

Катков Юрій Ігорович

доктор технічних наук, доцент, професор кафедри комп'ютерних наук

Державний університет інформаційно-комунікаційних технологій, Київ, Україна

ORCID ID <https://orcid.org/0009-0007-1194-4014>

kyi12kyi12@gmail.com

Вишнівський Олександр Вікторович

аспірант

Державний університет інформаційно-комунікаційних технологій, Київ, Україна

ORCID ID <https://orcid.org/0009-0008-0209-9549>

o.vyshnivskiy@gmail.com

МЕТОД УПРАВЛІННЯ КОМП'ЮТЕРНОЮ МЕРЕЖЕЮ SD-WAN МЕТОДАМИ МАШИННОГО НАВЧАННЯ НА ОСНОВІ МАТЕМАТИЧНОЇ МОДЕЛІ У ПРОСТОРИ СТАНІВ

Анотація. Розвиток технологій побудови комп'ютерних мереж привело до створення програмно-визначених глобальних мереж SD-WAN. Цей напрямок являється одним із ключових тенденцій в розвитку ефективної мережевої інфраструктури. В цьому випадку забезпечується програмне централізоване управління всією комп'ютерною розподіленою мережею використовуючи спеціалізований контролер, який відокремлює систему управління від системи передачі даних.

Метою даної роботи являється підвищення ефективності функціонування розподілених комп'ютерних мереж SD-WAN методами машинного навчання на основі математичної моделі у просторі станів. В роботі вирішено актуальне наукове завдання з розробки методу управління комп'ютерною мережею SD-WAN на основі машинного навчання у просторі станів. Розроблено математичну модель мережі SD-WAN у просторі станів, яка представляє собою сукупність векторів стану, управління та функцію якості обслуговування. Вона враховує динаміку завантаження каналів, затримки, втрати пакетів та стан буферів вузлів. Дана модель може бути застосована у вигляді лінеаризованого варіанту для проведення аналітичних розрахунків, так і у формі повної нелінійної форми для проведення симуляції. Визначені умови стійкості та керованості комп'ютерної мережі. Для лінійної моделі отримано аналітичний розв'язок задачі оптимального управління SD-WAN, яке ґрунтується на основі рівняння Річчати. На основі глибокого навчання з підкріпленням розроблено алгоритм управління мережею. Для дослідження ефективності отриманих результатів проведено симуляцію розподіленої комп'ютерної мережі різними методами управління SD-WAN. Результати проведеної симуляції підтвердили перевагу запропонованого методу управління комп'ютерною мережею SD-WAN на основі машинного навчання у просторі станів. Вони показали зниження затримки на 65%, рівня втрат пакетів на 85% та значення функціоналу якості на 61% порівняно з базовим методом ECMP.

Ключові слова: розподілена комп'ютерна мережа, SD-WAN, методи машинного навчання, математична модель, метод простору станів, управління, якість обслуговування, канал зв'язку.

Katkov Yuriy

Doctor of Technical Sciences, Associate Professor, Professor of Computer Science Department

State University of Information and Communication Technologies / Department of Computer Science, Kiev, Ukraine.

ORCID ID <https://orcid.org/0009-0007-1194-4014>

kyi12kyi12@gmail.com

Vyshnivskiy Oleksandr

postgraduate

State University of Information and Communication Technologies / Department of Computer Science, Kiev, Ukraine.

ORCID ID: <https://orcid.org/0009-0008-0209-9549>

o.vyshnivskiy@gmail.com

METHOD OF CONTROLLING THE SD-WAN COMPUTER NETWORK USING MACHINE LEARNING METHODS BASED ON A MATHEMATICAL MODEL IN THE STATE SPACE

© 2026 Катков Ю.І., Вишнівський О.В. Цей матеріал ліцензовано за умовами **CC BY 4.0**.
<https://creativecommons.org/licenses/by/4.0/>

Abstract. *The development of computer network construction technologies has led to the creation of software-defined wide area networks (SD-WAN). This direction is one of the key trends in the development of effective network infrastructure. In this case, software centralized management of the entire distributed computer network is provided using a specialized controller that separates the control system from the data transmission system.*

The purpose of this work is to improve the efficiency of functioning of distributed computer networks SD-WAN using machine learning methods based on a mathematical model in the state space. The work solves the current scientific problem of developing a method for controlling a computer network SD-WAN based on machine learning in the state space. A mathematical model of the SD-WAN network in the state space has been developed, which represents a set of state vectors, control and a quality of service function. It takes into account the dynamics of channel loading, delay, packet loss and the state of node buffers. This model can be used in the form of a linearized version for analytical calculations, and in the form of a full nonlinear form for simulation. The conditions for stability and controllability of a computer network are determined. For the linear model, an analytical solution to the optimal SD-WAN control problem is obtained, which is based on the Riccati equation. A network control algorithm is developed based on deep reinforcement learning. To study the effectiveness of the obtained results, a simulation of a distributed computer network using different SD-WAN management methods was conducted. The simulation results confirmed the superiority of the proposed state-space machine learning-based SD-WAN computer network management method. They showed a 65% reduction in latency, an 85% reduction in packet loss, and a 61% improvement in the quality functional compared to the basic ECMP method.

Keywords: *distributed computer network, SD-WAN, machine learning methods, mathematical model, state space method, management, quality of service, communication channel.*

1. Вступ

Стрімкий розвиток технологій побудови комп'ютерних мереж привело до створення програмно-визначених глобальних мереж (Software-Defined Wide Area Network, SD-WAN). Цей напрямок являється одним із ключових тенденцій в розвитку ефективної мережевої інфраструктури. В цьому випадку забезпечується програмне централізоване управління всією комп'ютерною розподіленою мережею використовуючи спеціалізований контролер, який відокремлює систему управління від системи передачі даних [1], [2]. Тому з'являється можливість перерозподіляти трафік використовуючи різні типи каналів зв'язку на основі динамічно управляємої маршрутизації. Для цього можуть використовуватись LTE/5G, широкосмугові Internet-канали, виділені лінії MPLS. Їх застосування буде залежати від поточного завантаження всієї мережі, а також встановлених правил для забезпечення заданого рівня якості обслуговування (QoS).

2. Постановка проблеми

При використанні класичних детермінованих методів задання управління SD-WAN представляється складним, а в деяких випадках практично не доцільним. Це обумовлено підвищенням складності комп'ютерної розподіленої мережевої топології, динамічними змінами станів каналів зв'язку, постійною неоднорідністю трафіку та іншими факторами. Для забезпечення ефективності управління необхідно врахувати велику кількість параметрів комп'ютерної мережі у реальному масштабі часу. Такими параметрами можуть бути пропускна здатність каналів, затримки пакетів, рівня втрати пакетів, джиттера та інших.

Тому, в цьому випадку, комп'ютерну мережу доцільно представити у вигляді управляємої динамічної моделі, де трафік мережі та стани каналів зв'язку будуть представляти собою впливи, а сигнал контролера SD-WAN буде управлінням по розподілу трафіку. Для формалізації завдання забезпечення ефективного управління на основі SD-WAN пропонується метод простору станів. Це дозволяє використати сучасний апарат теорії автоматичного управління для розробки математичної моделі мережі і на її основі оптимізувати управління методами машинного навчання Machine Learning (ML), зокрема застосовуючи навчання з підкріпленням Reinforcement Learning (RL) для управління розподіленими комп'ютерними мережами [3], [4].

Таким чином, необхідно розробити метод управління комп'ютерною мережею SD-WAN на основі машинного навчання у просторі станів.

3. Аналіз останніх досліджень і публікацій

Дослідженням використання теорії автоматичного управління та оптимізації мережевого управління на основі машинного навчання написано велику кількість наукових праць. При цьому архітектурну основу комп'ютерної розподіленої мережі з управлінням SD-WAN можуть пов'язувати з принципами SDN. Ці принципи полягають в логічній централізації управління та використанням програмного забезпечення для представлення контрольних площин в абстрактному вигляді [5]. Проте управління SD-WAN ґрунтується на використанні централізованих політик та управлінні динамічним перерозподілом трафіку на основі вимірювання якості додатків [6].

В роботах [7], [8] та [9] показано, що SD-WAN будується на таких принципах, як використання WAN для глобальної оптимізації при централізованому керуванні потоками даних та управлінні на основі інтеграції вимірювань показників якості мережі.

Використовуючи теорію мережевого трафіку на основі апаратів теорії черг та стохастичних процесів, можна визначити режимні зміни інтенсивності потоків в каналах зв'язку [10]. Такі моделі можна застосувати для відносно постійних статистичних інтенсивностей трафіку і описати детермінованими рівняннями управління.

З точки зору управління трафіком, теорія робастного синтезу через LMI (BRL/KYP) дає можливість знаходження гарантованих меж для найгірших вхідних впливів [11]. Оптимальне управління (LQR, LQG) надає базову форму «м'якої» оптимізації, а MPC – практичну технологію оптимізації з обмеженнями [12].

У [13] пропонується підхід для оптимізації маршрутизації в SDN, який ґрунтується на використанні лінійно-квадратичного регулятора (LQR). Однак, ці роботи не враховують особливості SD-WAN, які полягають в наявності різних технологій передачі даних та необхідності динамічного перерозподілу трафіку в реальному масштабі часу.

Що стосується технологій машинного навчання для управління розподіленими комп'ютерними мережами, то вони використовуються для управління QoS на основі Deep Q-Network (DQN) та його варіантів [14, 15], або алгоритму Proximal Policy Optimization (PPO) [16] для завдань з неперервним простором дій. Проте в цих роботах пропонуються рішення для окремих аспектів управління. Вони не надають універсальної математичної моделі мережі у вигляді простору станів для SD-WAN.

Тому необхідно розробити метод побудови комплексної математичної моделі SD-WAN у просторі станів на основі теорії автоматичного управління та методи машинного навчання. В роботі пропонується інтегральне рішення. SD-WAN представляється як система з синтезом управління із чітко визначеними станом, виходом та впливами.

4. Мета і задачі дослідження

Метою даної роботи являється підвищення ефективності функціонування розподілених комп'ютерних мереж SD-WAN методами машинного навчання на основі математичної моделі у просторі станів. Для досягнення поставленої мети необхідно вирішити актуальне наукове завдання з розробки методу управління комп'ютерною мережею SD-WAN на основі машинного навчання у просторі станів. Для цього побудуємо математичну модель мережі SD-WAN у просторі станів, яка буде представляти собою сукупність векторів стану, управління та функцію якості обслуговування. На їх основі необхідно забезпечити стійкість моделі мережі SD-WAN та визначити умови оптимальності. Для цього на основі глибокого навчання з підкріпленням розробити алгоритм управління мережею. Для дослідження ефективності отриманих результатів провести моделювання розподіленої комп'ютерної мережі з різними методами управління SD-WAN.

5. Результати дослідження

5.1 Математична модель розподіленої комп'ютерної мережі SD-WAN у просторі станів

З точки зору концепції SD-WAN вся мережа представляє собою одну велику динамічну систему. Контролер SD-WAN повинен підтримувати систему в певному стані, видаючи управляючі команди на основі виміру телеметрії. Розглянемо архітектуру розподіленої комп'ютерної мережі SD-WAN та її основні компоненти.

Представимо комп'ютерну мережу SD-WAN у вигляді системи, яка складається з N вузлів, що пов'язані між собою через M гетерогенних каналів передачі даних різних типів (MPLS, Broadband, LTE/5G). В цьому випадку управління буде здійснюватися центральним SD-WAN контролером. Даний контролер отримує дані про телеметрію мережі та надає управляючі впливи на перерозподіл трафіку.

Нехай граф комп'ютерної мережі буде представляти собою вираз

$$G = (V, E), \quad (1)$$

де $V = \{v_1, v_2, \dots, v_n\}$ – множина вузлів комп'ютерної мережі, $E = \{e_1, e_2, \dots, e_m\}$ – множина ребер комп'ютерної мережі (каналів передачі даних різних типів). Для кожного каналу $e_k \in E$ певний набір його стандартних та поточних характеристик. Це максимальна пропускна здатність C_k , поточне завантаження $l_k(t)$, затримка $d_k(t)$, рівень втрати пакетів $p_k(t)$.

Визначимо вектор стану комп'ютерної мережі. Вектор стану системи $x(t) \in R^n$ в просторі n у дискретний момент часу t формується з таких компонентів

$$x(t) = [xL(t)^T, xD(t)^T, xP(t)^T, xB(t)^T]^T \in R^n, \quad (2)$$

де $xL(t) = [l_1(t), l_2(t), \dots, l_m(t)]^T$ – вектор завантаження каналів комп’ютерної мережі, $l_k(t) = L_k(t)/C_k \in [0, 1]$; $xD(t) = [d_1(t), d_2(t), \dots, d_m(t)]^T$ – вектор затримок каналів комп’ютерної мережі (мс); $xP(t) = [p_1(t), p_2(t), \dots, p_m(t)]^T$ – вектор ймовірностей втрати пакетів; $xB(t) = [b_1(t), b_2(t), \dots, b_n(t)]^T$ – вектор стану буферів вузлів комп’ютерної мережі (ступінь заповненості).

Повна розмірність вектору стану становить $n = 3M + N$.

Для управління комп’ютерною мережею визначимо вектор управляючих впливів $u(t) \in R^m$ в просторі m , який буде визначати розподіл трафіку між каналами

$$u(t) = [u_1(t), u_2(t), \dots, u_m(t)]^T, \quad (3)$$

де $u_k(t) \in [0, 1]$ – частина трафіку класу сервісу s , що проходить через канал e_k у момент t . При цьому необхідно, щоб виконувалися обмеження

$$\sum_k u_k^s(t) = 1, \quad \forall s \in S, \quad \forall t, \quad (4)$$

де S – множина класів сервісу. При необхідності вектор управління може включати і інші параметри комп’ютерної мережі, наприклад, такі як формування трафіку та пріоритетизації черг.

Розглянувши архітектуру комп’ютерної мережі, основні компоненти SD-WAN, вектори стану $x(t)$ та управляючих впливів $u(t)$ побудуємо узагальнену модель у просторі станів.

Динаміка мережі SD-WAN описується нелінійним рівнянням стану з дискредитацією часу наступного вигляду

$$x(t+1) = f(x(t), u(t), w(t)), \quad (5)$$

де $w(t)$ – вектор впливів, а саме фактори, на які контролер не може вплинути безпосередньо, але які впливають на стан системи (наприклад, обрив каналу, раптовий сплеск трафіку, DDoS-атака). біля робочої точки (x^*, u^*) отримаємо

$$\Delta x(t+1) = A \cdot \Delta x(t) + B \cdot \Delta u(t) + \Gamma \cdot w(t), \quad (6)$$

$$y(t) = C \cdot \Delta x(t) + D \cdot \Delta u(t) + v(t), \quad (7)$$

де $\Delta x(t) = x(t) - x^*$, $\Delta u(t) = u(t) - u^*$ – відхилення від точки лінеаризації; $A \in R^{n \times n}$ – матриця, що описує внутрішню динаміку мережі комп’ютерної мережі; $B \in R^{n \times m}$ – матриця, що описує вплив керувань на стан (матриця управління); $C \in R^{l \times n}$ – матриця, що описує, які стани ми можемо безпосередньо спостерігати; $D \in R^{l \times m}$ – матриця прямого зв’язку; $\Gamma \in R^{n \times p}$ – матриця впливів; $v(t)$ – шум вимірювань.

Нелінійна модель мережі SD-WAN (5) може бути представлена через компонентні рівняння, які залежать від архітектури комп’ютерної мережі.

Розглянемо завантаження каналу e_k . Динаміка завантаження каналу описується рівнянням черги M/G/1 наступного вигляду

$$l_k(t+1) = l_k(t) + \Delta t / C_k \cdot [\sum_i \lambda_{ik}(t) \cdot u_{ik}(t) - \mu_k(t) \cdot l_k(t)] + w_k^1(t), \quad (8)$$

де $\lambda_{ik}(t)$ – інтенсивність вхідного трафіку з вузла i через канал k ; $\mu_k(t)$ – швидкість обслуговування; Δt – крок дискретизації часу.

Знайдемо затримку пакетів каналу. До моделі затримки в каналі будуть входити наступні складові: поширення, передачі та очікування в черзі. Використовуючи модель M/M/1 отримаємо

$$d_k(t) = d_k^{\text{prop}} + L_{\text{pkt}} / C_k + l_k(t) / (\mu_k \cdot (1 - \rho_k(t))), \quad (9)$$

де d_k^{prop} – затримка поширення, яка є постійною величиною; L_{pkt} – усереднена довжина пакету; $\rho_k(t) = l_k(t)$ – коефіцієнт завантаження каналу.

Знайдемо імовірність втрати пакетів використовуючи формула Ерланга-В при умові для кінцевого буфера з розміром B_k . В цьому випадку модель втрати пакетів буде

$$p_k(t) = (\rho_k(t)^{B_k}/B_k!)/\sum_{j=0}^{B_k} B_k(\rho_k(t)^j/j!). \quad (10)$$

Визначимо елементи матриці A . Вони визначаються частинними похідними для всіх компонент функції f вектору стану x у точці лінеаризації, а саме

$$A_{ij} = \partial f_i(x, u)/\partial x_j |_{(x^*, u^*)}. \quad (11)$$

У матриці A блочна структура наступного вигляду

$$A = [ALL \ ALD \ ALP \ ALB; \ ADL \ ADD \ 0 \ 0; \ APL \ 0 \ APP \ 0; \ ABL \ 0 \ 0 \ ABB]. \quad (12)$$

В цьому вигляді блоки представляють собою взаємовпливи компонентів вектору стану комп'ютерної мережі. Для прикладу, $ALD = \partial l_k/\partial d_j$ буде визначати впливи затримки на завантаження, а $ALL = I - \Delta t \cdot \text{diag}\{\mu_k\}$ буде описувати самозатухання завантаження каналів.

Елементами матриці управління B , які її визначають наступні

$$B = [\Delta t/C_k \cdot \text{diag}\{\lambda_{ik}\}; \ \partial d/\partial u; \ \partial p/\partial u; \ 0]. \quad (13)$$

Таким чином, отримано узагальнену математичну модель SD-WAN у просторі станів (1) - (13). Ця модель залежить від динаміки завантаження каналів, втрати пакетів, затримки та стану буферів вузлів. На основі цієї моделі можна розглядати лінеаризований її варіант для аналітичних розрахунків. Також її повна нелінійна форма може бути використана для проведення симуляції комп'ютерної мережі з управлінням SD-WAN.

Для лінійної моделі система (6) - (7) буде асимптотично стійкою тільки тоді, коли всі власні числа матриці замкненої системи Азам знаходяться всередині одиничного кола на комплексній площині

$$|\lambda_i(\text{Азам})| < 1, \ \forall i = 1, \dots, n, \quad (14)$$

де у випадку зворотнього зв'язку $u(t) = -K \cdot x(t)$ отримаємо Азам = $A - B \cdot K$. Матриця K може визначатися методами оптимального управління або методами машинного навчання.

Перевіримо умову керованості системи (5). Для цього визначимо матрицю керованості

$$\text{rank}(C) = \text{rank}[B \ | \ AB \ | \ A^2B \ | \ \dots \ | \ A^{n-1}B] = n. \quad (15)$$

Для перевірки можливості спостережуваності системи представимо матрицю спостережуваності

$$\text{rank}(O) = \text{rank}[C^T \ | \ A^T C^T \ | \ (A^T)^2 C^T \ | \ \dots \ | \ (A^T)^{n-1} C^T] = n. \quad (16)$$

Вираз для функції якості управління SD-WAN буде визначається у вигляді зваженої суми відхилень основних показників від їх цільових значень

$$J = \sum_{t=0}^T [x^T(t)Qx(t) + u^T(t)Ru(t)] + x^T(T)Px(T), \quad (17)$$

де $Q \in R^{n \times n}$ – вагова матриця стану; $R \in R^{m \times m}$ – матриця вартості управління; P – матриця штрафу. Вагова матриця Q може задаватися у вигляді блоку

$$Q = \text{diag}\{qL \cdot IM, \ qD \cdot IM, \ qP \cdot IM, \ qB \cdot IN\}, \quad (18)$$

де qL, qD, qP, qB – представляють собою зважені коефіцієнти відповідно для завантаження, затримки, втрати пакетів та заповненості буферів. Вибір ваги кожного коефіцієнта проводиться відповідно до вимог стандарту обслуговування для різних класів трафіку.

У випадку лінійної моделі (6) завдання мінімізації (17) буде зведено до вирішення рівняння Річчати

$$P = Q + A^T P A - A^T P B (R + B^T P B)^{-1} B^T P A, \quad (19)$$

а оптимальна матриця зворотнього зв'язку буде визначатися наступним чином

$$K^* = (R + B^T P B)^{-1} B^T P A. \quad (20)$$

Для узагальненої математичної моделі SD-WAN у просторі станів встановлено умови стійкості (14) та керованості (15) комп'ютерної мережі. Для випадку використання лінійної моделі було отримано аналітичний розв'язання завдання оптимального управління використовуючи рівняння Річчати (19) - (20).

5.2 Алгоритм управління SD-WAN на основі глибокого навчання з підкріпленням

У випадку побудови алгоритму управління SD-WAN на основі глибокого навчання з підкріпленням завдання оптимального управління SD-WAN можна формалізувати у вигляді марківського процесу прийняття рішень (MDP)

$$M = \langle S, A, P, R, \gamma \rangle, \quad (21)$$

де S – представляє собою простір станів, а $s(t) \equiv x(t) \in R^n$ – вектор стану мережі (2); A – простір дій, $a(t) \equiv u(t) \in R^m$ – управляючі впливи (3); $P(s'|s, a)$ – функція переходу стану, яка задана нелінійною моделлю (5); $R(s, a)$ – функція винагороди; $\gamma \in (0, 1)$ – коефіцієнт дисконтування.

Знайдемо функціонал винагороди. Він визначається у вигляді від'ємного значення функціоналу якості (17) на одному кроці

$$r(t) = -[\alpha L \|xL(t) - xL^*\|^2 + \alpha D \|xD(t) - xD^*\|^2 + \alpha P \|xP(t)\|^2 + \alpha u \|u(t)\|^2], \quad (22)$$

де xL^*, xD^* – представляють собою цільові значення завантаження та затримки; $\alpha L, \alpha D, \alpha P, \alpha u$ – вагові коефіцієнти.

Мета агента в навчанні з підкріпленням, це максимізувати сумарну дисконтовану винагороду. Агент намагається знайти таку стратегію поведінки, яка забезпечить найбільший "прибуток" у довгостроковій перспективі. Тому необхідно максимізувати сумарну дисконтовану винагороду

$$G(t) = \sum_{k=0}^{\infty} \gamma^k r(t+k). \quad (23)$$

Для дискретного управління застосуємо алгоритм DQN. При цьому Q -функція буде апроксимуватися нейронною мережею θ

$$Q(s, a; \theta) \approx Q^*(s, a). \quad (24)$$

Параметри розглядаємої мережі будуть оновлюватися на основі мінімізації функції втрат Беллмана

$$L(\theta) = E[(y - Q(s, a; \theta))^2], \quad (25)$$

$$y = r + \gamma \cdot \max_{a'} Q(s', a'; \theta^-), \quad (26)$$

де θ^- – параметри комп'ютерної мережі, що оновлюється кожні τ кроків. Градієнт функції втрат буде мати вигляд

$$\nabla_{\theta} L(\theta) = -2(y - Q(s, a; \theta)) \cdot \nabla_{\theta} Q(s, a; \theta) \quad (27)$$

Навчання проводиться при використанні буфера досвіду $D = \{(s_i, a_i, r_i, s_i')\}$ розміром $|D|$. Мінібатч величиною B випадково вибирається з D для визначення градієнта.

Для того, щоб сформувати оптимальні неперервні управляючі впливи вектору $u(t)$ необхідно використати алгоритм PPO. Поліси апроксимуються параметричною стохастичною функцією $\pi\theta(a|s)$

$$\pi\theta(a|s) = N(\mu\theta(s), \Sigma\theta(s)). \quad (28)$$

де $\Sigma\theta(s)$ та $\mu\theta(s)$ – коваріаційна матриця та середнє нормального розподілу, що формуються актор-неймережею. Вигляд цільової функції PPO з обмеженням на зміну полісу буде мати вигляд

$$L^{\wedge}CLIP(\theta) = E_t[\min(r_t(\theta) \cdot A_t, \text{clip}(r_t(\theta), 1 - \varepsilon, 1 + \varepsilon) \cdot A_t)] \quad (29)$$

де $r_t(\theta) = \pi\theta(a_t|s_t)/\pi\theta_{\text{old}}(a_t|s_t)$ – відношення ймовірностей; ε – гіперпараметр обмеження ($\varepsilon = 0.1 \div 0.2$); A_t – оцінка функції переваги (advantage function).

Оцінимо функцію переваги методом GAE у навчанні з підкріпленням, що призначений для обчислення функції переваги з метою оновлення політики

$$A_t^{\wedge}GAE(\lambda) = \sum_{k=0}^{\infty} (\gamma\lambda)^k \cdot \delta_{t+k}, \quad \delta_t = r_t + \gamma \cdot V(s_{t+1}) - V(s_t), \quad (30)$$

де $V(s)$ – функція цінності стану, яка буде апроксимуватися критич-неймережею φ . Critic оновлюється мінімізацією помилки прогнозу

$$L^{\wedge}VF(\varphi) = E_t[(V\varphi(s_t) - V_t^{\text{target}})^2]. \quad (31)$$

Таким чином, розроблено метод управління комп'ютерною мережею SD-WAN на основі машинного навчання у просторі станів.

6. Архітектура симуляції мережі SD-WAN

Розглянемо архітектуру нейронної мережі. Для апроксимації Q -функції та полісу використаємо Feedforward Neural Network, нейронну мережу прямого поширення з наступною архітектурою.

1. Вхідний шар: $\dim(s) = n$ нейронів.
2. Приховані шари: два шари з відповідно 256 та 128 нейронів та функція активації ReLU у вигляді

$$\varphi_1(z) = \max(0, z), \quad h_1 = \varphi_1(W_1 \cdot h_{1-1} + b_1). \quad (30)$$

3. Вихідний шар DQN: K нейронів (по числу дискретних дій), лінійна активація.
4. Актор PPO: m нейронів, активація \tanh для обмеження виходу в $[-1, 1]$, масштабування до допустимої області управлiнь.
5. Critic PPO: 1 нейрон, лінійна активація.
6. Загальна кількість параметрів мережі

$$N\theta = n \cdot 256 + 256 + 256 \cdot 128 + 128 + 128 \cdot m + m \quad (33)$$

Розробимо алгоритм навчання агента SD-WAN. Загальна схема для навчання агента буде описуватися наступним чином (реалізовано на мові C++)

```

Вхід: гіперпараметри { $\gamma, \lambda, \varepsilon, \alpha_{\text{actor}}, \alpha_{\text{critic}}, T, K_{\text{epochs}}$ }
Вихід: оптимізована стохастична поліса  $\pi\theta^*(a|s)$ 
Policy PPO_Train(double gamma, double lambda, double eps,
    double alpha_actor, double alpha_critic,
    int T, int K_epochs, int MAX_EPISODES)
{
    Actor theta = XavierInitActor();
    Critic phi = XavierInitCritic();
    State s = EnvReset();
    Policy pi_best;
    double J_best = -INF;

    for (int episode = 1; episode <= MAX_EPISODES; ++episode){
        Trajectory tau;
        for (int t = 1; t <= T; ++t){
            Action a = Sample(theta, s); //  $a_t \sim \pi\theta(\cdot|s_t)$ 
            StepResult step = EnvStep(a); // apply to SD-WAN
        }
    }
}

```

```
tau.push_back({s, a, step.r, step.s_next});
s = step.s_next;}
```

```
Vector V_target(T), A(T);
for (int t = 0; t < T; ++t)
    V_target[t] = tau[t].r + gamma * Value(phi, tau[t].s_next);
```

```
A = ComputeGAE(tau, phi, gamma, lambda); // formula (28)
Normalize(A, eps);
```

```
for (int k = 0; k < K_epochs; ++k){
    MiniBatch B = SampleMiniBatch(tau);
    double L_clip = ComputeClipLoss(theta, B, A); // formula (27)
    double L_vf = ComputeValueLoss(phi, B, V_target); // formula (29)
    double L_total = -L_clip + c1 * L_vf - c2 * Entropy(theta, B);
```

```
theta = theta - alpha_actor * GradTheta(L_total, theta);
phi = phi - alpha_critic * GradPhi(L_vf, phi);}
```

```
double J = EvaluateReturn(theta);
if (J > J_best) { J_best = J; pi_best = theta.ToPolicy(); }
```

```
}
```

```
return pi_best;
```

```
}
```

7. Аналіз результатів

Для навчання агента була проведена симуляція, яка реалізує рівняння стану (5)–(10). Були проведені дослідження на симуляційному стенді розробленої математичної моделі мережі SD-WAN у просторі станів та алгоритму управління мережею. Комп’ютерна мережа містила $N = 10$ вузлів, $M = 15$ каналів зв’язку трьох типів (MPLS: 5 каналів по 100 Мбіт/с, Broadband: 6 каналів по 50 Мбіт/с, LTE: 4 канали по 20 Мбіт/с), $S = 3$ класи сервісу. Навчання проводилось протягом 2000 епізодів по $T = 200$ кроків кожен.

Збіжність алгоритму навчання з підкріпленням PPO підтверджується постійним зростанням середньої накопиченої винагороди $G(t)$ зі стандартним відхиленням, що зменшується на протязі навчання. Агент досягає стаціонарного режиму приблизно після 800–1000 епізодів навчання.

Порівняння якості управління SD-WAN на основі трьох методів управління наведено в таблиці 1.

Таблиця 1

Порівняння методів управління SD-WAN

Назва методу	Завант. (avg)	Затримка (мс)	Втрати (%)	J (сумарний)
ECMP (базовий)	0.68	47.3	2.14	-312.4
LQR (лінійна мод.)	0.52	28.7	0.89	-198.6
DQN (дискр. д-ї)	0.47	22.1	0.61	-167.3
PPO (запропонов.)	0.38	16.4	0.32	-121.7

Результати проведеної симуляція показали, що розроблений підхід PPO на основі математичної моделі мережі SD-WAN у просторі станів забезпечує кращі показники: середнє завантаження каналів знижується на 44% порівняно з базовим методом ECMP, середня затримка – на 65%, рівень втрати пакетів – на 85%, а значення функціоналу якості (17) покращується на 61%.

Основною перевагою PPO перед DQN є її здатність до неперервної оптимізації вектору управліннь $u(t)$. Це дозволяє точніше адаптуватися до поточного стану комп’ютерної мережі SD-WAN. Перевага перед LQR полягає у врахуванні нелінійностей реальної комп’ютерної мережі, які не відображає лінеаризована модель (5).

6. Висновки та перспективи подальших досліджень

1. У статті вирішено актуальне наукове завдання з розробки методу управління комп’ютерною мережею SD-WAN на основі машинного навчання у просторі станів.

2. Побудовано математичну модель мережі SD-WAN у просторі станів (1)–(13), яка представляє собою сукупність векторів стану, управління та функцію якості обслуговування. Вона враховує динаміку завантаження каналів, затримки, втрати пакетів та стан буферів вузлів. Дана модель може бути застосована у вигляді лінеаризованого варіанту для проведення аналітичних розрахунків, так і у формі повної нелінійної форми для проведення симуляції.

3. Визначені умови стійкості (14) та керованості (15) комп'ютерної мережі. Для лінійної моделі отримано аналітичний розв'язок задачі оптимального управління SD-WAN, яке ґрунтується на основі рівняння Річчати (19)–(20).

4. Завдання управління SD-WAN представлено у вигляді MDP. Розроблено алгоритм навчання агента на основі навчання з підкріпленням PPO і функцією переваги GAE (29)–(30). Це забезпечує проведення стабільного навчання у просторі неперервних дій.

5. Проведені дослідження симуляції мережі SD-WAN. Результати проведеної симуляції підтвердили перевагу запропонованого методу управління комп'ютерною мережею SD-WAN на основі машинного навчання у просторі станів. Вони показали зниження затримки на 65%, рівня втрат пакетів на 85% та значення функціоналу якості на 61% порівняно з базовим методом ECOMP.

Перспективами подальших досліджень є: розширення моделі на сценарії відмов каналів та кібератак; застосування математичної оптимізації та машинного навчання для вирішення задачі вибору контрзаходів у комп'ютерній мережі при кібератаках.

Внесок авторів Юрій Катков – концептуалізація; аналіз джерел, підготовка огляду літератури; Олександр Вишнівський – метод; програмне забезпечення; виконання досліджень; узагальнення результатів.

Декларація про штучний інтелект

При написанні статті штучний інтелект використовувався для пошуку літератури, граматичної та стилістичної підтримки і виправлення стилю посилань на літературу (ChatGPT).

Конфлікт інтересів

Автори заявляють про відсутність конфлікту інтересів та підтверджують, що під час підготовки цієї роботи не існувало жодних комерційних, фінансових чи інших взаємовідносин, які могли б бути розцінені як такі, що здатні вплинути на результати дослідження або їх інтерпретацію. Робота виконана відповідно до принципів академічної доброчесності, етичних норм проведення наукових досліджень та вимог редакційної політики щодо запобігання конфлікту інтересів.

Список використаної літератури

1. Гніденко М.П., Вишнівський В.В., Зінченко О.В., Іщераков С.М. Новий вимір реалізації ключових функцій та можливостей новітніх технологій HPE Aruba / Монографія / – К.:ДУІКТ, ФОП Гуляєва В.М., 2025. – 289 с.
2. Meyer D. Software-Defined Networking: A Comprehensive Survey / D. Meyer, J. Clarke, I. Moraes. – Proceedings of the IEEE. – 2015. – Vol. 103, No. 1. – P. 14–76.
3. Xiao Y. Reinforcement Learning-Based QoS/QoE-Aware Service Function Chaining in Software-Defined and Virtualized Multimedia Networks / Y. Xiao, Q. Zhang // IEEE Transactions on Multimedia. – 2017. – Vol. 19, No. 7. – P. 1555–1564.
4. Liu S. Deep Reinforcement Learning for Dynamic Multichannel Access in Wireless Networks / S. Liu, W. Wang // IEEE Transactions on Cognitive Communications and Networking. – 2018. – Vol. 4, No. 2. – P. 257–265.
5. K. Pentikousis et al., “Software-Defined Networking (SDN): Layers and Architecture Terminology,” RFC 7426, Jan. 2015.
6. X. Wu, K. Lu, and G. Zhu, “A Survey on Software-Defined Wide Area Networks,” Journal of Communications, vol. 13, no. 5, pp. 253-258, 2018, doi: 10.12720/jem.13.5.253-258.
7. S. Jain et al., “B4: Experience with a Globally-Deployed Software Defined WAN,” ACM SIGCOMM Computer Communication Review, vol. 43, no. 4, pp. 3-14, 2013.
8. C. Hong et al., “Achieving High Utilization with Software-Driven WAN,” ACM SIGCOMM Computer Communication Review, vol. 43, no. 4, pp. 15-26, 2013.
9. A. Gupta et al., “SDX: A Software Defined Internet Exchange,” ACM SIGCOMM Computer Communication Review, vol. 44, no. 4, pp. 551-562, 2015. DOI:10.1145/2740070.2626300

10. M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York, NY, USA: Wiley, 1994.
11. A. Rantzer, "On the Kalman-Yakubovich-Popov lemma," *Systems & Control Letters*, vol. 28, no. 1, pp. 7-10, 1996.
12. J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd ed. Madison, WI, USA: Nob Hill Publishing, 2017.
13. Mestres A. *Knowledge-Defined Networking* / A. Mestres, A. Rodriguez-Natal, J. Carner et al. // *ACM SIGCOMM Computer Communication Review*. – 2017. – Vol. 47, No. 3. – P. 2–10.
14. Mnih V. *Human-level control through deep reinforcement learning* / V. Mnih, K. Kavukcuoglu, D. Silver et al. // *Nature*. – 2015. – Vol. 518. – P. 529–533.
15. Stampa G. *A Deep-Reinforcement Learning Approach for Software-Defined Networking Routing Optimization* / G. Stampa, M. Arias, D. Sanchez-Charles et al. – arXiv:1709.07080. – 2017.
16. Schulman J. *Proximal Policy Optimization Algorithms* / J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov. – arXiv:1707.06347. – 2017.

References

17. Gnidenko M.P., Vyshnivskiy V.V., Zinchenko O.V., Ishcheryakov S.M. *A new dimension of implementing key functions and capabilities of the latest HPE Aruba technologies* / Monograph / – Kyiv: DUKIT, FOP Gulyaeva V.M., 2025. – 289 p.
18. Meyer D. *Software-Defined Networking: A Comprehensive Survey* / D. Meyer, J. Clarke, I. Moraes. – *Proceedings of the IEEE*. – 2015. – Vol. 103, No. 1. – P. 14–76.
19. Xiao Y. *Reinforcement Learning-Based QoS/QoE-Aware Service Function Chaining in Software-Defined and Virtualized Multimedia Networks* / Y. Xiao, Q. Zhang // *IEEE Transactions on Multimedia*. – 2017. – Vol. 19, No. 7. – P. 1555–1564.
20. Liu S. *Deep Reinforcement Learning for Dynamic Multichannel Access in Wireless Networks* / S. Liu, W. Wang // *IEEE Transactions on Cognitive Communications and Networking*. – 2018. – Vol. 4, No. 2. – P. 257–265.
21. K. Pentikousis et al., "Software-Defined Networking (SDN): Layers and Architecture Terminology," RFC 7426, Jan. 2015.
22. X. Wu, K. Lu, and G. Zhu, "A Survey on Software-Defined Wide Area Networks," *Journal of Communications*, vol. 13, no. 5, pp. 253-258, 2018, doi: 10.12720/jcm.13.5.253-258.
23. S. Jain et al., "B4: Experience with a Globally-Deployed Software Defined WAN," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4, pp. 3-14, 2013.
24. C. Hong et al., "Achieving High Utilization with Software-Driven WAN," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4, pp. 15-26, 2013.
25. A. Gupta et al., "SDX: A Software Defined Internet Exchange," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 4, pp. 551-562, 2015. DOI:10.1145/2740070.2626300
26. M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York, NY, USA: Wiley, 1994.
27. A. Rantzer, "On the Kalman-Yakubovich-Popov lemma," *Systems & Control Letters*, vol. 28, no. 1, pp. 7-10, 1996.
28. J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd ed. Madison, WI, USA: Nob Hill Publishing, 2017.
29. Mestres A. *Knowledge-Defined Networking* / A. Mestres, A. Rodriguez-Natal, J. Carner et al. // *ACM SIGCOMM Computer Communication Review*. – 2017. – Vol. 47, No. 3. – P. 2–10.
30. Mnih V. *Human-level control through deep reinforcement learning* / V. Mnih, K. Kavukcuoglu, D. Silver et al. // *Nature*. – 2015. – Vol. 518. – P. 529–533.
31. Stampa G. *A Deep-Reinforcement Learning Approach for Software-Defined Networking Routing Optimization* / G. Stampa, M. Arias, D. Sanchez-Charles et al. – arXiv:1709.07080. – 2017.
32. Schulman J. *Proximal Policy Optimization Algorithms* / J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov. – arXiv:1707.06347. – 2017.

Надійшла до редакції: 11.12.25

Прийнята до друку: 17.03.26

Опубліковано: 30.03.26