

Адаптивная система управления анимата в физической среде, основанная на теории функциональных систем

Путинцев Н.И.¹⁾, Исупов О.В.²⁾, Витяев Е.Е.^{1,2)}

¹⁾Новосибирский государственный университет, Новосибирск
putintsevnika@gmail.com, cypt@hotmail.com

²⁾Институт математики им. С. Л. Соболева СО РАН, Новосибирск, vityaev@math.nsc.ru

Аннотация. Система управления, основанная на теории функциональных систем и семантическом вероятностном выводе, была адаптирована для работы в физической среде. В систему была добавлена поддержка непрерывных сенсоров, а также способность учитывать продолжительность действий. Семантический вероятностный вывод был распространен на случай непрерывных сенсоров. Были проведены эксперименты в виртуальной и реальной среде. В ходе экспериментов была продемонстрирована способность системы решать задачи в физической среде.

1. Введение

Существует широкий круг работ, в которых исследуются адаптивные системы управления. Наиболее важными из них являются те, которые основаны на известных физиологических теориях. Одной из наиболее разработанных физиологических теорий является теория функциональных систем П.К. Анохина.

В настоящее время существует несколько адаптивных систем управления, разработанных на основе теории функциональных систем. Наиболее близкими к данной работе являются работы К.В. Анохина и В.Г. Редько и др. [1, 2] и А.А. Жданова [3].

В работах А.А. Жданова была разработана система управления, основанная на предложенной им вероятностной модели нейрона [3], которая близка по смыслу к разрабатываемому нами вероятностному семантическому выводу. На основе системы управления в [3] было разработано несколько контроллеров различного назначения и проведена серия успешных экспериментов на роботах, например [4]. В системе управления, разработанной в [3] необходимо задавать структуру системы управления заранее, что ухудшает её адаптивные свойства. Эта проблема частично решается применением эволюционных алгоритмов [5].

Система управления, разработанная в [1, 2], для формирования действий и прогнозирования результата использует нейронные сети. Обучение нейронных сетей происходит при помощи обучения с подкреплением и эволюционного алгоритма. Основным отличием нашей работы является использование метода семантического вероятностного вывода вместо нейронных сетей, потенциальные преимущества этого метода были описаны в работе [6].

Стоит отметить, что исследование различных адаптивных систем, основанных на биологических теориях, с помощью роботов, функционирующих в физической среде, является распространенной практикой. Ярким примером могут служить «устройства на основе мозга» (brain-based devices),

Система управления анимата в физической среде

исследованные в работах [7-9]. Наиболее известным семейством устройств на основе мозга является семейство роботов Darwin, разрабатываемое в The Neurosciences Institute in La Jolla, California с 1981 по 2007 годы в рамках проекта Nomad [8,9], с использованием опыта которого была построена экспериментальная часть нашей работы. В проекте Nomad также использовались роботизированные платформы с датчиками и перед системой управления ставились задачи, в которых она должна была проявлять адаптивное поведение. Основным отличием проекта Nomad от нашей работы, является то, что их системы управления симулируют частные аспекты адаптивного поведения, например, ориентирование на местности, в то время как мы, опираясь на теорию функциональных систем, стараемся воссоздать общие принципы.

Нами ранее была разработана система управления аниматом, основанная на теории функциональных систем и семантическом вероятностном выводе [10, 11]. Стоит также отметить, что в отличие от большинства других адаптивных систем, например [3, 8], структура которых задается заранее, наша система может перестраивать свою структуру в соответствии с условиями среды, в которой она находится.

Направлением развития нашей системы, которое выбрано в данной работе, является её адаптация к управлению аниматом в физической среде. Такая адаптация, с одной стороны, демонстрирует способность системы работать в среде, приближенной к той, в которой функционируют животные, с другой, позволяет испытать разработанный алгоритм на реальных задачах.

В ходе работы было выделено две задачи. Во-первых, необходимо было расширить возможности системы для эффективной работы в физической среде, в частности добавить поддержку непрерывных сенсоров, и поставить компьютерный эксперимент. Во-вторых, необходимо было распространить семантический вероятностный вывод на случай непрерывных сенсоров.

Системы управления, реализованные в предыдущих работах [12, 13], использовали бинарные датчики и функционировали в пошаговом режиме, т.е. все действия анимата имели одинаковую продолжительность. Однако большинство сенсоров, используемых в робототехнике, а также рецепторы живых организмов дают на выходе непрерывный сигнал. Кроме того, различные действия в реальном мире имеют различную протяженность во времени, что необходимо учитывать при выборе способа достижения цели. Система была дополнена возможностью использовать датчики с непрерывным вещественным сигналом, а также варьировать продолжительность своих действий при выборе способа достижения цели. При этом возможность использования семантического вероятностного вывода сохранилась.

Симуляция механики реального мира представляет собой трудную вычислительную задачу, которую, однако, не нужно выполнять, если в качестве анимата использовать робота. Поэтому для проведения экспериментов в физической среде была сконструирована роботизированная платформа. Платформа может нести на себе несколько видов датчиков и передвигаться по командам, получаемым по беспроводной связи.

Для демонстрации работоспособности в физической среде система управления робота должна была научиться находить блоки, разбросанные по

комнате. При этом изначально система не знала ни о назначении имеющихся у неё сенсоров, ни о возможных результатах своих действий. В ходе экспериментов система управления вырабатывала правила поведения, позволяющие эффективно обнаруживать блоки и подтвердила свою работоспособность.

2. Математическая модель системы управления

Математическая модель разработанной нами системы основана на наших предыдущих работах [11-13]. Для того чтобы система управления могла работать с непрерывными вещественными сигналами датчиков и учитывать продолжительность действий анимата, в модель были внесены некоторые изменения. Изменения в основном коснулись процесса формирования системы управления, при этом основные принципы её работы сохранились.

2.1. Основные принципы работы

Коротко рассмотрим основные принципы работы системы управления. Будем предполагать, что система функционирует в дискретном времени $t = 0, 1, \dots$

Анимат имеет набор сенсоров S_1, \dots, S_n , характеризующих состояние как самого анимата, так и внешней среды. Показания каждого сенсора S_i являются действительной величиной, которая может принимать значения в некотором интервале $VS_i = (v_i^{\min}, v_i^{\max})$, где v_i^{\min}, v_i^{\max} – минимальное и максимальное значения показаний сенсора S_i , соответственно.

Анимат располагает набором действий A_1, \dots, A_m . Действия анимата могут быть выполнены с различной продолжительностью Δt . Любое действие анимата, начатое в момент времени t_i и выполняемое с продолжительностью Δt , может приводить в моменты времени $t_i + 1, t_i + 2, \dots, t_i + \Delta t$ к какому-либо изменению среды, и, как следствие, к изменению показаний сенсоров. Для каждого действия задана максимально возможная продолжительность выполнения Δt_{\max} .

Поскольку анимат получает информацию об окружающей среде только через сенсоры, то, с его точки зрения, состояние системы в момент времени t может быть представлено вектором показаний сенсоров $V(t) = (v_1, \dots, v_n)$, где $v_i \in VS_i$ – показания i -го сенсора в момент времени t . Совокупность всех возможных состояний анимата обозначим как $SS = (VS_1 \times VS_2 \times \dots \times VS_n)$.

Вследствие того, что сенсоры имеют физические ограничения и их возможностей не всегда достаточно для того, чтобы однозначно охарактеризовать текущее состояние окружающей среды, одно и то же действие, выполненное при одинаковых показаниях сенсоров, может переводить систему в несколько различных состояний. Таким образом, действие A_i анимата может

Система управления анимата в физической среде

быть представлено как отображение, переводящее систему «анимат – внешняя среда» из одного состояния в другое с некоторой вероятностью:

$$A_i(\Delta t): (SS_i) \rightarrow (SS \times P),$$

где SS_i – подмножество состояний системы, в которых действие A_i осуществимо, Δt – продолжительность действия A_i , $SS \times P$ – множество пар вида (ss, p) , где $ss \in SS$ – конечное состояние, $p \in [0, 1]$ – вероятность перехода системы в одно из состояний этого множества из начального состояния $ss \in SS_i$ при выполнении действия A_i с продолжительностью Δt .

Разобьем интервалы показаний каждого сенсора VS_i на более мелкие интервалы $vs_{i1}, \dots, vs_{ik_i}$. Тогда вектором $ST = (vs_1, \dots, vs_n)$ можно представить множество состояний $V = (v_1, \dots, v_n)$ таких, что $v_1 \in vs_{11}, \dots, v_n \in vs_{n1}$.

Задачей анимата является достижение некоторой цели. Зададим цель через вектор интервалов показаний сенсоров:

$$G = (vs_1, \dots, vs_n)$$

Достижение цели аниматом совершается при помощи функциональных систем. Функциональная система FS^{rank} может быть представлена набором:

$$FS^{rank} = (G^{rank}, R_1, \dots, R_v, FS_1^{rank+1}, \dots, FS_d^{rank+1}),$$

где G^{rank} – цель, достижение которой является основной задачей FS^{rank} , R_1, \dots, R_v – правила, FS_i^{rank+1} – подчиненные функциональные системы, соответствующие некоторым правилам из R_1, \dots, R_v .

Правило R является преобразованием $ST_0 \xrightarrow[\hat{p}]{A(\Delta t)} ST_e$, где:

ST_0 – множество начальных состояний системы, заданное интервалами (vs_1^0, \dots, vs_n^0) ;

ST_e – множество конечных состояний, заданное интервалами (vs_1^e, \dots, vs_n^e) . Если правило принадлежит FS^{rank} , то $ST_e = G^{rank}$;

A – действие анимата;

Δt – длительность выполнения этого действия;

\hat{p} – оценка вероятности, с которой действие A , выполненное с продолжительностью Δt , переводит систему из начального состояния в конечное.

Оценка вероятности \hat{p} правила R рассчитывается следующим образом: если a – число случаев, когда анимат в состоянии ST_0 выполнил некоторое действие, а b – число случаев, когда действие $A(\Delta t)$ перевело его из состояния ST_0 в состояние ST_e , то $\hat{p} = \frac{b}{a}$. Пару (a, b) будем называть статистикой правила.

Функциональная система FS^{rank} может формировать подчиненные системы FS^{rank+1} , задачей которых является достижение подцелей, включенных в начальное состояние одного из правил FS^{rank} . Пример иерархии изображен на рис. 1.

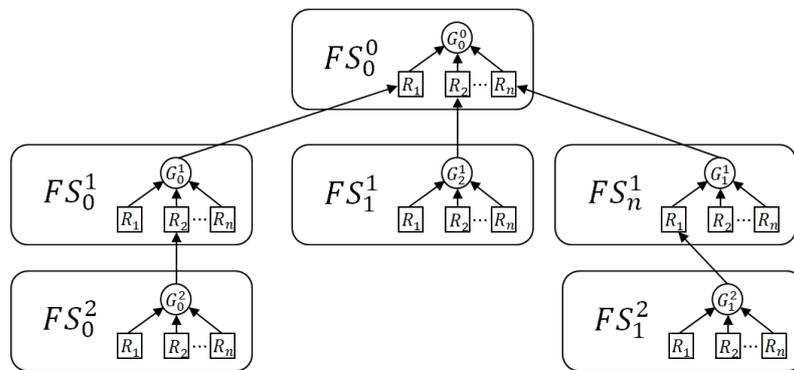


Рис. 1. Иерархия функциональных систем.

Для расчета оценки w_i^{rank} возможности достижения цели G^{rank} путем передачи управления подчиненной системе FS_i^{rank+1} используется тот факт, что цель G_i^{rank+1} является начальным состоянием некоторого правила вышестоящей системы $R_i^{rank} = G^{rank+1} \xrightarrow[\hat{p}_i]{A(\Delta t)} G^{rank}$. Это означает, что при достижении системой FS_i^{rank+1} своей цели G_i^{rank+1} , цель G^{rank} системы FS^{rank} может быть достигнута с оценкой вероятности \hat{p}_i . Для того, чтобы рассчитать w_i нам необходимо учесть как \hat{p}_i , так и оценку w_i^{rank+1} возможности достижения цели G_i^{rank+1} системой FS_i^{rank+1} .

Расчет w_i^{rank+1} производит подчиненная система FS_i^{rank+1} . Для этого она, аналогично FS^{rank} , производит оценку возможности достижения своей цели G_i^{rank+1} для доступных ей способов достижения цели (использование собственного правила или передача управления). Оценка w_i^{rank+1} принимается равной максимальной из оценок этих способов. Расчет ведется рекурсивно. Рекурсия завершается на функциональных системах самого нижнего уровня, у

Система управления анимата в физической среде

которых нет подчиненных систем и они вынуждены либо выбрать правило и вернуть его оценку, либо вернуть ноль, сообщая тем самым, что у них нет подходящего правила. После расчета w_i^{rank+1} и \hat{p}_i оценка w_i вычисляется как $w_i^{rank} = \gamma * \min(\hat{p}_i, w_i^{rank+1})$, где γ – заданный коэффициент дисконтирования, лежащий в интервале от 0 до 1. Коэффициент γ нужен для того, чтобы более короткий план, при одинаковой оценке вероятностей, имел преимущество: чем быстрее достигается цель, тем лучше. Типичное значение γ в компьютерных экспериментах составляло 0.95.

Как было показано выше, в вычислении оценки w^0 могут участвовать различные функциональные системы разного уровня иерархии, передающие управление друг другу. Каждой оценке w^0 достижения цели G^0 системой FS^0 верхнего уровня иерархии сопоставлена, кроме того, некоторая последовательность подчиненных друг другу функциональных систем FS^0, FS^1, \dots, FS^k и последовательность правил R^0, R^1, \dots, R^k , выбранных в этих функциональных системах для достижения их целей. Принятие решения осуществляется посредством выбора такой последовательности вызывающих друг друга функциональных систем и соответствующих правил, которая имеет максимальную оценку w^0 . Таким образом, система FS^0 выбирает одну из многих последовательностей правил R^0, R^1, \dots, R^k , которые затем выполняются в порядке от k до 0 соответствующими функциональными системами.

Если некоторая функциональная система для достижения своей цели воспользовалась правилом R и цель этой функциональной системы была достигнута, то правило подкрепляется и к его статистике добавляется положительный опыт, иначе правило наказывается и к статистике добавляется отрицательный опыт. В случае, если некоторая цель функциональной системой не достигнута, то вся система вызываемых функциональных систем перестраивается.

2.2. Формирование функциональных систем

Создание отдельной функциональной системы, её правил, а также выделение подцелей и создание иерархии происходит при помощи механизма формирования функциональных систем.

Создание функциональной системы FS^{rank} происходит в три этапа:

- Выделение цели G^{rank} ;
- Построение правил R_1, \dots, R_n достижения G^{rank} ;
- Выделение подцелей G^{rank+1} и формирование подчиненных систем F^{rank+1} .

Этот процесс продолжается, пока не будет достигнут заданный уровень подчиненности $rank_{max}$ или пока выделение подцелей не станет невозможно. Цель G^0 функциональной системы верхнего уровня задается извне. Цели G^{rank} для $rank > 0$ выделяются при помощи механизма генерации подцелей. Процесс время от времени перезапускается для обновления структуры системы управления.

Выделение подцелей и построение правил происходит при помощи статистических гиперкубов. Статистический куб GS^{A,ST_e} для некоторого множества состояний ST_e и действия A представляет собой многомерную таблицу размерностью $n+1$. Первые n её измерений соответствуют каждому из n сенсоров анимата. Число интервалов по каждому из этих измерений равно числу разбиений интервала показаний соответствующего сенсора, которое задается извне. Размерность $n+1$ служит для обозначения длительности действия. Количество ячеек этой размерности равно максимально возможной длительности действия Δt_{max} .

Каждая ячейка гиперкуба GS^{A,ST_e} соответствует конкретному вектору интервалов показаний сенсоров $ST = (vs_1, \dots, vs_n)$ и определенной продолжительности выполнения действия Δt . Ячейка содержит статистику (b, a) для правила $R = ST \xrightarrow[A(\Delta t)]{p} ST_e$. Заполнение ячеек происходит на основании истории событий.

Событием $E = (V_0, V_e, A, \Delta t)$ назовем единичный факт перехода системы из состояния с показаниями сенсоров V_0 в состояние с показаниями V_e при выполнении действия A длительностью Δt . Анимат регистрирует события и добавляет их в историю – упорядоченный во времени список событий. При рассмотрении события E в ячейку статистического гиперкуба $GS^{A,G}$, соответствующую показаниям V_0 и длительности действия Δt , заносится положительная статистика, если $V_e \in G$, и отрицательная в противном случае.

Рассмотрим процесс выделения подцелей для системы FS^{rank} . Для каждого правила R_i системы FS^{rank} анализируется его начальное условие ST_0 . Если ST_0 удовлетворяет приведенному ниже критерию формирования подцели, то для системы FS^{rank} создается подчиненная функциональная система FS^{rank+1} , целевым состоянием которой является ST_0 . Критерием формирования подцели является наличие в статистике гиперкуба GS^{A,ST_0} для одного из действий A анимата такой ячейки (b, a) , что:

$$\frac{b}{a} > x \wedge a > y, \text{ где}$$

x – пороговое значение оценки вероятности, а y – минимальное количество наблюдаемых событий. В ходе различных экспериментов были выбраны следующие значения $x = 0.4, y = 5$. Эти значения зависят от процесса обучения и его продолжительности во времени. Приведенные значения были выбраны на основе компьютерных экспериментов.

Для построения правил системы FS^{rank} с целью G^{rank} используются статистические гиперкубы $GS^{A_i, G^{rank}}$ для каждого доступного действия A_i анимата. Формирование правил происходит следующим образом:

Система управления анимата в физической среде

- Для каждого действия A_i фиксируется интервал показаний сенсоров ST , соответствующий некоторой ячейке $GS^{A_i, G^{rank}}$ и для всех возможных продолжительностей действия A_i по статистике рассчитываются оценки вероятности $p = \frac{b}{a}$. Затем выбирается такое действие A и его продолжительность Δt , чтобы величина оценки вероятности была наибольшей. Так для всех интервалов показаний сенсоров ST формируются правила вида $R = ST \xrightarrow{\frac{A(\Delta t)}{p}} G^{rank}$.
- Происходит обобщение интервалов показаний сенсоров, для которых отсутствует или собрано недостаточное количество статистики. Выдвигается гипотеза о том, что небольшое изменение показаний датчиков мало влияет на результат действия. Обобщение осуществляется следующим образом: из соседних ячеек извлекается и суммируется статистика достижения цели, затем на основании полученных величин рассчитывается оценка вероятности обобщенного правила $R = ST_0 \xrightarrow{\frac{A(\Delta t)}{p}} G^{rank}$, где ST_0 – вектор объединенных интервалов показаний сенсоров, соответствующий рассматриваемым ячейкам.
- Получившиеся правила обобщаются при помощи алгоритма семантического вероятностного вывода, который описан в работе [11]. Алгоритм позволяет избавиться от незначущих сенсоров и тем самым улучшить качество правил.

3. Эксперименты

Для проверки работоспособности предложенной системы управления были реализованы эксперименты в виртуальной и реальной среде.

Виртуальный мир (рис. 2) представлял собой ограниченную площадку, на которой случайным образом располагалась «пицца». Её «поедание» являлось основной целью анимата. «Пицца» и анимат были представлены окружностями с фиксированным радиусом. «Пицца» считалась съеденной, если окружность анимата по завершению какого-либо его действия пересекалась с окружностью «пицци».

Анимат располагал двумя непрерывными сенсорами: первый сообщал о расстоянии до ближайшего «съедобного» объекта, второй оценивал угол между направлением взгляда анимата и этим объектом.

Анимат мог выполнять три действия: движение в направлении взгляда, а также повороты налево и направо. Повороты и движение вперёд выполнялись с заданной скоростью. Время выполнения действий регулировалось системой управления в рамках предложенной модели.

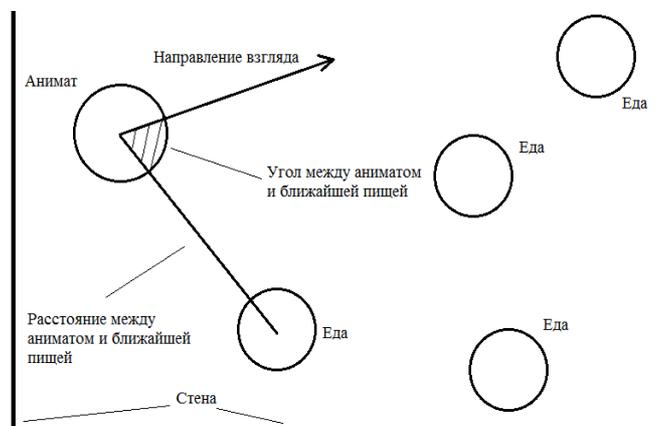


Рис. 2. Эксперимент в виртуальной среде.

В начале эксперимента анимат ничего не знал ни о назначении своих сенсоров, ни о возможных результатах действий. Методом «проб и ошибок» анимат обучался эффективно собирать «пищу». Реализация предложенной модели системы управления за 50-100 действий вырабатывала идеальное поведение: сначала анимат одним действием поворачивался к ближайшему объекту, а затем вторым действием приближался к нему и «поедал».

Для демонстрации способности системы управления работать в физической среде был придуман простой, но показательный эксперимент. Целью анимата в этом эксперименте так же являлся сбор «пищи».

В некоторой комнате размещались блоки, представляющие собой «пищу». В эту же комнату помещалась роботизированная платформа, управляемая предложенной нами системой. Целью системы управления являлся сбор «пищи». Пример расположения блоков и платформы показан на рис. 3.

На роботизированной платформе был установлен ряд сенсоров, которые давали системе управления частичную информацию о расположении блоков (рис. 3). Когда блок, находящийся в области видимости робота, оказывался достаточно близко, срабатывал специальный сенсор, и блок считался «собранным». Системе управления были доступны три действия: поворот направо, поворот налево и движение вперед. Действия выполнялись с фиксированной скоростью. Длительность выполнения действий регулировалась системой управления.

В начале каждого эксперимента система управления ничего не знала ни о расположении блоков, ни о назначении сенсоров, ни о возможных результатах своих действий. В ходе эксперимента система должна была самостоятельно обучиться эффективно обнаруживать блоки.

Эксперимент проводился в два этапа: обучение и контроль. На этапе обучения анимат в течение фиксированного времени исследовал окружающую среду, свои сенсоры, действия и их результаты. Во время обучения блоки и анимат располагались произвольным образом. При «сборе» блока робот автоматически откатывался назад на неизвестное системе управления

Система управления анимата в физической среде

расстояние и поворачивался на некоторый случайный угол. Расстояние, на которое откатывался робот, увеличивалось в процессе обучения.

На этапе контроля блоки и обученный анимат размещались показанным на рис. 3 образом. Когда робот «собирал» блок, эксперимент приостанавливался для удаления блока с поля. Контроль считался законченным, если робот собирал все блоки или если он не мог завершить сбор в течение десяти минут. Эффективность поведения анимата оценивалась временем, затраченным на сбор блоков.

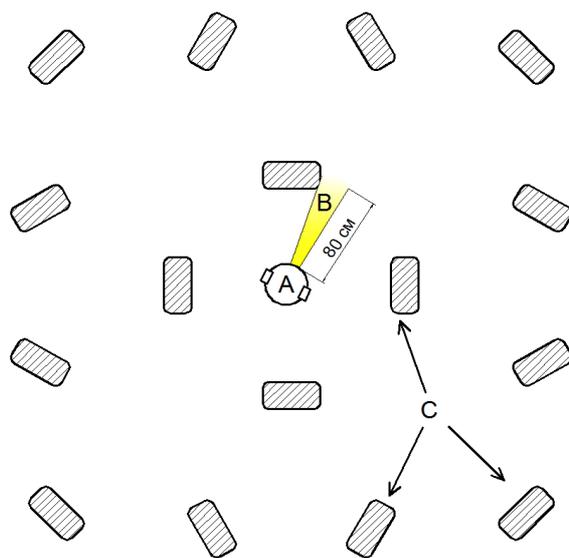


Рис. 3. Схема эксперимента в физической среде. А – роботизированная платформа; В – поле зрения платформы. Датчики позволяют видеть робота на 80 см. Блок считается съеденным, если оказался в поле зрения ближе 10 см; С – блоки «еды».

3.1. Роботизированная платформа

Для эксперимента на базе платы STM32VLDISCOVERY была разработана специальная роботизированная двухколесная платформа, оснащенная двумя наборами датчиков и Wi-Fi модулем для беспроводной связи с системой управления, запущенной на ПК.

Для ориентации в пространстве использовались два типа датчиков. Первый из них представлял собой инфракрасный дальномер, позволяющий измерять расстояние до объекта в пределах от 10 до 80 см. Второй тип – бинарный датчик, реагирующий на приближение к объекту ближе 10 см. Всего использовалось по два датчика каждого типа. Датчики фиксировались так, чтобы обеспечивать роботу поле зрения, изображенное на рис. 3. Датчики разных типов были закреплены один над другим и объединены в пары. Внешний вид платформы представлен на рис. 4.

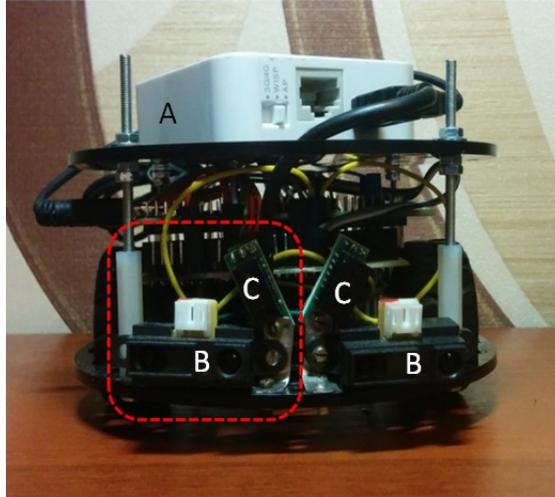


Рис. 4. Роботизированная платформа. А – Wi-Fi модуль; В – инфракрасные дальномеры; С – бинарные датчики дальности.

3.2. Результаты эксперимента в физической среде

В ходе экспериментов в систему управления был введен альтернативный способ оценки возможности достижения цели при передаче управления подчиненной функциональной системе.

В предыдущих работах [11,12] оценка возможности достижения цели G^{rank} системой FS^{rank} путем передачи управления подчиненной системе FS_i^{rank+1} происходила следующим образом: пусть w_i^{rank+1} – оценка возможности достижения цели G_i^{rank+1} подчиненной функциональной системой FS_i^{rank+1} , \hat{p} – оценка вероятности правила R^{rank} системы FS^{rank} , условие которого является целью системы FS_i^{rank+1} , тогда итоговая оценка возможности достижения цели при передаче управления системе FS_i^{rank+1} вычислялась как $w_i = w_i^{rank+1} * \hat{p}$. Произведение нескольких вероятностей давало малую величину. Такой способ оценки оказался неэффективным в физической среде. Из-за шумов в датчиках, ложных срабатываний, недочетов конструкции робота система управления верхнего уровня вырабатывала малоэффективные правила. Например, могло произойти срабатывание датчика близости без срабатывания дальномера, при этом подкреплялось следующее правило: двигаться вперед при слабом сигнале с дальномеров. Хотя это происходило очень редко, оценка такого правила часто оказывалась выше, чем рекурсивная оценка возможности достижения цели при передаче управления подчиненной системе с более эффективными правилами. Система управления могла довольно долго пользоваться таким правилом, прежде чем его оценка понижалась до уровня предпочтения подчиненных систем. Это серьезно замедляло процесс обучения.

Система управления анимата в физической среде

В данной работе введен альтернативный способ оценки. Итоговая оценка вычисляется как $w_i^{rank} = \gamma * \min(p_j, w_i^{rank+1})$, где γ – заданный коэффициент дисконтирования, лежащий в интервале от 0 до 1. Такой способ позволяет системе управления использовать подчиненные системы с хорошей оценкой достижения цели вместо правил систем верхних уровней иерархии с плохой оценкой. Коэффициент дисконтирования позволяет предпочитать хорошие правила верхнего уровня передаче управления подчиненным системам.



Рис. 5. Робот во время эксперимента.

Эксперименты проводились для различных версий системы управления: рандомизированной системы управления, выбирающей случайные действия и их продолжительность; системы управления, использующей оригинальный способ оценки возможности достижения цели [11,12], а также для системы управления с альтернативным способом оценки, приведенной выше. Для каждой системы проводилась серия из десяти экспериментов. Перед каждой серией экспериментов анимату отводился специальный период обучения, ограниченный тридцатью минутами. Продолжительность фазы контроля эксперимента ограничивалась десятью минутами. Производительность анимата оценивалась временем, затраченным на сбор всех блоков «еды». Результаты сведены в таблицу 1, робот во время эксперимента изображен на рис. 5.

Таблица 1. Результаты экспериментов в физической среде.

Тип системы управления	Среднее время	Минимальное время	Максимальное время
Рандомизированная	> 10 мин	> 10 мин	> 10 мин
Система с оригинальным способом оценки [11,12]	6 мин	4 мин 40 сек	> 10 мин
Система с предложенным выше альтернативным способом оценки	3 мин 40 сек	3 мин 10 сек	5 мин

Из таблицы 1 видно, что система управления способна эффективно собирать блоки, кроме того, введение нового способа прогнозирования улучшает её производительность.

4. Заключение

Система управления аниматом, основанная на теории функциональных систем, была адаптирована к работе в физической среде. В систему была добавлена возможность использования вещественных и непрерывных сигналов датчиков, а также учитывать и варьировать время выполнения действий. Был введен новый способ прогнозирования достижения цели, улучшающий работу в физической среде.

Система управления была протестирована при помощи виртуального эксперимента и эксперимента в физической среде. Для эксперимента в физической среде была разработана роботизированная платформа. В обоих экспериментах система показала хороший результат. В ходе экспериментов с роботом был выявлен и устранен ряд недостатков реализации алгоритма, а также продемонстрирована его способность решать задачи в физической среде.

Литература

1. Анохин К.В., Бурцев М.С., Зарайская И.Ю., Лукашев А.О., Редько В.Г. Проект «Мозг Анимата»: разработка модели адаптивного поведения на основе теории функциональных систем // Восьмая национальная конференция по искусственному интеллекту с международным участием. Труды конференции. М.: Физматлит, 2002. Том. 2. С. 781-789.
2. Red'ko V.G., Anokhin K.V., Burtsev M.S., Manolov A.I., Mosalov O.P., Nepomnyashchikh V.A., Prokhorov D.V. Project «Animat Brain»: Designing the animat control system on the basis of the functional systems theory // In: Butz, M.V., Sigaud, O., Pezzulo, G., Baldassarre, G (Eds.). *Anticipatory Behavior in Adaptive Learning Systems: From Brains to Individual and Social Behavior*. LNAI 4520, Berlin, Heidelberg: Springer Verlag, 2007. P. 94-107.
3. Жданов А.А. Метод автономного адаптивного управления // Известия академии наук. Теория и системы управления. 1999. № 5. С. 127-134.
4. Сыцко А.В. Система управления автономным мобильным роботом на основе адаптивного резонанса // Материалы XXIX Академических чтений по космонавтике. 2005. С. 93.
5. Земских Л.В. Возможности оптимизации системы автономного адаптивного управления с помощью генетических алгоритмов // М.: Препринт ИСП РАН, 2004.
6. Михиенко Е.В. Логическая спецификация нейронных сетей // Научная сессия МИФИ-2003. Нейроинформатика-2003 (V Всероссийская научно-техническая конференция). Теория нейронных сетей 2. С. 24-29.
7. Krichmar J.L., Edelman G.M. Brain-based devices for the study of nervous systems and the development of intelligent machines // *Artificial Life*. 2005. Vol. 11. No. 1-2. P. 63-78.
8. Krichmar J.L., Seth A.K., Nitz D.A., Fleischer J.G., Edelman G.M. Spatial navigation and causal analysis in a brain-based device modeling cortical-hippocampal interactions // *Neuroinformatics*. 2005. Vol. 3. No. 3. P. 197-222.

Система управления анимата в физической среде

9. Fleischer J.G., Krichmar J.L. Sensory integration and remapping in a medial temporal lobe model during maze navigation by a brain-based device // Journal of Integrative Neuroscience. 2007. Vol. 6. No. 3. P. 403-431.
10. Витяев Е.Е. Принципы работы мозга, содержащиеся в теории функциональных систем П. К. Анохина и теории эмоций П. В. Симонова // Нейроинформатика. 2008. Т. 3. № 1. С. 25-78. <http://www.niisi.ru/iont/ni/Journal/V3/N1/Vityaev.pdf>
11. Демин А.В., Витяев Е.Е. Логическая модель адаптивной системы управления // Нейроинформатика. 2008. Т. 3. № 1. С. 79 – 109. <http://www.niisi.ru/iont/ni/Journal/V3/N1/DeminVityaev.pdf>
12. Demin A.V., Vityaev E.E. Animal control system based on semantic probabilistic inference // Bulletin of the Novosibirsk Computing Center, volume of A.P. Ershov Institute of informatics systems, series: Computer Science, issue 24, 2006. P. 57-72.
13. Мухортов В.В., Хлебников С.В., Витяев Е.Е. Улучшенный алгоритм семантического вероятностного вывода в задаче 2-мерного анимата // Нейроинформатика. 2012. Т. 6. № 1. С. 50-62. <http://www.niisi.ru/iont/ni/Journal/V6/N1/MuhortovKhlebnikovVityaev.pdf>

Статья поступила 20 мая 2014 г.
После доработки 21 июля 2014 г.