

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерних наук _____
 Кафедра _____ програмної інженерії _____
 Рівень вищої освіти _____ другий (магістерський) _____
 Спеціальність _____ 121 – Інженерія програмного забезпечення _____
 Тип програми _____ освітньо-наукова програма _____
 Освітня програма _____ Інженерія програмного забезпечення _____
 (шифр і назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

«____» _____ 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові _____ Вовченку Артему Миколайовичу _____

(прізвище, ім'я, по батькові)

1. Тема роботи «Дослідження методів реалізації випадкових біржових ордерів із нелінійною частотою відправлення»

Затверджена наказом по університету від 29.03. 2024 р. № 250 Ст2. Термін подання студентом роботи до екзаменаційної комісії 25.06.2024

3. Вихідні дані до роботи визначення параметрів синтетичного біржового ордера, компоненти алгоритму генерації випадкових біржових ордерів, параметри компонентів алгоритму, оцінка результатів тестування реалізації моделі алгоритму

4. Перелік питань, що потрібно опрацювати в роботі аналіз процесів у біржових у біржових програмних системах, алгоритмічна торгівля, побудова моделі алгоритму, реалізація компонентів алгоритму, аналіз розробленого алгоритму.

КАЛЕНДАРНИЙ ПЛАН

| № | Назва етапів роботи | Термін виконання етапів роботи | Примітка |
|----|--|--------------------------------|-----------------|
| 1 | Видача завдання | 30.03.2024 | <i>виконано</i> |
| 2 | Аналіз предметної галузі | 01.04.2024 | <i>виконано</i> |
| 3 | Постановка задачі | 01.04.2024 | <i>виконано</i> |
| 4 | Експериментальні дослідження | 02.04 – 20.04.24 | <i>виконано</i> |
| 5 | Аналіз результатів експериментальних досліджень та розробка рекомендацій | 20.04 – 23.04.24 | <i>виконано</i> |
| 6 | Написання та оформлення статті та тез доповіді | 17.04 – 26.04.24 | <i>виконано</i> |
| 7 | Підготовка пояснювальної записки | 01.05 – 26.05.24 | <i>виконано</i> |
| 8 | Підготовка презентації та доповіді | 26.05 – 02.05.24 | <i>виконано</i> |
| 9 | Нормоконтроль | 14.06 – 17.06.24 | <i>виконано</i> |
| 10 | Рецензування | 18.06 – 20.06.24 | <i>виконано</i> |
| 11 | Занесення диплома в електронний архів | 22.06.2024 | <i>виконано</i> |
| 12 | Попередній захист | 22.06.2024 | <i>виконано</i> |
| 13 | Допуск до захисту у зав. кафедри | 23.06.2024 | <i>виконано</i> |

Дата видачі завдання «30» березня 2024 р.

Студент _____

(підпис)

Вовченко А.М. _____

Керівник роботи _____

д.т.н проф. Єрохін А. Л.

РЕФЕРАТ / ABSTRACT

Пояснювальна записка містить 64 ст., 6 рис., 3 табл., 22 джерела, 4 додатки

АВТОМАТИЗАЦІЯ ТОРГІВЛІ, АЛГОРИТМІЧНА ТОРГІВЛЯ, БІРЖОВІ ОРДЕРИ, НЕЛІНІЙНА ЧАСТОТА ВІДПРАВЛЕННЯ, РОЗПОДІЛ ВИПАДКОВОЇ ВЕЛИЧИНИ, ГЕНЕРАЦІЯ ОРДЕРІВ, ФІНАНСОВІ БІРЖІ.

Об'єктом дослідження є методи та алгоритми генерації випадкових біржових ордерів з використанням нелінійної частоти відправлення для автоматизації процесів торгівлі на фінансових ринках, симуляції та тренування, або валідації моделей та алгоритмів автоматизованої торгівлі.

У ході дослідження було визначено ключові аспекти функціонування біржового програмного забезпечення, створено та декомпоновано модель алгоритму генерації синтетичних ордерів та створено реалізацію моделі алгоритму з нелінійною частотою відправлення.

Результатом дослідження є реалізація створеної моделі алгоритму. Було проведено аналіз результатів роботи моделі, оцінено якість створених параметрів синтетичних ордерів, та проаналізовано нелінійні часові затримки між відправленнями ордерів.

ALGORITHMIC TRADING, FINANCIAL EXCHANGES, MARKET ORDERS, NONLINEAR DISPATCH FREQUENCY, RANDOM VARIABLE DISTRIBUTION, TRADING AUTOMATION.

The object of this research is the methods and algorithms for generating random exchange orders using nonlinear submission frequency for the automation of trading processes in financial markets, simulation and training, or validation of models and algorithms for automated trading.

During the research, the key aspects of the functioning of exchange software were identified, a model of the algorithm for generating synthetic orders was created and

decomposed, and an implementation of the algorithm model with nonlinear dispatch frequency was developed.

The result of the research is the implementation of the created algorithm model. An analysis of the model's performance was conducted, the quality of the generated parameters of synthetic orders was evaluated, and the nonlinear time delays between order submissions were analyzed.

Я, Вовченко Артем Миколайович, студент гр. ПЗМ-22-5, здобувач вищої освіти на другому (магістерському) рівні кафедри «Програмна інженерія», заявляю: моя кваліфікаційна робота на тему «Дослідження методів реалізації випадкових біржових ордерів із нелінійною частотою відправлення», що буде представлена в екзаменаційну комісію для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу EIArKhNURE. Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлений(на) з діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

ЗМІСТ

| | |
|---|----|
| Вступ..... | 9 |
| 1 Аналіз предметної галузі | 11 |
| 1.1 Загальні відомості..... | 11 |
| 1.2 Різновиди фінансових бірж | 12 |
| 1.3 Різновиди фінансових ордерів..... | 13 |
| 1.4 Функціонування фінансових ринків | 15 |
| 1.5 Алгоритмічна торгівля | 17 |
| 1.6 Алгоритми генерації ордерів | 19 |
| 2 Постановка мети дослідження | 20 |
| 3 Моделювання алгоритму генерації ордерів | 21 |
| 3.1 Визначення алгоритму генерації | 21 |
| 3.2 Генерація компонентів синтетичного ордера | 24 |
| 3.2.1 Генерація випадкової лімітної ціни..... | 24 |
| 3.2.2 Генерація випадкової кількості ордера | 26 |
| 3.2.3 Генерація випадкової категорії ордера..... | 27 |
| 3.2 Визначення випадкового часового лагу відправлення ордера | 28 |
| 3.2.1 Використання джитера для визначення частоти відправлення | 28 |
| 3.2.2 Використання розподілу випадкової величини..... | 29 |
| 4 Програмна реалізація алгоритму..... | 32 |
| 4.1 Деталі програмної реалізації | 32 |
| 4.1.1 Генератор часового лагу..... | 33 |
| 4.1.2 Генератори атрибутів ордера | 34 |
| 4.1.3 Взаємодія компонентів у програмі..... | 36 |
| 4.2 Метод генерації випадкових значень | 36 |
| 4.3 Перевірка роботи розробленої моделі | 38 |
| Висновки..... | 41 |
| Перелік джерел посилання | 42 |
| Перелік джерел посилання за науковими напрямками керівника та науковців кафедри програмної інженерії..... | 45 |

| | |
|-----------------|----|
| Додаток А | 46 |
| Додаток Б | 50 |
| Додаток В..... | 51 |
| Додаток Г | 54 |
| Додаток Д..... | 64 |

ПЕРЕЛІК СКОРОЧЕНЬ

МКТ – ринковий ордер;

LMT – лімітний ордер;

TIF – час дії ордера;

ІОС – виконати ордер негайно та відхилити решту;

FOK – виконати ордер негайно у повному обсязі чи відхилити;

PDF – функція щільності ймовірностей.

ВСТУП

Фінансові біржі виникли в давнину як місця, де торгувалися різні товари, валюти та цінні папери. Спочатку це були прості ринки, де купці й продавці зустрічалися, щоб укласти угоди. Однією з перших відомих фінансових бірж була Амстердамська фондова біржа, заснована у 1602 році. Це сталося після того, як Голландія видала акції Ост-Індської компанії - однієї з перших акціонерних компаній у світі. Ця подія стала каталізатором розвитку фондових бірж [1].

Протягом наступних століть фінансові біржі поширилися по всьому світу. Ці інституції стали центром фінансового життя, де торгувалися акції, облігації, товари та інші фінансові інструменти. У 20 столітті фінансові біржі стали більш сучасними, з впровадженням технологій, таких як телеграф та телефон, що значно полегшили торгівлю [2]. Завдяки цьому біржі стали більш доступними для широкої громадськості. У другій половині 20 століття з'явилися комп'ютеризовані торгові системи, які дозволили виконувати угоди швидше та ефективніше. Це сприяло розвитку глобальних фінансових ринків та збільшило обсяги торгівлі [3].

Сьогодні фінансові біржі залишаються ключовими центрами глобальної економіки, де мільйони торговців з усього світу здійснюють угоди щодня за допомогою передових технологій і засобів зв'язку. Ринки значною мірою впливають на світову економіку, формуючи базис для її подальшого розвитку. До основних факторів впливу на світову економіку можна віднести наступні можливості, які надають ринки:

- залучення довгострокових інвестицій, шляхом випуску акцій, що спрощує фінансування проектів та пришвидшує розвиток компаній;
- визначення оцінки ринкової вартості компаній, що дозволяє інвесторам приймати обґрунтовані рішення та точніше оцінювати потенційні ризики;
- забезпечення прозорості та відповідальності компаній, акції яких торгуються на фінансових ринках, завдяки вимогам до оприлюднення корпоративної інформації [4].

Розвиток автоматизованих систем для забезпечення торгівельного процесу а також поява алгоритмічної торгівлі призвели до необхідності використання програмних інструментів, мета яких – створення несправжньої ліквідності та тестових даних. Для забезпечення процесу тестування сумісності між біржовим та клієнтським програмним забезпеченням нерідко використовуються тестові середовища, транзакції у яких не потребують залучення капіталу. У контексті алгоритмічної торгівлі інструменти створення ліквідності нерідко використовуються для валідації моделей машинного навчання, натренованих з використанням історичних даних біржових транзакцій.

Актуальність даної роботи полягає у тому, що у контексті фінансових ринків, де автоматизація та алгоритмічна торгівля стають все більш поширеними, ефективне управління портфелем та максимізація прибутку потребують нових підходів до генерації торгових ордерів, що можуть використовуватися для владарці моделей штучного інтелекту, створених для автоматичного торгування. Генерація синтетичних ордерів також дозволить забезпечити торгівельну активність у тестових середовищах, які можуть бути використані для досягнення навчальних цілей або проведення тестування [5].

Використання нелінійної частоти відправки ордерів може бути ключем до оптимізації торгових стратегій. Це дозволяє адаптуватися до ринкових умов, які змінюються, та покращити реакцію на ринкову волатильність. Крім того, випадкова генерація ордерів може допомогти уникнути патернів, які можуть бути виявлені та використані іншими учасниками ринку [6].

Отже, дана робота актуальна для торгових фірм, інвестиційних банків, дослідницьких груп та всіх, хто зацікавлений у розробці інноваційних методів управління портфелем та виконанні угод на фондовому ринку.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Загальні відомості

Фінансові біржі – це спеціалізовані ринки, де купують та продають фінансові інструменти, такі як акції, облігації, товари, валюта та інші активи, основною метою яких є забезпечення зручності та ефективності торгівлі для учасників ринку. Основні принципи функціонування включають встановлення правил та регуляцій, механізми співробітництва між учасниками ринку, системи забезпечення безпеки та прозорості операцій [7].

Фінансові ордери – це інструкції, які надаються брокеру або біржі для виконання певних операцій на ринку від імені інвестора. Ордери можуть вказувати на покупку або продаж активу за певною ціною, обсягу чи певному часовому проміжку. Також існують різноманітні типи ордерів, такі як ринкові ордери, лімітні ордери, стоп-лімітні ордери тощо, кожен з яких має свої характеристики та цілі [8].

Загалом, фінансові біржі та фінансові ордери грають ключову роль у функціонуванні фінансових ринків, забезпечуючи ліквідність, прозорість та ефективність торгівлі.

Важливими критеріями при аналізі стану фінансового ринку є його ліквідність, широта та глибина. Ліквідністю активу (наприклад, цінних паперів або інших фінансових інструментів) називають його здатність бути купленими або проданими на ринку без великих змін у їхній ціні. Висока ліквідність означає, що активи легко можуть бути конвертовані у готівку без втрати вартості, тоді як низька ліквідність може призвести до складнощів у продажу активів або зниження їхньої ціни. Ліквідність визначається доступністю покупців та продавців на ринку, обсягом торгівлі, а також характеристиками самого активу [9].

Ширина фінансового ринку - це показник, який вказує на кількість та різноманітність фінансових інструментів, які доступні для торгівлі на конкретному ринку. Цей показник відображає розмаїття активів, таких як акції, облігації, товари, валюта та інші фінансові інструменти, які можуть бути куплені

або продані на ринку. Чим ширший ринок, тим більше можливостей для інвесторів та торгівців і тим більша ліквідність може бути на ринку [10].

Фінансова глибина ринку – норматив, що оцінюється за допомогою ключових показників, що характеризують рівень зрілості фінансової системи у відповідності до міжнародних стандартів. Дослідження цього показника є актуальним, адже недостатня фінансова глибина підриває довгострокову конкурентоспроможність країни та посилює вплив зовнішніх та внутрішніх ризиків на фінансовий ринок [11].

1.2 Різновиди фінансових бірж

Існує кілька видів фінансових ринків, кожен з яких має свої характеристики та цілі. Ось деякі з них:

- ринок акцій, що надає можливість компаніям продавати частки свого капіталу шляхом випуску акцій, а інвесторам ставати власниками компаній та отримувати частину їх прибутку у вигляді дивідендів;
- ринок облігацій, який дозволяє урядам, корпораціям та іншим організаціям залучати капітал шляхом випуску облігацій, що є цінними паперами, які підтверджують зобов'язання з виплати відсотків та повернення вкладених коштів у майбутньому;
- валютний ринок, у рамках якого торгуються різні валютні пари (USD/EUR, EUR/UAH, тощо), учасниками якого можуть бути банки, корпорації, інвестори та урядові установи [12];
- ринок товарів і сировини: який надає можливість торгувати фізичними товарами, такими як нафта, золото, сіль та інші сировини, та включає в себе як фізичну доставку товарів, так і фінансові інструменти, пов'язані з цими товарами, такі як ф'ючерси та опціони [13];
- фондовий ринок: Де торгуються інвестиційні фонди, що складаються з акцій, облігацій та інших цінних паперів. Ці фонди зазвичай мають портфель цінностей, розподілені між різними активами для диверсифікації ризиків [14];

- фондовий ринок індексів: Де торгуються фінансові інструменти, пов'язані з рухами фондових індексів, такі як ф'ючерси на індекси та ETF (фонди з індексним відтворенням).

У роботі за основу було обрано ринки цінних паперів, оскільки вони є найбільш відомими, наочними та зрозумілими для пересічного користувача та для математичного аналізу. Нижче наведений приклад однієї з сучасних фондових бірж – Cryptowatch (див.рис.1.1).

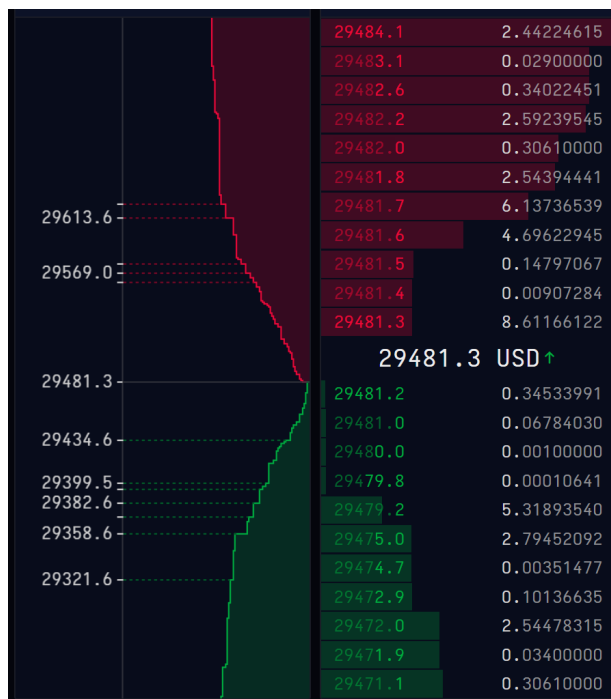


Рисунок 1.1 – Фінансова біржа Cryptowatch

У верхній половині цього рисунка ми бачимо усі ордери на купівлю, у нижній відповідно – на продаж. Таке розташування є досить зручним, оскільки найнижча ціна на купівлю завжди вища за найвищу ціну на продаж. Також для зручності ордери розташовані за однією ціною агреговані в один рядок на графіку.

1.3 Різновиди фінансових ордерів

Основні різновиди фінансових ордерів на ринку цінних паперів включають:

- МКТ: тип ордера вказує брокеру виконати угоду за поточною ринковою ціною; гарантує виконання угоди, але не гарантує конкретну ціну [15];

- LMT: тип ордеру встановлює максимальну ціну для покупки (якщо ордер створено для купівлі) або мінімальну ціну для продажу (якщо ордер створено для продажу), ордер буде виконаний лише за цією ціною або кращою;
- STL (Stop-Limit Order): тип ордеру складається з двох частин: стоп-ціни і лімітної ціни; коли ринкова ціна досягає або перетинає стоп-ціну, ордер перетворюється в лімітний ордер і надсилається на ринок за вказаною лімітною ціною або кращою;
- STP (Stop Order): тип ордеру активується тільки тоді, коли ринкова ціна досягає або перетинає певний рівень (стоп-ціну); після активації ордер стає ринковим і виконується за поточною ринковою ціною;
- TSO (Trailing Stop Order): тип ордеру автоматично коригується відповідно до змін в ринковій ціні. Він встановлюється як відсоток або фіксована відстань від поточної максимальної або мінімальної ціни, і рухається разом з ціною, але лише в одному напрямку.

Ці різновиди ордерів надають учасникам торговельного процесу різні можливості для управління ризиками та максимізації потенційних прибутків на ринку цінних паперів.

Важливим параметром для ордеру є також TIF – інструкція, яка використовується при розміщенні фінансових ордерів для визначення тривалості часу, протягом якого ордер залишатиметься активним до його виконання або анулювання. Вона фактично повідомляє брокеру, як довго ордер має бути дійсним.

Наведемо найбільш поширені значення TIF:

- DAY: ордер дійсний лише протягом торгового дня, в який він розміщений. Якщо він не виконується до кінця торгового дня, він анулюється;
- GTC: ордер залишається активним до тих пір, поки він не буде виконаний або явно анульований власником; деякі брокери можуть мати

максимальний термін (наприклад, тиждень або місяць), після закінчення якого ордер буде анульовано, якщо він не був виконаний;

- ІОС: ордер повинен бути виконаний негайно, будь-яка частина ордера, яка не може бути виконана, анулюється;
- ФОК: ордер повинен бути виконаний негайно і в повному обсязі, якщо весь ордер не може бути виконаний, він анулюється;
- GTD: ордер залишається активним до зазначеної дати та часу, якщо він не виконується до цього часу, він анулюється.

Ці інструкції ТІФ дозволяють учасникам торгівельного процесу керувати своїми ордерами відповідно до їхньої торгової стратегії та ринкових умов, забезпечуючи виконання їхніх ордерів відповідно до їхніх уподобань щодо часу та тривалості.

1.4 Функціонування фінансових ринків

Зазвичай, функціонування фондових ринків реалізовано за допомогою так званої книги ордерів. Книга ордерів — це постійно оновлюваний запис замовлень на купівлю і продаж. Коли учасник торгівельного процесу розміщує лімітне замовлення, воно розміщується в книзі ордерів за вказаною ціною на вказаній стороні. В результаті утворюється двосторонній ринок з різними цінами для покупців і продавців [16].

Книга замовлень поділяється на дві секції: заявка на купівлю (bid) і заявка на продаж (ask). Усі відкриті замовлення на купівлю відображаються на bid-стороні, тоді як усі відкриті замовлення на продаж відображаються на ask-стороні. Книга ордерів також відображає загальний обсяг замовлень на купівлю та продаж на кожному рівні цін.

Нижче наведено типовий приклад книги замовлень. Ми бачимо дві частини книги замовлень поряд: частину "Bid" і частину "Ask". Кожен ряд книги замовлень відповідає пропозиції на купівлю чи продаж за вказаною ціною (стовпці "Buy" та "Sell") і вказаній кількості (стовпець "Quantity"). Для даного ряду може бути одне чи кілька замовлень за одну ціну. Якщо є кілька замовлень,

кількість у стовпці "Orders" дорівнює сумі кількостей різних замовлень. До книги замовлень часто додається графік, який вказує на підсумкову кількість замовлень у книзі за вказаною ціною. Цей графік дає інформацію про ліквідність ринку в термінах розподілу, широти ринку та глибини ринку. Приклади подібних графіків наведені вище (рисунок 1.1 та рисунок 1.2)

| Order book ? | | | | | |
|---|------------|--------------|--------------|------------|-----------|
| Orders | Quantity | Buy | Sell | Quantity | Orders |
| 1 | 153 | 8.94 | 9.87 | 54 | 1 |
| 1 | 35 | 8.73 | 10.05 | 95 | 1 |
| 1 | 31 | 8.52 | 10.13 | 83 | 1 |
| 1 | 74 | 8.49 | 10.15 | 65 | 1 |
| 1 | 161 | 8.42 | 10.16 | 29 | 1 |
| 1 | 45 | 8.12 | 10.32 | 112 | 1 |
| 1 | 117 | 8.09 | 10.43 | 89 | 1 |
| 1 | 21 | 8.00 | 10.46 | 97 | 1 |
| 1 | 148 | 7.92 | 10.80 | 231 | 2 |
| 1 | 78 | 7.69 | 11.21 | 97 | 1 |
| 10 | 863 | Total | Total | 952 | 11 |

Рисунок 1.2. – Приклад книги ордерів

Варто зазначити, що на більшості ринків на значення ціни та кількості діють певні обмеження: мінімальні та максимальні значення а також мінімальна різниця між двома послідовними значеннями. Для ціни ці обмеження позначають як quantity minimum, quantity maximum та quantity multiple, для цін – price minimum, price maximum та price tick відповідно. Обмеження на ціну потрібні задля того, щоб забезпечити стабільність ринку та запобігти різким коливанням цін, обмеження на кількість у свою чергу запобігають зриву ринку через незвичайно великі замовлення, які також можуть спричинити значні коливання цін. Подібні випадки траплялися, наприклад, під час пандемії COVID-19: у 2020 році через безпрецедентну волатильність відбулося кілька спрацьовувань автоматичних вимикачів, що демонструє важливість таких механізмів під час екстремальних ринкових умов. [17]

Окрім ціни, кількості та сторони ордера для подальшого аналізу треба також визначити поняття пасивних (making) та агресивних (aggressive) ордерів.

З наведених вище прикладів можна побачити декілька властивостей, притаманних фондовим ринкам. По-перше, ордери з обох сторін відсортовані за ціною. У першому рядку розташований ордер з найкращою ціною, у останньому з найгіршою. По-друге, з цього випливає, що для bid-сторони найкращий ордер має найвищу ціну, у той час як для ask-сторони – найнижчу. Така логіка є цілком зрозумілою – будь-який продавець зацікавлений продати свій товар за найвищою ціною, також як і будь-який покупець зацікавлений придбати за найнижчою.

Пасивний ордер – це такий ордер, що не може створити торговий договір з іншою стороною, тобто з ціною гіршою, ніж найгірша ціна протилежної сторони. Відповідно, агресивний ордер – це такий ордер, що створює торговий договір, що має ціну кращу, ніж найгірша ціна протилежної сторони.

1.5 Алгоритмічна торгівля

Останнім часом все більше розповсюдженим явищем стає алгоритмічна торгівля. Алгоритмічна торгівля, часто називається алго-торгівлею - це процес використання комп'ютерних алгоритмів для виконання угод на фінансових ринках. Ці алгоритми призначені для виконання попередньо визначених інструкцій, правил або стратегій для автоматичного розміщення замовлень на купівлю або продаж без втручання людини [18].

Деякі популярні приклади подібних алгоритмів:

- алгоритм TWAP (Time Weighted Average Price), що передбачає рівномірне виконання загального обсягу заявки протягом певного проміжку часу шляхом подання заявок за цінами найкращого попиту або пропозиції через рівновеликі інтервали часу;
- алгоритм VWAP (Volume Weighted Average Price), який передбачає рівномірне виконання загального обсягу замовлення за визначену кількість ітерацій протягом певного проміжку часу шляхом подання заявок за цінами найкращого попиту або пропозиції, скоригованих на

визначену величину відсоткового відхилення, але не перевищуючих середньозважену ринкову ціну інструменту, розраховану з моменту запуску алгоритму;

- алгоритм Iceberg, що передбачає виконання загального обсягу замовлення шляхом подання котирувальних заявок з сумарним обсягом, що не перевищує визначену "видиму" кількість. Подання заявок триває до повного виконання загального обсягу замовлення. На деяких біржах, зокрема на LSE (London Stock Exchange), цей алгоритм реалізований на рівні ядра торгової системи, що дозволяє, поряд зі звичайними параметрами заявки, вказати її "видимий" обсяг.

Але найбільший інтерес в контексті даної роботи викликають алгоритми, що визначають саме коли виставляти ордери на ринок. Ось деякі з поширених типів таких алгоритмів:

- TFA (Trend Following Algorithms): ідентифікують та використовують ринкові тренди. Ці алгоритми зазвичай використовують технічні індикатори, такі як ковзні середні (MA), конвергенція/дивергенція ковзної середньої (MACD) та індекс відносної сили (RSI), щоб визначити напрямок тренду та правильний час для входу або виходу з угод [19];
- MRA (Mean Reversion Algorithms) – стратегії повернення до середнього базуються на ідеї, що ціни повернуться до свого середнього або середнього рівня. Ці алгоритми шукають перекуплені або перепродані умови та розміщують угоди відповідно: наприклад, при досягненні нижньої смуги або певного іншого порогу;
- EDA (Event-Driven Algorithms): алгоритми торгують на основі виникнення конкретних подій, таких як оголошення про прибутки, злиття та поглинання або макроекономічні новини, наприклад, на основі очікуваних або фактичних звітів про прибутки.

Саме через нелінійність частоти відправлення, нейронним мережам, що можуть реалізовувати автоматичну торгівлю, буде набагато складніше визначити, коли саме буде надісланий новий ордер і таких чином передбачити наші дії.

1.6 Алгоритми генерації ордерів

На даний момент деякі великі фінансові установи вже використовують алгоритми генерації ордерів у своїх продуктах. Більшим чином вони використовуються у симуляційних програмах, для тестування торгових стратегій. Як приклад таких можна навести XETRA Auction Simulator, TradeStation Simulator, Investopedia Simulator та інші.

На прикладі XETRA Auction Simulator можна побачити, що різноманіття задач, для яких вони використовуються, є досить великим:

- проведення бектестингу стратегій, реалізованих на аукціонах;
- можливість для учасників ринку оцінювати потенціал масштабування стратегій;
- моделювання результатів відкритих, щоденних та закриваючих аукціонів із залученням синтетичних ордерів;
- симуляція ринкових або лімітних ордерів та різного часу подання ордерів.

Нажаль, з'ясувати які саме алгоритми використовують ці компанії не є можливим через комерційну таємницю. Але використання та активний розвиток цього напрямку серед великих фінансових компаній підкреслює важливість та актуальність подібних досліджень.

2 ПОСТАНОВКА МЕТИ ДОСЛІДЖЕННЯ

З наведеного в першій частині роботи аналізу випливає, що алгоритми генерації випадкових біржових ордерів, становлять великий інтерес для сучасних фінансових клієнтських та біржових систем. Слід відзначити, що нелінійна залежність затримок у часі між відправленнями двох ордерів відіграє важливу роль у контексті тренування та валідації фінансових алгоритмів.

Метою дослідження є створення моделі алгоритму генерації випадкових біржових ордерів включно з його компонентами, а саме:

- генератор ціни випадкового ордера;
- генератор розміру випадкового ордера;
- генератор атрибутів випадкового ордера: сторони та TIF параметру;
- генератор випадкового часового лагу між відправленнями двох суміжних ордерів.

Необхідно створити дві моделі генератора випадкової затримки у часі між відправленнями двох синтетичних ордерів:

- генератор лінійної частоти відправлення у комбінації з джитером;
- генератор нелінійної частоти відправлення в основі якого лежить деякий неперервний розподіл випадкової величини.

Для проведення експериментальної частини дослідження у рамках роботи необхідно реалізувати створену модель генерації синтетичних біржових ордерів.

З метою порівняння між генерації з лінійною та нелінійною частотами відправлень та переваги останнього варіанта, треба також реалізувати алгоритм з лінійною частотою. Алгоритми, мають відповідати наступному критерію: результати, що більш наближені до базового значення повинні бути більш ймовірними, аніж віддалені результати. У рамках роботи мають бути обрані розподіл за степеневим законом та експоненційний розподіл.

3 МОДЕЛЮВАННЯ АЛГОРИТМУ ГЕНЕРАЦІЇ ОРДЕРІВ

3.1 Визначення алгоритму генерації

Для генерації довільного біржового ордера необхідно визначити множину обов'язкових параметрів, до яких належать:

- сторона ордера;
- тип ордера;
- кількість одиниць активу, яку необхідно купити, або продати;
- найгірша ціна ордера, за якої біржове програмне забезпечення може виконати ордер;
- TIF параметр, що визначає спеціальні умови виконання ордера та накладає обмеження на час його дії.

Метою довільного алгоритму генерації біржових ордерів є таке визначення наведених параметрів, яке призведе до появи нової заявки на біржі, або ж призведе до торгування синтетичного ордера.

Слід зазначити, що для LMT ордерів обов'язковою є лімітна ціна, яка накладає обмеження на можливість ордера торгуватися з ордерами, які належать до протилежної сторони. Однак цей параметр є недопустимим для MKT ордерів, адже за визначенням цього типу ордера він може бути зіставленим з будь-яким протилежним ордером. Таким чином, ми можемо визначити LMT та MKT ордери як множину їх обов'язкових параметрів, а результат генерації нового ордера як тип-суму множин LMT та MKT ордерів.

Визначення MKT та LMT ордерів можливо спростити до двох категорій:

- категорія пасивних ордерів, що не призводять до запуску механізмів зіставлення ордерів у біржових системах;
- категорія агресивних ордерів, що обов'язково призводять до його зіставлення з ордерами, що належать до протилежної сторони.

До категорії пасивних ордерів завжди відносять LMT ордери, адже лише цей тип ордера може залишатися у переліку ордерів на виконання у біржових системах за стандартних торгівельних умов. Категорія агресивних ордерів, як

правило, представляється MKT ордерами, які можуть зіставитися з протилежними пасивними ордерами. Однак, деяку підмножину LMT ордерів також можливо віднести до категорії агресивних ордерів за наступних умов:

- TIF параметр LMT ордера визначено як IOC;
- TIF параметр LMT ордера визначено як FOK.

За цих умов LMT ордер буде скасовано у разі його неповного виконання біржовою системою, що дозволяє симулювати поведінку MKT ордерів використовуючи LMT ордери.

Категорію агресивних та пасивних LMT ордерів також можливо розширити додавши значення сторони. Тим самим, ми зменшуємо кількість вихідних параметрів алгоритму, визначаючи тип та сторону ордера як його категорію. Для генерації агресивних та пасивних ордерів необхідно визначити чотири категорії ордерів, наведених у таблиці 3.1. Зазначимо, що значення FOK для TIF параметру не використовується, з огляду на те, що біржова система відхилить такий ордер, у разі якщо він не зможе повністю виконатися. Значення IOC дозволяє біржовій системі частково виконати LMT ордер та скасувати невиконану решту агресивного ордера.

Таблиця 3.1 – відповідність параметрів LMT ордера та категорій

| Категорія ордера | Значення сторони | Значення TIF параметра |
|------------------------|------------------|------------------------|
| Пасивний для купівлі | BUY | DAY |
| Пасивний для продажу | SELL | DAY |
| Агресивний для купівлі | BUY | IOC |
| Агресивний для продажу | SELL | IOC |

Наведемо основні відмінності кожної визначеної категорії синтетичних ордерів:

- пасивний ордер для купівлі: має ціну, яка є нижчою за кращу ціну ордерів на продаж, тим самим гарантуючи, що жодного зіставлення не відбудеться, і ордер залишиться активним у біржовій системі;

- пасивний ордер для продажу: має ціну, яка є вищою за кращу ціну ордерів на купівлю, тим самим гарантуючи відсутність зіставлень на стороні біржової системи та збереження ордера для майбутніх операцій;
- агресивний ордер для купівлі: має ціну, яка є вищою або рівною кращій ціні ордерів для продажу, що гарантує хоча б одну операцію зіставлення та рух кращих ринкових цін; ордер має бути скасований біржовою системою у разі його часткового виконання;
- агресивний ордер для продажу: має ціну, яка є нижчою або рівною кращій ціні ордерів для купівлі, що гарантує хоча б одну операцію зіставлення з ордерами для купівлі; ордер має бути скасований біржовою системою у разі його часткового виконання.

Окремим вихідним параметром моделі алгоритму генерації є часовий лаг, з яким синтетичний ордер буде відправлено до торгової системи. Часовий лаг є затримкою у часі між відправленнями двох синтетичних ордерів представлений у вигляді мікросекунд.. Цей компонент множини вихідних параметрів є незалежним від категорії ордера, що згенеровано.

Таким чином можливо визначити модель алгоритму генерації випадкових ордерів як функції над простором випадкових подій (формула 3.1):

$$\mathcal{A}: (\mathcal{C}_p, \mathcal{C}_q, \mathcal{C}_c, \mathcal{C}_\ell, \Omega_p, \Omega_q, \Omega_c, \Omega_l) \rightarrow \mathcal{O} \times \mathbb{N}^+ \quad (3.1)$$

де \mathcal{C}_p – параметри обмеження для випадкової лімітної ціни ордера;

\mathcal{C}_q – параметри генерації випадкової кількості;

\mathcal{C}_c – параметри генерації категорії синтетичного ордера;

\mathcal{C}_ℓ – параметри генерації випадкового часового лагу;

Ω_p – простір випадкових подій генерації лімітної ціни;

Ω_q – простір випадкових подій генерації кількості;

Ω_c – простір випадкових подій генерації категорії лімітного ордера;

Ω_l – простір випадкових подій генерації часового лагу;

\mathcal{O} – множина можливих параметрів синтетичного лімітного ордера;

\mathbb{N}^+ –множина натуральних чисел, що визначають часовий лаг.

Параметри моделі алгоритму генерації синтетичних біржових ордерів дозволяють налаштувати компоненти для генерації параметрів випадкового ордера. Для створення окремого атрибуту кожного ордера використовується відповідна випадкова подія. Вихідні значення алгоритму дозволяють визначити параметри випадкового ордера, а також обрахувати час відправлення нового ордера, використовуючи отримане значення часового лагу [20].

3.2 Генерація компонентів синтетичного ордера

3.2.1 Генерація випадкової лімітної ціни

Для забезпечення максимальної правдоподібності синтетичних цін під час генерації необхідно брати до уваги реальні поточні ціни на цільовий торговий інструмент. Біржові системи також нерідко виставляють обмеження на ціни, серед яких:

- обмеження на мінімальну зміну значення ціни для торгового інструменту;
- мінімальна допустима ціна для торгового інструменту;
- максимальна допустима ціна для торгового інструменту.

Окремо маємо відзначити, що алгоритм випадкових ордерів створюватиме як купівельні ордери, так і ордери для продажу. Це означає, що генератор цін має брати до уваги діапазон кращих цін (відомий як “price spread”), який визначається наступним чином (формула 3.2):

$$\Delta P = \begin{cases} P_s - P_b, & P_s > P_b \\ 0 & \end{cases} \quad (3.2)$$

де P_s – поточна краща ціна ордерів на купівлю інструменту;

P_b – поточна краща ціна ордерів на продаж інструменту.

Можливі випадки коли різниця кращих цін не може бути визначення стандартним шляхом. Прикладом такого випадку є схрещення цін біржових ордерів, тобто ситуація, коли краща ціна купівлі перевищує кращу ціну продажу. За дотриманням стандартних правил торгівлі така ситуація неможлива, однак біржові книги зазвичай схрещуються під час аукціонів на фондових, товарних ринках та ринках похідних інструментів.

У рамках моделі генератора цін визначимо випадкову ціну як (формула 3.3):

$$P = P_B \pm (\Delta R + \Delta P), (P_B, \Delta R, \Delta P) \in \mathbb{R}^+ \quad (3.3)$$

де P_B – базова ціна для створення синтетичної ціни;

ΔR – випадкове відхилення;

ΔP – статичний діапазон кращих цін.

Таким чином задача генерації випадкових цін зводиться до генерації випадкового відхилення. Слід зазначити, що випадкове відхилення не може набувати від’ємних значень. Для обмеження випадкового фактору синтетичної ціни модель має передбачати максимальне допустиме відхилення.

Нами обрано експоненційний розподіл випадкової величини для створення випадкового відхилення від базової ціни, графік PDF якого наведено на рисунку 3.1. У контексті генерації випадкового відхилення для синтетичних цін експоненційний розподіл має низку переваг [21]:

- значення випадкової величини визначено на проміжку дійсних додатних чисел;
- вірогідність отримати менше випадкове число більша;
- простота реалізації та конфігурації моделі, що використовує експоненційний розподіл.

Використання базової ціни дозволяє генератору випадкових цін створювати синтетичні ціни нового ордера, що є подібними до реальних, тобто таких, які на момент генерації існують у біржовій системі.

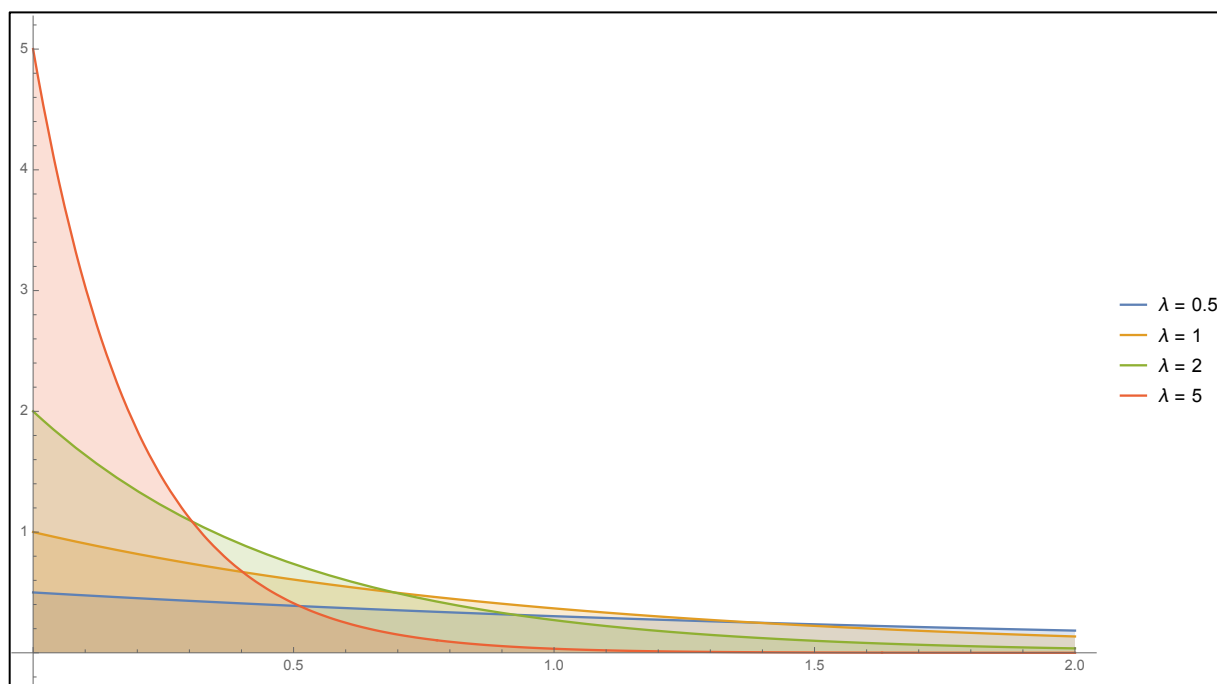


Рисунок 3.1 – Графік PDF для експоненційного розподілу випадкової величини

Такий підхід дозволяє збільшити кількість зіставлень, що, у свою чергу, дозволяє симулювати торгівельний процес для ліквідних інструментів.

3.2.2 Генерація випадкової кількості ордера

У контексті дослідження було обрано використати рівномірний розподіл випадкової величини для створення нової синтетичної кількості. Модель генератора для кількості ордера можна визначити наступним чином (формула 3.4):

$$Q \sim U(Q_{min}, Q_{max}) \quad (3.4)$$

де Q – випадкова кількість, розподілена відповідно до рівномірного розподілу випадкової величини;

Q_{min} – мінімальна допустима кількість нового ордера;

Q_{max} – максимальна допустима кількість нового ордера.

Така модель дозволяє створювати випадкову кількість нового ордера, що задовольняє потенційним вимогам біржових систем до граничних значень кількості ордерів.

3.2.3 Генерація випадкової категорії ордера

Модель визначення категорії нового ордера було побудовано на основі дискретного розподілу з чотирма можливими вихідними випадковими значеннями, кожне з яких відповідає однієї з категорій синтетичного LMT ордера. Визначення вірогідності генерації конкретної категорії ордера відбувається наступним чином (формула 3.5):

$$P(C_i) = \frac{w_i}{w_1 + w_2 + w_3 + w_4}, i \in \{1,2,3,4\} \quad (3.5)$$

де w_i – вага, що відповідає категорії i .

Отримана категорія ордера визначає змінює логіку формування випадкової ціни для ордера (формула 1). Залежність кроків формування випадкової ціни від згенерованої категорії синтетичного ордера наведено у таблиці 3.2.

Категорія випадкового ордера також визначає значення обов'язкових параметрів сторони та TIF. Відповідність категорій до значень параметрів ордера наведено у таблиці 3.1.

Таблиця 3.2 –Формування ціни для категорій синтетичного ордера

| Категорія випадкового ордера | Використання випадкового відхилення ціни ΔR | Використання діапазону кращих цін ΔP |
|------------------------------|---|---|
| Пасивний для купівлі | віднімається від базової ціни P_B | додається до випадкового відхилення ціни ΔR |
| Пасивний для продажу | додається до базової ціни P_B | додається до випадкового відхилення ціни ΔR |
| Агресивний для купівлі | додається до базової ціни P_B | – |
| Агресивний для продажу | додається до базової ціни P_B | – |

Введення категорій ордера дозволяє спростити модель алгоритму генерації шляхом зменшення кількості вхідних параметрів налаштування. Можливим недоліком підходу є той факт, що алгоритм втрачає здатність створювати синтетичні MKT ордери. Однак, деякі біржові системи забороняють використання такого типу ордерів, тож генерація LMT ордерів з різними значенням TIF параметра є більш надійним рішенням.

3.2 Визначення випадкового часового лагу відправлення ордера

Часовий лаг відправлення синтетичного ордера є окремою важливою частиною алгоритму генерації випадкових ордерів. Формально часовий лаг можна визначити наступним чином (формула 3.6):

$$\Delta t = T_2 - T_1 \quad (3.6)$$

де T_2, T_1 – значення часу відправлення двох суміжних ордерів.

Значення часових лагів відправлення ордерів визначає частоту відправлення. У рамках досліджуваної моделі алгоритму ми виділяємо дві категорії частоти відправлення синтетичних ордерів: лінійна та нелінійна частоти відправлень.

Лінійність частоти означає, що час між відправленням двох суміжних ордерів змінюється з постійною швидкістю. З чого випливає, що часових лаг між відправленнями двох ордерів є константою. Нелінійна частота відправлення, з іншої сторони, визначає, що різниця часу відправлення двох суміжних ордерів підпорядковується деякій нелінійній функції.

3.2.1 Використання джитера для визначення частоти відправлення

Джитер – це випадкові коливання або нестабільності в інтервалах часу між послідовними подіями. Для моделювання нелінійної частоти відправлення синтетичних ордерів, джитер представляє випадкове відхилення від очікуваного або середнього інтервалу часу між відправленням. При генерації випадкових

ордерів із нелінійною частотою, джитер може бути змодельований як випадкова величина, що додається до базового інтервалу часу [22].

Для моделювання джитера було обрано дискретний розподіл, що має два можливих значення випадкової величини. Перше значення визначає виконання такту генерації нового ордера та його відправки до торгівельної системи, а друге, відповідно, інструктує модель алгоритму пропустити такт генерації та не виконувати жодних дій. Формально модель генератора випадкового часового лагу з використанням джитера можна визначити наступним чином (формула 3.7):

$$t_{n+1} - t_n = \sum_{i=n}^{n+k} \Delta t \cdot J_i \quad (3.7)$$

де $t_{n+1} - t_n$ – часовий лаг між відправленням ордера n та $n + 1$;

Δt – базовий часовий лаг між тактами генерації;

J_i – випадкова величина, яка приймає значення 0 (виконати генерацію ордера), чи 1 (пропустити такт генерації).

Слід зазначити, що модель нелінійного генератора розширює стандартний генератор лінійного часового лагу, який у всіх випадках повертає однаковий визначений часовий лаг. Нелінійна залежність часових лагів досягається шляхом пропуску довільних тактів генерації в залежності від випадкової величини, отриманої з використанням вказаного дискретного розподілу.

3.2.2 Використання розподілу випадкової величини

Для створення випадкових часових лагів задля досягнення нелінійності частоти відправлення алгоритму можливо використовувати випадкові величини, розподілені за деяким непервинним законом розподілу випадкової величини, що має нелінійну PDF. Прикладами таких розподілів є: нормальний, експоненційний, гамма-розподіл та інші [22, 23].

Для моделювання торгівельного процесу ліквідних інструментів було обрано використати випадкові числа розподілені за степеневим (див. рис. 3.2) та експоненційним законами розподілення випадкової величини (див. рис. 3.1).

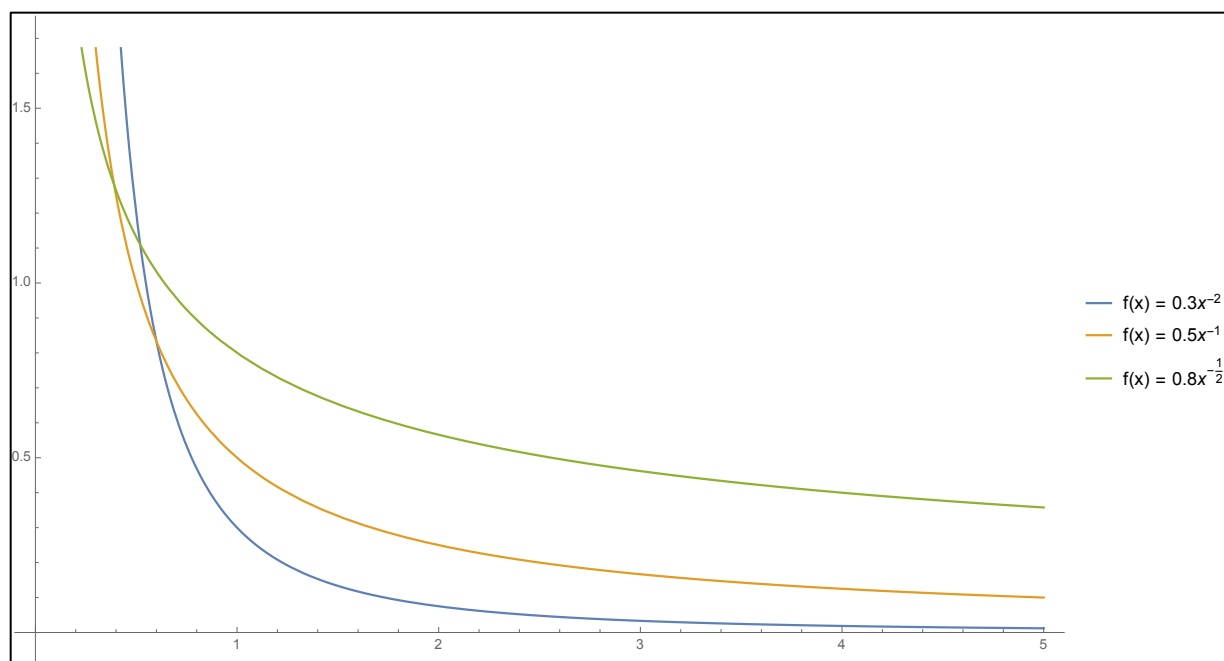


Рисунок 3.2 – графік PDF для степеневих розподілів випадкової величини

Використання вказаних законів розподілу дозволяє значною мірою відтворити деякі характеристики торгівельного процесу, а саме:

- менший лаг має більшу вірогідність бути згенерованим, алгоритм генерації відправлятиме ордери з меншою затримкою частіше;
- степеневий розподіл дозволяє створювати довші випадкові затримки у часі, хоча і зі значно меншою вірогідністю.

Слід зазначити, що модель генератора часових лагів має повертати псевдо-випадкове ціле число у заданому проміжку, який визначає діапазон допустимих лагів алгоритму. Тож, для кожного нового лагу необхідно виконати процес його нормалізації та масштабування до натурального числа, що належить вказаному інтервалу.

Генератор випадкових часових лагів, в основі якого лежить степеневий (формула 3.9) чи експоненційний (формула 3.10) розподіл випадкової величини можна визначити наступним чином:

$$t_{n+1} - t_n = X, \quad (3.9)$$

$$X = \left(\frac{1}{U}\right)^{\frac{1}{\alpha-1}}$$

$$X = -\frac{1}{\lambda} \ln(1 - U) \quad (3.10)$$

де X – випадковий лаг між часами відправлення ордерів t_{n+1} та t_n ;

U – випадкова величина, рівномірно розподілена на інтервалі $(0,1)$;

α – показник степеневого розподілу випадкової величини;

λ – параметр інтенсивності експоненційного розподілу.

Підхід до генерації випадкових часових лагів, заснований на деякій функції розподілу випадкової величини, також може бути інтегрований з джитером. Таким чином згенерований нелінійний лаг може бути скомбіновано алгоритмом з іншими випадковими затримками, що, тим не менш, може нівелювати наведені переваги симуляції торгівлі для ліквідних інструментів.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ АЛГОРИТМУ

4.1 Деталі програмної реалізації

Для реалізації програмного коду було обрано мову програмування C++, версія 20, для побудови додатку – система збірки CMake, для створення тестових даних було створено скрипт на мові програмування Python.

Структурно програму було вирішено поділити на 3 основних компонента: головну частину, генератор випадкового часового лагу та модуль генераторів параметрів синтетичного ордера. Для їх реалізації було обрано шаблон проектування PImpl (Pointer to Implementation) – техніку, що використовується в C++ для досягнення приховування інформації та роз'єднання інтерфейсу і деталей реалізації. Таким чином ми досягаємо наступних переваг:

- інкапсуляція: Приховування деталей реалізації від користувача класу;
- бінарна сумісність: Зміни в реалізації не вимагають перекомпіляції коду, який використовує клас, оскільки інтерфейс залишається незмінним;
- зменшення залежностей на етапі компіляції: Мінімізує залежності, включені до заголовкового файлу, зменшуючи потребу у перекомпіляції великих частин коду при внесенні змін.

Шаблон PImpl зазвичай включає клас, який містить приватний вказівник на непрозорий клас реалізації. Публічний клас (інтерфейс) пересилає виклики до приватної реалізації.

Окрім трьох основних компонентів було також розроблено два допоміжних. Для запобігання дублюванню створено загальний компонент (core), що містить реалізацію алгоритмів розподілу, які використовуються одночасно і в лаунчері і в генераторі. Для більш зручного пошуку помилок під час розробки, а також більш зручного отримання результатів виконання створено логгер (logger). Для реалізації логгера було використано сторонню бібліотеку – spdlog, досить швидко та зручно у використанні. Для завантаження цієї бібліотеки та підключення її до проекту використано пакетний менеджер Conan.

значення типу `bool` (так чи ні). Таким чином ми робимо алгоритми генерації ще більш непередбачуваними, навіть при використанні лінійного розподілу, для сторонніх користувачів, або систем штучного інтелекту, що можуть спробувати проаналізувати їх та певним чином використати ці дані у своїх цілях.

При виборі лінійного розподілу ми передаємо лише один аргумент – затримку між двома послідовно створеними ордерами у мікросекундах (структура `LinearScheduling`). Таким чином за нульової ймовірності джитера ордери завжди будуть генеруватися через рівні проміжки часу.

При виборі стохастичного розподілу потрібно визначити тип використаного розподілу (експоненційний чи степеневий), мінімально та максимально можливий час до наступного ордеру, та також мінімальну різницю між згенерованим часом (структура `StochasticScheduling`). Незалежно від того, який саме розподіл було обрано, ми передаємо один додатковий дійсний параметр – значення α для степеневого розподілу або значення λ – для експоненційного (структури `PowerLawDistribution` та `ExponentialDistribution` відповідно).

В залежності від типу розподілу ми обираємо відповідну реалізацію інтерфейсу `Scheduler`, який використовується у головному класі `Launcher` для генерації випадкового такту. Таким чином, за допомогою обраного нами патерну проектування ми приховуємо усі наведені вище деталі реалізації, відмінності у конфігурації, тощо і передаємо у головний клас однаковий зручний інтерфейс, що є дуже зручним при подальшому використанні.

4.1.2 Генератори атрибутів ордеру

Компонент генератору відповідає для визначення параметрів нового ордеру – у першу чергу його категорії, ціни та кількості. Для кожного з цих параметрів створено окремий, незалежний від інших інтерфейс, його реалізація за допомогою патерну вказівника на реалізацію а також окрему конфігурацію. На рисунку 4.2 наведено діаграму класів генераторів, створених для реалізації алгоритму створення синтетичних ордерів.

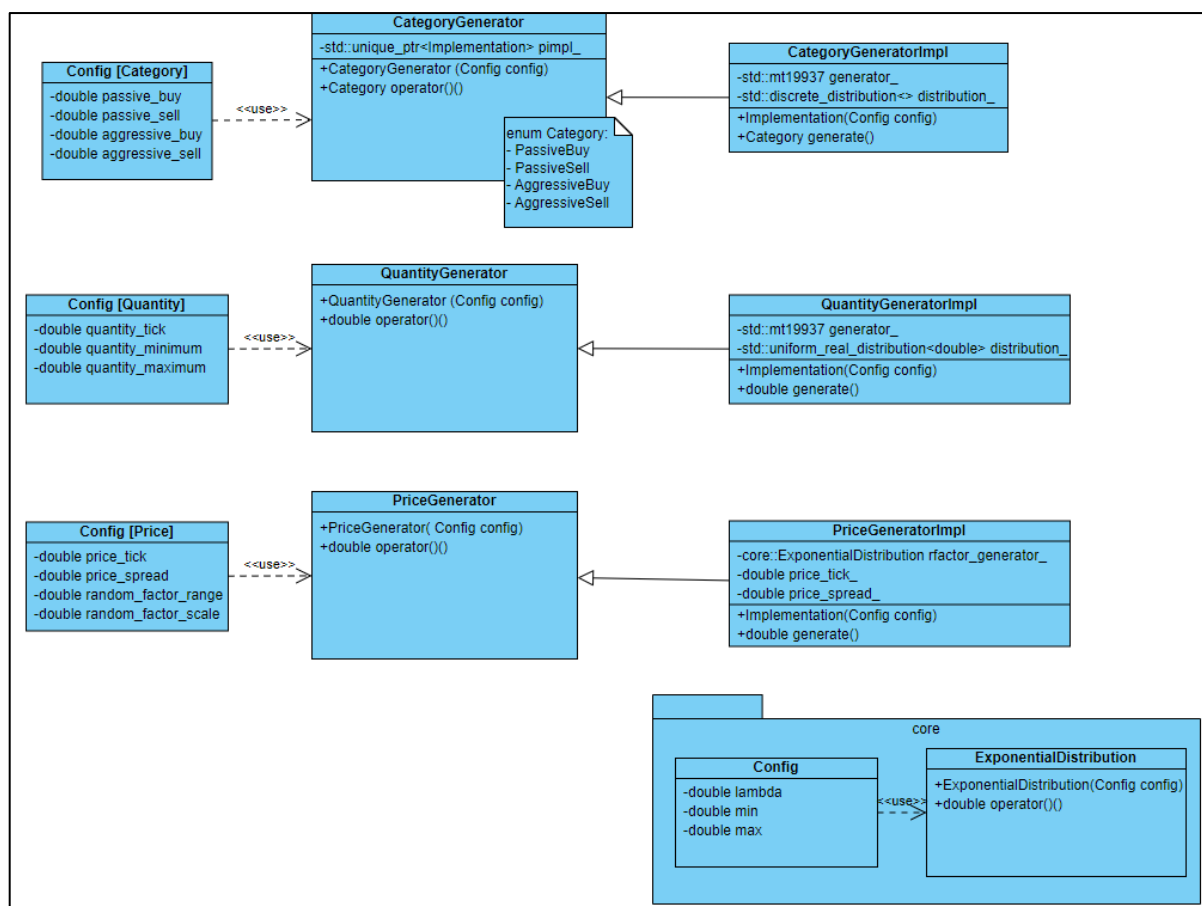


Рисунок 4.2 – Діаграма класів генератору

Розглянемо окремо кожний з наведених інтерфейсів. Генератор категорій визначає, який саме ордер ми будемо створювати – пасивний buy, пасивний sell, агресивний buy чи агресивний sell. Усі ці чотири опції поєднані за допомогою переліку `Category`, саме це значення ми отримуємо від цього генератора. Конфігурація для нього складається з чотирьох дійсних значень – ймовірностей відповідно для кожної з категорій. На кожній окремій ітерації програма обирає одну з них за допомогою дискретного розподілу. При цьому важливо переконатися, щоб сума ймовірностей складала 1 (або 100%).

Генератор кількості приймає у якості конфігурації параметри обмежень, а саме мінімальну та максимальну кількість та розмір кроку між двома послідовними значеннями кількості (`quantity_multiple`). Вибір кількості реалізований досить просто – за допомогою рівномірного розподілу з урахуванням заданого кроку.

Генератор ціни визначає фактично не саму ціну, а її зміщення у напрямку зростання або зменшування. Його функціонування засновано на експоненційному розподілі аналогічного до того, що використовується у лаунчері. Тож аналогічним чином, конфігурація цього генератора переважно складається з аргументів розподілу – максимального значення `random_factor_range` (мінімально вважаємо за 0), мінімального кроку між двома послідовними цінами `price_tick` та значення аргументу розподілу – `random_factor_scale`. Також, у конфігурації ми передаємо `price_spread` – різницю кращих цін між buy та sell-сторонами.

4.1.3 Взаємодія компонентів у програмі

Головний компонент поєднує у собі усі наведені вище компоненти і організовує їх взаємодію таким чином, щоб реалізувати повноцінний процес генерації ордерів. Перш за все за допомогою лаунчера ми отримуємо сигнали з випадковою частотою, що сигналізують про початок створення нового ордеру. Потім, отримуємо категорію нового ордеру, його кількість та зміщення ціни. Остаточна ціна залежить від категорії ордеру, саме тому було вирішено генерувати не ціну, а її відхилення і таким чином, позбавитися прямої залежності між генераторами ціни та категорії. Також саме у головному компоненті ми визначаємо ціну, яку буде обрано за основу і до якої ми будемо застосовувати зміщення.

У залежності від обраної категорії ми визначаємо чи будемо ми додавати (пасивний sell та агресивний buy) або відіймати (пасивний buy та агресивний sell) зміщення а також чи буде застосований spread (тільки для пасивних ордерів). Таким чином ми отримуємо остаточну ціну і, відповідно, усі необхідні параметри нового ордеру.

4.2 Метод генерації випадкових значень

Для реалізації розробленої моделі алгоритму важливим є метод, який буде використано для генерації випадкових чисел, що підпорядковуються заданому

розподілу випадкової величини. Більш того, для створення випадкового часового лагу важливим є генерація саме натуральних чисел у визначеному інтервалі.

Для вирішення поставленої проблеми було використано метод ITS (Inverse Transform Sampling) Метод ґрунтується на використанні зворотної функції розподілу ймовірностей, яка визначається як функція, обернена до кумулятивної функції розподілу деякої випадкової величини [23]. Таким чином, окреме випадкове число, що підпорядковується деякому закону розподілу можна визначити наступним чином (формула 4.1):

$$X = C^{-1}(U) \quad (4.1)$$

де C^{-1} – обернена функція кумулятивного розподілу вірогідності;

U – випадкове число, визначене на проміжку $[0,1]$

ITS було використано для генерації псевдо-випадкових чисел, розподілених за експоненційним законом, який, у рамках досліджуваної моделі, використовується для створення синтетичної ціни та випадкового часового лагу.

Важливим також є той факт, що генератор випадкових експоненційно розподілених чисел має повертати значення з визначеного діапазону. Наведемо формулу обрахування випадкової величини, розподіленої за експоненційним законом (формула 4.2):

$$C^{-1}(u) = -\frac{1}{\lambda} \ln(1 - u), \quad (4.2)$$

$$X = a + \left(C^{-1}(u) \frac{b - a}{\lambda} \right)$$

де $C^{-1}(u)$ – обернена кумулятивна функція експоненційного розподілу;

u – випадкова величина визначена на проміжку $(0,1)$;

λ – параметри масштабування експоненційного розподілу;

a, b – мінімальне та максимальне значення цільового випадкового значення відповідно.

Для визначення випадкових часових затримок, що підпорядковуються степеневому закону, також було використано метод ITS. Для масштабування випадкової величини для степеневого закону було визначено метод обрахування коефіцієнту нормалізації. Таким чином вдалося створити генератор випадкових чисел, що повертає значення, які належать заданому інтервалу. Наведемо формулу обрахування випадкового значення для степеневого закону (формула 4.3):

$$N = b^{\alpha+1} - a^{\alpha+1}, \quad (4.3)$$

$$X = (Nu + a^{\alpha+1})^{\frac{1}{\alpha+1}}$$

де N – коефіцієнт нормалізації степеневого розподілу для визначеного інтервалу;

a, b – мінімальне та максимальне значення вихідної випадкової величини відповідно;

α – показник ступеня степеневого закону розподілу;

u – випадкова величина визначена на проміжку $(0,1)$;

Таким чином для створення випадкових значень, розподілених за степеневим чи експоненційним законами, у контексті моделі алгоритму можливо використовувати лише стандартний генератор рівномірно розподілених дійсних чисел у інтервалі від нуля до одного. Принцип ITS дозволяє визначити математичну залежність між рівномірно розподіленою випадковою величиною та величиною, розподілену за експоненційним чи степеневим законами, що дозволяє значно оптимізувати реалізацію генераторів синтетичної ціни та випадкової часової затримки.

4.3 Перевірка роботи розробленої моделі

Програмна реалізація моделі була протестована. У ході роботи з моделлю були отримані записи про значення згенерованих параметрів кожного синтетичного ордера, а також часовий лаг між відправленнями кожного з двох суміжних ордерів.

Результати генерації випадкової ціни повністю збігаються з передбаченими. На рисунку А.1 наведено гістограму значень цін для синтетичних ордерів, що належать до категорії пасивних, а на рисунку А.2 показано гістограму випадкових цін для агресивних ордерів. Можливо побачити, що для пасивних ордерів генератор випадкових цін враховує значення цінового розриву між кращими цінами на купівлю та продаж. Для обох категорій ордерів ціни розподілені за експоненційним законом у рамках сторони купівлі чи продажу.

Було перевірено згенеровану кількість синтетичних ордерів. Хоча модель генератора кількості, заснована на рівномірному розподілі не, є комплексною, важливо перевірити що розроблений алгоритм створює нову кількість згідно з заданими параметрами. На рисунку А.3 наведено гістограму значень випадкової кількості для синтетичних ордерів. Одразу можна побачити, що значення розподілені за рівномірним законом. Слід також відзначити, що модель коректна, та створює випадкову кількість дотримуючись обмежень на мінімальну та максимально можливі значення кількості нових ордерів.

Визначення категорії випадкового ордеру було налаштовано з параметрами, наведеними у таблиці 4.1. На рисунку А.4 наведено гістограму згенерованих значень категорій ордерів. Як можна побачити, гістограма відповідає наведеним налаштуванням дискретного розподілу.

Таблиця 4.1 – Параметри генерації категорії синтетичного ордеру

| Категорія ордеру | Вірогідність генерації |
|------------------------|------------------------|
| Пасивний для купівлі | 0.3 |
| Пасивний для продажу | 0.3 |
| Агресивний для купівлі | 0.2 |
| Агресивний для продажу | 0.2 |

Розроблену модель було запущено з різними типами генераторів випадкового часового лагу:

- генератор лінійних часових затримок у комбінації з джитером, гістограму часових затримок якого наведено на рисунку А.5;
- генератор нелінійних часових затримок на базі експоненційного розподілу, гістограму часових затримок якого наведено на рисунку А.6;
- генератор нелінійних часових затримок в основі якого лежить степеневий закон розподілу випадкової величини, гістограму часових затримок якого наведено на рисунку А.7.

З неведених діаграм можна зробити висновок, що кожна з реалізацій генератора часового лагу забезпечує нелінійність затримок. Однак найбільш точно торговельний процес ліквідних інструментів описує генератор, в основі якого використано степеневий розподіл випадкової величини, що зумовлено частотою створення довших часових лагів у порівнянні з експоненційним розподілом.

ВИСНОВКИ

У роботі були досліджені та порівняні різні алгоритми генерації біржових ордерів з лінійною та нелінійною частотою. Застосування нелінійної частоти відправлення ордерів може суттєво вплинути на ефективність автоматизованих торгових стратегій. Більш реалістичне моделювання ринкових умов дозволяє точніше тестувати та оптимізувати алгоритми, що може призвести до покращення їх продуктивності у реальних умовах.

Порівняння з алгоритмом, що використовує лінійну частоту відправлення ордерів, показало переваги запропонованого підходу. Нелінійна модель більш точно відтворює статистичні характеристики реальних торговельних даних, такі інтервали між двома послідовно створеними ордерами, ціна та кількість ордеру, тощо.

У рамках роботи було розроблено програмний додаток, що реалізує усі розглянуті алгоритми генерації, отримано результати роботи цих алгоритмів за різних значень вхідних даних, створено та порівняно наочні діаграми на основі цих результатів.

Робота демонструє значний потенціал застосування нелінійних моделей для генерації біржових ордерів. Вона відкриває нові можливості для досліджень та розвитку у сфері фінансових технологій, а також для практичного впровадження у фінансових установах та серед трейдерів. Подальші дослідження можуть бути спрямовані на розширення моделі з урахуванням додаткових факторів, таких як новини, макроекономічні показники або поведінкові аспекти трейдерів. Це дозволить ще більше підвищити точність та адекватність моделювання ринкових умов.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. The trading companies VOC&WIC. URL – <https://www.canonvannederland.nl/en/vocenwic> (дата звернення: 01.04.2024);
2. Гончарук С.М., Долбнєва Д.В., Приймак С.В., Романів Є.М. Фінансовий контроль: теорія, термінологія, практика: навч. посіб. Львів : ЛНУ ім. І. Франка, 2019. 298 с;
3. Колінець Л.Б. Антикризове регулювання економіки в умовах світових економічних криз: орієнтація на реалізацію стратегії економічного розвитку // Економічний аналіз : збірник наукових праць кафедри економічного аналізу Тернопільського національного економічного університету // Тернопіль, 2013. – С. 276-279;
4. Ліфанова М.І. Трансформація міжнародних фінансових центрів у глобальній економіці : дис. ... канд. екон. наук : 08.00.02. Тернопіль, 2018. – с. 202;
5. Економічне управління підприємством // конспект лекцій для студ. денної та заочної форм навч. за спец. 051 "Економіка" // уклад. Л. С. Ларка. Харків, 2017. с 83;
6. Гафаров, Е. М. Алгоритмізація та оптимізація процесів біржової торгівлі. Київ, 2023. - 106 с.;
7. Кубіній Н.Ю., Данаїканич О.В., Кубіній В.В. Методичні аспекти аналізу доходу підприємства. Науковий вісник Ужгородського університету. Серія Економіка. №1 (55). 2020. с. 97-101;
8. Кравчук, І. С. Сучасні тенденції електронної торгівлі обіговими фінансовими інструментами. Вісник ЖДТУ: Економіка, управління та адміністрування. 2018. с. 116–122. [https://doi.org/10.26642/jen-2018-1\(83\)-116-122](https://doi.org/10.26642/jen-2018-1(83)-116-122);
9. Гавкалова Н.Л. Шутєєва О.Ю. Гавкалова Н.Л. Ринок цінних паперів: особливості та фінансове забезпечення його функціонування : монографія. Харків, 2015. – с. 184;
10. Шевченко Д. Особливості формування показників ліквідності у портфельному менеджменті фінансових інститутів. Socio-Economic Relations in the Digital Society, 2 (38), 2020. с. 81–87.;

11. Rekunenko, I. I., Hrytsenko, L. L., Voiarko, I. M., & Kostyrko, R. A. Financial and Credit Activity Problems of Theory and Practice. 2(29), 2019. с. 430 – 439.;
12. Носач Л. Л., Величко К. Ю., Печенка О. І., Зосимова Ж. С., Сокол Н. А. Міжнародні кредитно-розрахункові та валютні операції: навчальний посібник. Форт. Харків, 2016. 240 с.;
13. Світлова А.В. Інноваційні підходи до управління активами на світовому ринку дорогоцінних металів – Полтава, 2022. – 91 с.;
14. Семеняка В. В. Інвестиційні фонди на ринку цінних паперів – Суми, 2022. – 47 с.;
15. Методичні вказівки до практичних занять з курсу "Міжнародні кредитно – розрахункові та валютні операції" : для студентів спец. 073 "Менеджмент", 072 "Фінанси, банківська справа та страхування" // уклад.: С. В. Клепікова, М. М. Шевченко, О. С. Другова ; Нац. техн. ун-т "Харків. політехн. ін-т" // Харків, 2018. – 75 с.;
16. Understanding the Order Book: How It Impacts Trading. URL: – https://www.simtrade.fr/blog_simtrade/understanding-order-book-how-impacts-trading/ (дата звернення 08.05.2024);
17. Курбацький О.С. Система розрахунків арбітражної ситуації з візуальним представленням – Київ, 2021. – 74 с.;
18. Trend Following Algorithms for Technical Trading in Stock Market. URL – https://www.researchgate.net/publication/236973066_Trend_Following_Algorithms_for_Technical_Trading_in_Stock_Market (дата звернення 15.05.2024);
19. XETRA Auction Simulator on A7 Analytics Platform. Modeling the market impact of large orders in auctions. URL – <https://www.mds.deutsche-boerse.com/mds-en/analytics/A7-analytics-platform/xetra-auction-simulator> (дата звернення 15.05.2024);
20. Filatov V. O., Yerokhin A. L., Zolotukhin O. V., Kudryavtseva M. S. Hybrid simulation models for complex decision-making problems with partial uncertainty.

Information Extraction and Processing. 2022, 50(126), 78-86.
DOI:<https://doi.org/10.15407/vidbir2022.50.078>;

21. Dmytro Panchenko, Daniil Maksymenko, Olena Turuta, Andriy Yerokhin, Yana Daniil, Oleksii Turuta . Evaluation and Analysis of the NLP Model Zoo for Ukrainian Text Classification // Communications in Computer and Information Science, 2022, 1698 CCIS, pp. 109–123. DOI: 10.1007/978-3-031-20834-8_6;

22. Daniil Maksymenko, Nataliia Saichyshyna, Oleksii Turuta, Olena Turuta, Andriy Yerokhin, Andrii Babii. Improving the Machine Translation Model in Specific Domains for the Ukrainian Language // International Scientific and Technical Conference on Computer Sciences and Information Technologies, 2022, 2022-November, pp. 123–129. DOI: 10.1109/CSIT56902.2022.10000529.

**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ ЗА НАУКОВИМИ НАПРЯМАМИ
КЕРІВНИКА ТА НАУКОВЦІВ КАФЕДРИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ**

20. Filatov V. O., Yerokhin A. L., Zolotukhin O. V., Kudryavtseva M. S. Hybrid simulation models for complex decision-making problems with partial uncertainty. *Information Extraction and Processing*. 2022, 50(126), 78-86. DOI:<https://doi.org/10.15407/vidbir2022.50.078>;

21. Dmytro Panchenko, Daniil Maksymenko, Olena Turuta, Andriy Yerokhin, Yana Daniil, Oleksii Turuta . Evaluation and Analysis of the NLP Model Zoo for Ukrainian Text Classification // *Communications in Computer and Information Science*, 2022, 1698 CCIS, pp. 109–123. DOI: 10.1007/978-3-031-20834-8_6;

22. Daniil Maksymenko, Nataliia Saichyshyna, Oleksii Turuta, Olena Turuta, Andriy Yerokhin, Andrii Babii. Improving the Machine Translation Model in Specific Domains for the Ukrainian Language // *International Scientific and Technical Conference on Computer Sciences and Information Technologies*, 2022, 2022-November, pp. 123–129. DOI: 10.1109/CSIT56902.2022.10000529./