

УДК 004.72

DOI: 10.18372/2073-4751.78.18959

Кулаков Ю.О., д.т.н.,  
orcid.org/0000-0002-8981-5649,  
e-mail: ya.kulakov@gmail.com,

Коренко Д.В.,  
orcid.org/0000-0003-0463-189X,  
e-mail: korenko.dima98@gmail.com

## МЕТОД БАЛАНСУВАННЯ НАВАНТАЖЕННЯ В МЕРЕЖАХ SDN З ВИКОРИСТАННЯМ ШТУЧНОГО ІНТЕЛЕКТУ

Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»

### Вступ

Підходи штучного інтелекту (AI) та машинного навчання (ML) в даний час відіграють ключову роль в розв'язанні широкого спектру складних проблем, що виникають у сучасних мережах. Серед цих завдань можна відокремити маршрутизацію, класифікацію трафіку, кластеризацію потоків, виявлення вторгнень, балансування навантаження, виявлення збоїв, оптимізацію якості обслуговування (QoS), поліпшення якості користування (QoE), контроль доступу та розподіл ресурсів.

У контексті *Software-Defined Networking (SDN)* взаємодія між AI та ML стає особливо важливою. Дослідження [1] вказує на те, що зростання ролі AI в SDN є значущим та відображає напрямок розвитку промисловості та наукового співтовариства. Цей рух відзначається впровадженням різноманітних підходів AI в архітектуру SDN.

Серед найбільш поширених методів вирішення проблем в мережах можна визначити машинне навчання (ML), мета-евристику та вибіркового пошук (FS). Ці підходи відзначаються високою ефективністю та гнучкістю у вирішенні різноманітних завдань, таких як адаптація до змін у трафіку, оптимізація роботи маршрутизації, та підвищення загальної продуктивності та надійності мережевого середовища. Такий симбіоз ШІ та SDN відкриває нові можливості для ефективного керування та оптимізації мережевих систем у сучасних інформаційних технологіях.

### Огляд існуючих рішень

Нинішні дослідження в сфері багатошляхових мереж активно розглядають проблематику балансування навантаження. Дві важливі стратегії розподілу навантаження, які широко використовуються в традиційних багатошляхових мережах, – це *Equal-Cost MultiPath (ECMP)* [2] та *Valiant Load Balance (VLB)* [3]. Основна концепція ECMP полягає в рівномірному розподілі потоків даних між комутаторами наступного переходу. У свою чергу, VLB розподіляє трафік між доступними шляхами та вибирає комутатор наступного переходу випадковим чином. Обидві ці стратегії використовують статичні методи і не можуть адаптивно реагувати на зміни в навантаженні на шляху в режимі реального часу.

Однак, ECMP та VLB, не дивлячись на їхню ефективність у певних умовах, суттєво обмежені фіксованістю методів. Ці стратегії не володіють потрібною гнучкістю та адаптивністю, щоб ефективно враховувати та реагувати на динамічні зміни у стані мережі. Враховуючи це, виникає виражена необхідність у вдосконаленні методів балансування навантаження, спрямованих на створення більш гнучких та адаптивних стратегій.

Особливо важливим в цьому контексті є розробка нових, більш ефективних методів балансування навантаження, що можуть адаптуватися до змін у топології мережі, забезпечуючи оптимізацію використання ресурсів та підвищення продуктивності багатошляхових мереж.

У низці наукових досліджень, що стосуються оптимізації балансування мережевого навантаження в архітектурі *SDN*, були висунуті різноманітні системи балансування [4, 5]. У таких системах контролер використовується для аналізу інформації, отриманої від комутаторів *OpenFlow*, та вносить зміни в таблиці потоків відповідно до стратегії балансування навантаження. Цей підхід дозволяє ефективно планувати шлях передачі даних у *SDN*-мережі, пристосовуючись до змін у топології або обсязі трафіку.

Проте, досліджені стратегії відносяться до статичних методів балансування навантаження, що обмежує їхню здатність адаптивно реагувати на динамічні зміни у стані навантаження мережі в режимі реального часу. Важливо відзначити, що ці методи не використовують на повну можливість *SDN*, оскільки не враховують глобальний погляд на структуру мережі та не взаємодіють із всіма її компонентами.

Однією з найважливіших аспектів подальшого дослідження в цьому напрямі є розробка більш адаптивних та гнучких стратегій балансування навантаження в архітектурі *SDN*, які використовують всі переваги цієї технології та здатні ефективно враховувати та реагувати на динамічні зміни у мережевому середовищі.

### **Архітектура системи балансування навантаження**

У статі пропонується використовувати систему підвищення ефективності використання ресурсів мережі *SDN* за рахунок використання модуля балансування навантаження на базі штучного інтелекту для обрахунку оптимальних маршрутів передачі пакетів у мережі спираючись на особливості корпоративних *SDN* мереж.

До особливостей корпоративних *SDN* мереж можна віднести такі пункти як віртуальні мережі та можливість підтримки декількох таких мереж на одному фізичному мережевому середовищі, а також самоподібний трафік, що є однією з головних особливостей багатьох корпоративних мереж.

Віртуальні мережі (*Virtual Networks*) – це технологія, що дозволяє створювати віртуальні ізольовані мережі в межах одного фізичного мережевого середовища. Ця концепція часто використовується у корпоративних комп'ютерних мережах для розділення одного фізичного мережевого ресурсу на кілька логічно розділених мереж, кожна з яких працює незалежно від інших.

У свою чергу, самоподібний трафік (*Self-Similar Traffic*) – це вид трафіку, який властивий багатьом сучасним корпоративним мережам, особливо якщо ці мережі передають великий обсяг даних та мають інтенсивний обмін інформацією. Цей вид трафіку характеризується тим, що його статистичні властивості залишаються схожими на різних масштабах часу, тобто трафік виявляє самоподібну або фрактальну структуру.

У даній роботі для належного відображення стану завантаження кожного маршруту використовуються критерії оцінки, такі як коефіцієнт використання пропускну здатності, втрати пакетів, затримка передачі та кількість стрибків для передачі пакетів. Ці параметри становлять основні міри, які визначають ефективність та навантаження на кожному маршруті, що дозволяє системі адаптивно реагувати на зміни в умовах мережі та вибирати оптимальні шляхи для передачі даних. Застосування цих критеріїв оцінки сприяє вдосконаленню стратегії балансування навантаження, забезпечуючи оптимальні рішення у реальному часі.

Архітектура мережі для запропонованої системи балансування навантаження *SDN* показана на рис. 1.

З метою оптимізації управління ресурсами мережі та зменшення навантаження на контролер *SDN*, пропонується архітектура з використанням балансувальника навантаження. Для визначення найменш завантаженого шляху в режимі реального часу, він використовує інтегрований стан навантаження, заснований на інформації про різні шляхи, отриманій від контролера *SDN*. Контролер періодично

надсилає дані про навантаження кожного шляху до *Load Balancer*, що дозволяє останньому миттєво реагувати на зміни у стані мережі [6].

Процедура роботи системи передбачає обробку нового потоку даних комутаторами *OpenFlow*. У разі відсутності відповідної таблиці потоку, контролер *SDN* визначає шлях передачі даних, спираючись на інформацію від *Load Balancer*. За наявності кількох шляхів, контролер

передає дані про навантаження шляхів до *Load Balancer* для визначення оптимального.

Після вибору оптимального шляху контролер створює таблиці потоків для передачі даних комутаторам *OpenFlow*. Таким чином, система здатна інтелектуально керувати навантаженням у *SDN*-мережі та забезпечити оптимальну пропускну здатність та надійність передачі даних.

На рис 2 зображена розроблена процедура керування даними в системі.

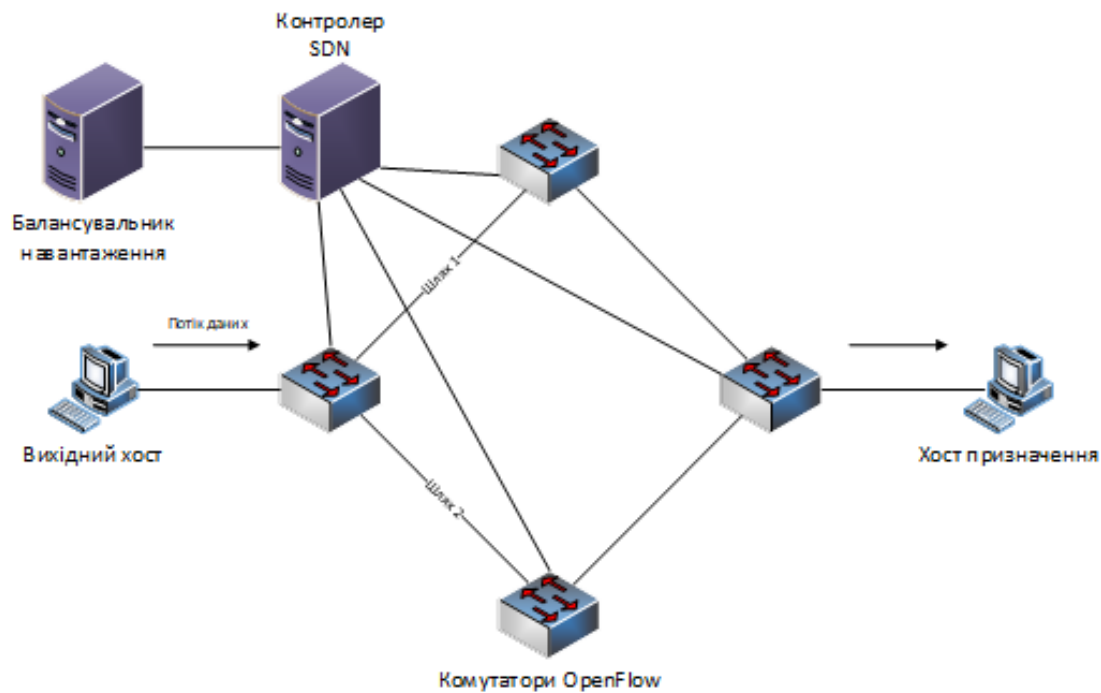


Рис. 1. Архітектура мережі

### Обчислення характеристик шляху

Процес передачі потоку даних у запропонованій концепції системи полягає у тому, що контролер *SDN* ініціює запит до таблиці потоків та збирає статистику для кожного порту від комутаторів *OpenFlow*, розташованих вздовж визначеного маршруту передачі. Зібрані дані про навантаження кожного каналу використовуються для розрахунку загального стану навантаження цього маршруту за допомогою штучної нейронної мережі.

Після періодичного збору інформації про маршрути з використанням протоколу *OpenFlow*, контролер *SDN* передає ці дані на сервер *Load Balancer* з метою

додаткової обробки. Це наближає архітектуру системи до стандартної моделі *SDN*, забезпечуючи ефективне зменшення навантаження на контролер.

Процес збору та обробки статистики відбувається динамічно, що дозволяє адаптивно реагувати на зміни в мережевому трафіку та забезпечує врахування актуального стану навантаження. У завершенні цього процесу обирається маршрут з найменшим навантаженням для остаточної передачі даних. Використання штучних нейронних мереж у даній стратегії дозволяє враховувати складні взаємодії мережевих факторів та прогнозувати оптимальний маршрут в змінних умовах мережевого середовища.

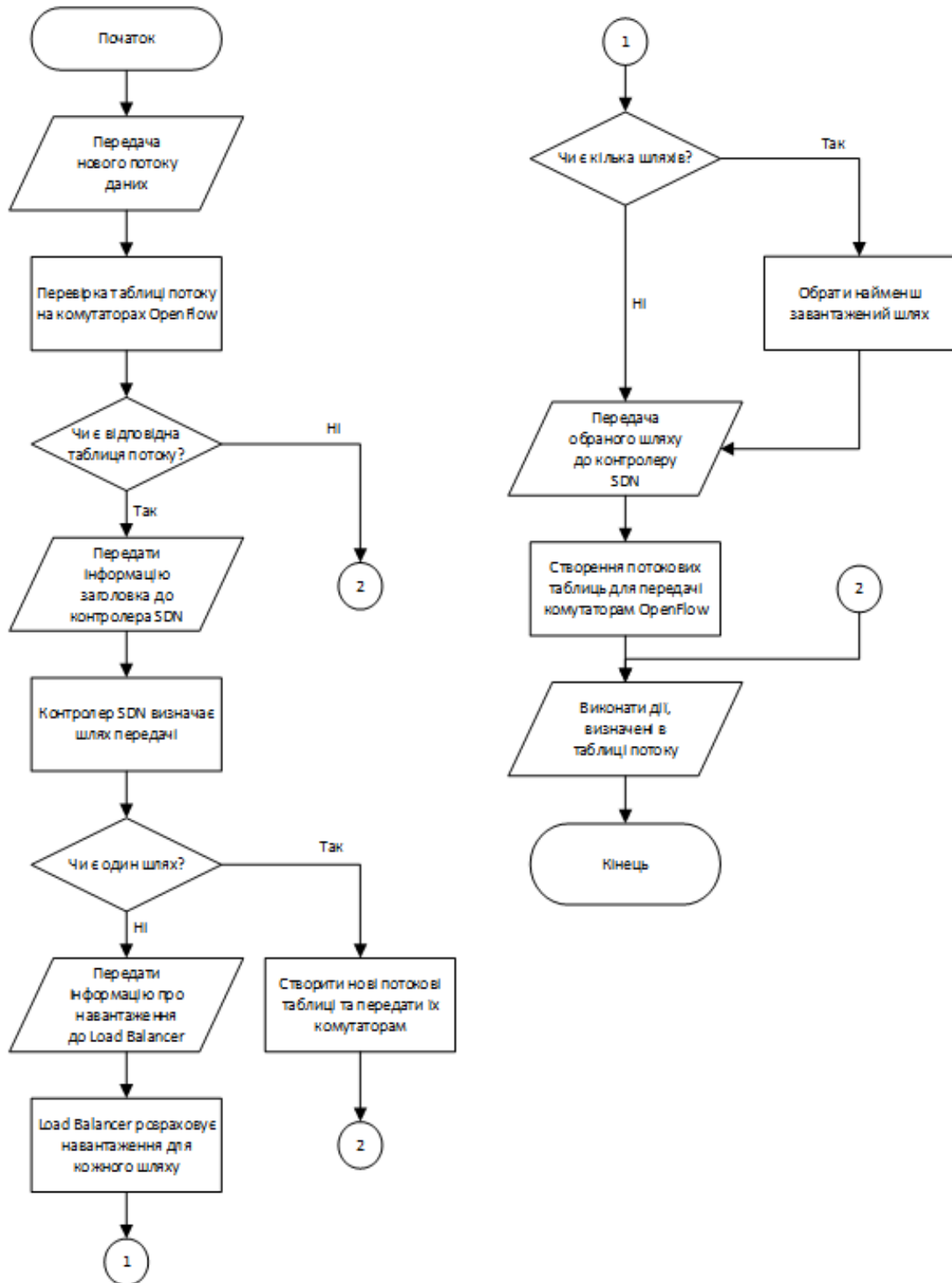


Рис. 2. Блок-схема роботи системи

В залежності від призначення мережі можна використовувати різні критерії для обчислення оптимального маршруту. Для розрахунку інтегрованого навантаження кожного маршруту у данній статі обрано чотири характеристики мережі –

коефіцієнт використання пропускної здатності, коефіцієнт втрат пакетів, затримка передачі та кількості стрибків для передачі пакетів

Коефіцієнт використання пропускної здатності (*Bandwidth*) представляє собою

важливий критерій для оцінки ефективності передачі даних у мережевому середовищі і відображає ступінь завантаження конкретного зв'язку. Значення коефіцієнту пропускної здатності, яке прямує до одиниці, вказує на високе використання пропускної здатності і може вказувати на потенційні перевантаження та затори в зв'язку.

Для обчислення коефіцієнту  $K_B$  контролер  $SDN$  збирає накопичені обсяги переданих байтів –  $N_b$  на відповідних портах комутаторів. Шляхом віднімання значень  $N_b$  та  $N_{b-1}$  (попередній період), отримуємо обсяг переданих даних за даний період, що репрезентує використану пропускну здатність зв'язку протягом цього часу. Останнім кроком є поділ використаної пропускної здатності на максимальну пропускну здатність  $V_{max}$ , що дозволяє отримати фінальне значення коефіцієнта використання пропускної здатності, яке можна обчислити за допомогою рівняння (1):

$$K_B = \frac{N_b - N_{b-1}}{V_{max}}. \quad (1)$$

Якщо конкретний маршрут включає в себе низку зв'язків, позначених як  $L_1, L_2, L_3 \dots L_n$ , кожен з них з характеристиками використання пропускної здатності  $K_{B1}, K_{B2}, K_{B3}, \dots, K_{Bn}$  відповідно, то співвідношення використання пропускної здатності для цього маршруту може бути обчислене шляхом взаємодії цих характеристик, виражене у формулі (2):

$$K_B = \frac{1}{n} \sum_{i=1}^n K_{Bi}. \quad (2)$$

Втрати пакетів є невід'ємною характеристикою передачі даних, яка виникає внаслідок переповнення комутаторів при обробці вхідних пакетів. Враховуючи це, коефіцієнт втрат пакетів ( $K_{PL}$ ) є важливим показником завантаження комутатора та загального стану маршруту. З метою оцінки цього параметра контролер  $SDN$  активно збирає інформацію про кількість переданих пакетів ( $P_T$ ) та отриманих пакетів ( $P_R$ ) на відповідних портах комутаторів. Отже, коефіцієнт втрат пакетів для

конкретного маршруту може бути визначений шляхом застосування формули (3):

$$K_{PL} = \frac{P_T - P_R}{P_T}. \quad (3)$$

Затримка передачі є важливим параметром, представляючи час, необхідний для комутатора-вузла на передачу даних. Цей показник тісно пов'язаний із продуктивністю комутаторів, розміром пакетів, які передаються, та станом черг передачі. Затримка передачі може служити індикатором умов переповнення зв'язку та загального навантаження комутатора.

Для оцінки затримки передачі ( $TD$ ) контролер  $SDN$  активно збирає інформацію про кількість переданих байтів ( $B$ ) за визначений період і швидкість передачі ( $TS$ ) на відповідних портах комутаторів *OpenFlow*. На основі цих даних затримка передачі може бути обчислена за допомогою формули (4):

$$TD_i = \frac{B}{TS}. \quad (4)$$

Якщо один маршрут складається з кількох зв'язків  $L_1, L_2, L_3 \dots L_n$  із відповідними затримками передачі  $TD_1, TD_2, TD_3 \dots TD_n$ , то загальна затримка цього маршруту може бути представлена як сума затримок на кожному зв'язку:

$$TD = \frac{1}{n} \sum_{i=1}^n TD_i. \quad (5)$$

Кількість переходів в мережі є суттєвим критерієм, який розглядається стратегією маршрутизації. Збільшення кількості переходів може призвести до збільшення ймовірності перевантаження та впливати на продуктивність мережі. В той час як при інших однакових умовах, передача пакету по маршруту із меншою кількістю переходів може зменшити ймовірність втрати пакетів і скоротити затримку передачі. Зберігаючи глобальну топологію мережі в базі даних контролера  $SDN$ , ми можемо отримати кількість переходів між двома комутаторами, використовуючи початковий комутатор та комутатор призначення як запит до бази даних.

Такий підхід дозволяє урахувувати фізичні характеристики мережі та

оптимізувати маршрутизацію, сприяючи ефективній передачі даних при мінімальній кількості переходів. Отже, критерій кількості переходів важливий для визначення оптимального маршруту з точки зору продуктивності та надійності мережі.

### **Підхід до балансування навантаження на основі штучних нейронних мереж**

Для вибору оптимального маршруту з найменшим навантаженням в мережі, необхідно аналізувати інтегровану інформацію про стан навантаження шляхів. Використання методів машинного навчання в цьому процесі важливо для покращення продуктивності та якості обслуговування мережі. Обчислювальна модель штучної нейронної мережі, яка є нелінійною та самоадаптованою, може бути ефективним інструментом для цієї задачі. У порівнянні з іншими методами машинного навчання, такими як логістична регресія, штучна нейронна мережа не обмежена векторами введення, що робить її ефективною для обробки невизначених даних трафіку в мережі.

Основою розробленої нейронної мережі є багатошаровий перцептрон. Ця мережа прямого поширення включає вхідний шар, один або декілька прихованих шарів та вихідний шар, через які сигнал проходить послідовно. Для навчання такої мережі застосовується метод зворотного поширення помилки, який базується на обчисленні градієнту функції помилки. У процесі навчання ваги нейронів коригуються з урахуванням сигналів, що надійшли з попереднього шару, і зсувів кожного шару, що обчислюється рекурсивно у зворотному напрямку від останнього шару до першого.

У даного виду мережі є декілька входів  $v_i$  ( $i=1, \dots, n$ ), один або декілька виходів та декілька внутрішніх нейронів. Через  $w_{ik}$  позначаємо вагу, що знаходиться на ребрі, що з'єднує  $i$ -й та  $k$ -й нейрони, а через  $o_i$  – вихід  $i$ -го нейрона. Використовуючи метод найменших квадратів, у випадку, якщо нам відомий навчальний приклад, функція помилки (6) матиме наступний вигляд:

$$E(\{w_{ik}\}) = \frac{1}{2} \sum_m (t_m - o_m)^2, \quad (6)$$

де  $t_m$  – правильні відповіді мережі, а  $m$  – вихідний нейрон.

Для модифікування ваг мережі буде використовуватись стохастичний градієнтний спуск. Тобто, після кожного навчального прикладу ваги на ребрах будуть коректуватись і пересуватись у багатовимірному просторі ваг. Оскільки нам потрібно дібратись до мінімуму помилки, то коректувати ваги ми будемо у сторону, протилежну градієнту. Тобто ми, опираючись на кожен групу правильних відповідей, будемо додавати до кожної ваги  $w_{ik}$

$$\Delta w_{ik} = -n \frac{\partial E}{\partial w_{ik}}, \quad (7)$$

де  $0 < n < 1$  – множник, що задає швидкість «руху».

Структура штучної нейронної мережі, яка складається з багатошарового перцептрону, ілюструється формулами (8) і (11). Такий підхід дозволяє моделі ефективно враховувати складні взаємозв'язки в даних трафіку та забезпечує високий рівень аналізу та передбачення стану навантаження шляху в мережі.

$$u_k = \sum_{i=1}^n w_{ik} v_i. \quad (8)$$

У рівнянні (7), де  $v_i$  ( $i=1, \dots, n$ ) відображає вектори введення, а  $w_{ik}$  ( $i=1, \dots, n$ ) представляє вагу нейрона  $k$ , визначається ваговий коефіцієнт, який вказує на ступінь впливу введення на дію нейрона. Зокрема, якщо  $w_{ik} > 0$ , це вказує на стан збудження, тоді як  $w_{ik} < 0$  вказує на стан стримування. Тут  $n$  позначає кількість векторів введення, що використовуються для навчання нейронної мережі. Параметр  $u_k$  визначається як лінійна комбінація вектора введення та ваги нейрона.

Похідна (9) у такому випадку, для нейрону останнього рівня, розраховується наступним чином:

$$\frac{\partial E}{\partial w_{ik}} = \frac{\partial E}{\partial u_k} \frac{\partial u_k}{\partial w_{ik}} = v_i \frac{\partial E}{\partial u_k}. \quad (9)$$

На загальну помилку  $u_k$  також впливає тільки в рамках вихідного нейрона всієї мережі:

$$\frac{\partial E}{\partial u_k} = \frac{\partial E}{\partial o_k} \frac{\partial o_k}{\partial u_k} = \left( \frac{\partial}{\partial o_k} \frac{1}{2} \sum_m (t_m - o_m)^2 \right) \left( \frac{\partial y_k}{\partial u_k} \right) = \left( \frac{1}{2} \frac{\partial}{\partial o_k} (t_k - o_k)^2 \right) (o_k(1 - o_k)) = -o_k(1 - o_k)(t_k - o_k), \quad (10)$$

де  $y_k$  – це функція активації (11).

$$y_k = f(u_k + b_k). \quad (11)$$

Функція активації  $f(x)$  (11) визначає реакцію нейрона на вхід, враховуючи ваги та порогове значення. Порогове значення нейрона (зсув) позначається як  $b_k$  і використовується для управління входом функції активації. Цей параметр визначає, чи має стан збудження або стримування певний нейрон. Вихід нейрона, позначений як  $y_k$ , є результатом функції активації та представляє собою фінальний вихід нейрона. Усі ці параметри та елементи структури штучної нейронної мережі визначають її здатність до адаптації та вирішення завдань навчання чи прогнозування в мережевому середовищі.

Для інтеграції інформації про шлях використовується 3-шарова штучна нейронна мережа з методом зворотного поширення. Процес навчання нейронної мережі складається з двох основних фаз. Перша фаза – це фаза навчання вперед. У цей період вхідні нейрони обробляють вхідні вектори з попередньо встановленими вагами за допомогою функції активації. Після цього вихід з прихованого шару передається на вихідний шар, де генеруються кінцеві результати. Цей процес навчання вперед передається, і стан кожного вузла впливає на вихід наступних вузлів.

Після завершення фази навчання нейронна мережа переходить до фази налаштування ваг. На цьому етапі помилка між кінцевими результатами та очікуваним виводом використовується для коригування ваг. У випадку, якщо результати вихідного шару надто великі, помилка використовується для зворотного розповсюдження, щоб змінити ваги нейронів у кожному шарі. Цей процес налаштування нейронної мережі продовжується до тих пір, поки помилка не досягне прийнятної рівня.

Для ефективного балансування трафіку у корпоративних SDN мережах використовується 4 характеристики як вектор введення, що відповідає наявності 4 вхідних нейронів у нейронній мережі. Однак визначення кількості нейронів у прихованому шарі важливо для досягнення оптимальних результатів. Для визначення кількості нейронів у прихованому шарі використовується формула (12) [9], що є одним із підходів до оптимізації структури нейронної мережі.

$$N_h = \sqrt{n_{out} + n_{in}} + a. \quad (12)$$

У цій формулі  $N_h$  позначає кількість нейронів у прихованому шарі,  $n_{in}$  – кількість нейронів у вхідному шарі,  $n_{out}$  – кількість нейронів у вихідному шарі, а – константа від 1 до 10. Для проведення експерименту будуть встановлені значення 1000 як кількість навчань, а ціль навчання встановлено як 0,001. Щодо кількості нейронів у прихованому шарі будуть вибрані різні значення константи  $a$ :  $a=1$ ,  $a=3$ ,  $a=5$ ,  $a=7$  та  $a=9$ . Відповідно, отримані кількості нейронів у прихованому шарі дорівнюють 3, 5, 7, 9 та 11.

Після визначення експериментальних параметрів, вибрані оптимальні параметри для створення штучної нейронної мережі зворотного поширення. Таким чином, структура штучної нейронної мережі зворотного поширення включає в себе вхідний шар із  $n_{in}$  нейронами, прихований шар із  $N_h$  нейронами та вихідний шар із  $n_{out}$  нейронами, відповідно до обраних значень константи  $a$  та кількості нейронів у прихованому шарі.

### Висновки

Застосування штучного інтелекту (ШІ) в корпоративних мережах SDN має потенціал для підвищення якості та ефективності мережі, зниження витрат і підвищення безпеки.

Використання методу балансування навантаження, який, за рахунок розрахунку інтегрованого навантаження кожного маршруту на основі характеристик мережі та використання системи штучного інтелекту, дозволяє обрати оптимальний маршрут з найменшим навантаженням для передачі потоку даних у корпоративній SDN мережі з урахуванням їх особливостей.

Застосування запропонованої схеми сприяє підвищенню ефективності використання ресурсів мережі SDN та забезпечує оптимальне розподілення навантаження, що в результаті сприяє підвищенню продуктивності та якості обслуговування у системах передачі даних.

### **Література**

1. Kulakov Y. O., Korenko D. V. Methods of applying artificial intelligence in software-defined networks. *Problems of Informatization and Control*. 2023. Vol. 1, no. 73. P. 23–27.
2. Chim T. W., Yeung K. L. Traffic distribution over equal-cost-multi-paths. *2004 IEEE International Conference on Communications* : proceedings, Paris, France, 20–24 June 2004 / IEEE. Piscataway, 2004. Vol. 2. P. 1207–1211.
3. William J. D., Towles B. Principles and practices of interconnection networks. Amsterdam : Morgan Kaufmann Publishers, 2004. 550 p.
4. Alnaser A. M., Kulakov Y. O., Korenko D. V. Modified Method of Traffic Engineering in DCN with a Ramified Topology. *International Journal of Advanced Computer Science and Applications*. 2021. Vol. 12, no. 12. P. 439–446.
5. Korenko D. V. et al. Creation of the method of multipath routing using known paths in software-defined networks. *Technology audit and production reserves*. 2022. Vol. 4, no. 2(66). P. 19–24.
6. Handigol N. et al. Plug-n-Serve: Load-Balancing Web Traffic using OpenFlow. *ACM SIGCOMM 2009 : Demos, Barcelona, Spain, 17–21 August 2009 / ACM SIG, 2009*. 2 p.
7. Wang R., Butnariu D., Rexford J. OpenFlow-based server load balancing gone wild. *11th USENIX conference on Hot topics in management of internet, cloud, and enterprise networks and services (Hot-ICE'11)* : proceedings, Boston, MA, USA, 29 March 2011 / USENIX Association. Berkeley, 2011. P. 12–12.
8. Hu Y. et al. BalanceFlow: Controller load balancing for OpenFlow networks. *2012 IEEE 2nd International Conference on Cloud Computing and Intelligence Systems* : proceedings, Hangzhou, China, 30 October–01 November 2012 / IEEE. Beijing, 2012. Vol. 2. P. 780–785.
9. Gao J. Artificial neural network principle and simulation examples. Beijing : Mechanical Industry Press, 2007.

**Кулаков Ю.О., Коренко Д.В.**

## **МЕТОД БАЛАНСУВАННЯ НАВАНТАЖЕННЯ В МЕРЕЖАХ SDN З ВИКОРИСТАННЯМ ШТУЧНОГО ІНТЕЛЕКТУ**

*Ця робота присвячена огляду методу балансування навантаження у корпоративних мережах з використанням технології SDN з використанням штучного інтелекту. У роботі розглянуті особливості та методи використання ШІ у цих мережах, а також виявлені потенційні проблеми, які можуть виникнути.*

*У роботі проведено огляд методу балансування навантаження у корпоративних SDN мережах з використанням ШІ. Виявлено, що ШІ дозволяє автоматизувати процес балансування навантаження у мережі. Використання ШІ сприяє покращенню ефективності мережі, зменшенню витрат і поліпшенню безпеки.*

*Загалом, робота висуває ідею використання штучного інтелекту у корпоративних мережах з використанням технології SDN для підвищення ефективності роботи*



*мережі та забезпечення безпеки. Оглянуто метод застосування ШІ в цих мережах та намічені перспективні напрямки досліджень.*

*Ця робота дає теоретичний огляд методу балансування навантаження у корпоративних SDN мережах з урахуванням їх особливостей, а також ставить основу для подальших досліджень у цій області. Впровадження штучного інтелекту в корпоративні мережі є актуальним завданням, оскільки воно сприяє покращенню ефективності та безпеки мереж та відкриває нові можливості для їх розвитку.*

**Ключові слова:** *штучний інтелект; корпоративні мережі; SDN; Smart Grids; транспортні мережі; БПЛА мережі.*

**Kulakov Y.O., Korenko D.V.**

## **LOAD BALANCING METHOD IN SDN NETWORKS USING ARTIFICIAL INTELLIGENCE**

*This paper provides an overview of load balancing in corporate networks using SDN technology and artificial intelligence. It discusses the features and methods of using AI in these networks and identifies potential problems.*

*The paper reviews the method of load balancing in corporate SDN networks using AI and concludes that AI can automate the process of load balancing in the network. The implementation of AI can enhance network efficiency, decrease costs, and bolster security. This paper suggests the use of artificial intelligence in corporate networks through SDN technology to improve network efficiency and security.*

*This paper suggests the use of artificial intelligence in corporate networks through SDN technology to improve network efficiency and security. The paper reviews the method of applying AI in these networks and outlines promising areas of research.*

*This paper presents a theoretical overview of load balancing in enterprise SDN networks, considering their features, and establishes the foundation for future research in this area. The integration of artificial intelligence into corporate networks is a pressing task as it enhances network efficiency and security while also creating new opportunities for development.*

**Keywords:** *artificial intelligence; enterprise networks; SDN; Smart Grids; transport networks; UAV networks.*