

АЛГОРИТМ ЗНАХОДЖЕННЯ ПАТЕРНІВ ЧАСОВИХ РЯДІВ В ЗАДАЧАХ АНАЛІЗУ ЯКОСТІ ПОТОКОВИХ СЕРВІСІВ

Розглянуте застосування алгоритмів та технік аналізу часових рядів в задачах моніторингу та прогнозування якості передачі даних в потокових сервісах, таких як VoIP. Запропоноване покращення алгоритму знаходження подібних патернів часових послідовностей з точки зору продуктивності обчислень.

The usage of algorithms and techniques of time-series analysis for the tasks of monitoring and prediction of quality of data transmitting in streaming services, such as VoIP, is investigated. The performance improvement of algorithm of similar pattern detection in time-series is proposed.

Ключові слова: VoIP, QoS, MOS, джитер, патерн, часовий ряд, PAA, SAX.

1. Вступ

У сучасних телекомунікаційних мережах важливого значення набуває практична проблема забезпечення якості обслуговування (Quality of Service, QoS) істотних обсягів різноманітного мережевого трафіку в умовах невизначеності динаміки мережевих потоків та мережевого середовища. Гарантування заданого рівня якості обслуговування при передачі даних відповідно до специфіки сучасних інформаційних потоків в умовах інтенсивного зростання кількості користувачів Інтернет і збільшення об'єму трафіка набуває критичного значення для сучасних мультисервісних мережевих інфраструктур [1].

Ефективність інформаційних систем суттєво залежить від якості управління відповідними ресурсами. Основна мета управління мережевими інфраструктурами – гарантоване забезпечення певних показників якості обслуговування.

Підтримка якості обслуговування в сучасних мультисервісних телекомунікаційних інфраструктурах є досить важким завданням і вимагає розв'язання цілого комплексу задач управління мережевими ресурсами. Для одночасного забезпечення різних вимог QoS у системі зв'язку потрібно впроваджувати технології та засоби керування трафіком, які, в свою чергу, повинні враховувати особливості різних класів трафіку і забезпечувати ефективний динамічний перерозподіл мережевих ресурсів [2].

Для оцінювання якості надання послуг використовують показник оцінювання якості

сприйняття послуг QoE (Quality of Experience), який прямопропорційно залежить від показника якості надання сервісу QoS. На відміну від QoS, QoE – це суб'єктивна оцінка послуги на прикладному рівні користувачем, який користується послугою. Найпростішим варіантом оцінки параметра QoE може слугувати оцінка MOS – Mean opinion score. MOS являє собою зважену оцінку роздратованості користувача через спотворення зв'язку, викликане втратою пакетів та іншими завадами, та виставляється з використанням п'ятибальної системи. Для оцінки QoE сервісу VoIP оцінюють такі параметри: час встановлення з'єднання; час реакції на виконання команди; завмирання зображення; розбиття зображення; синхронізацію зображення та голосу; чіткість та розбірливість звуку [3].

Для реалізації послуг з належним рівнем QoE, необхідно встановити залежність між характеристиками якості надання сервісу QoS та якістю самої послуги, що дозволить виявити та формалізувати закони розподілу характеристики QoS та їхній вплив на параметри QoE. Основними параметрами QoS під час передавання потокових видів трафіку в мультисервісних мережах є:

- надійність передавання – ймовірність втрати або блокування пакетів;
- затримка – середній час перебування пакета в системі;
- флуктуації – оцінка дисперсії часу очікування пакета в буфері;

- пропускна здатність – значення вхідного потоку, для якого забезпечується максимальне значення решти параметрів QoS.

2. Постановка задачі

Для того, щоб ідентифікувати неприйнятні умови передачі аудіо-сигналу та вчасно на них реагувати, існують різні підходи, більшість з яких спирається на низькорівневий аналіз параметрів прийому-передачі пакетів з використанням спеціалізованого ПЗ, наприклад, Wireshark.

Так чи інакше, будь-які системи моніторингу чи прогнозування трафіку використовують алгоритми аналізу та обробки часових рядів для виокремлення з них певних характеристик або закономірностей. Проте суттєвим недоліком більшості існуючих підходів є складність необхідних обчислень, що призводить до унеможливлення використання цих алгоритмів в задачах реального часу, де швидкодія відіграє вирішальну роль.

Таким чином, основною задачею даної роботи є покращення швидкодії одного з існуючих алгоритмів аналізу часових рядів з метою потенційного використання його для аналізу VoIP-трафіку, а також експериментальне порівняння швидкодії класично та модифікованої версій алгоритмів.

3. Огляд праць

Існує велика кількість праць, присвячених різним підходам до аналізу трафіку, алгоритмам їх моніторингу та навіть прогнозування. Наприклад, в роботах [4-6] продемонстровано через емпіричні дослідження, що VoIP-трафік має властивості самоподібності та містить довготривалі залежності. Більш того, в роботі [7] доведено, що VoIP-трафік має мультифрактальні залежності, а у праці [8] розглядається побудова QoS моделей VoIP-трафіку на основі мультифрактальності та Марковських моделей.

Низка праць присвячена можливості прогнозувати параметри QoS та деградацію якості сервісів. Наприклад, у роботі [9] розглядається можливість передбачення продуктивності мережевих інфраструктур за допомогою Марковських процесів. В [10] запропоновано використання нейронних мереж та лінійної регресії, як технік прогнозування майбутніх показників навантаження в

мультисервісних мережах на основі аналізу трафіку.

Багато робіт присвячені можливостям представлення VoIP-характеристик у термінах нечіткої логіки [11] або прогнозуванню рівня QoS з використанням штучних нейронних мереж [12] (або поєднанню цих методів).

Окрема група робіт описує підходи до моніторингу QoE високорівневими засобами, такими як аналіз соціальних мереж у реальному часі [13], які дозволяють відстежувати незадоволеність якістю сервісів користувачами у випадках, коли низькорівневі засоби не змогли детектувати погіршення якості. Загалом, оптимальною практикою є комбінування обох сімейств методів для досягнення комплексного рішення проблеми.

4. Параметри VoIP-трафіку та їх вплив на якість послуги

Залежно від того, які компоненти мережі скомпрометовані, користувачі можуть мати проблеми якості аудіо-сервісу. Оскільки голосовий зв'язок є сервісом реального часу, він є чутливим до затримок і зміни часу прибуття пакетів. Кодеки вимагають постійного, надійного потоку пакетів, щоб забезпечити прийнятну якість відтворення. Пакети, які прибувають занадто рано, занадто пізно, або з ривками, спотворюють відтворення аудіосигналу. Це явище називається *джитером*.

Джитер – це небажані фазові та/або частотні випадкові спотворення при передачі сигналу. Виникають внаслідок нестабільності тактового генератора, змін параметрів лінії передачі в часі і різної швидкості поширення частотних складових одного і того ж сигналу. Джитер може впливати на цифровий сигнал у двох широких областях: у процесі перетворення аналогового сигналу в цифровий (АЦП/ЦАП) і навпаки (ЦАП/АЦП), а також при передачі в цифровому вигляді. Джитером дискретизації або джитером семплінгу називають помилки вибору моментів часу квантування в процесі оцифровки в АЦП, при перетворення в аналоговий сигнал в ЦАП або в перетворювачах частоти дискретизації. Велике значення джитера в перерахованих випадках може призвести до відчутного погіршення якості сигналу. На відміну від поступового погіршення звуку при збільшенні джитера семплінгу, велике значення інтерфейсного

джитера при передачі звукових даних може призвести до втрати цілісності даних. Таким чином, стає важливим контролювати значення джитера при передачі даних. Джитер цифрових звукових інтерфейсів повинен знаходитися в певних допусках, щоб його можна було компенсувати на приймальній стороні.

Оскільки жодна мережа не може гарантувати абсолютно постійний потік пакетів в реальних умовах, VoIP телефони використовують джитер-буфери, щоб згладити передачу. Буфер джитера є кеш-пам'яттю, що функціонує за принципом First In, First Out (FIFO), яка збирає пакети по мірі їх надходження та направляє їх до кодеку рівномірно рознесеними в часі і в правильній послідовності для точного їх відтворення.

У той час як джитер-буфер може успішно замаскувати невеликі проблеми затримки, суттєвий джитер може призвести до переповнення джитер-буферу, що в свою чергу призводить до втрати пакетів. Збільшення розміру буфера може допомогти, але тільки до певної міри: буфер джитера, що збільшує загальну затримку проходження пакетів до 300мс робить нормальну розмову неможливою.

Також дуже суттєво на якість голосових сервісів впливає втрата пакетів. Існує пряма та виражена залежність між процентом втрати пакетів та якістю відтворення аудіосигналу (відображений у вигляді оцінки MOS) [14] (рис. 1).

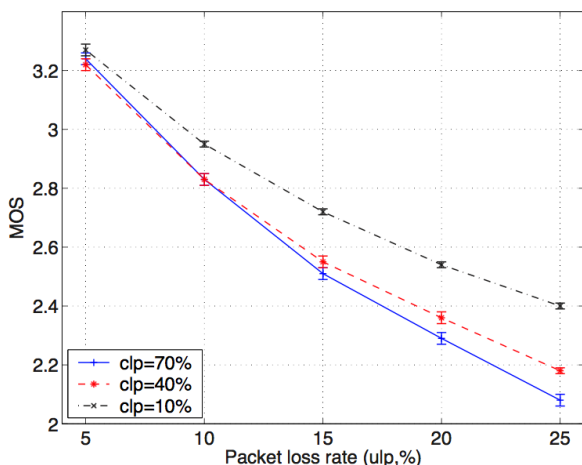


Рис. 1. Залежність MOS від втрати пакетів для кодеків G.729

5. Аналіз часових рядів для ідентифікації патернів параметрів якості мережевого трафіку

Дуже важливу роль в аналізі часових рядів, в тому числі в задачах моніторингу якості потокових сервісів, відіграє pattern matching – знаходження “подібних” фрагментів у послідовності даних (рис. 2).

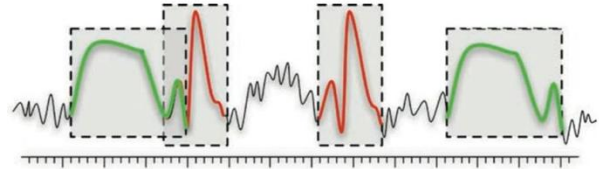


Рис. 2. Знаходження подібних фрагментів у часовому ряді

Суттєвим моментом для даної задачі є те, що подібність не є ідентичністю: на певному часовому відрізку дані можуть відрізнитись за амплітудою та мати певні мінімальні відмінності, наприклад, викликані шумовими завадами (рис. 3). Проте такі фрагменти все ж залишаються представниками одного класу патернів, тому алгоритми, що ідентифікують та виокремлюють їх, повинні бути інваріантними до таких завад.

Найчастіше для нівелювання несуттєвих відмінностей використовують нормалізацію вихідних даних та певний набір фільтрів (наприклад, фільтр Гауса для згладжування шуму, який може бути наявним у вихідних вибірках).

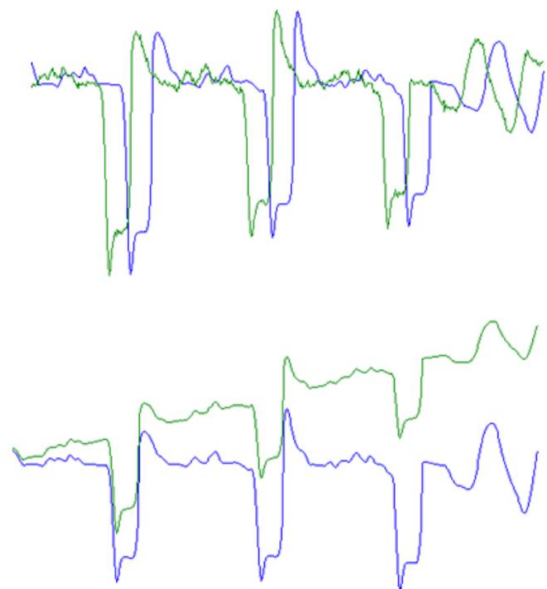


Рис. 3. Зсуви патернів за часом та відмінності за амплітудою

Існує декілька сімейств алгоритмів для кодування часових послідовностей з метою їх подальшого аналізу: дискретне перетворення Фур'є, часткова лінійна апроксимація, вейвлети Хаара, адаптивна часткова константна апроксимація та інші. В цій роботі розглянутий ще один алгоритм – символний. Він має низку переваг, таких як можливість хешування даних, застосування Марковських моделей, суфіксних дерев, дерев прийняття рішень та ін. [15]

Розглянутий алгоритм має назву SAX (Symbolic Aggregate approXimation) [16]. Алгоритм SAX дозволяє описати часовий ряд довільної довжини n рядком довільної довжини w ($w < n$, як правило $w \ll n$). Рядок складається з символів певного алфавіту, розмір якого – довільне ціле число α , де $\alpha > 2$.

Застосування підходу символного кодування дозволяє отримати представлення часових рядів в просторі символів деякого обраного алфавіту.

Алгоритм відрізняється від інших [17] процедурою дискретизації, в якій використовується проміжне представлення часового ряду. Дані спочатку конвертуються з використанням PAA (Piecewise Aggregate Approximation), а потім приводяться до символної послідовності.

Суть часткової агрегуючої апроксимації полягає в тому, що спочатку вихідні дані нормалізуються на певному інтервалі (також можлива попередня фільтрація для пригнічення шумових завад), а потім виконується дискретизація (рис. 4).

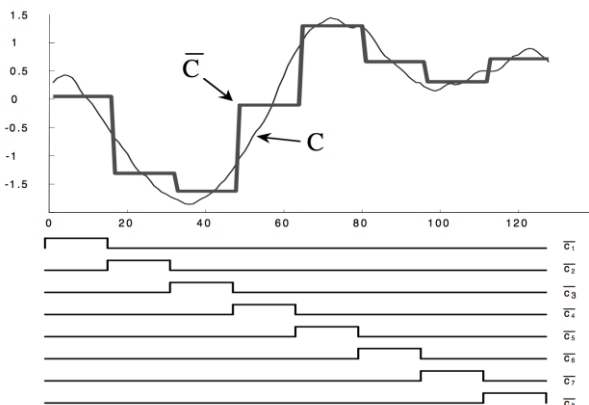


Рис. 4. PAA-розбиття часового ряду

Далі PAA-перетворенню ставиться у відповідність символна послідовність з використанням алгоритму SAX, який на основі квантилів наявного розподілу ставить у відповідність до кожного амплітудного рівня

символ із попередньо визначеного алфавіту A . Редукований вектор дискретизується за допомогою ліній розриву, які розташовані паралельно осі часу і ділять весь простір значень часового ряду на області. Кожен регіон повинен позначатися деяким символом алфавіту A . Потужність алфавіту вважається заданою і дорівнює кількості ліній розриву, які ми можемо розмістити. Кожен елемент PAA-вектора трансформується в символ відповідно до області на осі амплітуд, до якої він потрапив (рис. 5).

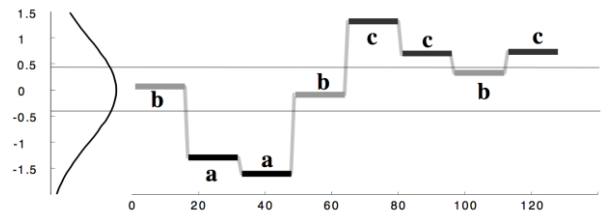


Рис. 5. Часовий ряд перетворюється у символну послідовність "baabccbc" з параметрами $n = 128$, $w = 8$, $\alpha = 3$

Даний метод дискретизації підтримує довільне розташування ліній розриву, проте, автори оригінального методу пропонують розставляти їх відповідно до нормального розподілу.

Далі для знаходження певних сталих патернів часовий ряд сканується відрізком певної довжини з заданим зсувом та кожний відрізок порівнюється з оригіналом (рис. 6).

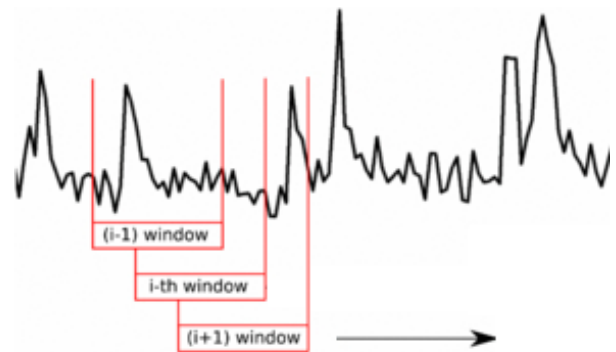


Рис. 6. Сканування часового ряду для знаходження шуканого патерну

Існує декілька метрик відстані (Евклідова, SDA, IMPACTS та ін.), які визначають подібність закодованих даних. Та SAX пропонує власну міру відстані, яка демонструє гарні результати порівняно з іншими (рис. 7).

6. Покращення продуктивності знаходження подібних патернів у часових рядах

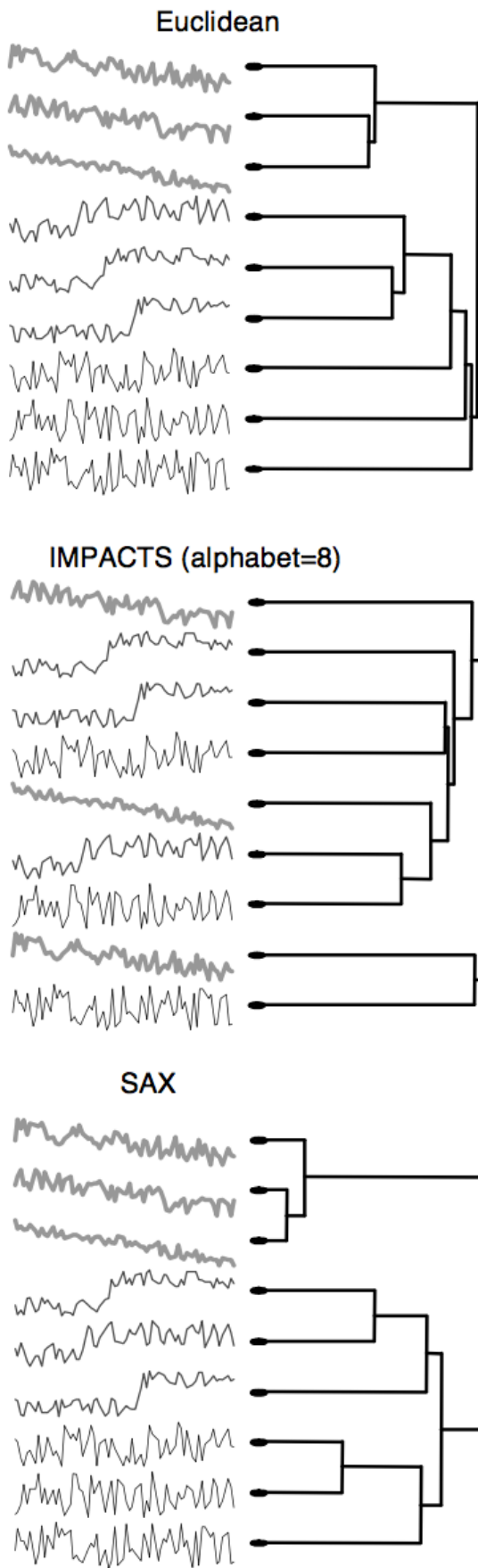


Рис. 7. Порівняння результатів кластеризації часових рядів з використанням різних метрик

Як було зазначено, класичний підхід ідентифікації патернів у часових послідовностях зводиться до сканування послідовності “вікнами” сталого розміру та ітеративного порівняння підпослідовностей з використанням певних метрик.

Цей підхід має суттєвий недолік з огляду на продуктивність обчислень. Якщо стоїть задача для двох закодованих часових послідовностей знайти всі співпадаючі підпослідовності, доводиться виконувати повне сканування вихідних послідовностей декілька разів, оскільки попередньо ми не знаємо якої довжини подібні патерни можемо зустріти. Тому сканування повторюється для вікон різної довжини: наприклад, якщо шукані патерни можуть мати довжину від 50 до 80 символів, то для точного ідентифікування подібностей необхідно виконати сканування та співставлення “вікон” 30 разів, збільшуючи розмір вікна з 50 до 80 з одиничним кроком.

Очевидно, якщо подібна процедура потребується для вирішення тієї чи іншої задачі аналізу поточкових даних, таких як VoIP-трафік, така ітеративність алгоритму є неприйнятною, з огляду на те, що шукані підпослідовності можуть бути довільної довжини та повний перебір усіх можливих вікон для співставлення даних унеможливить своєчасну обробку інформації в умовах реального часу.

Протягом досліджень нами була помічена цікава особливість, яка проявляється, якщо відобразити значення відстаней між усіма вікнами одного розміру у вигляді двомірної карти відстаней. Як було помічено, у такому відображенні дуже яскраво виділяються геометричні примітиви, які вказують на подібність підпослідовностей значно більшої довжини, ніж довжина вікна, яка використовувалась для складання візуальної карти даної ітерації. Наприклад, в даному випадку аналізувались дві часові послідовності довжиною 500 елементів, сканування відбувалось вікном розміром 50 елементів (рис. 8).

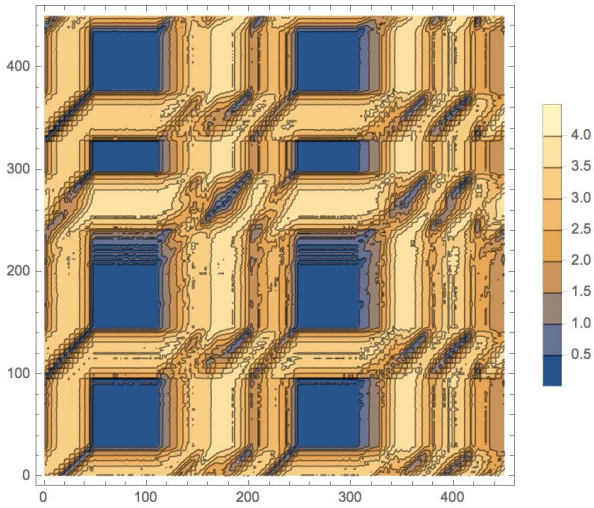


Рис. 8. Двовимірна карта відстаней підпоследовностей з розміром вікна 50

На цьому зображенні чітко відокремлені прямокутники, які можна виділити з загальної картини за допомогою порогової фільтрації та бінаризації. Для процедури фільтрації можна варіювати значення граничного коефіцієнту, що призведе до виокремлення графічних примітивів різного розміру (рис. 9).

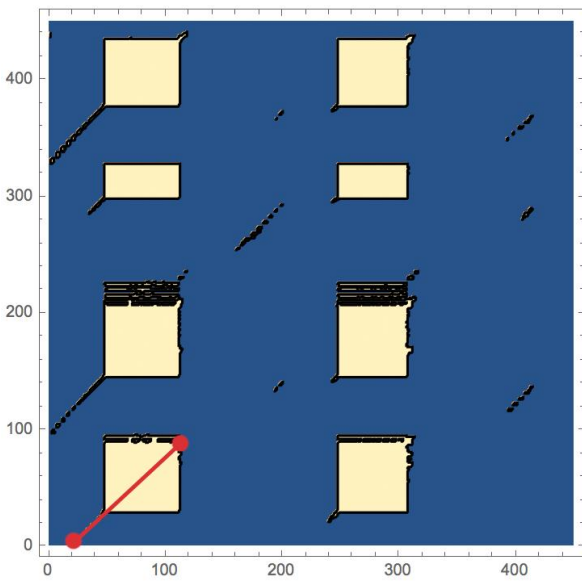


Рис. 9. Відфільтрована та бінаризована карта відстаней

Якщо зіставити, використовуючи осі абсцис та ординат, відрізки вихідних послідовностей, які відповідають виділеним діагональним відріzkам, то можна побачити, що ці фрагменти послідовностей є подібними. Наприклад, на рисунку 10 виділений відрізок між точками (20; 100) та (0; 80). Нижче наведені відповідні інтервали на аналізованих часових рядах (рис. 10).

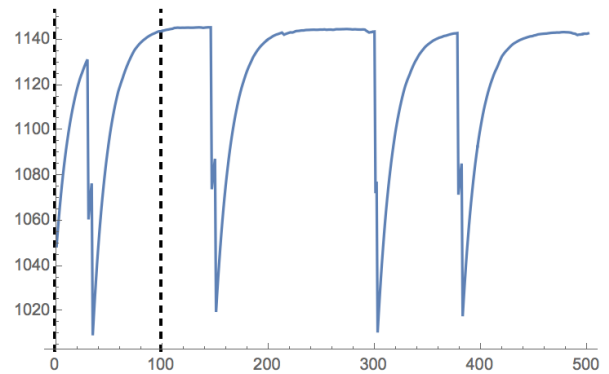
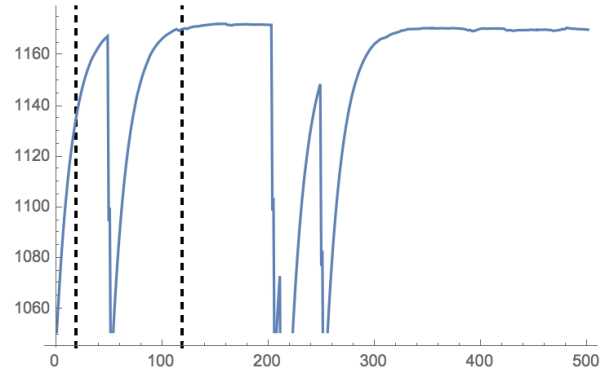


Рис. 10. Знайдені подібні підпоследовності у двох часових послідовностях

Як бачимо, з використанням вікна однієї довжини (50) вдалося знайти подібні патерни довжиною 80 елементів (без необхідності у додаткових 30 переборах, щоб збільшити розмір вікна до 80).

7. Експериментальне порівняння продуктивності класичного та удосконаленого підходів

Нами було розроблено два Java-додатки для експериментальної перевірки швидкодії алгоритмів: один додаток імплементує класичний підхід, інший – удосконалений.

Обидва додатки використовують спільні алгоритми для проведення PAA- та SAX-перетворень, проте різні процедури порівняння отриманих даних. Перший додаток для знаходження схожих часових патернів у двох послідовностях використовує почергове сканування даних вікнами змінної довжини (з 50 до 140), щоб для знаходження схожих патернів у цьому інтервалі довжини. Другий додаток базується на підході, запропонованому у п'ятому розділі.

Тестові дані для аналізу взяті з реальної VoIP-сесії та екстрактовані за допомогою програми Wireshark (рис. 11).

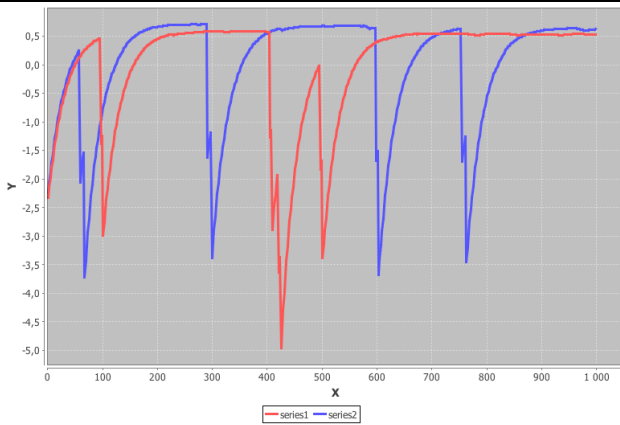


Рис. 11. Тестові дані – джитер

Обидві програми знайшли кілька співпадаючих патернів, найдовшим з яких є наступний (рис. 12).

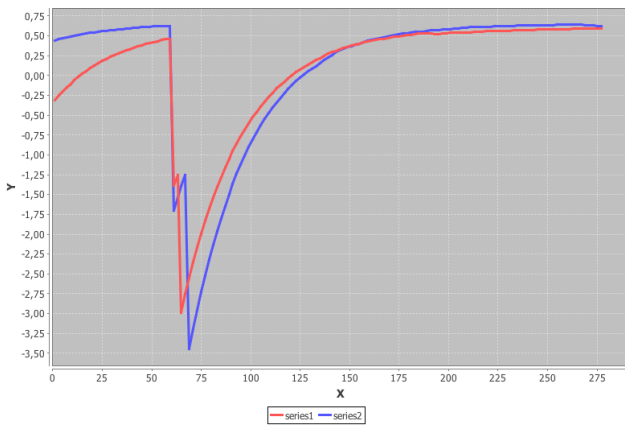


Рис. 12. Приклад знайденого патерну

Проте швидкодія програм суттєво відрізняється. Нижче наведений графік, який демонструє, скільки часу в мілісекундах пішло на сканування кожним новим вікном у класичному підході (90 ітерацій), та скільки часу зайняв модернізований підхід (одна ітерація) (рис. 13).

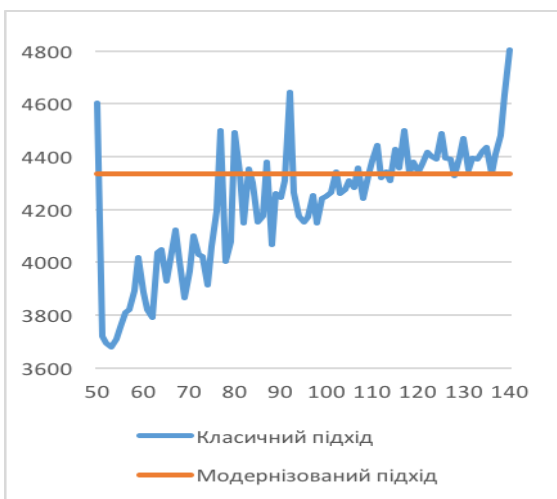


Рис. 13. Порівняльна діаграма швидкодії

Загалом повний цикл роботи програми, заснованої на класичному підході, зайняв **380930 мс**, у той час як виконання модернізованого алгоритму зайняло лише **4334 мс**, що вказує на приріст швидкодії більш ніж у **87 разів**.

Висновки

У роботі проаналізоване застосування алгоритмів та технік обробки часових рядів в задачах моніторингу та прогнозування якості передачі даних в потокових сервісах, таких як VoIP. Показано, які саме параметри впливають на якість аудіосигналу в голосових сесіях, та як можна їх представити у вигляді часового ряду.

Розглянуто символічний алгоритм кодування часових послідовностей SAX, який має низку переваг перед аналогами та оперує власною мірою відстаней між послідовностями. Запропоноване покращення алгоритму знаходження подібних патернів у часових послідовностях з точки зору продуктивності обчислень.

Створено два Java-додатки для знаходження всіх схожих підпослідовностей у часових рядах, які експериментально підтверджують перевагу у швидкодії запропонованого алгоритму порівняно з класичним.

Список посилань

1. Славко О.Г. Інформаційна технологія керування перевантаженнями в мультисервісних телекомунікаційних мережах / Славко О.Г. – 2011 – Вісник Кременчуцького національного університету імені Михайла Остроградського – Вип. 2 (67), част. 1. – С. 29-34.
2. Лаврів О.А. Модель системи управління ресурсами мультисервісних мереж в умовах самоподібності трафіку / Лаврів О. А. – 2012 – Вісник Національного Університету "Львівська політехніка". Радіоелектроніка та телекомунікації. – С. 165-172
3. Чернихівський Є.М., Червенець В. В., Білик О. Б. Визначення часових параметрів обслуговування потокового трафіку з пріоритетними класами / Чернихівський Є.М. – Вісник Національного університету "Львівська політехніка". – 2011. – № 705 : Радіоелектроніка та телекомунікації. – С. 167-170.
4. Rezaul K., Grout V., A Survey of Performance Evaluation and Control for Self-Similar Network Traffic, Proceedings of the Second International Conference on Internet Technologies and Applications (ITA), 4-7 September 2007, pp. 514-524.
5. Samorodnitsky G., Long Range Dependence, Foundations and Trends in Stochastic Systems, 1 (3) (2007) 163-257.
6. Zhang G., Xie G., Yang J., Zhang D., Self-similar characteristic of traffic in current metro area network, Proceedings of the 15th IEEE Workshop on Local and Metropolitan Area Networks, 10-13 June 2007, pp. 176-181.
7. Veitch D., Hohn N., Abry P., Multifractality in TCP/IP traffic: the case against, Computer Networks, 48 (3) (2005) 293-313.
8. Toral-Cruz, Homero, Pathan, Al-Sakib Khan, Pacheco, Julio Cesar Ramirez Accurate modeling of VoIP traffic QoS parameters in current and future networks with multifractal and Markov models – Mathematical and Computer Modelling 57, no. 11-12 (2013): 2832-2845.
9. Pacheco-Sanchez S., Casale G., Scotney B., McClean S., Parr G., Dawson S. Markovian Workload Characterization for QoS Prediction in the Cloud – 2011 IEEE 4th International Conference on Cloud Computing. 147154 (2011).
10. Islam S., Keung J., Lee K., Liu A.: Empirical prediction models for adaptive resource provisioning in the cloud. Future Generation Computer Systems. 28, 155162 (2012).
11. Mansouri, Taha, Nabavi, Ali, Ravasan, Ahad Zare and Ahangarbahan, Hamid. "A practical model for ensemble estimation of QoS and QoE in VoIP services via fuzzy inference systems and fuzzy evidence theory." Telecommunication Systems 61, no. 4 (2016): 861-873.
12. Ролик А.И. Оценка качества предоставления мультимедийных сервисов с использованием нейросетевого классификатора / Ролик А.И., Галушко Д.А., Барна В.В., Томашук А.В., Ясочка М.В. – Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка: збірник наукових праць. – К.: Століття+, 2015. – №63. – С. 25–30.
13. Тітов Д.С. Моніторинг соціальних мереж в системі реального часу / Тітов Д.С., Дорошенко А.Ю. – К.: Наукова дискусія: теорія, практика, інновації, 2015. – С. 93-96.
14. Lingfen S. Speech quality prediction for voice over Internet protocol networks: PhD diss.: 2004 / Lingfen Sun. – University of Plymouth, UK, 2004. – 218pp.
15. Lin, J., Keogh, E., Lonardi, S. & Chiu, B. (2003) A Symbolic Representation of Time Series, with Implications for Streaming Algorithms. In proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery. San Diego, CA. June 13.
16. Lin, J., Keogh, E., Lonardi, S. and Patel P. Finding motifs in time series. In Proc. 2nd Workshop on Temporal Data Mining, 2002. pp. 214-221.
17. Keogh, E., Chakrabarti, K., Pazzani, M., Mehrotra, S. Locally Adaptive Dimensionality Reduction for Indexing Large Time Series Databases. In proceedings of ACM SIGMOD Conference on Management of Data. Santa Barbara, CA, 2001. pp. 151-162.