

УДК 004

Є.Водзинський, магістр гр. КН-21М-1,4

Центральноукраїнський національний технічний університет

## ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ АДАПТИВНОЇ ОПТИМІЗАЦІЇ МАРШРУТИЗАЦІЇ МЕРЕЖІ ІНФОРМАЦІЙНИХ ТА КОМП'ЮТЕРНИХ СИСТЕМ

У статті розроблено програмне забезпечення, яке призначено для системи адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем. Метою розробки є дослідження та програмна реалізація системи адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем. Об'єктом дослідження є процес адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем. Предметом дослідження є методи адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем. Методи дослідження базуються на методах побудови комп'ютерних мереж, методах математичної статистики, методах розробки програмного забезпечення. Результат роботи – програмна реалізація системи адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем. В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

комп'ютерні науки, маршрутизація, мережі інформаційних та комп'ютерних систем

**Постановка проблеми.** Мережеву маршрутизацію, процес вибору шляхів маршрутизації в мережі зв'язку, можна оптимізувати для зниження витрат, покращення використання мережі та максимізації продуктивності.

У мережах з комутацією пакетів, таких як Інтернет, постачальники засобів зв'язку (включно з операторами зв'язку) мають багато варіантів – часто з кількома переходами та варіантами передачі – щодо того, як пакет переміщається до пункту призначення. У міру надходження пакетів програмне забезпечення мережевої маршрутизації оцінює ці варіанти та приймає рішення в режимі реального часу щодо того, як пакет повинен дістатися до місця призначення.

Мережі динамічні, умови змінюються щосекунди, що вводить новий рівень складності. Система мережевої маршрутизації має оптимізуватися для кожного пакета. Оптимізація може включати вартість, пропускну здатність, угоди про рівень обслуговування, кількість переходів, доступність тощо.

Для вирішення цих складнощів постачальникам послуг потрібен механізм прийняття рішень на основі правил, який базується на високопродуктивній базі даних із низькою затримкою та постійно оновлюється реальними вхідними даними, включаючи доступність мережі, пропускну здатність і вартість. Механізм прийняття рішень розглядає кожен пакет і його призначення, а потім запитує базу даних, щоб застосувати свої правила прийняття рішень для прийняття рішення про маршрутизацію.

Бізнес-вимоги бази даних для мережевої маршрутизації:

– Великі набори даних постійно оновлюються з мережевими вхідними даними, параметрами топології та вартістю.

– Рішення щодо маршрутизації в реальному часі на основі оптимізації механізму правил.

– Постійна доступність, завжди 24x7x365.

– Надзвичайно низька затримка, щоб не перешкоджати роботі мережі.

Система, яка розроблена у даній роботі, є гарним вибором для мережевої маршрутизації, оскільки він забезпечує:

- Передбачувану, високу продуктивність проти надзвичайно великих обсягів транзакцій.
- Провідну в галузі доступність і час безвідмовної роботи (п'ять 9с).
- Масштабування для обробки величезних обсягів даних.
- Значно нижчу ТСО порівняно з іншими технологіями.

**Аналіз останніх досліджень і публікацій.** При аналізі останніх досліджень і публікацій [1-10] було виявлено певні прогалини у забезпеченні системи адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем.

**Мета й завдання дослідження.** Метою роботи є дослідження та програмна реалізація системи адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем.
- Дослідження системи адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем.
- Програмна реалізація системи адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем.

*Об'єктом дослідження* є процес адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем.

*Предметом дослідження* є методи адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем.

*Методи дослідження* базуються на методах побудови комп'ютерних мереж, методах математичної статистики, методах розробки програмного забезпечення.

### **Виклад основного матеріалу**

За останні п'ятнадцять років Інтернет став важливою частиною світової комунікаційної інфраструктури. Інтернет складається з десятків тисяч доменів або автономних систем (AS) – частин інфраструктури, кожною з яких керує окрема установа, наприклад університет, корпорація або постачальник послуг Інтернету (ISP). Повідомлення, надіслане одним комп'ютером, зазвичай проходить через кілька AS, перш ніж досягти пункту призначення, завдяки чому продуктивність зв'язку залежить від потоку трафіку всередині та між AS на шляху. У цій роботі розглянемо роль оптимізації в управлінні маршрутизацією трафіку через Інтернет. Пояснимо, як складність показників продуктивності та протоколів мережевої маршрутизації призводить до проблем оптимізації, які не піддаються аналітичному вирішенню, що змушує використовувати ефективні методи пошуку для дослідження великого простору налаштовуваних параметрів.

Повідомлення, надіслані одним комп'ютером (скажімо, веб-клієнтом) на інший комп'ютер (скажімо, веб-сервер), поділяються на окремі дейтаграми або пакети, які проходять через мережу незалежно. Рішення базувати Інтернет-протокол (IP) на комутації пакетів сягає корінням у перші дні ARPAnet наприкінці 1960-х років [1,2]. На відміну від традиційного підходу комутації каналів телефонної мережі, комутація пакетів визнає, що передача даних часто відбувається з перебоями, коли користувачі та програми чергують періоди високої активності мережі та відносної тиші. Комутація пакетів має привабливість, що дозволяє зв'язкам у мережі мультиплексувати трафік між декількома парами відправників і отримувачів. Поки одна пара відправник-одержувач неактивна, інша може використовувати незатребувану пропускну здатність шляхом обміну трафіком на вищій швидкості.

Однак ця гнучкість має свою ціну. Якщо занадто багато хостів обмінюються пакетами одночасно, сукупний трафік може перевантажити пропускну здатність каналу, що призведе до затримок у передачі даних і, зрештою, до втрати пакетів у разі переповнення буфера, що

живить канал. Таким чином, Інтернет має відносно просту модель обслуговування найкращих зусиль без гарантії, що пакет даних досягне свого кінцевого пункту призначення вчасно.

Хоча багато програм (наприклад, електронна пошта) можуть допускати затримку в отриманні даних, відсутність інформації часто є неприпустимим. Замість створення надійної впорядкованої доставки даних в IP, кінцеві хости несуть відповідальність за повторну передачу втрачених пакетів і реконструкцію впорядкованого потоку даних на одержувачі. Ці функції виконує протокол керування передачею (TCP), реалізований в операційній системі на кінцевих комп'ютерах [1]. Крім того, відправник TCP визначає швидкість передачі даних, відстежуючи успішність (або невдачу) надсилання пакетів одержувачу. Коли пакети втрачаються, хост-відправник зменшує швидкість передачі, щоб допомогти зменшити явне перевантаження; коли пакети успішно доставлені, хост-відправник оптимістично збільшує швидкість надсилання, щоб отримати вигоду від доступної смуги пропускання на шляху до одержувача. Ця децентралізована схема контролю перевантаження забезпечує форму справедливого розподілу пропускну здатності каналів між конкуруючими парами відправників і отримувачів.

Однак транспортні протоколи, такі як TCP, не гарантують ефективну роботу мережі. Наприклад, одне посилення може бути сильно перевантаженим, тоді як інші посилення в мережі залишаються незначно завантаженими. Або виклик VoIP (VoIP) може проходити довгий шлях із високою затримкою розповсюдження, якщо доступний шлях із низькою затримкою. Відповідальність за вибір шляху, яким пакет слідує через мережу, лежить на протоколах маршрутизації, реалізованих окремими маршрутизаторами в мережі. Замість того, щоб використовувати жорстко з'єднані таблиці для пересилання пакетів, маршрутизатори обмінюються один з одним керуючими повідомленнями для розподіленого обчислення шляхів через мережу.

Розподілений підхід дозволяє набору маршрутизаторів автоматично адаптуватися до змін у топології мережі. Це робить IP-мережі надійними за наявності збоїв зв'язку та маршрутизатора, а також легко дозволяє розгортати нове обладнання в міру зростання мережі.

Однак протоколи маршрутизації, які використовуються в більшості IP-мереж, не включають інформацію про мережеве навантаження та продуктивність у вибір шляхів. Надані власним пристроям, маршрутизатори продовжують пересилати пакети через сильно завантажені канали зв'язку, що призводить до втрати пакетів, що змушує кінцеві хости зменшувати швидкість надсилання у відповідь.

Перші спроби розробки протоколів маршрутизації на основі затримки викликали занепокоєння щодо стабільності адаптивної маршрутизації в мережах з комутацією пакетів. При маршрутизації, чутливій до навантаження, пересилання пакетів даних відбуватиметься за такий же малий часовий проміжок, що й доставка інформації про переважне навантаження на окремі канали. Початковий досвід роботи в ARPAnet показав, що дія на основі застарілої інформації про завантаження каналів спонукає маршрутизатори спрямовувати трафік на, здавалося б, недостатньо використовувані канали, що призводить до перевантажень, які зрештою змушують маршрутизатори перемикатися на інші шляхи.

Незважаючи на деякі пропозиції, які намагаються запобігти подібним коливанням [1], звичайні протоколи IP-маршрутизації адаптуються лише до змін у топології мережі та конфігурації маршрутизатора. У той час як дослідження та стандартизація тривали (і все ще тривають) над більш гнучкими протоколами маршрутизації [1,6], практикам потрібен був ефективний спосіб впливати на потік трафіку через їхні мережі. На щастя, протоколи IP-маршрутизації мають різні налаштовуємі параметри, які мережеві оператори можуть налаштувати, щоб змінити шляхи, які маршрутизатори використовують для пересилання пакетів даних. Наприклад, оператор може налаштувати цілочисельні ваги зв'язку, які маршрутизатори використовують для обчислення найкоротших шляхів для передачі трафіку даних.

Однак визначення правильного налаштування цих параметрів у великій мережі є складним для людини-оператора, створення основи для застосування методів оптимізації.

Як альтернатива адаптації протоколів розподіленої маршрутизації до переважного трафіку, мережеві оператори або автоматизовані системи керування можуть змінювати конфігурацію «статичних» параметрів, які керують роботою протоколів маршрутизації [7]. Це призводить до «контура керування», де вимірювання робочої мережі служать вхідними даними для моделі «що, якщо» фіксує наслідки змін.

Вимірювання фіксують запропоноване навантаження на мережу, а також поточну топологію, тоді як модель передбачає результат розподіленого процесу вибору шляху, що виконується на маршрутизаторах для заданого налаштування налаштовуваних параметрів. Це дозволяє використовувати методи оптимізації для визначення налаштувань параметрів, які відповідають цілям продуктивності мережі. Після визначення хороших значень оператор-людина або автоматизована система можуть зв'язатися з маршрутизаторами, щоб змінити конфігурацію налаштовуваних параметрів. Потім маршрутизатори спільно діють на основі нових значень параметрів, щоб обчислити нові шляхи для пересилання пакетів даних через мережу, як передбачено моделлю.

Підхід має кілька переваг перед розширенням протоколів маршрутизації для автоматичної адаптації до трафіку.

По-перше, мережа може продовжувати використовувати звичайні протоколи IP-маршрутизації, а не розгортати або вмикати нові (потенційно складні) протоколи, які автоматично адаптуються до переважного трафіку.

По-друге, мережа уникає коливань протоколу, які можуть виникати, коли маршрутизатори реагують на локально створені (потенційно застарілі) уявлення про переважний трафік. Натомість зміни маршрутизації ретельно плануються на основі точного уявлення про трафік і топологію всієї мережі.

По-третє, маршрутизаторам не потрібно підтримувати статистику щодо навантаження на окремі канали, поширювати цю інформацію по всій мережі або перераховувати шляхи, коли ця інформація змінюється; уникнення цих операцій зменшує пропускну здатність і обчислювальні вимоги до маршрутизаторів.

По-четверте, і, мабуть, найважливіше, вибір параметрів маршрутизації може залежати від широкого спектру обмежень продуктивності та надійності, які може бути важко включити в розподілений протокол маршрутизації.

Ці обмеження та методи оптимізації для їх задоволення можуть продовжувати розвиватися з часом. Незважаючи на ці переконливі переваги, виникла проблема оптимізації важко вирішити через складність базових протоколів маршрутизації та різноманітність цілей мережі.

Звичайні протоколи IP-маршрутизації не були розроблені з урахуванням оптимізації, а оптимізація налаштовуваних параметрів є складною в обчислювальному плані навіть для найпростіших показників (наприклад, таких як мінімізація максимального використання каналу) [1]. Простір параметрів досить великий, і невеликі зміни значення параметра можуть призвести до значних відмінностей у потоці трафіку через мережу.

У ядрі Інтернету зміни маршрутизації в одній AS можуть вплинути на те, як трафік залишає цю мережу та потрапляє в наступну AS на шляху до місця призначення [9]. Це вносить додаткову складність у моделювання наслідків змін налаштовуваних параметрів. Нарешті, проблема оптимізації повинна враховувати численні цілі, такі як навантаження на канал, затримку розповсюдження та кількість переходів. Більш складні показники також беруть участь [10], такі як бажання обмежити загальний трафік, спрямований на кожен сусідню AS, або частоту змін конфігурації, оскільки трафік коливається з часом. У процесі оптимізації може також знадобитися врахувати вплив несправностей обладнання та планового технічного обслуговування на придатність налаштувань параметрів; крім того, кілька маршрутизаторів або каналів можуть вийти з ладу або відновитися разом, якщо вони мають спільні ризики, наприклад загальне джерело живлення або оптичний підсилювач.

Ці практичні проблеми ускладнюють (а часто й унеможливають) отримання аналітичних рішень проблеми оптимізації. Натомість вирішення проблеми оптимізації залежить від наявності ефективних моделей протоколів маршрутизації та ефективних способів дослідження великих частин простору параметрів. Щоб зменшити накладні витрати

на обчислення, моделі «що, якщо» не повинні імітувати детальну роботу протоколів маршрутизації з часом, коли маршрутизатори обмінюються гіпотетичними повідомленнями керування. Замість цього моделі повинні визначати результат протоколів маршрутизації – шляхи, які маршрутизатори в кінцевому підсумку виберуть для передачі трафіку.

Крім того, процес оптимізації не повинен вимагати повторного обчислення цих результатів з нуля для кожного можливого налаштування налаштовуваних параметрів; натомість слід використовувати інкрементні алгоритми, щоб мінімізувати витрати на обчислення для дослідження простору пошуку. Нарешті, експериментальні результати на реальних вхідних даних слід використовувати для керування евристичними методами, які обмежують простір пошуку з точки зору кількості параметрів та діапазону їхніх значень. У наступному розділі ми досліджуємо ці принципи в контексті оптимізації маршрутизації всередині однієї AS на основі звичайних протоколів маршрутизації за найкоротшим шляхом із регульованими вагами країв. Ця проблема широко вивчалася протягом останніх кількох років, і рішення були включені в інструменти управління, які використовуються в багатьох оперативних мережах IP [9,11,12].

Далі ми опишемо, як розширити модель маршрутизації, щоб точно охопити роботу великих магістральних мереж, які мають кілька точок виходу для спрямування трафіку до кожного пункту призначення. Потім ми досліджуємо роль оптимізації в налаштуванні політик міждоменного маршрутизації, які контролюють те, як трафік проходить між AS. У висновку ми коротко обговорюємо останні дослідницькі роботи з розробки протоколів розподіленої маршрутизації, які легше налаштовувати, а також централізованих підходів для прямого обчислення маршрутів від імені оперативних маршрутизаторів.

Багато автономних систем (AS) запускають протоколи маршрутизації, такі як Open Shortest Path First (OSPF) [13] або Intermediate System-Intermediate System (IS-IS) [14], які обчислюють найкоротші шляхи на основі налаштовуваних ваг зв'язку. У цьому розділі ми формулюємо задачу оптимізації для вибору ваг зв'язку на основі топології мережі, матриці трафіку та цільової функції. На перший погляд, налаштування єдиного цілого значення ваги для кожного посилення може здатися недостатньо гнучким, щоб задовольнити різноманітні цілі продуктивності. Тим не менш, експериментальні результати показують, що можливо досягти майже оптимальної маршрутизації для реальних топологій і матриць трафіку. Незважаючи на те, що проблема оптимізації є NP-складною, методи локального пошуку є напроцуд ефективними у пошуку хороших рішень і легко підтримують різноманітні цілі продуктивності та операційні обмеження.

Дослідження різних методів пошуку показали, що можна знайти майже оптимальні налаштування ваг зв'язку на графах із сотнями вузлів за розумний проміжок часу [1,13,14]. Крім того, правильне налаштування вагових коефіцієнтів зв'язку працює майже так само добре, як і теоретична верхня межа розв'язання проблеми потоку кількох товарів, яка передбачає більшу гнучкість у маршрутизації трафіку, ніж дозволяють алгоритми маршрутизації за найкоротшим шляхом. На практиці різниця в продуктивності між хорошим налаштуванням ваги та рішенням для багатокомпонентного потоку особливо мала для реальних мережевих топологій, які зазвичай мають відносно малий діаметр і вузький діапазон пропускної здатності зв'язку. Хоча виконання локального пошуку обчислювально недороге для невеликих графіків, визначення хороших налаштувань ваги у великих графіках може свідчити про альтернативний підхід. Багатообіцяючою альтернативою є двофазний алгоритм, де на першому етапі кожен елемент матриці трафіку розподіляється на один шлях, а на другому етапі намагається знайти налаштування вагових коефіцієнтів зв'язку, які можуть досягти цих шляхів [15].

Використання локального пошуку для оптимізації ваг зв'язку забезпечує величезну гнучкість у вирішенні багатьох практичних обмежень щодо встановлення ваг зв'язку в робочих мережах. Зокрема, мінімізація кількості та частоти змін ваги є дуже важливою, щоб уникнути перебоїв у роботі мережі. Коли вага зв'язку змінюється, нова інформація розповсюджується по всій мережі, і маршрутизатори перераховують свої найкоротші шляхи. Протягом цього періоду конвергенції маршрутизатори в мережі не мають узгодженого представлення найкоротших маршрутів для деяких пунктів призначення; тим часом пакети

можуть бути втрачені або мати тривалі затримки. Хоча нещодавня робота значно скоротила час конвергенції [16], затримки в кілька секунд не є рідкістю. Таким чином, оператори не змінюють ваги каналів, якщо поточна конфігурація маршрутизації не спричиняє проблеми з продуктивністю. Стійкість до варіацій у матриці трафіку:

Налаштування ваг зв'язку не повинно бути чутливим до невеликих варіацій у матриці трафіку  $M$ . Існуючі методи вимірювання або висновку матриці трафіку [15, 17] мають обмеження щодо їх точності, а сама матриця трафіку коливається через деякий час. Найпростішим способом врахування невизначеності було б збільшити елементи  $M_{i,j}$  матриці трафіку на деяку цільову величину.

Загалом, алгоритм локального пошуку міг би оцінити параметр-кандидат для вагових коефіцієнтів посилення на набір матриць трафіку, віддаючи перевагу рішенню, яке добре працює для кожної матриці трафіку, а не тому, яке найкраще працює для однієї за рахунок інших. Цей підхід надзвичайно ефективний для вибору вагових коефіцієнтів зв'язку, які враховують добові цикли трафіку, дозволяючи мережевим операторам мати єдине призначення вагових коефіцієнтів каналів як для денного, так і для нічного трафіку [10]. Фактично, нещодавня робота [17] показує, що можна знайти рішення маршрутизації, які добре працюють незалежно від матриці трафіку або в надзвичайно широкому діапазоні вимог трафіку. Пережиті збої обладнання та планове технічне обслуговування: в ідеалі призначення вагових зв'язків було б стійким до звичайних збоїв у мережі, таких як збої одного з'єднання. У разі збою зв'язку інформація розповсюджується по всій мережі, і маршрутизатори обчислюють нові найкоротші шляхи по краях, що залишилися. Запобігання перевантаженню після збою може вимагати від алгоритму локального пошуку оцінки параметрів кандидата вагових коефіцієнтів зв'язку для кожного сценарію збою [18]. Однак оцінка вагових коефіцієнтів за всіма помилками є непомірною для обчислень у великих мережевих налаштуваннях.

На щастя, можна ідентифікувати та оцінити набагато менший набір критичних сценаріїв [19]. На практиці ваги каналів, які добре працюють у вихідній топології мережі, продовжують працювати добре після більшості збоїв одного каналу. Однак для деяких сценаріїв збою може знадобитися змінити ваги посилення, щоб уникнути перевантаження. На щастя, зміни однієї або двох ваг зв'язку зазвичай достатньо, щоб зменшити перевантаження. Як профілактичний захід, необхідні зміни ваги можуть бути обчислені заздалегідь до будь-якого збою з'єднання та збережені системою керування мережею. Ці самі попередньо обчислені зміни ваги надзвичайно корисні, коли мережеві оператори повинні навмисно «вивести з ладу» канал, щоб виправити або оновити обладнання; у цьому випадку зміни ваги можна внести заздалегідь, до відключення обладнання для обслуговування.

Підтримка різноманітних обмежень щодо налаштувань шляхів: коли між двома маршрутизаторами існує кілька найкоротших шляхів, маршрутизатори вздовж шляхів розподіляють трафік між кількома вихідними посиленнями – практика, відома як рівноцінна багатошляхова маршрутизація. Замість того, щоб чергувати ці посилення на рівні пакетів, маршрутизатори зазвичай намагаються пересилати пакети для тієї самої пари джерело-одержувач по одному шляху; це зменшує ймовірність того, що пакети з того самого ТСП-з'єднання надходять до одержувача не в порядку. Балансування навантаження зазвичай досягається шляхом виконання геш-функції для IP-адрес джерела та призначення кожного пакета; значення геш-функції визначає, яке вихідне посилення має нести пакет. Як наслідок, зв'язки вносять невизначеність у точний розподіл трафіку за посиленнями; з двома вихідними посиленнями трафік не обов'язково ділиться точно навпіл. Локальний пошук може легко пояснити вплив невизначеності 10, штрафуючи рішення, які мають зв'язки (наприклад, встановивши  $P_{i,j}$  на 0,6, а не на 0, 1, неявно припускаючи, що обидва канали повинні нести 60% вихідного трафіку).

З іншого боку, наявність кількох найкоротших шляхів є перевагою в деяких налаштуваннях. Зокрема, якщо одне з двох з'єднань вийде з ладу, інший найкоротший шлях усе ще залишається, дозволяючи мережі об'єднуватися швидше, спричиняючи менше втрат пакетів і затримок [10]. Алгоритм локального пошуку може легко схилитися до рішень із більшою кількістю зв'язків, включивши кількість зв'язків у цільову функцію. Загалом,

використання локального пошуку дозволяє самій цільовій функції змінюватися з часом, оскільки важливі різні показники. Наприклад, для найкращого Інтернет-трафіку мінімізація максимального використання каналу (або суми  $f()$  для всіх каналів) є дуже ефективним способом максимізації пропускної здатності TCP. Однак ситуація зовсім інша, коли на сцену з'являються такі інтерактивні програми, як Voice-over-IP (VoIP) і відеоігри. Для цих програм важливо підтримувати низьку затримку розповсюдження (наприклад, нижче 100 мс), а збоїв під час конвергенції протоколу маршрутизації слід уникати шляхом зменшення частоти змін маршрутизації та вибору рішень, які мають кілька найкоротших шляхів. Базовий механізм локального пошуку є надзвичайно гнучким для включення складних показників і зміни їх важливості в оцінці налаштувань кандидатів на ваги послань.

Переважає більшість Інтернет-трафіку має пройти кілька автономних систем (AS) на шляху до місця призначення. У той час як попередній розділ зосереджувався на одній мережі окремо, у цьому розділі розглядаються додаткові проблеми, з якими стикаються транзитні AS (наприклад, великі Інтернет-провайдери), які підключаються до інших мереж Інтернету. Оптимізація вагових коефіцієнтів зв'язку в цих мережах потребує складнішої моделі, яка фіксує вплив маршрутизації з раннім виходом, а також показників оптимізації, які віддають перевагу рішенням, які уникають переміщення трафіку з однієї точки виходу в іншу.

#### **Пункти призначення з кількома точками виходу**

Хоча багато мереж здебільшого автономні, постачальники послуг транзиту підключаються до багатьох інших мереж у різних географічних точках. Наприклад, AS A дозволяє клієнтській мережі (наприклад, компанії або університетському містечку), підключеній до маршрутизатора  $i$ , досягати пунктів призначення (наприклад, веб-серверів) в AS B і C.

Великі транзитні AS часто взаємодіють один з одним у декілька місць для підвищення надійності та продуктивності AS. Дедалі частіше клієнти підключаються до своїх постачальників у кількох місцях для підвищення надійності. Кожне з'єднання складається з двох маршрутизаторів – по одному в кожному домені – які обмінюються інформацією про доступність за допомогою протоколу Border Gateway Protocol (BGP). По суті, BGP – це клей, який утримує разом різні частини Інтернету. Наприклад, маршрутизатор в AS B оголошуватиме шлях призначення dB до сусіднього маршрутизатора в AS A. Оскільки дві мережі є одноранговими у двох місцях, маршрутизатори в AS A мають дві можливі точки виходу (посилання на AS B у маршрутизаторах  $j$  і  $k$ ) для спрямування трафіку до місця призначення dB. На високому рівні два міждоменні шляхи здебільшого еквівалентні, наприклад, обидва шляхи проходять однакову кількість AS ( $i$ , фактично, ту саму AS) на шляху до місця призначення. Коли маршрутизатор має два «однаково хороші» міждоменні маршрути до пункту призначення, рішення щодо маршрутизації BGP залежить від вартості внутрішньодоменового шляху до кожного вихідного маршрутизатора. Наприклад, маршрутизатор  $i$  направляв би трафік, призначений до dB, через маршрутизатор  $j$ , оскільки внутрішньодоменний шлях вартості 10 до маршрутизатора  $j$  коротший, ніж шлях вартості 20 до маршрутизатора  $k$ . Загальна практика раннього виходу або гарячої маршрутизації намагається звести до мінімуму використання внутрішніх мережевих ресурсів, перекидаючи трафік до сусідньої AS якомога раніше. Насправді AS B буде виконувати маршрутизацію раннього виходу у зворотному напрямку для трафіку від dB до клієнта в мережі ASA, що призведе до асиметричної маршрутизації, коли трафік від клієнта залишає AS A через маршрутизатор  $j$ , а трафік до клієнта входить через маршрутизатор  $k$ . Враховуючи велику кількість пунктів призначення, розповсюджених по всьому Інтернету, транзитній AS доведеться вибирати між кількома точками виходу для значної частини пунктів призначення, що робить вкрай важливим фіксувати вплив маршрутизації раннього виходу на потік трафіку.

Потік трафіку через транзитну мережу залежить від маршрутів BGP, рекламаних сусідніми AS, а також від локальних політик, налаштованих на окремих маршрутизаторах. У цьому розділі ми розглянемо, як налаштування політик BGP впливає на пересилання трафіку даних. Далі ми опишемо, як розширити модель маршрутизації з попереднього розділу, щоб відобразити роль політик маршрутизації. Потім ми обговорюємо підходи до дослідження

дуже великого простору пошуку конфігурацій політики BGP, а також фундаментальні обмеження на здатність моделей передбачати навантаження на посилання в AS.

### **Політики BGP, що впливають на процес вибору шляху**

У найпростішому випадку маршрутизатори в AS вибирають маршрути BGP із найкоротшим шляхом AS, розриваючи зв'язки на основі близькості точок виходу. Загалом, маршрутизатор можна налаштувати для застосування політики, яка призначає локальні переваги маршруту, щоб вибрати один маршрут замість іншого з коротшим шляхом AS або ближчою точкою виходу. Сучасні маршрутизатори забезпечують надзвичайно гнучку «мову програмування» для вказівки правил для призначення атрибута локальних переваг. Наприклад, політики можуть розрізняти маршрути, отримані від різних сусідніх AS, на основі комерційних відносин. У транзитних мережах загальноприйнятою практикою є надання більшої переваги маршрутам BGP, отриманим від клієнтів, ніж маршрутам, отриманим від постачальників вищестоящего зв'язку, щоб гарантувати, що трафік даних проходить через сусідів, які платять клієнтам, навіть якщо шлях через постачальника коротший [14, 15]. Подібним чином мережа може призначити нижчу локальну перевагу маршрутам BGP, отриманим через канали з низькою пропускною здатністю, які існують лише для надання резервної служби [16], щоб гарантувати, що трафік даних перетинає високу пропускну здатність. основні ланки, за винятком випадків, коли вони вийшли з ладу.

На додаток до призначення переваг на основі відносин із сусідньою AS, оператори конфігурують політики, щоб впливати на навантаження на мережеві з'єднання [12, 17, 18]. Наприклад, припустімо, що AS вивчає маршрути BGP до пункту призначення від двох постачальників вищестоящего зв'язку. Призначаючи нижчі локальні переваги для одного маршруту, AS вирішує спрямувати трафік через маршрут, отриманий від іншого постачальника. Ретельне призначення локальних переваг над діапазоном пунктів призначення допомагає збалансувати навантаження на два посилання. У деяких випадках один провайдер може стягувати вищу ціну за пересилання трафіку або загалом мати нижчу продуктивність. Щоб зменшити фінансові витрати або підвищити продуктивність, AS може віддавати перевагу маршрутам через іншого провайдера, аж до моменту, коли канал стає занадто сильно завантаженим. Загалом, AS може підключатися до однієї AS у кількох географічних місцях. Якщо одне з посилань до сусідньої AS стає перевантаженим, оператори Provider local-pref 90 первинний шлях Transit AS local-pref 100 Customer вторинний шлях d можуть змінити конфігурацію сусіднього маршрутизатора, щоб призначити нижче локальне значення переваги деяким із отриманих маршрутів BGP. на цьому місці. Це гарантує, що маршрутизатори в AS спрямовують трафік для цих пунктів призначення через інші, малозавантажені посилання до тієї ж AS.

### **Моделювання впливу на потік трафіку**

Щоб відобразити вплив політики маршрутизації, потрібні розширення моделі, представленої раніше [19]. Змінюючи налаштування атрибута local-preference, налаштування політик BGP впливає на те, як маршрутизатор на периферії мережі вибирає найкращий маршрут. Це, у свою чергу, впливає на вихідний набір для кожного блоку адреси призначення (або префікса). Моделювання впливу політик BGP на потік трафіку вимагає:

– Маршрутів, отриманих BGP від сусідніх AS: вхідними даними для обчислення є набір маршрутів, отриманих BGP від сусідніх AS. На практиці маршрути, отримані за допомогою BGP, можна зафіксувати шляхом «скидання» таблиці маршрутизації BGP на кожному маршрутизаторі або моніторингу маршрутів BGP, коли вони оголошуються сусідами.

– Специфікація політик маршрутизації: політика маршрутизації BGP – це послідовність пунктів, де кожен пункт визначає набір маршрутів (наприклад, на основі префікса призначення або елементів у шляху AS) і призначає значення локального параметра для відповідності маршрути. Наприклад, політика може призначити локальне налаштування 100 для всіх маршрутів з AS «1234» як другий стрибок і 90 для всіх інших.

– Модель вибору шляху BGP: обчислення вихідного набору вимагає застосування політик до маршрутів BGP, отриманих із сусідніх запитів, а потім вибору «найкращого» зі змінених шляхів. Зокрема, модель повинна вибирати маршрути з найвищими локальними

перевагами та, серед них, маршрути з найкоротшою довжиною шляху AS. Зрештою, кожен маршрутизатор вибере маршрут із «найближчою» точкою виходу.

– Пропонований трафік від точок входу до пункту призначення: оцінка потоку трафіку вимагає накладення запитів трафіку  $v_i, d$  поверх шляхів через AS для досягнення вибраної точки виходу для кожної точки входу та префікса призначення. Маючи точне уявлення про визначені BGP маршрути та вимоги до трафіку, інструмент оптимізації може проводити локальний пошук можливих змін у політиках маршрутизації та оцінювати вплив на потік трафіку через мережу.

Обмеження простору пошуку Значна гнучкість у визначенні політики маршрутизації BGP у поєднанні з великою кількістю префіксів призначення робить простір пошуку надзвичайно великим – надто великим для повного вивчення. На щастя, евристика корисна для обмеження накладних витрат процесу оптимізації [12]:

– Дослідження поступових змін політики: локальний пошук може досліджувати поступові зміни конфігурації маршрутизатора, а не розглядати всі можливі політики з нуля. Наприклад, оптимізація може, помітивши, що одне межове з'єднання перевантажене, зосередитися на зміні політики в цьому одному місці, щоб призначити менші локальні переваги маршрутам BGP для деяких пунктів призначення. Цей підхід не тільки обмежує простір пошуку, але й зменшує накладні витрати на оцінку наслідків зміни конфігурації, оскільки інструменту оптимізації потрібно враховувати лише вимоги трафіку, які переміщуються до нових точок виходу.

– Зосередження лише на популярних префіксах призначення: типова магістральна мережа IP має маршрути BGP для більш ніж 150 000 блоків адрес призначення. На практиці переважна більшість цих напрямків сприяють отриманню лише невеликої частини трафіку [10]. Зміна значень локальних параметрів для невеликої кількості популярних префіксів місць призначення перемістить значну кількість трафіку з однієї точки виходу в іншу, мінімізуючи при цьому кількість змін маршруту та зменшуючи накладні витрати на локальний пошук. Крім того, ці популярні напрямки, як правило, мають більш стабільні кандидати BGP 17AS C AS A інший провайдер AS B після після нижнього сусіда до попереднього маршрути [11], що робить імовірним те, що нова конфігурація політики BGP буде ефективною в майбутньому.

– Зосередження на групах блоків адрес призначення: на практиці мережа має однакові (або схожі) параметри маршрутизації BGP для кількох префіксів призначення. Наприклад, великий клієнт може мати дюжину блоків адрес, які рекламуються точно так само, в тих самих місцях мережі. Визначення політик маршрутизації, які збігаються в групі пунктів призначення (скажімо, шляхом зіставлення загальних аспектів їхніх ASpaths), зменшує простір пошуку, а також робить маршрутизацію в мережі менш чутливою до невеликих змін в інших атрибутах маршруту. Крім того, групи пунктів призначення, як правило, мають більш стабільний сукупний обсяг трафіку, що робить прогнози навантаження трафіку в мережі більш точними.

Поєднання цих трьох методів суттєво зменшує накладні витрати на дослідження простору пошуку та, як правило, надає перевагу надійним рішенням. Однак у деяких випадках вплив зміни політики BGP може бути важко передбачити через побічні ефекти на сусідніх доменах [12].

У прикладі AS A і B рекламують шляхи до пунктів призначення в AS C. Спочатку існує п'ять «найкращих» маршрутів – два через AS A і три через AS B. Маршрутизатори на маршруті західного узбережжя через AS A і маршрутизатори на маршруті східного узбережжя через AS B. Припустимо, що крайнє ліве з'єднання до AS B іє перевантажене (як показано пунктирною лінією), і політику BGP для цієї точки виходу змінено 18, щоб призначити нижчу локальну перевагу маршрутам, що походять від AS C

Після цієї зміни деякі маршрутизатори можуть перемикатися з маршруту через крайнє ліве посилення до AS B на маршрут через крайнє праве посилення до AS A.

Ці маршрутизатори будуть рекламувати новий найкращий шлях до сусідніх мереж. Залежно від політики маршрутизації сусіда, нове оголошення може змусити сусіда вибрати іншу наступну AS (тобто іншого провайдера) для досягнення цього префікса. Це може

привести до непередбачуваного зменшення обсягу трафіку, що надходить у домен через цей маршрутизатор. Подібним чином зміна маршруту може спровокувати збільшення трафіку, якщо інші сусіди віддадуть перевагу маршруту (A,C) над маршрутом (B,C).

Такі побічні ефекти ускладнюють прогнозування того, як зміна політики маршрутизації вплине на обсяг трафіку в мережевих посиланнях. Щоб запобігти поширенню змін маршрутизації на сусідні домени, інструмент оптимізації повинен, коли це можливо, досягати цілей балансування навантаження шляхом коригування політики маршрутизації для пунктів призначення, для яких кожен потенційно найкращий маршрут має однаковий шлях AS. На практиці AS матиме багато таких адресатів. Наприклад, пункт призначення, який знаходиться в AS B, буде доступним через три точки виходу до ASB, які пропонують той самий шлях ASpath з одним стрибком.

Зміна політики маршрутизації для одного (або кількох) із цих префіксів призначення перемістить трафік від пунктирного посилання до одного з інших посилань до AS B, не змінюючи шляху AS, який бачить сусідній вузол. Залежно від реалізації BGP подальша AS може навіть не отримати нове оголошення BGP, оскільки шлях AS не змінився. Це дозволяє мережі зменшувати перевантаження на каналі, переміщуючи трафік на інші, малозавантажені канали, не змінюючи інформацію про маршрутизацію BGP, що надсилається до сусідніх domenів.

### **Алгоритм ARTCP**

Запропонований у даній роботі, удосконалений протокол Adaptive Rate Transmission Control Protocol (ARTCP) системи адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем, позичає деякі механізми від протоколу TCP. В ARTCP повністю переглянутий алгоритм керування потоком, що і відрізняє його від TCP. Разом з тим, запропонований протокол може забезпечувати сумісність із TCP.

Протокол ARTCP, запропонований у цій роботі, здатний працювати більш ефективно і якісно, ніж TCP, однак можна виділити кілька напрямків подальших досліджень нового протоколу, які можуть, по-перше, дати можливість ефективно використовувати його в асиметричних системах, а по-друге, досягти рівноправності між потоками з різною довжиною маршруту.

### **Асиметричні системи**

Оскільки в протоколі ARTCP усунута ACK-синхронізація, властива TCP, то відправлення сегментів відбувається незалежно від прибуття підтверджень аж до вичерпання максимального вікна, то на відміну від TCP, ARTCP може бути вдосконалений так, щоб ефективно працювати в системах з асиметричними каналами.

Для використання ARTCP у таких системах необхідно зменшити частоту підтверджень. Оскільки штучна затримка підтверджень викличе збільшення затримки в петлі зворотного зв'язку, то, вимір затримки передачі сегментів потрібно також зв'язати з одержувачем. Оскільки важко домогтися гарної синхронізації системних годин одержувача й відправника, то одержувач може лише помічати зміну часу передачі сегментів, якщо відправник використовує стандартне поле часової мітки [6]. Якщо різниця значень мітки в потоці й системних годинах одержувача змінюється, значить змінюється й абсолютне значення затримки. У цьому випадку одержувач повинен збільшити частоту підтверджень, щоб відправник міг зреагувати на зміну навантаження в мережі. Коли значення швидкості прибуття потоку й затримки передачі не міняються, частота підтверджень може бути знову зменшена.

Алгоритм ARTCP не містить перешкод для цього вдосконалення, крім того, модифікований таким способом ARTCP для асиметричних каналів збереже сумісність із представленим тут протоколом.

### **З'єднання з різними RTT**

Для з'єднань, що володіють різним часом RTT, через розходження довжин їхніх маршрутів, середнє значення коефіцієнта F обмежується деяким числом, меншим 1, як показано в роботах [10, 11, 12, 13]. Це вірно як для ARTCP, так і для TCP. Щоб досягти рівноправності поділу ПрЗд між ARTCP потоками з різною довжиною маршруту, необхідно

усунути залежність коефіцієнта *speedup* від мінімального часу RTT. Цього результату можна досягти шляхом модифікації алгоритму адаптації таким чином, щоб у режимі **FT** повністю відмовитися від використання обмірюваного RTT як індикатора перевантаження, використовуючи лише значення міжсегментних інтервалів потоку.

#### **Імітаційна модель**

Для дослідження можливостей протоколу й відпрацювання його механізмів була розроблена програмна імітаційна модель самого протоколу й мережних компонентів, у середовищі яких повинен функціонувати протокол ARTCP. Модель являє собою набір компонентів реальні об'єкти, що імітують, у складі мережі й об'єкти протоколів ARTCP і CBR (Constant Bit Rate). Досліджувана мережа може мати топологічну схему досить великої складності, що будується з необхідного числа екземплярів наявних класів.

Основними способами що дозволяють досліджувати й верифікувати складні мережні протоколи є стенди для тестування (експериментальні мережі), програмні емулятори й системи автоматизованої верифікації. Верифікацію набору процедурних правил протоколу можна здійснювати за допомогою програмного перебору станів кінцевого автомата представляючого протокол [16, 17, 18], наприклад, в інтерпретаторі мови PROMELA, такому як SPIN [1]. Однак верифікація набору процедурних правил, і вивчення ефективності протоколу переслідують різні цілі й, відповідно, для них застосовуються різні інструменти. Верифікація протоколу означає застосування до його набору процедурних правил формального методу, що дозволяє довести, що цей набір або моделююча його система кінцевих автоматів (з обміном даними) повна, не містить недосяжних станів, вільна від статичних і динамічних блокувань. Верифікація протоколу не ставить задачею навіть визначення кількісних характеристик його ефективності, з погляду верифікації важливо лише те, що прогресивний обмін даними взагалі відбувається. Внаслідок цього методи верифікації побудовані на повному або частковому аналізі доступних станів КА, що представляє протокол. Для систем з великим числом станів ( $> 10^5$ ) верифікація заснована на повному аналізі утруднена на практиці.

Моделювання протоколу не дає гарантії повного аналізу досяжних станів, але зате дозволяє досліджувати кількісні характеристики системи. Разом з тим моделювання дозволяє зробити статистично обґрунтований висновок про надійність протоколу, принаймні, щодо відсутності в ньому блокувань. Складність імітаційного моделювання в тому, що крім реалізації процедурних правил самого досліджуваного протоколу до складу моделі повинні входити всі його компоненти: середовище функціонування, словник і способи кодування повідомлень, модель сервісу протоколу. Однак при цьому моделювання вимагає все-таки менших витрат, ніж розгортання експериментальної мережі для дослідження властивостей протоколу. У цьому випадку наше завдання полягає не у верифікації протоколу ARTCP, а у визначенні чисельних значень його характеристик у різних умовах.

#### **Розробка структурної схеми**

У протоколі ARTCP повністю перероблені всі механізми керування потоком.

Механізм корекції помилок передачі в ARTCP не впливає на швидкість передачі. Від TCP збережені віконний механізм для керування завантаженням одержувача, алгоритми визначення RTT і установки таймера ТПП. Ознакою втрати сегмента служить спрацювання ТПП або прихід двох послідовних підтверджень одного сегмента. Алгоритм керування швидкістю містить у собі: функції диспетчеризації сегментів, виміру швидкості й адаптації швидкості (рисунок 1).

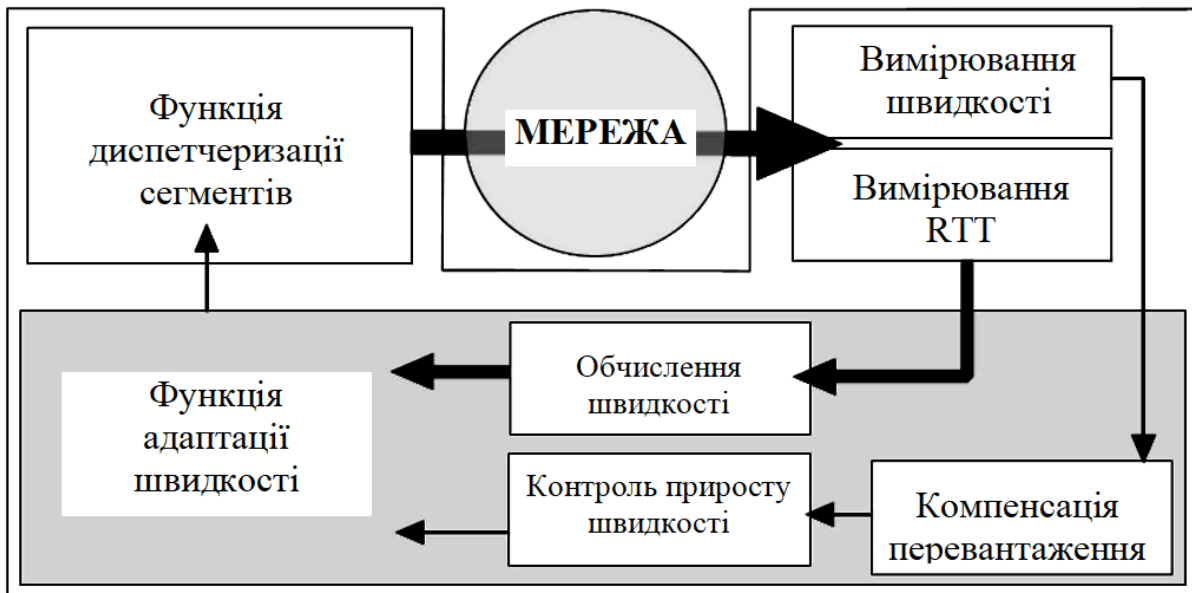


Рисунок 1 – Структурна схема системи

**Висновки.** У статті наведені теоретичне узагальнення й рішення наукового завдання дослідження методів адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем. Рішення даного завдання полягало у вирішенні наступних задач: Був проведений огляд існуючих систем адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем; Досліджена система адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем; На основі отриманих результатів досліджень створена програмна реалізація системи адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем; Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання адаптивної оптимізації маршрутизації мережі інформаційних та комп'ютерних систем. Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

## Список літератури

1. Smirnov, O., Neskorodieva, T., Fedorov, E., Rudakov, K., Neskorodieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» CEUR Workshop Proceedings, Volume 3187, 2022, pp. 1-12. (Scopus).
2. Smirnov O., Smirnova T., Anas M. Al-Oraiqat, Drieiev O., Polishchuk L., Sheroz Khan, Yassin M. Y. Hasan, Aladdein M. Amro, Hazim S. AlRawashdeh «Method for Determining Treated Metal Surface Quality Using Computer Vision Technology». Sensors (Basel, Switzerland) Volume 22, Issue 16, 6223, 2022. (Scopus).
3. Smirnov, O., Lakhno, V., Akhmetov, B., Chubaievskiy, V., Khorolska, K., Bebeshko, B. «Selection of a Rational Composition of Information Protection Means Using a Genetic Algorithm». In: Rajakumar, G., Du, K.L., Vuppalapati, C., Beligiannis, G.N. (eds) Intelligent Communication Technologies and Virtual Mobile Networks. Lecture Notes on Data Engineering and Communications Technologies, vol 131. 2023. Springer, Singapore. pp. 21-34. (Scopus).
4. Smirnov O.A., Al-Oraiqat A.M., Ulichev O.S., Meleshko Ye.V., Al-Rawashdeh H.S., Polishchuk L.I. «Modeling strategies for information influence dissemination in social networks». Journal of Ambient Intelligence and Humanized Computing Volume 13, Issue 5. Springer, Cham. 2022, pp. 2463-2477. (Scopus).
5. Smirnov O., Kuznetsov A., Kryvinska N., Kiian A., Kuznetsova K. «Full Non-Binary Constant-Weight Codes». SN Computer Science, Vol 2, 137, 2021. <https://doi.org/10.1007/s42979-021-00739-w> (Scopus).
6. Smirnov O., Kovalenko O., Kovalenko A., Kavun S. «Quantitative Risk Assessment Method Development in the Context of the SDLC-model». 2021 IEEE 8th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), 2021, pp. 203-208, doi: 10.1109/PICST54195.2021.9772143 (Scopus).
7. Smirnov O., Neskorodieva T., Fedorov E., Rymar P. «Neural Network Modeling Method of Transformations Data

- of Audit Production with Returnable Waste». CEUR Workshop Proceedings Volume 3101, 2021, Pages 192-207. (Scopus).
8. Smirnov O., Kuznetsov A., Kiiian A., Kuznetsova K. «Data hiding scheme based on spread sequence addressing». CEUR Workshop Proceedings Volume 2805, 2020, Pages 44-58. (Scopus).
  9. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-feedback shift register». International Journal of Computing; 2020, Volume 19, Issue 2 – Research Institute for Intelligent Computer Systems – 2020. – P. 247-256. (Scopus).
  10. Smirnov O., Kuznetsov A., Kiiian A., Kuznetsova T. «Non-binary constant weight coding technique». CEUR Workshop Proceedings. Volume 2740, 2020, Pages 102-114. (Scopus).
  11. Smirnov O.A., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». Journal of theoretical and applied information technology Vol.98. No 21, 2020, P. 3334-3346. (Scopus).
  12. Smirnov O., Kuznetsov A., Arischenko A., Chepurko I., Onikiychuk A., Kuznetsova T. «Pseudorandom sequences for spread spectrum image steganography». CEUR Workshop Proceedings Volume 2654, 2020, Pages 122-131. (Scopus).
  13. О.А. Смірнов, П.С. Усік, «дослідження перспектив використання технологічних рішень в мережах 5g» у Кібербезпека та інформаційні технології: монографія. – Х. : ТОВ «ДІСА ПЛЮС», 2020.С. 122-135.
  14. Смірнов О.А., Дреєва Г.М., «Метод генерування фрактального трафіку за допомогою моделі генератора на графі» у Інформаційна безпека та інформаційні технології: монографія / за заг. ред. В. С. Пономаренка. – Х. : Вид. Рожко С.Г. 2019. С. 123-139.
  15. Смирнов А.А., Коваленко А.В. Комплекс математических моделей технологии тестирования web-приложений. Информационные технологии: современный стан та перспективи: монографія / За загальною редакцією В.С. Пономаренка. – Х.: ТОВ «ДІСА ПЛЮС», 2018. – 461 с.
  16. Смирнов А.А., Коваленко А.В. Разработка метода управления рисками разработки программного обеспечения. Информационные технологии: проблемы та перспективи: монографія / За загальною редакцією В.С. Пономаренка. – Х.: Видавець Рожко С.Г., 2017. – 447 с.
  17. Смірнов О.А., Смірнова Т.В., Буравченко К.О., Кравченко С.С., Горбов В.О., «Хмарна система підтримки прийняття рішень технологічного процесу відновлення поверхонь конструкцій і деталей машин». Сучасні інформаційні системи. 2021. Т. 5, № 4. С. 79-95
  18. Смірнов, О.А., Полігенько О.О., Одарченко Р.С., Терещенко Л.Ю.Усік П.С., «Інформаційна технологія та програмне забезпечення для підвищення ефективності планування підсистеми базових станцій стільникового зв'язку». Проблеми телекомунікацій. № 1(26). С. 83-96. 2020.
  19. Смірнов О.А., Усік П.С., Миронець І.В., Буравченко К.О., Якименко Н.М. «Метод підвищення ефективності розподіленої обробки даних у комп'ютерних системах операторів стільникового зв'язку» Вісник Черкаського державного технологічного університету. Технічні науки. №4. С. 103-110. 2020.
  20. О.А.Смірнов, Т.В.Смірнова, Л.І. Поліщук, К.О. Буравченко, А.О.Макевнін, «Дослідження хмарних технологій як сервісів», Кібербезпека: освіта, наука, техніка. № 3(7). С. 43-62. 2020.