

ПРЕДСТАВЛЕНИЕ ЗНАНИЙ ИНТЕЛЛЕКТУАЛЬНОГО АГЕНТА МНОГОАГЕНТНОЙ СИСТЕМЫ

Введение. Интеллектуальная система (ИС) – это техническая или программная система, способная решать задачи, традиционно считающиеся творческими, принадлежащие конкретной предметной области, знания о которой хранятся в памяти такой системы. Структура интеллектуальной системы включает три основных блока – базу знаний, решатель и интеллектуальный интерфейс[1].

База знаний (БЗ) – важный компонент интеллектуальной системы. Это особого рода база данных, разработанная для оперирования знаниями (метаданными). База знаний содержит структурированную информацию, покрывающую некоторую область знаний, для использования с конкретной целью. Современные базы знаний работают совместно с системами поиска информации, имеют классификационную структуру и формат представления знаний[2].

Полноценные базы знаний содержат в себе не только фактическую информацию, но и правила вывода, допускающие автоматические умозаключения о вновь вводимых фактах и, как следствие, осмысленную обработку информации.

Иерархический способ представления в базе знаний набора понятий и их отношений называется онтологией. Онтологию некоторой области знаний вместе со сведениями о свойствах конкретных объектов также можно назвать базой знаний.

Постановка задачи. Цель статьи: анализ способов представления знаний и формирования базы знаний интеллектуального агента (ИА) многоагентной системы (МАС).

МАС определяется шестеркой[3]:

$$MAS = (X, E, R, AC, P, ST, EV), \quad (1)$$

где $X = A = \{1, \dots, n\}$ – множество неоднородных агентов; E – множество сред, в которых может функционировать данная МАС; R – семейство базовых отношений между агентами, причем это семейство отношений включает по крайней мере три типа отношений и может быть представлено разбиением

$$R = R_1 \cup R_2 \cup R_3, \quad (2)$$

R_1 – множество горизонтальных (симметричных) отношений, R_2 – множество асимметричных отношений, направленных «сверху вниз», R_3 – множество нечетких асимметричных отношений, направленных «снизу вверх»; AC – множество действий агентов; P – множество коммуникативных актов, образующих протокол коммуникации в МАС; ST – множество состояний МАС; EV – множество стратегий агентов.

В свою очередь, социальный (коллективный) агент из МАС определяется пятеркой[3]

$$SA = (ST, L, AC, SL, T), \quad (3)$$

где ST – множество состояний агента; L – множество языков общения между агентами; AC – множество действий; SL – множество социальных законов, т.е. правил поведения, причем ограничение задается парой (ac, φ) , а социальный закон sl есть множество ограничений (ac, φ) , $\forall ac_i \in AC, \varphi \in L, st | = \varphi$; T – обобщенная функция переходов, $T: ST \times AC \times SL \rightarrow 2^{ST}$, удовлетворяющая следующим условиям: а) для любых $st \in ST, ac \in AC, sl \in SL$, если состояние st удовлетворяет ограничению $\varphi, st | = \varphi$, и пара $(ac, \varphi) \in sl$, то $T(st, ac, sl) = \varphi$; б) для любых $st \in ST, ac \in AC, sl_1 \in SL, sl_2 \in SL$, если $sl_1 > sl_2$, то $T(st, ac, sl_1) \subseteq T(st, ac, sl_2)$.

Представление знаний агента. В простейших замкнутых средах[7] применяются агенты автоматного уровня сложности (модели коллективного поведения автоматов), а в случае более сложных замкнутых сред уже используют агенты, основанные на конечных наборах правил и сценариев их применения (например, нечеткие регуляторы). Агенты, основанные на правилах, активно используются в компьютерных сетях, действуя в рамках «клиент-серверного» подхода. В целом, когда среда замкнута, пара «среда-агент» может быть в принципе задано формальной системой, т.е. действия агента здесь могут быть описаны в рамках подходящего логического исчисления (например, модальные логики или логики предикатов высокого порядка). Для открытых сред требуется переход к семиотическому моделированию.

К настоящему времени сформировался довольно большой список свойств, которыми должны обладать ИА [4,5]:

– автономность (autonomy, autonomous functioning) – способность к самостоятельному формированию

целей и функционированию с самоконтролем своих действий и внутреннего состояния;

– общественное поведение (social ability, social behavior) – способность согласовать свое поведение с поведением других агентов в условиях определенной среды и правил поведения путем обмена сообщениями на языке коммуникации;

– реактивность (reactivity) – способность адаптировано воспринимать состояние внешней среды (среды функционирования и множества других агентов) и своевременно реагировать на происходящие изменения;

– активность (pro-activity) – способность проявлять инициативу, т. е. самостоятельно генерировать цели и действовать рационально для их достижения, а не только пассивно реагировать на внешние события;

– базовые знания (basic knowledge) – постоянная часть знаний агента о себе, о среде, а также постоянные знания о других агентах, которые не изменяются в рамках жизненного цикла агента;

– убеждения (beliefs) – переменная часть знаний агента о среде и других агентах, которая может изменяться во времени, но агент может об этом не знать и продолжать использовать их для своих целей;

– желания (desires) – состояния и/или ситуации, достижение которых является желательным и важным для агента, однако которые могут быть противоречивыми и не все будут достигнуты;

– цели (goals) – совокупность состояний, на достижение которых направлено текущее поведение агента;

– намерения (intentions) – это то, что агент обязан сделать в силу своих обязательств по отношению к другим агентам, или то, что вытекает из его желаний (т. е. непротиворечивое подмножество желаний, выбранное по тем или иным причинам и совместимое с принятыми на себя обязательствами);

– обязательства (commitments) – задачи, которые берет на себя агент по просьбе и/или поручению других агентов.

Для автономного агента (АА) постулируются такие свойства как адаптивность (способность действовать соответственно и согласованно с изменениями в среде), робастность (малые изменения в свойствах среды не должны приводить к общему коллапсу поведения АА), наличие тактики (способность устанавливать множественные цели и, вне зависимости от найденных им обстоятельств, решать, какие частные цели активно преследовать), гибкость (способность выполнять широкий круг задач) [4, 5].

К этому набору свойств могут добавляться и следующие:

– благожелательность (benevolence) – готовность агентов помогать друг другу и решать именно те задачи, которые им поручат владелец или пользователь;

– правдивость (veracity) – свойство агента не оперировать заведомо ложной информацией;

– рациональность (rationality) – способность агента действовать так, чтобы достигать своих целей, а не избегать их достижения, по крайней мере, в рамках своих знаний и убеждений.

Все основные действия агентов можно при этом разделить на три основные категории:

– действия по заранее заложенному сценарию для достижения поставленной цели;

– реакция на изменение внешней среды;

– реакция на действия других агентов, через протокол общения между агентами.

В контексте использования нотации UML 2.0 для отображения основных действий реализуемых ИА агента используем диаграмму вариантов использования:

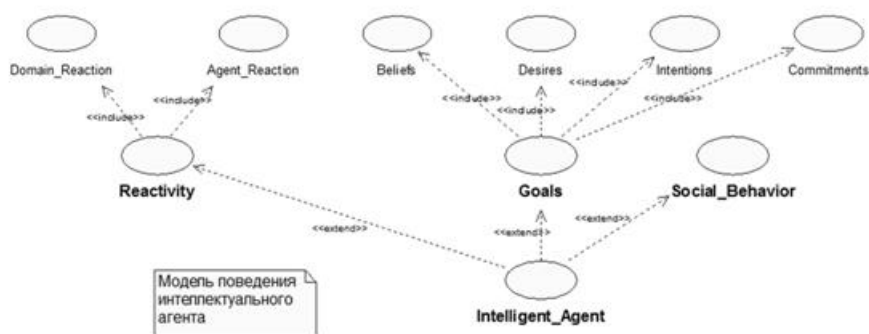


Рис.1. Диаграмма модели поведения агента

Для выполнения поставленных перед ИА целей, ему необходимы следующие знания:

– основные знания агента о предметной области,

– знания о других агентах среды,

– основные сценарии поведения,

- правила логического вывода новых знаний, сценариев поведения, т.е. (самообучение),
- правила рассуждений,
- протокол изменения параметров предметной области.

В контексте использования нотации UML 2.0 для проектирования ИА знания агента можно представить в виде диаграммы вариантов использования:



Рис.2. Диаграмма вариантов использования знаний агента

Существует множество моделей представления знаний для различных предметных областей[2]. Большинство из них может быть сведено к следующим классам:

- продукционные модели;
- семантические сети;
- фреймы;
- формальные логические модели.

У каждой из моделей есть свои слабые и сильные стороны, к примеру, сильные стороны продукционной модели: простота создания и понимания отдельных правил, простота пополнения и модификации, простота механизма логического вывода; слабые: сложность и громоздкость анализа целостного образа знаний, структура базы знаний системы не позволяет описывать метазнания и свойственную человеческому мышлению нечеткую логику.

С целью использования преимуществ продукционной модели представления знаний и компенсации ее недостатков, предлагается использование гибридной модели с нечетким логическим выводом.

Под нечетким логическим выводом называется получение заключения в виде нечеткого множества, соответствующего текущим значениям входов, с использованием нечеткой базы знаний и нечетких операций[8]. Основу нечеткого логического вывода составляет композиционное правило Заде: если известно нечеткое отношение \bar{R} между входной (x) и выходной (y) переменными, то при нечетком значении входной переменной $x = \tilde{A}$, нечеткое значения выходной переменной определяется:

$$y = \tilde{A} \circ \bar{R}, \tag{4}$$

где \circ – максимая композиция.

Для осуществления нечеткого логического вывода ИА по нечеткой базе знаний будем использовать алгоритм Мамдани:

$$\bigcup_{p=1}^{k_j} \left(\bigcap_{i=1}^n x_i = a_{i,jp} \text{ с весом } w_{jp} \right) \rightarrow y = d_j, j = \overline{1, m}, \tag{5}$$

где \cup – нечеткая логическая операция ИЛИ; \cap – нечеткая логическая операция И; $w_{jp} \in [0,1]$ – весовой коэффициент правила с номером jp ; $a_{j,jp}$ – нечеткий терм, которым оценивается переменная x_i в строке $jp (p = \overline{1, k_j})$;

При этом для задания многомерных зависимостей «входы-выходы» будем использовать нечеткие логические операции И и ИЛИ. Для удобства правила будем формулировать таким образом, чтобы внутри

каждого из правил переменные объединялись логической операцией И, а правила в базе знаний связывались операцией ИЛИ. Тогда нечеткую базу знаний, связывающую входы $X = (x_1, x_2, \dots, x_n)$ с выходом y , будем представлять в следующем виде:

$$\begin{aligned} & \text{ЕСЛИ } (x_1 = a_{1,j1}) \text{ И } (x_2 = a_{2,j1}) \dots \text{ И } (x_n = a_{n,j1}) \\ & \text{ИЛИ } (x_1 = a_{1,j2}) \text{ И } (x_2 = a_{2,j2}) \dots \text{ И } (x_n = a_{n,j2}) \\ & \dots \\ & \text{ИЛИ } (x_1 = a_{1,jk_j}) \text{ И } (x_2 = a_{2,jk_j}) \dots \text{ И } (x_n = a_{n,jk_j}), \\ & \text{ТО } y = d_j, \quad j = 1, m. \end{aligned} \tag{6}$$

Значения входных и выходной переменной в нечеткой базе знаний будем задавать нечеткими множествами. При этом:

$$\begin{aligned} \mu_{jp}(x_i) & - \text{ функция принадлежности входа } x_i \text{ нечеткому терму } a_{i,jp}, \text{ т.е. } a_{i,jp} = \int_{x_i}^{\bar{x}_i} \mu_{jp}(x_i) / x_i, \quad x_i \in [x_i, \bar{x}_i]. \\ \mu_{dj}(y) & - \text{ функция принадлежности выхода } y \text{ нечеткому терму } d_j, \text{ т.е. } d_j = \int_{y}^{\bar{y}} \mu_{dj}(y) / y, \quad y \in [y, \bar{y}]. \end{aligned}$$

Степени принадлежности входного вектора $X^* = (x_1^*, x_2^*, \dots, x_n^*)$ нечетким термам d_j из базы знаний ИА будут рассчитываться следующим образом:

$$\mu_{dj}(X^*) = \bigvee_{p=1,k_j} w_{jp} \cdot \bigwedge_{i=1,n} [\mu_{jp}(x_i^*)], \quad j = \overline{1,m}, \tag{7}$$

где $\bigvee(\bigwedge)$ – операция из s-нормы (t-нормы), т.е. из множества реализаций логической операций ИЛИ (И). Наиболее часто используются следующие реализации: для операции ИЛИ – нахождение максимума и для операции И – нахождение минимума.

В результате получаем такое нечеткое множество \tilde{y} , соответствующее входному вектору X^* :

$$\tilde{y} = \frac{\mu_{d1}(X^*)}{d_1} + \frac{\mu_{d2}(X^*)}{d_2} + \dots + \frac{\mu_{dm}(X^*)}{d_m}. \tag{8}$$

Особенностью этого нечеткого множества является то, что универсальным множеством для него является терм-множество выходной переменной y .

Для перехода от нечеткого множества, заданного на универсальном множестве нечетких термов $\{d_1, d_2, \dots, d_m\}$ к нечеткому множеству на интервале $[y, \bar{y}]$ совершаем: 1) «срезаем» функции принадлежности $\mu_{dj}(y)$ на уровне $\mu_{dj}(X^*)$; 2) объединяем (агрегируем) полученные нечеткие множества:

$$\tilde{y} = \text{agg}_{j=1,m} \left(\int_y^{\bar{y}} \min(\mu_{dj}(X^*), \mu_{dj}(y)) / y \right), \tag{9}$$

где agg - агрегирование нечетких множеств, которое наиболее часто реализуется операцией нахождения максимума.

Четкое значение выхода y , соответствующее входному вектору X^* определяется в результате дефаззификации нечеткого множества \tilde{y} . Выполним это по методу центра тяжести:

$$y = \frac{\int_y^{\bar{y}} y \cdot \mu_{\tilde{y}}(y) dy}{\int_y^{\bar{y}} \mu_{\tilde{y}}(y) dy}, \tag{10}$$

где \int - здесь символ интеграла.

Учитывая возможности программной реализации, рассуждения ИА можно классифицировать следующим образом:

– немонотонные рассуждения, т.е. рассуждения, полученные в результате логического вывода при некотором недостатке информации при помощи неполной системы правил (аксиом) или на основании

гипотез о достоверности некоторых фактов, которые впоследствии могут быть отвергнуты;

- вероятностные или правдоподобные рассуждения;
- рассуждения по аналогии, когда можно предположить, что истинность одних фактов подтверждается истинностью аналогичных;
- достоверные рассуждения, основанные на дедукции и правилах логического вывода.

Рассмотрим пример реакции координационного ИА на изменения окружающей среды. Состояния агента и среды заданы следующими терм-множествами:

состояния среды:

уровень модема: $T_{\text{MODEM}} = \{\text{«очень низкий»}, \text{«низкий»}, \text{«средний»}, \text{«высокий»}, \text{«очень высокий»}\}$,

качество соединения: $T_{\text{CONNECT}} = \{\text{«очень низкое»}, \text{«низкое»}, \text{«среднее»}, \text{«высокое»}, \text{«очень высокое»}\}$,

вероятность отклика сервера – значение неизвестно,

количество неоткликов: $T_{\text{NO RESPONSE}} = \{1, 2, 3, 4\}$;

поведение ИА:

цель – соединение с сервером $T_{\text{SERVER CONNECTION}} = \{\text{«да»}, \text{«нет особой необходимости»}, \text{«нет»}\}$,

время повтора (секунд) $T_{\text{REPLAY}} = \{15, 300, 6000\}$.

Тогда, с использованием нечеткого логического вывода Мамдани, при следующих значениях входных переменных:

состояния среды:

уровень модема: $x_{\text{MODEM}} = \text{«высокий»}$,

качество соединения: $x_{\text{CONNECT}} = \text{«очень высокое»}$,

вероятность отклика сервера – значение неизвестно,

количество неоткликов: $x_{\text{NO RESPONSE}} = 4$;

поведение ИА:

цель – соединение с сервером $x_{\text{SERVER CONNECTION}} = \text{«да»}$,

будет получено – значение время повтора $u_{\text{REPLAY}} = 6000$, что позволит не совершать непрерывные попытки соединения с сервером и не тратить время на ожидание ответа, что позволит сэкономить вычислительные ресурсы агента.

Выводы. Применение нечеткого логического вывода в рассуждения ИА позволяет избежать громоздкости правил, по сравнению с продукционным подходом, увеличить скорость вычисления результата и расширить его спектр за счет увеличения мощности терм-множеств, что приближает рассуждения ИА к человеческим. Использование метода Мамдани объясняется простотой программной реализации и естественностью получаемых результатов.

ЛИТЕРАТУРА:

1. Аверкин А.Н., Гаазе-Рапопорт М.Г., Поспелов Д.А. Толковый словарь по искусственному интеллекту. М.: Радио и связь, 1992.
2. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем – СПб.: Питер, 2001. – 384 с.
3. Тарасов В.Б. От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. — М.: Эдиториал УРСС, 2002. — С. 352.
4. Люгер Д. Ф. Искусственный интеллект: стратегии и методы решения сложных проблем: Пер. с англ. – М.: Издательский дом «Вильямс», 2003. – 864 с.
5. Рассел С., Норвиг П. Искусственный интеллект: современный подход.: Пер. с англ. – М.: Издательский дом «Вильямс», 2006. – 1408 с.
6. Швецов А.Н. Агентно-ориентированные системы: от формальных моделей к промышленным приложениям / Всероссийский конкурсный отбор обзорно-аналитических статей по приоритетному направлению «Информационно-телекоммуникационные системы», 2008. – 101 с.
7. Поспелов Д.А. От коллектива автоматов к мультиагентным системам // Труды Международного семинара «Распределенный искусственный интеллект и многоагентные системы» (DAIMAS'97, Санкт-Петербург, Россия, 15-18 июня 1997). – С.319-325.
8. Штовба С.Д. Проектирование нечетких систем средствами MATLAB. М.: Горячая линия – Телеком, 2007. – 288 с.

БОСКИН Олег Осипович – ст. преподаватель кафедры информационных технологий Херсонского национального технического университета.

Научные интересы: прогрессивные информационные технологии.