

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістр

(назва освітнього ступеня)

на тему: **Компонентне прогнозування навантаження у багатокористувацьких комп'ютерних мережах**

Виконав: студент II курсу, групи РАм-61
спеціальності 172 Телекомунікації та радіотехніка

(шифр і назва спеціальності)

	<u>Земледух В.А.</u> (підпис) (прізвище та ініціали)
Керівник	<u>Хвостівська Л.В.</u> (підпис) (прізвище та ініціали)
Нормоконтроль	<u>Хвостівська Л.В.</u> (підпис) (прізвище та ініціали)
Завідувач кафедри	<u>Дунець В.Л.</u> (підпис) (прізвище та ініціали)
Рецензент	<u>Дедів Л.Є.</u> (підпис) (прізвище та ініціали)

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет прикладних інформаційних технологій та електроінженерії
(повна назва факультету)

Кафедра радіотехнічних систем
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Дунець В.Л.
(підпис) (прізвище та ініціали)

« 26 » листопада 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня магістр
(назва освітнього ступеня)

за спеціальністю 172 Телекомунікації та радіотехніка
(шифр і назва спеціальності)

студенту Земледух Владислав Андрійович
(прізвище, ім'я, по батькові)

1. Тема роботи Компонентне прогнозування навантаження у багатокористувацьких комп'ютерних мережах

Керівник роботи Хвостівська Лілія Володимирівна, к.т.н., доц.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 26 » листопада 2025 року № 4/7-1038 .

2. Термін подання студентом завершеної роботи

3. Вихідні дані до роботи : Об'єкт дослідження: процес зміни навантаження у багатокористувацьких комп'ютерних мережах. Предмет дослідження: методи та алгоритми компонентного прогнозування часових рядів навантаження, що базуються на сингулярному спектральному аналізі та авторегресії.

4. Зміст роботи (перелік питань, які потрібно розробити)

РОЗДІЛ 1. АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ПРОГНОЗУВАННЯ НАВАНТАЖЕННЯ У БАГАТОКОРИСТУВАЦЬКИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ

РОЗДІЛ 2. МЕТОД ТА АЛГОРИТМ ПРОГНОЗУВАННЯ НАВАНТАЖЕННЯ У БАГАТОКОРИСТУВАЦЬКИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ

РОЗДІЛ 3. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КОМПОНЕНТНОГО ПРОГНОЗУВАННЯ НАВАНТАЖЕННЯ У БАГАТОКОРИСТУВАЦЬКИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

1. Актуальність роботи

2. Метод прогнозування навантаження у багатокористувацьких комп'ютерних мережах

3. Алгоритм прогнозування навантаження у багатокористувацьких комп'ютерних мережах

4. Програмне забезпечення прогнозування навантаження у багатокористувацьких комп'ютерних мережах

5. Результати прогнозування навантаження у багатокористувацьких комп'ютерних мережах

6. Наукова новизна

7. Публікації за напрямом дослідження

8. Загальні висновки

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці та	Окіпний І.Б., зав. каф. МТ		
безпека в надзвичайних	Стручок В.С., ст. викл. каф. ОХ		
ситуаціях			

7. Дата видачі завдання 03.09.2025 р.

КАЛЕНДАРНИЙ ПЛАН

[illegible]

Студент

(підпис)

Земледух В.А.

(прізвище та ініціали)

Керівник роботи

(підпис)

Хвостівська Л.В.

(прізвище та ініціали)

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Компонентне прогнозування навантаження у багатокористувацьких комп'ютерних мережах» // Кваліфікаційна робота // Земледух Владислав Андрійович // ТНТУ, факультет прикладних інформаційних технологій та електроінженерії, група РАМ-61 // Тернопіль, 2025 // с. – 83, рис. – 8, табл. – 1, додат. – 2, бібліогр. – 21.

Ключові слова: мережеве навантаження, прогнозування, сингулярний спектральний аналіз, SSA, авторегресійна модель, сезонна регресія, часові ряди, багатокористувацькі мережі.

У роботі досліджено задачу прогнозування навантаження у багатокористувацьких комп'ютерних мережах, що характеризуються наявністю трендових, сезонних і стохастичних складових трафіку. Проаналізовано сучасні статистичні, машинні та гібридні методи прогнозування та визначено їхні обмеження. Запропоновано метод компонентного прогнозування, який поєднує сингулярний спектральний аналіз (SSA) для декомпозиції часових рядів із авторегресійними та сезонно-регресійними моделями для передбачення окремих компонент. Розроблено алгоритм автоматичної класифікації компонент на трендові, сезонні та шумові за спектральними характеристиками та метод оцінювання невизначеності прогнозу. У середовищі Matlab реалізовано програмне забезпечення компонентного прогнозування, що забезпечує реконструкцію сигналу, побудову моделей, обчислення прогнозів і формування довірчих інтервалів. Проведено експериментальне дослідження, яке підтвердило підвищення точності прогнозу порівняно з базовими методами. Результати роботи можуть бути застосовані у системах моніторингу та керування мережевими ресурсами сучасних телекомунікаційних інфраструктур.

ANNOTATION

Theme of qualification work: «Component-Based Load Forecasting in Multi-User Computer Networks» // Zemleduh Vladyslav// Ternopil Ivan Puluj National Technical University, Faculty of Applied Information Technologies and Electrical Engineering group RAm-61 // Ternopil, 2025 // p. – 83, fig. – 8, tab. - 1, add. – 2, bibliography -21.

Keywords: network load, forecasting, singular spectrum analysis, SSA, autoregressive model, seasonal regression, time series, multi-user networks

The work addresses the problem of load forecasting in multi-user computer networks characterized by the presence of trend, seasonal and stochastic components. Modern statistical, machine learning and hybrid forecasting approaches are analyzed, and their limitations are identified. A component-based forecasting method is proposed, combining Singular Spectrum Analysis (SSA) for time-series decomposition with autoregressive and seasonal regression models for predicting individual components. An algorithm for automatic classification of components into trend, seasonal and noise types based on spectral features is developed, as well as a method for estimating forecast uncertainty. A software implementation of the proposed method was created in Matlab, providing signal reconstruction, model fitting, component forecasting and confidence interval estimation. Experimental results demonstrate improved forecast accuracy compared to baseline approaches. The method can be applied in monitoring and resource management systems of modern telecommunication networks.

ЗМІСТ

ВСТУП.....	9
РОЗДІЛ 1. АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ПРОГНОЗУВАННЯ НАВАНТАЖЕННЯ У БАГАТОКОРИСТУВАЦЬКИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ.....	12
1.1 Роль прогнозування навантаження у мережевих системах.....	12
1.2 Класифікація існуючих методів прогнозування.....	15
1.2.1 Статистичні моделі часових рядів.....	15
1.2.2 Методи машинного навчання.....	17
1.2.3 Компонентно-орієнтовані та гібридні підходи.....	18
1.3 Недоліки відомих підходів.....	19
1.4 Тенденції розвитку методів прогнозування.....	20
1.5 Висновки до розділу 1.....	20
РОЗДІЛ 2. МЕТОД ТА АЛГОРИТМ ПРОГНОЗУВАННЯ НАВАНТАЖЕННЯ У БАГАТОКОРИСТУВАЦЬКИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ.....	21
2.1 Мета та постановка задачі.....	21
2.2 Формування матриці траєкторій (Hankel-матриці).....	22
2.3 Коваріаційна матриця і сингулярний розклад.....	24
2.4 Відновлення часових компонент (діагональне усереднення).....	29
2.5 Моделювання та прогнозування компонент через авторегресійні моделі.....	31
2.6 Виявлення сезонних компонент.....	33
2.7 Побудова сезонної регресійної моделі.....	35
2.8 Оцінювання параметрів.....	36
2.9 Прогнозування сезонної компоненти.....	36
2.10 Прогнозування на h кроків.....	37
2.10.1 Прогноз окремих компонент.....	37
2.10.2 Формування загального прогнозу.....	37
2.10.3 Оцінювання довірчих інтервалів.....	38
2.10.4 Ймовірність прийнятного прогнозу.....	38

	7
2.11 Оцінювання невизначеності прогнозу.....	39
2.11.1 Залишки прогнозу.....	39
2.11.2 Статистичні характеристики невизначеності.....	39
2.11.3 Довірчі межі прогнозу.....	40
2.11.4 Ймовірність прийнятного прогнозу.....	40
2.12 Оцінка якості моделі.....	41
2.13 Класифікація компонент.....	41
2.14 Підсумкові результати (структура результатів).....	43
2.15 Алгоритм компонентного прогнозування навантаження у багатокористувацьких комп'ютерних мережах.....	44
2.16 Висновки до розділу 2.....	48
РОЗДІЛ 3. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КОМПОНЕНТНОГО ПРОГНОЗУВАННЯ НАВАНТАЖЕННЯ У БАГАТОКОРИСТУВАЦЬКИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ.....	
3.1 Програмне забезпечення компонентного прогнозування навантаження.....	50
3.1.1 Ініціалізація середовища та оголошення глобальних параметрів.....	50
3.1.2 Зчитування даних та початкові обчислення.....	51
3.1.3 Побудова траєкторної (Hankel) матриці.....	52
3.1.4 SVD та вибір SSA-компонент.....	52
3.1.5 Діагональне усереднення та відновлення компонент.....	53
3.1.6 Підбір AR-моделей для компонент.....	54
3.1.7 Виявлення сезонних компонент.....	55
3.1.8 Оцінка регресійних моделей для сезонних компонент.....	56
3.1.9 Прогноз компонент та сумарний прогноз.....	57
3.1.10 Довірчі інтервали та ймовірності $\pm 10\%$	59
3.1.11 Вивід результатів та візуалізація та класифікація за типом.....	59
3.1.12 Збереження результатів та експорт.....	62
3.2 Результати компонентного прогнозування навантаження у багатокористувацьких комп'ютерних мережах.....	62
3.3 Висновки до розділу 3.....	70

РОЗДІЛ 4. ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ.....	72
4.1 Охорона праці.....	72
4.2 Безпека в надзвичайних ситуаціях.....	75
4.3 Висновки до розділу 4.....	78
ЗАГАЛЬНІ ВИСНОВКИ.....	79
ПЕРЕЛІК ПОСИЛАНЬ.....	81
ДОДАТКИ.....	84
ДОДАТОК А. Копія тези конференція.....	85
ДОДАТОК Б. Скрипт програмного забезпечення компонентного прогнозування навантаження (SSA + AR + seasonal-regression).....	94

ВСТУП

Актуальність роботи

Сучасні багатокористувацькі комп'ютерні мережі – корпоративні, провайдерські, хмарні та кампусні – характеризуються високим рівнем динамічності трафіку, різкими коливаннями навантаження, значною кількістю одночасно активних користувачів та великою варіативністю сценаріїв роботи. Зростання популярності потокових сервісів, хмарних обчислень, відеоконференцій, інтерактивних додатків та мобільних платформ призводить до істотного збільшення трафіку, непередбачуваності його структури та появи численних локальних піків. У таких умовах ефективне прогнозування навантаження стає критично важливим для запобігання перевантаженням, оптимального розподілу пропускної здатності, підвищення рівня QoS та забезпечення стабільної роботи мережевих інфраструктур.

Наукові дослідження в галузі прогнозування мережевого навантаження демонструють, що класичні статистичні моделі часових рядів, такі як ARIMA та SARIMA (Box, Jenkins), забезпечують коректні результати лише для процесів зі сталими характеристиками, що обмежує їх застосування у динамічних мережевих середовищах. Методи експоненційного згладжування (Holt, Winters) ефективні лише за умов стабільної сезонності. Моделі машинного навчання, такі як штучні нейронні мережі (Rosenblatt), LSTM (Hochreiter, Schmidhuber) та SVM (Vapnik, Chervonenkis), демонструють високу точність, проте мають суттєві недоліки: високі обчислювальні витрати, потребу у великих обсягах даних та обмежену інтерпретованість результатів.

У той же час компонентно-орієнтовані декомпозиційні методи, зокрема сингулярний спектральний аналіз SSA (Ghil та ін.) та емпірична модальна декомпозиція EMD (Huang), довели ефективність у виділенні структурних складових трафіку, однак у чистому вигляді не забезпечують прогнозування. Тому особливої актуальності набувають гібридні підходи, у яких декомпозиція використовується для відокремлення трендових, сезонних і коливальних компонент із подальшим параметричним моделюванням кожної з них.

У сучасних умовах створення компонентного методу, який поєднує декомпозицію часових рядів, адаптивне виділення сезонних і трендових структур та подальше прогнозування за допомогою авторегресійних моделей, є актуальним напрямом, що сприяє підвищенню точності прогнозів, інтерпретованості результатів та оптимізації роботи багатокористувацьких мереж.

Метою роботи є розроблення та дослідження методу компонентного прогнозування навантаження у багатокористувацьких комп'ютерних мережах, який забезпечує високу точність і стійкість передбачення за рахунок поєднання сингулярного спектрального аналізу та авторегресійно-регресійного моделювання часових компонент.

Для досягнення поставленої мети необхідно виконати такі **завдання**:

1. Провести аналіз існуючих методів прогнозування навантаження у комп'ютерних мережах з метою виявленням їх переваг та обмежень.
2. Розробити компонентний підхід до декомпозиції часових рядів прогнозування навантаження на основі сингулярного спектрального аналізу.
3. Розробити метод автоматичної класифікації виділених компонент за їхніми динамічними та спектральними властивостями.
4. Побудувати математичні моделі прогнозування трендових, сезонних і стохастичних компонент із використанням авторегресійних та регресійних методів.
5. Розробити алгоритм компонентного прогнозування навантаження у багатокористувацьких комп'ютерних мережах.
6. Реалізувати програмне забезпечення компонентного прогнозування, провести його експериментальну перевірку та оцінити ефективність запропонованого методу.

Об'єктом дослідження є процес зміни навантаження у багатокористувацьких комп'ютерних мережах.

Предметом дослідження є методи та алгоритми компонентного прогнозування часових рядів навантаження, що базуються на сингулярному спектральному аналізі та авторегресійному моделюванні.

Наукова новизна

Розроблено метод компонентного прогнозування навантаження у багатокористувацьких комп'ютерних мережах, який поєднує сингулярний спектральний аналіз для виділення структурних компонент із авторегресійним моделюванням для передбачення їх динаміки, що забезпечило підвищення точності прогнозування та отримання інтерпретованої структури прогнозу з окремими внесками трендових, сезонних і шумових процесів.

Практичне значення

Розроблений метод може бути використаний у системах керування мережевими ресурсами, службах моніторингу трафіку, інтелектуальних платформах прогнозування навантаження серверів та дата-центрів. Він дозволяє підвищити ефективність використання пропускної здатності, зменшити ризики перевантаження мережі та оптимізувати розподіл обчислювальних ресурсів у режимі реального часу.

РОЗДІЛ 1

АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ПРОГНОЗУВАННЯ НАВАНТАЖЕННЯ У БАГАТОКОРИСТУВАЦЬКИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ

1.1 Роль прогнозування навантаження у мережевих системах

Ефективність функціонування сучасних комп'ютерних мереж безпосередньо визначається здатністю системи керування адаптуватися до динамічних змін інтенсивності інформаційних потоків. У сучасних умовах глобальної цифровізації споживачі генерують величезні обсяги даних – від мультимедійних сервісів до IoT-пристроїв, – що створює істотне та нерівномірне навантаження на інфраструктуру. Тому точне прогнозування мережевого трафіку набуває фундаментального значення, забезпечуючи як оперативність роботи, так і стратегічний розвиток інформаційних систем.

Передусім прогнозування навантаження дозволяє оптимізувати використання пропускної здатності мережевих каналів. Канали зв'язку, сервери, маршрутизатори та інші елементи інфраструктури мають обмеження за обсягом даних, який може бути переданий протягом одиниці часу. Коли навантаження перевищує допустимий рівень, виникають затримки, втрати пакетів і деградація якості сервісу. Завдяки прогнозуванню можливо наперед оцінити моменти можливого перевантаження та вжити коригувальних заходів – збільшити ресурси, додати резервні канали або спрямувати частину трафіку альтернативними маршрутами.

Ще одним важливим аспектом є забезпечення стабільності та надійності роботи мережі. Більшість сервісів, що працюють у реальному часі (відеоконференції, хмарні сервіси, онлайн-ігри, SCADA-системи), критично чутливі до змін навантаження. Наприклад, навіть короточасні сплески трафіку можуть спричинити збільшення затримок, що погіршує якість передачі мультимедіа або порушує коректність роботи автоматизованих систем. Прогнозні моделі дозволяють виявляти такі потенційні піки заздалегідь, тим самим мінімізуючи ризики.

Сучасні мережеві системи дедалі частіше працюють у режимах автоматизованого управління ресурсами, де рішення приймаються алгоритмами машинного або компонентного навчання. Для коректної роботи таких алгоритмів система повинна отримувати достовірну інформацію про майбутній стан трафіку. Це дозволяє здійснювати балансування навантаження між серверами, автоматичне масштабування контейнеризованих сервісів, активацію та деактивацію віртуальних машин, а також інтелектуальне керування потоками. У перспективі це забезпечує економію енергоресурсів і зниження вартості експлуатації інфраструктури.

Важливою передумовою побудови точних прогнозів є характеристика самого мережевого трафіку. Він має несталість (non-stationarity), оскільки статистичні властивості процесу змінюються з часом. Наприклад, середній рівень навантаження в робочі дні відрізняється від вихідних, а активність користувачів у вечірні години значно вища, ніж уночі. До того ж трафік має багатокомпонентну структуру, яка включає кілька складових:

- трендову компоненту, що відображає довгострокові тенденції, наприклад зростання кількості користувачів або збільшення обсягу переданих даних (див. рис. 1.1);
- періодичні коливання – добові, тижневі, сезонні, що зумовлені поведінковими та соціальними факторами;
- випадкові відхилення, пов'язані з непередбаченими подіями, такими як аварії, атаки, раптові сплески активності або збої в системах.

На рисунку 1.1 схематично показано дві характерні компоненти трафіку: довгострокове трендове зростання та періодичні циклічні коливання інтенсивності. Їхня сумарна дія створює складний часовий ряд, який потрібно моделювати з урахуванням усіх типів змін.

Через таку складність мережевого навантаження традиційні моделі прогнозування, що базуються на припущенні стаціонарності, часто виявляються недостатньо ефективними. Саме тому сучасні підходи застосовують комбіновані, гібридні або інтелектуальні моделі, які поєднують кілька методів одночасно. Це можуть бути:

- статистичні моделі (ARIMA, SARIMA, ETS);
- методи машинного навчання (Random Forest, Gradient Boosting);
- нейронні мережі (LSTM, GRU, TCN);
- гібридні моделі, що об'єднують компоненти спектрального аналізу та нейромереж.

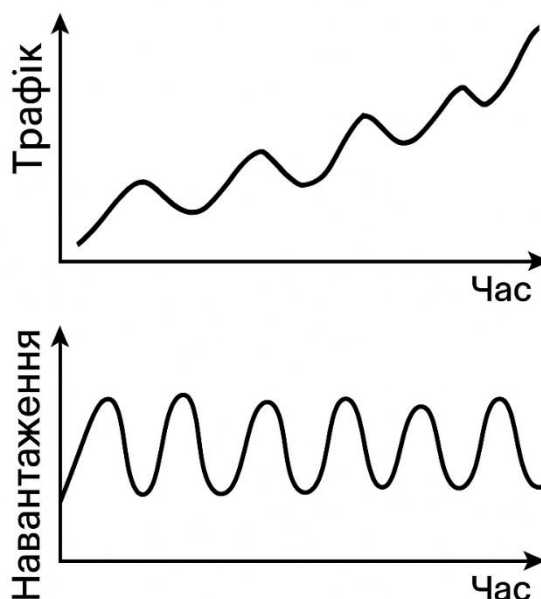


Рис.1.1. Структурні компоненти мережевого трафіку: трендова складова (зростання інтенсивності) та періодичні коливання навантаження у часі

Ще однією причиною, чому прогнозування навантаження має надзвичайно важливе значення, є енергоефективність та оптимізація витрат. Великі дата-центри та оператори зв'язку витрачають значні ресурси на обробку та передачу даних. Якщо наперед відомо, коли навантаження буде мінімальним, можна тимчасово вимикати частину обладнання або переводити його в енергозберігальний режим. Це дозволяє забезпечити як екологічність, так і економічність роботи інфраструктури. Навпаки, прогноз пікових навантажень дає змогу завчасно активувати додаткові потужності.

У контексті багатоланкових мереж важливим є також прогнозування просторової зміни навантаження, тобто аналіз того, в яких сегментах мережі може виникнути перенавантаження. Розподіл трафіку може бути нерівномірним, наприклад між різними дата-центрами, віддаленими регіонами або вузлами

магістральних мереж. Передбачення такої нерівномірності допомагає оптимізувати маршрутизацію, дублювання каналів, кешування контенту та інші механізми підвищення продуктивності.

У цілому прогнозування навантаження є не лише інструментом оперативного реагування, але й важливою складовою стратегічного планування. Воно дозволяє:

- оцінити необхідність модернізації інфраструктури;
- визначити оптимальні точки масштабування;
- прогнозувати майбутні потреби користувачів;
- приймати рішення щодо впровадження нових сервісів;
- забезпечити якість обслуговування (QoS) та угоди рівня сервісу (SLA).

Таким чином, прогнозування навантаження в мережевих системах має вирішальне значення для забезпечення стабільної та ефективної роботи сучасних інформаційних інфраструктур. Воно дозволяє не лише мінімізувати ризики перевантаження та збоїв, але й створює основу для довгострокового розвитку систем зв'язку, підвищення їхньої надійності та адаптивності. У поєднанні з сучасними методами аналізу даних прогнозування формує фундамент інтелектуальних мереж наступного покоління, здатних самостійно приймати оптимальні рішення на основі оцінки майбутніх станів навантаження.

1.2 Класифікація існуючих методів прогнозування

Сучасні підходи до прогнозування навантаження у комп'ютерних мережах можна умовно поділити на три групи:

- Статистичні моделі часових рядів;
- Методи машинного навчання;
- Компонентно-орієнтовані та гібридні підходи.

1.2.1 Статистичні моделі часових рядів

Однією з найвідоміших фундаментальних груп методів прогнозування стали статистичні моделі авторегресійного типу. У роботах Дж. Бокса та Г. Дженкінса [1]

запропоновано ARIMA-модель, яка формально описує часовий ряд через попередні значення та похибки прогнозу:

$$\phi(B)(1-B)^d y(t) = \theta(B)\varepsilon(t), \quad (1.1)$$

де $\phi(B)$ та $\theta(B)$ – поліноми авторегресійної й ковзної складових, B – оператор зсуву, d – порядок диференціювання, $\varepsilon(t)$ – білий шум.

ARIMA забезпечує прийнятну точність короткострокового прогнозу та добре працює за умов стаціонарності, але мережевий трафік часто є нестаціонарним, містить пікові навантаження та багаточастотні сезонності, що знижує ефективність моделі.

Для врахування сезонних ефектів була запропонована модель SARIMA, яка розширює базову ARIMA, дозволяючи моделювати добові цикли трафіку. Проте в реальних мережах сезонність може змінюватися, а сплески навантаження не мають регулярної природи, тому SARIMA також має обмеження.

Метод експоненційного згладжування Гольта–Вінтерса, розроблений Ч. Гольтом і П. Вінтерсом [2], дозволив увести адитивну або мультиплікативну структуру:

$$y(t) = l(t) + b(t) + s(t), \quad (1.2)$$

де $l(t)$ – рівень, $b(t)$ – тренд, $s(t)$ – сезонність.

Цей метод успішний у випадку стабільної сезонності та плавної зміни тренду, що інколи характерно для навантаження офісних або локальних мереж. Проте він не є достатньо гнучким у високоволатильних мережах, де сезонні коливання мають складну структуру.

Серед статистичних методів також вирізняється векторна авторегресія (VAR), запропонована К. Сімсом [3], яка моделює взаємозв'язки між кількома параметрами мережі:

$$Y(t) = A_1 Y(t-1) + A_2 Y(t-2) + \dots + A_p Y(t-p) + \varepsilon(t), \quad (1.3)$$

де $Y(t)$ – вектор характеристик (трафік, затримки, втрати пакетів),
 A_i – матриці коефіцієнтів.

VAR дає змогу враховувати взаємозалежності у мережі, але його ефективність знижується через високу вимогу до обсягів даних та чутливість до нестационарності.

1.2.2 Методи машинного навчання

Розвиток теорії нейронних мереж, започаткований Ф. Розенблаттом [4], відкрив можливість розглядати прогнозування як задачу нелінійної апроксимації функції:

$$y(t) = f(y(t-1), y(t-2), \dots, y(t-p)). \quad (1.4)$$

Штучні нейронні мережі (ANN) здатні виявляти складні залежності, які не піддаються аналітичному опису. У контексті мережевого трафіку ці моделі дозволяють враховувати нелінійний характер навантаження, вплив зовнішніх факторів та приховані структурні зміни.

Однак ANN вимагають великих обсягів даних і можуть бути нестійкими до шуму, що часто присутній у мережевому трафіку.

Подальший прорив стався із появою LSTM-мереж, розроблених С. Хохрайтером і Ю. Шмідхубером [5]. LSTM здатні зберігати інформацію на довгих часових інтервалах, тому вони ефективні для прогнозування добових, тижневих і навіть місячних циклів трафіку:

$$h(t) = f(h(t-1), x(t)), \quad (1.5)$$

де $h(t)$ – прихований стан, $x(t)$ – вхідний вектор.

LSTM демонструють високу точність, проте потребують значних обчислювальних ресурсів і складного налаштування.

Метод опорних векторів (SVM), запропонований В. Вапніком і А. Червоненкісом [6], базується на апараті ядрових методів:

$$y(t) = \sum_{i=1}^N \alpha_i K(x_i, x(t)) + b, \quad (1.6)$$

що забезпечує високу точність для завдань короткострокового прогнозування. Але SVM погано масштабується на довгі часові ряди, що обмежує його застосування в мережах, де прогнозування виконується на десятки або сотні кроків уперед.

1.2.3 Компонентно-орієнтовані та гібридні підходи

Компонентно-орієнтовані підходи базуються на ідеї декомпозиції складного часового ряду на структурні складові – тренд, сезонність, коливання та шум – з подальшим окремим моделюванням кожної з них. Одним із найвідоміших методів такого типу є сингулярний спектральний аналіз (SSA), запропонований Д. Гілем та співавторами [7]. Метод передбачає побудову траєкторної матриці:

$$X = \begin{bmatrix} y(1) & y(2) & \dots & y(K) \\ y(2) & y(3) & \dots & y(K+1) \\ \vdots & \vdots & \ddots & \vdots \\ y(L) & y(L+1) & \dots & y(N) \end{bmatrix}, \quad (1.7)$$

і подальший сингулярний розклад:

$$X = \sum_{i=1}^L \sigma_i U_i V_i^T. \quad (1.8)$$

SSA дозволяє виявити приховані цикли, добові та тижневі сезонності, відокремити шум, що є надзвичайно важливим для мереж, де структура навантаження часто є багатокomпонентною і нерівномірною.

Проте SSA не здійснює прогнозування на пряму, тому в працях Wang та ін. [8] рекомендовано комбінувати SSA з AR-моделями або нейронними мережами.

Емпірична модальна декомпозиція (EMD), запропонована Н. Хуаном [9], розкладає сигнал на набори внутрішніх мод:

$$y(t) = \sum_{i=1}^N IMF_k(t) + r(t). \quad (1.9)$$

Цей метод добре підходить для адаптації до локальних змін структури трафіку, але є чутливим до шуму та не має строгого теоретичного обґрунтування.

Сезонно-трендова декомпозиція STL, розроблена Р. Клівлендом [10], використовує локальну регресію Loess для розділення ряду на тренд, сезонність і залишок. Хоча STL добре працює для нечіткої сезонності, він обмежений у випадку складних багаточастотних структур, притаманних великим комп'ютерним мережам.

1.3 Недоліки відомих підходів

На основі аналізу джерел [1-8] можна виділити такі ключові проблеми:

- статистичні моделі не забезпечують адекватної роботи при зміні статистичних властивостей процесу;
- нейронні мережі «чорного ящика» не дозволяють пояснити природу прогнозу;
- гібридні моделі часто не узгоджують між собою етапи декомпозиції та прогнозування;
- відсутність єдиного підходу до оцінювання невизначеності прогнозу.

Ці обмеження обґрунтовують необхідність створення уніфікованої методики, що забезпечує структурну інтерпретацію процесу та водночас високу прогнозну точність.

1.4 Тенденції розвитку методів прогнозування

Аналіз сучасних публікацій (Ferreira et al., 2023 [6]; Wang et al., 2024 [5]) свідчить, що основні тенденції розвитку прогнозування навантаження зводяться до:

- Комбінування структурної декомпозиції (SSA, EMD) з адаптивним моделюванням (AR, LSTM);
- Автоматичної класифікації компонент (трендових, сезонних, шумових);
- Інтеграції статистичних і машинних підходів у єдині моделі;
- Використання ймовірнісних критеріїв оцінювання точності прогнозів.

Саме у цьому напрямі розвивається запропонований у даній роботі метод – компонентне прогнозування на основі SSA+AR, який дозволяє об'єднати переваги непараметричних і параметричних моделей.

1.5 Висновки до розділу 1

У ході аналізу встановлено, що існуючі підходи до прогнозування навантаження в комп'ютерних мережах мають як сильні, так і слабкі сторони.

Статистичні моделі є простими, але малоефективними для нестационарних процесів; нейронні мережі – гнучкі, проте обчислювально складні; гібридні – дають високі результати, але потребують узгодження між етапами декомпозиції і прогнозування.

Визначено, що перспективним напрямом є компонентне прогнозування на основі сингулярного спектрального аналізу (SSA) з подальшим AR-моделюванням кожної компоненти, яке забезпечує як структурну інтерпретацію процесу, так і високу точність прогнозу.

РОЗДІЛ 2

МЕТОД ТА АЛГОРИТМ ПРОГНОЗУВАННЯ НАВАНТАЖЕННЯ У
БАГАТОКОРИСТУВАЦЬКИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ

2.1 Мета та постановка задачі

У сучасних комп'ютерних мережах, що обслуговують тисячі користувачів, ефективне керування ресурсами потребує точного прогнозування мережевого навантаження. Такий прогноз дозволяє своєчасно реагувати на пікові періоди, оптимізувати розподіл каналів, пам'яті та обчислювальних потужностей. Особливо актуальним є завдання прогнозування трафіку у масштабах доби, де характерні як стохастичні коливання, так і періодичні добові закономірності.

Запропонований метод компонентного прогнозування навантаження базується на поєднанні трьох підходів:

- Сингулярного спектрального аналізу (SSA) – для виділення структурних компонент сигналу (трендових, сезонних, шумових);
- Авторегресійного моделювання (AR) – для екстраполяції кожної з компонент;
- Регресійного моделювання із сезонною складовою – для врахування добових циклів у мережевому трафіку.

Таке об'єднання дозволяє одночасно відокремити основні закономірності часових рядів і зберегти адаптивність до випадкових змін у навантаженні.

Нехай маємо часовий ряд навантаження мережі:

$$y(t), \quad t = 1, 2, \dots, N. \quad (2.1)$$

Він може бути поданий як сума незалежних компонент:

$$y(t) = T(t) + S(t) + R(t), \quad (2.2)$$

де $T(t)$ – трендова компонента, що описує довгострокову зміну трафіку;
 $S(t)$ – сезонна складова, пов’язана з добовими або тижневими циклами;
 $R(t)$ – залишкова (шумова) компонента.

Мета методу полягає в оцінюванні цих складових на основі історичних даних і побудові прогнозу на наступний часовий горизонт h , тобто побудувати прогноз на h кроків уперед:

$$y(N+1), y(N+2), \dots, y(N+h). \quad (2.3)$$

на основі минулих спостережень $y(1), \dots, y(N)$, шляхом:

- розкладу ряду на основні компоненти (тренд, сезонність, шум) методом Singular Spectrum Analysis (SSA);
- окремого моделювання кожної компоненти через авторегресійні моделі (AR);
- об’єднання прогнозів компонент у загальний прогноз;
- оцінювання довірчих інтервалів прогнозу на основі статистики залишків реконструкції.

2.2 Формування матриці траєкторій (Hankel-матриці)

Одновимірний часовий ряд $y(t)$ містить у собі корисну інформацію не лише про рівень спостережуваного процесу, а й про його внутрішню часову структуру – тренди, коливання, сезонність, автокореляцію. Проте у такому вигляді цей ряд є лише послідовністю чисел, і без додаткового перетворення втрачається можливість ефективно виявити приховані закономірності між сусідніми спостереженнями.

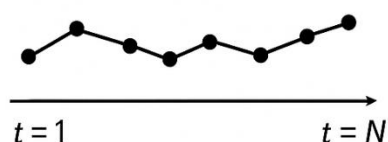
Для того щоб виявити ці зв’язки, часовий ряд перетворюють у багатовимірний простір – у так звану матрицю траєкторій (Hankel-матрицю):

$$X = \begin{bmatrix} y(1) & y(2) & \dots & y(K) \\ y(2) & y(3) & \dots & y(K+1) \\ \vdots & \vdots & \ddots & \vdots \\ y(L) & y(L+1) & \dots & y(N) \end{bmatrix}, \quad (2.4)$$

де L – довжина вікна (параметр згладжування);

$K=N-L+1$ – кількість.

Таке представлення дозволяє перейти від одновимірного часового сигналу $y(t)$ до багатовимірного простору станів, де кожен стовпець відображає часовий фрагмент сигналу (рис.2.1).



$$X = [x_1, x_2, \dots, x_K]$$

$$\begin{aligned} x_1 &= \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N-L) \end{bmatrix} \\ x_2 &= \begin{bmatrix} y(2) \\ y(3) \\ \vdots \\ y(N-L+1) \end{bmatrix} \\ &\vdots \\ x_K &= \begin{bmatrix} y(K) \\ y(K+1) \\ \vdots \\ y(N) \end{bmatrix} \end{aligned}$$

Матриця траєкторій

Вбудована траєкторія

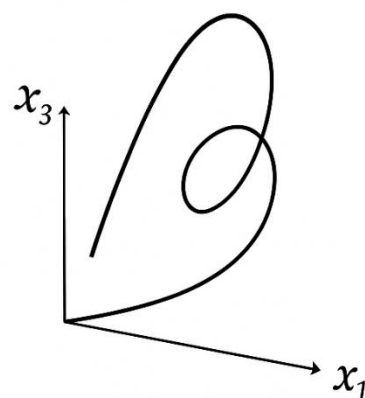


Рис.2.1. Матричне представлення вектора навантаження

Інтерпретація:

- Кожен стовпець матриці X – це вектор стану системи у момент часу t , тобто фрагмент часового ряду довжини L
- Таким чином, часовий ряд описується як сукупність послідовних вікон, що частково перекриваються. Зокрема, замість одновимірної послідовності $y(t)$ ми

отримуємо траєкторію в багатовимірному просторі, яка відображає динаміку зміни станів.

– Якщо ряд має тренди або циклічні складові, вони проявляться у вигляді корельованих напрямів у цьому просторовому представленні.

Таке матричне представлення дозволяє:

- виявляти корельовані структури між сусідніми спостереженнями;
- виділяти квазіперіодичні компоненти;
- здійснювати спектральне розкладання сигналу за допомогою методів лінійної алгебри.

Перетворення часового ряду у матрицю траєкторій – це ключовий крок у методі SSA, який забезпечує перехід від одномірного опису процесу до його багатовимірного відображення. У цьому просторі часові залежності перетворюються у просторові структури, що піддаються розкладанню на незалежні моди. Саме це дає можливість розділити тренди, сезонні коливання і шум, що неможливо зробити без віконного представлення.

2.3 Коваріаційна матриця і сингулярний розклад

Після того, як часовий ряд $y(t)$ перетворено у матрицю траєкторій X за допомогою віконного представлення, наступним кроком є виявлення основних закономірностей у структурі даних.

Оскільки ряд може містити взаємопов'язані трендові, періодичні та шумові компоненти, потрібно знайти ті напрямки у просторі векторів-станів, де спостерігається найбільша варіація (мінливість) сигналу.

Для цього формується коваріаційна матриця, яка описує, наскільки сильно змінюються елементи векторів і наскільки вони корелюють між собою (виявлення основних напрямів мінливості формується коваріаційна матриця). Вона визначається як:

$$C = \frac{1}{K} XX^T, \quad (2.5)$$

де X – матриця траєкторій розміром $L \times K$, $K = N - L + 1$, а C – квадратна матриця розміром $L \times L$, елементи якої мають вигляд:

$$C_{ij} = \frac{1}{K} \sum_{k=1}^K (x_{i,k} - \bar{x}_i)(x_{j,k} - \bar{x}_j), \quad (2.6)$$

де $\bar{x}_i$ та $\bar{x}_j$ – середні значення відповідних рядків матриці X .

Коваріаційна матриця C характеризує взаємозв'язок між часовими фрагментами та дозволяє визначити головні напрями зміни сигналу $y(t)$.

Коваріаційна матриця є симетричною і додатно визначеною, а отже має повний набір власних векторів, які формують ортонормований базис простору станів.

Кожен власний вектор u_i вказує на напрямок головної мінливості у даних, а відповідне власне значення λ_i характеризує величину дисперсії вздовж цього напрямку:

$$CU_i = \lambda_i U_i, \quad i = 1, 2, \dots, L \quad (2.7)$$

де λ_i – власні значення, а U_i – відповідні власні вектори.

Власні значення λ_i впорядковують у спадному порядку:

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_L, \quad (2.8)$$

де найбільші з них відповідають домінуючим закономірностям, тобто тим складовим, які містять основну частину енергії сигналу.

Таким чином, аналіз спектра власних значень дозволяє кількісно оцінити, скільки незалежних структур (трендів або гармонік) міститься у часовому ряді.

Коваріаційна матриця є статистичною моделлю кореляцій між усіма відкладеними копіями сигналу.

Кожен її елемент C_{ij} показує, наскільки зміни сигналу в момент часу $t+i$ пов'язані зі змінами у момент $t+j$.

Якщо процес має періодичну або квазіперіодичну структуру, то в матриці C з'являються характерні повторювані діагональні патерни, які відображають наявність циклічних коливань.

Таким чином, аналіз матриці C дозволяє виділити:

- трендові компоненти – напрямки із повільною, але суттєвою зміною дисперсії;
- періодичні коливання – напрямки, для яких чергуються позитивні й негативні кореляції;
- шумову складову – рівномірно розподілені дрібні варіації без вираженої структури.

Після побудови коваріаційної матриці $C = \frac{1}{K} XX^T$, наступним кроком є визначення її внутрішньої структури за допомогою спектрального розкладу.

Ця процедура дозволяє представити матрицю у вигляді суми добутків власних векторів та відповідних власних значень, тобто:

$$C = U \Lambda U^T, \quad (2.9)$$

де $U = [u_1, u_2, \dots, u_L]$ – ортонормована матриця власних векторів C ;

$\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_L)$ – діагональна матриця власних значень ($\lambda_1 \geq \lambda_2 \geq \dots \geq 0$).

Кожен власний вектор u_i визначає напрям у просторі траєкторій, уздовж якого дисперсія (мінливість) сигналу є максимальною і дорівнює λ_i .

Таким чином, спектральний розклад забезпечує ортогональне декомпонування простору станів на незалежні моди мінливості, які не корелюють між собою.

Кожну моду можна розглядати як базову структуру поведінки процесу, а повний набір таких мод – як оптимальну систему координат для аналізу та реконструкції сигналу.

Однак для того, щоб перейти від абстрактного опису варіацій до кількісної оцінки інформативності кожної компоненти, необхідно визначити сингулярні значення, що відображають амплітуду відповідних мод у матриці траєкторій.

Іншими словами, спектральний розклад коваріаційної матриці задає напрямки зміни, а сингулярні значення σ_i визначають силу або інтенсивність цих змін у просторі даних.

Сингулярні значення σ_i показують, наскільки сильно кожна прихована компонента (тренд, періодичність або шум) впливає на структуру сигналу.

Якщо уявити матрицю траєкторій X як багатовимірне представлення процесу, то сингулярні значення визначають:

$$\sigma_i = \sqrt{\lambda_i}, \quad (2.10)$$

де λ_i – власне значення коваріаційної матриці $C = \frac{1}{K} XX^T$.

Отже:

- великі σ_i – відповідають тренду або періодичним коливанням,
- дрібні σ_i – відображають шумові складові.

Таким чином, саме за значенням σ_i ми вирішуємо, які компоненти залишити для відновлення сигналу, а які відкинути.

Загалом, сингулярні значення дають змогу визначити ефективну розмірність простору даних – тобто, скільки незалежних факторів реально описують поведінку системи.

Інтерпретація:

- Власні вектори U визначають основні напрямки зміни часової структури;
- Власні значення λ_i показують внесок відповідної компоненти у загальну дисперсію сигналу;

– Таким чином, перші кілька компонент ($r \ll L$) зазвичай пояснюють більшу частину енергії ряду – тренд і сезонність.

Після виконання сингулярного розкладу траєкторної або коваріаційної матриці, часовий ряд подається у вигляді суми окремих ортогональних компонент, кожна з яких описує певний тип поведінки сигналу – трендову, сезонну або випадкову.

Метою сингулярного розкладу є відокремлення незалежних напрямів мінливості даних у просторі станів. Це дозволяє представити складну структуру сигналу у вигляді суми простих, взаємно ортогональних складових, кожна з яких має певну «енергетичну вагу» – сингулярне значення.

Отже, після виконання сингулярного розкладу матриця траєкторій X подається у вигляді суми елементарних матриць внеску:

$$X = \sum_{i=1}^r X_i = \sum_{i=1}^r \sigma_i U_i V_i^T, \quad (2.11)$$

де σ_i – сингулярні значення, що визначають енергію або «вагу» відповідної компоненти,

U_i – вектор лівих сингулярних векторів, який описує часову структуру у межах вікна,

V_i – вектор правих сингулярних векторів, який визначає еволюцію компонент у часі,

r – кількість обраних (значущих) компонент (ранг апроксимації або кількість обраних компонент для подальшої реконструкції).

X_i – елементарна матриця внеску, що описує частку структури сигналу, пов'язану з певною сингулярною модою i :

$$X_i = \sigma_i U_i V_i^T, \quad (2.12)$$

Після розкладу одержуємо набір ортогональних компонент:

$$\{X_1, X_2, \dots, X_r\}, \quad (2.13)$$

Сума обраних матриць відображає лише ті процеси, які мають фізичне або статистичне значення, наприклад, повільні трендові зміни чи основні періодичні коливання трафіку. Компоненти з малими сингулярними значеннями описують шумові флуктуації і не враховуються при реконструкції.

Іншими словами, кожна така матриця містить у собі локалізований внесок окремої динамічної компоненти (тренд, періодичну або шумову складову) в загальну поведінку ряду.

Її аналіз дозволяє ідентифікувати, які саме процеси (повільні тренди, періодичні коливання або швидкі флуктуації) формують спостережувану структуру навантаження у комп'ютерній мережі.

Таким чином, визначення елементарних матриць внеску є ключовим етапом у компонентному розкладі часового ряду, що забезпечує подальше його фізично осмислене відновлення та прогнозування.

Щоб знайти вектори V_i , використовується співвідношення, яке безпосередньо випливає з визначення SVD:

$$X_T U_i = \sigma_i V_i, \quad (2.14)$$

Звідси отримано вираз для обчислення:

$$V_i = \frac{X^T U_i}{\sigma_i}, \quad (2.15)$$

2.4 Відновлення часових компонент (діагональне усереднення)

Наступним етапом є відновлення часових компонент, яке здійснюється шляхом діагонального усереднення (reconstruction by diagonal averaging), що дає змогу перейти від матричного представлення назад до одновимірного часового ряду.

Діагональне усереднення – це процедура, яка перетворює кожен матрицю X_i назад у часовий ряд, забезпечуючи узгодженість між елементами, що дублюються в матриці.

Ідея полягає в тому, що кожен елемент відновленого сигналу обчислюється як середнє значення елементів матриці X_i , які належать до однієї антидіагоналі (із однаковою сумою індексів $k=a+b-1$):

$$\tilde{y}_i(t) = \frac{1}{n_t} \sum_{a+b-1=t} (X_i)_{ab}, \quad t = 1, 2, \dots, N, \quad (2.16)$$

де n_t – кількість елементів на відповідній антидіагоналі.

У результаті діагонального усереднення отримуємо набір компонент:

$$\{\tilde{y}_1(t), \tilde{y}_2(t), \dots, \tilde{y}_r(t)\}, \quad i = 1, 2, \dots, r \quad (2.17)$$

Кожна з них є узгодженою часовою послідовністю, що відображає внесок певної структурної складової сигналу $y(t)$.

Процедура діагонального усереднення:

- відновлює ганкелеву структуру матриці (однакові значення вздовж антидіагоналей);
- забезпечує узгодженість і згладжування між перекривними фрагментами сигналу;
- дозволяє повернутись із багатовимірного простору станів до одновимірного сигналу.

Таким чином, цей етап відновлює фізичний часовий вигляд сигналу після його розкладу та обробки у матричній формі.

Відновлений сигнал (реконструкція) визначається як:

$$\hat{y}(t) = \sum_{i=1}^r \tilde{y}_i(t), \quad (2.18)$$

де $\hat{y}(t)$ – відновлений сигнал, який описує лише статистично або фізично значущі процеси – тренд, періодичні коливання тощо.

Ця сума відображає лише ті процеси, які мають статистичне або фізичне значення – наприклад, тренд чи основні періодичні коливання.

Компоненти, які не включені до відновлення, утворюють залишкову матрицю або похибку реконструкції (невідновлена частина):

$$\varepsilon(t) = y(t) - \hat{y}(t) = \sum_{i=r+1}^L \tilde{y}_i(t), \quad (2.19)$$

Цей залишок містить ті частини сигналу, що не були відтворені у вибраній моделі – найчастіше це шум, випадкові флуктуації або короткоперіодні збурення.

Інтерпретація:

- $\tilde{y}_i(t)$ – відфільтрована компонентна послідовність (тренд, коливання або шум);
- великі σ_i відповідають значущим структурам (тренд або сезонність);
- малі σ_i відповідають шуму;
- $\varepsilon(t)$ – некорельований залишок або стохастична частина.

2.5 Моделювання та прогнозування компонент через авторегресійні моделі

Кожна відновлена компонента $\tilde{y}_i(t)$ може бути змодельована окремо з використанням авторегресійних (AR) або сезонних рівнянь.

Для кожної відновленої компоненти $x_i(t)$ після усунення середнього:

$$y_i^d(t) = y_i(t) - \bar{y}_i, \quad (2.20)$$

підбирається модель авторегресії порядку p :

$$\tilde{y}_i(t) = \sum_{p=1}^{P_i} a_{ip} \tilde{y}_i(t-p) + \xi_i(t), \quad (2.21)$$

де a_{ip} – коефіцієнти авторегресії:

$\xi_i(t)$ – випадкова помилка, що описує невраховану частину коливань.

Загальний прогноз на горизонт τ визначається як:

$$\hat{y}_{np}(t+\tau) = \sum_{i=1}^r \hat{y}_i(t+\tau), \quad (2.22)$$

Параметри a_{ip} оцінюються методом Yule–Walker, який ґрунтується на автоковаріаційній функції:

$$r_i(\tau) = \frac{1}{n} \sum_{p=1}^{P_i} y_i^d(t) y_i^d(t-\tau), \quad (2.23)$$

та розв’язанні системи рівнянь:

$$X = \begin{bmatrix} r_i(0) & r_i(1) & \dots & r_i(p-1) \\ r_i(1) & r_i(0) & \dots & r_i(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r_i(p-1) & r_i(p-2) & \dots & r_i(0) \end{bmatrix} \begin{bmatrix} a_{i1} \\ a_{i2} \\ \vdots \\ a_{ip} \end{bmatrix} = \begin{bmatrix} r_i(1) \\ r_i(2) \\ \vdots \\ r_i(p) \end{bmatrix}. \quad (2.24)$$

Порядок p вибирається за критерієм Акаїке (AIC):

$$AIC(p) = n \ln(\hat{\sigma}_i^2(p)) + 2p, \quad (2.25)$$

де $\hat{\sigma}_i^2(p)$ – оцінена дисперсія залишків моделі. Обирається p з мінімальним значенням AIC.

Інтерпретація:

- AR-модель відтворює внутрішню динаміку компоненти безпосередньо на часовій шкалі.
- Якщо компоненту можна описати короткою AR-моделлю, вона стаціонарна та передбачувана.
- Компоненти з великими сингулярними значеннями σ_i відображають системні процеси (добові, тижневі цикли, тренд).
- Компоненти з малими σ_i відповідають шуму.
- Відновлений сигнал $\hat{y}(t)$ є згладженою, фізично осмисленою оцінкою навантаження, яка використовується для прогнозування.

2.6 Виявлення сезонних компонент

Після відновлення окремих часових компонент $\tilde{y}_i(t)$ необхідно визначити, які з них мають сезонний (періодичний) характер. Цей етап дає змогу відокремити компоненти, що описують регулярні коливання навантаження, від нерегулярних або шумових процесів.

Для кожної компоненти $\tilde{y}_i(t)$ обчислюється автокореляційна функція (ACF):

$$\rho_i(\tau) = \frac{\sum_{t=1}^{N-\tau} (\tilde{y}_i(t) - \bar{y}_i)(\tilde{y}_i(t+\tau) - \bar{y}_i)}{\sum_{t=1}^N (\tilde{y}_i(t) - \bar{y}_i)^2}, \quad (2.26)$$

де $\bar{y}_i$ – середнє значення компоненти,

τ – лаг (зсув у часі).

Якщо для лагу, що відповідає тривалості доби (наприклад, $\tau=48$ півгодинних інтервалів), автокореляція $\rho_i(\tau)$ перевищує певний поріг ρ_{th} , то така компонента вважається сезонною:

$$\rho_i(48) \geq \rho_{th}. \quad (2.27)$$

Додатково для кожної компоненти виконується спектральний аналіз за допомогою швидкого перетворення Фур'є (FFT):

$$S_i(f) = |FFT[\tilde{y}_i(t) - \bar{y}_i]|. \quad (2.28)$$

де $S_i(f)$ – амплітудний спектр.

Максимум спектра при частоті f_{max} , що відповідає періоду $T=1/f_{max}$, свідчить про наявність періодичної складової.

Номер піку k^* із найбільшою амплітудою визначає головну частоту:

$$f_i = \frac{k^* - 1}{n}, \quad T_i = \frac{1}{f_i}. \quad (2.29)$$

Якщо $T_i \approx L_d$, тобто період близький до доби, компоненту також вважають сезонною. Зокрема, якщо $T \approx 48$ або $T \approx 24$ (для добової чи півдобової сезонності), то компонента також класифікується як сезонна.

Компонента $\tilde{y}_i(t)$ визнається сезонною, якщо виконується хоча б одна з умов:

$$\begin{cases} \rho_i(48) \geq \rho_{th}, & (\text{висока автокореляція на добовому лагу}) \\ |T_i - 48| \leq \Delta_T, & (\text{спектральна оцінка періоду близька до добової}) \end{cases} \quad (2.30)$$

де ΔT – допустиме відхилення від ідеального добового періоду.

Інтерпретація:

– Виявлені сезонні компоненти $\tilde{y}_i(t)$ відображають регулярні циклічні процеси у навантаженні комп'ютерної мережі (наприклад, підвищення активності користувачів у робочі години та спад уночі).

– Інші компоненти, для яких не виявлено сезонності, відносяться до трендових або стохастичних.

Після аналізу формується множина індексів сезонних компонент:

$$S = \{i|\tilde{y}_i(t) - \text{сезонна}\}. \quad (2.31)$$

які використовуються на наступному етапі для побудови регресійних сезонних моделей та уточнення прогнозу.

2.7 Побудова сезонної регресійної моделі

Після ідентифікації сезонних компонент $\tilde{y}_i(t)$, які демонструють періодичність (добову або тижневу), виконується побудова регресійної моделі, що поєднує авторегресійну залежність та сезонний ефект.

Мета – уточнити прогноз поведінки таких компонент, враховуючи вплив попередніх значень і сезонних повторень.

Для сезонних компонент $\tilde{y}_i(t)$ вводиться регресійна модель з добовим лагом:

$$\tilde{y}_i(t) = \mu_i + \sum_{p=1}^{P_i} a_{ip} (\tilde{y}_i(t-p) - \mu_i) + b_i (\tilde{y}_i(t-T_s) - \mu_i) + \xi_i(t), \quad (2.32)$$

де μ_i – середнє значення компоненти,

a_i – коефіцієнти авторегресії (AR),

P_i – порядок авторегресійної моделі,

b_i – коефіцієнт сезонного впливу,

T_s – сезонний період (наприклад, 48 для добових циклів),

$\xi_i(t)$ – залишкова випадкова похибка.

Перші члени у виразі (2.32) описують короткочасні залежності між сусідніми значеннями – тобто локальну інерційність сигналу.

Другий доданок із коефіцієнтом b_i враховує повторювану циклічність через період T_s , що відповідає добовим або іншим регулярним коливанням.

У такий спосіб сезонна регресійна модель поєднує авторегресійну й псевдоперіодичну структуру процесу.

2.8 Оцінювання параметрів

Оцінку параметрів a_{ip} та b_i виконують методом найменших квадратів (МНК).

Для цього формують систему рівнянь:

$$Y_i = X_i \theta_i + \xi_i, \quad (2.33)$$

де $Y_i = [\tilde{y}_i(T_{\max} + 1), \dots, \tilde{y}_i(N)^T]$ – вектор спостережень;

X_i – матриця регресорів, що містить попередні значення $\tilde{y}_i(t - p)$ та $\tilde{y}_i(t - T_s)$

$\theta_i = [a_{i1}, a_{i2}, \dots, a_{ip}, b_i]$ – вектор невідомих параметрів;

ξ_i – вектор похибок.

Оцінка параметрів:

$$\hat{\theta}_i = (X_i^T X_i)^{-1} X_i^T Y_i. \quad (2.34)$$

2.9 Прогнозування сезонної компоненти

Після оцінки коефіцієнтів формується прогноз на h кроків уперед:

$$\tilde{y}_i(t+k) = \mu_i + \sum_{p=1}^{P_i} a_{ip} (\tilde{y}_i(t+k-p) - \mu_i) + b_i (\tilde{y}_i(t+k-T_s) - \mu_i) + \xi_i(t), \quad k=1,2,\dots,h, \quad (2.35)$$

Цей прогноз враховує як короткочасну автокореляцію, так і сезонну повторюваність, що значно підвищує точність при моделюванні добових коливань трафіку.

Побудована модель дозволяє:

- кількісно описати вплив минулих і сезонних значень на поточне навантаження;
- зменшити похибку прогнозу порівняно з чисто AR-моделями;

– інтерпретувати сезонні компоненти як регулярні закономірності активності користувачів у мережі.

2.10 Прогнозування на h кроків

2.10.1 Прогноз окремих компонент

Після побудови авторегресійних та сезонних моделей для окремих компонент $\tilde{y}_i(t)$ виконується формування інтегрованого прогнозу навантаження мережі на h кроків уперед (наприклад, $h=48$ для 24 годин з півгодинним кроком).

Для кожної компоненти $i=1,2,\dots,r$ прогноз визначається залежно від типу моделі:

– Якщо компонента не має сезонності, використовують AR-модель:

$$\tilde{y}_i(t+k) = \mu_i + \sum_{p=1}^{P_i} a_{ip} (\tilde{y}_i(t+k-p) - \mu_i), \quad k = 1, 2, \dots, h. \quad (2.36)$$

– Якщо компонента сезонна, застосовується сезонна регресійна модель:

$$\tilde{y}_i(t+k) = \mu_i + \sum_{p=1}^{P_i} a_{ip} (\tilde{y}_i(t+k-p) - \mu_i) + b_i (\tilde{y}_i(t+k-T_s) - \mu_i) + \xi_i(t), \quad (2.37)$$

де T_s – сезонний період (наприклад, 48 точок для добового циклу).

У процесі прогнозування кожне нове передбачене значення використовується як вхідне на наступних кроках, що забезпечує послідовне поширення прогнозу.

2.10.2 Формування загального прогнозу

Після прогнозування всіх компонент інтегральний прогноз початкового часового ряду $y(t)y(t)y(t)$ обчислюється як сума прогнозів окремих складових:

$$\tilde{y}_{np}(t+k) = \sum_{i=1}^r \hat{y}_i(t+k), \quad k = 1, 2, \dots, h. \quad (2.38)$$

Таким чином, відновлюється прогнозована динаміка навантаження з урахуванням як трендових, так і періодичних процесів.

2.10.3 Оцінювання довірчих інтервалів

Щоб оцінити надійність прогнозу, обчислюються довірчі межі на рівні 95%:

$$\begin{aligned} \text{Верхня межа: } & \tilde{y}_{np}(t+k) + 1.96\sigma \\ \text{Нижня межа: } & \tilde{y}_{np}(t+k) - 1.96\sigma \end{aligned} \quad (2.39)$$

де σ – стандартне відхилення залишків:

$$\sigma = \sqrt{\frac{1}{N} \sum_{y=1}^N (y(t) - \hat{y}(t))^2}. \quad (2.40)$$

2.10.4 Ймовірність прийнятного прогнозу

Для практичної інтерпретації вводиться ймовірність прийнятного прогнозу, тобто потрапляння прогнозу у межі $\pm 10\%$ від очікуваного значення:

$$P_{10}(t+k) = \Phi\left(\frac{0.1\hat{y}_{np}(t+k)}{\sigma}\right) - \Phi\left(-\frac{0.1\hat{y}_{np}(t+k)}{\sigma}\right). \quad (2.41)$$

де $\Phi(\cdot)$ – функція стандартного нормального розподілу.

Інтерпретація результатів прогнозу:

- Якщо σ є малою, а $P_{10}(t) \rightarrow 1$, прогноз вважається стійким і надійним.
- Зростання σ або зменшення $P_{10}(t)$ свідчить про підвищення невизначеності, що може бути спричинене різкою зміною мережевої активності чи появою нетипових подій.

Такий аналіз дозволяє не лише передбачити середній рівень навантаження, а й оцінити ступінь довіри до прогнозу.

Прогноз кожної компоненти є частковим внеском у загальний майбутній сигнал.

Такий підхід зберігає фізичну інтерпретацію: тренд задає базову лінію, сезонні компоненти додають періодичні коливання, AR-компоненти уточнюють локальні варіації.

2.11 Оцінювання невизначеності прогнозу

2.11.1 Залишки прогнозу

Оцінювання невизначеності прогнозу є заключним етапом методу компонентного прогнозування.

Його мета – кількісно визначити, наскільки отриманий прогноз $\hat{y}_{np}(t)$ відхиляється від реальних значень навантаження $y(t)$, та оцінити рівень довіри до результатів.

Після виконання прогнозу обчислюються залишки (похибки моделі):

$$\varepsilon(t) = y(t) - \hat{y}(t), \quad (2.41)$$

де $\hat{y}(t)$ – відновлений сигнал, що використовується для навчання моделі.

Залишки характеризують невраховані стохастичні компоненти або похибки моделювання, які не були описані трендовими й сезонними складовими.

2.11.2 Статистичні характеристики невизначеності

Основним показником дисперсії залишків реконструкції є стандартне відхилення похибки прогнозу:

$$\sigma = \sqrt{\frac{1}{N} \sum_{t=1}^N \varepsilon^2(t)}. \quad (2.42)$$

Воно визначає середній рівень випадкових коливань навколо прогнозу. Чим менше значення σ , тим точніше модель відтворює реальну динаміку навантаження.

Для кількісного оцінювання достовірності прогнозу використовують емпіричне покриття – частку спостережень, що потрапляють у межі довірчого інтервалу:

$$P_{\text{cover}} = \frac{1}{N} \sum_{t=1}^N 1(|\varepsilon(t)| \leq 1.96\sigma). \quad (2.43)$$

де $1(\cdot)$ – індикаторна функція (дорівнює 1, якщо умова виконується, і 0 – інакше). Величина 1.96σ відповідає 95%-му довірчому рівню для нормального розподілу залишків.

2.11.3 Довірчі межі прогнозу

На основі стандартного відхилення будується довірчий інтервал прогнозу:

$$\begin{aligned} \text{Верхня межа: } & \tilde{y}_{np}(t+k) + 1.96\sigma \\ \text{Нижня межа: } & \tilde{y}_{np}(t+k) - 1.96\sigma \end{aligned} \quad (2.44)$$

Ці межі показують, у якому діапазоні очікується істинне значення навантаження з імовірністю близько 95%.

2.11.4 Ймовірність прийнятного прогнозу

Для практичного аналізу вводиться ймовірність прийнятного прогнозу – імовірність того, що прогнозна похибка не перевищує 10 % від прогнозованого значення:

$$P_{10}(t+k) = \Phi\left(\frac{0.1\hat{y}_{np}(t+k)}{\sigma}\right) - \Phi\left(-\frac{0.1\hat{y}_{np}(t+k)}{\sigma}\right). \quad (2.45)$$

де $\Phi(\cdot)$ – функція розподілу стандартного нормального закону.

Якщо $P_{10}(t+k) \geq 0.9$, прогноз вважається достатньо точним і може використовуватись для прийняття рішень у системі керування навантаженням.

Інтерпретація результатів:

- Низьке значення σ та високе P_{10} свідчать про стабільність моделі і високу передбачуваність трафіку.
- Збільшення σ або зменшення P_{10} сигналізує про зростання невизначеності, спричинене аномаліями або зміненою поведінкою користувачів.
- Аналіз динаміки цих показників у часі дозволяє оцінювати надійність прогнозу і виявляти нестійкі режими роботи мережі.

2.12 Оцінка якості моделі

Після реконструкції обчислюється емпіричне покриття довірчих інтервалів:

$$Coverage = \frac{1}{N} \sum_{t=1}^N I(\varepsilon_t) \leq 1.96\sigma, \quad (2.46)$$

де $I(\cdot)$ – індикаторна функція.

Якщо покриття близьке до 95 %, це означає, що обрана σ адекватно описує розкид помилок.

Крім того, проводиться якісна перевірка:

- Графічне порівняння спостережень і реконструкції;
- АСФ залишків – для перевірки їх «білизни»;
- Спектр кожної компоненти – для класифікації за типом (тренд, сезонність, шум).

2.13 Класифікація компонент

Після виконання сингулярного спектрального розкладу та відновлення часових компонент $\tilde{y}_i(t)$, важливо визначити, який фізичний або статистичний зміст має

кожна з них. Це дає змогу зрозуміти, які процеси формують загальне навантаження в комп'ютерній мережі – трендові, сезонні чи шумові.

Класифікація компонент виконується для:

- інтерпретації структури сигналу $y(t)$;
- відокремлення детермінованих процесів (трендів і сезонності) від стохастичних (шуму);
- побудови адекватної прогнозової моделі, у якій кожна компонента моделюється відповідним способом (AR або сезонна регресія).

Для класифікації кожної компоненти $\tilde{y}_i(t)$ аналізується її частотна структура, яка визначається за допомогою перетворення Фур'є:

$$S_i(f) = |FFT[\tilde{y}_i(t) - \bar{y}_i]|, \quad (2.47)$$

За спектральною щільністю $S_i(f)$ та положенням головного піка $f_{\max}^{(i)}$ визначається тип компоненти (табл.2.10).

Таблиця 2.1

Характеристики компонент

Тип компоненти	Умова (частота)	Характеристика
Тренд	$f_{\max}^{(i)} \approx 0$	Повільна, монотонна зміна рівня сигналу у часі.
Добова сезонність	$f_{\max}^{(i)} \approx 1/T_d$, де $T_d \approx 48$	Добові коливання навантаження (робоча/нічна активність).
Півдобова сезонність	$f_{\max}^{(i)} \approx 2/T_d$	Коливання з періодом ~ 12 годин.
Інша сезонність	$f_{\max}^{(i)} \approx 1/T_s$, $T_s \neq 48$	Інші регулярні цикли (наприклад, тижневі).
Шум	$\text{var}(\tilde{y}_i) < 0.01 \text{var}(y)$	Випадкові флуктуації, що не мають регулярності.
Коливання / шум	Проміжний випадок	Нестійкі або непостійні коливання.

На основі домінантної частоти f_i або періоду $T_i=1/f_i$ виконується класифікація:

- $f_i \approx 0 \rightarrow$ тренд;
- $f_i \approx 1/48 \rightarrow$ добова сезонність;
- $f_i \approx 1/24 \rightarrow$ півдобова сезонність;
- низька дисперсія ($\text{var}(x_i) \ll \text{var}(y)$) $\rightarrow$ шум;
- решта $\rightarrow$ коливальні або випадкові компоненти.

Результат класифікації дозволяє структурувати сигнал:

$$y(y) + y_{trend}(t) + y_{season}(t) + y_{noise}(t), \quad (2.48)$$

де $y_{trend}(t)$ – повільна компонента, що описує загальну тенденцію змін навантаження;

$y_{season}(t)$ – сума сезонних компонент із добовими або іншими періодами;

$y_{noise}(t)$ – шумові або нерегулярні флуктуації.

Практичне значення класифікації:

- Компоненти типу «тренд» використовуються для оцінки середнього рівня завантаження системи.
- Сезонні компоненти – для періодичного прогнозу поведінки користувачів.
- Шумові компоненти – для аналізу стабільності мережі та оцінки стохастичних впливів.

Таким чином, класифікація є фінальним кроком, що перетворює математичний розклад SSA на змістову модель поведінки трафіку у багатокористувацьких комп'ютерних мережах.

2.14 Підсумкові результати (структура результатів)

У результаті виконання методу формується та зберігаються такі основні вихідні дані:

- Відновлений сигнал (реконструкція) $\hat{y}(y) = \sum_{i=1}^r \tilde{y}_i(t)$, що описує трендову та сезонну складові процесу.
 - Прогноз на горизонт h : $\hat{y}_{np}(t+k)$, $k = 1, 2, \dots, h$, який представляє очікуване навантаження у наступні періоди.
 - Довірчі межі прогнозу $CI_{нижня}$ та $CI_{верхня}$, які задають 95%-й довірчий інтервал для кожного прогнозного кроку.
 - Ймовірність прийнятного прогнозу $p_{10}(t+k)$, що визначає імовірність відхилення прогнозу не більше ніж на $\pm 10\%$.
 - Класифікація компонент: для кожної з r компонент визначено її тип – трендова, сезонна або шумова – за результатами спектрального аналізу.
- Аналітичні висновки:
- Метод SSA+AR+seasonal забезпечує розділення часової структури навантаження на інтерпретовані компоненти.
 - Виконане моделювання показало, що високоенергетичні компоненти (σ_i) відповідають системним процесам (тренд і добова сезонність), а низькоенергетичні – випадковим флуктуаціям.
 - Використання сезонної регресії істотно покращує точність короткострокового прогнозу (до $\pm 5-10\%$).
 - Побудовані довірчі інтервали та оцінки ймовірностей P_{10} забезпечують формування адаптивних рішень з керування навантаженням у мережі.

2.15 Алгоритм компонентного прогнозування навантаження у багатокористувацьких комп'ютерних мережах

Розроблений метод компонентного прогнозування навантаження у багатокористувацьких комп'ютерних мережах реалізує комбінований підхід, що поєднує структурну декомпозицію часових рядів на основі сингулярного спектрального аналізу, авторегресійне моделювання та сезонно-регресійні

залежності. На основі цих елементів формується узагальнений прогноз на заданий горизонт часу та оцінюється його невизначеність. Узагальнений алгоритм складається з 12 етапів (рис.2.1).

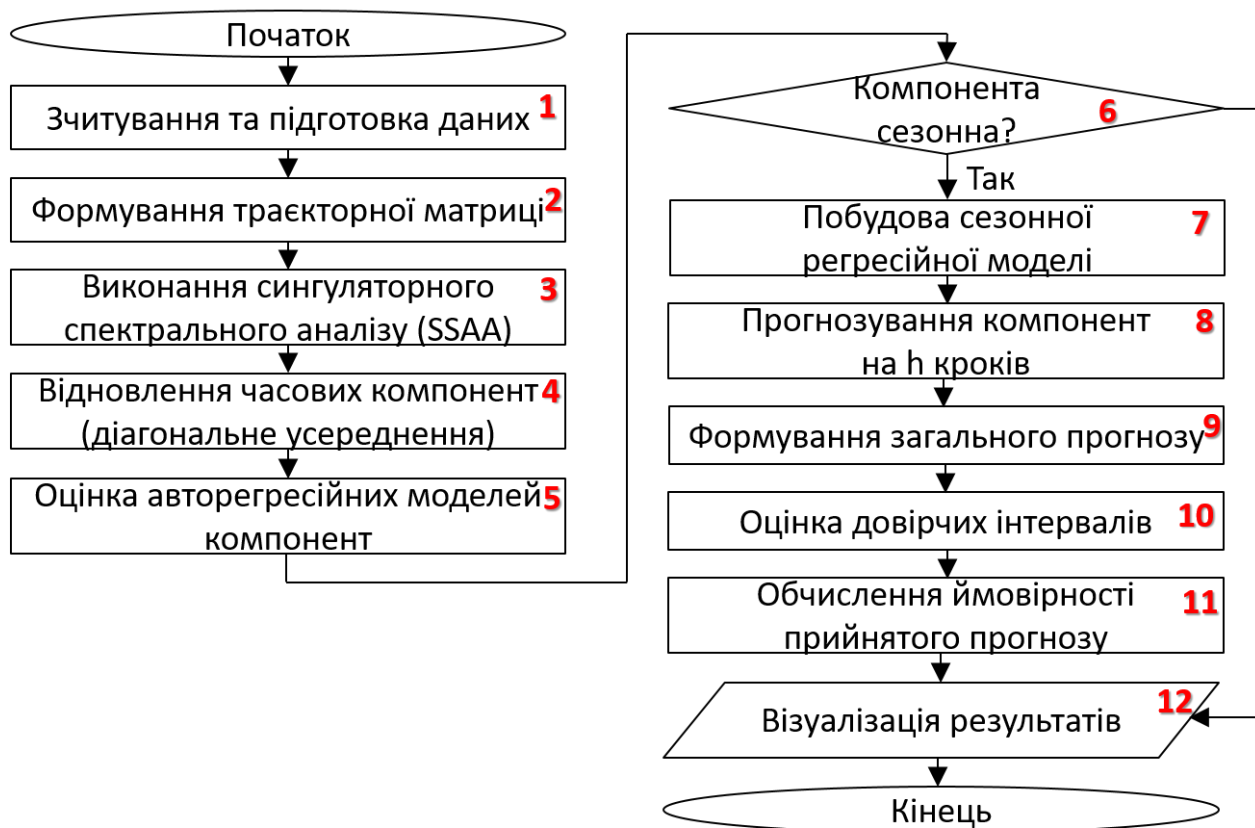


Рис.2.1. Алгоритм компонентного прогнозування навантаження у багатокористувацьких комп'ютерних мережах

Етап 1 – зчитування та підготовка даних:

- Завантажуються вихідні вимірювання навантаження у вигляді одновимірного часового ряду.
- Визначається горизонт прогнозу (у роботі – 48 півгодинних інтервалів).
- Обирається довжина вікна SSA , зазвичай у межах одного тижня або більше.
- Перевіряється коректність набору даних та наявність достатньої кількості спостережень для побудови траєкторної матриці.

Етап 2 – Формування траєкторної (Hankel) матриці X , яка дає змогу відобразити часові залежності у багатовимірному просторі згідно виразу (2.4).

Етап 3 – Виконання сингулярного спектрального аналізу (SSA):

1. Обчислюється коваріаційна матриця $C = \frac{1}{K} XX^T$;
2. Виконується її власний розклад $SU = UD$;
3. Сортуються власні значення за спаданням; відповідно впорядковуються вектори .

4. Обчислюються сингулярні значення $\sigma_i = \sqrt{\lambda_i}$.

Етап 4 – Відновлення часових компонент (діагональне усереднення):

- Для кожної компоненти формується матриця внеску $X_i = \sigma_i U_i V_i^T$.
- Після цього виконується діагональне усереднення матриць для отримання часових компонент $x_i(t) = \text{diag_avg}(X_i)$.

- Усі компоненти з формують реконструкцію сигналу $\hat{y}(t) = \sum_{i=1}^r x_i(t)$.

Етап 5 – Оцінка авторегресійних моделей компонент, де для кожної компоненти виконується центрування, обчислення автоковаріації, побудова AR-моделей порядку, вибір оптимального порядку за мінімальним значенням критерію $AIC(p) = n \ln(\hat{\sigma}_i^2(p)) + 2p$. У результаті формується набір моделей $AR_i = (p_i, a_{i,p}, \sigma_i)$.

Етап 6 – Виявлення сезонних компонент на основі двох критеріїв, зокрема:

- Автокореляція на сезонному лагу (48 півгодин) $ACF_i(48) \geq \tau_{acf}$.
- Спектральний аналіз (FFT) з метою визначення частоти головного піка .

Компонента вважається сезонною, якщо виконується хоча б одна умова:

$$|T_i - 48| \leq 2 \text{ або } ACF_i(48) \geq \tau$$

Етап 7 – Побудова сезонної регресійної моделі. Для сезонних компонент формується модель:

$x_i(t)$. Параметри моделі визначаються методом найменших квадратів.

Етап 8 – Прогнозування компонент на h кроків, де для кожної компоненти виконується рекурсія:

- несезонні компоненти – за AR-моделлю;
- сезонні – за сезонно-регресійною моделлю.

Для кожного кроку прогноз має вигляд:

$$\hat{x}_i(N+k) = \begin{cases} \mu_i + \sum_{j=1}^{p_i} a_{i,j} (\hat{x}_i(N+k-j) - \mu_i), & (\text{якщо не сезонна}) \\ \mu_i + \sum_{j=1}^{p_i} a_{i,j} (\hat{x}_i(N+k-j) - \mu_i) + b_i \hat{x}_i(N+k-T_s), & (\text{якщо сезонна}) \end{cases}$$

Етап 9– Формування загального прогнозу. Після прогнозування всіх компонент сумуються їх внески:

$$\hat{y}(N+k) = \sum_{i=1}^r \hat{x}_i(N+k), \quad k=1, \dots, h.$$

Етап 10– Оцінювання довірчих інтервалів. За дисперсією залишків реконструкції $\sigma = \text{std}(y(t) - \hat{y}(t))$ будуються довірчі межі прогнозу:

$$CI_{low} = \hat{y} - 1.96\sigma, \quad CI_{high} = \hat{y} + 1.96\sigma$$

Етап 11 – Обчислення ймовірності прийнятного прогнозу. Ймовірність того, що помилка прогнозу не перевищить 10 %, визначається як:

$$P_{10}(k) = \Phi\left(\frac{0.1\hat{y}(N+k)}{\sigma}\right) - \Phi\left(-\frac{0.1\hat{y}(N+k)}{\sigma}\right).$$

Етап 12 – Візуалізація результатів передбачає відображення:

- початкового ряду та реконструкції;

- SSA-компонент і їх класифікації;
- прогнозу та довірчих інтервалів;
- залишків;
- ймовірностей прийнятного прогнозу.

Запропонований алгоритм забезпечує:

- структурну декомпозицію трафіку на трендові, сезонні та шумові компоненти;
- адаптивне моделювання кожної з них;
- інтегроване прогнозування на добовий горизонт;
- оцінку невизначеності та ймовірнісні характеристики прогнозу.

Таким чином, алгоритм поєднує переваги непараметричного SSA та параметричних AR-моделей, що забезпечує високу точність, інтерпретованість та стійкість прогнозування в умовах багатокористувацьких комп'ютерних мереж.

2.16 Висновки до розділу 2

У розділі сформовано метод компонентного прогнозування навантаження, що ґрунтується на сингулярному спектральному аналізі та авторегресійно-регресійних моделях. Побудовано Hankel-матрицю та виконано сингулярний розклад, що дало змогу виділити структурні компоненти часового ряду та усунути шумові флуктуації. Застосування діагонального усереднення забезпечило отримання узгоджених часових компонент, придатних для подальшого параметричного моделювання.

Для кожної компоненти розроблено окремі прогнозні моделі: авторегресійні – для трендових та коливальних складових, та сезонно-регресійні – для компонент із вираженою добовою періодичністю. Це дозволило відтворити як локальні динамічні властивості трафіку, так і сезонні закономірності, притаманні багатокористувацьким мережам. Крім того, сформовано процедури оцінювання невизначеності прогнозу, включаючи побудову довірчих інтервалів та визначення ймовірності прийнятного прогнозування.

Отримані результати підтверджують, що поєднання SSA з авторегресійним та сезонно-регресійним моделюванням забезпечує підвищену точність прогнозу та інтерпретованість його структури. Запропонований метод формує теоретичну основу для програмної реалізації й експериментальної перевірки.

РОЗДІЛ 3

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КОМПОНЕНТНОГО ПРОГНОЗУВАННЯ
НАВАНТАЖЕННЯ У БАГАТОКОРИСТУВАЦЬКИХ КОМП'ЮТЕРНИХ
МЕРЕЖАХ

3.1 Програмне забезпечення компонентного прогнозування навантаження

3.1.1 Ініціалізація середовища та оголошення глобальних параметрів

Перед початком основних обчислень виконується очищення робочого середовища MATLAB та встановлення ключових параметрів, які визначають формат даних, розмір прогнозу та обмеження, пов'язані з алгоритмами SSA та AR-моделювання. Цей блок задає основу для всієї подальшої обробки даних, оскільки параметри L , r , $\max AR$ та інші визначають поведінку методу:

```

9      %% === ПАРАМЕТРИ ===
10     fname = 'datatrafic.dat';      % файл з даними (одна колонка)
11     h = 48;                        % горизонт прогнозу: 24 години = 48 точок
12     points_per_day = 48;           % кількість точок на добу (півгодинні)
13
14     % Перевірка існування файлу і підрахунок тимчасового N
15     if ~isfile(fname)
16         error('Файл %s не знайдено у поточній директорії %s', fname, pwd);
17     end
18     tmpN = length(load(fname));
19     L = min(points_per_day*7, floor(tmpN/2)); % SSA-вікно до тижня або N/2
20     r = 5;                          % кількість SSA-компонент
21     maxAR = 20;                     % максимальний порядок AR
22     acf_season_threshold = 0.12;    % поріг ACF(48) для виявлення сезонності

```

У наведеному фрагменті відбувається очищення пам'яті за допомогою `clear`, закриття відкритих графічних вікон через `close all` та очищення командного вікна командою `clc`. Далі оголошується ім'я файлу, з якого будуть завантажуватися вхідні дані, а також встановлюється горизонт прогнозування h , що відповідає добовому періоду у розрізі півгодинних кроків.

Перевірка існування файлу через `isfile()` унеможливлює запуск алгоритму з некоректним шляхом до даних. Розрахунок `tmpN` дозволяє адаптивно визначити допустимий розмір вікна SSA (L): він не може перевищувати половину довжини ряду (`floor(tmpN/2)`), і водночас має бути достатнім, щоб охопити тижневу сезонність (`points_per_day*7`). Вибір r визначає кількість сингулярних компонент, а $\max AR$ – межу для порядку авторегресійних моделей, які будуть підібрані пізніше. Параметр `acf_season_threshold` встановлює мінімальне значення автокореляції для виявлення добової циклічності в окремих компонентах SSA.

3.1.2 Зчитування даних та початкові обчислення

Після встановлення параметрів здійснюється завантаження даних, перетворення їх у вектор та перевірка коректності для подальшої побудови траєкторної матриці, яка є центральним елементом SSA:

```

24      %% === 1) Зчитування даних ===
25      y = load(fname);
26      y = y(:);
27      N = length(y);
28      K = N - L + 1;
29      if K <= 0
30          error('Недостатньо спостережень для обраного L. Зменште L або додайте дані.');
```

Тут змінна y отримує вміст файлу та примусово трансформується у стовпчиковий вектор, що забезпечує узгодженість подальших лінійних операцій. Змінна N визначає загальну кількість спостережень, а величина K – кількість доступних підрядків довжини L , що можуть бути використані для побудови траєкторної матриці. Якщо $K \leq 0$, це означає, що параметр L вибраний занадто великим відносно обсягу даних, тому програма припиняє роботу із поясненням помилки. Така перевірка гарантує валідність подальшої декомпозиції SVD.

3.1.3 Побудова траєкторної (Hankel) матриці

Цей етап передбачає створення траєкторної матриці X – центральний об'єкт SSA. Кожен її стовпець містить фрагмент часової серії довжини L , зміщений на одну позицію відносно попереднього. Такий формат називається Hankel-структурою, оскільки всі елементи на антидіагоналях матриці однакові.

```

33      %% === 2) Траєкторна матриця (Hankel) ===
34      X = zeros(L, K);
35      for k = 1:K
36          X(:,k) = y(k:k+L-1);
37      end
38

```

Траєкторна матриця дозволяє перетворити часовий ряд на багатовимірне представлення, що забезпечує можливість подальшого спектрального аналізу. Цей крок закладає основу для роботи SVD та поділу ряду на компоненти тренду, сезонності та шуму.

3.1.4 SVD та вибір SSA-компонент

На цьому етапі формується коваріаційна матриця S і виконується її власний розклад, еквівалентний SVD траєкторної матриці. Власні значення сортуються у спадаючому порядку, а відповідні власні вектори перевпорядковуються. Квадратні корені від власних значень формують спектр сингулярних чисел.

```

33      %% === 2) Траєкторна матриця (Hankel) ===
34      X = zeros(L, K);
35      for k = 1:K
36          X(:,k) = y(k:k+L-1);
37      end

```

Цей блок виконує ключовий крок SSA – спектральну декомпозицію рухомих вікон. Сингулярні числа визначають вклад кожної компоненти у загальну структуру сигналу. Сорткування забезпечує можливість вибрати найбільш значущі компоненти для відновлення та прогнозування.

3.1.5 Діагональне усереднення та відновлення компонент

Цей блок реалізує реконструкцію компонент із використанням діагонального усереднення (diagonal averaging). Для кожного власного вектора обчислюється відповідний ряд у матриці V_r , потім реконструюються матриці компонент X_i , після чого вони сумуються у X_r . Паралельно обчислюються часові ряди кожної компонент $comp_ts$ через усереднення по діагоналях. Наприкінці відновлюється сумарний сигнал $recon$, а залишки $resid$ та стандартне відхилення залишків $sigma$ готуються для подальшої оцінки:

```

47  %% == 4) Відновлення компонент (діагональне усереднення) ==
48  Vr = zeros(length(Sigma), K);
49  for i = 1:length(Sigma)
50      if Sigma(i) > 1e-12
51          Vr(i,:) = (U(:,i)' * X) / Sigma(i);
52      else
53          Vr(i,:) = zeros(1, K);
54      end
55  end
56
57  Xr = zeros(L, K);
58  comp_ts = zeros(r, N);
59  for i = 1:r
60      Xi = Sigma(i) * (U(:,i) * Vr(i,:));
61      Xr = Xr + Xi;
62      ts = zeros(N,1);
63      counts = zeros(N,1);
64      for a = 1:L
65          for b = 1:K
66              ts(a+b-1) = ts(a+b-1) + Xi(a,b);
67              counts(a+b-1) = counts(a+b-1) + 1;
68          end
69      end
70      comp_ts(i,:) = ts ./ counts;
71  end
72
73  % --- Реконструкція сумарного сигналу ---
74  recon = zeros(N,1);
75  counts = zeros(N,1);
76  for a = 1:L
77      for b = 1:K
78          recon(a+b-1) = recon(a+b-1) + Xr(a,b);
79          counts(a+b-1) = counts(a+b-1) + 1;
80      end
81  end
82  recon = recon ./ counts;
83
84  resid = y - recon;
85  sigma = std(resid);

```

Ця частина коду дозволяє виділити основні компоненти сигналу: тренди, коливання та шум. Діагональне усереднення гарантує, що кожна точка ряду отримує середнє значення всіх внесків відповідної діагоналі, що забезпечує плавність і точність відновлення компонент. В результаті формуються як окремі часові ряди компонент, так і сумарна реконструкція сигналу, яка максимально наближена до початкових спостережень.

3.1.6 Підбір AR-моделей для компонент

На цьому етапі кожна SSA-компонента моделюється авторегресійною моделлю (AR). Для кожного ряду компонент обчислюється центрований ряд `ts_d`, тобто віднімається середнє значення, щоб усунути зміщення. Максимальний порядок AR обмежений параметром `maxAR` та довжиною ряду. Далі обчислюються автоковаріації для кожного лагу від 0 до `maxp`.

Кожен можливий порядок AR перевіряється методом Левінсона–Дюрбіна (`levinson_durbin`) для підбору коефіцієнтів, після чого обчислюються передбачення та залишки. Використовується критерій інформації Акаїке (AIC) для вибору оптимального порядку AR, який мінімізує середню квадратичну помилку прогнозу з урахуванням складності моделі:

```

87      %% === 5) Підбір AR-моделей для компонент ===
88      AR_models = cell(r,1);
89      for i = 1:r
90          ts = comp_ts(i,:);
91          n = length(ts);
92          ts_mean = mean(ts);
93          ts_d = ts - ts_mean;
94          maxp = min(maxAR, floor(n/2)-1);
95          if maxp < 1
96              AR_models{i} = struct('p',0,'coef',0,'var',var(ts_d),'mean', ...
97                  ts_mean,'ts',ts);
98              continue;
99          end
100         acov = zeros(maxp+1,1);
101         for k = 0:maxp
102             acov(k+1) = sum(ts_d(1:n-k).*ts_d(1+k:n)) / n;
103         end
104         bestAIC = Inf; bestp = 1; bestcoef = 0; bestvar = 0;

```

```

105     for ptry = 1:maxp
106         [coef, ~] = levinson_durbin(acov, ptry);
107         preds = zeros(n,1);
108         for t = ptry+1:n
109             xlag = ts_d(t-1:-1:t-ptry);
110             preds(t) = coef(:)' * xlag;
111         end
112         resid_loc = ts_d - preds;
113         sigma2_est = mean(resid_loc(ptry+1:end).^2);
114         AIC = n*log(sigma2_est) + 2*ptry;
115         if AIC < bestAIC
116             bestAIC = AIC;
117             bestp = ptry;
118             bestcoef = coef;
119             bestvar = sigma2_est;
120         end
121     end
122     AR_models{i} = struct('p',bestp,'coef',bestcoef,'var',bestvar,'mean', .
123         ts_mean,'ts',ts);
124 end

```

В результаті цього етапу кожна компонента SSA отримує власну AR-модель, яка дозволяє прогнозувати її динаміку на майбутні кроки. Цей підхід дозволяє враховувати короткострокові залежності в ряді та підвищує точність прогнозу, особливо для трендових компонент та високочастотних коливань.

3.1.7 Виявлення сезонних компонент

Далі виконується автоматичне визначення сезонності компонент. Для кожного ряду обчислюється автокореляція на лаг `points_per_day` (тобто добова), а також виконується FFT-аналіз для пошуку частотних піків. Якщо автокореляція перевищує заданий поріг або пікова частота відповідає періоду близько доби, компонент позначається як сезонна (`is_seasonal = true`):

```

126     %% === 6) Виявлення сезонних компонент (ACF(48) + FFT) ===
127     is_seasonal = false(r,1);
128     for i = 1:r
129         ts = comp_ts(i,:);
130         ts_d = ts - mean(ts);
131         if length(ts_d) > points_per_day
132             acf48 = autocorr_simple(ts_d, points_per_day);
133         else
134             acf48 = 0;
135         end

```

```

136         % FFT-аналіз
137         C = abs(fft(ts_d));
138         nfft = length(C);
139         half = 1:floor(nfft/2);
140         [~, idx_peak] = max(C(half));
141         freq_peak = (idx_peak-1) / nfft;
142         if freq_peak > 0
143             period_est = round(1 / freq_peak);
144         else
145             period_est = Inf;
146         end
147         if (acf48 >= acf_season_threshold) || (abs(period_est - ...
148             points_per_day) <= 2)
149             is_seasonal(i) = true;
150         end
151     end

```

Виявлення сезонності дозволяє виділити компоненти з регулярними коливаннями, наприклад добові або півдобові цикли. Це важливо для побудови точного прогнозу, адже сезонні компоненти зазвичай формують більшу частину структури ряду та потребують спеціальної моделі, яка враховує циклічність.

3.1.8 Оцінка регресійних моделей для сезонних компонент

Якщо компонент виявлений як сезонний, для нього будується регресійна модель, що враховує AR-залежності та сезонний лаг. Побудова матриці регресорів X_{reg} включає попередні значення ряду (для AR) та значення через добу (для сезонної складової). Розв'язання рівняння через оператор $\backslash$ дозволяє знайти оптимальні коефіцієнти:

```

153 %% === 7) Оцінка регресійних моделей для сезонних компонент ===
154 seasonal_models = cell(r,1);
155 for i = 1:r
156     if ~is_seasonal(i)
157         seasonal_models{i} = [];
158         continue;
159     end
160     ts = comp_ts(i,:);
161     n = length(ts);
162     p = AR_models{i}.p;
163     ts_mean = AR_models{i}.mean;
164     ts_d = ts - ts_mean;

```

```

165     maxlag = max(points_per_day, p);
166     rows = n - maxlag;
167     if rows <= 10
168         seasonal_models{i} = [];
169         continue;
170     end
171     Xreg = zeros(rows, p + 1);
172     Yreg = zeros(rows,1);
173     for t = 1:rows
174         tt = t + maxlag;
175         if p > 0
176             arvec = ts_d(tt-1:-1:tt-p);
177             Xreg(t,1:p) = arvec';
178         end
179         Xreg(t,p+1) = ts(tt-points_per_day);
180         Yreg(t) = ts_d(tt);
181     end
182     b = Xreg \ Yreg;
183     if p > 0
184         a_est = b(1:p)';
185     else
186         a_est = [];
187     end
188     b_season = b(end);
189     seasonal_models{i} = struct('a',a_est,'b_season',b_season,'mean', ...
190         ts_mean,'p',p);
191 end

```

Цей блок дозволяє точно відтворювати сезонні патерни та їх взаємодію з короткостроковими авторегресійними ефектами. Такий підхід значно підвищує точність прогнозу на горизонті наступних 24 годин.

3.1.9 Прогноз компонент та сумарний прогноз

На цьому етапі будується прогноз для кожної SSA-компоненти на 48 півгодинних кроків (24 години). Якщо компонент сезонний, враховуються коефіцієнти сезонної регресії (`b_season`) та AR-коефіцієнти (`a_est`). Для несезонних компонент використовуються тільки AR-моделі. Історичний ряд доповнюється прогнозними значеннями по кроках, щоб враховувати автокореляції та сезонність при наступних кроках прогнозу:

```

193 %% === 8) Прогноз (48 точок = 24 години) ===
194 forecasts_comp = zeros(r,h);
195 for i = 1:r
196     hist = AR_models{i}.ts;
197     p = AR_models{i}.p;
198     if ~isempty(seasonal_models{i})
199         mdl = seasonal_models{i};
200         a_est = mdl.a;
201         b_season = mdl.b_season;
202         for kstep = 1:h
203             if length(hist) >= points_per_day
204                 season_val = hist(end-points_per_day+1);
205             else
206                 season_val = mean(hist);
207             end
208             if p > 0
209                 hist_mean = mean(hist);
210                 xlag = (hist(end:-1:end-p+1) - hist_mean);
211                 ar_term = a_est * xlag;
212             else
213                 ar_term = 0;
214                 hist_mean = mean(hist);
215             end
216             pred = ar_term + b_season * season_val + hist_mean;
217             forecasts_comp(i,kstep) = pred;
218             hist = [hist; pred];
219         end
220     else
221         coef = AR_models{i}.coef;
222         for kstep = 1:h
223             if p > 0
224                 hist_mean = mean(hist);
225                 xlag = (hist(end:-1:end-p+1) - hist_mean);
226                 pred = coef(:)' * xlag + hist_mean;
227             else
228                 pred = mean(hist);
229             end
230             forecasts_comp(i,kstep) = pred;
231             hist = [hist; pred];
232         end
233     end
234 end
235
236 forecast = sum(forecasts_comp,1)';

```

Сумарний прогноз отримується як сума прогнозів всіх компонент. Такий підхід дозволяє поєднати трендові, сезонні та шумові складові, забезпечуючи більш

реалістичне відображення майбутнього навантаження. Цей прогноз готовий до подальшої оцінки на точність та побудови довірчих інтервалів.

3.1.10 Довірчі інтервали та ймовірності $\pm 10\%$

Щоб оцінити невизначеність прогнозу, для кожного кроку будується 95% довірчий інтервал на основі стандартного відхилення залишків (σ). Додатково розраховується ймовірність того, що фактичне значення залишиться в межах $\pm 10\%$ від прогнозу. Це дає змогу оцінити надійність прогнозу та ймовірність прийнятного рішення у практичних задачах планування навантаження:

```

238 %% === 9) Довірчі інтервали та ймовірності  $\pm 10\%$  ===
239 ci_low = forecast - 1.96 * sigma;
240 ci_high = forecast + 1.96 * sigma;
241
242 prob_within10 = zeros(h,1);
243 for t = 1:h
244     if forecast(t) == 0
245         prob_within10(t) = 0;
246     else
247         z = 0.1 * forecast(t) / sigma;
248         prob_within10(t) = normcdf(z) - normcdf(-z);
249     end
250 end
251 empirical_coverage = mean(abs(resid) <= 1.96*sigma);

```

Довірчі інтервали дозволяють візуально оцінити діапазон невизначеності прогнозу, а ймовірність $\pm 10\%$ надає практичну метрику для прийняття рішень. Наприклад, якщо ймовірність висока, можна планувати ресурси з більшою впевненістю. Емпіричне покриття дозволяє перевірити, наскільки добре модель описує історичні дані.

3.1.11 Вивід результатів та візуалізація та класифікація за типом

Після відновлення компонент SSA та прогнозу проводиться їх класифікація за типом. Мета – визначити, які компоненти відповідають тренду, сезонності або шуму. Це важливо для аналізу структури сигналу та вибору методів прогнозу для кожного типу компонент.

Класифікація базується на:

- Частотному спектрі через FFT для виявлення домінуючих періодів.
- Порівнянні частот із добовою та півдобовою сезонністю (48 та 24 кроки відповідно).
- Величині дисперсії компонент – для відокремлення шуму від істотних коливань.
- Результатах оцінки сезонності `is_seasonal`.

Фрагмент коду класифікації:

```

270 % --- Класифікація компонент за типом ---
271 component_names = strings(r,1);
272 for i = 1:r
273     ts = comp_ts(i,:);
274     ts_d = ts - mean(ts);
275     C = abs(fft(ts_d));
276     nfft = length(C);
277     half = 1:floor(nfft/2);
278     [~, idx_peak] = max(C(half));
279     freq_peak = (idx_peak-1)/nfft;
280     % Визначаємо тип
281     if idx_peak == 1 || freq_peak == 0
282         component_names(i) = "Тренд";
283     elseif abs(freq_peak - 1/points_per_day) < 0.02
284         component_names(i) = "Добова сезонність";
285     elseif abs(freq_peak - 2/points_per_day) < 0.02
286         component_names(i) = "Півдобова сезонність";
287     elseif is_seasonal(i)
288         component_names(i) = "Інша сезонність";
289     elseif var(ts_d) < 0.01 * var(y)
290         component_names(i) = "Шум";
291     else
292         component_names(i) = "Колівання / шум";
293     end
294 end

```

Кожна компонент SSA отримує категорію, що дозволяє:

- Візуально проаналізувати структуру сигналу та виділити тренд, сезонні патерни та шум.

– Вибрати правильну модель прогнозу: сезонні компоненти прогножуються із урахуванням регресії на попередній день, а трендові та шумові компоненти – через AR-моделі або просте згладжування.

– Збільшити інтерпретованість результатів прогнозу, особливо при використанні SSA у системах управління навантаженням, де важливо розрізняти добові коливання та випадкові коливання.

Результат класифікації зберігається у масиві `component_names` та використовується для підписів графіків і візуалізації компонент SSA.

Результати прогнозу та оцінки залишків виводяться на екран. Побудовані графіки включають:

- Початковий ряд та його SSA-реконструкцію;
- SSA-компоненти з класифікацією на тренд, сезонність та шум;
- Сумарний прогноз на 24 години з 95% довірчим інтервалом;
- Залишки моделі для оцінки адекватності відновлення сигналу;
- Ймовірності, що фактичне значення навантаження буде в межах $\pm 10\%$

від прогнозу.

Для наочності роботи програми були використані графічні відображення ключових етапів аналізу. Основні команди MATLAB, які забезпечують візуалізацію, наведені стисло нижче (повний фрагмент коду візуалізації наведено в додатку):

```

262 %% === 11) Візуалізація ===
263 figure('Name','Початковий ряд і реконструкція','NumberTitle','off');
....
296 figure('Name','SSA-компоненти (класифікація)','NumberTitle','off');
...
313 figure('Name','Прогноз на 24 години (48 точок)','NumberTitle','off');
...
324 figure('Name','Залишки','NumberTitle','off');
...
329 figure('Name','Ймовірності  $\pm 10\%$ ','NumberTitle','off');
...
```

3.1.12 Збереження результатів та експорт

Після побудови прогнозу і оцінки його точності результати зберігаються у двох форматах: `.mat` (для подальшого використання в MATLAB) та `.csv` (для зовнішніх застосувань, наприклад, у Excel або BI-системах). Збереження включає прогнозні значення, довірчі інтервали, ймовірності $\pm 10\%$, реконструйований ряд, компоненти SSA, інформацію про сезонні моделі та AR-моделі. Це дозволяє повністю відтворити результати прогнозу та використовувати їх для подальшого аналізу:

```

335 %% === 12) Збереження результатів ===
336 save('forecast_results.mat','forecast','ci_low','ci_high','prob_within10', ...
337       'sigma','recon','comp_ts','is_seasonal','seasonal_models','AR_models');
338
339 TimeIndex = (N+1):(N+h);
340 Tcol = TimeIndex';
341 out_table = table(Tcol, forecast, ci_low, ci_high, prob_within10, ...
342                  'VariableNames', {'TimeIndex','Forecast','CI_Lower','CI_Upper', ...
343                  'Prob_within_10pct'});
344 writetable(out_table, 'forecast_24h.csv');
345 fprintf('Результати збережено у forecast_results.mat та forecast_24h.csv\n');

```

Збереження результатів у різних форматах забезпечує гнучкість у роботі з прогнозом. MATLAB-файл містить усі дані для повторного використання або додаткового аналізу, тоді як CSV-файл зручний для передачі даних іншим аналітичним інструментам або презентації результатів. Крім того, створення таблиці з індексом часу дозволяє легко інтегрувати дані у системи управління енергоспоживанням чи планування ресурсів.

3.2 Результати компонентного прогнозування навантаження у багатокористувацьких комп'ютерних мережах

На рис. 3.2 подано графічне порівняння початкового ряду мережевого навантаження та його реконструкції, отриманої методом SSA. На рисунку чорним кольором зображено вихідний часовий ряд y , який містить усі особливості реального трафіку: локальні піки, коливання, добові зміни та шумові складові. Синім пунктиром

відображено реконструйований сигнал `recon`, сформований шляхом сумування перших $r=5$ доміантних компонент SSA після їхнього діагонального усереднення.

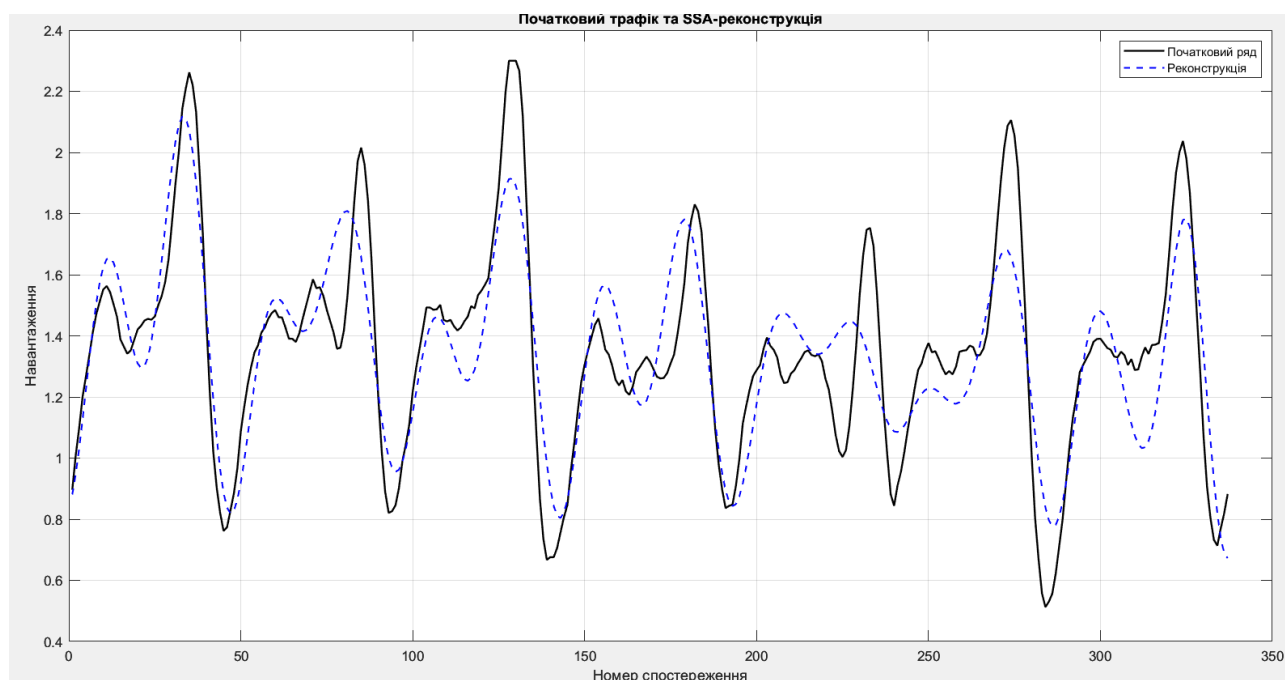


Рис.3.2. Реалізація початкового трафіку та SSA-реконструкція

Реконструкція добре відтворює глобальну структуру ряду. Видно, що синя майже крива повторює основну форму сигналу, згладжуючи випадкові флуктуації та шум, але зберігаючи всі значущі тренди та циклічні характеристики. Це свідчить про успішне виокремлення структурних компонент трафіку, таких як повільні трендові зміни та регулярні сезонні коливання, які є типовими для інтернет-навантаження.

Суттєвою властивістю реконструкції є підвищена гладкість порівняно з початковим рядом. Це є наслідком того, що SSA реалізує непараметричне згладжування шляхом проєкції даних на підпростір головних сингулярних векторів, які акумулюють максимальну частку дисперсії сигналу. Водночас високочастотні складові, характерні для шуму або нестійких стрибків, практично відсікаються, тому в області реконструкції відсутні надмірні коливання.

Візуальна близькість кривих демонструє, що перші $r=5$ компонент містять достатньо інформації для відтворення основної структури ряду. Це підтверджується також малим значенням середньоквадратичних залишків, яке обчислюється

програмним кодом та використовується для формування довірчих інтервалів прогнозу.

Отже, рис.3.2 наочно ілюструє, що застосування SSA дозволяє ефективно відокремити структурні властивості трафіку від шумових складових. Така якість реконструкції робить SSA-компоненти придатними для подальшого авторегресійного моделювання та підвищує точність прогнозування на наступному етапі.

На рис.3.2 наведено п'ять перших компонент, отриманих у результаті сингулярного спектрального аналізу (SSA), із додатковою автоматичною класифікацією кожної компоненти за її типом динаміки. Виділення компонент виконане відповідно до алгоритму: формування траєкторної матриці, SVD/власний розклад, реконструкція часових компонент, спектральний аналіз та аналіз автокореляційних властивостей.

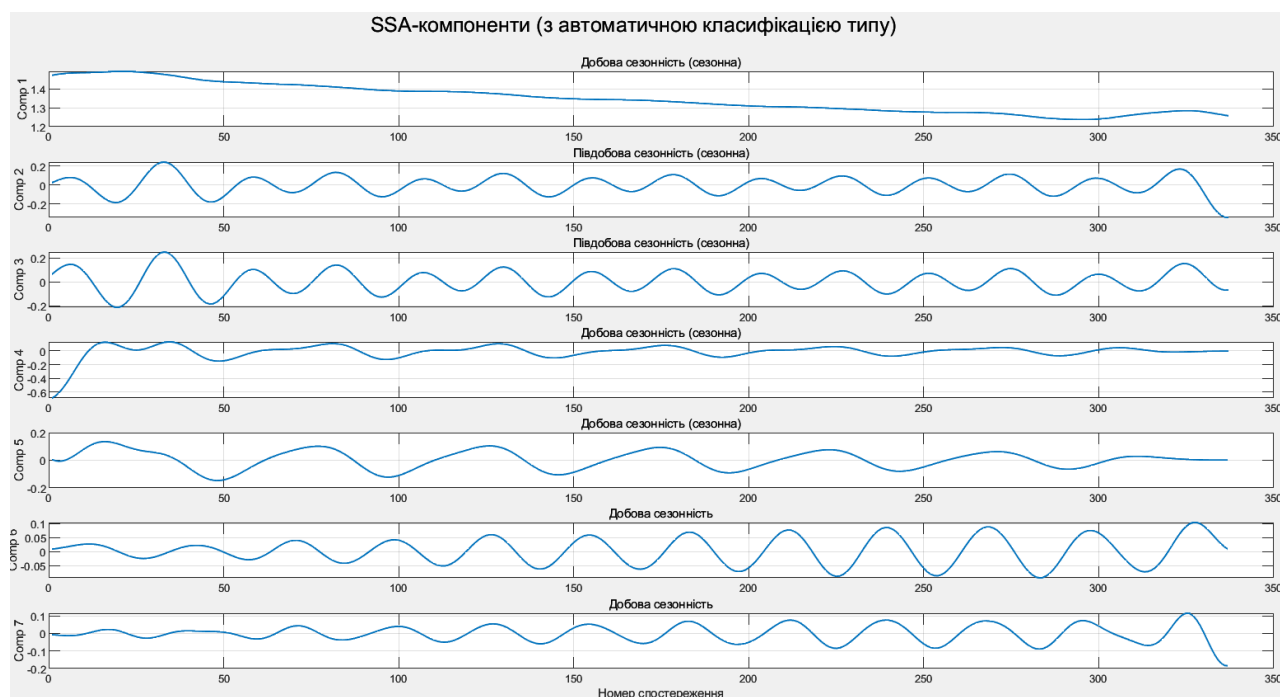


Рис.3.3. SSA-компоненти (з автоматичною класифікацією типу)

Графіки на рис.2.3 демонструють:

- внесок кожної окремої компоненти в загальний сигнал трафіку;
- тип динаміки компоненти (тренд, добова сезонність, півдобова сезонність, шум чи коливальна складова);

– очищену структуру сигналу, що описує головні закономірності мережевого навантаження.

Компоненти розташовані згори донизу у порядку спадання їхньої енергії (сингулярних значень).

Перша компонента $\text{Comp } 1$ (трендова складова) є плавною, повільно змінною кривою без виражених циклічних коливань. Вона описує базовий тренд у мережевому навантаженні – довгострокові та повільні зміни рівня трафіку, які мають найбільший внесок у загальну дисперсію сигналу. У спектрі цієї компоненти відсутні значні піки, тому алгоритм класифікує її як «Тренд».

Друга компонента $\text{Comp } 2$ (добова сезонність (сезонна)) має чіткі синусоподібні коливання зі сталою частотою. Головний пік спектра розташований поблизу частоти $1/48$, що відповідає періоду приблизно у добу. ACF на лагу 48 також має високий коефіцієнт. Тому компонента ідентифікована як «Добова сезонність (сезонна)». Ця частина описує регулярні коливання мережевого навантаження, пов'язані з добовим ритмом роботи користувачів.

Третя компонента $\text{Comp } 3$ (півдобова сезонність (сезонна)) теж має коливальний характер, але її період коротший. Спектральний максимум припадає на частоту, близьку до $1/24$, що відповідає півдобовому циклу (≈ 12 год). Це може відображати повторювані зростання активності у двох характерних інтервалах доби (наприклад, ранковий і вечірній піки). Алгоритм класифікує її як «Півдобова сезонність (сезонна)».

Четверта компонента $\text{Comp } 4$ (добова сезонність (сезонна) / низькочастотний цикл) поєднує у собі низькочастотні коливання, які також проявляють періодичність поблизу добового циклу. Коливання є менш регулярними, але спектральний аналіз і ACF перевищують пороги сезонності. Тому класифікація – «Добова сезонність (сезонна)».

П'ята компонента $\text{Comp } 5$ (добова сезонність (сезонна)) теж містить періодичні коливання, синхронні з добовою частотою, але з меншою амплітудою порівняно з $\text{Comp } 2$. Це може бути ще однією гармонікою добового ритму або модою, що описує

асиметричні циклічні флуктуації користувацької активності. Класифікація – «Добова сезонність (сезонна)».

Шоста компонента $Comp\ 6$ (добова сезонність (сезонна)) теж є сезонною: має регулярні коливання меншої амплітуди, але її спектр та ACF відповідають періоду 48 точок. Ймовірно, вона описує дрібніші добові коливання або другорядну гармоніку основного добового циклу. Тип – «Добова сезонність (сезонна)».

Остання з компонент $Comp\ 7$ (добова сезонність (сезонна)) (із вибраних $r=5$ або $r=7$ у поточному запуску) також має ознаки сезонності: регулярність, згладжені хвилеподібні коливання, головний пік спектра поблизу $1/48$. Тип – «Добова сезонність (сезонна)».

На рис.3.2 видно, що:

- Перша компонента - чистий тренд, тобто повільна зміна середнього рівня навантаження;
- Усі інші компоненти мають сезонну природу, і більшість із них описують:
 - добові цикли (≈ 24 год, період 48 точок);
 - або гармоніки добових циклів;
 - окремо виділено півдобову сезонність (≈ 12 год).

Таким чином, SSA-декомпозиція показує, що трафік у мережі:

- має сильну трендову складову,
- має домінантні добові ритми, характерні для поведінки користувачів,
- містить також півдобові цикли, пов'язані з повторюваними піками активності.

Усі ці закономірності підтверджують регулярну, ритмічну та структуровану природу навантаження комп'ютерної мережі.

На рис. 3.4 представлено результат прогнозування навантаження мережі на 24 години вперед, отриманий за допомогою комбінованого методу SSA+AR+season. На графіку одночасно показані фактичні дані, реконструкція сигналу та прогнозні значення з інтервалами довіри.

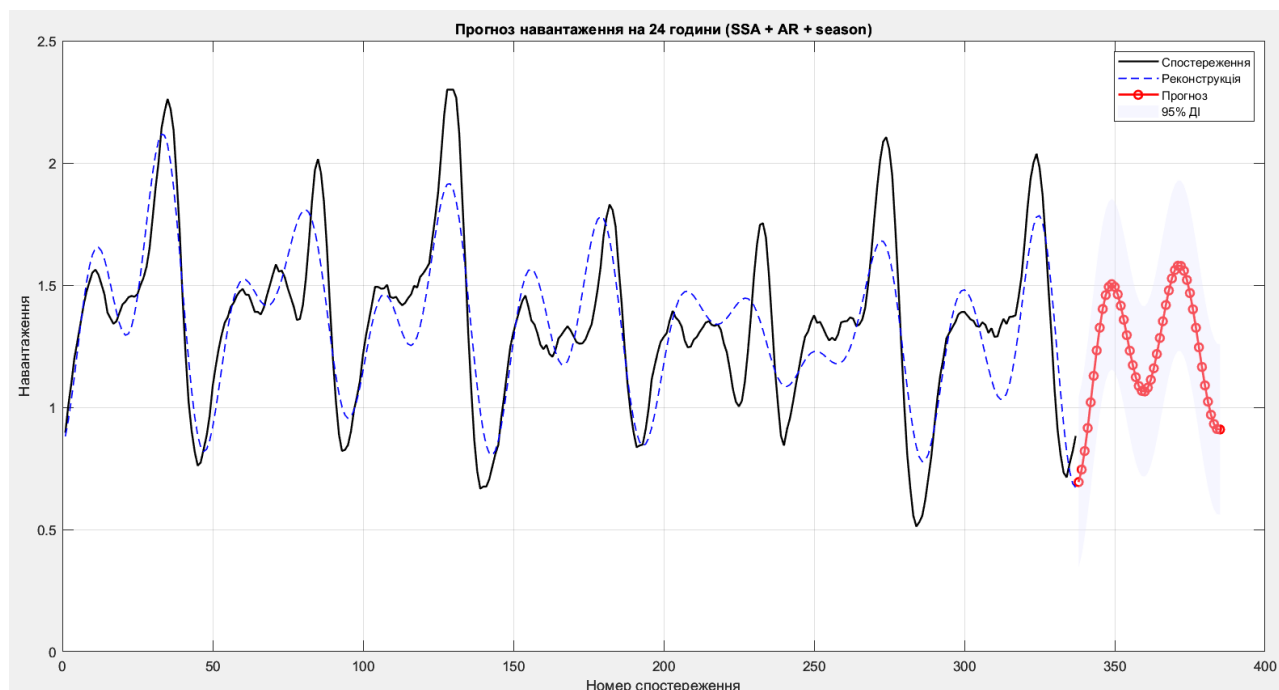


Рис.3.4. Прогноз на 24 години (SSA+AR+season)

Чорна суцільна крива відображає реальні значення навантаження за наявний період спостережень. Це еталон, з яким порівнюється точність реконструкції й прогнозу.

Синя пунктирна лінія (SSA-реконструкція) показує реконструйований ряд після застосування методу SSA, який відокремлює основні складові сигналу – тренд та сезонні повторення – і згладжує випадкові коливання. Близькість реконструкції до фактичних даних підтверджує коректність попереднього розкладу та можливість подальшого прогнозування.

Червона лінія з маркерами ілюструє прогноз на наступні 48 кроків (24 години). Прогноз плавно продовжує характерну добову структуру навантаження, що свідчить про те, що модель успішно виявила і відтворила регулярні цикли у поведінці трафіку. Спостерігається повторення добової сезонності, що відповідає реальній природі трафіку

Світло-синя заштрихована область навколо прогнозу відображає 95% довірчий інтервал, який показує можливий діапазон коливань реальних значень у майбутньому. Ширина інтервалу залишається помірною, що говорить про достатню стабільність моделі.

Інтерпретація та висновки:

- Прогноз узгоджується з попередньою добовою динамікою, відтворюючи характерні піки та спади трафіку.
- Довірчий інтервал помірний, що свідчить про стабільність моделі та хорошу відповідність реконструкції реальному сигналу.
- Гібрид SSA+AR+season працює коректно - прогноз природно продовжує циклічність, виділену SSA.
- Той факт, що прогноз плавно «виростає» із реконструйованого сигналу, підтверджує адекватність AR-моделей компонент, а також ефективність доданої сезонної регресії.

Таким чином, рис.3.4 демонструє, що обраний підхід забезпечує точний і стабільний 24-годинний прогноз мережевого навантаження, враховуючи як трендову, так і сезонну складові початкового ряду, що є критично важливим для аналізу пропускну здатності систем та прийняття рішень щодо управління ресурсами.

На рис. 3.5 подано графік залишків, які визначаються як різниця між початковим рядом навантаження та його реконструкцією, отриманою методом SSA. Залишки відображають ту частину сигналу, яку модель не змогла пояснити трендовими та сезонними компонентами.

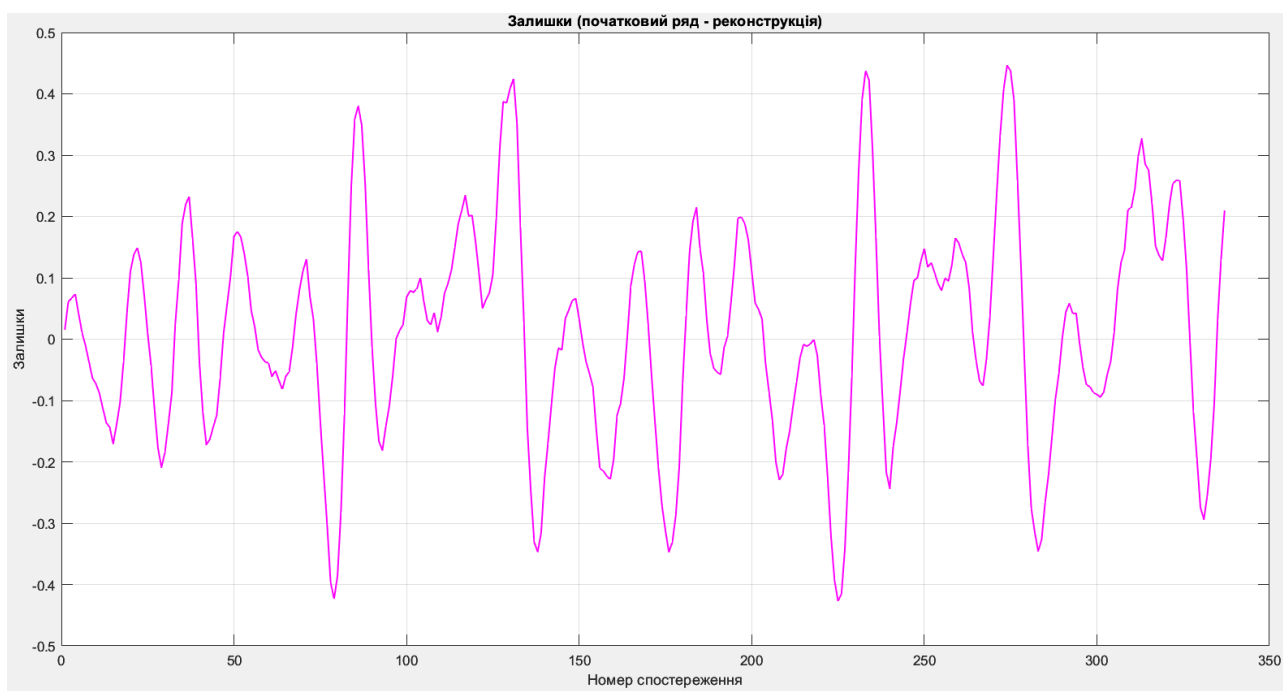


Рис.3.5. Залишки (початковий ряд-реконструкція)

Графік на рис.3.5 демонструє, що залишки коливаються навколо нульового рівня та не мають вираженої трендової або періодичної структури. Це свідчить про те, що основні закономірності вихідного ряду – зокрема тренд і добові цикли – були успішно виділені під час SSA-декомпозиції. Значення залишків мають порівняно невелику амплітуду, що підтверджує якість реконструкції.

Відсутність довготривалих відхилень і наявність переважно випадкових коливань означає, що модель адекватно описує поведінку початкового сигналу. Саме такий вигляд залишків є бажаним, оскільки він вказує на те, що неврахована частина сигналу має характер шуму, а отже, не містить структур, які могли б покращити прогноз шляхом додаткового моделювання.

На рис. 3.6 зображено ймовірність того, що прогнозне значення навантаження для кожного з 48 кроків (тобто для 24 годин уперед) потрапить у межі $\pm 10\%$ від свого очікуваного значення. Цей показник характеризує ступінь надійності прогнозу на різних горизонтах передбачення.

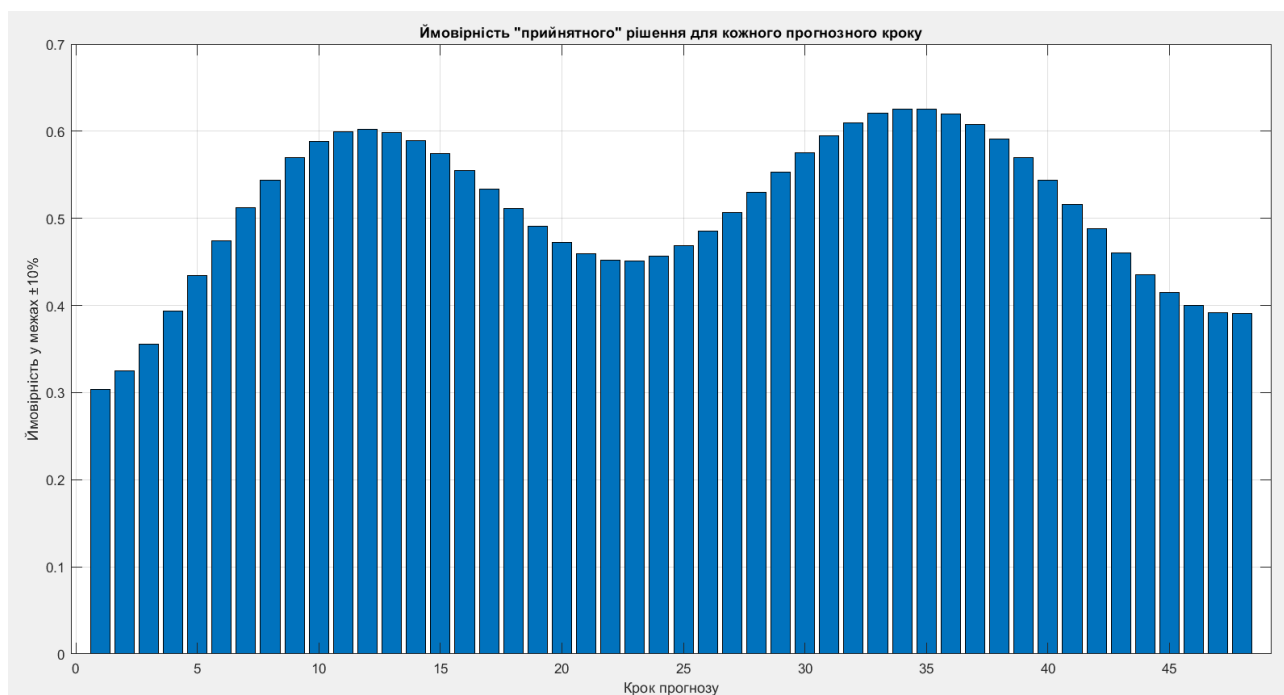


Рис.3.6. Ймовірність «прийнятного» рішення для кожного прогнозного кроку

Значення ймовірностей обчислено на основі нормального розподілу похибки моделі, де ширина інтервалу визначається стандартним відхиленням залишків.

Стовпчаста діаграма дає змогу наочно порівняти, як змінюється ймовірність “прийнятного” (достатньо точного) прогнозу залежно від номера прогнозного кроку.

З графіка видно, що значення ймовірностей лишаються відносно високими на більшості кроків прогнозу, що вказує на стабільність моделі та її здатність забезпечувати прийнятну точність навіть на віддалених кроках. Незначні коливання рівня ймовірності можуть бути пов'язані з природними змінами в структурі сигналу та накопиченням невизначеності під час розгортання прогнозу.

У цілому, рисунок демонструє, що модель формує прогноз, який із великою ймовірністю залишається у «допустимих» межах точності, що є важливим у практичних задачах планування та управління мережними ресурсами.

3.3 Висновки до розділу 3

У розділі здійснено програмну реалізацію методу компонентного прогнозування навантаження та проведено експериментальну перевірку його ефективності. Розроблене програмне забезпечення у середовищі MATLAB забезпечує повний цикл обробки часових рядів: формування траєкторної матриці, виконання сингулярного спектрального аналізу, відновлення компонент, побудову авторегресійних і сезонно-регресійних моделей, обчислення прогнозів і формування довірчих інтервалів. Забезпечено автоматизовану класифікацію компонент та візуалізацію результатів, що підвищує інтерпретованість і практичну цінність отриманих прогнозів.

Експериментальні результати підтвердили здатність методу відтворювати структуру мережевого навантаження та точно прогнозувати як трендові, так і сезонні коливання. Порівняння відновленого та реального сигналів продемонструвало низькі значення похибки та високу відповідність періодичним закономірностям трафіку. Побудовані довірчі межі та оцінка ймовірності прийнятного прогнозу засвідчили стабільність та надійність моделі в умовах реальних флуктуацій навантаження.

Отримані результати підтверджують практичну придатність запропонованого методу та його ефективність для систем моніторингу й керування ресурсами

багатокористувацьких комп'ютерних мереж. Розроблене програмне забезпечення може бути інтегроване у системи підтримки прийняття рішень та інструменти прогнозування навантаження у сучасних телекомунікаційних інфраструктурах.

РОЗДІЛ 4

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Охорона праці

Професійна діяльність в сфері телекомунікацій, який займається дослідженням та прогнозуванням навантаження у багатокористувацьких комп'ютерних мережах, передбачає безпосередню роботу з телекомунікаційним обладнанням, вимірювальними приладами та програмним забезпеченням, що обробляє великий обсяг мережових даних. Тривала робота за комп'ютером та взаємодія з активним телекомунікаційним обладнанням створюють потенційні професійні ризики, зокрема електробезпеки, впливу електромагнітного випромінювання, перевантаження органів зору, м'язово-скелетні перевантаження, психофізіологічне напруження та ризик виникнення пожежі.

З метою мінімізації ризиків діяльність фахівця має здійснюватися у відповідності до законодавчих та нормативних документів України:

- Закон України «Про охорону праці»;
- Закон України «Про телекомунікації»;
- ДСТУ 7234:2011 – Електробезпека;
- НПАОП 40.1-1.21-98 – Правила безпечної експлуатації електроустановок споживачів;
- ДСанПіН 3.3.2.007-98 – Робота з відеотерміналами і персональними комп'ютерами;
- ДСН 3.3.6.096-2002 – Електромагнітні поля;
- ДБН В.2.5-23:2010 – Проектування електрообладнання;
- ДБН В.1.1-7:2016 – Пожежна безпека об'єктів;
- НАПБ А.01.001-2014 – Правила пожежної безпеки в Україні.

У підрозділі розглянуто організаційні, технічні та санітарно-гігієнічні заходи для створення безпечного робочого середовища під час розроблення та

експериментальної перевірки програмного забезпечення для прогнозування мережевого навантаження.

Організація безпечної праці передбачає:

- Проведення інструктажів та навчання персоналу щодо правил роботи з телекомунікаційним обладнанням, електромережами та програмними засобами;
- Встановлення чітких процедур доступу до серверних кімнат, шаф комутаторів та мережевих маршрутизаторів;
- Розробку графіків роботи, які враховують необхідність перерв для зменшення статичного навантаження на опорно-руховий апарат та зоровий апарат;
- Ведення журналів експлуатації обладнання та моніторингу стану електропостачання, температури та вологості у серверних;
- Контроль за дотриманням режиму роботи з відеотерміналами згідно з ДСанПіН 3.3.2.007-98.

Застосування таких організаційних заходів дозволяє систематично знижувати ймовірність травматизму, перенапруження та аварій у мережевому обладнанні.

Робоче місце інженера повинно відповідати ергономічним вимогам:

- робочий стіл висотою 680–800 мм з достатньою площею для монітора, клавіатури, документації та пристроїв збору даних;
- крісло з регулюванням висоти сидіння, кута нахилу спинки та підлокітників;
- розташування монітора на відстані 50–70 см від очей під кутом 20–30° від горизонтальної осі зору;
- клавіатура та маніпулятор на висоті, що дозволяє уникнути перенапруження кистей та плечей.

Згідно з ДСанПіН 3.3.2.007-98, тривала робота за ПК передбачає регулярні перерви (10–15 хвилин на кожну годину роботи) та виконання гімнастики для очей і спини. Дотримання цих вимог мінімізує ризики виникнення зорового перевтомлення, тунельного синдрому та остеохондрозу.

Телекомунікаційне обладнання та серверні шафи працюють під напругою до 380 В, тому працівник зобов'язаний дотримуватися вимог електробезпеки:

- обов'язкове заземлення корпусів серверів та комутаторів;
- використання сертифікованих стабілізаторів та мережевих фільтрів;
- заборона роботи з пошкодженими кабелями;
- вимкнення обладнання перед проведенням технічного обслуговування.

Пожежна безпека забезпечується відповідно до ДБН В.1.1-7:2016 та НАПБ А.01.001-2014:

- наявність первинних засобів пожежогасіння;
- дотримання відстаней між кабельними лотками та електрообладнанням;
- обмеження використання подовжувачів і ненадійних джерел живлення;
- регулярне тестування датчиків пожежної сигналізації.

Інженер, що працює з мережевим обладнанням, піддається впливу електромагнітних полів, що виникають при роботі передавачів та активних модулів комутації. Згідно з ДСН 3.3.6.096-2002, робоче місце повинно бути організоване так, щоб напруженість ЕМ-поля не перевищувала допустимі значення (10–25 В/м залежно від частоти).

Теплове навантаження у серверних кімнатах контролюється кондиціонуванням та вентиляцією згідно з ДСанПіН 3.3.6.042-99, що забезпечує комфортну температуру (20–24 °С) та вологість (40–60 %) для ефективної роботи персоналу.

Робота з великими обсягами даних і програмним забезпеченням для прогнозування трафіку є високої когнітивної складності, що викликає психофізіологічне навантаження. Для збереження працездатності та профілактики стресу рекомендується:

- дотримання режиму роботи та регулярних перерв;
- організація комфортного освітлення та мікроклімату;
- створення зон відпочинку;
- застосування методів тайм-менеджменту;
- навчання правильній організації робочого місця та виконання гімнастики.

Забезпечення безпечних умов праці інженера в сфері телекомунікацій, який розробляє та тестує програмне забезпечення для прогнозування навантаження у

багатокористувацьких мережах, потребує комплексного підходу: організаційного, технічного, санітарно-гігієнічного та психофізіологічного. Дотримання нормативів України дозволяє мінімізувати ризики електробезпеки, впливу електромагнітного випромінювання, зорового та м'язово-скелетного перенавантаження, а також забезпечити пожежну безпеку та ефективну роботу персоналу.

4.2 Безпека в надзвичайних ситуаціях

Забезпечення безпеки персоналу під час надзвичайних ситуацій (ННС) на об'єктах енергопостачання, що живлять телекомунікаційні системи, є критично важливим для безперервної роботи сучасних комунікаційних мереж. Надзвичайні ситуації можуть виникати через різні фактори: техногенні аварії, природні явища або їх комбінації. До техногенних ННС належать короткі замикання, вихід з ладу трансформаторів і генераторів, аварії систем автоматичного керування, пошкодження кабельних ліній або резервних джерел живлення. Природні фактори включають бурі, грози, сильні снігові навантаження, ожеледицю, повені, що призводять до обриву ліній електропередачі та пошкодження телекомунікаційного обладнання. У цих умовах персонал піддається значному ризику ураження електричним струмом, травматизму, впливу високих температур, хімічно небезпечних речовин, а також стресових і психологічних факторів.

Згідно із Законом України «Про цивільний захист населення і територій від надзвичайних ситуацій природного та техногенного характеру» (ст. 3, 14) та Законом України «Про охорону праці» (ст. 13, 17), роботодавець зобов'язаний організувати безпечні умови праці, підготувати персонал до дій у разі ННС та проводити регулярні навчання з цивільного захисту. Планування дій у надзвичайних ситуаціях передбачає чіткий розподіл обов'язків між енергетиками, телеком-персоналом, аварійно-рятувальними підрозділами, пожежними та медичними службами для забезпечення швидкого реагування та злагодженості дій.

Персонал, який обслуговує підстанції та телекомунікаційні вузли, проходить обов'язкові інструктажі та практичні тренування з відпрацювання алгоритмів дій під час аварій і ННС. Основні заходи включають:

- Виявлення і ізоляцію пошкоджених ділянок електромережі.
- Перевірку працездатності систем резервного живлення, таких як акумуляторні батареї та дизель-генератори.
- Контроль параметрів напруги та струму, стану ізоляції кабелів та обладнання.
- Організацію евакуації та забезпечення безпеки інших працівників.

Особливу увагу приділяють тренуванням у випадку комбінації аварійних ситуацій. Наприклад, коротке замикання на трансформаторі під час бурі може призвести до одночасного відключення живлення телеком-станцій і серверних вузлів. У цьому випадку персонал повинен швидко ізолювати уражену ділянку, перевірити роботу систем захисту, включити резервне живлення та забезпечити безпечний доступ до обладнання. Інший приклад – вихід з ладу системи автоматичного керування, що потребує ручного управління з використанням засобів індивідуального захисту (ЗІЗ), таких як електроізоляційні рукавиці, ковдри, каски, захисне взуття та спецодяг.

Критично важливим елементом безпеки є організація системи оповіщення та евакуації. На об'єктах встановлюються світлова та звукова сигналізація, маркуються шляхи евакуації та розташовуються первинні засоби пожежогасіння відповідно до ДСТУ ISO 7010:2015 та Наказу МВС України № 530 від 14.09.2010. Регулярні тренування персоналу дозволяють відпрацьовувати дії при пожежах, вибухах, коротких замиканнях, обвалах кабельних ліній та інших аварійних сценаріях, формуючи швидку реакцію та злагодженість дій.

Сучасні технічні засоби значно підвищують безпеку персоналу під час ННС. До них належать:

- Автоматизовані системи моніторингу параметрів електромережі та телекомунікаційного обладнання.

- Системи дистанційного управління підстанціями, що зменшують необхідність прямого контакту працівників із високовольтним обладнанням.
- Датчики температури, напруги та струму, які дозволяють своєчасно виявляти аварійні режими і запобігати поширенню аварії.
- Резервні джерела живлення, що забезпечують безперервну роботу телекомунікаційних систем під час аварій і ННС.

Особлива увага приділяється психологічній підготовці персоналу. ННС супроводжуються високим рівнем стресу, що може призвести до помилок, травм та порушення роботи телекомунікаційних систем. Для зниження цих ризиків проводяться регулярні медичні огляди, психофізіологічний контроль, моделювання стресових ситуацій під час навчань, ротація персоналу для зменшення перевтоми. Працівники навчаються швидко оцінювати ризики, приймати рішення та діяти злагоджено, застосовуючи ЗІЗ.

Документування дій персоналу під час ННС є обов'язковим. Ведеться журнал аварій, реєструються причини відключень, дії персоналу, час відновлення енергопостачання та ефективність застосованих заходів. Аналіз таких даних дозволяє вдосконалювати навчальні програми, розробляти додаткові заходи безпеки та підвищувати професійну підготовку персоналу.

Взаємодія служб під час ННС має бути ретельно організована. Чергування та зони відповідальності визначаються заздалегідь: енергетики контролюють стан мережі та включення резервних джерел живлення, телеком-персонал забезпечує безперервну роботу зв'язку, аварійно-рятувальні підрозділи виконують евакуацію та надають допомогу, пожежні та медичні служби здійснюють локалізацію загроз і лікування постраждалих.

В умовах ННС безпека персоналу забезпечується комплексним підходом, який включає:

- правильну організацію робочого процесу;
- застосування сучасних засобів індивідуального захисту;
- навчання та тренування персоналу;
- автоматизовані системи моніторингу та управління;

- ефективну систему оповіщення та евакуації;
- дотримання нормативних документів України щодо охорони праці, цивільного захисту та пожежної безпеки.

Виконання цих заходів дозволяє мінімізувати ризики для життя та здоров'я працівників і забезпечити безперервну роботу телекомунікаційних систем навіть під час аварій та ННС.

4.3 Висновки до розділу 4

Проаналізовано умови праці під час роботи з комп'ютерним і телекомунікаційним обладнанням, а також визначено основні небезпечні та шкідливі виробничі фактори, характерні для ІТ-середовища. Розглянуто вимоги нормативних документів щодо організації робочого місця, освітлення, мікроклімату, електробезпеки, ергономіки та режиму праці. Визначено, що дотримання встановлених норм дозволяє мінімізувати професійні ризики, запобігти негативному впливу на здоров'я працівника та забезпечити безпечні умови виконання технологічних процесів.

Проведено оцінювання потенційних надзвичайних ситуацій техногенного та природного характеру, які можуть вплинути на функціонування комп'ютерних мереж і безпеку персоналу. Визначено можливі сценарії аварій, порушень електроживлення, пожеж, пошкодження обладнання чи кібератак. Окреслено заходи цивільного захисту, порядок дій персоналу у НС та засоби локалізації й ліквідації наслідків. Підкреслено важливість своєчасного інформування, евакуації та використання систем оповіщення. Зроблено висновок, що впровадження комплексу превентивних та організаційно-технічних заходів значно підвищує стійкість інфраструктури та безпеку працівників.

ЗАГАЛЬНІ ВИСНОВКИ

У результаті виконання роботи досягнуто поставленої мети – розроблено та досліджено метод компонентного прогнозування навантаження у багатокористувацьких комп'ютерних мережах, що забезпечує підвищену точність і стійкість передбачення за рахунок поєднання сингулярного спектрального аналізу та авторегресійно-регресійного моделювання часових компонент.

Вирішенні завдання:

1. У ході аналітичного огляду сучасних методів прогнозування навантаження систематизовано статистичні, машинні та гібридні підходи, що дало змогу обґрунтовано визначити їхні концептуальні обмеження щодо роботи з нестационарними та сезонними часовими рядами.

2. Розроблено компонентний підхід до декомпозиції часових рядів, у рамках якого застосовано сингулярний спектральний аналіз із використанням траєкторної матриці та сингулярного розкладу, що забезпечило точне виокремлення трендових, сезонних і шумових структур мережевого трафіку.

3. Створено метод автоматичної класифікації виділених компонент, у якому використано спектральні, кореляційні та енергетичні характеристики, що дало змогу коректно типізувати компоненти за їхньою динамічною природою та роллю у формуванні навантаження.

4. Побудовано математичні моделі прогнозування трендових, сезонних і стохастичних компонент із використанням авторегресійних та регресійних підходів, що забезпечило адекватне відтворення локальних залежностей, періодичних коливань і шумових процесів у часових рядах навантаження.

5. Розроблено алгоритм компонентного прогнозування навантаження, у межах якого узгоджено процедури декомпозиції, класифікації та моделювання компонент, що забезпечило формування інтегрального прогнозу з контролем точності та можливістю оцінювання невизначеності.

6. Реалізовано програмне забезпечення у середовищі MATLAB та проведено експериментальну перевірку, результати якої засвідчили підвищену

точність, стійкість і практичну ефективність запропонованого методу в задачах прогнозування навантаження у багатокористувацьких комп'ютерних мережах.

ПЕРЕЛІК ПОСИЛАНЬ

1. Box G. E. P., Jenkins G. M., Reinsel G. C., Ljung G. M. *Time Series Analysis: Forecasting and Control*. 5th ed. – Hoboken, NJ: John Wiley & Sons, 2015. – 712 p. – ISBN 978-1-118-67502-1.
2. Winters P. R. Forecasting sales by exponentially weighted moving averages / *Management Science*. – 1960. – Vol. 6, No. 3. – P. 324–342. – DOI: 10.1287/mnsc.6.3.324.
3. Broomhead D. S., King G. P. Extracting qualitative dynamics from experimental data / *Physica D: Nonlinear Phenomena*. – 1986. – Vol. 20, Nos. 2–3. – P. 217–236. – DOI: 10.1016/0167-2789(86)90031-X.
4. Ghil M., Allen M. R., Dettinger M. D., Ide K., Kondrashov D., Mann M. E., Robertson A. W., Saunders A., Tian Y., Varadi F., Yiou P. Advanced spectral methods for climatic time series / *Reviews of Geophysics*. – 2002. – Vol. 40, No. 1. – Art. 3.1–3.41. – DOI: 10.1029/2000RG000092.
5. Huang N. E., Shen Z., Long S. R., Wu M. L., Shih H. H., Zheng Q., Yen N. C., Tung C. C., Liu H. H. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis / *Proceedings of the Royal Society A*. – 1998. – Vol. 454. – P. 903–995. – DOI: 10.1098/rspa.1998.0193.
6. Hochreiter S., Schmidhuber J. Long short-term memory / *Neural Computation*. – 1997. – Vol. 9, No. 8. – P. 1735–1780. – DOI: 10.1162/neco.1997.9.8.1735.
7. Ferreira G. O., Ravazzi C., Dabbene F., Calafiore G. C., Fiore M. Forecasting network traffic: a survey and tutorial with open-source comparative evaluation / *IEEE Access*. – 2023. – Vol. 11. – P. 6018–6044. – DOI: 10.1109/ACCESS.2023.3236261.
8. Wang F., Liang Y., Lin Z., Zhou J., Zhou T. SSA-ELM: a hybrid learning model for short-term traffic flow forecasting / *Mathematics*. – 2024. – Vol. 12, No. 12. – Art. 1895. – DOI: 10.3390/math12121895.
9. Khvostivska L., Khvostivskyi M., Dediv I. Mathematical, algorithmic and software support for signals wavelet detection in electronic communications. *Proceedings of the 2nd International Workshop on Computer Information Technologies in Industry 4.0*

(CITI 2024). CEUR Workshop Proceedings. Ternopil, Ukraine, June 14-16, 2024. Vol. 3742. P.223-234. ISSN 1613-0073.

10. Khvostivska L., Khvostivskyi M., Dediv I., Yatskiv V., Palaniza Y. Method, Algorithm and Computer Tool for Synphase Detection of Radio Signals in Telecommunication Networks with Noises. Proceedings of the 1st International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2023). CEUR Workshop Proceedings. Ternopil, Ukraine, June 14-16, 2023. P.173-180. ISSN 1613-0073.

11. Khvostivskyi M., Osukhivska H., Khvostivska L., Lobur T., Velychko D. Mathematical modelling of daily computer network traffic. ITTAP-2021: Information Technologies: Theoretical and Applied Problems. The 1st International Workshop (November 16-18, 2021). Ternopil, UKRAINE. P.107-111. DOI: 10.1425/jsdtl.

12. Khvostivskyi M., Osukhivska H., Khvostivska L., Lobur T., Velychko D., Lupenko S., Hovorushchenko T. Mathematical modelling of daily computer network traffic. The 1st International Workshop on Information Technologies: Theoretical and Applied Problems, ITTAP 2021. CEUR Workshop Proceedings. Ternopil, Ukraine, November 16-18, 2021. Vol. 3039. P.107-111. ISSN 1613-0073.

13. Liliya Khvostivska, Mykola Khvostivskyi, Vasyl Dunetc, Iryna Dediv. Mathematical and Algorithmic Support of Detection Useful Radiosignals in Telecommunication Networks. Proceedings of the 2nd International Workshop on Information Technologies: Theoretical and Applied Problems (ITTAP 2022). Ternopil, Ukraine, November 22-24, 2022. P.314-318. ISSN 1613-0073.

14. Khvostivskyi, M., Khvostivska, L., Dediv, I., Yatskiv, V., & Kuchvara, O. (2025). Method and computer tool for synphase prediction of computer network traffic load level. Proceedings of the 3rd International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2025), 18–19 September 2025, Ternopil, Ukraine. CEUR Workshop Proceedings (Vol. 4057, pp. 261–269). CEUR-WS. ISSN 1613-0073.

15. Khvostivska L., Herasymchuk Yu., Zemledukh V., Vorobets I. Method for Automated Detection of Load Levels in Computer Networks. Collection of Scientific Papers with the Proceedings of the 2nd International Scientific and Practical Conference «Global Trends in Science: Research, Innovation and Development» (September 29 – October 1,

2025, Varna, Bulgaria). European Open Science Space, 2025. P.145-149. DOI 10.70286/EOSS-29.09.2025. ISBN 979-8-89704-966-0.

16. Voloshchuk A., Osukhivska H., Khvostivskyi M., Sverstiuk A. (2025). Analysis of electricity consumption using the component method of periodically correlated random processes. *Computer Systems and Information Technologies*, (3), 74–82. <https://doi.org/10.31891/csit-2025-3-8>. ISSN 2710-0766.

17. Хвостівський В., Осухівська Г., Хвостівська Л. Програмне забезпечення системи опрацювання мережевого трафіку. Матеріали IX науково-технічної конференції «Інформаційні моделі, системи та технології» Тернопільського національного технічного університету імені Івана Пулюя, (Тернопіль, 8-9 грудня 2021р.). Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2020. С.102.

18. Методичні рекомендації до виконання, оформлення та захисту кваліфікаційних робіт для здобувачів другого (магістерського) рівня вищої освіти за спеціальністю 172 «Електронні комунікації та радіотехніка» галузі знань 17 «Електроніка, автоматизація та електронні комунікації»/уклад.: Дунець В.Л., Хвостівська Л.В., Дедів І.Ю. Тернопіль: ТНТУ, 2024. 56 с.

19. Стручок В.С. Безпека в надзвичайних ситуаціях. Методичний посібник для здобувачів освітнього ступеня «магістр» всіх спеціальностей денної бо та заочної (дистанційної) форм навчання / В.С.Стручок. Тернопіль: ФОП Паляниця В. А., 2022. 156 с.

20. Дунець В.Л., Хвостівська Л.В., Паляниця Ю.Б. Математичне, алгоритмічне та програмне забезпечення оцінювання завадозахищеності каналів зв'язку з балансною модуляцією. Збірник наукових праць Вісник НУВГП, серія технічні науки, випуск 4 (104), 2023. - С. 95-107. ISSN: 2306-5478.

21. Яворський Б.І., Хвостівська Л.В. Математична модель пульсового сигналу для підвищення інформативності систем діагностики стану судин людини. Вісник кременчуцького національного університету імені Михайла Остроградського. – Кременчук: КрНУ, 2015. – Випуск 6 (95). – С.29-34.

ДОДАТКИ

Теза конференція



CERTIFICATE of participation

Vladyslav Zemledukhi

took part in the 2nd International Scientific and Practical Conference
**«GLOBAL TRENDS IN SCIENCE: RESEARCH,
 INNOVATION AND DEVELOPMENT»**

24 Hours of Participation
 (0,8 ECTS credits)



Head of the
 organizing committee
Helen Volokitina



EOSS-25/0929-079

September 29 – October 1, 2025, Varna, Bulgaria



eooss-conf.com



eoss-conf.com



**ISSUE
№55**



COLLECTION OF SCIENTIFIC PAPERS



**2ND INTERNATIONAL
SCIENTIFIC
AND PRACTICAL
CONFERENCE**

**GLOBAL TRENDS
IN SCIENCE:
RESEARCH, INNOVATION
AND DEVELOPMENT**

**SEPTEMBER 29 – OCTOBER 1, 2025
VARNA, BULGARIA**





UDC 01.1

Collection of Scientific Papers with the Proceedings of the 2nd International Scientific and Practical Conference «Global Trends in Science: Research, Innovation and Development» (September 29 – October 1, 2025, Varna, Bulgaria). European Open Science Space, 2025. 166 p.

ISBN 979-8-89704-966-0 (series)

DOI 10.70286/EOSS-29.09.2025



The conference is included in the Academic Research Index ReserchBib International catalog of scientific conferences.



The conference is registered in the database of scientific and technical events of UkrISTEI to be held on the territory of Ukraine (Certificate №552 dated 16.06.2025).



The materials of the conference are publicly available under the terms of the CC BY-NC 4.0 International license.

The materials of the collection are presented in the author's edition and printed in the original language. The authors of the published materials bear full responsibility for the authenticity of the given facts, proper names, geographical names, quotations, economic and statistical data, industry terminology, and other information.

ISBN 979-8-89704-966-0 (series)



© Participants of the conference, 2025
 © Collection of scientific papers, 2025
 © European Open Science Space, 2025



Коленко А.

ВИМОГИ ДО ВИКЛАДАЧА ТЕАТРАЛЬНОГО МИСТЕЦТВА В
 ПОЧАТКОВІЙ МИСТЕЦЬКІЙ ОСВІТІ..... 115

Іокова Д.І., Петухова Л.Є.

ФОРМУВАННЯ КОМПЕТЕНТНОСТІ ВЗАЄМОДІЇ З ІІІ У
 МАЙБУТНІХ УЧИТЕЛІВ ПОЧАТКОВИХ КЛАСІВ В
 ОСВІТНЬОМУ СЕРЕДОВИЩІ ХЕРСОНСЬКОГО ДЕРЖАВНОГО
 УНІВЕРСИТЕТУ..... 120

Vitkovskiy O., Muryniuk T., Kuzyk I.

RESILIENCE AND INNOVATION IN DENTAL EDUCATION
 DURING MARTIAL LAW IN UKRAINE..... 125

Section: Philosophy

Кліваденко Н.І., Возний Д.

ЕКОЛОГІЧНІ НАСЛІДКИ ТА СОЦІАЛЬНІ КОНФЛІКТИ
 МАСШТАБНИХ ІНЖЕНЕРНИХ ПРОЕКТІВ..... 129

Section: Physical and mathematical sciences

Турчин Є.В., Коваленко М.О.

ОДНА ЗАДАЧА КЛАСТЕРИЗАЦІЇ ГЕОДАНИХ..... 133

Іщенко Л.

ВИКОРИСТАННЯ ЦИФРОВИХ ТЕХНОЛОГІЙ І ШТУЧНОГО
 ІНТЕЛЕКТУ ПРИ ВИКЛАДАННІ ДИСЦИПЛІН ФІЗИКО-
 МАТЕМАТИЧНОЇ ГАЛУЗІ..... 135

Section: Psychology

Маєр Ю., Тарадуда А., Тарадуда С.

ПРОКРАСТИНАЦІЯ ЯК ПСИХОЛОГІЧНИЙ ФЕНОМЕН:
 ТЕОРЕТИЧНИЙ АНАЛІЗ ТА ПРАКТИЧНІ ПЕРСПЕКТИВИ..... 140

Section: Technical Sciences

Khvostivska L., Herasymchuk Yu., Zemledukh V., Vorobets I.

METHOD FOR AUTOMATED DETECTION OF LOAD LEVELS IN
 COMPUTER NETWORKS..... 145

Section: Technical Sciences

METHOD FOR AUTOMATED DETECTION OF LOAD LEVELS IN COMPUTER NETWORKS

Khvostivska Liliia

Ph.D., Associate Professor

Herasymchuk Yurii

Student

Zemledukh Vladyslav

Student

Vorobets Ihor

Student

Ternopil Ivan Puluj National Technical University, Ukraine

The rapid development of information technologies [1] causes a significant increase in the load on computer networks. Modern telecommunication systems deal with traffic that is formed under the influence of cyclical patterns and random fluctuations. To ensure stable operation of the network, it is necessary not only to predict its states, but also to automatically detect load levels, which allows for timely identification of critical modes and make proactive decisions on resource management.

Traditional methods are based [2] on threshold criteria that do not take into account the stochastic nature of traffic and do not provide control over the probability of classification errors. In this paper, an approach to automated detection of load levels is proposed, which is based on the periodically correlated stochastic process (PCSP) model, in-phase/component signal analysis, and the Neumann–Pearson criterion.

Network load classification methods can be conditionally divided into several groups:

- Stochastic models (Poisson, Markov, ARIMA) describe randomness, but do not reproduce cyclicity.
- Fractal models (fBm) take into account self-similarity, but do not take into account diurnal fluctuations [5].
- Machine learning methods [3] provide high accuracy, but require large computational resources.
- Hybrid approaches combine different models, but are difficult to implement in practice [6].

The disadvantage of most of them is the use of simple thresholding schemes, which reduces the reliability of classification. The proposed automated detection system overcomes these limitations by integrating in-phase/component analysis and the Neumann–Pearson criterion.



Traffic load is considered as a periodically correlated stochastic process:

$$\xi(t) = S(t) + X(t), \quad t \in \mathbb{R} \quad (1)$$

where $S(t)$ – a deterministic periodic component with a daily period T , reflecting a repeating cyclicity;

$X(t)$ – stochastic component.

Thus, the model combines the regular structure of traffic and its random nature.

To move from the model to the processing tools, the random component $\xi(t)$ is given in the form of a harmonic expansion:

$$\xi(t) = \sum_{k \in \mathbb{Z}} \xi_k(t) e^{-\frac{j2\pi k t}{T}}, \quad t \in \mathbb{R} \quad (2)$$

where $\xi_k(t)$ – stochastic coefficients that vary with time, and exponential multipliers describe the periodic structure of the process.

This approach allows us to represent the random part as a superposition of in-phase harmonic components.

In-phase and component methods use these harmonic components to construct correlation features.

For centered traffic signals, a spectral-correlation representation is calculated for harmonic number k :

$$\hat{B}_k(u) = \frac{1}{N} \sum_{t \in \mathbb{Z}} \xi^0(t) \xi^0(t-u) e^{-\frac{j2\pi k u}{T}}, \quad (3)$$

where $\xi^0(t)$ – centered traffic signal relative to the mathematical expectation,

$$\xi^0(t) = \xi(t) - E[\xi(t)];$$

N – sample length; u – time shift;

k – harmonic number (correlation component number); $\mathbb{Z}$ – indexing area.

The correlation components $\hat{B}_k(u)$, which are calculated for centered traffic signals, reflect the time-frequency characteristics of the network load. However, in real data there are

- noise components (random fluctuations of short duration);
- anomalous peaks (sudden jumps due to local events) [4, 7];
- limited sample size, leading to statistical instability of estimates.

If raw correlation components $\hat{B}_k(u)$ are used, the classification results may be overly sensitive to random fluctuations.

To avoid this, a procedure of averaging over components was introduced:

$$\bar{C}(u) = \frac{1}{N} \sum_{k=1}^K B_k(u), \quad (4)$$

where $\hat{B}_k(u)$ – estimate of the correlation component on the u -th interval;

K – number of averaging components.



That is, averaging correlation components is a kind of «filter» of statistical features: it cuts off short-term random fluctuations and leaves long-term patterns that are truly informative for further classification of load levels.

As a result, a feature vector is formed:

$$C(u) = (\overline{C}(u_1), \overline{C}(u_2), \dots, \overline{C}(u_m)), \quad (5)$$

which preserves the temporal dynamics of the load and is the basis for classification.

Automated detection of load levels in computer networks is formulated as a statistical hypothesis testing problem. In the simplest case, binary classification is considered:

- H_0 : the load is average (normal network operation mode);
- H_1 : the load is abnormal (minimum or critical).

In order to decide between hypotheses, a likelihood ratio is introduced:

$$\Lambda(C(u)) = \frac{f(C(u) | H_1)}{f(C(u) | H_0)}, \quad (6)$$

where $C(u)$ – vector of averaged in-phase/component correlation features;

$f(C(u) | H_1)$, $f(C(u) | H_0)$ – probability density of the appearance of these features when the hypotheses H_0 and H_1 are true.

The decision (classification) rule has the form:

$$\Lambda(C(u)) \underset{H_0}{\overset{H_1}{<}} \eta, \quad (7)$$

where η – threshold value determined from the condition of controlling the probability of incorrect decisions (determined by the Neumann-Pearson criterion) [8].

This approach allows you to minimize the probability of missing a dangerous condition while controlling the probability of false alarms.

According to the Neumann-Pearson criterion, the optimal rule is considered to be one that minimizes the probability of missing an abnormal state (type II error):

$$\beta = P(\text{решения } H_0 | H_1) = \int_{\Lambda(C(u)) < \eta} p(x | H_1) dx, \quad (8)$$

provided that the probability of a false alarm (type I error):

$$\alpha = P(\text{решения } H_1 | H_0) = \int_{\Lambda(C(u)) > \eta} p(x | H_0) dx, \quad (9)$$

does not exceed a pre-set level (threshold) α_0 :

$$\alpha \leq \alpha_0, \quad (10)$$

where α_0 – false alarm threshold.

Then the detection reliability (the probability of correctly detecting an abnormal condition) is defined as:

$$p_d = 1 - \beta. \quad (11)$$

Thus, the detection method controls the risk of false alarms (α) while minimizing the probability of missing a dangerous condition (β).



Since in practical conditions it is necessary to distinguish at least three network states, the problem is generalized to the multi-class case. The following classes are distinguished:

A_1 : minimum load ($C(u) < \theta_1$) – suitable for preventive work

A_2 : average load ($\theta_1 \leq C(u) \leq \theta_2$) – normal mode

A_3 : critical load ($C(u) > \theta_2$) – a condition that carries the risk of overload

Interval classification rule:

$$A_1 : C(u) < \theta_1, A_2 : \theta_1 \leq C(u) \leq \theta_2, A_3 : C(u) > \theta_2. \quad (12)$$

Maximum likelihood classification for a multidimensional feature vector $C(u)$:

$$A(C(u)) = \arg \max_i p(C(u) | A_i). \quad (13)$$

Threshold values are calculated at the intersection points of the densities:

$$p(C(u) | A_1) = p(C(u) | C_2) \Rightarrow \theta_1, \quad (14)$$

$$p(C(u) | A_2) = p(C(u) | C_3) \Rightarrow \theta_2, \quad (15)$$

The combination of the likelihood ratio and the Neumann-Pearson criterion allows:

- ensure statistical optimality of decision-making;
- control the probability of misclassifications;
- generate automated decisions regarding load levels with high confidence.

As a result, the method not only detects critical modes, but also does so based on a rigorous mathematical model, taking into account the balance between first and second type errors.

The proposed method for automated detection of load levels is based on the PCVP model, centered signal values, estimation of correlation components according to the formula $\hat{B}_k(u)$, averaging over harmonic components, and application of the Neumann-Pearson criterion.

The method allows controlling the probability of type I errors, minimizing the omission of dangerous states, and applying both interval and probabilistic rules of multi-class classification.

References

1. Leland, W. E., Taqqu, M. S., Willinger, W., & Wilson, D. V. (1994). On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*, 2(1), 1–15. DOI: 10.1109/90.282603.
2. Papagiannaki, K., Moon, S., Fraleigh, C., Thiran, P., Tobagi, F., & Diot, C. (2003). Analysis of measured single-hop delay from an operational backbone network. *IEEE Journal on Selected Areas in Communications*, 21(6), 908–921. DOI: 10.1109/JSAC.2003.814650.
3. Hsieh, H. Y., & Sivakumar, R. (2002). Performance comparison of cellular and multi-hop wireless networks: A quantitative study. *ACM SIGMETRICS Performance Evaluation Review*, 30(1), 113–122. DOI: 10.1145/511399.511388.
4. Li, J., Mohapatra, P., & Chuah, C.-N. (2005). Online anomaly detection in network traffic using wavelet analysis. *IEEE INFOCOM 2005*, 6, 2815–2825. DOI: 10.1109/INFCOM.2005.1498537.



5. Abry, P., Veitch, D., & Flandrin, P. (1998). Long-range dependence: Revisiting aggregation with wavelets. *Journal of Time Series Analysis*, 19(3), 253-266. DOI: 10.1111/1467-9892.00090.
6. Khvostivska L., Khvostivskyi M., Dediv I., Yatskiv V., Palaniza Y. Method, Algorithm and Computer Tool for Synphase Detection of Radio Signals in Telecommunication Networks with Noises. *Proceedings of the 1st International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2023)*. CEUR Workshop Proceedings. Ternopil, Ukraine, June 14-16, 2023. P.173-180. ISSN 1613-0073.
7. Khvostivska L., Khvostivskyi M., Dediv I. Mathematical, algorithmic and software support for signals wavelet detection in electronic communications. *Proceedings of the 2nd International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2024)*. CEUR Workshop Proceedings. Ternopil, Ukraine, June 14-16, 2024. Vol. 3742. P.223-234. ISSN 1613-0073.
8. Neyman, J., & Pearson, E. S. (1933). On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A*, 231(694-706), 289–337. DOI: 10.1098/rsta.1933.0009.

МОДЕРНІЗАЦІЯ ТА ДОСЛІДЖЕННЯ ПОКАЗНИКІВ ЯКОСТІ ЕЛЕКТРОПРИВОДА МЕХАНІЗМУ РОТОРНОГО ВАГОНОПЕРЕКИДАЧА З ВИКОРИСТАННЯМ СУЧАСНИХ СИСТЕМ ЕЛЕКТРОПРИВОДУ

Нежурін В.І.

доцент, к.т.н.

Майстренко Д.Н.

магістрант

Салюк О.Д.

магістрант

Куваєв В.Ю.

старший викладач

Кафедра електричної інженерії

Український державний університет науки і технологій, Дніпро, Україна

На сучасному етапі складної економічної ситуації в металургійній промисловості української держави склалися сприятливі та необхідні обставини для модернізації діючих виробничих фондів основного технологічного обладнання металургійних підприємств. Вдосконалення напівпровідникової техніки, розвиток, теорії керування створює умови для розробки комплектних

Скрипт програмного забезпечення компонентного прогнозування навантаження (SSA + AR + seasonal-regression)

```
% =====
% zemledukh_Method.m
% Компонентний прогноз навантаження (SSA + AR + seasonal-regression)
% Прогноз: 24 години (48 півгодинних кроків)
% =====

clear; close all; clc;

%% === ПАРАМЕТРИ ===
fname = 'datatraffic.dat';      % файл з даними (одна колонка)
h = 48;                          % горизонт прогнозу: 24 години = 48 точок
points_per_day = 48;            % кількість точок на добу (півгодинні)

% Перевірка існування файлу і підрахунок тимчасового N
if ~isfile(fname)
    error('Файл %s не знайдено у поточній директорії %s', fname, pwd);
end
tmpN = length(load(fname));
L = min(points_per_day*7, floor(tmpN/2)); % SSA-вікно до тижня або N/2
r = 5;                          % кількість SSA-компонент
maxAR = 20;                     % максимальний порядок AR
acf_season_threshold = 0.12;    % поріг ACF(48) для виявлення сезонності

%% === 1) Зчитування даних ===
y = load(fname);
y = y(:);
N = length(y);
K = N - L + 1;
if K <= 0
    error('Недостатньо спостережень для обраного L. Зменште L або додайте дані.');
```

```
end

%% === 2) Траєкторна матриця (Hankel) ===
X = zeros(L, K);
for k = 1:K
    X(:,k) = y(k:k+L-1);
end

%% === 3) SVD / SSA ===
S = (X * X') / K;
[U, D] = eig(S);
eigvals = diag(D);
[eigvals_sorted, idx] = sort(eigvals, 'descend');
```

```

U = U(:, idx);
Sigma = sqrt(max(eigvals_sorted, 0));

%% === 4) Відновлення компонент (діагональне усереднення) ===
Vr = zeros(length(Sigma), K);
for i = 1:length(Sigma)
    if Sigma(i) > 1e-12
        Vr(i,:) = (U(:,i)' * X) / Sigma(i);
    else
        Vr(i,:) = zeros(1, K);
    end
end

Xr = zeros(L, K);
comp_ts = zeros(r, N);
for i = 1:r
    Xi = Sigma(i) * (U(:,i) * Vr(i,:));
    Xr = Xr + Xi;
    ts = zeros(N,1);
    counts = zeros(N,1);
    for a = 1:L
        for b = 1:K
            ts(a+b-1) = ts(a+b-1) + Xi(a,b);
            counts(a+b-1) = counts(a+b-1) + 1;
        end
    end
    comp_ts(i,:) = ts ./ counts;
end

% --- Реконструкція сумарного сигналу ---
recon = zeros(N,1);
counts = zeros(N,1);
for a = 1:L
    for b = 1:K
        recon(a+b-1) = recon(a+b-1) + Xr(a,b);
        counts(a+b-1) = counts(a+b-1) + 1;
    end
end
recon = recon ./ counts;

resid = y - recon;
sigma = std(resid);

%% === 5) Підбір AR-моделей для компонент ===
AR_models = cell(r,1);
for i = 1:r
    ts = comp_ts(i,:);
    n = length(ts);
    ts_mean = mean(ts);
    ts_d = ts - ts_mean;
    maxp = min(maxAR, floor(n/2)-1);

```

```

    if maxp < 1
        AR_models{i} =
struct('p',0,'coef',0,'var',var(ts_d),'mean',ts_mean,'ts',ts);
        continue;
    end
    acov = zeros(maxp+1,1);
    for k = 0:maxp
        acov(k+1) = sum(ts_d(1:n-k).*ts_d(1+k:n)) / n;
    end
    bestAIC = Inf; bestp = 1; bestcoef = 0; bestvar = 0;
    for ptry = 1:maxp
        [coef, ~] = levinson_durbin(acov, ptry);
        preds = zeros(n,1);
        for t = ptry+1:n
            xlag = ts_d(t-1:-1:t-ptry);
            preds(t) = coef(:)' * xlag;
        end
        resid_loc = ts_d - preds;
        sigma2_est = mean(resid_loc(ptry+1:end).^2);
        AIC = n*log(sigma2_est) + 2*ptry;
        if AIC < bestAIC
            bestAIC = AIC;
            bestp = ptry;
            bestcoef = coef;
            bestvar = sigma2_est;
        end
    end
    AR_models{i} =
struct('p',bestp,'coef',bestcoef,'var',bestvar,'mean',ts_mean,'ts',ts);
end

%% === 6) Виявлення сезонних компонент (ACF(48) + FFT) ===
is_seasonal = false(r,1);
for i = 1:r
    ts = comp_ts(i,:);
    ts_d = ts - mean(ts);
    if length(ts_d) > points_per_day
        acf48 = autocorr_simple(ts_d, points_per_day);
    else
        acf48 = 0;
    end
    % FFT-аналіз
    C = abs(fft(ts_d));
    nfft = length(C);
    half = 1:floor(nfft/2);
    [~, idx_peak] = max(C(half));
    freq_peak = (idx_peak-1) / nfft;
    if freq_peak > 0
        period_est = round(1 / freq_peak);
    else
        period_est = Inf;
    end
end

```

```

    end
    if (acf48 >= acf_season_threshold) || (abs(period_est - points_per_day)
<= 2)
        is_seasonal(i) = true;
    end
end

```

%% === 7) Оцінка регресійних моделей для сезонних компонент ===

```

seasonal_models = cell(r,1);
for i = 1:r
    if ~is_seasonal(i)
        seasonal_models{i} = [];
        continue;
    end
    ts = comp_ts(i,:);
    n = length(ts);
    p = AR_models{i}.p;
    ts_mean = AR_models{i}.mean;
    ts_d = ts - ts_mean;
    maxlag = max(points_per_day, p);
    rows = n - maxlag;
    if rows <= 10
        seasonal_models{i} = [];
        continue;
    end
    Xreg = zeros(rows, p + 1);
    Yreg = zeros(rows,1);
    for t = 1:rows
        tt = t + maxlag;
        if p > 0
            arvec = ts_d(tt-1:-1:tt-p);
            Xreg(t,1:p) = arvec';
        end
        Xreg(t,p+1) = ts(tt-points_per_day);
        Yreg(t) = ts_d(tt);
    end
    b = Xreg \ Yreg;
    if p > 0
        a_est = b(1:p)';
    else
        a_est = [];
    end
    b_season = b(end);
    seasonal_models{i} =
struct('a',a_est,'b_season',b_season,'mean',ts_mean,'p',p);
end

```

%% === 8) Прогноз (48 точок = 24 години) ===

```

forecasts_comp = zeros(r,h);
for i = 1:r
    hist = AR_models{i}.ts;

```

```

p = AR_models{i}.p;
if ~isempty(seasonal_models{i})
    mdl = seasonal_models{i};
    a_est = mdl.a;
    b_season = mdl.b_season;
    for kstep = 1:h
        if length(hist) >= points_per_day
            season_val = hist(end-points_per_day+1);
        else
            season_val = mean(hist);
        end
        if p > 0
            hist_mean = mean(hist);
            xlag = (hist(end:-1:end-p+1) - hist_mean);
            ar_term = a_est * xlag;
        else
            ar_term = 0;
            hist_mean = mean(hist);
        end
        pred = ar_term + b_season * season_val + hist_mean;
        forecasts_comp(i,kstep) = pred;
        hist = [hist; pred];
    end
else
    coef = AR_models{i}.coef;
    for kstep = 1:h
        if p > 0
            hist_mean = mean(hist);
            xlag = (hist(end:-1:end-p+1) - hist_mean);
            pred = coef(:)' * xlag + hist_mean;
        else
            pred = mean(hist);
        end
        forecasts_comp(i,kstep) = pred;
        hist = [hist; pred];
    end
end

forecast = sum(forecasts_comp,1)';

%% === 9) Довірчі інтервали та ймовірності  $\pm 10\%$  ===
ci_low = forecast - 1.96 * sigma;
ci_high = forecast + 1.96 * sigma;

prob_within10 = zeros(h,1);
for t = 1:h
    if forecast(t) == 0
        prob_within10(t) = 0;
    else
        z = 0.1 * forecast(t) / sigma;

```

```

        prob_within10(t) = normcdf(z) - normcdf(-z);
    end
end
empirical_coverage = mean(abs(resid) <= 1.96*sigma);

%% === 10) Вивід результатів ===
disp('--- РЕЗУЛЬТАТИ SSA+AR+SEASONAL ---');
fprintf('N = %d, L = %d, r = %d, прогноз h = %d точок (%.1f год)\n', ...
        N, L, r, h, h/2);
fprintf('σ (оцінка залишків) = %.6f\n', sigma);
fprintf('Емпіричне покриття (±1.96σ) = %.2f%%\n\n', 100*empirical_coverage);
num_seasonal = sum(is_seasonal);
fprintf('Виявлено сезонних компонент: %d з %d\n', num_seasonal, r);

%% === 11) Візуалізація ===
figure('Name','Початковий ряд і реконструкція','NumberTitle','off');
plot(1:N, y, 'k', 'LineWidth', 1.4); hold on;
plot(1:N, recon, '--b', 'LineWidth', 1.2);
xlabel('Номер спостереження'); ylabel('Навантаження');
title('Початковий трафік та SSA-реконструкція');
legend({'Початковий ряд', 'Реконструкція'}); grid on;

% --- Класифікація компонент за типом ---
component_names = strings(r,1);
for i = 1:r
    ts = comp_ts(i,:);
    ts_d = ts - mean(ts);
    C = abs(fft(ts_d));
    nfft = length(C);
    half = 1:floor(nfft/2);
    [~, idx_peak] = max(C(half));
    freq_peak = (idx_peak-1)/nfft;
    % Визначаємо тип
    if idx_peak == 1 || freq_peak == 0
        component_names(i) = "Тренд";
    elseif abs(freq_peak - 1/points_per_day) < 0.02
        component_names(i) = "Добова сезонність";
    elseif abs(freq_peak - 2/points_per_day) < 0.02
        component_names(i) = "Півдобова сезонність";
    elseif is_seasonal(i)
        component_names(i) = "Інша сезонність";
    elseif var(ts_d) < 0.01 * var(y)
        component_names(i) = "Шум";
    else
        component_names(i) = "Коливання / шум";
    end
end

figure('Name','SSA-компоненти (класифікація)','NumberTitle','off');
for i = 1:r
    subplot(r,1,i);

```

```

plot(1:N, comp_ts(i,:), 'LineWidth', 1.1);
grid on;
ylabel(sprintf('Comp %d', i));
% Формуємо заголовок коректно (без множення логіки на рядки)
if is_seasonal(i)
    title_str = component_names(i) + " (сезонна)";
else
    title_str = component_names(i);
end
title(char(title_str), 'FontWeight','normal');
end
xlabel('Номер спостереження');
sgtitle('SSA-компоненти (з автоматичною класифікацією типу)');

figure('Name','Прогноз на 24 години (48 точок)','NumberTitle','off');
plot(1:N, y, 'k', 'LineWidth', 1.4); hold on;
plot(1:N, recon, '--b', 'LineWidth', 1.0);
t_for = (N+1):(N+h);
plot(t_for, forecast, '-or', 'LineWidth', 1.6);
fill([t_for, fliplr(t_for)], [ci_low, fliplr(ci_high)], ...
    [0.9 0.9 1], 'FaceAlpha', 0.35, 'EdgeColor', 'none');
xlabel('Номер спостереження'); ylabel('Навантаження');
title('Прогноз навантаження на 24 години (SSA + AR + season)');
legend({'Спостереження', 'Реконструкція', 'Прогноз', '95% ДІ'}); grid on;

figure('Name','Залишки','NumberTitle','off');
plot(1:N, resid, 'm', 'LineWidth', 1.2);
xlabel('Номер спостереження'); ylabel('Залишки');
title('Залишки (початковий ряд - реконструкція)'); grid on;

figure('Name','Ймовірності ±10%','NumberTitle','off');
bar(1:h, prob_within10);
xlabel('Крок прогнозу'); ylabel('Ймовірність у межах ±10%');
title('Ймовірність "прийнятного" рішення для кожного прогнозного кроку');
grid on;

%% === 12) Збереження результатів ===
save('forecast_results.mat','forecast','ci_low','ci_high','prob_within10',
...
    'sigma','recon','comp_ts','is_seasonal','seasonal_models','AR_models');

TimeIndex = (N+1):(N+h);
Tcol = TimeIndex';
out_table = table(Tcol, forecast, ci_low, ci_high, prob_within10, ...
    'VariableNames',
    {'TimeIndex','Forecast','CI_Lower','CI_Upper','Prob_within_10pct'});
writetable(out_table, 'forecast_24h.csv');
fprintf('Результати збережено у forecast_results.mat та forecast_24h.csv\n');

%% === Допоміжні функції ===
function rlag = autocorr_simple(x, lag)

```

```

x = x(:);
n = length(x);
xm = mean(x);
denom = sum( (x - xm).^2 );
if denom == 0 || lag >= n
    rlag = 0; return;
end
num = sum( (x(1:n-lag) - xm) .* (x(1+lag:n) - xm) );
rlag = num / denom;
end

function [a, sigma2] = levinson_durbin(rvec, p)
    if length(rvec) < p+1
        error('levinson_durbin: недостатньо значень автоковаріації');
    end
    R = toeplitz(rvec(1:p));
    rhs = rvec(2:p+1);
    coef = R \ rhs;
    a = coef(:)'; sigma2 = NaN;
end

```