

А.А. БАТЕНКОВ, К.А. БАТЕНКОВ, А.Б. ФОКИН
**МЕТОДЫ ФОРМИРОВАНИЯ МНОЖЕСТВ СОСТОЯНИЙ
ТЕЛЕКОММУНИКАЦИОННЫХ СЕТЕЙ ДЛЯ РАЗЛИЧНЫХ
МЕР СВЯЗНОСТИ**

Батенков А.А., Батенков К.А., Фокин А.Б. Методы формирования множеств состояний телекоммуникационных сетей для различных мер связности.

Аннотация. Задачи анализа надежности, живучести и устойчивости характерны не только для телекоммуникаций, но и для систем, чьи компоненты подвержены одному или нескольким видам отказов, например транспортные, энергетические, механические системы, интегральные цепи и даже программное обеспечение. Логический подход предполагает декомпозицию системы на ряд небольших функциональных элементов, и в рамках телекоммуникационных сетей они обычно представляют собой отдельные сетевые устройства (коммутаторы, маршрутизаторы, терминалы и т. п.), а также линии связи между ними (медножильные, оптоволоконные, коаксиальные кабели, беспроводная среда и другие среды передачи). Функциональные взаимосвязи задают и логические соотношения между отказами отдельных элементов и отказом сети в целом. Также используется допущение, что отказы устройств являются сравнительно менее вероятными, чем отказы линий связи, что подразумевает использование предположения об абсолютной устойчивости (надежности, живучести) данных устройств. Модель телекоммуникационной сети представлена в виде обобщенной модели Эрдеша – Реньи. В контексте устойчивости телекоммуникационной сети под анализируемым свойством понимается связность сети в той или иной форме. Основываясь на представлении понятия стохастической связности сети как соответствия некоторого случайного графа свойства связности заданному набору вершин, традиционно выделяют три меры связности: двухполюсная, многополюсная и всеполюсная. Представлены процедуры формирования для сетей произвольной структуры множеств путей, деревьев и, как их обобщение, многополюсных деревьев. Отмечено, что многополюсные деревья являются наиболее общим понятием относительно простых цепей и остовых деревьев. Решение подобных задач позволит в дальнейшем перейти к вычислению вероятности связности графов для различных мер связности.

Ключевые слова: сеть связи, граф, структура, вероятность связности, двухполюсная связность, многополюсная связность, всеполюсная связность.

1. Введение. Существенный рост как размеров, так и сложности телекоммуникационных сетей не только приводит к их большей подверженности различного рода отказам, но и выносит на первый план проблемы, связанные с анализом устойчивости (надежности, живучести). При этом подобные системы могут занимать как сравнительно незначительные пространства, например локальные сети, так и иметь габариты, сравнимые с земным шаром, – глобальная сеть Интернет [1].

Отметим, что задачи анализа надежности, живучести и устойчивости характерны не только для телекоммуникаций, но и для систем, чьи компоненты подвержены одному или нескольким видам отказов, например транспортные, энергетические, механические системы, инте-

гральные цепи и даже программное обеспечение. Главный признак данных систем – возможность их представления в форме сети, описывающей взаимосвязь ее отдельных элементов. Так, для телекоммуникаций каждая вершина сети представляет собой отдельное устройство, например, маршрутизатор, коммутатор, сервер и тому подобное, а ребра – линии связи между ними [2, 3]. При этом в зависимости от состояния ребер и вершин (работоспособное или отказ) сама телекоммуникационная сеть может быть работоспособной либо нет.

Для большинства сетей связи постоянная доступность заданного набора вершин является необходимым условием нормального функционирования. Высокая надежность – неотъемлемое требование любой сети [4, 5]. В связи с этим в большинстве случаев сети разрабатываются с учетом наличия сразу нескольких путей распространения информации между любой парой вершин.

В идеальном случае наличие сразу нескольких разнесенных путей между заданной парой вершин приводит к высокой вероятности существования хотя бы одного маршрута передачи. Однако стоимость подобных конструкций оказывается серьезным ограничением на пути их физического воплощения [6, 7]. Как компромисс используют допущение, предполагающее наличие между любой парой вершин нескольких непересекающихся множеств путей, а также маршрутов, включающих общие ребра.

Основной проблемой в данной области является определение надежности подобных сетей, а также задание некоторых количественных показателей требуемых характеристик сети.

Цель настоящей работы – обобщение и доработка отдельных этапов методов формирования множеств состояний телекоммуникационных сетей в условиях произвольных требований наличия взаимосвязи между заданным набором вершин. В качестве основного результата можно выделить формализованное описание методов формирования наиболее часто используемых структур при анализе устойчивости сетей (взаимодействие между парой или всеми узлами) и обобщение данных методов на случай произвольного количества узлов, основывающихся на представлении сети как модели случайного графа Эрдеша – Реньи.

2. Случайная графовая модель телекоммуникационной сети. В настоящее время оценка устойчивости (надежности, живучести) немислима без использования теории графов [8, 9]. При подобном анализе практически невозможно рассмотреть сложную или большую систему во всей ее полноте и богатстве. Логический подход предполагает декомпозицию системы на ряд небольших функциональных элементов, и в рамках теле-

коммуникационных сетей они обычно представляют собой отдельные сетевые устройства (коммутаторы, маршрутизаторы, терминалы и т. п.), а также линии связи между ними (медножильные, оптоволоконные, коаксиальные кабели, беспроводная среда и другие среды передачи). Функциональные взаимосвязи задают и логические соотношения между отказами отдельных элементов и отказом сети в целом. Отметим, что обычно считают отказы устройств сравнительно менее вероятными, чем отказы линий связи, что подразумевает использование предположения об абсолютной устойчивости (надежности, живучести) данных устройств.

Модель телекоммуникационной сети целесообразно представить в виде обобщенной модели Эрдеша – Реньи [10]. В ней рассматривается случайный граф G , в котором жестко задано множество вершин $V = \{1, \dots, v\}$ с мощностью v . Естественно, что случайными в нем являются только ребра, что вполне согласуется с предположением об абсолютной устойчивости сетевых устройств. В классическом подходе анализируются исключительно графы, в которых отсутствуют кратные ребра (мультиграфы), петли (псевдографы) и направленные ребра (орграфы). Всего максимально возможное количество ребер в графе G равно $l = C_n^2$ штук. Вероятность существования ребра (работоспособности линии связи) между вершинами i и j обозначена через $p_{i,j} \in [0,1]$ и не зависит от всех остальных $l-1$ ребер. Отметим, что данная вероятность может рассматриваться и как коэффициент готовности, и как коэффициент оперативной готовности [11], и как другие показатели, характеризующие работоспособность элементов телекоммуникационной сети [12]. Предполагая, что L – случайное множество ребер, существующее после реализации отдельного опыта, случайный граф G приобретает форму обобщенной модели Эрдеша – Реньи, то есть $G = (V, L)$. Заметим, что вероятность p_G существования конкретного графа G в результате опыта есть:

$$p_G = \prod_{(i,j) \in L} p_{i,j} \prod_{(i,j) \notin L} (1 - p_{i,j}).$$

В подобной постановке вероятность $p(S)$ обладания случайным графом G некоторым свойством трактуется как сумма вероятностей p_G существования графов, принадлежащих множеству S , для которого этого свойства выполняется, то есть:

$$p(S) = \sum_{G \in S} p_G.$$

В контексте устойчивости телекоммуникационной сети под рассматриваемым свойством обычно понимают связность сети в той или иной форме, а следовательно, вероятность $p(S)$ связности сети оценивается как сумма вероятностей p_G всех связанных графов (на заданном множестве вершин), формирующих множество S . Отметим также, что вероятность связности сети может трактоваться в рамках терминов коэффициента готовности, коэффициента оперативной готовности [11] и других показателей, характеризующих работоспособность телекоммуникационной сети [12].

3. Меры связности. Основываясь на представлении понятия стохастической связности сети как соответствия некоторого случайного графа свойства связности заданному набору вершин, традиционно выделяют три меры связности: двухполюсная, многополюсная и всеполюсная [9, 13]. Отметим, что в зависимости от трактовки английских понятий иногда используют термины двухтерминальная, k -терминальная, и всетерминальная связности.

Двухполюсная связность – это свойство сети, характеризующее наличием хотя бы одного пути между заданной парой вершин.

Всегополюсная связность – это свойство сети, характеризующее наличием хотя бы одного пути между любой парой вершин. Следует отметить, что данное понятие эквивалентно классическому определению термина *связность* в теории графов [14]. Также необходимо упомянуть, что иногда используют синонимы глобальная и g -связность.

Всегополюсная связность соответствует состоянию случайного графа, когда все его вершины имеют связь друг с другом. В результате событие связности можно трактовать как существование хотя бы одного остового дерева, включающего все вершины случайного графа, а также в котором присутствует только один путь между любой парой вершин, то есть отсутствуют циклы.

Многополюсная связность – это свойство сети, характеризующее наличием хотя бы одного пути между заданными вершинами. Предполагается, что существует связь между любой парой вершин из заданного множества. В результате событие связности можно трактовать как существование хотя бы одного дерева (неостового), включающего все заданные вершины случайного графа.

Естественно, что понятие многополюсной связности является обобщением и двухполюсной, и всеполюсной связности. При этом целесообразно рассматривать событие связности как существование некоторого подграфа S_i в конкретной реализации графа G , содержащего заданные вершины исходного случайного графа G и однозначно

являющегося деревом, листьями которого могут быть только эти заданные вершины. Для общего случая многополюсной сети этими подграфами S_i оказываются деревья, имеющие заданный набор вершин, для двухполюсной сети – деревья, содержащие заданную пару вершин, которые, по сути, являются путями; для всеполюсной сети – остовые деревья, включающие все вершины случайного графа G .

Таким образом, множество S графов, для которого выполняется свойство связности, имеет следующий вид:

$$S = \{G : \exists S_i \subseteq G\}, \quad (1)$$

то есть множество S состоит из всех графов G , для которых существует хотя бы один подграф S_i , содержащийся в графе G .

Таким образом, для расчета устойчивости телекоммуникационных сетей на основе данного подхода предварительно необходимо сформировать множество состояний графа (модели сети), представляющие собой некоторые подграфы S_i которые обладают свойством связности между заданными вершинами. В современных исследованиях методы формирования (перечисления) множеств состояний графа (наборов путей) традиционно разделяют на две категории [9, 15, 16]. Первая из них предполагает использование информации, содержащейся только в матричном представлении графа [17-19], а вторая – в самой его структуре путем обхода графа в глубину или ширину, либо полный перебор [20, 21].

Попытаемся представить методы формирования множеств состояний графа на основе обоих подходов, когда в качестве исходных данных используется матричное представление, но в процессе поиска решения применяется также и способы обхода графа.

4. Метод формирования множества путей. Формально процедуру формирования множества путей целесообразно разбить на три этапа. На первом этапе синтезируют множество путей от источника ко всем возможным вершинам, на втором – отбирают из полученного множество только те пути, стоки которых совпадают с заданным стоком, а на третьем – преобразуют пути в векторную форму.

Исходными данными будет матрица смежностей $\mathbf{A} = \{a_{i,j}\}_{i,j=1,v}$ случайного графа G [22-24], элементами $a_{i,j}$ которой являются:

$$a_{i,j} = \begin{cases} 1, & p_{i,j} \neq 0, \\ 0, & p_{i,j} = 0. \end{cases}$$

Отметим, что естественным упрощением для детерминированного графа, вероятности связности ребер которого могут принимать значения только ноль или единица, то есть $p_{i,j} = \{0;1\}$, является представление, основанное на множестве L' относительно надежных ребер:

$$L' = \{(i, j) : p_{i,j} \neq 0\}.$$

Тогда элементы матрицы связностей принимают классический вид [14]:

$$a_{i,j} = \begin{cases} 1, & (i, j) \in L', \\ 0, & (i, j) \notin L'. \end{cases}$$

Заданное направление связи образуется двумя вершинами: истоком с номером v_s и стоком с номером v_t .

Изначально образуют матрицу путей единичной длины \mathbf{W}_0 по следующему правилу. Каждый новый k -й столбец этой матрицы добавляется к уже существующим в случае наличия соответствующего ребра (v_s, i) у графа G , а последовательный перебор производится по всем v вершинам графа:

$$\mathbf{W}_1^1 = \left\{ \left[\begin{array}{c} v_s \\ \min(i) \end{array} \right] : a_{v_s, i} = 1, i = 1, \dots, v \right\}; \quad (2)$$

$$\mathbf{W}_1^k = \left\{ \left[\begin{array}{c} v_s \\ \min(i) \end{array} \right] : a_{v_s, i} = 1, [\mathbf{W}_1]_{2, k-1} < v, i = [\mathbf{W}_1]_{2, k-1} + 1, \dots, v \right\}, k \leq v, \quad (3)$$

где \mathbf{X}^k – k -й столбец матрицы \mathbf{X} ; $\begin{bmatrix} i \\ j \end{bmatrix}$ – вектор, построчно составленный из элементов i и j ; $[\mathbf{X}]_{i,j}$ – элемент матрицы \mathbf{X} , стоящий на i -й строке в j -м столбце.

Отметим, что в случае отсутствия путей единичной длины матрица \mathbf{W}_1 формироваться не будет (случай графа без ребер), а при их наличии – она будет состоять из двух строк, где первая строка заполняется элементами, значения которых будут равны номеру истока v_s , а во второй строке будут последовательно от меньшего к большему перечислены все смежные истоку вершины.

Последующие матрицы \mathbf{W}_m , $m = 2, \dots, v-1$ являются матрицами путей длины m и формируются так, чтобы не образовывались петли. Каждый новый k -й столбец таких матриц добавляется к уже существующим в случае наличия соответствующего ребра (s', j) , $j = 1, \dots, v$, у графа G , а последовательный перебор производится сначала по всем строкам матрицы \mathbf{W}_{m-1} , а затем по всем v вершинам графа:

$$\mathbf{W}_m^1 = \left\{ \left[\begin{array}{c} \mathbf{W}_{m-1}^i \\ \min(j) \end{array} \right] : a_{s',j} = 1, j \notin \mathbf{W}_{m-1}^i, j = 1, \dots, v \right\}, \quad (4)$$

$$\mathbf{W}_m^k = \left\{ \left[\begin{array}{c} \mathbf{W}_{m-1}^i \\ \min(j) \end{array} \right] : a_{s',j} = 1, j \notin \mathbf{W}_{m-1}^i, [\mathbf{W}_m]_{m+1,k-1} < v; \right. \\ \left. j = [\mathbf{W}_m]_{m+1,k-1} + 1, \dots, v, s' = [\mathbf{W}_{m-1}]_{m,i}, i = 1, \dots, \text{cols}(\mathbf{W}_{m-1}) \right\}, \quad (5)$$

где $\left[\begin{array}{c} \mathbf{X} \\ j \end{array} \right]$ – блочный вектор, составленный из вектора \mathbf{X} и скаляра j ; $\text{cols}(\mathbf{X})$ – число столбцов матрицы \mathbf{X} .

Отметим, что путь максимальной длины в любом графе на единицы меньше числа вершин графа, что соответствует матрице \mathbf{W}_{v-1} .

В результате данной процедуры образуется $v-1$ матриц путей, строками в которой являются произвольные вершины. Для отбора путей, полезных для анализа, необходимо отбросить все пути, за исключением тех, строки которых соответствуют вершине v_t . Тогда формируются матрицы путей \mathbf{W}'_m , $m = 1, \dots, v-1$ от истока v_s к стоку v_t путем последовательного перебора по всем матрицам путей \mathbf{W}_m и столбцам каждой из этих матриц:

$$\mathbf{W}_m^1 = \left\{ \mathbf{W}_m^{\min(i)} : v_t = [\mathbf{W}_m]_{m+1,i}, i = 1, \dots, \text{cols}(\mathbf{W}_m) \right\}, \quad (6)$$

$$\mathbf{W}_m^{k'} = \left\{ \mathbf{W}_m^{\min(i)} : v_t = [\mathbf{W}_m]_{m+1,i}, [\mathbf{W}'_m]_{m+1,k-1} < \text{cols}(\mathbf{W}_m); \right. \\ \left. i = [\mathbf{W}'_m]_{m+1,k-1} + 1, \dots, \text{cols}(\mathbf{W}_m) \right\}. \quad (7)$$

Отметим, что в результате некоторые матрицы \mathbf{W}'_m могут просто не существовать из-за отсутствия пути длины m между истоком v_s и стоком v_t .

Конечным этапом формирования множества путей является преобразование полученных матриц путей \mathbf{W}'_m к общему виду единой матрицы путей $\mathbf{W} = \{w_{i,j}\}_{i=1,\dots,l, j=1,\dots,s}$, где s – общее число путей, каждый

столбец которой соответствует определенному пути, а строка – ребру. Для этого все ребра $(i,j) \in L'$, $i, j = 1, \dots, v$ упорядочиваются и им присваиваются обычно в порядке возрастания номеров инцидентных вершин определенные номера $k = 1, \dots, l$, то есть пары вершин сопоставляются с ребрами: $(i,j) \rightarrow k$. Таким образом, элементы матрицы путей \mathbf{W} образуются по следующему правилу:

$$w_{i,j} = \begin{cases} 1, i \in L'_j, \\ 0, i \notin L'_j, \end{cases} \quad (8)$$

где $L'_j \in L'$ – j -й путь.

Процедура формирования матрицы путей \mathbf{W} на основе матриц \mathbf{W}'_m довольно проста и использует последовательный перебор всех столбцов ранее определенных матриц \mathbf{W}'_m , $m = 1, \dots, v-1$:

$$w_{i,j} = \begin{cases} 1, i \in \mathbf{W}'_m{}^{tk}, \\ 0, i \notin \mathbf{W}'_m{}^{tk}, \end{cases} \quad (9)$$

$$j = \begin{cases} k, j \leq \text{cols}(\mathbf{W}'_1), \\ k + \sum_{r=1}^{m-1} \text{cols}(\mathbf{W}'_r), j > \text{cols}(\mathbf{W}'_1). \end{cases}$$

Отметим также, что общее количество путей s между заданной парой вершин, основываясь только на виде матрицы смежностей \mathbf{A} , заранее определить практически невозможно [25], однако матрицы путей позволяют это сделать:

$$s = \text{cols}(\mathbf{W}) = \sum_{r=1}^m \text{cols}(\mathbf{W}'_r).$$

Рассмотрим реализацию предложенных процедур на примере.

Структура сети представлена на рисунке 1. Источник и сток (полюса) на рисунке отмечены квадратами: $v_s = 1$, $v_t = 5$. Матрица смежностей графа данной сети имеет вид:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}. \quad (10)$$

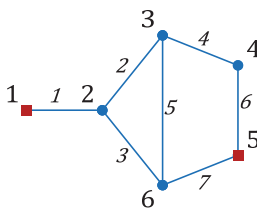


Рис.1. Двухполюсный граф исследуемой сети связи

Поскольку вершин в графе всего шесть ($v = 6$), то матриц путей \mathbf{W}_i – пять ($m = 5$), а согласно формулам (2)–(5)

$$\mathbf{W}_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad \mathbf{W}_2 = \begin{bmatrix} 1 & 1 \\ 2 & 2 \\ 3 & 6 \end{bmatrix}, \quad \mathbf{W}_3 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 6 & 6 \\ 4 & 6 & 3 & 5 \end{bmatrix},$$

$$\mathbf{W}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 6 & 6 \\ 4 & 6 & 3 & 5 \\ 5 & 5 & 4 & 4 \end{bmatrix}, \quad \mathbf{W}_5 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 6 & 6 \\ 4 & 6 & 3 & 5 \\ 5 & 5 & 4 & 4 \\ 6 & 4 & 5 & 3 \end{bmatrix}.$$

В соответствии с выражениями (6)–(7) непустых матриц путей от первой к шестой вершине оказывается всего три:

$$\mathbf{W}'_1 = \mathbf{W}'_2 = \emptyset, \mathbf{W}'_3 = \begin{bmatrix} 1 \\ 2 \\ 6 \\ 5 \end{bmatrix}, \mathbf{W}'_4 = \begin{bmatrix} 1 & 1 \\ 2 & 2 \\ 3 & 3 \\ 4 & 6 \\ 5 & 5 \end{bmatrix}, \mathbf{W}'_5 = \begin{bmatrix} 1 \\ 2 \\ 6 \\ 3 \\ 4 \\ 5 \end{bmatrix}.$$

Для построения обобщенной матрицы путей \mathbf{W} необходимо пронумеровать все ребра графа (на рисунке 1 нумерация ребер приведена курсивным шрифтом), что проще всего сделать, используя матрицу инциденций [26]:

$$\mathbf{B} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Таким образом, согласно (9) обобщенная матрица путей \mathbf{W} имеет размер 7×4 ($s = 4$).

$$\mathbf{W} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}, \quad (11)$$

а множество путей изображено на рисунке 2.

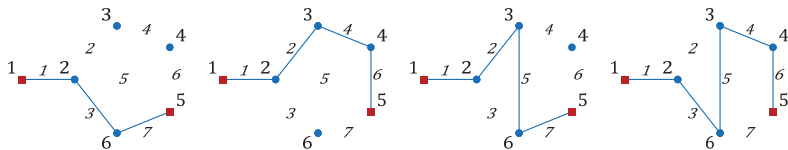


Рис. 2. Пути (простые цепи) в направлении 1–5 графа, представленного на рисунке 1

5. Метод формирования множества деревьев. Предлагаемый подход основывается на методе формирования множества деревьев в форме декартового произведения $(v-1)$ вершинных сечений [9, 27, 28]. При этом принципиальным отличием выступает процедура проверки связности получаемого подграфа на последнем этапе, которая, по сути, аналогична ранее используемой проверке на наличие циклов.

Формально процедуру формирования множества деревьев целесообразно разбить на четыре этапа. На первом этапе синтезируют $(v-1)$ множество вершинных сечений, на втором – находят декартово произведение вплоть до $(v-2)$ полученного множества сечений с учетом неповторяемости ребер, на третьем – отдельно вычисляют последнее произведение на $(v-1)$ -е множество сечений, одновременно контролируя не только уникальность ребер, но и связность полученного подграфа, а на четвертом – преобразуют деревья в векторную форму.

Исходными данными здесь является только матрица смежностей A случайного графа G . Для дальнейших выкладок ее необходимо преобразовать в модифицированную матрицу смежности $A' = \{a'_{i,j}\}_{i,j=1,\dots,v}$, учитывающую нумерацию ребер. Каждый ненулевой элемент этой матрицы соответствует номеру ребра, упорядоченного в порядке возрастания номеров инцидентных им вершин. А поскольку матрица неориентированного графа симметрическая, то целесообразно формировать сразу два элемента модифицированной матрицы:

$$a'_{i,j} = a'_{j,i} = \begin{cases} \sum_{r=1}^j a_{1,r}, & a_{i,j} = 1, \\ 0, & a_{i,j} = 0, \end{cases} \quad (12)$$

$$j = 1, \dots, v,$$

$$a'_{i,j} = a'_{j,i} = \begin{cases} \sum_{k=1}^{i-1} \sum_{r=k}^v a_{k,r} + \sum_{r=k}^j a_{i,r}, & a_{i,j} = 1, \\ 0, & a_{i,j} = 0, \end{cases} \quad (13)$$

$$i = 2, \dots, v, j = 1, \dots, v.$$

Множество вершинных сечений сформировать довольно просто. Достаточно в матрице смежностей рассмотреть все единичные элементы, принадлежащие соответствующему столбцу или строке. Для упрощения дальнейших вычислений целесообразно исключить вершину, валентность которой имеет максимальную величину, так как она породит наибольшее число комбинаций в декартовом произведении. Номер пропускаемой в ходе анализа вершины:

$$v_e = \min \left[\arg \max (\mathbf{A} \mathbf{1}_v) \right], \quad (14)$$

где $\mathbf{1}_v = \{u_i\}_{i=1, \dots, v}$ – единичный вектор длины v , $u_i = 1$.

Здесь используется внешний оператор минимизации для отбора только одной вершины в случае, если вершин с максимальной валентностью сразу несколько.

Для удобства программной реализации множество сечений целесообразно синтезировать в форме векторов сечений \mathbf{c}_r , $r = 1, \dots, v-1$. Для их определения производится последовательная процедура выделения по строкам ненулевых элементов матрицы смежности:

$$[\mathbf{c}_r]_1 = \left\{ a'_{i, \min(j)} : a_{i,j} = 1, j = 1, \dots, v \right\}, \quad (15)$$

$$[\mathbf{c}_r]_k = \left\{ a'_{i, \min(j)} : a_{i,j} = 1, [\mathbf{c}_i]_{k-1} < v, j = [\mathbf{c}_i]_{k-1} + 1, \dots, v \right\}; \quad (16)$$

$$i \neq v_e, i = 1, \dots, v, r = \begin{cases} i, & i < v_e, \\ i-1, & i > v_e. \end{cases}$$

В результате формируется $(v-1)$ векторов сечений, над которыми осуществляется последовательная процедура декартового умножения с контролем уникальности, входящих в каждое произведение ребер. Исходной матрицей декартовых произведений \mathbf{W}_1 яв-

ляется транспонированный вектор сечений первой или второй ($v_e = 1$) вершин:

$$\mathbf{W}_1 = \mathbf{c}_1^T, \quad (17)$$

где T – оператор транспонирования.

Далее результаты декартовых произведений формируются в виде матриц \mathbf{W}_r , k -й столбец каждой из которых образуется только в случае отсутствия повторяющихся ребер, а последовательный перебор выполняется сначала по столбцам матрицы декартовых произведений \mathbf{W}_{r-1} , а затем по элементам векторов сечений \mathbf{c}_r :

$$\mathbf{W}_r^k = \left\{ \left[\begin{array}{c} \mathbf{W}_{r-1}^i \\ [\mathbf{c}_r]_j \end{array} \right] : [\mathbf{c}_r]_j \notin \mathbf{W}_{r-1}^i, i = 1, \dots, \text{cols}(\mathbf{W}_{r-1}), j = 1, \dots, \text{rows}(\mathbf{c}_r) \right\}, \quad (18)$$

$$r = 2, \dots, v-2.$$

Последнее ($v-2$) декартово произведение дополнительно проверяется на связность с целью выявить подграфы, содержащие петли:

$$\mathbf{W}_{v-1}^k = \left\{ \left[\begin{array}{c} \mathbf{W}_{v-2}^i \\ [\mathbf{c}_{v-1}]_j \end{array} \right] : [\mathbf{c}_{v-1}]_j \notin \mathbf{W}_{v-2}^i, S \left(\left[\begin{array}{c} \mathbf{W}_{v-2}^i \\ [\mathbf{c}_{v-1}]_j \end{array} \right] \right) = 1; \right. \\ \left. i = 1, \dots, \text{cols}(\mathbf{W}_{v-2}), j = 1, \dots, \text{rows}(\mathbf{c}_{v-1}) \right\}, \quad (19)$$

где $S(\mathbf{W})$ – оператор проверки связности, результат выполнения которого равен единице, если граф, состоящий из ребер \mathbf{W} , связный, и нулю, если несвязный.

Стандартным подходом для реализации этого оператора служит один из двух поисков в ширину или глубину [28, 29]. Для его выполнения первоначально выбирается произвольная вершина v_1 , из которой начинается поиск. В ходе поиска на каждом шаге используется вспомогательный вектор $\mathbf{w}_k = \{w_i\}_{i=1, \dots}$, размер которого изменяется в ходе итерационного процесса, а каждый шаг эквивалентен проверке

отсутствия неучтенных в текущем векторе \mathbf{w}_k вершин. Первоначально этот вектор состоит только из одного элемента, значение которого соответствует номеру начальной вершины поиска, то есть $\mathbf{w}_1 = [v_1]$. Далее на каждом k -м шаге проверяются инцидентные данной вершине вершины на присутствие в векторе \mathbf{w}_{k-1} :

$$\mathbf{w}_k = \left\{ \left[\begin{array}{c} \mathbf{w}_{k-1} \\ \mathbf{i} \end{array} \right] : \mathbf{i} = \{i\}_{i=1, \dots, v}, a_{w_{k-1}, i} = 1, i \notin \mathbf{w}_{k-1}, i = 1, \dots, v \right\},$$

$$k = 1, \dots, \text{rows}(\mathbf{w}_k).$$

Итерационный процесс завершается после того, как номер шага k превысит количество элементов, то есть при $k > \text{rows}(\mathbf{w}_k)$. После этого проверяется, все ли вершины присутствуют в конечном векторе \mathbf{w}_k :

$$S(\mathbf{A}) = \begin{cases} 1, & \text{rows}(\mathbf{w}_k) = v, \\ 0, & \text{rows}(\mathbf{w}_k) > v. \end{cases}$$

Сформированная матрица \mathbf{W}_{v-1} содержит описание всех деревьев исследуемого графа, однако она имеет структуру, где каждое дерево формализовано как перечисление номеров входящих в него ребер, что несколько неудобно для последующих выкладок. В связи с этим на последнем этапе предполагается преобразование данной матрицы \mathbf{W}_{v-1} в матрицу деревьев $\mathbf{W} = \left\{ w_{i,j} \right\}_{\substack{i=1, \dots, l \\ j=1, \dots, s}}$, аналогичной по структуре матрице (8), каждый элемент которой определяется выражением:

$$w_{i,j} = \begin{cases} 1, & i \in L'_j, \\ 0, & i \notin L'_j, \end{cases}$$

где $L'_j \in L'$ – j -е дерево; s – общее число деревьев.

Отметим, что здесь и далее все описания связанных и несвязных подграфов будут иметь общие обозначения, если иное не оговорено дополнительно.

Таким образом, процедура формирования матрицы деревьев \mathbf{W} на основе матрицы \mathbf{W}_{v-1} оказывается еще более простой, поскольку все

деревья содержат одинаковое количество ребер, и использует последовательный перебор всех столбцов ранее определенной матрицы \mathbf{W}_{v-1} :

$$w_{i,j} = \begin{cases} 1, & i \in \mathbf{W}_{v-1}^j, \\ 0, & i \notin \mathbf{W}_{v-1}^j, \end{cases} \quad (20)$$

$$i = 1, \dots, l, j = 1, \dots, s.$$

Отметим также, что общее количество деревьев s , основываясь только на виде матрицы смежностей \mathbf{A} , заранее определить возможно на основе теоремы Кирхгофа [14], что позволяет дополнительно проверять правильность реализации алгоритмов, так как здесь также:

$$s = \text{cols}(\mathbf{W}),$$

где $\text{cols}(\mathbf{X})$ – число столбцов матрицы \mathbf{X} .

Рассмотрим реализацию предложенных процедур на примере.

Структура сети представлена на рисунке 3. Матрица смежностей графа задается формулой (10), а модифицированная матрица в соответствии с уравнениями (12)–(13):

$$\mathbf{A}' = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 2 & 0 & 0 & 3 \\ 0 & 2 & 0 & 4 & 0 & 5 \\ 0 & 0 & 4 & 0 & 6 & 0 \\ 0 & 0 & 0 & 6 & 0 & 7 \\ 0 & 3 & 5 & 0 & 7 & 0 \end{bmatrix}. \quad (21)$$

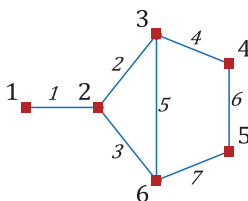


Рис. 3. Всеполюсный граф исследуемой сети связи

Согласно выражению (14) пропускаемая в ходе анализа вершина имеет значение $v_e = 2$, так как ее степень $\text{deg } v_2 = 3$. Тогда в

соответствии с процедурами (15)–(16) рассматривается всего пять векторов сечений:

$$\mathbf{c}_1 = [1], \mathbf{c}_2 = \begin{bmatrix} 2 \\ 4 \\ 5 \end{bmatrix}, \mathbf{c}_3 = \begin{bmatrix} 4 \\ 6 \end{bmatrix}, \mathbf{c}_4 = \begin{bmatrix} 6 \\ 7 \end{bmatrix}, \mathbf{c}_5 = \begin{bmatrix} 3 \\ 5 \\ 7 \end{bmatrix}.$$

Согласно выражениям (17)–(19) матрицы декартовых произведений:

$$\mathbf{W}_1 = [1], \mathbf{W}_2 = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 4 & 5 \end{bmatrix}, \mathbf{W}_3 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 4 & 5 & 5 \\ 4 & 6 & 6 & 4 & 6 \end{bmatrix};$$

$$\mathbf{W}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 4 & 5 & 5 & 5 \\ 4 & 4 & 6 & 6 & 4 & 4 & 6 \\ 6 & 7 & 7 & 7 & 6 & 7 & 7 \end{bmatrix},$$

$$\mathbf{W}_5 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 4 & 5 & 5 & 5 \\ 4 & 4 & 4 & 4 & 4 & 6 & 6 & 6 & 4 & 4 & 6 \\ 6 & 6 & 6 & 7 & 7 & 7 & 7 & 7 & 6 & 7 & 7 \\ 3 & 5 & 7 & 3 & 5 & 3 & 5 & 3 & 3 & 3 & 3 \end{bmatrix}.$$

В результате согласно формуле (20) матрица деревьев имеет размер 7×11 :

$$\mathbf{W} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \end{bmatrix},$$

а множество остовых деревьев изображено на рисунке 4.

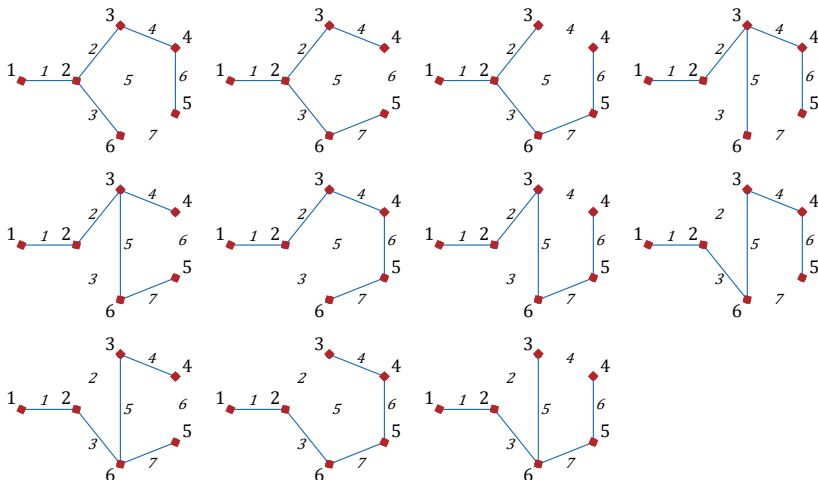


Рис. 4. Остовы (остовые деревья) графа, представленного на рисунке 3

6. Метод формирования множества многополюсных деревьев.

Как упоминалось ранее, многополюсное дерево – это такой подграф исходного графа, у которого сохранены все заданные вершины, и между любой парой из них существует один и только один путь. Отметим, что иногда многополюсное дерево называют также k -полусным, что несколько неудобно, так как под k уместно понимать некоторый счетчик.

Естественно, что подобный подграф является деревом, но не обязательно остовым, а кроме того, его листьями могут быть только заданные k вершин (полюса). Предлагаемый здесь подход к формированию множества многополюсных деревьев основывается на методе, описанном в [9, 30], предполагающим последовательное преобразование всего множества остовых деревьев в многополюсные. В отличие от способа [9, 30], последовательно сравнивающего остовые деревья с сечениями вершин, которые не являются полюсами, и устранившего только те ребра, которые повторяются один раз, целесообразно воспользоваться свойством многополюсного дерева включать в качестве листьев только полюса. В результате оказывается возможным последовательно удалять из остового дерева висячие вершины и таким образом добиваться формирования требуемого подграфа. Кроме того, на каждом шаге целесообразно проверять, является ли сформированное многополюсное дерево подграфом оставшихся непроанализированных остовых деревьев, и если да, то данные деревья из дальнейшего рассмотрения убирать [31]. Такая процедура существенно упрощает про-

цесс формирования деревьев, так как исключает из анализа дубликаты многополюсных деревьев.

Формально процедуру формирования множества многополюсных деревьев целесообразно разбить на три этапа. На первом этапе синтезируют множество остовых деревьев в форме матрицы деревьев, на втором – определяют множество неполюсных вершинных сечений, а на третьем – непосредственно формируют множество многополюсных деревьев путем последовательно удаления висячих неполюсных вершин и удаления избыточных остовых деревьев в матричном виде.

Процедура синтеза множества остовых деревьев в форме матрицы деревьев описана ранее в виде соотношений (12)–(20). В результате ее выполнения образуется матрица \mathbf{W} остовых деревьев.

Процесс формирования множества неполюсных вершинных сечений основан на выражениях (15)–(16) и, по сути, отличается лишь тем, что из рассмотрения необходимо удалить не вершину с максимальной валентностью, а неполюсные вершины и образовывать не вектора, а матрицы неполюсных вершинных сечений \mathbf{C} . Таким образом, целесообразно первоначально образовать вектор, содержащий номера вершин, не являющихся полюсами. Так, если номера вершин-полюсов v_i образуют вектор $\mathbf{k} = \{k_i\}_{i=1, \dots, v'}$, то вектор

$$\mathbf{k}' = \{k'_i\}_{i=1, \dots, v-v'} :$$

$$\mathbf{k}' = \{\{i\} : i \notin \mathbf{k}, i = 1, \dots, v\}$$

включает номера только неполюсных вершин.

Следовательно, матрица неполюсных вершинных сечений $\mathbf{C} = \{c_{i,j}\}_{\substack{i=1, \dots, l \\ j=1, \dots, v-v'}}$ аналогична по структуре матрице (8), каждый элемент которой определяется выражением:

$$c_{i,j} = \begin{cases} 1, i \in L'_j, \\ 0, i \notin L'_j, \end{cases} \quad (22)$$

где $L'_j \in L'$ – j -е вершинное сечение.

Данная матрица формируется путем последовательной процедуры сравнения элементов модифицированной матрицы \mathbf{A}' смежностей с индексами образуемой матрицы \mathbf{C} сечений:

$$c_{i,j} = \{a_{k,j} : a'_{k,j} = i, k = 1, \dots, v\}, i = 1, \dots, l, j = 1, \dots, v - v'. \quad (23)$$

Синтез матрицы многополюсных деревьев разбивается на ряд этапов, на каждом из которых рассматривается одно из остовых деревьев на предмет исключения листьев, не являющихся полюсами, а затем в рамках того же этапа удаляются те остовые деревья, которые включают в себя полученное многополюсное дерево.

Удаление из остовых деревьев листьев представляется в виде последовательной процедуры сравнения текущего вектора остового дерева \mathbf{W}^k со всеми столбцами матрицы непольных сечений \mathbf{C} . Причем подобное сравнение каждого с каждым необходимо проводить до тех пор, пока не наступит ситуация отсутствия непольных листьев, поскольку удаление листа может привести к возникновению новой непольной висячей вершины. Таким образом, образуется вектор \mathbf{w}_r , содержащий номера ребер подграфа по следующему закону:

$$\mathbf{w}_1 = \left\{ \mathbf{W}^k - \mathbf{q} : \mathbf{q} = \frac{\mathbf{W}^k + \mathbf{C}^i}{2}, \mathbf{q}^T \mathbf{1}_l = 1, i = 1, \dots, v - v' \right\}; \quad (24)$$

$$\mathbf{w}_2 = \left\{ \mathbf{w}_1 - \mathbf{q} : \mathbf{w}_1 \neq \mathbf{W}^k, \mathbf{q} = \frac{\mathbf{w}_1 + \mathbf{C}^i}{2}, \mathbf{q}^T \mathbf{1}_l = 1, i = 1, \dots, v - v' \right\}; \quad (25)$$

$$\mathbf{w}_r = \left\{ \mathbf{w}_{r-1} - \mathbf{q} : \mathbf{w}_{r-1} \neq \mathbf{w}_{r-2}, \mathbf{q} = \frac{\mathbf{w}_{r-1} + \mathbf{C}^i}{2}, \right. \\ \left. \mathbf{q}^T \mathbf{1}_l = 1, i = 1, \dots, v - v' \right\}, r = 1, \dots \quad (26)$$

В конечном счете, когда вектор подграфа \mathbf{w}_r не изменяется ($\mathbf{w}_{r-1} \neq \mathbf{w}_{r-2}$), то он превращается в многополюсное дерево, формализуемое как столбец матрицы $\mathbf{W}' = \left\{ w'_{i,j} \right\}_{\substack{i=1,\dots,l \\ j=1,\dots,s}}$ многополюсных деревьев, где s – общее число многополюсных деревьев, аналогичных по структуре матрице (8), каждый элемент которой определяется выражением:

$$w'_{i,j} = \begin{cases} 1, i \in L'_j, \\ 0, i \notin L'_j, \end{cases}$$

где $L'_j \in L'$ – j -е многополюсное дерево.

Следовательно:

$$\mathbf{W}^k = \mathbf{w}_r. \quad (27)$$

Удаление избыточных остовых деревьев также производится путем сравнения, но уже текущего многополюсного дерева \mathbf{w}_r и всех нерассмотренных остовых деревьев \mathbf{W}^k :

$$\mathbf{W}^r = \{ \mathbf{W}^j : \mathbf{w}_r \subset \mathbf{W}^j, j = k+1, \dots, \text{cols}(\mathbf{W}) \}.$$

Таким образом, после просмотра всех остовых деревьев формируется матрица \mathbf{W}' многополюсных деревьев. Отметим, что вследствие усечения матрицы \mathbf{W} количество этапов процедуры может быть существенно меньше общего числа остовых деревьев.

Рассмотрим реализацию предложенных процедур на примере.

Структура сети представлена на рисунке 5. Полюсами данного графа являются первая, четвертая и пятая вершины (отмечены квадратами). Таким образом, вектор полюсов равен:

$$\mathbf{k} = \begin{bmatrix} 1 \\ 4 \\ 5 \end{bmatrix}.$$

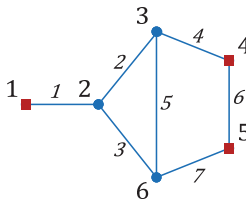


Рис. 5. Многополюсный граф (полюсы – первая, четвертая и пятая вершины) исследуемой сети связи

Матрица смежностей графа задается формулой (10), а модифицированная матрица – выражением (21).

В соответствии с процедурой (23) рассматривается матрица неполюсных вершинных сечений размером 7×3 :

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

В результате согласно формулам (24)–(27) матрица многополюсных деревьев имеет размер 7×7 :

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 \end{bmatrix},$$

а множество многополюсных деревьев изображено на рисунке 6.

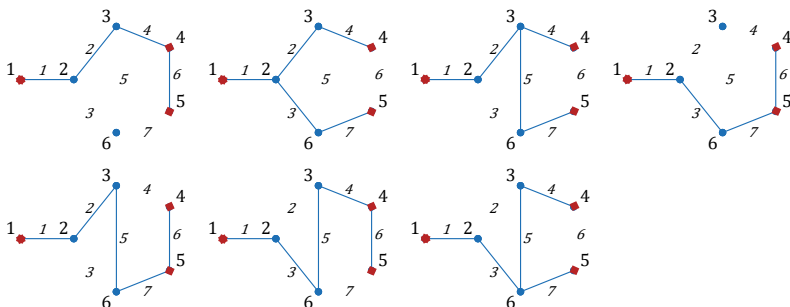


Рис. 6. Многополюсные деревья (полюсы – первая, четвертая и пятая вершины) графа, представленного на рисунке 5

7. Формирование множеств состояний на примере магистральной сети Ростелеком «Европа – Азия». В качестве тестовой сети связи используется магистральная сеть Ростелеком, развернутая с целью формирования потоков в направлении «Европа – Азия», помеченный граф которой представлен на рисунке 7. Исходная топологическая структура данной сети приведена в [31].

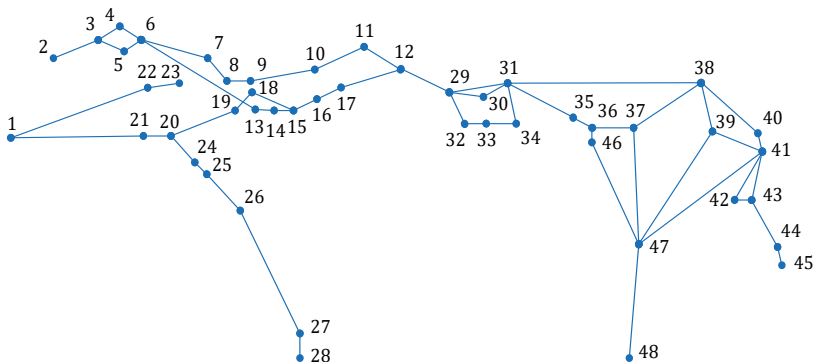


Рис. 7. Граф – модель магистральной сети Ростелеком [31]

В таблице 1 приведены сводные данные по числу состояний графа и временных затрат на их формирование для различных мер связности. Отметим, что реализация вышеописанных процедур анализа выполнена в программной среде MathCad 15, а вычисления проводились на компьютере с процессором AMD Ryzen 5 3500U, оперативной памятью 16 Гб и 64-разрядной операционной системой Windows 10.

При анализе двухполюсной связности исследовался весь граф сети Ростелеком (рис. 7). Выигрыш по производительности по сравнению с методом поиска на основе матрицы связностей (*traversing through connection matrix*), изложенным в [9], составил величину до 25 % и наиболее ярко выражен для случаев малого количества состояний (табл. 1). Данное обстоятельство вызвано, прежде всего, использованием в предложенном методе на первоначальном этапе процедуры синтеза множества путей от источника ко всем возможным вершинам, что позволяет для незначительно разветвленных графов образовывать подобные множества со сравнительно малыми мощностями. Однако с ростом сложности сети эти множества становятся все более мощными, что и нивелирует положительный эффект предложенного метода формирования.

Таблица 1. Данные результатов формирования состояний магистральной сети Ростелеком [31] (рис. 7)

Мера связности	Номера вершин–полусов	Число состояний (подграфов)	Время формирования, с	
			известные методы	предложенные методы
ДВУХПОЛУСНАЯ	1, 28	1	0,070	0,056
	1, 45	216	0,097	0,092
	1, 48	72	0,075	0,063
	2, 23	4	0,217	0,188
	2, 28	4	0,217	0,189
	2, 45	432	0,265	0,266
	2, 48	144	0,224	0,201
	23, 28	1	0,071	0,057
	23, 45	216	0,100	0,095
	23, 48	72	0,077	0,067
	28, 45	216	0,102	0,099
28, 48	72	0,079	0,068	
45, 48	22	0,481	0,423	
ВСЕПОЛУСНАЯ	1–28	48	0,153	0,170
	1–34	672	11,192	3,180
	1–38	3360	16,471	9,492
	1–41	13440	–	71,181
	29–48	22848	–	47,931
МНОГОПОЛУСНАЯ	1, 2, 28	4	0,024	0,011
	1, 2, 23	1	0,027	0,007
	1, 23, 28	1	0,031	0,004
	2, 23, 28	4	0,028	0,005
	1, 2, 23, 28	4	0,022	0,005
	29, 45,48	216	5,952	2,763
	37, 45,48	64	6,431	0,704
	29, 37, 45,48	414	6,363	4,721
	29, 37, 41, 45,48	414	5,831	4,822
29, 33, 37, 41, 45,48	690	6,241	6,400	

Анализ всеполюсной связности сети Ростелеком проводился на усеченных графах, поскольку на исходном графе количество деревьев составило величину более миллиона единиц [31], что не позволило на имеющихся вычислительных ресурсах проводить расчеты ни по одному из рассматриваемых методов. В соответствии с таблицей 1 для всеполюсной меры связности исследуемые подграфы состояли только из тех вершин, номера которых указаны и являются полюсами. По сравнению с известным методом, использующим декартово произведение вершинных сечений (using the cartesian product of the node cut sets [27]), выигрыш в вычислительной слож-

ности оказался достаточно весомым (более, чем в три раза) для разветвленных сетей. Это обстоятельство продиктовано внесением в предложенный метод дополнительного усложнения, связанного с проверкой формируемых подграфов на наличие циклов, то есть, по сути, на свойство древовидности. В известном методе данная проверка не выполняется, что приводит, во-первых, к появлению подграфов, которые не являются деревьями как таковыми, а, во-вторых, к чрезмерной загрузке результирующей матрицы деревьев. На слаборазветвленных сетях данной проблемы практически не наблюдается, что влечет за собой сопоставимость получаемых результатов. Отметим, что стоящие в соответствующих графах проверки указывают на отсутствие выходных данных алгоритмов, вызванных переполнением памяти.

Временные параметры для многополюсной связности (табл. 1) учитывают только процедуру непосредственного образования матрицы многополюсных деревьев из остовых. Таким образом, для реальной оценки временных затрат необходимо еще учесть время, требуемое для формирования последних. Сравнительный анализ предложенного метода с известным подходом [30] показывает преимущество первого (в несколько раз) при незначительном числе полюсов в исследуемой сети. Данное обстоятельство связано с существенной избыточностью полного набора остовых деревьев и массовой включенностью результирующих многополюсных деревьев в качестве их подграфов. Таким образом, в ходе формирования состояний сети оказывается весьма эффективными процедуры как отбрасывания лишних остовых деревьев, проводимая для каждого анализируемого подграфа, так и анализа неполюсных вершин графа на наличие листьев.

8. Заключение. Следует отметить, что многополюсные деревья являются наиболее общим понятием относительно простых цепей и остовых деревьев [32-34]. Первые из них представляют собой многополюсные деревья всего с двумя полюсами, а последние – случай, когда все вершины и есть полюсы. Несмотря на возможность подобного обобщения, в практических приложениях целесообразно рассматривать именно частные случаи простых цепей и остовов вследствие их меньшей вычислительной сложности [35]. Как следствие, можно предположить, что вполне логичным окажется и поиск более эффективных алгоритмов формирования множеств многополюсных деревьев с фиксированным числом полюсов, например конкретно с тремя, четырьмя и так далее, либо с количеством полюсов, зависящим от общего количества вершин в графе, например на единицу, двойку и так далее меньше.

В целом же предложенные методы оказываются вычислительно более эффективными по сравнению с известными при различных исходных данных. Так, метод формирования состояний для двухполюсной связности наиболее предпочтителен в случае незначительно разветвленных сетей, для всеполюсной связности – наоборот, хорошо себя демонстрирует на сравнительно мощных графах, что в совокупности с методом для многополюсной связности, позволяет весьма эффективно решать задачи для малого числа полюсов.

В качестве основного направления дальнейших исследований целесообразно рассматривать анализ и обобщение существующих методов точной оценки вероятности связности, основывающихся на множествах состояний графов. Кроме того, весьма разумным продолжением данной работы можно считать исследование методов формирования состояний графов не как подграфов, описывающих состояния связности, а методов, образующих множества несвязных состояний, то есть сечений.

Литература

1. *Xu S., Qian Y., Hu R.* Reliable and resilient access network design for advanced metering infrastructures in smart grid // IET Smart Grid. 2018. vol. 1. no. 1. pp. 24–30.
2. *Xu S., Qian Y., Hu R.Q.* A data-driven preprocessing scheme on anomaly detection in big data applications // 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). 2017. pp. 814–819.
3. *Ye F., Qian Y., Hu R.Q., Das S.K.* Reliable energy-efficient uplink transmission for neighborhood area networks in smart grid // IEEE Transactions on Smart Grid. 2015. vol. 6(5). pp. 2179–2188.
4. *Ye F., Qian Y., Hu R.Q.* Energy efficient self-sustaining wireless neighborhood area network design for smart grid // IEEE Transactions on Smart Grid. 2014. vol. 6. no. 1. pp. 220–229.
5. *Ye F. et al.* Design and analysis of a wireless sensor based monitoring network for transmission lines in smart grid // Wireless Communications and Mobile Computing. 2016. vol. 16(10). pp. 1209–1220.
6. *Xu S., Qian Y.* Quantitative study of reliable communication infrastructure in smart grid NAN // 2015 11th International Conference on the Design of Reliable Communication Networks (DRCN). 2015. pp. 93–94.
7. *Gharavi H., Hu B.* Multigate communication network for smart grid // Proceedings of the IEEE. 2011. vol. 99(6). pp. 1028–1045.
8. *Lin M., Ting C.* A polynomial-time algorithm for computing K-terminal residual reliability of d-trapezoid graphs // Information Processing Letters. 2015. vol. 115(2). pp. 371–376.
9. *Chaturvedi S.K.* Network Reliability Measures and Evaluation // John Wiley & Sons. 2016. 237 p.
10. *Райгородский А.М.* Модели случайных графов и их применения // Труды МФТИ. 2010. Т. 2. № 4. С. 130–140.
11. ГОСТ Р 53111–2008. Устойчивость функционирования сети связи общего пользования. Требования и методы проверки // М.: Стандартинформ. 2009. 16 с.
12. *Острейковский В.А.* Теория надежности. Учебник // Высшая школа, Абрис 2003. 463 с.

13. *Paredes R.* Network Reliability Estimation in Theory and Practice // Reliability Engineering and System Safety. 2018. 26 p.
14. *Tamm V.* Теория графов // М.: Мир. 1988. 424 с.
15. *Housni K.* An Efficient Algorithm for Enumerating all Minimal Paths of a Graph // International Journal of Advanced Computer Science and Applications. 2019. vol. 10. no. 1. pp. 450–460.
16. *Roberts B.* Estimating the Number of s-t Paths in a Graph // Journal of Graph Algorithms and Applications. 2007. vol. 11. pp. 195–214.
17. *Lin Y.K., Chang P. C.* Maintenance reliability estimation for a cloud computing network with nodes failure // Expert Systems with Applications. 2011. vol. 38. pp. 14–185.
18. *Lin S.G.C.* On performance evaluation of ERP systems with fuzzy mathematics // Expert Systems with Applications. 2009. vol. 36. pp. 6362–6367.
19. *Lin Y.K., Yeh C.T.* Determine the optimal double-component assignment for a stochastic computer network // Omega. 2012. vol. 40. no. 1. pp. 120–130.
20. *Lin Y.K.* A novel algorithm to evaluate the performance of stochastic transportation systems, Expert Systems with Applications // Expert Systems with Applications. 2010. vol. 37. no. 2. pp. 968–973.
21. *Bai G.H., Tian Z.G., Zuo M.J.* An improved algorithm for finding all minimal paths in a network // Reliability Engineering and System Safety. 2016. vol. 150. pp. 1–10.
22. *Зыков А.А.* Основы теории графов // М.: Наука. 1987. 384 с.
23. *Батенков К.А.* Числовые характеристики структур сетей связи // Труды СПИИРАН. 2017. Вып. 4(53). С. 5–28.
24. *Батенков К.А., Батенков А.А.* Анализ и синтез структур сетей связи по детерминированным показателям устойчивости // Труды СПИИРАН. 2018. Вып. 3(58). С. 128–159.
25. *Цицаишвили Г.Ш., Оситова М.А., Лосев А.С.* Асимптотика вероятности связности графа с низконадёжными ребрами // Прикладная дискретная математика. 2013. № 1(19). С. 93–98.
26. *Харари Ф.* Теория графов // М.: Едиториал УРСС. 2003. 296 с.
27. *Aggarwal K., Rai S.* Reliability Evaluation in Computer Communication Networks // IEEE Transaction on Reliability. 1981. vol. 30. no. 1. pp. 32–35.
28. *Белоусов А.И., Ткачев С.Б., Зарубина В.С., Крищенко А.П.* Дискретная математика: Учеб. для вузов // МГТУ им. Н.Э. Баумана. 2004. 744 с.
29. *Батенков К.А.* Синтез детерминированных нелинейных дискретных отображений непрерывных каналов связи // Труды СПИИРАН. 2016. Вып. 2(45). С. 75–101.
30. *Rath D.A., Soman K.P.* Simple Method for Generating k-Trees of a Network // Microelectronics and Reliability. 1993. vol. 33(9). pp. 1241–1244.
31. *Батенков К.А.* Точные и граничные оценки вероятностей связности сетей связи на основе метода полного перебора типовых состояний // Труды СПИИРАН. 2019. Т. 18. № 5. С. 1093–1118.
32. *Zuev K. M., Wu S., Beck J. L.* Network reliability problem and its efficient solution by Subset Simulation // Probabilistic Engineering Mechanics. 2015. pp. 25–35.
33. *Zhang H.C. et al.* Connection effect on amplitude death stability of multi-module floating airport // Ocean Engineering. 2017. pp. 46–56.
34. *Pino W., Gomes T., Kooij R.* A Comparison between Two All-Terminal Reliability Algorithms // Journal of Advances in Computer Networks 2015. vol. 3. no. 4. pp. 284–290.
35. *Silva J. et al.* An effective algorithm for computing all-terminal reliability bounds // Networks. 2015. vol. 66(4). pp. 282–295.

Батенков Александр Александрович — д-р техн. наук, профессор, профессор, кафедра электроники, радиотехники и систем связи, Орловский государственный университет им. И.С. Тургенева. Область научных интересов: статистическая теория связи, модели и методы обработки сигналов, вычислительные алгоритмы. Число научных публикаций — 110. batenkov1957@mail.ru; ул. Комсомольская, 95, 302026, Орёл, Россия; р.т.: +7-906-570-16-66.

Батенков Кирилл Александрович — д-р техн. наук, доцент, сотрудник, Академия Федеральной службы охраны Российской Федерации (Академия ФСО России). Область научных интересов: статистическая теория связи, модели и методы обработки сигналов, анализ качества сетей связи. Число научных публикаций — 150. rustur@yandex.ru; ул. Приборостроительная, 35, 302015, Орёл, Россия; р.т.: +7(4862)54-98-28.

Фокин Александр Борисович — сотрудник, Академия Федеральной службы охраны Российской Федерации (Академия ФСО России). Область научных интересов: анализ качества сетей связи. Число научных публикаций — 5. tatarin57ru@mail.ru; ул. Приборостроительная, 35, 302015, Орёл, Россия; р.т.: +7(4862)54-98-28.

A. BATENKOV, K. BATENKOV, A. FOKIN
**METHODS FOR FORMATION OF TELECOMMUNICATION
NETWORK STATES SETS FOR DIFFERENT MEASURES OF
CONNECTIVITY**

Batenkov A., Batenkov K. Fokin A. Methods for Formation of Telecommunication Network States Sets for Different Measures of Connectivity.

Abstract. Reliability, survivability, and stability analysis tasks are typical not only for telecommunications, but also for systems whose components are subject to one or more types of failures, such as transport, power, mechanical systems, integrated circuits, and even software. The logical approach involves the decomposition of the system into a number of small functional elements, and within telecommunications networks they are usually separate network devices (switches, routers, terminals, etc.), as well as communication lines between them (copper-core, fiber-optic, coaxial cables, wireless media, and other transmission media). Functional relationships also define logical relationships between the failures of individual elements and the failure of the network as a whole. The assumption is also used that device failures are relatively less likely than communication line failures, which implies using the assumption of absolute stability (reliability, survivability) of these devices. Model of a telecommunication network in the form of the generalized model of Erdos–Renyi is presented. In the context of the stability of the telecommunications network, the analyzed property is understood as the connectivity of the network in one form or another. Based on the concept of stochastic connectivity of a network, as the correspondence of a random graph of the connectivity property between a given set of vertices, three connectivity measures are traditionally distinguished: two-pole, multi-pole, and all-pole. The procedures for forming an arbitrary structure of sets of paths and trees for networks are presented, as well as their generalization of multipolar trees. It is noted that multipolar trees are the most common concept of relatively simple chains and spanning trees. Solving such problems will allow us to proceed to calculating the probability of connectivity of graphs for various connectivity measures.

Keywords: Communication Network, Graph, Structure, Connectivity Probability, Two-pole Connectivity, Multi-pole Connectivity, All-pole Connectivity.

Batenkov Aleksandr — Ph.D., Dr.Sci., Professor, Professor, Department of Electronics, Radio Engineering and Communication Systems, Orel State University named after I.S. Turgenev. Research interests: statistical communication theory, models and methods of signal processing, computational algorithms. The number of publications — 110. batenkov1957@mail.ru; 95, Komsomolskaya str., 302026, Orel, Russia; office phone: +7-906-570-16-66.

Batenkov Kirill — Ph.D., Dr.Sci., Associate Professor, Researcher, Academy of Federal Security Guard Service of the Russian Federation. Research interests: statistical communication theory, models and methods of signal processing, communications network quality analysis. The number of publications — 150. pustur@yandex.ru; 35, Priborostroitel'naya str., 302015, Orel, Russia; office phone: +7(4862)54-98-28.

Fokin Aleksandr — Researcher, Academy of Federal Security Guard Service of the Russian Federation. Research interests: communications network quality analysis. The number of publications — 5. tatarin57ru@mail.ru; 35, Priborostroitel'naya str., 302015, Orel, Russia; office phone: +7(4862)54-98-28.

References

1. Xu S., Qian Y., Hu R. Reliable and resilient access network design for advanced metering infrastructures in smart grid. *IET Smart Grid*. 2018. vol. 1. no. 1. pp. 24–30.
2. Xu S., Qian Y., Hu R.Q. A data-driven preprocessing scheme on anomaly detection in big data applications. 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). 2017. pp. 814–819.
3. Ye F., Qian Y., Hu R.Q., Das S.K. Reliable energy-efficient uplink transmission for neighborhood area networks in smart grid. *IEEE Transactions on Smart Grid*. 2015. vol. 6(5). pp. 2179–2188.
4. Ye F., Qian Y., Hu R.Q. Energy efficient self-sustaining wireless neighborhood area network design for smart grid. *IEEE Transactions on Smart Grid*. 2014. vol. 6. no. 1. pp. 220–229.
5. Ye F. et al. Design and analysis of a wireless sensor based monitoring network for transmission lines in smart grid. *Wireless Communications and Mobile Computing*. 2016. vol. 16(10). pp. 1209–1220.
6. Xu S., Qian Y. Quantitative study of reliable communication infrastructure in smart grid NAN. 2015 11th International Conference on the Design of Reliable Communication Networks (DRCN). 2015. pp. 93–94.
7. Gharavi H., Hu B. Multigate communication network for smart grid. *Proceedings of the IEEE*. 2011. vol. 99(6). pp. 1028–1045.
8. Lin M., Ting C. A polynomial-time algorithm for computing K-terminal residual reliability of d-trapezoid graphs. *Information Processing Letters*. 2015. vol. 115(2). pp. 371–376.
9. Chaturvedi S.K. *Network Reliability Measures and Evaluation*. John Wiley & Sons. 2016. 237 p.
10. Raigorodskiy A.M. [Random graph models and their applications]. *Trudy MFTI – Proceedings of MIPT*. 2010. Issue 2. vol. 4. pp. 130–140 (In Russ.).
11. GOST R 53111–2008. [Stability of the public communication network. Requirements and verification methods]. M.: Standardinform. 2009. 16 p. (In Russ.).
12. Ostreikovskiy V.A. *Teoriya nadezhnosti. Uchebnik [Reliability Theory. Textbook]*. Vysshaya shkola, Abris. 2003. 463 p. (In Russ.).
13. Paredes R. *Network Reliability Estimation in Theory and Practice. Reliability Engineering and System Safety*. 2018. 26 p.
14. Tutte W.T. *Graph Theory*. Addison–Wesley Publishing Company. 1984. 333 p. (Russ. ed.: Tatt U. *Teoriya grafov*. M.: Mir. 1988. 424 p.).
15. Housni K. An Efficient Algorithm for Enumerating all Minimal Paths of a Graph. *International Journal of Advanced Computer Science and Applications*. 2019. vol. 10. no. 1. pp. 450–460.
16. Roberts B. Estimating the Number of s-t Paths in a Graph. *Journal of Graph Algorithms and Applications*. 2007. vol. 11. pp. 195–214.
17. Lin Y.K., Chang P.C. Maintenance reliability estimation for a cloud computing network with nodes failure. *Expert Systems with Applications*. 2011. vol. 38. pp. 14–185.
18. Lin S.G.C. On performance evaluation of ERP systems with fuzzy mathematics. *Expert Systems with Applications*. 2009. vol. 36. pp. 6362–6367.
19. Lin Y.K., Yeh C.T. Determine the optimal double-component assignment for a stochastic computer network. *Omega*. 2012. vol. 40. no. 1. pp. 120–130.
20. Lin Y.K. A novel algorithm to evaluate the performance of stochastic transportation systems, *Expert Systems with Applications. Expert Systems with Applications*. 2010. vol. 37. no. 2. pp. 968–973.
21. Bai G.H., Tian Z.G., Zuo M.J. An improved algorithm for finding all minimal paths in a network. *Reliability Engineering and System Safety*. 2016. vol. 150. pp. 1–10.

22. Zykov A.A. *Osnovy teorii grafov* [Fundamentals of graph theory]. M.: Nauka.1987. 384 p. (In Russ.).
23. Batenkov K.A. [Numerical characteristics of the structures of communication networks]. *Trudy SPIIRAN – SPIIRAS Proceedings*. 2017. vol. 4(53). pp. 5–28. (In Russ.).
24. Batenkov K.A., Batenkov A.A. [Analysis and synthesis of communication network structures based on deterministic stability indicators]. *Trudy SPIIRAN – SPIIRAS Proceedings*. 2018. vol. 3(58). pp. 128–159. (In Russ.).
25. Ciciashvili G.Sh., Osipova M.A., Losev A.S. [Asymptotics of the probability of connectivity of a graph with low-confidence edges]. *Prikladnaya diskretnaya matematika – Applied discrete mathematics*. 2013. vol. 1(19). pp. 93–98. (In Russ.).
26. Harari F. *Graph Theory*. M.: Editorial URSS. 2003. (Russ ed.: Harari F. *Teoriya grafov*. M.: Editorial URSS. 2003. 296 p.).
27. Aggarwal K., Rai S. Reliability Evaluation in Computer Communication Networks. *IEEE Transaction on Reliability*. 1981. vol. 30. no. 1. pp. 32–35.
28. Belousov A.I., Tkachev S.B. Diskretnaya matematika: Uchebnik dlya vuzov [*Discrete mathematics: Studies. for higher education institutions*]. MGTU im. N.E. Bauman. 2004. 744 p. (In Russ.).
29. Batenkov K.A. [Synthesis of deterministic nonlinear discrete mappings of continuous communication channels]. *Trudy SPIIRAN –SPIIRAS Proceedings*. 2016. vol. 2(45). pp. 75–101. (In Russ.).
30. Rath D.A., Soman K.P. Simple Method for Generating k-Trees of a Network. *Microelectronics and Reliability*. 1993. vol. 33(9). pp. 1241–1244.
31. Batenkov K.A. [Accurate and boundary estimates of the connectivity probabilities of communication networks based on the method of full iteration of typical States]. *Trudy SPIIRAN –SPIIRAS Proceedings*. 2019. Issue 18. vol. 5. pp. 1093–1118 (In Russ.).
32. Zuev K. M., Wu S., Beck J. L. Network reliability problem and its efficient solution by Subset Simulation. *Probabilistic Engineering Mechanics*. 2015. pp. 25–35.
33. Zhang H.C. et al. Connection effect on amplitude death stability of multi-module floating airport. *Ocean Engineering*. 2017. pp. 46–56.
34. Pino W., Gomes T., Kooij R. A Comparison between Two All-Terminal Reliability Algorithms. *Journal of Advances in Computer Networks*. 2015. vol. 3. no. 4. pp. 284–290.
35. Silva J. et al. An effective algorithm for computing all-terminal reliability bounds. *Networks*. 2015. vol. 66(4). pp. 282–295.