

УДК 004.896:007.52

АДАПТИВНАЯ СИСТЕМА УПРАВЛЕНИЯ МОБИЛЬНЫМ АГЕНТОМ В ФИЗИЧЕСКОЙ СРЕДЕ, ОСНОВАННАЯ НА ТЕОРИИ ФУНКЦИОНАЛЬНЫХ СИСТЕМ

© 2014 г. Н.И. Путинцев¹, О.В. Исупов², Е.Е. Витяев^{1,2}

¹ Федеральное государственное автономное образовательное учреждение высшего образования «Новосибирский национальный исследовательский государственный университет», Новосибирск, Россия,
e-mail: putintsevnikita@gmail.com, cyrt@hotmail.com;

² Федеральное государственное бюджетное учреждение науки
Институт математики им. С.Л. Соболева
Сибирского отделения Российской академии наук, Новосибирск, Россия,
e-mail: vityaev@math.nsc.ru

Поступила в редакцию 26 мая 2014 г. Принята к публикации 31 октября 2014 г.

Нами ранее была разработана система управления мобильным агентом, основанная на теории функциональных систем и семантическом вероятностном выводе (Витяев, 2008; Демин, Витяев, 2008). В данной работе была проведена ее адаптация к управлению роботом 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 расширить семантический вероятностный вывод на случай непрерывных сенсоров. 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. 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 обучиться находить разбросанные в помещении блоки. Разработанный нами алгоритм позволил решить эту задачу и выработать набор правил для эффективного поиска блоков.

Ключевые слова: адаптивная система управления, мобильный агент, машинное обучение, теория функциональных систем.

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, разрабо-

танных на основе теории функциональных систем. Наиболее близкими к данной работе являются работы К.В. Анохина с соавт. (2002), В.Г. Редько с соавт. (Red'ko *et al.*, 2007) и А.А. Жданова (1999).

В работах А.А. Жданова была разработана система управления, основанная на предложенной им вероятностной модели нейрона (Жданов, 1999), которая близка по смыслу к разрабатываемому нами вероятностному семантическому выводу. На основе системы управления было

разработано несколько контроллеров различного назначения (Жданов, 1999) и проведена серия успешных экспериментов на роботах (например, Сыцко, 2005). В системе управления, разработанной Ждановым (1999), необходимо задавать структуру системы управления заранее, что ухудшает ее адаптивные свойства. Эта проблема частично решается применением эволюционных алгоритмов (Земских, 2004).

Система управления, разработанная К.В. Анохиным с соавт. (2002) и В.Г. Редько с соавт. (Red'ko *et al.*, 2007) для формирования действий и прогнозирования результата использует нейронные сети. Настройка параметров 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 Е.В. Михиенко (2003).

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), исследованные в работах J. Krichmar и G. Edelman (2005), Krichmar с соавт. (2005), J. Fleischer и J. Krichmar (2007). Наиболее известным семейством устройств на основе мозга является семейство роботов Darwin, разрабатываемое в The Neurosciences Institute in La Jolla, California с 1981 г. по 2007 г. в рамках проекта Nomad (Krichmar *et al.*, 2005; Fleischer, Krichmar, 2007), с использованием опыта которого была построена экспериментальная часть нашей работы. В проекте Nomad также использовались роботизированные платформы с датчиками, 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 мобильным агентом, основанная на

теории функциональных систем и семантическом вероятностном выводе (Витяев, 2008; Демин, Витяев, 2008). Стоит также отметить, что в отличие от большинства других адаптивных систем, структура которых задается заранее (Жданов, 1999; Krichmar *et al.*, 2005), 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 ее адаптация к управлению мобильным агентом 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 (Demin, Vityaev, 2006; Мухортов и др., 2012), использовали бинарные датчики и функционировали в пошаговом режиме, т. е. все действия агента 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 разную протяженность во времени, что необходимо учитывать при выборе способа достижения цели. Система была дополнена возможностью использования датчиков с непрерывным вещественным сигналом, а также варьирования продолжительности 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 агента использовать робота. Поэтому

для проведения экспериментов в физической среде была сконструирована роботизированная платформа. Она может нести на себе несколько видов датчиков и передвигаться по командам, получаемым по беспроводной связи.

В качестве эксперимента в физической среде роботизированная платформа, не имея какой-либо информации о назначении своих сенсоров и последствиях своих действий, должна была обучиться находить разбросанные в помещении блоки. Разработанный нами алгоритм позволил решить эту задачу и выработать набор для эффективного поиска блоков.

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ СИСТЕМЫ УПРАВЛЕНИЯ

Математическая модель разработанной нами системы основана на наших предыдущих работах (Demín, Vityaev, 2006; Демин, Витяев, 2008; Мухортов и др., 2012). Для того чтобы система управления могла работать с непрерывными вещественными сигналами датчиков и учитывать продолжительность действий агента, в модель были внесены некоторые изменения. Изменения в основном коснулись процесса формирования системы управления, при этом основные принципы ее работы сохранились.

Основные принципы работы

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

Агент имеет набор сенсоров S_1, \dots, S_n , характеризующих состояние как самого агента, 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.

Агент располагает набором действий A_1, \dots, A_m . Действия агента могут быть выполнены с различной продолжительностью Δt . Любое действие агента, 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 агент 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 агента 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 агента A_i может быть представлено как отображение, переводящее систему «агент–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 агента является достижение некоторой цели. Зададим цель через вектор интервалов показаний сенсоров:

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

Достижение цели агентом 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_1^{rank+1} – подчиненные функциональные сис-

темы, соответствующие некоторым правилам из R_1, \dots, R_n .

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, v_n^0) ; ST_e – множество конечных состояний, заданное интервалами (vs_1^e, \dots, v_n^e) . Если правило принадлежит FS^{rank} , то $ST_e = G^{rank}$, A – действие агента; 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 агент 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.

Для расчета оценки w_i^{rank} возможности достижения цели G^{rank} путем передачи управления подчиненной системе FS_1^{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 озна-

чает, что при достижении системой FS_1^{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_1^{rank+1} системой FS_1^{rank+1} .

Расчет w_i^{rank+1} производит подчиненная система FS_1^{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 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 $w_i^{rank} = \gamma \cdot \min(\hat{p}_i, w_i^{rank+1})$, 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,

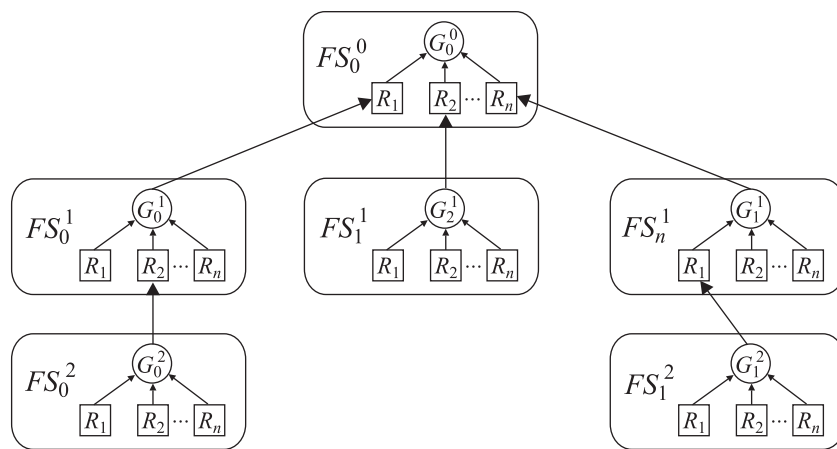


Рис. 1. Иерархия функциональных систем.

Fig. 1. The hierarchy of functional systems.

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.

Формирование функциональных систем

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

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

1. Выделение цели G^{rank} ;
2. Построение правил R_1, \dots, R_n достижения G^{rank} ;
3. 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 невозможным. 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 происходят 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 ее измерений соответствуют каждому из n сенсоров агента. 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 $R = ST \xrightarrow{\frac{A(\Delta t)}{\hat{p}}} ST_e$. Заполнение ячеек происходит на основании истории событий.

Событием $E = (V_0, V_e, A, \Delta t)$ назовем единственный факт перехода системы из состояния с показаниями сенсоров V_0 в состояние с показаниями V_e при выполнении действия A длительностью Δt . Агент 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 агента такой ячейки (b, a) , что:

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

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 агента. Формирование правил происходит следующим образом:

1. 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 $R = ST \xrightarrow{\frac{A(\Delta t)}{\hat{p}}} G^{rank}$.

2. 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 получившихся величин рассчитывается оценка вероятности обобщенного правила $R = ST_0 \xrightarrow{\frac{A(\Delta t)}{\hat{p}_i}} G^{rank}$, где ST_0 – вектор объединенных интервалов показаний сенсоров, соответствующий рассматриваемым ячейкам.

3. Получившиеся правила обобщаются при помощи алгоритма семантического вероятностного вывода, который описан в работе Демина и Витяева (2008). Алгоритм позволяет избавиться от незначущих сенсоров и тем самым улучшить качество правил.

ЭКСПЕРИМЕНТЫ

Для проверки работоспособности предложенной системы управления были реализова-

ны эксперименты в виртуальной и реальной среде.

Виртуальный мир (рис. 2) представлял собой ограниченную площадку, на которой случайным образом располагалась «пища». Ее «поедание» являлось основной целью агента. «Пища» и агент были представлены окружностями с фиксированным радиусом. «Пища» считалась съеденной, если окружность агента по завершении какого-либо его действия пересекалась с окружностью «пищи».

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

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

В начале эксперимента агент ничего не знал ни о назначении своих сенсоров, ни о возможных результатах действий. Методом «проб и ошибок» агент обучался эффективно собирать «пищу». Реализация предложенной модели системы управления за 50–100 действий способствовала выработке идеального поведения: сначала агент 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

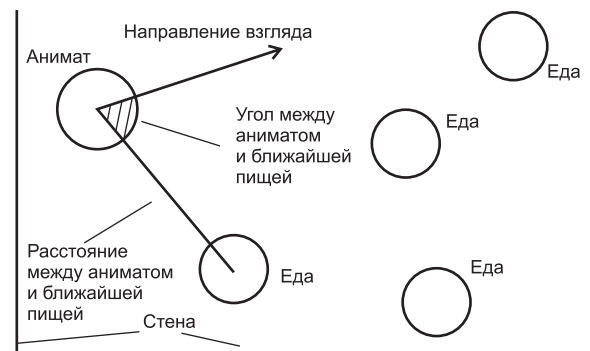


Рис. 2. Эксперимент в виртуальной среде.

Fig. 2. Experiment in the virtual environment.

придуман простой, но показательный эксперимент. Целью агента в этом эксперименте также 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.

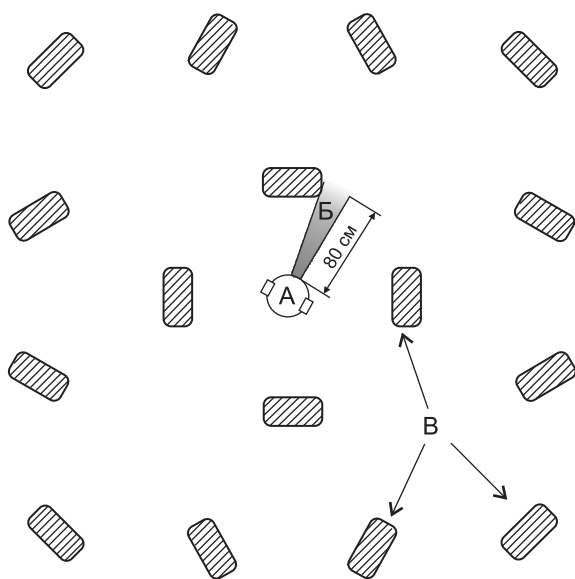


Рис. 3. Схема эксперимента в физической среде.

А – роботизированная платформа; Б – поле зрения платформы. Датчики позволяют роботу видеть на 80 см. Блок считается съеденным, если оказался в поле зрения ближе 10 см; В – блоки «еды».

Fig. 3. Experiment in the physical environment.

А – robotic platform; В – visual field of the robotic platform. Sensors allow robot to «see» in a range of 80 cm. A brick is considered to be eaten if it appears in the visual field closer than 10 cm to the robot; С – «Food» bricks.

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 агент в течение фиксированного времени исследовал окружающую среду, свои сенсоры, действия и их результаты. Во время обучения блоки и агент располагались произвольным образом. При «сборе» блока робот автоматически откатывался назад на неизвестное системе управления расстояние и поворачивался на некоторый случайный угол. Расстояние, на которое откатывался робот, увеличивалось в процессе обучения.

На этапе контроля блоки и обученный агент 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 агента оценивалась временем, затраченным на сбор блоков.

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 использовалось 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.

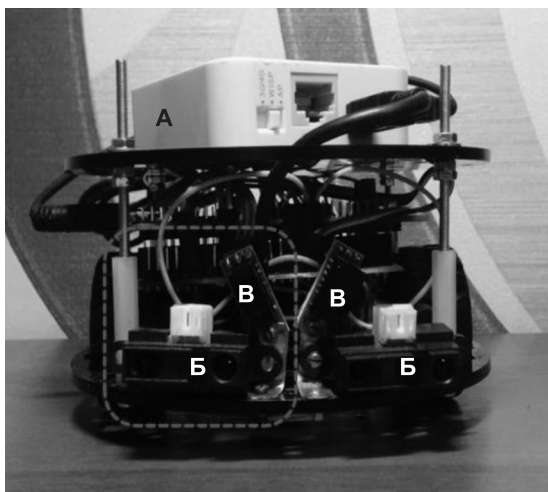


Рис. 4. Роботизированная платформа.

A – Wi-Fi модуль; Б – инфракрасные датчики; В – бинарные датчики дальности.

Fig. 4. Robotic platform.

A – Wi-Fi module; B – infrared proximity sensors; C – binary proximity sensors.

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 (Demin, Vityaev, 2006; Демин, Витяев, 2008) оценка возможности достижения цели G^{rank} системой FS^{rank} путем передачи управления подчиненной системе FS_1^{rank+1} происходила следующим образом: пусть w_i^{rank+1} – оценка возможности достижения цели G_i^{rank+1} , подчиненной функциональной системой FS_1^{rank+1} ; \hat{p} – оценка вероятности правила R^{rank} системы FS^{rank} , условие которого является целью системы FS_1^{rank+1} , тогда итоговая оценка возможности достижения цели при передаче управления системе FS_1^{rank+1} вычислялась как $w_i = w_i^{rank+1} \cdot \hat{p}$. 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 датчика дальности, 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.

В данной работе введен альтернативный способ оценки. Итоговая оценка вычисляется как $w_i^{rank} = \gamma \cdot \min(p_j, w_i^{rank+1})$, 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.

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 (Demin, Vityaev, 2006; Демин, Витяев, 2008), а также для системы управления с альтернативным способом оценки, приведенной выше. Для каждой системы проводилась серия из 10 экспериментов. Перед каждой серией экспериментов агенту отводился специальный период обучения, ограниченный 30 минутами. Продолжительность фазы контроля эксперимента ограничивалась 10 минутами. Производительность агента оценивалась временем, затраченным на сбор всех блоков «еды». Результаты сведены в таблицу, робот во время эксперимента изображен на рис. 5.

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



Рис. 5. Робот во время эксперимента.

Fig. 5. The robot during the experiment.

Таблица

Результаты экспериментов в физической среде

Тип системы управления	Время		
	среднее	минимальное	максимальное
Рандомизированная система с оригинальным способом оценки (Demin, Vityaev, 2006; Демин, Витяев, 2008)	> 10 мин	> 10 мин	> 10 мин
	6 мин	4 мин 40 с	> 10 мин
с предложенным выше альтернативным способом оценки	3 мин 40 с	3 мин 10 с	5 мин

ЗАКЛЮЧЕНИЕ

Система управления мобильным агентом, 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 возможность 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 алго-

ритма, а также продемонстрирована его способность решать задачи в физической среде.

БЛАГОДАРНОСТИ

Работа выполнена при финансовой поддержке интегративного проекта СО РАН № 136.

ЛИТЕРАТУРА

- Анохин К.В., Бурцев М.С., Зарайская И.Ю., Лукашев А.О., Редько В.Г. Проект «Мозг Агента»: разработка модели адаптивного поведения на основе теории функциональных систем // Восьмая национальная конф. по искусственному интеллекту с междунар. участием. Тр. конф. М.: Физматлит, 2002. Т. 2. С. 781–789.
- Anokhin K.V., Burtsev M.S., Zarayskaya I.U., Lukashov A.O., Red'ko V.G. Project «Agent Brain»: Development of adaptive behavior model based on functional systems theory // Vos'maja nacional'naja konf. po iskusstvennomu intellektu s mezhdunar. uchastiem. Trudi konferencii. M.: Fizmatlit, 2002. V. 2. P. 781–789. (In Russian).

- 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>
- Vityaev E.E. The principles of the brain from the Anokhin's theory of functional systems and P.V. Simonov's theory of emotions // *Nejroinformatika*. 2008. V. 3. No. 1. P. 25–78. <http://www.niisi.ru/iont/ni/Journal/V3/N1/Vityaev.pdf> (In Russian).
- Демин А.В., Витяев Е.Е. Логическая модель адаптивной системы управления // *Нейроинформатика*. 2008. Т. 3. № 1. С. 79–109.
- Demin A.V., Vityaev E.E. Logical model of the adaptive control system // *Nejroinformatika*. 2008. V. 3. No. 1. P. 79–109. (In Russian).
- Жданов А.А. Метод автономного адаптивного управления // *Изв. АН. Теория и системы управления*. 1999. № 5. С. 127–134.
- Zdanov A.A. Method of adaptive control // *Izvestija Akademii Nauk. Teorija i sistemy upravlenija*. 1999. No. 5. P. 127–134. (In Russian).
- Земских Л.В. Возможности оптимизации системы автономного адаптивного управления с помощью генетических алгоритмов. М.: Препринт ИСП РАН, 2004.
- Zemskih L.V. The possibility of autonomous adaptive control system optimization using genetic algorithms. M.: ISP RAS Preprint, 2004. (In Russian).
- Михиенко Е.В. Логическая спецификация нейронных сетей // *Науч. сессия МИФИ-2003. Нейроинформатика-2003 (V Всерос. науч.-техн. конф.)*. Теория нейронных сетей 2. С. 24–29.
- Mikheenko E.V. Logical specification of neural networks // *Scientific session of the Moscow Engineering Physics Institute-2003. Neuroinformatics-2003 (5th All-Russian Science and Technology Conference)*. Theory of neural networks. Nauch. sessija MIFI-2003. *Nejroinformatika-2003 (V Vseros. nauch.-tehn. konf.)*. Teorija nejronnyh setej 2. P. 24–29. (In Russian).
- Мухортов В.В., Хлебников С.В., Витяев Е.Е. Улучшенный алгоритм семантического вероятностного вывода в задаче 2-мерного агента // *Нейроинформатика*. 2012. Т. 6. № 1. С. 50–62. <http://www.niisi.ru/iont/ni/Journal/V6/N1/MuhortovKhlebnikovVityaev.pdf>
- Mukhortov V.V., Khlebnikov S.V., Vityaev E.E. An improved algorithm of the semantic probabilistic inference for the 2-dimensional agent problem // *Nejroinformatika*. 2012. V. 6. No. 1. P. 50–62. <http://www.niisi.ru/iont/ni/Journal/V6/N1/MuhortovKhlebnikovVityaev.pdf> (In Russian).
- Сыцко А.В. Система управления автономным мобильным роботом на основе адаптивного резонанса // *Матер. XXIX Акад. чтений по космонавтике*. 2005. С. 93.
- Sutsko A.V. The control system of autonomous mobile robot based on adaptive resonance // *Mater. XXIX Akad. chtenij po kosmonavtike*. 2005. P. 93. (In Russian).
- Demin A.V., Vityaev E.E. Animal control system based on semantic probabilistic inference // *Bul. of the Novosibirsk Computing Center, volume of A.P. Ershov Institute of Informatics Systems. Ser. Computer Science*. I. 24. 2006. P. 57–72.
- Fleischer J.G., Krichmar J.L. Sensory integration and remapping in a medial temporal lobe model during maze navigation by a brain-based device // *J. Integr. Neurosci.* 2007. V. 6. No. 3. P. 403–431.
- Krichmar J.L., Edelman G.M. Brain-based devices for the study of nervous systems and the development of intelligent machines // *Artificial Life*. 2005. V. 11. No. 1/2. P. 63–78.
- 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. V. 3. No. 3. P. 197–222.
- 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 // *Anticipatory Behavior in Adaptive Learning Systems: From Brains to Individual and Social Behavior / Eds M.V. Butz, O. Sigaud, G. Pezzulo, G. Baldassarre*. LNAI 4520, Berlin, Heidelberg: Springer Verlag, 2007. P. 94–107. <http://www.niisi.ru/iont/ni/Journal/V3/N1/DeminVityaev.pdf>

**ADAPTIVE CONTROL SYSTEM FOR A MOBILE AGENT
IN PHYSICAL ENVIRONMENT
BASED ON THE FUNCTIONAL SYSTEMS THEORY**

N.I. Putintsev¹, O.V. Isupov², E.E. Vityaev^{1,2}

¹ Novosibirsk State University, Novosibirsk, Russia,
e-mail: putintsevnikita@gmail.com, cypt@hotmail.com;

² Sobolev Institute of Mathematics SB RAS, Novosibirsk,
e-mail: vityaev@math.nsc.ru

Summary

Earlier we developed an adaptive mobile agent control system based on the functional systems theory and semantic probabilistic inference (Vityaev, 2008; Demin, Vityaev, 2008). In the present work, we extended the potential of the system by introducing the ability to control a robot in physical environment. On the one hand, this ability shows that the system can act in a real environment. On the other hand, it allows us to test the developed algorithm on actual physical problems. We identified two objectives. The first was to extend the capabilities of the system so that it could work effectively in the physical environment. In particular, it was necessary to add support for continuous sensors and conduct a simulated experiment. The second task was to extend the semantic probabilistic inference to the work with continuous sensors. We added the ability to use sensors with a continuous real signal and vary the duration of its actions while achieving the goal. The benefits of the semantic probabilistic inference were preserved. We constructed a robotic platform for experiments in the physical environment. The platform could carry several types of sensors and move executing commands received wirelessly. To show the ability to act in the physical environment the system was supposed to learn how to find bricks scattered around the room. Initially, the system did not know neither about the sensors available to it nor about the possible outcome of its actions. During the experiments, the control system generated rules for the effective detection of bricks and proved its efficiency.

Key words: adaptive control system, mobile agent, machine learning, functional systems theory.