

# Разработка высокоуровневой системы управления антропоморфным роботом\*

Корягин Е.В.

Институт прикладной математики и информатики,  
Балтийский федеральный университет им. И.Канта, г. Калининград  
*koryagin.evgeniy@gmail.com*

Описана архитектура разрабатываемой системы управления для человекоподобного полноразмерного робота российского производства. Рассмотрены основные принципы построения когнитивных архитектур и современные достижения в области самообучающихся систем и систем обработки рассуждений. Предложенная модульная структура предполагает высокую организацию данных и способность к адаптации. Обозначены основные вопросы, которые предстоит решить в ходе разработки системы.

**Ключевые слова:** антропоморфный робот, когнитивная архитектура, SOINN, ментальная модель, система обработки знаний и рассуждений, KnowRob, сильный искусственный интеллект.

## Введение

На базе лаборатории интеллектуальной робототехники БФУ им. И. Канта ведется разработка высокоуровневой системы управления (ВСУ) для полноразмерного антропоморфного робота AP-600 (рис.1). AP-600 – это единственный гуманоидный робот, производимый в России. Его разработчики – ОАО «Научно-производственное объединение «Андроидная техника» [1] (далее НПО «АТ»). AP-600 – это человекоподобная высоко-адаптивная платформа. У робота имеются:

- 2 антропоморфных манипулятора (руки)
- 2 антропоморфных педипулятора (ноги)
- Система видеонаблюдения
- Аудио система
- Удаленное управление посредством оператора

Также сотрудниками НПО «АТ» разработан специальный костюм (рис. 3), позволяющий осуществлять телеоперационное управление руками робота.

По конструкции робот AP-600 является аналогом японского робота ASIMO (рис. 2) — это двуногая платформа с головой, двумя пятипальными руками. В отличие от ASIMO, робот AP-600 не имеет сколько-нибудь развитой системы управления, которая позволила бы ему достаточно быстро передвигаться (шагать), двигаться по кривым траекториям, шагать боком, выполнять самостоятельно задачи манипулирования предметами. Также робот AP-600 лишен

---

\* Статья публикуется в порядке дискуссии

## Разработка высокоуровневой системы управления робота

тактильных рецепторов на кончиках пальцев, что ограничивает его манипуляционные возможности. Современное программное обеспечение робота ASIMO появилось в результате кооперативной работы нескольких японских институтов, каждый из которых решал свой комплекс задач. Таким же образом собирается компания НПО «АТ»: проводятся переговоры с университетами и научно-исследовательскими предприятиями с целью разработки программного обеспечения для робота AP-600.



Рис. 1. Робот AP-600.

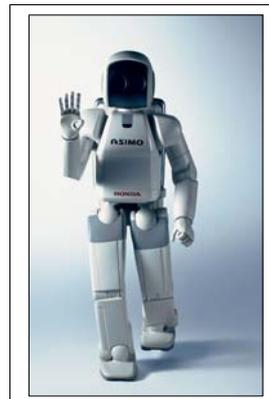


Рис. 2. Робот ASIMO.

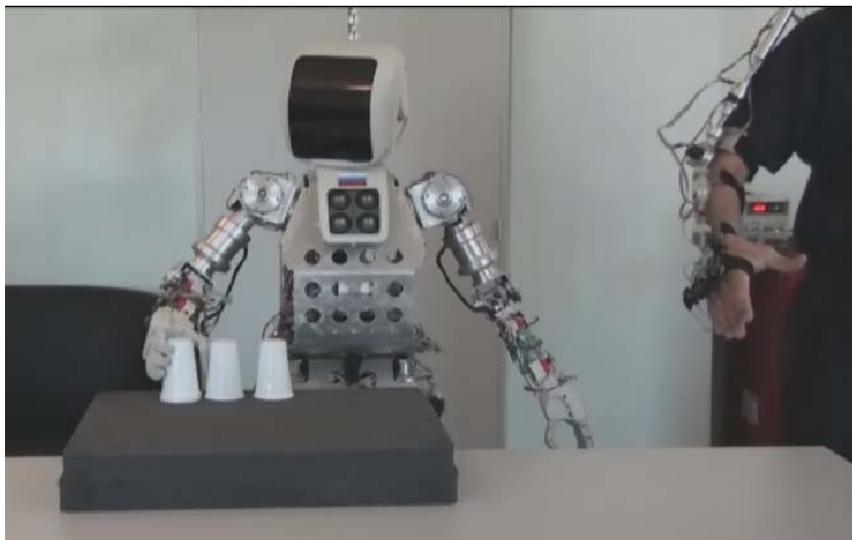


Рис. 3. Костюм управления роботом AP-600.

Разрабатываемая высокоуровневая система будет главным элементом управления антропоморфным роботом AP-600. Система управления таким сложным объектом должна иметь модульную структуру, быть гибкой, отказоустойчивой и адаптивной. Основные задачи, которые должна уметь решать качественная система управления агентом/роботом:

- распознавание объектов/ситуаций, их кластеризация и категоризация
- восприятие окружающей среды, оценка ситуации
- принятие решений (выбор из доступных альтернатив)
- прогнозирование ситуаций, результатов действий
- планирование (нахождение пути достижения поставленной цели)
- выполнение действий
- коммуникация (взаимодействие с человеком, другим агентом)

Пример использования системы: лицо, имеющее на это право (согласно внутренним инструкциям робота) подходит к нему и дает команду: принеси из соседней комнаты маленькую зеленую коробку. Робот распознает речь, анализирует, вычленяет команду, принимает решение, выполняет передвижение, позиционирование, распознавание предмета, захват, перенос предмета и передает предмет человеку, давшему команду.

Статья состоит из нескольких частей. В первой – представлен краткий обзор когнитивных архитектур и основных требований к ним. Разрабатываемая система управления описана во второй части. В третьей части описаны различные технологии построения машинного сознания и адаптивных систем и возможности применения этих технологий в описываемом проекте. Завершают статью краткие выводы и перспективы развития системы.

## 1. Когнитивные архитектуры

Высокоуровневая интеллектуальная система управления роботом должна уметь выполнять некоторые функции так, как это делает человек. Такая система будет состоять из большого числа взаимосвязанных модулей, образующих так называемую когнитивную архитектуру. Когнитивные архитектуры описывают те аспекты поведения интеллектуального агента, которые постоянны во времени и не зависят от конкретной предметной области [2]. Например, когнитивная архитектура должна включать в себя:

- кратковременную и долговременную память, которая хранит накопленные знания, текущие цели и состояния агента (робота)
- структуру представления и взаимосвязи элементов, хранящихся в памяти
- механизмы обработки данных, включая накопление знаний и обучение.

Когнитивная архитектура интеллектуального агента аналогична архитектуре здания – кроме постоянных характеристик, таких как основание, крыша, стены, существуют и временные – мебель или бытовая техника, которые можно передвигать или менять [3].

Основная цель исследований в области когнитивных архитектур (КА) – построение интеллектуального агента, который бы обладал теми же возможностями, что и человек. В какой-то степени когнитивные архитектуры являются противоположностью экспертных систем, которые предоставляют поведение в заранее заданном контексте. КА, напротив, ориентированы на решение как

## Разработка высокоуровневой системы управления робота

можно более широкого диапазона задач.

Ниже рассмотрены наиболее известные когнитивные архитектуры.

### 1.1. АСТ-R

Первые варианты АСТ-R появились еще в 70-х годах прошлого века [4, 5]. Когнитивная архитектура АСТ-R состоит из набора модулей, каждый из которых обрабатывает свой тип информации: сенсорный модуль (для обработки визуальной информации), моторный модуль, модуль целей и декларативный модуль (для долговременного хранения декларативных знаний). Каждый модуль имеет буфер (пространство для хранения текущих данных). Все вместе буферы образуют кратковременную память АСТ-R (рис. 4).

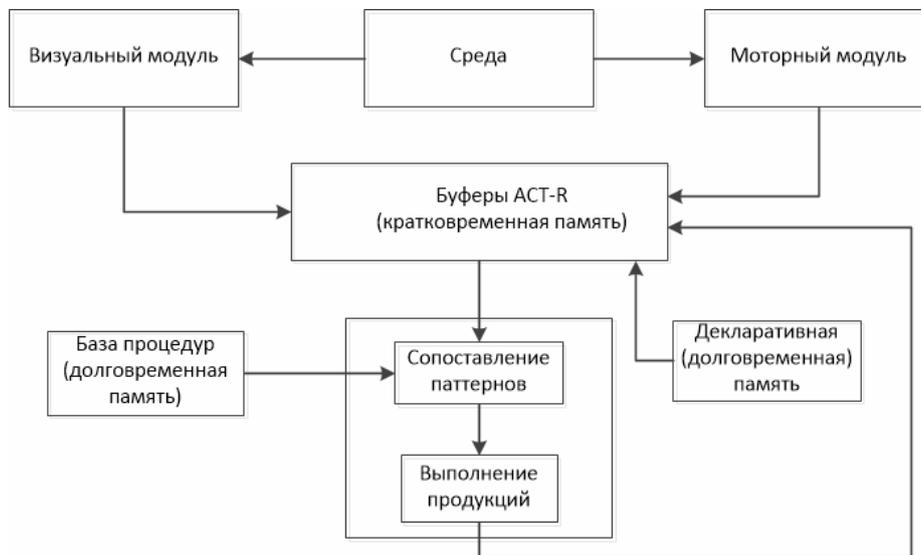


Рис. 4. Схема когнитивной архитектуры АСТ-R.

Хранящиеся в долговременной памяти продукционные правила координируют работу модулей. Каждая продукция в результате своей работы может изменить некоторые текущие данные, или запустить команду моторного модуля, или выполнить запрос к определенным данным из долговременной памяти. Каждый объект данных имеет порог активации, отражающий опыт его прошлого использования, каждая продукция имеет стоимость и вероятность успеха.

В каждом цикле работы АСТ-R определяет, какие продукции соответствуют текущему состоянию кратковременной памяти. Для этого используются пороги активации каждого объекта памяти, удовлетворяющего условиям продукции. Система вычисляет выгоду (utility) для каждой подходящей продукции как разность между ожидаемым выигрышем (приоритет цели, умноженный на вероятность успеха ее достижения) и ожидаемыми затратами (стоимостью). АСТ-R выбирает продукцию с максимальной выгодой и выполняет ее действия, что приводит к появлению новой ситуации, и цикл повторяется.

Обучение в АСТ-R происходит на разных уровнях. Например, пороги ак-

тивации объектов памяти увеличиваются при использовании их в продукциях и уменьшаются в противном случае. Стоимость и вероятность успеха продукций обновляется при каждом успешном или неуспешном выполнении соответствующей продукции.

Сообщество разработчиков ACT-R применяло данную архитектуру для моделирования различных феноменов экспериментальной психологии, таких как память, внимания, рассуждения, обработка языка. Также данная система сыграла важную роль в разработке обучающих систем для школы [6]. Она также применялась для разработки систем управления роботами, основной задачей которых является взаимодействие с человеком [7].

## 1.2. SOAR

Архитектура SOAR [8, 9] появилась в начале 1980-х годов и с тех пор постоянно дорабатывалась и совершенствовалась. В основе продукционной системы в SOAR лежат операторы, ассоциированные с предметной областью. Некоторые операторы представляют собой примитивные действия, которые изменяют внутреннее состояние агента, некоторые – примитивные внешние действия, а некоторые – описывают более абстрактные понятия.

Все задачи в SOAR формулируются в виде попыток достижения цели. Основной цикл работы системы сначала определяет перечень операторов, которые могут быть реализованы в текущих условиях, затем выбирает нужные, выполняет их и оценивает результирующее состояние (рис. 5). В процессе работы системы могут возникать тупики, когда неясно, какой оператор выбрать. В таких ситуациях SOAR создает новую цель – определить, какой оператор выбрать. Таким образом, может быть построена иерархия целей, и задачи могут быть разбиты на подзадачи.

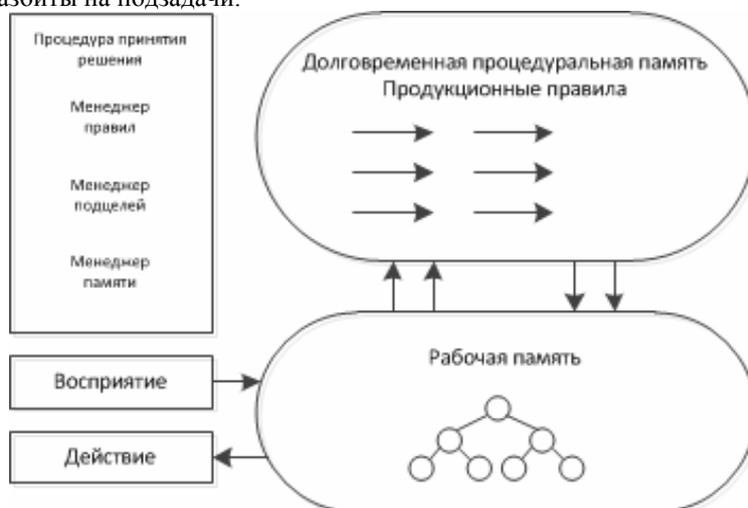


Рис. 5. SOAR.

В качестве основного механизма обучения в SOAR используется обучение с подкреплением (reinforcement learning [39]). Исследователи использовали SOAR для разработки различных сложных агентов, которые обладали внуши-

## Разработка высокоуровневой системы управления робота

тельной функциональностью. Наиболее известная система – боевой тренажер ВВС США TAC-Air-SOAR [10]. Также SOAR применялась в интерактивных компьютерных играх для управления персонажами [11].

### 1.3. ICARUS

Icarus – это относительно новая когнитивная архитектура [12], она хранит два разных типа знаний. Концепты (concepts) описывают классы ситуаций окружающей среды (через концепты-подклассы). Навыки (skills) определяют пути достижения целей в виде упорядоченных подцелей. Концепты и навыки используют отношения между объектами и имеют иерархическую структуру хранения данных в долговременной памяти.

Система работает в цикле «распознавай-действуй». На каждом шаге архитектура сохраняет описания видимых объектов в специальный буфер. Система сравнивает низкоуровневые концепты с содержимым буфера и помещает совпавшие объекты в кратковременную память в качестве гипотез. Эти гипотезы могут совпадать с описаниями некоторых концептов, находящихся выше в иерархии. Такие концепты также добавляются в кратковременную память. Процесс продолжается до тех пор, пока не будут добавлены все возможные концепты. Затем для цели верхнего уровня строится путь ее достижения через иерархию навыков. Для каждого навыка в этом пути удовлетворены начальные условия, но не удовлетворены его цели. Когда путь достигает низшего уровня с простейшим навыком, система выполняет соответствующее ему действие. Это приводит к изменению окружающей среды, и цикл повторяется.

Icarus применялся для решения таких задач, как игры «Ханойская башня», «Солитёр», и планирование в логистике.

### 1.4. PRODIGY

Архитектура Prodigy [13] – это еще одна когнитивная архитектура, которая появилась и развивалась в 1980-х годах. В долговременной памяти Prodigy хранятся два типа данных – операторы, которые описывают следствия применения действий, и правила, которые определяют, когда система должна выбрать или отклонить конкретный оператор. Кратковременная память включает в себя описания состояний и путь достижения цели. Путь этот строится путем выбора операторов, которые уменьшают разницу между текущим состоянием и целью. Модуль обучения анализирует различные ветки решений и создает правила выбора или отклонения определенных операторов для последующего ускорения процесса поиска. Хотя все исследования этой архитектуры сводились к решению задачи планирования, архитектура Prodigy была также основой для очень качественной системы планирования для мобильного робота, которая поддерживала асинхронные запросы от пользователя [14].

Несмотря на многие концептуальные достижения, полученные в течение трех десятков лет исследований в области когнитивных архитектур, и, несмотря на многие успешные реализации некоторых из систем при решении практических проблем, все еще есть необходимость в дополнительной работе в этой важной области. Практически все архитектуры ставят во главу угла задачу поиска решения проблемы или выполнения действий. Но кластеризация и обоб-

щение данных, понимание этих процессов тоже являются важным аспектом, которому пока что не уделяется должное внимание. Большинство КА использует аппарат логики или нечто близкое к нему для представления знаний, однако человек способен воспринимать образную, визуальную, звуковую информацию. Для поддержки таких форм знаний нужны расширенные системы, которые могли бы связать все эти типы информации и использовать их для более интеллектуального и эффективного поведения. Также стоит учесть, что реальные физические агенты имеют ограниченные ресурсы для приема информации об окружающем мире и для воздействия на него. При практической реализации система должна уметь управлять ресурсами агента и фокусировать его внимание на тех задачах, которые наиболее важны в конкретный период времени и в конкретной обстановке.

## 2. Описание разрабатываемой системы

К разрабатываемой для антропоморфного робота AP-600 системе управления (рис. 6) предъявляются серьезные требования. Эта система управления должна:

- аккумулировать данные из остальных систем робота (низкоуровневых модулей)
- планировать поведение робота, требуемое для решения задач
- регламентировать границы самостоятельности робота и окружающих его людей с помощью заранее прописанных правил
- обеспечивать безопасное функционирование робота (контролируя питание, избегая некорректных команд)

Система должна быть размещена на бортовом компьютере робота (модель VDX-6318 на базе процессора 800MHz Vortex86DX). На этом компьютере будет установлена операционная система Ubuntu Linux v.12.04 (или выше) с развернутой средой ROS (Robotics Operating System). ROS [15] – это надстройка над операционной системой, которая предоставляет набор библиотек и инструментов (драйвера, средства контроля отправки/получения сообщений, визуализации, управления пакетами) для создания программного обеспечения для роботов. основополагающим преимуществом ROS является клиент-серверная архитектура, реализованы: механизм пересылки сообщений между различными объектами, возможность построения распределенных систем, предоставление «bridge'ей» к языкам C++ и Python. Все низкоуровневые системы робота и высокоуровневая система управления будут реализованы в виде набора пакетов (так называемый stack в терминологии ROS). Каждая система будет запущена в виде отдельного процесса (node). Общаться узлы между собой будут через специальные именованные каналы (topic) посредством сообщений (msg) определенного формата.

Также в ROS реализованы средства виртуального моделирования и симуляции. Коллективом разработчиков БФУ им. И.Канта будет создана полноценная виртуальная модель робота AP-600, что позволит тестировать все разрабатываемые алгоритмы в симуляции, перед тем как применять их на реальном роботе.

## Разработка высокоуровневой системы управления робота

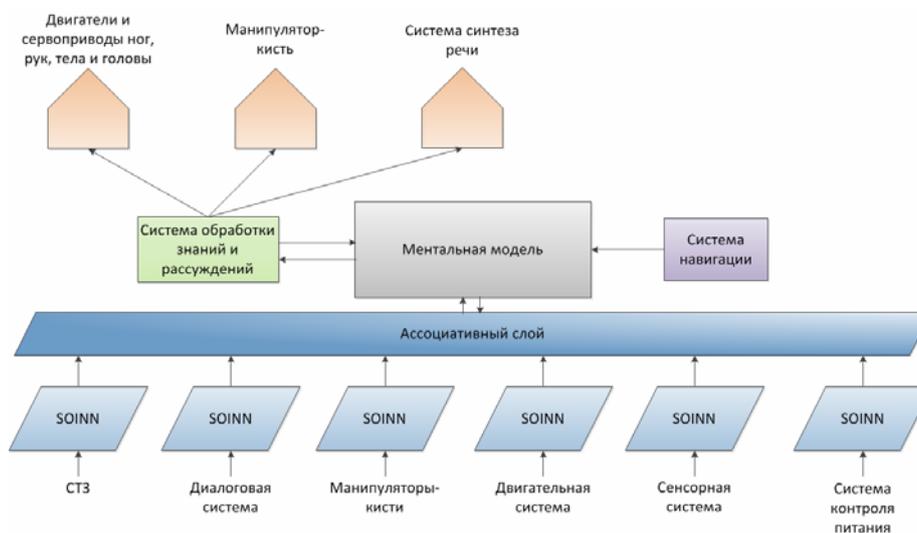


Рис. 6. Общая схема ВСУ. СТЗ – система технического зрения.

### 2.1. Обработка сенсорных данных

Для обработки сенсорных данных и данных с других систем робота будет использоваться набор сетей SOINN (Self-Organizing Incremental Neural Network [16]). Выбор именно SOINN обусловлен тем, что эта сеть не требует предварительно заданной структуры, необходимое количество нейронов формируется в процессе работы системы, она способна обрабатывать новые знания без потери старых, и устойчива к шуму.

Разработанная в Технологическом институте Токио профессором Осаму Хасегавой сеть SOINN [17, 18] является комбинацией SOM (Self-Organizing Map) [19] и Растущего Нейронного Газа (GNG – Growing Neural Gas) [20]. Она позволяет воспроизвести топологию представления данных, обеспечивает on-line обучение без учителя. Основные свойства системы SOINN состоят в следующем:

- нейроны сами организуются, нет необходимости заранее задавать структуру или размер сети
- подходящее количество нейронов адаптивно определяется в процессе работы сети
- система реализует on-line инкрементное обучение без каких-либо заранее заданных условий
- строит узлы-прототипы для больших наборов данных
- устойчива к шуму.

На рис. 7 представлена структура двухслойной SOINN.

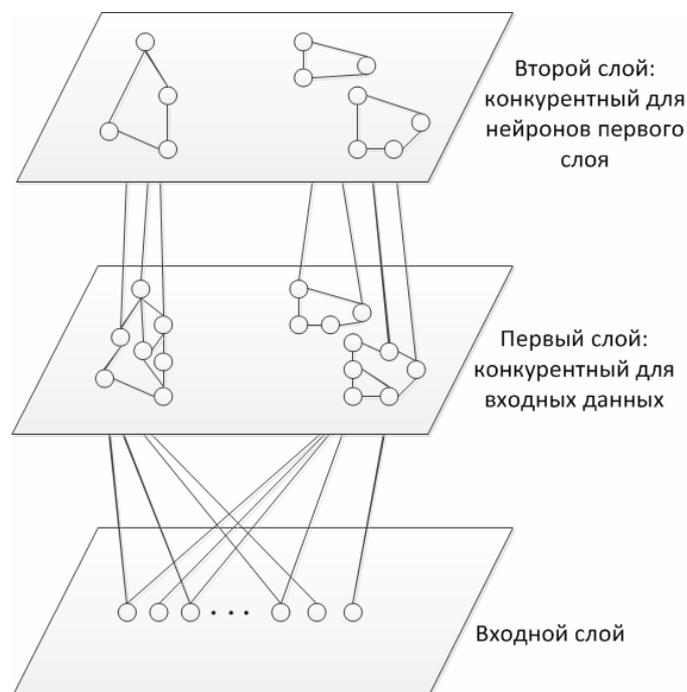


Рис. 7. Двухслойная нейронная сеть SOINN.

В первом слое происходит кластеризация входных векторов. Близко расположенные нейроны соединяются связями, образуя классы. Входной вектор может быть добавлен к слою (тогда он будет отнесен к ближайшему классу или сформирует свой класс). Также нейроны, формирующие определенный класс, могут быть сдвинуты, чтобы усреднить положение класса с учетом поступившего входного вектора.

Аналогичный процесс происходит и во втором слое, что приводит к формированию более четких классов с меньшим числом нейронов-прототипов.

В разрабатываемой ВСУ для АР-600 будет использован модифицированный вариант ассоциативной SOINN [18].

Входные данные с каждой системы робота будут поступать на соответствующую SOINN. Кластеризованные в первом слое данные будут участвовать в формировании ассоциативного слоя. Ассоциативный слой будет содержать по одному нейрону-прототипу каждого класса из каждой системы (рис. 8). Наличие связи между двумя нейронами в ассоциативном слое будет означать, что между этими данными установлена ассоциация.

Перечень систем робота для аккумуляции данных:

- диалоговая система (синтез и распознавание речи)
- система технического зрения
- сенсорная система робота
- система управления рукой
- двигательная система
- система контроля питания.

## Разработка высокоуровневой системы управления робота

Таким образом, при одновременном (или с некоторой задержкой) поступлении информации на различные системы робота будут активированы определенные нейроны-прототипы. Все активные в данный момент нейроны ассоциативного слоя будут соединены связями. Так будет строиться ассоциативная связь между, например, визуальным образом предмета (распознанным изображением шарика) и его звуковым обозначением (произнесенным в этот момент человеком его названием).

Кроме распространения сигнала снизу вверх можно будет запустить активацию нейронов сверху вниз. Например, SOINN диалоговой системы распознала слово "кубик", тогда через соответствующие ассоциации будут активированы связанные с этим понятием классы в зрительной и тактильной SOINN и, спуская возбуждение до нижнего, входного уровня, робот "вспомнит"/"представит" соответствующие кубику ощущения. Такие "фантазии" будут использоваться как для генерации ответного воздействия на окружающую среду (команды низкоуровневым системам), так и для предварительного проектирования, предугадывания возможных ситуаций. Этот метод еще нуждается в дополнительном исследовании и сейчас ведется разработка тестового приложения для проверки такого алгоритма.

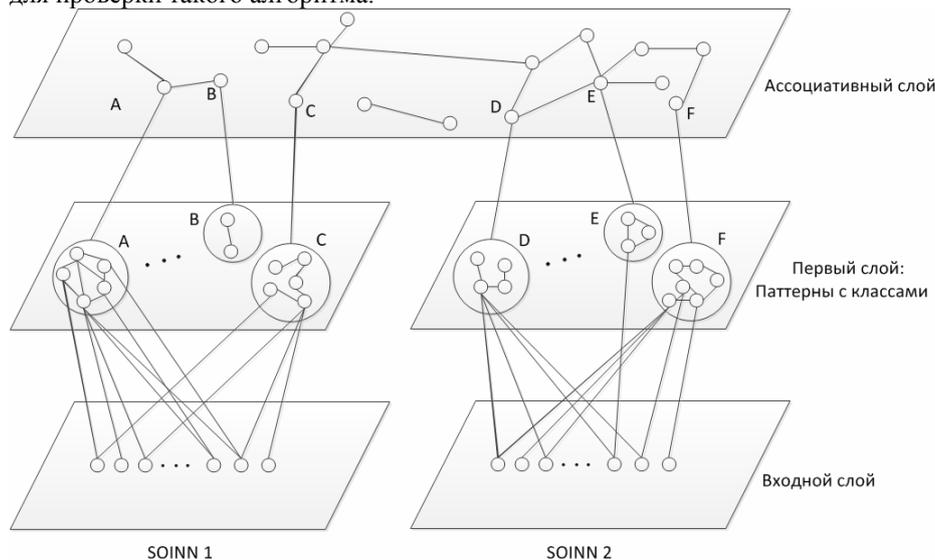


Рис. 8. SOINN с ассоциативной памятью.

### 2.2. Навигация

Для навигации робота в пространстве будет использован метод одновременной локализации и построения карты (Simultaneous Localization And Mapping - SLAM) [21, 22]. Этот алгоритм очень популярен и с его помощью хорошо решаются обе задачи, составляющие название метода: 1) построение карты местности (при этом используется 3D-облако точек, полученное с лазерного сканера) и 2) локализация положения робота на этой карте. Оценка фактического местоположения робота ведется с учетом накопленной информации и сравнивается с текущим набором данных. Два независимых процесса (построение карты

и локализация) связываются в непрерывный цикл последовательных вычислений, при котором результаты одного процесса участвуют в вычислениях другого процесса.

Для ROS уже имеется большое количество реализаций алгоритма SLAM. Наиболее популярный среди них – пакет `gmapping` [23]. Он отличается хорошим качеством и легко встраивается в любые системы. Он идеально подходит для решения наших задач. Пакет `gmapping` будет встроен в виде отдельного узла (node) в ВСУ AP-600, что позволит сэкономить время команде разработчиков и сконцентрироваться на остальных задачах.

### 2.3. Система обработки знаний и рассуждений Knowrob

Необходимым элементом системы управления робота является система вывода. Основные требования к ней:

- возможность хранить и задавать цели управления
- проецировать текущую картину мира на свою базу правил
- генерировать план действий, необходимых для достижения цели
- оценивать результаты выполнения действий
- постоянно пополнять и совершенствовать свою базу.

Основные минусы стандартных производственных систем: данные для них должны быть должным образом структурированы, обычно это сложно при взаимодействии с окружающей средой, реальным миром.

Разработанный немецкой группой ученых из Intelligent Autonomous Systems Group Технического университета Мюнхена фреймворк KnowRob [24, 25] является системой обработки знаний, которая сочетает в себе представление знаний, логический вывод и алгоритмы приобретения новых знаний, и может служить общей семантической средой для интеграции информации из различных источников (рис. 9). Система реализована в виде набора пакетов для ROS [25], т.е. она очень удобна для интеграции в описываемую ВСУ. KnowRob уже содержит серьезную базу данных. Описаны как классические энциклопедические знания, так и знания общего плана, описания задач, моделей окружающей среды, информация об обозреваемых объектах, иерархия задач, планы действий, собранных из самых разных источников: добавленных в систему вручную, из сенсорных данных робота или из Интернета (включая универсальную базу данных для роботов RoboEarth [26]). Система поддерживает различные детерминистские и вероятностные механизмы рассуждений, кластеризации и классификации и имеет свой интерфейс запросов к данным и свои инструменты визуализации [25].

Все знания в KnowRob представлены на языке OWL (Web Ontology Language [27]). Отношения между объектами описаны на диалекте Description Logic [28]. Рассуждения реализованы на языке Prolog.

В системе KnowRob представлены все основные типы данных, действий и явлений, используемых при решении роботом бытовых задач (есть описания для объектов, траекторий, поверхностей, ситуаций, процессов передвижения, слежения, захвата).

Дополнительно в систему управления будут вручную добавлены все правила безопасности. Также будут добавляться действия, изученные с помощью телеоперационного костюма и в разрабатываемой виртуальной среде. Кроме

## Разработка высокоуровневой системы управления робота

того, сохраненные последовательности команд к двигателям будут добавлены к онтологической базе знаний KnowRob для последующего использования (рис. 6).

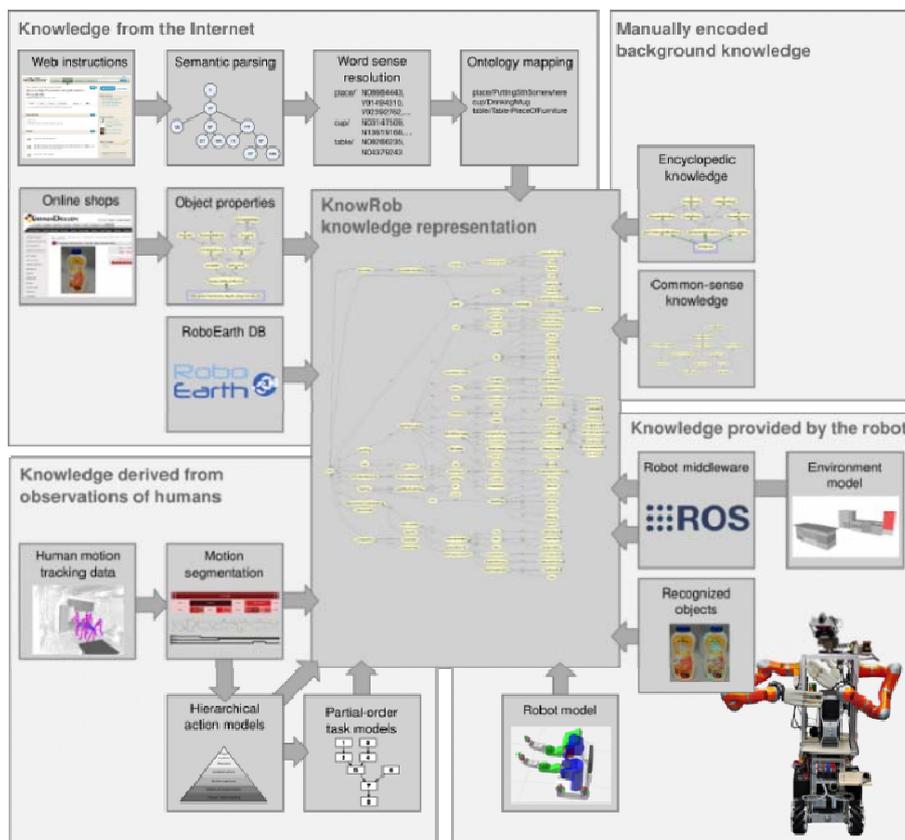


Рис. 9. Схема системы KnowRob [24].

### 2.4. Ментальная модель

В ментальной модели робота (рис. 10) хранится текущая информация об объектах, с которыми идет работа (сколько объектов, их цвет, форма, положение), об активных ассоциациях, о местоположении, о текущей задаче и ее статусе. Это кратковременная память робота. Описанная выше система «фантазий» в виде копии набора сенсорных SOINN будет реализована в рамках ментальной модели. Также на одном из будущих этапов разработки системы предполагается снабдить ментальную модель аппаратом эмоций. В качестве эмоциональных оценок предполагается использовать результаты выполнения (успешные или неуспешные) задач, поставленных перед роботом. Однако пока не ясно, как именно данные оценки будут влиять на поведение робота, этот вопрос еще остается открытым.



Рис. 10. Ментальная модель.

В целом вся система будет представлять собой набор асинхронных пакетов ROS, которые будут общаться между собой (стрелки на рис. 6) посредством сообщений специального формата. В качестве основного языка программирования будет использоваться Python. Это относительно молодой язык, но он очень быстро развивается и имеет большое сообщество разработчиков. Основные правила безопасности и некоторые описания внешней среды (офисного помещения) будут описаны в KnowRob на языке OWL. Также KnowRob поддерживает обращение к своим базам данных и алгоритмам вывода на Python.

### 3. Разработка машинного сознания и сильного искусственного интеллекта

В дальнейших работах над проектом планируется углубиться в исследование вопросов создания сильного искусственного интеллекта и машинного сознания, разработку методик пожизненного (life-long) поведения через адаптацию, обучение и исследование. Тем более в этом направлении уже ведутся интересные работы, в том числе и российскими учеными [29, 30, 31, 32].

Отметим некоторые из интересных подходов.

А.А. Жданов в [33] говорит о важности перехода к технологиям создания адаптивных систем, подобным естественным системам управления. Системы, способные к самоприспособлению, самообучению и саморазвитию являются логическим развитием технологий искусственного интеллекта, и мы стоим на пороге их появления.

В работе В.Г. Редько [34] описана возможность эволюционного возникновения сознания. «Нервная система» робота представляет собой блочно-иерархическую структуру, блоки могут появляться, исчезать и изменяться в процессе «жизни» робота. Тогда вследствие сложности «нервной системы» и нетривиальных мутаций вероятно возникновение блока, ответственного за субъективное Я, который позволяет роботу осознать себя как личность в окружающем мире. В [34] также отмечена важность осознания роботом схемы его организации. В [35] описано «сознание» робота, которое формирует модель самого себя, т.е. внутреннюю модель собственного тела в виде подвижного существа, состоящего из стержней и шарниров. При этом если происходила поломка робота (повреждалась одна из 4-х ног), то робот, как хромая собака, мог подстроить модель самого себя к поломке, и научиться перемещаться, прихрамы-

## Разработка высокоуровневой системы управления робота

вая.

В работах [36, 37, 38] описан подход к разработке систем с постоянной глобальной целью, которая делает поведение агента автономным, позволяя самостоятельно выбирать задачи, которые он будет решать. Эта цель – достижение баланса между поиском новой, непредсказуемой стимуляции и стремлением к предсказуемости результатов своего поведения. Разработанная методика была протестирована на роботе-собаке Aibo (рис. 11)



Рис. 11. Робот Aibo на экспериментальной площадке [36].

Единственной целью поведения робота было максимально успешное обучение. Данный пример показал наиболее существенное следствие этого подхода – способность автономно формировать задачи поведения, осваивать различные навыки от передвижения в пространстве до манипулирования предметами. Робот не решает конкретных задач. Однако разнообразное поведение, которое у него автономно формируется, могло бы стать основой для адаптивного поиска определенных предметов или решения других задач.

Все описанные выше методики могут быть исследованы и протестированы на роботе AP-600 в качестве развития разрабатываемой высокоуровневой системы управления после реализации ее первой версии, описываемой в данной работе.

### 4. Заключение

В рамках работ над проектом разработки высокоуровневой системы управления антропоморфным роботом AP-600 была построена общая структура системы управления, определены основные компоненты хранения и обработки информации.

Сенсорные данные будет обрабатывать набор сетей SOINN, выходы из которых (нейроны-паттерны, обозначающие конкретные признаки) будут аккумулироваться в ассоциативном слое.

Вся информация о текущем местоположении, целях и сенсорная инфор-

мация хранятся в ментальной модели робота (кратковременная память). Описаны возможные усложнения ментальной модели аппаратом «фантазий» и эмоций, однако эти механизмы пока не до конца проработаны и нуждаются в дополнительном исследовании.

В качестве системы хранения знаний и логического вывода выбрана система KnowRob. Она позволяет хранить информацию из различных источников (вручную добавленные знания, сенсорная информация, данные из сети Интернет) и реализует алгоритмы логического вывода для построения планов действий для достижения поставленных задач.

Описаны возможности развития разрабатываемой системы, изучены существующие методики и технологии создания машинного сознания и адаптивного поведения. Работы над данной системой управления имеют потенциал сотрудничества как с российскими институтами, так и с крупнейшими европейскими исследовательскими центрами.

### Список литературы.

1. НПО Андроидная Техника <http://npo-at.com/about/>
2. Laird, J. E., Wray, R. E. III, Marinier, R. P. III, & Langley, P. (2009). Claims and Challenges in Evaluating Human-Level Intelligent Systems // Proceedings of the Second Conference on Artificial General Intelligence.
3. Langley P., Laird J. E., Rogers S. (2008) Cognitive architectures: Research issues and challenges, Science Direct // Cognitive Systems Research. Vol.10. No. 2. PP. 141-160.
4. Anderson, J. R., & Lebiere, C. (1998). The atomic components of thought. Mahwah, NJ: Lawrence Erlbaum.
5. Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind // Psychological Review. Vol. 111. No. 4. PP. 1036–1060.
6. Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. (1997). Intelligent tutoring goes to school in the big city // International Journal of Artificial Intelligence in Education Society. Vol. 8, No. 1. PP. 30–43.
7. Trafton, J. G., Cassimatis, N. L., Bugajska, M., Brock, D., Mintz, F., & Schultz, A. (2005). Enabling effective human–robot interaction using perspective-taking in robots // IEEE Transactions on Systems, Man and Cybernetics, Part A. Vol. 35, No. 4. PP. 460–470.
8. Laird, J. E. (2008). Extending the Soar cognitive architecture // In: Proceedings of the artificial general intelligence conference. Memphis, TN: IOS Press.
9. Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). Soar: An architecture for general intelligence // Artificial Intelligence. Vol. 33, No. 1. PP. 1–64.
10. Tambe, M., Johnson, W. L., Jones, R. M., Koss, F., Laird, J. E., Rosenbloom, P. S., et al. (1995). Intelligent agents for interactive simulation environments // AI Magazine, Vol. 16, No. 1. PP. 15–39.
11. Magerko, B., Laird, J. E., Assanie, M., Kerfoot, A., & Stokes, D. (2004). AI characters and directors for interactive computer games // In: Proceedings of the sixteenth innovative applications of artificial intelligence conference (pp. 877–884). San Jose, CA: AAAI Press.
12. Langley, P., Cummings, K., & Shapiro, D. (2004). Hierarchical skills and cognitive architectures // In: Proceedings of the twenty-sixth annual conference of the cognitive science society (pp. 779–784). Chicago, IL.
13. Carbonell, J. G., Knoblock, C. A., & Minton, S. (1990). PRODIGY: An integrated

## Разработка высокоуровневой системы управления робота

- architecture for planning and learning // In: K. Van Lehn (Ed.), Architectures for intelligence. Hillsdale, NJ: Lawrence Erlbaum.
14. Haigh, K., & Veloso, M. (1996). Interleaving planning and robot execution for asynchronous user requests // In: Proceedings of the international conference on intelligent robots and systems (pp. 148–155). Osaka, Japan: IEEE Press.
  15. ROS. <http://www.ros.org/wiki/>
  16. Research: Hasegawa Lab. <http://haselab.info/soinn-e.html>
  17. Shen Furoo, (2009). An Algorithm for Incremental Unsupervised Learning and Topology Representation, Ph.D Thesis, Tokyo Institute of Technology. URL: [http://haselab.info/papers/shen\\_doctoralThesis.pdf](http://haselab.info/papers/shen_doctoralThesis.pdf)
  18. Furoo Shen and Osamu Hasegawa, Self-organizing Incremental Neural Network and its Applications // Tutorial, International Joint Conference on Neural Networks (IJCNN 2009).
  19. Self-organizing map. [http://en.wikipedia.org/wiki/Self-organizing\\_map](http://en.wikipedia.org/wiki/Self-organizing_map)
  20. Neural gas. [http://en.wikipedia.org/wiki/Neural\\_gas](http://en.wikipedia.org/wiki/Neural_gas)
  21. SLAM.  
[http://en.wikipedia.org/wiki/Simultaneous\\_localization\\_and\\_mapping](http://en.wikipedia.org/wiki/Simultaneous_localization_and_mapping)
  22. Sebastian Thrun, (April 2000) Probabilistic Algorithms in Robotics // CMU-CS-00-126.
  23. GMapping. <http://www.ros.org/wiki/gmapping>
  24. KnowRob: Knowledge processing for robots.  
<https://ias.in.tum.de/research/knowledge>
  25. KnowRob wiki. <http://www.ros.org/wiki/knowrob/>
  26. RoboEarth – World Wide For Robots. <http://www.roboearth.org/>
  27. Web Ontology Language  
[http://en.wikipedia.org/wiki/Web\\_Ontology\\_Language](http://en.wikipedia.org/wiki/Web_Ontology_Language)
  28. Description Logic [http://en.wikipedia.org/wiki/Description\\_Logic](http://en.wikipedia.org/wiki/Description_Logic)
  29. Карпов В.Э., Частные механизмы лидерства и самосознания в групповой робототехнике // Тринадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2012. Труды конференции. Белгород: Изд-во БГТУ, 2012. Т. 3. С. 275-283.
  30. Turchin V.F. (1987) A constructive interpretation of the full set theory // The Journal of Symbolic Logic. Vol. 52. No. 1. PP. 172 -201.
  31. Павловский В.Е., Кедровская М.Н., Емельянова, М.С. Волкова Т.А.. Интеллектуализация мобильного робота: диалог и сенсорика // Тринадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2012. Труды конференции. Белгород: Изд-во БГТУ, 2012. Т. 3. С. 284-291.
  32. Витяев Е.Е. Извлечение знаний из данных, компьютерное познание, модели процессов. Новосиб. гос. ун-т. Новосибирск, 2006.
  33. Жданов А.А. Адаптивные машины - неизбежное направление развития техники. Задачи и проблемы // XII Всероссийская научно-техническая конференция «Нейроинформатика-2010»: Лекции по нейроинформатике. - М.: НИЯУ МИФИ, 2010. с.162-211.
  34. Редько В.Г. Подходы к разработке компьютерных моделей сознания // Проблема сознания в философии и науке. Сборник под ред. Д.И. Дубровского. М.: «Канон+» РООИ «Реабилитация», 2009. С. 419-424.
  35. Bongard J., Zykov V., Lipson H. Resilient machines through continuous self-modeling // Science, 2006. Vol. 314. No. 5802. PP. 1118-1121.
  36. Непомнящих В.А. Адаптация к решению частных задач и «глобальные» цели в поведении животных // Нейроинформатика, 2012, том 6, № 1. URL:

Корягин Е.В.

<http://www.niisi.ru/iont/ni/Journal/V6/N1/Nepomnyashchikh.pdf>

37. Ay Nihat, Der Ralf, Prokopenko Mikhail. Information driven self-organization: The dynamical system approach to autonomous robot behavior // *Theory in Biosciences*, September 2012, Volume 131, Issue 3, PP. 161-179.
38. Martius Georg, Der Ralf, Ay Nihat. Information driven self-organization of complex robotic behaviors // *arXiv:1301.7473v1 [cs.RO]* Feb 2013.
39. Саттон Р.С., Барто Э.Г. Обучение с подкреплением. М.: Бином, 2011.

Статья поступила 11 марта 2013 г.

После доработки 25 апреля 2013 г.