

Искусственный Домовой

Аннотация. В работе дан краткий обзор основных понятий, технологий и проблем построения интеллектуальных систем для помощи человеку в повседневной деятельности (Ambient Intelligence). Для построения таких систем, предлагается использовать технологию гибридных экспертных систем на основе оболочки ESWin и нейронных сетей.

Ключевые слова: гибридные интеллектуальные системы, экспертные системы, нейронные сети, умный дом.

Введение

В последние полтора десятилетия в мире стало популярным направление исследований «Ubiquitous Computing» [1] (вездесущие вычисления). Под ним понимается множество информационных технологий, поддерживающих внедрение вычислительных и коммуникационных функций в различные устройства, с которыми имеют дело люди. Такими устройствами могут быть мобильный телефон, различные бытовые приборы, а также, обычные предметы повседневного пользования (предметы посуды, мебель и т.п.). Толчком для появления работ в этом направлении явилось осознание того факта, что население развитых стран неуклонно стареет [2]. И необходимо автоматизировать уход за престарелыми и немощными людьми, т.е. дать им возможность самостоятельно пользоваться всеми ресурсами дома или другого места проживания, а также, обеспечить мониторинг их здоровья с выходом через коммуникационные возможности на доктора или близких родственников [3]. В рамках этого направления появилось множество новых терминов, таких как «smart environment» (умная среда) с умными объектами (smart objects) [4], “Ambient intelligence” [5, 6] (обволакивающий интеллект), “pervasive computing” [7] (чувствующие вычисления), “proactive computing” [8] (активные вычисления), “Intelligent Space” [9] (интеллектуальное пространство) и т.п. Эти направления развиваются в основном в университетской среде и финансируются различными на-

циональными и международными научными фондами. Ежегодно в мире проводится множество конференций полностью или частично посвященных вездесущим вычислениям. Термин «Ambient Intelligence» и ему подобные плохо переводятся на русский язык. Учитывая основные цели такой системы, помощь человеку в повседневной деятельности, совпадающие с назначением мифического персонажа русского фольклора «Домового», ее можно назвать «Искусственным Домовым».

С другой стороны есть направление развития строительной индустрии и автоматики «smart home» или «умный дом» [10]. В рамках его разрабатываются системы автоматизации жилых и офисных зданий и помещений, целью внедрения которых является, прежде всего, тепло- и энергосбережение, а также, повышение комфорта жизни при использовании разнообразной бытовой техники и обеспечение безопасности жилища.

Можно ожидать, что в ближайшем будущем эти два направления сольются – «вездесущие вычисления» перестанут быть научным направлением с набором экспериментальных технологий, а «умный дом» вберет в себя технологии искусственного интеллекта и станет одной из сторон дальнейшей «киборгизации» человека, т.е. слияния его с искусственным окружением.

В статье приводятся примеры сценариев умного окружения, дан краткий обзор технологий, используемых для его разработки, и примеров проектов исследований и разработок в этом на-

правлении (раздел 1). В разделе 2 приводится обобщенная архитектура умного окружения. В разделе 3 предлагается архитектура гибридной экспертной системы, основанной на представлении знаний и логическом выводе, совместимой с экспертной оболочкой ESWin [32, 33] и взаимодействующей с нейронной сетью. Приводится пример фрагмента базы знаний.

1. Технологии умного окружения

1.1. Примеры сценариев использования умного окружения

В общем случае «умное окружение» (smart environment) с «умными вещами» (smart objects) представляет собой информационную систему, состоящую из сенсорной беспроводной сети, сервера для централизованного сбора информации и управления, беспроводной сенсорной сети, сети «умных объектов», оснащенных локальными встроенными интеллектуальными системами, набора мобильных устройств доступа к системе, обеспечивающих пользовательский интерфейс с встроенными сервисами.

В качестве сенсоров могут использоваться видео-сенсоры (микро-видеокамеры), температурные датчики, датчики освещенности, датчики прикосновения или близости, вмонтированные в мебель, звуковые сенсоры.

В качестве «умных объектов», снабженных интеллектом, могут использоваться повседневные вещи от чашек до одежды. Все эти интеллектуальные предметы обладают некоторой степенью автономности. В то же время, с помощью беспроводной связи они включаются в централизованную систему управления умным домом. С помощью своих сенсоров они могут не только обеспечивать информацией о своем состоянии (например, температура жидкости в чашке или кастрюли), но и определять местоположение человека рядом с собой и некоторые элементы его активности, например, "взял тарелку со стола", "поставил чашку на стол" и т.п. Последнее необходимо для распознавания активности человека системой умного окружения с целью предсказания его желаний/действий и опасных с точки зрения здоровья ситуаций.

Примеры поведения умных вещей:

1. Закладка прицепляется на книгу и на вопрос пользователя голосом "где моя книга?" она отвечает "я здесь", и начинает мигать светодиод.

2. Кофейная чашка предупреждает пользователя голосом или звуковым сигналом о том, что жидкость в ней остыла до минимально приемлемого уровня и отзывается при ее поиске так же как в предыдущем случае.

3. Когда человек садится в кресло перед письменным столом, включается локальный свет и компьютер на столе.

4. Система следит за количеством человек в комнате и включает или выключает освещение в ней.

5. Если человек подошел к книжной полке, включается локальное освещение, помогающее просматривать книги.

6. Если в комнату, предназначенную для отдыха, в обеденное время зашел человек, включается его любимая музыка или любимый телевизионный канал.

7. Умная кровать притушит свет, если на нее лег человек, включит локальный свет, если человек взял книгу для чтения (выключив при этом телевизор), определит положение тела на кровати и его переворачивания, а по этой информации, а также, по звуку дыхания определит, спит человек или нет. По звуку храпа она может разбудить человека и посоветовать ему перевернуться на бок и т.п.

1.2. Технологии умного окружения

Существуют следующие ключевые технологии, лежащие в основе вездесущих вычислений:

- беспроводные сенсорные сети [11],
- контекстно-зависимые вычисления [12, 13, 14],
- много-агентные системы [15].

Кроме того, при построении умного окружения используются различные технологии искусственного интеллекта, направленные на решение следующих основных задач:

- 1) описание (формализация) поведения системы,
- 2) восприятие (perception) системой состояние контролируемой среды (объектов и субъектов),
- 3) удобное взаимодействие системы с субъектами среды на естественном языке,
- 4) обучение системы в процессе взаимодействия со средой поведению и восприятию,
- 5) мониторинг контролируемой среды, включающий в себя задачи диагностики и прогнозирования,
- 6) задачи локализации и отслеживания объектов и субъектов в умном окружении,

7) интеллектуализация предметов повседневного обихода (создание “smart object”) [16], например, посуды («умная чашка»), мебели (кровать, стул, стол), приборов бытовой техники.

Среди технологий искусственного интеллекта, используемых для решения этих задач, наиболее применяемыми являются:

- 1) системы, основанные на знаниях - экспертные системы и онтологии,
- 2) технологии построения обучающихся и обучаемых систем - нейронные сети, решающие деревья, байесовые сети, цепи Маркова,
- 3) семантически-ориентированные методы понимания естественного языка – шаблоны, падежные фреймы,
- 4) методы технического зрения и распознавания образов,
- 5) методы построения гибридных интеллектуальных систем.

Наиболее перспективным подходом к построению интеллектуального окружения является гибридный подход, основанный на комбинации нейронных сетей и экспертных систем [17-24]. Его перспективность в сравнении с другими методами представления и обработки знаний объясняется, во-первых, возможностью формулировать на определенном уровне обработки знания в удобном для восприятия символическом виде (как для ввода знаний в систему, так и для вывода знаний с целью объяснений или отладки системы), а во-вторых, тем, что искусственные нейронные сети являются наиболее близким аналогом природных механизмов обработки данных и приобретения знаний в мозге человека. При этом можно сформулировать следующие требования к нейронным сетям в такой гибридной системе [25]:

- 1) работа в реальном времени как в режиме обучения, так и в режиме использования обученной нейронной сети,
- 2) способность к продолжающемуся обучению (incremental learning) и переобучению,
- 3) возможность легкого извлечения знания из обученной нейронной сети.

1.3. Примеры проектов умного окружения

Все такого рода системы можно разделить на два класса: 1) специфическое умное окружение, ориентированное на решение довольно узкого круга задач, как правила, связанных с обслуживанием престарелых, больных или людей с ограниченными возможностями, и 2) умное

окружение, ориентированное на решение широкого круга задач, связанных с обеспечением комфорта жизни и/или работы.

К первым относятся, например, проекты ALARM-NET [3], SOPRANO [26, 27], PERSONA [28], SC³ [29]. Ко второму – OXIGEN [30], AMIGO [31, 32].

Система ALARM-NET ориентирована на мониторинг за состоянием здоровья пациента за пределами лечебного заведения. Включает в себя домашнюю сенсорную беспроводную сеть, нательную сенсорную сеть (Mobile Body Networks), базу данных с подсистемой анализа на сервере, пользовательский интерфейс через персональный компьютер или смартфон со встроенной системой. Нательная сенсорная сеть включает в себя сенсоры для определения пульса, ЭКГ, акселерометры для обнаружения движений. Она может конфигурироваться в зависимости от потребностей конкретного пациента. Домашняя сенсорная сеть включает в себя датчики температуры, движения, освещенности, запыленности/задымления и предназначена для сбора контекстной информации о местоположении пациента и окружающей среде, а также, для передачи информации от нательной сети к встроенной системе AlarmGate или на сервер. Встроенная система AlarmGate запускается на персональном или мобильном компьютере и предназначена для обеспечения пользовательского интерфейса с системой, связи с Интернет, аутентификации пользователя и предоставления ему ряда сервисов, связанных с управлением системой (безопасности, энергоснабжения и т.п.) и отображения ее состояния. Сервер обеспечивает длительное хранение данных, собираемых сенсорными сетями, и анализ суточной активности пациента с целью создания шаблонов поведения пациента. Эти шаблоны используются для прогнозирования поведения пациента для управления освещением помещений и безопасностью. Пользовательский интерфейс ориентирован на использование разными типами пользователей: доктором, медсестрой, пациентом, членом семьи.

Проект SOPRANO также ориентирован на уход за престарелыми людьми и основан на комбинации онтологий и сервис-ориентированной архитектуры. Ядром проекта является так называемое Middleware Framework или SAM (SOPRANO ambient middleware) - программное обеспечение, предназначенное для связи между

аппаратурой – интеллектуальными сенсорами, исполнительными механизмами, устройствами для доступа к системе, – и серверным аппаратно-программным обеспечением (термин *middleware* является общепринятым в разработке различных проектов умного окружения). Это программное обеспечение планируется использовать в домашних условиях с использованием таких сенсоров как для определения температуры, определения дыма, состояния дверей, местоположения пользователя посредством радаров или RFID, его состояния здоровья и т.п. ПО SAM является интегрирующим компонентом, обеспечивающим открытую сервисную инфраструктуру. Открытость и семантическая согласованность обеспечиваются использованием онтологий, разработка которых является первым шагом при введении нового сервиса и основывается на установлении соглашений между разработчиками решений по удаленному мониторингу здоровья (Telecare Solution Engineer, разрабатывает шаблоны поведения системы), разработчиками ПО (Software Developers, разрабатывают рассуждения на основе контекста Context Reasoning), разработчиками сенсоров и исполнительных устройств (Device Technology Provider).

Проект PERSONA базируется на двух основных концепциях: открытая и масштабируемая платформа, разработка AAL сервисов над невидимой платформой. Эти сервисы являются попыткой обеспечения: социальной интеграции, поддержки повседневной жизни, чувства надежности и защищенности, мобильности. Система реализована на языке Java.

Система SC³ предназначена для ухода за престарелыми людьми в домашних условиях. Она основана на беспроводной сенсорной сети, средствах доступа к облачным данным (Cloud Data, данным в Интернет), средствах аутентификации и распознавания активности пользователя и набора сервисов для пациентов с различными недугами, в частности, страдающих болезнью Альцгеймера. Информация об активности и местоположении пациента формализуется в виде онтологий в облачных данных для анализа его активности и принятия решений. Ядром системы SC³ является программное обеспечение HARE (Human Activity Recognition Engine). Оно состоит из следующих компонентов: отслеживание местоположения (Location Tracking), распознавание активности (Activity

Recognizer, использует 2D- и 3D-видеокамеры), преобразователь (Schema Mapping and XML-Transformer, преобразует информацию об активности в формат OWL), принятие решений (Context-Aware Activity Manipulation Engine, делает выводы об активности высокого уровня на основе первичного распознавания элементарной активности и принимает решения). Кроме этого, в состав ПО HARE входит ряд вспомогательных компонентов: AR Fusion and Collaborator для объединения и взаимодействия различных источников данных об активности, HARE Repository для хранения данных об активности.

Проект OXIGEN реализуется силами MIT и ориентирован на автоматизацию повседневной деятельности в помещении (дома или в офисе). Его особенностью является то, что он базируется на собственных разработках всех подсистем и устройств. Система OXIGEN состоит из устройств Enviro21s (E21s), встроенных в домашнее или офисное окружение, а также, автомобиль, отслеживает и реагирует на изменения в окружающей среде. Эти устройства создают интеллектуальное пространство дома, в здании, в офисе, в автомобиле и обеспечивают речевой пользовательский интерфейс, а также, интерфейсы с различными устройствами, такими как камеры, микрофоны, дисплеи и т.п. Мобильные устройства (handheld devices) Handy21s (H21s) обеспечивают пользователя связью и сервисами независимо от того, где он находится. Это устройство поддерживает речевой и визуальный ввод информации и может реконфигурироваться для поддержки различных протоколов и полезных функций (использоваться как мобильный телефон, GPS навигатор, радио- и телевизионный приемник, видеокамера, персональный ассистент). Динамическая самоконфигурируемая сеть N21s помогает всем компонентам (машинам, людям, сервисам, ресурсам) системы определять местоположение друг друга. При этом она поддерживает различные протоколы для связи в помещении с низким потреблением энергии (low-power point-to-point), для связи внутри здания (building-wide), для связи внутри кампуса (campus-wide). Сеть обеспечивает полностью децентрализованные механизмы именования узлов, определения местоположения и поиска ресурсов. Программное обеспечение O2S адаптируется к изменениям внешней среды и требованиям пользователя и

помогает ему делать то, что он хочет. Адаптация может быть инициирована при использовании нового устройства пользователем, по требованию пользователя или некоторого приложения, при доступности нового ПО или апгрейда, при возникновении ошибки и т.п. ПО O2S использует для этого управляющие и планирующие абстракции, спецификации и память постоянных объектов (persistent object stores) с трансформационной семантикой (transactional semantics) для поддержки процесса изменений. Технологии распознавания речи и технического зрения в проекте OXIGEN взаимно дополняют друг друга. Для распознавания речи используется помимо звукового сигнала распознавание движения губ, мимики лица, особенностей взгляда. Технологии, ориентированные на пользователя (user technologies), в проекте OXIGEN делятся на три категории: 1) технологии автоматизации (Automation technologies), 2) технологии сотрудничества (Collaboration technologies), 3) технологии доступа к знаниям (Knowledge access technologies). Технологии автоматизации позволяют легко описывать в виде скриптов поведение объектов, например, дверей или системы отопления. Технологии сотрудничества позволяют объединять взаимодействующие регионы системы, которые включают в себя крайне мобильных людей или вычисления и могут использоваться, например, для организации совещаний или семинаров. Технологии доступа к знаниям позволяют пользователю обращаться к своей собственной базе знаний, базам знаний своих друзей или корпоративной базе знаний, а также, к знаниям в Интернете.

В основе проекта AMIGO лежит программное обеспечение Basic Middleware, реализованное на Java. Оно состоит из следующих компонентов:

- ПО для разработчиков (The Programming and Deployment Framework),
- Служба управления контекстом – сбора и хранения сенсорной информации (Context Management Service),
- Служба уведомления об изменении контекста (Awareness and Notification),
- Аутентификация и авторизация пользователя (Privacy and Security),
- Моделирование пользователя и его предпочтений (User Modeling and Profiling),

- Преобразование сообщения из одного протокола в другой (Interoperable Service Discovery & Interaction Middleware),

- Представление множества разных устройств как сети унифицированных сервисов (Domotic Infrastructure),

- Преобразование данных в формат DLNA (Content Distribution),

- Подсистема хранения данных и приложений (Content Storage),

- Управление платежными документами в формате IPDR (Accounting and Billing),

- Набор сервисов пользовательского интерфейса (User Interface Services), обеспечивающих речевое общение, распознавание поз и мимики и т.п.

- Описание семантики сервисов и их адаптации (Semantic Service Description) с использованием языка Amigo-S,

- Приложение на Java для редактирования и визуализации онтологий в формате OWL, а также, доступа к ним (VantagePoint),

- Консоль для управления и диагностики системы в целом (Management Console).

2. Архитектура умного окружения

В основе умного окружения лежит информационная система, основанная на беспроводной сенсорной сети, беспроводной сети умных вещей и сервера (Рис. 1).

К сенсорной беспроводной сети подключаются как самые простые датчики, например, тактильные сигнализирующие о том, сел ли кто-нибудь на стул, или закрыта ли дверь, так и микро-видеокамеры, множество которых может представлять собой распределенную систему технического зрения [33]. К беспроводной сети умных вещей могут быть подключены такие виды умных вещей как двери, окна, мебель, посуда, книги, светотехника, бытовая техника, мобильные роботы, смартфоны, а также, обычные персональные компьютеры. Следует иметь в виду, что эти две сети могут различаться на логическом уровне, а на физическом уровне они представляют собой одну беспроводную локальную сеть.

Подсистема распознавания выполняет следующие основные функции: распознавание активности (activity recognition) субъектов, распознавание ситуаций, распознавание субъектов и объектов. Она может быть реализована с ис-

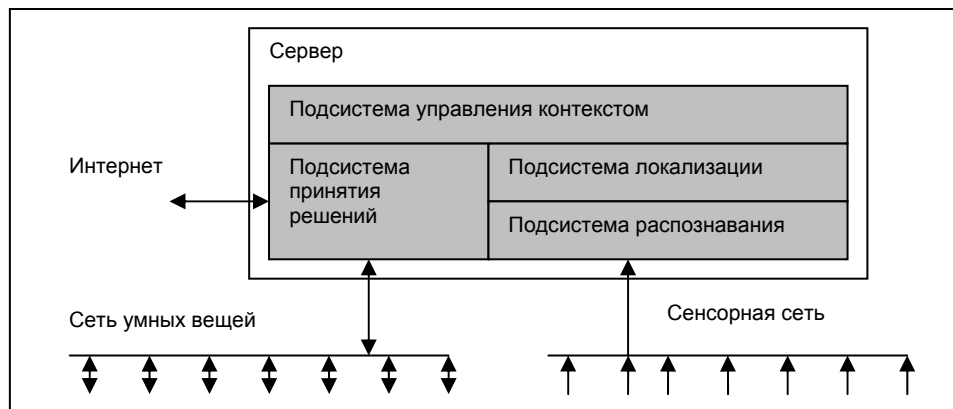


Рис. 1. Структура информационной системы умного окружения

пользованием нейронных сетей, вероятностных моделей, а также, гибридного подхода, включающего в себя и принятие решений, основанное на знаниях.

Подсистема локализации (localization) предназначена для определения местоположения объектов и субъектов на основе локальной информации из различных источников, например, визуальной информации с различных камер. Локализация субъекта может быть основана на использовании распознавания координат по вектору силы сигналов (электромагнитных или ультразвуковых) от нескольких маяков, наподобие того, как это предложено в [34, 35] с использованием нейронных сетей.

Подсистема управления контекстом занимается сбором, сохранением и обновлением контекста, используемого в подсистеме принятия решений. Под контекстом понимается время, место, решаемая задача, состояние решения и другие важные аспекты распознаваемых ситуаций и предпринимаемых действий.

Подсистема принятия решений является наиболее важной частью всей системы, в которой воплощены все сервисы и задачи, решаемые системой. Эта подсистема обычно рассматривается как прикладное ПО в отличие от других подсистем, которые обычно реализуются как, так называемое промежуточное ПО (middleware). Взаимодействие с пользователем эта подсистема реализует через сенсоры (или устройства ввода) и актуаторы умных вещей, в том числе, смартфонов, персонального компьютера или робота. Эта подсистема может включать в себя приложения, выполняющие такие функции как управление освещением, управление мультимедиа аппаратурой, поиск предметов, беседа, скрашивающая одиночество, вопросно-ответная система по раз-

личным вопросам, напоминание о делах (в частности, о приеме лекарств и проведении лечебных и оздоровительных процедур), обеспечение безопасности жилища, мониторинг за состоянием пациента, помощь в ориентации для слабо-видящих людей. Связь с Интернетом необходима для ряда из этих приложений, например, для вопросно-ответной системы при поиске фактографической информации, для системы мониторинга за состоянием пациента при необходимости связи с медперсоналом или владельцем квартиры, оставившем престарелого одного дома.

Умная вещь (smart object) должна обладать встроенной аппаратно-программной платформой, обеспечивающей восприятие соответствующих событий или состояний внешнего мира через сенсоры, воздействовать на внешний мир через актуаторы и обмениваться с другими умными вещами и/или сервером умного окружения через беспроводную сеть (Рис. 2)

Ниже предлагается архитектура гибридной экспертной системы, которая может быть основой для создания подсистемы принятия решений на сервере или в составе умной вещи.

3. Архитектура встроенной гибридной экспертной системы для реализации домового интеллекта

В [36] автором было предложено использовать для создания «интеллекта» умных вещей встроенную гибридную экспертную систему, совместимую с экспертной оболочкой ESWin [37, 38] (доступна на сайте <http://www.insycom.ru>). Оболочка ESWin первоначально была разработана для создания диалоговых экспертных систем для решения задач диагностики, выдачи реко-

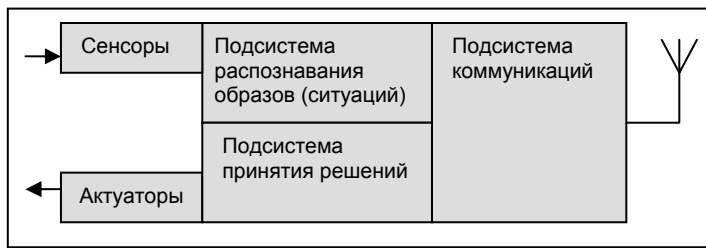


Рис. 2. Структура информационной системы умной вещи (smart object)

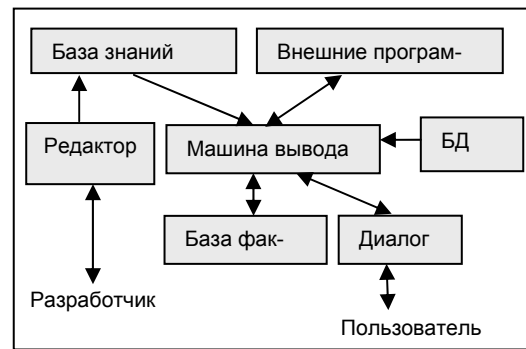


Рис. 3. Архитектура ESWin

мендаций, идентификации и т.п. в хорошо структурируемой предметной области. Но в настоящее время она развивается в направлении возможности построения встроенных гибридных экспертных систем для решения задач управления, например, мобильным роботом. Встраивание подразумевает возможность легкого переноса на разные платформы, которые могут применяться в умном окружении, такие как ПО для «умных вещей», сервер для обеспечения сервисов умного окружения, мобильные телефоны или смартфоны для общения пользователя с умным окружением.

Предлагаемая архитектура гибридной интеллектуальной системы может быть отнесена к модульной в соответствии с классификацией гибридных систем, предложенной в [39].

3.1. Представление знаний

Представление знаний в ESWin (Рис. 3, Рис.4) основано на правилах, фреймах и лингвистических переменных. Фреймы используются для описания структуры мира (предметной области) и пользовательского интерфейса и включают в себя свойства и иерархию сущностей (объектов, понятий) как фреймов-классов. Кроме того, фреймы используются для сохранения текущих фактов о сущностях во фреймах-экземплярах. Правила используются для описания решения задач, в частности, поведения умных объектов. Лингвистические переменные применяются для описания нечетких свойств сущностей, которые могут принимать как качественные (символьные) так и количественные (численные) значения.

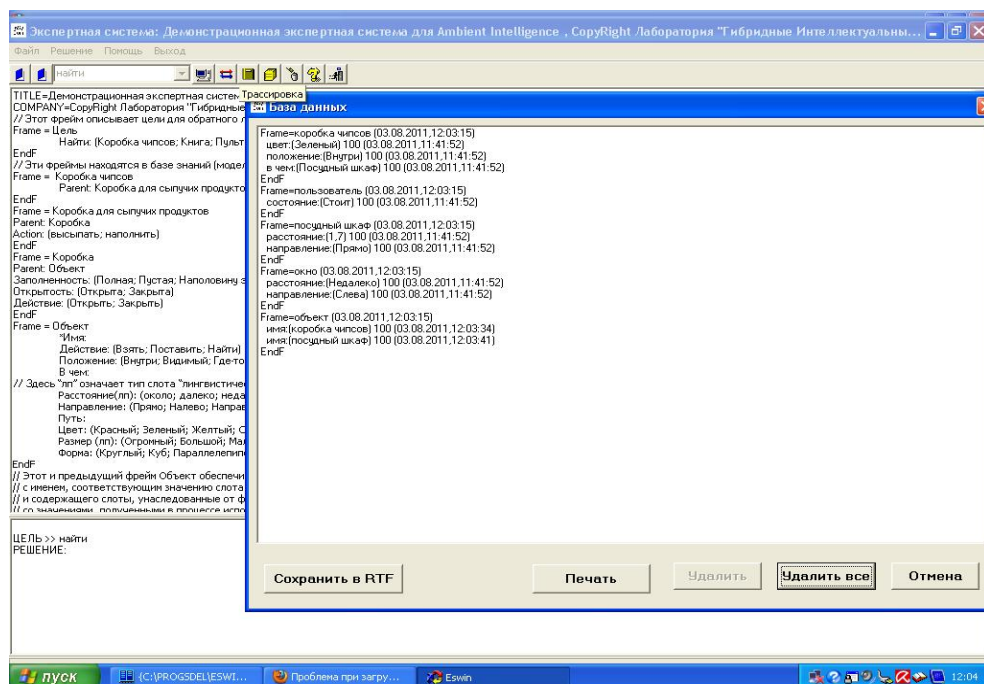


Рис. 4. Скриншот оболочки ESWin для разработчиков

Решение задачи в ESWin представляет собой обратный или прямой нечеткий логический вывод, использующий факты из фреймов-примеров и от сенсоров. Обратный вывод обеспечивает поведение или принятие решения в зависимости от текущего набора фактов и цели. Цель определяет, что мы хотим получить от поведения. Например, в простейшем случае целью может быть название поведения, в частности, «найти». Цель является слотом фрейма «Цель» и при решении задачи получает значение. Например, для цели «название объекта» значением может быть название места, где он находится, при решении за-

дачи поиска объекта. Если в качестве цели используется «найти» (как в примере ниже), то необходимо в используемом правиле позаботиться о вводе имени объекта.

Предположим, что мы имеем дело с сервисом, обеспечивающим помощь для престарелых людей в поиске и манипулировании объектами в помещении, и одна или более видеокамер обеспечивают мониторинг за местоположением и состоянием пользователя. Ниже представлен фрагмент базы знаний, реализованной на языке представления знаний ESWin.

// Этот фрейм описывает цели для обратного логического вывода

Frame = Цель

Найти: (Коробка чипсов; Книга; Пульт управления ТВ; Таблетки; Термометр)

EndF

// Эти фреймы находятся в базе знаний (модель предметной области)

Frame = Коробка чипсов

Parent: Коробка для сыпучих продуктов

EndF

Frame = Коробка для сыпучих продуктов

Parent: Коробка

Action: (высыпать; наполнить)

EndF

Frame = Коробка

Parent: Объект

Заполненность: (Полная; Пустая; Наполовину заполнена)

Открытость: (Открыта; Закрыта)

Действие: (Открыть; Закрыть)

EndF

Frame = Объект

**Имя:*

Действие: (Взять; Поставить; Найти)

Положение: (Внутри; Видимый; Где-то)

В чем:

// Здесь “лп” означает тип слота “лингвистическая переменная”

Расстояние(лп): (около; далеко; недалеко; вплотную)

Направление: (Прямо; Налево; Направо; Сзади)

Путь:

Цвет: (Красный; Зеленый; Желтый; Синий; Белый; Черный)

Размер (лп): (Огромный; Большой; Маленький; Микроскопический)

Форма: (Круглый; Куб; Параллелепипед; Сложной формы)

EndF

// Этот и предыдущий фрейм Объект обеспечивают генерацию фрейма-факта с именем,

// соответствующим значением слота Name фрейма-родителя Объект, и содержащего слоты,

// унаследованные от фрейма Объект, со значениями, полученными в процессе использования

// в правилах соответствующих слотов

Frame = Имя

Parent: Объект

EndF


```

Frame = Пользователь
    Действие: (Повернуться направо; Повернуться налево; Двигаться вперед; Стоп;
    Повернуться назад; Идти к; Искать)
    Состояние: (Стоит; Лежит; Сидит; Идет)
    Направление взгляда:
EndF
// Правила для поиска объекта, находящегося в посудном шкафу
Rule 1
    =(Объект.Имя; *Объект)
    =( *Объект.Положение; Внутри)
    =( *Объект.В чем; *Объект_2)
    =( *Объект_2.Путь; апу)
DO
    =(Цель.Найти; *Объект)
    MS(Действие.Мs; Стоп, возьмите *Объект из *Объект_2)
EndR
Rule 2
    =(Объект.Имя; *Объект_2)
    =( *Объект.В чем; *Объект_2)
    =( *Объект_2.Направление; Прямо)
    <> ( *Объект_2.Расстояние; Вплотную)
DO
    =( *Объект_2.Путь; Прямо)
    MS(Действие.Мs; Идите прямо)
EndR
Rule 3
    =(Объект.Имя; *Объект_2)
    =( *Объект.В чем; *Объект_2)
    =( *Объект_2.Направление; Налево)
    <> ( *Объект_2.Расстояние; Вплотную)
DO
    =( *Объект_2.Путь; Налево)
    MS(Действие.Мs; Поверните налево)
EndR
// эти фреймы находятся в базе фактов и являются результатом обработки сенсорной
// информации или предварительного ввода информации о расположении объектов
Frame=коробка чипсов (экземпляр)
    цвет:(Зеленый)
    положение:(Внутри)
    в чем:(Посудный шкаф)
EndF
Frame=пользователь (экземпляр)
    состояние:(Стоит)
EndF
Frame=посудный шкаф (экземпляр)
    расстояние:(1,7)
    направление:(Прямо)
EndF
Frame=окно (экземпляр)
    расстояние:(Недалеко)
    направление:(Слева)
EndF

```

Здесь выражение **Объект* является аналогом идентификатора переменной, которой пользователь задает значение при выборе цели, например, *Коробка чипсов*. В правиле 1 сначала проверяется, есть ли в окружении распознанный объект *Коробка чипсов*. Если да, создается фрейм-пример с именем *Коробка чипсов*, к слотам которого далее происходит обращение при интерпретации правил. При этом должно происходить взаимодействие оболочки ESWin с подсистемой восприятия/распознавания, реализованной, например, на нейронной сети, получающей данные с датчиков, в частности, визуальных датчиков на базе миниатюрных видеокамер. В этом примере данными, получаемыми от сенсоров, являются: имя распознанного объекта (*коробка чипсов*, *кухонный шкаф*), эти значения присваиваются переменным **Объект* и **Объект_2*, соответственно, при проверке условий, в которых впервые присутствуют эти идентификаторы переменных), значения слотов *Направление* и *Расстояние* применительно к определенным объектам (*Кухонный шкаф* или *Окно*), *Форма* (*Шар*, *Куб* и т.д.), *Цвет* (*Зеленый*, *Красный* и т.д.), *Состояние* и т.п. Некоторые из них могут быть получены при срабатывании правил, например, «*Кухонный шкаф.Направление*» может быть получен из *Пользователь.Направление взгляда* при срабатывании правила (если переменная **Объект_2* имеет значение *Кухонный шкаф*):

Rule 4

=(*Пользователь.Направление взгляда*;

**Объект_2*)

DO

=(**Объект_2.Направление*; *Прямо*)

EndR

Слоты *Объект.Расстояние* и *Объект.Размер* являются лингвистическими переменными, т.е. при логическом выводе используется символическое значение и соответствующий ему коэффициент достоверности, полученные как результат фаззификации из первоначального численного значения, полученного от сенсора.

Слот **Имя* во фрейме *Объект* обеспечивает создание во фрейме-экземпляре *Объект* слота с именем, введенным при запросе слота **Имя*, и этот слот не используется как факт при проверке условий, а используется только для сохранения статистики (какие объекты вовлекались в процесс решения задачи).

Для того чтобы получить более выразительный язык представления знаний для моделирования восприятия и поведения в умном окружении, необходимо модифицировать ESWin в следующих направлениях:

1) Для обеспечения возможности решения задач в течение длительного интервала времени необходимо ввести два режима интерпретации базы знаний: короткое время для принятия решений, и долгое время для мониторинга, например, в предыдущем примере.

2) Для обеспечения возможности управлять несколькими процессами одновременно (например, для реакции на распознаваемые ситуации с учетом их важности) надо ввести в интерпретатор возможность мультизадачности. Это означает, что, начав некоторую цепочку интерпретации правил, мы инициируем некоторый процесс. И должно быть обеспечено взаимодействие между процессами, в частности, прерывание, ожидание и прекращение процессов. В выше приведенном примере интерпретация правила *Rule 1* может быть приостановлена до тех пор, пока не будет выполнено условие $\langle (\text{Кухонный шкаф.Расстояние}; 0,3) \text{ или "расстояние между субъектом и кухонным шкафом"} < 0,3 \text{ метров} \rangle$, т.е. ждать, когда человек подойдет к кухонному шкафу. Очевидно, что надо иметь и механизм отмены этого ожидания. Для того, чтобы обеспечить способность реагировать на распознаваемые ситуации, необходимо, чтобы оболочка ESWin поддерживала прямой логический вывод. Версия 2.1 такой возможностью не обладает. Хотя возможно имитировать реактивность системы с помощью обратного логического вывода и мультизадачности, этот путь является не достаточно эффективным, удобным и прозрачным.

3.2. Взаимодействие экспертной системы с нейронной сетью

Для того чтобы реализовать взаимодействие между принятием решений на уровне правил в ESWin и восприятием на уровне нейронной сети, в [32] был предложен следующий механизм взаимодействия. Специальная фреймоподобная структура данных SOURCE в базе знаний описывает необходимые параметры для определения взаимодействия экспертной оболочки с другими источниками информации помимо диалога с пользователем, в частности, с

нейронной сетью, реализованной в виде внешней программы. Если интерпретатор базы знаний при логическом выводе обнаруживает этот специальный фрейм в условии выполняемого правила, он приостанавливается и посылает запрос соответствующего факта другой программе (нейронной сети) через сокет. После получения факта процесс интерпретации правила продолжается. Если запрашиваемый факт не приходит за определенное время, процесс вывода продолжается с учетом того, что такого факта нет. Если же этот специальный фрейм используется в заключении правила, экспертная система посылает факт другой программе. Ниже приведен пример описания взаимодействия с нейронной сетью.

SOURCE = NN1

Type: Connection

Port: 1234

IP: 127.0.0.1

Timeout: (10) // время ожидания ответа в секундах

Old: (60) // время устаревания факта, полученного от нейронной сети NN1

EndF

Эта структура данных содержит всю информацию, необходимую для взаимодействия с нейронной сетью с именем NN1 посредством сокета через порт 1234. Использование сокета позволяет размещать модули гибридной интеллектуальной системы в разных узлах сети умного окружения.

Переменные, значения которых генерируются нейронной сетью, описываются во фрейме с таким же названием, что и специальный фрейм SOURCE:

FRAME = NN1

Форма: (Куб; Сфера; Параллелепипед; Сложная)

Цвет: (Зеленый; Красный; Желтый; Синий; Белый; Черный)

ENDF

Если используется прямой логический вывод, нейронная сеть может генерировать факт в произвольный момент времени, когда произошло распознавание объекта или ситуации. В любом случае результатом работы нейронной сети должен быть факт, понятный механизму логическому выводу. Для преобразования сигналов, получаемых от нейронной сети в виде вектора, в символьное представление в виде слов или словосочетаний используется словарь.

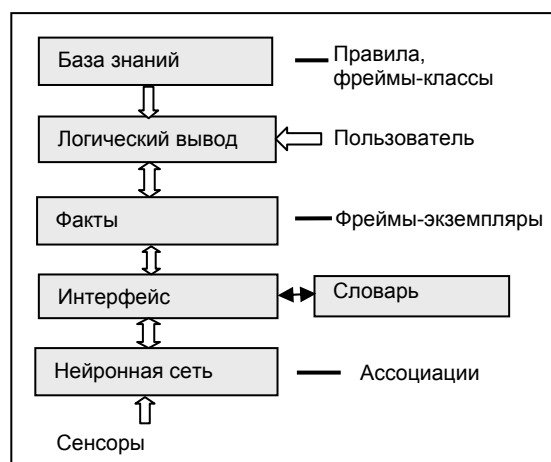


Рис. 5. Уровни обработки информации

В нашем примере словарь может содержать следующие слова: форма, куб, шар, параллелепипед, сложная форма (для распознанной формы), состояние, стоит, лежит, сидит (для распознанного положения/состояния человека), объект, кухонный шкаф, коробка чипсов, чашка, бутылка, окно (для распознанного объекта). Следует заметить, что, если проверяемый факт уже находится в базе фактов, т.е. был получен ранее от нейронной сети, он используется без запроса его от нейронной сети, пока не истечет время устаревания этого факта.

На Рис. 5 показаны уровни представления и использования знаний в предлагаемой гибридной экспертной системе.

В качестве нейронной сети, удовлетворяющей сформулированным в п. 2.2 требованиям, может использоваться сеть ARTMAP. Модель ARTMAP [40] используется для создания ассоциативной памяти для запоминания ассоциаций между двумя разными образами, один из которых может быть кодом слова (понятия) из словаря. В частности, в работе [34] эту модель было предложено использовать в умном окружении для локализации мобильного устройства на основе распознавания вектора величин радиосигналов от различных маяков.

Заключение

В статье рассмотрены основные понятия, особенности и проблемы построения интеллектуального окружения, названного в статье «домовым интеллектом». Приводится краткий обзор существующих проектов в этой области. Предлагается подход к встраиванию интеллекта

в интерьер и «умные вещи», основанный на применении технологии гибридных экспертных систем. В качестве примера рассматривается применение модифицированной экспертной оболочки ESWin. В настоящее время разрабатывается ее модификация для встраивания в программируемые контроллеры. В качестве альтернативы или развития архитектуры «умных вещей» в направлении обеспечения их активности и обучаемости можно предложить подход, основанный на концепции контекстно-зависимого обучения-программирования с использованием диалога на естественном языке, предложенный автором в [41-43] и выходящий за рамки данной статьи. Для разработки интеллекта, встроенного в умные вещи, можно использовать также подход, предложенный в работе [44].

Предложенные подходы могут применяться для разработки умного окружения для престарелых людей или людей с ограниченными двигательными или сенсорными способностями.

Литература

- Weiser, M. "Ubiquitous Computing". IEEE Computer, 26(10):71—72, 1993.
- Martha E. Pollack. Intelligent Technology for an Aging Population: The Use of AI to Assist Elders with Cognitive Impairment. AI Magazine, Vol. 26, No. 2, 2005.
- A. Wood, G. Virone, T. Doan, Q. Cao, L. Selavo, Y. Wu, L. Fang, Z. He, S. Lin, J. Stankovic. "ALARM-NET: Wireless Sensor Networks for Assisted-Living and Residential Monitoring", Technical Report CS-2006-13, University of Virginia, Department of Computer Science, 2006.
- Ju-Jang Lee, Kap-Ho Seo, Changmok Oh and Z. Zenn Bien, "Development of a future Intelligent Sweet Home for the disabled," Artificial Life and Robotics, 11, 2007, pp. 8-12.
- W.Weber, J.M.Rabaey, E.A.Arts (Eds.). Ambient Intelligence. Springer-Verlag, 2005.
- P. Remagnino, G.L.Foresti, T.Ellis (Eds.). Ambient Intelligence. A Novel Paradigm. Springer, 2005.
- M. Satyanarayanan, "Pervasive Computing: Vision and Challenges," IEEE Personal Communications, Aug. 2001, pp. 10-17.
- D.Tennenhouse. Proactive computing. Association for Computing Machinery. Communications of the ACM, 2000, Vol. 43, № 5. - P. 43.
- Hideki Hashimoto, "Present state and future of Intelligent Space—Discussion on the implementation of RT in our environment", Artificial Life and Robotics, 11, 2007. - Pp.1-7.
- C.В.Богданов. Умный дом. Наука и техника, 2005.
- K.Sohraby, D.Minoli, T.Znati. Wireless Sensor Networks. Wiley-Interscience, 2007.
- Baldauf, M., Dustdar S., Rosenberg F. A survey on context-aware systems. Int. J. Ad Hoc and Ubiquitous Computing, Vol. 2, No. 4, 2007. – Pp. 263-277.
- Hung Q. Ngo, Anjum Shehzad, Saad Liaquat Kiani, Maria Riaz, Kim Anh Ngoc, and Sungyoung Lee, "Developing Context-Aware Ubiquitous Computing Systems with a Unified Middleware Framework", The 2004 International Conference on Embedded & Ubiquitous Computing (EUC2004), Springer-Verlag Lecture Notes in Computer Science, August 26-28, 2004, pp. 672-681.
- L.Mehedy, S.-Y.Lee, S.M.S.Zabir, Y.-K. Lee. Scalable and Adaptive Context Delivery Mechanism for Context-Aware Computing. International Journal of Pervasive Computing and Communications, Vol. 4 Issue: 2, 2008. - Pp.166 – 184.
- Multiagent Systems. (Ed. G.Weiss). MIT Press, 1999.
- C.Goumopoulos, A.Kameas. Smart Objects as Components of UbiComp Applications. International Journal of Multimedia and Ubiquitous Engineering, Vol. 4, No. 3, July, 2009.-Pp. 1-20.
- Gavrilov A.V. Hybrid Intelligent Systems in Ubiquitous Computing. Chapter in book "Designing Solutions-Based Ubiquitous and Pervasive Computing: News Issues and Trends" (Eds. F. Milton, P. Fernandes), IDEA Publishing Inc., 2010. – Pp. 263-281.
- Larry R.Medsker. Hybrid Intelligent Systems. 1995.
- Stefan Wermter, Ron Sun, Hybrid Neural Systems. Springer-Verlag, Heidelberg, Germany. 2000.
- Колесников А.В. Гибридные интеллектуальные системы: Теория и технология разработки. – СПб: Изд-во СПбГТУ, 2001. – 711с.
- Гаврилов А.В. Гибридные интеллектуальные системы. - Новосибирск: Изд-во НГТУ, 2003. - 162с.
- Ярушкина Н.Г. Основы теории нечетких и гибридных систем: Учеб. пособие.- М.: Финансы и статистика, 2004.
- Рыбина Г.В. Теория и технология построения интегрированных экспертных систем. М.: Научтехлитиздат, 2008.
- Рыбина Г.В. Инструментальные средства построения интегрированных экспертных систем: развитие комплекса АТ-ТЕХНОЛОГИЯ//Искусственный Интеллект и принятие решений. 2010. №1.
- A.V.Gavrilov, "Usage of Neural Networks in Ubiquitous Computing Systems," in Proc. of 3rd Int. Forum on Strategic Technologies IFOST-2008, Novosibirsk, June 23-29, 2008.
- Michael Klein, Andreas Schmidt, and Rolf Lauer. Ontology-Centred Design of an Ambient Middleware for Assisted Living: The Case of SOPRANO. In: Towards Ambient Intelligence: Methods for Cooperating Ensembles in Ubiquitous Environments AIM-CU, 30th Annual German Conference on Artificial Intelligence KI 2007, Osnabrück, September 10, 2007.
- P. Wolf, A. Schmidt, M. Klein. SOPRANO—An extensible, open AAL platform for elderly people based on semantical contracts. – In Proc. of 3rd Workshop on Artificial Intelligence Techniques for Ambient Intelligence. 18th European Conf. on Artificial Intelligence ECAI-08, Patras, Greece, 2008.
- M.R. Tazari, K. Plößer. User-Centric Service Brokerage in a Personal Multi-Agent Environment. In Proc. of Int. Conf. IEEE KIMAS'03, Cambridge MA (USA), 2003.

29. Asad Masood Khattak, La The Vinh, Dang Viet Hung, Phan Tran Ho Truc, Le Xuan Hung, D. Guan, Zeeshan Pervez, Manhyung Han, Sungyoung Lee, Young-Koo Lee. Context-aware Human Activity Recognition and Decision Making. In Proc. of Int. Conf. IEEE HealthCom-2010, 2010. – Pp. 112-118.
30. Eric S. Brown, Project Oxygen's New Wind, Technology Review, December 2001.
<http://www.technologyreview.com/computing/12706/>
31. Georgantas, N., Ben Mokhtar, S., Bromberg Y.D., Issarny, V., Kalaoja, J., Kantorovitch, J., G  rodolle, A., Mevissen, R. The Amigo Service Architecture for the Open Networked Home Environment. - In Proceedings of 5th Working IEEE/IFIP Conference on Software Architecture (WICSA), 2005.
32. Reinhold Haeb-Umbach, Basilis Kladis, Joerg Schmalenstro  er. Speech Processing in the Networked Home Environment. A View on the Amigo Project. Proc. of Int. Conf. INTERSPEECH-2005, 2005.
33. I.Tosic, P.Frossard. Geometry-based scene representation with distributed vision sensors. Report TR-ITS-2007.11, EPFL, 2007.
34. Uzair Ahmad, Andrey V.Gavrilov, Sungyoung Lee, Young-Koo Lee. Self-scalable Fuzzy ArtMap for Received Signal Strength Based Location Systems. "Soft Computing", Vol.12, N.7, February, 2008. – Pp.699-713.
35. Uzair Ahmad, Andrey V.Gavrilov, Sungyoung Lee, Young-Koo Lee. A Modular Classification Model for Received Signal Strength Based Location Systems. "Neurocomputing", Vol. 71, Issue 13-15, August, 2008. – Pp.2657-2669.
36. A.V.Gavrilov, "Hybrid Rule and Neural Network based Framework for Ubiquitous Computing.", The 4th Int. Conf. on Networked Computing and Advanced Information Management NCM2008, Vol. 2, Gyeongju, Korea, September 2-4, 2008. – Pp. 488-492.
37. Gavrilov A.V., Novickaja J.V. The Toolkit for development of Hybrid Expert Systems. – The 5-th Int. Symp. "KORUS-2001", Tomsk: TPU, 2001. - Proceedings. - Vol.1. - P. 73-75.
38. Gavrilov A.V., Chistykov N.A. An architecture of the toolkit for development of hybrid expert systems. // Proc. Of the Second IASTED Int. Multi-Conference ACIT-2005, Automation, Control and Applications, Novosibirsk, 2005. – Pp. 116-120.
39. McGarry K., Wermter S. & MacIntyre J. Hybrid Neural Systems: From Simple Coupling to Fully Integrated Neural Networks. Neural Computing Surveys, 2, 1999. – Pp. 62-94.
40. G.A. Carpenter, S. Grossberg, N. Markuzon, J.H. Reynolds and D.B. Rosen. Fuzzy ArtMap: neural network architecture for incremental supervised learning of analog multidimensional maps. IEEE Trans. on Neural Networks, vol. 3, 1992. - Pp. 698-713.
41. Andrey V.Gavrilov. Context and Learning based Approach to Programming of Intelligent Equipment. The 8th Int. Conf. on Intelligent Systems Design and Applications ISDA'08. Kaohsiung City, Taiwan, November 26-28, 2008. – Pp. 578-582.
42. Andrey V.Gavrilov. New Paradigm of Context based Programming-Learning of Intelligent Agent. Proc. of 1st Workshop on Networked embedded and control system technologies. In conjunction with 6th International Conference on Informatics in Control, Automation and Robotics ICINCO-2009, Milan, Italy, 2-5 July, 2009. – Pp. 94-99.
43. Гаврилов А.В. Архитектура программного обеспечения для поиска документов по запросу на естественном языке. - Труды Межд. конф. KDS-2001 "Знание-Диалог-Решение". С.-Петербург, 2001. - Т.1, с. 124-130.
44. Прохоров М. Д. Федунев Б.Е. Вывод по прецеденту в базах знаний бортовых интеллектуальных систем, размещаемых на борту антропоцентрических объектов// Искусственный интеллект и принятие решений. 2010.

Гаврилов Андрей Владимирович. Доцент кафедры автоматизации производственных процессов в машиностроении Новосибирского государственного технического университета. Окончил Новосибирский электротехнический институт (ныне НГТУ) в 1974 году. К.т.н., доцент. Количество печатных работ - 129, в том числе 1 монография. Область научных интересов: гибридные интеллектуальные системы, нейронные сети, методы представления нечетких знаний, моделирование понимания естественного языка, интеллектуальные роботы, философия искусственного интеллекта.
E-mail: andr_gavrilov@yahoo.com