

ИНСТИТУТ СИСТЕМНОГО ПРОГРАММИРОВАНИЯ
РОССИЙСКОЙ АКАДЕМИИ НАУК

На правах рукописи

Караваев Максим Владимирович

**ПРИМЕНЕНИЕ НЕЧЕТКОЙ ЛОГИКИ
В СИСТЕМАХ АВТОНОМНОГО АДАПТИВНОГО УПРАВЛЕНИЯ**

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

2006

Работа выполнена:
в Институте Системного Программирования РАН.

Научный руководитель: доктор физико-математических наук
Жданов Александр Аркадьевич

Официальные оппоненты: доктор технических наук,
Ющенко Аркадий Семенович

доктор физико-математических наук,
Рязанов Владимир Васильевич

Ведущая организация: Факультет Вычислительной математики
и кибернетики Московского
Государственного Университета
им. М.В. Ломоносова

Защита диссертации состоится « 8 » сентября 2006 г. в 15⁰⁰ часов 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 « 1 » августа 2006 г.

Ученый секретарь
специализированного совета

/Прохоров С.П./

Общая характеристика работы

Актуальность темы

Постоянно растущая сложность и разнообразие задач, решение которых возлагается на автоматические системы, в последнее время определяют повышенную потребность в системах управления с возможно более универсальными свойствами. Однако, ни один из существующих на текущий момент подходов к построению систем управления – ни теория автоматического управления (ТАУ), ни нечеткие системы, ни искусственные нейронные сети, ни другие технологии, не обладают необходимой универсальностью. С другой стороны, имеет место уверенность в возможности построения систем с искомыми свойствами, которая основана на наблюдении действия таких систем в природе – нервные системы живых организмов.

Возможно, что претендовать на более широкий охват типов решаемых задач могут методы, основанные на концептуальных подходах, моделирующих функциональность и структуру нервных систем живых организмов, поскольку в таких методах объект управления описывается совокупностью формализованных знаний, заранее заданных, либо эмпирически приобретенных системой, и менее привязанных к той или иной функциональной зависимости от параметров объекта управления, чем это имеет место в моделях, выраженных в аналитической форме уравнений. К таким концептуальным подходам относится и метод автономного адаптивного управления (ААУ), развиваемый в Отделе имитационных систем Института Системного Программирования РАН.

Метод ААУ определяет общие принципы и некоторые частные реализации систем управления объектами, для которых отсутствует математическая модель и его логико-лингвистическое описание, а известны только общие критерии качества управления (целевая функция).

Однако все первые системы ААУ строились на основе дискретных устройств, что ограничивало область их применения, в особенности, при работе таких систем управления в «реальном» непрерывном мире. Для обеспечения достаточно гладких зависимостей между входными и выходными переменными необходимо было разбивать входное и выходное признаковые пространства на достаточно большое количество диапазонов, и на каждом из них проводить пробные управляющие воздействия для нахождения закономерных реакций. Это приводило к экспоненциальному росту объема базы знаний и времени обучения

системы при линейном увеличении критериев качества управления или количества входных переменных системы.

Несмотря на то, что были разработаны специальные алгоритмы, позволяющие значительно ускорить обучение системы, основываясь на различных эвристиках, в целом эффективность систем ААУ оставалась реализованной далеко не полностью.

Применение нечеткой логики, основанное на теореме FAT (Fuzzy Approximation Theorem), доказанной Б. Коско (B. Kosko), и гласящей, что любая математическая система может быть аппроксимирована системой на основе нечеткой логики, значительно повышает эффективность систем ААУ.

Актуальность данной работы заключается, с одной стороны, в необходимости наделения систем ААУ непрерывными свойствами, за счет использования теории нечетких множеств, с целью улучшения их характеристик и расширения области применения. С другой стороны, в работе предлагаются возможности расширения области применения традиционных нечетких контроллеров (систем управления, работающих по закону, заданному набором нечетких продукционных правил) за счет применения в них методов обучения и адаптации, используемых в системах ААУ. Разработанные в работе методы могут использоваться для построения широкого класса прикладных систем управления.

Цель работы

Цель диссертационной работы состояла в том, чтобы найти подходы и разработать методы построения управляющих систем ААУ на основе теории нечетких множеств. Такие методы позволят преодолеть ограничения систем ААУ, связанные с обязательным требованием четкой дискретизации информационных объектов.

Для достижения указанной цели необходимо было решить следующие задачи:

1. Разработать метод автоматического определения входных и выходных нечетких множеств, заданных при помощи функций принадлежности.
2. Разработать метод автоматической генерации правил управления, описывающих функциональную взаимосвязь между входом и выходом.
3. Разработать метод автоматической адаптации набора правил базы знаний (БЗ) к изменяющимся свойствам объекта управления (ОУ) и среды.

4. Разработать метод оценивания образов и принятия решений.
5. Осуществить проверку работоспособности и эффективности разработанных методов на примерах прикладных систем.

Научная новизна

Научная новизна работы заключается в том, что в данной работе впервые была предпринята попытка построения способной к самообучению автономной адаптивной системы управления на основе нечеткой логики. А именно:

- введено описание системы ААУ в терминах теории нечетких множеств. Разработана кодировка входных и выходных нечетких множеств и нечетких правил управления специального вида;
- предложен метод генерации нечетких правил управления, а также входных и выходных функций принадлежности на основе модифицированного метода кластеризации методом вычетов;
- предложен метод адаптации набора правил в базе знаний к изменяющимся свойствам объекта управления и среды посредством введения для каждого правила специальной величины, названной «степенью адекватности», и способов вычисления ее значения;
- предложен метод принятия решений, учитывающий степени адекватности правил управления в базе знаний и оценки результирующих образов правил;
- на примере модели прикладной системы управления показаны преимущества разработанного подхода в скорости обучения перед системами управления, построенными на основе парадигмы обучения с подкреплением, в традиционной тестовой задаче балансирования перевернутым маятником.¹

Практическая ценность

Практическая ценность работы состоит в том, что разработанные методы могут быть применены для построения прикладных самообучающихся управляющих систем для широкого класса объектов управления.

¹ Задача состоит в том, чтобы при помощи перемещений тележки удерживать физический маятник, на который действует сила притяжения, в вертикальном положении. Маятник нижним концом прикреплен к тележке при помощи шарнира, а тележка может с помощью привода перемещаться вдоль прямой, лежащей на горизонтальной плоскости.

Возможность построения прикладных систем управления различными объектами и их преимущества показаны на примерах системы балансирования перевернутым маятником, системы стабилизации углового движения космического аппарата и системы управления моделью мобильного робота Pioneer 3-DX.

Апробация работы и публикации

По материалам диссертации опубликовано 10 работ [1-10].

Основные положения работы докладывались на следующих конференциях и семинарах:

- всероссийской научно-технической конференции Нейроинформатика-2004, 2005 МИФИ,
- семинаре «Экобионика» МГТУ им. Н.Э. Баумана,
- XXIX Академических чтениях по космонавтике, 2005 год,
- всероссийской научно-технической конференции "Наука – производство – технологии – экология» ВятГУ (г. Киров).

Структура и объем диссертации

Работа состоит из введения, четырех глав, заключения и списка литературы. Общий объем диссертации составляет 106 страниц. Список литературы содержит 49 наименований.

Краткое содержание работы

В 1-й главе представлен обзор известных методов и проблем построения адаптивных систем управления.

Во 2-й главе приводится обобщенная формализация системы ААУ с применением теории нечетких множеств, а также описываются разработанные в диссертации методы синтеза таких нечетких систем ААУ и их адаптации к изменениям свойств объекта управления и среды.

В 3-ей главе приводится описание разработанной в ходе диссертационной работы модели прикладной системы управления.

В 4-ой главе представлены результаты нескольких серий экспериментов, в ходе которых производилось сравнение разработанной модели прикладной системы с аналогичными системами, а также уточнялись параметры разработанной системы управления, оптимальные для различных объектов управления и целевых функций.

В 1-ой главе представлен анализ основных технологий построения систем управления с точки зрения возможности организации самообучения и адаптации: теории автоматического управления, дискретных автоматов, нечетких систем управления (нечетких контроллеров), систем управления, построенных на основе искусственных нейронных сетей (нейроконтроллеров), и концептуальных моделей нервных систем.

Сделаны выводы, что системы ТАУ (построенные в соответствии с теорией автоматического управления), нечеткие и нейроконтроллеры обладают особенностями, которые затрудняют построение на их основе адаптивных самообучающихся систем управления. Для первых двух технологий это ограничение связано с тем, что они основываются соответственно на математической модели и логико-лингвистическом описании объекта управления, которые не всегда доступны и часто не поддаются изменениям в реальном времени. Как следствие – не существует хорошо разработанных методов наделения систем ТАУ и нечетких контроллеров свойствами адаптивности и самообучаемости. Такие методы отсутствуют и для нейроконтроллеров, которым свойственна проблема катастрофического забывания, вследствие чего нейросети теряют старые «навыки» в процессе получения новых. Помимо этого, для первоначального обучения нейросети необходимо иметь обучающую выборку, которая не всегда доступна.

Для дискретных автоматов существует большее количество алгоритмов их самообучения и адаптации, к тому же автоматы обладают активным поведением, в противовес реактивному поведению (при котором система лишь отвечает на входные воздействия), свойственным системам управления, построенным в соответствии с предыдущими методами. Однако при использовании автоматов для взаимодействия с реальным миром часто возникают сложности из-за несоответствия дискретной природы автоматов и непрерывной природы реального мира.

Среди известных концептуальных подходов к моделированию нервных систем до уровня решения прикладных задач доведены только исследования

французской лаборатории AnimatLab под руководством Ж. А. Мейера (J. A. Meyer), основывающиеся на парадигме обучения с подкреплением, и, в большей степени, – метод ААУ, развиваемый в отделе Имитационных систем ИСП РАН под руководством А.А. Жданова.

Однако все существующие методы построения прикладных систем по методу ААУ используют конечные автоматы и обладают всеми особенностями, присущими этим дискретным устройствам. Из этого вытекают и такие недостатки, как, например, экспоненциальный рост времени обучения и объема БЗ при увеличении количества входных переменных или необходимой точности управления. Такой экспоненциальный рост обусловлен равномерным разбиением множеств значений входных и выходных переменных системы на диапазоны и привязкой к каждому их сочетанию минимум одного элемента системы.

Устранение этих ограничений за счет применения нечеткой логики и является главной целью данной работы.

Во 2-й главе содержится обобщенная формализация системы ААУ с применением теории нечетких множеств, а также описываются разработанные в диссертации методы синтеза таких нечетких систем ААУ и их адаптации к изменениям свойств объекта управления и среды.

Краткое описание обобщенной системы автономного адаптивного управления в базе теории нечетких множеств

Пусть в среде, близкой к стационарной, находится объект управления (ОУ) со встроенной управляющей системой (УС), состоящей из блока датчиков, блока формирования и распознавания образов (ФРО), БЗ, аппарата эмоций, блока принятия решений и исполняющих механизмов. Пусть УС является системой замкнутого цикла, в которой каждый блок периодически получает входные значения и формирует выходные значения, а моменты генерации выходных значений отстоят друг от друга на равные интервалы времени. Согласно базовым положениям метода ААУ, УС имеет две целевые функции: обеспечение выживания ОУ и накопление знаний о закономерностях его функционирования в среде.

Для наблюдения за параметрами ОУ и среды УС имеет N входных переменных, связанных с датчиками, а для воздействия на среду и объект – M выходных переменных, связанных с исполнителями. Датчики отображают значения некоторых параметров среды и ОУ в множества значений входных

переменных x_1, x_2, \dots, x_N . Множества значений выходных переменных обозначим y_1, y_2, \dots, y_M . В общем случае все эти множества бесконечны, но обязательно имеют верхнюю и нижнюю границы. Тогда входные и выходные множества УС будут равны соответственно декартовым произведениям:

$$X = x_1 \times x_2 \times \dots \times x_N \text{ и } Y = y_1 \times y_2 \times \dots \times y_M. \quad (1)$$

В УС априорно заложено множество S оценок состояний ОУ, которое дискретно, полностью упорядочено и состоит из конечного числа элементов, т.е. $S = \{s_1, s_2, \dots, s_P\}$, отношение порядка на нем: $s_1 < s_2 < \dots < s_P$, а также определенная на подмножестве X множества X функция $E(x)$, ставящая каждому элементу X в соответствие один элемент множества S (здесь использование подмножества X основано на том, что не все датчики измеряют параметры среды и ОУ, непосредственно влияющие на выживание ОУ). Таким способом задается первая целевая функция управления (обеспечение выживания ОУ в среде).

Для соблюдения одного из условий построения систем по методологии ААУ – условия дискретности, которое говорит о конечном количестве элементов в структуре УС (в том числе и элементов памяти) невозможно использовать элементы в общем случае бесконечных множеств X и Y непосредственно для управления. Для соблюдения этого условия покроем множества X и Y конечным количеством нечетких (возможно пересекающихся) подмножеств: O_1, O_2, \dots, O_L и A_1, A_2, \dots, A_K , которые назовем «образами» и «действиями» соответственно.

Для каждого нечеткого множества O_1, O_2, \dots, O_L и A_1, A_2, \dots, A_K зададим набор функций $\mu_{i1}^O(x_1) \dots \mu_{iN}^O(x_N)$ и $\mu_{j1}^A(y_1) \dots \mu_{jM}^A(y_M)$, где i – номер входного образа, $i=[1, L]$, а j – выходного $j=[1, K]$, определяющих степени принадлежности им каждого элемента из множеств X и Y соответственно. Пусть множеством значений всех функции принадлежности в системе являются действительные числа из диапазона $[0; 1]$. При таком покрытии входного и выходного множеств нечеткими подмножества и заключается одно из ключевых отличий предлагаемого подхода от базовой методологии ААУ, в которой образы и действия носят исключительно дискретный характер.

Каждой входной и выходной переменной x_1, x_2, \dots, x_N и y_1, y_2, \dots, y_M можно поставить в соответствие лингвистическую переменную, которая будет принимать в качестве своих значений нечеткие множества, которыми покрывается диапазон значений соответствующей входной или выходной переменной.

Задачи и методы синтеза и адаптации нечетких систем ААУ

Для обеспечения выживания ОУ в среде и накопления знаний о ней, методология ААУ предписывает УС решать следующие задачи:

1. Нахождение закономерностей в информации, поступающей во входные переменные, сохранение их в виде образов и распознавание этих образов при появлении их признаков на входах, т.е. решение задач автоматической кластеризации, распознавания образов.

Для решения этих задач в структуре УС существует блок ФРО, который в нашем случае формирует образы O_i , определяя параметры их функций принадлежности. Также на каждом такте блок ФРО вычисляет принадлежности текущих значений входных переменных каждому из сформированных образов, фактически выполняя традиционную для нечетких контроллеров процедуру фаззификации. Такой способ распознавания образов позволяет наделять базовый метод ААУ, в котором факт распознавания образа определялся лишь двумя числами 0 или 1, преимуществами нечетких контроллеров.

2. Оценивание образов с точки зрения полезности или опасности их для ОУ, используя множество «элементарных» оценок S и функцию $E(x)$, задающую отображение $X \rightarrow S$.

Решением этой задачи в системе ААУ занимается другой блок – аппарат эмоций, который для каждого образа O_i вычисляет его качественную оценку E_i .

3. Нахождение закономерностей управления ОУ в среде в виде продукций $O_i \& A_j \rightarrow O_k$, где O_i и A_j – нечеткие множества, которым принадлежали значения входных и выходных переменных на предыдущем такте, а O_k – множество, которому принадлежат значения входных переменных на текущем такте.

БЗ сохраняет пространственно-временные закономерности в виде нечетких правил вида $P_h: O_i \& A_j \rightarrow O_k$, где P_h – h -ое правило управления в БЗ.

4. Вычисление выходных воздействий на каждом такте работы УС, подаваемых в среду или ОУ через «исполнительные органы», основываясь на качественных оценках распознанных образов и обнаруженных закономерностях управления, сохраненных в БЗ.

Эту задачу решает блок принятия решений, который на каждом такте работы системы анализирует содержимое БЗ, вычисляет выходные значения всех

правил в текущей ситуации, и, принимая во внимание качественные оценки результирующих образов правил, вычисляет выходные воздействия УС (значения выходных переменных). Этот процесс схож с процедурой дефаззификации, при помощи которой происходит принятие решений в нечетких контроллерах. В отличие от базового способа принятия решений в методе ААУ, при таком подходе выходные значения блока ФРО могут носить непрерывный характер, а не выбираться из конечного набора фиксированных значений.

Рассмотрим решение этих задач, предлагаемых в данной диссертационной работе.

Формирование образов, как пространственных в блоке ФРО, так и пространственно-временных в БЗ, происходит с использованием процедуры кластеризации методом вычетов.

Входными данными для процедуры кластеризации является история управления, которая накапливается в специальной буферной памяти системы за определенное количество тактов работы УС и записывается в виде (2).

$$\{x_{t,1}, x_{t,2}, \dots, x_{t,N}\}, \{y_{t,1}, y_{t,2}, \dots, y_{t,M}\}, \{x_{t+1,1}, x_{t+1,2}, \dots, x_{t+1,N}\}, \{y_{t+1,1}, y_{t+1,2}, \dots, y_{t+1,M}\}, \dots \quad (2)$$

где $x_{t,i}$ – значение i -й входной переменной (показание i -го датчика) в момент времени t , а $y_{t,i}$ – значение i -й выходной переменной (номер действия, выполненного i -ым исполнителем) в момент времени t , N и M – количество входных и выходных переменных соответственно.

Процедура генерации правил запускается при полном заполнении буферной памяти, после выполнения кластеризации память полностью очищается.

Для генерации одноктоковых правил управления используются значения входных и выходных параметров t -го такта и значения входных параметров $(t+1)$ -го такта, которые задают координаты одной точки в $(N+M+N)$ -мерном пространстве.

Рассмотрим пример системы управления, балансирующей перевернутым физическим маятником, установленным на тележке. Маятник нижним концом прикреплен к тележке при помощи шарнира, тележка может с помощью привода перемещаться вдоль прямой, лежащей на горизонтальной плоскости. Задача состоит в том, чтобы при помощи перемещений тележки удерживать маятник, на который действует сила притяжения, в вертикальном положении.

Для решения задачи балансирования маятником система управления должна получать информацию о текущем угле и угловой скорости маятника и определять значение управляющего воздействия на тележку. В этом случае одна точка на входе процедуры кластеризации будет задаваться значениями угла, угловой скорости и управляющего воздействия на тележку на t -ом такте и значениями угла и угловой скорости на такте $t+1$. В результате, на вход процедуры кластеризации поступает набор точек, заданных своими координатами в $(N+M+N)$ -мерном пространстве. В примере с балансированием маятником размерность пространства получается равной $2+1+2=5$.

Процедура кластеризации выполняется методом вычетов. На вход процедуры поступает набор точек и максимальный размер кластера R , а на выходе получается набор кластеров, заданных центральной точкой и набором точек, также принадлежащих кластеру.

Формирование входных нечетких множеств (функций принадлежности)

Входные нечеткие множества традиционно для нечетких контроллеров задаются при помощи функций принадлежности. Функции принадлежности имеют трапецеидальный вид и задаются x -координатами четырех вершин трапеции: x_{lb} , x_{lt} , x_{rt} , и x_{rb} (рис. 1). Такая форма задания позволяет определять помимо трапецеидальных функций, также функции треугольного вида (при $x_{lt} = x_{rt}$).

Выбор трапецеидального вида функций принадлежности основан на сравнительной простоте вычисления данного вида функций, а также возможности интерполяции с их помощью большинства распространенных функций принадлежности, например, треугольных и колоколообразных, с достаточной для большинства задач точностью. К тому же, часто объекты управления обладают поведением, близким к линейному на небольших интервалах значений входных и выходных переменных, поэтому для управления ими можно применять нечеткие контроллеры с трапецеидальными функциями принадлежности.

Функция принадлежности генерируется на основе набора точек, отстоящих не более чем на евклидово расстояние R от центра кластера, полученного в результате работы процедуры кластеризации.

Рассмотрим автоматическую процедуру генерации функций принадлежности по координатам точек кластера, соответствующим угловому отклонению (другими словами – по проекциям точек кластера на одну из осей координатного пространства).

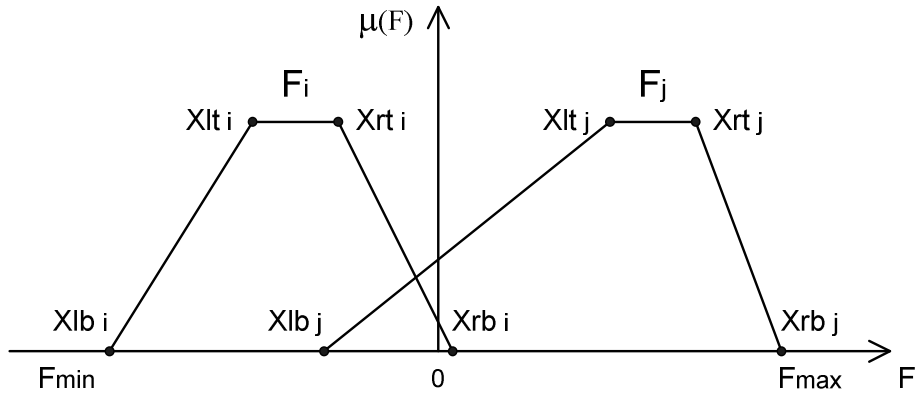


Рисунок 1. Пример задания функций принадлежности для углового отклонения.

Пусть R – максимальный размер кластера, c_i – координата i -ой точки кластера (например, i -ое значение углового отклонения для системы управления перевернутым маятником), c_{mid} – координата центра кластера, а N – количество точек в кластере ($N \geq 2$), причем точки кластера отсортированы по возрастанию значений. Тогда координаты вершин трапеции, задающей функцию принадлежности трапецеидального вида, будут вычисляться по формуле (3).

$$\begin{aligned}
 X_{lb} &= 2 \cdot c_1 - c_2 \\
 X_{rb} &= 2 \cdot c_N - c_{N-1} \\
 X_{lt} &= c_2 \\
 X_{rt} &= c_{N-1}
 \end{aligned} \tag{3}$$

При большом разбросе точек кластера может возникнуть ситуация, в которой ширина основания функции принадлежности ($x_{rb} - x_{lb}$) превысит максимальный размер кластера R . В этом случае координаты корректируются по формулам (4).

$$\begin{aligned}
 X_{rb} &= X_{rb} - ((X_{rb} - X_{lb}) - R) \cdot \frac{X_{rb} - X_{rt}}{(X_{rb} - X_{rt}) + (X_{lt} - X_{lb})} \\
 X_{lb} &= X_{lb} - ((X_{rb} - X_{lb}) - R) \cdot \frac{X_{lt} - X_{lb}}{(X_{rb} - X_{rt}) + (X_{lt} - X_{lb})}
 \end{aligned} \tag{4}$$

Перед добавлением функции принадлежности в базу знаний, производится проверка на совпадение новой функции с уже существующими функциями в БЗ.

Пусть G_i – площадь под графиком i -й функции принадлежности, уже существующей в БЗ ($i=1,..,D$, где D – количество функций принадлежности в БЗ), G_c – площадь новой функции принадлежности, $G_{i,c}$ – площадь пересечения двух

фигур (трапеций), образованных графиками соответствующих функций. Тогда, при выполнении условия (5) для некоторого i , добавление новой функции в базу знаний не происходит, а вместо новой функции используется уже существующая функция i из БЗ, которая модифицируется в соответствии с формулой (6). Фактически формула (6) выполняет операцию дизъюнкции над новым и уже существующим нечеткими множествами.

$$(G_i + G_c - G_{i,c}) / (G_i + G_c) < Pa, \quad (5)$$

где Pa – заданный априорно порог «похожести» двух функций принадлежности.

$$\begin{aligned} X_{lbi} &= \min(X_{lbi}, X_{lbc}) \\ X_{rbi} &= \max(X_{rbi}, X_{rbc}) \\ X_{lci} &= \min(X_{lci}, X_{lcc}) \\ X_{rci} &= \max(X_{rci}, X_{rcc}). \end{aligned} \quad (6)$$

Генерация нечетких правил управления

После генерации блоком ФРО функций принадлежности для результирующего образа, вычисляется качественная оценка E_h этого образа. Правило, с использованием выходных данных процедуры кластеризации, записывается в виде (7):

$$\begin{aligned} P_h: & (\mu_{i1}^x(x_1) \& \mu_{i2}^x(x_2) \& \dots \& \mu_{iN}^x(x_N)) \& (\mu_{i1}^y(y_1) \& \mu_{i2}^y(y_2) \& \dots \& \mu_{iM}^y(y_M)) \rightarrow \\ & \rightarrow (\mu_{i+1\ 1}^x(x_1) \& \mu_{i+1\ 2}^x(x_2) \& \dots \& \mu_{i+1\ N}^x(x_N)) \& E_h, \end{aligned} \quad (7)$$

где выражение $(\mu_{i1}^x(x_1) \& \mu_{i2}^x(x_2) \& \dots \& \mu_{iN}^x(x_N))$ задает входной образ (набор входных множеств) правила, $(\mu_{i1}^y(y_1) \& \mu_{i2}^y(y_2) \& \dots \& \mu_{iM}^y(y_M))$ – набор управляющих воздействий, $(\mu_{i+1\ 1}^x(x_1) \& \mu_{i+1\ 2}^x(x_2) \& \dots \& \mu_{i+1\ N}^x(x_N))$ – образ результата (набор выходных множеств), который должен распознаваться при выполнении данного правила, а E_h – оценка результирующего образа.

Распознавание образов в нечетких системах ААУ

Процесс распознавания выполняется традиционной для нечетких контроллеров процедурой – фаззификацией, – вычислением степеней принадлежности текущих значений входных параметров входным нечетким множествам. Далее, по значениям входных функций принадлежности вычисляются результирующие послылки правил в соответствии с принятой в системе реализацией

операции «t-конорма» (операция нечеткое «И» может быть реализована через умножение, взятие минимального числа или другую операцию над значениями функций принадлежности).

Значения результирующих посылок β_i всех правил поступают на вход подсистемы принятия решений.

Принятие решений в нечетких системах ААУ

Алгоритм принятия решений состоит из следующих последовательных шагов:

1. Вычисление посылок каждого из правил в соответствии со значениями входных функций принадлежности.
2. Корректировка посылки каждого правила в соответствии со значением степени адекватности правила. Для i -го правила

$$\beta'_i = \beta_i \cdot Q_i. \quad (8)$$

3. Корректировка посылки каждого правила в соответствии с оценкой результирующего образа правила по одной из следующих формул (выбирается в зависимости от вида объекта управления и других критериев):

$$\beta''_i = \beta'_i \cdot \frac{a \tan \left(\left(\frac{E_i}{E_{\max}} - 0.5 \right) \cdot 13 \right) \cdot 1.1}{\pi} + 0.5, \quad (9)$$

$$\beta''_i = \beta'_i \cdot \left(\frac{E_i}{E_{\max}} \right)^2, \quad (10)$$

$$\beta''_i = \begin{cases} 0, & \text{если } E_i < E_{\min} \cdot E_{\max} \\ \beta'_i \cdot \frac{E_i}{E_{\max}}, & \text{если } E_i \geq E_{\min} \cdot E_{\max} \end{cases}, \quad (11)$$

где $E_{\min} = [0,1]$ – коэффициент, задающий нижний порог, отсекающий правила с низкими оценками, а E_{\max} – максимальное значение оценки.

Первая формула реализует сигмоидальную зависимость – она «подавляет» правила с низкой оценкой результирующего образа и «вытягивает» правила с оценкой выше перегиба функции (0.5).

Вторая формула реализует квадратичную зависимость посылки правила от оценки результирующего образа. Эту формулу следует применять для систем, в которых состояния с высокими оценками достижимы из большинства других состояний. Правила с низкими оценками в соответствии с этой формулой достаточно сильно «подавляются».

Третья формула задает пороговую зависимость посылок правил от оценок результирующих образов. Все правила с оценкой ниже порога исключаются из процесса принятия решения, тем самым исключается попадание ОУ в состояния с нежелательной оценкой. Эта формула пригодна для применения в системах, управляющих объектами, переход которых в состояние с низкой оценкой может привести к нежелательным последствиям.

Далее результирующие посылки всех правил суммируются и полученная сумма сравнивается с априорно заданным порогом P_{Smin} . Если величина суммы меньше порога, значит в БЗ недостаточно правил управления, предлагающих действия для текущего состояния ОУ, ведущие к состоянию с удовлетворительной оценкой, и действие выбирается случайным способом.

Если величина суммы больше порога, то происходит вычисление результирующего управляющего воздействия по одному из алгоритмов нечетких контроллеров: вычисления средневзвешенного значения (12), «победитель забирает все» (выбор управляющего воздействия правила с максимальной посылкой), или по методу центра масс (13).

$$y_j = \frac{\sum (a_{jh} \cdot \beta_h)}{\sum \beta_h}, \quad (12)$$

$$y_j = \frac{\int \beta_h \cdot \mu_{jh}^y(a_{jh}) \cdot da_{jh}}{\int_{y_j} \mu_{jh}^y(a_{jh}) \cdot da_{jh}}, \quad (13)$$

где y_j – значение j -ой выходной переменной, $j=[1,M]$, β_h – значение результирующей посылки h -го правила, a_{jh} – значение выходного управляющего воздействия h -го правила для j -ой выходной переменной.

Второй способ вычисления применим для объектов управления, ближайшие состояния которых по значениям параметров не всегда обладают близкими оценками. И наоборот, состояния с близкими оценками не всегда имеют близкие значения параметров.

Далее вычисленное управляющее воздействие подается на вход исполняющего органа, который в соответствии с ним воздействует на ОУ.

Адаптация (управление базой знаний) нечетких систем ААУ

Исходя из основного условия задачи, согласно которому система управления должна иметь способность управлять ОУ в близкой к стационарной среде, т.е. в такой среде, свойства которой могут меняться во времени, система должна адаптироваться к изменениям свойств среды.

Под адаптацией мы понимаем переобучение системы – изменение алгоритма управления объектом.

Ввиду того, что знания в системе хранятся в виде нечетких правил управления (продукций), изменение алгоритма управления возможно двумя способами: либо изменением уже существующих правил, либо удалением существующих правил и генерацией новых.

Исходя из рассмотренной организации БЗ, при которой описания входных и выходных нечетких множеств хранятся отдельно от правил, и каждое нечеткое множество может быть использовано несколькими разными правилами, изменение входных множеств нежелательно, поскольку попытка изменить одно правило может повлечь изменения в других правилах, изменять которые не нужно.

Если исходить из условия, что входные и выходные множества остаются неизменными, можно заключить, что первый способ изменения алгоритма управления равносильен второму. То есть адаптация системы осуществляется за счет удаления старых правил, которые становятся неадекватными изменившимся свойствам ОУ и среды, и генерации новых правил.

Каждое правило, находящееся в базе знаний, должно контролироваться с точки зрения адекватности описания им закономерностей взаимодействия ОУ и среды. Для этой цели введем для каждого правила некоторый параметр Q_j , при помощи которого будем задавать степень адекватности j -го правила. Пусть $Q_j = 0$ говорит о полной неадекватности правила текущим свойствам ОУ и среды, а $Q_j = 1$ – об обратном. Правила с $Q_j < Q_{min}$, где Q_{min} – априорно заданный порог, ухудшают качество управления, и в некоторых реализациях систем управления могут быть удалены из БЗ для экономии памяти.

Очевидно, величина Q_j должна меняться в процессе работы УС в соответствии с изменениями свойств объекта управления и среды. То есть значение

этой величины можно использовать для отслеживания изменений свойств объекта управления и среды.

Изменение величины Q_j на очередном такте должно зависеть от того, насколько j -ое правило принимало участие в управлении на предыдущем такте и на сколько получившийся при этом результат соответствует результату, предсказываемому правилом.

Тогда модуль изменения величины Q_j на t -ом такте будет зависеть от степени принадлежности входных величин множествам входных переменных правила (посылка) и от степени принадлежности реально выполненного действия выходным переменным правила, что можно выразить в виде зависимости (14):

$$\Delta Q_j \propto w_j^x(O^{t-1}) \cdot w_j^y(A^{t-1}), \quad (14)$$

где, исходя из того, что

$$O^{t-1} = \begin{bmatrix} x_1^{t-1} \\ x_2^{t-1} \\ \dots \\ x_N^{t-1} \end{bmatrix} \quad A^{t-1} = \begin{bmatrix} y_1^{t-1} \\ y_2^{t-1} \\ \dots \\ y_M^{t-1} \end{bmatrix}, \quad (15)$$

величины w_j^x и w_j^y примут значения

$$w_j^x(O^{t-1}) = \min_{i=1,N} (\mu_{ij}^x(x_i^{t-1})), \quad (16)$$

$$w_j^y(A^{t-1}) = \min_{i=1,M} (\mu_{ij}^y(y_i^{t-1})).$$

Знак и величина изменения степени адекватности должны зависеть от принадлежности результирующих величин образу, описываемому данным правилом.

Введем также некоторую величину q – константу, задаваемую априорно, которая будет определять скорость изменения степени адекватности правил – некоторый аналог скорости сходимости для нейронных сетей.

Тогда значение величины Q_j на каждом такте работы УС можно записать в виде формулы (17):

$$Q_j^{t+1} = Q_j^t + w_j^x(O^{t-1}) \cdot w_j^y(A^{t-1}) \cdot q \cdot (w_j^x(O^t) - Q_j^t). \quad (17)$$

Полученное значение степени адекватности j -го правила Q_j может быть использовано в процессе двумя способами: как поправочный коэффициент,

корректирующий результирующую посылку j -го правила, и как индикатор необходимости присутствия данного правила в БЗ.

В 3-ей главе приводится описание разработанной в диссертации модели прикладной системы управления.

Разработка программной реализации нечеткой системы ААУ проведена с использованием CASE-системы 4GN, созданной с участием автора специально для ускорения разработки и унификации интеллектуальных систем управления, основанных на методе ААУ. Схема системы 4GN изображена на рис.2.

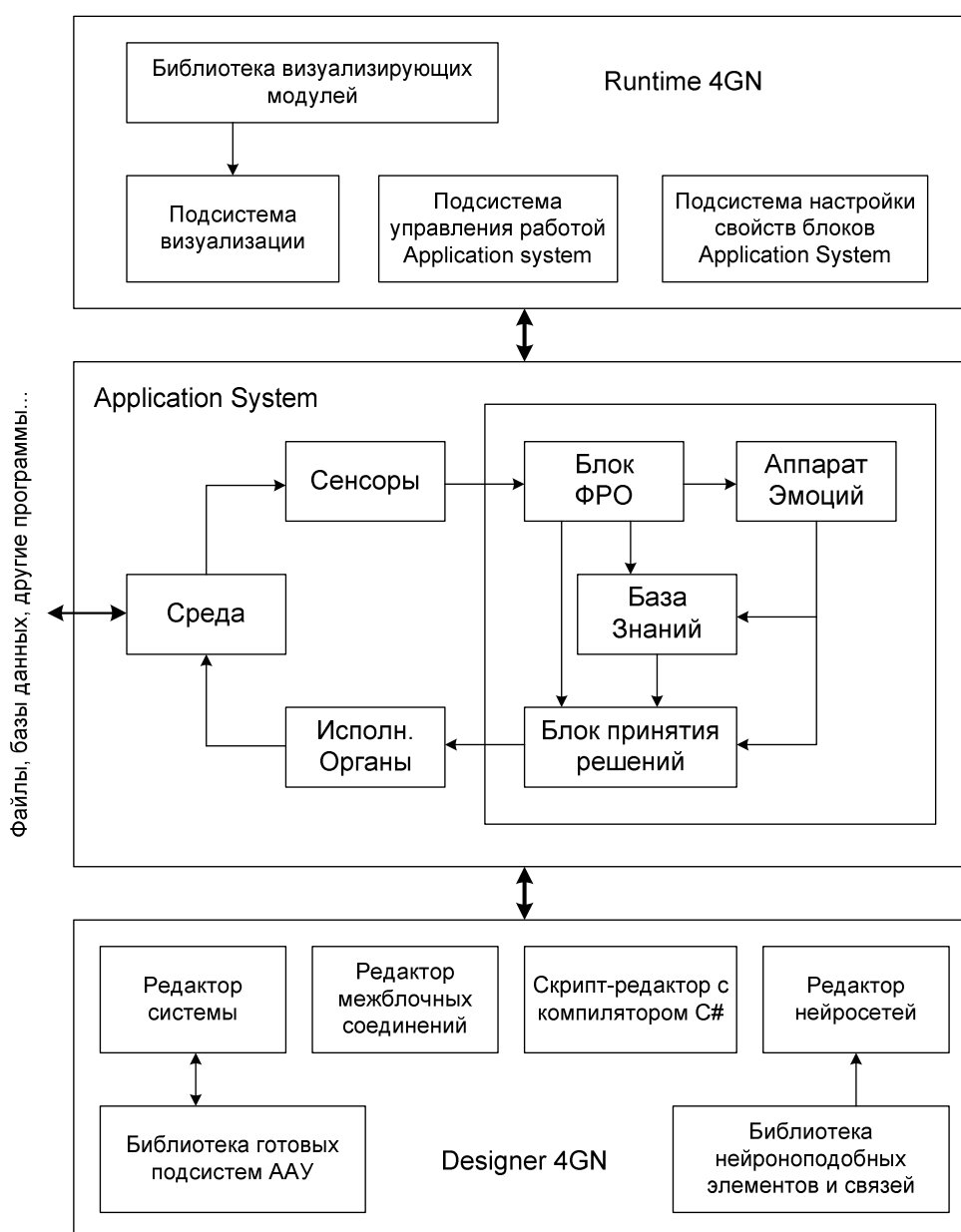


Рисунок 2. Общая схема системы 4GN.

Помимо структуры и основных функций 4GN, в главе описана структура разработанного в рамках данной диссертационной работы программного обеспечения. А именно: базовых классов системы ААУ для среды 4GN и модулей, реализующих:

- модель нечеткой системы управления ААУ, состоящей из блоков FuzzyPFR, FuzzyQAS, FuzzyKB и FuzzyDM;
- модели объектов управления – перевернутого маятника и корпуса космического аппарата – PendulumCO и SatelliteCO;
- подсистемы визуализации реального времени для 4GN;
- модулей для подсистемы визуализации ViewMembershipFunctions, ViewString и ViewPendulum, отображающих соответственно функции принадлежности, правила управления в текстовом виде и изображение перевернутого маятника.

В 4-ой главе представлены результаты нескольких серий экспериментов, в ходе которых уточняются параметры разработанной системы управления, оптимальные для различных объектов управления и целевых функций.

На прикладной системе управления перевернутым маятником были исследованы зависимости качества управления, скорости обучения системы и эффективности базы знаний от максимального размера кластера и длительности периода генерации правил.

В частности, было экспериментально установлено, что

- эффективность БЗ линейно повышается при увеличении длительности периода генерации правил с 10 до 100 тактов;
- качество управления системы, а также количество правил в БЗ и время обучения системы падает при увеличении максимального размера кластера с 0,2 до 0,35. При значениях размера кластера, выходящих за пределы этого диапазона система теряет способность к обучению.

Также в этой главе разработанная прикладная система управления сравнивается с другими системами управления, решающими аналогичные задачи:

1. проводится сравнение разработанной системы с системами управления, основанными на различных нейросетевых реализациях обучения с подкреплением (reinforcement learning) на традиционной тестовой задаче

балансирования перевернутым маятником. В ходе проведения экспериментов, результаты которых представлены в таблицах 1 и 2, показано преимущество нечеткой системы ААУ над другими системами, участвовавшими в экспериментах в скорости обучения примерно на 9% при балансировании маятником из фиксированного начального положения и примерно на 115% при балансировании маятником из случайного начального положения при равных критериях качества управления;

Таблица 1. Результаты опытов по балансированию перевернутым маятником из фиксированного начального положения.

Система управления	GENITOR	SANE	Igel	НААУ	НААУ
Среднее количество попыток	1846	535	283	259	395
Стандартное отклонение (SD)	1396	329	138	154	279
Максимальное отклонение маятника	12°	12°	12°	12°	8°

Таблица 2. Результаты опытов по балансированию перевернутым маятником из произвольного начального положения.

Система управления	GENITOR	SANE	Igel	НААУ
Среднее количество попыток	2578	1691	967	448
Стандартное отклонение (SD)	2092	984	1148	84
Максимальное отклонение маятника	12°	12°	12°	12°

2. На примере задачи стабилизации углового движения космического аппарата проводилось сравнение разработанной нечеткой системы ААУ с ранее разработанными вариантами системы ААУ, не использующими нечеткую логику. В ходе экспериментов нечеткая система ААУ показала преимущество в скорости обучения примерно в 8 раз (таблица 3). На этой же задаче была продемонстрирована способность разработанной системы к переобучению (адаптации);

Таблица 3. Результаты опытного сравнения различных вариантов системы управления угловым движением космического аппарата.

Система управления	Pilot 1	Pilot 2	НААУ 1	НААУ 2
Среднее количество тактов обучения	7590	6905	862	925
Стандартное отклонение (SD)	972	735	176	115

3. На примере программной модели мобильного робота Pioneer 3-DX с целевой функцией избежания столкновения с препятствиями показана возможность применения разработанной нечеткой системы для управления объектами этого типа. Сравнение полученной системы управления с другими системами не проводилось из-за отсутствия общепринятых тестовых задач для мобильных роботов.

Основные результаты работы

В данной диссертационной работе представлен подход к построению систем автономного адаптивного управления на основе теории нечетких множеств.

Основные результаты работы:

- 1) разработано обобщенное описание системы ААУ на основе теории нечетких множеств;
- 2) разработаны методы синтеза подсистем системы ААУ на основе нечеткой логики, а именно:
 - представление (кодирование) алгоритма работы системы управления в виде входных и выходных нечетких множеств и нечетких правил управления,
 - метод автоматического разбиения множеств значений входных и выходных переменных системы на нечеткие множества, отражающие пространственные закономерности во входных и выходных данных системы,
 - метод автоматической генерации нечетких правил управления с использованием автоматической кластеризации методом вычетов, при

помощи которых описываются функциональные взаимосвязи между входами и выходами системы;

- метод коррекции степеней адекватности правил управления, позволяющий менять влияние каждого правила на процесс управления, адаптируя алгоритм управления к изменениям свойств объекта управления и среды;

3) на основе разработанных методов создана модель прикладной системы управления в виде модуля для системы Designer4GN на языке С#, которая была отлажена на задачах балансирования перевернутым маятником, стабилизации углового движения космического аппарата и управления мобильным роботом;

4) проведены несколько серий компьютерных экспериментов с разработанной системой управления и моделями объектов управления:

- на примере системы балансирования перевернутым маятником нечеткая система ААУ показала высокую скорость обучения, которая на 9%-115% выше по сравнению с системами, построенными на основе обучения с подкреплением, при равных критериях качества,
- на примере системы стабилизация углового движения космического аппарата нечеткая система ААУ продемонстрировала существенное сокращение (примерно в 8 раз) скорости обучения, по сравнению с существующими системами ААУ,
- на примере системы управления мобильным роботом была продемонстрирована возможность применения разработанной системы для эффективного управления объектами этого типа.

Публикации

1. Жданов А. А., Караваев М. В. Применение нечеткой логики в имитационной системе автономного адаптивного управления. Труды Института Системного Программирования Российской Академии Наук: Том 3. – М.: ИСП РАН, 2002, с. 119-135.
2. Караваев М.В., Жданов А.А. Применение нечеткой логики в системах автономного адаптивного управления. Сборник материалов всероссийской научно-технической конференции "Наука – производство – технологии – экология": Том 1. Киров: Изд-во ВятГУ, 2002, с. 13-14.

3. Alexander Zhdanov, Maxim Karavaev and Helen Maklakova, Claire Medigue, Michel Sorine. Simulation of control mechanisms in the cardio-vascular system. French-Russian A.M. Liapunov Institute for Applied Mathematics and Computer Science. Transactions. Vol. 4. Pp. 233-245. Moscow. 2003.
4. Караваев М. В. Применение нечеткой логики в имитационной системе автономного адаптивного управления. Труды Института Системного Программирования Российской Академии Наук: Том 7 (под ред. А.А.Жданова). – М.: ИСП РАН, 2004, с. 41-53.
5. Караваев М.В. Правила формирования связей между нейроноподобными элементами в системах автономного адаптивного управления. С. 102-108. Сборник научных трудов Всероссийской научно-технической конференции Нейроинформатика-2004: Часть 2. М.: МИФИ. С. 102-108.
6. 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.
7. А.А. Жданов, А.Е. Устюжанин, М.В. Караваев, Д.Б. Липкевич. 4GN – инструмент для разработки нейроноподобных адаптивных систем управления на основе метода автономного адаптивного управления. Сборник научных трудов Всероссийской научно-технической конференции Нейроинформатика-2005: Часть 1. М.: МИФИ. С. 203-209.
8. Жданов А.А., Караваев М.В. Разработка адаптивной системы управления мобильным роботом с применением нечеткой логики. Всероссийская научно-техническая конференция "Наука – производство – технологии – экология". Сборник материалов: Том 1. Киров: Изд-во ВятГУ, 2005, с. 34-36.
9. М.В. Караваев. Применение нечеткой логики в системах автономного адаптивного управления. Труды Международных научно-технических конференций «Интеллектуальные системы» (AIS'05) и «Интеллектуальные САПР» (CAD-2005). Научное издание в 4-х томах. - М.: ФИЗМАТЛИТ, 2005.
10. М.В. Караваев, А.Е. Устюжанин, А.А. Жданов. 4GN – программный инструмент для проектирования интеллектуальных систем управления. Труды Международных научно-технических конференций «Интеллектуальные системы» (AIS'05) и «Интеллектуальные САПР» (CAD-2005). Научное издание в 4-х томах. - М.: ФИЗМАТЛИТ, 2005.