(\lambda a_1^{(i)} p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa a_1^{(i)} p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \sum_{i=1}^n \sum_{\substack{l=1 \\ l \neq i}}^n jx_i jx_l \kappa p_i p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau + \\
& \left. + \sum_{i=1}^n \sum_{\substack{l=1 \\ l \neq i}}^n jy_i jy_l \kappa a_1^{(i)} p_i a_1^{(l)} p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau + \sum_{i=1}^n \sum_{\substack{l=1 \\ l \neq i}}^n jx_i jy_l \kappa a_1^{(l)} p_i p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau \right\},
\end{aligned}$$

подставляя которое в (17), получаем

$$\begin{aligned}
\mathbf{F}_2(\mathbf{x}, \mathbf{y}, t, \varepsilon) = & \mathbf{r} \exp \left\{ \sum_{i=1}^n \frac{(jx_i)^2}{2} \left(\lambda p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \right. \\
& + \sum_{i=1}^n \frac{(jy_i)^2}{2} \left(\lambda a_2^{(i)} p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa (a_1^{(i)})^2 p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \\
& + \sum_{i=1}^n jx_i jy_i \left(\lambda a_1^{(i)} p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa a_1^{(i)} p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \sum_{i=1}^n \sum_{\substack{l=1 \\ l \neq i}}^n jx_i jx_l \kappa p_i p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau + \\
& \left. + \sum_{i=1}^n \sum_{\substack{l=1 \\ l \neq i}}^n jy_i jy_l \kappa a_1^{(i)} p_i a_1^{(l)} p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau + \sum_{i=1}^n \sum_{\substack{l=1 \\ l \neq i}}^n jx_i jy_l \kappa a_1^{(l)} p_i p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau \right\}.
\end{aligned}$$

Выполним замены, обратные к (13) и (15), запишем приближённое равенство для характеристической функции $\mathbf{H}(\mathbf{u}, \mathbf{v}, t)$:

$$\begin{aligned}
\mathbf{H}(\mathbf{u}, \mathbf{v}, t) \approx & \mathbf{r} \exp \left\{ N \lambda \sum_{i=1}^n (ju_i + jv_i a_1^{(i)}) p_i \int_{t_0}^t S_i(\tau) d\tau + \right. \\
& + N \sum_{i=1}^n \frac{(ju_i)^2}{2} \left(\lambda p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \\
& + N \sum_{i=1}^n \frac{(jv_i)^2}{2} \left(\lambda a_2^{(i)} p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa (a_1^{(i)})^2 p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \\
& + N \sum_{i=1}^n ju_i jv_i \left(\lambda a_1^{(i)} p_i \int_{t_0}^t S_i(\tau) d\tau + \kappa a_1^{(i)} p_i^2 \int_{t_0}^t S_i^2(\tau) d\tau \right) + \\
& + N \sum_{i=1}^n \sum_{\substack{l=1 \\ l \neq i}}^n ju_i ju_l \kappa p_i p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau + N \sum_{i=1}^n \sum_{\substack{l=1 \\ l \neq i}}^n jv_i jv_l \kappa a_1^{(i)} p_i a_1^{(l)} p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau + \\
& \left. + N \sum_{i=1}^n \sum_{\substack{l=1 \\ l \neq i}}^n ju_i jv_l \kappa a_1^{(l)} p_i p_l \int_{t_0}^t S_i(\tau) S_l(\tau) d\tau \right\}. \tag{19}
\end{aligned}$$

Теорема 2 доказана.

Следствие. Полагая в (19) $t = T$, $t_0 \rightarrow -\infty$, учитывая (1), получим асимптотическую характеристическую функцию стационарного многомерного распределения вероятностей чисел занятых приборов и суммарного объёма занятых ресурсов каждого типа, которая совпадает с характеристической функцией многомерного гауссовского распределения с вектором математического ожидания $\mathbf{a} = \mathcal{N}\lambda[\mathbf{a}_1 b_1 \quad \mathbf{a}_2 b_2 \quad \dots \quad \mathbf{a}_n b_n]$, где

$$\mathbf{a}_i = [1 \quad a_1^{(i)}], \quad b_i = p_i \int_0^{\infty} (1 - B_i(\tau)) d\tau, \quad \text{и ковариационной матрицей } \mathbf{K} = N(\lambda \mathbf{K}^{(1)} + \kappa \mathbf{K}^{(2)}),$$

где

$$\mathbf{K}^{(1)} = \begin{bmatrix} \mathbf{K}_1^{(1)} b_1 & 0 & \dots & 0 \\ 0 & \mathbf{K}_2^{(1)} b_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \mathbf{K}_n^{(1)} b_n \end{bmatrix}, \quad \mathbf{K}^{(2)} = \begin{bmatrix} \mathbf{K}_{11}^{(2)} & \mathbf{K}_{12}^{(2)} & \dots & \mathbf{K}_{1n}^{(2)} \\ \mathbf{K}_{21}^{(2)} & \mathbf{K}_{22}^{(2)} & \dots & \mathbf{K}_{2n}^{(2)} \\ \dots & \dots & \dots & \dots \\ \mathbf{K}_{n1}^{(2)} & \mathbf{K}_{n2}^{(2)} & \dots & \mathbf{K}_{nn}^{(2)} \end{bmatrix},$$

$$\mathbf{K}_i^{(1)} = \begin{bmatrix} 1 & a_1^{(i)} \\ a_1^{(i)} & a_2^{(i)} \end{bmatrix}, \quad \mathbf{K}_{ii}^{(2)} = \begin{bmatrix} 1 & a_1^{(i)} \\ a_1^{(i)} & a_1^{(i)} a_1^{(i)} \end{bmatrix} p_i p_i \int_0^{\infty} (1 - B_i(\tau))(1 - B_i(\tau)) d\tau.$$

Заключение

Была исследована неоднородная ресурсная СМО с неограниченным числом приборов с входящим ММРР-поток. С помощью метода асимптотического анализа показано, что совместное асимптотическое распределение вероятности чисел занятых приборов и суммарного объёма занятого ресурса каждого типа сходится к многомерному гауссовскому распределению в асимптотическом условии растущей интенсивности входящего потока.

ЛИТЕРАТУРА

1. Вишневикий В.М. Теоретические основы проектирования компьютерных сетей. – М.: Техносфера, 2003. – 506 с.
2. Матальцкий М.А., Тихоненко О.М., Паньков А.В. Теория массового обслуживания и ее применения // Гродно: ГрГУ, 2008. – 771 с.
3. Панкратова Е.В. Исследование системы массового обслуживания MAP|M|∞ с разнотипным обслуживанием методом асимптотического анализа в условии предельно редких изменений состояний входящего потока // Распределенные компьютерные и телекоммуникационные сети: Управление, вычисление, связь. – 2015. – С. 585–592.
4. Панкратова Е.В. Исследование системы массового обслуживания GI/GI/∞ с двумя типами заявок // Информационные технологии и математическое моделирование (ИТММ-2015): материалы XIV Международной конференции им. А.Ф. Терпугова. – 2015. – Ч. 1. – С. 152–157.
5. Pankratova E., Moiseeva S. Queueing System GI/GI/∞ with n Types of Customers // Communications in Computer and Information Science. – Switzerland: Springer. – 2015. V. 564. – P. 216–225.
6. Dudin S., Kim C., Dudina O. MMAP|M|N queueing system with impatient heterogeneous customers as a model of a contact center // Computers & Operations Research. – 2013. – V. 40. – № 7. – P. 1790–1803.
7. Моисеева С.П., Назаров А.А. Методы асимптотического анализа в теории массового обслуживания. – Томск: Изд-во НТЛ, 2006. – 112 с.
8. Крысанова К.А., Моисеева С.П. Исследование системы параллельного обслуживания кратных заявок потока марковского восстановления методом асимптотического анализа // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2012. – № 1. (18). – С. 49–55.
9. Ефросинин Д.В. Методы анализа управляемых динамических систем: дис...доктора физ.-мат. наук: 05.13.01 / Москва, 2013. 332 с.
10. Моисеев А.Н., Назаров А.А. Бесконечнолинейные системы и сети массового обслуживания. – Томск: Изд-во НТЛ, 2015. 240 с.
11. Назаров А.А., Моисеева С.П. Метод асимптотического анализа в теории массового обслуживания // Томск: Изд-во НТЛ, 2006. – 112 с.
12. Лисовская Е.Ю., Моисеева С.П. Асимптотический анализ системы ММРР|GI|∞ с обслуживанием требований случайного объема // Труды Томского государственного университета. – Т. 299. Серия физико-математическая: Математическое и программное обеспечение информационных, технических и экономических систем: материалы IV Международной молодежной научной конференции. Томск, 20–21 мая 2016 г. / под общ. ред. И.С. Шмырина. – Томск: Издательский Дом Томского государственного университета, 2016. – С. 99–104.

13. Лисовская Е.Ю., Моисеева С.П. Исследование бесконечнолинейной системы массового обслуживания требований случайного объема с входящим ММРР-поток // Информационные технологии и математическое моделирование (ИТММ–2016): материалы XV Международной конференции им. А.Ф. Терпугова (12–16 сентября 2016 г.). Томск: Изд-во Том. ун-та, 2016, Ч. 1. – С. 77–82.

14. Лисовская Е.Ю., Моисеева С.П. Суммарный объем заявок в бесконечнолинейной системе массового обслуживания с рекуррентным входящим потоком // Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2016): материалы Девятнадцатой международной научной конференции, 21–25 нояб. 2016 г.: в 3 т.; под общ. ред. В.М. Вишневого и К.Е. Самуйлова – М.: РУДН, 2016. – С. 313–325.

АСИМПТОТИЧЕСКИЙ АНАЛИЗ РЕСУРСНОЙ ГЕТЕРОГЕННОЙ СМО ВИДА ММРР^(v1,v2)|GI_∞

Е.А. Павлова, С.П. Моисеева

Томский государственный университет
pavlovakatya_2010@mail.ru, smoiseeva@mail.ru

Введение

В настоящее время телекоммуникации играют большую роль и встречаются на каждом шагу. Требования к эффективности и производительности устройств связи постоянно растут, поэтому так актуально построение и исследование математических моделей [1]. Для описания процессов в телекоммуникационных системах применяются системы массового обслуживания (СМО) с неоднородными обслуживающими устройствами. При построении математических моделей таких процессов часто используются системы с непуассоновскими входящими потоками [4]. Например, с помощью ММРР-потока можно моделировать изменчивый характер трафика.

В настоящей статье помощью метода асимптотического анализа [2,6] приводится исследование числа занятых приборов и объемов ресурсов в бесконечнолинейной гетерогенной СМО [3,8], на вход которой поступает ММРР-поток.

1. Постановка задачи

Рассмотрим систему массового обслуживания с неограниченным числом обслуживающих приборов двух различных типов (рис. 1).

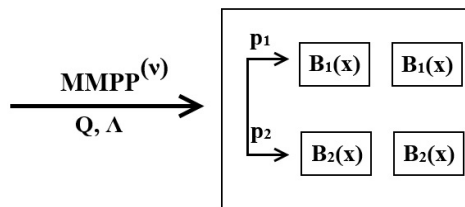


Рис. 1. Система массового обслуживания ММРР^(v1, v2)|GI_∞ повторными обращениями

Каждая поступающая заявка требует случайное количество ресурса одного из двух типов. На вход системы поступает ММРР-поток заявок, управляемый цепью Маркова $k(t)$ с конечным числом состояний K , заданный матрицей инфинитезимальных характеристик $Q = \|q_{ij}\|$, $i, j = \overline{1, K}$ и матрицей интенсивностей $\Lambda = \text{diag}[\lambda_1, \dots, \lambda_K]$.

Поступающая заявка потребует прибор i -го типа с вероятностью p_i , $i = 1, 2$. Заявка становится на прибор соответствующего типа, где обслуживается случайное время с функцией распределения $B_i(x)$, $i = 1, 2$, также зависящей от типа заявки. Пусть каждая заявка i -го типа требует некоторый ресурс случайного объема $V_i > 0$, $i = 1, 2$ с функцией распределения $G_i(y)$, $i = 1, 2$.

Обозначим $\{i_1(t), i_2(t)\}$ и $\{V_1(t), V_2(t)\}$ – число занятых приборов каждого типа в системе в момент времени t и объём их ресурса соответственно. Ставится задача исследования 4-хмерного случайного процесса $\{i_1(t), i_2(t), V_1(t), V_2(t)\}$. Так как процесс не является марковским, воспользуемся методом динамического просеивания для его исследования (рис. 2).

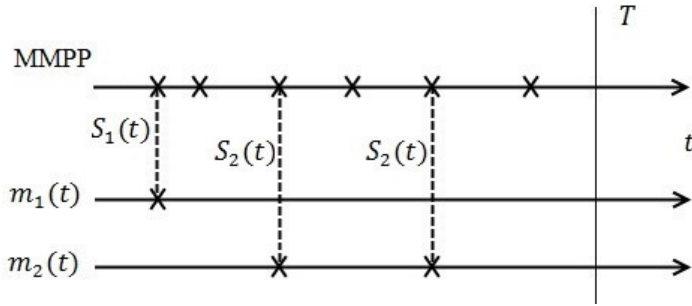


Рис. 2. Генерация событий в просеянных потоках

Предположим, что в системе не было заявок в момент времени t_0 , зафиксируем некоторый момент времени T . С вероятностью $S_i(t)$ просеивается заявка i -го типа, поступающая в систему в момент времени t , не завершившая обслуживание к моменту времени T , $S_i(t) = 1 - B_i(T - t)$, $i = 1, 2$.

Обозначим $m_1(t)$, $m_2(t)$ и $w_1(t)$, $w_2(t)$ – число событий в просеянном потоке и объём их ресурса соответственно.

Распределение вероятностей числа заявок в системе в момент времени T совпадает с распределением вероятностей числа событий в просеянном потоке:

$$P\{i_1(T) = m_1, i_2(T) = m_2\} = P\{m_1(T) = m_1, m_2(T) = m_2\}, \quad m_i = 0, 1, 2, \dots$$

Нетрудно показать это свойство и для процесса $\{i_1(t), i_2(t), V_1(t), V_2(t)\}$:

$$\begin{aligned} & P\{i_1(T) = m_1, i_2(T) = m_2, V_1(T) < w_1, V_2(T) < w_2\} = \\ & = P\{m_1(T) = m_1, m_2(T) = m_2, w_1(T) < w_1, w_2(T) < w_2\}, \quad m_i = 0, 1, 2, \dots \end{aligned}$$

2. Система дифференциальных уравнений Колмогорова

Рассмотрим пятимерный марковский процесс $\{k(t), m_1(t), m_2(t), w_1(t), w_2(t)\}$ и обозначим

$$P(k, m_1, m_2, w_1, w_2, t) = P\{k(t) = k, m_1(t) = m_1, m_2(t) = m_2, w_1(t) < w_1, w_2(t) < w_2\},$$

тогда по формуле полной вероятности [5] можем записать следующую систему дифференциальных уравнений Колмогорова

$$\begin{aligned} & \frac{\partial P(k, m_1, m_2, w_1, w_2, t + \Delta t)}{\partial t} = -\lambda_k P(k, m_1, m_2, w_1, w_2, t) + \\ & + \lambda_k p_1 (1 - S_1(t)) P(k, m_1, m_2, w_1, w_2, t) + \lambda_k p_2 (1 - S_2(t)) P(k, m_1, m_2, w_1, w_2, t) + \\ & + \lambda_k p_1 S_1(t) \int_0^{w_1} P(k, m_1 - 1, m_2, w_1 - y, w_2, t) dG_1(y) + \\ & + \lambda_k p_2 S_2(t) \int_0^{w_2} P(k, m_1, m_2 - 1, w_1, w_2 - y, t) dG_2(y) + \sum_v q_{vk} P(v, m_1, m_2, w_1, w_2, t), \end{aligned} \quad (1)$$

$$k = \overline{1, K}, \quad m_i = 0, 1, 2, \dots, \quad w_i > 0, \quad i = 1, 2.$$

Введём частичные характеристические функции следующего вида:

$$h(k, u_1, u_2, z_1, z_2, t) = \sum_{m_1=0}^{\infty} \sum_{m_2=0}^{\infty} e^{ju_1 m_1} e^{ju_2 m_2} \int_0^{\infty} e^{jz_1 w_1} e^{jz_2 w_2} P(k, m_1, m_2, dw_1, dw_2, t), \quad j = \sqrt{-1}.$$

Тогда система (1) примет вид

$$\begin{aligned} \frac{\partial h(k, u_1, u_2, z_1, z_2, t)}{\partial t} &= \lambda_k h(k, u_1, u_2, z_1, z_2, t) \cdot \left[p_1 S_1(t) \cdot (e^{ju_1} G_1^*(z_1) - 1) + \right. \\ &\quad \left. + p_2 S_2(t) \cdot (e^{ju_2} G_2^*(z_2) - 1) \right] + \sum_{\nu} q_{\nu k} h(\nu, u_1, u_2, z_1, z_2, t), \end{aligned} \quad (2)$$

$$k = \overline{1, K}, \quad G_i^*(z_i) = \int_0^{\infty} e^{jz_i y} dG_i(y), \quad i = 1, 2.$$

Перепишем систему (2) в матричном виде

$$\frac{\partial \mathbf{h}(u_1, u_2, z_1, z_2, t)}{\partial t} = \mathbf{h}(u_1, u_2, z_1, z_2, t) \cdot \left[\sum_{i=1}^2 p_i (e^{ju_i} G_i^*(z_i) - 1) S_i(t) \mathbf{\Lambda} + \mathbf{Q} \right] \quad (3)$$

с начальным условием $\mathbf{h}(u_1, u_2, z_1, z_2, t_0) = \mathbf{r}$, где

$$\mathbf{h}(u_1, u_2, z_1, z_2, t) = [h(1, u_1, u_2, z_1, z_2, t), \dots, h(K, u_1, u_2, z_1, z_2, t)]$$

и $\mathbf{r} = [r(1), \dots, r(K)]$ – стационарное распределение вероятностей цепи Маркова, удовлетворяющее системе уравнений

$$\begin{cases} \mathbf{rQ} = 0, \\ \mathbf{r}\mathbf{e} = 1, \end{cases}$$

где $\mathbf{e} = [1, \dots, 1]^T$.

3. Метод асимптотического анализа

В общем случае точное решение уравнения (3) невозможно, но оно может быть найдено в некоторых асимптотических условиях. Для поиска решения воспользуемся методом асимптотического анализа [7] в условии эквивалентного роста времени обслуживания на приборах разного типа.

Характеристическая функция распределения вероятностей многомерного процесса $\{k(t), m_1(t), m_2(t), w_1(t), w_2(t)\}$ в первой асимптотике имеет вид

$$\mathbf{h}^{(1)}(u_1, u_2, z_1, z_2, t) = \exp \left\{ j\kappa_1 \sum_{i=1}^2 p_i (u_i + z_i a_i) b_i \right\},$$

где $\kappa_1 = \mathbf{r} \cdot \mathbf{\Lambda} \cdot \mathbf{e}$, $b_i = \int_0^{\infty} (1 - B_i(x)) dx$ – среднее значение времени обслуживания заявки

на приборе i -го типа, $a_i = \int_0^{\infty} y dG_i(y)$ – среднее значение объема i -го типа ресурса, требуемого заявкой.

Асимптотическая характеристическая функция второго порядка распределения вероятностей процесса $\{k(t), m_1(t), m_2(t), w_1(t), w_2(t)\}$:

$$\begin{aligned} \mathbf{h}^{(2)}(u_1, u_2, z_1, z_2, t) &= \exp \left\{ j\kappa_1 \sum_{i=1}^2 p_i (u_i + z_i a_i) b_i + \right. \\ &\quad \left. + \frac{j^2}{2} \kappa_1 \sum_{i=1}^2 p_i (u_i^2 + z_i^2 a_{2i}) b_i + j^2 \kappa_1 \sum_{i=1}^2 p_i u_i z_i a_i^2 b_i + \right. \end{aligned}$$

$$+ j\kappa_2 \left. \left. \sum_{i=1}^2 p_i^2 (u_i + z_i a_{1i})^2 b_i + \sum_{i=1}^2 \sum_{\substack{g=1 \\ g \neq i}}^2 p_i p_g (u_i + z_i a_{1i})(u_g + z_g a_{1g}) b_i b_g \right\} \right\}$$

где $\kappa_2 = \mathbf{f}_2 (\mathbf{A} - \kappa_1 \mathbf{I}) \mathbf{e}$, $a_{2i} = \int_0^{\infty} y^2 dG_i(y)$, а вектор-строка \mathbf{f}_2 удовлетворяет линейной матричной системе уравнений

$$\begin{cases} \mathbf{f}_2 \mathbf{Q} = \mathbf{r} (\kappa_1 \mathbf{I} - \mathbf{A}), \\ \mathbf{f}_2 \mathbf{e} = 0. \end{cases}$$

Полученное выражение будем называть асимптотикой второго порядка числа занятых приборов и объемов ресурсов каждого типа в системе $\text{MMPP}^{(V_1, V_2)} | \text{GI} |_{\infty}$ в момент времени t .

Заключение

В работе рассмотрена математическая модель СМО $\text{MMPP}^{(V_1, V_2)} | \text{GI} |_{\infty}$, с помощью метода асимптотического анализа получены асимптотические приближения моментов первого и второго порядка числа занятых приборов и объемов ресурса, требуемого заявками, двух типов в рассматриваемой системе в условии эквивалентного роста времени обслуживания.

ЛИТЕРАТУРА

1. *Lisovskaya E., Moiseeva S., Pagano M., Potatueva V.* Study of the MMPP/GI/ Queueing System with Random Customers' Capacities // Informatics and Applications. – 2017. – Vol. 11, No. 4. – P. 111–119.
2. *Боровков А.А.* Асимптотические методы в теории массового обслуживания. – М.: Наука, 1980. – 382 с.
3. *Моисеев А.Н., Назаров А.А.* Бесконечнолинейные системы и сети массового обслуживания // Томск: Изд-во НТЛ, 2015. – 240 с.
4. *Моисеева С.П.* Разработка методов исследования математических моделей немарковских систем обслуживания с неограниченным числом приборов и непугассоновскими входящими потоками: дис. д-р. физ.-мат. наук: 05.13.18. – Томск, 2014. – 280 с.
5. *Назаров А.А., Терпугов А.Ф.* Теория массового обслуживания. – Томск: Изд-во НТЛ, 2005. – 228 с.
6. *Назаров А.А.* Асимптотический анализ марковизируемых систем. – Томск: Изд-во Том. ун-та, 1991. – 153 с.
7. *Назаров А.А., Моисеева С.П.* Метод асимптотического анализа в теории массового обслуживания. – Томск: Изд-во НТЛ, 2006. – 112 с.
8. *Панкратова Е.В.* Исследование системы массового обслуживания GI/GI/∞ с двумя типами заявок // Информационные технологии и математическое моделирование (ИТММ–2015): материалы XIV Междунар. конф. им. А.Ф. Терпугова (Анжеро-Судженск, 18–22 ноября 2015). – Томск: Изд-во Том. ун-та, 2015. – Ч. 1. – С. 152.

ИССЛЕДОВАНИЕ СИСТЕМЫ $\text{MMPP}(n) | \text{M}(n) |_{\infty}$ МЕТОДОМ МОМЕНТОВ

Е.П. Полин, С.П. Моисеева

*Томский государственный университет
polin_evgeny@mail.ru, smoiseeva@mail.ru*

Введение

Развитие техники, телефонии, спутниковых, компьютерных, беспроводных и мобильных сетей связи привело к необходимости применения более адекватных математических моделей процессов передачи и обработки данных и дало существенный толчок к развитию исследований систем с изменяемыми параметрами [1]. Особенно актуальным представляется исследование таких систем при оценке ситуации и организации управления в современных и перспективных информационно-вычислительных сетях и сетях связи.

Большинство современных технических систем, в том числе систем передачи информации и телекоммуникационных систем, функционируют в условиях изменяющейся внешней среды, которые носят как регулярный, так и случайный характер [2]. Такие системы в теории массового обслуживания называются системами массового обслуживания (СМО) в случайной среде [3]. Эти СМО более адекватно по сравнению с классическими марковскими системами отображают реальные процессы, связанные с изменяющейся во времени внешней случайной средой и реакцией самой системы на эти изменения.

В данной статье приводится исследование методом моментов [4] системы массового обслуживания, функционирующей в условиях изменяющейся внешней среды.

1. Постановка задачи

Рассматривается неоднородная система массового обслуживания с неограниченным числом приборов, функционирующая в случайной среде (рис. 1).

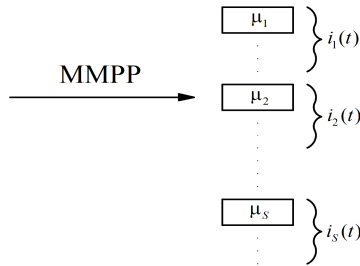


Рис. 1. Система массового обслуживания MMPP(n)|M(n)|∞

Входящий MММР-поток задается матрицами $(\mathbf{Q}, \mathbf{\Lambda})$, где \mathbf{Q} – матрица инфинитесимальных характеристик управляющей цепи Маркова $s(t)$, $\mathbf{\Lambda}$ – матрица интенсивностей. Дисциплина обслуживания определяется следующим образом: если заявка поступает с интенсивностью λ_n , то она обслуживается случайное время, распределённое по экспоненциальному закону $F(x) = 1 - e^{-\mu_n x}$, не меняющееся при изменении состояния среды.

Таким образом, ставится задача исследования многомерного случайного процесса $\mathbf{i}(t) = [i_1(t), i_2(t), \dots, i_s(t)]$ – числа занятых приборов разного типа в системе.

Так как $\{i_1(t), i_2(t), \dots, i_s(t)\}$ – немарковский случайный процесс, будем рассматривать многомерную цепь Маркова $\{i_1(t), i_2(t), \dots, i_s(t), s(t)\}$.

2. Система дифференциальных уравнений Колмогорова

Обозначим $P(i_1, i_2, \dots, i_s, n, t) = P\{i_1(t) = i_1, i_2(t) = i_2, \dots, i_s(t) = i_s, s(t) = n\}$, $\mathbf{e}_1 = [1, 0, \dots, 0]$, $\mathbf{e}_2 = [0, 1, \dots, 0]$, ..., $\mathbf{e}_l = [0, 0, \dots, 1, \dots, 0]$, тогда для распределения вероятностей получаем систему дифференциальных уравнений Колмогорова

$$\frac{\partial P(\mathbf{i}, n, t)}{\partial t} = -\left(\lambda_n + \sum_{l=1}^s \mu_l i_l\right) P(\mathbf{i}, n, t) + \lambda_n P(\mathbf{i} - \mathbf{e}_n, n, t) + \sum_{l=1}^s \mu_l (i_l + 1) P(\mathbf{i} + \mathbf{e}_l, n, t) + \sum_{v=1}^S q_{vn} P(\mathbf{i}, v, t), \quad n = \overline{1, S}.$$

Для стационарного распределения вероятностей $P(\mathbf{i}, n) \equiv P(\mathbf{i}, n, t)$ перепишем данную систему в виде

$$-\left(\lambda_n + \sum_{l=1}^S \mu_l i_l\right) P(\mathbf{i}, n) + \lambda_n P(\mathbf{i} - \mathbf{e}_n, n) + \sum_{l=1}^S \mu_l (i_l + 1) P(\mathbf{i} + \mathbf{e}_l, n) + \sum_v q_{vn} P(\mathbf{i}, v) = 0, \quad n = \overline{1, S}.$$

Тогда для частичных характеристических функций вида

$$H(u_1, u_2, \dots, u_S, n) = H(\mathbf{u}, n) = \sum e^{j\mu_1 i_1} \sum e^{j\mu_2 i_2} \dots \sum e^{j\mu_S i_S} P(\mathbf{i}, n), \quad n = \overline{1, S}$$

имеем следующую систему уравнений

$$j \sum_{k=1}^S \mu_k (e^{-j\mu_k} - 1) \frac{\partial H(\mathbf{u}, n)}{\partial u_k} = \lambda_n H(\mathbf{u}, n) (e^{j\mu_n} - 1) + \sum_v q_{vn} H(\mathbf{u}, v), \quad n = \overline{1, S}.$$

Систему уравнений для $H(\mathbf{u}, n)$ в матричном виде запишем следующим образом:

$$j\boldsymbol{\mu}(\mathbf{u}) \frac{\partial \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}} = \mathbf{h}(\mathbf{u}) (\boldsymbol{\Lambda}(\mathbf{u}) + \mathbf{Q}), \quad (1)$$

где

$$\mathbf{h}(\mathbf{u}) = [H(\mathbf{u}, 1), H(\mathbf{u}, 2) \dots H(\mathbf{u}, S)], \quad \boldsymbol{\mu}(\mathbf{u}) = [\mu_1 (e^{-j\mu_1} - 1) \quad \mu_2 (e^{-j\mu_2} - 1) \quad \dots \quad \mu_S (e^{-j\mu_S} - 1)],$$

$$\frac{\partial \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}} = \begin{bmatrix} \frac{\partial H(\mathbf{u}, 1)}{\partial u_1} & \dots & \frac{\partial H(\mathbf{u}, S)}{\partial u_1} \\ \dots & \dots & \dots \\ \frac{\partial H(\mathbf{u}, 1)}{\partial u_S} & \dots & \frac{\partial H(\mathbf{u}, S)}{\partial u_S} \end{bmatrix}, \quad \boldsymbol{\Lambda}(\mathbf{u}) = \begin{bmatrix} \lambda_1 (e^{j\mu_1} - 1) & 0 & \dots & 0 \\ 0 & \lambda_2 (e^{j\mu_2} - 1) & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \lambda_S (e^{j\mu_S} - 1) \end{bmatrix},$$

$$\mathbf{Q} = \begin{bmatrix} q_{11} & \dots & q_{1S} \\ \dots & \dots & \dots \\ q_{S1} & \dots & q_{SS} \end{bmatrix}.$$

Полученная система уравнений позволяет определить основные характеристики рассматриваемой системы.

3. Моменты первого порядка

Для нахождения начальных моментов первого порядка продифференцируем систему (1) по \mathbf{u} , в результате дифференцирования получим уравнение

$$j \left(\frac{\partial \boldsymbol{\mu}(\mathbf{u})}{\partial \mathbf{u}} \cdot \frac{\partial \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}} + \boldsymbol{\mu}(\mathbf{u}) \frac{\partial^2 \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}^2} \right) = \frac{\partial \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}} (\boldsymbol{\Lambda}(\mathbf{u}) + \mathbf{Q}) + \mathbf{h}(\mathbf{u}) \frac{\partial \boldsymbol{\Lambda}(\mathbf{u})}{\partial \mathbf{u}}, \quad (2)$$

где

$$\frac{\partial \boldsymbol{\mu}(\mathbf{u})}{\partial \mathbf{u}} = \begin{bmatrix} -j\mu_1 e^{-j\mu_1} & 0 & \dots & 0 \\ 0 & -j\mu_2 e^{-j\mu_2} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & -j\mu_S e^{-j\mu_S} \end{bmatrix}, \quad \frac{\partial \boldsymbol{\Lambda}(\mathbf{u})}{\partial \mathbf{u}} = \begin{bmatrix} j\lambda_1 e^{j\mu_1} & 0 & \dots & 0 \\ 0 & j\lambda_2 e^{j\mu_2} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & j\lambda_S e^{j\mu_S} \end{bmatrix}.$$

В уравнение (2) подставим $\mathbf{u} = \mathbf{0}$, получим следующую систему уравнений для нахождения моментов первого порядка

$$\mathbf{A} \cdot \mathbf{M} - \mathbf{M} \cdot \mathbf{Q} = \mathbf{R}\mathbf{A}, \quad (3)$$

$$\text{где } \mathbf{M} = \begin{bmatrix} m_1^{(1)}(1) & \dots & m_1^{(1)}(S) \\ \dots & \dots & \dots \\ m_1^{(S)}(1) & \dots & m_1^{(S)}(S) \end{bmatrix} \text{ — матрица условных моментов первого порядка,}$$

$$\mathbf{A} = \begin{bmatrix} \mu_1 & 0 & \dots & 0 \\ 0 & \mu_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \mu_s \end{bmatrix}, \quad \mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \lambda_s \end{bmatrix},$$

$$\mathbf{R} = \begin{bmatrix} R_1 & 0 & \dots & 0 \\ 0 & R_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & R_s \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} q_{11} & \dots & q_{1s} \\ \dots & \dots & \dots \\ q_{s1} & \dots & q_{ss} \end{bmatrix}.$$

Домножив уравнение (3) на единичный вектор-столбец и учитывая, что $\mathbf{Qe} = \mathbf{0}$,

получим $\mathbf{Am} = \mathbf{RLe}$, откуда $\mathbf{m} = \mathbf{A}^{-1}\mathbf{RLe}$. Здесь $\mathbf{m} = \begin{bmatrix} m_1^1 \\ \vdots \\ m_1^s \end{bmatrix}$ – вектор-столбец первых

моментов числа занятых приборов каждого типа.

4. Моменты второго порядка

Для нахождения моментов второго порядка продифференцируем систему (2) по \mathbf{u} , в результате дифференцирования получим уравнение

$$j \left(\frac{\partial^2 \boldsymbol{\mu}(\mathbf{u})}{\partial \mathbf{u}^2} \cdot \frac{\partial \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}} + 2 \frac{\partial \boldsymbol{\mu}(\mathbf{u})}{\partial \mathbf{u}} \cdot \frac{\partial^2 \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}^2} + \boldsymbol{\mu}(\mathbf{u}) \frac{\partial^3 \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}^3} \right) =$$

$$= \frac{\partial^2 \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}^2} (\mathbf{\Lambda}(\mathbf{u}) + \mathbf{Q}) + 2 \frac{\partial \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}} \cdot \frac{\partial \mathbf{\Lambda}(\mathbf{u})}{\partial \mathbf{u}} + \mathbf{h}(\mathbf{u}) \frac{\partial^2 \mathbf{\Lambda}(\mathbf{u})}{\partial \mathbf{u}^2}, \quad (4)$$

где

$$\frac{\partial^2 \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}^2} = \begin{bmatrix} \frac{\partial^2 H(\mathbf{u}, 1)}{\partial u_1^2} & \dots & \frac{\partial^2 H(\mathbf{u}, S)}{\partial u_1^2} \\ \dots & \dots & \dots \\ \frac{\partial^2 H(\mathbf{u}, 1)}{\partial u_s^2} & \dots & \frac{\partial^2 H(\mathbf{u}, S)}{\partial u_s^2} \end{bmatrix}, \quad \frac{\partial^2 \boldsymbol{\mu}(\mathbf{u})}{\partial \mathbf{u}^2} = \begin{bmatrix} j^2 \mu_1 e^{-j\mu_1} & 0 & \dots & 0 \\ 0 & j^2 \mu_2 e^{-j\mu_2} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & j^2 \mu_s e^{-j\mu_s} \end{bmatrix},$$

$$\frac{\partial^2 \mathbf{\Lambda}(\mathbf{u})}{\partial \mathbf{u}^2} = \begin{bmatrix} j^2 \lambda_1 e^{j\mu_1} & 0 & \dots & 0 \\ 0 & j^2 \lambda_2 e^{j\mu_2} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & j^2 \lambda_s e^{j\mu_s} \end{bmatrix}.$$

В уравнение (4) подставим $\mathbf{u} = \mathbf{0}$, получим следующую систему уравнений для нахождения моментов второго порядка $2\mathbf{A} \cdot \mathbf{M}_2 - \mathbf{M}_2 \mathbf{Q} = 2\mathbf{M} \cdot \mathbf{\Lambda} + \mathbf{A} \cdot \mathbf{M} + \mathbf{R}\mathbf{L}$, где

$\mathbf{M}_2 = \begin{bmatrix} m_2^{(1)}(1) & \dots & m_2^{(1)}(S) \\ \dots & \dots & \dots \\ m_2^{(S)}(1) & \dots & m_2^{(S)}(S) \end{bmatrix}$ – матрица условных моментов второго порядка.

5. Моменты первого и второго порядков системы MMPP(2)|M(2) $_{\infty}$

Рассмотрим частный случай исследуемой системы, при котором изменяющаяся внешняя среда принимает только два различных состояния.

Для распределения вероятностей $P(i_1, i_2, s, n) = P\{i_1(t) = i_1, i_2(t) = i_2, s(t) = n\}$ можно записать следующую систему дифференциальных уравнений Колмогорова:

$$\begin{aligned} \frac{\partial P(i_1, i_2, 1, t)}{\partial t} &= \lambda_1 P(i_1 - 1, i_2, 1, t) + \{\mu_1 (i_1 + 1) P(i_1 + 1, i_2, 1, t) + \mu_2 (i_2 + 1) P(i_1, i_2 + 1, 1, t)\} - \\ &\quad - (\lambda_1 + \{i_1 \mu_1 + i_2 \mu_2\}) P(i_1, i_2, 1, t) + q_{11} P(i_1, i_2, 1, t) + q_{21} P(i_1, i_2, 2, t), \\ \frac{\partial P(i_1, i_2, 2, t)}{\partial t} &= \lambda_2 P(i_1 - 1, i_2, 2, t) + \{\mu_1 (i_1 + 1) P(i_1 + 1, i_2, 2, t) + \mu_2 (i_2 + 1) P(i_1, i_2 + 1, 2, t)\} - \\ &\quad - (\lambda_2 + \{i_1 \mu_1 + i_2 \mu_2\}) P(i_1, i_2, 2, t) + q_{12} P(i_1, i_2, 1, t) + q_{22} P(i_1, i_2, 2, t). \end{aligned}$$

В стационарном режиме эта система принимает вид:

$$\begin{aligned} 0 &= \lambda_1 P(i_1 - 1, i_2, 1) + \{\mu_1 (i_1 + 1) P(i_1 + 1, i_2, 1) + \mu_2 (i_2 + 1) P(i_1, i_2 + 1, 1)\} - \\ &\quad - (\lambda_1 + \{i_1 \mu_1 + i_2 \mu_2\}) P(i_1, i_2, 1) + q_{11} P(i_1, i_2, 1) + q_{21} P(i_1, i_2, 2), \\ 0 &= \lambda_2 P(i_1 - 1, i_2, 2) + \{\mu_1 (i_1 + 1) P(i_1 + 1, i_2, 2) + \mu_2 (i_2 + 1) P(i_1, i_2 + 1, 2)\} - \\ &\quad - (\lambda_2 + \{i_1 \mu_1 + i_2 \mu_2\}) P(i_1, i_2, 2) + q_{12} P(i_1, i_2, 1) + q_{22} P(i_1, i_2, 2). \end{aligned}$$

Введём частичные характеристические функции:

$H(u_1, u_2, n) = \sum_{i_1=0}^{\infty} e^{ju_1 i_1} \sum_{i_2=0}^{\infty} e^{ju_2 i_2} P(i_1, i_2, n)$, $n = 1, 2$. Домножив на $e^{ju_1 i_1}$, $n = 1, 2$ и просуммировав по i от 0 до ∞ , получим систему уравнений для частных характеристических функций:

$$\begin{aligned} j \left\{ \mu_1 \frac{\partial H(u_1, u_2, 1)}{\partial u_1} (e^{-ju_1} - 1) + \mu_2 \frac{\partial H(u_1, u_2, 1)}{\partial u_2} (e^{-ju_2} - 1) \right\} &= \\ = \lambda_1 H(u_1, u_2, 1) (e^{ju_1} - 1) + q_{11} H(u_1, u_2, 1) + q_{21} H(u_1, u_2, 2), \\ j \left\{ \mu_1 \frac{\partial H(u_1, u_2, 2)}{\partial u_1} (e^{-ju_1} - 1) + \mu_2 \frac{\partial H(u_1, u_2, 2)}{\partial u_2} (e^{-ju_2} - 1) \right\} &= \\ = \lambda_2 H(u_1, u_2, 2) (e^{ju_2} - 1) + q_{12} H(u_1, u_2, 1) + q_{22} H(u_1, u_2, 2). \end{aligned} \quad (5)$$

Продифференцируем систему (5) по u_1 и u_2 , в полученных выражениях положим $u_1 = 0$ и $u_2 = 0$. Получим систему уравнений для нахождения моментов первого порядка:

$$\begin{aligned} \mu_1 m_1^{(1)}(1) &= \lambda_1 R_1 + q_{11} m_1^{(1)}(1) + q_{21} m_1^{(1)}(2), \\ \mu_2 m_1^{(2)}(1) &= q_{11} m_1^{(2)}(1) + q_{21} m_1^{(2)}(2), \\ \mu_1 m_1^{(1)}(2) &= q_{12} m_1^{(1)}(1) + q_{22} m_1^{(1)}(2), \\ \mu_2 m_1^{(2)}(2) &= \lambda_2 R_2 + q_{12} m_1^{(2)}(1) + q_{22} m_1^{(2)}(2). \end{aligned}$$

Для нахождения моментов второго порядка дважды продифференцируем систему (5) по u_1 и u_2 и в полученных выражениях положим $u_1 = 0$ и $u_2 = 0$. Система уравнений для моментов второго порядка и смешанных моментов имеет вид:

$$\begin{aligned}
2\mu_1 m_2^{(1)}(1) - \mu_1 m_1^{(1)}(1) &= q_{11} m_2^{(1)}(1) + q_{21} m_2^{(1)}(2) + 2\lambda_1 m_1^{(1)}(1) + \lambda_1 R_1, \\
2\mu_1 m_2^{(1)}(2) - \mu_1 m_1^{(1)}(2) &= q_{12} m_2^{(1)}(1) + q_{22} m_2^{(1)}(2), \\
\mu_2 m_{1,2}(1) + \mu_1 m_{1,2}(1) &= q_{11} m_{1,2}(1) + q_{21} m_{1,2}(2) + \lambda_1 m_1^{(2)}(1), \\
\mu_2 m_{1,2}(2) + \mu_1 m_{1,2}(2) &= q_{12} m_{1,2}(1) + q_{22} m_{1,2}(2) + \lambda_2 m_1^{(1)}(2), \\
2\mu_2 m_2^{(2)}(1) - \mu_2 m_1^{(2)}(1) &= q_{11} m_2^{(2)}(1) + q_{21} m_2^{(2)}(2), \\
2\mu_2 m_2^{(2)}(2) - \mu_2 m_1^{(2)}(2) &= q_{12} m_2^{(2)}(1) + q_{22} m_2^{(2)}(2) + 2\lambda_2 m_1^{(2)}(2) + \lambda_2 R_2.
\end{aligned} \tag{6}$$

Решая систему (6), получим выражения для моментов второго порядка:

$$\begin{aligned}
m_2^{(1)} &= \frac{1}{2\mu_1} \left\{ 2\lambda_1 m_1^{(1)}(1) - \lambda_1 R_1 - \mu_1 \left(m_1^{(1)}(1) + m_1^{(1)}(2) \right) \right\}, \\
m_2^{(2)} &= \frac{1}{2\mu_2} \left\{ 2\lambda_2 m_1^{(2)}(2) - \lambda_2 R_2 - \mu_2 \left(m_1^{(2)}(1) + m_1^{(2)}(2) \right) \right\}.
\end{aligned}$$

Таким образом, можно записать решение в общем виде для произвольного числа состояний внешней среды:

$$m_2^{(n)} = \frac{1}{2\mu_n} \left\{ 2\lambda_n m_1^{(n)}(n) - \lambda_n R_n - \mu_n \left(m_1^{(n)}(1) + m_1^{(n)}(2) + \dots + m_1^{(n)}(S) \right) \right\}, \quad n = \overline{1, S},$$

где S – число состояний.

Заключение

В данной работе рассмотрена математическая модель системы $MMPP(n)|M(n)|\infty$, функционирующей в условии изменяющейся внешней среды, методом моментов получены уравнения для нахождения моментов первого и второго порядков, исследован частный случай рассматриваемой системы для двух состояний изменяющейся внешней среды.

ЛИТЕРАТУРА

1. Дудин С.А., Дудина О.С. Многоканальная система обслуживания с марковским входным потоком нетерпеливых запросов, функционирующая в случайной среде // Информатика. Математическое моделирование. – 2015. – №1. – С. 45–55.
2. Моисеева С.П., Полин Е.П. Исследование математической модели бесконечнолинейной СМО с входящим МАР-потоком переменной интенсивности // Проблемы оптимизации сложных систем: сборник докладов Двенадцатой международной азиатской школы-семинара. Новосибирск, Академгородок, 12-16 декабря 2016 г. – Новосибирск, 2016. – С. 380–384.
3. Коротаев И.А., Спивак Л.Р. Системы массового обслуживания в полумарковской случайной среде // Автоматика и телемеханика. – 1992. – №7. – С. 86–92.
4. Назаров А.А., Терпугов А.Ф. Теория массового обслуживания. – Томск: Изд-во НТЛ, 2005. – 228 с.

ИССЛЕДОВАНИЕ СИСТЕМЫ $MMPP|GI|\infty$ МЕТОДОМ МОМЕНТОВ

С.А. Сеченова

Томский государственный университет
sechenovasofi@mail.ru

Введение

Системы массового обслуживания (СМО) с заявками случайного объёма позволяют решать задачи проектирования информационных систем, объектом преобразования в которых является информация, поступающая порциями в виде дискретных или непрерывных сообщений. Сообщения или заявки обладают различным информационным объёмом, который представляет собой случайную величину.

Задача исследования СМО, в которых каждая поступающая в систему заявка наряду со случайной длиной имеет случайный объём, причем суммарный объём всех находящихся в системе заявок ограничен, как было замечено еще в работах [1,8,9], играет

важную роль при моделировании работы самых разнообразных технических устройств, в частности – современных информационно-вычислительных систем.

Однако аналитических решений задачи, когда время обслуживания каждой заявки зависит от ее объема, для бесконечнолинейных СМО с произвольной функцией времени обслуживания до сих пор не найдено, поскольку для корректного построения соответствующего марковского процесса, описывающего функционирование СМО, необходимо учитывать объемы всех заявок в системе. В настоящей статье предлагается решать поставленную задачу методом просеянного потока позволяющим учитывать объемы заявок находящихся в системе в момент времени t и не учитывать покинувшие и найти моменты первого и второго порядка числа занятых приборов и полной суммы объемов требований.

1. Постановка задачи

Рассмотрим систему массового обслуживания, на вход которой поступает марковский модулированный поток (ММРП-поток), управляемый цепью Маркова $k(t)$, заданной матрицей инфинитезимальных характеристик $\mathbf{Q} = \|q_{ij}\|$ и диагональной матрицей условных интенсивностей $\mathbf{\Lambda}$. Считаем, что продолжительность обслуживания заявки имеет произвольную функцию распределения, одинаковую для всех приборов, которую обозначим $B(x)$. Будем предполагать, что каждое требование характеризуется некоторым случайным объемом $\sigma = \frac{\xi}{C} > 0$. Объемы различных требований – независимые случайные величины с функцией распределения $A(x) = P\{\sigma < x\} = P\{\xi < Cx\} = B(Cx)$. По окончании обслуживания заявка покидает систему и «уносит» свой объем [2–4].

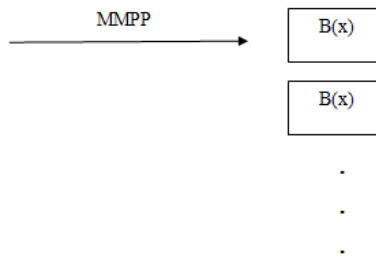


Рис. 1. Система массового обслуживания ММРП|GI|∞

Пусть $i(t)$ – число заявок, находящихся на обслуживании в системе в момент t , $V(t)$ – полная сумма объемов требований, находящихся в системе в момент времени t . Поставим задачу нахождения характеристик двумерного случайного процесса $\{i(t), V(t)\}$. Отметим, что исследуемый процесс не является марковским, поэтому для его исследования будем использовать метод динамического просеивания [5–7].

Построим просеянный поток для рассматриваемой системы массового обслуживания ММРП|GI|∞. Для этого зафиксируем некоторый момент времени T . Полагаем, что заявка входящего потока, поступившая в систему в момент времени $t < T$, с вероятностью $S(t) = 1 - B(T - t)$ формирует событие просеянного потока, а с вероятностью $1 - S(t)$ эта заявка не рассматривается.

Обозначим $n(t)$ – число событий просеянного потока, наступивших до момента времени t , $V(t)$ – суммарный объем просеянных заявок к моменту времени t . Тогда,

если в начальный момент $t_0 < T$ система была свободна, то для момента времени T для любых m и v выполняется равенство $P\{i(T) = m, V(T) < v\} = P\{n(T) = m, V(T) < v\}$.

Следует отметить, что использование метода динамического просеивания потока позволяет более точно определить характеристики процесса $V(t)$, т.к. в просеянном потоке присутствует только те заявки, которые не закончат обслуживание к моменту времени T .

2. Система дифференциальных уравнений Колмогорова

Введём обозначение $P(k, n, z, t) = P\{k(t) = k, n(t) = n, V(t) < z\}$ – распределение вероятностей трёхмерного марковского процесса, где $k(t)$ – состояние управляющей цепи Маркова в момент времени t , $n(t)$ – число событий, наступивших в просеянном потоке к моменту времени t , $V(t)$ – суммарный объём требований, находящихся в просеянном потоке к моменту времени t .

По формуле полной вероятности запишем равенство:

$$P(k, n, z, t + \Delta t) = P(k, n, z, t)(1 - \lambda_k \Delta t)(1 - q_{kk} \Delta t) + P(k, n, z, t) \lambda_k \Delta t (1 - S(t)) + \lambda_k S(t) \Delta t \int_0^z P(k, n - 1, z - y, t) dA(y) + \sum_{v \neq k} q_{vk} \Delta t P(v, n, z, t) + o(\Delta t), \quad (1)$$

$$k = \overline{1, K}, n = 0, 1, 2, \dots, z > 0,$$

при этом будем считать, что $P(n, 0, t) = 0$.

Из (1) получаем систему дифференциальных уравнений Колмогорова

$$\frac{\partial P(k, n, z, t)}{\partial t} = \lambda_k S(t) \left[\int_0^z P(k, n - 1, z - y, t) dA(y) - P(k, n, z, t) \right] + \sum_v q_{vk} \Delta t P(v, n, z, t),$$

$$k = \overline{1, K}, n = 0, 1, 2, \dots, z > 0.$$

Введём частичные характеристические функции вида:

$$H(k, u_1, u_2, t) = M \left\{ \exp(ju_1 n(t) + ju_2 V(t)) \right\} = \sum_{n=0}^{\infty} e^{ju_1 n} \int_0^{\infty} e^{ju_2 z} P(k, n, z, t) dz.$$

Учитывая, что

$$\begin{aligned} & \sum_{n=0}^{\infty} e^{ju_1 n} \int_0^{\infty} e^{ju_2 z} \int_0^z P(k, n - 1, z - y, t) dA(y) dz = \\ & = e^{ju_1} \sum_{n=0}^{\infty} e^{ju_1 (n-1)} \int_0^{\infty} e^{ju_2 y} \cdot e^{ju_2 (z-y)} \int_0^z P(k, n - 1, z - y, t) dA(y) dz = \\ & = e^{ju_1} \int_0^{\infty} e^{ju_2 y} \left[\sum_{n=0}^{\infty} e^{ju_1 (n-1)} \int_0^{\infty} e^{ju_2 (z-y)} P(k, n - 1, z - y, t) dz \right] dA(y) = \\ & = e^{ju_1} \int_0^{\infty} e^{ju_2 y} H(k, u_1, u_2, t) dA(y) = e^{ju_1} H(k, u_1, u_2, t) \int_0^{\infty} e^{ju_2 y} dA(y) = e^{ju_1} H(k, u_1, u_2, t) A^*(u_2), \end{aligned}$$

можно записать следующую систему уравнений:

$$\frac{\partial H(k, u_1, u_2, t)}{\partial t} = \lambda_k S(t) H(k, u_1, u_2, t) [e^{ju_1} A^*(u_2) - 1] + \sum_v q_{vk} H(k, u_1, u_2, t), \quad (2)$$

где $A^*(u_2) = M \left\{ \exp(ju_2 V(t)) \right\} = \int_0^\infty e^{ju_2 y} dA(y)$ – преобразование Фурье – Стилтеса функции $A(y)$.

Учитывая, что $A(y) = P\{\sigma < y\} = P\{\xi < Cy\} = B(Cy)$ и сделав замену $Cy = t$, перепишем систему (2) в виде:

$$\frac{\partial H(k, u_1, u_2, t)}{\partial t} = \lambda_k S(t) H(k, u_1, u_2, t) \left[e^{ju_1 B^* \left(\frac{u_2}{C} \right)} - 1 \right] + \sum_v q_{vk} H(k, u_1, u_2, t), \quad (3)$$

где $B^* \left(\frac{u_2}{C} \right) = \int_0^\infty e^{j \frac{u_2}{C} t} dB(t)$ – преобразование Фурье – Стилтеса функции $B(t)$.

Запишем систему (3) в матричном виде

$$\frac{\partial \mathbf{H}(u_1, u_2, t)}{\partial t} = \mathbf{H}(u_1, u_2, t) \left[\Lambda S(t) \left(e^{ju_1 B^* \left(\frac{u_2}{C} \right)} - 1 \right) + \mathbf{Q} \right] \quad (4)$$

с начальным условием $\mathbf{H}(u_1, u_2, t_0) = \mathbf{R}$, где

$$\mathbf{H}(u_1, u_2, t) = [H(1, u_1, u_2, t), H(2, u_1, u_2, t), \dots, H(K, u_1, u_2, t)],$$

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \lambda_K \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} q_{11} & q_{12} & \dots & q_{1K} \\ q_{21} & q_{22} & \dots & q_{2K} \\ \dots & \dots & \dots & \dots \\ q_{K1} & q_{K2} & \dots & q_{KK} \end{bmatrix}, \quad \mathbf{R} = [R(1), R(2), \dots, R(K)].$$

Здесь и далее \mathbf{R} – вектор-строка стационарного распределения управляющей цепи Маркова, определяемая системой линейных уравнений $\begin{cases} \mathbf{RQ} = 0, \\ \mathbf{RE} = 1, \end{cases}$ \mathbf{E} – единичный вектор-столбец.

Уравнение (4) позволяет находить не только характеристическую функцию двумерного процесса, но и другие характеристики рассматриваемой системы обслуживания, в частности, моменты любого порядка для числа занятых приборов и полной суммы объемов требований.

3. Нахождение моментов различных порядков числа занятых приборов и полной суммы объемов требований в системе ММРР|GI| ∞

Известно, что производная ν -го порядка в нуле от характеристической функции равна моменту ν -го порядка, умноженному на ν -ю степень мнимой единицы. Используя этот факт, найдём моменты числа занятых приборов и полной суммы объемов требований.

Моменты первого порядка

Для нахождения первого начального момента для процесса $i(t)$ продифференцируем по u_1 систему уравнений (4), получим

$$\frac{\partial^2 \mathbf{H}(u_1, u_2, t)}{\partial u_1 \partial t} = \frac{\partial \mathbf{H}(u_1, u_2, t)}{\partial u_1} \left[\Lambda S(t) \left(e^{ju_1 B^* \left(\frac{u_2}{C} \right)} - 1 \right) + \mathbf{Q} \right] + \mathbf{H}(u_1, u_2, t) \Lambda S(t) j e^{ju_1 B^* \left(\frac{u_2}{C} \right)}. \quad (5)$$

Обозначив $\left. \frac{\partial \mathbf{H}(u_1, u_2, t)}{\partial u_1} \right|_{u_1=u_2=0} = j\mathbf{m}_1(i, t)$, из (5) получим неоднородную систему

дифференциальных уравнений

$$\frac{\partial \mathbf{m}_1(i, t)}{\partial t} = \mathbf{m}_1(i, t)\mathbf{Q} + \mathbf{R}\Lambda S(t). \quad (6)$$

Решение системы (6) в стационарном режиме представляет собой первый начальный момент для числа занятых приборов, и имеет вид $m_1(i) = \kappa_1 b_1$, где $\kappa_1 = \mathbf{R}\Lambda \mathbf{E}$.

Для нахождения первого начального момента для полной суммы объемов требований продифференцируем по u_2 систему уравнений (4), получим

$$\begin{aligned} \frac{\partial^2 \mathbf{H}(u_1, u_2, t)}{\partial u_1 \partial u_2} &= \frac{\partial \mathbf{H}(u_1, u_2, t)}{\partial u_2} \left[\Lambda S(t) \left(e^{ju_1} B^* \left(\frac{u_2}{C} \right) - 1 \right) + \mathbf{Q} \right] + \\ &+ \mathbf{H}(u_1, u_2, t) \Lambda S(t) e^{ju_1} d \int_0^\infty j \frac{t}{C} e^{j \frac{u_2}{C} t} dB(t). \end{aligned} \quad (7)$$

Обозначив $\left. \frac{\partial \mathbf{H}(u_1, u_2, t)}{\partial u_2} \right|_{u_1=u_2=0} = j\mathbf{m}_1(V, t)$, из (7) получим неоднородную систему

дифференциальных уравнений

$$\frac{\partial \mathbf{m}_1(V, t)}{\partial t} = \mathbf{m}_1(V, t)\mathbf{Q} + \mathbf{R}\Lambda S(t) \frac{b_1}{C}. \quad (8)$$

Решение системы (8) в стационарном режиме представляет собой первый начальный момент для процесса $V(t)$ и имеет вид $m_1(V) = \kappa_1 \frac{b_1^2}{C}$.

Моменты второго порядка

Для нахождения момента второго порядка числа занятых приборов продифференцируем по u_1 систему (5), получим

$$\begin{aligned} \frac{\partial^3 \mathbf{H}(u_1, u_2, t)}{\partial u_1^2 \partial t} &= \frac{\partial^2 \mathbf{H}(u_1, u_2, t)}{\partial u_1^2} \left[\Lambda S(t) \left(e^{ju_1} B^* \left(\frac{u_2}{C} \right) - 1 \right) + \mathbf{Q} \right] + \\ &+ 2 \frac{\partial \mathbf{H}(u_1, u_2, t)}{\partial u_1} \Lambda S(t) j e^{ju_1} B^* \left(\frac{u_2}{C} \right) + \mathbf{H}(u_1, u_2, t) \Lambda S(t) j^2 e^{ju_1} B^* \left(\frac{u_2}{C} \right). \end{aligned} \quad (9)$$

Обозначив $\left. \frac{\partial^2 \mathbf{H}(u_1, u_2, t)}{\partial u_1^2} \right|_{u_1=u_2=0} = j^2 \mathbf{m}_2(i, t)$, из (9) получим неоднородную систему

линейных дифференциальных уравнений с постоянными коэффициентами

$$\frac{\partial \mathbf{m}_2(i, t)}{\partial t} = \mathbf{m}_2(i, t)\mathbf{Q} + 2\mathbf{m}_1(i, t)\Lambda S(t) + \mathbf{R}\Lambda S(t).$$

Суммируя все уравнения этой системы, и полагая $t=0$, получим для второго момента процесса $i(t)$ следующее равенство: $\mathbf{m}_2(i, t) = \kappa_1 b_1 + 2 \int_{t_0}^0 \mathbf{m}_1(i, t) S(t) dt \Lambda \mathbf{E}$.

Таким образом, для определения второго момента числа занятых приборов необходимо найти значение интеграла в последнем слагаемом, который определяется решением $\mathbf{m}_1(i, t)$ неоднородной системы (6) линейных дифференциальных уравнений с постоянными коэффициентами.

Для нахождения момента второго порядка полной суммы объёмов требований, находящихся в системе, продифференцируем по u_2 систему уравнений (7), получим

$$\begin{aligned} \frac{\partial^3 \mathbf{H}(u_1, u_2, t)}{\partial u_2^2 \partial t} &= \frac{\partial^2 \mathbf{H}(u_1, u_2, t)}{\partial u_2^2} \left[\Lambda S(t) \left(e^{j u_1} B^* \left(\frac{u_2}{C} \right) - 1 \right) + \mathbf{Q} \right] + \\ &+ 2 \frac{\partial \mathbf{H}(u_1, u_2, t)}{\partial u_2} \Lambda S(t) e^{j u_1} \int_0^\infty j \frac{t}{C} e^{j \frac{u_2}{C} t} dB(t) + \mathbf{H}(u_1, u_2, t) \Lambda S(t) e^{j u_1} \int_0^\infty j^2 \frac{t^2}{C^2} e^{j \frac{u_2}{C} t} dB(t). \end{aligned} \quad (10)$$

Обозначив $\left. \frac{\partial^2 \mathbf{H}(u_1, u_2, t)}{\partial u_2^2} \right|_{u_1=u_2=0} = j^2 \mathbf{m}_2(V, t)$, из (10) получим неоднородную систему

линейных дифференциальных уравнений с постоянными коэффициентами

$$\frac{\partial \mathbf{m}_2(V, t)}{\partial t} = \mathbf{m}_2(V, t) \mathbf{Q} + 2 \mathbf{m}_1(V, t) \Lambda S(t) \frac{b_1}{C} + \mathbf{R} \Lambda S(t) b_2.$$

Суммируя все уравнения этой системы, и полагая $t=0$, получим для второго момента процесса $V(t)$ следующее равенство:

$$\mathbf{m}_2(V, t) = \kappa_1 b_1 b_2 + 2 \frac{b_1}{C} \int_{t_0}^0 \mathbf{m}_1(V, t) S(t) dt \Lambda \mathbf{E}.$$

Таким образом, для определения второго момента полной суммы объёмов требований необходимо найти значение интеграла в последнем слагаемом, который определяется решением $\mathbf{m}_1(V, t)$ неоднородной системы (8) линейных дифференциальных уравнений с постоянными коэффициентами.

Смешанный момент

Для нахождения смешанного момента первого порядка для числа заявок, находящихся в системе, и полной суммы объёмов требований продифференцируем сначала по u_1 , а затем по u_2 систему уравнений (4) и получим

$$\begin{aligned} \frac{\partial^3 \mathbf{H}(u_1, u_2, t)}{\partial u_1 \partial u_2 \partial t} &= \frac{\partial^2 \mathbf{H}(u_1, u_2, t)}{\partial u_1 \partial u_2} \left[\Lambda S(t) \left(e^{j u_1} B^* \left(\frac{u_2}{C} \right) - 1 \right) + \mathbf{Q} \right] + \\ &+ \frac{\partial \mathbf{H}(u_1, u_2, t)}{\partial u_1} \Lambda S(t) e^{j u_1} \int_0^\infty j \frac{t}{C} e^{j \frac{u_2}{C} t} dB(t) + \frac{\partial \mathbf{H}(u_1, u_2, t)}{\partial u_2} \Lambda S(t) j e^{j u_1} B^* \left(\frac{u_2}{C} \right) + \\ &+ \mathbf{H}(u_1, u_2, t) \Lambda S(t) j^2 e^{j u_1} \int_0^\infty \frac{t}{C} e^{j \frac{u_2}{C} t} dB(t). \end{aligned} \quad (11)$$

Обозначив $\left. \frac{\partial^2 \mathbf{H}(u_1, u_2, t)}{\partial u_1 \partial u_2} \right|_{u_1=u_2=0} = j^2 \mathbf{m}_1(i, V, t)$, из (11) получим неоднородную систему

линейных дифференциальных уравнений с постоянными коэффициентами

$$\frac{\partial \mathbf{m}_1(i, V, t)}{\partial t} = \mathbf{m}_1(i, V, t) \mathbf{Q} + \mathbf{m}_1(i, t) \Lambda S(t) \frac{b_1}{C} + \mathbf{m}_1(V, t) \Lambda S(t) + \mathbf{R} \Lambda S(t) \frac{b_1}{C}.$$

Суммируя все уравнения этой системы, и полагая $t=0$, получим для процессов $i(t)$ и $V(t)$ следующее равенство:

$$\mathbf{m}_1(i, V, t) = \kappa_1 \frac{b_1^2}{C} + \frac{b_1}{C} \int_{t_0}^0 \mathbf{m}_1(i, t) S(t) dt \Lambda \mathbf{E} + \int_{t_0}^0 \mathbf{m}_1(V, t) S(t) dt \Lambda \mathbf{E}.$$

Таким образом, для определения смешанного момента числа занятых приборов и полной суммы объёмов требований необходимо найти значения интегралов, которые

определяются решением $m_1(i, t)$ и $m_1(V, t)$ неоднородных систем линейных дифференциальных уравнений с постоянными коэффициентами (6) и (8) соответственно.

Заключение

В настоящей работе рассмотрена математическая модель системы $M|G|I|_{\infty}$ с пропорциональным объёмом времени обслуживания, с помощью метода моментов получены начальные моменты первого и второго порядков, смешанный момент для числа занятых приборов и полной суммы объёмов требований в рассматриваемой системе.

ЛИТЕРАТУРА

1. Александров А.М., Кац Б.А. Обслуживание потоков неоднородных требований // Изв. АН СССР. Технич. Кибернетика. – 1973. – №2. – С. 47–53.
2. Лисовская Е.Ю. Характеристическая функция распределения вероятностей суммарного объёма заявок в системе $M|G|I|_{\infty}$ // Материалы 54-й Международной научной студенческой конференции МНСК-2016: Математика / Новосиб. гос. ун-т. Новосибирск, 2016. – с. 108.
3. Лисовская Е.Ю., Моисеева С.П. Исследование суммарного объёма требований в бесконечнолинейной системе массового обслуживания вида $M|G|I|_{\infty}$ // Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем: материалы Всероссийской конференции с международным участием. Москва, РУДН, 18-22 апреля 2016 г. – Москва: РУДН, 2016. – С. 28–30.
4. Лисовская Е.Ю., Моисеева С.П. Асимптотический анализ системы $M|G|I|_{\infty}$ с обслуживанием требований случайного объёма // – Математическое и программное обеспечение информационных, технических и экономических систем: материалы IV Международной молодежной научной конференции. Томск, ТГУ, 20-21 мая 2016 г. – Томск: ТГУ, 2016. – С. 99–104.
5. Моисеев А.Н., Назаров А.А. Бесконечнолинейные системы и сети массового обслуживания. – Томск: Изд-во НТЛ, 2015. – 240 с.
6. Назаров А.А., Тертугов А.Ф. Теория массового обслуживания: учебное пособие. 2-е изд., испр. – Томск: Изд-во НТЛ, 2010. – 228 с.
7. Назаров А.А., Моисеева С.П. Методы асимптотического анализа в теории массового обслуживания. – Томск: Изд-во НТЛ, 2006. – 112 с.
8. Ромм Э.Л., Скитович В.В. Об одном обобщении задачи Эрланга // Автоматика и телемеханика. – 1971. – № 6. – С. 164–167.
9. Тихоненко О.М. Моделирование процессов и систем обработки информации : курс лекций. – Минск: БГУ, 2008. – 148 с.

К АНАЛИЗУ МОДЕЛИ СХЕМЫ БЕСПРОВОДНОЙ СЕТИ LTE С СИСТЕМОЙ LSA В ВИДЕ СМО С ОТКАЗАМИ И ВОССТАНОВЛЕНИЯМИ*

Д.В. Иванова, Е.А. Карнаухова, Е.В. Маркова

Российский университет дружбы народов

daria.i1996@gmail.com, ek.karnauhova@gmail.com, markova_ev@pfur.ru

Введение

В связи с развитием телекоммуникационных беспроводных сетей последующих поколений, в частности сетей на базе технологии LTE, повышается спрос на высокоскоростные услуги, а также увеличивается число пользователей мобильных сетей, что приводит к экспоненциальному росту объёма передаваемого в мобильных сетях трафика. По данным компании Cisco Systems [1], объём трафика, генерируемого в мобильных сетях, в период с 2016 г. по 2021 г. увеличится на 46% и к 2021 г. составит 48,3 экзабайт в месяц. В то же время повышаются требования к уровню качества обслуживания пользователей (Quality of Service, QoS) [2], что в совокупности со стремительным ростом объёмов трафика приводит к проблеме нехватки радиоресурсов [3]. Одним из возможных подходов к решению этой проблемы является привлечение дополнительных ресурсов, что может быть реализовано за счёт системы совместного использования радиочастотного спектра LSA (Licensed Shared Access) [3,4].

* Публикация подготовлена при финансовой поддержке РФФИ в рамках научного проекта № 18-37-00231

В данной работе проводится исследование одной из возможных моделей функционирования LSA-полосы, взятой мобильным оператором в аренду у владельца, имеющего абсолютный приоритет в использовании своих радиочастот. При необходимости возвращения арендованных частот владельцу, так называемом отказе LSA-полосы, обслуживание пользователей арендатора не прерывается, а переходит в режим ожидания до момента восстановления полосы.

1. Описание модели

Проведём построение модели в терминах теории массового обслуживания [5]. Рассмотрим систему массового обслуживания (СМО), состоящую из C ненадежных приборов, способных одновременно отказывать с интенсивностью α или восстанавливаться с интенсивностью β [6], и очереди длины r [7], причём будем считать, что $r > C$ (рис. 1). Предположим, что находящиеся на приборах заявки сохраняют за собой место в очереди. Пусть входящий поток заявок имеет пуассоновское распределение с параметром λ , а среднее время обслуживания μ^{-1} распределено по экспоненциальному закону, тогда предложенная нагрузка определяется по формуле $\rho = \lambda / \mu$.

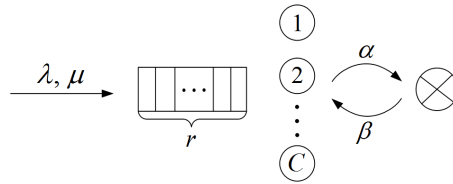


Рис. 1. Схема СМО с отказами, восстановлениями и конечной очередью

Состояние системы описывает двумерный вектор (n, s) , где $n = \{0, 1, \dots, r\}$ – количество заявок в системе; $s \in \{0, 1\}$ – состояние ненадежных приборов (0 – приборы недоступны, 1 – доступны). Тогда пространство состояний системы будет иметь вид

$$X = \{(n, s) : n = 0, \dots, r; s = 0 \vee n = 0, \dots, r; s = 1\}.$$

2. Правила приема и обслуживания заявок

При поступлении новой заявки на обслуживание возможны следующие варианты:

- заявка поступает на обслуживание, занимает прибор и сохраняет за собой место в очереди, если в системе имеются свободные приборы, находящиеся в доступном состоянии;
- заявка поступает в очередь, если в системе нет свободных приборов, находящихся в доступном состоянии, или приборы находятся в недоступном состоянии, а в очереди есть свободные места;
- заявка блокируется, если в очереди нет свободных мест.

На рис. 2 представлена диаграмма интенсивностей переходов для данной системы.

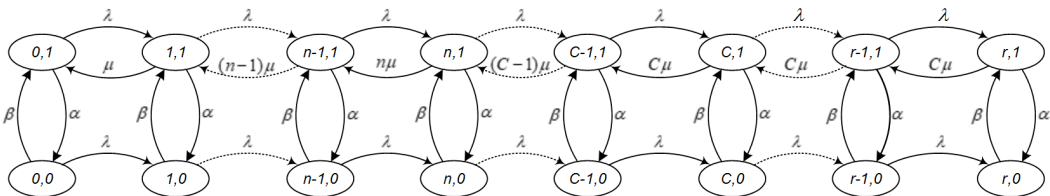


Рис. 2. Диаграмма интенсивностей переходов

3. Матрица интенсивностей переходов

Запишем систему уравнений глобального баланса (СУГБ):

$$\begin{aligned} & [\lambda \cdot I(n < r) + n\mu \cdot I(n < C, s = 1) + C\mu \cdot I(n \geq C, s = 1) + \alpha \cdot I(s = 1) + \beta \cdot I(s = 0)] \cdot p(n, s) = \\ & = \lambda \cdot I(n > 0) p(n-1, s) + \alpha \cdot I(s = 0) p(n, 1) + \\ & + \beta \cdot I(s = 1) p(n, 0) + (n+1)\mu \cdot I(n < C, s = 1) p(n+1, 1) + \\ & + C\mu \cdot I(C \leq n \leq r-1, s = 1) p(n+1, 1), \quad n = \overline{0, r}, \quad s = \overline{0, 1}, \end{aligned}$$

где $p(n, s)$, $(n, s) \in \mathbf{X}$ – стационарное распределение вероятностей.

Стационарное распределение вероятностей может быть получено путём численного решения СУР $\mathbf{p}^T \cdot \mathbf{A} = \mathbf{0}^T$, где \mathbf{A} – инфинитезимальная матрица, представимая в блочном трёхдиагональном виде:

$$\mathbf{A} = \begin{bmatrix} \mathbf{N}_0 & \mathbf{\Lambda}_0 & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{M}_1 & \mathbf{N}_1 & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{N}_{r-1} & \mathbf{\Lambda}_{r-1} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{M}_r & \mathbf{N}_r \end{bmatrix},$$

где \mathbf{N}_n , \mathbf{M}_n , $\mathbf{\Lambda}_n$ – подматрицы, а n – количество заявок в системе. Рассмотрим блоки матрицы отдельно.

Подматрицы интенсивностей поступления описываются следующим образом:

$$\mathbf{\Lambda}_n = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}, \quad n = \overline{0, r-1}.$$

Подматрицы интенсивностей обслуживания описываются следующим образом:

$$\mathbf{M}_n = \begin{bmatrix} 0 & 0 \\ 0 & n\mu \end{bmatrix}, \quad n = \overline{1, C},$$

$$\mathbf{M}_n = \begin{bmatrix} 0 & 0 \\ 0 & C\mu \end{bmatrix}, \quad n = \overline{C+1, r}.$$

Диагональные подматрицы описываются следующим образом:

$$\mathbf{N}_n = \begin{bmatrix} -(\beta + \lambda) & \beta \\ \alpha & -(n\mu + \alpha + \lambda) \end{bmatrix}, \quad n = \overline{0, C},$$

$$\mathbf{N}_n = \begin{bmatrix} -(\beta + \lambda) & \beta \\ \alpha & -(C\mu + \alpha + \lambda) \end{bmatrix}, \quad n = \overline{C+1, r-1},$$

$$\mathbf{N}_n = \begin{bmatrix} -\beta & \beta \\ \alpha & -(C\mu + \alpha) \end{bmatrix}, \quad n = r.$$

4. Рекуррентный алгоритм

Помимо матричного решения, для расчёта стационарного распределения вероятностей $p(n, s)$, $(n, s) \in \mathbf{X}$ может быть разработан рекуррентный алгоритм. Обозначим $q(n, s)$ ненормированную вероятность того, что число заявок в системе равно n , а приборы находятся в состоянии s . Тогда ненормированные вероятности вычисляются по формулам

$$q(0, 0) = 1,$$

$$q(0,1) = x,$$

$$q(n,s) = \gamma_{ns} + \delta_{ns}x, (n,s) \in \mathbf{X}; n > 0.$$

Коэффициенты γ_{ns} и δ_{ns} вычисляются по рекуррентным соотношениям

$$\gamma_{00} = 1, \delta_{00} = 0, \gamma_{01} = 0, \delta_{01} = 1, \gamma_{10} = \frac{\lambda\mu - \alpha\beta}{(\beta + \lambda)\mu}, \delta_{10} = \frac{\alpha}{\beta + \lambda} \cdot \frac{\alpha + \lambda}{\mu}, \gamma_{11} = -\frac{\beta}{\mu}, \delta_{11} = \frac{\alpha + \lambda}{\mu},$$

$$\gamma_{n1} = \frac{1}{n} \left[(n-1 + \delta_{11})\gamma_{n-1,1} - \rho\gamma_{n-2,1} + \gamma_{11}\gamma_{n-1,0} \right], n = \overline{2, C-1},$$

$$\delta_{n1} = \frac{1}{n} \left[(n-1 + \delta_{11})\delta_{n-1,1} - \rho\delta_{n-2,1} + \gamma_{11}\delta_{n-1,0} \right], n = \overline{2, C-1},$$

$$\gamma_{n0} = \frac{\lambda\gamma_{n-1,0} + \alpha\gamma_{n1}}{\beta + \lambda}, \delta_{n0} = \frac{\lambda\delta_{n-1,0} + \alpha\delta_{n1}}{\beta + \lambda}, n = \overline{2, r-1},$$

$$\gamma_{n1} = \frac{1}{C} \left[(C + \delta_{11})\gamma_{n-1,1} - \rho\gamma_{n-2,1} + \gamma_{11}\gamma_{n-1,0} \right], n = \overline{C, r},$$

$$\delta_{n1} = \frac{1}{C} \left[(C + \delta_{11})\delta_{n-1,1} - \rho\delta_{n-2,1} + \gamma_{11}\delta_{n-1,0} \right], n = \overline{C, r},$$

$$\gamma_{r0} = \frac{\lambda\gamma_{r-1,0} + \alpha\gamma_{r1}}{\beta}, \delta_{r0} = \frac{\lambda\delta_{r-1,0} + \alpha\delta_{r1}}{\beta}.$$

Для нахождения x воспользуемся уравнением глобального баланса для состояния $(r,1)$, записав его с использованием коэффициентов γ_{ns} и δ_{ns} :

$$(\alpha + C\mu)p(r,1) = \beta p(r,0) + \lambda p(r-1,1),$$

$$(\alpha + C\mu)(\gamma_{r1} + \delta_{r1} \cdot x) = \beta(\gamma_{r0} + \delta_{r0} \cdot x) + \lambda(\gamma_{r-1,1} + \delta_{r-1,1} \cdot x),$$

$$((\alpha + C\mu)\delta_{r1} - \beta\delta_{r0} - \lambda\delta_{r-1,1}) \cdot x = \beta\gamma_{r0} + \lambda\gamma_{r-1,1} - (\alpha + C\mu)\gamma_{r1},$$

$$x = \frac{\beta\gamma_{r0} + \lambda\gamma_{r-1,1} - (\alpha + C\mu)\gamma_{r1}}{(\alpha + C\mu)\delta_{r1} - \beta\delta_{r0} - \lambda\delta_{r-1,1}}.$$

Стационарное распределение вероятностей определяется по формуле

$$p(n,s) = \frac{q(n,s)}{\sum_{(i,j) \in \mathbf{X}} q(i,j)}, (n,s) \in \mathbf{X}.$$

5. Показатели эффективности

Основными характеристиками модели являются:

- вероятность блокировки $B = \sum_{s=0}^1 p(r,s) = p(r,0) + p(r,1)$;
- средняя длина очереди $Q = \sum_{n=1}^r np(n,0) + \sum_{n=C+1}^r (n-C)p(n,1)$;
- средняя длина очереди при доступной полосе $Q_{\text{дост.}} = \sum_{n=C+1}^r (n-C)p(n,1)$;
- средняя длина очереди при недоступной полосе $Q_{\text{недост.}} = \sum_{n=1}^r np(n,0)$.

Заключение

В настоящей работе рассмотрена модель схемы распределения радиоресурсов в виде СМО с ненадёжными приборами и конечной очередью, разработан рекуррентный алгоритм для расчёта стационарного распределения вероятностей, получены формулы для расчёта основных характеристик модели.

ЛИТЕРАТУРА

1. Cisco Visual Networking Index: Forecast and Methodology, 2017.
2. Тихвинский В.О., Бочечка Г.С. Перспективы сетей 5G и требования к качеству их обслуживания // Электросвязь. – 2014. – №11. – С. 40–43.
3. Gundlach M., Hofmann J., Markwart C., Mohyeldin E. Recent advances on LSA in standardization, regulation, research and architecture design // Proc. of the 1st International Workshop on Cognitive Cellular Systems CSS 2014. – IEEE. – 2014. – P. 1–5.
4. Ponomarenko-Timofeev A., Pyattaev A., Andreev S., Koucheryavy Ye., Mueck M., Karls I. Highly dynamic spectrum management within licensed shared access regulatory framework // IEEE Communications Magazine. – 2015. – Vol. 54, No. 3. – P. 100–109.
5. Башарин Г.П. Лекции по математической теории телеграфика. – М.: РУДН, 2009. – 341 с.
6. Mokrov E., Sopin E., Markova E., Poluektov D., Gudkova I., Shorgin S., Hosek J., Masek P., Samouylov K. Modelling And Response Time Analysis For Web Browsing Under Interruptions In LTE Network. // Proc. of the 31th European Conference on Modelling and Simulation ECMS-2017. – 2017. – P. 706–712. DOI: <https://doi.org/10.7148/2017-0706>
7. Gudkova I., Korotysheva A., Zeifman A.I., Shilova G., Korolev V., Shorgin S., Razumchik R. Modeling and analyzing licensed shared access operation for 5G network as an inhomogeneous queue with catastrophes. // Proc. of the 8th International Congress on Ultra Modern Telecommunications and Control Systems ICUMT-2016. – 2016. – P. 336–341. DOI: <https://doi.org/10.1109/ICUMT.2016.7765372>

АНАЛИЗ ВЕРОЯТНОСТНЫХ ХАРАКТЕРИСТИК МОДЕЛИ СХЕМЫ ФУНКЦИОНИРОВАНИЯ ОДНОЙ LSA-ПОЛОСЫ С НЕТОЛЕРАНТНЫМ К ЗАДЕРЖКАМ ТРАФИКОМ*

Д.В. Иванова, Е.А. Карнаухова, Е.В. Маркова

Российский университет дружбы народов

daria.i1996@gmail.com, ek.karnauhova@gmail.com, markova_ev@pfur.ru

Введение

В настоящее время развитие телекоммуникационных беспроводных сетей на базе технологии LTE предлагает пользователям широкий спектр различных мультимедийных услуг. По прогнозам международной компании Cisco Systems, приведенным в отчете за 2017 г., к 2021 г. размер ежемесячного интернет-трафика достигнет 20 ГБ на душу населения, по сравнению с 10 ГБ на душу населения в 2016 г. [1]. В условиях стремительного роста, генерируемого различными приложениями интернет-трафика, необходимо искать решения, позволяющие более эффективно использовать имеющиеся радиоресурсы, например, за счёт их совместного использования несколькими сторонами. Данное решение было предложено Европейским институтом по стандартизации в области телекоммуникаций (ETSI), и называется системой совместного использования лицензированного спектра частот (Licensed Shared Access, LSA) [2,3]. Структура этой системы позволяет обеспечить ограниченному числу арендаторов контролируемый доступ к спектру частот владельца. В работе рассматривается одна из возможных моделей схем распределения радиоресурсов [4] в беспроводной сети передачи данных LTE с системой LSA. Исследуется только одна полоса частот – полоса совместного использования – частоты, взятые мобильным оператором в аренду у оператора-владельца. Управление ресурсами полосы построено так, что в случае её отказа, т.е. при необходимости возврата полосы оператору-владельцу, обслуживание пользователей не прерывается, а переходит в режим ожидания до момента восстановления полосы.

* Публикация подготовлена при финансовой поддержке РФФИ в рамках научного проекта № 18-37-00231

Далее проведём построение модели в терминах теории массового обслуживания [5]. Для этого опишем функционирование полосы совместного использования с помощью системы массового обслуживания (СМО) с ненадёжными приборами и конечной очередью.

1. Описание математической модели

Рассмотрим СМО, состоящую из C ненадёжных приборов и очереди длины r , причём $r > C$. Предположим, что находящиеся на приборах заявки сохраняют за собой место в очереди. Приборы могут одновременно отказывать (переходить в недоступное состояние) с интенсивностью α и восстанавливаться (переходить в доступное состояние) с интенсивностью β . При отказе приборов заявки, находящиеся на обслуживании, возвращаются в очередь. Входящий поток заявок имеет пуассоновское распределение с параметром λ . Среднее время обслуживания μ^{-1} распределено по экспоненциальному закону.

Пространство состояний системы имеет вид:

$$X = \{0, 1, \dots, 2r + 1\},$$

где $n = 0, 2, \dots, 2k, \dots, 2r$ – состояния, в которых приборы недоступны для обслуживания заявок, соответствующие отказу LSA-полосы, $n = 1, 3, \dots, 2k + 1, \dots, 2r + 1$ – состояния, в которых приборы доступны, соответствующие восстановлению LSA-полосы, $k = 0, 1, \dots, r$ – общее число заявок в системе.

Заявки, поступающие в систему, обслуживаются по следующим правилам:

- заявка поступает на обслуживание, занимает прибор и резервирует за собой место в очереди, если в системе имеются свободные приборы, находящиеся в доступном состоянии, т.е. $n = 2k + 1, k < C$;
- заявка поступает в очередь, если в системе нет свободных приборов, находящихся в доступном состоянии, или приборы находятся в недоступном состоянии, а в очереди есть свободные места, т.е. $n = 2k + 1, C \leq k < r$ или $n = 2k, k < r$;
- заявка будет заблокирована, если в очереди нет свободных мест, т.е. $n \geq r$.

Диаграмму интенсивностей переходов модели можно представить в следующем виде (рис. 1):

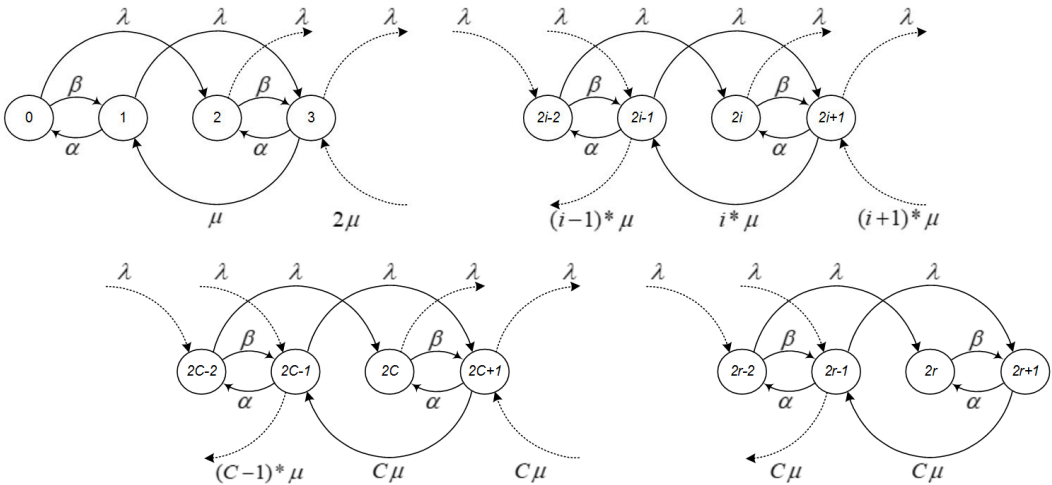


Рис. 1. Диаграмма интенсивностей переходов в общем виде

С учётом сформулированных выше правил приема и обслуживания заявок элементы матрицы интенсивностей переходов определяются следующим образом:

$$a_{n'} = \begin{cases} \alpha, & \text{если } n' = n-1, n = 2k+1, k = 0, \dots, r, \\ \beta, & \text{если } n' = n+1, n = 2k, \\ \lambda, & \text{если } n' = n+2, n < 2r, \\ \frac{n-1}{2} \mu, & \text{если } n' = n-2, n = 2k+1, k = 1, \dots, C, \\ C\mu, & \text{если } n' = n-2, n = 2k+1, k = C+1, \dots, r, \\ *, & \text{если } n' = n, \\ 0, & \text{в противном случае,} \end{cases}$$

$$\text{где } * = - \left(\frac{n-1}{2} \mu \cdot I(n = 2k+1, k = 1, \dots, C) + C\mu \cdot I(n = 2k+1, k = C+1, \dots, r) + \right. \\ \left. + \alpha \cdot I(n = 2k+1) + \beta \cdot I(n = 2k) + \lambda \cdot I(n < 2r) \right).$$

В таком случае систему уравнений глобального баланса можно представить в следующем виде:

$$\begin{aligned} & [\lambda \cdot I(n > 2r) + \frac{n-1}{2} \mu \cdot I(n = 2k+1, k = 1, \dots, C) + \alpha \cdot I(n = 2k+1) + \\ & + C\mu \cdot I(n = 2k+1, k = C+1, \dots, r) + \beta \cdot I(n = 2k)] \cdot p_n = \\ & = \lambda \cdot I(n > 1) p_{n-2} + \frac{n-1}{2} \mu \cdot I(n = 2k+1, k = 0, \dots, C-1) p_{n+2} + \\ & + \alpha \cdot I(n = 2k) p_{n+1} + C\mu \cdot I(n = 2k+1, k = C, \dots, r) p_{n+2} + \\ & + \beta \cdot I(n = 2k+1) p_{n-1}, k = \overline{0, r}, \end{aligned}$$

где $p_n, n \in X$ – стационарное распределение вероятностей системы.

2. Численный анализ вероятностных характеристик модели

Основными характеристиками модели являются вероятность блокировки (1) и среднее число заявок в очереди (2), а также среднее число заявок в очереди, когда LSA-полоса доступна (3) или недоступна (4):

$$B = \sum_{i=0}^1 p_{2r+i} = p_{2r} + p_{2r+1}, \quad (1)$$

$$Q = \sum_{k=1}^r k p_{2k} + \sum_{k=C+1}^r (k-C) p_{2k+1}, \quad (2)$$

$$Q_{\text{дост.}} = \sum_{k=C+1}^r (k-C) p_{2k+1}, \quad (3)$$

$$Q_{\text{недост.}} = \sum_{k=1}^r k p_{2k}. \quad (4)$$

Одним из ключевых факторов, влияющих на стремительный рост трафика данных, является увеличение числа M2M (Machine to Machine) приложений. Это тесно связано с развитием автоматизированных технологий в рамках реализации проекта «умный город». Данная технология является примером коммуникационной парадигмы M2M соединений в сетях пятого поколения 5G [6]. В связи с тем, что большинство приложений «умного города» являются машиноориентированными, взаимосвязанные датчики могут стать неотъемлемой частью этой среды.

Для проведения численного анализа рассмотрим городской случай использования системы LSA, при условии, что владелец совместного спектра частот не часто использует свои частотные ресурсы, примерно каждые 20-30 минут. В связи с этим будем счи-

тять, что LSA полоса становится недоступной для оператора каждые 20 или 30 минут, т.е. $\alpha^{-1} = 1200$ или 1800 сек. При этом у оператора есть возможность запрашивать неиспользуемый спектр каждые 20 секунд, т.е. $\beta^{-1} = 20$ сек. Число запросов, поступающих в секунду от датчиков на передачу данных, будем считать меняющимся в пределах от 2 до 20. Предположим, что одна сота беспроводной сети может одновременно обслужить не более 30 датчиков, а среднее время обслуживания при этом составляет 0,1 сек. Сведём все перечисленные исходные данные в табл. 1.

Таблица 1

Исходные данные

Обозначение	Значение
C	30
r	50
α^{-1}	1200, 1800 сек.
β^{-1}	20 сек.
λ	2-20
μ^{-1}	0,1 сек.

Рассмотрим зависимость вероятности блокировки (1) и средней длины очереди (2) от интенсивности поступления запросов на передачу данных – рис. 2 и рис. 3 соответственно.

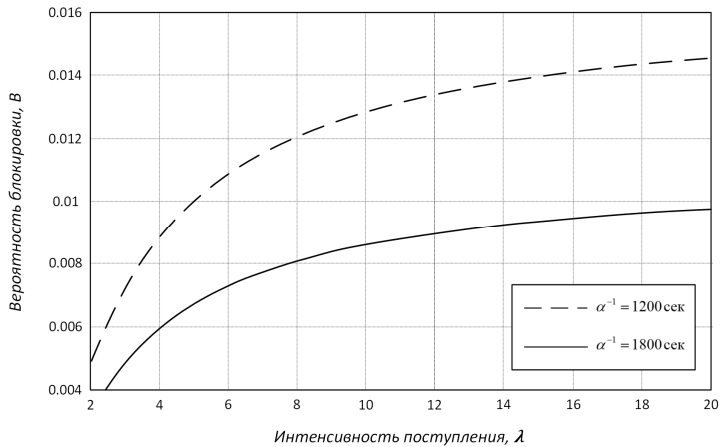


Рис. 2. Зависимость вероятности блокировки от интенсивности поступления

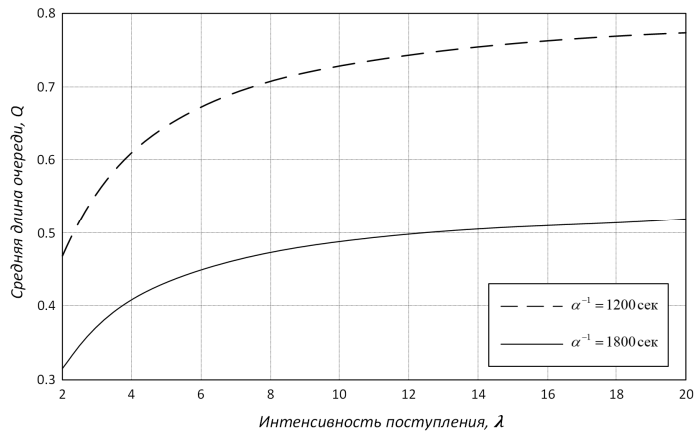


Рис. 3. Зависимость средней длины очереди от интенсивности поступления

Согласно полученным результатам (рис. 2 и рис. 3) можно сделать вывод, что чем реже LSA-полоса переходит в недоступный режим, тем ниже вероятность блокировки запросов на передачу данных и меньше длина очереди.

Заключение

В ходе проделанной работы была рассмотрена модель одной LSA-полосы для нетолерантного (потокowego) к задержкам трафика. Для исследования модели была построена диаграмма интенсивностей переходов, получены формулы для расчёта основных вероятностных характеристик системы – вероятности блокировки, средней длины очереди, а также средней длины очереди при доступной и недоступной LSA-полосе. Описание модели с помощью одномерного графа позволит в дальнейшем провести исследование зависимости рассмотренного случайного процесса от времени, тем самым, максимально приблизить исследуемую модель к реальным условиям.

ЛИТЕРАТУРА

1. Cisco. The Zettabyte Era: Trends and Analysis, 2017
2. Ponomarenko-Timofeev A., Pyataev A., Andreev S., Koucheryavy Ye., Mueck M., Karls I. Highly dynamic spectrum management within licensed shared access regulatory framework // IEEE Communications Magazine. – 2015. – Vol. 54, No. 3. – P. 100–109.
3. ETSI TS 103 235: System architecture and high level procedures for operation of Licensed Shared Access (LSA) in the 2 300 MHz – 2 400 MHz band. – ETSI. – 2015.
4. Seo H., Lee K.-D., Yasukawa S., Peng Y., Sartori P. LTE Evolution for Vehicle-to-Everything Services. // IEEE Communications Magazine. – 2016. – Vol. 54, No. 6. – P. 22–28.
5. Башарин Г.П. Лекции по математической теории телетрафика: Учеб. пособие. Изд. 3-е, испр. и доп. – М.: РУДН, 2009. – 342 с.
6. Laya A., Alonso L., Alonso-Zarate J., Dohler M. Green MTC, M2M, Internet of Things. // Green Communications: Principles, Concepts and Practice. – 2015. – P. 217–236.

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ЗВЕНА ДОСТУПА В СЕТЯХ СВЯЗИ МИЛЛИМЕТРОВОГО ДИАПАЗОНА ДЛЯ ОДНОАДРЕСНЫХ И МНОГОАДРЕСНЫХ СОЕДИНЕНИЙ

О.С. Крупко, Ф.А. Москалева

Российский университет дружбы народов
alesiakrupko@yandex.ru, faina3586@mail.ru

Введение

В последнее время беспроводные сети миллиметрового диапазона частот (30 ГГц – 300 ГГц) привлекли внимание научного сообщества благодаря возможности поддерживать высокую скорость передачи данных. Высокая скорость делает возможным использование таких сетей как для предоставления современных широкополосных услуг связи, например, видео по запросу или транслирование несжатого видео, так и для услуг будущего поколения, например, услуг дополненной реальности.

Физические свойства волн миллиметрового диапазона частот несут в себе как достоинства, так и недостатки. Среди достоинств – высокая направленность. Поскольку передающие лучи генерируются в определённых направлениях, снижаются потери энергии, а проблемы конфиденциальности и безопасности стоят не так остро. Также снижается уровень интерференции, что позволяет повысить пропускную способность радиоканалов. К недостаткам следует отнести высокие потери энергии при распространении в среде и возможность блокировки объектами из-за слабой дифракционной способности волн. Например, любое препятствие, такое, как человек или предмет мебели, может значительно ухудшить связь.

Анализ литературы показывает, что предыдущие работы по сетям миллиметрового диапазона были сосредоточены на оптимизации передачи данных одноадресных соеди-

нений. Многоадресная передача на основе миллиметровых волн слабо исследована как в отечественной, так и в зарубежной литературе [1,2]. В данной работе предлагается модель процесса обслуживания соединений услуг мультивещания и одноадресных соединений звеном мультисервисной сети.

1. Основная часть

Модель предполагает наличие соединений двух типов – одноадресное (unicast) и многоадресное (multicast). Входящие параметры модели: интенсивности поступления заявок для одноадресного (λ_u) и многоадресного (λ_m) соединений, интенсивности обслуживания заявок для одноадресного (μ_u) и многоадресного (μ_m) соединений, а также требование к ресурсам – постоянная величина d для одноадресного соединения и случайная величина ξ для многоадресного соединения, а также общий объём ресурса C . При построении модели мы используем терминологию ресурсных систем массового обслуживания [3]. В терминологии классических систем массового обслуживания под ресурсом понимаются приборы, а также места в очереди при её наличии [4]. Закон распределения требований к ресурсам для заявок мультивещания определяется с учётом случайного положения пользователей в зоне обслуживания и возможной блокировки пути распространения сигнала. Время обслуживания заявок одноадресных и многоадресных соединений распределено по экспоненциальному закону. Предполагается, что все многоадресные заявки, принятые в систему с уже находящимися на обслуживании многоадресными заявками, покинут систему вместе с первой многоадресной заявкой, принятой на обслуживание, т.е. время обслуживания запросов многоадресного соединения определяется временем обслуживания первой поступившей заявки. Последующие заявки на многоадресное соединение могут увеличить объём ресурса для обслуживания уже установленного многоадресного соединения, если требование к объёму для вновь поступившей заявки больше уже выделенного для обслуживания многоадресных заявок объёма ресурса. Если в момент поступления заявки одного из типов соединений в системе нет достаточного количества свободного ресурса для её обслуживания, заявка считается потерянной.

Описываемая модель схематически представлена на рис. 1.

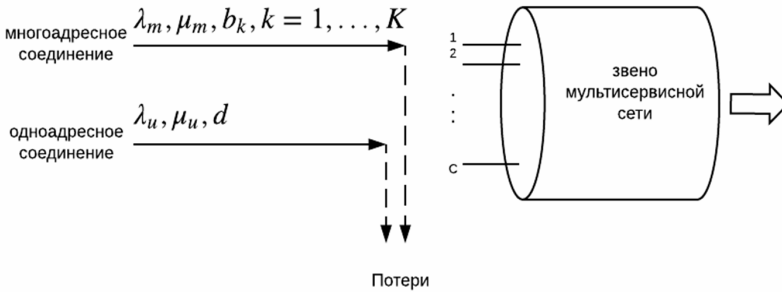


Рис. 1. Модель звена мультисервисной сети с двумя типами трафика

Согласно используемым предположениям о модели распространения радиосигнала, объём ресурса, требуемого для обслуживания многоадресных заявок, растёт по мере удаления пользователя от базовой станции. Мы предлагаем разделить пользователей на K классов (рис. 2) относительно количества ресурса, требуемого для обслуживания многоадресной заявки.

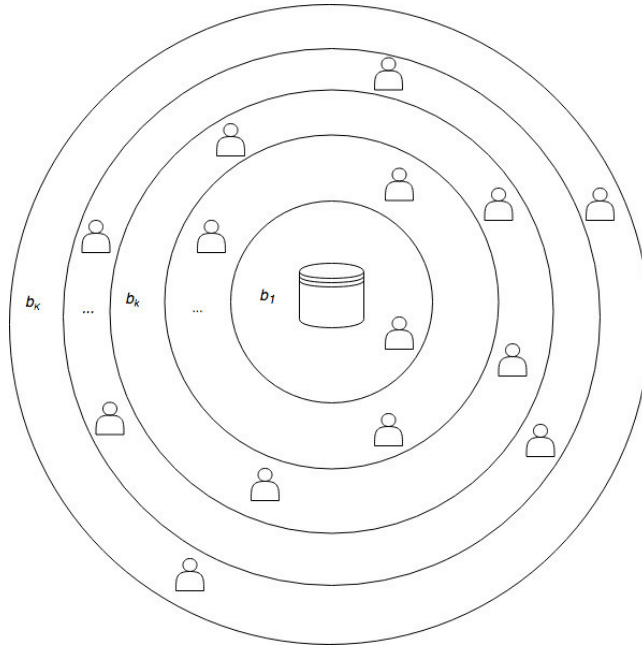


Рис. 2. Схема разделения пользователей на классы в зависимости от требования к ресурсам

Объём запрашиваемых заявкой мультивещания ресурсов является дискретной случайной величиной ξ с рядом распределения $\phi_k = P\{\xi = b_k\}$, $k = \overline{1, K}$, $\sum_{k=1}^K \phi_k = 1$.

Функционирование системы описывает марковский процесс $X(t)$ с пространством состояний

$$X = \{(n, r) : 0 \leq n \leq [C/d]; 0 \leq r \leq C\},$$

где n – число одноадресных заявок в системе, r – суммарный объём ресурса, занятого соединениями обоих типов.

Построенное таким образом пространство состояний системы позволяет получить все необходимые для дальнейших исследований характеристики, а именно – количество одноадресных заявок в системе, объём ресурса, занятого мультивещанием и одноадресными соединениями, взятыми в отдельности.

Фрагмент графа интенсивностей переходов представлен на рис. 3.

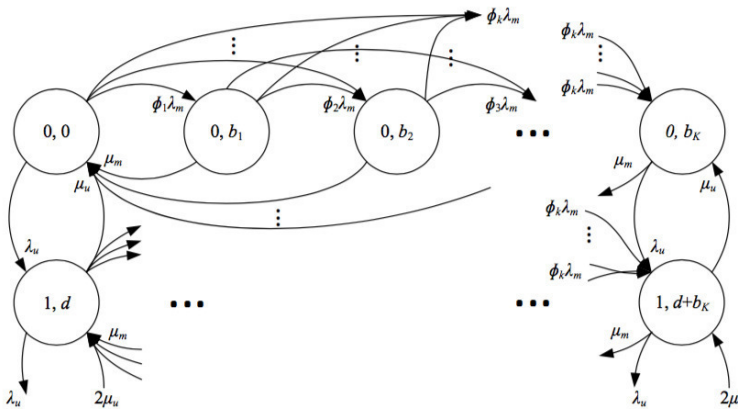


Рис. 3. Фрагмент графа интенсивностей переходов

На основе построенного графа интенсивностей переходов мы вывели систему уравнений глобального баланса.

$$\begin{aligned}
 (\lambda_u + \lambda_m) p_{0,0} &= \mu_u p_{1,d} + \mu_m \sum_{i=1}^K p_{0,b_i}, \quad n=0, \quad k=0, \\
 \left(\lambda_u + \lambda_m \sum_{i=k+1}^K \phi_i + \mu_m \right) p_{0,b_k} &= \phi_k \lambda_m \sum_{i=1}^{k-1} p_{0,b_i} + \phi_k \lambda_m p_{0,0} + \mu_u p_{1,d+b_k}, \quad n=0, \quad k=\overline{1, K-1}, \\
 (\lambda_u + \mu_m) p_{0,b_K} &= \mu_u p_{1,d+b_K} + \phi_K \lambda_m \sum_{i=1}^{K-1} p_{0,b_i} + \phi_K \lambda_m p_{0,0}, \quad n=0, \quad k=K, \\
 (\lambda_u + \lambda_m + n\mu_u) p_{n,nd} &= (1+n)\mu_u p_{n+1,d(n+1)} + \mu_m \sum_{i=1}^K p_{n,nd+b_i} + \lambda_u p_{n-1,d(n-1)}, \quad n=1, \quad \left[\frac{C}{d} \right], \quad k=0, \\
 (\lambda_u + \lambda_m + n\mu_u) p_{n,nd+b_K} &= \phi_K \lambda_m \sum_{i=1}^{K-1} p_{n,nd+b_i} + \phi_K \lambda_m p_{n,nd} + (1+n)\mu_u p_{n+1,d(n+1)+b_K} + \\
 &\quad + \lambda_u p_{n-1,d(n-1)+b_K}, \quad n=1, \quad \left[\frac{C-b_K}{d} \right], \quad k=K, \\
 \left(\lambda_u + \lambda_m \sum_{i=k+1}^K \phi_i + n\mu_u + \mu_m \right) p_{n,nd+b_k} &= \phi_k \lambda_m \sum_{i=1}^{k-1} p_{n,nd+b_i} + \phi_k \lambda_m p_{n,nd} + (1+n)\mu_u p_{n+1,d(n+1)+b_k} + \\
 &\quad + \lambda_u p_{n-1,d(n-1)+b_k}, \quad n=1, \quad \left[\frac{C-b_K}{d} \right], \quad k=\overline{1, K}, \\
 n\mu_u p_{n,nd} &= \lambda_u p_{n-1,d(n-1)}, \quad n = \left[\frac{C}{d} \right], \\
 (n\mu_u + \mu_m) p_{n,nd+b_k} &= \phi_k \lambda_m \sum_{i=1}^{k-1} p_{n,nd+b_i} + \phi_k \lambda_m p_{n,nd} + \\
 &\quad + \lambda_u p_{n-1,d(n-1)+b_k}, \quad n = \left[\frac{C-b_K}{d} \right], \left[\frac{C}{d} \right], \quad b_k \leq C - nd \leq b_{k+1}.
 \end{aligned}$$

2. Расчёт вероятностных характеристик

Из системы уравнений глобального баланса мы можем получить стационарное распределение $\{p_{n,r}, (n,r) \in X\}$. Таким образом, каждому числу занятых ресурсов r соответствует набор вероятностей $p_{n,r}$ того, что в системе будет занято ровно r ресурсов, nd из которых будет занято одноадресными заявками, а $b_k = r - nd$ ресурсов будет занято соединением мультивещания, $0 \leq n \leq [C/d]$.

Среднее число занятых ресурсов в модели можно найти как математической ожидание $r^{(1)}$ случайной величины r , а именно

$$r^{(1)} = \sum_{r=0}^C r \cdot \sum_{n=0}^{[C/d]} p_{n,r}.$$

Величина $r^{(1)} / C$ представляет собой коэффициент использования звена.

Следует напомнить, что одноадресная заявка считается потерянной, если в системе недостаточно ресурсов для её обслуживания. Таким образом, множество потерь одноадресных заявок имеет вид

$$B_u = \{(n,r) \in X : r + d > C\}.$$

Заявка мультивещания класса k считается потерянной, если в системе в момент поступления данной заявки нет других заявок класса k и нет свободного ресурса объёма b_k . Множество потерь заявок мультивещания класса k имеет вид

$$B_m^k = \{(n, r) \in X : r - nd + b_k > C\}.$$

Значения вероятностей $B_u = P\{(n, r) \in B_u\}$ и $B_m^k = P\{(n, r) \in B_m^k\}$ можно получить из стационарного распределения вероятностей $\{p_{n,r}, (n, r) \in X\}$.

Вероятность потери заявки одноадресного соединения представима в виде

$$B_u = \sum_{(n,r) \in B_u} p_{n,r}.$$

Вероятность потери заявки мультивещания класса k имеет вид

$$B_m^k = \sum_{(n,r) \in B_m^k} p_{n,r}.$$

Заключение

В данной работе построена модель звена мультисервисной сети с двумя типами трафика в виде марковского процесса с двумя компонентами. Построен граф интенсивностей переходов, получена система уравнений глобального баланса, а также такие характеристики, как вероятности блокировки для одноадресных и многоадресных соединений в случае недостатка свободных ресурсов, среднее значение количества занятых ресурсов и коэффициент использования звена. Построенная модель объединяет известные подходы к моделированию систем с случайным объёмом ресурса [3] и моделированию обслуживания многоадресных и одноадресных соединений [4]. Задачей дальнейших исследований может стать рассмотрение более общих распределений требований к объёму ресурса для многоадресных соединений.

Авторы благодарят доцентов кафедры прикладной информатики и теории вероятностей РУДН, сотрудников Технологического университета г. Тампере (Финляндия) Д.А. Молчанова и А.К. Самуйлова за постановку задачи и советы по построению математической модели, доцента кафедры прикладной информатики и теории вероятностей РУДН Ю.В. Гайдамака за советы по расчёту вероятностных характеристик модели.

ЛИТЕРАТУРА

1. Sundaesan K., Ramachndran K., Rangarajan S. Optimal Beam Scheduling for Multicasting in Wireless Networks in MobiCom '09 Proceedings of the 15th annual international conference on Mobile computing and networking, P. 205–216, 2009.
2. Kim W., Song T., Pack S. Rate Adaptation for Directional Multicast in IEEE 802.11ad Networks in IEEE International Conference on Consumer Electronics, P. 364–365, January 2012.
3. Naumov V., Samouylov K., Sopin E., Andreev S. Two approaches to analyzing dynamic cellular networks with limited resources. // Proc. of 6th International Congress on Ultra Modern Telecommunications and Control Systems (ICUMT-2014), Saint-Petersbourg, Russia, 2014. – P. 585–588.
4. Наумов В.А., Самуйлов К.Е., Гайдамака Ю.В. Мультипликативные решения конечных цепей Маркова. – М.: Изд-во РУДН, 2015. – 159 с.

АНАЛИЗ УСТАНОВЛЕНИЯ СОЕДИНЕНИЯ ПАР ПЕРЕДАТЧИК-ПРИЁМНИК НА MMWAVE

Е.Д. Макеева, П.А. Харин, Н.А. Поляков, Е.В. Маркова, И.А. Гудкова

Российский университет дружбы народов

len16730637@yandex.ru, gruzavjeg@mail.ru, goto97@mail.ru, markova_ev@pfur.ru,
igudkova@sci.pfu.edu.ru

Введение

Система mmWave, использующая миллиметровый диапазон частот, позволяющий передавать данные на высоких скоростях, позволяет решить проблему нехватки частного диапазона в беспроводных сетях. Данная система отличается низкой дальностью сигнала, а также достаточно точным определением диапазона, скорости и угла его распределения. В работе рассмотрена задача определения области действия сигналы пары передатчик-приёмник.

1. Постановка задачи исследований

В данной работе рассматривается произвольная область на плоскости, в которой равномерно и непрерывно возникают пары точек. Обозначим эти точки А и В. Пусть точка А передаёт данные (передатчик), точка В принимает (приёмник). Сигналы между точками распространяются в пределах некоторой области. Задача исследования состоит в том, чтобы определить максимальное количество пар передатчик-приёмник, которые могут возникнуть в заданной плоскости, так чтобы области распространения сигнала каждой пары не пересекались друг с другом, в противном случае передача сигнала от вновь возникнувшей пары будет заблокирована. Для этих целей в работе предложен алгоритм определения границ распространения сигнала между произвольной парой передатчик-приёмник, рассмотрены результаты программной реализации алгоритма, получены значения вероятности блокировки и среднего числа устанавливаемых в секунду соединений.

2. Анализ границ распространения сигнала

Пусть точка А имеет координаты (x_1, y_1) , точка В – (x_2, y_2) . Угол распространения сигнала равен α , дальности действия передатчика и приемника одинаковы – R (рис. 1).

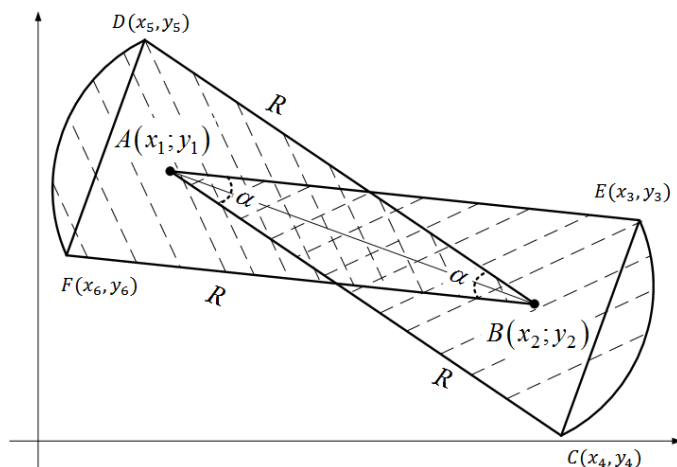


Рис. 1. Построение области распространения сигнала между точками А и В

Обозначим расстояние между точками А и В как d . Будем считать, что в случае, когда расстояние d больше, чем дальность действия R , то соединение не может быть

установлено. Таким образом, для установления соединения необходимо выполнение следующего неравенства:

$$\sqrt{(x_1 - x_2)^2 - (y_1 - y_2)^2} < R.$$

Перейдём к определению границ распространения сигнала между точками А и В, согласно рис. 1 – определению границ треугольника АЕС и ВДФ.

В качестве примера рассмотрим анализ только одного из них – АЕС, для этого разобьём его на два треугольника АВЕ и АВС.

Шаг 1. Найдем длину АВ, равную d , по формуле

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}.$$

Шаг 2. Рассчитаем длины сторон ВЕ и ВС по теореме косинусов

$$BE = BC = \sqrt{R^2 + d^2 - 2Rd \cos \frac{\alpha}{2}}.$$

Шаг 3. Составим систему уравнений, решение которой даёт координаты точек Е и С, т.е. пары (x_3, y_3) и (x_4, y_4) соответственно

$$\begin{cases} BE^2 = BC^2 = (x_2 - x_i)^2 + (y_2 - y_i)^2, \\ R^2 = (x_1 - x_i)^2 + (y_1 - y_i)^2, \end{cases} \quad (1)$$

где $i = 3, 4$.

Шаг 4. Решим систему уравнений (1) методом вычитания.

1. Вычтем из первого уравнения второе, получим следующее уравнение:

$$R^2 - BE^2 + (x_2 - x_i)^2 + (y_2 - y_i)^2 - (x_1 - x_i)^2 - (y_1 - y_i)^2 = 0, \quad i = 3, 4.$$

2. Раскроем скобки и сгруппируем слагаемые следующим образом

$$(R^2 - BE^2 + x_2^2 + y_2^2 - x_1^2 - y_1^2) + 2(x_1 - x_2)x_i + 2(y_1 - y_2)y_i = 0, \quad i = 3, 4. \quad (2)$$

3. Введём обозначения для констант

$$C = R^2 - BE^2 + x_2^2 + y_2^2 - x_1^2 - y_1^2,$$

$$C_x = 2(x_1 - x_2),$$

$$C_y = 2(y_1 - y_2).$$

4. Уравнение (2) примет вид

$$C + C_x x_i + C_y y_i = 0, \quad i = 3, 4.$$

5. Если $y_1 \neq y_2$, то выразим y_i через x_i ,

$$y_i = -\frac{C + C_x x_i}{C_y}, \quad i = 3, 4. \quad (3)$$

1) Подставим выражение (3) в первое уравнение системы (1) и сгруппируем слагаемые следующим образом

$$\left(-BE^2 + x_2^2 + y_2^2 + \left(\frac{C}{C_y}\right)^2\right) + 2\left(-x_2 + \frac{C_x}{C_y}\left(\frac{C}{C_y} + y_2\right)\right)x_i + \left(1 + \left(\frac{C_x}{C_y}\right)^2\right)x_i^2 = 0, \quad i = 3, 4.$$

2) Введём обозначения для констант

$$A = 1 + \left(\frac{C_x}{C_y}\right)^2,$$

$$B = 2 \left(-x_2 + \frac{C_x}{C_y} \left(\frac{C}{C_y} + y_2 \right) \right),$$

$$E = -BE^2 + x_2^2 + y_2^2 + \left(\frac{C}{C_y} \right)^2.$$

3) Получим квадратное уравнение

$$Ax_i^2 + Bx_i + E = 0, \quad i = 3, 4. \quad (4)$$

4) Найдем корни уравнения (4)

$$x_i = \frac{-B \pm \sqrt{B^2 - 4AE}}{2A} \quad i = 3, 4.$$

5) Подставим найденные корни в уравнение (3)

$$y_i = -\frac{C + C_x \frac{-B \pm \sqrt{B^2 - 4AE}}{2A}}{C_y}, \quad i = 3, 4.$$

6) Если $y_1 = y_2$, то выразим x_i через y_i

$$x_i = -\frac{C + C_y y_i}{C_x}, \quad i = 3, 4.$$

Далее аналогично выполняем пункты 1) – 5).

7. Получим следующие формулы для расчета координат (x_3, y_3) и (x_4, y_4)

$$x_i = -\frac{C + C_y \frac{-B \pm \sqrt{B^2 - 4AE}}{2A}}{C_x}, \quad i = 3, 4,$$

$$y_i = \frac{-B \pm \sqrt{B^2 - 4AE}}{2A}, \quad i = 3, 4,$$

где

$$A = \left(\frac{C_y}{C_x} \right)^2 + 1,$$

$$B = 2 \frac{C_y}{C_x} \left(x_2 + \frac{C}{C_x} \right) - 2y_2,$$

$$E = -BE^2 + x_2^2 + 2 \frac{x_2 C}{C_x} + \left(\frac{C}{C_x} \right)^2 + y_2^2.$$

Шаг 5. Запишем уравнения прямых АЕ, АС, ЕС, подставив координаты (x_1, y_1) , (x_3, y_3) , (x_4, y_4) в формулу уравнения прямой, проходящей через две точки:

$$\frac{y - y_j}{y_i - y_j} = \frac{x - x_j}{x_i - x_j},$$

где $i = 3, 4$, $j = 1, 3$.

Шаг 6. Выпишем неравенства, определяющие область треугольника АЕС

$$\begin{cases} AE: y \leq \frac{(x-x_1)(y_3-y_1)}{x_3-x_1} + y_1, \\ AC: y \leq \frac{(x-x_1)(y_4-y_1)}{x_4-x_1} + y_1, \\ CE: y \leq \frac{(x-x_3)(y_4-y_3)}{x_4-x_3} + y_3. \end{cases}$$

Расчёт границ второго треугольника BDF проводится по аналогии.

3. Программная реализация алгоритма

Для получения численных результатов было разработано программное средство на языке программирования Java. Пусть в системе равномерно и непрерывно с интенсивностью λ , определённой в соответствии с законом Пуассона, генерируется N пар передатчик-приёмник. Время передачи данных является экспоненциально распределённым и равно μ^{-1} . Сведём используемые в программе начальные данные в табл. 1.

Таблица 1

Начальные данные для реализации программы

N	100 000
α	20
R	35
λ	1.0
μ	0.1

Графическая реализация программы представлена на рис. 2. Следует отметить, что после окончания передачи данных, т.е. по истечении времени μ^{-1} , пара передатчик-приёмник исчезает с экрана.

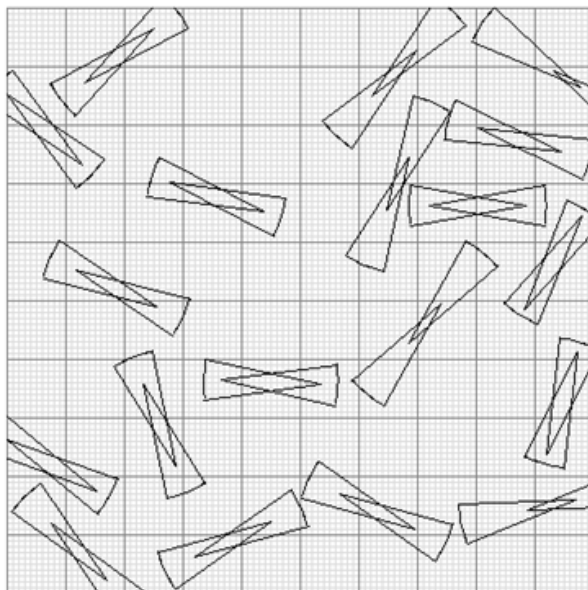


Рис. 2. Графическая реализация программы

Результаты выполнения программы представлены в табл. 2.

Результаты выполнения программы

Количество случаев, когда соединение не будет установлено	85 510
Количество случаев, когда передача сигнала будет заблокирована	11 002
Вероятность блокировки установления соединения	0,75928
Среднее число активных пар передатчик-приемник в секунду	3,939168

Заключение

В работе рассмотрена произвольная область на плоскости, в которой равномерно и непрерывно возникают передатчики с приёмниками. Предложен алгоритм расчёта границ распространения сигнала между произвольной парой передатчик-приёмник. Представлены результаты программной реализации алгоритма, позволяющей графически отобразить области возникновения пар передатчик-приёмник, а также получены конкретные значения для основных характеристик системы – вероятности блокировки установления соединения и среднего числа активных пар передатчик-приёмник в секунду.

МОДЕЛЬ МОБИЛЬНОСТИ В ВИДЕ СЕМО С "ПАССИВНЫМ" УЗЛОМ

В.Н. Савич, В.А. Бесчастный, Д.Ю. Острикова, И.А. Гудкова

Российский университет дружбы народов

valeria.savich16@gmail.com, {ostrikova_dyu, beschastnyy_va, gudkova_ia}@rudn.university

Введение

В данной работе рассматривается сеть массового обслуживания с M «активными» узлами и дополнительным «пассивным» узлом. В работе исследуется технология мультимедийного вещания – форма передачи данных, при которой адресом назначения сетевого пакета является многоадресная группа.

Число и разнообразие мобильных многоадресных приложений растёт в быстром темпе. Поставщики мобильной сети сталкиваются с резким увеличением нагрузки на многоадресный трафик, и ожидается, что этот рост продолжится в сетях пятого поколения (5G). Основные проблемы связаны с тем, что многоадресный трафик не только нацелен на группы конечных пользователей; он также включает в себя коммуникацию машинного типа (MTC) для интернета вещей (Internet of Things). Увеличение нагрузки MTC, предсказанное для 5G, ставит под сомнение эффективность текущего мультимедийного широкоадресного многоадресного обслуживания (MBMS). Спрос на многоадресные приложения сотовых систем продолжает быстро расти. Как следствие, многоадресная передача будет играть ключевую роль в появляющихся сетях 5G, как указано в технических документах, исследовательских проектах и стандартных документах от проекта партнерства третьего поколения (3GPP). Действительно, многоадресная передача представляет собой жизнеспособное и эффективное решение для одновременной передачи данных в группу терминалов посредством связи «point-to-multipoint» (PtM) с положительными последствиями для пропускной способности и эффективности спектра сотовых систем. Обе функции имеют решающее значение для внедрения сетей 5G. Сценарий многоадресной рассылки 5G становится более сложным в том случае, когда мини-соты, лежащие в основе макропокрытия, будут использоваться для усиления сигнала уровнями, связанных с человеческими службами, и для увеличения пропускной способности в сценариях MTC, где устройства обычно расположены в труднодоступных местах. Становится очевидным, что услуга Multimedia Broadcast Multicast Service (MBMS) 3GPP нуждается в новых архитектурных и процедурных определениях для удовлетворения различных ограничений ожидаемых услуг многоадресной передачи 5G.

Хотя многоадресная передача привлекает всё большее внимание со стороны широкого исследовательского сообщества, многие проблемы по-прежнему препятствуют его эффективному размещению в сетях 5G. Недостаточно внимания было уделено разработке архитектурных и процедурных определений для решения новых задач многоадресной рассылки в сетях 5G, в которых тот же приоритет следует отдавать как человеческому, так и машинному типу обслуживания [4].

Исходя из распределения вероятностей для многоадресной группы можно улучшить производительность сети. Таким образом, мы можем оптимально сформировать подгруппы для достижения максимальной ADR (Aggregate Data Rate) среди всех пользователей в соте. Распределение вероятностей модели CeMO с «пассивными» узлом осваивается на модели с «активными» узлами.

1. Многоадресные подходы в сетях LTE

LTE является наиболее благоприятным стандартом для поддержки многоадресных передач. При многоадресной рассылке, когда нескольким пользователям требуется одно и то же содержимое, выбор параметров передачи выполняется в соответствии с базисом для каждой группы. Таким образом, необходимо учитывать несколько проблем, связанных с различными условиями канала пользователей. В частности, выбор схемы модуляции и кодирования (MCS) зависит от индикатора качества канала (Channel quality indicator, CQI). В условиях мобильности CQI каждого пользователя периодически изменяется. Это включает в себя изменение конфигурации многоадресной группы, которая должна быть пересчитана каждый интервал времени передачи (TTI).

Обычная схема многоадресной рассылки (CMS) является методом справедливого распределения ресурсов, т.к. все пользователи многоадресной рассылки обслуживаются с одинаковой скоростью передачи данных. С другой стороны, CMS страдает низкой спектральной эффективностью из-за адаптации уровня MCS к уровню пользователя с худшим состоянием канала.

Для преодоления этой проблемы вероятностная схема многоадресной рассылки (OMS) использует разнообразие пользовательской соте. В некоторый промежуток времени OMS обслуживает только группы членов многоадресной рассылки в целях обеспечения максимальной заданной целевой функции (т.е. пропускной способности). Ограничение такого подхода связано с увеличением многоадресной рассылки. Кроме того, OMS страдает от краткосрочной справедливости, поскольку не все пользователи обслуживаются в каждом TTI, но она достигает долгосрочной справедливости. Чтобы преодолеть вышеупомянутые ограничения, авторы [3] предлагали многоадресную подгруппу, где все пользователи, принадлежащие к различным многоадресным подгруппам, обслуживаются каждым TTI. Многоадресная подгруппа минимизирует негативные последствия, связанные с пользователями с наименьшим коэффициентом усиления каналов, и гарантирует хороший компромисс между пропускной способностью и справедливостью.

2. Построение математической модели с «активными» узлами

В мобильных сетях пользователи могут быть как «активными», т.е. пользоваться сотовой связью в текущий момент времени, так и «пассивными» – не получать сигнала базовой станции. Таким образом, абоненты постоянно перемещаются по узлам с различными вероятностями переходов между узлами.

Рассматриваемая сеть является замкнутой – это сеть с множеством узлов без источника и стока, в которой циркулирует постоянное число N однотипных заявок [1]. Ранее в [2] была рассмотрена сеть массового обслуживания только с «активными» пользователями, т.е. абоненты, в данном случае, пользуются мобильной сетью постоянно. Схема такой сети изображена на рис. 1:

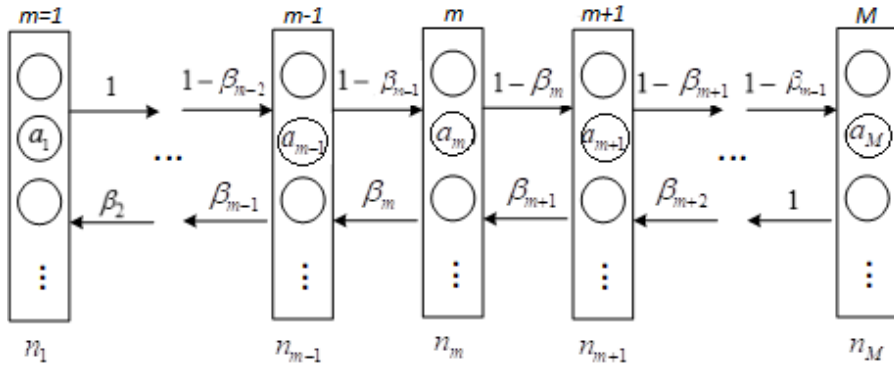


Рис. 1. Схема сети массового обслуживания с «активными» узлами

Здесь M – количество узлов, N – общее количество заявок системе; через β_m обозначена вероятность события, когда пользователь перемещается дальше от BS (Base station) и изменяет показатель CQI (Channel Quality Indicator) уровня от m до $m-1$; $\alpha_m = 1 - \beta_m$ – вероятность события, когда пользователь перемещается ближе к BS и изменяет показатель CQI уровня от m до $m-1$; и a_m – среднее время, когда пользователь не меняет свой уровень CQI.

Пространство состояний системы $X(M, N)$ удовлетворяет соотношению:

$$X(M, N) = \left\{ n : n_i = \overline{0, N}; \sum_{i=1}^M n_i = N \right\}.$$

В данном случае стационарные вероятности $p(n)$ состояний системы вычисляются следующим образом: $p(n) = G^{-1}(M, N) \prod_{m=1}^M \frac{(h_m a_m)^{n_m}}{n_m!}$, где

$$G(M, N) = \sum_{n \in X(M, N)} \prod_{m=1}^M \frac{(h_m a_m)^{n_m}}{n_m!}, \quad N \geq 1, \quad M > 0.$$

Здесь h_m , $m \in M$ – количество раз, когда пользователь сообщает m -уровень CQI в течение указанного интервала времени – рассчитывается следующим образом: $h_1 = 1$, $h_2 = \frac{1}{\beta_2}$, $h_m = \frac{1}{\beta_m} h_{m-1} - \frac{\alpha_{m-2}}{\beta_m} h_{m-2}$, $m = \overline{3, M}$.

По алгоритму Бузена константа нормализации $G(M, N)$ рассчитывается по следующей рекурсии: $g(m, n) = \sum_{v=0}^n G(m-1, v) f_m(n-v)$, $m = \overline{1, M}$, $n = \overline{0, N}$, $g(m, 0) = 1$,

$$m = \overline{0, M}, \quad G(0, n) = 0, \quad n = \overline{1, N}, \quad \text{где } f_m(v) = \begin{cases} 1, & n = 0, \\ \frac{h_m a_m}{v} f_m(v-1), & v = \overline{1, N}. \end{cases}$$

Зная стационарное распределение вероятностей $p(n)$ числа членов многоадресной рассылки по уровню m -CQI, мы применяем схему оптимального формирования подгруппы для расчета максимального ADR $b(n)$ всех пользователей в многоадресной группе и его среднего значения $\bar{b} = \sum_{n \in X(M, N)} p(n) \cdot b(n)$.

Данная модель не является совершенной, потому что пользователи не всегда находятся в сети. Таким образом, сеть с одним «пассивным» узлом, которая представляет для нас интерес, является расширением для данной модели.

3. Построением модели с «пассивным» узлом

Схема модели с «пассивным» узлом, изображена на рис. 2:

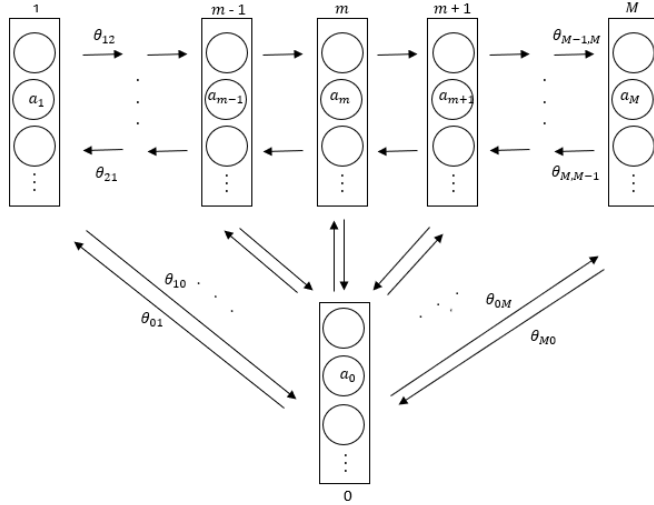


Рис. 2. Схема сети массового обслуживания с «пассивным» узлом

Здесь M – количество активных узлов, нулевым обозначен «пассивный узел», N – общее количество заявок системе, как в активных, так и пассивном узле; через θ_{mm} , $m = \overline{1, M}$ обозначены вероятности переходов между соседними узлами, где находятся активные пользователи, через θ_{0m} , $m = \overline{1, M}$ обозначены вероятности переходов от «пассивного» узла к активным. Среднее время, пока пользователь не перемещается из области m , обозначено через a_m .

Таким образом, для того, чтобы доказать, что данная физическая система существует, она должна находиться в одном из доступных ей состояний, т.е. удовлетворять условию нормировки: $\sum_{m=1}^M \theta_{0m} = 1$.

Заключение

В данной работе рассмотрена сеть массового обслуживания с «активными» узлами и одним «пассивным», выведено условие нормировки, удовлетворяющее условию существования данной системы. Модель может использоваться для анализа обслуживания мобильных пользователей в сети LTE.

ЛИТЕРАТУРА

1. Башарин Г.П. Лекции по математической теории телетрафика. – Москва: Российский университет дружбы народов, 2007. – 268 с.
2. Ostrikova Darya, Rinaldi Federica, Beschastnyi Vitalii, Gudkova Irina, Militano Leonardo, Araniti Giuseppe, Iera Antonio, Samouylov Konstantin. Analytical Model for Multicast Subgrouping in 5G-Mobile eMBMS Environment // IEEE, New Jersey, Piscataway, USA. – 2017. – P. 13–19.
3. Araniti G., Condoluci M., Militano L., Iera A. Adaptive Resource Allocation to Multicast Services in LTE Systems // IEEE Transaction on Broadcasting, Dec. 2013.
4. Araniti Giuseppe, Condoluci Massimo, Scopelliti Pasquale, Molinaro Antonella, Iera Antonio. Multicasting over Emerging 5G Networks: Challenges and Perspectives // March-April 2017.

СХЕМА МОДЕЛИ БЕСПРОВОДНОЙ СЕТИ LTE С ТЕХНОЛОГИЕЙ СОВМЕСТНОГО ДОСТУПА К НЕЛИЦЕНЗИРОВАННОМУ ДИАПАЗОНУ ЧАСТОТ WI-FI*

Синицына А.А., Филипова В.Р., Гудкова И.А., Маркова Е.В.

Российский университет дружбы народов

aa.sinityna@yandex.ru, valeryflp@gmail.com, igudkova@sci.pfu.edu.ru, markova_ev@pfur.ru

Введение

За последние несколько лет число пользователей мобильных сетей и их потребность в высокоскоростных услугах резко возросли [1]. Это привело к значительному увеличению объёма генерируемого в мобильных сетях трафика, для передачи которого требуется не только высокая пропускная способность сети, но и возможность операторов предоставлять пользователям услуги с требуемым качеством обслуживания (Quality of Service, QoS), которое также постоянно повышается.

Наиболее предпочтительным спектром для качественного предоставления услуг является лицензированный диапазон частот, позволяющий предоставлять высококачественные услуги с максимальной спектральной эффективностью. Однако, в условиях стремительного роста объёма мобильных данных, возникает проблема нехватки частот лицензированного диапазона, требующая от операторов и органов стандартизации поисков эффективного решения. Существует два типа решения данной проблемы: привлечение дополнительных ресурсов, например, использование более высоких частот, таких, как миллиметровые волны [2], или более эффективное использование уже имеющихся, например, за счёт временного использования лицензированных диапазонов спектра (система Licensed Shared Access, LSA) [3]; использование полосы ISM (Industrial Scientific and Media), к которой относится система совместного доступа (Licensed Assisted Access, LAA) [4,5] к нелицензированным частотам Wi-Fi.

Впервые система LAA была представлена консорциумом 3GPP [6] как часть технологии LTE Advanced (LTE-A), и определена как система, способная обеспечить сосуществование сетей LTE и Wi-Fi в нелицензированных диапазонах. Согласно данным аналитиков, Wi-Fi – наиболее популярная и успешная технология, предоставляющая беспроводное обслуживание на нелицензированных частотах. Благодаря низкой стоимости и высокой скорости передачи данных, системы Wi-Fi до сих пор занимали главенствующую позицию на всех нелицензированных диапазонах в 2,4 ГГц и 5 ГГц, поэтому большинство пользовательских электронных устройств в настоящее время оснащены встроенными модулями Wi-Fi. Однако, при высокой загрузке, эффективность спектра Wi-Fi сети довольно низкая и не даёт пользователям никаких гарантий по уровню QoS. В отличие от Wi-Fi, технология LTE способна более эффективно управлять ресурсами беспроводной сети и предоставлять пользователям услуги с требуемым уровнем QoS. Использование системы LAA позволяет улучшить эффективность спектра на нелицензированных частотах [7]. Следует отметить, что в текущей версии LAA частоты агрегируются только на нисходящей линии связи.

Согласно спецификации [6] консорциума 3GPP, одной из главных задач системы LAA является обеспечение эффективного, бесперебойного и справедливого сосуществования с сети Wi-Fi с сетью LTE. Основным требованием сосуществования этих сетей является то, что базовая станция LAA не должна интерферировать с точкой доступа Wi-Fi больше, чем она интерферировала бы с любой другой точкой доступа Wi-Fi, работающей на том же канале.

* Публикация подготовлена при поддержке Программы РУДН «5-100» и при финансовой поддержке РФФИ в рамках научных проектов № 16-07-00766 и № 18-07-00576.

В работе рассматривается схема модели беспроводной сети LTE, для которой при помощи системы LAA реализован доступ к нелицензированным частотам сети Wi-Fi. По сети LTE осуществляется передача потокового трафика, характеризующегося гарантированной скоростью передачи данных, по сети Wi-Fi – адаптивного потокового трафика, скорость передачи которого зависит от загрузки системы.

1. Основные обозначения

Пусть ёмкость рассматриваемой системы равна C . Запросы на установление сессии Wi-Fi или LTE поступают согласно закону Пуассона с интенсивностью λ_1 или λ_2 соответственно. Предположим, что среднее время обслуживания одной Wi-Fi или LTE-сессии является экспоненциальным и равно μ_1^{-1} или μ_2^{-1} . Тогда предложенная нагрузка, создаваемая соответственно Wi-Fi или LTE-сессиями, определяется по формуле $\rho_1 = \lambda_1/\mu_1$ или $\rho_2 = \lambda_2/\mu_2$. Пользователям сети Wi-Fi услуги предоставляются без гарантий к качеству обслуживания, при этом ограничение на минимальную скорость передачи данных определяется как b_1 . Для предоставления пользователям сети LTE услуг с требуемым уровнем QoS скорость передачи данных b_2 должна быть гарантированной, положим её равной 1.

Опишем состояние системы в некоторый момент времени $t \geq 0$ при помощи двумерного вектора (n_1, n_2) , где n_1 – число установленных Wi-Fi сессий, а n_2 – число установленных LTE сессий. Тогда пространство состояний системы имеет вид:

$$X = \{(n_1, n_2) : n_1 = 0, 1, 2, \dots, n_2 = 0, 1, \dots, C - n_1, n_1 + n_2 \leq C\}.$$

Отметим, что трафик, передаваемый по сети Wi-Fi, является адаптивным, текущая скорость его передачи $b_1(n_1, n_2)$ зависит от количества установленных в системе сессий и при большой загрузке сети может стремиться к нулю, определим её с помощью следующей формулы:

$$b_1(n_1, n_2) = \begin{cases} 1, & n_1 + n_2 \leq C, \\ \frac{C}{n_1}, & n_1 > C. \end{cases}$$

2. Построение математической модели

Для дальнейшего анализа системы рассмотрим граф интенсивностей переходов, представленный на рис. 1, и соответствующее центральное состояние (рис. 2).

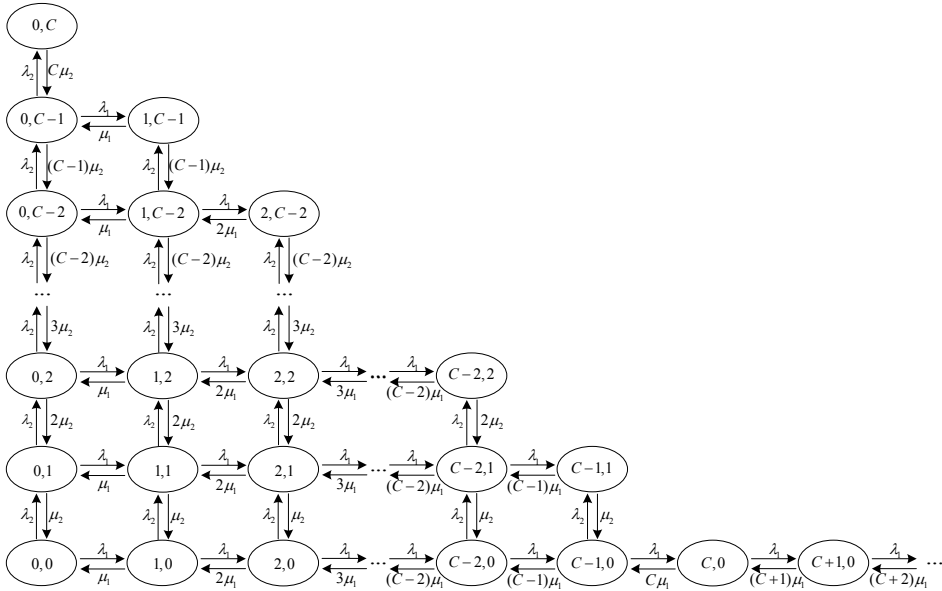


Рис. 1. Граф интенсивностей переходов

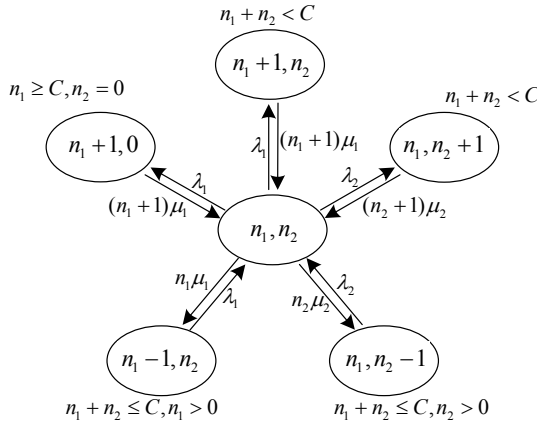


Рис. 2. Центральное состояние системы

Тогда система уравнений глобального баланса (СУГБ) имеет вид:

$$\begin{aligned}
 & p(n_1, n_2) \left[\lambda_1 (I(n_1 \geq C, n_2 = 0) + I(n_1 + n_2 \leq C)) + \lambda_2 I(n_1 + n_2 < C) + \right. \\
 & \left. + n_2 \mu_2 (n_1 + n_2 \leq C, n_2 > 0) + n_1 \mu_1 I(n_1 + n_2 \leq C, n_1 > 0) \right] = p(n_1 + 1, 0)(n_1 + 1)\mu_1 I(n_1 \geq C, n_2 = 0) + \\
 & + p(n_1, n_2 + 1)(n_2 + 1)\mu_2 I(n_1 + n_2 < C) + p(n_1 + 1, n_2)(n_1 + 1)\mu_1 I(n_1 + n_2 < C) + \\
 & + p(n_1, n_2 - 1)\lambda_2 I(n_1 + n_2 \leq C, n_2 > 0) + p(n_1 - 1, n_2)\lambda_1 I(n_1 + n_2 \leq C, n_1 > 0),
 \end{aligned}$$

где $p(n_1, n_2)$, $(n_1, n_2) \in \mathbf{X}$ – стационарное распределение вероятностей состояний системы.

Так как для данной системы выполнен критерий Колмогорова – Чепмена [8], для нахождения $p(n_1, n_2)$, $(n_1, n_2) \in \mathbf{X}$ составим систему уравнений частичного баланса (СУЧБ):

$$\begin{cases} \lambda_1 p(n_1 - 1, n_2) = n_1 \mu_1 p(n_1, n_2), \\ \lambda_2 p(n_1, n_2 - 1) = n_2 \mu_2 p(n_1, n_2). \end{cases}$$

Рассмотрим первое уравнение системы, получим

$$p(n_1, n_2) = \frac{\lambda_1}{n_1 \mu_1} p(n_1 - 1, n_2) = \frac{\rho_1}{n_1} p(n_1 - 1, n_2) = \frac{\rho_1^2}{n_1 (n_1 - 1)} p(n_1 - 2, n_2) = \dots = \frac{\rho_1^{n_1}}{n_1!} p(0, n_2).$$

Аналогично, используя второе уравнение, получим

$$p(n_1, n_2) = \frac{\lambda_2}{n_2 \mu_2} p(n_1, n_2 - 1) = \frac{\rho_2}{n_2} p(n_1, n_2 - 1) = \frac{\rho_2^2}{n_2 (n_2 - 1)} p(n_1, n_2 - 2) = \dots = \frac{\rho_2^{n_2}}{n_2!} p(n_1, 0).$$

Тогда стационарное распределение $p(n_1, n_2)$, $(n_1, n_2) \in \mathbf{X}$ вероятностей состояний системы определяется следующим образом: $p(n_1, n_2) = p(0, 0) \frac{\rho_1^{n_1} \rho_2^{n_2}}{n_1! n_2!}$.

Для нахождения $p(0, 0)$ воспользуемся условием нормировки. Учитывая особенности пространства состояний \mathbf{X} , $p(0, 0)$ можно выразить как $p(0, 0) = \left(\sum_{i=1}^{\infty} \sum_{j=1}^{C-i} \frac{\rho_2^j \rho_1^i}{j! i!} \right)^{-1}$.

Основными вероятностными характеристиками модели являются вероятность блокировки B_1 и B_2 запросов на передачу данных по сети Wi-Fi или LTE соответственно:

$$B_1 = \sum_{n_1=0}^{C-1} p(n_1, C - n_1), \quad B_2 = \sum_{n_1=0}^{C-1} p(n_1, C - n_1) + \sum_{n_1=C}^{\infty} p(n_1, 0).$$

Заключение

В статье рассмотрена одна из возможных схем доступа технологии LTE к нелицензированному частотному диапазону Wi-Fi при помощи системы LAA. Услуги, предоставляемые пользователям сетей, генерируют потоковый трафик, в случае передачи по сети Wi-Fi этот трафик является адаптивным.

В дальнейшем планируется рассмотреть модели схем совместного доступа к нелицензированному диапазону Wi-Fi с приоритетом запросов, обслуживаемых сетью Wi-Fi, и различными типами трафика – потоковым и эластичным, найти вероятностные характеристики для моделей с потоковым трафиком и вероятностно-временные характеристики для моделей с эластичным, провести сравнительный анализ всех построенных моделей с системой LAA.

ЛИТЕРАТУРА

1. Cisco. The Zettabyte Era: Trends and Analysis. – 2017.
2. Rappaport T.S., Sun S., Mayzus R., Zhao H., Azar Y., Wang K., Wong G.N., Schulz J.K., Samimi M., Gutierrez F. Millimeter Wave Mobile Communications for 5G Cellular: It Will Work! // IEEE Access. – 2013. – Vol. 1. – P. 335–349.
3. Ponomarenko-Timofeev A., Pyataev A., Andreev S., Koucheryavy Ye., Mueck M., Karls I. Highly dynamic spectrum management within licensed shared access regulatory framework // IEEE Communications Magazine. – 2015. – Vol. 54. – No. 3. – P. 100–109.
4. Mukherjee A., Cheng J.-F., Falahati S., Koorapaty H., Karaki R., Falconetti L., Larsson D. et al. Licensed-assisted access LTE: Coexistence with IEEE 802.11 and the evolution toward 5G // IEEE Communications Magazine. – 2015. – Vol. 54. – No. 6. – P. 50–57.
5. Maule M., Moltchanov D., Kustarev P., Komarov M., Andreev S., Koucheryavy Y. Delivering Fairness and QoS Guarantees for LTE/Wi-Fi Coexistence under LAA Operation // IEEE Access, IET. – 2018. – Vol. 6. – P. 7359–7373.
6. 3GPP TR.36.889. Feasibility study on licensed-assisted access to unlicensed spectrum. – Technical Report, Release 13. – 2016.
7. Yin R., Yu G., Maaref A., Li G.Y. Adaptive LBT for Licensed Assisted Access LTE Networks // IEEE Globecom 2015. – 2015. – P. 1–6.
8. Башарин Г.П. Лекции по математической теории телетрафика. – М.: Изд-во РУДН, 2009. – 342 с.

ПРИМЕНЕНИЕ МОДЕЛИ ДВИЖЕНИЯ ГАУССА – МАРКОВА К АНАЛИЗУ ИНТЕРФЕРЕНЦИИ В БЕСПРОВОДНОЙ D2D-СЕТИ

О.О. Цветкова

Российский университет дружбы народов
oksana.tsvetkova.1995@mail.ru

Введение

Согласно прогнозам Cisco, опубликованным в отчёте «Наглядный индекс развития сетевых технологий» (Cisco Visual Networking Index™ Complete Forecast, Cisco VNI) [1], ежемесячный глобальный мобильный трафик данных будет составлять 49 экзабайт (1018 байт) к 2021 г., а годовой трафик будет превышать половину зеттабайта (1021 байт). При этом количество мобильных подключенных устройств на душу населения достигнет 1,5 к 2021 г. По России [2] мобильный трафик данных вырастет с 2016 по 2021 гг. пятикратно, его годовой прирост составит 37%. Если в 2016 г. на долю мобильного трафика приходилось 13% всего IP-трафика, в 2021 г. этот показатель составит 28%. В 2021 г. в России будет 841,3 млн сетевых устройств, или 6 устройств на душу населения (в 2016 г. этот показатель составлял 3,6). Эти цифры указывают на то, что в будущем сети подвижной связи будут важнейшей составляющей при предоставлении услуг связи пользователям. На сегодняшний день инновационной сетью следующего поколения, уже функционирующей в России, является сеть 5G (Fifth Generation), которая поддерживает предоставление услуг в рамках концепции Internet of Things (IoT), в том числе, технологию Device-to-Device (D2D) Communications. Технология D2D позволяет беспроводным устройствам, находящимся неподалеку друг от друга, обмениваться данными напрямую, без участия сети. Преимуществом такого подхода является снятие нагрузки с сетевого оборудования, а именно, с базовых станций сотовой сети подвижной связи, а также возможность использования для передачи данных нелицензируемой части радиоспектра, что позволяет дополнительно разгрузить сеть. Недостатком технологии является необходимость контроля интерференции, которая возникает при использовании одной и той же или близких радиочастот для обмена данными между несколькими парами беспроводных устройств. Интерференция снижает скорость передачи данных в радиоканале, связывающем приёмник и передатчик, что влияет на качество предоставления услуги для этой пары. В данной работе исследуется влияние модели движения беспроводных устройств на интерференцию и, следовательно, на качество предоставления услуг в сетях подвижной связи поколения 5G.

1. Постановка задачи

Одним из основных показателей качества соединения в радиоканале между приёмным и передающим беспроводными устройствами является параметр SIR (Signal to Interference Ratio) – отношение уровней полезного $S(l)$ и интерферирующего сигналов $I(l)$, которое измеряется на приёмнике:

$$SIR = \frac{S(l)}{I(l)}. \quad (1)$$

Уровень сигнала в числителе и в знаменателе формулы (1) определяется традиционной моделью распространения радиосигнала – так называемой «path loss model» [3]. Эта модель основана на предположении, что среднее значение уровня сигнала обратно пропорционально расстоянию между передатчиком и приемником и имеет вид

$$S(l) = g_0 I_0^{-\gamma_0}, \quad (2)$$

$$I(l) = \sum_{i=1}^{N-2} g_i I_i^{-\gamma_i}, \quad (3)$$

где g_i - базовая мощность сигнала i -го передатчика, l_i - расстояние между приёмником и i -м передатчиком, γ_i - коэффициент затухания сигнала, $i = \overline{0, N}$. Численное значение параметра γ зависит от условий распространения радиоволны между приёмником и передатчиком и может принимать значения от 2 при прямой видимости до 6 в худшем случае.

Для статического случая эта модель применялась, например, в [4], где с использованием аппарата стохастической геометрии был разработан метод исследования интерференции для различных распределений абонентов в ограниченном пространстве. Поскольку в сетях подвижной связи беспроводные устройства перемещаются в процессе получения услуг, расстояния между ними, а, следовательно, и значение параметра SIR, изменяется во времени. Для динамического случая при анализе меняющейся интерференции и зависящего от нее качества предоставления услуг необходимо уметь моделировать перемещения абонентов, получающих услуги связи с помощью беспроводных устройств. В настоящее время для моделирования передвижения беспроводных устройств используются как модели, основанные на анализе собранной статистики о перемещениях абонентов, так и модели мобильности, традиционно используемые для описания перемещения объектов [5]. Одной из моделей индивидуальной мобильности является так называемая «модель движения Гаусса—Маркова», относящаяся к классу моделей случайного блуждания (Random Walk, Random Waypoint, Random Direction и т.п.), для которых в дискретном времени на каждом временном такте случайным образом меняется скорость и направление движения объекта.

В статье при помощи модели движения Гаусса – Маркова моделируется перемещение в ограниченном пространстве N объектов, использующих технологию D2D для обмена данными. Два из этих объектов соответствуют целевой паре «приемник-передатчик», которая обменивается полезным сигналом, остальные $N - 2$ объектов моделируют интерферирующие передатчики, использующие для обмена данными со своими приемниками ту же или близкую радиочастоту.

2. Модель движения Гаусса-Маркова

Модель движения Гаусса – Маркова относится к классу моделей индивидуальной мобильности, т.е. движение объекта моделируется вне зависимости от движения других объектов в области моделирования. Моделирование движения объекта происходит в дискретном времени с заданной продолжительностью T шагов (тактов), при этом в начале каждого такта разыгрываются скорость и направление движения.

Скорость s_n движения объекта на текущем такте n зависит от нескольких параметров: скорости s_{n-1} его движения на предыдущем такте, заданных постоянных величин – средней скорости \bar{s} и параметра хаотичности α , а также от случайной величины $s_{x_{n-1}}$, имеющей гауссовское (нормальное) распределение. Модель предполагает аналогичные зависимости для угла d_n , на который меняется направление движения на текущем такте n по сравнению с направлением движения на предыдущем такте $n - 1$. Величины s_n и d_n рассчитываются по формулам

$$s_n = \alpha s_{n-1} + (1 - \alpha) \bar{s} + \sqrt{(1 - \alpha^2)} s_{x_{n-1}},$$

$$d_n = \alpha d_{n-1} + (1 - \alpha) \bar{d} + \sqrt{(1 - \alpha^2)} d_{x_{n-1}}, \text{ где } 0 \leq \alpha \leq 1; 0 < S_{\min} \leq \bar{s} \leq S_{\max}, 0 \leq \bar{d} \leq 360.$$

Заметим, что «Гаусс» в названии модели Гаусса – Маркова связано с гауссовским (нормальным) распределением случайных величин $s_{x_{n-1}}$ и $d_{x_{n-1}}$, а «Марков» объясняется тем, что будущее местоположение объекта зависит только от текущего местоположения и не зависит от предыдущей траектории движения объекта. Заметим также, что при $\alpha = 0$ модель определяет полностью хаотическое движение (броуновское движение), а при $\alpha = 1$ – прямолинейное движение. Среднее значение \bar{s} скорости движения

не меняется вне зависимости от местоположения объекта, а с помощью среднего значения \bar{d} угла может регулироваться изменение направления движения при приближении к границам области или к препятствиям внутри области моделирования, как это сделано, например, в [5].

Координаты (x_n, y_n) местоположения беспроводного устройства в конце такта рассчитываются по формулам $x_n = x_{n-1} + s_{n-1} \cos d_{n-1}$, $y_n = y_{n-1} + s_{n-1} \sin d_{n-1}$, где (x_{n-1}, y_{n-1}) – координаты текущего местоположения (в начале такта), s_{n-1} – скорость на текущем такте, d_{n-1} – угол, на который изменилось направление на текущем такте по сравнению с направлением движения на предыдущем такте.

Пример реализации модели движения Гаусса – Маркова показан на рис. 1. Объект начинает движение из точки с координатами $(x_0, y_0) = (100, 150)$ и движется в прямоугольной области размера 300x300 с заданными средними значениями скорости $\bar{s} = 7$ и угла $\bar{d} = 80$ в течение $T = 100$ тактов. Влияние параметра хаотичности α на вид траектории движения иллюстрируется на рис. 1а ($\alpha = 0$) и рис. 1б ($\alpha = 0.75$).

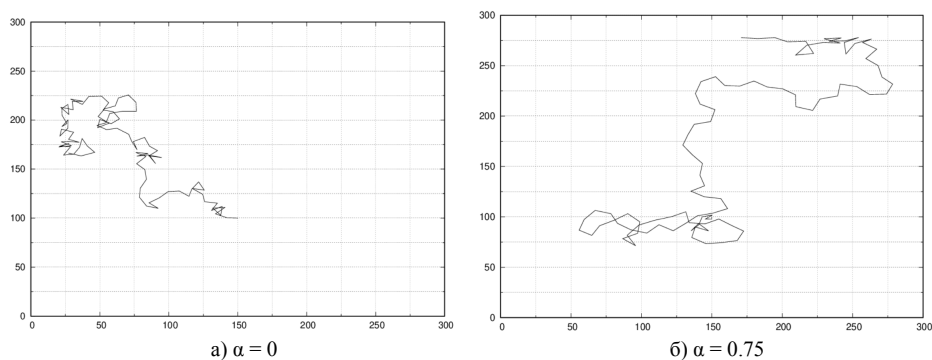


Рис. 1. Траектория движения объекта

3. Анализ отношения сигнал / интерференция

Рассмотрим зону моделирования с несколькими движущимися беспроводными устройствами (БУ). На рис. 2 представлены траектории движения трёх БУ с точками начала движения $(150, 100)$, $(150, 150)$, $(100, 100)$ для 1-го, 2-го и 3-го устройства соответственно. Параметры движения: $\alpha = 0,75$, $T = 100$, $\bar{s} = 7$, $\bar{d} = 80$.

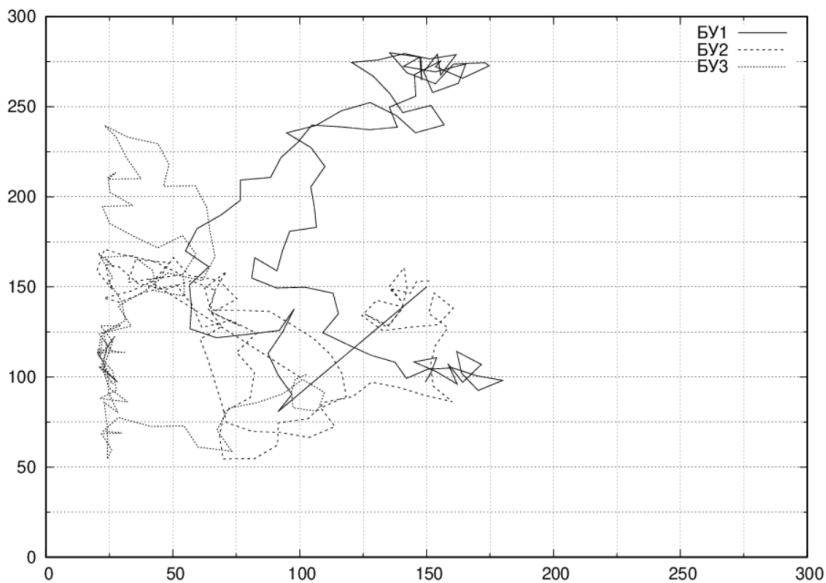


Рис. 2. Траектории движения 3-х БУ.

Показатель SIR рассчитан для целевой пары «приемник-передатчик», где БУ1 – передатчик полезного сигнала, БУ2 – приемник, БУ3 – интерферирующее устройство, в предположении, что передатчики имеют одинаковую базовую мощность сигнала g_i (2), (3). На рис. 3 представлена динамика изменения показателя SIR на каждом шаге движения.

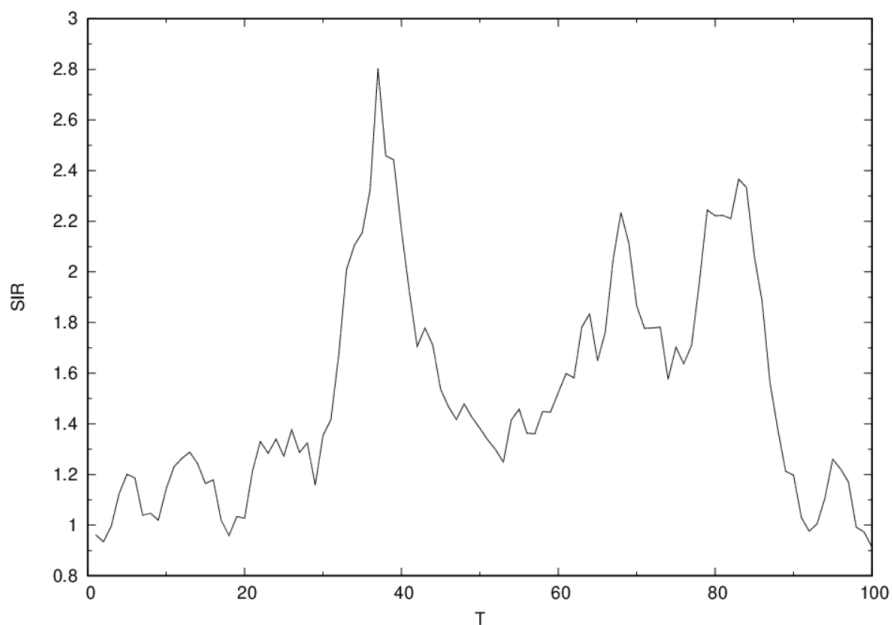


Рис. 3. Изменение SIR во времени

Такой график для SIR позволяет выделить периоды, в которые SIR падает ниже предельно допустимого уровня, когда связь в целевой паре становится невозможной и соединение прерывается.

Заключение

В настоящей работе для моделирования движения беспроводных устройств в сетях 5G исследована модель движения Гаусса – Маркова, с помощью которой получены траектории движения нескольких беспроводных устройств, независимо перемещающихся в ограниченном пространстве. На каждом шаге моделирования определялось расстояния от целевого приёмника до целевого и интерферирующих передатчиков, от которых зависит отношение сигнал / интерференция, являющееся ключевым показателем качества передачи данных в радиоканале целевой пары. Моделирование показало наличие и длительности периодов падения показателя SIR ниже порогового уровня, определенного стандартами. Предложенный метод исследования позволяет определить максимальную плотность передатчиков, при которой уровень сигнала в целевой паре будет удовлетворять требованиям к качеству соединения в радиоканале.

ЛИТЕРАТУРА

1. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021 White Paper. – Cisco, Updated: March 28, 2017. <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html>. Дата обращения: 15.04.2018.
2. Cisco Россия https://www.cisco.com/c/ru_ru/about/press/press-releases/2017/06-09b.html. Дата обращения: 15.04.2018.
3. *Theodore S. Rappaport* Wireless Communications: Principles and Practice. – Prentice Hall, 2002.
4. *Гайдамака Ю.В., Андреев С.Д., Сопин Э.С., Самуйлов К.Е., Шоргин С.Я.* Анализ характеристик интерференции в модели взаимодействия устройств с учетом среды распространения сигнала // Информатика и её применения. – 2016. – Т. 10. – Вып. 4. – С. 2–10.
5. *Camp T., Boleng J., Davies V.* A survey of mobility models for ad hoc network research // Wirel. Commun. Mob. Comput. – 2002. – 2. – P. 483–502.

СЕКЦИЯ IX. COMPUTER SCIENCE

STUDY OF THE MMPP/GI/1 QUEUEING SYSTEM WITH RANDOM CUSTOMERS' CAPACITIES

T.V. Bushkova, S.P. Moiseeva

Tomsk State University

bushkova70@mail.ru, smoiseeva@mail.ru

Introduction

Queueing systems represent a powerful mathematical tool for investigating the performance of a wide variety of real-life systems, ranging from telecommunication networks to financial markets, from computer architectures to supply chain management and airplane traffic control, just to cite a few. Analytical tractability of the corresponding models strongly depends on the nature of the underlying processes (Poisson arrivals have many nice features that strongly simplify the analysis) and on the system geometry.

Although physical resources are always finite, quite often it is easier to study queueing systems in which the corresponding parameters assume infinite values. For instance, the overflow probability is often used as an upper bound for the loss probability in finite-buffer queues and, indeed, asymptotic results are available even for strongly non-Markovian systems [1]. Moreover, infinite-server queueing systems may be applicable in case of models with a limited number of server devices as described in [2].

In this work, an infinite-server queueing system, fed by non-Poisson arrivals with random customers' capacities, is considered. Queues with random customers' capacities are useful for analysis and design issues in high-performance computer and communication systems, in which service time and customer volume are the independent quantities (see [3,4] and references therein). For instance, in [3], performance analysis of LTE (Long Term Evolution) networks is carried out in terms of flow-level dynamics and the amount of required radio resources does not depend on the duration of the flow. Such queues are also important in modeling devices, where it is necessary to calculate a sufficient volume of buffer for data storing [5,6]. The results for single-server queues with limited buffer and LIFO (last in, first out) service discipline were presented in [7], where algorithms for the calculation of stationary characteristics were derived.

A new trend in the study of queueing systems is the analysis of the systems with non-Poisson arrivals and nonexponential service time. So, in the works [2,8–11], queues with renewal arrivals, Markovian Arrival processes (MAP), and MMPP are studied under various asymptotic conditions. The main contribution of this paper consists in extending such analysis, focusing on the properties of the bidimensional process describing the number of customers and the total capacity in the system when an infinite-server queue is fed by MMPP arrivals with random capacities and nonexponential service time distribution.

1. Mathematical model

Consider a queue with infinite number of servers (Fig. 1) and assume that customers arrive according to an MMPP.

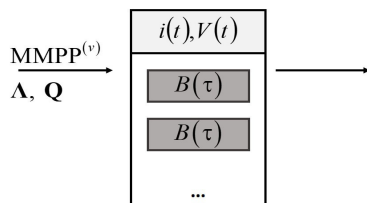


Fig. 1. Queue MMPP/GI/∞ with random customers capacities

The input process is defined by its generator matrix $Q = \|q_{ij}\|$ of size $K \times K$ and the conditional rates $\lambda_1, \dots, \lambda_K$, typically composed into the diagonal matrix $\Lambda = \text{diag}\{\lambda_1, \dots, \lambda_K\}$. Denote the underlying Markov chain of the MMPP as $k(t) \in 1, 2, \dots, K$. Let each customer has some random capacity $v > 0$ with distribution function $G(y)$. An arriving customer instantly occupies a server in the system and its service time has distribution function $B(x)$; when the service is completed, the customer leaves the system. Customers' capacities and service times are mutually independent and do not dependent on the epochs of customers' arrivals.

Denote by $i(t)$ and $V(t)$ the number of customers in the system at time t and their total capacity, respectively. Let us obtain the probabilistic characteristics of two-dimensional process $\{i(t); V(t)\}$. This process is not Markovian; therefore, the dynamic screening method has been used for its investigation.

Consider two time axes that are numbered as 0 and 1 (Fig. 2). Let axis 0 shows the epochs of customers' arrivals, while axis 1 corresponds to the screened process.

Let us introduce a function $S(t)$ (dynamic probability) that satisfies the condition $0 \leq S(t) \leq 1$. Let us assume that a customer, arriving in the system at time t , is screened to axis 1 with probability $S(t)$, and not screened with probability $1 - S(t)$.

Let the system be empty at moment t_0 and let us fix some arbitrary moment T in the future. $S(t)$ represents the probability that a customer arriving at time t will be serviced in the system by the moment T . It is easy to show [11] that $S(t) = 1 - B(T - t)$ for $t_0 \leq t \leq T$.

Denote by $n(t)$ and $W(t)$ the number of arrivals screened before the moment t on axis 1 and their total capacity, respectively.

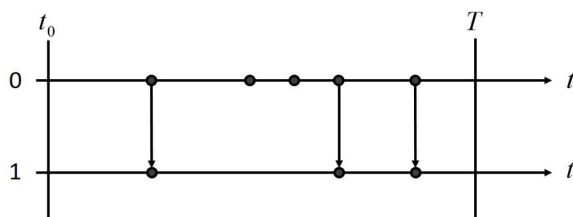


Fig. 2. Screening of the customers' arrivals

As it is shown in [9], the probability distribution of the number of customers in the system at the moment T coincides with the probability distribution of the number of screened arrivals on the axis:

$$P\{i(T) = m\} = P\{n(T) = m\}$$

for all $m = 1, 2, \dots$. It is easy to prove the same property for the extended process $\{i(t), V(t)\}$:

$$P\{i(T) = m, V(T) < z\} = P\{n(T) = m, W(T) < z\} \quad (1)$$

for all $m = 1, 2, \dots$ and $z \geq 0$. Let us use Eqs. (1) for the investigation of the process $\{i(t), V(t)\}$ via the analysis of the process $\{n(t), W(t)\}$.

2. Kolmogorov differential equations

Let us consider the three-dimensional Markovian process $\{k(t), n(t), W(t)\}$. Denoting the probability distribution of this process by $P(k, n, w, t) = P\{k(t) = k, n(t) = n, W(t) \leq w\}$ and taking into account the formula of total probability, one can write the following system of Kolmogorov differential equations:

$$\frac{\partial P(k, n, w, t)}{\partial t} = \lambda_k S(t) \left[\int_0^z P(k, n-1, w-y, t) dG(y) - P(k, n, w, t) \right] + \sum_v P(v, n, w, t) q_{vk}$$

for $k = \overline{1, K}$; $n = 1, 2, \dots$; $w > 0$.

Let us introduce the partial characteristic function:

$$h(k, u_1, u_2, t) = \sum_{n=0}^{\infty} e^{ju_1 n} \int_0^{\infty} e^{ju_2 w} P(k, n, dw, t)$$

where $j = \sqrt{-1}$ is the imaginary unit. Then, one can write the following equations:

$$\frac{\partial h(k, u_1, u_2, t)}{\partial t} = h(k, u_1, u_2, t) \lambda_k S(t) (e^{ju_1} G^*(u_2) - 1) + \sum_v h(v, u_1, u_2, t) q_{vk}$$

for $k = \overline{1, K}$ where $G^*(u) = \int_0^{\infty} e^{juv} dG(y)$.

Let us rewrite this system in the matrix form:

$$\frac{\partial h(u_1, u_2, t)}{\partial t} = h(u_1, u_2, t) [\Lambda S(t) (e^{ju_1} G^*(u_2) - 1) + Q] \quad (2)$$

with the initial condition

$$h(u_1, u_2, t_0) = r \quad (3)$$

where $h(u_1, u_2, t) = [h(1, u_1, u_2, t), \dots, h(K, u_1, u_2, t)]$ and $r = [r(1), \dots, r(K)]$ represents the stationary distribution of the underlying Markov chain, i.e., vector r satisfies the following linear system:

$$\begin{cases} rQ = 0, \\ re = 1, \end{cases} \quad (4)$$

where e is the column vector with all entries equal to 1.

3. Asymptotic analysis

In general, the exact solution of Eq. (2) is not available, but it may be found under asymptotic conditions. In this paper, the case of infinitely growing service time is considered.

Denote by

$$b_1 = \int_0^{\infty} x dB(x) = \int_0^{\infty} (1 - B(x)) dx$$

the mean service time; then, the asymptotic condition is $b_1 \rightarrow \infty$.

Let us solve Problem (2)–(3) under such asymptotic condition and we obtain approximate solutions with different order of accuracy, named as “first-order asymptotic” $h(u_1, u_2, t) \approx h^{(1)}(u_1, u_2, t)$ and “second-order asymptotic” $h(u_1, u_2, t) \approx h^{(2)}(u_1, u_2, t)$.

3.1. First-order asymptotic analysis

Let us formulate and prove the following statement.

Lemma. *The first-order asymptotic characteristic function of the probability distribution of the process $\{k(t), n(t), W(t)\}$ has the form:*

$$h^{(1)}(u_1, u_2, t) = r \exp \left\{ \left(ju_1 k_1 + ju_2 k_1 a_1 \right) \int_{t_0}^t S(v) dv \right\}$$

where $k_1 = r\Lambda e$ and $a_1 = \int_0^\infty y dG(y)$ is the mean customer capacity.

Proof. By performing the substitutions

$$\varepsilon = \frac{1}{b_1}, \quad \varepsilon t = \tau, \quad \varepsilon t_0 = \tau_0, \quad u_1 = \varepsilon x_1, \quad u_2 = \varepsilon x_2, \quad S(t) = S_1(\tau), \quad h(u_1, u_2, t) = f_1(x_1, x_2, \tau, \varepsilon)$$

in expressions (2) and (3), one obtains

$$\varepsilon \frac{\partial f_1(x_1, x_2, \tau, \varepsilon)}{\partial \tau} = f_1(x_1, x_2, \tau, \varepsilon) \left[\Lambda S_1(\tau) (e^{j\varepsilon x_1} G^*(\varepsilon x_2) - 1) + Q \right] \quad (5)$$

with the initial condition

$$f_1(x_1, x_2, \tau_0, \varepsilon) = r. \quad (6)$$

Let us find the asymptotic solution of Problem (5)–(6) $f_1(x_1, x_2, \tau) = \lim_{\varepsilon \rightarrow 0} f_1(x_1, x_2, \tau, \varepsilon)$ in two steps.

Step 1. Let $\varepsilon \rightarrow 0$ in (5)–(6); then, one obtains the following system of equations:

$$\begin{cases} f_1(x_1, x_2, \tau) Q = 0; \\ f_1(x_1, x_2, \tau_0) Q = r. \end{cases}$$

Taking into account (4), one can conclude that $f_1(x_1, x_2, \tau)$ can be expressed as

$$f_1(x_1, x_2, \tau) = r \Phi_1(x_1, x_2, \tau) \quad (7)$$

where $\Phi_1(x_1, x_2, \tau)$ is some scalar function which satisfies the condition

$$\Phi_1(x_1, x_2, \tau_0) = 1. \quad (8)$$

Step 2. Let us multiply (5) by vector e , substitute (7), divide the result by ε , and perform the asymptotic transition $\varepsilon \rightarrow 0$. Then, taking into account that $Qe = 0$ and $re = 1$, one obtains the following differential equation for the function $\Phi_1(x_1, x_2, \tau)$:

$$\frac{\partial \Phi_1(x_1, x_2, \tau)}{\partial \tau} = \Phi_1(x_1, x_2, \tau) S_1(\tau) (jx_1 k_1 + jx_2 k_1 a_1). \quad (9)$$

The solution of Problem (8)–(9) is as follows:

$$\Phi_1(x_1, x_2, \tau) = \exp \left\{ (jx_1 k_1 + jx_2 k_1 a_1) \int_{\tau_0}^{\tau} S_1(v) dv \right\}.$$

Substituting this expression into (7), one obtains

$$f_1(x_1, x_2, \tau) = r \exp \left\{ (jx_1 k_1 + jx_2 k_1 a_1) \int_{\tau_0}^{\tau} S_1(v) dv \right\}.$$

Therefore, one can write

$$h^1(u_1, u_2, t) = f_1(x_1, x_2, \tau, \varepsilon) \approx f_1(x_1, x_2, \tau) = r \exp \left\{ (jx_1 k_1 + jx_2 k_1 a_1) \int_{\tau_0}^{\tau} S_1(v) dv \right\} =$$

$$= r \exp \left\{ (ju_1k_1 + ju_2k_1a_1) \int_{\tau_0}^{\tau} S_1(v) dv \right\}.$$

Thus, the proof is complete.

3.2. Second-order asymptotic analysis

The main result is the following theorem.

Theorem. *The second-order asymptotic characteristic function of the probability distribution of the process $\{k(t), n(t), W(t)\}$ has the form:*

$$h^2(u_1, u_2, t) = r \exp \left\{ (ju_1k_1 + ju_2k_1a_1) \int_{t_0}^t S_1(v) dv \right\} + \frac{(ju_1)^2}{2} \left(k_1 \int_{t_0}^t S(v) dv + k_1 \int_{t_0}^t S^2(v) dv \right) + \frac{(ju_2)^2}{2} \left(k_1a_2 \int_{t_0}^t S(v) dv + k_2a_1^2 \int_{t_0}^t S^2(v) dv \right) + ju_1ju_2 \left(k_1a_1 \int_{t_0}^t S(v) dv + k_2a_1 \int_{t_0}^t S^2(v) dv \right),$$

where $k_2 = 2g(\Lambda - k_1I)e$, $a_2 = \int_0^{\infty} y^2 dG(y)$ and the row vector g satisfies the linear matrix system

$$\begin{cases} gQ = r(k_1I - \Lambda), \\ ge = \text{const}. \end{cases}$$

Proof. Let $h_2(x_1, x_2, t)$ be a vector function that satisfies the equation:

$$h(u_1, u_2, t) = h_2(u_1, u_2, t) \exp \left\{ (ju_1k_1 + ju_2k_1a_1) \int_{t_0}^t S(v) dv \right\}. \quad (10)$$

Substituting this expression into (2) and (3), one obtains the following problem:

$$\frac{\partial h_2(u_1, u_2, t)}{\partial t} = h_2(u_1, u_2, t) \left[(e^{ju_1} G^*(u_2) - 1) S(t) \Lambda - (ju_1k_1 + ju_2k_1a_1) S(t) I + Q \right] \quad (11)$$

with the initial condition

$$h_2(u_1, u_2, t_0) = r \quad (12)$$

where I is the identity matrix.

Let us make the substitutions:

$$\varepsilon^2 = \frac{1}{b_1}; \quad \varepsilon^2 t = \tau; \quad \varepsilon^2 t_0 = \tau_0; \quad u_1 = \varepsilon x_1; \quad u_2 = \varepsilon x_2; \quad (13)$$

$$S(t) = S_1(\tau); \quad h_2(u_1, u_2, t) = f_2(x_1, x_2, \tau, \varepsilon).$$

Using these notations, Problem (11)–(12) can be rewritten in the form

$$\varepsilon^2 \frac{\partial f_2(x_1, x_2, \tau, \varepsilon)}{\partial \tau} = f_2(x_1, x_2, \tau, \varepsilon) \left[\Lambda S_1(\tau) (e^{j\varepsilon x_1} G^*(\varepsilon x_2) - 1) - (j\varepsilon k_1 x_1 + j\varepsilon k_1 x_2 a_1) S_1(\tau) I + Q \right] \quad (14)$$

with the initial condition

$$f_2(x_1, x_2, \tau_0, \varepsilon) = r. \quad (15)$$

Let us find the asymptotic solution of this problem $f_2(x_1, x_2, \tau) = \lim_{\varepsilon \rightarrow 0} f_2(x_1, x_2, \tau, \varepsilon)$ in three steps.

Step 1. Letting $\varepsilon \rightarrow 0$ in (14)–(15), one obtains the following system of equations:

$$\begin{cases} f_2(x_1, x_2, \tau)Q = 0; \\ f_2(x_1, x_2, \tau_0) = r. \end{cases}$$

Therefore, taking into account (4), one can write:

$$f_2(x_1, x_2, \tau) = r\Phi_2(x_1, x_2, \tau) \quad (16)$$

where $\Phi_2(x_1, x_2, \tau)$ is some scalar function which satisfies the condition

$$\Phi_2(x_1, x_2, \tau_0) = 1. \quad (17)$$

Step 2. Using (16), the function $f_2(x_1, x_2, \tau)$ can be represented in the expansion form:

$$f_2(x_1, x_2, \tau, \varepsilon) = \Phi_2(x_1, x_2, \tau) [r + gS_1(\tau)(j\varepsilon x_1 + j\varepsilon x_2 a_1)] + O(\varepsilon^2) \quad (18)$$

where g is the row vector that satisfies the condition $ge = \text{const}$ and $O(\varepsilon^2)$ is the row vector of the secondorder infinitesimals. Let us use substitution (18) and the expansion $e^{j\varepsilon x} = 1 + j\varepsilon x + O(\varepsilon^2)$.

In Eq. (14). Taking into account (4) and making the transition $\varepsilon \rightarrow 0$, one obtains the following matrix equation for the vector g : $gQ = r(k_1 I - \Lambda)$.

Step 3. Let us multiply Eq. (14) by vector e and use expression (18) and the second-order expansion: $e^{j\varepsilon x} = 1 + j\varepsilon x + \frac{(j\varepsilon x)^2}{2} + O(\varepsilon^3)$.

After some transformations, using the notation $k_2 = 2g(\Lambda - k_1 I)e$, one obtains the following differential equation for the function $\Phi_2(x_1, x_2, \tau)$:

$$\begin{aligned} \frac{\partial \Phi_2(x_1, x_2, \tau)}{\partial \tau} = \Phi_2(x_1, x_2, \tau) & \left[\frac{(jx_1)^2}{2} (k_1 S_1(\tau) + k_2 S_1^2(\tau)) + \frac{(jx_2)^2}{2} (k_1 a_2 S_1(\tau) + \right. \\ & \left. + k_2 a_1^2 S_1^2(\tau)) + jx_1 jx_2 (k_1 a_1 S_1(\tau) + k_2 a_1 S_1^2(\tau)) \right]. \end{aligned}$$

The solution of this equation with initial condition (17) is as follows:

$$\begin{aligned} \Phi_2(x_1, x_2, \tau) = \exp & \left\{ \frac{(jx_1)^2}{2} \left(k_1 \int_{\tau_0}^{\tau} S_1(v) dv + k_2 \int_{\tau_0}^{\tau} S_1^2(v) dv \right) + \frac{(jx_2)^2}{2} \left(k_1 a_2 \int_{\tau_0}^{\tau} S_1(v) dv + \right. \right. \\ & \left. \left. + k_2 a_1^2 \int_{\tau_0}^{\tau} S_1^2(v) dv \right) + jx_1 jx_2 \left(k_1 a_1 \int_{\tau_0}^{\tau} S_1(v) dv + k_2 a_1 \int_{\tau_0}^{\tau} S_1^2(v) dv \right) \right\}. \end{aligned}$$

Substituting this expression in formula (16) and performing the substitutions that are inverse to (13) and (10), one obtains

$$\begin{aligned} h^2(u_1, u_2, t) = r \exp & \left\{ (ju_1 k_1 + ju_2 k_1 a_1) \int_{t_0}^t S(v) dv + \frac{(ju_1)^2}{2} \left(k_1 \int_{t_0}^t S(v) dv + k_2 \int_{t_0}^t S^2(v) dv \right) + \right. \\ & \left. + \frac{(ju_2)^2}{2} \left(k_1 a_2 \int_{t_0}^t S(v) dv + k_2 a_1^2 \int_{t_0}^t S^2(v) dv \right) + ju_1 ju_2 \left(k_1 a_1 \int_{t_0}^t S(v) dv + k_2 a_1 \int_{t_0}^t S^2(v) dv \right) \right\} \end{aligned}$$

for the asymptotic characteristic function of the process $\{k(t), n(t), W(t)\}$. The proof is complete.

Corollary. Assuming $t = T$ and $t_0 \rightarrow \infty$ and using Eqs. (1), one obtains the steady-state characteristic function of the process under study $\{i(t), V(t)\}$:

$$h(u_1, u_2) = \exp \left\{ \left(ju_1 k_1 b_1 + ju_2 k_1 a_1 b_1 \right) + \frac{(ju_1)^2}{2} (k_1 b_1 + k_2 b_2) + \frac{(ju_2)^2}{2} (k_1 a_2 b_1 + k_2 a_1^2 b_2) + \right. \\ \left. + ju_1 ju_2 (k_1 a_1 b_1 + k_2 a_1 b_2) \right\}. \quad (19)$$

where $b_1 = \int_0^{\infty} (1 - B(v)) dv$, $b_2 = \int_0^{\infty} (1 - B(v))^2 dv$.

From the form of the characteristic function (19), it is clear that the probability distribution of the two dimensional process $\{i(t), V(t)\}$ is asymptotically Gaussian with vector of means $a = [k_1 b_1, k_1 a_1 b_1]$ and covariance matrix

$$K = \begin{bmatrix} \sigma_1^2 & K_{12} \\ K_{12} & \sigma_2^2 \end{bmatrix} = \begin{bmatrix} k_1 b_1 + k_2 b_2 & k_1 a_1 b_1 + k_2 a_1 b_2 \\ k_1 a_1 b_1 + k_2 a_1 b_2 & k_1 a_2 b_1 + k_2 a_1^2 b_2 \end{bmatrix}.$$

Therefore, the correlation coefficient is given by

$$r = \frac{K_{12}}{\sigma_1 \sigma_2} = \frac{k_1 a_1 b_1 + k_2 a_1 b_2}{\sqrt{k_1 b_1 + k_2 b_2} \sqrt{k_1 a_2 b_1 + k_2 a_1^2 b_2}}.$$

Conclusion

In the paper, the queue with MMPP arrivals, infinite number of servers, and nonexponential service time is considered. Moreover, random customers' capacities, independent of their service time, are assumed. The analysis is performed under the asymptotic condition of an infinitely growing service time. It is shown that two-dimensional probability distribution of customers' number and total capacity in the system is two-dimensional Gaussian under this asymptotic condition. Numerical results show that asymptotic results have enough accuracy for the marginal distributions of number of customers and of the total capacity in the system when the service rate exceeds the fundamental rate of arrivals by at least 15 times.

REFERENCES

1. Mandjes M. Large deviations of Gaussian queues. – Chichester: Wiley, 2007. – 340 p.
2. Melikov A., Zadiranova L., Moiseev A. Two asymptotic conditions in queue with MMPP arrivals and feedback // Comm. Com. Inf. Sc., 2016. – 678. – P.231–240. – doi: 10.1007/978-3-319-51917-3 21.
3. Naumov V., Samouylov K., Sopin E., Andreev S. Two approaches to analyzing dynamic cellular networks with limited resources // 6th Congress (International) on Ultra Modern Telecommunications and Control Systems and Workshops. – St. Petersburg. – 2015. P. 485–488. – doi: 10.1109/ICUMT.2014.7002149.
4. Morozov E., Potakhina L., Tikhonenko O. Regenerative analysis of a system with a random volume of customers // Comm. Com. Inf. Sc. – 2016. – 638. – P. 261–272. – doi: 10.1007/978-3-319-44615-8 23.
5. Tikhonenko O.M., Kempa W. Queueing systems with processor sharing and limited memory under control of the AQM mechanism // Automat. Rem. Contr. – 2015. – 76(10). – P. 1784–1796. – doi: 10.1134/S0005117915100069.
6. Naumov V.A., Samuilov K. E., Samuilov A.K. On the total amount of resources occupied by serviced customers // Automat. Rem. Contr. – 2016. – 77(8). – P. 1419–1427.
7. Tikhonenko O.M. Queueing system with processor sharing and limited resources // Automat. Rem. Contr. – 2010. – 71(5). – P. 803–815.
8. Pankratova E. V., Moiseeva S.P. Queueing system MAP/M/1 with n types of customers // Comm. Com. Inf. Sc. – 2014. – 487. – P. 356–366.
9. Moiseev A., Nazarov A. Tandem of infiniteserver queues with Markovian arrival process // Comm. Com. Inf. Sc. – 2016. – 601. – P. 323–333. – doi: 10.1007/978-3-319-30843-2 34.
10. Lisovskaya E., Moiseeva S., Pagano M. The total capacity of customers in the infinite-server queue with MMPP arrivals // Comm. Com. Inf. Sc. – 2016. – 678. – P. 110–120. doi: 10.1007/978-3-319-51917-3 11.
11. Moiseev A., Nazarov A. Queueing network MAP/(GI/1)K with high-rate arrivals // Eur. J. Oper. Res. – 2016. – 254. – P. 161–168. – doi: 10.1016/j.ejor.2016.04.011.

SURVEY ON MATHEMATICAL MODELS OF QUEUEING SYSTEMS IN RANDOM ENVIRONMENT

E.V. Zhukova

Tomsk State University

water_melon93@mail.ru

One of the most important point of scientific literature are queueing systems subject with external stochastic influences. For example, Markov modulation and random environment are the part of considerate interest in scientific literature. System breakdowns or arrivals of priority customers (including batch arrivals) are often characterized by such influences. They are the force that provides the behavior of system at a different mode.

In this paper a brief review of papers that are devoted to mathematical models of queueing systems functioning in a random environment is presented.

The infinite-server queue is one of the radical models in queueing theory Its specific property is the existence of an infinite number of servers, so that requests are served right away and there are no waiting times. H.M. Jansen, M.R.H. Mandjes, K. De Turck, S. Wittevrongel in the article [1] studied an infinite-server queue in a random environment and a full large deviation principle (LDP) for the transient number of jobs in the system was proved by the authors. The proof of this LDP includes two substantial aspects, namely the result that the temporary number of jobs in the system has a Poisson distribution with a random parameter and the supposition that the random parameter satisfies an LDP.

In the paper [2] focus is on an extended version of the M/M/1 queue in a n-phase Markovian random environment with disasters that destroy all customers present in the system. The authors applied two methods of analysis, namely (i) probability generating functions and (ii) matrix geometric approach, whereas there was revelation of some relationships between them. Mean queue sizes, mean (conditional and unconditional) waiting times and fraction of customers lost were calculated by the authors, and they investigated two special cases further.

Recently a great deal of attention is paid to studying queueing systems with vacations; for example, the surveys (Doshi) and the monographs (Takagi and Tian and Zhang). Queueing systems with vacations can be characterized by the feature that each time a busy period ends and the systems become vacant, the server starts a vacation of random length of time.

In the paper [3] a discrete-time vacation queue operating in random environment where the time axis is derived into slots with equal length was considered by the J. Li, L. Liu. And the authors showed that it is possible to approximate M/G/1 queue with vacations in random environment by its discrete-time counterpart.

In the article [4] mathematical models of systems and queueing networks in a random environment were considered in queueing theory in connection with the plenty of various applications to transport models and systems with a hysteresis control strategy. However, issues of ergodicity in models of this type have usually been solved in the form of only sufficient, and not necessary and sufficient conditions. The methods of obtaining ergodicity criteria for queueing systems in a random environment proposed in the article can be extended from a single-channel system to other systems.

G.L. Choudhury, A. Mandelbaum, M.I. Reiman, W. Whitt A in their paper [5] noticed that significant feature of many complicated systems is the existence of different behavior on various time scales. In the context of stochastic models, various time scales can be represented by nearly completely-decomposable (NCD) Markov chains. Once by the authors was identified the connection between the fluid models and the associated queueing models, the authors decided that this fact may not actually be needful to apart consider the fluid models. They pointed out that any Markov-modulated fluid model is a limit of MMPP/G/1 queues, where G can be some distribution, including exponential (M). The MMPP/M/1 model is especially amenable because its queue-length process is a quasi-birth-death (QBD) process.

In the article [6] the research was directed on the $M/G/\infty$ queue, operating in a Markovian random environment. In this way, the authors considered necessarily to analyze the 2-dimensional stochastic process, which is non-Markovian. The authors studied the case when the service-time distribution of customers which are straight away being served does not turn until the customer is totally serviced, even if the environment has jumped to another state. The authors as well deduced the expressions for its parameters — mean and variance.

In the paper [7] an $M/GI/\infty$ queueing system operating in a semi-Markovian random environment was considered by the authors. Thereby, state transitions are changed by the arrival rate and service-time distribution according to the external semi-Markov process. Using the method of dynamic screening, the authors considered a non-stationary Markov point process $n(t)$ vice of non-Markovian $i(t)$ which is the number of customers in the system. In this way, using to the method of supplementary variable the authors defined the residual sojourn time $z(t)$ in the present state of the environment process $s(t)$ to be capable to analyze it with theory of Markov processes tools. Using derived system of differential equations in terms of vector characteristic functions of the number of customers in the system, the asymptotic analysis of the system in question was conducted by the authors.

Standard mathematical description of the class of models is time homogeneous Markovian vector processes, where the behaviour of one of the queues is represented by each coordinate. This is stated in the article of R. Krenzler, H. Daduna [8]. Product form networks can be characterized by the fact that in steady state (at any fixed time t) the joint distribution of the multi-dimensional (over nodes) queueing process is the product of the stationary marginal distributions of the individual nodes' (non Markovian) queueing processes.

In the paper [9] factors that have influence on the traffic in roadways were considered. In recent years vehicles are the safest while having more power, and the roads have become wider. These and other factors affect on the road traffic, the congestion of vehicles on the road and their speed. Vehicles have higher speed, and this may cause a road section to contain more vehicles every second. Accordingly, a large number of vehicles can provoke incidents on the roadway: they have to minify their speed, or they may not have ability to pass the road section at all. In this research, two kinds of queueing models which describe the traffic flow on a roadway were proposed by the authors. Each model have queue phases, where the various phases represent different situations of the traffic condition and flow. The authors study cases where every phase is characterized either by an $M/M/\infty$ queue or by an $M/M/1$ queue, and investigate the corresponding solutions.

In [10] the author studied a special case where $\frac{\lambda_i}{\mu_i} = c$ for every phase i , author identifies

a full reconcilability which appears between-phase $M/M/\infty$ model and the regular $M/M/\infty$ queue. In this work proposals were presented of application a general approach for cases with $n \geq 2$ phases. Considering road traffic, the author precisely revealed the dependence which appears between the transition rates between phases and the number of vehicles in the system. Respectively, considering safety requirements of distance between vehicles on the road have dependence of their speed, one can find the optimum driving speed for a road section, considering incidents which might happen while the vehicles are driving along a given road section.

In the paper [11] written by O.J. Boxma studied an $M/G/1$ queue in a multi-phase random environment with disasters. Using the supplementary variable technique, the author identified a Markov process and gained the probability generating function of the stationary queue size at an arbitrary epoch. Performance measures such as the mean system length, the mean length of a cycle and the probability that the system in phase also were obtained by the author. Moreover, the author derived the Laplace – Stieltjes transform (LST) of the stationary sojourn time distribution of an arbitrary customer and the server's working time in a cycle, relatively.

Lastly, some numerical examples whose purpose is to show the effect of parameters on the mean system length were given by the authors.

Subject of the article of S. Sophia, B. Praba [12] is an M/M/1 queue with the special feature that the speed of the server alternates between two constant values s_L and s_H . The high-speed periods have exponential distribution, and the low-speed periods have a regularly varying distribution. By the authors was obtained evident asymptotics for the tail of the workload distribution. The two cases in which the offered traffic load is smaller respectively and larger than the low service speed are shown to result in completely different asymptotics.

In this article [13], an M/M/1 queue with vacation in a multi-phase random environment was studied. For this model, by the author was obtained probability generation function (PGF) of the steady-state queue length at arbitrary epoch. Performance measures such as the average queue length, the average length of a cycle, and the probability that the system is in phase that was also presented by the author. Further, the author gained the LST of the steady-state sojourn time distribution of an arbitrary customer and the server's working time distribution in a cycle, respectively. The authors investigated two special cases of that model. Finally, they gained some numerical examples to demonstrate the impact of parameters on the average queue length and the average sojourn time, accordingly.

In the paper of B. D'Auria it's shown that using a matrix-geometric approach it is possible to resolve the problem of detection the factorial moments of the random number of customers in an M/M/ ∞ system when it parameters are modulated by a quasi-markovian random environment. By the authors was showed that this is possible by looking at this random variable as the random measure of a bidimensional random set by a modulated Poisson random measure. Like so, the case when the environment includes only 2 states is more deeply investigated and it is shown that explicit formulas are obtainable given that one state has exponential sojourn times. It is then colorable to believe that for this last case it would be possible to get an explicit expression for the complete characteristic function.

The authors in [14] considered an M/G/ ∞ queue where customers arrive according to a Poisson process with constant rate λ and require independent service, σ , that is distributed according to a general probability function $G(d\sigma)$. The queue is supposed to operate in a random environment $\Gamma(t)$ that is a non-negative stochastic process independent of the arrival process. They were noticed the space of the sample paths of $\Gamma(t)$ by G . The authors assumed that at time t all servers operate at the same speed, $v(\Gamma(t))$, that is a function of the random environment. In particular they supposed that $v: [0, \infty] \rightarrow [0, 1]$. Before proving the stochastic decomposition formula for the M/G/ ∞ queue, the authors first proved the stability of the queue in a more general setting. Also they supposed that the queue is of G/G/ ∞ kind, whose input is a stationary marked point process with intensity λ , and whose service speeds depend on the stationary and ergodic process $\Gamma(t)$.

REFERENCES

1. Jansen H.M., Mandjes M.R.H., De Turck K., Wittevrongel S. A large deviations principle for infinite-server queues in a random environment // Queueing Syst. – 2016. – 82. – P. 199–235.
2. Paz Noam, Yechiali Uri An M/M/1 Queue in Random Environment with Disasters // Asia Pacific Journal of Operational Research. – 2014.
3. Li Jianjun, Liu Liwei On the Discrete-Time Geo/G/1 Queue with Vacations in Random Environment // Discrete Dynamics in Nature and Society. – 2016.
4. Thithiashvili G.Sh. Ergodicity of single-channel queueing system in random environment // Bulletin of Tomsk State University. – 2013. – № 3.
5. Choudhury G.L., Mandelbaum A., Reiman M.I., Whitt W. Fluid and diffusion limits for queues in slowly changing environments // Stochastic Models. – 1997. – 13. – P. 121–146.
6. Nazarov A., Baymeeva G. The M/G/ ∞ Queue in Random Environment // CCIS. – 2014. – 487. – P. 312–324.
7. Nazarov A., Baymeeva G. The M/GI/ ∞ System Subject to Semi-Markovian Random Environment // CCIS. – 2015. – 564. – P. 128–140.

8. *Krenzler R., Daduna H.* Loss systems in a random environment // *Probability.* – 2013.
9. *Paz N.* Markovian Queues in Random Environment with System Failures // *Oper. Res. Lett.* – 2007. – 35. – P. 805–812.
10. *Jiang T., Liu L., Li J.* Analysis of the M/G/1 queue in multi-phase random environment with disasters // *J. Appl. Prob.* – 2015. – 18. – 236–244.
11. *Boxma O.J.* The M/M/1 queue in a heavy-tailed random environment // *Statistica Neerlandica.* – 2000. – 54. – P. 221–236.
12. *Sophial S., Praba B.* The M/M/1 queue in a heavy-tailed random environment // *International Journal of Pure and Applied Mathematics.* – 2015. – P. 267–279.
13. *Zaiming Liu I., Senlin Yu.* The M/M/C queueing system in a random environment // *Journal of Mathematical Analysis and Applications.* – 2016. – 436. – P. 556–567.
14. *Jianjun Li, Liwei Liu.* Performance analysis of a complex queueing system with vacations in random environment // *Advances in Mechanical Engineering.* – 2017. – 9(8). – P. 1–9.

SIMPLE MODEL – A WAY TO INTEGRATE SOFTWARE DEVELOPMENT PROCESSES AND PROJECT MANAGEMENT SOFTWARE

D.O. Zmeev, D. Sokolov, O.A. Zmeev

Tomsk State University

denis.zmeev@accounts.tsu.ru, danila.sokolov@accounts.tsu.ru, ozmeyer@gmail.com

Introduction

In the present condition of the industrial software development project managers have a special role. The manager's role consists in managing a project, which implies monitoring its progress, risk analysis and identifying the key stages of development, building and managing a team, documentation and other activities.

To make the work of the manager easier there are many manuals, reference books, knowledge bases, that describe the project activities in quite different ways. The manager can approach studying those sources more competently to solve problems more efficiently. Many of the difficulties that project managers face are not exclusive to their subject area. For example, a skyscraper architect, a head of a department developing an operating system or even a student studying project management can face similar problems of motivating the team members [6]. But, naturally, there is a great influence of the subject area on project management. In software development this influence resulted in so-called software development process (SDP).

An SDP is effectively a set of instructions that should help organize teamwork in the most efficient way to reach the goals of a project. These instructions can range from a set of abstract principles describing the main work stages and correlation with the tasks to strict rules that specify the order of artefacts that should be added at each step of the working process of the project team. Practical application of the SDP has shown that there is no ideal and universal process. Every one of them has its advantages and disadvantages and is applicable for a certain group of projects. A project manager in software development uses either an existing SDP (it can be adapted for a certain project team) or tries to manage work relying on personal experience (but usually his experience is based on methods from different SDPs).

There are different project management software solutions (PM) created to aid managers. Such systems allow managers and their teams to formalize the work on the projects. At present there are many examples of software solutions the most popular of which are Microsoft Project [4], Team Foundation Server [5], Redmine [2] and Jira [3]. Some of the PM are targeted for specific development processes which means that their features focus on the specific aspects of a certain development process (for example, digital Kanban board for marking the progress of tasks). Nonetheless it is evident that regardless of the system (even if it is created for specific purposes) the observance of the development process in the software development project is the professional responsibility of the manager. That leads to the following problems:

1. Following the rules of an SDP in a PM leads to indirect costs (Which can be expressed as price for a processes-based PM or time to configure the PM according to the SDP rules). The more formalized process is the higher the costs.
2. Accordance of the SDP and the real process used by the team depends on the people involved. Often it is the case that the manager believes that he follows a certain method of an SDP when in fact he does not.

This work consists of the follows parts: first, we describe the main idea of the simple model of the SDP in the Section 1. The example of using the simple model of the SDP for the Feature Driven Development [1] software process and its implementation to Redmine project management software is discussed in the Sections 2 and 3. Finally, in the Conclusin the conclusions and plans for the future work are presented.

1. Simple model of SDP

We develop a simple model (not a meta-model) of an SDP for designing the solution which allow to integrate an SDP and PM. Such model allows to formalize basic interaction between different entities in an SDP. Additional advantage of this simplified approach is the possibility to avoid using specific terms and usual tools for different SDPs. A simple model of SDP is shown in terms of a UML class diagram in the Fig. 1. The suggested model does not represent the full description of the development process because its main purpose is to structure the main working entities which compose the SDP. Using the simple structure allows to transform the SDP into a sequence of tasks implemented in the PM and makes possible to develop automatic tools for such kind of implementation. This feature distinguishes this model from other ways of describing an SDP.

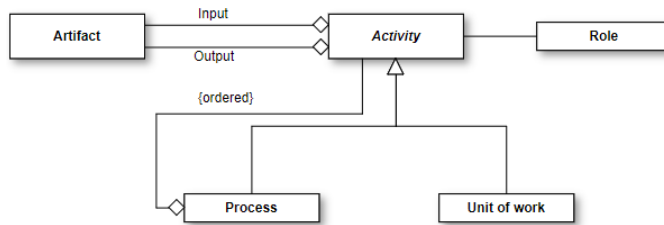


Fig. 1. Simple model of SDP

This model is based on the composite pattern, which allows creating hierarchical structures from elements. These elements are the entities: unit of work, process and activities. In this case the basic element (leaf for composite pattern) is the unit of work that in the implementation of the development process transforms into a task. The aggregating entity (composite pattern) [9] is the process that depending on the difficulty of the SDP structure can be the whole project (if it has no complicated hierarchical structure of actions) or can be an iteration, sprint, phase, etc. (if the SDP implies the using of such entities). Characteristically for the composite pattern while creating the aggregation of entities the order of their addition should be noted (in most developing processes the tasks follow each other in a certain sequence and it is impossible to run test-case before building the system).

Other important entities are artifacts and roles. An artifact denotes an activity in any SDP that is aimed at creation and modifying the output set of artifacts (code, project approval documents, terms of reference, working artifacts for the system design, etc.) from the input set of artifacts (which can be empty, especially at the early stages of the project, for example, collecting the primary requirements for the project). A role denotes a set of responsibilities in a certain activity in the SDP (architect, test engineer, developer, etc.), by assigning a task the project manager assigns a certain role to the team member. In case of the structured development process the roles are defined precisely and in detail, but nonetheless at different points in time any one team member can perform different roles. And if there is no role-based structure there are two default roles in the SDP: the manager and the team member.

Finally, to establish the interaction between the theoretical SDP and the actual implementation of the SDP to the project object-oriented programming dichotomy can be used. If this approach is applied to the development process the abstract activities, iterations, phases described in the development process (classes) implement in real projects into specific tasks, categories and labor costs that are characteristic for projects (objects). Due to this notion it is sufficient to describe the development process with only fundamentally different entities (the concept of a sprint in general), which in practice will be used to describe specific elements of the PM within which the project is implemented.

2. FDD expression using simple model of SDP

To implement the idea of integrating a simple model of an SDP and a PM firstly you need to design the structure of the selected software development process. In this article we use the Feature Driven Development (FDD). The FDD process is based on the main entity – feature and organizes all the work of a team to form, analyze, design, implement and promote features to the software. It is one of the Agile-based development processes, so it is also has the sprint entity to formalize the time-management part of the software project management. The structure of the FDD in terms of the simple model of SDP is as follows (Fig. 2 shows a UML class diagram describing the main entities related to the feature used at FDD for task management).

There are two entities that are used for task management at FDD: a top-level activity and a specific task. As for the roles which are not presented at Fig. 2, the FDD does not formalize the strict model of roles, one of the main ideas is to separate the whole team into small mixed groups (team members with different competencies) and distribute features between them. As for the artifacts we do not show them for clarity (they make the diagram overloaded) and despite each task has a certain type of artifacts in the FDD, we do not use them for implementation into the PM because each team has their own template for different artifacts. The main part of the FDD is to divide the whole sprint into 5 main activities and determine the default task for executing for each activity. Usually, this task transforms to task related to a specific feature which is determined in the first (Develop Overall Model) activity.

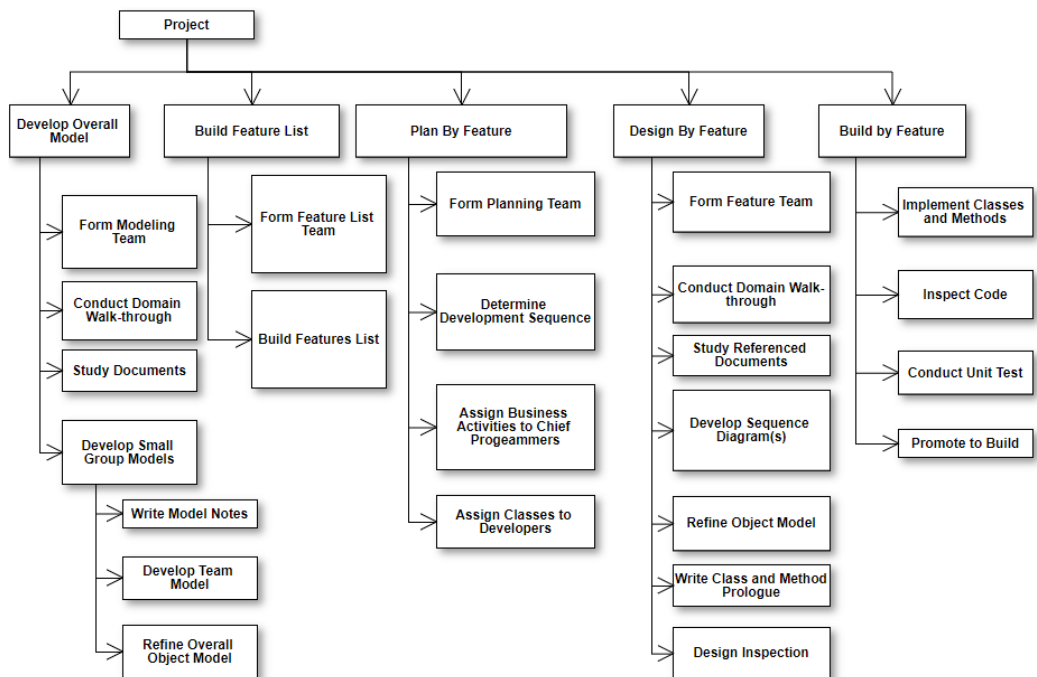


Fig. 2. Simple development process model of FDD

3. Integration between Simple model of FDD and Redmine

At the next step the software development process is integrated with the project management software. To do this, it is necessary to create correspondence between the elements of the development process expressed in the simple model hierarchy and the specific elements of the real project management system (Redmine in this work). The result is shown in Fig. 3.

Let us analyze this correspondence: **Project** is the same entity both in the SDP and the PM. **Sprint** is the entity which can be determined as the best time management practice for the Agile-based development processes correspond to versions mechanism in Redmine which allows the manager to determine time, amount of work, and tasks for each version. **Activities** is the general description for different stages of the sprint, there are two possible solution for it in Redmine, one of them uses specific trackers for each activity and the other is the parent issue feature. Given that the specific trackers make Redmine more limited to a determined implementation of the FDD we choose parent issue feature to express this hierarchy which allows to use other modifications of the FDD and other practices of different SDPs. And finally, the leaf element is **task** for the simple model of the SDP and **issue** for the Redmine system.

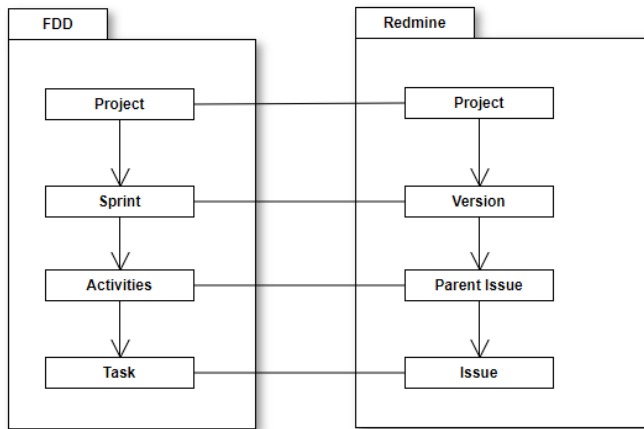


Fig. 3. Implementing the model into Redmine

At the last stage, the Redmine is filled with an empty FDD-prototype project, which allows to create new project based on the FDD development process by copying the prototype instead of repeated filling of projects by the same settings, default initial tasks and usual configuration. The example of the prototype filled “Develop Overall Model” activity as the parent issue is shown in Fig. 4

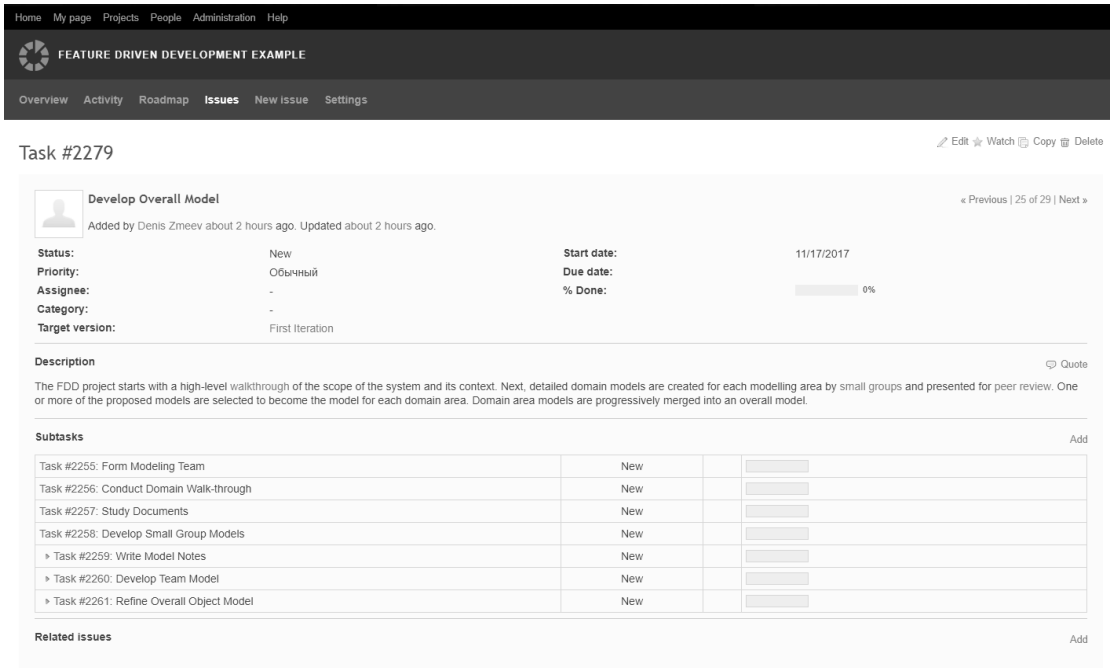


Fig. 4. Example of "Develop Overall Model" activity

It should be noted that the fact that to integrate software development processes and the project management software were used only the native features of Redmine. It makes possible to migrate created prototype development process projects between different teams, companies, Redmine, which allows to apply reusability [8] to the software project management practices.

Conclusion

The simple model allows to express different development processes in understandable way which can be used as hierarchical structure for implementing to different project management software. At the current state of research this model allows to create pre-configured prototype project that allows different teams to use the same implementation of the SDP without spending additional time or financial resources. Future plans include integration of the simple model with the SEMAT [7] initiative which will allow to configure different software development processes as the cooperation of used practices in software engineering.

REFERENCES

- 1 En.wikipedia.org. (2017). Feature-driven development. [online] Available at: https://en.wikipedia.org/wiki/Feature-driven_development [Accessed 29 Nov. 2017].
- 2 Redmine.org. (2017). Overview – Redmine. [online] Available at: <http://www.redmine.org> [Accessed 29 Nov. 2017].
- 3 Atlassian. (2017). Jira | Issue & Project Tracking Software | Atlassian. [online] Available at: <https://www.atlassian.com/software/jira> [Accessed 29 Nov. 2017].
- 4 Products.office.com. (2017). Project Management Software | Microsoft Project. [online] Available at: <https://products.office.com/en/project/project-and-portfolio-management-software> [Accessed 29 Nov. 2017].
- 5 Webster, L. (2017). Agile, Git, CI with TFS | Team Foundation Server. [online] Visual Studio. Available at: <http://www.visualstudio.com/ru/tfs/> [Accessed 29 Nov. 2017].
- 6 Pmi.org. (2017). PMBOK Guide and Standards | Project Management Institute. [online] Available at: <https://www.pmi.org/pmbok-guide-standards> [Accessed 29 Nov. 2017].
- 7 Semat.org. (2017). Welcome - SEMAT. [online] Available at: <http://semat.org/home> [Accessed 29 Nov. 2017].
- 8 En.wikipedia.org. (2017). Reusability. [online] Available at: <https://en.wikipedia.org/wiki/Reusability> [Accessed 29 Nov. 2017].
- 9 En.wikipedia.org. (2017). Composite pattern. [online] Available at: https://en.wikipedia.org/wiki/Composite_pattern [Accessed 29 Nov. 2017].

DATA MINING CLASSIFICATION TECHNIQUES FOR CREDIT SCORING IN BANKS

T.A. Inkhireeva

Tomsk Polytechnic University

tai2@tpu.ru

Introduction

Loan granting is still one of the main services provided by banks. Therefore, credit risk management is crucial for banking and its importance is increasing over time. Credit risk appears when a wrong decision on loan granting is taken. A large number of such a decisions leads to increase in a number of defaulters causing bankruptcy. An example of consequences of credit risk decisions is the USA subprime mortgage crisis of 2007, which developed into an international banking crisis.

Credit scoring models have become more widespread, as banks realized the importance of credit risk management. Credit risk scoring is the method that banking organizations used when analyzing credit data with credit scoring systems to decide whether applicants would be good or bad clients, with the former being able to pay off the debt and the latter having a higher possibility of breaching the contract [1].

Credit scoring methods have been developed to classify customers according to their different risk levels based on available credit history information. Therefore, credit scoring information can be modelled as data mining classification problem [2]. Lots of data mining techniques such as Decision Trees (DT), Logistic Regression (LOGR), Neural Networks (NN), K-Nearest Neighbor (KNN), Bayesian networks (BN), Linear Regression (LR) etc. are used for developing credit scoring models.

In this paper is given an overview of data mining techniques mentioned above and results of comparison discussed advantages and disadvantages of the methods.

Only IEEE Xplore digital library and Google Scholar databases were used. The articles from published journals and conference papers are included. Masters and doctoral theses, working papers, internal reports, textbooks are excluded.

1. Methodological framework

Considering applicability of the Data Mining techniques for building a scoring model, it is necessary to note the possibility of linking the lifecycle stages of a scoring system viewed as a standalone software product and Data Mining work cycle. Both cycles represent a set of repetitive functional steps. Each of the 11 basic lifecycle stages of the scoring model (Fig. 1) can be unambiguously associated with the development stages of the Data Mining model (Fig. 2).

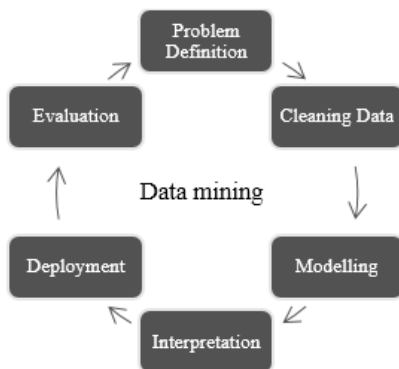


Fig. 1. Data Mining process

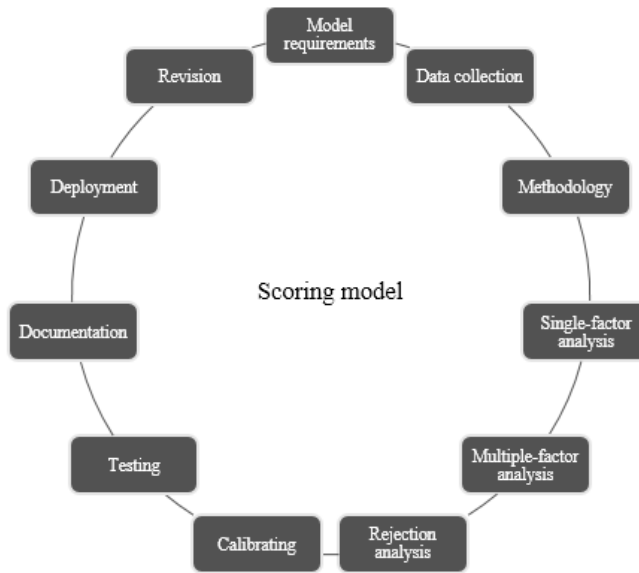


Fig. 2. Scoring model lifecycle

2. Main classification methods in credit scoring

Classification is building categories in feature sets according to according to targets' attributes. Classification models describe features and relationships between categories in order to classify information and assist with decision-making. In this paper will be provided six classification techniques. The techniques can be divided into two categories: Statistical and Artificial Intelligence. DT, LR, LOGR, BN and KNN belong to the Statistical category. NN is the only Artificial Intelligence technique represented in the article

3. Linear regression

The LR is used in credit scoring applications even though the response variable is a two class problem. The technique sets a linear relationship between the characteristics of borrowers $X = \{X_1, \dots, X_p\}$ and the target variable Y ,

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p + \varepsilon.$$

Ordinary least squares is the traditional procedure to estimate coefficients [3].

4. Logistic regression

The LOGR model considers a group of explanatory variables $X = \{X_1, \dots, X_p\}$ and a response variable with two categories $Y = \{Y_1, Y_2\}$. The technique of LOGR consists in the estimation of a linear combination between X and the logit transformation of Y . Thus, if we consider y_1 as the category of interest for analysis, the model can be represented as $\ln\left(\frac{\pi}{1-\pi}\right) = X\beta$, where $\pi = P(Y = y_1)$ and β is the vector containing the model's coefficients [3].

The LOGR model is a traditional method, often compared with other techniques or used in technique combinations. Other possible and particular methods of LOGR are regularized LOGR and limited LOGR.

5. Decision trees

DT is a classification method which uses historical data to construct so-called decision rules organized into tree-like architectures. In general, the purpose of this method is to determine a set of if-then logical conditions that permit prediction or classification of cases. There are three usual tree's algorithms: chi-square automatic interaction detector (CHAID), classification and regression tree (CART) and C5, which differ by the criterion of tree construction, CART uses gini as the splitting criterion, C5 uses entropy, while CHAID uses the chi-square test. A rule based model implementation is used in a stock selection. CHAID and CART are used to verify the segmentation value in the performance capability. A combination of a Bayesian behavior scoring model and a CART-based was proposed for credit scoring model. Other possible and particular methods of decision trees are C4.5 decision trees algorithm and J4.8 decision trees algorithm.

6. Bayesian network

A BN is based on calculating a posterior probability of each observation belongs to a specific class. In other words, it finds the posterior probability distribution $P(Y|X)$, where $Y = (y_1, y_2, \dots, y_k)$ is a random variable to be classified featuring k categories, and $X = (X_1, X_2, \dots, X_p)$ is a set of p explanatory variables. A Bayesian classifier may be seen as a Bayesian network: a directed acyclic graph represented by the triplet (V, E, P) , where V are the nodes, E are the edges and P is a set of probability distributions and its parameters. In this case, the nodes represent the domain variables and edges the relations between these variables.

A conditional Bayesian independence graph is used to extract insightful information on the variables association structure in credit scoring applications. BNs were used in a credit database of annual reports of Czech engineering enterprises.

Possible methods in BNs are naive Bayes, tree augmented naive Bayes and Gaussian naive Bayes.

7. K-Nearest neighbor

KNN classify a data instance by considering only the k -most similar data instances in the training set. The class label is then assigned according to the class of the majority of the k -nearest neighbors. Ties can be avoided by choosing k odd. One commonly opts for the Euclidean distance as the similarity measure:

$$d(x_i, x_j) = \|x_i - x_j\| = \left[(x_i - x_j)^T (x_i - x_j) \right]^{1/2},$$

where x_i, x_j are the input vectors of data instance i and j , respectively. Also more advanced distance measures are used in some cases.

8. Neural networks

A NN is a system based on input variables, also known as explanatory variables, combined by linear and non-linear interactions through one or more hidden layers, resulting in the output variables, also called response variables. NNs were created in an attempt to simulate the human brain, since it is based on sending electronic signals between a huge number of neurons.

The NN structure have elements which receive an amount of stimuli (the input variables), create synapses in several neurons (activation of neurons in hidden layers), and results in responses (output variables). Neural networks differ according to their basic structure. In general, they differ in the number of hidden layers and the activation functions applied to them. There are many NN models suggested for credit scoring: radial basis function NN, a two-stage hybr-

id of the discriminant analysis approach with NN technique, probabilistic NN, partial logistic NN, metaplasticity NN and hybrid NN. Possible particular methods of neural networks are feedforward neural network, multi-layer perceptron, modular neural networks, radial basis function neural networks and self-organizing network.

9. Model comparison

LOGR models are widely used for developing credit scoring models [3,4]. In these studies, the LOGR credit scoring models achieved better in terms of accuracy when LOGR in comparison with other models such as LR, NN, KNN and DTs. LOGR is identified as a good alternative to NN. However, it has weakness due to the model assumption, that independent variables must be linearly related to the logit of the dependent variable [4]. In some datasets, the neural networks have the highest average correct classification rate when compared with other traditional techniques, such as discriminant analysis and logistic regression, taking into account the fact that results were very close.

DTs, rule-based classifiers and any other rule extraction techniques are welcomed in the credit scoring and banking industry because of their explicit conditions in accepting/rejecting applicants so that they are easily understandable by business people in comparison with other techniques [5].

The choice of the technique is a subjective matter and depends on analyst. The most commonly used technique is NN.

Conclusion

In this paper were considered and compared six classification techniques for credit scoring.

The result of the study demonstrates that NN classifiers perform very well. However, simple LR and LOGR also give good performances. It may indicate that most credit scoring data sets are only slightly non-linear. Although performance of other classification techniques is lower, they may compete with each other.

REFERENCES

1. *Huang S.C., Day M.Y.* A comparative study of data mining techniques for credit scoring in banking // IEEE 14th International Conference on Information Reuse & Integration (IRI), San Francisco, CA. – 2013. – P. 684–691.
2. *Kambal E., Osman I., Taha M., Mohammed N., Mohammed S.* Credit scoring using data mining techniques with particular reference to Sudanese banks // International Conference On Computing, Electrical And Electronic Engineering (ICCEEE). – Khartoum, 2013. – P. 378–383.
3. *Louzada F., Ara A., Fernandes G.B.* Classification methods applied to credit scoring: Systematic review and overall comparison // Surveys in Operations Research and Management Science. – 2016. – V. 21. – Is. 2. – P. 117–134.
4. *Baesens B., Seow H.V., Thomas L.C.* Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research // European Journal of Operational Research. – 2015. – V. 247. – Is. 1. – P. 124–136.
5. *Sadatrasoul S.M., Gholamian M.R., Siami M., Hajimohammadi Z.* Credit scoring in banks and financial institutions via data mining techniques: A literature review // Journal of AI and Data Mining. – 2013. – V. 1. – No. 2. – P. 119–129.

THE USE OF «IS-A»-RELATIONS IN INFORMATION TECHNOLOGY

A.M. Babanov, E.S. Kvach

Tomsk State University

babanov2000@mail.ru, kvachelena@gmail.com

Introduction

Any sphere of human activity can not do without using «IS-A»-relations which make it possible to pass from the individual phenomena to their abstractions of different levels and vice versa, for example, «student IS-A person», «bachelor IS-A student». In order to analyze the features of «IS-A»-relations in information systems, various precedents for usage of gene-

ralized and specialized concepts in artificial intelligence, as well as semantic data models and programming the authors were considered.

1. «IS-A»-relations in artificial intelligence

In artificial intelligence, «IS-A»-relations are used in semantic networks. Also, they are considered to be the main elements in the network. The «IS-A» relationship establishes a type hierarchy in the network. Moreover, it is used to define the most specific semantic type [1]. A type hierarchy is a complex network of types or concepts which are organized according to levels of generality, where the concepts get more abstract as one moves up the hierarchy. In a hierarchically structured semantic net, the properties and relations of any given type can be inherited by all of its subtypes [2]. In semantic networks, «IS-A» relationships are distinguished between classes (IS-a type of) and between instances and classes (IS-an instance of) [3]. Fig. 1 shows an example of representation of «IS-A»-relations in semantic networks.

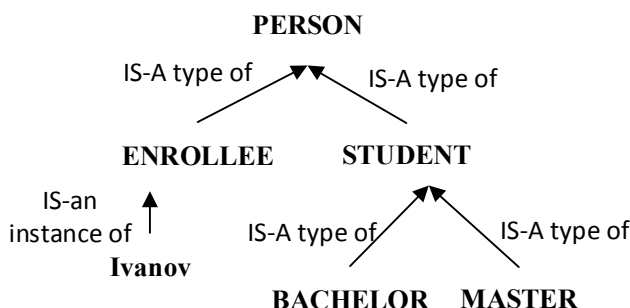


Fig. 1. Representation of «IS-A»-relations in semantic networks

Important integrity constraints are set for «IS-A»-relations: they are transitive, directed, and do not form loops. «Because of the transitivity of the relations, properties may be inherited through many levels of the type hierarchy» [4]. If every bachelor is a student and every student is a person, then every bachelor is a person, whether or not there is «IS-A»-relation explicitly saying so. «Also, we need to check that the hierarchies do not have loops and to always mark either upward or downward from a node or set of nodes, never both at once» [4].

According to Fahlman S.E. «the type hierarchy is the most important single organizational structure in the knowledge network». If information system is drawn (on paper or mentally) as a chaotic tangle of links of all kinds, it is much harder to visualize its behavior as an orderly, structured flow of events [4].

2. «IS-A»-relations in semantic data models

In data modeling the form of the information representation, which is comprehensive for people, it appeared together with the Enhanced Entity-Relationship Model (EER-model) in 80s of the last century [5]. In EER-model, «IS-A»-relation is referred to as «specialization». It is also used to denote the top-down process of «IS-A»- hierarchy formation with full inheritance of characteristics. The reverse process is called generalization. According to data modeling specialists, specialization is:

- «the process of identifying subclasses in the superclass, which is based on grouping unique characteristics and relationships of the subtypes» [5];
- «the process of maximizing the differences between members of an entity» [6];
- «the process of defining a set of subclasses of an entity type on the basis of some distinguishing characteristic of the entities in the superclass» [7];
- «the process of conceptual refinement to form specialized subclasses for entity sets» [8];

– «procedure in which subtypes of a more general object type are introduced to declare that specific roles are recorded only for these subtypes» [9].

Many researchers of the semantic models and the authors, who study them, use the term «specialization» to denote the process of generating subclasses. At the same time it refers to the resulting graph known as a two-level hierarchy, e.g. Terry Halpin [9], the author of Object-Role Model (OR-model), defines specialization as «procedure in which subtypes of a more general object type are introduced», and as «a subtype graph may arise in a top-down way, by specializing an object type into subtypes».

Important concepts accompanying the consideration of «IS-A»-relations are the concepts applied to the elements of these relations.

«If instances of an object type are classified into a more specific type, this specialized type is known as a subtype. A supertype is a more general form of its subtypes, and a subtype is a special form of its supertype(s).» [9].

«Superclass – an entity type that includes one or more distinct subgroupings of its occurrences, which require to be represented in a data model. Subclass – a distinct subgrouping of occurrences of an entity type, which require to be represented in a data model. The relationship between a superclass and a subclass is one-to-one (1:1) and is called a superclass/subclass relationship [6].

Consequently, the main purpose of specialization in data modeling is the possibility of grouping characteristics i.e. generalized ones rise to the level of a superclass, whereas specialized ones descend to the corresponding subclass. It should be noted that data modeling specialists [5–9] can mainly see structural and restrictive features of data classes. In addition, Terry Halpin [9], notes that specializations let make optional superclass roles mandatory for a subclass and thereby introduce additional integrity constraints. Similar to restrictions on «IS-A»-relations in artificial intelligence [4] T. Halpin argues, that specialization is transitive and any pattern of type-subtype relationships forms a directed acyclic graph: «Since no type can be a proper subtype of itself, it follows that no cycles or loops are permitted, no matter what the length of the cycle».

Additional rules for subtypes are introduced by the ER-model in Barker notation in Oracle Designer: mutually exclusive rule (each supertype instance is simultaneously an instance of one and only one subtype) and exhaustive (each supertype instance must be an instance of one of the subtypes) [10]. However, these rules only cover a full disjoint specialization. In addition to this type of specializations:

- full overlapping (each supertype instance necessarily must be an instance of one or more subtypes);
- partial disjoint (a supertype instance may be (or may not be) an instance of no more than one subtype);
- partial overlapping (a supertype instance may be (or may not be) an instance of one or more subtypes)

are also used in many data models.

Fig. 2 shows the options of representations of full disjoint specialization in semantic data models.

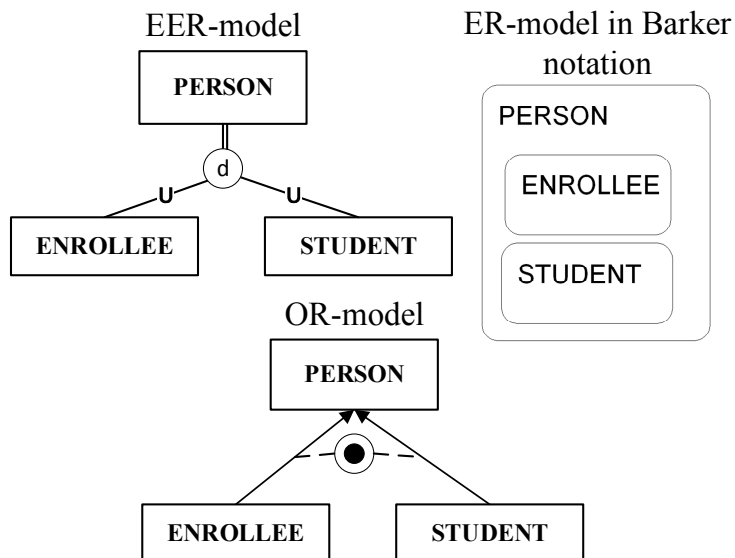


Fig. 2. Options for the representation of «IS-A»-relations in semantic data models

3. «IS-A»-relations in programming

In object-oriented programming (OOP) «IS-A»-relations can be found in the Unified Modeling Language (UML). The term generalization is used to denote:

- «the relationship between a class and one or more of its variations. The generalization unites classes by their general properties» [11];
- «the link between the entity of a general nature and more specific entity» [12];
- «a relationship in which a subclass inherits all the methods and fields of the superclass and can override inherited methods» [13];
- «the process associated with the identification of the common ground between the concepts and the definition of the superclass and its associated subclasses» [14].

A generalization connects generalized classes to more specialized classes, and is therefore called the inheritance relationships (class-subclass or parent-child). Generalization is called the «IS-A»-relations: for example, a student's class is-a kind of a more General entity, a class of a person [12].

In accordance with the features of the OOP, in UML, besides attributes and relations, subtypes inherit also the methods of supertype. Moreover, the implementation of methods in the subtype replaces the implementation of the same supertype method – this phenomenon is called polymorphism [12]. Experts studying UML, additionally introduce the notion of discriminator [13] or a set of generalizations [11,12]. A generalization with a name means the decomposition of a superclass into subclasses in some particular aspect called the generalization set [12]. The name of the generalization set is an enumerable attribute that indicates which aspect of the subclasses is generalized. Each generalization should be made on one aspect. For example, the generalization aspect for a *person* class is the *order for admission*. The values of the generalization set are in one – to-one correspondence with the subclasses (if there is an *order for admission*, then the *person* is a *student*, otherwise *person* is an *enrollee*). Fig. 3 shows an example of representation of «IS-A»-relations UML.

Larman K. [14] proposes two rules for subclasses: compatibility of subclass definitions (100% rule) and compatibility of multiple subclasses (IS-a rule). The 100% rule states that 100% of the superclass definition must be applied to the subclass. The subclass must 100% match its superclass in the following parameters: attributes, associations. This means that the *enrollee* and *student* subclasses must contain the *name* attribute and be associated with the

address class. The rule «IS-A» states that all the elements of a subclass set must be elements of its superclass set. Therefore, subclass *student* should be an element of *people* set.

UML also specifies that there can be no cycles in the inheritance system: no class can be its own ancestor. The generalization relation is asymmetric: the child inherits the parent properties, but the parent does not contain any specific information about its childs [12]. Finally, the generalization is transitive and operates through an arbitrary number of hierarchy levels [11].

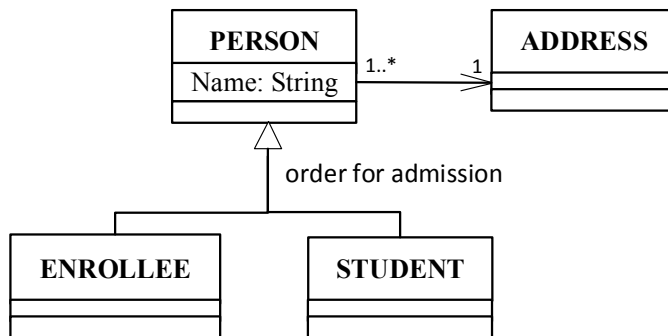


Fig. 3. Representation of «IS-A»-relations in UML

Conclusion

«IS-A»-relations are an important element of human intellectual activity (not only scientific). This notion has been long studied by researchers, however, in each of the presented areas of knowledge different definitions of generalized and specialized concepts are given. Also different integrity constraints and design methods are proposed. Figs. 1–3 show possible variants of representation «IS-A»-relations in information technology.

After analyzing the features of «IS-A»-relations in various information technologies, the authors have concluded there is a need for stricter formalization of generalizations as well as the introduction of a new basic concept of «IS-THE»-relations in the semantic Entity-Relationship-Mapping Model [15].

ACKNOWLEDGMENTS

The authors thank senior teacher L.V. Artamonova for help in translating the report into English.

REFERENCES

1. U.S. National Library of Medicine, The Unified Medical Language System, UMLS Reference Manual, URL: <https://www.ncbi.nlm.nih.gov/books/NBK9676/>
2. Eileen C.W. Knowledge Representation and Metaphor. – Springer Science+Business Media Dordrecht, 1991. – 271 p.
3. Bundy. A. Catalogue of Artificial Intelligence Tools. – Springer-Veriag, Berlin, Heidelberg New York Tokyo, 1984. – 150 p.
4. Fahlman S.E. NETL: A System for Representing and Using Real-World Knowledge.– MIT Press, 1979. – 269 p.
5. Coronel C., Morris S., Rob P. Database Systems: Design, Implementation, and Management. – USA.: Cengage Learning, 2011. – 724 p.
6. Connolly T., Begg C. Database Systems: A Practical Approach to Design, Implementation, and Management, edition 4th. – Addison-Wesley, 2010. – 1400 p.
7. Elmasi R., Navahe S.B. Fundamentals of Database Systems, edition 6th. – Addison-Wesley, 2011. – 1201 p.
8. Ferraggine V.E., Doorn J.H., Rivero L.C. Handbook of Research on Innovations in Database Technologies and Applications: Current and Future Trends. – 2009. – 1124 p.
9. Halpin T., Morgan T. Information Modeling and Relational Databases, edition 2. – Morgan Kaufman, 2008. – 943 p.
10. Barker R. CASE Method: Entity Relationship Modelling. – Addison-Wesley. –1990. – 240 p.
11. Рамбо Д., Блаха М. UML 2.0. Объектно-ориентированное моделирование и разработка. 2-е изд. – СПб.:Питер, 2007. – 544 с.

12. Буч Г., Рамбо Д., Якобсон А. Язык UML. Руководство пользователя. – ДМК Пресс, 2007. – 496 с.
13. Фаулер М., Скотт К. UML основы. Второе издание. – Символ-Плюс, 2002. – 188 с.
14. Ларман К. Применение UML и шаблонов проектирования. 2-е издание Пер. с англ. – М.: Издательский дом «Вильямс», 2004. – 624 с.
15. Бабанов А.М., Квач Е.С. «IS-THE»-отношения в семантических моделях данных: основные понятия и разновидности // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2016. – № 1(34). – С. 69–78.

АВТОРСКИЙ УКАЗАТЕЛЬ

Андреева Валентина Валерьевна.....	17,22
Андреевская Анна Андреевна.....	107
Аношкин Роман Александрович.....	241
Бабанов Алексей Михайлович.....	365
Балашова Ольга Михайловна.....	55
Бесчастный Виталий Александрович.....	334
Бойко Сергей Олегович.....	245
Броч Михаил Ильич.....	14
Бушкова Татьяна Владимировна.....	347
Воробьева Мария Геннадьевна.....	205
Выговская Ольга Анатольевна.....	291
Галилейская Анастасия Александровна.....	295
Гальцева Александра Ивановна.....	257
Головчинер Михаил Наумович.....	34,41,74,77,257
Горлова Анастасия Андреевна.....	195
Гудкова Ирина Андреевна.....	330,334,338
Дарибаева Нурзия Туребековна.....	237
Дмитренко Анатолий Григорьевич.....	55
Дмитриев Юрий Глебович.....	116
Добров Эльдар Робертович.....	68
Жукова Екатерина Викторовна.....	354
Завгородняя Мария Евгеньевна.....	129
Замятин Александр Владимирович.....	278
Зенкова Жанна Николаевна.....	99,103,107,112
Зенкова Наталья Андреевна.....	121
Змеев Денис Олегович.....	357
Змеев Олег Алексеевич.....	357
Зубова Екатерина Игоревна.....	62
Зуевич Екатерина Владимировна.....	226
Иванова Дарья Вадимовна.....	317,321
Инхиреева Татьяна Александровна.....	362
Кабанова Татьяна Валерьевна.....	195
Камагаева Анастасия Валерьевна.....	91
Карнаухова Екатерина Андреевна.....	317,321
Квач Елена Сергеевна.....	365
Кеба Анастасия Владимировна.....	157
Ким Константин Станиславович.....	211

Козлов Алексей Николаевич.....	103
Конев Виктор Васильевич.....	185
Крупко Олеся Сергеевна.....	325
Лившиц Климентий Исаакович.....	200,205
Лисовская Екатерина Юрьевна.....	295
Лопсанчап Буян Орланович.....	164
Макеева Елена Дмитриевна.....	330
Малахова Татьяна Евгеньевна.....	195
Маркова Екатерина Викторовна.....	317,321,330,338
Матросова Анжела Юрьевна.....	3,10
Мизюркина Ольга Игоревна.....	34
Моисеев Александр Николаевич.....	172
Моисеева Светлана Петровна.....	291,303,306,347
Москалева Фаина Александровна.....	325
Назаренко Богдан Николаевич.....	185
Назаров Анатолий Андреевич.....	177,284
Нежелская Людмила Алексеевна.....	140,149,157,164
Никитина Валерия Васильевна.....	268
Орлов Андрей Александрович.....	257
Осенчук Вячеслав Анатольевич.....	77
Острикова Дарья Юрьевна.....	334
Охрименко Александра Сергеевна.....	99
Павлова Анна Илларионовна.....	68
Павлова Екатерина Алексеевна.....	303
Пальянова Юлия Евгеньевна.....	116
Пашкова Наталья Николаевна.....	17
Петренко Екатерина Андреевна.....	22
Петров Евгений Денисович.....	172
Поволоцкая Татьяна Александровна.....	200
Погуда Алексей Андреевич.....	177
Полин Евгений Павлович.....	306
Поляков Никита Алексеевич.....	330
Попеляева Ксения Николаевна.....	38
Попырин Дмитрий Андреевич.....	220
Провкин Виктор Алексеевич.....	10
Пупков Андрей Викторович.....	191
Ревенко Николай Михайлович.....	180
Савич Валерия Николаевна.....	334

Самохина Светлана Ивановна	83,88
Саринова Асия Жумабаевна	278
Серебренникова Кристина Андреевна	112
Сеченова Софья Алексеевна	311
Сидорова Екатерина Филипповна	140
Синицына Анна Алексеевна	338
Скворцов Евгений Олегович	74
Смагин Валерий Иванович	211,216
Смородинова Светлана Сергеевна	216
Соколов Данила Александрович	357
Табольжина Юлия Евгеньевна	129
Томков Владислав Владимирович	38
Тумашкина Диана Александровна	149
Тычинский Вячеслав Зиновьевич	49
Ульянова Екатерина Сергеевна	200
Устинова Вероника Александровна	88
Федоров Александр Викторович	95
Филипова Валерия Романовна	338
Харин Петр Александрович	330
Худяшова Екатерина Евгеньевна	284
Хуракай Долма Май-Ооловна	273
Хусаинова Гузалия Ядкарровна	91,95
Цветкова Оксана Олеговна	342
Цветницкая Светлана Александровна	220
Чаусова Елена Владимировна	121
Чернышов Семен Владимирович	3
Чупрасова Людмила Витальевна	261
Шитина Анна Александровна	133
Шишкин Михаил Максимович	83
Шмаков Ярослав Игоревич	41
Эшмурадов Даур Уралович	231

СОДЕРЖАНИЕ

СЕКЦИЯ I. МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ МАШИН И КОМПЬЮТЕРНЫХ СЕТЕЙ.....	3
Чернышов С.В., Матросова А.Ю. Поиск ложных путей в последовательностных схемах с использованием ROBDD.....	3
Матросова А.Ю., Провкин В.А. О повышении защищённости логических схем от внедрения вредоносных подсхем.....	10
Броч М.И. Построение всех тестовых наборов, обнаруживающих константные неисправности литер ЭНФ.....	14
Пашкова Н.Н., Андреева В.В. Исследование и разработка алгоритмов упрощения КНФ при решении проблемы выполнимости.....	17
Петренко Е.А., Андреева В.В. Статическое тестирование динамических структур данных.....	22
Мизюркина О.И., Головчинер М.Н. Адаптивный нейро-нечеткий классификатор инновационно-технологических решений.....	34
Попеляева К.Н., Томков В.В. Разработка USE-инструмента для создания пользовательского интерфейса.....	38
Шмаков Я.И., Головчинер М.Н. Клеточно-автоматное моделирование распространения звука в задаче поиска оптимального расположения акустических систем.....	41
Тычинский В.З. Разработка и реализация специализированного программного обеспечения для курса «Методы оптимизации».....	49
Дмитренко А.Г., Балашова О.М. Численный метод решения задачи электромагнитного рассеяния на тонких параллельных идеально проводящем и диэлектрическом цилиндрах.....	55
Зубова Е.И. О системе управления аудиометром с реализацией методики тональной аудиометрии и объективной методики ASSR.....	62
Павлова А.И., Добров Э.Р. Разработка экспертной информационной системы для контроля пассажироперевозок.....	68
Скворцов Е.О., Головчинер М.Н. Генератор психодиагностических тестов.....	74
Осенчук В.А., Головчинер М.Н. Разработка автоматизированной системы для проведения психодиагностических исследований.....	77
Самохина С.И., Шишкин М.М. Программный комплекс моделирования двухкомпонентного разложения кардиологической кривой.....	83
Устинова В.А., Самохина С.И. Информационная система анализа состояния здоровья человека «H.TOGETHER».....	88
Хусаинова Г.Я., Камагаева А.В. Разработка пользовательского приложения средствами Borland Delphi.....	91
Хусаинова Г.Я., Федоров А.В. Инфологическое проектирование на примере биржи труда.....	95
СЕКЦИЯ II. ПРИКЛАДНОЙ СТАТИСТИЧЕСКИЙ АНАЛИЗ.....	99
Зенкова Ж.Н., Охрименко А.С. ABC-XYZ анализ ассортимента предприятия г. Томска по случайно цензурированным данным.....	99
Зенкова Ж.Н., Козлов А.Н. ABC-анализ при наличии дефицита.....	103

Андриевская А.А., Зенкова Ж.Н. Метод ценообразования Штоцеля для случайно цензурированных данных.....	107
Зенкова Ж.Н., Серебренникова К.А. Расчет чистой нетто-премии по однократно справа цензурированной выборке	112
Дмитриев Ю.Г., Пальянова Ю.Е. Прогнозирование тенденции максимальной цены акции.....	116
Зенкова Н.А., Чаусова Е.В. Анализ товарного ассортимента торгового предприятия г. Прокопьевска.....	121
СЕКЦИЯ III. ПРИКЛАДНОЙ ВЕРОЯТНОСТНЫЙ АНАЛИЗ.....	129
Завгородняя М.Е., Табольжина Ю.Е. Оценивание параметров простейшего потока событий и мёртвого времени, распределенного по закону Рэлея.....	129
Шитина А.А. Статистические эксперименты на имитационной модели пуассоновского потока событий при продлеваемом случайном мёртвом времени	133
Нежелская Л.А., Сидорова Е.Ф. Статистические эксперименты на имитационной модели обобщённого синхронного потока событий второго порядка с непродлеваемым мёртвым временем.....	140
Нежелская Л.А., Тумашкина Д.А. Статистические эксперименты на имитационной модели полусинхронного потока событий второго порядка в условиях его неполной наблюдаемости.....	149
Кеба А.В., Нежелская Л.А. Статистические эксперименты на имитационной модели обобщённого МАР-потока событий с произвольным числом состояний	157
Лопсанчип Б.О., Нежелская Л.А. Статистические эксперименты на имитационной модели МАР-потока событий с двумя состояниями в условиях продлеваемого мёртвого времени.....	164
Петров Е.Д., Моисеев А.Н. Стохастическое моделирование территории парковки автомобилей.....	172
Погуда А.А., Назаров А.А. Математическая модель обработки текстовой информации в системах тестирования	177
Ревенко Н.М. Система управления ресурсами с кусочно-постоянной скоростью поступления ресурсов и постоянной интенсивностью моментов потребления ресурсов	180
СЕКЦИЯ VI. ИДЕНТИФИКАЦИЯ ДИНАМИЧЕСКИХ СИСТЕМ.....	185
Конев В.В., Назаренко Б.Н. О доверительном оценивании параметров AR(1) процесса с неизвестным средним по зашумлённым наблюдениям.....	185
Пупков А.В. Скорейшее обнаружение разладки авторегрессионных процессов при неизвестной конечной модели	191
Малахова Т.Е., Кабанова Т.В., Горлова А.А. Обработка медицинских данных, содержащих информацию о факторах сердечно-сосудистых заболеваний.....	195
СЕКЦИЯ V. МАТЕМАТИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ СОЦИАЛЬНО-ЭКОНОМИЧЕСКОЙ ДЕЯТЕЛЬНОСТИ	200
Лившиц К.И., Поволоцкая Т.А., Ульянова Е.С. Розничная продажа продукции с ограниченным сроком годности при ММР-потоке моментов продаж.....	200

Воробьева М.Г., Лившиц К.И. Простая аппроксимация вероятности разорения страховой компании для модели Крамера – Лундберга с ММР-потокотом страховых выплат.....	205
Ким К.С., Смагин В.И. Робастная экстраполяция в дискретных системах со случайными скачками с двумя состояниями	211
Смординова С.С., Смагин В.И. Задача управления запасами с учётом транспортных ограничений.....	216
Попырин Д.А., Цветницкая С.А. Управление моделью средней фирмы.....	220
Зуевич Е.В. Исследование нескольких инвестиционных стратегий с использованием короткой и длинной позиции	226
Эшмурадов Д.У. Анализ экономического роста в расширенной модели Солоу с человеческим капиталом	231
Дарибаева Н.Т. Проект корпоративной информационной системы управления полётным расписанием авиакомпаний.....	237
СЕКЦИЯ VI. ГЕОИНФОРМАТИКА. ВЕБ ПРИЛОЖЕНИЯ	241
Аношкин Р.А. Исследование и реализация эффективных средств управления web-контентом	241
Бойко С.О. Создание масштабируемой и отказоустойчивой системы передачи пакетов данных.....	245
СЕКЦИЯ VII. ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ И ВИЗУАЛИЗАЦИЯ ДАННЫХ	257
Гальцева А.И., Орлов А.А., Головчинер М.Н. Оценка эффективности методов прогнозирования в задаче заполнения пропусков	257
Чупрасова Л.В. Построение реалистичных изображений с использованием GLSL-шейдеров OpenGL.....	261
Никитина В.В. Разработка и анализ модели с помощью технологии Process Mining	268
Хуракай Д.М. Анализ журнала событий методами Process Mining.....	273
Саринова А.Ж., Замятин А.В. Методика разработки алгоритмов для сжатия гиперспектральных аэрокосмических изображений, применимых на борту космических аппаратов.....	278
СЕКЦИЯ VIII. ТЕОРИЯ МАССОВОГО ОБСЛУЖИВАНИЯ И ТЕЛЕТРАФИКА	284
Худяшова Е.Е., Назаров А.А. Исследование системы $M GI _{\infty}$ с неординарным входящим потоком и бесконечным значением среднего времени обслуживания	284
Выговская О.А., Моисеева С.П. Асимптотический анализ двухлинейной RQ-системы вида $MMPP M 2$ с нетерпеливыми заявками в условии долгой терпеливости	291
Галилейская А.А., Лисовская Е.Ю. Стационарное асимптотическое распределение вероятностей числа заявок и объёмов занятых ресурсов в неоднородной ресурсной бесконечнолинейной СМО с входящим ММРР-потокотом.....	295
Павлова Е.А., Моисеева С.П. Асимптотический анализ ресурсной гетерогенной СМО вида $MMPP(V1, V2) GI _{\infty}$	303
Полин Е.П., Моисеева С.П. Исследование системы $MMPP(n) M(n) _{\infty}$ методом моментов.....	306
Сеченова С.А. Исследование системы $MMPP GI _{\infty}$ методом моментов	311

Иванова Д.В., Карнаухова Е.А., Маркова Е.В. К анализу модели схемы беспроводной сети LTE с системой LSA в виде СМО с отказами и восстановлениями	317
Иванова Д.В., Карнаухова Е.А., Маркова Е.В. Анализ вероятностных характеристик модели схемы функционирования одной LSA-полосы с нетолерантным к задержкам трафиком	321
Крупко О.С., Москалева Ф.А. Математическая модель звена доступа в сетях связи миллиметрового диапазона для одноадресных и многоадресных соединений	325
Макеева Е.Д., Харин П.А., Поляков Н.А., Маркова Е.В., Гудкова И.А. Анализ установления соединения пар передатчик-приёмник на mmWave	330
Савич В.Н., Бесчастный В.А., Острикова Д.Ю, Гудкова И.А. Модель мобильности в виде СеМО с "пассивным" узлом	334
Синицына А.А., Филипова В.Р., Гудкова И.А., Маркова Е.В. Схема модели беспроводной сети LTE с технологией совместного доступа к нелицензированному диапазону частот Wi-Fi.....	338
Цветкова О.О. Применение модели движения Гаусса – Маркова к анализу интерференции в беспроводной D2D-сети	342
СЕКЦИЯ IX. COMPUTER SCIENCE.....	347
Bushkova T.V., Moiseeva S.P. Study of the MMPP/GI/1 Queueing System with Random Customers' Capacities.....	347
Zhukova E.V. Survey on Mathematical Models of Queueing Systems in Random Environment	354
Zmееv D.O., Sokolov D., Zmееv O.A. Simple Model – a Way to Integrate Software Development Processes and Project Management Software.....	357
Inkhireeva T.A. Data Mining Classification Techniques for Credit Scoring in Banks	362
Babanov A.M., Kvach E.S. The Use of «Is-A»-Relations in Information Technology.....	365
АВТОРСКИЙ УКАЗАТЕЛЬ	371

Научное издание

МАТЕРИАЛЫ
VI Всероссийской молодежной
научной конференции
«МАТЕМАТИЧЕСКОЕ
И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
ИНФОРМАЦИОННЫХ, ТЕХНИЧЕСКИХ
И ЭКОНОМИЧЕСКИХ СИСТЕМ»

Томск, 24–26 мая 2018 г.

Под общей редакцией
кандидата технических наук И.С. Шмырина

Издание подготовлено в авторской редакции

Подписано в печать 15.10.2018 г. Формат 70×108 ¹/₁₆

Печ. л. 23,6; усл.-печ. л. 33.

Тираж 30 экз. Заказ № 3439.

Отпечатано на оборудовании
Издательского Дома
Томского государственного университета
634050, г. Томск, пр. Ленина, 36
Тел. 8+(382-2) 52-98-49
Сайт: <http://publish.tsu.ru>; e-mail: rio.tsu@mail.ru

ISBN 978-5-94621-755-2



9 785946 217552