Модел. и анализ информ. систем. Т. **20**, № **6** (2013) 10–21

УДК 517.51+514.17

Общие знания в хорошо структурированных системах с абсолютной памятью

Гаранина Н.О.[1]

Институт систем информатики им. А.П. Ершова СО РАН
630090, Россия, г. Новосибирск, проспект Лаврентьева, 6

e-mail: garanina@iis.nsk.su

получена 22 ноября 2013

Ключевые слова: логика общих знаний, мультиагентные системы с абсолютной памятью, хорошо структурированные системы, проверка моделей

В данной работе исследуется задача проверки моделей для логики общих знаний и неподвижных точек μPLC$_n$ в хорошо структурированных мультиагентных системах с абсолютной памятью. Показано, что среда с абсолютной памятью, порожденная хорошо структурированной средой и снабженная специальным PRS-порядком, образует хорошо структурированную среду. Из этого следует, что проверка моделей для дизъюнктивного фрагмента μPLC$_n$ разрешима.

1. Введение

Комбинации традиционных программных логик [14, 4] с логиками знаний [5] — это хорошо известный формализм для выражения ряда свойств мультиагентных систем [9]. Многие исследования посвящены разработке техник проверки моделей для мультиагентных систем, специфицированных различными комбинированными логиками [2, 3, 11, 15, 16].

Мы исследуем задачу проверки моделей в следовой синхронной мультиагентной системе с абсолютной памятью для различных комбинаций логик знаний с логиками действий и времени. В таких системах агенты имеют своего рода память: их знания зависят от их предыдущих состояний и действий. Мы можем описывать такой тип агентов благодаря тому, что семантика знаний определена на следах, т.е. конечных последовательностях состояний и действий, и каждый агент может отличать следы, содержащие различные последовательности доступной ему информации. Каждый элемент последовательности представляет состояние системы в некоторый момент времени.

[1]Работа выполнена при финансовой поддержке РФФИ (проект № 13-01-00643-а) и президиума РАН (интеграционный проект СО РАН № 15/10 "Математические и методологические аспекты интеллектуальных информационных систем").

Задача проверки моделей для таких систем рассматривалась в [7, 18, 19, 8]. В работе [7] было показано, что задача проверки моделей в классе конечно-порожденных следовых синхронных систем с абсолютной памятью неразрешима для логик *Act*-CTL-C_n, μPLK$_n$, и μPLC$_n$ (при $n > 1$), но разрешима для логики знаний и действий *Act*-CTL-K_n (с неэлементарной нижней границей по времени). Статья [18] представляет прямой алгоритм для проверки логики *Act*-CTL-K_n в синхронных средах с абсолютной памятью. Этот алгоритм проверяет формулы в специальной модели, элементами которой являются "деревья знаний". В статье [19] показано, что эта модель, снабженная специальным частичным порядком на деревьях, образует хорошо структурированную помеченную систему переходов [1, 6], где каждая формула *Act*-CTL-K_n может быть охарактеризована конечным вычислимым множеством максимальных деревьев, удовлетворяющих данному свойству. Эта особенность модели на деревьях позволяет снизить временную сложность предложенного алгоритма проверки моделей, но она всё ещё высока. В работе [8] новая форма деревьев знаний делает возможным экспоненциальное снижение временной сложности предыдущего варианта алгоритма проверки моделей для *Act*-CTL-K_n.

Однако задача проверки моделей для более мощных логик знаний, содержащих модальности общих знаний, является неразрешимой в системах с абсолютной памятью и, насколько нам известно, не было предложено алгоритмов проверки моделей даже для фрагментов этой логики. Одним из возможных путей решения данной задачи представляется применение какого-либо специального вида мультиагентных систем с абсолютной памятью. В данной статье мы предлагаем для этой цели использовать хорошо структурированные помеченные системы переходов [1, 6], расширенные агентами. Это является основой для комбинирования подхода, основанного на знаниях, к протоколам передачи данных [10] и хорошо структурированных систем переходов как моделей для взаимодействующих систем с ненадежной коммуникацией [6].

В данной статье синхронная мультиагентная система с абсолютной памятью (PRS-среда) порождается хорошо структурированной мультиагентной системой, обладающей свойством совместимости агентов и действий. Мы покажем, что такая PRS-среда, снабженная специальным порядком, также является хорошо структурированной системой. Как следствие, задача проверки моделей для дизъюнктивного фрагмента логики с общими знаниями и неподвижными точками μPLC$_n$ оказывается разрешимой.

2. Базовые логики и модели

Приведем адаптированные для данной работы определения пропозициональной логики с общими знаниями и неподвижными точками из [7]. Эта логика является комбинацией пропозициональной логики с общими знаниями (PLC) [5] и μ-исчисления (μC) [13]. Семантика μPLC$_n$ определена в терминах отношения выполнимости $\models$ в средах, которые являются специальным видом помеченных систем переходов.

Наше определение среды отличается от традиционных из работ [5, 7] и др. тем, что мы явно определяем отношение групповой неразличимости, хотя она выражается через отношения индивидуальной неразличимости.

Определение 1. *(Среда)*
Среда — это набор $E = (D, \sim, \approx, I, V)$, где

- D — непустое множество состояний (или миров);
- для каждого агента $i \in [1..n]$, отношение неразличимости $\overset{i}{\sim}$ является отношением эквивалентности на D $\sim : [1..n] \to 2^{D\times D}$;
- для каждой группы агентов $G \subseteq [1..n]$, отношение групповой неразличимости $\overset{G}{\approx}$ является отношением эквивалентности на D $\approx : 2^{[1..n]} \to 2^{D\times D}$, при этом $\overset{G}{\approx} = (\bigcup_{i\in G} \overset{i}{\sim})^*$;
- интерпретация действий I — всюду определённое отображение $I : Act \to 2^{D\times D}$;
- означивание V — всюду определённое отображение $V : Prp \to 2^D$.□

Каждое отношение неразличимости, которое не является равенством, отражает тот факт, что агент (группа агентов) имеет неполную информацию о состояниях системы. Из определения среды непосредственно следует, что группа агентов G не различает миры w и w' среды E, т.е. $w \overset{G}{\approx} w'$ тогда и только тогда, когда существует конечная последовательность миров $ws \in D^+$ и конечная последовательность агентов $as \in G^+$ такие, что w — первый мир в ws, w' — последний мир в ws, $|ws| = |as|+1$ и $ws_j \overset{as_j}{\sim} ws_{j+1}$ для всех $j \in [1..|as|]$.

Пусть $\{true, false\}$ — булевы константы, Prp и Act — непересекающиеся конечные алфавиты пропозициональных переменных и символов действий, а конечное множество натуральных чисел $[1..n]$ представляет имена агентов ($n \in \mathbb{N}$).

Определение 2. *(синтаксис μPLC_n)*
Синтаксис μPLC$_n$ состоит из формул, построенных из булевых констант, пропозициональных переменных, связок $\neg$, $\wedge$, $\vee$, и следующих правильных конструкций. Пусть $i \in [1..n]$ — агент, $G \subseteq [1..n]$ — группы агентов, $a \in Act$ — действие, $x \in Prp$ — пропозициональная переменная, и φ — формула логики μPLC$_n$. Тогда правильными конструкциями являются

- модальности знаний: $K_i\varphi$ и $S_i\varphi$
(читается как ‘агент i знает φ’ и ‘агент i предполагает φ’);
- модальности общих знаний: $C_G\varphi$ и $J_G\varphi$
(читается как ‘для агентов из группы G факт φ является общеизвестным’ и ‘агенты из группы G имеют общую гипотезу φ’);
- модальности действий: $[a]\varphi$ и $\langle a\rangle\varphi$
(читается как ‘после действия a необходимо выполняется φ’ и ‘после действия a возможно выполняется φ’);
- неподвижные точки: $\mu x.\varphi$ и $\nu x.\varphi$
(читается как ‘наименьшая неподвижная точка формулы φ’ и ‘наибольшая неподвижная точка формулы φ’).□

Семантика μPLC$_n$ следует семантикам логик PLC [5] и μC [13].

Определение 3. *(Семантика μPLC_n)*
Отношение выполнимости $\models$ между средами, мирами и формулами определяется индуктивно по структуре формулы. Для булевых констант, пропозициональных переменных и связок отношение выполнимости стандартно. Для модальностей знаний, общих знаний, действий и неподвижных точек семантика определяется следующим образом. Пусть E — среда, $w \in D$, $i \in [1..n]$, $G \subseteq [1..n]$, $a \in Act$, φ формула логики μPLC$_n$. Пусть $S \subseteq D$, $x \in Var$, обозначим $E_{S/x}$ среду, которая равна среде E повсюду, кроме $V(x) = S$. Для формулы ϕ без негативных вхождений x, $\lambda S.\ E_{S/x}(\phi)\ :\ S \mapsto E_{S/x}(\phi)$ является монотонным неубывающим отображением на 2^D наименьшей и наибольшей неподвижными точками $\mu(\lambda S.\ E_{S/x}(\phi))$ и $\nu(\lambda S.\ E_{S/x}(\phi))$. Тогда

- $w \models_E K_i\varphi \Leftrightarrow$ для каждого w': $w \stackrel{i}{\sim} w'$ влечёт $w' \models_E \varphi$;
- $w \models_E S_i\varphi \Leftrightarrow$ для некоторого w': $w \stackrel{i}{\sim} w'$ и $w' \models_E \varphi$;
- $w \models_E C_G\varphi \Leftrightarrow$ для каждого w': $w \stackrel{G}{\approx} w'$ влечёт $w' \models_E \varphi$;
- $w \models_E J_G\varphi \Leftrightarrow$ для некоторого w': $w \stackrel{G}{\approx} w'$ и $w' \models_E \varphi$;
- $w \models_E [a]\varphi \Leftrightarrow$ для каждого w': $(w, w') \in I(a)$ влечёт $w' \models_E \varphi$;
- $w \models_E \langle a\rangle\varphi \Leftrightarrow$ для некоторого w': $(w, w') \in I(a)$ и $w' \models_E \varphi$;
- $w \models_E \mu x.\varphi \Leftrightarrow w \in \mu(\lambda S.\ E_{S/x}(\varphi))$;
- $w \models_E \nu x.\varphi \Leftrightarrow w \in \nu(\lambda S.\ E_{S/x}(\varphi))$.

Семантикой формулы φ в среде E является множество всех миров E, в которых формула φ выполняется: $E(\varphi) = \{w \mid w \models_E \varphi\}$.$\square$

Заметим, что модальности K_i (S_i) — это синтаксический сахар для групповых модальностей C_G (J_G) при $G = \{i\}$. Далее мы рассматриваем только *нормальные формулы* μPLC$_n$, в которых отрицание применяется только к пропозициональным переменным. Каждая формула μPLC$_n$ эквивалентна некоторой нормальной формуле в силу законов “Де Моргана”.

Мы исследуем следовую синхронную среду с абсолютной памятью (PRS-среду), порожденную базовой конечной средой. В PRS-среде (1) состояния являются последовательностями миров исходной среды с историей порождающих действий; (2,3) агент (группа агентов) не различает такие последовательности, если базовая среда исполняет ту же последовательность действий и агент (группа) не может отличить эти последовательности шаг за шагом; (4) переход из одной последовательности в другую посредством действия a осуществляется расширением исходной последовательности состоянием, достижимым с помощью a из последнего состояния этой последовательности; (5) пропозициональные переменные означиваются в последнем состоянии последовательностей в соответствии с их означиванием в базовой среде.

Определение 4. *(PRS-среда)*
Пусть E — это среда $(D, \sim, \approx, I, V)$. Следовая *среда с абсолютной памятью*, порожденная средой E, — это другая среда $E_{PRS} = (D_{PRS(E)}, \underset{\text{prs}}{\sim}, \underset{\text{prs}}{\approx}, I_{PRS(E)}, V_{PRS(E)})^2$:

(1) $D_{PRS(E)}$ — множество всех пар (ws, as), где $ws \in D^*$, $as \in Act^*$ не пусты, $|ws| = |as| + 1$, и $(ws_j, ws_{j+1}) \in I(as_j)$ для каждого $j \in [1..|as|]$;

пусть $(ws, as),\ (ws', as') \in D_{PRS(E)}$:

(2) для каждого $i \in [1..n]$: $(ws, as) \overset{i}{\underset{\text{prs}}{\sim}} (ws', as') \Leftrightarrow$
$$as = as' \text{ и } ws_j \overset{i}{\sim} ws'_j \text{ для каждого } j \in [1..|ws|];$$

(3) для каждого $G \subseteq [1..n]$: $(ws, as) \overset{G}{\underset{\text{prs}}{\approx}} (ws', as') \Leftrightarrow$
$$as = as' \text{ и } ws_j \overset{G}{\approx} ws'_j \text{ для каждого } j \in [1..|ws|];$$

(4) для каждого $a \in Act$: $((ws, as),\ (ws', as')) \in I_{PRS(E)}(a) \Leftrightarrow$
$$as' = as^\wedge a,\ ws' = ws^\wedge w', \text{ и } (w_{|ws|}, w') \in I(a);$$

(5) для каждого $p \in Prp$: $(ws, as) \in V_{PRS(E)}(p) \Leftrightarrow ws_{|ws|} \in V(p).\square$

В PRS-средах агенты имеют своего рода память, поскольку их осведомленность выражена посредством отношений неразличимости, зависящих от истории развития системы.

Мы изучаем задачу проверки моделей для логики μPLC$_n$ в синхронных средах с абсолютной памятью, порожденных конечными средами [18].

Определение 5. *(Задача проверки моделей для μPLC_n)*
Задачей проверки моделей для μPLC$_n$ в синхронных средах с абсолютной памятью является подтвердить или опровергнуть $(ws, as) \models_{PRS(E)} \varphi$, где E — конечная среда, $(ws, as) \in D_{PRS(E)}$, φ — формула логики μPLC$_n$.

2.1. Пример среды с абсолютной памятью: Начальник и Подчиненный

Приведем простой пример среды, порождаемой ею среды с абсолютной памятью и формулы логики общих знаний с неподвижными точками. Рассмотрим игру под названием Начальник-Подчиненный (Boss-Inferior $B\&I$). *Начальник* загадал трансцендентное число, например π, а *Подчиненный* старается его угадать. Подчиненный называет произвольные целые числа, а Начальник сопоставляет эти числа с подпоследовательностями цифр задуманного числа, начиная с начала. Если число не совпадает с текущим отрезком последовательности цифр задуманного числа, то Начальник не принимает это число и выписывает Подчиненному штраф. Если очередное число совпадает с текущим отрезком последовательности, то Начальник принимает это число и выписывает Подчиненному премию. При этом в дальнейшем Подчиненный не может называть принятое однажды число, а Начальник сдвигает начало очередной подпоследовательности соответственно принятому

[2] В определении для каждого множества S запись S^* означает множество всех конечных последовательностей над S и операция $^\wedge$ — это конкатенация конечных слов.

числу. Цель Подчиненного угадать число, т.е. не получать штрафов, что означает хорошую работу. В данной модели для нас (и Начальника) не имеет значения, как именно подчиненный будет угадывать задуманное число. Однако мы считаем, что Подчиненный должен обладать памятью, чтобы учитывать принятые числа при выборе очередного кандидата. Поэтому для формализации данной задачи необходимо задать базовую среду, по которой строится среда с абсолютной памятью.

Рассмотрим следующее высказывание ST: Начальник знает, что Подчиненный знает, что Начальник знает, $\overset{\infty}{\ldots}$, что Подчиненный знает, что Начальник знает, что Подчиненный хорошо работает. Это общее знание обеспечивает уверенность Подчиненного в том, что ему увольнение не грозит, и уверенность Начальника в том, что Подчиненный и дальше будет стараться. Ниже мы выразим это высказывание с помощью логики $\mu\mathrm{PLC}_n$. Начальника и Подчиненного можно рассматривать как процессы, соединенные ненадежным каналом связи, надежность которого, т.е. количество испорченных сообщений, изменяется со временем.

Чтобы задать среду $B/I = (D_{B/I}, \sim, \approx, I_{B/I}, V_{B/I})$ для игры $B\&I$, необходимо задать (1) агентов, (2) состояния среды, (3) отношения неразличимости агентов, (4) множество действий и их интерпретацию и (5) пропозициональные переменные и их означивание.

1. Агенты

Среда содержит двух агентов: B для Начальника и I для подчиненного, $A = \{B, I\}$.

2. Состояния среды

Состояния среды задаются значениями следующих локальных переменных:
(1) $\pi \in \mathbb{N}^+$ — (неполная) последовательность натуральных чисел, кодирующая число π;
(2) $Lst \in \mathbb{N}$ — последнее принятое число;
(3) $Num \in \mathbb{N} \cup \{null\}$ — число-кандидат, названное Подчиненным;
(4) $Nms \subseteq \mathbb{N}$ — подмножество натуральных чисел, которые может называть Подчиненный;
(5) $Acc \in \mathbb{N}$ — счет Подчиненного.
Таким образом, каждое состояние это $(\pi, Lst, Num, Nms, Acc) \in D_{B/I} = \mathbb{N}^+ \times \mathbb{N} \times \mathbb{N} \times 2^{\mathbb{N}} \times \mathbb{N}$. Для $w \in D_{B/I}$ и локальной переменной $Var \in \{\pi, Lst, Num, Nms, Acc\}$ обозначим значение локальной переменной Var в состоянии w как $w(Var)$.

3. Отношения неразличимости агентов

Определим наблюдаемые переменные агентов.
$Obs_B = \{\pi, Lst, Num, Acc\}$ — Начальник видит число, которое он задумал, последнее принятое число, число-кандидат и счет Подчиненного, чтобы начислять штрафы и премии.
$Obs_I = \{Lst, Num, Nums, Acc\}$ — Подчиненный видит последнее принятое число, число-кандидат, оставшиеся у него в распоряжении числа и свой счет.
Отношения неразличимости определяются равенством наблюдаемых переменных, т.е. для всех $w, w' \in D_{B/I}$ и $i \in \{B, I\}$ верно $w \overset{i}{\sim} w' \Leftrightarrow \forall\ Var \in Obs_i\ w(Var) = w'(Var)$.

4. Множество действий и их интерпретация

Опишем действия агентов и их интерпретацию в терминах пред- и постусловий: $(pre_1, ..., pre_5) \longrightarrow (post_1, ..., post_5)$, где для каждого $j \in [1..5]$ $pre_j/post_j$ — предусловие/постусловие для соответствующей переменной.

Действия Начальника.
(1) принять число $accept$:
$$(\pi = n\hat{\ }postfix, Lst = m, Num = n, Nms = S, Acc = k) \longrightarrow$$
$$(\pi = postfix, Lst = n, Num = null, Nms = S \setminus \{n\}, Acc = k + 1)$$
Начальник стирает у задуманного им числа префикс, соответствующий предложенному числу, и выписывает премию в переменной Acc, при этом последнее угаданное число фиксируется в Lst и удаляется из множества чисел Nms, которые можно предложить в дальнейшем;
(2) отвергнуть число $reject$:
$$(\pi = n\hat{\ }postfix, Lst = m, Num = p, Nms = S, Acc = k) \longrightarrow$$
$$(\pi = n\hat{\ }postfix, Lst = m, Num = null, Nms = S, Acc = k - 1)$$
Начальник выписывает штраф в переменной Acc.
Действия Подчиненного.
(1) предложить число try :
$$(\pi = n\hat{\ }postfix, Lst = m, Num = null, Nms = S, Acc = k) \longrightarrow$$
$$(\pi = n\hat{\ }postfix, Lst = m, Num = x, Nms = S, Acc = k)$$
Подчиненный предлагает число в переменной Num.
5. Пропозициональные переменные и их означивание
Определим переменную, характеризующую хорошую работу подчиненного:
$GoodJob = \{w \in D_{B/I} \mid w(Acc) > 0\}$.

Теперь можно выразить высказывание ST в логике $\mu\mathrm{PLC}_n$ следующим образом: $ST = \mu x.\ (C_{\{B,I\}} GoodJob \vee [try]([accept]x \vee [reject]x)$, т.е. как бы ни развивались события в системе, наступит момент, когда Начальник и Подчиненный будут обладать общим знанием, что работа выполняется хорошо.

3. Хорошо структурированные системы с абсолютной памятью

Мы рассматриваем хорошо структурированные среды. Сущность техники проверки моделей с бесконечным числом состояний [1, 6] для хорошо структурированных систем переходов состоит в том, что можно проверять формулы только в некотором конечном репрезентативном множестве состояний, а не во всём пространстве состояний. Напомним основные определения из [19].

Определение 6. *(Хорошо предупорядоченная система переходов)*
Пусть D — множество. *Хороший предпорядок* — это рефлексивное и транзитивное бинарное отношение $\leq$ на D, где каждая бесконечная последовательность $d_1, \ldots d_i, \ldots$ элементов D содержит пару элементов d_m и d_n, таких что m меньше n и $d_m \leq d_n$. Пусть $(D, \leq)$ — *хорошо предупорядоченное множество* с хорошим предпорядком $\leq$. *Идеал* (синоним: *конус*) — это замкнутое вверх подмножество D, т.е. множество $\{C \subseteq D \mid \forall d', d'' \in D : d' \leq d'' \wedge d' \in C \Rightarrow d'' \in C\}$. Каждый элемент $d \in D$ генерирует *конус* $(\uparrow d) \equiv \{e \in D \mid d \leq e\}$. Для каждого подмножества $S \subseteq D$, *базис* S — это множество его минимальных элементов $\{B \subseteq S \mid \forall s \in S\ \exists b \in B : b \leq s\}$. *Хорошо предупорядоченная система переходов (WPTS)* — это тройка $(D, \leq, I)$, такая что $(D, \leq)$ — хорошо предупорядоченное множество и (D, I) — структура Крипке.$\square$

Мы рассматриваем хорошо предупорядоченные системы переходов, в которых хороший предпорядок и интерпретация разрешимы и совместимы. Условие разрешимости для хорошего предпорядка стандартно: отношение $\leq$ является разрешимым.

Определение 7. *(Модель с идеалами)*
Пусть $(D, \leq, I)$ является WPTS.

- *Условие разрешимости (вычислимое прошлое)*: существует вычислимая всюду определенная функция $BasPre : D \times Act \to 2^D$ такая, что для каждого $w \in D$, для каждого $a \in Act$, $BasPre(w, a)$ является конечным базисом $\{u \in D \mid (u, v) \in I(a) \text{ и } w \leq v\}$.

- *Условие совместимости*: предпорядок R совместим с интерпретацией $I(a)$ каждого символа действий $a \in Act$, т.е.
$$\forall s_1, s_2, s_1' \ \exists s_2' \ : \ s_1 \xrightarrow{I(a)} s_1' \wedge s_1 \leq s_2 \ \Rightarrow \ s_2 \xrightarrow{I(a)} s_2' \wedge s_1' \leq s_2'.$$

Модель с идеалами — это помеченная система переходов $\mathcal{I} = (D, \leq, I, V)$ с разрешимым предпорядком $\leq$, $\mathcal{I}$ удовлетворяет условиям совместимости и вычислимого прошлого, и интерпретация $V(p)$ каждой пропозициональной переменной $p \in Prp$ является конусом.$\square$

Здесь мы рассматриваем индивидуальные и групповые отношения неразличимости как, в некотором роде, “действия знаний”, поскольку они ассоциированы с соответствующими модальностями. Введем понятие обратной *совместимости отношений неразличимости и действий*, которое отражает тот естественный факт, что каждое действие $a \in Act$ трансформирует одинаковую локальную информацию агентов одним и тем же образом в разных состояниях системы.

Определение 8. *(Совместимость неразличимости)*
Для каждого действия $a \in Act$, для каждой пары миров $(w_1, w_1') \in I(a)$ и каждого мира w_2' такого, что $w_1' \sim w_2'$ существует w_2 такой, что $(w_2, w_2') \in I(a)$ и $w_1 \sim w_2$.$\square$

Очевидно, что если действия совместимы с индивидуальной неразличимостью, то они совместимы с групповой неразличимостью.

Определение 9. *(Среда с идеалами)*
Среда с идеалами — это набор $\mathcal{E} = (D, \leq, \sim, \approx, I, V)$, где $E = (D, \sim, \approx, I, V)$ является средой, в которой I удовлетворяет условиям совместимости с неразличимостью и $\mathcal{I} = (D, \ \leq, \ \sim \cup \approx \cup \ I, \ V)$ — модель с идеалами.$\square$

Пусть далее $\mathcal{E}$ — среда с идеалами и $\mathcal{E}_{PRS}$ — это ее PRS-среда. Определим бинарное отношение *PRS-порядок* $\preceq$ на $D_{PRS(\mathcal{E})}$.

Определение 10. *(PRS-порядок $\preceq$)*
Для всех следов равной длины (ws, as) и (ws', as') в $D_{PRS(\mathcal{E})}$ пишем $(ws, as) \preceq (ws', as')$ если, и только если $ws_{|ws|} \leq ws'_{|ws|}$.$\square$

Теорема 1. *Бинарное отношение $\preceq$ является частичным порядком на следах таким, что PRS-среда $\mathcal{E}_{PRS}$, снабженная этим частичным порядком, оказывается средой с идеалами.*

Эскиз доказательства. Доказательство довольно техническое, поэтому представим здесь только основные идеи.

Во-первых, $\preceq$ является частичным порядком. Это так же хороший предпорядок и разрешимое отношение в силу того, что отношение $\leq$ имеет те же свойства.

Во-вторых, $\preceq$ удовлетворяет условию вычислимого прошлого, поскольку мы можем найти прообраз каждого следа для каждого перехода по 'действиям', для каждого перехода по 'знаниям' и 'общим знаниям' (определяемым $I_{PRS(\mathcal{E})}(a)$, $\underset{\text{prs}}{\overset{i}{\approx}}$, и $\underset{\text{prs}}{\overset{G}{\approx}}$, соответственно для $a \in Act$, $i \in [1..n]$, и $G \subseteq [1..n]$, в определении 4), поскольку каждый след имеет конечную длину и порождающая среда $\mathcal{E}$ удовлетворяет условию вычислимого прошлого.

В-третьих, $\preceq$ совместимо со всеми переходами по 'действиям' и всеми переходами по 'знаниям' и 'общим знаниям'. (1) для каждого действия $a \in Act$ и для каждой пары следов $(ws_1, as_1) \preceq (ws_1', as_1')$ существует следующий след $(ws_2', as_2') = (ws_1'^\wedge w', as_1'^\wedge)$ такой, что $(ws_1^\wedge w, as_1^\wedge) \preceq (ws_2', as_2')$ по определению PRS-порядка и совместимости действий в порождающей среде $\mathcal{E}$. (2) для каждого действия знаний $\underset{\text{prs}}{\overset{i}{\approx}}$ $(i \in [1..n])$, и для каждой пары следов $(ws_1, as_1) \preceq (ws_1', as_1')$ с $(ws_1, as_1) \underset{\text{prs}}{\overset{i}{\approx}} (ws_2, as_2)$ существует некоторый след $(ws_2', as_2') \underset{\text{prs}}{\overset{i}{\approx}} (ws_1', as_1')$ такой, что $(ws_2, as_2) \preceq (ws_2', as_2')$, поскольку каждый след имеет конечную длину и имеет место совместимость неразличимости и совместимость действий в порождающей среде $\mathcal{E}$. (3) каждое действие общих знаний $\underset{\text{prs}}{\overset{G}{\approx}}$ $(G \subseteq [1..n])$ совместимо с $\preceq$ по тем же причинам, что и действия знаний $\underset{\text{prs}}{\overset{i}{\approx}}$ $(i \in [1..n])$.

В-четвертых, $\mathcal{E}_{PRS}$ является моделью с идеалами. Очевидно, что означивание каждой пропозициональной переменной образует конус с базисом, состоящим из следов, в которых последнее состояние содержится в базисе соответствующего конуса означивания этой переменной в $\mathcal{E}$.■

Легко показать, что можно выразить модальности общих знаний через модальности индивидуальных знаний и неподвижные точки, используя следующие *эквивалентности общих знаний* [7]:

– $C_G\varphi \leftrightarrow \nu x.(\varphi \wedge (\bigwedge_{i\in G} K_i x))$; – $J_G\varphi \leftrightarrow \mu x.(\varphi \vee (\bigvee_{i\in G} S_i x))$.

Приведем определение из [12].

Определение 11. *(Дизъюнктивный фрагмент μ-исчисления)*
Дизъюнктивными формулами μ-исчисления являются:

$$\varphi ::= p \mid (\varphi \vee \varphi) \mid \langle a\rangle\varphi \mid \mu x.\varphi.$$

Следующая теорема доказана в [12].

Теорема 2. *Задача проверки моделей разрешима для моделей с идеалами и дизъюнктивных формул пропозиционального μ-исчисления. Она также разрешима для формул интуиционистской модальной логики с наименьшими неподвижными точками $\mu\boldsymbol{FS}$ в моделях с вычислимым прошлым.*

Определение 12. *(Дизъюнктивный фрагмент μPLC_n)*
Дизъюнктивными формулами μ-исчисления являются μPLC_n:

$$\varphi ::= p \mid (\varphi \vee \varphi) \mid \langle a\rangle\varphi \mid S_i\varphi \mid J_G\varphi \mid \mu x.\varphi.$$

Следствие 1 является прямым выводом теоремы 1 данной статьи, теоремы 2 из [12] и эквивалентностей общих знаний:

Следствие 1. *Задача проверки моделей разрешима для PRS-сред с идеалами и дизъюнктивных формул логики μPLC_n.*

3.1. Пример среды с идеалами: Начальник и Подчиненный

Введем порядок $\leq$ на состояниях среды B/I и покажем, что среда с этим порядком является средой с идеалами. Пусть $w_1, w_2 \in D_{B/I}$. Мы считаем, что $w_1 \leq w_2$, если и только если $w_1.Lst \notin w_2.Nms$ и $w_1.Acc \leq w_2.Acc$.

Утверждение 1. *Бинарное отношение $\leq$ является частичным порядком на состояниях среды B/I, таким что B/I снабженная этим частичным порядком оказывается средой с идеалами.*

Эскиз доказательства.

(1) Порядок $\leq$ является хорошим предпорядком по определению.

(2) Среда удовлетворяет условию вычислимого прошлого, поскольку прообраз каждого действия вычислим.

(3) Порядок совместим с действиями в силу того, что принятые числа не повторяются в дальнейшем.

(4) Действия в среде совместимы с отношениями неразличимости, поскольку они одинаковую локальную информацию агентов преобразуют одним и тем же образом в разных состояниях системы.

(5) Означивание пропозициональной переменной $GoodJob$ является конусом, поскольку зависит от локальной переменной Acc. ■

К сожалению, формула ST, выражающая уверенность Начальника и Подчиненного друг в друге, не входит в дизъюнктивный фрагмент $\mu\mathrm{PLC}_n$. Ему принадлежит более слабая формула $ST' = \mu x.\ (J_{\{B,I\}}GoodJob \vee \langle try\rangle(\langle accept\rangle x \vee \langle reject\rangle x)$, которая говорит о том, что существует возможность, что агенты могут выдвинуть общую гипотезу о том, что работа выполняется хорошо.

4. Заключение

В данной статье показано, что синхронные среды с абсолютной памятью над хорошо структурированными средами с совместимостью неразличимости и конической интерпретацией пропозициональных переменных, снабженные PRS-порядком, являются средами с идеалами. Этот факт делает возможной проверку моделей для дизъюнктивного фрагмента логики общих знаний и неподвижных точек $\mu\mathrm{PLC}_n$.

В статье [18] предложен метод проверки моделей для несравнимого фрагмента логики $\mu\mathrm{PLC}_n$, а именно *Act*-CTL-K_n, состоящего из формул с ограниченной глубиной знаний. Этот метод развивался в работах [19] и [8], но данная техника исключала модальности общих знаний.

Дальнейшие исследования включают следующие темы: (1) поиск более широкого фрагмента $\mu\mathrm{PLC}_n$, чем дизъюнктивный, с более интересными формулами, содержащими настоящие знания, для которого возможна проверка моделей; (2) комбинирование подхода, основанного на знаниях, к задаче передачи данных [10] и хорошо структурированных систем переходов с ненадежной коммуникацией [6]; (3) проверка моделей реальных хорошо структурированных систем.

Список литературы

1. Abdulla P.A., Ĉerâns K., Jonsson B., and Tsay Yih-Kuen. Algorithmic analysis of programs with well quasi-ordered domains // Information and Computation. 2000. V. 160(1–2). P. 109–127.

2. Bordini R.H., Fisher M., Visser W., Wooldridge M. Verifying Multi-agent Programs by Model Checking // Autonomous Agents and Multi-Agent Systems. 2006. 12(2). P. 239–256.

3. Cohen M., Lomuscio A. Non-elementary speed up for model checking synchronous perfect recall // Proceeding of the 2010 conference on ECAI 2010. Amsterdam, IOS Press, 2010. P. 1077-1078.

4. Emerson E.A. Temporal and Modal Logic // Handbook of Theoretical Computer Science. V. B. Elsevier and MIT Press, 1990. P. 995–1072.

5. Fagin R., Halpern J.Y., Moses Y., Vardi M.Y. Reasoning about Knowledge. MIT Press, 1995.

6. Finkel A., Schnoebelen Ph. Well-structured transition systems everywhere! // Theor. Comp. Sci. 2001. V. 256(1–2). P. 63–92.

7. Garanina N.O., Kalinina N.A. and Shilov N.V. Model checking knowledge, actions and fixpoints // Proc. of Concurrency, Specification and Programming Workshop CS&P'2004, Germany, 2004. Humboldt Universitat, Berlin, Informatik-Bericht Nr. 170, V. 2. P. 351–357.

8. Garanina N.O. Exponential Improvement of Time Complexity of Model Checking for Multiagent Systems with Perfect Recall // Programming and Computer Software. 2012. Vol. 38, No 6. P. 294–303.

9. Halpern J.Y., van der Meyden R., Vardi M.Y. Complete Axiomatizations for Reasoning About Knowledge and Time // SIAM J. Comp. 2004. 33(3). P. 674–703.

10. Halpern J.Y. and Zuck L.D. A little knowledge goes a long way: knowledge-based derivations and correctness proofs for a family of protocols // Journal of the ACM. 1992. 39(3). P. 449–478.

11. Huang X., van der Meyden R. The Complexity of Epistemic Model Checking: Clock Semantics and Branching Time // Proc. of 19th ECAI. Lisbon, Portugal, August 16–20, Frontiers in Artificial Intelligence and Applications. V. 215. IOS Press, 2010. P. 549–554.

12. Kouzmin E.V., Shilov N.V., Sokolov V.A. Model Checking μ-Calculus in Well-Structured Transition Systems // Proc. of 11th International Symposium on Temporal Representation and Reasoning (TIME'04). France, IEEE Press. P. 152–155.

13. Kozen D. Results on the Propositional Mu-Calculus // Theoretical Computer Science. 1983. V. 27, No 3. P. 333–354.

14. Kozen D., Tiuryn J. Logics of Programs // Handbook of Theoretical Computer Science. V. B. Elsevier and MIT Press, 1990. P. 789–840.

15. Kwiatkowska M.Z., Lomuscio A., Qu H. Parallel Model Checking for Temporal Epistemic Logic // Proc. of 19th ECAI. Lisbon, Portugal, August 16-20, Frontiers in Artificial Intelligence and Applications. V. 215. IOS Press, 2010. P. 543–548.

16. Lomuscio A., Penczek W., Qu H. Partial order reductions for model checking temporal epistemic logics over interleaved multi-agent systems // Proc. of 9th AAMAS, Toronto, Canada, May 10-14, 2010. V. 1. IFAAMAS, 2010. P. 659–666.

17. van der Meyden R., Shilov N.V. Model Checking Knowledge and Time in Systems with Perfect Recall // Lect. Notes Comput. Sci. 1999. V. 1738. P. 432–445.

18. Shilov N.V., Garanina N.O., and Choe K.-M. Update and Abstraction in Model Checking of Knowledge and Branching Time // Fundamenta Informaticae. 2006. V. 72. No 1–3. P. 347–361.

19. Shilov N.V., Garanina N. O. Well-structured Model Checking of Multiagent Systems // Proceedings of 6th International Conference on Perspectives of System Informatics, Novosibirsk, Russia, June 27-30, 2006 // Lecture Notes in Computer Science. 2006. V. 4378.

Common Knowledge in Well-structured Perfect Recall Systems

Garanina N.O.

A.P. Ershov Institute of Informatics Systems RAS, Siberian Branch
Acad. Lavrentjev pr., 6, Novosibirsk, 630090, Russia

Keywords: logic of common knowledge, perfect recall, well-structured systems, model checking

We investigate a model checking problem for the logic of common knowledge and fix-points $\mu\mathrm{PLC}_n$ in well-structured multiagent systems with perfect recall. In this paper we show that a perfect recall synchronous environment over a well-structured environment forms a well-structured environment provided with a special PRS-order. This implies that the model checking problem for the disjunctive fragment of $\mu\mathrm{PLC}_n$ is decidable.

Сведения об авторе:
Гаранина Наталья Олеговна,
Институт систем информатики им. А.П. Ершова СО РАН,
лаборатория Теоретического программирования,
научный сотрудник.