УДК 004.421
О.А. Юхта[1], О.О. Ройко[2]
[1]Східноєвропейський національний університет імені Лесі Українки, м. Луцьк, Україна
[2]Волинський коледж Національного університету харчових технологій, м. Луцьк, Україна

ВИКОРИСТАННЯ ОНЛАЙН-АЛГОРИТМІВ У ЕКОНОМІЧНИХ ПРОЦЕСАХ НА ПРИКЛАДІ ВИСОКОЧАСТНОГО ТРЕЙДИНГУ HFT

О.А. Юхта, О.О. Ройко. Використання онлайн-алгоритмів у економічних процесах на прикладі високочастотного трейдингу HFT. В статті розглядаються особливості використання онлайн-алгоритмів у економічних процесах на прикладі високочастотного трейдингу HFT. Це дає змогу вирішити такі проблеми, як оцінка поточної середньої ліквідності, поточна оцінка мінливості і робочої лінійної регресії за допомогою однопрохідних онлайн-алгоритмів.
Ключові слова: online-алгоритм, HFT, однопрохідний алгоритм, онлайн-алгоритм дисперсії, онлайн-алгоритм середнього, трейдинг .

А.А. Юхта, О.А. Ройко. Использование онлайн-алгоритмов в экономических процессах на примере высокочастотного трейдинга HFT. В статье рассматриваются особенности использования онлайн-алгоритмов в экономических процессах на примере высокочастотного трейдинга HFT. Это позволяет решить такие проблемы, как оценка текущей средней ликвидности, текущая оценка изменчивости и рабочей линейной регрессии с помощью однопроходных онлайн-алгоритмов.
Ключевые слова: online-алгоритм, HFT, однопроходный алгоритм, онлайн-алгоритм дисперсии, онлайн-алгоритм среднего, трейдинг.

O.A. Yukhta, O.O. Roiko. The use of online-algorithms in economic processes on an example of high-frequency trading HFT. The article discusses the features of using online algorithms in economic processes on the example of high-frequency trading. This allows solving problems such as the estimation of current average liquidity, the current estimation of variability and working linear regression with the help of one-pass online algorithms.
Keywords: online-algorithm, HFT, one-pass algorithm, online-algorithm of dispersion, online algorithm of average, trading.

Постановка проблеми. На практиці існують задачі, котрі потрібно вирішувати, не знаючи заздалегідь усі вхідні дані. Наприклад, системна програма, що контролює роботу операційної системи, повинна генерувати деякі дії (приймати рішення, записувати певні дані на вихід) до того, як надійдуть всі дані. Такі алгоритми називаються алгоритмами, що працюють в режимі реального часу, або online-алгоритмами. Як правило, online-алгоритм отримує послідовність запитів. По кожному із запитів він повинен надати деякий сервіс до того, як отримає наступний запит. Звичайно існує декілька можливостей надати сервіс, кожній з яких відповідає деяка вартість даної можливості.

Важливість online-алгоритмів проявляється як і суто в науці, так і для багатьох непомітно у повсякденному житті. В багатьох прикладних областях, де виникають геометричні задачі, ряд обчислень повинен виконуватися по мірі надходження точок (даних). В загальному випадку алгоритм, що обробляє дані згідно міри їх надходження називається відкритим. Основною рисою відкритих алгоритмів вважається відсутність обмежень на час корекції, що рівносильне тому, що новий елемент даних (точка) вводиться за запитом відразу ж після того, як завершується корекція, пов'язана з попереднім елементом даних. Для прикладу, у нейромережах алгоритми, обробляючі дані по мірі їх надходження, через створення бази знань, що з допомогою їх автоматично обновляється, дозволяють зберігати всі переваги індуктивного (нейроінформатика) та дедуктивного (нечітка логіка) підходів.

Сьогодні все більша кількість сервісів працює з алгоритмами, що обробляють дані згідно міри їх надходження: це і служби, що мають відношення як до суто роботи комп'ютера у вигляді, наприклад, відомої задачі кешування (paging problem), що розглядає ефективність використання пам'яті комп'ютера через звернення до дискової та оперативної пам'яті відповідно, так і служби, що працюють у мережі, як так звані хмарні технології з основою у вигляді хмари - деякого ЦОД (дата-центр, сервер) або їх мережі, де зберігаються дані та програми, що з’єднуються з користувачами через Інтернет.

Аналіз досліджень. HFT (High Frequency Trading – високочастотний трейдинг) – вид автоматичного трейдинга, при якому пошук торгівельних можливостей здійснюється шляхом використання спеціальних комп’ютерних алгоритмів. HFT-трейдери використовують так званих торгових роботів для аналізу вхідних даних та організації торгівельних стратегій. Інвестиційна позиція у процесі торгів утримується протягом дуже нетривалого часу, через що кількість операцій у високочастотному трейдингу досягає декількох тисяч у день.

Не дивлячись на те, що зовнішній вигляд місця процесу та його учасників різко змінились, метою усіх трейдерів, як електронних, так і живих людей, залишається те саме: купити актив з одного місця/трейдера і продати його в інше місце/трейдеру по більш високій ціні. Визначаючим фактором між людиною-трейдером і HFT є те, що останній може реагувати швидше, частіше і має дуже короткі періоди проведення угод.

Типовий алгоритм HFT працює на субмілісекундній часовій шкалі, з якою людина-трейдер не може конкурувати, оскільки миготіння людського ока займає близько 300 мілісекунд. Коли HFT-алгоритми конкурують один з одним, вони стикаються з двома основними проблемами: вони отримують великі обсяги даних кожну мікросекунду та вони повинні бути взмозі діяти дуже швидко щодо отриманих даних, оскільки рентабельність сигналів, за якими вони слідкують, розпадається дуже швидко.

Онлайн-алгоритми, що працюють у режимі реального часу, забезпечують природний клас алгоритмів, що підходять для застосування в HFT. В онлайн-задачах нові вхідні дані отримуються та показуються послідовно. Після кожних нових вхідних даних алгоритм повинен прийняти рішення – наприклад, рішення, погоджуватися чи не погоджуватися з купівлею/продажем. Це різко контрастує з оффлайн-проблемою, згідно якої всі вхідні дані доступні в момент прийняття рішення. Багато з практичних задач оптимізації, що розглядаються у комп'ютерній науці і наукових дослідженнях є саме онлайн-проблеми.

Окрім вирішення онлайн-задачі, HFT-алгоритм також повинен швидко реагувати на оновлення ринку. Щоб гарантувати швидкодію реакції, ефективна обробка пам'яті являється необхідністю для «живого» торгівельного алгоритму. Зберігання великої кількості даних у пам'яті буде гальмувати який завгодно процесор, тому важливо, щоб алгоритм використовував лише мінімальну кількість даних та параметрів, які можуть бути збережені у пам'яті швидкого доступу, такій як кеш L1. Крім того, ці фактори повинні відображати поточне становище ринку і повинні оновлюватися при отриманні нових значень даних. Таким чином, чим менша кількість параметрів, які потрібно зберігати у пам'яті, і чим простіше обчислення кожного необхідного параметра, тим швидше алгоритм може реагувати на оновлення ринку.

На основі швидкісних вимог і онлайн-природі HFT-проблеми клас однопроходних алгоритмів особливо підходить для застосування в HFT. Ці алгоритми отримують одну точку даних і в той же час використовують її для оновлення набору факторів. Після оновлення точка даних відкидається і лише оновлені фактори залишаються у пам'яті.

Три основні проблеми можуть виникнути при використанні HFT: оцінка поточної середньої ліквідності, оцінка робочої волатильності та оцінка робочої лінійної регресії.

Кожна з цих проблем може бути вирішена ефективно, використовуючи однопрохідний онлайн-алгоритм. Перевірку продуктивності однопрохідного алгоритму можна оцінити у межах таблиці заявок та описати як вивіряти ці алгоритми на практиці.

Однією з переваг, що HFT має у порівнянні з іншими учасниками ринку, є швидкість реакції. HFT-фірми можуть бачити усі дії на ринку, тобто інформацію, що міститься у таблиці замовлень, і реагувати напротязі мікросекунд. Хоча деякі алгоритми HFT можуть основувати свої дії на джерела інформації поза ринком (наприклад, шляхом аналізу звітів, вимірювання температури або оцінки настрою ринків), найбільше спираючись у своїх рішеннях виключно на повідомленнях, що поступають на ринок. За деякими оцінками на Нью-Йоркській фондовій біржі відбувається приблизно 215000 оновлень в секунду. Задачею HFT являється обробка цих даних таким чином, що дозволить їм приймати рішення, наприклад, для зменшення ризиків. У статті припускається, що HFT може проводити кожне оновлення даних з кращими цінами купівлі та продажу, в тому числі з кращою пропозицією та розмірами попиту.

Наступні приклади онлайн-алгоритмів, пов'язаних с HFT, описані далі.

Основні результати досліджень.

Онлайн-алгоритм середнього. Ілюструє побудову фактора, що передбачає доступну ліквідність, що визначається як сума розмірів по кращій ціновій пропозиції і кращому ціновому запиту у фіксованій у майбутньому точці. Ця величина може бути корисною при оцінці того, який розмір замовлення найкраще виконати у випадку кращого котирування при даній затримці.

Онлайн-алгоритм дисперсії. Ілюструє побудову фактора, що передбачає реалізовану волатильність протягом фіксованої межі у майбутньому. Ця величина може бути корисною при оцінці короткочасного ризику проведення інвентаризації.

У всіх випадках алгоритм має один параметр – альфа, який контролює швидкість з якою забувається стара інформація. На ***графіку 1*** побудована чиста міра ліквідності (сума ставки та попиту) синім кольором. Червоний та зелений показують онлайн-фактор ліквідності з альфа = 0,9 та альфа = 0,99 відповідно. Варто відзначити, що чим ближче альфа наближається до 1, тим гладкішим стає «сигнал» і ефективно відслідковується тенденція в вихідних даних.

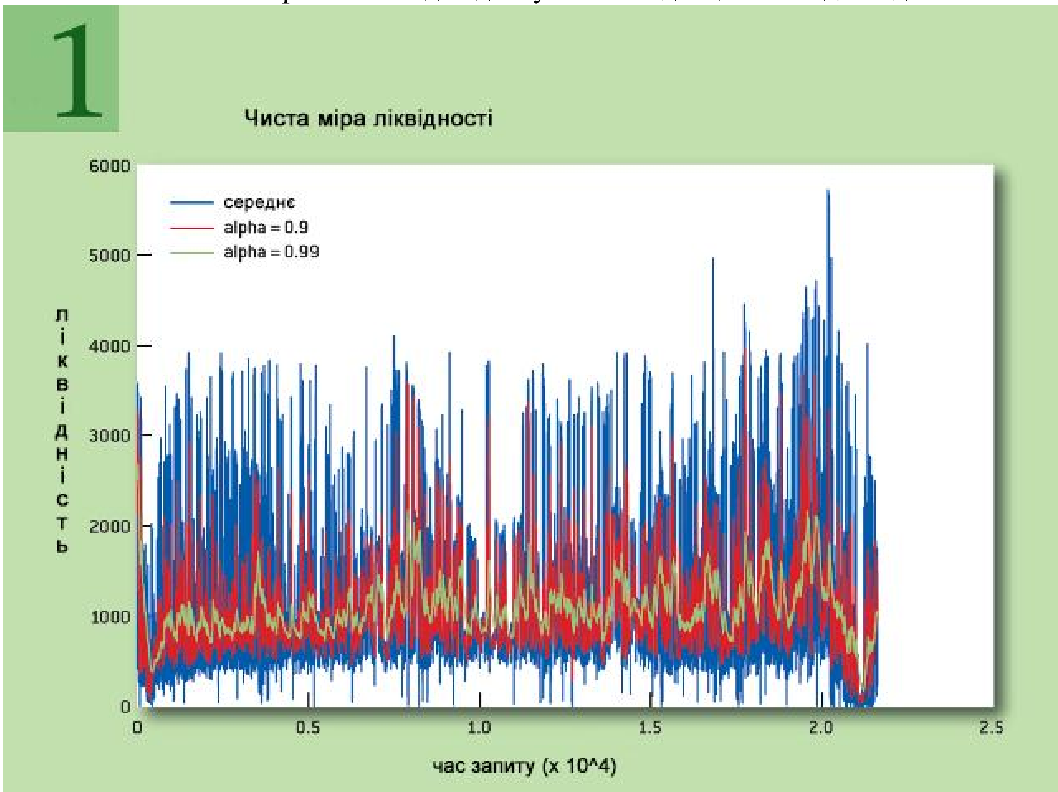

Рис. 1. Чиста міра ліквідності

Графік 2 зображує онлайн волатильність для різних значень альфа. Хоча більше значення альфа і забезпечує плавний «сигнал», він також і залишається далеко позаду основного тренду, оскільки це дає багато лишньої ваги старих даних. Вибір значення альфа балансує між гладкістю сигналу і пониженим відставанням тренду.

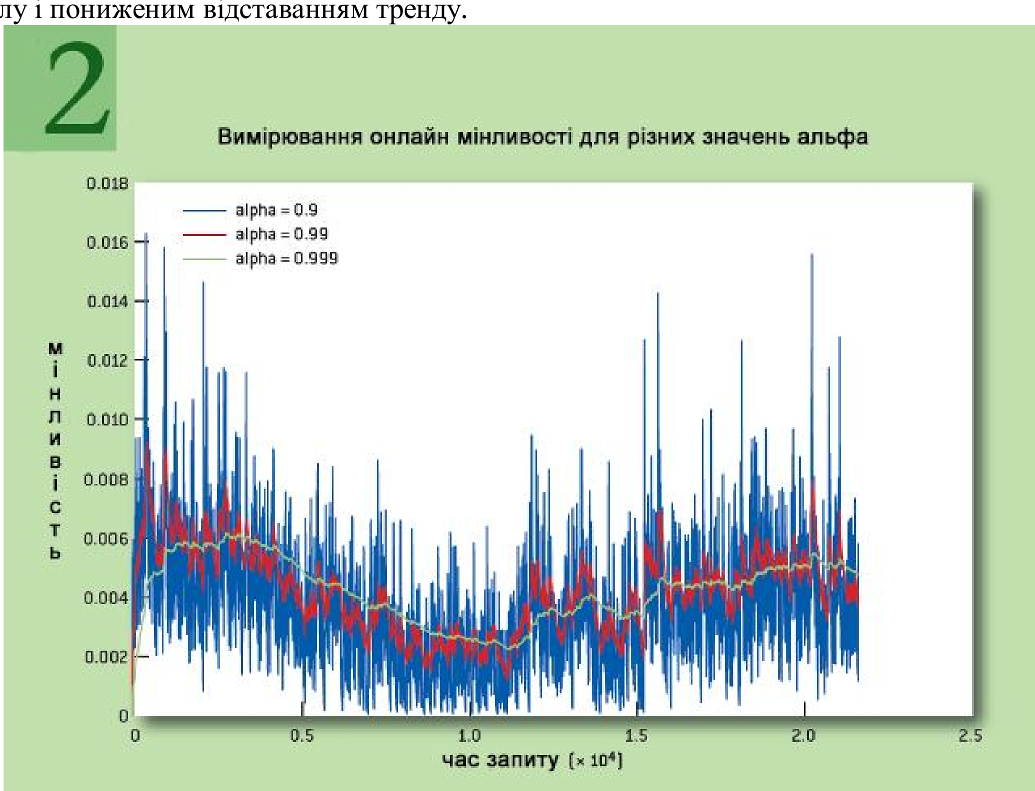

Рис. 2. Вимірювання онлайн мінливості для різних значень

Однопрохідні алгоритми

Як указано в його назві, однопрохідний алгоритм читає кожну змінну один раз, а потім відкидає її. Цей тип алгоритму являється досить ефективним з точки зору обробки пам'яті, оскільки він потребує мінімальну кількість даних для збереження у ній.

Розглянемо важливі приклади однопрохідних онлайн-алгоритмів на експоненціальному рухомому середньому та експоненціальній зваженій дисперсії. По-перше, розглянемо рухоме середнє часового ряду. Це оцінка середнього значення часового ряду по рухомому вікну фіксованого розміру. В фінансах вона часто використовується для для виявлення тенденції у ціні, а саме, порівнюючи два рухомих середніх – одне протягом довгострокового вікна, інше – протягом короткого. У іншому застосуванні середні об'єми продаж за останні п'ять хвилин можуть служити для передбачення об'єму продаж у наступну хвилину. На відміну від експоненціального рухомого середнього, просте рухоме середнє не може бути вирішене з допомогою однопрохідного алгоритму.

Нехай $(X_t)_t = X_0, X_1, X_2, \ldots$ буде послідовністю вхідних змінних. У будь-який момент часу t ми хочемо визначити наступне X_{t+1}. Для $M > 0$ і $t \geq M$ просте рухоме середнє з вікном розміру M визначається як останнє з M-спостережень часового ряду $(X_t)_t$, тобто $\widehat{X}_{t,M} = \frac{1}{M}\sum_{j=0}^{M-1} X_{t-j}$. Рухоме середнє може бути обраховане з допомогою наступної рекурсії $\widehat{X}_{t,M} = \widehat{X}_{t-1,M} - \frac{X_{t-M}}{M} + \frac{X_t}{M}$.

Хоча це і онлайн-алгоритм, але він не є однопроходним, оскільки він повинен відкривати кожну точку вхідних даних рівно двічі: один раз, щоб додати її до рухомого середнього, а потім знову «викинути» її з рухомої середньої оцінки. Такий алгоритм називається двохпроходним і необхідно утримувати весь масив розміру M у пам'яті.

Приклад 1: Однопрохідне експоненціальне середньозважене

На відміну від звичайного середнього $\overline{X} = \frac{1}{t+1}\sum_{j=0}^{t} X_j$, експоненціальне середньозважене призначає експоненціальне ваги до попередніх спостережень:

$$\widehat{X}_{t,\alpha} = (1-\alpha)\sum_{j=0}^{t-1} \alpha^j X_{t-j} + \alpha^t X_0$$

Тут α являється параметром ваги, обирається користувачем та повинен задовольняти умові $0 < \alpha \leq 1$. В цьому випадку експоненціальне середньозважене дає більше значення для більш пізніших вхідних даних в порівнянні зі старими точками. Це часто вважається хорошим наближенням для звичайного рухомого середнього.

Для порівняння з простим рухомим середнім експоненціальне середньозважене приймає до уваги усі попередні значення, а не лише останні M спостережень. Для порівняння на ***графіку 3*** показано, скільки точок даних отримано при 80, 90, 95, 99 і 99,9% від ваги в оцінці в залежності від α. Наприклад, якщо α = 0,95, то останні M = 90 точок даних, що спостерігаються, призводять до 99% від розрахункової вартості.

Причиною надання переваги експоненціальному рухомому середньому над простим рухомим середнім в HFT є те, що ефективне рішення можливе за допогомою однопроходного алгоритма вперше представленого Брауном у 1956 р.

$$\begin{aligned} \widehat{X}_{0,\alpha} &= X_0, \\ \widehat{X}_{t,\alpha} &= (1-\alpha)X_t + \alpha\widehat{X}_{t-1,\alpha}. \end{aligned}$$

Ця формула також представляє просту інтепретацію параметра α як контрольного – скільки ваги надається останньому спостереженню у порівнянні з усіма попередніми спостереженнями.

Приклад 2: Однопрохідна зважена дисперсія

У фінансах мінливість часового ряду часто являється важливим фактором. Мінливість повинна захопити скільки часу серії коливаються навколо її середнього. Немає загальноприйнятого визначення мінливості для даних HFT. Дисперсія випадкової величини визначається як $Var(X) = E\left[X - E[X]^2\right]$. Оцінка експоненціально зваженої дисперсії часового ряду потребує двох оцінок: одна оцінює середнє $E[X]$, а інша дисперсію:

$$\widehat{X}_{0,\alpha} = X_0$$
$$\widehat{V}_0 = 1$$
$$\widehat{X}_{t,\alpha} = (1-\alpha)X_{t,\alpha} + \alpha\widehat{X}_{t-1,\alpha}$$
$$\widehat{V}_{t,\alpha} = (1-\alpha)\left(X_{t,\alpha} - \widehat{X}_{t,\alpha}\right)^2 + \alpha\widehat{V}_{t-1,\alpha}$$

Стандартне відхилення наступної точки X_{t+1} виміру далі оцінюється як $\sqrt{\widehat{V}_{t,\alpha}}$. Знову ж таки, вхідний параметр α ∈ (0,1) обирається користувачем і відображає, скільки ваги надається минулим значенням у порівнянні з останніми вхідними даними. Тут ми ініціалізуємо оцінювання дисперсії з 1, що є достатньо довільним вибором.

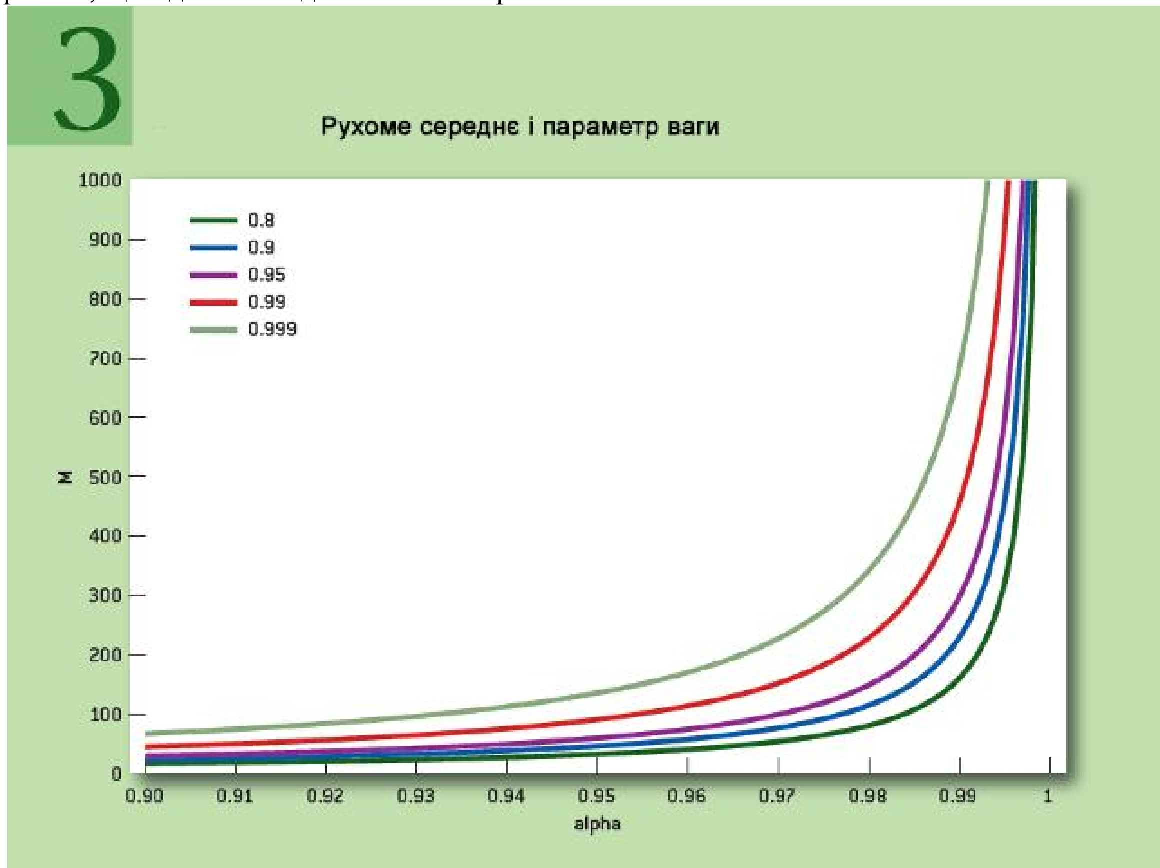

Рис. 3. Рухоме середнє і параметр ваги

Висновки. Отже, сьогодні однопрохідні онлайн-алгоритми грають важливу роль у високочастотній торгівлі, де вони отримують великі об'єми даних кожної мікросекунди і повинні бути взмозі діяти дуже швидко з отриманою інформацією, ефективно використовуючи ресурс пам'яті. Зараз однопрохідні онлайн-алгоритми можуть допомогти вирішити такі проблеми, як оцінка поточної середньої ліквідності, поточна оцінка мінливості і робочої лінійної регресії.

1. Albers S. Online algorithms: a survey / Susanne Albers. // Mathematical Programming. – 2003. – №97. – C. 3–26.
2. Åström K. Adaptive Control: Second Edition / K. Åström, B. Wittenmark. – New York: Dover Publications, 2008. – 573 c.
3. Brown R. Exponential Smoothing for Predicting Demand / Robert Brown. – Cambridge, Massachusetts: Arthur D. Little Inc., 1956. – 15 c.
4. Clark C. Improving speed and transparency of market data [Електронний ресурс] / Colin Clark // Exchanges. – 2011. – Режим доступу до ресурсу: https://exchanges.nyx.com/cclark/improving-speed-and-transparency-market-data
5. HFT (высокочастотный трейдинг) [Електронний ресурс] – Режим доступу до ресурсу: https://fomag.ru/investopedia/hft-vysokochastotnyy-treyding/
6. Loveless J. The challenges faced by competing HFT algorithms / J. Loveless, S. Stoikov, R. Waeber. // Development. – 2013. – volume 11, issue 8. – C. 2–5.