

МОДЕЛЬ ПОВЕДЕНИЯ АВТОНОМНОГО АГЕНТА НА ОСНОВЕ ТЕОРИИ АВТОМАТОВ

Запропоновано концептуальний підхід побудови системи автономного управління інформаційними системами та розподіленими базами даних із застосуванням технології програмних агентів. Обґрунтовано вибір модифікованої структури фрейма як логічної моделі програмного агента. Надана модель поведінки агента у мультиагентному просторі на підставі теорії автоматів, що дозволяє ефективно реалізувати управління та адміністрування розподілених інформаційних ресурсів обчислювальних систем.

Введение

Идея управления информационными системами и ресурсами при помощи автономных программ-диспетчеров по сценарию не нова. Она начала воплощаться в жизнь еще в 60-х, обеспечивая возможность передачи пакетов заданий с мини-компьютеров на большую машину и получения результатов вычислений. Позднее, в 70-х, программы-диспетчеры в реальном времени поддерживали распределенные вычисления в сетях мини-компьютеров. Сейчас эта идея лежит в основе технологии программных агентов с мобильными сценариями поведения, реализованных на одном из языков программирования.

Существует большое количество определений понятия "агент" в зависимости от взгляда на распределенную обработку знаний. С точки зрения распределенных вычислений, агент – это самостоятельный процесс, выполняемый параллельно, имеющий определенное состояние и способный взаимодействовать с другими агентами посредством передачи сообщений. В этом смысле его можно рассматривать как естественное развитие парадигмы объектно-ориентированного параллельного программирования. По существу программные агенты выполняют задачи, делегируемые им пользователями. В зависимости от конкретной задачи агенты могут размещаться на персональных компьютерах разного назначения вплоть до Web-серверов. С точки зрения объектно-ориентированного программирования программного агента можно рассматривать как технологию обмена распределенными знаниями и функциями. Каждый агент – это процесс, обладающий определенной частью знаний об объекте с возможностью обмениваться этими знаниями с другими агентами [1].

Существующие проблемы и перспективы в развитии агентных технологий

В современном информационном пространстве уже существуют интеллектуальные агенты, но они, как правило, функционируют изолированно и выполняют единичные, сравнительно простые задачи, например, фильтрацию электронной почты и поиск информации в соответствии с заданными параметрами.

В настоящее время рассматривают четыре типа архитектур интеллектуальных агентов:

- простой рефлексивный агент (Simple reflex agent);
- агент, помнящий о состояниях мира (Agent that keep track of the world);
- целевой агент (Goal-based agent);
- бережливый агент (Utility-based agent).

Сложные системы на базе агентов находят все более широкое применение в информационных технологиях. Агенты получают возможность находить друг друга в кибернетическом пространстве, если будут созданы технологии взаимодействия программных агентов, которые функционируют как координаторы или центры обмена информацией, накапливающие данные о других агентах и выполняемых ими функциях.

Учитывая динамику развития информационных технологий, агентные системы в ближайшем будущем найдут свое применение в таких областях как:

- автоматизированного проектирования,
- управления движением на транспорте,
- системы интеграции гетерогенной информации,
- управления масштабными военными действиями,
- защита данных и администрирования информационных ресурсов,
- извлечение "знаний" из данных (data mining).

Существуют ряд препятствий на пути широкомасштабного развертывания в информационном пространстве мультиагентных систем. Среди первоочередных проблем следует выделить задачу согласования стандартов, обеспечивающих совместную работу агентов, созданных различными разработчиками [2].

Цель исследования

Агенты распределенной мультиагентной системы могут решать сложные задачи информационной поддержки. В качестве внешних управляющих воздействий могут выступать сообщения и команды от менеджера мультиагентного пространства, а также сообщения о состоянии информационной среды. При помощи специальных функций и процедур агент может взаимодействовать с различными объектами вычислительной среды: от каталогов и файлов до структур распределенных баз данных.

Концептуальная часть задачи проектирования и создания программных агентов состоит в разработке механизмов их поведения в информационном пространстве и взаимодействия между собой.

Целью проводимых исследований является разработка и исследование эффективных моделей поведения программных агентов в информационном пространстве.

Разработка модели взаимодействия программного агента с внешней средой

В соответствии с основными требованиями, предъявляемыми к свойствам программных агентов – автономностью функционирования и способностью выполнять целесообразные действия, рассмотрим один подход к построению модели поведения программного агента. В качестве базовой структуры для построения модели поведения программных агентов в информационной среде может быть рассмотрена модель на основе теории автомата [3].

$$\varphi(t+1) = \Phi[\varphi(t), Q(t+1)] \quad (1)$$

$$f(t) = F[\varphi(t)] \quad (2)$$

Уравнение (1) описывает зависимость изменения состояний автомата под воздействием входной переменной $Q(t)$, а уравнение (2) – действия автомата $f(t)$ от его состояний. Входная переменная имеет только два значения и задает пару отображений в себя множества состояний автомата. Одно из этих отображений задано для $Q=0$, другое – для $Q=1$. Эти отображения записываются в виде матрицы состояний $\|\alpha_{ij}(Q)\|$, $(i, j=1, 2, \dots, n)$. Для определения задачи о поведении автомата необходимо дать характеристику среды, с которой взаимодействует автомат.

Автомат U находится в случайной стационарной среде $C = C(a_1, a_2, \dots, a_x)$, если действия автомата и значения его входной переменной связаны следующим образом:

действие $f_\alpha, \alpha = 1, 2, \dots, x$, произведенное автоматом в момент времени t , влечет за собой в момент времени $t+1$ – проигрыш $Q=0$ с вероятностью $p_\alpha = (1 - a_\alpha)/2$ или выигрыш $Q=1$ с вероятностью $p_\alpha = (1 + a_\alpha)/2$.

Обозначим через $\sigma_\alpha, \alpha = 1, 2, \dots, x$ сумму финальных вероятностей состояний Φ_i , которым соответствует действие f_α . Тогда величины σ_α будут иметь смысл вероятности действия f_α автомата U в среде C . Математическое ожидание выигрыша для автомата U в среде C выражается соотношением:

$$W(U, C) = \sum_{\alpha=1}^x \sigma_\alpha a_\alpha \quad (3)$$

Автомат с достаточно большим набором состояний (емкостью памяти) обладает целесообразным поведением в среде C [4].

Представление логической модели программного агента в виде фрейма позволяет эффективно применить теорию автоматов для построения и исследования модели поведения программного агента.

В общем виде такая модель может быть записана следующим образом:

$$FR \langle R_1, C_{11}, C_{12}, \dots, C_{1m}, \rangle, \dots \langle R_2, C_{21}, C_{22}, \dots, C_{2m}, \rangle, \dots \langle R_{k1}, C_{km} \rangle \quad (4)$$

где FR – имя фрейма (агента); пара $\langle Ri, Ci \rangle$ – i -й слот фрейма; Ri , – имя слота, Ci – значение слота.

Слот в модели (4) является логической конструкцией для реализации конкретных заданий фрейму - программному агенту. Слоты из фрейма можно удалять, добавлять, изменять функциональное назначение слота-задания. Однако в виде (4) классическое представление слота в виде <имя>, <значение> не может в полной мере отражать требования логической модели программного агента.

Модифицируем структуру слота и приведем его к следующему виду:

$$\text{Slot} = \langle Y, D, \text{dom}, r_i, \Theta, \Sigma \rangle \quad (5)$$

где Y – множество имен атрибутов, D – множество доменов, dom – отображение $Y \Rightarrow D$, r_i – модель-кортеж i -го задания агента, Σ – множество операций над отношениями. $r_i = \{ \{R\}_i \}$, где R_i – множество состояний кортежа r_i , Θ – множество, определяющее начальные условия и признаки выполнения действий в структуре задания.

Слот может быть спроектирован при помощи типового набора атрибутов $Y = \{ \langle \text{OBG} \rangle, \langle \text{ACT} \rangle, \langle \text{CON} \rangle, \langle \text{STA} \rangle \}$. Описание возможных базовых типов приведено в таблице 1.

Таблица 1

Базовые типы атрибутов

Сущность	Имя сущности	Описание
OBJECT	OBG	База данных, файл, папка, диск, PC
ACTION	ACT	Копировать, наблюдать, защищать
CONDITION	CON	IF- THEN, предикат
STATUS	STA	1 – действие над объектом выполнено успешно, 0 – действие не выполнено, * – результат действия неопределенность

С учетом (1), (2) и (5) может быть предложена логическая модель фрейма-программного агента, представленная на рис.1.

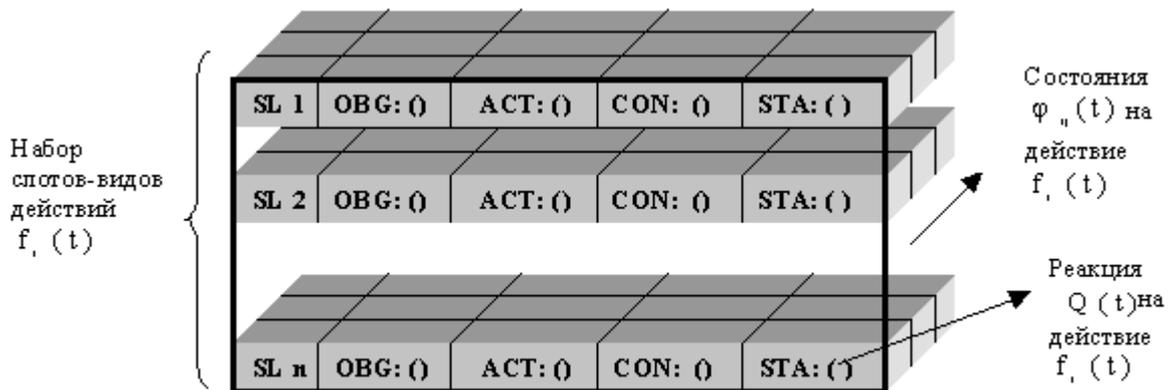


Рис.1. Логическая модель фрейма-программного агента

В предлагаемой модели "фрейм - программный агент", n -слотов образуют конечное число внутренних состояний агента $\Phi_i (i = 1, 2, \dots, n)$. Каждый слот решает одну задачу в информационной среде и обладает одним определенным действием $f_i (i = 1, 2, \dots, n)$, т.е. одному состоянию агента соответствует одно действие. На каждое состояние среда отвечает ответным сигналом $Q(t)$. Тогда поведение программного агента может быть задано уравнениями вида (1) и (2).

Уравнение (1) определяет смену внутренних состояний программного агента под воздействием входной переменной $Q(t)$, а уравнение (2) – зависимость выходного действия агента от его внутреннего состояния [4].

Предположим, что входная переменная $Q(t)$ в общем случае может принимать три значения "0", "1" и "неопределенность". Значение $Q=1$ будет соответствовать успешному завершению действия программного агента в момент времени $t = t^*$ на действие агента $f(t^*)$. Значение $Q=0$ – неудачному действию и действию, находящемуся в режиме обслуживания или выполнения, будет соответствовать состоянию Q ="неопределенность".

Переменная $\varphi(t)$ может принимать n различных значений $\varphi_1, \varphi_2, \dots, \varphi_n$. Эти значения будем называть состояниями программного агента; n – его информационная емкость. Значения переменной $f(t)$ будем называть действиями программного агента. Тогда можно утверждать следующее: в момент времени t программный агент A_g находится в j -м состоянии $j = 1, 2, \dots, n$, если $\varphi(t) = \varphi_j$. Действие f_j называется действием, соответствующим состоянию φ_j , $F(\varphi_j) = f_j$. Уравнение (2) описывает зависимость действий программного агента от его состояний, уравнение (1) – изменения его состояний под воздействием входной переменной. Функция перехода программного агента может быть задана системой матриц. Каждому значению Q_1 соответствует матрица $\|\alpha_{ij}(Q_1)\|$, ($i, j = 1, 2, \dots, n$), которая определяет смену состояний программного агента под воздействием сигнала Q_1 .

Достоинство логического представления модели программного агента, рассмотренного на рис.1., состоит в том, что она может быть объединена с матрицей состояний и, таким образом, отражать текущее состояние и функцию перехода программного агента в зависимости от значений входной переменной $Q(t)$.

Структура фрейма-агента с интегрированной матрицей состояний приведена на рис.2.

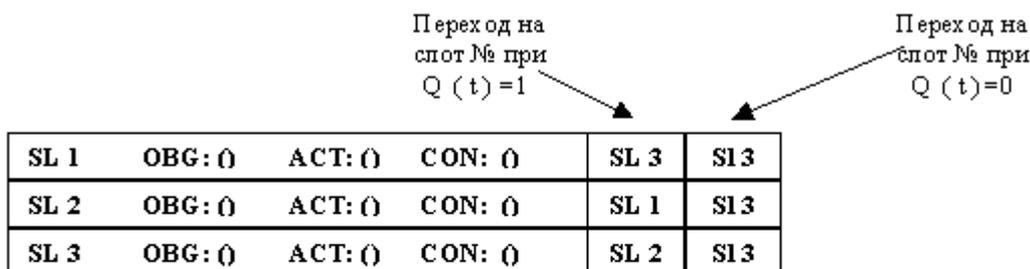


Рис.2. Модель программного агента с присоединенной управляющей матрицей

Выводы

В статье рассмотрены вопросы построения модели автономного поведения программного агента для решения задач управления информационными ресурсами вычислительной системы. Обоснован формальный подход к построению модели поведения программных агентов на основании теории автоматов. Предложенный подход к построению модели поведения программного агента может эффективно использоваться для разработки мультиагентных систем решения задач управления и мониторинга информационных ресурсов распределенных вычислительных систем.

ЛИТЕРАТУРА:

1. Филатов В.А., Чалая Л.С. Модель интеллектуального агента как абстрактный тип данных // Вестник Херсонского государственного технического университета. № 2 (18) – 2003. – с. 216-219.
2. Пономаренко Л.А., Филатов В.А., Цыбульник Е.Е. Агентные технологии в задачах поиска информации и принятия решений // Международный научный журнал "Управляющие системы и машины". № 1 – 2003. – с.36-41.
3. Цетлин М. Л. Исследования по теории автоматов и моделированию биологических систем. – М.: "Наука", 1969. – 316с.
4. Варшавский В.И. Коллективное поведение автоматов. – М.: "Наука", 1973. – 408с.

MODEL OF THE INDEPENDENT AGENT'S BEHAVIOUR BASED ON THE AUTOMATA THEORY

V. Filatov

This clause is devoted to a problem of management the corporate system's information space. The management is carried out on the technology of the program agents. The approach to construction the program agent's model on a basis of frame structure is considered. The frame is fixed as universal extension system. For the decision of concrete tasks the functional modules - slots can be added in it.

Base structure for model of behaviour of the program agents in information environment is the model based on the finite-state machine. The offered models can be used for development the systems for administration of information resources in the allocated computing systems.

МОДЕЛЬ ПОВЕДІНКИ АВТОНОМНОГО АГЕНТА НА ОСНОВІ ТЕОРІЇ АВТОМАТІВ

В.О. Філатов

Запропоновано концептуальний підхід побудови системи автономного управління інформаційними системами та розподіленими базами даних із застосуванням технології програмних агентів. Обґрунтовано вибір модифікованої структури фрейма як логічної моделі програмного агента. Надана модель поведінки агента у мультиагентному просторі на підставі теорії автоматів, що дозволяє ефективно реалізувати управління та адміністрування розподілених інформаційних ресурсів обчислювальних систем.

МОДЕЛЬ ПОВЕДЕНИЯ АВТОНОМНОГО АГЕНТА НА ОСНОВЕ ТЕОРИИ АВТОМАТОВ

В.А. Филатов

В статье рассмотрены вопросы построения модели автономного поведения программного агента для решения задач управления информационными ресурсами вычислительной системы. Обоснован формальный подход к построению модели поведения программных агентов на основании теории автоматов. Предложенный подход к построению модели поведения программного агента может эффективно использоваться для разработки мультиагентных систем решения задач управления и мониторинга информационных ресурсов распределенных вычислительных систем.