

Информационные системы и технологии

УДК 681.05.007

А.В. Ищенко, В.С. Рогоза, д-р техн. наук

Агентные технологии: гибридные интеллектуальные системы

Выполнен обзор исследований по вопросам разработки гибридных интеллектуальных систем, как инструментальных средств решения сложных задач. Проанализированы особенности алгоритмической и структурной организации таких систем, представлены их классификационные схемы и этапы построения. Также показана роль агентов в реализациях гибридных интеллектуальных систем.

The current paper presents the survey of investigations on developing hybrid intelligent systems as a tool for solution of complex problems. The specific features of developing hybrid intelligent systems algorithmic and structural building are analyzed, classification schemes are given, the stages of building developing hybrid intelligent systems are discussed, and the role of they implementations with the use of agents are pointed out in the paper.

Ключевые слова: агенты, агентно-ориентированная реализация, гибридные интеллектуальные системы, мультиагентная система, мягкие вычисления, синергизм.

Введение

В связи с усложнением задач, решаемых с привлечением компьютерных средств, особого внимания заслуживают исследования технологий построения компьютерных систем, составленных из «интеллектуальных» компонентов. Хотя понятие «интеллектуальности» искусственных объектов до сих пор четко не определено, однако в практических реализациях компьютерных систем под эту категорию обычно попадают такие объекты, которые способны действовать рационально в среде, подверженной непрерывным изменениям. Поведение объектов считается рациональным, если все их действия направлены на достижение поставленной цели при минимальных вычислительных затратах.

Современные интеллектуальные информационные системы предоставляют пользователям богатый арсенал инструментальных средств для решения самых разнообразных задач из области экономики, инженерного проектирования, дистанционного обучения, управле-

ния виртуальными производствами и научных исследований. Неоспоримым является факт успешного внедрения таких систем во многие сферы нашей жизни. Достаточно вспомнить примеры Web-услуг, предоставляемых Интернетом, средства поиска и отбора информации Data Mining, инструменты мультимедиа или, скажем, Grid-технологии распределенных научных исследований.

Принципиальную особенность интеллектуальных информационных систем исследователи усматривают в том, что по своим функциям такие системы являются сложными объектами для исследований, поскольку они характеризуются такими специфическими свойствами, как [1–5]:

- неоднородность состава;
- способность компонентов системы к адаптации поведения с учетом изменяющихся условий окружающей среды;
- невозможность во многих случаях точного предсказания будущих действий системы по результатам оценки ее прошлых состояний.

В развитии математической платформы интеллектуальных информационных систем важную роль играют как традиционные математические дисциплины (например, как системный анализ), так и новые направления исследований, разрабатываемые на стыке математики и информатики, которые получили развитие в последние десятилетия. К ним, прежде всего, следует отнести методы искусственного интеллекта (ИИ), которые в среде исследователей получили название методов «мягких» вычислений (soft-computing), а именно: нечеткую логику, нейронные сети, эволюционные вычисления, машинное обучение, методы выявления данных в распределенных средах (методы Data Mining), методы теории игр и комбинаторной оптимизации, методы логического вывода и экспертные системы. Альтернативой этим методам являются традиционные методы прикладной математики, которые иногда называют термином «жесткие» вычисления («hard-computing»). Методы решения задач, в основе которых используются комбинации методов первой и второй групп или только лишь методы мягких вычислений, составляют математическую основу для

построения так называемых *гибридных интеллектуальных систем* (ГИС). Одним из выразительных средств программной реализации ГИС можно считать *агентные системы*, разнообразные реализации которых появились в последние десять лет.

Авторы, насколько это было возможно, старались отдавать предпочтение ранним публикациям, которые положили начало исследованиям тех или иных обсуждаемых в работе вопросов.

Целью работы является обзор исследований по вопросам разработки гибридных интеллектуальных систем как инструментальных средств решения сложных задач. В работе описаны две классификационные схемы ГИС, перечислены этапы их построения. Отмечена роль агентов в реализациях гибридных интеллектуальных систем.

1. Определения гибридных интеллектуальных систем

Гибридные интеллектуальные системы представляют собой программные комплексы, предназначенные для решения сложных задач системного проектирования и построенные с использованием комбинаций методов мягких и жестких вычислений, т. е. с использованием гибридных методов вычислений.

В этом определении не оговаривается способ программной реализации таких комплексов, поскольку он определяется тем кругом задач и, соответственно, теми методами, которые выбраны для решения задач в определенной предметной области, а также платформой, на которой реализуется ГИС. Вместе с тем, нужно отметить факт известной «специализации» методов мягких вычислений. Например, нейронные сети изначально создавались как инструментарий для аппроксимации функций, поэтому с помощью нейронных сетей успешно решаются те прикладные задачи, в которых центральное место занимают методы аппроксимации, интерполяции и экстраполяции функций. К таким задачам относятся, например задачи адаптивного управления, построения самообучающихся систем, распознавания образов и речи и ряд других. В задачах, требующих адаптации методов и стратегий решения, нейронные сети имеют явное преимущество перед другими подходами, разрабатываемыми в теории ИИ, например, такими, как экспертные системы.

Однако нейронные сети не являются универсальным инструментом решения всех возможных задач. Так, например, в задачах принятия решений, требующих явных объяснений пу-

тей получения искомым результатов, нейронные сети оказываются беспомощными по сравнению, скажем, с методами вывода правил по индукции, дедукции или абдукции, которые применяются в экспертных и советующих системах.

Часто в системах интеллектуальных вычислений используются нечетко определенные данные, на основе которых строятся так называемые нечеткие рассуждения и делаются приближенные выводы. Структуры данных в таких системах могут содержать нечетко заданные числа и описания в словесной форме (последние относятся к категории лингвистических переменных и констант). Поскольку нечеткость высказываний характерна для человеческих рассуждений, такие системы приближены к способу описания мира в категориях, близких человеку: убеждений, суждений, предпосылок, приближенных числовых характеристик, мнений и т. п. Методы нечеткой логики находят применение в построении экспертных и советующих систем, систем перевода с одного языка на другой, при построении нечетких систем управления и в ряде других приложений.

Наконец, если для решения задач в системе используются эволюционные вычисления, она приобретает способности производить систематизированный случайный поиск и находить оптимальные решения даже в тех случаях, когда целевые функции характеризуются свойствами, усложняющими применение классических методов оптимизации, например, если эти функции мультимодальны. Поэтому генетические алгоритмы успешно применяются для решения задач многоэкстремальной комбинаторной оптимизации.

Однако чаще всего решение сложных задач предполагает использование одновременно нескольких подходов. Рассмотрим пример из области решения экономических задач. Положим, создается интеллектуальная система, предназначенная для моделирования и прогноза состояния рынка ценных бумаг. В этой системе нейронные сети могут быть успешно использованы для классификации различных форм, используемых на рынке ценных бумаг, а также при решении задач планирования финансовых инвестиций. Генетические алгоритмы можно использовать в такой системе для предсказания процентных ставок, а приближенные рассуждения, основанные на нечеткой логике, – для оценки способности клиентов сохранять устойчивость к финансовому риску [6–8].

Объединение методов мягких вычислений может осуществляться двумя путями – на мак-

ро- и микроуровне. Системная интеграция методов на *макроуровне* предполагает формирование ГИС из набора специализированных модулей, каждый из которых реализует один из подходов мягких вычислений и функционирует независимо от других. Системная интеграция методов на *микроуровне* является более высокой формой кооперации методов интеллектуальных вычислений и предполагает реализацию комбинированных подходов непосредственно в методах мягких вычислений; такой способ создания интегрированных методов мягких вычислений определен в литературе термином «синергизм». Например:

- методы, сочетающие в себе методы нейронных сетей и нечеткой логики, относят к категории нейро-нечеткого синергизма (*neuro-fuzzy sznergism*);
- методы, в которых объединены технологии нейронных сетей и генетических алгоритмов, относят к категории нейро-генетического синергизма (*neuro-genetic_algorithm synergism*);
- методы, сочетающие в себе технологии нечетких систем и генетических алгоритмов, относят к разряду нечетко-генетического синергизма (*fuzzy-genetic_algorithm synergism*) и т.д.

Такие сочетания методов мягких вычислений составляют основу так называемых *гибридных синергетических методов интеллектуальных вычислений* [7].

Развитие методов интеллектуальных вычислений, и, прежде всего, гибридных синергетических методов, стимулировало разработку парадигмы *интеллектуальных программных агентов*, как способа программной реализации указанных методов. Некоторые авторы работ по ИИ склонны считать, что искусственный интеллект – это по существу теоретическая платформа интеллектуальных агентов [6].

С точки зрения программной реализации агенты – это компоненты централизованных или распределенных информационных систем, которые характеризуются двумя важными свойствами: во-первых, они обладают способностью к *автономным действиям*, и во-вторых, если речь идет о распределенных системах агентов (такие системы называется *мультиагентными*), то в них агенты способны взаимодействовать между собой путем обмена сложными структурами данных и самостоятельно принимать решения о дальнейших действиях, анализируя состояния среды; подобные действия агентов относят к категории *социальной активности*. Точнее говоря, под социальной активностью

агентов понимаются такие их действия, которые направлены на достижение поставленной цели путем установления сотрудничества с другими агентами. В мультиагентных системах отдельные агенты являются специализированными объектами и выполняют свои собственные задачи, а координация их взаимодействия осуществляется специальным *агентом-менеджером* [6, 8, 9].

2. Гибридные интеллектуальные системы как инструмент решения сложных задач

Гибридные интеллектуальные системы позволяют оперировать различными типами и формами представления данных и знаний, которые могут поступать от различных источников, а также уточнять информацию о предмете исследования, которые используются в процессах рассуждений и принятия решений, достигая, таким образом, более эффективных результатов по сравнению с традиционными подходами [10, 11]. Одной из функций ГИС является способность поиска путей решения задач в средах с нечетко и/или не полностью определенными состояниями, поэтому их программные реализации, такие, например, как агенты, должны обладать способностью оперировать неточной и неполной информацией и достигать поставленных целей путем адаптации своего поведения. В терминологии, предложенной Л. Заде, системы моделирования, функционирующие в нечетко определенных условиях, должны иметь высокий коэффициент машинных интеллектуальных способностей – КМИС (*MIQ, machine intelligent quotient*). Установлено, что из всех возможных версий интеллектуальных систем наибольшим КМИС обладают именно ГИС с системной интеграцией методов на микроуровне [12]. Однако при реализации подобных систем возникает проблема сложности управления процессами вычислений, поскольку они, как правило, состоят из большого числа взаимодействующих между собой программных компонентов, причем эти взаимодействия обычно возникают в непредсказуемые моменты времени по неопределенным заранее причинам между произвольными компонентами. Эти особенности функционирования ГИС принципиально отличают их от традиционных систем моделирования и решения задач и именно они служат основанием для того, чтобы считать действия ГИС интеллектуальными.

Многими исследователями отмечается тот факт, что, хотя программирование методов мягких вычислений, составляющих математическую основу ГИС, является трудной задачей

самой по себе, но все же главной проблемой создания подобных систем является организация системы управления и механизмов взаимодействия компонентов системы. При этом разработчикам необходимо решать следующие вопросы:

- какие методы подходят для определенных типов задач (классификация методов и задач);
- какими сообщениями компоненты ГИС могут обмениваться между собой, и в какой форме эти сообщения следует представлять (если формой реализации ГИС является распределенная среда, то здесь речь идет о протоколах обмена данными);
- как обеспечить легкую замену одних методов на другие, оказавшиеся более удачными для решения данной конкретной задачи, а также каким образом облегчить легкий доступ к данным, хранящимся на различных компонентах системы (вопросы открытости и масштабируемости системы);
- как определять местонахождение тех или иных данных и компонентов в системе, в особенности в тех случаях, когда в распределенной структуре некоторые компоненты системы могут мигрировать от одного компьютера к другому, и каким образом в условиях неоднородной компьютерной среды обеспечить эффективную совместную работу компонентов системы для решения сложных задач, учитывая разнообразие языков программирования, операционных систем и аппаратного окружения.

Хотя эти вопросы являются предметом активных обсуждений в периодических изданиях уже в течение довольно продолжительного периода времени, необходимо признать тот факт, что по сей день отсутствуют универсальные рекомендации, касающиеся практической реализации систем управления ГИС. Для выработки таких рекомендаций в литературе выполняется определенная систематизация основных подходов и их практической реализации. В работе [13] на примерах практической реализации ГИС обоснованы следующие положения:

- Сложность часто принимает форму иерархии. Т. е. система состоит из взаимосвязанных подсистем, каждая из которых, в свою очередь, проявляет структурную иерархию. Указанная иерархия по-разному реализуется в различных системах, однако обнаруживаются некоторые общие формы управления подобными иерархическими структурами, которые реализованы в некоторых известных моделях, таких, например,

как модели распределенных систем *клиент-сервер*, *равноправных узлов* и командной обработки на основе модели *классной доски*. Дополнительная сложность, присущая именно ГИС, состоит в том, что взаимосвязи между ее компонентами, как правило, не статичны, а могут изменяться во времени.

- ГИС с устойчивыми промежуточными уровнями иерархии более открыты и масштабируемы по сравнению с теми структурами, в которых такие промежуточные уровни не найдены.
- При проектировании ГИС следует различать два типа взаимодействий: те, которые осуществляются между небольшими частями системы, выполняющими относительно простые функции, и те, которые реализуются между крупными подсистемами ГИС. Первые более быстрые и более предсказуемые, чем вторые; соответственно, необходимо предусмотреть различные механизмы управления такими взаимодействиями.

Разделение взаимодействий компонентов ГИС на указанные две группы можно схематически представить так, как показано на рис. 1 [14]. На рисунке показано, что система может быть разделена на ряд крупных частей (подсистем), взаимодействия между которыми характеризуются относительно малой частотой и называются *слабыми связями*.

Внутри каждой подсистемы выделяются ряд более мелких компонентов с относительно высокой частотой взаимодействий, то есть компонентов с *сильными связями*. Двухнаправленными стрелками отражены слабые связи между подсистемами, пунктирными линиями – слабые связи между конкретными компонентами этих подсистем, а сплошными линиями – сильные связи между компонентами внутри подсистем.

Руководствуясь изложенными выше соображениями, разработчики программного обеспечения (ПО) сформулировали три базовых принципа реализации ГИС [14]:

- *Декомпозиция*, т. е. разделение функций ГИС на более простые действия. Этот принцип, давно применявшийся при реализации программных комплексов, оказался действенным и в реализациях ГИС, в особенности в условиях специализации отдельных частей синергетических ГИС.
- *Абстрагирование* – это процесс определения упрощенной модели системы, в которой выделены наиболее существенные свойства и детали проекта (которые должны быть отражены в общей структурной схеме ГИС) и

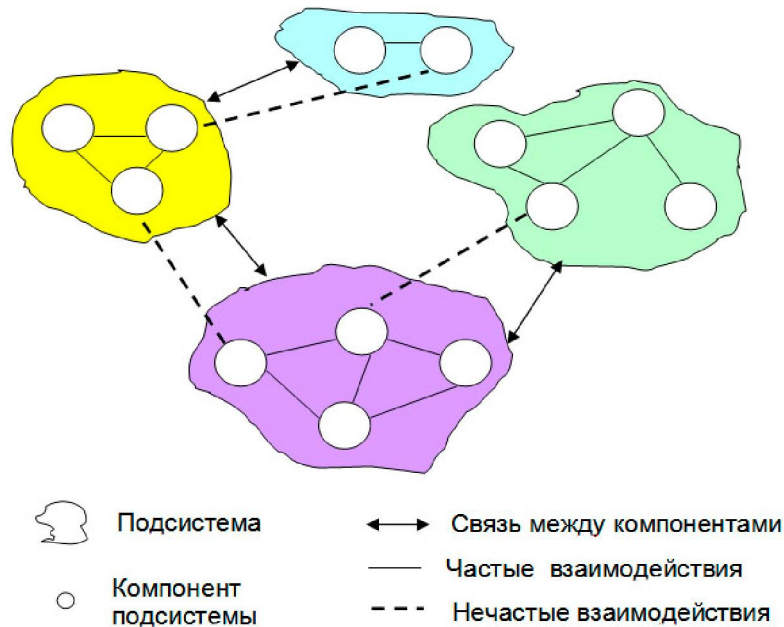


Рис. 1.

опущены второстепенные детали. Цель абстрагирования – создание «настолько упрощенной модели системы, насколько это возможно, но не проще».

- *Иерархическая структурная организация*, т.е. разделение функций ГИС по уровням иерархии. В процессе такого разделения осуществляется идентификация функций частей системы и формулирование правил управления взаимодействиями компонентов решения задачи, расположенных на различных уровнях иерархии.

Эти принципы позволяют формализовать процессы реализации проекта ГИС, помогая разработчикам преодолевать трудности и избегать ошибок, которые сопутствовали проектировщикам первых ГИС.

3. Агенты как инструментальный построения ГИС

Агентные технологии, развивавшиеся поначалу параллельно и независимо от методов построения ГИС, в настоящее время предоставляют в руки разработчиков эффективные инструментальные средства программной реализации ГИС. Принципы, закладываемые в организации агентных систем, оказались близки к тем, на которых строятся ГИС, а кроме того, эти два направления дополняют друг друга, способствуя расширению области применений компьютерного моделирования, основанного на методах ИИ [15, 16]. В качестве подтверждения этих положений, приведем определение агентов, известное в среде разработчиков программного обеспечения [9, 15, 16]:

Агент – это специализированный программный компонент, который функционирует в некоторой среде и способен к гибким автономным действиям в этой среде, направленным на достижение проектных целей прозрачно (то есть, незаметно) для пользователя.

Областями, в которых агенты нашли особенно широкое применение, являются интеллектуальные производства и интеллектуальные системы инженерного проектирования. Если рассматривать производственные и проектные задачи с точки зрения парадигмы агентно-ориентированного проектирования, то становится понятным, что решение большинства таких задач требует использования множество специализированных агентов, действующих в среде децентрализованно. Таким образом, особенностью среды агентов является наличие не только управляющего органа, осуществляющего общее планирование процессов решения задач, но и множества локальных пунктов управления, причем каждый такой пункт управления реализует интересы вполне определенной группы агентов.

В некоторых высокоорганизованных мультиагентных системах интересы одной группы могут конкурировать с интересами другой группы. Здесь конкуренцию следует понимать как способ получения приемлемых результатов решения задач параллельно несколькими группами агентов, каждая из которых может следовать своим путем. Обычно принимается одно из альтернативных найденных решений по принципу «прав тот, кто решил задачу быстрее других». Такое решение должно удовлетворять оп-

ределенным заранее критериям, но вместе с тем, оно может не оказаться наилучшим среди всех возможных. Следует отметить, что упомянутый принцип принят во многих существующих системах интеллектуальных вычислений и вытекает из тех соображений, что в практической деятельности мы часто можем быть вполне удовлетворены «хорошим» (в определенном смысле) решением, несмотря на то, что оно может оказаться и не самым лучшим.

Некоторой аналогией мультиагентной среды с конкурирующими компонентами можно считать рыночную систему, исследованную в теории игр фон Неймана и Моргенштерна [17]. Однако, в отличие от рыночных субъектов, изучаемых в теории игр, агенты в большей степени, чем эти субъекты, нуждаются во взаимодействии друг с другом для достижения своих собственных целей и в управлении этими взаимодействиями. Поэтому мультиагентная система может быть определена как «слабо связанная» (loosely-coupled) или «сильно связанная» (closely-coupled) сеть сущностей (агентов), которые работают вместе для принятия решений, выходящих за пределы возможностей отдельных сущностей или их знаний» [18]. Эти сущности автономны и образуют вместе гетерогенную среду. На основании этих соображений, характеристики мультиагентных систем определены в работе [19] следующим образом:

- каждый агент обладает неполной информацией или возможностями для принятия решений для конкретной задачи, т. о. каждый агент имеет ограниченный кругозор;
- общая для всех агентов система управле-

ния, как правило, не является единственной системой управления в системе агентов, а порой она вообще отсутствует; агенты находятся под управлением также и локальных систем управления, которые могут быть реализованы непосредственно в составе каждого агента; поэтому говорят, что агентные системы характеризуются децентрализованным управлением;

- вычисления, выполняемые агентами, асинхронны.

В данном контексте уместно применить термин «интеллектуальные» агенты для обозначения сущностей, которые, действуя автономно, вместе с тем проявляют социальную активность для «достижения своих целей и выполнения своих задач, оптимизируя некоторые заданные характеристические параметры» [20].

Таким образом, парадигма мультиагентных систем вполне согласуется с концепцией гибридных интеллектуальных систем. По аналогии с канонической структурой ГИС, приведенной на рис. 1, в работе [21] приведены несколько примеров канонических форм мультиагентных систем, одна из которых показана на рис. 2.

Агентно-ориентированная реализация ГИС предполагает декомпозицию решаемой задачи на множество подзадач, решаемых отдельными группами агентов. Последние способны взаимодействовать друг с другом для достижения общих целей. Обмен данными и взаимодействие агентов могут быть реализованы с использованием известных технологий, например, с применением программных средств Web-систем.

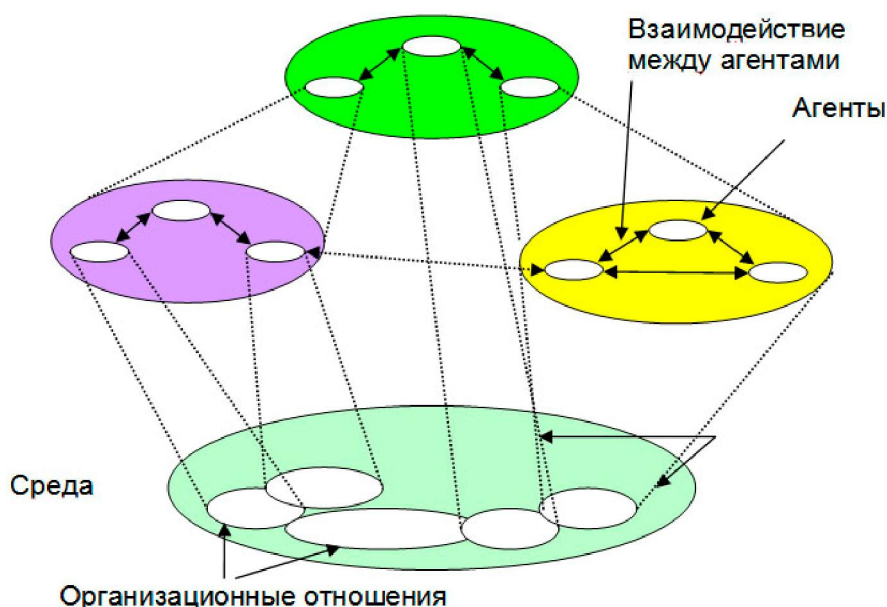


Рис. 2.

4. Классификация гибридных интеллектуальных систем и этапы построения ГИС

Интеллектуальные системы могут быть классифицированы на основе различных критериев. Рассмотрим две типовые схемы классификации по Медскеру-Бейли и Гунатилейку-Хеббалу.

В работе [22] Медскер (Medsker) и Бейли (Bauly) исследовали вопросы построения синергетических ГИС на основе объединения методов экспертных систем и нейронных сетей и выделили пять различных стратегий гибридной реализации таких систем (рис. 3). Дополнительные сведения по этому вопросу можно найти в работах [21–24]. В качестве составляющих элементов модели *автономных комбинированных интеллектуальных систем* выступают программные модули, один из которых реализует методы нейронных сетей, а второй – методы экспертных систем.

В рамках этой модели ГИС предполагается, что указанные модули работают независимо друг от друга и не взаимодействуют между собой в процессе решения задач. Одним из преимуществ такой системы является относительная простота и легкость построения ПО с использованием доступных в продаже коммерческих пакетов. Вместе с тем, каждый модуль работает независимо, поэтому преимущества одного из них не восполняют недостатки другого.

Модель системы преобразования отличается от предыдущей тем, что в этом случае ГИС начинают свою работу как система одного типа (например, как нейронная сеть), а заканчивает – как система другого типа (например, как экспертная система). Когда одна подсистема завершает свою работу, полученные ею данные преобразуются к виду, который может быть воспринят второй подсистемой, после чего первая подсистема заканчивает свою работу. Таким образом, данная модель реализует идею интеграции методов мягких вычислений в ограниченном виде. К недостаткам данной модели относят следующие: во-первых, не существует полностью автоматизированных средств преобразования экспертной системы к нейронной сети или обратно; программная реализация таких преобразований возлагается на проектировщиков и не является тривиальной задачей; и, во-вторых, внесение изменений в систему может повлечь за собой необходимость разработки дополнительных преобразователей форматов данных, которыми обмениваются подсистемы.

Модели ГИС со слабыми связями и сильными связями являются формами ГИС с интеграцией компонентов на высоком уровне и основаны на использовании категорий слабых системных связей (*loosely-coupled interconnections*) и сильных системных связей (*closely-coupled interconnections*) (см. пояснения к рис.1). Сильная связь между компонентами системы приводит



Рис. 3.

к тому, что изменения в устройстве и функционировании одного компонента влекут соответствующие изменения в других компонентах, с которыми данный компонент обменивается сообщениями с высокой частотой. Поскольку оба способа связей, как правило, используются в технологии экспертных систем и нейронных сетей, то обычной схемой реализации сильно связанных ГИС считается такая, при которой такие системы пересылают сообщения в форме структур данных с использованием резидентной памяти, а не через внешние файлы данных. Это улучшает свойства интерактивности тесно связанных модулей. По сравнению с системами с сильными связями, модель со слабыми связями легче реализовать на практике и она допускает большую свободу в выборе интерфейсов и модификации составляющих программных модулей. Однако в обеих моделях не предусмотрен доступ извне к внутренним процессам каждой из подсистем, что может повлечь за собой частичное дублирование вычислительных процедур, выполняемых каждой из подсистем.

Общей для обеих систем особенностью является то, что вследствие использования специального интерфейса передачи файлов, коммуникация усложняется, соответственно время работы возрастает по сравнению с предыдущими вариантами ГИС. В целом, сильная связность обеспечивает большую гибкость и робастную интегральность, чем слабая связность.

Полностью интегрированные системы построены на принципе совместного использования обеими подсистемами структур данных и баз знаний. Коммуникация между различными компонентами осуществляется дуальным способом (напр., посредством передачи сложных структур данных, содержащих одновременно символьную информацию, интерпретируемую на семантическом уровне, и наборы числовых данных). Рассуждения формируются либо кооперативно, либо с использованием дополнительного компонента, работающего как некий универсальный анализатор.

К преимуществам модели полной интеграции следует отнести высокий уровень робастности и возможность получения потенциально лучших решений сложных задач по сравнению с предыдущими подходами. Эти свойства являются результатом дуальной природы представления знаний и структур данных. Кроме того, такие ГИС характеризуются сравнительно небольшой избыточностью. И наконец, полностью интегрированные модели в принципе позволяют реализовать весь набор «интеллектуальных» свойств подобных систем, таких как адаптация,

устойчивость к случайным сбоям, а также реализация способности к объяснению результатов логического вывода в сочетании с высоким уровнем обучаемости.

Недостатком модели полной интеграции является сложность реализации межмодульных взаимодействий и управления потоками сообщений. Проблема широкого использования подобного класса ГИС усугубляется тем обстоятельством, что на рынке ПО пока отсутствуют программные средства, которые отвечали бы требованиям совместимости и открытости. Не полностью решены также вопросы верификации и контроля подобных ГИС.

Авторы второй классификационной системы Гунатилейк (Goonatilake) и Хеббал (Hebbal) [7, 25] отмечали, что существуют три главных причины для создания гибридных интеллектуальных систем:

- улучшение качества технологий решения интеллектуальных задач;
- многообразии прикладных задач, решение которых связано с привлечением методов ИИ;
- разнообразии целей применения ГИС.
- Основываясь на эти тезисы, они разделили гибридные системы на три класса (рис. 4).

В *гибридных системах с замещением функций* реализуется методология объединения функций в общей интеллектуальной компьютерной среде. В процессе решения задачи одна функция (метод) может быть заменена другой функцией (методом), если первая не позволяет получить ожидаемых результатов. Таким образом, главной мотивацией таких замещений функций является стремление добиться наилучших результатов при минимизации объемов вычислений.

Гибридные системы со взаимодействующими между собой компонентами состоят из независимых, самостоятельных, интеллектуальных процессорных модулей, которые обмениваются информацией и выполняют отдельные функции для выработки совместных решений. Такие системы по своим действиям напоминают рассмотренные выше системы преобразования (по классификации Медскера-Бейли).

Если задача может быть разделена на отдельные подзадачи, тогда для решения каждой из них можно использовать независимые интеллектуальные модули, предназначенные для решения задач данного класса. Работа этих независимых модулей, которые совместно решают поставленную задачу, координируется общим модулем управления.

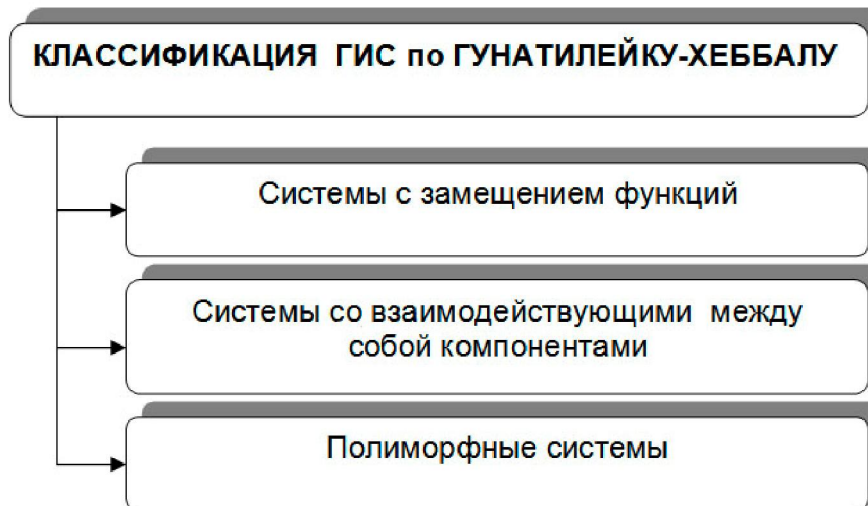


Рис. 4.

Гибридные полиморфные системы – это такие системы, которые используют единую процессорную архитектуру для реализации функций различных интеллектуальных методов мягких вычислений. Общей мотивацией создания таких систем является стремление разработчиков к реализации многофункциональности в рамках единой вычислительной архитектуры.

Выводы

Проанализированы возможности развития специальных программных технологий, например таких, как парадигма агентно-ориентированного, аспектно-ориентированного, компонентного и эволюционного программирования. Рассмотрена проблематика построения адаптивных систем с привлечением новых компьютерных технологий вычислений, таких, как генетические алгоритмы, ДНК-вычисления, квантовые вычисления и ряд других.

Представлены следующие перспективные принципы разработки гибридных интеллектуальных систем [8, 9, 16, 21–25]:

- *Агентно-ориентированная декомпозиция.* ГИС состоят из ряда иерархически связанных подсистем, которые взаимодействуют между собой для достижения общих целей и управляются единым программным модулем или набором локальных управляющих модулей. Построение ГИС в подобной форме связано с декомпозицией задач, которая может быть реализована на основании агентно-ориентированной парадигмы программирования.
- *Установление соответствия между абстрактным описанием задач, решаемых ГИС, и их агентно-ориентированной реализацией в составе ПО.* При проектирова-

нии ГИС наиболее удачной формой декомпозиции программной модели считается такая, при которой программные модули, из которых составляется ГИС, ориентированы на решение тех или иных классов подзадач и, следовательно, являются специализированными объектами. Подобный принцип декомпозиции естественным образом может быть положен в основу агентно-ориентированной реализации ГИС.

- *Учет требования гибкого управления изменяющимися организационными структурами.* Замена или модификации, вносимые в те или иные компоненты ГИС, не требуют перестройки остальных компонентов, поскольку все функции по согласованию взаимодействия этих компонентов возлагаются на промежуточные агенты.

Литература

1. *Винер Н.* Кибернетика, или управление и связь в животном и машине. – М.: Сов. радио, 1968. – 328 с.
2. *Ивахненко А.Г., Зайченко Ю.П., Димитров В.Д.* Принятие решений на основе самоорганизации. – М.: Сов. радио, 1976. – 280 с.
3. *Фельдбаум А.А.* Основы теории оптимальных автоматических систем. – М.: Наука, 1966. – 623 с.
4. *Цыпкин Я.З.* Адаптация и обучение в автоматических системах. – М.: Наука, 1968. – 400 с.
5. *Эшби У.Р.* Конструкция мозга. Происхождение адаптивного поведения. – М.: Изд-во иностр. лит-ры, 1962. – 400 с.
6. *Рассел С., Норвиг П.* Искусственный интеллект. Современный подход. – М.: Вильямс, 2003. – 1408 с.

7. *Goonatilake S., Campbell J., Ahmad, N. Genetic-Fuzzy Systems for Financial Decision Making // Advances in Fuzzy Logic, Neural Networks and Genetic Algorithms / Ed. T. Furuhashi. – Springer, 1995. – P. 202–223.*
8. *Wooldrige M., Jennings N., Kinney D. The Gaia Methodology for Agent-Oriented Analysis and Design // Journal of Autonomous Agents and Multi-Agent Systems. – 2000. – Vol. 3, № 3. – P. 285–312.*
9. *Jennings N.R. On Agent-Based Software Engineering // Artificial Intelligence. – 2000 – Vol. 117. – P. 277–296.*
10. *Booch G. Object-Oriented Analysis and Design with Applications. – Addison Wesley. – 1994. – 650 p.*
11. *Dong Chun Lee, Keun Wang Lee. Design and Implementation of Multi-Agents for Learner-oriented Course Scheduling on the Internet // E-Service Intelligence. Methodologies, Technologies and Applications / Ed. J. Lu, D. Ruan, G. Zhang. – Springer, 2007. – P. 601–626.*
12. *Zadeh L.A. The roles of fuzzy logic and soft computing in the conception, design and deployment of intelligent systems // Software Agents and Soft Computing: Concepts and Applications / Ed. H.S. Nwana, N. Azarmi. – Berlin: Springer-Verlag, 1997. – P. 183.*
13. *Simon H. The Sciences of the Artificial. – MIT Press, 1966. – 412 p.*
14. *Bergenti F., Marie-Pierre Gleizes, and Zambonelli F. Methodologies and Software Engineering for Agent Systems. The Agent-Oriented Software Engineering Handbook. – Kluwer Academic Publishers, 2004. – 505 p.*
15. *Jennings N., Wooldridge M. Agent-Oriented Software Engineering // Handbook of Agent Technologies / Ed. J. Bradshaw. – MIT Press, 2001. – P. 180–196.*
16. *Wooldridge M. Agent-Based Software Engineering // IEEE Proc. Software Engineering. – 1997. – Vol. 144, № 1. – P. 26–37.*
17. *Von Neumann J., Morgenstern O. Theory of Games and Economic Behavior. – Princeton, New Jersey: Princeton University Press, 1944. – 512 p.*
18. *Durfee E.H. and Lesser V. Negotiating Task Decomposition and Allocation Using Partial Global Planning // Distributed Artificial Intelligence / Ed. L. Gasser, M. Huhns. – Pitman Publishing and Morgan Kaufmann, 1989. – Vol.II. – P. 229–244.*
19. *Jennings N.R., Sycara K., and Wooldridge M. A Roadmap of Agent Research and Development. // Autonomous Agents and Multi-Agent Systems. – 1998. – vol. 1. – P. 7–38.*
20. *Weiss G. Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. – MIT Press, 1999. – 480 p.*
21. *Jennings, N.R. An Agent-Based Approach for Building Complex Software Systems // Communications of the ACM. – 2001. – vol. 44, № 4. – P. 35–41.*
22. *Medsker L.R. and Bailey D.L. Models and Guidelines for Integrating Expert Systems and Neural Networks // Hybrid Architectures for Intelligent Systems / Ed. A. Kandel, G. Langholz. – CRC Press, 1992. – P. 154–171.*
23. *Medsker L.R. Hybrid Intelligent Systems. – Kluwer Academic Publishers, 1995. – 512 p.*
24. *Wang L. Balasubramanian S. and Norrie D. Agent-based Intelligent Control System Design for Real-time Distributed Manufacturing Environments // Working Notes of the Agent-Based Manufacturing Workshop. – Minneapolis: MN, 1998. – P. 152–159.*
25. *Dong Chun Lee, Keun Wang Lee. Design and Implementation of Multi-Agents for Learner-oriented Course Scheduling on the Internet // E-Service Intelligence. Methodologies, Technologies and Applications / Ed. J. Lu, D. Ruan, and G. Zhang. – Springer, 2007. – P. 601–626.*