

12. Кучеров С.А., Рогозов Ю.И., Свиридов А.С. Метод построения структурно-независимых баз данных с использованием реляционных технологий // Новые технологии. Информационные технологии. – 2011. – № 2. – С. 52-59.
13. Fischer G. and E. Giaccardi. Meta Design: A framework for the future of end user development. End User Development: Empowering People to flexibly Employ Advanced Information and Communication Technology. H. Lieberman, F. Paterno and V. Wulf, Springer. 9: – P. 427-457.

Статью рекомендовал к опубликованию д.т.н., профессор В.Н. Иванченко.

Кучеров Сергей Александрович – Южный федеральный университет; e-mail: sergey.kutcherov@gmail.com; 347928, г. Таганрог, ГСП-17А, Некрасовский, 44; тел.: 89281922577; кафедра системного анализа и телекоммуникаций; ассистент.

Kucherov Sergey Alexandrovich – Southern Federal University; e-mail: sergey.kutcherov@gmail.com; 44, Nekrasovsky, Taganrog, GSP-17A, 347928, Russia; phone: +79281922577; the department of system analysis and telecommunications; assistant.

УДК 004.436.4; 510.643

А.И. Миков

НОРМАТИВНЫЕ МУЛЬТИАГЕНТНЫЕ СИСТЕМЫ*

Рассмотрено понятие нормативного агента, цели деятельности которого и ограничения в действиях формулируются на языке деонтической логики. Агент руководствуется внешними нормами (нормативными документами), описывающими окружение (предметную область) и возможности и запреты во взаимодействии агента с окружением. Нормативные мультиагентные системы представляют собой множество взаимодействующих нормативных агентов, решающих общую задачу в некоторой распределенной системе. Нормы могут образовывать иерархическую нормативную систему. Приведено разделение норм по видам, таким как регулирующие, декларативные и процедурные. Отношения между агентами в мультиагентной системе также представлены в виде норм. Обсуждается роль отношения "count as" в нормативных высказываниях. В работе описаны различные варианты деонтических операторов для нормативных агентов, варианты действий агентов, операторы статуса действий, деонтически замкнутые по действиям множества. Обсуждается сотрудничество агентов в мультиагентной системе, коалиции, проблемы формализации норм, заданных на естественном языке, ограничения на поведение агентов во времени с использованием модели ветвящегося времени. Приведены примеры преобразования нормативного текста в формулы прикладной модальной логики. Сравниваются теоретико-множественное определение нормативных систем и их задание формальными грамматиками.

Деонтическая логика; норма; программный агент; дескриптивный язык.

A.I. Mikhov

NORMATIVE MULTI-AGENT SYSTEMS

The concept of normative agent's goals and action constraints are formulated in the language of deontic logic. Agent is guided by external norms (regulations), describing the environment (subject area) and opportunities and prohibitions in conjunction with the agent environment. Normative multi-agent systems represent a set of interacting regulatory agents that solve a common problem in a distributed system. Norms may form a hierarchical regulatory system. Some

* Работа частично поддержана грантом РФФИ и Администрации Краснодарского края 13-07-96506.

kinds of norms are distinguished such as regulatory, declarative and procedural. Relationships between agents in multi-agent system are also presented in the form of regulations. The role of the "count as" relation in normative statements is discussed. This paper describes some different types of deontic operators for normative agents, variants of agent actions, statements of action status, the set which are deontic closed by actions. Also the paper discusses cooperation of agents, coalitions, the problem of formalizing the rules specified in natural language, restrictions on the behavior of agents in time using the model of branching time. The examples of normative texts transformation into formulas of applied modal logics are given. The set-theoretic definitions of normative systems and their description by formal grammars are compared.

Deontic logic; norm; program agent; descriptive language.

Введение. Современный компьютеринг все более смещается в сторону всеобъемлющего, *всепроницающего* (ubiquitous, pervasive) компьютеринга. В «вычислительном» процессе одновременно участвует несколько исполнителей и несколько взаимодействующих программ. По мнению Р. Милнера [1, 2], необходима новая математическая модель компьютеринга, соединяющая логику и традиционную компьютерную математику.

Можно представить себе, что в вычислительном процессе участвует несколько *агентов* (людей, программ, технических устройств). Они уже не могут действовать исключительно на основе императивной парадигмы. Они должны разрешать (**P**), запрещать (**F**) друг другу некоторые действия, а также обязывать (**O**). Таким образом, возникают условия для применения в модели компьютеринга деонтической логики. Разрешения и запреты могут быть персональными (одного агента другому), но это – ограниченный вариант. Практика показывает, что ориентироваться нужно на *неопределенный* круг агентов (лиц). Следовательно, описание вычислительного процесса должно включать в себя свод разрешающих, запрещающих и обязывающих правил – *нормативную систему* [3].

В [4] нормативная мультиагентная система определена как множество агентов, чье взаимодействие управляется нормами. Нормы предписывают, как агенты (в идеале) должны вести себя и как они не должны вести себя. Важно, что нормы предусматривают возможность того, что реальное поведение иногда может отклоняться от идеального, т.е. что нарушения обязательств агентов могут случаться.

Язык программирования для агентов также должен включать соответствующие конструкции для описания нормативной подсистемы вычислительного процесса. При этом нормативные высказывания могут касаться не только функционирования вычислительных устройств (как в обычном языке программирования), но всех агентов [5], участвующих в процессе. Поскольку процесс протекает во времени, высказывания могут быть функциями времени.

1. Деонтические операторы для агентов. Первоначально введенные в деонтическую логику операторы **P**, **F**, **O** не позволяют достаточно точно отражать содержание норм, в связи с чем отмечаются парадоксы первоначальных вариантов деонтической логики. Прежде всего это относится к вопросу об обязательствах. Например, фраза «обязательно записывать информацию в базу данных» не содержит указаний на то, *кто* это должен делать. Это – *безличная* обязанность. Без знания контекста невозможно понять, как она может быть выполнена. Фраза «администратор системы должен записывать информацию в базу данных» содержит указание на агента обязательства (администратор системы). Это – *агентская* обязанность. Агентская обязанность может быть непосредственной, как в данном примере, но могут быть и более сложные варианты агентской обязанности. Агент не всегда может сам выполнить некоторое действие. Но он может, в свою очередь, обязать другого агента (в чью компетенцию, в чьи умения входит выполнение действия) выполнить это действие. Например, организация берется за интеграцию нескольких информационных систем [6]. Директор (первый агент) заключает дого-

вор, принимая на себя ответственность. Директор поручает работу руководителю отдела (второй агент), который, в свою очередь, формирует группу и назначает ее руководителя (третий агент). Это типичная схема. При этом всю полноту ответственности несет первый агент. Третий агент может даже уволиться, а первый и второй агенты должны найти ему замену. Из этого примера видно, что в одном деонтическом выражении скрывается порой вся система взаимоотношений, бизнес-процессов, локальных норм внутри целой организации. Более того, агентская обязанность может быть выполнена не только по поручению агента, но и помимо его воли другим лицом, в чьи обязанности данное действие не входит. Например, это касается возникающих нештатных ситуаций.

Для предмета обязательства также существуют варианты. Прежде всего, предметом может быть выполнение некоторого действия. Это относительно простой вариант. Далее предметом может быть пребывание агента в определенном месте в определенное время (в заданной части распределенной системы). Предметом может быть постоянное совместное с другими агентами выполнение некоторых действий, сотрудничество. Совместное выполнение двумя агентами, связанными как личности, обязанностей, возникающих у одного из них (разделение обязанностей), тоже является вариантом предмета обязательств. К примеру, это может быть у соавторов произведения (они связаны пожизненно личными неимущественными правами авторства) и т.д.

Поэтому высказывания деонтической логики приобретают практический смысл только как часть описания предметной области информационной системы, совместно с другими частями описания (онтологиями, ролями, бизнес-процессами, целями). В связи с этим многие авторы дополняют стандартную деонтическую логику новыми операторами.

В работе [7] предложен оператор предопределенности (неизбежности): $\mathbf{PR}p$ = «Для любого агента предопределено, что имеет место утверждение p ». Иначе говоря, любой агент в своих действиях должен учитывать наличие в данный момент и в будущем факта p . Оператор \mathbf{PR} поддерживается следующими двумя аксиомами в дополнение к тавтологиям стандартной логики: $\mathbf{PR}(p \rightarrow q) \rightarrow (\mathbf{PR}p \rightarrow \mathbf{PR}q)$; $\mathbf{PR}p \rightarrow p$.

Авторы статьи [8] ввели для агентов операторы статуса действия: $\mathbf{W}\alpha$ и $\mathbf{Do}\alpha$. Оператор $\mathbf{W}\alpha$ читается как «отказ от обязательства выполнить действие α ». Оператор $\mathbf{Do}\alpha$ читается как «агент выполняет действие α ». Формула $\text{Op}(\alpha(t_1, \dots, t_n))$, где $\text{Op} \in \{\mathbf{P}, \mathbf{F}, \mathbf{O}, \mathbf{W}, \mathbf{Do}\}$, α – имя действия, а t_i – термы, т.е. объекты или переменные, или типы данных, называются атомом статуса действия. Из основных атомов строится множество S , называемое множеством статусов и удовлетворяющее аксиомам деонтической согласованности:

- 1) если $\mathbf{O}\alpha \in S$, то $\mathbf{W}\alpha \notin S$;
- 2) если $\mathbf{P}\alpha \in S$, то $\mathbf{F}\alpha \notin S$;
- 3) если $\mathbf{P}\alpha \in S$, то действие α выполнимо в данном состоянии агента.

Под состоянием агента понимается множество экземпляров объектов – структур данных в программном представлении объекта.

Множество S называется деонтически замкнутым по действиям, если для каждого основного действия α выполняются следующие ограничения:

- 1) $\mathbf{O}\alpha \in S \rightarrow \mathbf{P}\alpha \in S$;
- 2) $\mathbf{O}\alpha \in S \rightarrow \mathbf{Do}\alpha \in S$;
- 3) $\mathbf{Do}\alpha \in S \rightarrow \mathbf{P}\alpha \in S$.

Таким образом, некоторые модальности конфликтуют друг с другом. В частности, \mathbf{P} конфликтует с \mathbf{F} ; \mathbf{F} конфликтует с \mathbf{P} , \mathbf{O} , и \mathbf{Do} ; \mathbf{O} конфликтует с \mathbf{F} и \mathbf{W} ; \mathbf{W} конфликтует с \mathbf{O} ; \mathbf{Do} конфликтует с \mathbf{F} .

Если рассмотреть отрицания модальностей, то конфликты выглядят следующим образом. В дополнение к конфликтующей паре $(\mathbf{P}, \neg \mathbf{P})$ и подобным ей, имеются конфликтующие пары: $(\mathbf{O}, \neg \mathbf{P})$, $(\mathbf{O}, \neg \mathbf{Do})$, $(\mathbf{Do}, \neg \mathbf{P})$.

2. Нормативные логики. Проблема формализации норм является более неопределенной, чем, например, формализация физических законов. Предписанные нормы, в отличие от законов Природы, могут не выполняться, могут в ряде случаев противоречить друг другу. Работа информационно-телекоммуникационных систем подчиняется определенным инструкциям (локальным нормам), но иногда исполнители могут этими инструкциями пренебрегать. Автоматизированная система, в отличие от автоматической, требует своей логики описания предметной области. Автором предложен [9] подход, заключающийся в описании организационных предметных областей на основе построения систем уравнений в нормативной логике (реляционных уравнений), представляющих собой аксиоматику для мультиагентных систем.

Приведем пример преобразования фрагмента нормативного текста (Федерального Закона):

«конфиденциальность информации – обязательное для выполнения лицом, получившим доступ к определенной информации, требование не передавать такую информацию третьим лицам без согласия ее обладателя»

в такую деонтическую формулу:

$$((p \text{ has access}(I) \wedge p \neq \text{owner}(I) \wedge \neg \mathbf{P}^{\text{owner}(I)} \text{ distribute}(I)) \rightarrow \mathbf{O}_p \neg \text{distribute}(I).$$

Здесь I – информация, $\text{owner}(I)$ – собственник информации, p – некоторый агент, принадлежащий «неопределенному кругу лиц», $\text{access}(I)$ – доступ к информации, $\text{distribute}(I)$ – действие по распространению информации.

В [5] представлен теоретико-множественный подход к определению нормативной системы на основе конечного множества A агентов. С каждым агентом $i \in A$ связывается множество Act_i возможных действий. Причем множества действий разных агентов не пересекаются. Это упрощает модель, хотя, конечно, и очень ограничивает ее применимость.

Из агентов могут составляться коалиции. *Коалицией* называется подмножество множества агентов. В понятие коалиции не входит требование наличия какой-то общей цели у агентов. Множество Act_C действий коалиции C – это объединение множеств действий агентов, составляющих коалицию. Набор $\langle \alpha_1, \dots, \alpha_k \rangle$, в котором разные действия α_m совершаются разными агентами одной коалиции, называется совместным действием коалиции C и обозначается j_C . Множество всех совместных действий коалиции обозначается J_C . Действие из набора $j \in J_C$, совершаемое агентом с номером i , обозначается j_i .

Агенты существуют и действуют в некоторой базовой системе (например, информационной системе, или в организации), о которой известно, что она может пребывать в одном из конечного множества Q состояний. Среди этих состояний выделяется q_0 – начальное состояние.

Для того чтобы описать, как действуют агенты в базовой системе и к чему их действия могут привести, вводится понятие системы переходов – набора $Tr = \langle Q, q_0, A, Act_1, \dots, Act_n, \pi, \delta, \Xi, \xi \rangle$.

Обозначения π, δ, Ξ, ξ имеют следующий смысл.

$\pi : Act_A \rightarrow 2^Q$ – функция предусловий для действий. Для каждого действия каждого агента задается множество состояний системы, в которых это действие может быть произведено.

$\delta : Q \times J_A \rightarrow Q$ – функция переходов состояний. В результате совместного действия агентов система может перейти из одного состояния в другое. Поскольку не во всех состояниях базовой системы возможно выполнение всех действий, функция δ является частичной.

Ξ – конечное непустое множество атомарных утверждений.

$\xi : Q \rightarrow 2^\Xi$ – функция интерпретации. Задает для каждого состояния q множество $\xi(q)$ примитивных утверждений, справедливых в этом состоянии. Например, если пропозициональная переменная $p \in \xi(q)$, то ее значение истинно в этом состоянии.

На приведенные функции и множества налагаются два условия: 1) условие нетривиальности: в любом состоянии любой агент может выполнить хотя бы одно действие; 2) условие согласованности функций π и δ . Значение функции δ на наборе (q, j) определено тогда и только тогда, когда $q \in \pi(j_i)$ для любого агента i .

Функция π определяет для каждого агента i и состояния q множество опций $H(i, q)$, т.е. множество действий, которые агент может выполнить в этом состоянии. Опции задают ограничения *законности* для агентов.

Для каждого агента можно задать функцию $\sigma_i : Q \rightarrow Act_i$, такую, что $\sigma_i(q) \in H(i, q)$ для всех состояний q . Такую функцию естественно назвать *стратегией* агента. Стратегия коалиции – это набор $\sigma_C = \langle \sigma_1, \dots, \sigma_k \rangle$, в который включено по одной стратегии для каждого агента коалиции. Функция $jump(\sigma_C, q)$ описывает одношаговый переход, т.е. множество возможных состояний системы, в которые она может перейти из состояния q после выполнения одного действия каждым из агентов коалиции.

Нормативная система определяется как функция $\nu : Act_A \rightarrow 2^Q$. Интерпретируется она так: если $q \in \nu(\alpha)$, то нормативная система ν запрещает действие α в состоянии q . Функция ν двойственна функции π (π разрешает действие).

Говорят, что стратегия σ_i агента допускается нормативной системой ν , если для него никогда не выбирается действие, запрещенное этой нормативной системой. Это утверждение можно записать в виде некоторой функции $D(\sigma_i, \nu)$, принимающей значения истина или ложь. Аналогичная функция $D(\sigma_C, \nu)$ определена для стратегии коалиции σ_C .

Ясно, что возможно дальнейшее развитие нормативных систем в этом направлении. Недостатком приведенного определения нормативной системы является то, что запрет определяется для конкретных агентов, а не для типов агентов. Кроме того, запрет не структурирован на отдельные нормы, поэтому, вообще говоря, это не система. Сложно представить себе применимость описанного подхода к организационным системам, в которых множество состояний Q имеет очень большую мощность. В связи с этим возникают технические трудности в описании функции ν .

Поскольку агенты совершают некоторые действия, важным является не только изменение состояний, но и существование системы во времени. В работе [11] строится модель агентной системы на базе ветвящегося времени в рамках теории недетерминизма. *Структура ветвящегося времени* $F = \langle Tree, < \rangle$ представляет собой непустое множество *Tree* моментов времени и транзитивное неререфлексивное отношение порядка. Отношение $<$ удовлетворяет следующему условию: для любых моментов времени $m_1, m_2, m_3 \in Tree$, из $(m_1 < m_3) \wedge (m_2 < m_3)$ следует $(m_1 = m_2) \vee (m_1 < m_2) \vee (m_2 < m_1)$. Графическое изображение такой структуры напоминает дерево.

Необходимо учитывать, что, поскольку агентам предоставлено право выбора действий, то поведение мультиагентной системы не предопределено, могут быть его варианты во времени. Максимальные линейно упорядоченные множества $h \subseteq Tree$ в структуре F называются *историями*. Разумеется, историй в структуре F может быть много. Некоторый момент m может принадлежать истории h , т.е. $m \in h$. История h проходит через момент m . На графическом изображении дерева история представляет собой ветвь дерева $Tree$. Исходя из недетерминизма, считается, что некоторый момент может принадлежать нескольким историям. Множество историй, проходящих через момент m , обозначается $H_m = \{h : m \in h\}$. Поэтому для определенности обозначают m/h – момент m в истории h .

Модель ветвящегося времени M представляет собой пару $\langle F, v \rangle$, где $F = \langle Tree, < \rangle$, а v – функция оценки, отображающая каждое предложение, записанное на базовом языке, в множество пар m/h из $Tree$. Предполагается, что в эти моменты предложение рассматривается как истинное. Вообще же говоря, одно и то же предложение в разные моменты времени и/или в разных историях может иметь различные значения истинности, что отражается бинарным отношением, состоящим из пар $(m/h, A)$ и записываемое в виде $m/h \models A$, где A – формула. Срез этого отношения для некоторого момента m и формулы A определяет множество историй $|A|_m = \{h \in H_m : m/h \models A\}$.

Для описания действий агента в [11] введен новый двуместный деонтический оператор, записываемый в форме $\odot[\alpha \text{ cstit} : A]$, где α и A – операнды этого оператора. Интуитивный смысл этого оператора – агенту α следует наблюдать за тем, чтобы A (α ought to see to it that A). Здесь *stit* – первые буквы выражения «see to it that», *cstit* – один из вариантов *stit*. Более точно семантику оператора можно определить так. Имеется конечный выбор ситуаций, в которых гарантированно непустым является множество оптимальных действий, доступных агенту α в некоторый момент времени m . Выражение $\odot[\alpha \text{ cstit} : A]$ истинно только в случае, если истинность A гарантирована каждым оптимальным действием, доступным агенту в момент времени m , т.е. при $K \subseteq |A|_m$ для каждого $K \in \text{Optimal}_\alpha^m$. Здесь Множество Optimal_α^m состоит из оптимальных действий, доступных агенту в момент времени m .

3. Варианты описаний нормативных действий агентов. В теории информационных систем хорошо известно отношение, обозначаемое **is_a**. В нормативных системах возникает необходимость использовать, кроме этого, родственное отношение **count_as** (считаться).

Для иллюстрации можно вспомнить известный пример из романа Макса Фриша «Назову себя Гантенбайн». Герой романа объявил себя слепым. Выполнив все необходимые процедуры, он получил в мэрии удостоверение слепого и желтую нарукавную повязку, т.е. он стал *считаться* слепым. Но дело в том, что зрение у него было отменное. Это случай, когда имеет место отношение **count_as**, но не имеет места отношение **is_a**.

Из этого можно сделать вывод, что *установленный факт* – это факт, подтвержденный неким *уполномоченным* (наделенным властью) *агентом*.

В государственные и муниципальные информационные системы (ИС) сведения вносятся на основании документов, отражающих отношение **count_as**. В таких ИС *считается*, что **count_as = is_a**. Иначе говоря, используя инфиксную запись для отношений и рассматривая отношение как метапредмет предметной области, можно написать: **(count_as) count_as (is_a)** – отношение **count_as** считается отношением **is_a**. В социальных же сетях в Интернете (облаке) довольно часто можно встретиться с подменой имен, фотографий и даже пола.

Среди норм выделяют [12] регулирующие, декларативные и процедурные. Регулирующие нормы выражаются в терминах обязанности, запрета или разрешения и описываются деонтической логикой. Декларативные нормы дополняют регулирующие и основываются на отношениях **count_as** для того, чтобы представлять установленные факты как юридически имеющие место. Процедурные нормы задают взаимоотношение регулирующих и декларативных норм, смысл их в обязанности следить за состоянием дел, за выполнением установленных норм или, иначе, в отсутствии действий, *считающихся* нарушениями. Распознаванием нарушений, санкциями за нарушения, а также признанием фактов должны заниматься специальные агенты в нормативных системах.

Делегирование власти имеет две составляющие: полномочия в установлении факта – как действия некоторого агента устанавливает факт; делегирование цели – как этот агент может выполнить целеуказание делегирующего и как он при этом мотивирован. Цель делегирования определяется [13] так: агент *A* хочет, чтобы другой агент *B* выполнил некоторое действие и включает это в свой план; иными словами, *A* пытается достичь некоторых своих целей посредством определенного поведения (действий) агента *B*. Цель агента *A* состоит в том, чтобы *B* выполнил заданное действие или обладал заданным поведением.

Нормативный уровень описания МАС может быть расширен *контекстным нормативным* [14] уровнем. На этом уровне абстрактные нормы конкретизируются значениями из предметной области.

Например, абстрактная норма «Покупатель должен оплачивать товар в национальной валюте страны продавца» превращается в конкретную, определяемую окружением, норму «Покупатель должен оплачивать товар в России русскими рублями». Используя отношение «*A counts-as B in context C*» и перефразируя его как «*A is a sub-concept of B in context C*», можно сформулировать утверждение о классификации: «*Рубль is a sub-concept of 'законная валюта' in context Россия*».

Агентно-ориентированный подход становится все более распространенным в программировании и, в частности, в проектировании информационных систем. Одной из важнейших задач организационных информационных систем (ИС) является автоматизация бизнес-процессов. Бизнес-процессы определяются действиями, которые выполняют агенты, и ограничениями как на отдельные действия, так и на их последовательности, называемыми бизнес-правилами.

Описание и анализ нормативной МАС – достаточно сложная и громоздкая задача. Поэтому привлечение для этих целей средств из компьютерных наук вполне естественно. Одним из таких средств в последние годы стало создание языков описания нормативных мультиагентных систем и использование аппарата формальных грамматик для задания этих языков.

В работе [15] для описания норм при обмене информацией (сообщениями) предлагается формальная грамматика контекстно-свободного типа. В ней норма задается как последовательность нормативного условия, т.е. контекста, в котором норма применяется, условия нарушения нормы, механизма раскрытия нарушения, санкций и восстановления.

Здесь мы предложим более общую форму основных правил грамматики дескриптивного языка формальных норм:

1) ⟨норма⟩ ::= ⟨деонтический оператор⟩ (**do** ⟨формула высказывания⟩ [⟨период времени⟩] [**if** ⟨условное выражение⟩])

2) ⟨деонтический оператор⟩ ::= **obliged** | **permitted** | **forbidden**

3) ⟨период времени⟩ ::= **before** ⟨момент времени⟩ | **after** ⟨момент времени⟩ | **between** (⟨момент времени⟩, ⟨момент времени⟩) | **before done** ⟨формула высказывания⟩ | **after done** ⟨формула высказывания⟩ | **between (done** ⟨формула высказывания⟩), **done** ⟨формула высказывания⟩)

- 4) $\langle \text{условное выражение} \rangle ::= \neg(\langle \text{список условий} \rangle) \mid \langle \text{список условий} \rangle$
- 5) $\langle \text{список условий} \rangle ::= [\neg] \langle \text{условие} \rangle [\langle \text{условное выражение} \rangle]$
- 6) $\langle \text{условие} \rangle ::= \langle \text{величина} \rangle \langle \text{знак сравнения} \rangle \langle \text{величина} \rangle \mid \mathbf{done} \langle \text{формула высказывания} \rangle \mid \langle \text{деонтический оператор} \rangle (\mathbf{do} \langle \text{формула высказывания} \rangle)$
- 7) $\langle \text{величина} \rangle ::= \langle \text{атрибут} \rangle \mid \langle \text{обращение к функции} \rangle \mid \langle \text{значение} \rangle$
- 8) $\langle \text{атрибут} \rangle ::= \langle \text{имя агента} \rangle . \langle \text{имя атрибута агента} \rangle \mid \langle \text{переменная} \rangle$
- 9) $\langle \text{санкции} \rangle ::= \mathbf{sanction} (\langle \langle \text{список действий} \rangle \rangle) \mathbf{if} \langle \text{вид отношения} \rangle (\langle \langle \text{норма} \rangle \rangle)$
- 10) $\langle \text{вид отношения} \rangle ::= \mathbf{violated} \mid \mathbf{complied}$
- 11) $\langle \text{действие} \rangle ::= \langle \text{атрибут} \rangle = \langle \text{обращение к функции} \rangle \mid \langle \text{обращение к функции} \rangle$.

Семантика описанного языка предполагает взаимодействие программных агентов в рамках некоторой нормативной мультиагентной системы. Предикат **do**(формула высказывания) представляет еще не выполненное действие, выраженное формулой высказывания, например, «совершить акт передачи сообщения». Это единственный предикат, перед которым могут стоять деонтические операторы. Предикат **done**(формула высказывания) используется для обозначения того, что соответствующее действие, заданное формулой высказывания, выполнено. Этот предикат может использоваться в условных конструкциях нормативных правил.

Из приведенного выше фрагмента грамматики видно, что многие правила являются рекурсивными, что позволяет строить очень сложные нормативные предложения, т.е. язык является весьма мощным.

Заключение. Нормативные мультиагентные системы – активно развивающаяся область исследований, зародившаяся в середине 90-х гг. и привлекающая все большее количество ученых. Эта область находится на пересечении математической логики, компьютерных наук, искусственного интеллекта, теории права и обещает в будущем как получение интересных теоретических результатов, так и серьезные приложения.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Milner R.* Elements of interaction. Turing Award lecture // *Communication of the ACM.* – 1993. – Vol. 36, № 1. – P. 78-89.
2. *Milner R., Parrow J., Walker D.* Modal logics for mobile processes // *Proceedings of 2nd International Conference on Concurrency Theory (Amsterdam, The Netherlands, August 26–29, 1991 (CONCUR '91)).* – Lecture Notes in Computer Science. – 1991. – Vol. 527. – P. 45-60.
3. *Миков А.И.* Информационные процессы и нормативные системы в ИТ: Математические модели. Проблемы проектирования. Новые подходы. – М.: Книжный дом «ЛИБРОКОМ», 2013. – 256 с.
4. *Jones A., Carmo J.* Deontic logic and contrary-to duties // *Handbook of Philosophical Logic (D. Gabbay, ed.).* – Kluwer, 2001. – P. 203-279.
5. *Миков А.И., Воробьев В.В.* Моделирование агентов с деонтической логикой, функционирующих в распределенных системах // *Информатизация и связь.* – 2012. – № 5. – С. 82-85.
6. *Миков А.И.* Модели сложности разделения и интеграции информационных систем // *Информатизация и связь.* – 2013. – № 2. – С. 92–96.
7. *McNamara P.* Agential obligation as non-agential personal obligation plus agency // *Journal of Applied Logic.* – 2004. – Vol. 2. – P. 117-152.
8. *Eiter T., Subrahmanian V.S., Rogers T.J.* Heterogeneous active agents, III: Polynomially implementable agents // *Artificial Intelligence.* – 2000. Vol. 117. – P. 107-167.
9. *Миков А.И.* Аксиоматика отношений в предметных областях и реляционные отношения // *Информатизация и связь.* – 2013. – № 5. – С. 70-74.

10. Wooldridge M., van der Hoek W. On obligations and normative ability: Towards a logical analysis of the social contract // Journal of Applied Logic. – 2005. – Vol. 3. – P. 396-420.
11. Horty J.F. Agency and Deontic logic. – Oxford: Oxford University Press, 2001.
12. Boella G., van der Torre L. Substantive and Procedural Norms in Normative Multiagent Systems // Journal of Applied Logic. – 2008. Vol. 6. – P. 152-171.
13. Castelfranchi C. Modeling social action for AI agents // Artificial Intelligence. – 1998. – Vol. 103, № 1-2. – P. 157-182.
14. Felicissimo C., de Lucena C.J. P., Briot J.-P. A Norm-Based Approach for the Modeling of Open Multiagent Systems // Int. Conf. on Agents and Artificial Intelligence (Porto, Portugal, ICAART'09). – P. 540-546.
15. Garcia-Camino A., Noriega P., Rodriguez-Aguilar J.A. Implementing norms in electronic institutions // Proceedings of the 4th international joint conference on Autonomous agents and multiagent systems (The Netherlands). – 2005. – P. 667-673.

Статью рекомендовал к опубликованию д.т.н., профессор В.Н. Марков.

Миков Александр Иванович – Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Кубанский государственный университет»; e-mail: alexander_mikov@mail.ru; 340050, г. Краснодар, ул. Ставропольская, 149; тел.: 89183456364; кафедра вычислительных технологий; д.ф.-м.н.; профессор; зав. кафедрой.

Mikov Alexander Ivanovich – Federal State Educational Establishment of Higher Professional Education “Kuban State University”; e-mail: alexander_mikov@mail.ru; 149, Stavropolskaya street, Krasnodar, 340050, Russia; phone: +79183456364; the department of computer technologies; dr. of phis.-math. sc.; professor; head of department.

УДК 004.652.4(045)

Ю.В. Полищук, Т.А. Черных

ФОРМИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ БОЛЬШИХ И СЛОЖНЫХ ТЕХНИЧЕСКИХ ОБЪЕКТОВ НА ОСНОВЕ КВАЗИСТРУКТУРИРОВАННЫХ МОДЕЛЕЙ ИНФОРМАЦИОННОГО НАПОЛНЕНИЯ

Рассмотрены основные задачи формирования информационных систем больших и сложных технических объектов на основе квазиструктурированных моделей информационного наполнения и предложены пути их решения. Применение квазиструктурированных моделей информационного наполнения реализует единый интерфейс доступа к фактографическим данным сопутствующего эксплуатационного контента. Обработка последнего необходима при принятии управленческих решений для больших и сложных технических объектов. В контексте данной работы рассмотрена математическая модель информационного наполнения документа и алгоритм синтеза моделей данного вида. Алгоритм синтеза моделей базируется на использовании пяти параметров и построенной на их основе целевой функции, применение которой позволяет проводить анализ квазиструктурированных моделей с позиции эффективности описания информационного наполнения документа. Основным преимуществом использования информационных систем больших и сложных технических объектов на основе квазиструктурированных моделей информационного наполнения является возможность обработки фактографического контента без извлечения из документа-первоисточника, что способствует снижению количества ошибок при обработке данных.

Моделирование документов; обработка электронных документов; квазиструктурированная информация; автоматизированные информационные системы.