

Ганенко Л.Д.

Центральноукраїнський державний університет ім. В. Винниченка, Кропивницький

Поперешняк С.В.

Державний університет телекомунікацій, Київ

## ВИКОРИСТАННЯ ТЕХНОЛОГІЇ СОКЕТІВ У МЕРЕЖНІЙ СИСТЕМІ ВІРТУАЛЬНОЇ ЛАБОРАТОРІЇ НА БАЗІ АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ

**Анотація.** Використання систем віртуальних лабораторій стало сучасним трендом у багатьох сферах людської діяльності. Сучасні мережеві віртуальні лабораторії використовують технологію віртуальної реальності для реалізації лабораторного моделювання та технологію інтерактивного середовища для керування моделюванням. У роботі розглядаються переваги, принципи побудови віртуальної лабораторії та існуючі проблеми в механізмі комунікації. Зокрема, при обміні даними між вузловими машинами в системі мережевої віртуальної лабораторії мережевий рівень не може забезпечити однакову передачу пакетних даних. У статті розглядаються сучасні методи машинного навчання: навчання з наглядом, без нагляду, напівнаставлення та навчання з підкріпленням. Визначено, що основними характеристиками архітектури системи віртуальної лабораторії є відкритість, модульність та зв'язок у режимі реального часу. Механізм зв'язку системи віртуальної лабораторії не відповідає вимогам високої пропускної здатності та двостороннього зв'язку в реальному часі. Для вирішення проблеми технічних вимог високої пропускної здатності та наднизької затримки системи віртуальної лабораторії запропоновано технологію комунікаційного планування на основі алгоритму глибокого машинного навчання. У роботі використовується механізм Socket для реалізації зв'язку на основі протоколу TCP/IP. Відповідно до технічних вимог високої пропускної здатності та наднизької затримки мережевої системи віртуального експерименту запропоновано технологію самонастроювання та оптимального планування комунікаційної мережі на основі алгоритму глибокого машинного навчання. Запропонована технологія дозволяє прийняти оптимальне рішення щодо планування ресурсів та реалізувати оптимізацію непрямого вибору параметрів системи.

**Ключові слова:** мережева віртуальна лабораторія, машинне навчання, сокети, протокол TCP/IP, технологія планування.

Hanenko L. D.,

Volodymyr Vynnychenko Central Ukrainian State University, Kropyvnytskyi

Poperehnyak S. V.

State University of Telecommunications, Kyiv

## USE OF SOCKETS TECHNOLOGY IN THE NETWORK SYSTEM OF A VIRTUAL LABORATORY BASED ON MACHINE LEARNING ALGORITHMS

**Abstract.** The use of virtual laboratory systems has become a modern trend in many areas of human activity. Modern networked virtual laboratories use virtual reality technology to implement laboratory simulations and interactive environment technology to control simulations. The work examines the advantages, principles of building a virtual laboratory and existing problems in the communication mechanism. In particular, when exchanging data between node machines in a network virtual laboratory system, the network layer cannot ensure the same transmission of packet data. The article reviews modern methods of machine learning: supervised, unsupervised, semi-supervised and reinforcement learning. It was determined that the main characteristics of the virtual laboratory system architecture are openness, modularity and real-time communication. The

*communication mechanism of the virtual laboratory system does not meet the requirements of high bandwidth and real-time two-way communication. To solve the problem of technical requirements of high throughput and ultra-low delay of the virtual laboratory system, a communication planning technology based on a deep machine learning algorithm is proposed. The work uses the Socket mechanism to implement communication based on the TCP/IP protocol. In accordance with the technical requirements of high throughput and ultra-low latency of the network system of the virtual experiment, a technology of self-tuning and optimal planning of the communication network based on the deep machine learning algorithm is proposed. The proposed technology makes it possible to make the optimal decision regarding resource planning and implement optimization of the indirect selection of system parameters.*

**Keywords:** network virtual laboratory, machine learning, sockets, TCP/IP protocol, planning technology.

### 1. Постановка проблеми.

Завдяки стрімкому розвитку комп'ютерних технологій використання систем віртуальних лабораторій стало сучасною тенденцією для багатьох сфер людської діяльності. Зокрема віртуальні лабораторії дозволяють як реалізувати дистанційне навчання в освітньому процесі [1], так і виконати складні експериментальні наукові дослідження [2]. Прерогативами віртуальних лабораторій є: низька вартість, висока надійність, безпека виконання експериментів, можливість імітації екстремальних умов експерименту, гнучкість та зручність.

Сучасні мережеві віртуальні лабораторії використовують технологію віртуальної реальності для реалізації лабораторного моделювання та технології інтерактивного середовища для управління моделюванням. В той же час механізм зв'язку цих технологій не відповідає вимогам високої пропускної здатності та двостороннього зв'язку в реальному часі. Тому дослідження даної проблеми є актуальним.

### 2. Аналіз останніх досліджень і публікацій.

Багато вітчизняних та закордонних вчених досліджували віртуальні лабораторії та їх механізми зв'язку. Американські вчені вперше висунули концепцію системи віртуальної лабораторії, яка використовується у викладанні електронних технологій із застосуванням мережевої віртуальної лабораторії у моделюванні схем. [3, 4]. Подібним чином система MASH у Сполучених Штатах є великомасштабною мережевою системою віртуальної лабораторії. Як мережева експериментальна платформа, вона має важливе значення для мережевих досліджень у всьому світі [5]. Відповідними дослідницькими інститутами Китаю було запропоновано ряд систем віртуальних лабораторій, таких як онлайн-віртуальна лабораторія для інженерних випробувань [6, 7]. Механізм зв'язку Socket, як механізм зв'язку у системі віртуальної лабораторії, вивчався великою кількістю вчених. Корпорація Майкрософт вперше вивчила механізм зв'язку Socket і запровадила новий технічний стандарт [8, 9]. На початку 21 століття відповідні науково-дослідні інститути запустили HTML5, завдяки чому комунікаційна технологія Socket вступила в новий технологічний період [10, 11].

### 3. Мета і задачі дослідження.

Метою дослідження є розробка технології самонастроювання та оптимального планування комунікаційної мережі на основі алгоритму глибокого машинного навчання, відповідно до технічних вимог високої пропускної здатності та наднизької затримки системи віртуального лабораторії.

Сформулюємо задачі дослідження:

- проаналізувати основні принципи та поточні проблеми системи віртуальної лабораторії;
- проаналізувати принципи машинного навчання та його застосування в комунікаційних технологіях;

– відповідно до технічних вимог великої пропускної здатності та наднизької затримки поточної системи віртуальної лабораторії запропонувати загальний підхід, заснований на алгоритмі глибокого машинного навчання.

#### 4. Результати дослідження

##### 4.1. Принцип комунікації Socket та механізм роботи у системі віртуальної лабораторії

Розглянемо принципи побудови віртуальної лабораторії та існуючі проблеми в механізмі зв'язку.

*Огляд системи віртуальної лабораторії.* Система віртуальної лабораторії є продуктом розвитку комп'ютерних мережевих технологій. Вона базується на комп'ютерній архітектурі та технології для створення наближеного до реального середовища наукових досліджень. У порівнянні з традиційними лабораторіями мережева лабораторна система має наступні переваги:

1. Зручний дизайн. Моделює компоненти, які необхідні лабораторії, за допомогою програмних елементів. У системному програмуванні мережевої віртуальної лабораторії передача даних у мережі реалізується субмодулем обміну повідомленнями на основі протоколу TCP/IP.

2. Безпека та надійність. Можна проводити нереалістичні експерименти з моделюванням. Завдяки віртуальному експериментальному середовищу ми можемо уникнути пов'язаних з цим небезпек і одночасно досягти експериментальних результатів.

3. Стійкість експерименту. Віртуальний експеримент можна постійно оновлювати та змінювати, таким чином вирішуючи проблему традиційного експериментального обладнання. Як правило, у клієнтсько-серверній програмі, написаній у сокетах, серверна програма прослуховує певний порт і чекає запиту на з'єднання від користувача. Цей порт зазвичай відкритий для всіх клієнтів, тому клієнту сервер відомий.

4. Дистанційний експеримент можна реалізувати будь-де та будь-коли. Система віртуального експерименту вирішує обмеження часу та простору експериментатора.

Основними характеристиками архітектури системи віртуальної лабораторії є відкритість, модульність і зв'язок у режимі реального часу.

Відкритість системи віртуальної лабораторії полягає у постійному підвищенні продуктивності і функції моделювання мережі за допомогою оновлення ядра моделювання мережі. У той же час система віртуальної лабораторії може забезпечити інтерфейс для нещодавно розробленої мережевої технології та застосування імітаційної моделі.

Модульність означає, що вся система віртуальної лабораторії може бути логічно відокремлена, а кожна частина системи може бути відокремлена незалежно від обміну даними. Характеристики модульності призначені для задоволення експериментальних потреб трьох аспектів: – вимоги та дизайн експериментатора для конкретної мережі в галузі дослідження; – незалежність і цілісність розвитку всієї системи віртуальної лабораторії; – досягнення синхронного розвитку з новою технологією.

Характеристика зв'язку в реальному часі означає, що механізм зв'язку системи віртуальної лабораторії повинен відповідати вимогам великої пропускної здатності та двостороннього зв'язку в реальному часі та забезпечити надання дружнього інтерфейсу для користувачів системи. Механізм зв'язку є важливим ядром системи віртуальної лабораторії.

Рис. 1 ілюструє архітектуру системи віртуальної лабораторії, яка базується на серверній системі NS2, за допомогою відповідних сценаріїв моделювання для зібрання статистичних даних, моделювання збою зв'язку та змін маршрутизації.

Фізична топологія мережевої лабораторії складається з трьох частин: клієнта, сервера та контролера. Клієнт безпосередньо взаємодіє з клієнтом. Кілька клієнтів можуть входити на сервер одночасно та проводити різні експерименти.

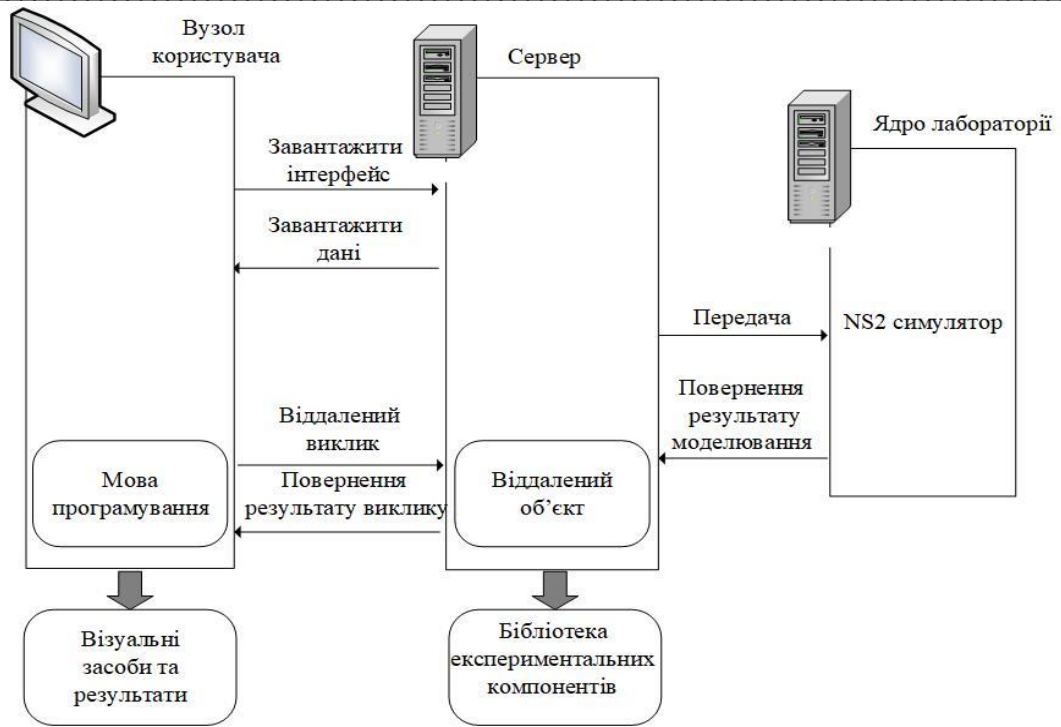


Рис. 1. Схема системи віртуальної лабораторії.

Сервер і всі експериментальні консолі знаходяться у високошвидкісній локальній мережі лабораторії під час кожного експерименту. Експеримент можна проводити в будь-який час. Вид клієнта та його структуру зображено на рис. 2.

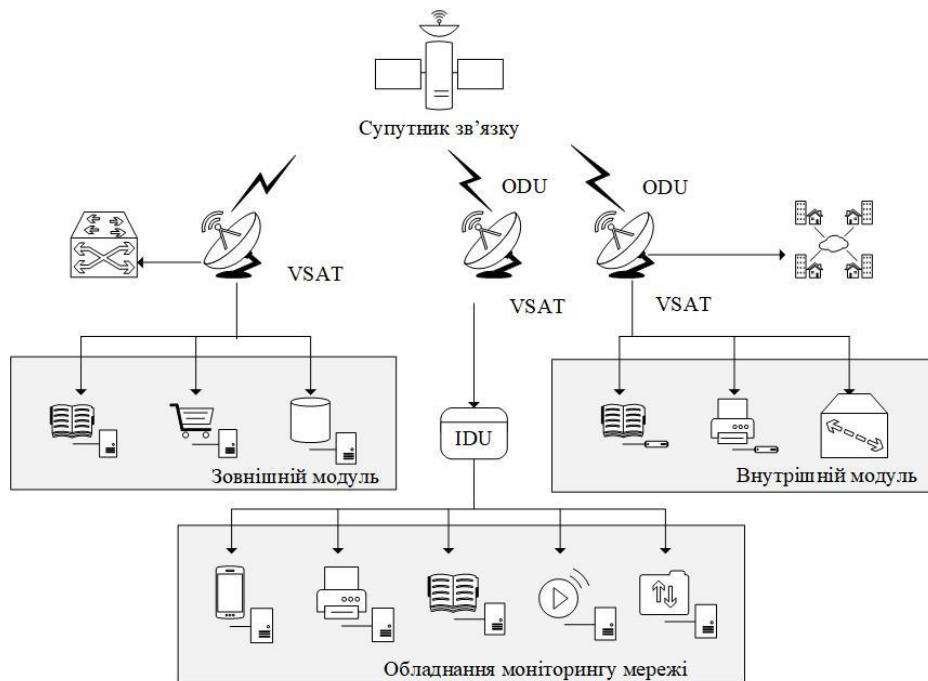


Рис. 2. Функціональна схема системи віртуальної лабораторії

**Механізм зв'язку Socket.** Розглянемо механізм зв'язку Socket, його застосування та проблеми механізму зв'язку Socket у системі віртуальної лабораторії.

Зв'язок Socket – це зручний засіб передачі даних між віддаленими вузлами. Його перевага полягає в тому, що не потрібно виділяти певний вузол під серверні потреби. Будь-який комп'ютер в мережі, на якому встановлена серверна програма, може бути сервером і обробляти повідомлення, які надходять. Основні компоненти Socket:

1. Релевантні протоколи (визначено протоколи, які використовуються для механізму зв'язку). Загальними протоколами є протокол TCP/IP або протокол UDP. Механізм зв'язку Socket, який досліджується, здійснюється за протоколом TCP/IP.

2. Локальна адреса (адреса, яку Інтернет призначає хосту за протоколом TCP/IP).

3. Номер локального порту (використовується для розрізнення програм, пов'язаних із локальною роботою).

4. Віддалена адреса (адреса, яка призначена клієнтському хосту відповідно до протоколу TCP/IP).

5. Номер віддаленого порту (використовується для розпізнавання програми, запущеної віддаленим хостом).

Основою механізму зв'язку Socket є функція Socket. У механізмі зв'язку Socket використано дев'ять основних функцій, а саме:

1. Створити функцію Socket - Socket (). Для створення зв'язку із використанням механізму зв'язку Socket, потрібно використовувати цю функцію як початкову.

2. Використати функцію Bind (), щоб указати локальну IP-адресу та відповідний номер порту.

3. Connect () використовується для зв'язку з клієнтами.

4. Accept () використовується для підключення клієнтів.

5. Прослухати підключення listen ().

6. Передача даних send ().

7. Отримання даних Rec().

8. Вибір стану сокету select (). Для механізму зв'язку Socket можуть існувати різноманітні стани, які можна запитувати за допомогою наведених вище функцій.

9. Закрити сокет closesocket (). Після завершення передачі даних згадані вище функції можна вимкнути.

Весь процес зв'язку Socket можна підсумувати так: «Відкрити-Читати-Записати-Закрити». За протоколом TCP/IP програма спочатку відкриває мережеве з'єднання, а потім може закрити з'єднання за допомогою безперервної операції читання-запису, згаданої вище. Відповідний процес відображено на рис. 3.

У мережевій системі віртуальної лабораторії механізм зв'язку Socket використовується для забезпечення своєчасного зв'язку. Основним додатком у системі віртуальної лабораторії є Socket.IO. Цей програмний пакет використовується в рамках Node.js. Він підтримує різноманітні протоколи зв'язку та може виконувати кросбраузерні запити. Socket.IO може реалізувати взаємодію між браузерами та комунікаційними пристроями, водночас ізолюючи відмінності між різними режимами передачі. Зв'язок Socket певною мірою вирішує питання складності та своєчасності зв'язку у системі віртуальної мережевої лабораторії.

#### **4.2. Оптимізація комунікаційної технології Socket на основі глибокого машинного навчання**

*Технологія глибокого машинного навчання.* Машинне навчання можна класифікувати за стилем навчання та алгоритмічною подібністю.

За стилем навчання машинне навчання ділиться на:

1) Контрольоване навчання. Даний метод використовує марковані дані для навчання алгоритмів, які класифікують дані або прогнозують результати. Він дозволяє спрогнозувати події за допомогою минулого досвіду. Метод контрольованого навчання аналізує відомий навчальний набір даних і вводить функцію, на основі якої здійснює прогноз. [12].

2) Неконтрольоване навчання. У цьому методі навчання немає необхідності маркувати дані, прогнозувати будь-яку цільову змінну чи змінну результату. Метою створення відповідної моделі є створення внутрішньої структури даних. Звичайне неконтрольоване навчання включає: розкладання сингулярного значення, аналіз головних компонентів, аналіз незалежних компонентів тощо. [13]

3) Напівконтрольоване навчання. Цей метод є поєднанням контрольованого та неконтрольованого навчання (часткове навчання). Він використовує невелику кількість

маркованих даних і велику кількість немаркованих. Цей метод навчання використовується для прогнозування, яке потребує вивчення структури даних моделі [14].

4) Навчання з підкріпленням. Даний метод дозволяє покращити здатність машин приймати самостійні рішення у невідомому середовищі. Щоб використовувати отримані знання для найточнішого оцінювання поведінки, машинам потрібно постійно пробувати та навчатися помилок.

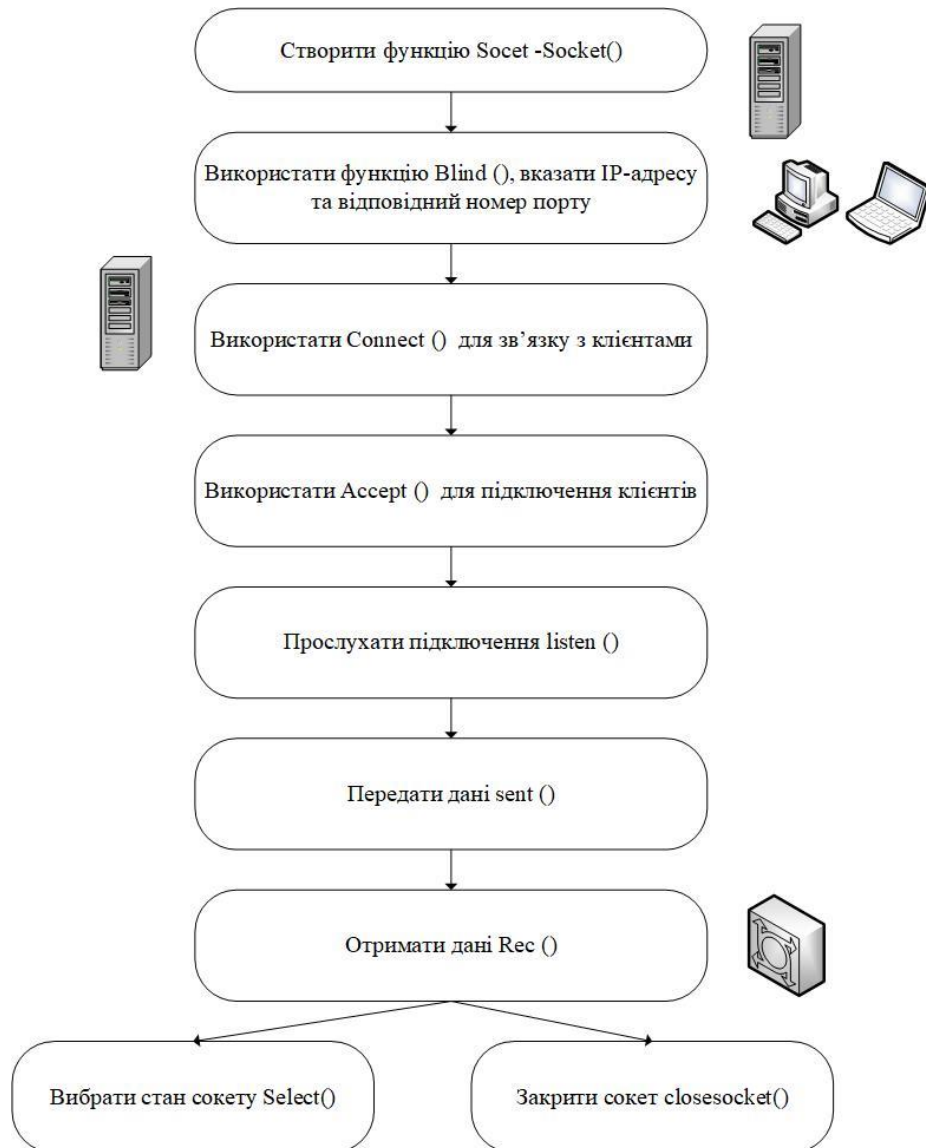


Рис. 3. Схема архітектури механізму зв'язку Socket

За алгоритмічною подібністю виділяють:

1. Алгоритм регресії. Типовий алгоритм, який включає логістичну регресію, метод найменших квадратів, алгоритм множинної адаптивної регресії, алгоритм покрокової регресії та оцінку локального згладжування розсіювання.
2. Навчання дерева рішень. Встановлює модель рішення деревоподібної структури на основі атрибутів, що вирішує проблеми регресійної моделі.
3. Алгоритм Байеса.
4. Алгоритм кластеризації.
5. Алгоритм глибокого навчання. Алгоритм, який базується на штучному нейронному алгоритмі. Переваги машинного алгоритму глибокого навчання із зростанням обчислювальної потужності комп'ютерів стають все більш очевидними.

На рис. 4 представлена відповідна схема класифікації.



Рис. 4. Класифікація алгоритмів машинного навчання

З метою вивчення характеристик механізму мережевої комунікації у віртуальній лабораторії розглянемо алгоритм навчання з підкріпленням [15]. Даний алгоритм машинного навчання є алгоритмом навчання без моделі. Його головною особливістю є асинхронне динамічне програмування. Агент навчання в алгоритмі називається інтелектуальним агентом навчання. Даний процес навчання можна є марковським процесом прийняття рішень.

Розглянемо ключові компоненти. Набір агентів середовища представлено як  $S(s_0, s_1, s_2, s_3, s_4 \dots)$ . Комбінацію дії відповідних агентів можна підсумувати як  $A(a_1, a_2, a_3, a_4, a_5, \dots)$

Агенти вибирають дію, яка максимізує основну функцію машинного навчання з підкріпленням – функцію  $Q$  (функція оцінювання). Дану функцію представлено таким чином:

$$Q(s, a) = R(s, a) + \gamma V^*(\Phi(s, a)) \quad (1)$$

де  $R(s, a)$  – негайна винагорода,  $\gamma$  – константа.

Відповідну оптимальну стратегію вибору дій можна задати формулою:

$$\pi^* = \arg \text{Max}_{\pi} V^{\pi}(s) (\forall s \in S) \quad (2)$$

Відповідні критерії оцінювання наведено у формулі:

$$V^{\pi}(s_t) = R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \dots = \sum_{i=0}^{\infty} \gamma^i R_{t+i} \quad (3)$$

На рис. 5 зображено загальний базовий процес встановлення механізму зв'язку Socket.

В проектуванні програми мережевої системи віртуальної лабораторії потрібно обробляти інформацію кількох клієнтів одночасно, тому підмодуль обміну повідомленнями також включає принцип так званого «конкурентного сервера». Реалізація паралельних серверів є типовим застосуванням багатопоточності та зв'язку через Socket у Java. Основний принцип полягає в тому, що обслуговуюча сторона встановлює потік для обробки інформації для входу користувача та виконує необхідну обробку після отримання запиту користувача.

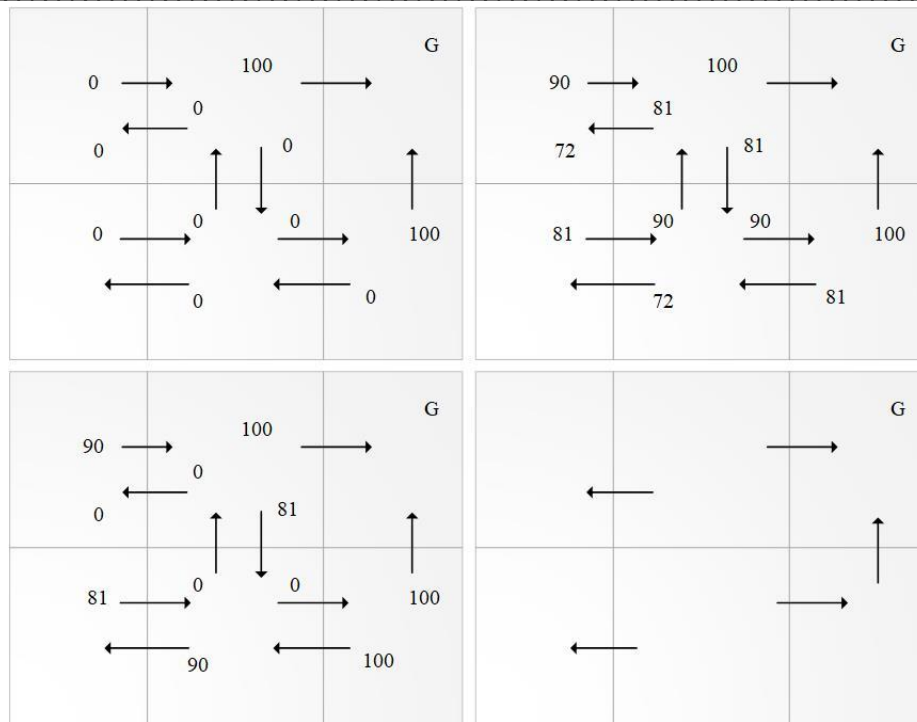


Рис. 5. Діаграма стану поглинання відповідного агента

**Оптимізація комунікаційної технології Socket на основі глибокого машинного навчання.** Розглянемо технологію оптимального планування комунікаційної мережі з самонастроюванням на основі алгоритму глибокого машинного навчання, який включає алгоритм навчання з підкріпленням. Використовуючи запропоновану технологію можна розв’язати такі задачі як прийняття рішень щодо планування ресурсів та оптимізація непрямого вибору параметрів системи.

Система віртуальної лабораторії є багатовузловою системою зв’язку. При виборі моделі агента вибрано багатоагентну систему. У цій системі однаковий режим обміну інформацією машинного навчання, але відповідні агенти різні. На основі різних агентів стратегія централізованого навчання з макробазовою станцією як агентом і стратегія розподіленого машинного навчання з головою кластера як агентом додаються до стратегії зв’язку Socket.

За стратегію вибору моделі прийнято бездротову ультращільну макро-мікро гетерогенну стільникову мережу. Після отримання інформації, надісланій одержувачем  $B$ , хост  $A$  спочатку перевірить значення номер підтвердження. Якщо це 1 і  $ack=1$ , це правильно. У цей час хост  $A$  надсилає номер підтвердження  $= (seq+1 \text{ отримувача } B)$ ,  $ack=1$  отримувачу  $B$ , а  $B$  отримує номер підтвердження та підтвердження після його отримання. Якщо він правильний, механізм «рукоштовання» завершено, і дві сторони починають звичайне спілкування. Відповідний вираз наведено у формулі:

$$T_p = \sum_{C_i \in C} \sum_{f \in C_i} \sum_{u_f \in U_f} \sum_{n=1}^{N_{num}} \rho_{f,u_f}^n \Delta B \log_2(1 + \gamma_{f,u_f}^n) \quad (4)$$

Відношення сигнал/шум між відповідним користувачем  $u$  і базовою станцією  $f$ , до якої він звертається, подано формулою:

$$\gamma_{f,u}^m = \frac{P_T G_{f,u}}{\sum_{C_i \in C/C_k} \sum_{f' \in C_i} \sum_{u_n \in U_{f'}} \rho_{f',u_n}^m P_T G_{f',u} + N_0} \quad (5)$$

Підсилення каналу між відповідним користувачем і базовою станцією обчислюється за формулою:

$$G_{f,u} = S_u d_{f,u}^{-\alpha} g_{f,u} \quad (6)$$



Відповідні обмеження оптимізації системи оброблюються за формулою:

$$\sum_{f \in C_i} \sum_{u_f \in U_f} \rho_{f,u_f}^n = 1 \quad (7)$$

Для повного використання обмежених ресурсів спектру в механізмі зв'язку Socket та вирішення проблеми кількості спектрів використовується мультиплексування з частотним поділом. Усі базові станції в цій архітектурі мережі використовують однакові ресурси смуги частот. Для вирішення проблеми перешкод, технологія мультиплексування з частотним поділом поєднується з технологією координації перешкод між осередками, а в мультиплексуванні з частотним поділом використовується режим м'якого та жорсткого поділу частот. У режимі мультиплексування з жорстким розділенням частот різні ресурси піддіапазону розподіляються для різних макрозон, що дозволяє ефективно уникати перешкод та покращувати ефективність зв'язку в реальному часі та пропускну здатність усієї системи.

На основі наведеного вище алгоритму оптимізації можна запропонувати загальну структуру системи віртуальної лабораторії. Як показано на рис. 6, основна частина ядра системи розділена на дві частини: розробка клієнтської програми та розробка серверної програми. В основі сервера лежить встановлення та підтримка комунікаційного з'єднання Socket. Алгоритм машинного навчання, Q-підсилення, збільшує пропускну здатність усієї системи та ефективність комунікації в реальному часі. Зі сторони клієнта необхідно встановити Socket-з'єднання між запитом, надісланим користувачем, і сервером, а також зібрати і обробити необхідну інформацію.

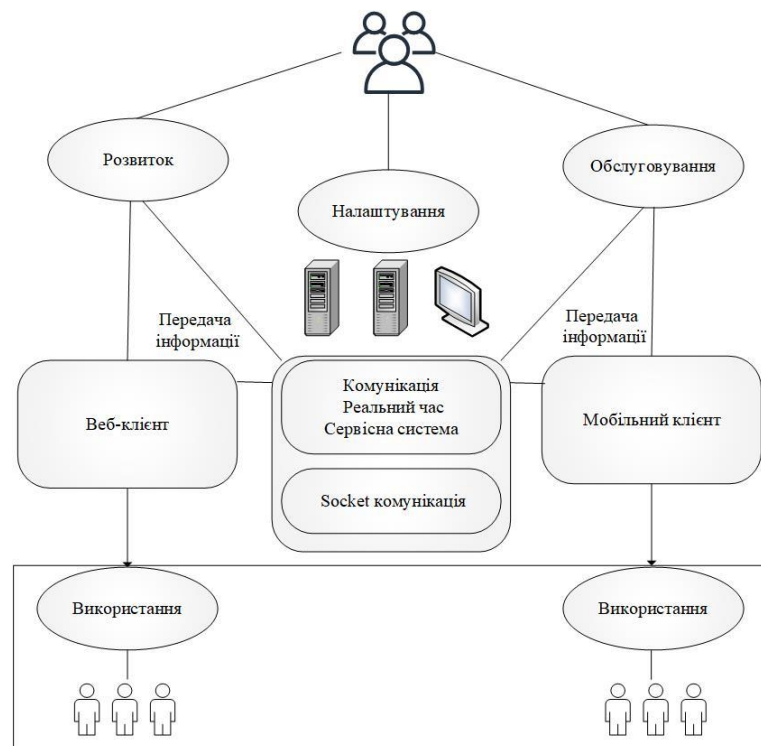


Рис. 6. Загальна структура системи віртуальної лабораторії.

### 5. Висновки і перспективи подальших досліджень.

Механізм зв'язку системи віртуальної лабораторії не відповідає вимогам високої пропускну здатності та двостороннього зв'язку в реальному часі. В даній роботі було використано механізм Socket для реалізації комунікації на основі протоколу TCP/IP. Відповідно до технічних вимог високої пропускну здатності та наднизької затримки мережевої системи віртуального експерименту, запропоновано технологію самонастроювання та оптимального планування комунікаційної мережі на основі алгоритму глибокого

машинного навчання. Запропонована технологія дозволяє прийняти оптимальне рішення щодо планування ресурсів та реалізувати оптимізацію непрямого вибору параметрів системи.

#### Список використаних джерел

1. Budai T., Kuczmann M. Towards a Modern, Integrated Virtual Laboratory. System. Acta Polytechnica Hungarica. 2018. Vol. 15, No. 3, P. 191-204.
2. Zeng W., Hao F.. Study on Virtual Simulation Experiment System of Fire Escape and Emergency Evacuation. Proceedings of the 2021 International Conference on Diversified Education and Social Development (DESD 2021). 4 August 2021. P.68-72.
3. Caminero A. C., Ros S., Hernández R., Robles-Gómez A., Tobarra L. and Granjo P. J. T. Virtual remote laboratories management system (TUTORES): Using cloud computing to acquire University practical skills. IEEE Trans. Learn. Technol. 2016. Vol. 9, No 2, P. 133-145.
4. Hou W., Ning Z., Guo L., Chen Z., Obaidat M. S. Novel framework of risk-aware virtual network embedding in optical data center networks. IEEE Syst. J. 2018. Vol. 12, No 3. P. 2473-2482.
5. Torre L., Guinaldo M., Heradio R., Dormido S. The ball and beam system: A case study of virtual and remote lab enhancement with moodle. IEEE Trans. Ind. Informat. 2015. Vol. 11, No. 4. P. 934-945.
6. Buczak A. L., Guven E. A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun. Surveys Tuts. 2016. Vol. 18, No 2. P. 1153-1176.
7. Sun Y., Babu P., Palomar D. Majorization-minimization algorithms in signal processing communications and machine learning. IEEE Trans. Signal Process. 2016. Vol. 65, No 3. – P. 794-816.
8. Yarkoni T., Westfall J. Choosing prediction over explanation in psychology: Lessons from machine learning. Perspect. Psychol. Sci. 2017. Vol. 12, No. 6. P. 1100-1122.
9. Parish E. J., Duraisamy K. A paradigm for data-driven predictive modeling using field inversion and machine learning. J. Comput. Phys. 2016. Vol. 305. P. 758-774.
10. Fontana F. A., Mäntylä M. V., Zanoni M., Marino A. Comparing and experimenting machine learning techniques for code smell detection. Empirical Softw. Eng., 2016. Vol. 21, No 3, P. 1143-1191.
11. Pasolli E., Truong D. T., Malik F., Waldron L., Segata N. Machine learning meta-analysis of large metagenomic datasets: Tools and biological insights. PLoS Comput. Biol. 2016. Vol. 12, No 7.
12. .Deva Kumari A., Prem Kumar J., Prakash V. S., Divya K. S. Supervised Learning Algorithms: A Comparison. Kristu Jayanti Journal of Computational Sciences (KJCS). 2021. Vol.1, No 1, P. 01-12.
13. Grishkun E., Kravchenko S., Levchenko A., Lysogor Y. Machine learning methods. Znanstvena misel. 2020. Vol. 1, No 39. C.55-58
14. Turhan K., Kayikcioglu T. Implementation of a virtual private network-based laboratory information system serving a rural area in Turkey. Lab. Med. 2006. Vol. 37. P. 527-531.
15. Engelen J.E., Hoos H.H. A survey on semi-supervised learning. Mach Learn. 2020.No109, P. 373–440.
16. Мальцев А. Ю. Обзор принципов глубокого обучения как динамической теории искусственного интеллекта. Вчені записки ТНУ імені В.І. Вернадського. Серія: Технічні науки 2021. Том 32 (71) № 6. С. 97-102.
17. Si H., Sun C., Chen B., Shi L., Qiao H. Analysis of Socket Communication Technology Based on Machine Learning Algorithms Under TCP/IP Protocol in Network Virtual Laboratory System. in IEEE Access. 2019. Vol. 7. P. 80453-80464
18. Phoemphon S., So-In C., Leelathakul N. Fuzzy weighted centroid localization with virtual node approximation in wireless sensor networks. IEEE Internet Things J. 2018. Vol. 5, No 6, P. 4728-4752.
19. Bai J., Wang W., Lu M., Wang H., J. Wang. TD-WS: A threat detection tool of WebSocket and Web storage in HTML5 websites. Secur. Commun. Netw. 2016. Vol. 9, No. 18, P. 5432-5443.

19. Samain J., Carofiglio G., Muscariello L., Papalini M., Sardara M., Tortelli M. Dynamic adaptive video streaming: Towards a systematic comparison of ICN and TCP/IP. *IEEE Trans. Multimedia*. 2017. Vol. 19, No. 10. P. 2166-2181.
20. Wang J.-S., Yang G.-H. Data-driven methods for stealthy attacks on TCP/IP-based networked control systems equipped with attack detectors. *IEEE Trans. Cybern.* 2019. Vol. 49, No 8, P. 3020-3031.
21. Quan L., Xu Z., Li Z. Implementation of hardware TCP/IP stack for DAQ systems with flexible data channel. *Electron. Lett.* 2017. Vol. 53, No. 8. P. 530-532.
22. Gomez-Sacristan A., Sempere-Paya V. M., Rodriguez-Hernandez M. A. Virtual laboratory for QoS study in next-generation networks with metro Ethernet access. *IEEE Trans. Educ.* 2016. Vol. 59, No 3. P. 187-193.

### References:

1. Budai T., Kuczmann M. Towards a Modern, Integrated Virtual Laboratory. *System. Acta Polytechnica Hungarica*. 2018. Vol. 15, No. 3, P. 191-204.
2. Zeng W., Hao F.. Study on Virtual Simulation Experiment System of Fire Escape and Emergency Evacuation. *Proceedings of the 2021 International Conference on Diversified Education and Social Development (DESD 2021)*. 4 August 2021. P.68-72.
3. Caminero A. C., Ros S., Hernández R., Robles-Gómez A., Tobarra L. and Granjo P. J. T. Virtual remote laboratories management system (TUTORES): Using cloud computing to acquire University practical skills. *IEEE Trans. Learn. Technol.* 2016. Vol. 9, No 2, P. 133-145.
4. Hou W., Ning Z., Guo L., Chen Z., Obaidat M. S. Novel framework of risk-aware virtual network embedding in optical data center networks. *IEEE Syst. J.* 2018. Vol. 12, No 3. P. 2473-2482.
5. Torre L., Guinaldo M., Heradio R., Dormido S. The ball and beam system: A case study of virtual and remote lab enhancement with moodle. *IEEE Trans. Ind. Informat.* 2015. Vol. 11, No. 4. P. 934-945.
6. Buczak A. L., Guven E. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Commun. Surveys Tuts.* 2016. Vol. 18, No 2. P. 1153-1176.
7. Sun Y., Babu P., Palomar D. Majorization-minimization algorithms in signal processing communications and machine learning. *IEEE Trans. Signal Process.* 2016. Vol. 65, No 3. – P. 794-816.
8. Yarkoni T., Westfall J. Choosing prediction over explanation in psychology: Lessons from machine learning. *Perspect. Psychol. Sci.* 2017. Vol. 12, No. 6. P. 1100-1122.
9. Parish E. J., Duraisamy K. A paradigm for data-driven predictive modeling using field inversion and machine learning. *J. Comput. Phys.* 2016. Vol. 305. P. 758-774.
10. Fontana F. A., Mäntylä M. V., Zanoni M., Marino A. Comparing and experimenting machine learning techniques for code smell detection. *Empirical Softw. Eng.*, 2016. Vol. 21, No 3, P. 1143-1191.
11. Pasolli E., Truong D. T., Malik F., Waldron L., Segata N. Machine learning meta-analysis of large metagenomic datasets: Tools and biological insights. *PLoS Comput. Biol.* 2016. Vol. 12, No 7.
12. .Deva Kumari A., Prem Kumar J., Prakash V. S., Divya K. S. Supervised Learning Algorithms: A Comparison. *Kristu Jayanti Journal of Computational Sciences (KJCS)*. 2021. Vol.1, No 1, P. 01-12.
13. Grishkun E., Kravchenko S., Levchenko A., Lysogor Y. Machine learning methods. *Znanstvena misel*. 2020. Vol. 1, No 39. C.55-58Turhan K., Kayikcioglu T. Implementation of a virtual private network-based laboratory information system serving a rural area in Turkey. *Lab. Med.* 2006. Vol. 37. P. 527-531.
14. Engelen J.E., Hoos H.H. A survey on semi-supervised learning. *Mach Learn.* 2020.No109, P. 373–440.

15. Maltsev A. Yu. Review of the principles of deep learning as a dynamic theory of artificial intelligence. Academic notes of TNU named after V.I. Vernadskyi. Series: Technical Sciences 2021. Volume 32 (71) No. 6. P. 97-102.
16. Si H., Sun C., Chen B., Shi L., Qiao H. Analysis of Socket Communication Technology Based on Machine Learning Algorithms Under TCP/IP Protocol in Network Virtual Laboratory System. in *IEEE Access*. 2019. Vol. 7. P. 80453-80464
17. Phoemphon S., So-In C., Leelathakul N. Fuzzy weighted centroid localization with virtual node approximation in wireless sensor networks. *IEEE Internet Things J.* 2018. Vol. 5, No 6, P. 4728-4752.
18. Bai J., Wang W., Lu M., Wang H., J. Wang. TD-WS: A threat detection tool of WebSocket and Web storage in HTML5 websites. *Secur. Commun. Netw.* 2016. Vol. 9, No. 18, P. 5432-5443.
19. Samain J., Carofiglio G., Muscariello L., Papalini M., Sardara M., Tortelli M. Dynamic adaptive video streaming: Towards a systematic comparison of ICN and TCP/IP. *IEEE Trans. Multimedia.* 2017. Vol. 19, No. 10. P. 2166-2181.
20. Wang J.-S., Yang G.-H. Data-driven methods for stealthy attacks on TCP/IP-based networked control systems equipped with attack detectors. *IEEE Trans. Cybern.* 2019. Vol. 49, No 8, P. 3020-3031.
21. Quan L., Xu Z., Li Z. Implementation of hardware TCP/IP stack for DAQ systems with flexible data channel. *Electron. Lett.* 2017. Vol. 53, No. 8. P. 530-532.
22. Gomez-Sacristan A., Sempere-Paya V. M., Rodriguez-Hernandez M. A. Virtual laboratory for QoS study in next-generation networks with metro Ethernet access. *IEEE Trans. Educ.* 2016. Vol. 59, No 3. P. 187-193.