

Российская Академия Наук  
Институт Системного Программирования

УДК 519.7

*На правах рукописи*

Устюжанин Андрей Евгеньевич

**РАЗРАБОТКА АССОЦИАТИВНОЙ ПАМЯТИ ДЛЯ СИСТЕМ  
АВТОНОМНОГО АДАПТИВНОГО УПРАВЛЕНИЯ НА ОСНОВЕ  
СИСТЕМ ДЕТЕРМИНИРОВАННОГО ХАОСА**

Специальность 05.13.11 –  
математическое и программное обеспечение  
вычислительных машин, комплексов и компьютерных сетей

**Автореферат**  
диссертации на соискание ученой степени  
кандидата физико-математических наук

Москва  
2007

Работа выполнена в Институте системного программирования РАН.

Научный руководитель: доктор физико-математических наук,  
профессор  
*Жданов Александр Аркадьевич*

Официальные оппоненты: доктор физико-математических наук,  
профессор  
*Зенкевич Станислав Леонидович*

доктор физико-математических наук,  
профессор  
*Дмитриев Александр Сергеевич*

Ведущая организация: Институт точной механики и  
вычислительной техники  
им. С.А. Лебедева РАН

Защита диссертации состоится «7» сентября 2007 г. на заседании диссертационного совета Д.002.087.01 при Институте системного программирования РАН по адресу:

109004, Москва, Б. Коммунистическая 25, Институт системного программирования РАН, конференц-зал.

С диссертацией можно ознакомиться в библиотеке Института системного программирования РАН.

Автореферат разослан «7» августа 2007 г.

Ученый секретарь  
диссертационного совета  
кандидат физ.-мат. наук

/Прохоров С.П./

## Общая характеристика работы

**Актуальность работы.** Развитие способов представления информации как средства эффективного управления искусственными и естественными системами не теряет своей актуальности с начальных этапов становления кибернетики. В наши дни управляющие системы в состоянии контролировать достаточно сложные объекты, масштабы которых варьируются от нанороботов до комплексов атомных электростанций. Значительные темпы развития демонстрирует в наши дни индустрия мобильных роботов, выполняющих разнообразные полезные функции в различных практических сферах. Примером, показавшим принципиальный скачок возможностей автономной робототехники, стал кросс непилотируемых автомобилей по аризонской пустыне в 2005 году. Другими примерами мобильных роботов, завоевавших отдельную нишу на современном рынке, являются роботы, используемые в силовых ведомствах (разведка, разминирование, работы во вредных для человека средах), мобильные автономные роботы-гиды для музеев и выставок, роботы-пылесосы.

Темпы развития индустрии мобильной робототехники в наши дни сопоставимы с темпами развития отрасли персональных компьютеров в начале 80-х годов: на рынке нет устоявшегося лидера и нет традиционных подходов, которые могли бы считаться безусловными авторитетами. Напротив, число проектных решений, архитектур роботов, предлагаемых исследовательскими и производственными коллективами, изобилует разнообразием, но все они, как правило, ориентированы на решение частных задач, отличающихся определенной узостью постановки. На сегодняшний день нет общепринятых методов разработки систем управления, подходящих для роботов, особенно если речь идет об адаптивных методах управления, которые пока практически не используются. Тем не менее, такие методы существуют, и одним из них является метод Автономного Адаптивного Управления (ААУ).

Метод ААУ потенциально обладает широким спектром применимости. Методология автономного адаптивного управления предполагает необходимость наличия в адаптивной систем управления определенного набора функциональных блоков, связанных друг с другом определенную в структуру, независимую от прикладной задачи. Способ реализации каждого блока может варьироваться в зависимости от характера задач, решаемых системой в целом. Среди задач, нашедших свое решение в рамках данной методологии,

присутствуют разнообразные по характеру примеры прототипов: от прогноза развития экономических систем до систем управления устройствами автомобиля и космическими аппаратами.

Для систем управления важную роль играет «глубина памяти» – объем сохраненной предыстории поведения управляемого объекта. Эффективность работы системы управления напрямую зависит от глубины памяти. В настоящей работе в рамках метода ААУ предлагается новый подход к построению модуля ассоциативной памяти для систем ААУ - памяти, которая способна идентифицировать протяженные во времени или в пространстве закономерности входных сигналов. В работах С.Н. Гринченко указывается на системы с ассоциативной памятью, как на системы, обладающие большей «глубиной памяти», поскольку они естественным образом приспособлены для работы с более сложными образами. В частности, системы ассоциативной памяти способны распознавать сохраненный образ протяженного во времени или в пространстве явления по небольшому фрагменту или по искаженному прообразу. Примеры, для которых наличие такой памяти необходимо, можно найти в любой области, требующей эффективного прогнозирования при управлении. Например, можно указать на следующие прикладные задачи:

- управление мобильным роботом в лабиринте,
- адаптивное управление подвеской автомобиля,
- управление коллективами агентов,
- управление искусственным спутником.

Известным примером, отражающим основные этапы обработки информации управляющими системами, является модель мобильного робота, помещенного в замкнутый лабиринт, заполненный повторяющимися препятствиями нескольких типов. Преимуществом этого примера является его широкая популярность не только среди исследователей в области робототехники, но и популярность среди исследователей из смежных областей. В качестве основной исследовательской модели выбран именно мобильный робот, поведенческие характеристики которого соотносятся с характеристиками живых существ.

**Научная новизна.** Научная новизна выносимых на защиту результатов работы состоит в следующем.

1. Предложен и разработан метод построения ассоциативной памяти для систем управления на основе хаотических процессоров. Создание и реализация такой памяти является развитием теории ААУ, позволяющим системам

управления оперировать образами качественно более сложных объектов – протяженных во времени и пространстве наблюдаемых явлений, представимых в виде последовательности многомерных векторов.

2. Разработана система управления для мобильного робота с подсистемой ассоциативной памяти, которая позволяет оперировать качественно новыми объектами – образами протяженных во времени и пространстве наблюдаемых явлений, повышая тем самым эффективность управления.

3. На основе разработанной технологии создана прикладная система распознавания для адаптивной системы управления лучезапястным суставом протеза руки. Такая ассоциативная адаптивная система распознавания позволяет автоматически научиться распознавать типовое движение руки по начальной фазе его выполнения, и передавать этот результат адаптивной подсистеме принятия решений для поиска оптимального завершения этого движения.

Одну из ключевых ролей в построении систем автономного адаптивного управления играет подсистема формирования и распознавания образов (ФРО). На сегодняшний день технологии, использованные для реализации этой подсистемы, позволяли работать лишь с образами непродолжительных во времени явлений (например, временных рядов, наблюдаемых в течение не более 5 тактов работы системы). «Глубина» такой памяти невелика, и небольшой объем запоминаемой предыстории обусловлен тем, что основной технологией работы системы ФРО является использование системы нейроподобных элементов. Возможности отслеживать динамику и длинные последовательности наблюдаемых явлений и процессов посредством нейроподобных структур наталкиваются на серьезные ограничения, вызванные тем, что пока еще не развита технология использования нейроподобных сетей для работы с образами протяженных пространственно-временных объектов. Использование предложенного в данной диссертационной работе подхода к построению ассоциативной памяти на основе хаотических процессоров позволит значительно расширить границы применимости методологии ААУ, охватив, тем самым, более сложные с технологической точки зрения области.

Предлагаемый подход представляет ценность для самого широкого спектра задач, требующих эффективного анализа регулярных структур во временных последовательностях данных. Например, в задачах разработки

систем навигации и локализации мобильных роботов существующие подходы требуют существенных ограничений для свойств лабиринта - заранее известной карты лабиринта или запрета на появление движущихся препятствий и т.п. условий. Подход, предлагаемый в данной работе, позволяет существенно ослабить такие ограничения.

**Практическая значимость работы.** Основным практическим результатом данной работы является разработанный способ построения системы управления, включающей в свой состав подсистему памяти на основе хаотических процессоров. Система памяти, организованная предложенным способом, позволяет эффективно различать и распознавать протяженные во времени и пространстве структуры. Например, такую память можно использовать в системах управления мобильными роботами, которые должны научиться распознавать препятствия лабиринта с высокой эффективностью (малым числом ошибок и за короткое время). Для мобильного робота ассоциативность памяти оказывается критическим свойством при решении целого ряда практических задач. Например, такими задачами являются распознавание протяженных в пространстве препятствий, задачи локализации (определения своего расположения) и навигации (прокладывание пути) в лабиринте.

Другим примером объектов, в которых можно использовать разработанную систему памяти, могут служить адаптивные манипуляторы и протезы. Последние могут позволить человеку повысить точность и качество выполняемых с помощью протеза операций. Актуальность разработки автоматизированных протезов конечностей очень высокая и не уменьшается со временем в связи с большим количеством травм бытового, производственного и военного характера, а также из-за большого числа ампутаций в связи с сосудистыми заболеваниями.

**Методы исследования.** В качестве основных методов исследования в данной работе использовались положения из теории метода автономного адаптивного управления, теории детерминированного хаоса, математической логики, теории графов, теории множеств, анализ временных рядов, методы математического моделирования.

**Апробация работы и публикации.** Результаты работы докладывались на международных конференциях «Интеллектуальные системы» (Дивноморское, 2003 г. и 2004 г.), «Нейроинформатика-2006» (Москва, 2006 г.), 44-й и 48-й научных конференциях МФТИ (Долгопрудный, 2001 г., 2005 г.).

По теме диссертации опубликованы 13 работ, раскрывающих основные научные результаты диссертации.

**Структура и объем работы.** Диссертация состоит из введения, пяти глав, заключения, списка литературы и одного приложения. Список литературы включает 94 названия. Основной текст диссертации занимает 126 страниц.

## Содержание работы

Во **Введении** обосновываются цели и задачи данной работы, её актуальность, научная новизна и практическая значимость. Приводится краткий обзор работы.

В **главе 1** рассматриваются результаты исследований, близких к теме настоящей диссертации, а также методы и подходы, использованные при написании работы. А именно, представлены:

- Методология Автономного Адаптивного Управления;
- Информационные потоки в динамических системах. Хаотические процессоры;
- Меры близости многомерных последовательностей.

Методология «Автономного адаптивного управления» (Жданов А.А., 1999) – подход, разрабатываемый в Институте системного программирования РАН и нацеленный на проектирование и разработку систем управления, обладающих свойством адаптации к изменениям условий окружающей среды. Предлагаемый в данной работе метод построения ассоциативной памяти позволяет существенно повысить эффективность подсистем распознавания образов и памяти образов в системах ААУ. Центральным понятием системы распознавания является понятие *образа*. Используется понятие образа, принятого в алгебраических системах, где под образом понимается результат отображения из множества прообразов в множество образов. Множеством прообразов в случае систем управления может быть множество объектов, сцен или процессов, наблюдаемых управляющей системой (УС) в окружающей ее среде посредством сенсоров. Множество образов расположено в памяти УС (называемой *памятью образов*) и представлено множеством идентификаторов образов. Рассматриваемое отображение реализует некоторое определенное правило, накладывающее ограничения на структуру и статистические свойства прообразов, которым разрешается иметь свои отображения в памяти образов,

т.е. собственно образы. Память образов формируется в процессе автоматического обучения системы распознавания.

Под *прообразом*, анализируемым разработанной системой управления, понимается последовательность многомерных векторов, заданных в некотором  $N$ -мерном пространстве. Работа датчиков, отображающих информацию о наблюдаемых объектах и процессах в форму входных векторов, во многих случаях описывается тривиальными правилами. Например, для мобильного робота, обладающего 10-ю ультразвуковыми датчиками расстояния, поток информации от этих датчиков можно представить как последовательность 10-мерных векторов. Характерные длины таких последовательностей определяются спецификой конкретной предметной области.

Построение памяти образов базируется на принципе так называемых «хаотических процессоров» (Дмитриев А.С., 1997) – систем, работающих с потоками информации, и обладающими следующими свойствами:

- итеративность,
- работа с последовательностями данных,
- адресация элементов по содержанию.

«Хаотический процессор» определяется как информационная система, реализующая отображение  $F$

$$X_{i+1} = F(X_i, a),$$

где  $X_i$  –  $n$ -мерный вектор,  $a$  — вектор параметров, размерность которого определяется спецификой задачи. Итеративно применяя отображение  $F$  к начальному вектору  $X_0$ , получаем последовательность векторов  $X_1, X_2, \dots, X_m$ . Такую последовательность называют *траекторией системы в векторном пространстве*. Множество последовательностей  $n$ -мерных векторов длиной не большей  $t$  будем обозначать  $V_m^n$ .

При определенных условиях, среди возможных траекторий системы существуют особенные последовательности – *аттракторы*, к которым стягиваются любые траектории, попавшие в некоторую окрестность аттрактора. Окрестность притяжения аттрактора – область, откуда последовательность  $\{X_i\}$  стягивается к аттрактору, называется *бассейном аттрактора*. В работах Дмитриева А.С. рассматривается способ построения хаотических процессоров, реализующих преобразование  $F$  и функции хранения цепочек одно- и двумерных данных на базе аттракторов этих отображений. Обсуждаются



требования, которым должны соответствовать хаотические процессоры, чтобы их использование было допустимо при работе с многомерными данными.

В заключительной части главы 1 рассматриваются численные меры близости, используемые для сравнения многомерных последовательностей. Были отобраны следующие часто используемые меры близости:

- Евклидова мера близости – сумма квадратов поэлементных разностей двух сравниваемых последовательностей,
- Мера наибольшей общей подпоследовательности (Longest Common Subsequence – LCSS), по которой последовательности тем ближе, чем больше максимальная длина общей подпоследовательности двух сравниваемых последовательностей,
- Мера динамического растяжения по шкале времени (Dynamic Time Warping – DTW), для вычисления которой поэлементно сравниваются две последовательности без учета частот дискретизаций этих последовательностей.

В главе 2 описывается подход к построению ассоциативной памяти, позволяющей запоминать протяженные во времени и пространстве цепочки данных. Здесь и далее рассматриваются два множества: множество ключей  $\mathbf{K}$ , каждый из элементов которого принадлежит  $V_m^n$ , и множество идентификаторов последовательностей  $\mathbf{O}$ , каждый из элементов которого является целым числом. Под *ассоциативной памятью* понимается:

**Определение 1.** *Ассоциативной* будем называть память, обеспечивающую хранение пар (ключ, идентификатор) и поиск идентификаторов, ключи которых совпадают с поисковым запросом. Другими словами память обеспечивает сюръективное отображение между множеством ключей  $\mathbf{K}$  и множеством значений идентификаторов  $\mathbf{O}$ , упорядоченных по возрастанию от 0 до  $N-1$  (где  $N$  – мощность множества значений  $\mathbf{O}$ ). Извлечение идентификатора из памяти осуществляется по запросу – последовательности  $Q \in V_m^n$ . т.е. память реализует операции:

- сохранение пары (ключ  $K_i$ , идентификатор  $O_j$ ),  $K_i \in \mathbf{K}$ ,  $O_j \in \mathbf{O}$ ;
- поиска идентификатора сохраненных ключей  $K_i$ , по запросу  $\{Q_{ij}\} \in V_m^n$ ;

При работе с ключами, представленными в виде последовательности векторов, получаемых от физических источников (например, сенсоров мобильного робота), важным оказывается свойство устойчивости ассоциативной памяти.

**Определение 2.** *Устойчивой ассоциативной памятью* назовем такую ассоциативную память, которая в дополнение к операциям обычной ассоциативной памяти позволяет находить идентификаторы ключей по неточным запросам:

- поиск идентификаторов  $O_j$  сохраненных ключей  $K_i$ , ключи которых *похожи* на запрос  $\{Q_i\} \in V_m^n$ ;
- поиск идентификаторов  $O_j$  сохраненных ключей  $K_i$ , которые включают последовательности, *похожие* на запрос  $\{Q_i\}$ .

Похожесть определяется значением некоторой вычислимой функцией от двух аргументов – запроса и ключа, возвращающей целое положительное число. Большие значения соответствуют большей мере похожести ее аргументов. Далее под ассоциативной памятью будем понимать устойчивую ассоциативную память.

Далее рассматриваются подсистемы ААУ, для которых использование ассоциативной памяти оказывается оправданным. Сравнивая функциональность подсистемы ААУ «ФРО» и «Базы Знаний» с функциональностью ассоциативной памяти, приходим к выводу, что именно для этих подсистем использование ассоциативной памяти может дать значительные преимущества.

Затем рассматривается вопрос определения количественной меры, которую можно использовать для оценки близости двух преобразов. Такая мера должна быть устойчива к различным искажениям, т.е. при некоторых величинах искажений из близости последовательностей  $A$  и  $B$  можно судить о близости последовательностей  $A'$  и  $B'$ , полученных из  $A$  и  $B$  внесением искажений. Такими искажениями могут быть, например, изменения длины, изменение частот дискретизации, всплески и фрагментарность (частичная потеря данных).

В качестве меры, устойчивой к подобным искажениям, вводится мера на основе минимальных ограничивающих регионов.

*Минимальным ограничивающим параллелепипедом* или *минимальным ограничивающим регионом* (МОП или MBR – minimum bounding rectangle) для последовательности  $\{S_i\}$ , все элементы которой принадлежат  $\mathcal{H}^n$  ( $S_i \in \mathcal{H}^n$ ), будем называть  $n$ -мерный параллелепипед  $R = \text{MBR}(\{S_i\})$  минимального объема со сторонами, параллельными осям координат, который целиком содержит точки этой последовательности. *Расширением параллелепипеда*  $R$  на величину  $\delta$  является параллелепипед  $R'$ , такой что, центр  $R'$  совпадает с

центром параллелепипеда  $R$ , а координаты вершин  $R'$  отличаются от координат соответствующих вершин  $R$  на величину  $\delta$  вдоль каждой из осей в направлении от центра  $R$ .

Алгоритм вычисления MBR меры следующий. Пусть даны последовательность  $\{S_{ij}\}$  и запрос  $\{Q_{ij}\}$ , для которых необходимо вычислить меру близости. Для этого разобьем  $\{S_{ij}\}$  на подпоследовательности, и для каждой из подпоследовательностей построим минимальный ограничивающий параллелепипед  $R_i$ . Каждый параллелепипед увеличим на  $\delta$  (для каждого параллелепипеда построим расширение на величину  $\delta$ ). Таким образом, последовательности  $\{S_{ij}\}$  ставим в соответствие последовательность  $\{R_{ij}\}$ . Заметим, что между любыми двумя  $R_i, R_j \in \{R_{ij}\}$  установлено отношения *непосредственного следования* – для них можно определить, является ли  $R_i$  непосредственным предшественником  $R_j$  или нет. Тогда процедура вычисления меры близости последовательностей  $\{Q_{ij}\}$  и  $\{S_{ij}\}$  выглядит следующим образом:

- 1)  $v = v_{min}$  (минимальное значение, которое нужно, чтобы операция деления на  $v$  была всегда допустима)
- 2) по последовательности  $\{S_{ij}\}$  строим  $\{R_{ij}\}$ ,
- 3) последовательно для каждого элемента  $Q_i \in \{Q_{ij}\}$  проверяем:
  - a) принадлежит ли  $Q_i$  какому-либо  $R_i \in \{R_{ij}\}$ , и если принадлежит, то запоминаем индексы этих регионов (индексы текущего шага). Если принадлежность найдена, то увеличиваем значение  $v$  на величину  $\alpha_1$ ;
  - b) сравниваем индексы текущего шага с индексами прошлого шага. Если оказывается, что индексы текущего следуют за индексами прошлого, то увеличиваем  $v$  на величину  $\alpha_2$ .
- 4) Величина меры близости вычисляется как  $1/v$ .

Величина  $\delta$  характеризует устойчивость алгоритма к пространственным искажениям. Параметрам алгоритма  $\alpha_1$  и  $\alpha_2$  были присвоены значения 1 и 2 соответственно.

Описанный способ сравнения пространственно-временных последовательностей удовлетворяет выдвинутым требованиям и оказывается лучше аналогов (DTW и LCSS). Этот вывод подкрепляется численными

экспериментами. В качестве тестовых данных для численных экспериментов использовались данные из следующих источников: 1) данные, полученные в ходе работы над совместным проектом Института системного программирования и Санкт-Петербургского научно-практического центра медико-социальной экспертизы, протезирования и реабилитации инвалидов им. Г.А.Альбрехта; 2) набор слов австралийского языка глухонемых, который является стандартом де-факто для проверки работы различных алгоритмов с многомерными данными.

Из анализа результатов вычислительных экспериментов были сделаны следующие выводы:

- 1) в целом выбранные меры ведут себя приблизительно одинаково хорошо, когда длина запроса близка к длине последовательности;
- 2) мера LCSS ведет себя тем хуже, чем больше соотношение длины последовательности к длине запроса;
- 3) мера MBR ведет себя в целом не хуже, чем евклидова мера;
- 4) евклидова мера ведет себя хуже, чем мера MBR при увеличении параметра соотношения длины последовательности к длине запроса;
- 5) для выбранных задач мера MBR является наиболее выразительной среди анализируемых мер, т.е. процент ошибок классификации на искаженных данных у нее ниже, чем процент ошибок, полученных с использованием других мер.

Далее предлагается следующий способ построения ассоциативной памяти на основе хаотического процессора. Схема хранения данных основана на подходе аппроксимации бассейна аттрактора посредством многомерных геометрических примитивов и хранении этих элементов в многомерной индексной структуре. Пример построения МОП-ов разбиения последовательности приведен на Рис.1.

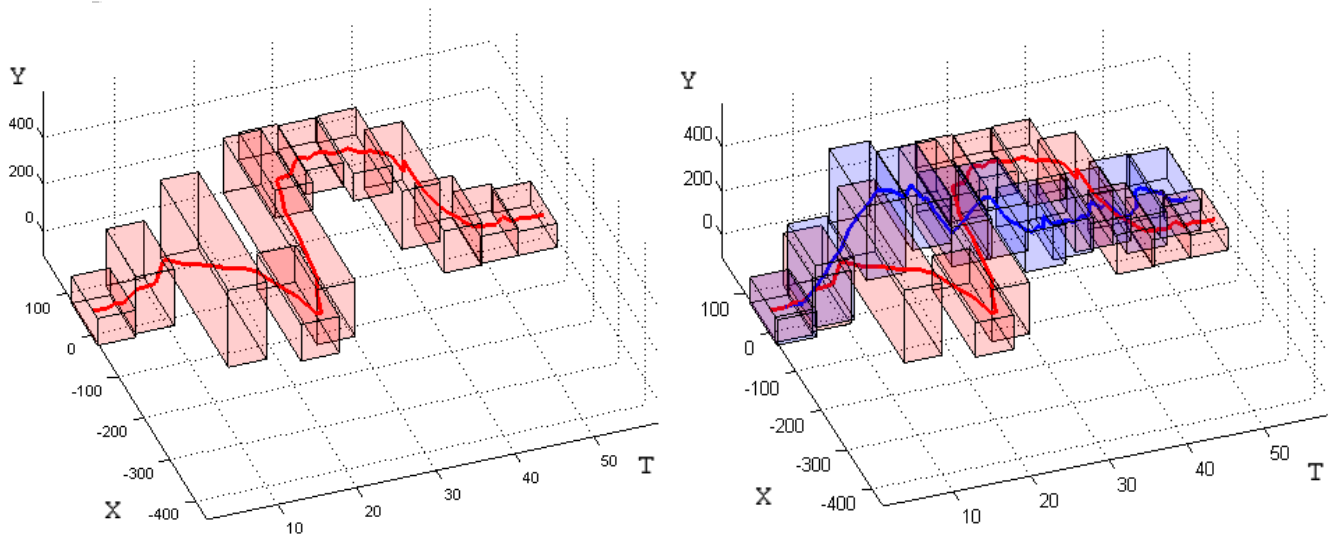


Рис. 1. Пример разбиения многомерной последовательности и построения ограничивающих регионов

Процедура построения ассоциативной памяти выглядит следующим образом. Пусть имеется последовательность  $\{X_i\}$  и ее идентификатор  $O_i$ , которые необходимо запомнить. Построим такое отображение  $F$ , для которого  $\{X_i\}$  будет являться аттрактором.

В работе формулируется и доказывается ряд утверждений, о том, что

- отображение  $F$  обладает хаотическим режимом,
- аттракторами этого отображения являются  $\{X_i\}$
- бассейн аттрактора включает подмножество объединения  $\{R_i\}$

На Рис.2 представлены примеры бассейнов притяжения траекторий-аттракторов. Система, реализующая отображение  $F$  является хаотическим процессором, который в последующих главах будет использован для построения систем управления.

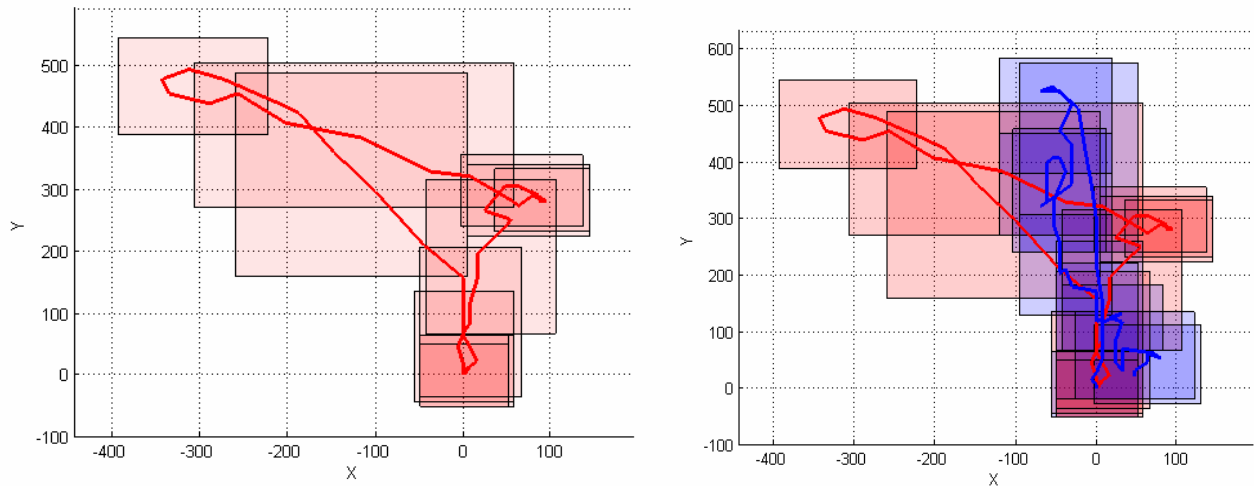


Рис. 2. Бассейны притяжения аттракторов

Для реализации отображения  $F$  необходимо каждой сохраняемой последовательности  $\{X_{ij}\}$  сопоставить последовательность МОП-ов  $\{R_{ij}\}$ . Каждому МОП-у поставим в соответствие следующие данные:

- идентификатор всей последовательности  $O_j$ ,
- индекс МОП-а в последовательности  $\{R_{ij}\}$ ,
- центр следующего МОП-а (или первого, если текущий МОП является последним в  $\{R_{ij}\}$ ),
- признак конца последовательности, который равен 1, если текущий МОП является последним в последовательности  $\{R_{ij}\}$ , и 0 в противном случае.

Последовательность параллелепипедов  $\{R_{ij}\}$  и связанных с ними данных сохраняется в структуре многомерного индекса – R-дереве с приоритетами.

Приводится описание процедуры поиска последовательности по запросу  $Q$ . Показывается, что алгоритм поиска в предложенной ассоциативной памяти сводится к вычислению MBR меры близости запроса и сохраненных последовательностей. На этом основании делается вывод, что построенная ассоциативная память обладает свойствами MBR меры, в частности, является устойчивой к искажениям.

На основании оценки сложности операции поиска по R-дереву показывается, что сложность операции поиска при использовании предложенного подхода составляет  $O(m(\frac{N}{B})^{1-\frac{1}{n}})$ , где  $m$  – длина запроса  $Q$ ,  $N$  число параллелепипедов размерности  $n$ , сохраненных в дереве,  $B$  – размер блока дисковой памяти, используемого для хранения дерева на постоянном

носителе. Эмпирические оценки эффективности работы предложенной памяти приводятся в 5-ой главе.

Описанная системы ассоциативной памяти может быть использована при разработке системы управления мобильным роботом. Построение такой системы управления приводится в последующих главах.

**В главе 3** приводятся основные примеры, которые демонстрируют возможности применимости ассоциативной памяти в мобильных роботах. Описание примеров включает перечень основных объектов: мобильного робота, системы управления, сенсоров, актуаторов, окружающей среды, а также описание основных критериев успешности его работы.

Программная модель робота, рассматриваемая в этом примере, соответствует современной модели мобильного робота Pioneer II DX. Такой робот оснащен следующими датчиками:

- лазерные датчики расстояния,
- датчик цвета,
- датчики столкновения,
- датчик направления (компас).

Основным датчиком, использовавшимся для работы системы управления, был лазерный датчик, определяющий расстояние до ближайшего к роботу препятствия. Угол обзора датчика составляет 180 градусов, который разбивается на 180 равных угловых диапазонов. Информация о расстоянии до ближайшего препятствия возвращается в виде вектора размерности 180, каждая компонента которого – расстояние до препятствия в соответствующем угловом диапазоне. На следующем рисунке представлена последовательность значений таких векторов от времени, полученная при обходе препятствия:

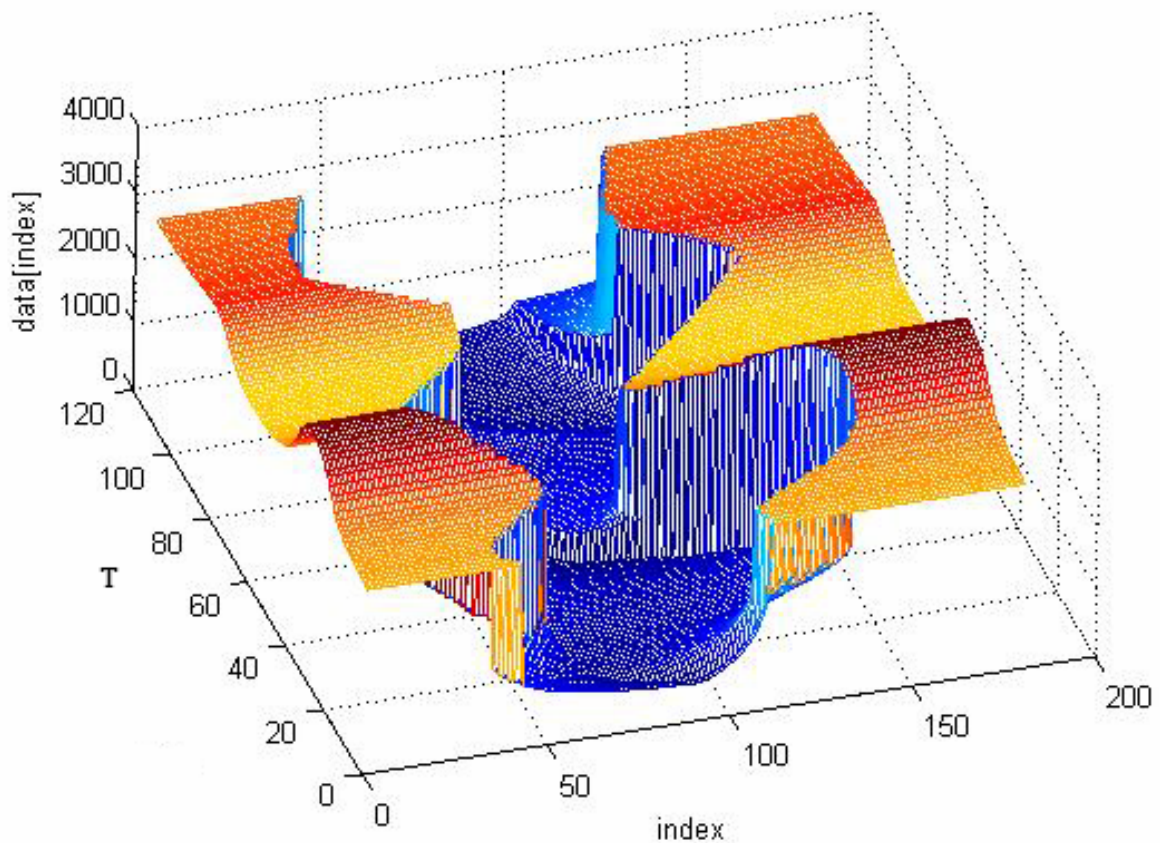


Рис. 3. Последовательность значений, полученных от лазерных сенсоров.

Для облегчения работы с такими с такими векторами были использованы дискретные вейвлет-преобразования по базису вейвлетов Хаара. Использование такого преобразования позволило уменьшить абберации и размерность входных сигналов. Характерная длина преобразов (длина последовательности векторов) препятствий варьировалась от 50 до 100 элементов. Множество идентификаторов последовательностей состоит из чисел от 0 до  $(T-1)$ , где  $T$  – число типов препятствий.

Управляющая система (УС) такого мобильного робота была спроектирована с использованием методологии ААУ, схема которой представлена на Рис.4.



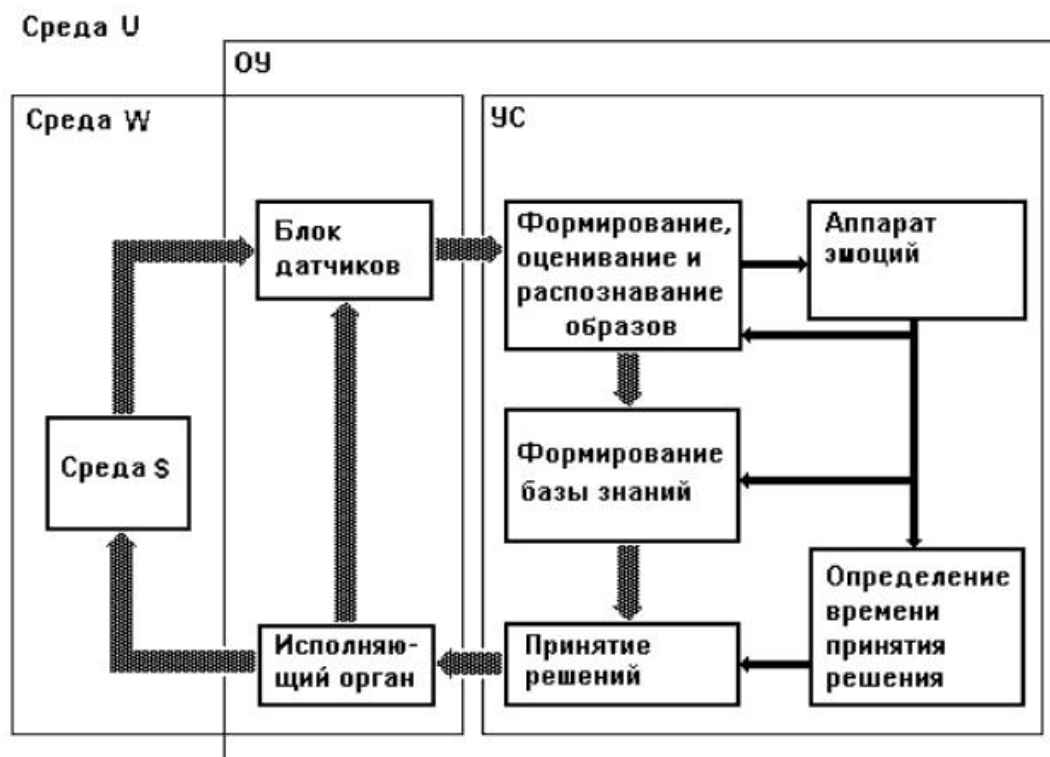


Рис. 4. структура управляющей системы (УС) согласно методологии ААУ. ОУ – объект управления, Среда S, Среда W и Среда U – обозначения внешней среды с различных точек зрения (Жданов А.А., 1999).

Система «Формирования и распознавания образов» такой системы управления построена с использованием системы ассоциативной памяти на основе систем детерминированного хаоса.

Приведены три типа задач, с которыми сталкивается мобильный робот:

- анализ и распознавание объектов по искаженным входным данным,
- распознавание препятствия по фрагментарной информации,
- прогнозирование потока информации по имеющимся данным.

Первый тип задач связан с распознаванием препятствий различной формы, которые достаточно похожи друг на друга, чтобы быть неразличимыми при разовом осмотре.

Для определения типа препятствия на основании показаний лазерных датчиков мобильному роботу необходимо пройти определенный путь вдоль препятствия – осмотреть его, чтобы сопоставить последовательность показаний датчиков с уже запомненными последовательностями,

соответствующими известным препятствиям. Такую особенность структуры препятствий будем называть условием *односторонней неразличимости*.

В качестве типов препятствий были выбраны следующие три набора. 1-й набор препятствий содержит два класса препятствий, каждый из которых представляет собой искаженную трапецию. Препятствия 2-го набора выбраны из теста на уровень интеллекта (IQ), в котором необходимо было определить различия между препятствиями. Препятствия 3-го набора были взяты с карты мира и представляют собой очертания Багамских островов.

Сформулированы следующие типы задач, которые ставились перед мобильным роботом и по результатам выполнения которых оценивалась эффективность системы управления

Первая задача, названная «Слон», должна показать возможности системы управления по распознаванию препятствия, несмотря на преднамеренно создаваемые сложности распознавания. Один тип сложностей связан с тем, что масштаб препятствий может варьироваться в заданном диапазоне, т.е. размер препятствий при управлении роботом отличается от размера препятствия, который они имели при обучении управляющей системы. Второй тип сложностей состоит в увеличении скорости перемещения робота в режиме управления. Эффективность работы системы управления определяется как соотношение корректно определенных типов препятствий к общему числу попыток распознавания.

Эффективность работы системы управления и метрик различных типов рассматриваются в зависимости от степени создаваемых сложностей.

Вторая задача, названная «Сталкер», состоит в «сборе» как можно большего числа полезных препятствий. В лабиринт, о котором системе управления известно только то, что в нем присутствуют препятствия двух типов («хорошие» и «плохие»), помещается мобильный робот. Ни очертаний препятствий, ни их взаимное расположение системе управления не известны. Система управления может отличить препятствия по маячку, установленному на каждом препятствии. Но для того, чтобы увидеть маяк, необходимо почти полностью обойти это препятствие. Робот должен «научиться» отличать «хорошие» препятствия от «плохих» по выявленным им характерным особенностям очертаний. Перемещаясь по лабиринту, робот должен коснуться как можно большего числа «хороших» препятствий, избегая столкновений с «плохими» препятствиями. При каждом касании, в зависимости от типа

препятствия, робот приобретает или теряет баллы. Число набранных баллов за отведенное время является критерием успешности работы системы управления.

Лабиринт третьей задачи, условно названной «Колумб», представляет собой полосу препятствий, которую необходимо преодолеть роботу. Изначально мобильный робот располагается у нижней кромки полосы, и ему необходимо добраться до верхней стороны за минимальное время. В этой задаче роботу не требуется касаться препятствия, но препятствия составляют таким образом, что обход препятствия с одной стороны может занять существенно меньше времени, чем обход с другой стороны. То есть при быстром и правильном распознавании типа препятствия робот может изменить направление обхода и сэкономить время. Показателем эффективности работы системы управления является зависимость вертикальной координаты положения робота от времени.

Количественные характеристики поведения системы управления в имитационных экспериментах использованы для сравнения предлагаемой системы памяти с ее альтернативами в 5-ой главе.

В **главе 4** приводится описание практической реализации ассоциативной памяти, системы имитационного моделирования мобильных роботов и системы управления мобильными роботами. В описание включены основные компоненты, интерфейсы и протоколы, разработанные в ходе работ. Также в этой главе приводится краткое описание библиотек сторонних производителей и программного обеспечения, используемого в ходе работ. В качестве платформы для создания системы имитационного моделирования примеров, описанных в 3-ей главе, была выбрана система Player/Stage. Эта система ориентирована на исследователей в области робототехники и позволяет разрабатывать системы управления для различных мобильных роботов (Pioneer, Nerega, MIONIX, и т.д.), предоставляя разработчику удобные программные интерфейсы взаимодействия с основными датчиками и актуаторами роботов. Помимо богатого набора устройств, поддерживаемыми Player-ом, на ее выбор оказали влияние следующие немаловажные факторы. Во-первых, таким фактором стала расширяемость платформы – при необходимости исследователь может самостоятельно разработать новый тип датчика и драйвер, обеспечивающий взаимодействия с программной системой управления. Во-вторых, таким фактором является архитектура системы, позволяющая без изменений в логике программы перейти от управления программной моделью робота к управлению реальным роботом.

Для целей исследования все эксперименты проводились с использованием имитационного сервера Stage. Для моделирования использовались два компьютера: имитационный сервер и клиент. На сервер имитации работала система Player и система имитации Stage под управлением операционной системы Linux. Компьютер клиентского приложения работал под операционной системой MS Windows 2003 Professional, на нем выполнялась система управления мобильным роботом, написанная на языке Java и взаимодействующая с сервером имитации посредством библиотеки Java-client.

В главе подробно описывается набор программных интерфейсов, позволивших абстрагироваться от особенностей реализации используемой системы управления и построить имитационные эксперименты задач главы 3 для различных алгоритмов управления и различных реализаций ассоциативной памяти. Общим интерфейсом для всех систем управления является интерфейс IStageExplorer, в котором содержится метод doExec, реализующий один шаг обработки входной информации. Кроме того, каждый из классов систем управления является реализацией интерфейса IDebuggable. Этот интерфейс содержит необходимые методы для сбора статистики работы системы и вычисления эффективности ее работы. В каждом случае рейтинг вычисляется, исходя из критериев, сформулированных в соответствующей задаче.

Затем даются описание правил обработки входных сигналов системой управления и правила взаимодействия системы управления с актуаторами мобильного робота. Приводится способ интеграции данной программной модели в архитектуру ААУ.

Таким образом, созданная инфраструктура позволяет загрузить разработанные системы управления мобильным роботом и сравнить их эффективность, исходя из условий, описанных в 3-ей главе.

В главе 5 приводятся результаты численных имитационных экспериментов, которые демонстрируют практические аспекты применения ассоциативной памяти для систем управления.

Задача оценки эффективности работы хаотического процессора разбита на три подзадачи, имеющих целью расчет следующих показателей:

- объема занимаемой памяти,
- времени сохранения последовательностей,
- времени поиска последовательности.

Имитационное моделирование имеет две цели: а) анализ эффективности модуля ассоциативной памяти на основе хаотического процессора и сравнение ее характеристик с характеристиками систем ассоциативной памяти, построенной на альтернативных принципах и б) анализ эффективности системы управления мобильным роботом, разработанной согласно методологии ААУ. В реализации системы ААУ модуль формирования и распознавания образов реализован с использованием предлагаемого механизма работы ассоциативной памяти.

В качестве данных для проведения вычислений взяты последовательности австралийского языка глухонемых, рассмотренные во 2-ой главе.

Результаты проведенных экспериментов показывают преимущества использования памяти на основе хаотического процессора. Во-первых, хаотический процессор позволяет реализовать выразительную функцию меры близости (MBR меру), предложенную во 2-ой главе. Во-вторых, время поиска сохраненных идентификаторов в такой реализации значительно (до 6 раз) меньше времени поиска в альтернативных реализациях. Кроме того, время поиска посредством хаотического процессора гораздо меньше зависит от объема сохраненной информации.

Затем следует исследование эффективности работы ассоциативной памяти в составе системы управления мобильным роботом в задачах, описанных в главе 3. На рис. 5 представлены зависимости эффективности работы систем управления для первой из предложенных задач.

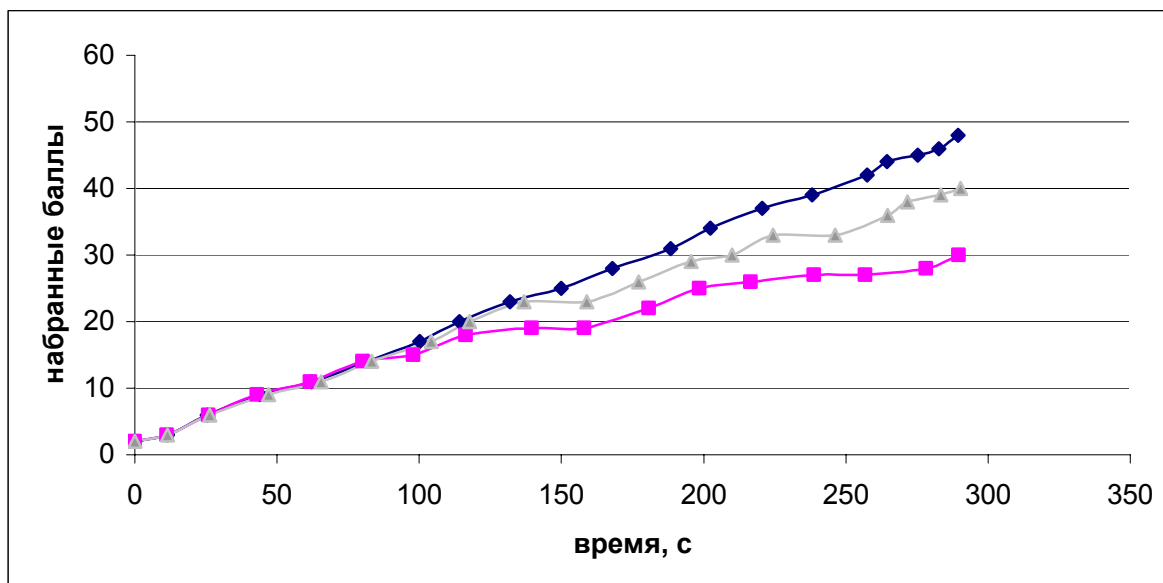


Рис. 5. Зависимость результативности системы управления мобильным роботом от времени при скорости движения робота  $1000\text{c}^{-1}$ . Квадратные значки – система управления с ФРО на основе евклидовой меры, треугольные значки – на основе меры LCSS, ромбовидные – на основе меры MBR.

В ходе испытаний, при движении робота со скоростью в 2 раза превышающей скорость его движения при обучении, результативность системы управления с ассоциативной памятью на основе хаотического процессора превысила на 65% результативность системы управления с ассоциативной памятью на основе евклидовой меры и на 25% - с ассоциативной памятью на основе метода LCSS.

Аналогичные качественные результаты получены при изменении линейного размера препятствий. Во всех рассмотренных примерах система управления на основе евклидовой меры демонстрировала худшие результаты, на основе хаотического процессора – наилучшие, а результативность системы управления на основе LCSS находилась на промежуточном уровне.

Проведенное сравнение показывает успешность распознавания роботами заданных препятствий.

Количественное сравнение эффективности различных систем управления приводится для задач «Сталкер» и «Колумб». Зависимости рейтинга,

набранного различными системами управления, от времени представлены на рис. 6, 7.

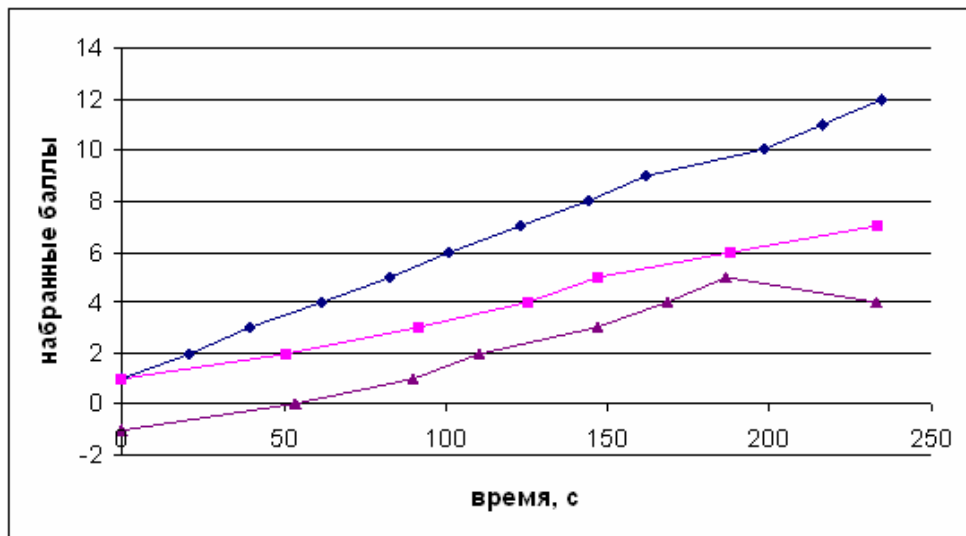


Рис. 6. Зависимость рейтинга систем управления мобильным роботом от времени в задаче «Сталкер». Треугольные значки соответствуют системе управления со случайным механизмом принятия решений (ExplorerRandom), квадратные – с выбором левого пути обхода (ExplorerLeft), а ромбовидные – системе управления ААУ.

Сравнение эффективности работы системы управления для задачи «Сталкер» проходило с двумя другими типами систем управления: 1) ExplorerRandom – система управления принимает случайное решение по касанию или избеганию препятствия; 2) ExplorerLeft – система управления обходит препятствие до тех пор, пока не натолкнется на маяк и тип препятствия не станет известен точно.

За время эксперимента около 5 минут показатели системы ААУ превысили показатели ExplorerRandom в 2 раза, а показатели ExplorerLeft в 1,7 раз. Сравнение зависимостей говорит об успешности использования подсистемы ассоциативной памяти для системы управления.

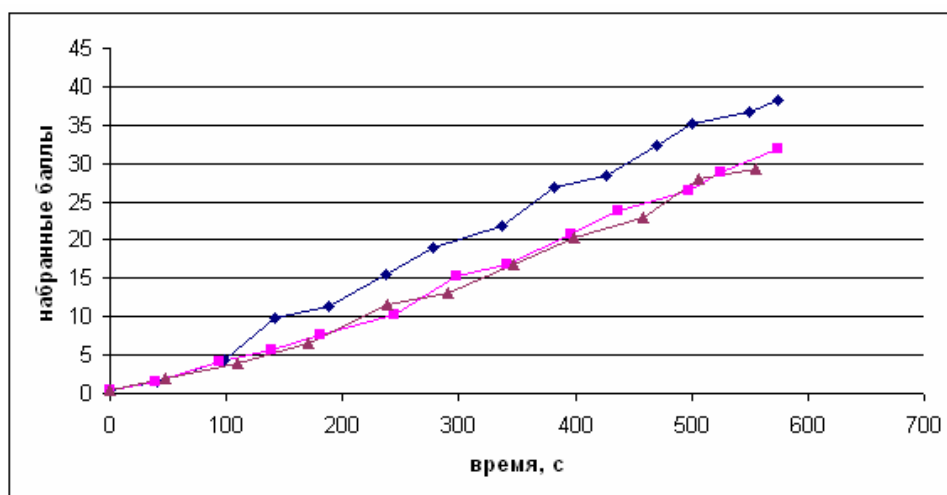


Рис. 7. Зависимость рейтинга систем управления мобильным роботом от времени в задаче «Колумб». Треугольные значки соответствуют системе управления со случайным механизмом принятия решений, квадратные – с выбором левого пути обхода, а ромбовидные – системе управления ААУ.

Сравнение эффективности работы системы управления для задачи «Колумб» проходило с двумя другими типами систем управления: 1) ExplorerRandom – система управления принимает случайное решение по обходу препятствий слева или справа; 2) ExplorerLeft – система управления принимает одно и то же решение обхода препятствия слева.

Сравнение зависимостей показывает успешность использования подсистемы ассоциативной памяти для системы управления в задаче «Колумб». За время эксперимента показатели системы ААУ превысили показатели ExplorerRandom в 1,27 раз а показания ExplorerLeft в 1,2 раза.

Качественный анализ полученных сравнений говорит о применимости и целесообразности применения ассоциативной памяти, организованной согласно подходу, предложенному в данной работе.

В **Заключении** диссертации приводятся выводы о применимости предлагаемого подхода, и перечисляются полученные результаты работы.



## Основные выводы и результаты

В работе были получены следующие основные результаты:

1. Предложена и разработана организация системы ассоциативной памяти на основе хаотического процессора.
2. Разработан критерий для сравнения пространственно-временных последовательностей, основанный на использовании меры минимальных ограничивающих регионов (Minimum Bounding Region (MBR) меры). В соответствии с предложенным критерием разработан алгоритм эффективного поиска близких пространственно-временных последовательностей, основанный на использовании структур пространственного индексирования. Разработаны оценки ресурсоемкости и производительности для сравнения предложенного подхода с альтернативными подходами.
3. Разработана система управления мобильным роботом с ассоциативной памятью на основе хаотического процессора. Для обеспечения возможности сравнения полученных результатов реализованы также системы управления мобильным роботом, базирующиеся на альтернативных способах управления.
4. Разработаны критерии для оценки применимости ассоциативной памяти на примере задач управления мобильным роботом для трех случаев лабиринтов и нескольких видов препятствий. Проведен количественный анализ эффективности предложенного подхода в соответствии с разработанными критериями на примерах нескольких тестовых прикладных задач.

## Публикации

По теме диссертации опубликованы следующие работы:

1. Жданов А.А., Устюжанин А.Е., Возможности использования технологии детерминированного хаоса в системах автономного адаптивного управления. // Тр. Института системного программирования: Том. 2. М.: ИСП РАН, 2001. с. 141-179.

2. Устюжанин А.Е., Об использовании многомерного представления данных для анализа документарных источников // XLIV научная конференция МФТИ, 2001.
3. Жданов А.А., Устюжанин А.Е., Использование технологий детерминированного хаоса для построения ассоциативных баз знаний систем автономного адаптивного управления. // Тр. Конференции IEEE AIS'02: М.: Физматлит, 2002. С. 121-127.
4. Жданов А.А., Устюжанин А.Е., Использование технологий детерминированного хаоса для реализации памяти систем автономного адаптивного управления // МФТИ-2002.
5. Устюжанин А.Е., Метаповедение как способ динамической конфигурации поведения агента. // IEEE AIS-2003, с 57-65.
6. Бородин М.В., Устюжанин А.Е., О возможностях динамического изменения поведения агентов на базе многоагентной платформы JADE // МФТИ, XLVI конференция, 2003 (тезисы).
7. Бородин М.В., Устюжанин А.Е., Построение обучаемых многоагентных систем на базе платформы JADE // IEEE AIS-2004, с 73-82.
8. Жданов А.А., Устюжанин А.Е., Караваев М.В., Липкевич Д.Б., 4GN - инструмент для разработки нейроподобных адаптивных систем управления на основе метода автономного адаптивного управления // Нейроинформатика-2005.
9. Жданов А.А., Устюжанин А.Е., Караваев М.В. Нейросетевой самообучаемый метод адаптивного управления динамическими объектами. // Материалы XXIX Академических чтений по космонавтике, 2005 год. М.: 2005. с. 93.
10. Устюжанин А.Е., Ассоциативная память контурных объектов на основе вейвлет представлений // МФТИ-2005, с. 88-101.
11. Сурин Н.В., Устюжанин А.Е., Визуализация многомерных данных с помощью самоорганизующихся карт // МФТИ, XLVIII конференция
12. Ющенко А.В., Устюжанин А.Е., Распознавание и интеграция метаданных разнородных источников // МФТИ, XLVIII конференция, 2005
13. Устюжанин А.Е., Метод построения системы памяти для хранения и поиска многомерных пространственно-временных последовательностей // Вестник Московского государственного технического университета им. Н.Э. Баумана, серия «Приборостроение», номер 2, 2007, с. 104- 112.