УДК 519.68:681.51

ВЕРОЯТНОСТНЫЕ МУЛЬТИАГЕНТНЫЕ СИСТЕМЫ: СЕМАНТИКА И ВЕРИФИКАЦИЯ[1]

Валиев М.К.*, **Дехтярь М.И.****
* Институт прикладной математики им. М.В.Келдыша РАН
** Тверской государственный университет, кафедра информатики

Поступила в редакцию 27.11.2008, после переработки 03.12.2008.

Рассматриваются системы взаимодействующих вероятностных интеллектуальных агентов. У них имеются два источника неопределенности: неопределенность времени передачи сообщений в каналах связи и неопределенность в результатах выполняемых действий. Показано, как таким системам за полиномиальное время могут быть сопоставлены конечные цепи Маркова. Это позволяет перенести известные результаты о верификации динамических свойств конечных цепей Маркова на вероятностные мультиагентные системы.

Probabilistic systems of interacting intelligent agents are considered. They have two sources of uncertainty: uncertainty of communication channels and uncertainty of the results of actions. We show how such systems can be polynomially transformed to finite state Markov chains. This allows one to transfer known results on verifying temporal properties of the finite state Markov chains to the probabilistic multi-agent systems of considered type.

Ключевые слова: мультиагентная система, логическая программа, цепь Маркова, темпоральная логика, верификация, сложность алгоритмов.
Keywords: multiagent system, logic program, Markov chain, temporal logic, verification, complexity of algorithms.

1. Введение

Системы взаимодействующих интеллектуальных агентов представляют собой одно из самых активно развивающихся направлений в области искусственного интеллекта и прикладного программирования. (см., например, [14, 15, 21, 23]). Агенты используются в качестве автономных компонент при построении систем искусственного интеллекта, работы в Интернете, обработке изображений, полученных со спутников, управлении бизнес процессами, электронной торговле и др. В таких системах агенты могут работать и взаимодействовать по весьма сложным правилам, что делает поведение системы во времени очень сложным. Это приводит к

[1]Работа выполнена при финансовой поддержке РФФИ, проекты 07-01-00637 и 08-01-00241.

необходимости разработки формальных методов для верификации динамических свойств таких систем.

В литературе известно несколько подходов, связанных с определением моделей агентов, мультиагентных систем (МАС) и языков спецификации свойств таких систем. В частности, в [17, 19] рассматривается поведение абстрактных агентов без внутренней структуры, в [2, 11] агенты специфицируются формулами некоторых темпоральных логик. Другой популярный подход к описанию агентов основан на модели «Believe-Desire-Intention», предложенной в [14] (см. также [3, 4, 20]). В наших предыдущих работах [7, 8] мы рассматривали сложность верификации динамическиз свойств систем агентов, построенных на основе архитектуры IMPACT, введенной в [16].

Во всех этих работах предполагается, что агенты действуют, имея полную и точную информацию о своем состоянии и окружающем мире, а передача информации от одного агента другому происходит без всяких потерь и занимает некоторое фиксированное время. При этом различаются детерминированные и недетерминированные системы. В первых текущее состояние однозначно определяет выбор выполняемых действий и следующее состояние системы. В недетерминированных системах в текущем состоянии имеется несколько разных допустимых действий и, следовательно, возможны несколько различных следующих состояний. При анализе поведения таких систем рассматриваются все возможные варианты. Однако в реальных приложениях агенты часто имеют неполную и неточную информацию о своем состоянии, окружающем мире, времени передачи сообщений и о результатах выполняемых действий, и при этом известны лишь распределения вероятностей для соответствующих параметров агентов и системы. Это приводит к тому, что для данного текущего состояния возможные переходы в одно из нескольких следующих состояний имеют, вообще говоря, различные вероятности. При анализе поведения таких систем естественно говорить о вероятностях выполнения для них тех или иных свойств.

В работе [10] была предложена модель вероятностных агентов, в которой основной причиной неопределенности была неточность знаний агента о своем состоянии и окружающем мире. Могут быть также и другие источники неопределенности поведения МАС. В этой работе мы рассматриваем два из них: неопределенность времени передачи сообщений между агентами в каналах связи и неопределенность в результатах выполняемых агентами действий. А именно, мы предполагаем, что продолжительность доставки сообщений по каналам может быть случайной, и некоторые сообщения могут теряться при передаче. Кроме того, действия агентов могут иметь альтернативы, выбор которых для выполнения производится с некоторыми вероятностями. Однако мы предполагаем, что сам выбор действий для выполнения является детерминированным, т.е. рассматриваемые МАС не являются недетерминированными (concurrent) в смысле М.Варди [18].

Основной результат статьи состоит в том, любая такая вероятностная МАС может быть эффективно преобразована в некоторую конечную Марковскую цепь с полиномиально вычислимыми (относительно размера МАС) вероятностями переходов.[2] Имеется некоторое число статей, посвященных исследованию сложности верификации конечных Марковских цепей. Наше преобразование МАС в Мар-

[2]Некоторые варианты этого утверждения были без доказательств сформулированы нами в работах [9, 22].

ковские цепи позволяет применить результаты этих статей к оценке сложности верификации поведения МАС с вероятностными каналами и действиями.

Упомянем некоторые из этих статей. Исследование сложности верификации конечных марковских цепей было начато в вышеупомянутой работе Варди. Его результаты по сложности верификации формул пропозициональной логики линейного времени (PLTL) на Марковских цепях и процессах принятия решений были улучшены в [6]. Аналогичные результаты для верификации формул вероятностных логик ветвящегося времени (PCTL and PCTL*) были получены в [12, 13].

Статья организована следующим образом. Раздел 2 содержит синтаксическое определение нашего варианта вероятностных МАС. В разделе 3 описывается операционная семантика этих МАС. Раздел 4 содержит алгоритм вычисления вероятностей переходов для цепей Маркова, соответствующих этим МАС. В разделе 5 приводятся результаты по сложности верификации вероятностных МАС, полученных применением результатов из [6, 12].

2. Вероятностные МАС

В этом разделе мы опишем рассматриваемую модель вероятностной мультиагентной системы. Наши определения в основном (с некоторыми упрощениями) следуют [16].

Мультиагентная система (МАС) $\boldsymbol{A} = \{A_1,...,A_n\}$ состоит из конечного множества $\{A_1,...,A_n\}$ взаимодействующих интеллектуальных агентов A_i. У интеллектуального агента A имеется *внутренняя база данных* (БД) I_A, содержащая конечное множество базисных атомов (т.е. выражений вида $p(c_1,\ldots, c_k)$, где p — предикатный символ, $c_1,\ldots, c_k$ — константы, причем множество используемых данной системой констант ограничено), и почтовый ящик $MsgBox_A$. Текущие содержимые внутренней БД и почтового ящика агента A составляют его текущее локальное состояние $IM_A =< I_A, MsgBox_A >$.

Агенты из $\boldsymbol{A}$ общаются между собой посредством передачи сообщений вида msg(Sender, Receiver, Msg), где Sender и Receiver — имена агентов (источника и адресата), а Msg — (передаваемый) базисный атом.

Для каждой пары агентов A, B из $\boldsymbol{A}$ имеется канал связи CH_{AB}, в который попадают сообщения, посылаемые агентом A агенту B. Затем из этого канала они попадают в почтовый ящик $MsgBox_B$. Время пребывания каждого сообщения «в пути» мы будем рассматривать как случайную величину, задаваемую конечным дискретным распределением вероятностей. Через $p_{AB}(t)$ обозначим вероятность того, что B получит сообщение, посланное ему агентом A, ровно через $t \geq 1$ шагов (тактов) после его отсылки (t0 будет обозначать минимальное число такое, что $p_{AB}(t) = 0$ для всех $t > t0$.

Для разных сообщений соответствующие случайные величины будем считать независимыми. Мы будем предполагать, что $\sum_{t=1}^{\infty} p_{AB}(t) \leq 1$. Тогда разность $1 - \sum_{t=1}^{\infty} p_{AB}(t)$ определяет вероятность того, что сообщение никогда не достигнет адресата, т.е. будет утеряно в канале. Например, $p_{AB}(1) = 1$ означает, что каждое сообщение, посланное агентом A агенту B, будет получено адресатом на следующем шаге после его посылки. Когда это верно для всех используемых в системе распределений вероятностей, мы приходим к синхронным МАС, которые рассматривались в работах [7, 8]. Если $p_{AB}(1)$=0.5, $p_{AB}(2)$=0.4 и $p_{AB}(t)$=0 при t

>2, то половина всех сообщений A агенту B будет получена на следующем шаге после их отсылки, еще 4/10 будут находиться в пути 2 такта, и в среднем десятая часть таких сообщений будет утеряна.

Текущее состояние канала CH_{AB} будет включать все сообщения, посланные агентом A агенту B, которые еще не дошли до B, с указанием времени их нахождения в канале. Мы будем обозначать текущее состояние канала так же как и сам канал, т.е. $CH_{AB} = \{(Msg, t)|$ *сообщение Msg от агента A агенту b находится в канале t тактов*$\}$. Мы будем также использовать сокращения CH_{ij} и p_{ij} для CH_{AiAj} и p_{AiAj}, соответственно.

С каждым агентом A связана его база ACT_A *параметризованных действий.* Каждое из этих действий имеет имя вида a(X_1,...,X_m) и множество альтернатив
$a^1 =< ADD_a^1(X_1, \ldots, X_m), DEL_a^1(X_1, \ldots, X_m), SEND_a^1(X_1, \ldots, X_m) >, \ldots,$
$a^k =< ADD_a^k(X_1, \ldots, X_m), DEL_a^k(X_1, \ldots, X_m), SEND_a^k(X_1, \ldots, X_m) >.$
На этих альтернативах определено некоторое вероятностное распределение $\mathrm{p}_a(\mathrm{j})$. Множества $\mathrm{ADD}_a^j(\mathrm{X}_1,\ldots,\mathrm{X}_m)$ и DEL_a^j $(\mathrm{X}_1,\ldots,\mathrm{X}_m)$ — списки атомов вида $\mathrm{p}(\mathrm{t}_1,\ldots, \mathrm{t}_k)$, где p — k-местный предикат из сигнатуры внутренней БД, $\mathrm{t}_1,\ldots,\mathrm{t}_k$ — либо константы, либо параметры $\mathrm{X}_1,\ldots,\mathrm{X}_m$. Эти множества определяют изменения внутренней БД (добавления и удаления фактов) при выполнении данной альтернативы. SEND_a^j $(\mathrm{X}_1,\ldots,\mathrm{X}_m)$ определяет аналогично список сообщений вида msg(A,B, $\mathrm{p}(\mathrm{t}_1,\ldots,\mathrm{t}_k)$), отправляемых другим агентам. Пусть $\mathrm{c}_1,\ldots,\mathrm{c}_m$ — константы. Обозначим через ADD_a^j $(\mathrm{c}_1,\ldots,\mathrm{c}_m)$ множество фактов, получаемых подстановкой $\mathrm{c}_1,\ldots,\mathrm{c}_m$ вместо $\mathrm{X}_1,\ldots,\mathrm{X}_m$ в атомы из ADD_a^j $(\mathrm{X}_1,\ldots,\mathrm{X}_m)$. Аналогично определяются DEL_a^j $(\mathrm{c}_1,\ldots,\mathrm{c}_m)$ и SEND_a^j $(\mathrm{c}_1,\ldots,\mathrm{c}_m)$. Базисные атомы вида $\mathrm{a}(\mathrm{c}_1,\ldots, \mathrm{c}_l)$ назовем базисными именами действий (или просто базисными действиями).

Конкретный выбор действий для выполнения в зависимости от текущего локального состояния агента определяется парой $< \mathrm{LP}_A$, $\mathrm{Sel}_A >$. Здесь LP_A — *логическая программа* агента, определяющая совместно с фактами из текущего локального состояния агента некоторое множество Perm ($= \mathrm{Perm}_{At}$) допустимых для выполнения в данный момент базисных имен действий, а полиномиально вычислимая функция Sel_A выделяет из Perm некоторое базисное действие $\mathrm{a}(\mathrm{c}_1,\ldots\mathrm{c}_q)$, альтернативы которого должны выполняться в текущий момент с соответствующими вероятностями (один из естественных вариантов функции выбора связан с определением некоторого отношения приоритета на действиях). Выполнение такой альтернативы (скажем, a^j) состоит в следующем:

1) следующее состояние внутренней БД агента A получается из текущего состояния удалением элементов из DEL_a^j $(\mathrm{c}_1,\ldots,\mathrm{c}_m)$ и добавлением элементов из ADD_a^j $(\mathrm{c}_1,\ldots,\mathrm{c}_m)$,

2) для каждого агента $\mathrm{B} \neq \mathrm{A}$ в канал CH_{AB} добавляются пары вида (Ms, 0) для всех экземпляров сообщений Ms, для которых msg(A, B, Ms) $\in \mathrm{SEND}_a^j$ $(\mathrm{c}_1,\ldots,\mathrm{c}_m)$.

В качестве программы LP_A мы рассматриваем логическую программу с предложениями вида

$$H : -L_1, ..., L_n$$

со следующими свойствами: H — атом действия, т.е. имеет вид $a(t_1, \ldots, t_m)$, где a — имя действия, $t_1, \ldots, t_m$ — либо константы, либо переменные; литералы L_i — либо литералы действия, либо (экстенсиональные) литералы с предикатами из сигнатуры внутренней БД, либо литералы сообщений вида msg(Sender, A, Msg) или **not** msg(Sender, A, Msg), либо некоторые вычислимые в полиномиальное время встроенные предикаты. Программа агента называется *базисной*, если она не содержит переменных. Система является *базисной*, если базисными являются программы всех ее агентов.

Мы предполагаем, что предложения в программах агентов являются *безопасными* в том смысле, что все переменные из головы предложения H входят *позитивно* в тело предложения $L_1,...,L_n$. Кроме того, мы считаем, что программа LP_A является стратифицированной [1]. Тогда в каждом локальном состоянии *state*= $<I_A, MsgBox_A>$ программа

$$LP_{A,state} = LP_A \cup I_A \cup MsgBox_A,$$

определяющая множество действий, которые в принципе может выполнить агент в текущем состоянии, является стратифицированной.

Хорошо известно (см. [1]), что стратифицированные логические программы имеют единственную минимальную модель. Обозначим такую модель программы $LP_{A,state}$ через $M_{A,state}$. Стандартная процедура вычисления неподвижной точки позволяет вычислить эту модель по базисной развертке gr($LP_{A,state}$) программы $LP_{A,state}$. Отметим, что размер gr($LP_{A,state}$) может быть экспонентой от размера $LP_{A,state}$. Напомним также, что предположение о замкнутости области, которое мы используем, предполагает вычислимость за полиномиальное время всех встроенных предикатов. Поэтому сохраняется вычислимость неподвижной точки за полиномиальное время.

Упомянутое выше множество Perm определяется тогда как множество базисных имен действий, содержащихся в минимальной модели
$M_{A,state}$. Обозначим через Sem функцию, которая по $LP_{A,state}$ строит это множество Perm.

3. Семантика вероятностных МАС

Глобальное состояние S системы **A** включает в себя локальные состояния ее агентов и состояния всех ее (n^2-n) каналов:

$$S =< I_1, \ldots, I_n; CH_{1,2}, CH_{2,1}, \ldots, CH_{n-1,n}, CH_{n,n-1} > .$$

Обозначим через $\mathbf{S}_A$ множество всех глобальных состояний МАС **A.** Тогда *одношаговая семантика* МАС **A** задает отношение S $\Rightarrow_A$ S′ перехода (за один шаг) на множестве $\mathbf{S}_A$, а вероятности $p_{i,j}(t)$ и $p_a(j)$ индуцируют вероятности таких переходов $p(S, S')$.

Переход S $\Rightarrow_A$ S′ начинается с работы каналов и формирования нового содержимого почтовых ящиков. Сначала каждый канал увеличивает на 1 счетчик времени у всех находящихся в нем сообщений. Пары вида (Msg, t), у которых $t>t_0$, удаляются из канала $CH_{i,j}$ Затем для каждой пары (Msg, t) $\in CH_{i,j}$ в почтовый ящик $MsgBox_j$ агента A_j с вероятностью $p_{i,j}(t)$ помещается факт msg(A_i, A_j ,Msg). После этого каждый агент $A_i \in \mathbf{A}$ формирует множество всех допустимых на данном шаге действий $Perm_i$ = Sem($LP_{A_i,state}$). А затем, используя свой оператор выбора Sel_{A_i}, определяет *выполняемое действие* $\mathrm{a}(\mathrm{c}_1,\ldots \mathrm{c}_q)$. Почтовые ящики всех агентов МАС **A** после этого опустошаются, т.е. полученные сообщения «забываются». Разумеется, это не ограничивает общности, поскольку агент может все нужные ему данные перенести из почтового ящика в свою базу данных. Затем с вероятностью $\mathrm{p}_a(\mathrm{j})$ выбирается альтернатива a^j действия $\mathrm{a}(\mathrm{c}_1,\ldots \mathrm{c}_q)$ и состояния внутренней БД I_i и соответствующих каналов заменяются в соответствии с определением действия этой альтернативы.

Таким образом, переход $\mathrm{S} \Rightarrow_A \mathrm{S}'$ вычисляется следующим вероятностным алгоритмом:

A-шаг (**Вход:** S ; **Выход:** S′)

FOR EACH $A_i, A_j \in \mathbf{A}$ $(i \neq j)$ **DO**
    FOR EACH $(Msg, t) \in CH_{i,j}$ **DO**
        BEGIN $CH_{i,j} := (CH_{i,j} \ \setminus \ \{(Msg, t)\})$;
        IF $\mathrm{t} \leq \mathrm{t}_0$ **then** $\mathrm{CH}_{i,j} := (\mathrm{CH}_{i,j} \cup \{(\mathrm{Msg},\ \mathrm{t}+1)\})$
        END;
FOR EACH A_i, $\mathrm{A}_j \in \mathbf{A}$ $(\mathrm{i} \neq \mathrm{j})$ **DO**
    FOR EACH $(\mathrm{Msg},\ \mathrm{t}) \in \mathrm{CH}_{i,j}$ **DO** с вероятностью $p_{i,j}(t)$
        BEGIN $\mathrm{CH}_{i,j} := (\mathrm{CH}_{i,j} \setminus \{(\mathrm{Msg},\ \mathrm{t})\})$;
        $\mathrm{MsbBox}_j := \mathrm{MsbBox}_j \cup \{\mathrm{msg}(\mathrm{A}_i,\ \mathrm{A}_j\ ,\ \mathrm{Msg})\}$
        END;
FOR EACH $\mathrm{A}_j \in \mathbf{A}$ **DO**
    BEGIN $state_j := (I_j, MsgBox_j)$;
    $Perm_j := Sem(LP_{A_j,state_j})$;
    Пусть значение $Sel_{A_j}(Perm_j)$ равно $a_j(c_1,\ldots c_q)$;
    Пусть $a_j^1,\ldots,\ a_j^k$ — все альтернативы для a_j;
    Выберем альтернативу a_j^r $(1 \leq r \leq k)$ с вероятностью $p_{aj}(r)$;
    $I_j' := ((I_j \setminus DEL_{aj}^r(c_1,\ldots c_q)) \ \cup ADD_{aj}^r(c_1,\ldots c_q))$;
    FOR EACH $(\mathrm{m} \neq \mathrm{j})$ **DO**
        $CH_{j,m}' := (CH_{j,m} \cup$
            $\{(ms, 0) \mid msg(A_j, A_m, ms) \in SEND_{aj}^r(c_1,\ldots,c_q)\})$;
        $MsgBox_j \ := \ \emptyset$;
    END;
RETURN S′ .

В соответствии с вышеприведенным определением семантики с системой ***A*** можно связать Марковскую цепь **MC(A)** с множеством состояний $\mathbf{S}_A$ и вероятностями переходов $\mathrm{p}(\mathrm{S},\ \mathrm{S}')$ между ними (алгоритм вычисления $\mathrm{p}(\mathrm{S},\ \mathrm{S}')$ описан в следующем разделе). Поведение ***A*** в начальном глобальном состоянии S^0 описывается деревом $\mathrm{T}_A(\mathrm{S}^0)$ возможных траекторий этой цепи Маркова, начинающихся с S^0. Узлы этого дерева помечены глобальными состояниями системы, причем каждый узел, находящийся на (t+1)-ом уровне и помеченный состоянием S′ , связан с узлом на t-ом уровне с меткой S, из которого возможен переход $\mathrm{S} \Rightarrow_A \mathrm{S}'$ с вероятностью $\mathrm{p}(\mathrm{S},\ \mathrm{S}') > 0$.

Заметим, что количество состояний цепи **MC(A)** в худшем случае может быть экспоненциальным относительно размера **A**, если **A** — базисная, и даже дважды экспоненциальным, если **A** — не базисная ((в размер $|\mathbf{A}|$ системы **A** входят размеры всех сигнатур, множества констант, описаний агентов, включающих их базы действий и описания программ агентов, и распределения вероятностей).

4. Вычисление вероятностей переходов

Отметим, что источниками неопределенности в алгоритме **A-шаг** являются операторы в строках 5-9 и 14, которые определяют, как сообщения попадают в почтовые ящики агентов с учетом вероятностей времен их пересылки и как выбираются альтернативы для действия a_j. Мы предполагаем, что все вероятностные выборки независимы.

Пусть на очередном шаге выполнен переход $S \ \Rightarrow_{\mathbf{A}} \ S'$ из состояния

$$S =< I_1, \ldots, I_n; CH_{1,2}, \ldots CH_{i,j}, \ldots, CH_{n,n-1} >$$

в состояние

$$S' =< I'_1, \ldots, I'_n; CH'_{1,2}, \ldots CH'_{i,j}, \ldots, CH'_{n,n-1} > .$$

Следующие утверждения об алгоритме **A-шаг**, позволяют оценить вероятность такого перехода. Первое из них связано с формированием почтовых ящиков агентов и соответствующими изменениями «старых» сообщений в каналах.

Лемма 1. *(1) После завершения цикла в стр. 5-9 состояние почтового ящика $MsgBox_j$ агента A_j равно $\bigcup_{i \neq j} \bigcup_{m \in M_{ij}} \{msg(A_i, A_j, m)\}$, где множество M_{ij} сообщений, полученных на этом шаге агентом A_j от агента A_i, определено равенством*

$$M_{ij} = \{m \mid \exists t \leq t0[((m,t) \in CH_{ij}) \ \& \ ((m, t+1) \notin CH'_{i,j})]\}.$$

(2) Пусть $T_{ij}(m) = \{t \mid ((m, t-1) \in CH_{ij}) \ \& \ ((m,t) \notin CH'_{i,j})\}$. Тогда

(а) вероятность попадания сообщения $msg(A_i, A_j, m)$ в $MsgBox_j$ равна

$$pm_{ij}(m) = \prod_{t \in T_{ij}(m)} p_{ij}(t);$$

(б) вероятность получения агентом A_j множества сообщений M_{ij}от агента A_i равна

$$pm_{ij} = \prod_{m \in M_{ij}} pm_{ij}(m);$$

(в) вероятность получения агентом A_j множества сообщений $MsgBox_j$ равна

$$pm_j = \prod_{i \neq j} pm_{ij}.$$

Доказательство. (1) Пусть некоторое сообщение $m \in M_{ij}$. По определению M_{ij} это означает, что некоторая пара $(m, t), t \leq t0$, в состоянии S находилась в канале CH_{ij}, а после завершения перехода исчезла из этого канала: $(m, t+1) \notin CH'_{i,j}$.

Единственная возможность для «бесследного» исчезновения сообщения с временем пребывания $t \leq t0$ из канала — это быть удаленным в стр. 7 алгоритма **А-шаг**. Но тогда в стр. 8 этого алгоритма сообщение $msg(A_i, A_j, m)$ попало в почтовый ящик $MsgBox_j$ агента A_j. Обратно, если $msg(A_i, A_j, m) \in MsgBox_j$, то это могло произойти только при некотором выполнении оператора в стр. 8. Но тогда в предыдущей стр. 7 для некоторого $t,\ 0 \leq t \leq t0$, пара $(m, t+1)$ была удалена из CH_{ij}. Для этого она должна была ранее попасть в CH_{ij} в стр. 4. Но это возможно лишь, когда пара (m, t) удаляется из CH_{ij} в стр. 3. Тогда из определения M_{ij} следует, что $m \in M_{ij}$.
(2) В канале CH_{ij} может одновременно находиться несколько экземпляров сообщения m, посланного агентом A_i агенту A_j в разное время. Причиной появления этого сообщения в $MsgBox_j$ в рассматриваемом переходе являются те пары (m, t), входящие в CH_{ij} после цикла в стр. 2-5, которые удаляются в стр. 7. (Отметим, что сообщение $msg(A_i, A_j, m)$, которое появляется в $MsgBox_j$ в стр. 8, не зависит от t). Множество $T_{ij}(m)$ содержит времена пребывания в канале CH_{ij} из всех таких пар (m, t). Каждая из них была независимо выбрана в стр. 6 с вероятностью $p_{i,j}(t)$. Поэтому вероятность их совместного выбора равна $pm_{ij}(m) = \prod_{t \in T_{ij}(m)} p_{ij}(t)$. Пункты (б) и (в) следуют из утверждения (1) и независимости получения разных сообщений одного агента и сообщений от разных агентов. $\square$

Следующее утверждение позволяет оценить вероятность требуемого изменения базы данных каждого агента и посылки им новых сообщений.

Лемма 2. *Пусть в переходе $S \Rightarrow_{\mathbf{A}} S'$ для агента A_j в результате шагов в стр. 11-13 было определено выполняемое действие $a_j(c_1, \ldots c_q)$ с возможными альтернативами $a_j^1, \ldots, a_j^k$. Тогда*

(1) множество $Alts_j$ альтернатив, которые могли использоваться на этом переходе, состоит из таких a_j^r, для которых

(а) $I'_j := ((I_j \setminus DEL^r_{aj}(c_1, \ldots c_q)) \cup ADD^r_{aj}(c_1, \ldots c_q))$ и

(б) для каждого $m \neq j$ $\{ms | (ms, 0) \in CH'_{j,m}\} = SEND^r_{a_j}(c_1, \ldots, c_q)$;

(2) вероятность pr_j того, что будет выбрана альтернатива, выполнение которой приведет к состоянию I'_j агента A_j и к помещению в каждый канал $CH_{jm}(m \neq j)$ множества новых сообщений $\{ms | (ms, 0) \in CH'_{j,m}\}$ равна

$$\sum_{r \in Alts_j} p_{aj}(r).$$

Доказательство. Доказательство пункта (1) непосредственно следует из определения I'_j в стр. 16 алгоритма **А-шаг** и $CH'_{j,m}$ в стр. 18 этого алгоритма. Заметим, что в момент добавления в $CH'_{j,m}$ новых сообщений все старые сообщения в канале имеют временные метки $t > 0$. Поэтому множество пар вида $(m, 0)$ в $CH'_{j,m}$ однозначно определено множеством $SEND^r_{aj}(c_1, \ldots, c_q)$. Выполнение любой альтернативы действия $a_j(c_1, \ldots c_q)$ из множества $Alts_j$, удовлетворяющей свойствам (а) и (б), приводит к тем же множествам I'_j и $\{ms | (ms, 0) \in CH'_{j,m}\}$. Если же для некоторой альтернативы одно из свойств (а) или (б) не выполнено, то ее выполнение, очевидно, не может привести к переходу $S \Rightarrow_{\mathbf{A}} S'$. Поскольку все альтернативы из $Alts_j$ независимы и попарно несовместны, то вероятность выбора одной из них в стр. 15 равна $pr_j = \sum_{r \in Alts_j} p_{aj}(r)$. □

Эти леммы позволяют предложить следующую эффективную процедуру вычисления вероятности $p(S, S')$ перехода S $\Rightarrow_A$ S$'$:

Алгоритм *Prob(S, S$'$)*

(1) **FOR EACH** A_i, $A_j \in \mathbf{A}$ $(i \neq j)$ **DO**
(2) **BEGIN** M[i,j] := $\{(m, t) \mid ((m, t) \in CH_{i,j})$ &
$((m, t+1) \notin CH'_{i,j})\ )\}$;
(3) $pm_{ij} := \prod_{(m,t) \in M[i,j]} p_{i,j}(t)$
(4) **END**;
(5) **FOR EACH** $A_j \in \mathbf{A}$ **DO**
(6) **BEGIN** MsgBox$_j$:= $\emptyset$;
(7) **FOR EACH** $A_i \in \mathbf{A}$ $(i \neq j)$ **DO**
(8) $MsgBox_j := MsgBox_j \cup \{msg(A_i, A_j, m) \mid \exists t((m, t) \in M[i, j])\}$;
(9) $pm_j := \prod_{i \neq j} pm_{ij}$
(10) **END**;
(11) **FOR EACH** $A_j \in \mathbf{A}$ **DO**
(12) **BEGIN** $state_j := (I_j, MsgBox_j)$;
(13) Perm$_j$:= Sem(LP$_{A_j, state_j}$);
(14) Пусть значение Sel$_{A_j}$ (Perm$_j$) равно $a_j(c_1, \ldots c_q)$;
(15) Пусть $a_j^1, \ldots, a_j^k$ — все альтернативы для a_j;
(16) **FOR** $r = 1$ **TO** k **DO**
(17) **BEGIN** $L_r := (I_j \setminus DEL^r_{aj}(c_1, \ldots, c_q)) \cup ADD^r_{aj}(c_1, \ldots, c_q)$;
(18) **FOR EACH** $A_m \in \mathbf{A}$, $m \neq j$ **DO**
(19) $MS_r[j, m] := \{ms \mid msg(A_j, A_m, ms) \in SEND^r_{a_j}(c_1, \ldots, c_q)\}$
(20) **END**;
(21) $Alts_j := \{r \mid (I'_j = L_r)$ &
$\bigwedge_{m \neq j}(\{ms \mid (ms, 0) \in CH'_{j,m}\} = MS_r[j, m])$;
(22) $pr_j := \sum\{p_{aj}(r) \mid r \in Alts_j\}$
(23) **END**;
(24) $p(S, S') := \prod_{A_j \in \mathbf{A}} pm_j\ pr_j$;
(25) **RETURN** $p(S, S')$.

Теорема 1. *Алгоритм* $\boldsymbol{Prob}(S, S')$ *вычисляет вероятность* $p(S, S')$ *перехода* $S \Rightarrow_{\mathbf{A}} S'$ *за время, полиномиальное от суммы размеров МАС* $\boldsymbol{A}$ *и размеров исходного и результирующего состояний* S *и* S'*:* $|\boldsymbol{A}| + |S| + |S'|$.

Доказательство. Из определения семантики одного шага в алгоритме **А-шаг** следует, что переход $S \Rightarrow_{\mathbf{A}} S'$ однозначно определяется следующими событиями: выбором сообщений, помещаемых в почтовые ящики агентов в начале шага (стр. 1-9) и выбором одной из альтернатив выполняемого действия для каждого агента (стр. 15). Действительно, содержимое ящика $MsgBox_j$ агента A_j однозначно определяет множества «старых» пар сообщений вида (m, t) при $t > 0$ в каналах связи CH'_{ij} $(i \neq j)$. Выбор альтернативы действия однозначно определяет следующее состояние I'_j базы данных агента A_j и множество «новых» сообщений вида $(m, 0)$. Лемма 1 2(в) определяет формулу для вычисления вероятности pm_j получения заданного ящика $MsgBox_j$. Вычисления этих вероятностей для всех агентов очевидным образом реализованы в стр.1-4 и 9 алгоритма **Prob**. Правильность вычисления содержимого $MsgBox_j$ в стр. 6-8 следует из пункта (1) этой леммы. Так как функции Sem и Sel работают детерминированно, то выполняемое действие $a_j(c_1, \ldots c_q)$ в стр. 14 определено верно. По лемме 2 множество $Alts_j$ альтернатив, приводящих к нужному результату, определенное в стр. 21 и вероятность pr_j выбора одной из этих альтернатив в стр. 22 вычисляются правильно Тогда из независимости действий различных агентов следует, что вероятность $p(S, S')$ перехода $S \Rightarrow_{\mathbf{A}} S'$ в стр. 24 также вычисляется верно.

Для оценки времени работы алгоритма заметим, что каждое сообщение в каналах связи системы рассматривается не более одного раза в стр. 2 и 8. Вычисления функций Sem и Sel в стр. 13 и 14 требуют времени, полиномиально зависящего от размеров их аргументов. Число итераций циклов в стр. 5-10 и 11-20 равно числу агентов, а цикла в стр. 1-4 — не более квадрата этого числа. Действия с базами действий и базами данных агентов в стр. 17 и 21 требуют времени, пропорционального их размерам. Отсюда следует, что время выполнения алгоритма $\mathbf{Prob}(S, S'$ ограничено полиномом от размера его входных данных. □

5. Верификация динамических свойств МАС

Традиционно свойства поведения дискретных динамических систем задаются с помощью формул некоторых темпоральных логик (см., например, [5]). Имеются два основных вида таких логик: логики линейного времени и логики ветвящегося времени. Обычно состояния цепей Маркова не наделяются никакой внутренней структурой. Поэтому динамические свойства таких цепей можно адекватно выразить формулами пропозициональных вариантов таких логик.

Состояния рассматриваемых нами МАС имеют структуру конечных моделей 1-го порядка. Поэтому естественно расширить логики, описывающие их динамические свойства, включив средства логик 1-го порядка (как это сделано в [7, 8]). Рассматриваемое расширение состоит в том, что вместо пропозициональных переменных могут быть использованы обычные замкнутые формулы 1-го порядка в сигнатурах внутренних баз данных агентов и сигнатуре сообщений. Основное множество модели, соответствующей состоянию

$$S =< I_1, \ldots, I_n, CH_{1,2}, \ldots, CH_{i,j}, \ldots, CH_{n,n-1} >,$$

включает все встречающиеся в описании системы константы и целые числа. Не ограничивая общности, будем считать, что имена предикатов в базах данных различных агентов не пересекаются. Пусть k-местный предикат p относится к агенту A_i и $c_1, \ldots, c_k$ — некоторые константы. Тогда факт $p(c_1, \ldots, c_k)$ истинен в модели, соответствующей S, если $p(c_1, \ldots, c_k) \in I_i$, и ложен в противном случае. Сигнатура сообщений содержит имена предикатов-сообщений, индексированные номерами агентов системы и временами задержки сообщений в каналах связи:

$\Sigma_{msg} = \{msg\_p_{i,j,t}(X_1, \ldots, X_k) \mid 1 \leq i, j \leq n,\ t \in \mathbf{N},\ p$ — k-местный предикатный символ $\}$. Базисный факт $msg\_p_{i,j,t}(c_1, \ldots, c_k)$ истинен на глобальном состоянии S, если канал $CH_{i,j}$ содержит пару $(msg(A_i, A_j, p(c_1, \ldots, c_k)), t)$.

Хорошо известно, что проверка замкнутых формул 1-го порядка на конечных моделях может быть проведена в полиномиальной памяти(или даже за полиномиальное время для формул с ограниченной кванторной глубиной).

Задачи верификации динамических свойств для логик линейного и ветвящегося времени формулируются по-разному:

- *Линейное время*: для заданной вероятностной МАС $\mathbf{A}$, ее начального состояния S^0 и FLTL-формулы F, описывающей некоторое свойство траекторий, найти меру (вероятность) $p_A(S^0, F)$ множества траекторий дерева $T_A(S^0)$, удовлетворяющих F. Если эта вероятность равна 1, то скажем, что пара $(\mathbf{A}, S^0)$ удовлетворяет F.
- *Ветвящееся время*: главную роль в логиках ветвящегося времени играют формулы состояний (а не траекторий). Мера выполнимости таких формул не отражает стохастических свойств поведения системы. Поэтому в работе [12] было предложено заменить кванторы по траекториям на границы вероятностей. Например, формула $[Gf]_{>p}$ означает, что мера траекторий, идущих из текущего состояния, на всех состояниях которых выполнена формула f, больше чем p. Получившаяся логика названа PCTL.

Теперь можно сформулировать некоторые из многих результатов о сложности верификации динамических свойств МАС, которые получаются переносом соответствующих результатов для цепей Маркова.

5.1 Линейное время

В этом случае применим теорему 3.1.2.1 из статьи [6]. Она утверждает, что существуют два алгоритма: 1) алгоритм, проверяющий, удовлетворяет ли данная цепь Маркова $\mathbf{M}$ PLTL-формуле F, время работы которого $O(|M| 2^{|F|})$, а память может быть ограничена полиномом от $|F|$ или полиномом от логарифма $|M|$, и 2) алгоритм, вычисляющий вероятность $p_M(F)$ выполнения F на $\mathbf{M}$ за время экспоненциальное от $|F|$ и полиномиальное от $|M|$.

Для применения этой теоремы достаточно использовать теорему 1 и замечание из конца раздела об оценке размера $\mathbf{MC(A)}$ через размер $\mathbf{A}$. Мы приводим только некоторые из получаемых следствий.

Теорема 2. *(1) Существует алгоритм, который проверяет выполнимость FLTL-формулы F на состоянии S базисной вероятностной МАС $\boldsymbol{A}$ в памяти, полиномиальной от $|\boldsymbol{A}|$ и $|F|$.*

(2) Существует алгоритм, вычисляющий вероятность $p_A(S^0, F)$ для базисной вероятностной МАС $\boldsymbol{A}$ и формулы F за время, экспоненциально зависящее от $|\boldsymbol{A}|$ и $|F|$.

(3) Существует алгоритм, вычисляющий вероятность $p_A(S^0, F)$ для любой (небазисной) вероятностной МАС $\boldsymbol{A}$ и формулы F за время, экспоненциально зависящее от $|\boldsymbol{A}|$ и дважды экспоненциальное от $|F|$.

5.2 Ветвящееся время

В работе [12] построен алгоритм, который проверяет выполнимость PCTL-формулы F на цепи Маркова M. Время его работы оценивается как $O(|M|^3 * |F|)$. Используя этот алгоритм, получаем следующие утверждения для предикатного расширения FPCTL логики PCTL.

Теорема 3. *(1) Существует алгоритм, который проверяет выполнимость FPCTL-формулы F на состоянии S базисной вероятностной МАС $\boldsymbol{A}$ за время, зависящее экспоненциально от $|\boldsymbol{A}|$ и линейно от $|F|$.*

(2) Существует алгоритм, проверяющий выполнимость FPCTL-формулы F на состоянии S произвольной (небазисной) вероятностной МАС $\boldsymbol{A}$ за время, зависящее дважды экспоненциально от $|\boldsymbol{A}|$ и линейно от $|F|$.

Следует заметить, что оценки для $|\mathbf{MC(A)}|$ выше были указаны для худшего случая. На практике во многих случаях эти оценки могут быть сильно понижены. Например, если арности всех используемых предикатов и действий ограничены, то число глобальных состояний для произвольной (не обязательно базисной) МАС ограничено некоторой экспонентой от полинома от размера этой МАС. Поэтому в этом случае слова «двойное экспоненциальное» в теореме 2 могут быть заменены на «экспоненциальное с полиномиальным показателем». Кроме того, при построении $\mathbf{MC(A)}$ многие глобальные состояния могут оказаться недостижимыми или недопустимыми, что также может привести к значительному снижению вычислительной сложности проблемы верификации.

6. Заключение

В этой статье мы показали, как вероятностные мультиагентные системы могут быть преобразованы в конечные Марковские цепи. Это позволило получить некоторые результаты о сложности верификации динамических свойств МАС с помощью применеия соответствующих результатов для конечных Марковских цепей, известных в литертуре. Заметим, что мы рассматривали здесь только МАС с детерминированным выбором действий. Конечно, представляется интересным также рассмотрение систем с недетерминированным выбором действий. По-видимому, к этому случаю можно будет применить результаты о верификации недетерминированных Марковских цепей (процессов принятия решений) из [18, 6, 13].

Список литературы

[1] Apt K. R., Logic programming. In: J. van Leeuwen (Ed.), Handbook of Theoretical Computer Science. Volume B. Formal Models and Semantics, Chapter 10, Elsevier Science Publishers B.V. 1990, 493-574.

[2] Barringer, H., Fisher, M., Gabbay, D., Gough, G., and Owens R. METATEM: An Introduction. Formal Aspects of Computing, 1995, 7:533-549.

[3] Bordini R., FisherM., Pardavila C., and Wooldridge M. Model checking AgentSpeak. AAMAS 2003, 409-416.

[4] Benerecetti, M., Guinchiglia, F., and Serafini, L. Model checking multiagent systems. Technical Report # 9708-07. Instituto Trentino di Cultura, 1998.

[5] Clarke E.M., Grumberg O. and Peled D., Model checking, MIT Press, 2000.

[6] Courcoubetis C., Yannakakis M., The complexity of probabilistic verification. J. ACM, v. 42, 4, 1995, 857-907.

[7] Dekhtyar M., Dikovsky A., and Valiev M., On feasible cases of checking multi-agent Systems Behavior. Theoretical Computer Science, Elsievier Science, 2003, vol. 303, no. 1, 63-81.

[8] Dekhtyar M.I., Dikovsky A.Ja., Valiev M.K., On complexity of verification of interacting agent's behavior. Annals of Pure and Applied Logic, 141, 2006, 336-362.

[9] Dekhtyar M.I., Dikovsky A.Ja., Valiev M.K. Temporal Verification of Probabilistic Multi-Agent Systems. Pillars of Computer Science: Essays Dedicated to Boris (Boaz)Trakhtenbrot on the Occasion of His 85th Birthday, LNCS, N 4800, 2008, 256-265.

[10] Dix J., Nanni M., and Subrahmanian V. S. , Probabilistic agent reasoning. ACM Transactions of Computational Logic, 1(2), 2000, 201-245.

[11] Giordano L., Martelli A., and Schwind C. Verifying communication agents by model checking in a temporal action Logic. JELIA 2004, LNAI 3229, 57-69.

[12] Hansson H., Jonsson B. A logic for reasning about time and reliability. Formal Aspects of Computing, 6(5), 1994, 512-535.

[13] Kwiatkowska Marta. Model Checking for probability and time: from theory to practice. In: Proc. 18th IEEE Symposium on Logic in Computer Science, 2003, 351-360.

[14] Rao A.S., Georgeff M.P., Modeling rational agents within a BDI architecture. Proc 2nd Intern. Conf. on Principles of Knowledge Representation and Reasoning, Morgan Kaufman Publishers, 1991.

[15] Shoham, Y., Agent oriented programming. Artificial Intelligence, 1993, 60:51-92.

[16] Subrahmanian V. S., Bonatti P., Dix J., et al., Heterogeneous agent systems, MIT LPess, 2000.

[17] van der Hoek W., Wooldridge M., Tractable multiagent planning for epistemic goals. AAMAS'02, Bologna, Italy, 2002.

[18] Vardi M.Y., Automatic verification of probabilistic concurrent finite state programs. In: Proceedings of 26th IEEE Symposium on Foundations of Computer Science. IEEE, New York, 327-338.

[19] Wooldridge, M., Dunne, P.E., The Computational complexity of Agent Verification. In: J.-J. Meyer and M. Tambe (eds.), Intelligent Agents VIII. Springer-Verlag Lecture Notes in AI Volume, March 2002.

[20] M. Wooldridge, M. Fisher, M.-P. Huget, and S. Parsons. Model Checking Multiagent systems with MABLE. In: Proc. of the First Intern. Conf. on Autonomous Agents and Multiagent Systems (AAMAS-02), Bologna, Italy, July 2002.

[21] Wooldridge, M., Jennings N. Intelligent agents: Theory and practice. The Knowledge Engineering Review, 1995, 10(2).

[22] Валиев М.К., Дехтярь М.И., Диковский А.Я, О свойствах многоагентных систем с вероятностными каналами связи. Труды IV-ой Межд. научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте», М.; Физматлит, Коломна, 2007.

[23] Тарасов В.Б. От многоагентных систем к интеллектуальным организациям. Эдиториал УРСС, М., 2002.