УДК 519.62

А.А. Гришко, С.Г. Удовенко, Л.Э. Чалая
ХНУРЭ, г. Харьков, Украина, udovenko@kture.kharkov.ua

ГИБРИДНЫЕ МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ В СИСТЕМАХ УПРАВЛЕНИЯ ДИНАМИЧЕСКИМИ ОБЪЕКТАМИ

В работе проведен анализ проблемы машинного обучения применительно к цифровым системам управления динамическими объектами. Предложен и протестирован модифицированный метод формирования стратегий управления, основанный на использовании Q-обучения с подкреплением. Новый подход позволяет пользователю системы управления непрерывно получать, анализировать и использовать сигналы подкрепления для выработки управляющих воздействий с применением нейросетевой модели.

СИСТЕМА УПРАВЛЕНИЯ, МАШИННОЕ ОБУЧЕНИЕ, НЕЙРОННАЯ СЕТЬ, Q-ФУНКЦИЯ, СТРАТЕГИЯ

Введение

В последнее время получили распространение управляемые стохастические системы, в основе функционирования которых лежит метод обучения с подкреплением, также называемый методом подкрепляемого обучения. Метод подкрепляемого обучения является достаточно новым методом в группе методов машинного обучения и занимает промежуточное положение между методами обучения с учителем и без учителя. В основе метода обучения с подкреплением лежат те основополагающие принципы адаптивного поведения, которые позволяют живым организмам приспосабливаться к изменяющимся или неизвестным условиям обитания. Метод обучения с подкреплением (Reinforcement Learning) был представлен и подробно изложен в [1]. В данном методе в обобщенном виде рассматривается взаимодействие агента с внешней средой, в результате которого агент путем проб и ошибок самостоятельно определяет наиболее оптимальное поведение для достижения максимума некоторого критерия. Отличительной чертой метода обучения с подкреплением является наличие сигнала подкрепления, который получает агент в процессе взаимодействия с внешней средой и который является скалярной величиной, характеризующей, насколько удачно функционирует агент в данный момент времени. Целью функционирования агента является максимизация суммарного сигнала подкрепления, которое получит агент при взаимодействии с внешней средой. В исходном виде метод обучения с подкреплением предполагает конечное количество состояний внешней среды и возможных воздействий агента на внешнюю среду, а также взаимодействие агента с внешней средой в дискретные моменты времени. Обучение с подкреплением является методом, который позволяет находить оперативное решение, являющееся оптимальным в смысле получения максимального дохода в каждом из состояний. При этом он позволяет в процессе обучения допускать возможность кратковременных потерь, с тем, чтобы впоследствии максимизировать суммарный доход на длительном интервале. Вследствие этого, обучение с подкреплением является методом, наиболее приспособленным для эффективной работы в системах, характеризующихся высоким уровнем изменения внешних и внутренних воздействий (например, в трейдинговых системах электронной биржевой торговли) [2]. В соответствии с методом обучения с подкреплением сигналы подкрепления и состояния внешней среды должны обладать свойством марковости. Однако в [1] показано, что метод может быть успешно применен и в том случае, когда сигналы подкрепления и состояния внешней среды не обладают свойством марковости. Представляется целесообразным рассмотреть возможность применения обучения с подкреплением для управления динамическими объектами и предложить эффективный подход к реализации задачи такого управления.

1. Формализация задачи

В задаче обучения с подкреплением для агента управляемой системы используются следующие элементы:

– набор состояний системы S, возникающих при получении агентом информации из внешней среды;

– набор возможных управляющих воздействий A;

– функция выигрыша R_t, получаемого в состоянии s_t при выборе в текущем такте управляющего воздействия a_t.

Задача базового алгоритма Q-обучения с подкреплением – определить и реализовать стратегию $\pi: S \to A$, основанную на текущем состоянии s_t, т.е. $\pi(s_t) = a_t$. Обычно требуется найти стратегию, соответствующую максимальному значению длительной суммы сигналов подкрепления.

Долгосрочный выигрыш R_t может задаваться различными типами функций в зависимости от поставленных задач. Если задача состоит в выработке фиксированной стратегии на протяжении

заданного числа шагов, то выигрыш можно определять как сумму прогнозируемых сигналов подкреплений:

$$R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}\,, \qquad (1)$$

где $\gamma \in [0,1]$ – весовой коэффициент, задающий значимость будущих подкреплений.

В большинстве известных алгоритмов обучения с подкреплением используются обобщенные функции выигрыша V^{π}, которые позволяют для заданной стратегии π оценить средний выигрыш для одного состояния:

$$V^{\pi}(s) = E_{\pi}(R_t \,|\, s_t = s)\,. \qquad (2)$$

Эти функции могут также определяться не только состояниями, но и действиями агента в этих состояниях:

$$Q^{\pi}(s,a) = E_{\pi}(R_t \,|\, s_t = s, a_t = a)\,. \qquad (3)$$

Обобщенная функция (2) для состояния связана с $Q^{\pi}(s,a)$ соотношением вида:

$$V^{\pi}(s) = \sum_a \pi(s,a) Q^{\pi}(s,a)\,. \qquad (4)$$

Особенности функции (4) позволяют создавать различные модификации алгоритмов обучения с подкреплением, основанные на рекуррентном уравнении Беллмана следующего вида:

$$V^{\pi}(s) = \sum_a \pi(s,a) \sum_{s'} P^a_{ss'} [\Re^a_{ss'} + \gamma V^{\pi}(s')]\,, \qquad (5)$$

где $P^a_{ss'}$ – функция перехода, определяющая вероятность перехода системы в состояние s' при применении агентом в состоянии s действия a; $\Re^a_{ss'}$ – соответствующая усредненная фунция выигрыша.

Это уравнение, связывающее выигрыши для одного состояния с выигрышами для последующих состояний, можно получить следующим образом:

$$\begin{aligned} V^{\pi}(s) &= E_{\pi}(R_t \,|\, s_t = s) = \\ &= E_{\pi}\Big(\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \,\Big|\, s_t = s\Big) = \\ &= E_{\pi}\Big(r_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \,\Big|\, s_t = \\ &= \sum_a \pi(s,a) \sum_{s'} P^a_{ss'} [\Re^a_{ss'} + \gamma E_{\pi}\Big(\sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \,\Big|\, s_{t+1} = s'\Big)] = \\ &= \sum_a \pi(s,a) \sum_{s'} P^a_{ss'} [\Re^a_{ss'} + \gamma V^{\pi}(s')]. \end{aligned}$$

Функция (5) позволяет оценивать качество стратегии по выигрышам в будущих состояниях системы и, соответственно, сравнивать различные стратегии:

$$\pi \geq \pi' \Leftrightarrow \forall s, V^{\pi}(s) \geq V^{\pi'}(s)\,. \qquad (6)$$

В соответствии с неравенством (6) зададим функцию оценивания оптимальности стратегии, которая позволила бы реализовывать обучение с подкреплением обучения с подкреплением:

$$V^*(s) = \max_{\pi} V^{\pi}(s)\,. \qquad (7)$$

Каждой паре «состояние-действие» с учетом (7) поставим в соответствие функцию вида:

$$Q^*(s,a) = \max_{\pi} Q^{\pi}(s,a)\,, \qquad (8)$$

определяемую последующей рекурсией:

$$Q^*(s,a) = E(r_{t+1} + \gamma V^*(s_{t+1}) \,|\, s_t = a, a_t = a)\,. \qquad (9)$$

С учетом уравнений (5) и (9) можно определить уравнения оптимальности стратегий для функций V^* и Q^* соответственно:

$$Q^*(s,a) = E(r_{t+1} + \gamma V^*(s_{t+1}) \,|\, s_t = a, a_t = a)\,; \qquad (10)$$

$$\begin{aligned} V^*(s) &= \max_a E(r_{t+1} + \gamma V^*(s_{t+1}) \,|\, s_t = s, a_t = a) \\ &= \max_a \sum_{s'} P^a_{ss'} [\Re^a_{ss'} + \gamma V^*(s')] \end{aligned}\,. \qquad (11)$$

Очевидно, что применение стратегий в соответствии с (10) и (11) позволит агенту выбирать действия, максимизирующие обобщенный выигрыш.

Если динамику управляемой системы можно описать конечномерным управляемым марковским процессом, то уравнения (10) и (11) дают единственное решение, которое позволяет определить для каждого состояния максимальный возможный выигрыш. По известным значениям V^* нетрудно найти оптимальную стратегию, которая ставит в соответствие каждому состоянию действие a', максимизирующее уравнения оптимальности стратегий для функций V^* и Q^* соответственно:

$$a' = \arg\max_a E(r_{t+1} + \gamma V^*(s_{t+1}) \,|\, s_t = s, a_t = a)\,; \qquad (12)$$

$$a' = \arg\max_a Q^*(s,a)\,. \qquad (13)$$

Следует отметить, что для определения стратегий по V^* необходимо иметь модель среды, чтобы оценивать r_{t+1} и s_{t+1}. В случае применения функции Q^* наличие такой модели не является необходимым.

Таким образом, проблема обучения с подкреплением может быть сведена к выбору оптимальных стратегий по оценкам V^* и Q^*. Отметим, что для известных $P^a_{ss'}$ и $\Re^a_{ss'}$ можно непосредственно определять V^* и Q^*, решая систему уравнений оптимальности стратегий для каждого состояния методами динамического программирования. Однако такое решение трудно применить к системам управления динамическими объектами (в частности, роботами), так как зачастую полная модель управляемой системы (среды) является неизвестной, а ее большая размерность не позволяет агенту использовать метод проб и ошибок. Более приемлемыми для решения задач оценивания рассмотренных функций

выигрыша и последующего улучшения стратегия являются методы Монте-Карло, не требующие обязательного наличия такой модели. Базовый метод Монте-Карло основан на независимой оценке выигрыша для каждого состояния и позволяет выделить наиболее важные сегменты управляемого пространства, в то время как использование динамического программирования предполагает необходимость одновременного оценивания всех состояний. Этот метод применим для функции $Q(s,a)$, не требующей наличия полной модели среды и формирующейся в процессе получения сигалов подкрепления (текущих оценок выигрыша) после реализации действий a в состоянии s. При этом предполагается необходимость ограничения бесконечного в общем случае количества пар (s,a), что является серьезной проблемой, так как снижает качество получаемых решений. Решить эту проблему можно добавлением к процедуре определения стратегии специальной процедуры, гарантирующей возможность реализации всех допустимых действий с некоторой вероятностью. Существует два метода задания такой процедуры. Первый состоит в условном объединении всех стратегий с низкой вероятностью, близкой к некоторой величине ε, и их сопоставлении всем возможным действиям. Процедура обучения сходится при этом к оптимальной стратегии в окрестности этого объединенного класса стратегий. Это позволяет гарантировать субоптимальную работу системы, а значение вероятности ε может быть уменьшено, если в процессе работы системы агент получит достаточные для оценивания стратегий данные. Такой метод называют методом «on policy» (в пределах стратегии), так как он лишь модифицирует при необходимости стратегию, используемую агентом. Второй метод называют методом «off policy» (вне стратегии), так как он оценивает одну стратегию, в то время как агент использует другую. Эта другая стратегия выбирает, как правило, оригинальную стратегию с вероятностью $1-\varepsilon$ и случайное действие с вероятностью ε. Для оценивания оригинальной стратегии метод Монте-Карло использует конечные фрагменты последовательностей, для которых выбор действий соответствует выбору, применяемому для оригинальной стратегии, но модифицирует взвешивание выигрышей для компенсации разности вероятностей выбора действий по двум стратегиям. Обучение, использующее метод Монте-Карло, осуществляется, таким образом, на основе чередования оценок одной стратегии и ее модификации, вырабатывая действия, которые максимизируют выигыш для каждого состояния. Такое чередование может быть реализовано различными способами, как и для динамического программирования. Как динамическое программирование, так и метод Монте-Карло имеют свои преимущества и недостатки. В частности, динамическое программирование обладает интересным свойством оценивания каждого из состояний, именуемым «загрузчиком». Это свойство позволяет достичь более высокой сходимости (в единицах сочетаний «состояние-действие-сигнал подкрепления») по сравнению с методом Монте-Карло.

Рассмотрим теперь процесс обучения по методу временных разностей (TD), объединяющему в определенном смысле преимущества двух описанных выше методов и являющемуся наиболее приемлемым для практического применения в системах управления динамическими объектами. Оценивание стратегии TD – метода осуществим, как и для метода Монте-Карло, использующего усредненые оценки для одного состояния. Представление такого оценивания в итеративной форме приводит к следующему уравнению:

$$V(s_t) \leftarrow V(s_t) + \alpha[R_t - V(s_t)], \qquad (14)$$

где

$$R_t = r_{t+1} + \gamma r_{t+2} + \ldots + \gamma^p r_{t+p+1}. \qquad (15)$$

Уравнения (14) и (15) отражают идею использования оценки выигрыша, начиная со следующего сигнала подкрепления и следующего значения состояния, вместо использовани сигналов подкрепления экспериментальной последовательности. TD – метод использует при этом следующее преобразование:

$$V(s_t) \leftarrow V(s_t) + \alpha[r_{t+1} + \gamma V(s_{t+1}) - V(s_t)]. \qquad (16)$$

Как правило, применяется инкрементная форма преобразования (16), учитывающая действия агента. Такая форма для функции Q имеет следующий вид:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha[r_{t+1} + \gamma Q(s_{t+1},a_{t+1}) - Q(s_t,a_t)]. \qquad (17)$$

Метод, основанный на преобразовании (17), называют иногда SARSA-методом (State, Action, Reward, State, Action – состояние, действие, выигрыш, состояние, действие). Он реализует оценивание «on policy», так как используемое здесь действие a_{t+1} зависит от выбираемой стратегии.

Наиболее распространенный метод обучения с подкреплением – метод Q-обучения, являющийся вариантом «off policy» SARSA-метода, в котором используется максимум функции от прогнозируемых действий вместо реализуемого в текущий момент действия:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha[r_{t+1} + \gamma \max_a Q(s_{t+1},a) - Q(s_t,a_t)]. \qquad (18)$$

Этот алгоритм осуществляет сходимость Q к Q^* независимо от реализуемой стратегии, причем эта стратегия гарантирует приемлемые результаты (если вероятности отличны от нуля для всех действий во всех состояниях).

2. Гибридный метод машинного обучения

Методы, использующие временные разности, позволяют заменить конечные значения используемых в алгоритме оценок оценками, полученными по методу Монте-Карло для следующего состояния. При этом в процессе обучения такие прогнозируемые оценки могут оказаться недостаточно корректными и тогда их необходимо заменить другими значениями оценок.

При применении TD – метода n раз получим выигрыш, определяемый следующим уравнением:

$$R_t = R_t^n = r_{t+1} + \gamma r_{t+2} + \ldots + \\ +\gamma^n (r_{t+n+1} + \gamma V(s_{t+n+1})). \quad (19)$$

Использование оценки (19) позволяет получить процедуру вида:

$$V(s_t) \leftarrow V(s_t) + \alpha[R_t^n - V(s_t)]. \quad (20)$$

Введем в (20) усредненные взвешенные оценки R_t:

$$R_t = \sum_i a_i R_t^i; \quad \sum a_i = 1. \quad (21)$$

Частным случаем взвешенных оценок здесь являются экспоненциально взвешенные оценки, которые приводят к повышению значимости будущих результатов и их измерений по мере возрастания аргумента времени:

$$R_t = (1-\lambda)\sum_{i=1}^{\infty} \lambda^i R_t^i. \quad (22)$$

Это позволяет упростить реализацию TD – метода, используя предыдущие значения сигналов подкрепления вместо оценок будущих значений. Введем для этого вспомогательный параметр (назовем его коэффициентом приемлемости), определяемый по следующей рекурсии:

$$e_t(s) = \begin{cases} \gamma\lambda e_{t-1}(s), & s \neq s_t \\ \gamma\lambda e_{t-1}(s) + 1, & s = s_t. \end{cases} \quad (23)$$

Перерасчет обобщенных функций выигрыша будет теперь производиться для каждого состояния пропорционально значению его коэффициента приемлемости:

$$V(s_t) \leftarrow V(s_t) + \alpha e(s_t)[r_{t+1} + \gamma V(s_{t+1}) - V(s_t)]. \quad (24)$$

Суть предложенного подхода состоит в замене состояний, использующих прогнозируемые значения сигналов подкрепления предыдущими взвешенными состояниями, использующими текущие значения сигналов подкрепления. Соответствующий алгоритм, который является расширением SARSA-алгоритма и $Q(\lambda)$ -алгоритма, может быть представлен следующей последовательностью вычислительных операций:

1. Инициализировать $V(s)$ и принять $e(s) = 0$ для всех s.
2. **for all** итераций **do**
3. Инициализировать значение s.
4. **for all** шагов текущей итерации **do**
5. $a \leftarrow \pi(s)$.
6. Реализовать a, получить r и значение следующего состояния s'.
7. $\delta \leftarrow r + \gamma V(s') - V(s)$.
8. $e(s) \leftarrow e(s) + 1$.
9. **for all** s **do**
10. $V(s) \leftarrow V(s) + \alpha\delta e(s)$.
11. $e(s) \leftarrow \gamma\lambda e(s)$.
12. **end for** .
13. $s \leftarrow s'$.
14. **end for**
15. **end for**

3. Практическая реализация гибридного метода

В реальных системах робототехники и цифрового управления динамическими процессами датчики и исполнительные механизмы редко бывают дискретными или же, если они все-таки используются, число состояний должно быть очень большим. Описанные выше алгоритмы обучения с подкреплением (и, в частности, предложенный гибридный алгоритм) предполагают применение множеств состояний и управляющих воздействий такой размерности, чтобы соответствующие вычислительные процедуры могли сходиться в реальном масштабе времени. Для практической реализации обучения с подкреплением при управлении динамическими объектами можно использовать различные подходы. Наиболее простой из них заключается в обычной дискретизации непрерывного пространства состояний и выделении нескольких сотен наиболее вероятных состояний, которые впоследствии будут использоваться в упрощенных вариантах алгоритмов обучения (например, с непосредственным использованием таблиц значений $Q[s][a]$).

Второй подход предполагает возможность работы алгоритма обучения непосредственно в непрерывном пространстве состояний, фиксируемых датчиками, используя методы аппроксимации функций. В общем случае, для практической реализации алгоритмов обучения с подкреплением необходимо корректно оценивать значения функции $Q(s,a)$. Аппроксимация такой функции может быть успешно реализована с применением искусственных нейронных сетей (ИНС). Нейросетевые методы реализации алгоритмов обучения зачастую называют коннекционистскими. Один из вариантов ИНС типа «многослойный персептрон», предназначенных для аппроксимации функции $Q(s,a)$, приведен на рис. 1.

Сравнивая обычное использование персептрона для задачи аппроксимации и его использование в качестве составной части алгоритма с подкреплением, можно выделить два основных момента. Во-первых, в задачах обычной аппроксимации обучение производится на некотором обучающем множестве, элементы которого постоянно

повторяются. При обучении с подкреплением предварительно заданного обучающего множества нет, а входные образцы формируются при взаимодействии автономного агента со средой. Таким образом, в процессе обучения некоторые образцы встречаются чаще, другие реже, а при работе с непрерывной средой велика вероятность того, что входной образец встретится лишь однажды. Поэтому для задач обучения с подкреплением задача переобучения является неактуальной.

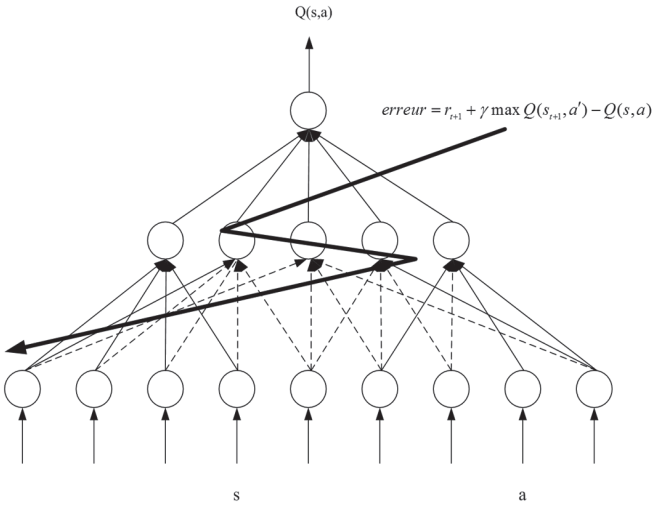

Рис.1. Структура искусственной нейронной сети, аппроксимирующей функцию $Q(s,a)$

Во-вторых, в задачах обычной аппроксимации обучение производится на известных результатах, т.е. известны истинные значения аппроксимируемой функции в определенных точках. При обучении с подкреплением истинные значения аппроксимируемой функции заранее неизвестны, и обучение происходит на оценках Q-значений, которые постепенно изменяются в процессе обучения.

При использовании Q-обучения коррекция Q-функции производится на основе оценок текущего и последующего состояний, т.е. на оценках смежных состояний. Гибридный алгоритм, описанный выше, позволяет учитывать оценки состояний, удаленных на большее расстояние друг от друга. Формула для алгоритма TD(λ) может быть представлена как в инкрементной, так и в рекурсивной форме. При использовании рекурсивной формулы обновление Q-функции производится только после того, как агент достигнет поглощающего состояния (когда задача агента будет выполнена). При использовании инкрементной формулы обновления производятся после каждого шага агента. Данные методы ускорения обучения могут использоваться как для табличного представления Q-функции, так и при использовании нейросетевой аппроксимации.

Для обучения многослойного персептрона используется алгоритм обратного распространения ошибки. Следует отметить, что при использовании такой архитектуры необходимо обновлять только веса, связанные с выходным узлом, действие которого было выбрано. Обновление производится после того, как агент совершил переход. Чтобы при обратном распространении ошибки не возникало конфликтных ситуаций с весами скрытого слоя, можно разделить единую нейронную сеть на несколько сетей с единственным выходом, каждая из которых относилась бы к отдельному действию. Конечно, такой подход может быть применен только при относительно малом количестве действий. При использовании множества сетей для вычисления оценок-значений на каждой итерации алгоритма на входы всех сетей подается состояние, а коррекция весов производится только для той сети, действие которой было выбрано.

Рассмотрим два примера, иллюстрирующих возможность применения гибридного метода обучения с подкреплением при управлении динамическими объектами.

Пример 1. Проведем анализ решения задачи управления мобильным роботом по гибридному алгоритму обучения с подкреплением на основе Q-таблиц и ИНС.

Перед роботом ставится задача – добраться до цели, избежав столкновения с препятствиями. Критерием оценки эффективности функционирования робота служит среднее значение выигрыша, полученного за время взаимодействия со средой. Информацию об окружающей среде робот получает при помощи 7 сенсоров: 1-5 сенсоры располагаются под углом 15° относительно друг друга и представляют информацию о расстоянии от робота до препятствия; 6-ой – информацию о расстоянии до цели; 7-ой – информацию об угле между направлением робота и целью. Схема дискретизации данных от сенсорных датчиков представлена на рис. 2. Для каждого сектора (от 0 до 6) заданы значения функции Θ. Значение Θ_i определяется наличием или отсутствием препятствий в зонах **a**, **b** и **c** (0, если препятствие в зоне **a**; 1, если в зоне **b** и 2 если в зоне **c**). Робот использует действия 3 видов: идти прямо, повернуть направо, повернуть налево. Текущий выигрыш (сигнал подкрепления) составляет +3, если робот выбрал действие продвижения вперед, и -10, если робот сталкивается с препятствиями. Состояние s определяется сочетанием 3 значений $\Theta_i : \Theta_i : \sum_k 3^k \Theta_k$.

В рассмотренном примере $\Theta_0 = 2$, $\Theta_1 = 2$, $\Theta_2 = 1$, $\Theta_3 = 1$, $\Theta_4 = 2$, $\Theta_5 = 2$, $\Theta_6 = 2$, следовательно, $s = 2150$.

При проведении экспериментов с классическим Q-обучением было установлено, что размер таблицы Q-значений по окончании обучения колеблется от 2800 до 3200 состояний. А это означает, что при использовании 2-х байт для хранения одного Q-значения потребуется 32000 байт для

всей таблицы. В то время как при использовании коннекшионистского Q-обучения (по гибридному алгоритму) с 3 нейронами скрытого слоя и 2-мя байтами для каждого веса нейронной сети потребовалось 810 байт.

Пример 2. Базовый метод обучения с подкреплением предполагает конечное количество состояний внешней среды и возможных воздействий агента на внешнюю среду, а также взаимодействие агента с внешней средой в дискретные моменты времени. Эти ограничения не позволяют широко использовать метод обучения с подкреплением в задачах управления техническими системами, т. к. сигналы в системах управления могут быть как дискретными, так и непрерывными по уровню и во времени. В то же время известны примеры успешного применения обучения с подкреплением в технических системах (например, для управления тележкой с шестом и плавающим роботом) [3].

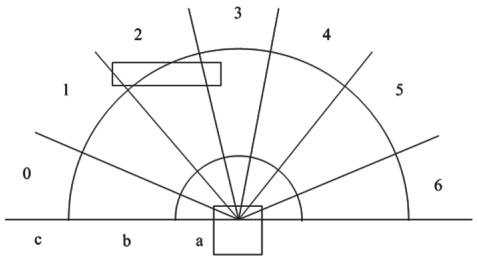

Рис. 2. Дискретизация пространства состояний мобильного робота

На основе метода подкрепляемого обучения в [4] предложена структурная схема обобщенной системы автоматического управления, функционирующей на основе метода обучения с подкреплением (МОП-САУ), используемая для моделирования SISO-систем, и алгоритмы работы структурных блоков. Приведенный выше гибридный алгоритм позволяет расширить воможности применения такой схемы. Входящий в состав модифицированной МОП-САУ объект управления (ОУ) должен удовлетворять следующим условиям: ОУ имеет один или несколько входов и один выход; в моменты дискретизации можно измерить сигналы, которые вместе с вектором управляющих воздействий $\boldsymbol{u}$ однозначно определяют значение выходной величины y в будущие моменты времени; транспотное запаздывание по каналам управления является незначительным. В результате обработки вектора входных сигналов адаптивный регулятор (АР) формирует векторное управляющее воздействие, значение которого является одним из элементов заранее определенного дискретного множества возможных воздействий. Под действием управляющего воздействия ОУ изменяет свое состояние. Вектор входных сигналов поступает на вход аналого-цифрового преобразователя (АЦП), который осуществляет дискретизацию по времени входных сигналов, необходимую для реализации гибридного метода обучения с подкреплением, который предполагает взаимодействие агента с внешней средой в дискретные моменты времени. На выходе АЦП формируется вектор дискретных сигналов, который поступает на анализирующее устройство (АУ) и на квантователь. АУ определяет значение сигнала подкрепления r, а квантователь определяет значение сигнала состояния внешней среды s, которое является одним из элементов заранее определенного множества возможных состояний внешней среды. Экстраполятор (ЭК) переводит дискретный сигнал, сформированный блоком «Агент-М» как воздействие на внешнюю среду, в непрерывное по времени управляющее воздействие на ОУ. Наличие в векторе входных сигналов производной входного воздействия и вектора переменных состояния ОУ позволяет частично учесть то, что в соответствии с методом обучения с подкреплением сигналы подкрепления и состояния внешней среды должны обладать свойством марковости. Блок «Агент-М» функционирует на основе метода обучения с подкреплением. Целью функционирования этого блока является максимизация суммарной оценки выигрыша для управления. Блок «Агент-М» состоит из устройства управления объектом (УУО) и устройства управления адаптацией (УУА). УУО формирует воздействие на основе информации о текущем состоянии внешней среды с использованием функции оценки воздействия (Q-функции). УУА осуществляет коррекцию Q-функции на основе анализа текущего состояния внешней среды и значения сигнала подкрепления как результата воздействия на внешнюю среду на предыдущем такте. В модифицированной МОП-САУ Q-функция представляется в виде таблицы соответствия, то есть для каждого возможного состояния внешней среды и для каждого возможного воздействия выделяется ячейка памяти, в которой хранится значение функции для данных значений аргументов. Экспериментальные исследования дискретных систем с различными ОУ показали, что Q-функции являются гладкими и непрерывными, что позволяет использовать для их представления функциональные аппроксиматоры. Проблему экспоненциального роста объема требуемой памяти предлагается устранить за счет представления Q-функции на основе трехслойной искусственной нейронной сети (ИНС) прямого распространения. Так как для хранения значений параметров ИНС не требуется больших объемов памяти, их применение позволит решить указанную проблему. Кроме того, входные и выходные сигналы ИНС могут быть непрерывными, что позволяет перейти от ограниченного множества возможных состояний ОУ к непрерывному пространству состояний ОУ. Первый слой ИНС является входным и

содержит столько нейронов, сколько сигналов содержится в векторе входных дискретных сигналов. Третий слой состоит из одного нейрона с линейной активационной функцией. Количество нейронов в среднем слое выбирается в зависимости от количества нейронов во входном слое. Изменение Q -функции осуществляется методом обратного распространения ошибки. Рассмотренная МОП-САУ была использована для моделирования систем цифрового управления некоторыми техническими ОУ, в частности процессом управления газоперекачивающим агрегатом (ГПА). Математическая модель ГПА представляется в виде системы дифференциальных уравнений третьего порядка. С помощью дискретной Q -функции в математическом пакете Scilab была обучена соответствующая трехслойная ИНС. Результаты моделирования показали, что среднеквадратическое отклонение значений исходной и аппроксимирующей функций друг от друга не превышает 1,5%, что свидетельствует о возможности использования ИНС для представления Q -функций. Применение ИНС позволяет не только устранить экспоненциальную зависимость объёма требуемой памяти от порядка ОУ, но также подтверждает возможность синтеза МОП-САУ для SISO и MISO-ОУ. Использование нейронных сетей в МОП-САУ затрудняется тем, что коррекция значения Q -функции на основе ИНС в одной точке приводит к изменению значений функции в других точках. Это связано с тем, что применение метода обратного распространения ошибки приводит к изменению параметров связей между нейронами, которые участвуют в формировании значений функции при любых значениях входных сигналов. С целью уменьшения влияния изменения значения Q -функции в одной точке на значения функции в других точках был применен следующий способ обучения ИНС: совместно с изменением значения Q -функции в этой точке осуществляется закрепление значений Q функции в нескольких точках из окрестности этой точки. Закрепление осуществляется за счет применения метода обратного распространения ошибки с нулевой ошибкой. Результаты экспериментов показали, что применение такого способа итерационного обучения ИНС позволяет уменьшить величину среднеквадратического отклонения значений функции в окрестности изменяемой точки от первоначальных значений.

Выводы

Результаты, полученные при моделировании систем управления роботом и газоперекачивающим агрегатом с применением предложенного гибридного метода определения управляющих стратегий, подтверждают работоспособность и перспективность применения методов машинного обучения с подкреплением в технических системах. Повышение эффективности практической реализации принципов интеллектуального управления динамическими объектами с использованием рассмотренных в настоящей процедур, может быть достигнуто в результате проведения дальнейших исследований применения методов машинного обучения с подкреплением для различных классов задач идентификации и управления в условиях неопределенности. Представляется целесообразным дальнейшее развитие функциональных возможностей рассмотренной в статье системы МОП-САУ с целью ее эффективного применения для синтеза реальных систем управления и в учебном процессе.

Список литературы: 1. *Sutton, R.S., Barto, A.G.* Reinforcement learning: An introduction. / R.S. Sutton, A.G. Barto// Cambridge, MA: MIT Press. –1998. – 432 p. **2.** *Hryshko A.* An Implementation of Genetic Algorithms as a Basis for a Trading System on the Foreign Exchange Market./ A.Hryshko, T. Downs// Proceedings of the 2003 Congress on Evolutionary Computation. – 2003. – P.1695-1701. **3.** *Filliat, David.* Robotique Mobile. / D. Filliat// Paris: ENSTA. – 2001. – 175p. **4.** *Вичугов, В.Н.* Нейросетевой метод подкрепляемого обучения в задачах автоматического управления [Текст] / В.Н. Вичугов // Известия Томского политехнического университета. – 2006. – Т. 309. № 7. – С. 92-96.

Поступила в редколлегию 11.11.2011

УДК 519.62

Гібридні методи машинного навчання в системах керування динамічними об'єктами / А.О. Гришко, С.Г. Удовенко, Л.Е. Чала // Біоніка інтелекту : наук.-техн. журнал. – 2012. – № 1 (78). – С. 78-84.

У статті розглядаються оптимальні стратегії у системах керування динамічними об'єктами , що використовують методи машинного навчання з підкріпленням. Запропонований підхід дозволяє отримати високу якість апроксимації функцій оцінювання оптимальності стратегій за допомогою багатошарових штучних нейронних мереж. Розглянуто приклади використання розроблених гібридних методів у задачах керування. Методи реалізовано програмно т та протестовано.

Іл. 2. Бібліогр.: 04 найм.

УДК 519.62

Hybrid machine learning methods in dynamic objects control systems / A.A. Hryshko, S.G. Udovenko, L.E. Chalaya. // Bionics of Intelligense: Sci. Mag. – 2012. – № 1 (78). – P. 78-84.

The paper considers the optimal strategies in control systems for dynamic objects using machine learning methods for reinforcement. The proposed approach allows to obtain high-quality approximation of the optimal strategies for evaluating functions by using multi-layer artificial neural networks. Examples of the use of hybrid methods developed in control. Methods implemented in software and tested.

Fig. 2. Ref.: 04 items.