Математические структуры и моделирование
2024. № 3 (71). С. 84–93

УДК 004.85
DOI 10.24147/2222-8772.2024.3.84-93

ДИФФЕРЕНЦИАЛЬНАЯ ИГРА «ПРЕСЛЕДОВАНИЕ–УКЛОНЕНИЕ» НА ОСНОВЕ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ

С.Н. Чуканов[1]
д.т.н., профессор, ведущий научный сотрудник, e-mail: ch_sn@mail.ru
И.С. Чуканов[2]
студент, e-mail: chukanov022@gmail.com
С.В. Лейхтер[3]
старший преподаватель, e-mail: leykhter@mail.ru

[1]Омский филиал Института математики им. С.Л. Соболева СО РАН, Омск, Россия
[2]Уральский федеральный университет имени первого Президента России Б.Н. Ельцина, Екатеринбург, Россия
[3]Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. В работе рассмотрены алгоритмы оптимального управления, основанные на схемах обучения актор/критик с подкреплением (RL). Алгоритмы используются для решения задач преследования–уклонения (PE) дифференциальных игр. Работа фокусируется на реализации решения политики агента в соответствии с концепцией адаптивного динамического программирования. Суть решения задачи PE-игры заключается в получении политики управления каждого агента (преследователя и уклоняющегося) с обеих сторон игры. В работе предложен метод адаптивного динамического программирования (ADP) для решения равновесных политик Нэша в дифференциальных играх преследования–уклонения для двух игроков. Используется метод аппроксимации функции стоимости для расчёта параметров нейросети (NN) без непосредственного решения уравнения Гамильтона–Якоби.

Ключевые слова: оптимальное управление, машинное обучение, обучение с подкреплением.

Введение

В последние годы проблема преследования–уклонения (PE) привлекает большое внимание из-за её широкого применения в конкурентных играх, оптимизации ресурсов интернета вещей и военных атаках. Однако из-за конфронтации между сторонами преследования и уклонения в реальном времени традиционная теория одностороннего управления не может точно решить проблему. Хотя существующие алгоритмы могут решить проблему дифференциальной игры во многих сценариях, автономный алгоритм не может реагировать в реальном времени на информацию агентов PE-игры с высокой производительностью в реальном времени. Данная рабо-

та фокусируется на проблеме онлайн-игр PE и реализует решение политики агента в соответствии с концепцией адаптивного динамического программирования (ADP).

Суть решения задачи PE-игры заключается в получении политики управления каждого агента с обеих сторон игры. В работе [1] Р. Айзекс ввёл в теорию игр современную теорию управления и создал дифференциальную теорию игр [2]. А. Фридман в работе [3] доказал существование седловых точек в дифференциальных играх, что позволило оптимизировать стратегии всех агентов в задаче PE. В работе [4] обсуждается единственность точки равновесия Нэша, так что аналитическое решение может быть получено для классической задачи дифференциальной игры.

В работе [5] П. Вербос и др. разработали структуры «актор–критик» для реализации алгоритмов в реальном времени, где механизмы обучения структур состоят из оценки и улучшения политики. В работе [6] Д. Берцекас и Дж. Цициклис представили методы обучения с подкреплением (RL) и сравнивают методы итерации политики (PI) и итерации значений (VI) для динамических систем с дискретным временем, которые изначально применяют идею RL к проблеме систем управления. В работе [7] П. Вербос разработал подход RL на основе VI для управления динамическими системами с дискретным временем с обратной связью с использованием аппроксимации функции стоимости (VFA). Доказано, что метод VFA пригоден для поиска оптимального управления в онлайн для задач управления с дискретным временем.

В настоящей работе предлагается метод ADP для онлайн-решения равновесных политик Нэша в дифференциальных играх преследования–уклонения для двух игроков.

1. Формулировка игры

Рассмотрим систему, содержащую два объекта и составляющую пару «преследователь–уклоняющийся».

Преследователь пытается схватить уклоняющегося, а уклоняющийся – уйти от преследователя. Игра «преследование–уклонение» в реальном времени представляет собой типичную задачу дифференциальной игры. Здесь уравнение движения каждого участника может быть выражено в виде пары дифференциальных уравнений, определённых в фиксированной системе координат. Игра с одним преследователем и одним уклоняющимся представляет собой типичную дифференциальную игру с нулевой суммой, поскольку выгоды обеих сторон исключают друг друга:

$$\dot{x}_p = Ax_p + Bu_p; \tag{1}$$

$$\dot{x}_e = Ax_e + Bu_e, \tag{2}$$

где x_p, u_p, x_e, u_e – переменные состояния и переменные управления двух игроков. Среди них переменная состояния содержит информацию о состоянии игроков, и могут существовать различные физические величины, представляющие действия игроков в соответствии с различными игровыми системами. Чтобы облегчить последующие операции в этой статье, переменные состояния здесь должны содержать информацию о местоположении агентов в каждом измерении. Переменные управления содержат элементы, которые реализованы для управления агентами в каждом измерении.

В задаче PE-игры состояние относительного движения агентов очень важно. Итак, пусть δ будет разницей в состояниях между двумя агентами:

$$\delta = x_p - x_e. \tag{3}$$

Преследователь пытается сократить расстояние двух агентов, заложенное в δ, а уклоняющийся пытается его увеличить. Подставив уравнения (1) и (2) в уравнение (3) и вычислив его производную по времени, получим:

$$\dot{\delta} = A\delta + B\left(u_p - u_e\right). \tag{4}$$

Для формулировки игры «преследование–уклонение» (PE) с нулевой суммой построим функцию стоимости интегральной формы:

$$J\left(\delta, u_p, u_e\right) = \int_0^\infty \left(\delta^T Q \delta + u_p^T R_p u_p - u_e^T R_e u_e\right) d\tau,$$

где Q – матрица неотрицательно определённых коэффициентов, R_p и R_e – положительно определённые матрицы. В интегральной функции $\delta^T Q \delta$ – это термин, который измеряет относительное состояние системы (4) и используется для определения пределов расстояния между агентами, $u_p^T R_p u_p$ и $u_e^T R_e u_e$ обозначают масштабы потребления, соответствующие двум агентам, которые используются для реализации ограничений средств управления.

Функция стоимости задаётся следующим образом, когда агенты выполняют определённую политику: $V\left(\delta\right) = \int\limits_0^\infty \left(\delta^T Q \delta + u_p^T R_p u_p - u_e^T R_e u_e\right) d\tau$.

Если и преследователь, и уклоняющийся применяют свои оптимальные стратегии на оптимальных путях, то оптимальное значение игры можно получить как:

$$V^*\left(\delta\right) = \min_{u_p} \max_{u_e} J = \min_{u_p} \max_{u_e} \int_0^\infty \left(\delta^T Q \delta + u_p^T R_p u_p - u_e^T R_e u_e\right) d\tau.$$

Целью работы является выяснение политики управления каждого агента. Сложность работы заключается в поиске численного решения политики каждого агента, в котором важны этапы итерации политики и выбор подходящей аппроксимации функции стоимости. В обучении с подкреплением политика требует некоторых итеративных шагов.

2. Решение игры «преследование–уклонение»

Заменим динамическую модель задачи PE-игры на принцип минимакса и получим аналитическое равновесие Нэша PE-игры.

PE-игра агентов рассматривается как своего рода дифференциальная игра, основанная на теории двустороннего оптимального управления. Оптимальные политики агентов получаются с использованием принципа минимакса. Дифференциальная игра относится к непрерывной игре с парой игроков в системах с непрерывным

временем. Каждый агент пытается достичь своей цели и увеличить свою выгоду. Игра закончится тем, что каждый участник достигнет политики равновесия Нэша. Используя теорему о минимаксе, мы можем гарантировать, что политика агентов является соответствующей оптимальной политикой.

В задаче PE-игры для двух игроков оптимальная политика преследователя пытается минимизировать функцию Гамильтона, тогда как политика уклоняющегося пытается максимизировать её. Следовательно, существует пара оптимальных политик $\left(u_p^*, u_e^*\right)$. Когда преследователь принимает решение u_p^*, а уклоняющийся принимает решение u_e^*, игра достигает равновесия Нэша. Назовём $\left(u_p^*, u_e^*\right)$ седловой точкой игры.

Из уравнений (1) и (2) имеем:

$$H\left(\delta\left(t\right), \nabla V, u_p, u_e\right) = \delta^T Q \delta + u_p^T R_p u_p - u_e^T R_e u_e + \dot{V} =$$
$$= \delta^T Q \delta + u_p^T R_p u_p - u_e^T R_e u_e + \nabla V^T \left(A\left(x_p - x_e\right) + B\left(u_p - u_e\right)\right),$$

где $H\left(\delta\left(t\right), \nabla V, u_p, u_e\right)$ – гамильтониан, u_p, u_e – допустимые политики управления преследователя и уклоняющегося соответственно; $\nabla V = \dfrac{\partial V}{\partial \delta}$.

Можно получить оптимальное управление каждым агентом согласно стационарному условию: $\dfrac{\partial H}{\partial u_p} = \dfrac{\partial H}{\partial u_e} = 0$.

Вторая производная гамильтониана по u_p, u_e должна удовлетворять:

$$\frac{\partial^2 H}{\partial u_p^2} = 2R_p > 0; \frac{\partial^2 H}{\partial u_e^2} = -2R_e < 0.$$

Оптимальные управления агентов получаются в виде:

$$u_p^* = -\tfrac{1}{2} R_p^{-1} B^T \nabla V^*;$$
$$u_e^* = -\tfrac{1}{2} R_e^{-1} B^T \nabla V^*.$$

Значение V можно аналитически получено решением следующего уравнения Гаильтона–Якоби–Беллмана (HJB):

$$\delta^T Q \delta + \left(u_p^*\right)^T R_p u_p^* - \left(u_e^*\right)^T R_e u_e^* + \left(\Delta V^*\right)^T \left(A\delta + B\left(u_p^* - u_e^*\right)\right) = 0.$$

Поскольку поведение двух агентов преследования–уклонения становится игрой с нулевой суммой, когда оба агента принимают свою оптимальную политику, которая называется теоретико-игровой политикой седловой точки, игра достигнет равновесия Нэша при этом условии.

Предположим, что V^* удовлетворяет уравнению HJB, в результате чего гамильтониан $H\left(\delta\left(t\right), \nabla V^*, u_p^*, u_e^*\right)$ обращается в 0. Тогда гамильтониан $H\left(\delta\left(t\right), \nabla V^*, u_p, u_e\right)$ преобразуется в

$$H\left(\delta\left(t\right), \nabla V^*, u_p, u_e\right) = \ = \left(\nabla V^*\right)^T B\left(\left(u_p - u_p^*\right) - \left(u_e - u_e^*\right)\right) +$$
$$+ \delta^T Q \delta + u_p^T R_p u_p - \left(u_p^*\right)^T R_p u_p^* - \left(u_e^T R_e u_e - \left(u_e^*\right)^T R_e u_e^*\right).$$

3. Численное решение PE-игры методом ADP

3.1. Итерация политики

Упростим функцию стоимости PE-игры:

$$V\left(\delta\left(t\right)\right)=\int_{t}^{\infty}r\left(\delta\left(\tau\right),u_{p}\left(\tau\right),u_{e}\left(\tau\right)\right)d\tau,$$

для $r\left(\delta\left(\tau\right),u_{p}\left(\tau\right),u_{e}\left(\tau\right)\right)=\delta^{T}Q\delta+u_{p}^{T}R_{p}u_{p}-u_{e}^{T}R_{e}u_{e}$. Для заданного интервала T можно записать:

$$V\left(\delta\left(t\right)\right)=\int_{t}^{t+T}r\left(\delta\left(\tau\right),u_{p}\left(\tau\right),u_{e}\left(\tau\right)\right)d\tau+\int_{t+T}^{\infty}r\left(\delta\left(\tau\right),u_{p}\left(\tau\right),u_{e}\left(\tau\right)\right)d\tau=$$
$$=\int_{t}^{t+T}r\left(\delta\left(\tau\right),u_{p}\left(\tau\right),u_{e}\left(\tau\right)\right)d\tau+V\left(\delta\left(t+T\right)\right).$$

Интервал T рассматривается как параметр ADP. Разделим весь период $[t,\infty)$ на сегменты интервалов и предположим, что $[t,t+T]$ – k-й интервал PE-игры, т. е. $t=k\cdot T$. Более того, политики, выполняемые двумя агентами в $[t,t+T]$, равны $u_{p}\left(\delta_{k}\right),u_{e}\left(\delta_{k}\right)$.

Тогда имеем:

$$V\left(\delta_{k}\right)=\int_{kT}^{(k+1)T}r\left(\delta,u_{p}\left(\delta_{k}\right),u_{e}\left(\delta_{k}\right)\right)d\tau+V\left(\delta_{k+1}\right).$$

Можно получить политики управления преследователя и уклоняющегося в виде:

$$u_{p}=-\frac{1}{2}R_{p}^{-1}B^{T}\nabla V\left(\delta_{k}\right);$$
$$u_{e}=-\frac{1}{2}R_{e}^{-1}B^{T}\nabla V\left(\delta_{k}\right).$$

Уравнения для $V^{(i)}\left(\delta_{k}\right)$ образуют i-й итерационный цикл метода PI с политиками управления $u_{p}^{(i)},u_{e}^{(i)}$. Для PE-игры пусть $u_{p}^{(0)},u_{e}^{(0)}$ являются допустимыми начальными управлениями преследователя и уклоняющегося. Функции $V^{(i)}\left(\delta_{k}\right)$ и управления $u_{p}^{(i)},u_{e}^{(i)}$ при $i\to\infty$ будут сходиться к $V^{*}\left(\delta\right),u_{p}^{*},u_{e}^{*}$, соответственно. Игра достигает равновесия Нэша, когда элементы управления сходятся.

3.2. Аппроксимация функции стоимости

Для большинства игр PE уравнение HJB сложно решить аналитически или оно может вообще не иметь аналитического решения. Поэтому мы используем процесс аппроксимации для получения решения уравнения HJB. Метод фокусируется на аппроксимации функции значения, которая называется алгоритмом VFA (Value Function Approximation).

Предположим, что линейно независимое интегрирование набора базисных функций $\varphi_j(\delta)$ способно аппроксимировать функцию стоимости V, которая выражается как: $V(\delta(t)) = \sum_{j=1}^{L} w_j \varphi_j(\delta) = w_L^T \phi_L(\delta)$, где L обозначает количество сохранённых функций, а $\phi_L(\delta)$ образует вектор L-размерности базовых функций. w обозначает определяемые параметры нейронной сети, которая состоит из каждого элемента $w_j\,(j = 1, ..., L)$.

Используя приведённую выше аппроксимацию функции стоимости (VFA) для функции стоимости, уравнение итерации политики можно выразить в виде:

$$w_L^T \phi_L(\delta_t) = \int_t^{t+T} r(\delta, u_p, u_e)\, d\tau + w_L^T \phi_L(\delta_{t+T}).$$

Представим дискретное время t в виде $t = kT$ и $\phi(\delta(t)) = \phi(\delta_k); \phi(\delta(t+T)) = \phi(\delta_{k+1})$ (индекс L для простоты опущен). Тогда

$$\left(w^{(i)}\right)^T \Phi(\delta_k) \approx \delta_k^T Q \delta_k + \left(h^{(i)}(\delta_k)\right)^T R h^{(i)}(\delta_k), \tag{5}$$

где $\Phi(\delta_k) = \varphi(\delta_k) - \gamma\varphi(\delta_{k+1})$, γ – коэффициент дисконтирования.

Обновить политику управления можно по формулам:

$$\begin{aligned} h_p^{(i+1)} &= -\frac{1}{2} R_p^{-1} B^T \nabla\phi^T w^{(i)}; \\ h_e^{(i+1)} &= -\frac{1}{2} R_e^{-1} B^T \nabla\phi^T w^{(i)}. \end{aligned} \tag{6}$$

Выражение для $\left(w^{(i)}\right)^T$ (5) представляет собой скалярное уравнение, тогда как вектор неизвестных параметров $w^{(i)} \in R^L$ имеет L элементов. Следовательно, для нахождения $w^{(i)}$ необходимы данные за несколько временных шагов.

На шаге i алгоритма итерации стратегия управления фиксируется на уровне $u = h^{(j)}(\delta)$. В каждый момент времени k измеряется набор данных $\left(\delta_k, \delta_{k+1}, r\left(\delta_k, h^{(i)}(\delta_k)\right)\right)$. Затем выполняется один шаг метода наименьших квадратов. Эта итерационная процедура повторяется до достижения параметров, соответствующих значению $V^*(\delta) = (w^*)^T \cdot \varphi(\delta)$.

4. Численное моделирование

Рассмотрим проведение численного моделирования игры «преследование-уклонение». На основе общей модели движения исследуется задача преследования-уклонения, рассматривающая в качестве управления ускорение обоих игроков. Положение и скорость агентов отслеживаются онлайн как переменные состояния.

Рассмотрим задачу PE-игры в двумерном пространстве, динамическая модель которой будет следующей:

$$\dot{s}_{px} = v_{px}; \dot{s}_{py} = v_{py}; \dot{v}_{px} = a_{px}; \dot{v}_{py} = a_{py}; \tag{7}$$

$$\dot{s}_{ex} = v_{ex}; \dot{s}_{ey} = v_{ey}; \dot{v}_{ex} = a_{ex}; \dot{v}_{ey} = a_{ey}, \tag{8}$$

где $s_{px}, s_{py}, v_{px}, v_{py}$ – координаты и скорости преследователя в направлениях x и y соответственно. Аналогично, $s_{ex}, s_{ey}, v_{ex}, v_{ey}$ – это координаты и скорости уклоняющегося в направлениях x и y, соответственно; (a_{px}, a_{py}) и (a_{ex}, a_{ey}) – это пары ускорений двух агентов, которые обозначают политики управления двух агентов, соответственно.

Вычтем модель (7) из (8) и получим систему разностей, переменные состояния которой равны $\delta = [l_x, \Delta v_x, l_y, \Delta v_y]$. Среди них обозначают расстояние в направлениях x и y соответственно. Полная система разностной модели:

$$\dot{l}_x = \Delta v_x; \Delta \dot{v}_x = a_{px} - a_{ex}; \dot{l}_y = \Delta v_y; \Delta \dot{v}_y = a_{py} - a_{ey},$$

или

$$\dot{\delta} = A\delta + B\left(a_p - a_e\right), \tag{9}$$

где

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}; B = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

Расстояние между двумя агентами можно рассматривать как условие захвата задачи PE-игры, которое задаётся следующим образом: $l = \sqrt{\left(s_{px} - s_{ex}\right)^2 + \left(s_{py} - s_{ey}\right)^2}$.

Примем матрицу Q в виде $Q = \operatorname{diag}\left(\begin{bmatrix} 1, & 0, & 1, & 0 \end{bmatrix}\right)$, $R_p = 0.3, R_e = 1$. Значения начальных состояний задаются в виде: $\delta^{(0)} = \begin{bmatrix} 1; & 0; & 1; & 0 \end{bmatrix}$, $a_{e0} = a_{p0} = 0$.

Построим однослойную нейронную сеть следующим образом:

$$\begin{aligned} V &= \sum_{j=1}^{6} w_j \varphi_j = \\ &= w_1 l_x^2 + w_2 l_x \Delta v_x + w_3 \Delta v_x^2 + w_4 l_y^2 + w_5 l_y \Delta v_y + w_6 \Delta v_y^2. \end{aligned} \tag{10}$$

Параметры $w^{(i)} = \begin{pmatrix} w_1^{(i)} & \dots & w_6^{(i)} \end{pmatrix}$ обновляются онлайн. Начальное значение каждой компоненты $w_j^{(0)}$ примем равным $w_j^{(0)} = 0.1 \cdot (rand() - 0.5); j = 1, \dots, 6$.

Функцию активации вектора критика выберем в виде:

$$\varphi(\delta) = \begin{bmatrix} l_x^2 & l_x \Delta v_x & \Delta v_x^2 & l_y^2 & l_y \Delta v_y & \Delta v_y^2 \end{bmatrix};$$

тогда:

$$(\nabla\varphi(\delta))^T = \begin{bmatrix} 2l_x & 0 & 0 & 0 \\ \Delta v_x & l_x & 0 & 0 \\ 0 & 2\Delta v_x & 0 & 0 \\ 0 & 0 & 2l_y & 0 \\ 0 & 0 & \Delta v_y & l_y \\ 0 & 0 & 0 & 2\Delta v_y \end{bmatrix}^T.$$

Примем политику управления в форме (6).

Для нахождения вектора w запишем систему уравнений для i-й итерации:

$$\begin{aligned} &\left(w^{(i)}\right)^T \cdot \Phi(1) = r\left(\delta_1, \delta_{1+1}, r\left(\delta_1, h^{(i)}(\delta_1)\right)\right); \\ &\dots \\ &\left(w^{(i)}\right)^T \cdot \Phi(k) = r\left(\delta_k, \delta_{k+1}, r\left(\delta_k, h^{(i)}(\delta_k)\right)\right); \\ &\dots \end{aligned} \tag{11}$$

Так как размерность вектора $w^{(i)}$ равна $L = 6$, то необходимо выполнение условия $k \geqslant 6$.

Сформируем матрицу

$$\Phi = \begin{pmatrix} \Phi(1) \\ \vdots \\ \Phi(k) \end{pmatrix}$$

и объединим скаляры $r\left(\delta_1, \delta_{1+1}, r\left(\delta_1, h^{(i)}(\delta_1)\right)\right), \dots, r\left(\delta_k, \delta_{k+1}, r\left(\delta_k, h^{(i)}(\delta_k)\right)\right), \dots$ в вектор

$$Rew^{(i)} = \begin{pmatrix} r\left(\delta_1, \delta_{1+1}, r\left(\delta_1, h^{(i)}(\delta_1)\right)\right) \\ \vdots \\ r\left(\delta_k, \delta_{k+1}, r\left(\delta_k, h^{(i)}(\delta_k)\right)\right) \end{pmatrix}.$$

Тогда (11) можно записать в виде:

$$\left(w^{(i)}\right)^T \Phi = Rew^{(i)}. \tag{12}$$

Представим решение (12) относительно $w^{(i)}$ как решение задачи минимизации выражения:

$$\left\|\Phi\left(w^{(i)}\right)^T - Rew^{(i)}\right\|_2^2 + \left\|\Gamma\left(w^{(i)}\right)^T\right\|_2^2 \to \min_{w^{(i)}},$$

с членом регуляризации А.Н. Тихонова [8] $\left\|\Gamma\left(w^{(i)}\right)^T\right\|_2^2$, где матрица Γ выбирается как скаляр α, кратный единичной матрице $\Gamma = \alpha \cdot I$. Тогда решение для оценки $\hat{w}^{(i)}$ может быть получено методом наименьших квадратов:

$$\left(\hat{w}^{(i)}\right)^T = \left(\Phi^T\Phi + \alpha^2 I\right)^{-1} \Phi^T \cdot Rew^{(i)}. \tag{13}$$

В результате получим: $w^* = [1.0; 0.5; 0.49; 1.0; 0.5; 0.5]$, т. е. функция стоимости аппроксимируется функцией:

$$V(\delta) = 1.0 \cdot l_x^2 + 0.5 \cdot l_x \Delta v_x + 0.49 \cdot \Delta v_x^2 + 1.0 l_y^2 + 0.5 \cdot l_y \Delta v_y + 0.5 \cdot \Delta v_y^2.$$

Заключение

В работе обсуждается решение игры «преследование–уклонение» двух игроков. С помощью принципа минимакса получено аналитическое решение равновесия Нэша и обсуждено необходимое условие, вызывающее возникновение захвата. Метод PI используется при решении онлайн-игры PE, а алгоритм VFA используется для предотвращения возможных неудобств при работе с уравнением HJB. Нет необходимости знать матрицу системы для получения политик, и игра приближается к аналитическому равновесному решению Нэша, которое проверяется в моделировании.

В будущем предполагается изучать более сложные задачи PE-игр с большим количеством агентов. Случай, когда существует связь между переменными состояния или переменными управления, также заслуживает дальнейшего изучения.

Литература

1. Isaacs R. Games of Pursuit. Rand; Santa Monica: Dover Publication Inc. 1965. 384 p.
2. Чуканов С.Н., Чуканов И.С., Цыганенко В.Н. Дифференциальные игры. Омск: ОмГТУ, 2024. 160 с.
3. Friedman A. Differential Games. John Wiley & Sons Inc., 1971. 370 p.
4. Yu Z. Linear–quadratic optimal control and nonzero-sum differential game of forward–backward stochastic system // Asian Journal of Control. 2011. No. 14. P. 173–185.
5. Werbos P.J., Miller W., Sutton R. A menu of designs for reinforcement learning over time // Neural Networks for Control. Cambridge, MA, USA: MIT Press, 1990. Vol. 3. P. 67–95.
6. Bertsekas D.P., Tsitsiklis J.N. Neuro-dynamic programming: An overview // Proceedings of the 1995 34th IEEE Conference on Decision and Control. New Orleans, LA, USA, 13–15 December, 1995. Vol. 1. P. 560–564.
7. Werbos P. Approximate dynamic programming for realtime control and neural modelling // Handbook of Intelligent Control: Neural, Fuzzy and Adaptive Approaches. New York, NY, USA: Van Nostrand Reinhold, 1992. P. 493–525.
8. Тихонов А.Н., Арсенин В.Я. Методы решения некорректных задач. М.: Наука, 1979. 285 с.

DIFFERENTIAL PURSUIT-EVASION GAME BASED ON REINFORCEMENT LEARNING

S.N. Chukanov[1]
Dr.Sc. (Technical), Professor, Leading Researcher, e-mail: a@a.ru
I.S. Chukanov[2]
Student, e-mail: chukanov022@gmail.com
S.V. Leykhter[3]
Assistant Professor, e-mail: leykhter@mail.ru

[1]Omsk branch of Sobolev Institute of Mathematics of the SB RAS, Omsk, Russia
[2]Ural Federal University named after the first President of Russia B.N. Yeltsin, Ekaterinburg, Russia
[3]Dostoevsky Omsk State University, Omsk, Russia

Abstract. The paper discusses optimal control algorithms based on actor/critic reinforcement learning (RL) schemes. The algorithms are used to solve pursuit-evasion (PE) problems of differential games. The work focuses on the implementation of agent policy decisions in accordance with the concept of adaptive dynamic programming. The essence of solving the PE game problem is to obtain the control policy of each agent (pursuer and evader) on both sides of the game. The paper proposes an adaptive dynamic programming (ADP) method for solving Nash equilibrium policies in differential pursuit-evasion games for two players. The cost function approximation method is used to calculate the parameters of a neural network (NN) without directly solving the Hamilton–Jacoby equation.

Keywords: optimal control, machine learning, reinforcement learning.

Дата поступления в редакцию: 01.05.2024