ISSN 2221-5646(Print) 2523-4641(Online)
Visnyk of V.N.Karazin Kharkiv National University
Ser. “Mathematics, Applied Mathematics and Mechanics”
2024, Vol. 99, p. 22–35
DOI: 10.26565/2221-5646-2024-99-02
УДК 519.6

Вісник Харківського національного університету імені В.Н. Каразіна
Серія "Математика, прикладна математика і механіка"
2024, Том 99, с. 22–35

В. В. Карєва
викладач кафедри прикладної математики
Харківський національний університет імені В. Н. Каразіна
майдан Свободи, 4, Харків, Україна, 61022
valerija.kareva@gmail.com *http://orcid.org/0000-0003-2121-5214*

С. В. Львов
науковий співробітник науково-дослідницького інституту біології
Харківський національний університет імені В. Н. Каразіна
майдан Свободи, 4, Харків, Україна, 61022
lvovser@gmail.com *http://orcid.org/0000-0003-4055-7172*

Методи адаптивно динамічного програмування для визначення оптимальної стратегії регенерації печінки

Кожен живий організм взаємодіє з навколишнім середовищем і використовує цю взаємодію для вдосконалення власних дій, щоб вижити та розвиватися. Процес еволюції показав, що види змінюють свої дії на основі взаємодії з навколишнім середовищем протягом тривалого часу, що призводить до природного відбору та виживання найбільш пристосованих. Це навчання, яке засноване на діях, або навчання з підкріпленням може охопити уявлення про оптимальну поведінку, що відбувається в природних системах. Ми описуємо математичні формулювання для навчання з підкріпленням і метод практичного впровадження, відомий як адаптивне динамічне програмування. Це дає нам уявлення про вигляд керування для штучних біологічних систем, які навчаються та демонструють оптимальну поведінку.

У даній роботі розглядається постановка задачі верхньої оцінки оптимальності, для якої оптимальна стратегія регуляції гарантовано краща чи еквівалентна об’єктивним правилам регуляції, які ми можемо спостерігати в реальних біологічних системах.

У випадку оптимальних алгоритмів навчання з підкріпленням процес навчання переміщується на вищий рівень, об’єктом інтересу якого є не деталі динаміки системи, а індекс продуктивності, який кількісно визначає, наскільки близько до оптимальності працює система керування. У такій схемі навчання з підкріпленням є засобом навчання оптимальній поведінці шляхом спостереження за реакцією оточення на неоптимальні стратегії керування.

Мета цієї статті полягає в тому, щоб показати корисність методів навчання з підкріпленням, зокрема сімейства методів, відомих як адаптивне динамічне програмування (АДП), для керування біологічними системами за допомогою зворотного зв’язку. У цій роботі викладено «он-лайн» методи вирішення задачі визначення верхньої оцінки оптимальності у постановці адаптивного динамічного програмування.

Ключові слова: **динамічне програмування; оптимальне керування; навчання з підкріпленням.**

2020 Mathematics Subject Classification: 90C39, 65K05.

1. Вступ

Розробка математичних моделей динаміки складних клітинних систем, що володіють задовільною пояснювальною та передбачувальною силою, є однією з фундаментальних проблем математичної біології.

Без явного уявлення принципів, правил і механізмів цілеспрямованої регуляції (керування) у «клітинних системах» будь-яка їхня математична модель дасть нам лише неосяжний набір потенційних стратегій, серед яких є справжня динаміка, що спостерігається в біологічному експерименті.

Ідентифікація об'єктивних принципів і правил регуляції «клітинної системи», що визначає серед усіх можливостей саме справжню динаміку, є необхідною умовою розробки математичних моделей із достатньою пояснювальною та передбачуваною силою.

Перспективним підходом до розв'язання цієї задачі є гіпотеза, що правила регуляції біологічних процесів підпорядковані деяким об'єктивним принципам, критеріям оптимальності[1]. Ця гіпотеза виникла з природнього припущення, що принципи, яким підкоряються правила регуляції процесів відновлення динамічного гомеостазу органів та тканин організму, відповідають процесу природного відбору під час його попередньої еволюції щодо деякого критерію оптимальності [2, 3].

Наразі розв'язати цю задачу, навіть у її спрощеній постановці, досить важко через безліч невизначеностей під час попередньої еволюції організму, динаміки зміни зовнішніх умов, в яких вона відбувалася, а також високої обчислювальної складності розв'язання такої задачі.

Розглядається значно простіша постановка задачі верхньої оцінки оптимальності, для якої оптимальна стратегія регуляції гарантовано краща чи еквівалентна об'єктивним правилам регуляції, які ми можемо спостерігати в реальних біологічних системах.

Задача пошуку верхньої оцінки оптимальності може оцінити особливості регуляції регенераційних процесів в сценаріях, які не охоплені у біологічних експериментах, та проаналізувати можливі процеси регуляції, спостереження яких є поки що технологічно недосяжні. Ця задача дозволяє на якісному рівні попередньо перевірити гіпотези щодо того, як відбувається регуляція процесів регенерації печінки з метою їх подальшої перевірки в біологічному експерименті.

Як було зазначено у роботах [4, 5] у регенерації печінки беруть участь процеси різного часового масштабу. При цьому швидкі процеси можуть відігравати важливу роль. При цьому часовий горизонт розгляду процесів регенерації організму може становити від тижня до кількох місяців.

Тому задача пошуку стратегії регенерації печінки є експоненційно складною щодо відліків часової шкали координат, що кодують переходи з попереднього стану безпосередньо у наступний стан. Природньо постає питання

про знаходження ефективних методів наближеного розв'язання цієї задачі за наявності тих чи інших обмежень.

Одним із найуспішніших методів є адаптивне динамічне програмування (АДП) та навчання з підкріпленням. Модифікацію дій на основі взаємодії з навколишнім середовищем навчанням з підкріпленням (reinforcement learning, RL) [6]. Існує багато типів навчання, включаючи контрольоване навчання, неконтрольоване навчання тощо. Навчання з підкріпленням стосується агента, який взаємодіє зі своїм середовищем і змінює свої дії або стратегію керування на основі стимулів, отриманих у відповідь на його дії. Це базується на оціночній інформації з навколишнього середовища і може називатися навчанням, заснованим на діях. RL передбачає причинно-наслідковий зв'язок між діями та винагородою чи покаранням.

Алгоритми RL побудовані на ідеї, що успішні контрольні рішення слід запам'ятовувати за допомогою сигналу підкріплення, щоб вони з більшою ймовірністю були використані вдруге. RL тісно пов'язаний з теоретичної точки зору з прямими та непрямими адаптивними методами оптимального керування. Адаптивне динамічне програмування та навчання з підкріпленням, крім «офлайн» методів передбачає «он-лайн» методи, що працюють у реальному режимі часу і які, зрештою, не вимагають знання рівнянь динаміки системи – методи, що базуються на даних, у тому числі методів, які обробляють дані, що надходять у реальному масштабі часу [12].

У цій роботі ми розглянемо «он-лайн» методи вирішення задачі визначення верхньої оцінки оптимальності у постановці адаптивного динамічного програмування.

2. Постановка задачі АДП визначення стратегії верхньої оцінки оптимальності регенерації печінки організму.

Попередньо було розроблено дискретну, детерміновану, автономну, керовану динамічну систему $S(X, U, f)$ у термінах індексів дискретних часів t [7]:

$$x_{t+1} = f(x_t, \tau_t, \lambda_t), 0 \leq \lambda_t \leq 1, x_0 = x^0, x_t \in X, \lambda_t \in U, t \in \mathbb{N}, \qquad (1)$$

де x_t - типи функціональних клітин печінки в момент часу t; $X \subseteq \mathbb{R}^n$ - простір допустимих станів системи; $x^0 \in X$ - заданий початковий розподіл функціональних клітин печінки; $U \subseteq \mathbb{R}^m$ - простір допустимих керуючих дій; τ_t - задана функція зовнішньої токсичності.

Функція $f(x_t, \tau_t, \lambda_t)$ задана у явному вигляді для моделі регенерації печінки, яку ми розробили [4, 7]. Запропонована модель процесів регенерації печінки включає такі моделі популяційної динаміки, як узагальнені рівняння Лотки-Вольтерра, рівняння Лотки-Вольтерра з переходами, рівняння Лотки-Вольтерра із запізненням.

Така система задовольняє однокроковій властивісті Маркова, оскільки її стан у момент часу $t + 1$ залежить лише від стану та вхідних даних у попередній момент часу t.

Для полегшення аналізу часто розглядають клас систем з дискретним часом, що описуються нелінійною динамікою у формі різницевого рівняння афінного простору станів:

$$x_{t+1} = g_1(x_t) + g_2(x_t)\lambda_t. \tag{2}$$

Аналіз таких форм зручний і може бути узагальнений на загальну вибірку даних форми (1).

Стратегія керування визначається як функція від простору станів до простору керування $\lambda : X \to U$. Тобто для кожного стану x_t стратегія визначає керуючу дію $\lambda_t = \lambda(x_t)$.

Кожному дискретному переходу системи з поточного стану x_t у наступний стан x_{t+1} під дією керування $x_t \xrightarrow{\lambda_t} x_{t+1}$ приписується його вартість:

$$r_t = r(x_t, \lambda_t),$$

де $r : X \times U \to \mathbb{R}$ є мірою вартості керування за один крок (utility).

Далі припускатимемо, що функція r обмежена. Принаймні для біологічних систем припущення, що вартість, ефективність, корисність стану не може бути необмежено великим, природно.

2.1. Оптимальна вартість.

На відміну від розглянутої в [7] постановки задачі визначення оптимальної стратегії, у задачі АДП розглядають дисконтовану сумарну вартість вперед або собівартість:

$$V_\lambda(x_t) = \sum_{i=t}^{\infty} \gamma^{i-t} r(x_i, \lambda_i), \tag{3}$$

$0 < \gamma \leq 1$ - коефіцієнт дисконтування. Коефіцієнт дисконтування відображає той факт, що ми менше турбуємося про обставини, які виникнуть в майбутньому.

Ми припускаємо, що система є стабілізованою на деякій множині $\Omega \subset \mathbb{R}^n$, тобто існує стратегія керування $\lambda_t = \lambda(x_t)$ така, що замкнута система (2) є асимптотично стійкою на Ω. Стратегія керування λ_t називається допустимою, якщо вона є стабілізуючою і дає кінцеву вартість $V_\lambda(x_t)$.

Метою теорії оптимального керування є вибір стратегії, яка мінімізує собівартість:

$$V^*(x_t) = \min_{\lambda(.)} \left(\sum_{i=t}^{\infty} \gamma^{i-t} r(x_i, \lambda(x_i)) \right), \tag{4}$$

яка відома як оптимальна вартість. Тоді оптимальна стратегія керування визначається як:

$$\lambda^*(x_t) = \arg\min_{\lambda(.)} \left(\sum_{i=t}^{\infty} \gamma^{i-t} r(x_i, \lambda(x_i)) \right). \tag{5}$$

Раніше, наприклад, ми визначали оптимальну стратегію керування як [7]:

$$\lambda_t^* = \arg\min_{\lambda(.)} \sum_{i=0}^{N} v_{t_i}(K - \Phi_{t_i})^2 \tag{6}$$

$\Phi_t = \sum_{i=1}^{n} c_i x_t^i$ – узагальнений показник функціональності печінки в момент часу t.

$t_i = i\Delta t, \Delta t$ – крок дискретизації, $[0, T] = \Delta t N$ – інтервал життєвого циклу організму.

K – оптимальна функціональна активність організму.

$0 \leq v_{t_i} \leq 1$ – відносна вага моменту життєвого циклу.

Зауваження. *Задача знаходження стратегії регенерації печінки має фізичний зміст кінцевого інтервалу часу (життєвий цикл організму кінцевий). Але формально систему рівнянь, що описують процеси регенерації печінки, можна продовжити на нескінченну вісь* $t \in \mathbb{N}$.

2.2. Рівняння Ляпунова і принцип оптимальності Беллмана.

Запишемо вираз (3) у вигляді:

$$V_\lambda(x_t) = r(x_t, \lambda_t) + \gamma \sum_{i=t+1}^{\infty} \gamma^{i-t-1} r(x_i, \lambda_i). \tag{7}$$

Вираз (7) еквівалентний наступному:

$$V_\lambda(x_t) = r(x_t, \lambda(x_t)) + \gamma V_\lambda(x_{t+1}), V_\lambda(x_0) = 0. \tag{8}$$

Вираз (8) є дискретним нелінійним рівнянням Ляпунова.

На підставі рівнянь (7) визначимо дискретний Гамільтоніан.

$$H(x_t, \lambda(x_t), \triangle V_t) = r(x_t, \lambda(x_t)) + \triangle V_t, \tag{9}$$

де $\triangle V_t = \gamma V_\lambda(x_{t+1}) - V_\lambda(x_t)$ - різницевий оператор, який виражає зміни дисконтованої вартості вперед під час переходу зі стану x_k у стан x_{k+1}, у результаті керуючої дії $\lambda(x_k)$. З рівняння Ляпунова випливає, що дискретний Гамільтоніан для будь-якого керування та будь-якого поточного стану дорівнює нулю.

Оптимальне значення можна записати за допомогою рівняння Беллмана як

$$V^*(x_t) = \min_{\lambda(.)} \left(r(x_t, \lambda(x_t)) + \gamma V_\lambda(x_{t+1}) \right). \tag{10}$$

Цю задачу оптимізації все ще важко вирішити.

Принцип Беллмана [8] є основою оптимального керування, і він стверджує, що "незважаючи на те, якими були попередні рішення (тобто керування), необхідно вибрати такий варіант керування, щоб собівартість на цьому

та всіх послідуючих кроках були мінімальними". З точки зору рівнянь, це означає, що:

$$V^*(x_t) = \min_{\lambda(.)} \left(r(x_t, \lambda(x_t)) + \gamma V^*(x_{t+1}) \right). \tag{11}$$

Рівняння (11) відоме як рівняння оптимальності Беллмана або рівняння Гамільтона-Якобі-Белмана (HJB) з дискретним часом. Тоді оптимальна стратегія виглядає як:

$$\lambda^*(x_t) = \arg\min_{\lambda(.)} \left(r(x_t, \lambda(x_t)) + \gamma V^*(x_{t+1}) \right). \tag{12}$$

Оскільки необхідно знати оптимальну стратегію в момент часу $t+1$ до (11) для визначення оптимальної стратегії в момент часу t, принцип Беллмана дає зворотну в часі процедуру для вирішення проблеми оптимального керування. Це основа для алгоритмів динамічного програмування, які широко використовуються в теорії систем керування, дослідженні операцій тощо.

Позначимо символами L_S і L_S^* безліч функцій Ляпунова і безліч оптимальних функцій Ляпунова шляхів динамічної системи S (1), відповідно:

$$L_S = \{V_\lambda(x) | x \in X, \lambda \in U\}$$

$$L_S = \{V^*(x) | x \in X, \lambda \in U\}$$

Розглянемо безліч функцій Ляпунова та оптимальних функцій Ляпунова динамічної системи S (1) як підмножини Банахова простору $l_\infty(\mathbb{N})$ обмежених функцій $v : \mathbb{N} \to \mathbb{R}, v \in l_\infty(\mathbb{N})$, з нормою $||v(.)||_\infty = \sup_{i \in \mathbb{N}} |v(i)|$.

З огляду на рівняння Ляпунова (8) визначимо пару відображень T і T^* Банахова простору у себе:

$$T : l_\infty(\mathbb{N}) \to l_\infty(\mathbb{N})$$

$$T(v(t)) = r(x_t, \lambda(x_t)) + \gamma v(t+1), v \in l_\infty(\mathbb{N}), t \in \mathbb{N}$$

$$T^* : l_\infty(\mathbb{N}) \to l_\infty(\mathbb{N})$$

$$T(v(t)) = \min_{\lambda(.)} \left(r(x_t, \lambda(x_t)) + \gamma v(t+1) \right), v \in l_\infty(\mathbb{N}), t \in \mathbb{N}$$

Твердження 1. Відображення $T : l_\infty(\mathbb{N}) \to l_\infty(\mathbb{N})$ і $T^* : l_\infty(\mathbb{N}) \to l_\infty(\mathbb{N})$ є стискаючими відображеннями в Банаховому просторі $l_\infty(\mathbb{N})$.

$\exists \alpha, 0 < \alpha < 1$:
$||v_1(t) - v_2(t)||_\infty \geq \alpha ||T(v_1(t)) - T(v_2(t))||_\infty, \forall v_1, v_2 \in l_\infty(\mathbb{N})$,
$||v_1(t) - v_2(t)||_\infty \geq \alpha ||T^*(v_1(t)) - T^*(v_2(t))||_\infty, \forall v_1, v_2 \in l_\infty(\mathbb{N})$.

Твердження 2. Стискаючі відображення $T : l_\infty(\mathbb{N}) \to l_\infty(\mathbb{N})$ і $T^* : l_\infty(\mathbb{N}) \to l_\infty(\mathbb{N})$ мають єдину «нерухому точку» і, якщо нерухома точка відображення T і нерухома точка відображення T^* належать безлічі функцій Ляпунова та безлічі оптимальних функцій Ляпунова динамічної системи S, то ці нерухомі точки є функції Ляпунова та оптимальні функції Ляпунова.

$$\exists! \tilde{v} \in l_\infty(\mathbb{N}) : T(\tilde{v}) = \tilde{v}, \tilde{v} \in L_S \Rightarrow \tilde{v} = V_\lambda(x_t).$$

$$\exists!\tilde{v} \in l_\infty(\mathbb{N}) : T^*(\tilde{v}) = \tilde{v}, \tilde{v} \in L_S^* \Rightarrow \tilde{v} = V^*(x_t).$$

Як відомо доказ теореми Банаха про нерухому точку заснований на послідовній ітераційній процедурі використання стискаючих відображень, в нашому випадку T і T^*. Це є математичним обгрунтуванням ітераційних алгоритмів АДП, які будуть наведені у наступному розділі.

3. Навчання з підкріпленням, АДП та адаптивне керування.

Оптимальним рішенням керування з використанням динамічного програмування є процедура зворотного руху в часі. У цьому розділі сформулюємо методи он-лайн навчання з підкріпленням у реальному часі для вирішення задачі оптимального керування [9, 12]. Ці методи широко називаються наближеним динамічним програмуванням (АДП) або нейродинамічним програмуванням (НДП) [10]. Є два ключових компоненти: похибка часової різниці і апроксимація функції вартості.

Похибка часової різниці. На основі рівняння Беллмана (8) визначимо рівняння похибки часової різниці:

$$e_t = r(x_t, \lambda(x_t)) + \gamma V_\lambda(x_{t+1}) - V_\lambda(x_t). \tag{13}$$

Слід зазначити, що права частина цього виразу є гамільтоновою функцією (9). Якщо виконується рівняння Беллмана, похибка часової різниці дорівнює нулю.

Похибка часової різниці може розглядатися як похибка передбачення між прогнозованою вартістю та спостережуваною вартістю у відповідь на керування, застосоване до системи.

Ключовою особливістю рівняння похибки часової різниці є те, що вона не вимагає знання явних рівнянь динаміки системи. Справді, якщо ми маємо такі дані: траєкторія системи, що спостерігається, на основі вимірювань $x_0, x_1, x_2, ...$; функція вартості кроку $r_t = r(x_t, \lambda_t)$; деяке передбачуване керування $\lambda(.)$ або деяка передбачувана оцінка вартості $V(x_t)$, тоді, відповідно до рівняння (1), ми можемо послідовно обчислити її похибку часової різниці.

Апроксимація функції вартості. Для апроксимації функції вартості можуть бути використані такі методи: лінійна регресія, нейронні мережі, дерева прийняття рішень, найближчі сусіди, тощо. Припустимо, що функція вартості може бути досить добре апроксимована найпростішою нейронною мережею (лінійною регресією):

$$\widehat{V}_\lambda(x) = W^T \phi(x). \tag{14}$$

де W - вектор коефіцієнтів (параметрів) нейронної мережі, $\phi(.)$ – базисна функція активації.

Апроксимація функції нейронною мережею означає обчислення параметрів (синаптичних ваг і зміщень, якщо такі є) мережі. Цей процес називається навчанням.

Для деякого керування $\lambda(.)$ похибка часової різниці набуває лінійного за параметрами W вигляду:

$$e_t = r(x_t, \lambda(x_t)) + \gamma W^T \phi(x_{t+1}) - W^T \phi(x_t). \tag{15}$$

Рівняння $e_t = 0$ є рівнянням з фіксованою точкою. Це рівняння узгодженості, яке задовольняється в кожен момент часу t для значення $V_\lambda(.)$, що відповідає поточній стратегії $\lambda(x_t)$. Таким чином, можна використовувати ітераційні процедури для вирішення рівняння часових різниць, включаючи ітерацію за стратегіями та ітерацію за значеннями.

Алгоритм ітерації за стратегіями он-лайн (On-line policy iteration, PI).

Ініціалізація. Виберіть будь-яку допустиму стратегію керування $\lambda_0(x_t)$.

Етап оцінки стратегії. Визначити розв'язок W_{i+1}:

$$W_{i+1}^T(\phi(x_t) - \gamma\phi(x_{t+1})) = r(x_t, \lambda_i(x_t)). \tag{16}$$

Зауважимо, що рівняння форми (16) - це саме ті рівняння, які розв'язуються методом найменших квадратів (least squares method, LS). Таким чином, метод найменших квадратів можна запускати в режимі он-лайн до збіжності. Запишемо (16) як:

$$W_{i+1}^T \Phi(t) = r(x_t, \lambda_i(x_t)). \tag{17}$$

$\Phi(t) = \phi(x_t) - \gamma\phi(x_{t+1})$ – вектор регресії. Зверніть увагу, що для збіжності методу найменших квадратів вектор регресії повинен обертатися.

Тоді:

$$W_{i+1}^T = r(x_t, \lambda_i(x_t))\Phi(t)^{-1}. \tag{18}$$

$$LS(W) = \sum_{t=1}^{T} \left(V_\lambda(x_t) - \widehat{V}_\lambda(x_t)\right)^2. \tag{19}$$

$$LS(W) = \sum_{t=1}^{T} \left(V_\lambda(x_t) - r(x_t, \lambda_i(x_t))\Phi(t)^{-1}\phi(x_t)\right)^2. \tag{20}$$

$$W_{i+1} = \arg\min_W \sum_{t=1}^{T} \left(V_\lambda(x_t) - r(x_t, \lambda_i(x_t))\Phi(t)^{-1}\phi(x_t)\right)^2. \tag{21}$$

Як альтернативу методу найменших квадратів, коли нейронна мережа апроксимації більш складна, можна використовувати метод градієнтного спуску і його модифікації.

Етап удосконалення стратегії. Визначте покращену стратегію за допомогою:

$$\lambda_{i+1}(x_t) = \arg\min_{\lambda(.)} \left(r(x_t, \lambda(x_t)) + \gamma W_{i+1}^T \phi(x_{t+1})\right). \tag{22}$$

Подібним чином можна надати он-лайн алгоритм навчання підкріплення на основі ітерації за значеннями.

Алгоритм ітерації за значеннями он-лайн (On-line value iteration, VI).

Ініціалізація. Виберіть будь-яку стратегію керування $\lambda_0(x_t)$, не обов'язково допустиму або стабілізуючу.

Етап оновлення значення. Визначити розв'язок W_{i+1}:

$$W_{i+1}^T \phi(x_t) = r(x_t, \lambda_i(x_t)) + \gamma W_i^T \phi(x_{t+1}). \tag{23}$$

Для знаходження параметрів W_{i+1} можна використовувати метод найменших квадратів так само як і в PI алгоритмі. Зверніть увагу, що старі параметри вагів знаходяться в правій частині (23). Таким чином, вектор регресії тепер $\phi(x_t)$, який повинен обертатися для збіжності методу найменших квадратів.

$$W_{i+1} = \arg\min_W \sum_{t=1}^{T} \left(V_\lambda(x_t) - (r(x_t, \lambda_i(x_t)) + \gamma W_i^T \phi(x_{t+1}))\phi(x_t)^{-1}\phi(x_t) \right)^2.$$

Для розв'язання в режимі реального часу можна також використовувати пакетні методи найменших квадратів, рекурсивних найменших квадратів або градієнтні методи.

Етап удосконалення стратегії. Визначте покращену стратегію за допомогою:

$$\lambda_{i+1}(x_t) = \arg\min_{\lambda(.)} \left(r(x_t, \lambda(x_t)) + \gamma W_{i+1}^T \phi(x_{t+1}) \right). \tag{24}$$

Алгоритм навчання з підкріпленням PI (або VI) розв'язує нелінійне рівняння Ляпунова на етапі оновлення значення кожного кроку i, спостерігаючи лише набір даних $x_t, x_{t+1}, r(x_t, \lambda_i(x_t))$ кожного разу вздовж траєкторій системи.

Таким чином, навчання з підкріпленням вирішує базове нелінійне рівняння Ляпунова (рівняння Беллмана) на кожному кроці в режимі он-лайн, використовуючи лише дані, що спостерігаються вздовж траєкторій системи.

Зауважимо, що втілення (16) не може бути легко реалізоване в нелінійному випадку, оскільки воно є неявно в керуванні, оскільки x_{t+1} залежить від $\lambda(.)$ і є аргументом нелінійної функції активації.

Ці проблеми вирішуються введенням другої нейронної мережі для стратегії керування, відомої як нейронна мережа діяча [11]. Тому введемо структуру параметричного апроксиматора діяча:

$$\lambda_t = \lambda(x_t) = U^T \sigma(x_t). \tag{25}$$

$\sigma(x) : \mathbb{R}^n \times U \to \mathbb{R}^M$ – вектор M функцій активації і $U \in \mathbb{R}^{M \times m}$ – матриця вагових коефіцієнтів або невідомих параметрів.

Реалізація навчання з підкріпленням з використанням двох нейронних мереж, однієї як критика, а іншої як діяча, дає структуру, показану на рис. 1. У цій системі керування критик і діяч налаштовуються послідовно як в PI, так

Рис. 1. Навчання з підкріпленням зі структурою актор/критик.
Pic.1. Reinforcement learning with an actor/critic framework.

і в VI. Тобто ваги однієї нейронної мережі зберігаються постійними, а ваги іншої налаштовуються до збіжності. Ця процедура повторюється до тих пір, поки обидві нейроннні мережі не зійдуться. Таким чином, це адаптивна система оптимального керування он-лайн, у якій параметри функції значення налаштовуються в режимі он-лайн, а збіжність відбувається до оптимального значення та керування. Збіжність нелінійної ітерації за значеннями з використанням двох нейронних була доведена в [13].

4. Q-навчання.

Щоб уникнути будь-якої інформації про динаміку системи, потрібно надати альтернативний шлях для отримання часткових похідних відносно вхідних даних керування, які не проходять через систему. Для цього Пол Вербос використав концепцію зворотного поширення, а Кріс Воткінс ввів подібні поняття для марковського процесу вирішування у дискретному просторі, який він назвав Q-навчанням [14].

Розглянемо рівняння Беллмана (8), яке дозволяє обчислити цінність будь-якої заданої допустимої стратегії $\lambda(.)$. Оптимальне керування визначається за допомогою (5) або (12). Отже, давайте визначимо функцію Q (quality), пов'язану зі стратегією $\lambda_t = \lambda(x_t)$:

$$Q_\lambda(x_t, \lambda_t) = r(x_t, \lambda_t) + \gamma V_\lambda(x_{t+1}). \tag{26}$$

Зауважте, що функція Q є функцією як стану x_t, так і керування λ_t у момент часу t. Вона відповідає «якості» дії, обраної в поточному стані. Визначимо оптимальну функцію Q:

$$Q^*(x_t, \lambda_t) = r(x_t, \lambda_t) + \gamma V^*(x_{t+1}). \tag{27}$$

З точки зору Q^*, можна записати рівняння оптимальності Беллмана і оптимальне керування в дуже простій формі:

$$V^*(x_t) = \min_{\lambda} \left(Q^*(x_t, \lambda)\right), \lambda^*(x_t) = \arg\min_{\lambda} \left(Q^*(x_t, \lambda)\right). \tag{28}$$

Під час вивчення функції вартості необхідно навчити та зберегти оптимальне значення для всіх можливих станів x_t. На відміну від цього при Q-навчанні потрібно зберігати оптимальну функцію Q для всіх значень (x_t, λ_t), тобто для всіх можливих керуючих дій, що виконуються в кожному можливому стані. Це набагато більше інформації.

Щоб застосувати методи підкріплення он-лайн для вивчення функції Q, потрібно визначити: рівняння з фіксованою точкою для Q і відповідну структуру параметричного апроксиматора для Q.

«Рівняння Беллмана» для Q є

$$Q_\lambda(x_t, \lambda(x_t)) = r(x_t, \lambda(x_t)) + \gamma Q_\lambda(x_{t+1}, \lambda(x_{t+1})). \tag{29}$$

Оптимальне значення Q задовольняє:

$$Q^*(x_t, \lambda_t) = r(x_t, \lambda_t) + \gamma Q^*(x_{t+1}, \lambda^*(x_{t+1})). \tag{30}$$

Рівняння (29) є рівнянням з фіксованою точкою або «рівнянням Беллмана» для Q. Тепер можна використовувати будь-який он-лайн метод навчання з підкріпленням вище як основу для АДП, включаючи PI та VI.

Для нелінійних систем допускається параметричний апроксиматор або нейронна мережа вигляду:

$$\widehat{Q}_\lambda(x, \lambda) = W^T \phi(x, \lambda). \tag{31}$$

де $\phi(x, \lambda)$ – множина базисних функцій активації. Тоді похибка часової різниці набуває вигляду:

$$e_t = r(x_t, \lambda(x_t)) + \gamma W^T \phi(x_{t+1}, \lambda_{t+1}) - W^T \phi(x_t, \lambda_t). \tag{32}$$

Для методів навчання з підкріпленням, включаючи PI або VI, етап оновлення стратегії буде базуватися на:

$$\frac{\partial}{\partial \lambda} Q_\lambda(x_t, \lambda) = \frac{\partial}{\partial \lambda} W^T \phi(x_t, \lambda) = 0. \tag{33}$$

Оскільки ця нейронна мережа явно залежить від керуючої дії λ, похідні можуть бути обчислені без знання динаміки системи. Щоб вирішити рівняння для λ і отримати явну стратегію $\lambda_t = \lambda(x_t)$, потрібно застосувати теорему про неявну функцію до цієї структури нейронної мережі.

Алгоритми PI і VI можна використовувати для Q-навчання.

5. Висновки

У цій статті представлено основні ідеї та алгоритми навчання з підкріпленням, зокрема сімейства методів, відомих як адаптивне динамічне програмування (ADP), а також продемонстрована корисність цих методів для визначення оптимальної стратегії керування біологічної системи процесів регенерації печінки людини.

Таким чином, в подальшому викладені методи будуть використані для розв'язання задачі знаходження верхньої оцінки оптимальності процесів регенерації печінки. Також отримані розв'язки планується верифікувати з даними, які отримані в біологічних експериментах.

Історія статті: отримана: 8 травня 2024; останній варіант: 22 травня 2024
прийнята: 8 червня 2024.

REFERENCES

1. E.T. Liu. Systems biology, integrative biology, predictive biology. Cell. – 2005. – Vol. 121(4). – P. 505–506. DOI: 10.1016/j.cell.2005.04.021

2. J.M. Smith. Optimization theory in evolution. Annu Rev Ecol Syst. – 1978. – Vol. 9(1). – P. 31–56. DOI: 10.1146/annurev.es.09.110178.000335

3. G.A. Parker, J.M. Smith et al. Optimality theory in evolutionary biology. Nature. – 1990. – Vol. 348(6296). – P. 27–33. DOI: 10.1038/348027a0

4. V. V. Karieva, S. V. Lvov. Mathematical model of liver regeneration processes: homogeneous approximation. Visnyk of V.N.Karazin Kharkiv National University. Ser. "Mathematics, Applied Mathematics and Mechanics". – 2018. – Vol. 87. – P. 29–41. DOI: 10.26565/2221-5646-2018-87-03

5. V. V. Karieva, S. V. Lvov, L. P. Artyukhova. Different strategies in the liver regeneration processes. Numerical experiments on the mathematical model. Visnyk of V.N.Karazin Kharkiv National University. Ser. "Mathematics, Applied Mathematics and Mechanics". – 2020. – Vol. 91. – P. 36–44. DOI: 10.26565/2221-5646-2020-91-03

6. J. M. Mendel, R. W. McLaren. Reinforcement-Learning Control and Pattern Recognition Systems. Mathematics in Science and Engineering. – 1970. – Vol. 66. – P. 287–318. DOI: 10.1016/S0076-5392(08)60497-X

7. V. V. Karieva, S. V. Lvov. Liver regeneration after partial hepatectomy: the upper optimality estimate. Visnyk of V.N.Karazin Kharkiv National University. Ser. "Mathematics, Applied Mathematics and Mechanics". – 2023. – Vol. 97. – P. 41–58. DOI: 10.26565/2221-5646-2023-97-04

8. R. E. Bellman. Dynamic Programming. Princeton, NJ: Princeton Univ. – 1957. – 392 p. ISBN: 9780691146683

9. R. S. Sutton, A. G. Barto. Reinforcement Learning—An Introduction. Cambridge, MA: MIT Press. – 1998. – 526 p. ISBN: 978-0-262-19398-6

10. D. P. Bertsekas, J. N. Tsitsiklis. Neuro-Dynamic Programming. MA: Athena Scientific. – 1996. – 512 p. DOI: 10.1007/978-0-387-74759-0-440

11. W. T. Miller III, R. S. Sutton, P. J. Werbos. Neural Networks for Control. The MIT Press. – 1995. – 544 p. ISBN: 9780262631617

12. F. L. Lewis, D. L. Vrabie. Reinforcement learning and adaptive dynamic programming for feedback control. IEEE Circuits and Systems Magazine. – 2009. – Vol. 9(3). – P. 32–50. DOI: 10.1109/MCAS.2009.933854

13. Al-Tamimi, F. L. Lewis, M. Abu-Khalaf. Model-free Q-learning designs for linear discrete-time zero-sum games with application to H-infinity control. Automatica. – 2007. – Vol. 43. – P. 473–481. DOI: 10.1016/j.automatica.2006.09.019

14. C. J. C. H. Watkins, P. Dayan. Q-learning. Machine Learning. – 1992. – Vol. 8. – P. 279–292. DOI: 10.1007/BF00992698

Article history: Received: 8 May 2024; Final form: 22 May 2024
Accepted: 8 June 2024.

Adaptive dynamic programming for the optimal liver regeneration control

V. V. Karieva[1], S. V. Lvov[2]
[1] *Departement of Applied Mathematics*
[2]*Research Institute of Biology*
V. N. Karazin Kharkiv National University
sq. Svobody, 4, Kharkiv, Ukraine, 61022

Every living organism interacts with an environment and uses that interaction for an improvement of its own adaptability, and, as a result, one's survival and overall development. The process of evolution shows us that different species change methods of interaction with an environment with passage of time, which leads to natural selection and survival of the most adaptive ones. This learning, which based on actions, or reinforcement learning may embrace the idea of optimal behavior occurring in environmental systems. We describe mathematical formulas for reinforcement learning and the practical integration method also known as adaptive dynamic programming. That gives us the overall concept of controllers for artificial biological systems that both learn and show the optimal behavior.

This paper reviews the formulation of the upper optimality problem, for which the optimal regulation strategy is guaranteed to be better or equivalent to objective regulation rules that can be observed in natural biological systems.

In cases of optimal reinforcement learning algorithms the learning process itself moves from the analysis of the item take on system dynamics to the much higher level. The object of interest now is not the details of the system dynamics, but the quantity efficiency index, which clearly represents how optimally the control system works. Such scheme of reinforcement learning is learning technique of optimal behavior in order to monitor the response to non-optimal control strategies.

The purpose of this article is to show the possibility of using of reinforcement learning methods, the adaptive dynamic programming (ADP) in particular, to control biological systems using feedback. This article shows the on-line methods for solving the problem of searching the upper optimality estimate with adaptive dynamic programming.

Keywords: **Dynamic programming; Optimal control; Reinforcement learning.**

How to cite this article:

V. V. Karieva, S. V. Lvov, Adaptive dynamic programming for the optimal liver regeneration control, Visnyk of V. N. Karazin Kharkiv National University. Ser. Mathematics, Applied Mathematics and Mechanics, Vol. 99, 2024, p. 22–35 (in Ukrainian). DOI: 10.26565/2221-5646-2024-99-02