

Принятие решений в автономных адаптивных системах управления, основанное на выявлении закономерных последовательностей действий

С. А. Бондаренко, А. А. Жданов, Б.М. Магомедов

Поддержано проектом РФФИ № 03-01-00323

Аннотация

Для общей схемы методологии Автономного Адаптивного Управления (ААУ) ставится задача построения удаптивного управления на основе выявления и использования регулярностей в последовательностях действий, совершаемых объектом управления, что формально является задачей обнаружения регулярностей над регулярностями, или знаний над знаниями. Предлагается два подхода к решению задачи – программно-прагматический и нейросетевой с использованием специальных моделей нейронов. Приводятся первые результаты моделирования данных подходов на примере системы адаптивного управления для мобильного робота.

Abstract

The paper describes a problem of detection of regularities in prehistory of a controlled object activity and using of the regularities in adaptive control system. Authors work in framework of the Autonomous Adaptive Control (AAC) methodology. Three modes of control are introduced in relation with regularities using. We elaborated two at least approaches for the problem solving: a program approach and a neural network one, which is based on special neural models. For application of the approaches we used a program model of the mobile robot Gnome № 8. The first results of simulation are presented.

Настоящая работа развивает метод *Автономного Адаптивного Управления (ААУ)* [1-3], представляющий собой концептуальную модель нервной системы, при этом на основе метода ААУ можно строить эффективные адаптивные управляющие системы (УС) для прикладных объектов. Согласно методологии ААУ, в биологических и искусственных системах управления можно различать несколько уровней механизмов принятия решений. Так в работах [4-7] показана возможность существования как минимум трех таких механизмов. В частности, механизм принятия решений *1-го уровня* состоит в анализе конкретного текущего состояния, в просмотре эмпирически накопленной базы знаний и поиске в ней лучшего (по определенным критериям) из вариантов состояний, которые управляющая система может вызвать совершением своих действий. Механизм управления *2-го уровня* связан с выявлением в предыстории и использованием при принятии решений неслучайных последовательностей выходных действий УС – моделей поведения. При обнаружении таких последовательностей принятие решения может осуществляться без применения 1-го механизма для выбора каждого очередного шага, но просто с помощью последовательного воспроизведения шагов найденной последовательности с одновременным контролем некоторых промежуточных результатов. Механизм принятия решений *3-го уровня* может быть связан с выбором лучшей из нескольких моделей поведения, совершение которых допустимо в текущей ситуации. Имеются и другие возможные механизмы принятия решений. Примеры таких механизмов принятия решений

можно наблюдать в биологических системах. Настоящая работа отражает результаты первых попыток моделирования 2-го из указанных механизмов принятия решений.

Как пример, рассмотрим программную модель мобильного робота «Гном №8», целевая функция которого состоит в выработке стереотипов поведения при объезде препятствий. Нейроподобная управляющая система робота [8,9] разработана в ИСП РАН и ИОНТ РАН в соответствии с принципами методологии ААУ. Робот эффективно самообучается объезжать единичные препятствия без столкновений, именно: а) вырабатывает «понимание» того, что сталкиваться с препятствием не следует, б) устанавливает зависимости (знания) между наблюдаемыми им ситуациями, своими действиями и их результатами, в) приобретает умение пользоваться этими знаниями для принятия решения на каждом шаге управления. Этот результат соответствует 1-му из описанных выше механизмов принятия решений, и его необходимо дополнить 2-м механизмом – умением автоматически обнаруживать повторяющиеся последовательности действий (модели поведения) и пользоваться ими в процессе принятия решений. Эффект циклического повторения принимаемых управляющей системой решений обусловлен, по-видимому, использованием дискретного представления действий. В случае с мобильным роботом такие циклы возникают, например, при попадании робота в углы ограждения полигона и при наезде на стенку, а также при движении по коридору. Так, циклическое «топтанье» робота в углу или у стенки является примером нежелательного заикливания, а поступательное, состоящее из повторяющихся однотипных шагов движение по прямому коридору есть пример полезного заикливания.

В случае обнаружения повторяющейся последовательности, управляющая система (УС) может при помощи *Подсистемы Эмоций* оценить, полезна или не полезна данная последовательность, и принять решение об автоматическом продолжении или прекращении такого цикла соответственно. Например, пусть в УС робота сформированы образ приближения к цели и образ удаления от цели, имеющие, соответственно, положительную и отрицательную эмоциональные оценки. В случае обнаружения заикливания мобильного робота у стенки или в углу, образ приближения к цели не будет распознаваться, положительная эмоциональная оценка не будет получена роботом, и тем самым УС может определить нежелательность данной закономерности в поведении. УС может заставить робота выйти из цикла, совершив некоторое, например, случайное действие. В случае поступательного движения по коридору, и распознавания образа периодически повторяющихся движений, будет распознан и образ приближения к цели с положительной оценкой. Поэтому такая модель поведения может быть принята к исполнению. Теперь УС может предугадывать момент следующего поворота, и принимать решения просто по найденной модели, не анализируя ситуацию каждый раз подробно.

Прекращение пользования моделью поведения можно осуществлять, опираясь также на показания *Подсистемы Эмоций*. Например, пусть робот, продвигаясь периодически повторяющимися шагами вдоль коридора по направлению к цели, проходит точку максимального сближения с целью и начинает удаляться от нее. В этой ситуации начнет распознаваться образ удаления от цели, имеющий отрицательную эмоциональную оценку, что явится основанием для прекращения выполнения циклических действий и необходимости возвращения к 1-му механизму принятия решений.

Возможны разные подходы к решению задачи обнаружения повторяющихся последовательностей команд, здесь мы представляем два из таких решений. Первое решение связано с применением формального алгоритмического метода, второе - состоит в применении специально разработанной нейроподобной сетевой структуры.

В общей схеме системы ААУ (рис. 1) *объект управления (ОУ)* взаимодействует с окружающей средой посредством датчиков и исполняющих органов, в управляющую систему

входят блок формирования и распознавания образов (ФРО), база знаний, аппарат эмоций и блок принятия решений.

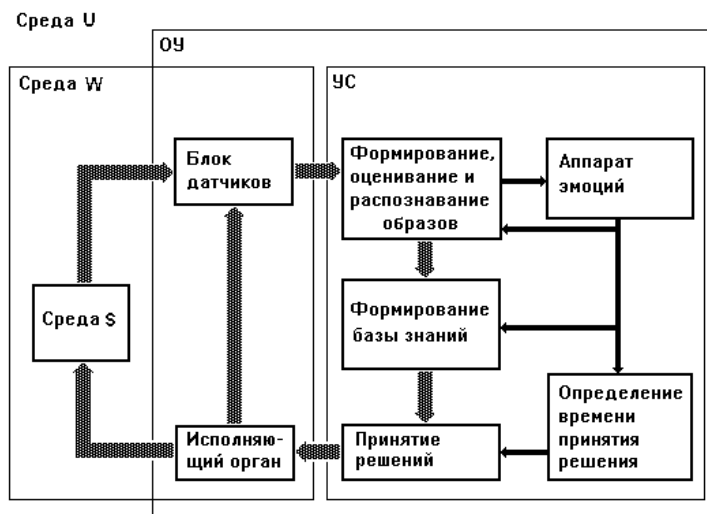


Рис. 1. Общая схема системы Автономного Адаптивного Управления.

Для формирования искомых образов протяженных пространственно - временных событий, которые отражали бы повторяющиеся последовательности выходных действий (такие образы формально не отличаются от образов других пространственно-временных событий), необходимы сенсоры, приносящие данные о действиях исполнителей. Будем называть такие сенсоры «сенсорами актуаторов». В живых организмах таким сенсорам соответствуют рецепторы, располагающиеся на мышцах и суставах, и регистрирующих их состояние и положение.

Пусть конкретный актуатор a_i в момент времени t может совершить только одно действие с номером v_i , $v_i = 0, 1, 2, \dots, \mu_i$. Пусть в каждый момент времени t на сенсоры актуаторов поступают данные о том, какие именно действия были совершены всеми актуаторами. Пусть эти данные представлены общим бинарным вектором, разбитым на секции a_i , каждая из которых закреплена за одноименным актуатором a_i , а «1», стоящая в позиции номер v_i секции a_i , указывает, что данный актуатор в этот момент времени совершил действие номер v_i . Временная последовательность таких векторов представляет историю действий данного объекта управления в виде временной диаграммы. На рис. 3 показан пример такой временной диаграммы, отражающей историю действий некоторого объекта управления, имеющего 2 актуатора, каждый из которых может совершить одно из 3-х действий, время движется вправо, крайний правый вектор соответствует последнему (текущему) моменту времени.

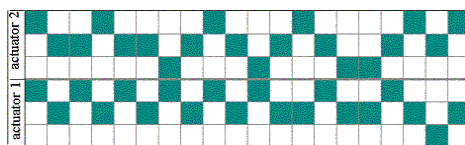


Рис. 3.

Сохраненная в памяти УС временная диаграмма активности системы, есть «память совершенных действий». Трудно хранить в памяти такую временную диаграмму по всему времени жизни УС, поэтому в методологии ААУ предполагается организация такой памяти в

виде ассоциативно связанных образов, обобщающих эти данные, при ограничении и уменьшении точности данных со временем.

Пусть история «совершенных действий» сохраняется точно на некотором скользящем временном окне. Анализируя временную диаграмму активности системы -«память совершенных действий», можно выявлять закономерности некоторых классов. Одним из простых классов является класс периодичностей. Если речь идет о сравнительно больших периодах, и при этом допускается аддитивность периодических компонент, данные зашумлены, либо имеют пропуски, то возможно применение известных методов выделения скрытых периодичностей, таких, как методы Бюй-Балло, Уиттекера, Шустера, метода максимальной энтропии, метода максимального правдоподобия, Фурье-анализа и т.п. Однако в нашей работе мы стремились к упрощению ситуации, чтобы ускорить процесс моделирования всей системы управления. Опишем два рассмотренных нами подхода к решению задачи и первые полученные результаты.

Для упрощения задачи анализа временной диаграммы действий отдельно анализировалась активность каждого из исполнителей. Периодичность с периодом T считалась закономерной, если она повторилась подряд заданное число n раз. Добавление в рассмотрение производной активности актуатора позволяет находить более сложные закономерности, такие как закономерность монотонности с постоянной производной. Примеры закономерностей периодичности и монотонности изображены на рис. 4.

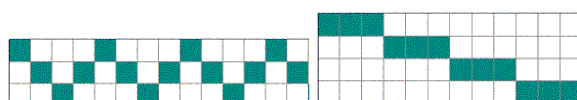


Рис. 4.

Алгоритм разработанной программы поиска периодичностей ActivityAnalyzer показан на рис. 5. После получения запроса от УС программа определяет текущее время по таймеру, и

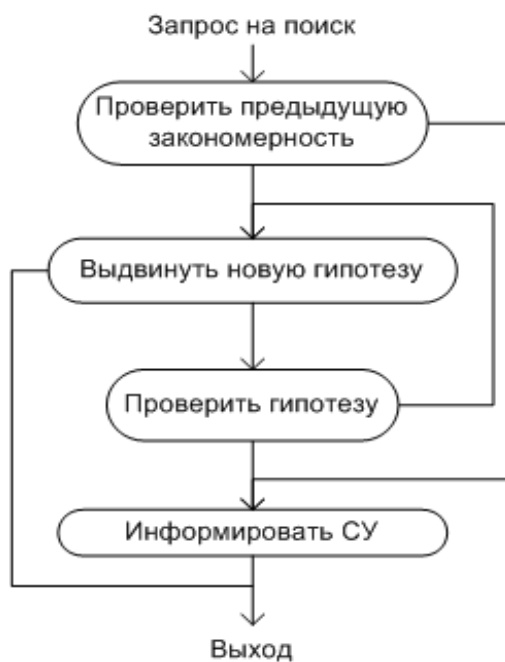


Рис. 5.

запрашивает в памяти активность за время от предыдущего запроса до текущего. Активность преобразуется в сжатое внутреннее представление в виде последовательностей специальных образов, и запоминается. После нахождения закономерности посылается сообщение о том, что закономерность найдена и предоставляются ее характеристики.

При внутреннем представлении диаграммы активности программа ActivityAnalyzer разбивает активность каждого из актуаторов на специальные образы - участки постоянной монотонности и выпуклости. Каждый такой образ характеризуется типом: парой {тип монотонности, тип выпуклости} и числовыми параметрами, такими как длина, и значения активности на концах участка. Примеры образов изображены на рис. 6. Период ищется в виде

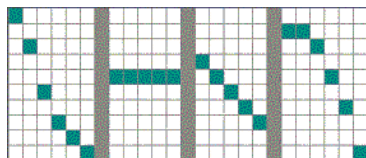


Рис. 6.

последовательности таких образов. Система запоминает предыдущие результаты поиска, и в случае, если при предыдущем вызове период был найден, проверяет, не продолжается ли этот период. Если период не повторился, то система начинает поиск заново. При этом, сначала на основании последовательности типов образов выдвигается гипотеза о возможной длине периода, а затем осуществляется проверка гипотезы с учетом количественных параметров образов. Если гипотеза не подтвердилась, выдвигаются новые гипотезы до выполнения некоторого правила останова. При проверке повторения последовательности образов используется условие равенства двух образов. Образы считаются равными, если они одного типа и их числовые параметры близки в смысле некоторой метрики. Это вносит гибкость в алгоритм поиска периода и позволяет учитывать, например, искажения периода.

Характеристиками результата поиска являются длина периода в тактах времени и число образов, которые его составляют, указание на то, новый ли это период, или тот же, что и при предыдущем поиске, а также последовательность образов, составляющих период.

На рис. 8 показаны примеры как простой, так и достаточно сложной периодичности, найденной программой ActivityAnalyzer, а также выявленного регулярного сигнала с переменным периодом.

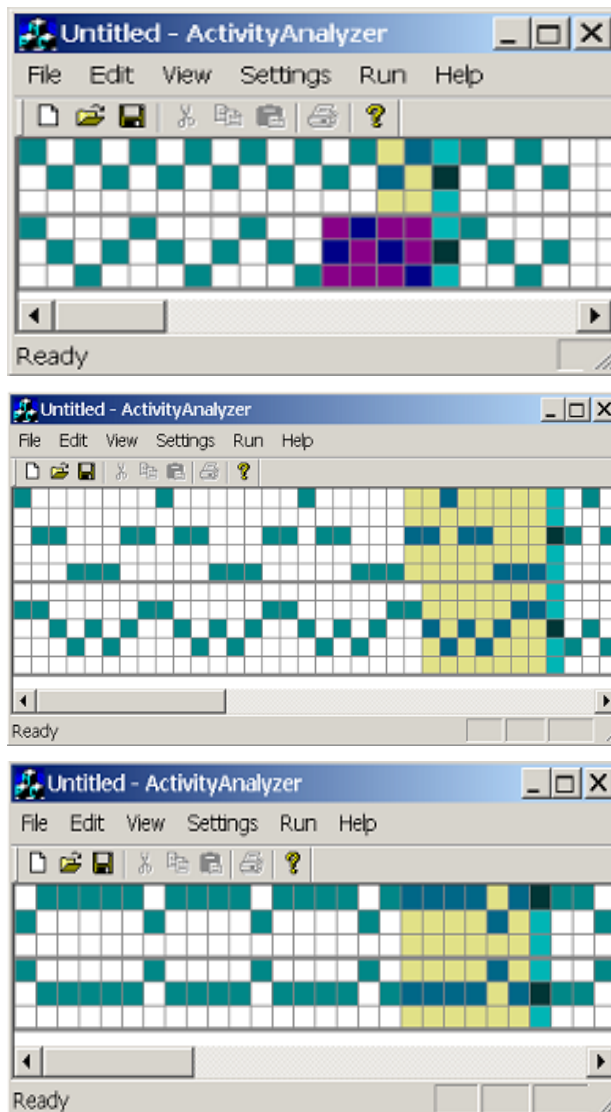


Рис. 8.

Авторы системы ААУ стремятся также к ее реализации в нейроподобном виде. Для этой цели разработаны специальные модели нейронов [10,11], назначением которых является автономное обнаружение коррелирующих компонент входного бинарного вектора (автоматическая классификация) с последующим распознаванием. Найденная совокупность коррелирующих компонент запоминается в нейроне и интерпретируется как *образ*, отражающий некоторую неслучайную, повторяющуюся ситуацию в системе «среда-объект управления-УС». Нейрон, сформировавший образ, может его распознавать. Одному нейрону в сети ААУ соответствует один образ. Сеть рассматривается как семантическая. Нейроны в системе ААУ имеют временные задержки, которые соответствуют синаптическим задержкам у биологических нейронов. При нерегулярном соединении нейронов в сеть, когда входные сигналы проходят на пути от сенсоров через разное число нейронов, появляется возможность работы с образами пространственно-временных явлений, в том числе – с образами причинно-следственных связей.

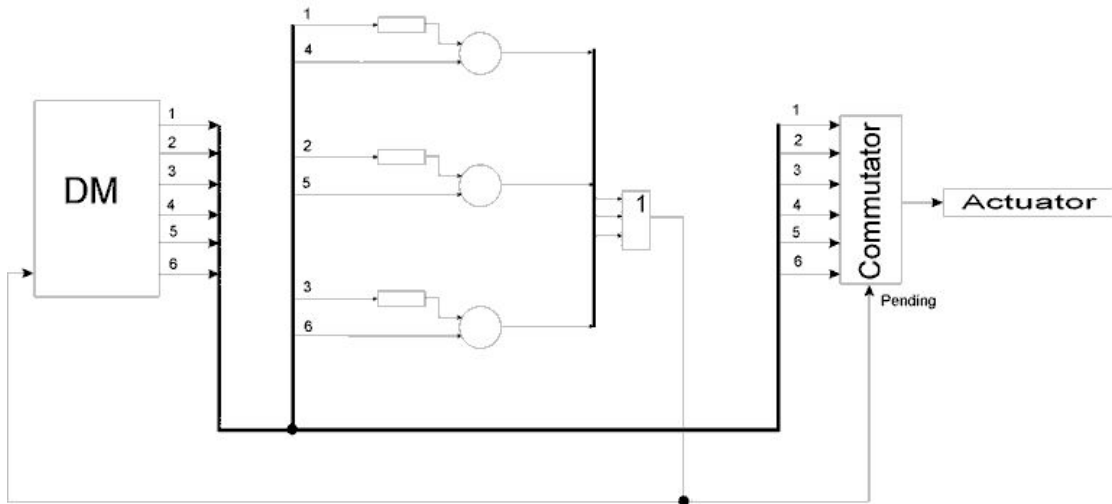


Рис. 12.

Для решения задачи поиска и использования периодических последовательностей действий, нами были разработаны специальные сетевые конструкции для блока ФРО из наиболее простых из указанных моделей нейронов, именно, нейрона, описанного в [1,3]. Для примера использовалась та же компьютерная модель мобильного робота. Эти нейроноподобные конструкции позволяли УС распознавать некоторые нежелательные повторяющиеся действия, связанные с заикливанием робота в углах ограждения, либо у стенок. Для обнаружения повторений использовались конструкции из двух нейронов, в которых один из нейронов играл роль элемента задержки сигнала. Для обнаружения некоторого числа повторяющихся комбинаций следует в блоке ФРО (при использовании таких простых моделей нейронов) задать столько таких конструкций, сколько гипотез необходимо проверить. В случае с мобильным роботом, оказалось достаточным ввести в блок ФРО всего 3 таких конструкции (рис. 12), которые реагировали на повторяющиеся оследовательности из двух действий, приводящих к заикливанию в углах и у стенок, именно: "вперед - назад", "вперед направо - назад направо" и "вперед налево – назад налево". Это позволило УС мобильного робота сформировать образы заикливания, распознать их и при распознавании давать сигнал на совершение случайного действия, что приводило к выходу из заикливания.

Рассмотренные в статье предложенные подходы позволяют управляющей системе робота обнаруживать заикливания, принимать их или выходить из них, что соответствует 2-му механизму принятия решений в методе ААУ. При этом программный подход обладает на настоящем этапе большими функциональными возможностями, чем нейросетевой. Однако нейросетевой подход также эффективно справляется с задачей и более бионичен. В последующих работах предусматривается развитие обоих подходов и переход к моделированию механизмов управления 3-го уровня.

Список литературы

1. Жданов А.А. Об одном имитационном подходе к адаптивному управлению//Вопросы кибернетики. Научный совет по комплексной проблеме «Кибернетика» РАН. Вып. 2. М.,1996.
2. Жданов А.А. Метод автономного адаптивного управления, Известия Академии Наук. Теория и системы управления, 1999, № 5, с. 127-134.

3. Жданов А.А. Накопление и использование информации при управлении в условиях неопределенности.// Информационная технология и численные методы анализа распределенных систем. Сб. науч. тр. Ин-та физ.-тех. проблем. М., 1992. С. 112-133.
4. Жданов А.А., Г.Я. Кантор, А.Б. Эфрон, И.Г. Новикова. Построение гомеостатического автомата с конечным количеством входных переменных. (депонированная работа) ВИНТИ "Депонированные научные работы", 1984, N2, стр.108, 336.
5. Zhdanov A.A. A principle of Pattern Formation and Recognition.// Pattern Recognition and Image Analysis vol.2, N3,1992, 249-264. (ISSN: 1054-6618). (on p. 263).
6. Жданов А.А. Принцип автономного адаптивного управления. Докторская диссертация, 1995 г. М., ВЦ РАН. (на стр. 39-41).
7. Жданов А. А. Моделирование высшей нервной деятельности // Наука и жизнь. 2000, № 1, с. 58-64, №2, с. 14-16. (на стр. 62).
8. Жданов А.А., Крыжановский М.В. Преображенский Н.Б. Нейронная адаптивная система управления. Труды международной конференции "Интеллектуальные и многопроцессорные системы" IMS'2002., С. 115-118. Казивели.
9. Жданов А.А., М.В. Крыжановский, Н.Б. Преображенский. Бионическая интеллектуальная автономная адаптивная система управления мобильным роботом. // Мехатроника. №1, 2004 (принята редакцией).
10. Zhdanov A.A., A.V. Ryadovikov. Neuron Models in the Autonomous Adaptive Control Method//Optical Memory and Neural Network, Allerton Press, Inc., Vol. 9, No 2, 2000, pp. 115-132.
11. Рядовилов А.В., Жданов А.А., О некоторых формальных моделях нейронов. // Сб. научн. тр. Всероссийской научн.-техн. конференции "Нейроинформатика-99", ч. 1. - М.: МИФИ. 1999. С. 202-211.