

Ю.А. МАЗУР, А.А. ЖДАНОВ

Институт точной механики и вычислительной техники им. С. А. Лебедева РАН,
Россия, Москва
yuamazur@ipmce.ru

МЕТОД ФОРМИРОВАНИЯ ВРЕМЕННЫХ ОБРАЗОВ ДЛЯ СИСТЕМ АВТОНОМНОГО АДАПТИВНОГО УПРАВЛЕНИЯ

В статье предложен и исследован простой метод автоматического формирования временных образов для интеллектуальных систем управления. Показана способность метода отбирать только фиксированное число наиболее важных по заданному критерию образов. В основе метода лежит рейтинг-функция, включающая частоту встречаемости образов и смысловую важность образов для системы управления.

Ключевые слова: автономные адаптивные системы управления, формирование образов, кластеризация

Введение

В процессе конструирования сложных автоматизированных систем управления (СУ) все чаще приходится сталкиваться с объектами и свойствами окружающей среды, которые изменяются со временем, что значительно затрудняет создание детерминированных систем управления. Растет спрос на СУ, способные самостоятельно принимать решения в непредусмотренных условиях, в частности, это относится к мобильным и автономным роботам.

Одним из возможных подходов к созданию подобных адаптивных систем является метод автономного адаптивного управления (ААУ) [1-3]. Метод базируется на концептуальной модели нервных систем и призван воспроизводить естественные механизмы адаптации живых существ к изменяющимся условиям существования. В системе ААУ важную роль играет «подсистема эмоций» [4, 5], которая, в частности, обеспечивает «мотивацию» к активному поведению объекта управления, старающегося максимизировать интегральную эмоциональную оценку своих текущего и будущих состояний. На рис. 1 представлена структура СУ в соответствии с методом ААУ.

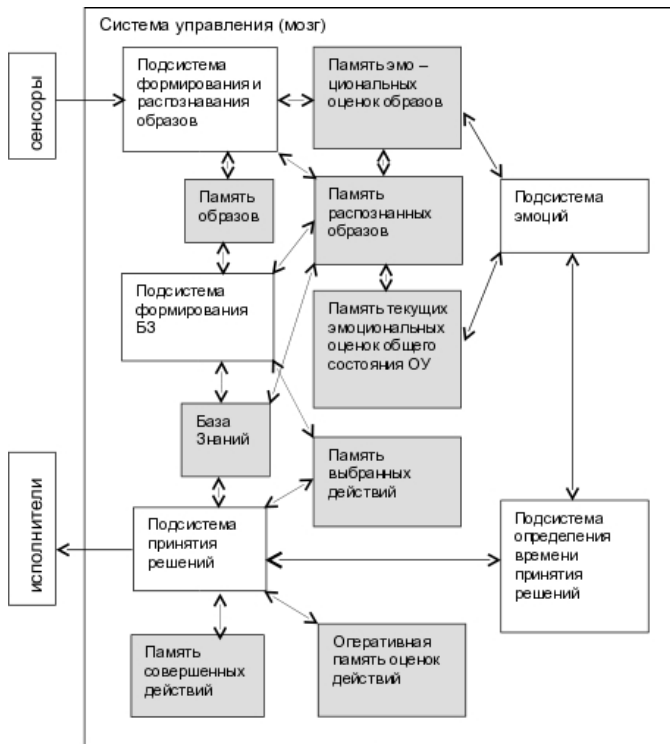


Рис.1. Структура функциональных подсистем и памяти управляющей системы ААУ

Одним из важнейших компонентов системы ААУ и подобных ей интеллектуальных СУ является подсистема формирования новых образов (подсистема кластеризации) на основе памяти и накопленного опыта. Задача данной подсистемы состоит в том, что она автоматически (по неким критериям) формирует новые образы (или кластеры), сохраняя описание новых кластеров и позволяя системе распознавать их во входной последовательности данных с сенсоров, используя их далее при накоплении знаний. Подобная подсистема необходима для таких интеллектуальных СУ, когда заранее невозможно описать все ситуации (образы), в которых она может оказаться.

К настоящему времени разработано множество различных систем кластеризации, опирающихся на методы распознавания вероятностные, геометрические, нечеткие и другие (см. например, обзор в [6]). Однако, во-

первых, в них используются модели, которые ограничивают применение этих систем, во-вторых, они в основном были разработаны для приложений Data Mining. Некоторые из них являются слишком сложными для автономных систем управления, работающими практически в реальном времени и с пространствами большой размерности. Наконец, в-третьих, они не используют информацию о важности кластеров для СУ, поскольку само это понятие характерно для СУ в комплексе, а не для отдельной подсистемы формирования образов.

Постановка задачи и описание алгоритма

Рассмотрим предлагаемый подход на примере задачи управления мобильным роботом с помощью системы ААУ [7]. Пусть в каждый момент времени $t(k), k = \{1, 2, 3, \dots, \infty\}$ на вход бортовой СУ поступает информация $\{x_1^{t(k)}, x_2^{t(k)}, \dots, x_N^{t(k)}\}$ с датчиков робота (где N – число датчиков). Благодаря наличию в СУ подсистемы распознавания образов, информация с датчиков приводит к распознаванию образа $O(t(k), j_k)$ (обозначим для краткости через $O(k)$), соответствующего одному из классов ω_{j_k} , где $1 \leq j_k \leq P$, содержащегося в алфавите классов $\Omega_1 = \{\omega_1, \omega_2, \dots, \omega_P\}$. В следующий момент времени $t(k+1)$ процесс повторяется, и мы получим образ $O(k+1)$. Назовем образы $\{O(1), O(2), \dots, O(k), \dots\}$ базовыми образами или образами 0-го уровня и обозначим их через $O^{L_0}(k) = O(k)$ (соответствующие им классы через $\omega_{j_k}^{L_0} = \omega_{j_k}$, а алфавит классов через $\Omega_1^{L_0}$). Эти образы обычно задаются вручную (через решающие правила, предварительное обучение нейронных сетей, непосредственное описание образов в признаковом пространстве и т.п.). Они несут в себе некоторую априорную информацию (наши знания в форме алгоритмов), чтобы помочь мобильному роботу лучше решать поставленные перед ним задачи в данной среде и сократить время поиска на начальном этапе работы СУ.

Однако очень сложно создать заранее универсальный алфавит классов $\Omega_1^{L_0}$, чтобы он эффективно работал в изменяющейся реальной среде (из-за ограничений на вычислительные ресурсы и память). В данной работе предлагается создавать новые образы более высокого уровня на основе наблюдаемой СУ временной последовательности образов более низкого

уровня. Для борьбы с экспоненциальным ростом количества таких образов (см. (2)) применяется рейтинг-функция S (которая будет описана ниже), вычисляемая для каждого нового создаваемого образа. Если вычисленное значение больше некоторого порогового $S_{threshold}$, тогда образ создается, сохраняется в алфавите классов и в дополнительной памяти (назовем его *сформированным* образом), в противном случае образ не создается, а лишь сохраняется в дополнительной памяти до ее заполнения, а потом удаляется (назовем такой образ *несформированным*). Для простоты число уровней ограничено двумя: $L0$ (нулевой) – задается, как было показано выше, и $L1$ (первый) – формируется в процессе работы СУ.

Пусть у СУ есть память (*память распознанных образов* по методологии ААУ, см. рис. 1), состоящая из двух частей. Первая часть запоминает последние w распознанных образов $\{O^{L0}(r+1), \dots, O^{L0}(r+w)\}$ в порядке их поступления (число w равно числу образов нулевого уровня $O^{L0}(k)$ на следующем (1-м) уровне). Вторая часть хранит Z , как сформированных, так и несформированных образов 1-го уровня $\{O^{L1}(k_1), O^{L1}(k_2), \dots, O^{L1}(k_z)\}$, где $k_1 < k_2 < \dots < k_z$.

Чтобы определить образ 1-го уровня, введем сначала алфавит классов 1-го уровня $\Omega_1^{L1} = \{\emptyset\}$ (изначально пустой). Описание j -го класса 1-го уровня ω_j^{L1} состоит из последовательности номеров классов 0-го уровня $\{\omega_1^{L0}, \omega_2^{L0}, \dots, \omega_w^{L0}\}$. Именно здесь мы видим преимущество подобного подхода, т.к. нам не пришлось опускаться до уровня описания классов на языке словаря признаков. Отметим, что вне зависимости от значения рейтинг-функции S , мы всегда создаем новый класс ω_j^{L1} с соответствующим описанием, за исключением случая, когда такой класс уже существует в алфавите классов Ω_1^{L1} . Если класс ω_j^{L1} соответствует несформированным образам и последний из этих образов удаляется из памяти, то этот класс и его описание также удаляются.

Определим образ 1-го уровня:

$$O^{L1}(t(r+1), t(r+w), j_{r+1, r+w}) = \{O^{L0}(r+1), \dots, O^{L0}(r+w)\}$$

(обозначим через $O^{L1}(r+1, r+w)$), где $j_{r+1, r+w}$ это номер соответствующего класса $\omega_{j_{r+1, r+w}}^{L1}$.

В этом случае образ $O^{L1}(r+1, r+w)$ будет хранить $(w+1)$ временную метку $\{t(r), \dots, t(r+w)\}$, а также несколько параметров определяющих интегральные характеристики всех w базовых образов.

Для дальнейшей работы с такими образами необходимо выбрать некоторую меру их разницы (dissimilarity measure). Используем простейшую, а именно:

$$\begin{aligned} DM(O^{L1}(r_1+1, r_1+w), O^{L1}(r_2+1, r_2+w)) &= \\ &= DM(\omega_i^{L1}, \omega_j^{L1}) = \begin{cases} 0, i = j \\ \infty, i \neq j \end{cases} \end{aligned} \quad (1)$$

где ω_i^{L1} и ω_j^{L1} классы образов $O^{L1}(r_1+1, r_1+w)$ и $O^{L1}(r_2+1, r_2+w)$ соответственно.

Теперь становится ясно, что максимальное число различных образов (в смысле введенной выше меры) 1-го уровня (количество элементов или мощность множества алфавита классов Ω_1^{L1}) равно:

$$|\Omega_1^{L1}| = |\Omega_1^{L0}|^w = P^w \quad (2)$$

Как было указано выше, для отбора потенциально наиболее важных для СУ образов используется рейтинг-функция S , которая имеет следующий вид:

$$\begin{aligned} S(O^{L1}(r+1, r+w)) &= \alpha * v(O^{L1}(r+1, r+w)) + \\ &\beta * Emotion(O^{L1}(r+1, r+w)) / MaxEmotion \end{aligned} \quad (3)$$

где:

α и β - весовые коэффициенты, причем $\alpha + \beta = 1$,

MaxEmotion – максимальная достижимая СУ эмоциональная оценка состояния согласно методологии ААУ, центрированная на ноль и взятая по модулю,

$Emotion(O^{L1}(r+1, r+w))$ – эмоциональная оценка $O^{L1}(r+1, r+w)$, вычисленная как среднее от оценок составляющих его образов $\{Emotion(O^{L0}(r+1)), \dots, Emotion(O^{L0}(r+w))\}$,

$$v(O^{L1}(r+1, r+w)) = \frac{MemCount(O^{L1}(r+1, r+w))}{MemorySize} \quad \text{– встречае-}$$

мость образа $O^{L1}(r+1, r+w)$ во второй части памяти (число образов в памяти с одинаковыми классами ω_j^{L1} деленое на ее размер). Если память имеет бесконечный или очень большой размер, тогда:

$$MemorySize = \max_{\forall O^{L1}(r+1, r+w)} \{MemCount(O^{L1}(r+1, r+w))\} \quad (4)$$

Основным слабым местом данного простого метода формирования временных образов является потенциально быстрый рост числа образов, сформированных во время работы СУ, т.е. не ясна эффективность введенной рейтинг-функции S с заранее заданным порогом.

Результаты и обсуждение

В качестве модельного объекта была выбрана виртуальная модель мобильного робота, созданная в среде Microsoft Robotics Development Studio. Задачей мобильного робота являлась езда по замкнутому треку с крутыми поворотами с минимальными отклонениями от центра дороги. Изначально база знаний СУ (см. рис. 1 и [1-3]) была пустой, т.е. робот не «знал», куда следует поворачивать в той или иной текущей ситуации. Количество образов нулевого уровня $L0$, созданных вручную, было равно 1331.

Ниже рассматриваются результаты моделирования работы данной системы с двумя уровнями $L0$ и $L1$, а также очень большой памятью для образов $O^{L1}(r+1, r+w)$ (т.е. образы хранятся и не удаляются). Рассматриваются 2 значения параметра w (10 и 5), а также влияние параметра $S_{threshold}$ (“S_threshold” на рис. 2 и 3) на количество сформированных образов. Весовые коэффициенты α и β в (3) приняты равными 0,7 и 0,3 соответственно.

Помимо вариантов с различными параметрами $S_{threshold}$ были реализованы также еще два базовых варианта:

- “BruteForce” – метод “грубой силы” или полного перебора. В этом случае $S_{threshold} = 0$, все образы $O^{L1}(r+1, r+w)$, которые встречаются, сразу же становятся сформированными.
- “Mean” – метод осреднения. В этом случае номер класса ω_j^{L1} вычисляется, как среднее от номеров классов $\{\omega_1^{L0}, \omega_2^{L0}, \dots, \omega_w^{L0}\}$, составляющих его. Затем проводится округление. Ясно, что в этом случае: $\lim_{k \rightarrow \infty} |\Omega_1^{L1}| = |\Omega_1^{L0}|$ и равно 1331 для рассматриваемой системы.

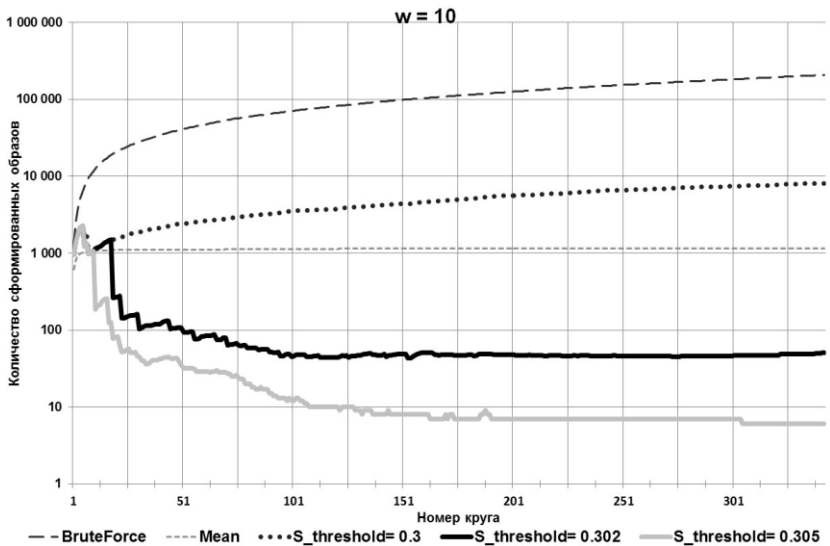


Рис.2. Зависимость количества сформированных образов для разных порогов $S_{threshold}$ от числа кругов, которые проехал мобильный робот. Параметр $w = 10$. Масштаб по оси ординат логарифмический

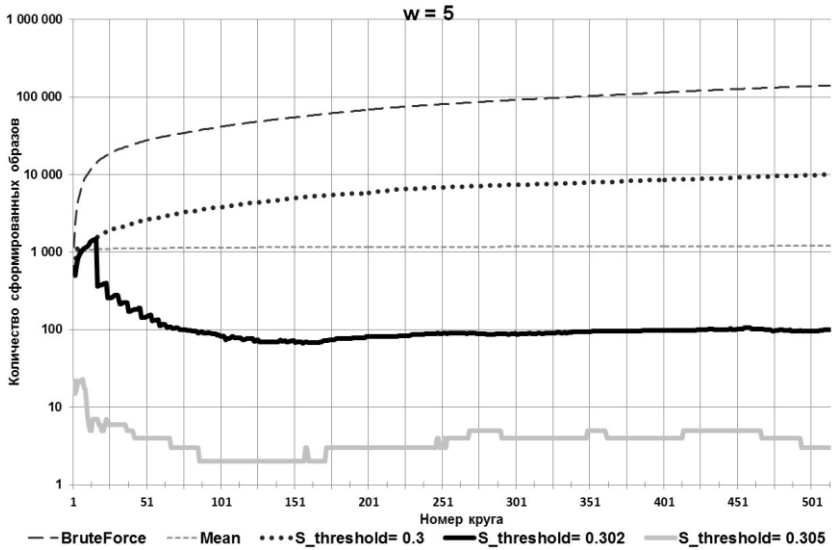


Рис.3. Зависимость количества сформированных образов для разных порогов

$S_{threshold}$ от числа кругов, которые проехал мобильный робот. Параметр $w = 5$. Масштаб по оси ординат логарифмический

По приведенным результатам моделирования можно сделать следующие выводы:

1. Созданная рейтинг-функция S позволяет сформировать небольшое количество новых образов при использовании порога $S_{threshold} > 0,3$. Причем этот порог оказывается эффективным, как в случае с $w = 10$ (кривые “ $S_{threshold}=0.302$ ” и “ $S_{threshold}=0.305$ ”), так и с $w = 5$ (кривые “ $S_{threshold}=0.302$ ” и “ $S_{threshold}=0.305$ ”)
2. Критическое значение порога $S_{threshold}^{crit} = 0,3$, по-видимому, связано с большим количеством образов 0-го уровня $O^{L0}(k)$, имеющих наивысшую (по модулю) эмоциональную оценку. Возможно, это объясняется ее дискретностью для $O^{L0}(k)$ и малым количеством уровней (6 уровней).

3. Исходя из выводов 1 и 2, можно предположить, что наибольший вклад в отбор формируемых образов вносит их частота появления $\nu(O^{L1}(r+1, r+w))$ в (3).
4. Как и ожидалось, вариант метода с использованием полного перебора работает неэффективно и количество образов, формируемое им, пропорционально времени работы СУ (рис. 2 и 3 для кривой “BruteForce”).

Заключение

В данной работе предложен простой метод формирования новых временных образов. Показано, что, используя рейтинг-функцию (3) на основе частоты встречаемости, а также эмоциональной оценки, возможно значительно сократить количество формируемых образов, устраняя тем самым основной недостаток простой идеи, лежащей в основе разработанного метода. Тем не менее, для полного доказательства эффективности данного метода необходимо проверить важность сформированных образов для интеллектуальной СУ, показав рост качества управления в соответствии с выбранным критерием.

В качестве дальнейших направлений работы по улучшению характеристик данного метода можно отметить:

- Использование более сложных алгоритмов сравнения последовательностей (Dynamic Time Warping (DTW), Longest Common Subsequence (LCSS) и т.д.) [8,9,10] вместо (1).
- Использование методов понижения размерности создаваемых образов (например, с помощью Discrete Fourier Transform (DFT), Discrete Wavelet Transform (DWT), Adaptive Piecewise Constant Approximation (APCA) и т.д.) [11], уменьшая тем самым параметр W в (2).

Созданный вариант метода не потеряет своей актуальности, даже при использовании указанных выше алгоритмов, и может быть использован для сравнения с новыми вариантами, как базовый, благодаря минимальному количеству сделанных предположений и своей универсальности.

Список литературы

- [1] Жданов А.А. Автономный искусственный интеллект // Монография. М.: БИНОМ. Лаборатория знаний, 2008.
- [2] Жданов А.А. Метод автономного адаптивного управления // Известия Академии Наук. Теория и системы управления, 1999, 5, С. 127-134.

- [3] Жданов А.А. О методе автономного адаптивного управления // Научная сессия МИФИ-2004. VI Всероссийская н.-т. конференция "Нейроинформатика-2004": Лекции по нейроинформатике. Ч. 2. - М.: МИФИ, 2004. С. 15-56.
- [4] Zhdanov A.A., Vinokurov A.N. Emotions Simulation in Methodology of Autonomous Adaptive Control // ISIC/ISAS'99. Special session Emotions and Intelligent Systems. 1999. Paper 99-002I-6.
- [5] Жданов А.А. О роли аппарата эмоций как системообразующего фактора в адаптивных системах управления // Труды Института системного программирования РАН. М.: 2004. С. 215-225.
- [6] Theodoridis S., Koutroumbas, K. Pattern Recognition 4th Ed // Elsevier Science. 2008.
- [7] Жданов А.А., Крыжановский М.В., Преображенский Н.Б. Бионическая интеллектуальная адаптивная система управления мобильным роботом // Искусственный интеллект. 2002. Т.4. С.341-350.
- [8] Keogh E., Pazzani M. Scaling up Dynamic Time Warping for Datamining Applications // In Proc. 6th Int. Conf. on Knowledge Discovery and Data Mining, Boston, MA, 2000.
- [9] Agrawal R., Lin K., Sawhney H.S., Shim K. Fast Similarity Search in the Presence of Noise, Scaling and Translation in Time-Series Databases // In Proc of VLDB Sept. 1995. P. 490–501
- [10] Устюжанин А.Е. Метод построения системы памяти для хранения и поиска многомерных пространственно-временных последовательностей // Вестник Московского государственного технического университета им. Н.Э. Баумана. серия «Приборостроение». номер 2. 2007. С.104- 112.
- [11] Camastra F. Data dimensionality estimation methods: A survey // Pattern Recognition. 2003. V. 36. P. 2945–2954.