

Расплывчатый модифицированный алгоритм построения обобщенной СППР представим в следующем виде:

1. Определение существенных с точки зрения ЛПР свойств системы.
2. Разделение выбранных свойств на внешние, внутренние, неконтролируемые (случайные и расплывчатые) и выходные параметры.
3. Выбор математической формы записи для выражения функциональных зависимостей между входными и выходными параметрами.
4. Построение локальных и общей математической модели.
5. Программно-аппаратная реализация ММ, позволяющая по заданным входным параметрам получать значения либо оценки выходных параметров.
6. Проведение имитационного или эволюционного моделирования для проверки изоморфизма моделей и реальных ситуаций.
7. Оценка адекватности, точности, степени универсальности, экономичности и погрешности ММ для обеспечения компромисса между ожидаемыми результатами математического моделирования и результатами вычислительных экспериментов.
8. Конец работы алгоритма.

Заметим, что для каждой ИИС можно построить большое число математических описаний, которые отражают те или иные ее свойства.

Использование предлагаемых принципов и нечетких алгоритмов анализа моделей позволяет повысить качество и скорость принятия решений в неопределенных условиях.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Грешиков А.А.* Математические методы принятия решений. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2006.
2. *Тихонов А.Н., Цветков В.Я.* Методы и системы поддержки принятия решений. – М.: МАКС Пресс, 2001.
3. *Трахтенгерц Э.А.* Компьютерная поддержка принятия решений. – М.: Синтег, 1998.
4. *Ларичев О.И.* Теория и методы принятия решений, а также Хроника событий в Волшебных Странах. – М.: Логос, 2000.
5. *Курейчик В.В., Сороколетов П.В.* Принятие решений в неопределенных условиях в задачах проектирования РЭА // Известия вузов. Северо-кавказский регион. Технические науки. № 1, 2007. – С. 19-23.
6. *Емельянов В.В., Курейчик В.В.* Теория и практика эволюционного моделирования. – М.: Физматлит, 2003. – 432 с.

УДК 681.3.053

С.А. Жмурко

ОБОБЩЕННАЯ МОДЕЛЬ АГЕНТА И МНОГОАГЕНТНОЙ СИСТЕМЫ*

К основным принципам создания современных перспективных САПР [1] относятся представление САПР как человеко-машинной системы, комплексный подход к автоматизации проектирования, представление САПР как совокупности распределенных информационно-согласованных подсистем, открытость САПР и т.д.

* Работа выполнена при поддержке: РФФИ (грант № 07-01-00174), РНП 2.1.2.3193, РНП 2.1.2.2238, г/б № Т.1.04.01, г/б № Т.12.8.08.

Одним из основных способов, которые позволяют учесть все основополагающие принципы современных САПР, является использование агентов и многоагентных систем в задачах проектирования.

Природа агентов исследуется различными группами ученых, работающими над стандартизацией многоагентных систем (МАС). Некоторые из этих групп представляют собой известные сообщества разработчиков, такие как FIPA (Foundation for Intelligent Physical Agents), OMG (Object Management Group), KAoS (Knowledge-able Agent-oriented System) и др. Современные исследования этих сообществ направлены на изучение природы распределения и взаимодействия агентов в многоагентной системе. Однако распределение агентов и их взаимодействие напрямую зависят от внутренней реализации самого агента и модели многоагентной системы.

Многие сообщества предлагают свои варианты архитектур агентов и МАС. Каждая из таких архитектур имеет ряд особенностей, поэтому разработчик перед тем, как выбрать ту или иную архитектуру, должен изучить все плюсы и минусы известных архитектур и выбрать наиболее подходящую для решения своей задачи, либо же путем синтеза свойств разработать свою архитектуру. Однако первоочередной задачей при анализе конкретных архитектур агентов является определение всех составляющих блоков обобщенной архитектуры агента, анализ основных абстрактных компонент агента.

В работе представлена обобщенная модель архитектуры агента, описывающая основные блоки агента, их назначение и взаимосвязь. Также в работе рассматривается модель многоагентной системы, отражающая взаимосвязь и взаимодействие агентов в многоагентной системе.

Обобщенная модель архитектуры агента. Любой агент представляет собой открытую систему, помещенную в некоторую среду, причем эта система обладает собственным поведением, удовлетворяющим некоторым экстремальным принципам [2]. Таким образом, агент считается способным воспринимать информацию из внешней среды с ограниченным разрешением, обрабатывать ее на основе собственных ресурсов, взаимодействовать с другими агентами и действовать на среду в течение некоторого времени, преследуя свои собственные цели.

Итак, агент – это программно или аппаратно реализованная система, обладающая следующими свойствами:

- ◆ автономность – способность функционировать без прямого вмешательства людей или компьютерных средств и при этом осуществлять самоконтроль над своими действиями и внутренними состояниями [3];
- ◆ общественное поведение (social ability), т.е. способность взаимодействия с другими агентами (а возможно, людьми), обмениваясь сообщениями с помощью языков коммуникации;
- ◆ реактивность – способность воспринимать состояние среды (физического мира, пользователя – через пользовательский интерфейс, совокупности других агентов, сети Internet или сразу все этих компонентов внешней среды);
- ◆ целенаправленная активность (pro-activity) – способность агентов не просто реагировать на стимулы, поступающие из среды, но и осуществлять целенаправленное поведение, проявляя инициативу.

Эти свойства являются основополагающими при разработке агентов и многоагентных систем. Таким образом, главным требованием, предъявляемым к архитектурам агента, является реализация всех вышеперечисленных свойств.

Каждый агент в общем виде состоит из трех основных блоков (рис. 1): блока рецепторов; блока эффекторов; процессора (блока переработки информации и блока памяти).

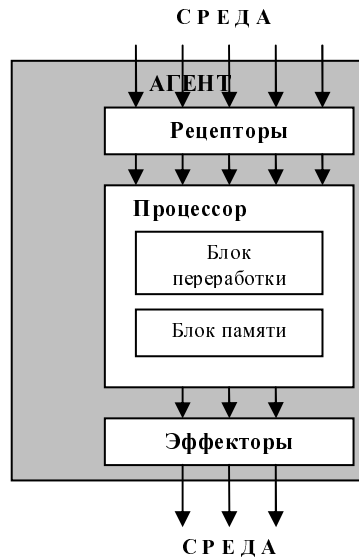


Рис. 1. Обобщенная архитектура агента

Каждый из блоков отвечает за выполнение строго определенных функций.

Блок рецепторов. Рецепторы образуют своеобразную систему восприятия агентов. Они обеспечивают прием информации, поступающей из среды и ее первичную обработку. После этого полученная и обработанная информация передается в блок памяти.

Блок памяти. Память агента должна хранить информацию о типовых реакциях на поступающие сигналы от рецепторов, а так же о состоянии эффекторов и имеющихся ресурсах. Таким образом, блок памяти обычно включает три основных компонента: систему фильтров, обеспечивающих выделение наиболее значимой информации, а так же внутреннюю модель мира и модель самого агента. Память также содержит программы переработки информации в управляющие сигналы эффекторов и результаты реакций на различные внешние ситуации. Немаловажным фактом является объем памяти и разнообразие хранимой в ней информации, так как именно это зачастую и определяет уровень автономности и интеллектуальности агентов.

Процессор. Процессор обеспечивает переработку разнородных данных, принятие решений о выполнении тех или иных действий и выработку реакций на входящую информацию о текущем состоянии среды.

Блок эффекторов. Эффекторы необходимы для воздействия на среду, выдаче информации в символьной форме, поддержании равновесия внутренней среды и т.д.

Иногда в архитектурах агентов выделяют также другие модули, но они не являются обязательными и зачастую входят в состав основных блоков обобщенной архитектуры. Так, для большинства современных агентов одним из основных свойств является коммуникативность. В связи с этим производится выделение всех коммуникативных сигналов, исходящих от агентов. Это приводит к тому, что вы-

деляют отдельный блок – блок коммуникации с другими агентами. Так же при использовании агентом различных методов, к примеру, методов эволюционного моделирования и генетических алгоритмов, разработчики выделяют соответствующие блоки, которые отражают специфику выбранного подхода к разработке агентов. Зачастую, это делается для наглядности, хотя, в действительности, схему коммуникации или использование эволюционного подхода таким же образом можно описать и с помощью процессора, рецепторов и эффекторов.

Необходимо отметить, что одна из ключевых особенностей интеллектуального агента – способность функционировать в условиях неточной, нечеткой и противоречивой информации. В связи с ограниченными возможностями рецепторов и эффекторов агента, он не может ни определять исчерпывающим образом параметры среды, ни точно предсказывать результаты намеченного действия [2]. В связи с этим возникает необходимость рассмотрения полиморфного или нечеткого соответствия между средой E и e моделью M у агента, которое можно назвать нечетким отношением моделирования:

$$R: E \times M \rightarrow [0,1]. \quad (1)$$

Одним из способов записи отношения моделирования является запись в виде матрицы размера $l \times n$, где на пересечении столбца e_j , $j=1, \dots, l$ и соответствующей строки m_i , $i=1, \dots, n$ находится число $\mu_R(e_j, m_i) \in [0,1]$. Это число представляет собой коэффициент соответствия между данным состоянием среды и его внутренним представлением агентом.

Модели многоагентной системы. Для разработки многоагентной системы понимания обобщенной модели агента недостаточно. Для этого необходимо представлять модель самой системы, так как именно она дает понять – какое место занимает агент в ней и как он соотносится с другими компонентами.

Одним из способов описания многоагентной системы является способ, при котором подразумевают, что она (система) $MAS = \{A_1, \dots, A_n; P\}$ состоит из конечного множества $\{A_1, \dots, A_n\}$ интеллектуальных агентов и специального почтового агента P , моделирующего сеть связей между агентами A_i [4]. У интеллектуального агента A имеется внутренняя база данных (БД) I_A , содержащая конечное множество базисных атомов и почтовый ящик $MsgBox_A$.

Согласно современным представлениям, управление командной работой должно выполняться распределенными иерархическими системами с развитыми функциями координации и управления. Ключевой проблемой в таких системах является координация в реальном времени коллективного поведения членов команды, распределено решающих общую задачу [5]. Сложность такого подхода кроется в том, что агенты, составляющие команду, обладают общими и частными навыками и собственным поведением, неполными знаниями об окружающей среде и ограниченными ресурсами. При этом за счет реализации своих индивидуальных задач и коммуникации с другими агентами каждый агент выполняет общую задачу, поставленную перед ними.

Агенты общаются между собой посредством передачи сообщений вида $msg(\text{Sender}, \text{Receiver}, \text{Msg})$, где Sender – агент-источник сообщения, Receiver – агент-адресат, а Msg – сообщение, (передаваемый) базисный атом. Общение агентов происходит через почтового агента P . Агенты посылают сообщения другим агентам системы, посредством передачи этих сообщений агенту P , и получают сообщения от агента P в свой почтовый ящик. При этом можно рассматривать два способа работы:

- ◆ *синхронный*, когда предполагается, что Р передает сообщение адресату сразу (в этом случае агент Р фактически можно исключить из системы);
- ◆ *асинхронный*, когда передача сообщения от Р к агентам может занять произвольное количество времени.

Синхронные МА-системы больше подходят для описания практических систем, работающих в рамках локальной сети (или в отдельном компьютере), в асинхронных МА-системах лучше отражаются свойства сильно распределенных (например, в Интернете) систем. Это связано со спецификой работы сетей, таких как Интернет.

Данный подход к описанию многоагентной системы является довольно доступным к пониманию и основан на представлении агента как сущности обладающей свойством коммуникации. Однако он не отражает всю специфику многоагентной системы.

Существует другой способ формального описания МАС, основанный на понятии алгебраической системы, которая выражается в виде:

$$AS = (X, P, \Omega), \quad (2)$$

где X – непустое множество, называемое носителем или основой системы, P – множество предикатов, Ω – множество операций. Очевидно, что система может быть многоосновной, и в этом случае $X = (X_1, \dots, X_n)$ [3].

Однако агенты редко являются одиночными системами. В настоящее время главный интерес представляют разработки, содержащие большое количество агентов разного назначения. К примеру, в большинстве многоагентных систем выделяют такие типы агентов, различающиеся по ролевому признаку:

- ◆ Агент-заказчик. Он формирует запрос и отправляет заказ на выполнение некоторого задания. Он инициирует и определяет общую задачу, участвует в выработке решений по выполнению задачи и контролирует процесс выполнения заказа. По окончании работы он ее оценивает и сообщает, удовлетворяет ли она его требованиям.
- ◆ Агент-координатор. Он создает и поддерживает сеть запросов и обязательств, необходимых для выполнения работы. Агент-координатор распределяет работу и отбирает требуемые ресурсы для ее успешного выполнения.
- ◆ Агент-исполнитель – это агент, занимающийся непосредственно выполнением задания
- ◆ Агент-субординатор. Он осуществляет общее управление процессами, протекающими на предприятии.
- ◆ Агент-наблюдатель. Он обладает достаточно полной информацией о деятельности предприятия и вырабатывает свои предложения об изменениях в базовой МАС, но может воздействовать на нее только косвенно через агента-субординатора.

Все эти агенты взаимодействуют друг с другом. Следовательно, каждый агент – это процесс, который владеет знаниями об объекте и возможностью обмениваться этими знаниями с другими агентами.

В общем виде многоагентная система включает в себя множество агентов и множество объектов, что может быть записано в виде $X = (A, O)$.

Когда $X = A$ и с учетом того, что МАС представляет собой открытую систему, ее в простейшем случае можно выразить как

$$MAS = (A, E, R, ORG), \quad (3)$$

где A – множество агентов; $E = \{e\}$ – среда, в которой находится данная МАС; R – множество взаимодействий между агентами (например, коммуникативные акты, аукционы и т. п.); ORG – множество базовых организационных структур, соответствующих конкретным функциям (ролям) агентов и установившимся отношениям между ними.

Существует также еще один вариант описания МАС (с акцентом на ее действия):

$$MAS = (A, ACT, \tau, L), \quad (4)$$

где A – множество агентов; ACT – конечное множество действий в МАС; $\tau: A \rightarrow 2ACT$, $ACT_a = \tau(a)$ – множество действий агента $a \in A$, L – подмножество расширенного множества ACT^* , которое описывает поведение всей МАС [6].

В настоящее время от САПР требуется все большее использование прогрессивных технологий. Одной из таких технологий является многоагентный подход к проектированию сложных систем. Применение данного подхода в САПР даст не только некий прирост производительности, но и прибавит гибкость и устойчивость системы при изменении границ решаемых задач. Рассмотренные обобщенная архитектура агента и модели построения системы являются основополагающими при разработке любой многоагентной системы, в том числе и многоагентной системы, используемой для задач проектирования. Реализация всех основных блоков агента (блока рецепторов, блока эффекторов и процессора) является залогом правильного проектирования новой системы. Общий смысл и назначение этих блоков неизменны и одинаковы для разных многоагентных систем, однако реализация этих блоков полностью зависит от разработчика и от поставленной перед ним задачи. Понимание же моделей МАС необходимо для правильного толкования роли и места агента в системе, так как именно модель отражает основные составляющие и их взаимосвязи внутри многоагентной системы.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Жмурко С.А. Многоагентные системы в САПР (часть 1) // Перспективные информационные технологии и интеллектуальные системы. – Таганрог, 2007, №1(29).
2. Тарасов В.Б. От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. – М.: УРСС, 2002.
3. Тарасов В.Б. Агенты, многоагентные системы, виртуальные сообщества: стратегическое направление в информатике и искусственном интеллекте // Новости искусственного интеллекта, 1998, №2. – С. 5-63.
4. Бурмистров М.Ю., Валиев М.К., Дехтярь М.И., Диковский А.Я. О верификации динамических свойств систем взаимодействующих агентов // <http://www.keldysh.ru/papers/2006/art01/obninsk06.htm>
5. Котенко И.В., Станкевич Л.А. Командная работа агентов в реальном времени // Новости искусственного интеллекта, №3(57), 2003. – С. 25-31.
6. Vittikh V.A. Multi-agent systems for modeling of self-organization and cooperation processes // <http://www.cs.brandeis.edu/dept/faculty/mataric>