НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
«КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»

МОГИЛЯНСЬКИЙ МАТЕМАТИЧНИЙ журнал

Том 7 ♦ 2024

Науковий журнал ♦ Щорічник ♦ Заснований у 1996 р.

Київ
2024

Засновник (1996 р.) і видавець журналу — Національний університет «Києво-Могилянська академія»

Виходив як частина багатосерійного видання «Наукові записки НаУКМА» («Фізико-математичні науки»)
З 2018 р. — окреме видання, що має назву «Могилянський математичний журнал» (англ. «Mohyla Mathematical Journal»)

Періодичне наукове видання «Могилянський математичний журнал» засновано з метою публікації результатів науково-дослідних робіт, теоретичних досліджень учених, науково-педагогічних працівників, аспірантів, магістрів і студентів, присвячених широкому колу питань сучасної математичної науки.

Тематичне розмаїття статей охоплює історію математики, виклад результатів теоретичних досліджень з математики і статистики, а також їх застосувань.

Мови видання: українська, англійська.

Здійснюється подвійне анонімне рецензування матеріалів

Засновник і видавець:
Національний університет
«Києво-Могилянська академія»

Ідентифікатор у Реєстрі суб'єктів у сфері медіа:
R40-04348

Внесено до Переліку наукових фахових видань України, в яких можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора наук, кандидата наук та ступеня доктора філософії, категорія «Б» (наказ МОН України від 15.04.2021 № 420)

УДК 511.7

Ліхачов А. Д., Олійник Б. В.

DOI: 10.18523/2617-7080720243-8

СХЕМА РОЗПОДІЛУ СЕКРЕТУ, ЩО БАЗУЄТЬСЯ НА КРИПТОСИСТЕМІ ГОЛДВАССЕР-ГОЛДРІХА-ХАЛЕВІ

З розвитком квантових технологій стає актуальним питання про дослідження та впровадження криптографічних примітивів, що базуються на складних задачах для квантових обчислень. Такі криптографічні примітиви є стійкими щодо квантового криптоаналізу. Прикладом задач, що мають експоненційну складність для квантових обчислень, є задачі на решітках, такі як пошук найкоротшого вектора або пошук найближчого вектора. Однією з перших і найвідоміших квантово-стійких криптосистем, що в основі свого математичного апарату використовує задачі на решітках, є криптосистема Голдвассер-Голдріха-Халеві.

Схема розподілення секрету є фундаментальним криптографічним примітивом, що допускає розподілення секрету між множиною учасників, при цьому відновлення секрету можливе тільки при авторизації всіх або певної частини учасників (порогу учасників). Також необхідною умовою схеми розподілення секрету є неможливість окремих учасників, або груп учасників, кількість яких менша за поріг, відновити секрет.

Варіанти побудови схем розподілу секрету на різних математичних моделях, у тому числі на решітках, наразі активно досліджуються, оскільки вони дозволяють проводити надійні багатосторонні обчислення, безпечно поширювати інформацію шляхом поширення і розподілення оригіналу даних між різними серверами, для побудови компіляторов схем із захистом від витоку тощо. У цій роботі запропоновано нову квантово-стійку n-порогову схему розподілу секрету для n учасників, що базується на криптосистемі Голдвассер-Голдріха-Халеві.

Ключові слова: цілочисельна решітка, алгоритм Бабаї, криптосистема Голдвассер-Голдріха-Халеві, схема розподілу секрету, асиметричний алгоритм шифрування.

Вступ

З розвитком квантових технологій виникає потреба в дослідженнях та впровадженнях криптографічних примітивів, що базуються на складних задачах для квантових обчислень. Такі криптографічні примітиви є стійкими щодо квантового криптоаналізу. Прикладом задач, що є складними для квантових обчислень (тобто мають експоненційну складність), є задачі на решітках, такі як пошук найкоротшого вектора або пошук найближчого вектора. Однією з перших і найвідоміших квантово-стійких криптосистем, яка в основі свого математичного апарату використовує задачу на решітках, є криптосистема Голдвассер-Голдріха-Халеві [3; 4; 10].

Схема розподілення секрету [8] є фундаментальним криптографічним примітивом, що допускає розподілення секрету між множиною учасників, при цьому відновлення секрету можливе тільки при авторизації всіх або певної частини учасників (порогу учасників). Також необхідною умовою схеми розподілення секрету є неможливість окремих учасників, або груп учасників, кількість яких менша за поріг, відновити секрет.

Схеми розподілення секрету використовуються для надійних багатосторонніх обчислень, для побудови компіляторов схем із захистом від витоку [2; 5], для аутсорсингу даних у безпечний спосіб, що дозволяє уникнути необхідності витрачати час на процес шифрування та дешифрування і проблем, які пов'язані з управлінням ключами [9], тощо. Варіанти побудови схем розподілу секрету на різних математичних моделях[2], у тому числі й на решітках [7], наразі активно досліджуються різними науковцями. У цій роботі запропоновано нову n-порогову схему розподілу секрету для n учасників, що базується на криптосистемі Голдвассер-Голдріха-Халеві.

Необхідні визначення

Спочатку нагадаємо основні означення.

Означення 1. [10] Нехай $v_1, \ldots, v_n \in \mathbb{R}^m$ є множиною лінійно незалежних векторів. Решіткою L, породженою $v_1, \ldots, v_n$, називають множину всіх лінійних комбінацій $v_1, \ldots, v_n$ з цілими коефіцієнтами, тобто

$$L = \{a_1v_1 + a_2v_2 + \ldots + a_nv_n \mid a_1, a_2, \ldots, a_n \in \mathbb{Z}\}.$$

«Поганим» базисом решітки L називають менш ортогональний її базис. Відповідно,

«хорошим» базисом решітки є її ортогональний базис або подібний до ортогонального.

Основними задачами на решітках, які є «складними», тобто мають експоненційну складність для звичайних і квантових обчислень, і які використовують у криптографії, є пошук найкоротшого вектора, пошук найближчого вектора та пошук найменшого базису.

Наведемо формулювання цих задач.

Означення 2. [10] Задачу пошуку найкоротшого вектора (Shortest Vector Problem, SVP) у решітці L можна описати одним із трьох таких способів:

- знайти ненульовий вектор x у решітці L, для котрого
$$\|x\| \leq \|y\|$$
для всіх ненульових $y \in L$, тобто $\|x\| = \lambda_1(L)$;
- SVP_γ: знаходження апроксимованого найменшого вектора x у решітці L, який
$$\|x\| \leq \gamma \cdot \lambda_1(L),$$
для малої константи γ;
- $uSVP_\gamma$: для константи $\gamma > 1$ та решітки L таких, що
$$\lambda_2(L) > \gamma \cdot \lambda_1(L),$$
знайти такий ненульовий вектор $x \in L$ довжини $\lambda_1(L)$.

Означення 3. [10] Маючи решітку L у n-вимірному дійсному просторі та $x \in \mathbb{R}^n, x \notin L$, проблему пошуку найближчого вектора (Closest Vector Problem, CVP) можна описати так:

- знайти такий вектор $y \in L$, щоб
$$\|x - y\| \leq \|x - z\|$$
для всіх $z \in L$;
- CVP_γ: знайти такий y, щоб
$$\|x - y\| \leq \gamma \cdot \|x - z\|$$
для всіх $z \in L$ та малої константи γ.

Означення 4. [3] Проблема пошуку найменшого базису (Smallest Basis Problem, SBP) : маючи базис B решітки L, треба знайти інший базис B' для цієї самої решітки, у якому добуток довжин його елементів буде найменшим.

Наведені задачі мають експоненційну складність для звичайних і для квантових обчислень. В загальному випадку наразі не існує алгоритму, за яким можливо було б знайти розв'язки за поліноміальний час, проте для ортогонального базису існують алгоритми, що працюють швидко. Найшвидші алгоритми для розв'язання наведених задач, які перераховані вище, досягають експоненційних множників і базуються на алгоритмі LLL [10]. Алгоритм LLL використовують для знаходження апроксимованого розв'язку задач пошуку найкоротшого вектора та найменшого базису. Для знаходження наближеного розв'язку задачі пошуку найближчого вектора — підхід Бабаї [10] з використанням змінених базисів.

Зауважимо також, що задачі SVP, CVP та SBP є NP–складними задачами [6].

Алгоритм Бабаї

Алгоритм Бабаї [1] використовують для знаходження найближчого вектора (CVP) з наближенням $\gamma = 2^{(n-2)/2}$ за експоненційним часом. Опишемо кроки цього алгоритму.

Нехай $b_1, b_2, \ldots, b_m$ є базисом решітки $L \subset \mathbb{R}^n$ та нехай $x \in \mathbb{R}^n$ є довільним вектором, відстань до якого необхідно знайти. Для цього алгоритму базисні вектори мають бути ортогональними. Якщо вони не є такими, доречним буде застосувати алгоритм LLL [10], який здійснює зміну базису до більш ортогонального. Задачу пошуку найближчого вектора розв'язує такий алгоритм [4]:

1. Визначають рівняння
$$x = t_1 b_1 + t_2 b_2 + \ldots + t_n b_n,$$
у якому коефіцієнти $t_1, t_2, \ldots, t_n \in \mathbb{R}$.
2. Знаходять розв'язок рівняння та призначають $a_i = \lfloor t_i \rceil$ для $i = 1, 2, \ldots, n$.
3. Знаходять вектор y у решітці L, наближений до вектора x:
$$y = a_1 b_1 + a_2 b_2 + \ldots + a_n b_n.$$

Криптосистема Голдвассер-Голдріха-Халеві

У цьому розділі ми наведемо опис криптосистеми з відкритим ключем Голдвассер-Голдріха-Халеві (GGH), яка є однією з перших і найвідоміших криптосистем, що базуються на складних задачах на решітках [3; 4; 10]. Для кращого розуміння, як працює криптосистема, ми опишемо процес шифрування, надсилання і дешифрування інформації як процес обміну інформацією між Алісою та Бобом [4].

Аліса обирає набір лінійно незалежних майже ортогональних між собою векторів
$$v_1, v_2, \ldots, v_n \in \mathbb{Z}^n.$$
Цей набір векторів V вважається «хорошим» базисом, який лежить в основі решітки L та є приватним ключем Аліси. Наступним кроком

генерується випадковим чином матриця U з цілими коефіцієнтами та $det(U) = \pm 1$. Генерування цієї матриці випадковим чином необхідно для знаходження набору векторів, який буде «поганим» базисом — публічним ключем Аліси. Одним із способів отримання матриці U є множення великої кількості випадково обраних елементарних матриць.

$$W = UV$$

Отже, ряд векторів $w_1, w_2, \ldots, w_n$ є новим базисом решітки L та є публічним ключем Аліси.

Тепер припустимо, що Боб хоче переслати повідомлення Алісі. До свого повідомлення m, яке також має векторну форму та може мати вигляд бінарного вектора, Боб додає малий вектор збурення r, який є також ефімерним (сесійним) ключем. Наприклад, координати вектора r можуть бути випадково обрані між δ та $-\delta$, де δ сталий публічний параметр. Далі відбувається процес шифрування:

$$e = mW + r = \sum_{i=1}^{m} m_i w_i + r,$$

де e — шифротекст, та $e \notin L$, але максимально наближений до точки $mW \in L$.

Процес дешифрування відбувається таким чином. Аліса застосовує алгоритм Бабаї з «хорошим» базисом $v_1, v_2, \ldots, v_n$, щоб знайти вектор у решітці L, який буде найближчим до вектора шифротексту e. Враховуючи те, що вона використовує «хороший» (а отже й ортогональний) базис, а вектор збурення r є малим, вектор решітки, який знаходить Аліса, являє собою mW. Помножуючи його на W^{-1}, вона знаходить оригінальне, дешифроване повідомлення від Боба m.

Схема розподілу секрету

Сформулюємо строге означення схеми розподілення секрету. Для цього спочатку нагадаємо визначення k-монотонної структури доступу і визначимо функцію розподілу доступу.

Означення 5 (**k-монотонна схема доступу**)**.** [9] Схема доступу A є монотонною, якщо для будь-якої множини $S \in A$ будь-яка надмножина S також є в A. Ми будемо називати таку схему доступу k-монотонною, якщо для будь-якої множини $S \in A$ буде виконуватись $|S| \geq k$.

Означення 6 (**Функція розподілення доступу**)**.** [9] Нехай $[n] = 1, 2, \ldots, n$ буде множиною ідентичностей n сторін. Нехай M буде областю секретів. Функція розподілення доступу $Share$ — це рандомізоване відображення з M на $S_1 \times S_2 \times \ldots \times S_n$, де S_i є областю поширень сторони з ідентичністю i. Для множини $T \subseteq [n]$ ми визначимо $Share(m)_T$ як обмеження для $Share(m)$ для її T записів.

Означення 7 (**$(A, n, \varepsilon_c, \varepsilon_s)$-Схема розподілення секрету**)**.** Нехай M скінченна множина секретів, де $|M| \geq 2$. Нехай $[n] = 1, 2, \ldots, n$ множиною ідентичностей для n сторін. Функція поширення $Share$ із областю секретів M являє собою $(A, n, \varepsilon_c, \varepsilon_s)$–схему розподілення секрету відносно монотонної схеми доступу A, якщо виконуються такі дві вимоги:

- **Коректність:** секрет може бути відновленим будь-якою множиною сторін, які є частиною схеми доступу A. Тобто для будь-якого набору $T \in A$ існує детермінована функція відновлення $Rec : \otimes_{i \in T} S_i \to M$ така, що кожне $m \in M$,

  $$Pr[Rec(Share(m)_T) = m] = 1 - \varepsilon_c$$

  де ймовірність перевищує випадковість функції $Share$.
- **Статистична конфідеційність:** будь-яке об'єднання сторін, які не є частиною схеми доступу, не повинно мати майже жодної інформації про основний секрет. Тобто для будь-якої неавторизованої множини $U \subseteq [n]$, такої, щоб $U \notin A$, а також для кожної пари секретів $m_0, m_1 \in M$, для кожного обчислювально необмеженого розрізнювача D зі значенням $\{0, 1\}$ має місце таке:

  $$|Pr[D(Share(m_0)_U) = 1] - Pr[D(Share(m_1)_U) = 1]| \leq \varepsilon_s.$$

Визначимо швидкість схеми розподілення секретами як

$$lim_{|m| \to \infty} \frac{|m|}{max_{i \in [n]} |Share(m)_i|}.$$

Означення 8 (**Порогова схема розподілення секрету**)**.** Визначимо t-порогову схему як $(t, n, \varepsilon_c, \varepsilon_s)$-схему розподілення секрету.

Узагальнюючи, схема розподілення секрету для певної схеми доступу та осіб складається з функцій $Share$, яка розподіляє секрет між учасниками схеми доступу, та Rec (Recombine), яка відновлює секрет для певної множини учасників, якщо їх кількості достатньо для відновлення секрету. Поріг схеми розподілення секрету (t, n) визначає, що секрет може бути відтворений t кількістю учасників із загальної кількості n.

Також, схему розподілу секрету вважають безпечною, якщо жоден нескінченно потужний

нападник не може нічого дізнатися про основний секрет, не маючи доступу до монотонної схеми доступу A. Насправді, такі схеми вважають інформаційно-теоретично захищеними, але ми просто називатимемо такі схеми безпечними.

Зараз ми продемонструємо можливість зберігання та відтворення секрету, використовуючи криптосистему Голдвассер-Голдріха-Халеві, яка була описана вище. Нова побудована схема буде n-пороговою, тобто (n, n), а отже відновлення секрету буде можливим лише за умови наявності всіх n сторін.

Кількість базисних векторів решітки напряму залежить від кількості учасників нашої схеми. Нехай ми маємо n учасників. Виберемо цілочисельну решітку L, базис якої складається з n векторів. Позначимо через $v_1, v_2, \ldots, v_n$ лінійно незалежні майже ортогональні вектори, що утворюють решітку L, тобто «хороший» базис нашої решітки.

Секрет, який має бути розподіленим поміж сторонами, позначимо як S. Секрет також має векторний вигляд. Шифруємо його таким чином:

$$S_{enc} = SW + r = \sum_{i=1}^{S} S_i w_i + r,$$

де r — малий вектор збурення, який є також ефімерним ключем.

Кожній стороні схеми видається комбінація (v_i, S_{enc}). Коли всі учасники схеми збирають свої комбінації, ми отримуємо повний «хороший» базис, тоді секрет S можна відновити з допомогою алгоритму Бабаї.

Знаходимо вектор у решітці L, який буде найближчим до вектора S_{enc}. Враховуючи те, що ми маємо «хороший» базис і публічний вектор збурення r є малим, вектор решітки, який ми знаходимо, являє собою SW. Помножуючи його на W^{-1}, ми відновлюємо оригінальний секрет S.

Також можна сформулювати таку теорему.

Теорема 1. *Схема розподілу секрету на основі криптосистеми Голдвассер-Голдріха-Халеві є коректною та статистично конфіденційною.*

Доведення. Розглянута схема є (n, n) пороговою, кожен учасник котрої має пару (v_i, S_{enc}), де v_i — частка лінійно незалежного ортогонального базису (приватного ключа) V, а S_{enc} — зашифрований секрет.

Схема розподілення секрету є коректною, оскільки

$$Pr[Rec(Share(S)_T) = m] = 1,$$

тобто секрет відновлюється з імовірністю 1 тоді і тільки тоді, коли наявний повний «хороший» базис, а це можливо тоді і тільки тоді, коли множина сторін n є повною, тобто наявні всі n сторін.

Схема є також статистично конфіденційною, оскільки за наявності меншої кількості учасників, ніж n, для відновлення секрету потрібно «доповнити»базис до n векторів, вибравши якимось чином вектори, ких не вистачає. При цьому, оскільки базис решітки L складається з n векторів, кожен вектор має не менше ніж n координат. Таким чином у випадку наявності пари секретів m_0 та m_1, навіть $n-1$ кількість учасників не матимуть можливість зрозуміти, до якого секрету відносяться наявні в них ключі, оскільки частинами ключів є пари (v_i, S_{enc}), а сукупність навіть $n-1$-го вектора v_i не дає можливість розшифрувати секрет. А тому вгадати, який саме з двох секретів був зашифрований, можна однаковою ймовірністю, близькою до $\frac{1}{2}$.

Як приклад розглянемо схему з трьома учасниками. Маємо такі три базисні вектори, що формують решітку L:

$$v_1 = \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix}, v_2 = \begin{pmatrix} 2 \\ 1 \\ -2 \end{pmatrix}, v_3 = \begin{pmatrix} 2 \\ -2 \\ 1 \end{pmatrix}.$$

Вони є «хорошим» базисом V.

Маємо секрет $S = (240, 80, 1991)$ і вектор збурення $r = (1, 0, -1)$. Шифруємо наш секрет:

$$S_{enc} = SW + r = \\ = (240, 80, 1991) \begin{pmatrix} 6 & 3 & 3 \\ 3 & 0 & 3 \\ 1 & -1 & -4 \end{pmatrix} + (1, 0, -1) = \\ = (3671, -1271, -7004).$$

Кожен із трьох учасників схеми отримує свою пару (v_i, S_{enc}):

- перший учасник отримує

$$(\begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix}, (3671, -1271, -7004));$$

- другий учасник отримує

$$(\begin{pmatrix} 2 \\ 1 \\ -2 \end{pmatrix}, (3671, -1271, -7004));$$

- третій учасник отримує

$$(\begin{pmatrix} 2 \\ -2 \\ 1 \end{pmatrix}, (3671, -1271, -7004)).$$

Для відновлення секрету застосуємо алгоритм Бабаї. Ми шукатимемо найближчий вектор решітки із базисними векторами V до S_{enc}. Запишемо S_{enc} у вигляді

$$S_{enc} = t_1 v_1 + t_2 v_2 + t_3 v_3,$$

отже отримуємо

$$\begin{pmatrix} 3671 \\ -1271 \\ -7004 \end{pmatrix} = t_1 \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix} + t_2 \begin{pmatrix} 2 \\ 1 \\ -2 \end{pmatrix} + t_3 \begin{pmatrix} 2 \\ -2 \\ 1 \end{pmatrix},$$

розв'язуючи це рівняння знаходимо

$$t_1 = -1431 = a_1, t_2 = 2231 = a_2, t_3 = 320 = a_3.$$

Далі обчислюємо

$$y = a_1 v_1 + a_2 v_2 + a_3 v_3$$

та отримуємо:

$$y = -1431 \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix} + 2231 \begin{pmatrix} 2 \\ 1 \\ -2 \end{pmatrix} + 320 \begin{pmatrix} 2 \\ -2 \\ 1 \end{pmatrix} = \begin{pmatrix} 3671 \\ -1271 \\ -7004 \end{pmatrix},$$

y буде найближчим вектором до S_{enc}. Тепер, щоб відновити секрет S, треба обчислити yW^{-1}:

$$S = yW^{-1} = (3671, -1271, -7004) \times \times \frac{1}{18} \begin{pmatrix} 1 & 3 & 3 \\ 5 & -9 & -3 \\ -1 & 3 & -3 \end{pmatrix} = \begin{pmatrix} 240 \\ 80 \\ 1991 \end{pmatrix}$$

Отриманий дешифрований результат збігається з первинним секретом.

Список літератури

1. Babai L. On Lovász' lattice reduction and the nearest lattice point problem. *Combinatorica*, 1986, Vol. 6. Pp. 1–13. https://doi.org/10.1007/BF02579403.
2. Binu V. P., Sreekumar A. Simple and Efficient Secret Sharing Schemes for Sharing Data and Image. *International Journal of Computer Science and Information Technologies*. 2015. Vol. 6 (1). Pp. 404–409.
3. Goldreich O., Goldwasser S., Halevi S. Publickey cryptosystems from lattice reduction problems. *Proceedings of 17th Annual International Cryptology Conference*. Santa Barbara, California, USA, 1997. Pp. 112–131.
4. . Junying Liang, Haipeng Peng, Lixiang Li, Fenghua Tong, Shuang Bao, Lanlan Wang. A secure and effective image encryption scheme by combining parallel compressed sensing with secret sharing scheme. *Journal of Information Security and Applications*. 2023. Vol. 75. 103487. https://doi.org/10.1016/j.jisa.2023.103487.
5. Hoffstein J., Pipher J., Silverman J. H. An Introduction to Mathematical Cryptography. Springer Science+Business Media, LLC, 2016.
6. Nguyen P. Cryptoanalysis of the Goldreich–Goldwasser–Halevi Cryptosystem from Crypto'97. *Advances in Cryptology — CRYPTO' 99. Lecture Notes in Computer Science*. Vol. 1666, Springer, Berlin, 1999. Pp. 288–304. https://doi.org/10.1007/3-540-48405-1_18.
7. Ravi P., Howe J., Chattopadhyay A., Bhasin S. Lattice-Based Key Sharing Schemes: A Survey. *ACM Computing Surveys*. 2021. Vol. 54, Issue 1, Article No. 9. Pp. 1–39.
8. Shamir A. How to share a secret. *Communications of the Association for Computing Machinery*. 1995. Vol. 22, No. 11. Pp. 612–613.
9. Srinvasan A., Vasudevan P. N. Leakage Resilient Secret Sharing and Applications. *Advances in Cryptology — CRYPTO' 2019. Lecture Notes in Computer Science*. Vol. 11693. Springer, Berlin, 2019. Pp. 480–509.
10. Smart N. P. Cryptography Made Simple. Springer International Publishing Switzerland, 2016.

References

1. L. Babai, Combinatorica. **6**, 1–13 (1986), https://doi.org/10.1007/BF02579403.
2. V. P. Binu and A. Sreekumar, International Journal of Computer Science and Information Technologies. **6** (1), 404–409 (2015).
3. O. Goldreich, S. Goldwasser, and S. Halevi, in: *Proceedings of 17th Annual International Cryptology Conference*(Santa Barbara, California, USA, 1997), pp. 112–131.
4. J. Hoffstein, J. Pipher, and J. H. Silverman, *An Introduction to Mathematical Cryptography* (Springer Science+Business Media, LLC, 2016).
5. Junying Liang, Haipeng Peng, Lixiang Li, Fenghua Tong, Shuang Bao, and Lanlan Wang, Journal of Information Security and Applications. **75** (2023), 103487. https://doi.org/10.1016/j.jisa.2023.103487.
6. P. Nguyen, in: *Advances in Cryptology—CRYPTO' 99. Lecture Notes in Computer Science*, Vol. 1666 (Springer, Berlin, 1999), pp. 288–304. https://doi.org/10.1007/3-540-48405-1_18.
7. . P. Ravi, J. Howe, A. Chattopadhyay, and S. Bhasin, ACM Computing Surveys. **54** (1), 1–39 (2021).
8. A. Shamir, Communications of the Association for Computing Machinery. **22** (11), 612–613 (1995).
9. A. Srinvasan and P. N. Vasudevan, in: *Advances in Cryptology — CRYPTO' 2019. Lecture Notes in Computer Science*, Vol. 11693 (Springer, Berlin, 2019), pp. 480–509.
10. N. P. Smart, *Cryptography Made Simple* (Springer International Publishing Switzerland, 2016).

A. Likhachov, B. Oliynyk

SECRET SHARING SCHEME BASED ON THE GOLDWASSER-GOLDRICH-HALEVI CRYPTOSYSTEM

With the development of quantum technologies, the issue of research and implementation of cryptographic primitives based on complex problems for quantum computing becomes relevant. Such cryptographic primitives are resistant to quantum cryptanalysis. Examples of problems with exponential complexity for quantum computing is lattice problems such as finding the shortest vector or finding the closest vector. One of the first and most famous quantum-resistant cryptosystems that use lattice problems as the basis of its mathematical apparatus is the Goldwasser-Goldrich-Halevi cryptosystem.

A secret distribution scheme is a fundamental cryptographic primitive that allows the distribution of a secret among a set of participants while the secret recovery is possible only when all or a certain part of the participants (the threshold of participants) is authorized. Also, a necessary condition for a secret distribution scheme is the impossibility of individual participants, or groups of participants whose number is less than the threshold, to recover the secret. Variants of constructing secret sharing schemes on various mathematical models, including lattices, are currently being actively studied since they allow for security multiparty calculations and secure information dissemination by distributing the original data between different servers. It is also used for constructing compilers of schemes with protection against leakage, etc. In this paper, a new quantum-stable n-threshold secret sharing scheme for n participants, based on the Goldwasser-Goldrich-Halevi cryptosystem, is proposed.

Keywords: integer lattice, Babai algorithm, Goldwasser-Goldrich-Halevi cryptosystem, secret sharing scheme, asymmetric encryption algorithm.

Матеріал надійшов 07.01.2025

УДК 519.17

Аверкін О. С., Тимошкевич Л. М.

DOI: 10.18523/2617-7080720249-16

ВІДНОВЛЮЮЧЕ СПЕКТРАЛЬНЕ ЧИСЛО ГРАФА K_4

Статтю присвячено дослідженню обернених спектральних задач для зважених графів. Розглянуто задачу щодо відновлення ваг на множині ребер графа за спектрами його індукованих підграфів.

Завдяки широкому колу застосувань, обернені спектральні задачі активно вивчають для різних класів матриць: зазвичай вони зводяться до відновлення матриці (або її частини) за спектром самої матриці чи її підматриць. Наша задача стосується класу нерозкладних симетричних матриць з невід'ємними елементами та нулями на головній діагоналі — матриць суміжності зв'язних зважених графів.

Ключовим поняттям цієї роботи є відновлююче спектральне число графа $Srn(G)$ — мінімальна кількість спектрів індукованих підграфів, необхідних для однозначного відновлення всіх ваг ребер графа G. Головним результатом дослідження є знаходження точного значення $Srn(K_4)$ для повного графа на чотирьох вершинах. Одержані результати та використані у роботі методи можуть бути застосовані в подальших дослідженнях, зокрема для визначення точних значень відновлюючого спектрального числа інших графів.

Ключові слова: спектр графа, власні числа, обернені спектральні задачі, зважений граф.

Вступ

Спектральна теорія графів є сучасним напрямом математики (див. [1]), який активно розвивається завдяки широким можливостям практичного застосування в різних областях, таких як хімія, фізика, біологія, комп'ютерні науки, економіка, соціальні науки та ін. (див. [2]). Наприклад, її використовують у машинному навчанні для покращення роботи згорткових нейронних мереж (див. [3]), у соціології — для пошуку соціальних сенсорів в епідеміологічних мережах (див. [4]).

Важливою складовою спектральної теорії графів є різноманітні задачі відновлення графів, що передбачають визначення структури графа або його характеристик (наприклад, ваг) на основі спектральних даних (див. [5–8]). З оглядом обернених спектральних задач можна ознайомитися за роботою [7].

Дослідження зосереджено на оберненій спектральній задачі для зважених графів, тобто графів, на множині ребер яких визначена додатна функція. У статті [10] вперше було введено поняття $Srn(G)$ — відновлюючого спектрального числа графа G, та знайдено його точні значення для деяких класів графів, зокрема доведено, що для графа-ланцюга $Srn(A_n) = 2$ за $n \geq 3$ та для графа-зірки $Srn(K_{1,n}) = n$.

У роботах [11; 12] наведено верхню оцінку Srn для дерев та уніциклічних графів через кількість висячих вершин. Проте знаходження точного значення $Srn(G)$ навіть для графів G невеликого порядку залишається досить складною задачею через її нелінійний характер та високу чутливість цього параметра до структури графа.

Метою цієї роботи є детальне дослідження оберненої спектральної задачі для повного графа K_4 та знаходження точного значення числа $Srn(K_4)$ (Теорема 2).

Основні означення та твердження

У цій статті під терміном *граф* розуміємо впорядковану пару $G = (V, E)$, в якій V (множина вершин) — деяка непорожня множина, E (множина ребер) — довільна підмножина множини усіх невпорядкованих пар різних елементів з V.

Будемо використовувати такі позначення: $E(G)$ — множина ребер графа G, $V(G)$ — множина вершин графа G, (u, v) — ребро, що з'єднує вершини u і v. Кількість вершин графа називають його порядком. У цій статті розглядаємо лише графи скінченного порядку.

Означення 1. *Зваженим графом* **G** називають впорядковану пару (G, w), де G — граф, а $w : E \to (0, +\infty)$ — вагова функція, яка ставить у відповідність кожному ребру e додатне число $w(e)$.

Зважений граф **G** зручно зображати за допомогою діаграми графа G, приписуючи над кожним ребром e його вагу $w(e)$.

Надалі часто будемо опускати слово «зваже-

ний», якщо з контексту та позначень зрозуміло, що мова йде саме про зважений граф.

Із кожним графом $\mathbf{G} = (G, w)$ та нумерацією його вершин натуральними числами від 1 до n, де n — порядок графа, пов'язують *матрицю суміжності* $A(\mathbf{G}) = (a_{ij})_{i,j=1}^{n}$, в якій елемент a_{ij} i-го рядка та j-го стовпця дорівнює w_{ij}, якщо вершини з номерами i та j є суміжними (через w_{ij} позначаємо значення ваги ребра (i, j)), і дорівнює 0 в іншому випадку.

Означення 2. *Спектром графа* $\mathbf{G}$ називають мультимножину власних значень його матриці суміжності. Позначають $\sigma(\mathbf{G})$.

Оскільки матриця суміжності $A(\mathbf{G})$ симетрична, то спектр графа містить лише дійсні числа.

Зауважимо, що *спектр зваженого графа не залежить від способу нумерації вершин та є його інваріантом.*

Позначимо *характеристичний многочлен графа* $\mathbf{G}$ через $P_{\mathbf{G}}(\lambda) = |\lambda I - A(\mathbf{G})|$.

Якщо $\sigma(\mathbf{G}) = \{\lambda_1, \lambda_2, ..., \lambda_n\}$ — спектр зваженого графа $\mathbf{G}$, тоді $P_{\mathbf{G}}(\lambda) = \prod_{i=1}^{n}(\lambda - \lambda_i)$.

Розглянемо необхідні для формулювання узагальненої теореми Захса означення та позначення (див. [9; 11]).

Лінійний підграф графа G є підграфом, компонентами зв'язності якого є лише пари суміжних вершин (з ребром, що їх з'єднує) та прості цикли. Позначимо його як H_k, де k — це кількість вершин у цьому підграфі. Під вагою компоненти зв'язності, що є парою суміжних вершин $\{i, j\}$, будемо мати на увазі w_{ij}^2, а під вагою компоненти зв'язності, що є простим циклом, — добуток значень w_{ij} за усіма ребрами циклу (i, j).

Введемо позначення: $r(H_k)$ — кількість компонент зв'язності лінійного підграфа H_k; $c(H_k)$ — кількість компонент зв'язності лінійного підграфа H_k, що є циклами; $w(H_k)$ — вага H_k, яка є добутком усіх ваг його компонент зв'язності.

Теорема 1. *(узагальнена теорема Захса) Якщо* $P_{\mathbf{G}}(\lambda) = \sum_{k=0}^{n} c_k \lambda^{n-k} = \lambda^n + c_1\lambda^{n-1} + c_2\lambda^{n-2} + \cdots + c_n$, *— характеристичний многочлен графа* $\mathbf{G} = (G, w)$, *то*

(1) $c_1 = 0$;

(2) $c_2 = -\sum_{e \in E(G)} w(e)^2$

(3) $c_k = \sum_{\{H_k\}} (-1)^{r(H_k)} 2^{c(H_k)} w(H_k)$

для $k = 1, ..., n$, *(сума береться по всіх підграфах* H_k *графа* G*).*

Нагадаємо, що індукований підграф графа G — підграф, утворений підмножиною вершин графа G разом з усіма ребрами G, які сполучають ці вершини.

Означення 3. Зважений граф $\mathbf{G}_1 = (G_1, w_1)$ називають *індукованим підграфом* зваженого графа $\mathbf{G} = (G, w)$, якщо G_1 — індукований підграф G, і для довільного ребра e графа G_1 має місце рівність $w_1(e) = w(e)$.

Розглянемо таку *обернену спектральну задачу* для зваженого графа: нехай нам відомий граф G, і ми хочемо однозначно відновити вагову функцію w зваженого графа $\mathbf{G} = (G, w)$ за спектрами певних його індукованих підграфів. Тобто потрібно, щоби за значеннями спектрів вибраних підграфів ваги на ребрах графа G визначалися однозначно для будь-якої вагової функції w. Спектр індукованого підграфа будемо називати *підспектром.*

Зауважимо, що задача відновлення ваг за спектрами підграфів еквівалентна задачі відновлення за характеристичними многочленами цих підграфів, оскільки за спектром зваженого графа однозначно відновлюється його характеристичний многочлен та навпаки.

Означення 4. *Відновлююче спектральне число* $Srn(G)$ — мінімальна кількість індукованих підграфів G таких, що за спектрами відповідних зважених індукованих підграфів завжди однозначно відновлюється вагова функція зваженого графа $\mathbf{G}$.

Зауваження. Термін *відновлююче спектральне число* використовують відповідно до попередніх публікацій (див. [10–12]), у яких було введено це поняття. З погляду сучасної мовної норми, коректнішою формою є *відновлювальне спектральне число.* Проте задля збереження термінологічної послідовності ми залишаємо усталену назву.

Оскільки вагову функцію будь-якого зваженого графа можна однозначно відновити за спектрами всіх підграфів, породжених парами суміжних вершин, то одержуємо таку верхню оцінку для відновлюючого спектрального числа:

$$Srn(G) \leq |E(G)|.$$

Рівність у цій оцінці досягається, зокрема, для графа-зірки $K_{1,n}$, оскільки в роботі [10] доведено, що $Srn(K_{1,n}) = n$, що відповідає кількості ребер у графі.

Задача відновлення ваг для повного графа $\mathbf{K}_4$

Дослідимо обернену спектральну задачу для $\mathbf{K}_4$. Для зручності як множину вершин оберемо

множину $\{1,2,3,4\}$ та позначимо ваги на його ребрах через $a_1, a_2, ..., a_6$ згідно з рис. 1.

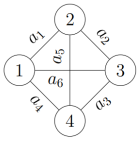


Рис. 1. Зважений граф $\mathbf{K}_4$

З нерівності, наведеної вище, випливає, що шести підспектрів достатньо для відновлення ваг на усіх ребрах. Обґрунтуємо детальніше: для кожного індукованого підграфа $\mathbf{K}_4$, породженого двома вершинами, характеристичний поліном має вигляд $P(\lambda) = \lambda^2 - a_i^2$, тому за шістьма підспектрами можна відновити усі ваги $a_1, a_2, a_3, a_4, a_5, a_6$.

Постає природне запитання: *чи можна відновити ваги графа* $\mathbf{K}_4$ *за п'ятьма підспектрами?*

Далі ми розглянемо всі можливі набори підспектрів (з точністю до перенумерації вершин графа K_4) та покажемо, що п'яти підспектрів недостатньо (а отже, і меншої кількості).

Теорема 2. $Srn(K_4) = 6$.

Доведення. Наведемо всі можливі структури наборів, що складаються з п'яти індукованих підграфів, з точністю до ізоморфізму вибраних підграфів. C_3 позначає цикл довжини 3, а A_2 — ланцюг довжини 1. Число, яке стоїть перед позначенням графа, вказує на кількість підграфів цього виду у наборі.

Маємо такі типи наборів:

1. K_4, $4C_3$;
2. K_4, $3C_3$, A_2;
3. K_4, $2C_3$, $2A_2$;
4. K_4, C_3, $3A_2$;
5. K_4, $4A_2$;
6. $4C_3$, A_2;
7. $3C_3$, $2A_2$;
8. $2C_3$, $3A_2$;
9. C_3, $4A_2$;
10. $5A_2$.

Розглянемо окремо кожен тип набору та наведемо приклади двох різних зважених графів $\mathbf{K}_4$ (інакше кажучи, графів K_4 з різними ваговими функціями, заданими на множині його ребер), для яких спектри відповідних вибраних індукованих підграфів є рівними. Таким чином, ми доведемо, що п'яти підспектрів недостатньо для відновлення ваг на ребрах зваженого графа $\mathbf{K}_4$.

1. $\mathbf{K}_4$***,*** $4\mathbf{C}_3$***.*** Легко переконатися, що характеристичні поліноми усіх циклів довжини три, що є підграфами графів $\mathbf{K}_4$, наведених на рис.2, рівні та мають такий вигляд:

$$P_{\mathbf{C}_3}(\lambda) = \lambda^3 - \lambda(a^2 + b^2 + c^2) - 2abc.$$

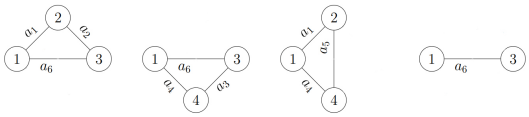
Також характеристичні поліноми графів $\mathbf{K}_4$ з рис.2 рівні, оскільки ці графи переходять один в одного при перестановці вершин 1 та 3, а як відомо, спектр графа є його інваріантом і не залежить від нумерації його вершин.

Рис. 2. Графи $\mathbf{K}_4$ з різними ваговими функціями на ребрах

2. $\mathbf{K}_4$***,*** $3\mathbf{C}_3$***,*** $\mathbf{A}_2$***.*** Без обмеження загальності, можемо вибрати цикли на таких множинах вершин: $\{1,2,3\}, \{1,2,4\}, \{3,4,6\}$. Щодо ланцюга A_2 (далі для зручності будемо називати його ребром), то існують два принципово різні варіанти вибору.

Розглянемо *перший випадок*, коли ребро належить двом з обраних циклів. Наприклад, розглянемо вибір ребра з вагою a_6 (інші варіанти розглядаються аналогічно).

Для кращого розуміння методу знаходження прикладів графів $\mathbf{K}_4$ з різними ваговими функціями, спектри відповідних вибраних індукованих підграфів яких є рівними, розглянемо детальніше цей випадок.



$\mathbf{H}_3^1$ $\mathbf{H}_3^2$ $\mathbf{H}_3^3$ $\mathbf{A}_2^1$

Рис. 3. Набір вибраних індукованих власних підграфів

Запишемо за узагальненою теоремою Захса (Теорема 1) характеристичні многочлени графа $\mathbf{K}_4$ (рис. 1) та його вибраних власних підграфів, зображених на рис. 3:

$$P_{\mathbf{A}_2^1}(\lambda) = \lambda^2 - a_6^2$$
$$P_{\mathbf{H}_3^1}(\lambda) = \lambda^3 - (a_1^2 + a_2^2 + a_6^2)\lambda - 2a_1a_2a_6$$
$$P_{\mathbf{H}_3^2}(\lambda) = \lambda^3 - (a_3^2 + a_4^2 + a_6^2)\lambda - 2a_3a_4a_6$$
$$P_{\mathbf{H}_3^3}(\lambda) = \lambda^3 - (a_1^2 + a_4^2 + a_5^2)\lambda - 2a_1a_4a_5$$
$$P_{\mathbf{K}_4}(\lambda) = \lambda^4 - (a_1^2 + a_2^2 + a_3^2 + a_4^2 + a_5^2 + a_6^2)\lambda^2 - 2(a_1a_4a_5 + a_1a_2a_6 + a_2a_3a_5 + a_3a_4a_6)\lambda + a_1^2a_3^2 + a_2^2a_4^2 + a_5^2a_6^2 - 2(a_1a_2a_3a_4 + a_1a_3a_5a_6 +$$

$+ a_2a_4a_5a_6$).

З характеристичного многочлена $P_{\mathbf{A}_2}(\lambda)$ знаходимо значення ваги a_6. Далі, зі значень коефіцієнтів многочленів $P_{\mathbf{H}_3^1}(\lambda)$, $P_{\mathbf{H}_3^2}(\lambda)$, $P_{\mathbf{K}_4}(\lambda)$ та $P_{\mathbf{H}_3^3}(\lambda)$, одержуємо значення ваги a_5, а також невпорядковані пари значень $\{a_1, a_2\}$, $\{a_3, a_4\}$ та $\{a_1, a_4\}$.

Аналіз показує, що у випадку, якщо $\{a_1, a_2\} = \{a_1, a_4\} = \{a_3, a_4\}$, наприклад, при $a_2 = a_4 = a$, $a_1 = a_3 = b$ та $a \neq b$, впорядкованих набір ваг $(a_1,\ a_2,\ a_3,\ a_4)$ неможливо відновити однозначно. З метою ілюстрації наведемо приклад двох графів $\mathbf{K}_4$, які мають різні вагові функції, але однакові спектри відповідних індукованих підграфів (рис. 4).

Рис. 4

Аналогічно до першого, розглянемо *другий випадок* — коли ребро входить лише до одного з вибраних циклів; наприклад, ребро з вагою a_5. Приклад двох зважених графів, поданих на рис. 4, демонструє неможливість однозначного відновлення вагової функції за відповідним набором підспектрів.

3. $\mathbf{K}_4$, $2\mathbf{C}_3$, $2\mathbf{A}_2$. Можемо розглянути довільні два цикли довжини три. Наприклад, виберемо цикли на вершинах $\{1, 2, 3\}$ та $\{1, 3, 4\}$.

Щодо вибору двох ребер маємо п'ять принципово різних варіантів.

- Спільне ребро вибраних циклів та ребро, яке суміжне з ним, наприклад, ребро $\{1, 2\}$ (рис. 5).

$\mathbf{H}_3^1$ $\mathbf{H}_3^2$ $\mathbf{A}_2^1$ $\mathbf{A}_2^2$

Рис. 5

  Наведемо приклад двох графів $\mathbf{K}_4$, які мають різні вагові функції ($a_3 \neq a_4$), але однакові спектри вибраних підграфів (рис. 6).

Рис. 6

  Наступні три варіанти аналогічні першому.

- Ребро, яке не належить двом вибраним циклам, та суміжне з ним ребро.

- Ребро, яке не належить двом вибраним циклам, та несуміжне з ним ребро.

- Два суміжні ребра в одному з вибраних циклів, які не є спільними для цих двох циклів.

- Два несуміжні ребра, кожне з яких належить рівно одному з двох вибраних циклів. Наприклад, ребра $\{1, 2\}$ та $\{3, 4\}$.
  Приклад двох зважених графів на рис. 7, де $a \neq b$, демонструє неможливість однозначного відновлення вагової функції за відповідним набором підспектрів.

Рис. 7

4. $\mathbf{K}_4$, $\mathbf{C}_3$, $3\mathbf{A}_2$. Цикл можемо вибрати довільно. Розглянемо, наприклад, цикл на вершинах $\{1, 2, 3\}$. Для вибору трьох ребер існує шість принципово різних варіантів.

- Три ребра, що виходять з однієї вершини, яка не належить вибраному циклу, наприклад, з вершини 4. Тобто ребра $\{1, 4\}, \{2, 4\}, \{3, 4\}$. Наведемо приклад двох графів $\mathbf{K}_4$, які мають різні вагові функції, але однакові спектри вибраних підграфів, де a, b, c, d — попарно різні додатні числа (рис. 8).

Рис. 8

- Три ребра, що виходять з однієї вершини, яка належить вибраному циклу, наприклад, з вершини 1.
  На рис. 9 наведено приклад двох графів $\mathbf{K}_4$, які мають різні вагові функції, але однакові спектри вибраних підграфів, де a, b, c — попарно різні додатні числа.

Рис. 9

- Три ребра, які належать вибраному циклу. На рис. 10 наведено приклад двох графів $\mathbf{K}_4$, які мають різні вагові функції, але однакові спектри вибраних підграфів, де $a \neq b$.

Рис. 10

- Три ребра, які належать одному циклу довжини три, відмінному від циклу $\{1,2,3\}$. Наприклад, ребра циклу $\{1,2,4\}$. Приклад двох графів $\mathbf{K}_4$, які мають різні вагові функції, але однакові спектри вибраних підграфів (де $b \neq d$), наведено на рис. 11.

Рис. 11

- Три ребра, що належать одному ланцюгу довжини три, який має з вибраним циклом $\{1,2,3\}$ два спільні ребра. Наприклад, ребра $\{1,2\},\{2,3\},\{3,4\}$. Приклад двох графів $\mathbf{K}_4$, які мають різні вагові функції, але однакові спектри вибраних підграфів (де $a \neq b$), наведено на рис. 12.

Рис. 12

- Три ребра, що належать одному ланцюгу довжини три, який має з вибраним циклом $\{1,2,3\}$ одне спільне ребро. Наприклад, ребра $\{1,2\},\{1,4\},\{3,4\}$. Приклад двох графів $\mathbf{K}_4$, які мають різні вагові функції, але однакові спектри вибраних підграфів (де $a \neq b$), наведено на рис. 13.

Рис. 13

Отже, з наведених прикладів випливає, що цього набору підспектрів недостатньо для однозначного відновлення ваг усіх ребер.

5. $\mathbf{K}_4$**,** $4\mathbf{A}_2$**.** Розглянемо два принципово різні варіанти вибору ребер:

- Чотири ребра, що утворюють цикл довжини 4. Наприклад, це можуть бути ребра $\{1,2\},\{2,3\},\{3,4\},\{4,1\}$. Наведемо приклад двох різних зважених графів $\mathbf{K}_4$, для яких вибрані набори підспектрів є рівними, хоча вагові функції різні при $a_5 \neq a_6$ (рис. 14).

Рис. 14

- Три ребра, що утворюють цикл довжини 3, та ще одне ребро. Наприклад, ребра $\{1,2\},\{2,3\},\{1,3\},\{1,4\}$. Наведемо приклад двох різних зважених графів $\mathbf{K}_4$, для яких вибрані набори підспектрів є рівними, хоча вагові функції різні при $a_3 \neq a_5$ (рис. 15).

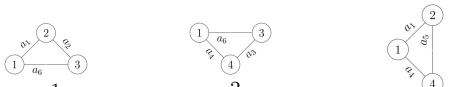
Рис. 15

6. $4\mathbf{C}_3$**,** $\mathbf{A}_2$**.** Неважливо, яке саме ребро вибрати. Наприклад, ребро $\{2,4\}$. Розглянемо наступний приклад двох різних зважених графів $\mathbf{K}_4$, де $a \neq b$, для яких вибрані набори підспектрів є рівними (рис. 16).

Рис. 16

Отже, такий набір підспектрів також не дозволяє однозначно відновити ваги всіх ребер.

7. $3\mathbf{C}_3$, $2\mathbf{A}_2$. Розглянемо довільні три цикли довжини 3. Наприклад, цикли на множинах вершинах {1, 2, 3}, {1, 3, 4}, {1, 2, 4} (рис. 17).



$\mathbf{H}_3^1$ $\mathbf{H}_3^2$ $\mathbf{H}_3^3$

Рис. 17

Щодо вибору двох ребер маємо чотири принципово різні варіанти.

- Два ребра, кожне з яких входить до двох вибраних циклів. Наприклад, розглянемо вибір ребер $\{1,2\},\{1,3\}$.
  Наведемо наступні два різні набори ваг: $a_3 = a_5 = a$, $a_4 = b$ та $a_3 = a_5 = b$, $a_4 = a$, де $a \neq b$. Спектри всіх вибраних підграфів рівні, хоча вагові функції різні (рис. 18).

Рис. 18

- Два несуміжні ребра: одне з них входить до двох вибраних циклів, інше — лише до одного. Наприклад, ребра $\{1,2\},\{3,4\}$. Наведемо наступні два різні набори ваг: $a_2 = a_4 = a$, $a_5 = a_6 = b$ та $a_2 = a_4 = b$, $a_5 = a_6 = a$, де $a \neq b$. Спектри всіх вибраних підграфів рівні, хоча вагові функції різні (рис. 19).

Рис. 19

- Два суміжні ребра: одне з них входить до двох вибраних циклів, інше — лише до одного. Наприклад, ребра $\{1,4\},\{2,4\}$. Наведемо наступні два різні набори ваг: $a_2 = a_3 = a$, $a_6 = b$ та $a_2 = a_3 = b$, $a_6 = a$, де $a \neq b$. Спектри всіх вибраних підграфів рівні, хоча вагові функції різні (рис. 20).

Рис. 20

- Два ребра, які трапляються у вибраних циклах по одному разу. Наприклад, ребра $\{2,3\},\{3,4\}$. Наведемо наступні два різні набори ваг: $a_1 = a_4 = a = 1$, $a_6 = b = 2$, $a_5 = d = 4\sqrt{\frac{2}{5}}$ та $a_1 = a_4 = b = 2$, $a_6 = a = 1$, $a_5 = c = \sqrt{\frac{2}{5}}$. Спектри всіх вибраних підграфів рівні, хоча вагові функції різні (рис. 21).

Рис. 21

Зауважимо, що приклад значень a, b, c, d підібрано таким чином, щоб спектри відповідних підграфів $\mathbf{H}_3^3$ були рівними, тобто цей набір є розв'язком такої системи:

$$\begin{cases} 2b^2 + c^2 = 2a^2 + d^2; \\ b^2 c = a^2 d. \end{cases}$$

8. $2\mathbf{C}_3$, $3\mathbf{A}_2$. Без обмеження загальності, можемо розглянути довільні два цикли довжини три. Наприклад, цикли на вершинах $\{1,2,4\}$ та $\{2,3,4\}$ (рис. 22).

$\mathbf{H}_3^3$ $\mathbf{H}_3^4$

Рис. 22

Щодо вибору трьох ребер: для відновлення всіх ваг необхідно обрати ребро, яке не входить до жодного з вибраних циклів, тобто ребро $\{1,3\}$. Далі є чотири принципово різні варіанти вибору ще двох ребер:

- Два ребра, які належать лише одному з вибраних циклів та не належать іншому. Наприклад, ребра $\{1,2\},\{1,4\}$.
- Два ребра, одне з яких є спільним для вибраних циклів. Наприклад, ребра $\{1,2\},\{2,4\}$.
  В обох цих випадках, коли $a_2 \neq a_3$, можемо поміняти ці ваги місцями, і спектри всіх вибраних графів залишатимуться незмінними, хоча вагові функції будуть різними.
- Два суміжні ребра, але які одночасно не належать жодному з вибраних циклів. Наприклад, ребра $\{1,2\},\{2,3\}$.
  Розглянемо наступні два різні набори ваг: $a_3 = a_4 = a$, $a_5 = b$ та $a_3 = a_4 = b$, $a_5 =$

$= a$, де $a \neq b$ (рис. 23). Спектри всіх вибраних підграфів залишатимуться незмінними, хоча вагові функції будуть різними.

Рис. 23

- Два несуміжні ребра. Наприклад, ребра $\{1,2\},\{3,4\}$. Розглянемо наступні два різні набори ваг: $a_2 = a_4 = a$, $a_5 = b$ та $a_2 = a_4 = b$, $a_5 = a$, де $a \neq b$. Спектри всіх відповідних вибраних підграфів рівні, хоча вагові функції різні.

9. $\mathbf{C_3}$, $4\mathbf{A}_2$. Без обмеження загальності, можемо вибрати довільний цикл довжини три. Щодо вибору ребер, то для відновлення усіх ваг необхідно взяти три ребра, які не належать вибраному циклу. Четверте ребро можна взяти будь-яке з циклу, оскільки кожне з трьох ребер циклу суміжне рівно з двома з решти ребер.

Наприклад, розглянемо цикл на множині вершин $\{1,2,3\}$, а також ребра на наступних парах вершин $\{1,2\},\{3,4\},\{1,4\},\{2,4\}$ (рис. 24).

$\mathbf{H}_3^1$ $\mathbf{A}_2^1$ $\mathbf{A}_2^2$ $\mathbf{A}_2^3$ $\mathbf{A}_2^4$

Рис. 24

Очевидно, що якщо поміняти місцями ваги a_2 і a_6, то спектри вибраних підграфів не зміняться. Таким чином, довільний впорядкований набір додатних ваг, у якому $a_2 \neq a_6$, неможливо однозначно відновити.

10. $5\mathbf{A}_2$. Очевидно, що п'яти довільних ребер недостатньо для відновлення всіх ваг графа $\mathbf{K}_4$, оскільки в такому випадку нічого не буде відомо про вагу шостого ребра.

Проаналізовано всі можливі варіанти вибору п'яти індукованих підграфів K_4 та доведено, що на основі відповідного набору підспектрів неможливо однозначно відновити ваги всіх ребер. А отже, п'яти підспектрів недостатньо для однозначного відновлення ваг у графах типу $\mathbf{K}_4$ в загальному випадку. Оскільки для відновлення ваг завжди достатньо шести спектрів індукованих підграфів, кожен з яких породжений парою суміжних вершин, то $Srn(K_4) = 6$, таким чином, Теорему 2 доведено.

Зауважимо, що з доведеної теореми випливає, що K_4 є нетривіальним прикладом графа, для якого досягається рівність у верхній оцінці відновлюючого спектрального числа:

$$Srn(G) \leq |E(G)|.$$

Список літератури

1. Brouwer A. E., Haemers W. H. Spectra of Graphs. New York : Springer, 2011.
2. Cvetković D. Applications of Graph Spectra: An Introduction to the Literature. *Zbornik Radova*. 2011. Issue 22. Pp. 9–34.
3. Salim A., Sumitra S. Spectral Graph Convolutional Neural Networks in the Context of Regularization Theory. *IEEE Transactions on Neural Networks and Learning Systems*. 2024. Vol. 35, No. 4. Pp. 4373–4384.
4. Sen S., Pal S., Sengupta S. Social Sensors in Epidemiological Networks via Graph Eigenvectors. arXiv preprint arXiv:2112.14385. 2021.
5. Hogben L. Spectral graph theory and the inverse eigenvalue problem of a graph. *Chamchuri Journal of Mathematics*. 2009. Vol. 1. Pp. 51–72.
6. Chu M. T. Inverse eigenvalue problems. *SIAM Review*. 1998. Vol. 40, No. 1. Pp. 1–39.
7. Chu M. T., Golub G. H. Structured inverse eigenvalue problems. *Acta Numerica*. 2002. Vol. 11. Pp. 1–71.
8. Nizhnik L. P., Rabanovich V. I. On new inverse spectral problems for weighted graphs. *Methods of Functional Analysis and Topology*. 2017. Vol. 23, No. 1, Pp. 66–75.
9. Sachs H. Beziehungen zwischen den in einem graphen enthaltenen kreisenund seinem charakteristischen polynom. *Publ. Math. Debrecen*. 1964. Bd. 11. S. 119–134.
10. Тимошкевич Л. М. Обернені спектральні задачі на реберно-зважених графах. *Науковий часопис НПУ імені М. П. Драгоманова*. Серія 1. Фізикоматематичні науки. 2013. Т. 14. С. 165–175.
11. Тимошкевич Л. М. Прямі та обернені спектральні задачі зважених скінченних графів і зліченних графів Коксетра: дис. канд. фіз.-мат. наук: 01.01.06. Київ, 2015. 160 с.
12. Пилипіва О. В., Тимошкевич Л. М. Обернені спектральні задачі для зважених графів. *Могилянський математичний журнал*. 2022. Т. 5. С. 26–32.

References

1. A. E. Brouwer and W. H. Haemers, *Spectra of Graphs* (Springer, New York, 2011).
2. D. Cvetković, Zbornik Radova. **22**, 9–34 (2011).
3. A. Salim and S. Sumitra, IEEE Transactions on Neural Networks and Learning Systems. **35** (4), 4373–4384 (2024).
4. S. Sen, S. Pal, and S. Sengupta, arXiv preprint arXiv:2112.14385 (2021).
5. L. Hogben, Chamchuri Journal of Mathematics. **1**, 51–72 (2009).
6. M. T. Chu, SIAM Review. **40** (1), 1–39 (1998).
7. M. T. Chu and G. H. Golub, Acta Numerica. **11**, 1–71 (2002).
8. L. P. Nizhnik and V. I. Rabanovich, Methods of Functional Analysis and Topology. **23** (1), 66–75 (2017).
9. H. Sachs, Publ. Math. Debrecen. **11**, 119–134 (1964).
10. L. M. Tymoshkevych, Nauk. chasopys NPU imeni M. P. Dragomanova. Seriya 1. Fizyko-matematychni nauky. **14**, 165–175 (2013).
11. L. M. Tymoshkevych, PhD diss., Kyiv, 2015.
12. O. V. Pylypiva and L. M. Tymoshkevych, *Mohylyanskyi matematychnyi zhurnal.* **5**, 26–32 (2022).

O. Averkin, L. Tymoshkevych

SPECTRAL RECONSTRUCTION NUMBER OF GRAPH K_4

In this work, we introduce new formulations of inverse spectral problems for weighted graphs in which certain spectral data (namely, the spectra of selected induced subgraphs) uniquely determine the edge weights of the original graph. To quantify this, we define the spectral reconstruction number of a graph $Srn(G)$ as the minimum number of spectra of induced subgraphs required to uniquely recover all edge weights of G.

Motivated by their broad range of applications, inverse spectral problems for various classes of matrices have been actively studied in the literature. These problems typically involve recovering a matrix, or part of it, from the spectrum of the matrix itself or from the spectra of its submatrices.

From a matrix-theoretic perspective, the problem concerns irreducible symmetric matrices with zero diagonal and nonnegative off-diagonal entries, which are adjacency matrices of connected edge-weighted graphs. Thus, the results obtained here offer new inverse spectral formulations for this class of matrices.

The main contribution of this paper is the exact determination of the spectral reconstruction number for the complete graph on four vertices.

Keywords: spectra of a graph, eigenvalues, inverse spectral problems, weighted graph, subgraphs of a graph.

Матеріал надійшов 10.04.2025

УДК 519.21

Чорней Р. К.

DOI: 10.18523/2617-70807202417-23

ПРО ДЕЯКІ ЗАСТОСУВАННЯ КЕРОВАНИХ ВИПАДКОВИХ ПОЛІВ З ЛОКАЛЬНОЮ СТРУКТУРОЮ ВЗАЄМОДІЇ

У статті розглянуто керовані випадкові поля з локальною структурою взаємодії та їхні застосування. Основну увагу приділено питанням застосування оптимального керування випадковими системами на графах, зокрема в аналізі ризику катастроф, моделюванні соціальних мереж та психометричному мережевому аналізі. Описано математичні підходи, що дозволяють формалізувати та вирішувати задачі стохастичної оптимізації в таких системах. Результати роботи можуть бути застосовані в економіці, кібербезпеці, соціальних науках та інших сферах.

Ключові слова: керовані випадкові поля, локальна взаємодія, оптимальні стратегії.

Вступ

Теорія стохастичної оптимізації і теорія керованих випадкових процесів достатньо повно представлені в науковій і навчальній літературі, натомість теорію керованих випадкових полів наразі висвітлено у відносно невеликій кількості публікацій. При цьому застосування теорії випадкових полів і керування ними потенційно доволі широке. У першу чергу завдяки отриманим результатам щодо існування і пошуку оптимальних стратегій керування випадковими полями для функцій ризику доволі широкого класу. В цій роботі розглядаються деякі можливі застосування теорії керованих випадкових полів з локальною структурою взаємодії до задач оцінки ризику катастроф, для моделювання і аналізу взаємодій у соціальних мережах. а також у психометричному мережевому аналізі.

Керовані випадкові поля з локальною структурою взаємодії

У цій роботі для опису керованих випадкових полів з елементами, що взаємодіють синхронно на підставі інформації про локальний стан, будемо використовувати матеріал роботи [1].

Для систем з локально взаємодіючими координатами структура взаємодії визначається неорієнтованим скінченним графом околів $\Gamma = (V, B)$ без петель і кратних ребер. Граф має множину вершин V і ребер B. Позначимо $\{k, j\}$ ребро графа, що з'єднує вершини k і j. Окіл вершини k — це множина вершин $N(k) = \{j\colon \{k, j\} \in B\}$. Повний окіл вершини k — це $\widetilde{N}(k) = N(k) \cup \{k\}$; тобто окіл вершини k, включно з k. Для всіх підмножин $K \subset V$ визначимо окіл $N(K) = \bigcup\limits_{k\in K} N(k) - K$, і повний окіл $\widetilde{N}(K) = N(K) \cup K$.

Нехай для кожної вершини $i \in V$ $(X_i, \mathfrak{X}_i)$ заданий деякий польський вимірний простір $X_i \neq \varnothing$ з борелівською σ-алгеброю $\mathfrak{X}_i$. X_i, що відповідає $(X_i, \mathfrak{X}_i)$, називатимемо локальним простором станів вершини i. Далі обладнаємо $X := \times\limits_{i\in V} X_i$ мультиплікативною σ-алгеброю $\mathfrak{X} = \sigma\Big\{\times\limits_{i\in V} \mathfrak{X}_i\Big\}$, породженою добутком $\times\limits_{i\in V} \mathfrak{X}_i$. X, що відповідає $(X, \mathfrak{X})$, — глобальний простір станів системи. Для кожної підмножини вершин $K \subset V$ позначимо маргінальний вектор стану $x = (x_i\colon i \in V)$ так: $x_K = (x_k\colon k \in K) \in X_K = \times\limits_{i\in K} X_i$.

Визначимо для $K \subseteq V$ σ-алгебру $\mathfrak{X}_K = \sigma\Big\{\times\limits_{i\in K} \mathfrak{X}_i\Big\} = \bigotimes\limits_{i\in K} \mathfrak{X}_i$, породжену $\times\limits_{i\in K} \mathfrak{X}_i$, а $\mathfrak{X}_V = \mathfrak{X}$.

Означення 1 (див. означення 3.2 в [1])**.** 1) Випадкову величину

$$\xi\colon (\Omega, \mathcal{F}, \Pr) \to (X, \mathfrak{X}) = \Big(\times\limits_{i\in V} X_i, \bigotimes\limits_{i\in V} \mathfrak{X}_i\Big)$$

називають випадковим полем на $\Gamma = (V, B)$ (або просто випадковим полем на V). Для $\varnothing \neq K \subseteq V$ маргінальні випадкові величини зі значеннями в просторі X_K позначатимемо ξ_K. У разі $K = \{k\}$ писатимемо ξ_k.

2) Випадкове поле ξ на (V, B) називають марковським полем, якщо для всіх $k \in V$ виконується:

$$\Pr\big\{\xi_k \in C_k \mid \xi_{V-\{k\}} = x_{V-\{k\}}\big\} =$$

$$= \Pr\{\xi_k \in C_k \mid \xi_{N(k)} = x_{N(k)}\}, \quad \forall\, x \in X, \quad C_k \in \mathfrak{X}_k, \quad (1)$$

де $\Pr\{\xi_k \in C_k \mid \xi_{N(k)} = x_{N(k)}\}$, відповідно $\Pr\{\xi_k \in C_k \mid \xi_{V-\{k\}} = x_{V-\{k\}}\}$, — регулярні умовні ймовірності на $(X_k, \mathfrak{X}_k)$ для заданого $\xi_{N(k)} = x_{N(k)}$, відповідно $\xi_{V-\{k\}} = x_{V-\{k\}}$.

3) (Канонічне випадкове поле) Часто припускатимемо, що $(\Omega, \mathcal{F}, \Pr)$ у частині (1) означення є канонічним простором. Тоді ξ_K і ξ_k — відповідні проекції і $\Pr^{\xi} = \Pr$.

4) (Дискретні стани) Якщо всі локальні простори станів дискретні, то зазвичай утотожнюватимемо розподіл випадкового поля та його зліченні щільності і писатимемо у випадку наявності канонічного основного ймовірнісного простору $\Pr(\xi = x) = \Pr^{\xi}\{x\} = \Pr(x)$, $x \in X$.

Якщо марковський процес ξ за його побудовою відповідає системі околів $\{N(k)\colon k \in V\}$ на (V, B), природно припустити, що у відповідності до еволюції в часі ймовірність події $\{\xi_k^t = x_k\}$ в k-й вершині залежить від попередніх станів усієї системи тільки через значення станів вершин з околу $\widetilde{N}(k)$ (включно з k) у момент часу $t-1$. Для опису цього введемо поняття марковської властивості у просторі і часі відповідно до означення 1.

Означення 2 (див. означення 3.10 в [1]). Нехай $\xi = \{\xi^t, t = 0, 1, \dots\}$, $\xi^t\colon (\Omega, \mathcal{F}, \Pr) \to (X, \mathfrak{X})$ — марковський процес з дискретним часом і простором станів

$$(X, \mathfrak{X}) = \left(\underset{i \in V}{\times} X_i, \bigotimes_{i \in V} \mathfrak{X}_i\right).$$

Перехідні ймовірності ξ називають *локальними* [2, с. 100], якщо

$$\Pr\{\xi_k^{t+1} \in C_k \mid \xi^t = x^t, \dots, \xi^0 = x^0\} = \Pr\{\xi_k^{t+1} \in C_k \mid \xi^t_{\widetilde{N}(k)} = x^t_{\widetilde{N}(k)}\} \quad (2)$$

для $k \in V$, $x^0, \dots, x^t \in X$, $C_k \in \mathfrak{X}_k$ виконується $\Pr^{(\xi^t, \dots, \xi^0)}$-майже всюди; тобто перехідна ймовірність вершини k залежить тільки від стану її повного околу у попередній момент часу.

Перехідні ймовірності ξ називають *синхронними* [2, с. 100], якщо

$$\Pr\{\xi_K^{t+1} \in C_K \mid \xi^t = x^t\} = \prod_{k \in K} \Pr\{\xi_k^{t+1} \in C_k \mid \xi^t = x^t\} \quad (3)$$

для всіх $K \subset V$, $x^t \in X$, $C_K = \underset{k \in K}{\times} C_k \in \mathfrak{X}_K$ виконується $\Pr^{\xi^t}$-майже всюди.

Якщо ξ задовольняє (2) і (3), то її називають *марковським процесом з локально взаємодіючими синхронними компонентами* на (Γ, X), коротко — *(залежне від часу) марковське випадкове поле*.

Далі опишемо структуру моделей прийняття рішень з локальними та синхронними координатами.

Означення 3 (див. означення 3.16 в [1]). Послідовність моментів прийняття рішень (моментів керування) — часова шкала $\mathbb{N}$.

1) Простір рішень (множина керуючих впливів), можливих в моменти прийняття рішень, — $A = \underset{i \in V}{\times} A_i$ на Γ, де A_i — множина можливих дій (рішень) для вершини i. Припускатимемо, що A_i — польський простір з борелівською σ-алгеброю $\mathfrak{A}_i$. $\mathfrak{A}$ — борелівська σ-алгебра добутку на A.

2) Якщо для рішень в вершині i в момент часу t з історією $h^t \in H^t$ множина керуючих впливів обмежена $A_i^t(h^t) \subseteq A_i$, називатимемо $A_i^t(h^t)$ множиною локально допустимих дій (рішень) в момент часу t з історією $h^t \in H^t$.

3) Припускатимемо, що таким чином визначене множиннозначне відображення

$$A_i^t\colon H^t \to 2^{A_i} - \{\varnothing\}, \quad h \to A_i^t(h), \qquad i \in V,$$

залежить тільки від локальної історії і вимірне за Борелем.

Тут для заданої історії

$$h^t = \left(x^0, a^0, x^1, a^1, x^2, \dots, x^{t-1}, a^{t-1}, x^t\right) \in H^t$$

локальна історія $h_i^t \in H_i^t$ вершини i визначається як

$$h_i^t = \left(x^0_{\widetilde{N}(i)}, a_i^0, x^1_{\widetilde{N}(i)}, a_i^1, x^2_{\widetilde{N}(i)}, \dots, x^{t-1}_{\widetilde{N}(i)}, a_i^{t-1}, x^t_{\widetilde{N}(i)}\right) \in H_i^t.$$

H_i^t наділена слідом $\mathfrak{H}_i^t$ σ-алгебри добутку простору $\left(X_{\widetilde{N}(i)} \times A_i\right)^t \times X_{\widetilde{N}(i)}$. Також припускатимемо, що множини $K_i^t = \left\{\left(h_i^t, a_i\right)\colon h_i^t \in H_i^t, a_i \in A_i^t\left(h^t\right)\right\}$ містять графік вимірного відображення і є вимірними за Борелем в сліді σ-алгебри добутку $\mathfrak{K}_i^t := K_i^t \cap \left(\mathfrak{H}_i^t \times \mathfrak{A}_i\right)$.

Крім того, множини

$$\kappa_i^t = \left\{\left(x_{\widetilde{N}(i)}, a_i\right)\colon x_{\widetilde{N}(i)} \in X_{\widetilde{N}(i)}, a_i \in A_i^t\left(x_{\widetilde{N}(i)}\right)\right\}$$

припускаються вимірними за Борелем множинами в просторі $X_{\widetilde{N}(i)} \times A_i$, а $\kappa^t = \underset{i \in V}{\times} \kappa_i^t$ вимірною за Борелем в просторі $\widetilde{X} \times A$, де $\widetilde{X} = \underset{i \in V}{\times} X_{\widetilde{N}(i)}$ і $\widetilde{\mathfrak{X}} = \sigma\left\{\underset{i \in V}{\times} \mathfrak{X}_{\widetilde{N}(i)}\right\}$.

Якщо відображення A_i^t не залежать від t, писатимемо $\kappa_i := \kappa_i^t$, $t \in \mathbb{N}$, і $\kappa := \kappa^t$, $t \in \mathbb{N}$.

Означення 4 (див. означення 3.17 в [1]). Нехай α_i^t — рішення, що обирається в вершині i в момент часу t, $\alpha^t := \big(\alpha_i^t \colon i \in V\big)$ — сумісний вектор рішень в момент часу t.

1) Рандомізована стратегія (політика) $\pi = \big(\pi^t \colon t \in \mathbb{N}\big)$ керованої системи з взаємодіючими компонентами і простором рішень у формі добутку визначається як вектор локальних політик $\pi = (\pi_i, i \in V)$, де для вершини i $\pi_i = \big\{\pi_i^0, \dots, \pi_i^t, \dots\big\}$ — послідовність перехідних імовірностей

$$\pi_i^t = \pi_i^t\big(\cdot \mid x^0, a^0, \dots, x^{t-1}, a^{t-1}, x^t\big).$$

Таким чином, π_i^t — ймовірнісна міра на $(A_i, \mathfrak{A}_i)$ для всіх $\big(x^0, a^0, , \dots, x^{t-1}, a^{t-1}, x^t\big)$ і вимірним чином залежить від історії $h^t = \big(x^0, a^0, \dots, x^{t-1}, a^{t-1}, x^t\big)$ системи до t-го переходу. Отже, маємо для будь-яких $B_i \in \mathfrak{A}_i$

$$\begin{aligned}\Pr\big\{&\alpha_i^t \in B_i \mid \xi^0 = x^0, \alpha^0 = a^1, \dots,\\ &\xi^{t-1} = x^{t-1}, \alpha^{t-1} = a^{t-1}, \xi^t = x^t\big\}\\ &= \pi_i^t\big(B_i \mid x^0, a^0, \dots, x^{t-1}, a^{t-1}, x^t\big). \quad (4)\end{aligned}$$

2) Паралельно з синхронними переходами і локальністю перехідних ядер завжди припускатимемо, що прийняття рішень у вершинах здійснюється умовно незалежно від заданої історії системи. Це призводить до управління процесом, керованим синхронним ядром переходу

$$\begin{aligned}\Pr\Big\{&\alpha^t \in \mathop{\times}_{i\in V} B_i \mid \xi^0 = x^0, \alpha^0 = a^0, \dots\\ &\dots, \xi^{t-1} = x^{t-1}, \alpha^{t-1} = a^{t-1}, \xi^t = x^t\Big\}\\ &= \prod_{i\in V} \Pr\big\{\alpha_i^t \in B_i \mid \xi^0 = x^0, \alpha^0 = a^0, \dots\\ &\dots, \xi^{t-1} = x^{t-1}, \alpha^{t-1} = a^{t-1}, \xi^t = x^t\big\}\\ &= \prod_{i\in V} \pi_i^t\big(B_i \mid x^0, a^0, \dots, x^{t-1}, a^{t-1}, x^t\big),\\ &\qquad B_i \in \mathfrak{A}_i, \quad a^s \in A, \quad x^s \in X. \quad (5)\end{aligned}$$

Означення 5 (див. означення 3.18 в [1]). 1)Нехай в моменти переходу $t = 0, 1, \dots$, множини допустимих рішень згідно з означенням 3 **(3)** залежать тільки від локальної історії, і рішення α_i^t вершини i виробляється відповідно до ймовірності π_i^t на підставі інформації тільки про локальну історію $h_i^t = \Big(x^0_{\widetilde{N}(i)}, a_i^0, \dots, x^{t-1}_{\widetilde{N}(i)}, a_i^{t-1}, x^t_{\widetilde{N}(i)}\Big)$ станів околу $\widetilde{N}(i)$ вершини i і попередніх рішень в i. Якщо $\pi_i^t\Big(A_i^t\big(h_i^t\big) \mid h_i^t\Big) = 1$, то π_i^t називатимемо локально допустимою, а послідовність перехідних імовірностей (рішень) $\pi_i = \big\{\pi_i^t, t \in \mathbb{N}\big\}$ — допустимою локальною стратегією для вершини i.

$\pi = (\pi_i, i \in V)$ називатимемо допустимою локальною стратегією для моделі прийняття рішень.

2) Допустиму локальну стратегію $\pi = (\pi_i, i \in V)$ називають допустимою локальною марковською стратегією, якщо

$$\begin{aligned}\pi_i^t\Big(\cdot \mid x^0_{\widetilde{N}(i)}, a_i^0, \dots, x^{t-1}_{\widetilde{N}(i)}, a_i^{t-1}, x^t_{\widetilde{N}(i)}\Big)\\ = \pi_i^t\Big(\cdot \mid x^t_{\widetilde{N}(i)}\Big), \qquad i \in V.\end{aligned}$$

Зазначимо, що як тільки маємо справу з локальними марковськими стратегіями, можемо припустити, що $A_i^t\big(h_i^t\big)$ залежить тільки від h_i^t через $x^t_{\widetilde{N}(i)}$; ця зменшена залежність виражається як $A_i^t\big(h_i^t\big) =: A_i^t\Big(x^t_{\widetilde{N}(i)}\Big)$.

3) Допустиму локальну марковську стратегію $\pi = (\pi_i, i \in V)$ називають допустимою локальною стаціонарною (марковською) стратегією, якщо $\pi_i^{t'}\Big(\cdot \mid x_{\widetilde{N}(i)}\Big) = \pi_i^{t''}\Big(\cdot \mid x_{\widetilde{N}(i)}\Big)$, $i \in V$, для всіх t', t'' і всіх x.

4) Допустиму локальну стаціонарну (марковську) стратегію $\pi = (\pi_i, i \in V)$ називають допустимою локальною стаціонарною детермінованою (нерандомізованою) стратегією, якщо $\pi_i\Big(\cdot \mid x_{\widetilde{N}(i)}\Big)$, $i \in V$, — одноточкова міра на $A_i^t\Big(x_{\widetilde{N}(i)}\Big)$, $i \in V$, для всіх $x \in X$.

Клас усіх допустимих локальних стратегій позначатимемо LS; підклас допустимих локальних марковських стратегій — LS_M. Як LS_S позначатимемо клас допустимих локальних стаціонарних стратегій, LS_P — клас допустимих локальних детермінованих (= чистих) стратегій, LS_{PM} — клас допустимих локальних стаціонарних (марковських) детермінованих стратегій.

Зауваження 1. Маємо $LS_D \subseteq LS_S \subseteq LS_M \subseteq LS$ і $LS_D \subseteq LS_{PM} \subseteq LS_P \subseteq LS$.

Наша мета полягає в тому, щоб прийти до марковської структури, подібної до означення 2. Побудова процесу приведе нас до припущення, що закони руху для системи можна охарактеризувати інваріантним у часі набором ймовірностей переходу. Тобто, коли система перебуває в стані y і прийнято рішення, що діє a, то, незалежно від своєї історії, наступний стан вибирається відповідно до закону переходу, який залежить тільки від (y, a).

Використовуючи це, стратегія π визначить імовірнісну міру на просторі послідовностей $\big(a^0, a^1, \dots\big)$ для кожного фіксованого початко-

вого стану x^0. Пару (ξ, π) будемо називати керованою версією ξ з використанням стратегії π. Керований процес взагалі не буде марковським, тому що функції π_i^t, $i \in V$, залежать не тільки від станів $x^t_{\widetilde{N}(i)}$, $i \in V$, а й від попередніх (локальних) станів $x^0_{\widetilde{N}(i)}, \ldots, x^{t-1}_{\widetilde{N}(i)}$. Властивість Маркова введена в наступному означенні, після чого обговорюватимуться принципи.

Означення 6 (див. означення 3.21 в [1])**.** Пара (ξ, π) — керований процес з локально взаємодіючими синхронними компонентами, заданий на скінченному графі взаємодій $\Gamma = (V, B)$, якщо $\xi = \left(\xi^t \colon t \in \mathbb{N}\right)$ — процес з простором станів $X = \underset{i \in V}{\times} X_i$ і $\pi = (\pi_i \colon i \in V)$ — допустима локальна стратегія.

Пару (ξ, π) називають керованим марковським процесом з локально взаємодіючими синхронними компонентами на (Γ, X), коротше — (залежне від часу) кероване марковське випадкове поле, якщо переходи ξ визначаються таким чином:

Для всіх t $\Pr^{(\xi^0, \alpha^0, \ldots, \xi^{t-1}, \alpha^{t-1}, \xi^t, \alpha^t)}$-майже всюди визначені умовні ймовірності, що задовольняють для всіх $K \subseteq V$, $C_j \in \mathfrak{X}_j$, $j \in K$, $y \in X$, $a_j \in A_j\left(y_{\widetilde{N}(j)}\right)$, $C_K = \underset{j \in K}{\times} C_j$

$$
\begin{aligned}
&\Pr\left\{\xi_K^{t+1} \in C_K \mid \xi^0 = x^0, \alpha^0 = a^0, \ldots\right.\\
&\left.\ldots, \xi^{t-1} = x^{t-1}, \alpha^{t-1} = a^{t-1}, \xi^t = y, \alpha^t = a\right\}\\
&\overset{(1)}{=} \Pr\left\{\xi_K^{t+1} \in C_K \mid \xi^t = y, \alpha^t = a\right\}\\
&\overset{(2)}{=} \prod_{j \in K} \Pr\left\{\xi_j^{t+1} \in C_j \mid \xi^t = y, \alpha^t = a\right\}\\
&\overset{(3)}{=} \prod_{j \in K} \Pr\left\{\xi_j^{t+1} \in C_j \mid \xi^t_{\widetilde{N}(j)} = y_{\widetilde{N}(j)}, \alpha_j^t = a_j\right\}\\
&\overset{(4)}{=} \prod_{j \in K} \mathsf{Q}_j\left(C_j \mid y_{\widetilde{N}(j)}, a_j\right)\\
&\overset{(5)}{=} \mathsf{Q}_K\left(C_K \mid y, a\right). \qquad (6)
\end{aligned}
$$

Якщо $K = V$, будемо писати $\mathsf{Q}_V(C_V \mid y, a) = \mathsf{Q}(C \mid y, a)$.

Марковське ядро $\mathsf{Q} = \prod_{j \in V} \mathsf{Q}_j$ називатимемо локальним і синхронним.

Застосування

Просторовий опис катастроф. У роботі [3] пропонується застосувати результати роботи [1] до розв'язання проблеми катастрофічних ризиків. Припускається, що можливі стани x_i^t і втрати z_i^t системи в вершині i у момент часу t визначаються випадковим процесом (наприклад, природною катастрофою) ξ^t з можливими станами $y \in Y$. Через $H^i\left(x_i^t, z_i^t \mid x^{t-1}_{\widetilde{N}}, z^{t-1}_{\widetilde{N}(i)}, y; u^{t-1}_{\widetilde{N}(i)}\right)$ позначимо умовні розподіли пари (x_i^t, z_i^t) в момент часу t за умови, що у попередній момент часу стани і втрати в околі $\widetilde{N}(i)$ були $\left(x^{t-1}_{\widetilde{N}}, z^{t-1}_{\widetilde{N}(i)}\right)$, процес ξ^{t-1} був у стані y і були прийняті рішення $u^{t-1}_{\widetilde{N}(i)}$. Такий розподіл визначає динаміку змін в системі згідно з таким співвідношенням:

$$
\begin{aligned}
&p\left(t, x_i^t, z_i^t\right)\\
&\quad = \sum_{y \in Y} H^i\left(x_i^t, z_i^t \mid x^{t-1}_{\widetilde{N}(i)}, z^{t-1}_{\widetilde{N}(i)}, y; u^{t-1}_{\widetilde{N}(i)}\right)\\
&\qquad\qquad \times P\left(\xi^{t-1} = y\right),
\end{aligned}
$$

де $p\left(t, x_i^t, z_i^t\right)$ — розподіл пар (x_i^t, z_i^t) в момент часу t в вершині i. Додаткова фіксація початкового розподілу станів $p\left(0, x_i, z_i\right)$ для всіх $i \in V$ повністю визначає динаміку змін станів системи.

Далі автори роботи [3] пропонують просторове узагальнення моделі керування довгостроковими інвестиціями у безпеку. У цьому разі стани вершин графа трактуються як рівні багатства агентів економічної системи, а ребра — наявність зв'язків між агентами. У випадкові моменти часу в кожній вершині $i \in V$ може відбутися катастрофічна подія або терористична атака (незалежно від подій в інших вершинах), які характеризуються випадковою величиною ξ_i. Припускається, що деякі ресурси u_{ij} віднімаються від x_i і потім інвестуються в заходи безпеки в вершину $j \in N(i)$. Повні інвестиції в заходи безпеки в вершині j становлять $\sum_{i \in \widetilde{N}(j)} u_{ij}$. Інвестиції в заходи безпеки зменшують інтенсивність атак, які у такому разі вже залежать і від інвестицій, тобто $p_i = p_i\left(x_i, u_i\right)$. Крім того, інвестиції в засоби захисту у вузлі i можуть зменшити збитки з коефіцієнтом пропрційності

$$
\xi_i\left(x_i, u_i\right) = \begin{cases} 0 \text{ з імовірністю } 1 - p_i\left(x_i, u_i\right), \\ \zeta_i\left(x_i, u_i\right) \text{ з імовірністю } p_i\left(x_i, u_i\right). \end{cases}
$$

Далі автори припускають, що темпи економічного розвитку ρ_i^t у вузлі i залежать від власних індикаторів зростання $(1 + r_i)$ і збитків $(1 - \xi_i)$, а також факторів зростання/збитків у сусідніх вузлах, тобто $\rho_i^t = \rho_i^t\left(r^t_{\widetilde{N}(i)}, \xi^t_{\widetilde{N}(i)}\right)$. Відповідна динамічна модель економіки набуває вигляду

$$
\begin{aligned}
x_i^{t+1} &= (x_i^t - u_i^t)\rho_i^t\left(r^t_{\widetilde{N}(i)}, \xi^t_{\widetilde{N}(i)}\right),\\
&\qquad x_i^0 = x_i, \quad t = 0, \ldots, T-1,
\end{aligned}
$$

де $r^t_{\widetilde{N}(i)} = \left(r_j^t, j \in \widetilde{N}(i)\right)$ — сукупність випадкових індикаторів зростання у вузлах $\widetilde{N}(i)$; $\xi^t_{\widetilde{N}(i)}$ — сукупність показників збитків у вузлах $\widetilde{N}(i)$ в

період часу t від атаки на вузол i або на сусідні вузли $j \in \widetilde{N}(i)$; $u_i^t = \sum_{j\in\widetilde{N}(i)} u_{ji}^t$ — інвестиції у безпеку вузла i з самого вузла i і з сусідніх вузлів $j \in N(i)$. Залежність добробуту даного вузла i мережі від сусідів може спонукати вузол i зробити інвестиції $u_{ij} > 0$ в сусідні вузли $j \in N(i)$.

Цільовий функціонал для оптимізації витрат на безпеку може мати вигляд

$$\vec{F}(x,u) = \mathbf{E}\sum_{k=0}^{T}\gamma^k \sum_{i\in V}\vec{f}_i\left(x_i^t\right) \longrightarrow \max_{u\in U}$$

або

$$\vec{F}(x,u) = \lim_{T\to\infty}\mathbf{E}\frac{1}{T+1}\sum_{k=0}^{T}\sum_{i\in V}\vec{f}_i\left(x_i^t\right) \longrightarrow \max_{u\in U},$$

де $\vec{f}_i(\cdot,\cdot)$ — векторна функції корисності капіталу у вузлі i; $\gamma \in (0;1]$ — дисконтуючий множник; $\mathbf{E}$ — математичне сподівання, $x = \{x_i, i \in V\}$, $u = \{u_{ij}, i,j \in V\}$, $U = \left\{u_{ij} \geqslant 0 \mid \sum_{j\in\widetilde{N}(i)} u_{ij} \leqslant x_i, i,j \in V\right\}$. Обидві задачі багатокритеріальні, причому компоненти векторної функції $\vec{f}_i(\cdot,\cdot)$ можуть відображати різні аспекти корисності та ризику у вузлі i.

Соцмережі. Не менш цікавим застосуванням теорії керованих марковських полів є моделювання соцмереж. Наприклад, у роботі [4] розглядається така графічна модель соціальної взаємодії.

Розглянемо соціальну мережу з n індивідів або вузлів. Вузол i асоціюється з впливом A_i, результатом Y_i і, можливо, коваріантами. Наприклад, Y може представляти думки або рекламні кампанії; Y може відображати поведінку, а A — заохочувальні втручання, або Y може представляти інфекційне захворювання, а A — вакцинацію. У аналізі рішень Верховного суду США, поданому в роботі [4], Y_i — це бінарна змінна, яка відображає, чи було рішення судді i ліберальним чи консервативним, а A, яке одночасно охоплює всіх суддів, є індикатором сфери розгляду справи. Коли переконання або думки індивідів проходять фазові переходи до впорядкованих станів, наприклад, коли є зовнішній тиск досягти одностайного консенсусу, або коли можна стверджувати, що розподіл поведінки, переконань, думок або інших результатів досягає рівноваги в межах зв'язків мережі, тоді ланцюговий граф може бути правильним підходом для моделювання спільного розподілу результатів у мережі та впливів на ці результати. Наприклад, у даних Верховного суду США результати представляють рішення, ухвалені за умов часових обмежень та під тиском дев'яти суддів досягти одностайного рішення; ці рішення можуть справді перебувати в стані рівноваги.

У статті розглядається застосування керованих марковських полів у контексті каузального аналізу в соціальних мережах та ланцюгових графах. Дослідження обґрунтовує використання ланцюгових графів як інструменту для моделювання взаємодій між індивідуальними одиницями, що піддаються впливу соціальних зв'язків, інтерференції та поширення результатів.

Напрямлені ациклічні графи (DAG) і ланцюгові графи використовують для визначення статистичних і причиново-наслідкових моделей. Статистичні графічні моделі пов'язують спостережуваний розподіл даних $p(V)$ із графом, де вершини пов'язані з випадковими величинами у V. Графічні моделі часто визначаються за допомогою факторизації, де $p(V)$ можна записати як добуток менших факторів із правилом для отримання цих факторів, заданим графом. Причиново-наслідковий висновок із даних спостережень полягає в тому, щоб робити висновки щодо фактичних або потенційних значень випадкових величин на основі розподілу даних спостереження $p(V)$. Параметри причиново-наслідкових зв'язків, що становлять першочерговий інтерес, як правило, є слабовимірними підсумками, отриманими з розподілів, а не самих розподілів. Наприклад, середній причиново-наслідковий ефект визначається як середній контраст $\mathbf{E}\big[Y(a)\big] - \mathbf{E}\big[Y(a')\big]$. Причиново-наслідкові моделі DAG є потужними інструментами для визначення причиновості з використанням даних спостережень і набули широкого застосування в епідеміології, соціальних науках та інших галузях і можуть бути використані для виведення теорії ідентифікації у складних багатовимірних причиново-наслідкових системах.

Керовані марковські поля використовують як спосіб параметризації мережевих даних, що дозволяє зменшити вимоги до обсягу даних і полегшує обчислення й оцінювання моделей. У статті продемонстровано, що такі моделі можуть ефективно описувати процес формування колективних рішень, зокрема на прикладі рішень Верховного суду США.

Авторами обґрунтовано, що традиційні методи каузального аналізу, основані на напрямлених ациклічних графах (DAGs), можуть бути непридатними для аналізу взаємодій у соціальних мережах через їхню складність і вимоги до обсягу даних. Натомість ланцюгові графи забезпечують більш компактне подання залежностей та дають змогу формалізувати рівноважні стани взаємодій у соціальних мережах.

Психологія. Яскравим прикладом застосування графічних моделей у психології є стаття [5], яка фокусується на використанні психометричного мережевого аналізу для дослідження структури та взаємозв'язків у багатовимірних даних, зокрема в психологічних дослідженнях. При цьому слід виділити такі основні моменти:

- для аналізу мережі використовують графічні моделі, де вузли представляють змінні, а ребра — умовні асоціації між ними, мережі дозволяють виявляти патерни асоціацій, які часто відображають латентну структуру даних;
- для оцінки мережевих структур визначається топологія мережі, зокрема щільність зв'язків, центральність вузлів та структура кластерів, мережі аналізуються через контекст тимчасових даних (наприклад, часові ряди) або панельних досліджень;
- щодо застосування результатів можна виділити такі особливості: у дослідженнях особистості мережі дозволяють описати взаємозв'язок між рисами характеру та мотиваційними цілями; у сфері вивчення ставлень моделюються зміни в оцінках важливості ставлень, що допомагає пояснити поляризацію; у дослідженнях психічного здоров'я мережі відображають взаємозв'язки між симптомами, що сприяє кращому розумінню причин та можливих втручань;
- виділяють такі переваги парних марковських випадкових полів: гнучкість у дослідженні умовних залежностей без необхідності жорстких апріорних припущень, мережі забезпечують наочне представлення даних, сприяючи генерації гіпотез;
- разом з тим, є певні обмеження та виклики, серед яких труднощі з інтерпретацією центральності вузлів і залежність від вибору змінних, необхідність додаткових досліджень для вдосконалення методів оброблення даних, наприклад, для пропущених або неоднорідних даних.

Стаття наголошує на важливості мережевого аналізу як потужного інструменту для розуміння складних взаємозв'язків у багатовимірних даних, особливо в психологічній науці.

Висновки

Теорія керованих випадкових полів, попри свою відносну малодослідженість порівняно з теорією стохастичної оптимізації та керованих випадкових процесів, демонструє значний потенціал для практичного застосування. Важливим досягненням у цій галузі є отримання фундаментальних результатів щодо існування та методів пошуку оптимальних стратегій керування випадковими полями для широкого класу функцій ризику. Особливу увагу привертає можливість застосування цієї теорії у випадках, де наявна локальна структура взаємодії. Напрями практичного використання не обмежуються оцінкою ризиків катастроф, моделюванням та аналізом взаємодій у соціальних мережах, а також психометричним мережевим аналізом. Це свідчить про міждисциплінарний характер теорії та її потенційну універсальність у застосуванні до різних предметних галузей. Подальший розвиток теорії потребує розширення наукової та навчальної літератури, а також розроблення методології застосування до конкретних практичних задач. Локальна структура взаємодії виступає ключовою характеристикою, що визначає специфіку застосування теорії, тому важливим є розвиток методів оптимізації для різних класів функцій ризику, що відповідають реальним практичним ситуаціям. Для ефективного впровадження теоретичних результатів необхідно зосередити увагу на розробленні спеціалізованих методів для кожного з визначених напрямів застосування, розвитку математичного апарату для більш точного опису локальних структур взаємодії та створенні відповідних програмних інструментів. Особливу увагу слід приділити розробленню методів, що дозволяють враховувати специфіку локальних взаємодій у конкретних практичних ситуаціях. Таким чином, незважаючи на відносно обмежену представленість у науковій літературі, теорія керованих випадкових полів має значний потенціал для практичного застосування в різних галузях. Подальший розвиток теорії та методології її застосування може суттєво вплинути на ефективність розв'язання важливих практичних задач у різних сферах людської діяльності, від технічних аспектів оцінки ризиків катастроф до соціальних та психологічних аспектів аналізу мереж взаємодії.

Список літератури

1. Chornei R. K., Daduna H., Knopov P. S. Control of Spatially Structured Random Processes and Random Fields with Applications. Springer Science + Business Media, Inc. 2006.
2. Vasilyev N. B. Bernoulli and Markov stationary measures in discrete local interactions. Dobrushin R. L., Kryukov V. I., Toom, A. L., eds. *Locally Interacting Systems and Their Application in Biology, Lecture Notes in Mathematics*. Springer Verlag, 1978. Pp. 99–112.
3. Haivoronskyy O. O., Ermoliev Yu. M., Knopov P. S., Norkin V. I. Mathematical Modeling of Distributed Catastrophic and Terrorist Risks. Cybernetics and Systems Analysis. 2015. Vol. 51 (1). Pp. 85–95.
4. Ogburn E. L., Shpitser I., Lee Y. Causal inference, social networks and chain graphs. *J. R. Stat. Soc. Ser. A Stat. Soc.* 2020. Vol. 183 (4). Pp. 1659–1676.
5. Borsboom D., Deserno M. K., Rhemtulla M., Epskamp S., Fried E. I., McNally R. J., Robinaugh D. J., Perugini M., Dalege J., Costantini G., Isvoranu A.- M., Wysocki A. C., van Borkulo C. D., R. van Bork and L. J. Waldorp. Network analysis of multivariate data in psychological science. *Nature Reviews Methods Primers*. 2021. Vol. 1 (1). Pp. 1–18.

References

1. R. K. Chornei, H. Daduna, and P. S. Knopov, *Control of Spatially Structured Random Processes and Random Fields with Applications* (Springer Science + Business Media, Inc., 2006).
2. N. B. Vasilyev, in: R. L. Dobrushinand, V. I. Kryukov and, A. L. Toom, eds., *Locally Interacting Systems and Their Application in Biology, Lecture Notes in Mathematics* (Springer Verlag, 1978), pp. 99–112.
3. O. O. Haivoronskyy, Yu. M. Ermoliev, P. S. Knopov, and V. I. Norkin, Cybernetics and Systems Analysis. **51** (1), 85–95 (2015).
4. E. L. Ogburn, I. Shpitser, and Y. Lee, J. R. Stat. Soc. Ser. A Stat. Soc. **183** (4), 1659–1676 (2020).
5. D. Borsboom, M. K. Deserno, M. Rhemtulla, S. Epskamp, E. I. Fried, R. J. McNally, D. J. Robinaugh, M. Perugini, J. Dalege, G. Costantini, A.-M. Isvoranu, A. C. Wysocki, C. D. van Borkulo, R. van Bork, and L. J. Waldorp, Nature Reviews Methods Primers. **1** (1), 1–18 (2021).

R. Chornei

ON SOME APPLICATIONS OF CONTROLLED RANDOM FIELDS WITH LOCAL INTERACTION STRUCTURE

This paper explores controlled random fields with a local interaction structure and the fields' potential applications. The primary focus is on optimal control problems for stochastic systems defined on graphs, emphasizing risk assessment, social network modeling, and psychometric network analysis. The study formalizes mathematical approaches that facilitate stochastic optimization and decision-making in complex systems with locally structured interactions.

The theoretical framework is developed within the context of Markov random fields, where interactions are defined on finite graphs. The article introduces a mathematical model that captures local dependencies among interacting elements and derives methods for optimizing their collective behavior. A key result concerns the existence and characterization of optimal control strategies in stochastic environments, demonstrating their applicability to risk management and dynamic decision-making.

The paper also discusses the use of controlled Markov fields in social network modeling. Specifically, it examines how individuals influence each other within structured networks and how equilibrium states emerge under specific interaction rules. This modeling technique proves useful in predicting opinion dynamics, social polarization, and decision-making in hierarchical systems.

A further application is psychometric network analysis, where controlled random fields facilitate the study of cognitive and psychological interactions among individuals. The methodology enables the identification of latent structures within high-dimensional psychological data, improving predictive accuracy in behavioral sciences.

The results contribute to interdisciplinary research at the intersection of mathematics, economics, and social sciences. The findings provide valuable insights into how locally structured systems can be effectively managed and optimized in various applied domains.

Keywords: controlled random fields, local interactions, optimal strategies.

Матеріал надійшов 01.03.2025

УДК 519.21

D. Zubritska, N. Shchestyuk, D. Sluchynskyi

DOI: 10.18523/2617-70807202424-34

FRACTIONAL CALCULUS AND ITS APPLICATION IN FINANCIAL MATHEMATICS

Fractional calculus extends classical calculus by allowing differentiation and integration of non-integer orders, providing valuable tools for analyzing complex systems. In this part of the paper we demonstrate the main methods of fractional calculus, including Euler's, Riemann-Liouville, and Caputo approaches. The behavior of functions such as x^n, $e^{\lambda x}$, and $\sin(x)$ is analyzed for fractional orders, demonstrating how fractional differentiation results in varying patterns of growth and decay.

The second part explores the application of fractal derivatives in financial mathematics. We present the use of the Riemann-Liouville derivative to model stock prices in illiquid markets, where the price of an asset may remain unchanged for some time. For this, subdiffusion processes and a fractal integro-differential equation with the Riemann-Liouville derivative are used. The idea of subdiffusion models is to replace the calendar time t in the risk-free bond motion and classical GBM by some stochastic process H_t, which represents a hitting time, which is interpreted as the first time at which G_t hits the barrier t.

Next, we focus on the pricing of a European option when the underlying asset is illiquid. The option price is found as a solution to a fractal Dupire integro-differential equation, in which the time derivative is replaced by the Dzerbayshan– Caputo (D–K) derivative. The D–K derivative is a generalization of the Caputo approach. The form of the D–K derivative depends on a random process G_t, called the subordinate. We take a standard inverse Gaussian process with parameters (1,1) as the subordinate G_t and formulate the Proposition about the form of the fractal Dupire equation for the chosen subordinate. These approaches provide tools that allow the investor to take into account the illiquidity of the financial markets.

Keywords: fractional calculus, Riemann-Liouville derivative, Euler's approach, Riemann-Liouville approach, Caputo's approach, subdiffusion, Dupire equation, Black-Scholes model, Partial Integro-Differential Equations, Dzerbayshan–Caputo derivatives, subordinator.

Introduction

Differentiation and integration are fundamental concepts in mathematics that have been studied intensively for centuries. In its simplest form, differentiation involves calculating the slope of a function at a given point, while integration involves finding the area under a curve. These concepts are well known and have been thoroughly studied over the years, leading to clear and well-known results that are widely used in a wide variety of fields.

An interesting question is the existence of differentiation and integration for fractional order, the so-called fractional calculus. As explained in [2], the classical derivative restricted by rate of change falls short to describe many phenomena that could not be constructed properly by integer order calculus encompassed by fractional calculus. Due to this fact, fractional derivatives are proposed for capturing the past history as in the classical integration. Hence, both fractional derivative and integral have past memory making them much more advantageous than classical counterparts. The history of fractional calculus can be traced back to the work of Euler and Laplace in the 18th century. Later, other prominent mathematicians such as Caputo, Liouville, and Riemann also made significant contributions to the field. Over the past few decades, this branch of mathematical analysis has gained attention due to its significant potential for applications in various fields including physics, engineering, finance, and biology. The main idea of fractional calculus is to extend the concepts of differentiation and integration to functions with non-integer orders. This allows for a more accurate description of complex phenomena, such as anomalous diffusion [7], viscoelasticity [11], and fractal behaviour [12].

The purpose of this paper is to study approaches to fractional calculus, illustrate them by visualizing the results in the Python programming language and demonstrate how Dzerbayshan– Caputo (D–C) derivative is used for option evaluat-

ing. By achieving this goal, this study aims to fill the gap in the existing literature on this topic and provide a better understanding of the potential of fractional calculus in financial mathematics.

The paper is organized as follows. The second section consists of two subsections. In the first subsection the comparison between classical and fractional calculus interpretations is discussed. Also we review the main approaches to fractional calculus: Euler, Liouville, Riemann, and Caputo. The second subsection focuses on the Riemann–Liouville approach to fractional calculus. This approach builds upon the Riemann method and the Cauchy integral formula, allowing for the generalization of integration to non-integer orders using the Gamma function. The fractional integral is defined, and its important properties, such as the additive property of fractional integrals and the relationship between fractional integration and differentiation, are discussed. The Riemann-Liouville approach has a huge application in financial mathematics and it is used for stock price modeling on illiquid markets. The Caputo's approach modifies the Riemann-Liouville definition to simplify initial condition handling in fractional differential equations, making it highly valuable for real-world modeling. The updated approach is known as Dzerbayshan–Caputo derivative introduced later and is applied for option pricing on illiquid markets. In the last subsection, we examine how the fractional order α influences the behavior of derivatives across the Euler, Caputo, and Riemann-Liouville approaches. The behavior of functions such as x^n, $e^{\lambda x}$, and $\sin(x)$ is analyzed for fractional orders, demonstrating how fractional differentiation results in varying patterns of growth and decay.

The third section is devoted to the applications of fractional calculus in financial mathematics, particularly for describing the dynamics of the illiquid markets. Classical models, like Black-Scholes, assumes that asset prices follow Brownian motion, a process with independent and stationary increments. However, these models often fail to account for the irregularities and memory effects observed in illiquid markets, where asset prices exhibit anomalous behaviors like stationarity or jumps. In this context, fractional calculus and subdifusive models which incorporate hitting times and irregular trading activity provides a natural extension to incorporate such complexities, offering a more accurate representation of the underlying dynamics of financial illiquid assets.

First, we mention the usual model of subdiffusion, which is the celebrated Fractional Fokker-Planck equation (see for example [8]). This equation is based on the Riemann- Liouville fractional derivative and describes the probability density function $w(t)$ of the sub-diffusive stock process. This theory fully detailed in the literature (see for example [7], [6], [8]). The application of the Fractional Fokker-Planck equation to the risk measuring in financial mathematics you can find in [22].

After that we focus on the option pricing problem under subdiffusion. The main idea of subdiffusive is to replace calendar time t by hitting time H_t, which interpreted as the first time at which stochastic process (so called subordinator) G_t hits the barrier t. Initially for the option pricing under subdiffusion was used the method of discounted mathematical expectation of the payoff function under risk-neutral measure (see for example [7], [6], [23], [24]). A new method was proposed recently by the Donaten and Leonenko (see [20]), which uses a fractional Dupire equation with Dzerbayshan–Caputo derivatives for deriving the European call option.

In this study we just apply the idea of this approach for the standard IG process (SIG), which simplifies the equation and recovers the fractional Dupire form under specific conditions. It is noteworthy that this approach was detailed for inverse $\alpha-$ stable and inverted Poisson processes in [20], for inverse inversian Gaussian in [21], for Gamma in [22].

Finally, we formulate the proposition about application of the fractal Dupire PIDE in the case of the SIG subordinator. By incorporating fractional calculus, we have used for SIG a model that captures the non-local and memory-dependent nature of market dynamics, offering a more accurate and flexible tool for pricing financial instruments in such environments.

Interpretations and approaches to fractional calculus

Main approaches to fractional calculus. Fractional calculus is an extension of traditional integral integration and differentiation. Similarly, fractional exponents are an extension of integer exponents.

Integer calculus has clear and well-known physical and geometric interpretations. For example, the geometric value of a first-order derivative at some point x_0 is equal to the tangent of the tangent line to the graph of the function at the point

with the abscissa x_0 and is equal to the angular coefficient of this tangent line.

In the case of differentiation and integration of arbitrary order, there were no clear geometric and physical interpretations for almost 300 years. Eventually, however, interpretations were found. In [10], the geometric interpretation is the so-called 'shadows on the walls', and the physical interpretation is 'shadows of the past'.

Here is an explanation of what exactly these interpretations are. For example, the geometric interpretation of fractional integration is to add a third dimension to the standard pair $\tau, f(\tau)$. If τ is time, then the added dimension can be called a 'deformed' timescale. The physical or mechanical interpretation of fractional calculus is to use two types of time in calculations: cosmic and individual.

Since this paper is devoted more to the mathematical side of the issue, it is worth describing the 'shadows on the walls' in a little more detail. The geometric interpretation of the fractional integral is to display the so-called 'fence' on two walls, as is clear from this sentence, fractional calculus provides a third dimension for analysing a function. Together with the 'fence', whose shape changes according to the change of time t from 0 to b, its shades on the walls also change, representing the right-handed Riemann-Liouville fractional integral and the classical integral with a moving lower bound. [10]

The history of fractional calculus starts from the work of Euler and Laplace in the 18th century. In 1730, Euler proposed a generalization of this formula:

$$\frac{(d^n x^m)}{(dx^n)} = m(m-1)\dots(m-n+1)x^{(m-n)}$$

Using the properties of the Gamma function:

$$\Gamma(m+1) = m(m-1)\dots(m-n+1)\Gamma(m-n+1)$$

he came up with the following formula:

$$\frac{(d^n x^m)}{(dx^n)} = \frac{\Gamma(m+1)}{\Gamma(m-n+1)}x^{(m-n)}$$

This formula is very useful and easy to use for calculating fractional differentials of functions of the form $f(x) = x^a$, where $a \in R$. [9]

In the period from 1832 to 1855, Liouville proposed three important definitions for fractional calculus. In the first definition, using the exponential representation of the function $f(x) = \sum_{n=0}^{\infty} c_n e^{a_n x}$, he generalized $\frac{(d^m e^{a} x)}{(dx^n)} = a^m e^a x$ as:

$$\frac{d^v f(x)}{dx^v} = \sum_{n=0}^{\infty} c_n a_n^v e^{a_n x}$$

Its second definition is a fractional integral [9]:

$$\int^{\mu} \Phi(\mathrm{x})\mathrm{dx}^{\mu} = \frac{1}{(-1)^{\mu}\Gamma(\mu)} \int_0^{\infty} \Phi(x+\alpha)\alpha^{\mu-1}d\alpha$$

$$\int^{\mu} \Phi(\mathrm{x})\mathrm{dx}^{\mu} = \frac{1}{\Gamma(\mu)} \int_0^{\infty} \Phi(x-\alpha)\alpha^{\mu-1}d\alpha$$

By replacing $x+\alpha$ and $x-\alpha$ with τ in the above formulas, the following formulas were obtained:

$$\int^{\mu} \Phi(\mathrm{x})\mathrm{dx}^{\mu} = \frac{1}{(-1)^{\mu}\Gamma(\mu)} \int_x^{\infty} \Phi(\tau)(\tau-x)^{\mu-1}d\tau$$

$$\int^{\mu} \Phi(\mathrm{x})\mathrm{dx}^{\mu} = \frac{1}{\Gamma(\mu)} \int_x^{\infty} \Phi(\tau)(\tau-x)^{\mu-1}d\tau$$

The third definition is a fractional differential:

$$\frac{d^{\mu}F(x)}{dx^{\mu}} = \frac{(-1)^{\mu}}{h^{\mu}}\left(F(x)\frac{\mu}{1}F(x+h) + \right.$$
$$\left. + \frac{\mu(\mu-1)}{1\cdot 2}F(x+2h) - \cdots\right.$$

$$\frac{d^{\mu}F(x)}{dx^{\mu}} = \frac{1^{\mu}}{h^{\mu}}\left(F(x)\frac{\mu}{1}F(x-h) + \right.$$
$$\left. + \frac{\mu(\mu-1)}{1\cdot 2}F(x-2h) - \cdots\right.$$

From 1847 to 1876, Riemann proposed the other definition of the fractional integral:

$$D^{-v}f(x) = \frac{1}{\Gamma(v)}\int_c^x (x-t)^{v-1}f(t)dt + \psi(t)$$

The Riemann-Liouville definition is one of the two most famous in the field of fractional calculus, it is a combination of the previous two definitions: the definition of the derivative of the Cauchy integral formula and the Riemann definition.

$${}_aD_t^{\alpha}f(t) = \frac{1}{\Gamma(n-\alpha)}\left(\frac{d}{dt}\right)^n \int_a^t \frac{f(\tau)d\tau}{(t-\tau)^{\alpha-n+1}}$$

In this formula, n is the so-called 'ceiling' of α, which means that n is the smallest integer greater than the number whose ceiling it is, in our case $n-1 \le \alpha < n$.[9]

Another well-known definition is Caputo's definition, created in 1967, and as mentioned earlier, it is an improvement of the Riemann-Liouville definition for the calculation of fractal equations. [9]

$${}_a^C D_t^{\alpha}f(t) = \frac{1}{\Gamma(\alpha-n)}\int_{\alpha}^t \frac{f^n(\tau)d\tau}{(t-\tau)^{\alpha+1-n}}, \quad (n-1 \le \alpha < n) \qquad (1)$$

The Riemann–Liouville approach. The Riemann–Liouville approach is based on the Riemann approach and the Cauchy integral formula.

By using the Cauchy formula for repeated integration over parameters, we can calculate the antiderivative α of the function order several times, which leads to the following formula:

$$I^{\alpha} f(t) = \frac{1}{(\alpha-1)!} \int_0^t (t-\tau)^{\alpha-1} f(\tau)\, d\tau$$

As mentioned in another section, the generalisation of the factorial is the so-called Gamma function. So, to improve the already obtained formula, we will replace this factorial with the Gamma function, generalising the result.

$$I^{\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t-\tau)^{\alpha-1} f(\tau)\, d\tau, \alpha > 0$$

This formula is a working formula for fractional integration. It is called the Riemann-Liouville left-handed integral. This integral is considered one of the easiest formulas to understand in the world of fractional calculus. The main note is that α can be a complex number due to the limitations of the Gamma function, but always with a real part greater than zero.

This integral has the following important dependencies:

$$I^{\alpha}\left(I^{\beta} f\right) = I^{\alpha+\beta} f \frac{d}{dx} I^{\alpha+1} f = I^{\alpha} f$$

Unfortunately, we cannot simply say that a differential of order α will be equal to an integral of order $-\alpha$. Due to the presence of the Gamma function in the Riemann-Liouville left-handed integral formula, the use of negative order is not possible, and hence it cannot be used to define a fractional order differential.

To start converting an integral to a differential, you should start with the fact that after differentiating n times, the integration will be equal to the original function itself.

$$\frac{d^n}{dt^n}\left(I^n f(t)\right) = f(t)$$

This means that the derivative is the left-hand side of the integral. However, the integral is not the left-hand side of the derivative because the integral adds an arbitrary constant. That is, in general, the inverse of the previous property is not true. Under this condition, we would still like to be able to define differentiation through operations that are understandable and possible. Such an operation, which has the desired properties, would be:

$$D^{\alpha} f = \frac{d^{\lceil\alpha\rceil}}{dt^{\lceil\alpha\rceil}}\left(I^{\lceil\alpha\rceil-\alpha} f\right)$$

Here, $\lceil\alpha\rceil$ is the 'ceiling' of α, the result of rounding the number to the next smallest integer greater than the given number. Let's write this record in more detail:

$${}_a D_t^{\alpha} f(t) = \frac{1}{\Gamma(n-\alpha)} \left(\frac{d}{dt}\right)^n \int_a^t \frac{f(\tau)\, d\tau}{(t-\tau)^{\alpha-n+1}}, \tag{2}$$

where n is the ceiling of α. This is the left-handed Riemann-Liouville fractional derivative. Most fractional calculations are long and complicated, if not completely intractable, if performed manually without the help of a computer.

Illustration of fractional calculus approaches to some basic functions. In this subsection, we will illustrate and visualize the Euler, Riemann-Liouville, and Caputo approaches to fractional calculus for some functions.

a) Euler's approach.

$$\frac{d^n x^m}{dx^n} = \frac{\Gamma(m+1)}{\Gamma(m-n+1)} x^{m-n} \tag{3}$$

The simplest example is the following function:

$$f(x) = 1$$

for which:

$$\frac{d^{\alpha} 1}{dx^{\alpha}} = \frac{x^{-\alpha}}{\Gamma(1-\alpha)}$$

In this case, we substitute $m = 0, n = \alpha$, where *alpha* is the order of differentiation, into the formula (3). Using Python, we visualize the graphs of the differentials of the function $f(x) = 1$ for the following orders: $\frac{1}{2}, \frac{3}{2}, -\frac{1}{2}, -\frac{3}{2}$.

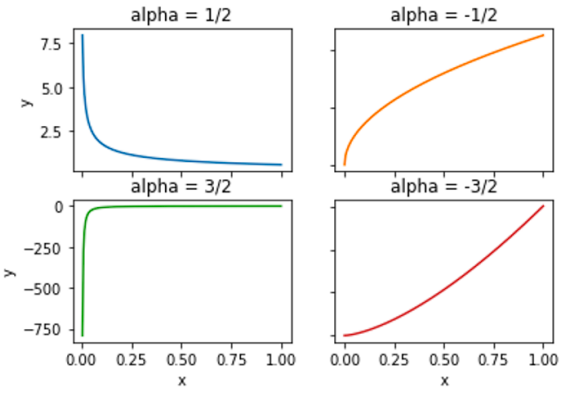


Figure 1. Graphical representation of the fractional differentials of the function $f(x) = 1$ using the formula (3) in Python.

In order to obtain Figure 1, the matplotlib library was used to calculate the result of the Gamma function, using the gamma() method, which returns the value of the function depending on the input x. Another example is solved below for Euler's formula, in this case the function has the following form:

$$f(x) = x$$

Let's repeat the steps described above. In formula (2), we will make the following substitutions: $m = = 1$, $n = \alpha$, where α is again the order of differentiation. After performing these steps, we will get the following function:

$$\frac{d^\alpha x}{dx^\alpha} = \frac{x^{1-\alpha}}{\Gamma(2-\alpha)}$$

Again using Python and its matplotlib library and the gamma() method, we will calculate and display the graphs of the differentials of the following function of orders: $\frac{1}{2}, \frac{3}{2}, -\frac{1}{2}, -\frac{3}{2}$.

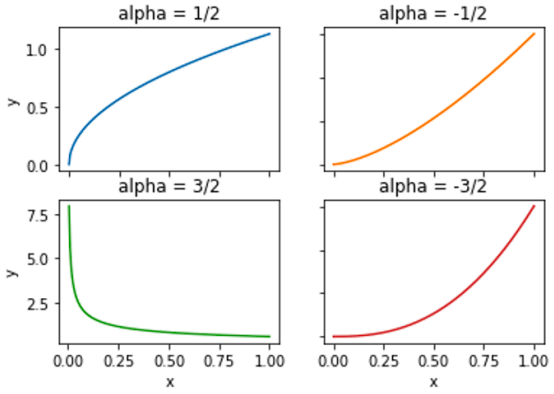


Figure 2. Graphical representation of the fractional differentials of the function $f(x) = x$ using formula (2) in Python.

As you can see from the previous examples, the Euler approach is very convenient for calculating fractional differentials of functions of a type:

$f(x) = x^n$, $n \in Q$. Analyzing the graphs of the derivative functions shown in Figures 1 and 2, we can draw the following conclusion: there is no single law by which these functions are constructed. For example, for $\alpha = 3/2 f(x) = 1$ will be monotonic and strictly increasing, and for $f(x) = = x$ the fractional differential will give us a monotonic strictly decreasing function. Similarly, for $\alpha = 1/2$, the function is strictly decreasing for $f(x) = 1$ and strictly increasing for $f(x) = x$. While for the other two alphas, no such dynamics is observed.

b) Caputo's approach

At first glance, this approach (see (1)) seems overly complicated and requires too many calculations. However, according to Theorrm 5 of Maria Ishtev [5], the differential of an exponential function is of the form:

$$f(x) = e^{\lambda x}$$

and after a number of transformations, it looks like:

$$\frac{d^\alpha e^{\lambda x}}{dx^\alpha} = \sum_{k=0}^{\infty} \frac{\lambda^{k+n} x^{k+n-\alpha}}{\Gamma(k+1+n-\alpha)} = \lambda^n x^{n-\alpha} E_{1,n-\alpha+1}, \quad (4)$$

where $\lambda \in C, n-1 < \alpha < n, \alpha \in R, n \in N$. The proof of this theorem is based on the generalised Mittag-Lefler function for two parameters:

$$E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + \beta)},$$
$$\alpha, \beta > 0, \ \alpha, \beta \in R, \ z \in C$$

and the facts from [5]:

$$D_*^\alpha f(t) = D^\alpha f(t) - \sum_{k=0}^{n-1} \frac{t^{k-\alpha}}{\Gamma(k+1-\alpha)} f^{(k)}(0),$$
$$t > 0, \alpha \in R, \ n-1 < \alpha < n$$

and

$$D^\alpha e^{\lambda t} = t^{-\alpha} E_{1,1-\alpha}(\lambda t).$$

With this formula, we can already write solutions for several examples. Let's start with the function:

$$f(x) = e^x$$

Let's use the formula (4) and get it:

$$\frac{d^\alpha e^x}{dx^\alpha} = \sum_{k=0}^{\infty} \frac{x^{k+n-\alpha}}{\Gamma(k+n+1-\alpha)}$$

Using WolframAlpha, we visualize graphs of order differentials: $\frac{1}{2}, -\frac{1}{2}, \frac{3}{2}, -\frac{3}{2}$.

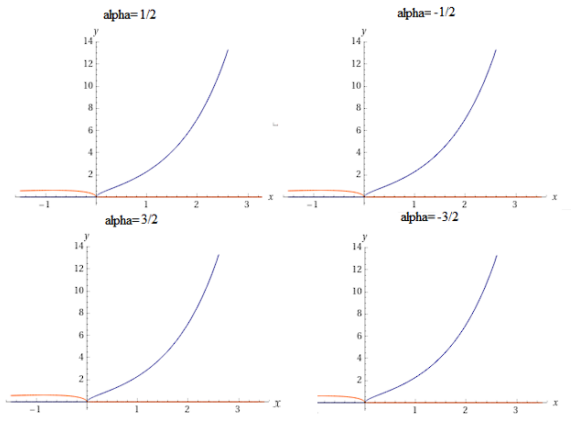


Figure 3. Graphical representation of the fractional differentials of the function $f(x) = e^x$ using the formula (4), using WolframAlpha.

For the next example, let's look at a function:

$$f(x) = e^{2x}$$

Using the formula (4), we get the following differential function:

$$\frac{d^\alpha e^{2x}}{dx^\alpha} = \sum_{k=0}^{\infty} \frac{2^{k+n} x^{k+n-\alpha}}{\Gamma(k+n+1-\alpha)}$$

Using WolframAlpha, we visualise graphs of order differentials: $\frac{1}{2}, -\frac{1}{2}, \frac{3}{2}, -\frac{3}{2}$.

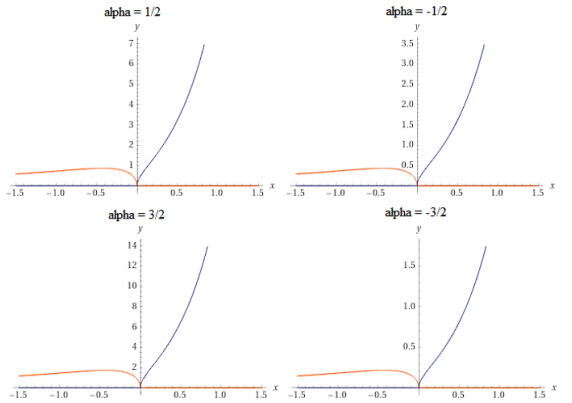


Figure 4. Graphical representation of the fractional differentials of the function $f(x) = e^{2x}$ using the formula (4) in WolframAlpha.

Analyzing the graphs of the derivative functions shown in Figures 3 and 4, we can draw the following conclusion: these functions have a clear pattern. It can be noted that in both figures, the graphs correspond to the behaviour of the integral differential for functions of the form $f(x) = e^{nx}$. Thus, we see that the change in α changes the y-value of the point of intersection of the graphs with the ordinate axis. The growth dynamics of the graphs also has a single pattern that corresponds to the whole number.

c) The Riemann-Liouville approach.

The application of the Riemann-Liouville formula (2) requires numerical methods of calculation, which is a separate complex task. It is also important to note that Python library for it has very large limitations. This library contains methods for calculating two approaches: Riemann-Liouville and Grunwald-Letnikov.

Main Function	Usage
GLpoint	Computes the GL differintegral at a point
GL	Computes the GL differintegral over an entire array of function values using the Fast Fourier Transform
GLI	Computes the improved GL differintegral over an entire array of function values
RLpoint	Computes the RL differintegral at a point
RL	Computes the RL differintegral over an entire array of function values using matrix methods

Figure 5. A set of functions and their functionality of the differint library.

Using this library, let's give an example for a trigonometric function:

$$f(x) = \sin(x)$$

Let's use the RL() function to calculate the order differentials: $\frac{1}{2}, -\frac{1}{2}, -\frac{3}{2}$. And visualise the results using the mathplotlib library:

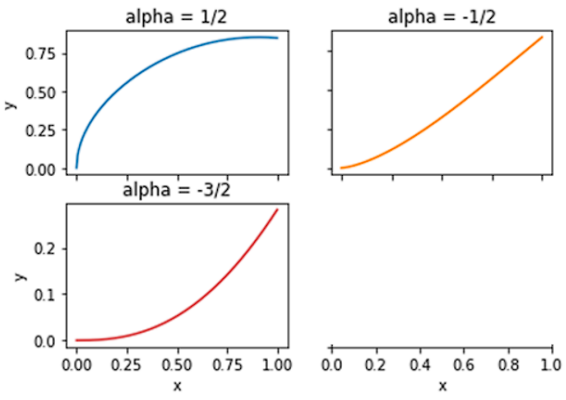


Figure 6. Graphical representation of the fractional differentials of the function $f(x) = \sin(x)$ using the (2) approach, using the Python library differint.

Analyzing the graphs in Figure 6, we can only note the monotonicity of each of the functions shown on it. So for all three values of α used, the functions are increasing. However, we cannot observe any single behaviour that would depend on the order of differentiation and would correspond to the behaviour of the function $f(x) = \sin(x)$ in the integer domain.

Fractional calculus in financial mathematics: Option pricing for subdiffusion model

In traditional financial markets, the Black-Scholes model is widely used for option pricing(see [4], [3]). However, in illiquid markets where trading delays and irregularities occur, classical diffusion models often fall short. Subdiffusive models[1], which incorporate waiting times and irregular trading activity, offer a more accurate way to represent such markets.

The idea of subdiffusion models is to replace the calendar time t in the risk-free bond motion and classical GBM by some stochastic process H_t, which represents a hitting time, defined as:

$$H_t = \inf\{\tau > 0 : G_\tau \geq t\}. \tag{5}$$

and interpreted as the first time at which G_t hits the barrier t. H_t is positive, non-decreasing and has all the properties to be used as a stochastic clock. By construction, the inverted process may be constant. Therefore, any process subordinated by H_t exhibits motionless periods.

The definition (5) of hitting time is based on the use of some other random process called a subordinator G_t. The subordinator G_t is generally a non-decreasing stochastic process.

The usual model of subdiffusion is the celebrated Fractional Fokker-Planck equation (see for example [8]). This equation describes the probability density function $w(t)$ of the sub-diffusive stock process:

$$\frac{\partial w}{\partial t} = {}_aD_t^\alpha \left[-\mu \frac{\partial}{\partial x} + \frac{\sigma^2}{2}\frac{\partial^2}{\partial x^2}\right] w(x,t), \tag{6}$$

where ${}_aD_t^\alpha f(t)$ is is the left-handed Riemann-Liouville fractional derivative (see sections above and formula (2)). This application of the fractional derivatives to the financial mathematics is a quite important, but the more detailed consideration of this problem is outside the framework of this paper.

This study focuses on the option pricing problem under subdiffusion.

In the classical diffusion model, the fair price of a European call option on an asset with price S_t at time t is provided by the Black-Scholes formula. The alternative way to compute the fair price using Dupire equation [15].

For the subdiffusive we derive the European call price using a fractional Dupire equation with Dzerbayshan–Caputo derivatives[13].

For this aim we start with classical Dupire equation and consider the case when $\sigma(S_t, t) = \sigma$ is the constant. Then the Dupire equation has a form

$$\begin{aligned}\frac{\partial C(T,K)}{\partial T} + r(T)C = \mu(T)C - \\ -\mu(T)K\frac{\partial C}{\partial K} + \frac{1}{2}\sigma^2K^2\frac{\partial^2 C}{\partial K^2},\end{aligned} \tag{7}$$

where $C(T,K)$ is the option price at time T with strike price K, $r(T)$ is the risk-free rate, $\mu(T) = = r(T) + q(T)$ is the drift, $q(T) = 0$ is the continuous dividend rate and σ is the volatility.

After that we input variable $k = \ln K$. The derivatives with respect to K are then transformed as follows:

$$\frac{\partial}{\partial K} = \frac{1}{K}\frac{\partial}{\partial k}, \quad \frac{\partial^2}{\partial K^2} = \frac{1}{K^2}\frac{\partial^2}{\partial k^2}$$

Substituting these expressions into the original equation, we obtain the Dupire equation in terms of $k = \ln K$:

$$\frac{\partial C(T,k)}{\partial T} = -r\frac{\partial C(T,k)}{\partial k} + \frac{\sigma^2}{2}\frac{\partial^2 C(T,k)}{\partial k^2}. \tag{8}$$

For the option pricing in subdiffusion model we just replace the derivative for the time in the Dupire equation by a Dzerbayshan–Caputo (D–C) derivative (see [20], [13]). So, the fractional Dupire PIDE has a form

$${}^\Psi DC_H(T,k) = -r\frac{\partial}{\partial k}C_H(T,k) + \frac{\sigma^2}{2}\frac{\partial^2}{\partial k^2}C_H(T,k),$$

where ${}^\Psi Du(t)$ is the convolution-type derivative, called the Dzerbayshan–Caputo (D–C) derivative.

The Dzerbashyan–Caputo derivative generalizes the classical Caputo[14] derivative by incorporating a convolution kernel. This adaptation provides greater flexibility in modeling market behavior influenced by memory effects and irregular temporal dynamics. Specifically, it allows the model to accurately capture the heavy-tailed waiting time distributions and subdiffusive characteristics often observed in illiquid markets. By combining fractional calculus with the dynamics of Lévy subordinators, this approach bridges the gap between

theoretical models and observed market anomalies. We focus on this derivative to better account for the stochastic time changes driven by inverse subordinators, making it particularly well-suited for subdiffusive option pricing.

The D–C derivative for a function $u(t)$ is given by:

$$^{\Psi}Du(t) = b\frac{d}{dt}u(t) + \int_0^t \frac{\partial}{\partial t}u(t-s)\nu(s)ds. \quad (9)$$

Here, the function Ψ represents the Lévy exponent associated with the subordinator G_t.

In this study, we use the Standard Inverse Gaussian (SIG) process as the subordinator G_t. Inverse Gaussian (IG) subordinator G_t is a non-decreasing Lévy process, where the increments $G_{t+s} - G_s$ follow the inverse Gaussian distribution $\varrho(\delta t, \gamma)$ with probabilities density function (PDF):

$$g(x,t) = \frac{\delta t}{\sqrt{2\pi x^3}} e^{\delta\gamma t - (\delta^2 t^2/x + \gamma^2 x)/2}, \quad x > 0;$$

and with Lévy measure

$$\tilde{\nu}(dx) = \frac{\delta}{\sqrt{2\pi x^3}} e^{\left(-\frac{\gamma^2 x}{2}\right)} dx, \quad x > 0, t > 0. \quad (10)$$

For $\gamma = \delta = 1$ we have the standard IG distribution in the form

$$f(x,t) = \frac{t}{\sqrt{2\pi x^3}} e^{\left(-\frac{(x-t)^2}{2x}\right)}, \quad x > 0, t > 0.$$

For a given subordinator G_t, its inverse, denoted as H_t, is defined by the hitting time H_t (5). The density function $h(x,t)$ of H_t has an integral representation [16] and for standard IG distribution has a form:

$$h(x,t) = \frac{1}{\pi} e^{x-\frac{1}{2}} \int_0^\infty \frac{e^{-ty}}{y+\frac{1}{2}} (\sin(x\sqrt{2y}) + \\ + \sqrt{2y}\cos(x\sqrt{2y}))dy.$$

The moments of H_t can be numerically evaluated using $h(x,t)$, and explicit formulas for the first and second moments were obtained via Laplace transforms. Asymptotic behavior shows that for large t [17]:

$$E(H_t) \sim \begin{cases} \left(\frac{\gamma}{\delta}\right) t, & \gamma > 0 \\ \left(\frac{1}{\delta}\sqrt{\frac{2t}{\pi}}\right) t, & \gamma = 0, \end{cases}$$

$$Var(H_t) \sim \left(\frac{\gamma}{\delta}\right)^2 t^2.$$

For the standard case ($\delta = 1, \gamma = 1$), we have $E(H_t) \sim t$ and $Var(H_t) \sim t^2$ and this fact explaines why we choose these parameters.

Thus we focus on standard inverse Gausian subordinator(see [18]) G_t Its Lévy-Khintchine representation can be written as:

$$\Psi(x) = \int_0^{+\infty} (1 - e^{-sz})\tilde{\nu}(dz),$$

where $\tilde{\nu}$ is the Lévy measure.

The Lévy measure for standard IG subordinator will be:

$$\tilde{\nu}(dz) = \frac{1}{\sqrt{2\pi z^3}} e^{-\frac{z}{2}} dz, \quad z > 0, , t > 0. \quad (11)$$

Thus, the integral kernel $\nu(s)$ in (11) is the integral of $\tilde{\nu}$ over (s, ∞):

$$\nu(s) = \int_s^{+\infty} \frac{1}{\sqrt{2\pi z^3}} e^{-\frac{z}{2}} dz = \\ = \frac{2e^{-\frac{s}{2}}}{\sqrt{2\pi s}} - \operatorname{erfc}\left(\frac{\sqrt{s}}{\sqrt{2}}\right) = \\ = \frac{2e^{-\frac{s}{2}}}{\sqrt{2\pi s}} + \operatorname{erf}\left(\frac{\sqrt{s}}{\sqrt{2}}\right) - 1$$

Here, $\operatorname{erf}(x)$ denotes the error function, which is related to the standard normal cumulative distribution function $\Phi(x)$:

$$\nu(s) = 2\Phi(s) - 2 + \frac{2e^{-\frac{s}{2}}}{\sqrt{2\pi s}}$$

Then we can represent the D-C derivative as:

$$^{\Psi}Du(t) = 2\int_0^t \frac{\partial}{\partial t}u(t-s)\left(\Phi(s) - 1 + \frac{e^{-\frac{s}{2}}}{\sqrt{2\pi s}}\right) ds.$$

Now, substituting this to (8), we obtain:

$$\int_0^T \frac{\partial}{\partial T} C_H(T-s,k)\left(\Phi\left(\sqrt{s}\right) - 1 + \frac{e^{-\frac{s}{2}}}{\sqrt{2\pi s}}\right) ds = \\ = -\frac{r}{2}\frac{\partial}{\partial k}C_H(T,k) + \frac{\sigma^2}{4}\frac{\partial^2}{\partial k^2}C_H(T,k), \quad (12)$$

Thus we can state the following proposition.

Proposition 1. *If the subordinator G_t for the hitting time (5) is the Standard Inverse Gaussian (SIG) process, the fair price $C_H(T,k)$ of the European option with time to maturity T and strike price K is the solution of the PIDE (12), where:*

- *$\Phi(s)$ is the standard normal cumulative distribution function;*
- *r is the risk-free rate;*
- *σ is the asset volatility,*

- $k = \log K$.

It is worth noting, that the Dzherbashyan-Caputo fractional derivative plays a crucial role in this model cause it incorporates the nonlinear dynamics of the market, particularly the delays modeled by the subordinator. The convolution kernel of this derivative includes the function $\Phi(s)$, which captures heavy tails and the slow decay of the waiting time distribution. This enables the model to accurately reflect the behavior of illiquid markets and pricing anomalies.

Remark 1. To solve the PIDE numerically, the time T and space k variables are discretized into grids with steps Δt and Δk (see [20]). The integral term is approximated using the trapezoidal rule or quadrature, while the derivatives $\frac{\partial}{\partial k}$ and $\frac{\partial^2}{\partial k^2}$ are computed with finite difference methods. An implicit time-stepping scheme is used for stability, with initial conditions $C_H(0,k) = (e^k - K)^+$ + and asymptotic boundary conditions applied at $k \to \pm\infty$. The equation is transformed into a system of algebraic equations and solved iteratively using numerical tools like Python or MATLAB, ensuring accuracy and stability of the solution.

Another calculation algorithm was presented by Omid Nikan et al [19].

Conclusion

Fractional calculus is a branch of mathematics that extends classical calculus to allow differentiation and integration of non-integer orders. This study looks at the main methods of fractional calculus: Euler's, Riemann-Liouville, and Caputo approaches.

The study analyzes the functions x^n, $e^{\lambda x}$, and $\sin(x)$ for fractional orders like $1/2$, $-1/2$, $3/2$, and $-3/2$. Euler's method was used to find analytical solutions for the functions $f(x) = 1$ and $f(x) = = x$. The Riemann-Liouville method was applied to the function $f(x) = \sin(x)$ using the Python `differint` library. The graphs of the differentials showed that the behavior of the functions changes depending on the specific case, with no general pattern across all cases. The graphs demonstrated consistent trends of either growth or decay, similar to what is observed in integer-order differentiation, where different orders of differentiation lead to different behaviors.

The Caputo method was used on the functions $f(x) = e^x$ and $f(x) = e^{2x}$, with approximation methods applied. Unlike the previous cases, the fractional differentials for these functions followed a consistent pattern, with the graphs behaving similarly but differing only in the intersection point with the y-axis, depending on the order of differentiation.

In the Riemann-Liouville approach, the study used the `RL()` function from the `differint` library to calculate the fractional derivatives of the function $f(x) = \sin(x)$ for orders $1/2$, $-1/2$, and $-3/2$. The graphs of the resulting fractional derivatives showed that all the functions exhibited monotonic growth. However, there was no clear pattern that could be attributed to the order of differentiation in the same way as integer-order derivatives.

In financial modeling, the Riemann-Liouville fractional derivative has been used to describe subdiffusive processes, improving the Black-Scholes model by accounting for market features that traditional models don't capture, such as irregular trading and delays. By adding subdiffusion with a fractional Partial Integro-Differential Equation (PIDE) using the Dzerbayshan–Caputo derivative, the model better reflects how asset prices move by considering memory effects and subdiffusive behavior.

The study finds that subdiffusive models are more accurate and sensitive, especially in capturing market behaviors like periods of price stability. However, these models are computationally heavy and not suitable for real-time use by most investors. While fractional calculus is a powerful tool for modeling complex systems like fluids and fractals, it requires a lot of computing power and time. While these models are useful for specific tasks, they are not necessary for general financial applications. Therefore, developing simpler approximation methods remains an important area of research. This study shows that fractional calculus can improve financial modeling.

References

1. S. Alshammari, N. Iqbal and M. Yar, Journal of Function Spaces. **2022**, 1–12.
2. K. Assaleh and W. M. Ahmad, in: 2007 9th International Symposium on Signal Processing and Its Applications (ISSPA), Sharjah, United Arab Emirates, February 12–15, 2007, pp. 1–4.
3. Black-Scholes formula, Encyclopedia of Finance, URL: http://encyclopediaofmath.org/index.php?title=Black-Scholes$_formulaoldid$ = 50024
4. A. Hayes, What is the BlackScholes model? Investopedia, URL: https://www.investopedia.com/terms/b/blackscholes.asp
5. M. Ishteva, R. Scherer, L. Boyadjiev, Mathematical Sciences Research Journal. **9** (6), 5–7 (2025).
6. M. Magdziarz, Journal of Statistical Physics.**136** (3), 553–564 (2009).
7. M. Magdziarz, S. Orzel and A. Weron, Journal of Statistical Physics. **145** (1), 187–203 (2011).
8. R. Metzler and J. Klafter, Physics Reports. **339** (1), 1–77 (2000).
9. D. Mehdi and B. Majid, Applied Mathematical Sciences. **4** (21), 1–4 (2010).
10. I. Podlubny, Fractional Calculus and Applied Analysis. **5** (4), 367–386 (2002).
11. E. Soczkiewicz, Molecular and Quantum Acoustics. **23**, 397–404 (2002).
12. . F. B. Tatom, Fractals. **3** (1), 217–229 (1995).
13. M. M. Dzherbashian and A. B. Nersesian, Fractional Calculus and Applied Analysis. **23**, 1810–1836 (2020).
14. M. Caputo, Geophysical Journal of the Royal Astronomical Society. **13**, 529– 539 (1967).
15. M. Haugh, in: IEOR E4707: Financial Engineering: Continuous-Time Models (2013).
16. P. Vellaisamy and A. Kumar, ArXiv. 1105.1468 (2011).
17. B. Jørgensen, Lecture Notes in Statistics. **9**, 188 (2012).
18. A. Kumar and P. Vellaisamy, Statistics and Probability Letters. **103**, 134–141 (2015).
19. O. Nikan, J. Rashidinia, and H. Jafari, Alexandria Engineering Journal. **112**, 235–245 (2024).
20. H. Donatien and N. N. Leonenko, Journal of Computational and Applied Mathematics. **381**, Article 112995 (2021).
21. N. Shchestyuk and S. Tyshchenko, Modern Stochastics: Theory and Applications. **12** (2), 136–152 (2025).
22. N. Shchestyuk, S. Drin, and S. Tyshchenko, in: Mathematical and Statistical Methods for Actuarial Sciences and Finance: Conference proceedings: International Conference of the Mathematical and Statistical Methods for Actuarial Sciences and Finance (MAF 2024) (Le Havre, France, Springer, Cham, 2024), pp. 286–291.
23. V. Pauk, O. Petrenko, and N. Shchestyuk, Mohyla Mathematical Journal. **5**, 38–45 (2022).
24. N. Shchestyuk and S. Tyshchenko, Bulletin of the Taras Shevchenko National University of Kyiv. Physics and Mathematics. **2**, 85–92 (2021).

Зубріцька Д. Є., Щестюк Н. Ю., Случинський Д. Ю.

ФРАКЦІЙНЕ ЧИСЛЕННЯ ТА ЙОГО ЗАСТОСУВАННЯ У ФІНАНСОВІЙ МАТЕМАТИЦІ

Фракційне числення розширює класичне числення, дозволяючи диференціювання та інтегрування довільного (нецілого) порядку, що надає цінні інструменти для аналізу складних систем. У цій частині роботи ми демонструємо основні методи фракційного числення, зокрема підходи Ейлера, Рімана-Ліувілля та Капуто. Аналізується поведінка функцій, таких як x^n, $e^{\lambda x}$ і $\sin(x)$, для фракційних порядків, що демонструє, як фракційне диференціювання призводить до різних закономірностей зростання та згасання.

У другій частині досліджується застосування фрактальних похідних у фінансовій математиці. Ми представляємо використання похідної Рімана-Ліувілля для моделювання динаміки цін акцій на неліквідних ринках, де вартість активу може залишатися незмінною протягом деякого часу. Для цього використовуються субдифузійні процеси та фрактальне інтегро-диференціальне рівняння з похідною Рімана-Ліувілля.

Ідея субдифузійних моделей полягає в тому, щоб замінити календарний час t у русі безризикової облігації та класичному геометричному броунівському русі (GBM) деяким стохастичним процесом H_t, який є моментом досягнення певного рівня. Його можна інтерпретувати як перший момент, коли процес G_t досягає бар'єру t.

Далі ми зосереджуємося на оцінюванні ціни європейського опціону у випадку, коли базовий актив є неліквідним. Ціна опціону визначається як розв'язок фрактального інтегро-диференціального рівняння Дюпіра, в якому похідна за часом замінюється похідною Джербашяна-Капуто (D–K). Похідна D–K є узагальненням підходу Капуто. Форма похідної D–K залежить від випадкового процесу G_t, який називають субординатою. Ми розглядаємо стандартний обернений гаусівський процес із параметрами (1,1) як субординату G_t і формулюємо твердження про вигляд фрактального рівняння Дюпіра для вибраної субординати.

Завдяки запропонованим підходам інвестор отримує інструменти, що дозволяють йому враховувати неліквідність фінансових ринків.

Ключові слова: фракційне числення, похідна Рімана-Ліувілля, підхід Ейлера, підхід Рімана-Ліувілля, підхід Капуто, субдифузія, рівняння Дюпіра, модель Блека-Шоулза, часткові інтегро-диференціальні рівняння, похідні Джербашяна-Капуто, субордината.

Матеріал надійшов 17.11.2024

УДК 519.85:004.42

M. Poliakov, N. Shvai

DOI: 10.18523/2617-70807202435-43

GAN-GENERATED STROKES EXTENSION FOR PAINT TRANSFORMER

Neural painting produces a sequence of strokes for a given image and artistically recreates it using neural networks. In this paper, we explore a novel Transformer-based framework named the Paint Transformer to predict the parameters of a stroke set with a feed-forward neural network. The Paint Transformer achieves better painting results than previous methods with more inexpensive training and inference costs. The paper proposes a novel extension to the Paint Transformer that adds more complex GAN-generated strokes to achieve a more artistically abstract painting style than the original method. This research was originally published as a Master's thesis [1].

Keywords: neural painting, transformer, GAN.

Introduction

Painting has been an excellent way for humans to record what they perceive or even imagine the universe around them and has long been known to demand proficiency. Computer-aided art design essentially fills this gap and enables us to make our creative pieces, particularly with the appearance of AI. However, most current generative AI painting methods are still centered on teaching computers how to "paint" at the pixel level to achieve or mimic some painting style, for example, purely GAN-based approaches [2] and style transfer [3]. Humans create artworks through a stroke-by-stroke process, using brushes from coarse to fine. It is of great potential to make machines imitate such a stroke-by-stroke process to develop more genuine and human-like paintings. Thus, as an emerging research topic, stroke-based neural painting is analyzed to generate a series of strokes to mimic how human painters create artistic works. Generating stroke sequences for the painting process is challenging even for skilled human painters, especially when the targets have complicated compositions and rich textures. Some previous work tackles this problem by a sequential process of generating strokes one by one, such as greedy search step-by-step [4], recurrent neural networks [5], and reinforcement learning [6]. Using an iterative optimization process, techniques [7] tackle this problem via stroke parameter searching.

Although these methods generate attractive painting results, considerable room for advancement in both efficiency and effectiveness still exists. Sequence-based methods such as RL are relatively quick in inference but suffer from lengthy training time and unstable agents. Meanwhile, optimization-based approaches do not need training, but their optimization process is highly time-consuming. Distinct from earlier techniques, in this paper, we explore the painting process as a set prediction task and a novel Transformer-based framework, named Paint Transformer, proposed by [8], to predict the parameters of a stroke set with a feed-forward neural network. Paint Transformer achieves better painting results than previous methods with more inexpensive training and inference costs.

Despite excellent Paint Transformer results, there is room for further improvement. The paper proposes a novel extension to Paint Transformer that adds more complex GAN-generated strokes to achieve a more artistically abstract painting style than the original method.

Related work

Paint Transformer. Paint Transformer is a progressive stroke prediction procedure. The model predicts multiple strokes in parallel at each step to minimize the difference between the current canvas and our target image. Paint Transformer has two modules: *Stroke Renderer* and *Stroke Predictor*. Provided a target picture, I_t, and an intermediate canvas picture, I_c, *Stroke Predictor* yields a set of parameters to choose the current stroke set S_r. Then, *Stroke Renderer* renders the stroke picture for each stroke in S_r and plots them onto the canvas I_c, creating the resultant image I_r [8]. Or simply:

$$I_r = PaintTransformer(I_c, I_t) \qquad (1)$$

Only *Stroke Predictor* is trainable in Paint Transformer, while *Stroke Renderer* is a parameter-free and differentiable module. *Stroke Predictor* has a self-training pipeline that uses randomly sampled strokes. During training, in each iteration, a

list of foreground stroke parameters S_f and a list of background stroke parameters S_b are randomly sampled. *Stroke Renderer* then generates a canvas picture I_c by taking as input S_b and producing a target picture I_t by overlaying S_f onto I_c. In the end, *Stroke Predictor* taking I_c and I_t as input can predict a stroke list S_r, after which *Stroke Renderer* can produce a predicted image I_r taking S_r and I_c as intake. Therefore, *Stroke Predictor* optimization is conducted on both stroke and pixel levels and the training goal can be stated as:

$$\mathcal{L} = \mathcal{L}_{stroke}(S_r, S_f) + \mathcal{L}_{pixel}(I_r, I_t) \qquad (2)$$

where $\mathcal{L}_{pixel}$ and $\mathcal{L}_{stroke}$ are pixel loss and stroke loss, respectively. Strokes are randomly sampled so that unlimited data for training can be generated. Thus, Paint Transformer does not need any prepared-in-advance training dataset.

In Paint Transformer, a stroke is a simple 1-channel brush image, called primitive brush, which can be transformed by shape parameters and color parameters. The shape parameters of a stroke include height h, width w; a center point coordinates x, y, and rotation angle θ. Color parameters contain RGB values represented as r, g, and b. Thus, a stroke s can be denoted as $\{x, y, h, w, \theta, r, g, b\}$. Let I_{in} and I_{out} be an input and output canvases, and $S = \{s_i\}_{i=1}^n$ is a list of n strokes. Given a primitive brush image I_b and a stroke s_i, *Stroke Renderer* can change its color, and affine transforms its shape and location in the canvas Cartesian coordinate system, obtaining its rendered stroke image $\bar{I}_b^i$. Also, the renderer generates a 1-channel alpha map α_i with the same shape of $\bar{I}_b^i$, as a binary mask of s_i. Representing $I_{mid}^0 = I_{in}$ and $I_{mid}^n = I_{out}$, it is possible to write the stroke rendering operation:

$$I_{mid}^i = \alpha^i \cdot \bar{I}_b^i + \left(1 - \alpha^i\right) \cdot I_{mid}^{i-1} \qquad (3)$$

and the whole *Stroke Renderer* process as:

$$I_{out} = StrokeRenderer(I_{in}, S) \qquad (4)$$

The purpose of a *Stroke Predictor* is to predict a set of strokes that can cover the distinctions between an intermediate target and a canvas image. Taking I_c, I_t with dimension 3 x m x m as input (here, m is the stroke image's width and height, and 3 is the number of channels), *Stroke Predictor* passes I_c and I_t through two independent convolution neural networks to extract their feature maps as F_c, F_t with dimension c $\times$ $\frac{m}{4}$ $\times$ $\frac{m}{4}$.

Afterward, F_c, F_t, and a learnable positional encoding [9] are concatenated and flattened as the intake of the Transformer encoder. The decoder part takes N learnable stroke vectors as intake. Ultimately, the decoder predicts initial stroke parameters $\bar{S}_r = \{s_i\}_{i=1}^N$ and stroke confidence $C_r = \{c_i\}_{i=1}^N$ using two branches of fully-connected layers.

Furthermore, in the forward phase, confidence score c_i can be transformed to a decision $d_i = BinarySign(c_i)$, where *BinarySign* is a binary function whose value is 1 if c_i is positive and is 0 otherwise. The decision d_i is used to decide whether a predicted stroke should be painted on the canvas image. Because the *BinarySign* function has zero gradient almost everywhere to enable backpropagation sigmoid function $\sigma(x)$ is used to compute the gradient [8]:

$$\frac{\partial d_i}{\partial c_i} = \frac{\partial \sigma\left(c_i\right)}{\partial c_i} = \frac{\exp\left(-c_i\right)}{\left(1 + \exp\left(-c_i\right)\right)^2} \qquad (5)$$

Collecting all inferred strokes with positive decisions, it is possible to get the final $S_r = \{s_i\}_{i=1}^N$ with N strokes and define *Stroke Predictor* as:

$$S_r = StrokePredictor(I_c, I_t) \qquad (6)$$

Paint Transformer can simultaneously minimize the differences between target and prediction on both image and stroke levels. Here is a list of losses that are used to train Paint Transformer [8]: Pixel loss is straightforwardly used to recreate a target image. Thus, the pixel-wise loss $\mathcal{L}_{pixel}$ between I_r and I_t is minimized on the whole image level:

$$\mathcal{L}_{pixel} = \left\|I_r - I_t\right\|_1 \qquad (7)$$

Stroke loss is essential to define suitable metric for estimating the difference between strokes on the stroke level. To achieve the best results, three-component losses are combined in the loss. Stroke $\mathcal{L}_\infty$ distance is intuitive (s_u and s_v indicate parameters of strokes u and v, respectively):

$$\mathcal{D}_{L_1}^{u,v} = \left\|s_u - s_v\right\|_1 \qquad (8)$$

Wasserstein distance is used because using the L_1 metric alone ignores different scales for large and little strokes. A rectangular rotational stroke with parameters $\{x, y, h, w, \theta\}$ can be represented as a 2D Gaussian distribution $N(\mu, \sum)$ by the next equations as stated in [10]:

$$\begin{aligned} &\mu = (x, y), \\ &\Sigma^{\frac{1}{2}} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \frac{w}{2} & 0 \\ 0 & \frac{h}{2} \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \\ &= \begin{bmatrix} \frac{w}{2}\cos^2\theta + \frac{h}{2}\sin^2\theta & \frac{w-h}{2}\cos\theta\sin\theta \\ \frac{w-h}{2}\cos\theta\sin\theta & \frac{w}{2}\sin^2\theta + \frac{h}{2}\cos^2\theta \end{bmatrix} \end{aligned} \qquad (9)$$

Hence, the Wasserstein distance between Gaussian distributions $N(\mu_u, \sum_u)$ and $N(\mu_v, \sum_v)$ is (where Tr denotes the trace of a matrix):

$$\mathcal{D}_W^{u,v} = \left\|\mu_u - \mu_v\right\|_2^2 + \mathrm{Tr}\left(\Sigma_u + \Sigma_v - 2\left(\Sigma_u^{\frac{1}{2}}\Sigma_v\Sigma_u^{\frac{1}{2}}\right)^{\frac{1}{2}}\right) \qquad (10)$$

Binary cross-entropy is used to predict a stroke's confidence with the positive (negative) ground-truth decision should be as high (low) as possible. Let's assume s_v as a target stroke with ground-truth label g_v and s_u as a predicted stroke with confidence c_u and, where $g_v = 0$ if s_v is an empty stroke and $g_v = 1$ if s_v is a valid stroke:

$$\mathcal{D}_{bce}^{u,v} = -g_v \cdot \log \sigma(c_u) - (1 - g_v) \cdot \log(1 - \sigma(c_u)) \quad (11)$$

The number of valid ground-truth strokes varies during training. Paint Transformer has a matching instrument between the prediction set $\bar{S}_r$ of N strokes and the ground-truth set S_g of a maximum N strokes (there could be both empty and valid strokes in S_g) to compute the loss function. Paint Transformer uses the permutation of strokes that yields the minimal stroke-level matching cost to calculate final loss using the Hungarian algorithm. For prediction set $\bar{S}_r$ that has a stroke s_u and for the target set S_g that has a stroke s_v, their cost value is (corresponding cost for empty target strokes is always 0):

$$M_{u,v} = g_v(\mathcal{D}_{L_1}^{u,v} + \mathcal{D}_{W}^{u,v} + \mathcal{D}_{bce}^{u,v}) \quad (12)$$

Thus, marking the optimal permutations for predicted and target strokes as X and Y, respectively, that are provided by the Hungarian algorithm, respectively, the stroke loss is given by:

$$\mathcal{L}_{stroke} = \frac{1}{n} \sum_{i=1}^{n} g_{Y_i}(\mathcal{D}_{L_1}^{X_i Y_i} + \mathcal{D}_{W}^{X_i Y_i} + \mathcal{D}_{bce}^{X_i Y_i}) \quad (13)$$

Paint Transformer imitates a coarse-to-fine algorithm to mimic an artist and yield painting results during prediction. Provided a photo of dimension $H \times W$, Paint Transformer runs from coarse to fine in order on K rankings. Painting on each ranking is conditional on the result of the prior ranking. The target image and current canvas are cut into the number of non-overlapping $P \times P$ patches before being processed by the *Stroke Predictor*.

Neural Painters. The paper by Reiichiro Nakano investigates different experiments with neural painters built on differentiable simulations of a non-differentiable painting program. Firstly, two methods of training a neural painter using VAEs and GANs, respectively, are presented. Secondly, the paper recreates *SPIRAL* reconstruction results [11] using a non-RL learning adversarial technique with a neural painter. Thirdly, the use of a neural painter as a differentiable image parameterization is suggested. By optimizing strokes directly using backpropagation, a method is suggested to visualize pre-trained image classifiers by letting them to paint classes they were trained to determine [12]. For the purposes of this paper, we are specifically interested in the GAN-reconstruction of a non-differentiable painting program brushstrokes.

The action space represents the set of parameters that are used as control inputs for the MyPaint. The action space maps a single action to a single stroke in the MyPaint. An agent paints by sequentially yielding actions and spreading full strokes on a canvas. The action space consists of the next parameters [12]:

- Brush coordinates are a set of three Cartesian coordinates pairs representing the stroke shape. The coordinates describe a start point, end point, and middle control point, forming a quadratic Bezier curve. We denote them as $\{x_s, y_s, x_e, y_e, x_c, y_c\}$ respectively.
- Start and end pressure describe the pressure used on the brush at the start and end of the stroke. We denote them as $\{p_s, p_e\}$ respectively.
- Brush size that specifies the brush radius and denoted as s.
- Color consists of three variables that represent the RGB color of the brush and specified as $\{r, g, b\}$.

To recreate a MyPaint brushstroke using a neural network [12] proposes VAE and GAN methods. We focus here on the GAN [13] method because it produces sharper images than VAE and thus more accurate strokes. An adversarial loss is used to directly learn a mapping from actions to strokes. Unlike a typical GAN, the noise is not injected into the intake of the generator. Instead, the generator takes the input action and maps it directly to a stroke. The discriminator is provided with real and generated action-stroke pairs and tries to decide whether the pair is real. This is comparable to a conditional GAN [14]. Pairs of true strokes on the left and the complementary GAN neural painter results on the right.

Methodology

GAN-generated strokes training. In this paper, we propose combining GAN-stroke rendering system referenced above with Paint Transformer to introduce more complex strokes. Current Paint Transformer *Stroke Renderer* has only eight parameters, while GAN-stroke rendering has 12 with potential to increase up to 50 parameters that MyPaint supports. First of all, we needed to set up the MyPaint program to generate strokes for the training, so we prepared a setup script for macOS [15]. The training of the GAN network was done locally on an Intel CPU based Mac laptop. Since MyPaint generated the strokes using CPU,

there is a bottleneck for training performance even if GPU is available. The training code is based on the implementation of Neural Painters by [12]. The modification is that we directly feed generated MyPaint strokes into the discriminator in real-time from a data loader. At the same time, [12] pre-generated stroke images first and trained the network afterward. We also simplified the MyPaint API calling code and wrapped it in a data loader [15] for convenience. The following stroke parameters are used: $\{x_s, y_s, x_e, y_e, x_c, y_c, s, p_s, p_e\}$. We do not use color parameters to train strokes compared to the original implementation because we can colorize the strokes later in the Paint Transformer. We trained GAN strokes for about 11.7 million iterations, and it took about 36 hours to do so due to the CPU bottleneck.

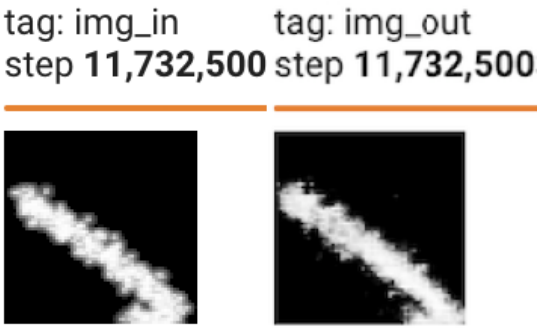


Figure 1. A GAN result sample on 11.7 million iterations

The sample of the result on the final iteration is provided in Figure 1. On the left (img_in) is the image painted by MyPaint and on the right (img_out) is the GAN-generated image on the same set of action parameters. The discriminator loss, generator score, and real score are provided in Figures 2, 3, and 4, respectively. The x-axis depicts iterations, and the y-axis is the numeric score.

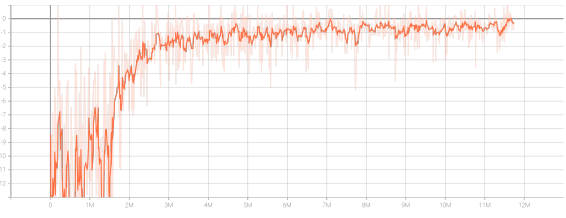


Figure 2. GAN-generated strokes discriminator loss

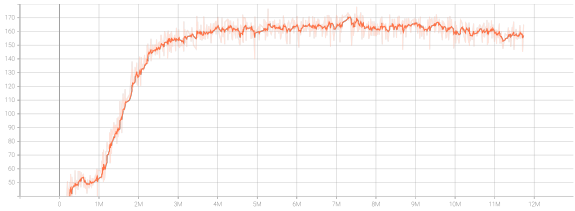


Figure 3. GAN-generated strokes generator score

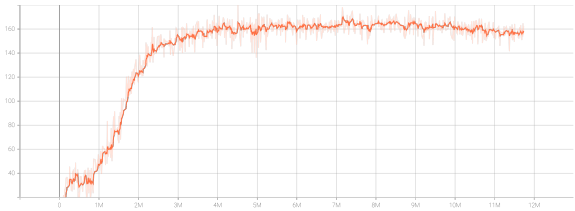


Figure 4. GAN-generated strokes real score

GAN Stroke Renderer. Once the GAN stroke predictor is trained, we can quickly predict the MyPaint stroke shape using nine parameters on the GPU with comparable quality to MyPaint. We can now utilize the GAN-generated strokes we trained, as a basis for a new stroke renderer for Paint Transformer that would yield more advanced strokes than the original. We denote the new stroke renderer as a *GAN Stroke Renderer*. The set of parameters is now $\{x_s, y_s, x_e, y_e, x_c, y_c, s, p_s, p_e, r, g, b\}$. As a first step, we infer set stroke shapes from $\{x_s, y_s, x_e, y_e, x_c, y_c, s, p_s, p_e\}$ using GAN-generated strokes pre-trained weights (step 1 denoted in Figure 5). Note that we do not train GAN-generated strokes anymore and use them as a predictor. The GAN output does not have clear zero pixels and has numbers close to zero instead. Therefore, we cannot create a binary mask immediately and must utilize a denoising solution. Considering Equation 3 for original *Stroke Renderer*, we form an alpha map via $\alpha^i = \bar{I}_b^i > > Q_{0.8}(\bar{I}_b^i)$, where $Q_{0.8}$ is 80th-percentile. By also forming a color map c^i from $\{r, g, b\}$ we can rewrite Equation 3 as (steps 2, 3 denoted in Figure 5):

$$I_{mid}^i = \alpha^i \cdot \bar{I}_b^i \cdot c^i + \left(1 - \alpha^i\right) \cdot I_{mid}^{i-1} \qquad (14)$$

Stroke Predictor modification. We would need to modify the Stroke Predictor so that it could work with the new GAN Stroke Renderer. We change the architecture of the Transformer to accept 12 parameters instead of 8 original. The main challenge is to modify the loss function so that it would take new parameters. Specifically, the Wasserstein distance needs a modification (Equation 9). Initially,

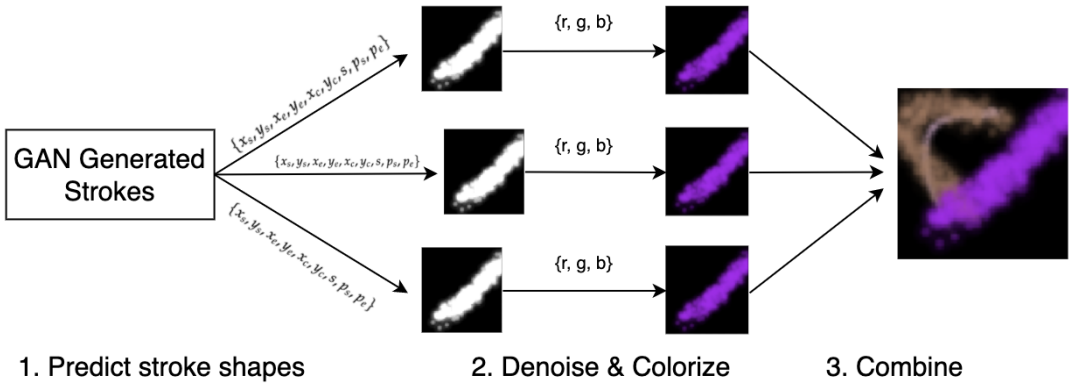
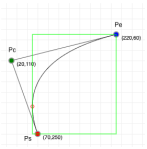


Figure 5. *GAN Stroke Renderer*

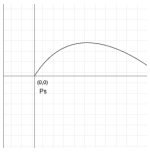
it accepts $\{x, y, h, w, \theta\}$, and we need to take $\{x_s, y_s, x_e, y_e, x_c, y_c, s, p_s, p_e\}$. Instead of modifying this loss function directly, we decided instead to translate $\{x_s, y_s, x_e, y_e, x_c, y_c, s, p_s, p_e\}$ into $\{x, y, h, w, \theta\}$ by creating a rotating bounding box around the stroke.

We know that the stroke is a quadratic Bezier curve represented by, where $P_0 = P_s = (x_s, y_s)$, $P_1 = P_c = (x_c, y_c)$, $P_2 = P_e = (x_e, y_e)$ and $t \in \in [0, 1]$:

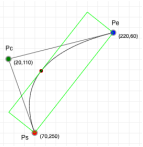
$$\begin{aligned}\mathbf{B}(t) &= \sum_{i=0}^{2} B_i^2(t) \cdot P_i = \sum_{i=0}^{2} t^i (1-t)^{2-i} \cdot P_i \\ &= (1-t)^2 \cdot P_0 + 2t(1-t) \cdot P_1 + t^2 \cdot P_2 \\ &= (1-t)^2 \cdot P_s + 2t(1-t) \cdot P_c + t^2 \cdot P_e\end{aligned} \tag{15}$$



(a) Non-rotating bounding box



(b) Curve alignment



(c) Rotating bounding box

Figure 6. Bezier curve [16]

Firstly, we can find a non-rotating bounding box by finding extremities of the Bezier curve by finding maxima and minima on the component functions, solving the equation $\mathbf{B}'(t) = 0$ [16]:

$$\begin{aligned}\mathbf{B}'(t) = 2(1-t)(P_c - P_s) + 2t(P_e - P_c) = 0 \implies \\ t = \frac{P_s - P_c}{-2 \cdot P_c + P_s + P_e}\end{aligned} \tag{16}$$

Now when we know t, we could find the solution and compare it with P_s and P_e. The lowest value is the lower point $P_{min} = min(\mathbf{B}(t), P_s, P_e)$, and the highest is the upper point for the bounding box $P_{max} = max(\mathbf{B}(t), P_s, P_e)$ (Figure 5a). To get a rotated bounding box, we need to make $P_s = (0, 0)$ and align the curve on the x-axis via (Figure 5b):

$$\begin{aligned}\alpha = \arctan\frac{y_e}{x_e} \implies R = \begin{bmatrix} \cos(-\alpha) & -\sin(-\alpha) \\ \sin(-\alpha) & \cos(-\alpha) \end{bmatrix} \\ \dot{P}_s = P_s - P_s = (0, 0) \\ \dot{P}_c = (P_c - P_s) \cdot R \\ \dot{P}_e = (P_e - P_s) \cdot R\end{aligned} \tag{17}$$

Afterward, we calculate a non-rotating bounding box for $\dot{P}_s, \dot{P}_c, \dot{P}_e$ via Equations 15, 16 and make a reverse transformation [16]:

$$\begin{aligned}\dot{P}_{min} = min(\dot{\mathbf{B}}(t), \dot{P}_s, \dot{P}_e) = (\dot{x}_{min}, \dot{y}_{min}) \\ \dot{P}_{max} = max(\dot{\mathbf{B}}(t), \dot{P}_s, \dot{P}_e) = (\dot{x}_{max}, \dot{y}_{max}) \\ (x_{max}, y_{max}) = (\dot{x}_{max}, \dot{y}_{max}) \cdot R^{-1} + P_s \\ (x_{min}, y_{min}) = (\dot{x}_{min}, \dot{y}_{min}) \cdot R^{-1} + P_s \\ (x'_{max}, y'_{max}) = (\dot{x}_{max}, \dot{y}_{min}) \cdot R^{-1} + P_s \\ (x'_{min}, y'_{min}) = (\dot{x}_{min}, \dot{y}_{max}) \cdot R^{-1} + P_s\end{aligned} \tag{18}$$

Thus, a rotating bounding box can be represented by four points $(x_{min}, y_{min}), (x_{max}, y_{max}), (x'_{min}, y'_{min}), (x'_{max}, y'_{max})$ as depicted in Figure 5c. We also need to account for start and end pressure and size; we found empirically that we can modify $\{x_s, y_s, x_e, y_e, x_c, y_c\}$ with $\{s, p_s, p_e\}$ before calculating the bounding box, which yields better results (clamp is used to

restrict a value between 0 and 1):

$$\begin{aligned} x_s &= \text{clamp}(x_s + 0.15 \cdot p_s,\ 0,\ 1); \\ y_s &= \text{clamp}(y_s + 0.15 \cdot p_s,\ 0,\ 1) \\ x_e &= \text{clamp}(x_e + 0.15 \cdot p_e,\ 0,\ 1); \\ y_e &= \text{clamp}(y_s + 0.15 \cdot p_e,\ 0,\ 1) \\ x_c &= \text{clamp}(x_c - 0.15 \cdot s,\ 0,\ 1); \\ y_c &= \text{clamp}(y_c - 0.15 \cdot s,\ 0,\ 1) \end{aligned} \tag{19}$$

Finally, we need to convert four coordinate points into $\{x, y, h, w, \theta\}$:

$$\begin{aligned} x &= \frac{x_{max} + x_{min}}{2}; \quad y = \frac{y'_{max} + y_{min}}{2} \\ w &= \sqrt{(y'_{max} - y_{max})^2 + (x'_{min} - x_{max})^2} \\ h &= \sqrt{(y_{max} - y'_{min})^2 + (x_{max} - x'_{max})^2} \\ \theta &= \arctan \frac{y'_{max} - y_{max}}{x'_{max} - x_{max}} \end{aligned} \tag{20}$$

The code implementation can be found in `get_rotated_bounding_box` [15] method, and the resulting bounding boxes on GAN-generated strokes are shown in Figure 7.

Figure 7. Rotated bounding box on GAN-generated strokes

Experiments

Training. Once the *Stroke Predictor* is optimized for the new set of parameters, we can train Paint Transformer with a *GAN Stroke Renderer* system. We trained Paint Transformer for 180 epochs, and the training process took about 6 hours on NVIDIA RTX 5000. In comparison, the original Paint Transformer takes about 5-5.5 hours to train 180 epochs on the same GPU. This means that the training time for our GAN extension did not increase significantly. In Figures 8 and 9, we can notice the training charts for pixel and stroke L_1 distance losses. Wasserstein distance loss and a binary cross-entropy decision loss are depicted in Figures 10 and 11, respectively. The x-axis depicts iterations, and the y-axis shows the numeric score. Canvas-target-predict triads S_b, S_f, and S_r, are shown in Figure 12 during the training. We also changed the monitoring framework from Visdom to the commonly used Tensorboard.

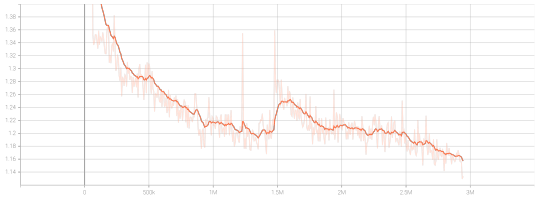


Figure 8. Paint Transformer pixel loss

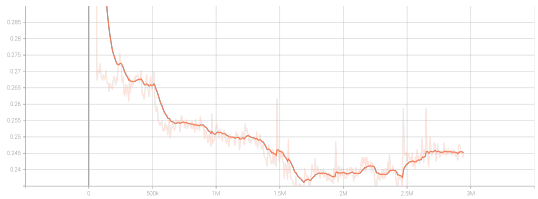


Figure 9. Paint Transformer stroke L_1 distance loss

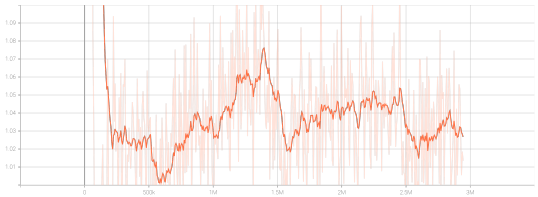


Figure 10. Paint Transformer Wasserstein distance loss

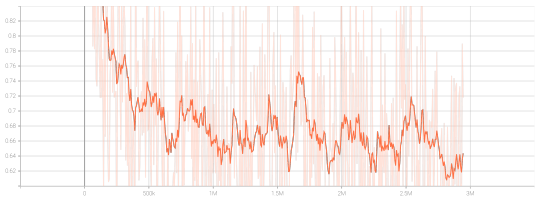


Figure 11. Paint Transformer binary cross-entropy decision loss

Results. We modified the inference module so it could work with *GAN Stroke Renderer* and obtained the results depicted in Figure 13c. We take Figure 13a as an input, and we also show the original Paint Transformer results in Figure 13b. For comparison, we choose the images in different settings: sunflower and frog are macro images, while the fjord and the city are landscape images. Overall, we can notice that our Paint Transformer extension paints the resulting pictures in a more granular fashion (using the same value of K as the original). This creates a more abstract painting style, especially in the fjord, city, and sunflower cases. We can also notice that the Paint Transformer extension does not paint larger strokes for uniform patches. We can also notice that the Paint Transformer extension does not paint larger

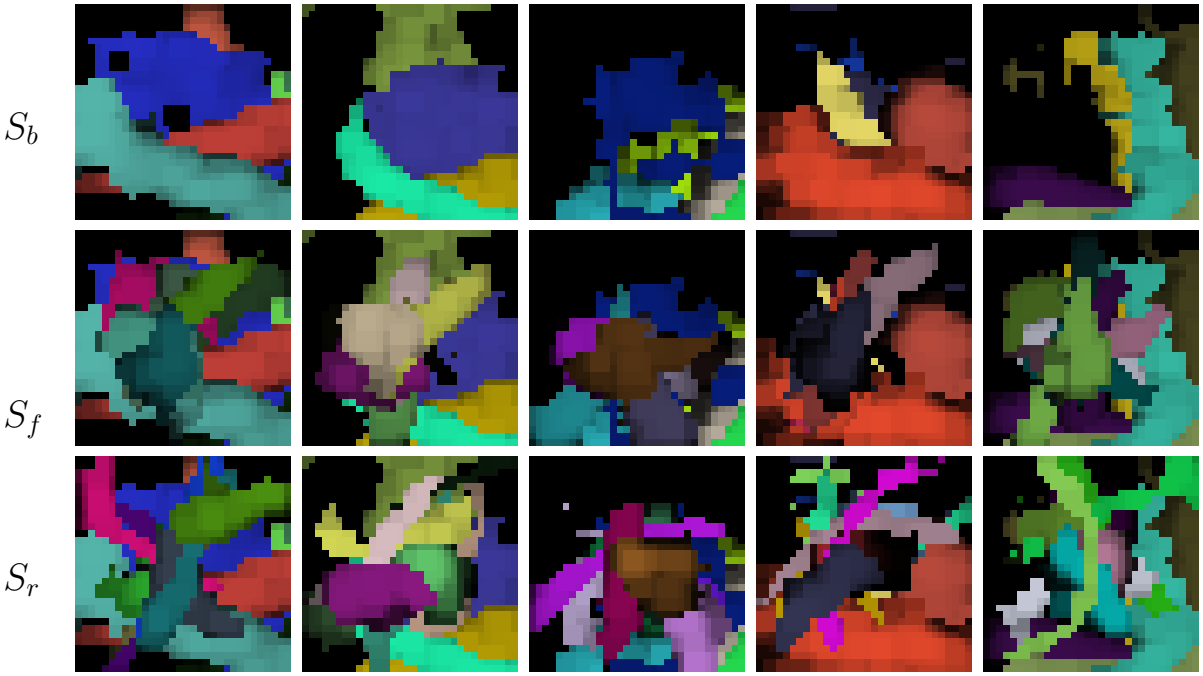
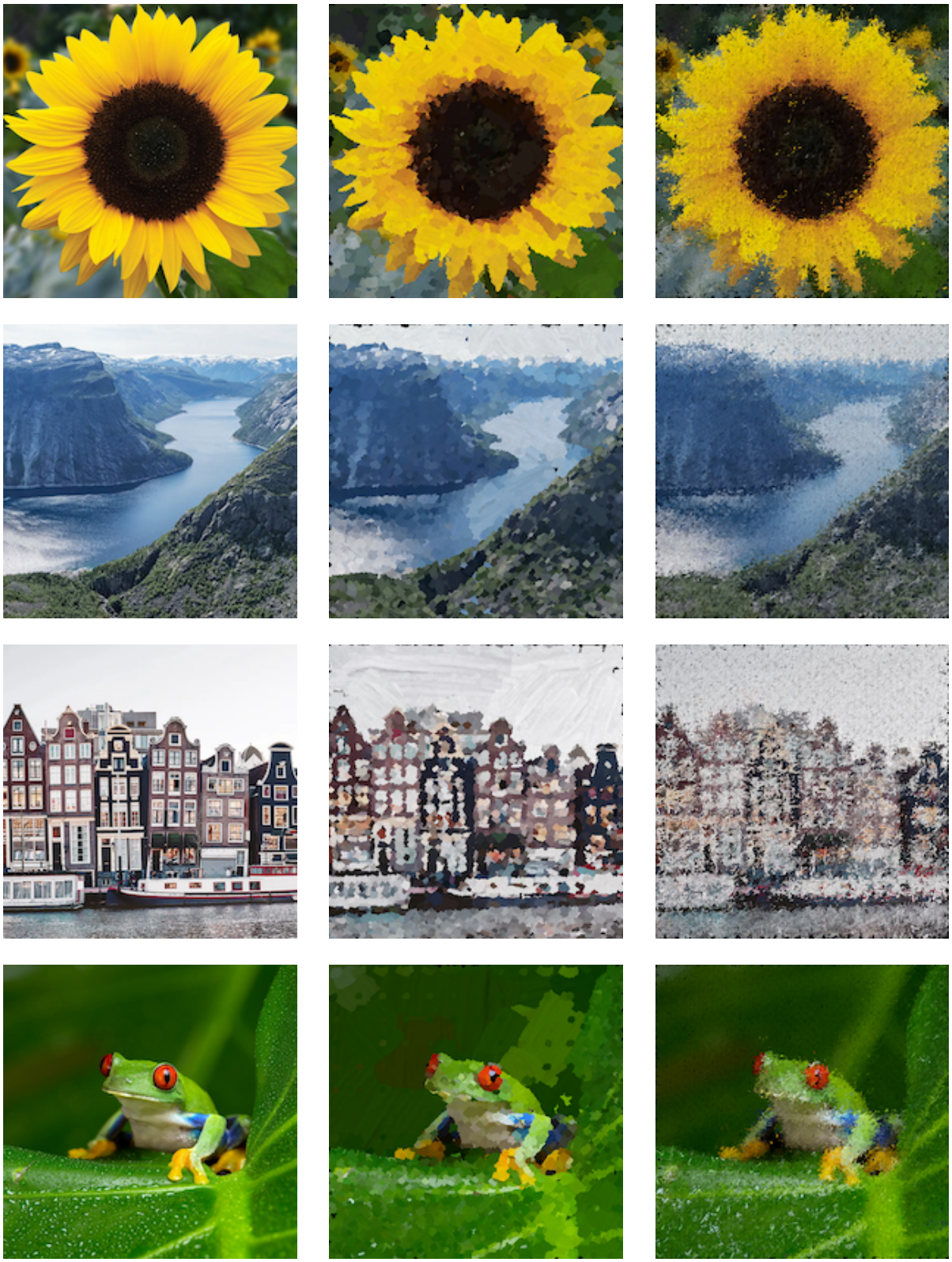


Figure 12. Canvas-target-predict triads in training

strokes for the uniform patches. We need to further modify Wasserstein distance loss and the binary cross-entropy decision loss to improve the results.

Conclusions

We proposed a GAN strokes extension to the Paint Transformer aimed at introducing more complex strokes. We refined the Stroke Rendering system, which generates strokes using a pre-trained GAN and has 12 parameters compared to the original 8. We partly modified the loss function to accept a new parameter list. The results have a different painting style and are more abstract; however, the extension paints strokes of similar size. This indicates that we need to make further effort in modifying Wasserstein distance loss and the binary cross-entropy decision loss to improve the results, which we plan to address in our future work. In addition, converting the network architecture to use 4-channel images might further enhance the results by removing artifacts on the generated strokes.



(a) Input image (b) Original output (c) GAN-extension output

Figure 13. Comparison of the input image, original output and GAN-extension output.

References

1. M. Poliakov, https://ekmair.ukma.edu.ua/handle/123456789/28820.
2. A. Elgammal, https://arxiv.org/abs/1706.07068.
3. L. A. Gatys, https://arxiv.org/abs/1508.06576.
4. P. Haeberli, in: Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques. — SIGGRAPH '90 (New York, NY, USA: Association for Computing Machinery, 1990), pp. 207–214. https://doi.org/10.1145/97879.97902.
5. D. Ha, https://arxiv.org/abs/1704.03477.
6. T. Zhou, https://arxiv.org/abs/1810.05977.
7. Z. Zou, https://arxiv.org/abs/2011.08114.
8. S. Liu, https://arxiv.org/abs/2108.03798.
9. A. Vaswani, https://arxiv.org/abs/1706.03762.
10. X. Yang, https://arxiv.org/abs/2101.11952.
11. Y. Ganin, https://arxiv.org/abs/1804.01118.
12. R. Nakano, https://arxiv.org/abs/1904.08410.
13. I. J. Goodfellow, https://arxiv.org/abs/1406.2661.
14. M. Mirza, https://arxiv.org/abs/1411.1784.
15. M. Poliakov, https://github.com/mxpoliakov/PaintTransformerGAN.
16. Pomax. A primer on bezier curves, https://pomax.github.io/bezierinfo.

Поляков М. Х., Швай Н. О.

РОЗШИРЕННЯ МОЖЛИВОСТЕЙ PAINT TRANSFORMER З ГЕНЕРУВАННЯМ МАЗКІВ ПЕНЗЛЯ ЗА ДОПОМОГОЮ GAN

Нейронне малювання створює послідовність мазків для заданого зображення і художньо відтворює його за допомогою нейронних мереж. У цій статті ми досліджуємо нову архітектуру, основану на Transformer, під назвою Paint Transformer, яка прогнозує параметри набору мазків за допомогою прямопрохідної нейронної мережі. Paint Transformer забезпечує кращі результати малювання порівняно з попередніми методами, маючи нижчі витрати на навчання та використання. У статті також пропонується нове розширення Paint Transformer, яке додає більш складні мазки, згенеровані GAN, для досягнення більш художнього та абстрактного стилю малювання, ніж оригінальний метод.

Ключові слова: нейронне малювання, трансформер, GAN.

Матеріал надійшов 07.01.2025

УДК 519.226:04.855

O. Mytnyk

DOI: 10.18523/2617-70807202444-50

ROBUST BAYESIAN REGRESSION MODEL IN BERNSTEIN FORM

In this paper, we present an inductive method for constructing robust Bayesian Polynomial Regression (BPR) models in Bernstein form, referred to as PRIAM (Polynomial Regression Inductive AlgorithM). PRIAM is an algorithm designed to determine stochastic dependence between variables. The triple nature of PRIAM combines the advantages of Bayesian inference, the interpretability of neurofuzzy models in Bernstein form, and the robustness of the support vector approach. This combination facilitates the integration of state-of-the-art machine learning techniques in decision support systems. We conduct experiments using well-known datasets and real-world economic, ecological, and meteorological models. Furthermore, we compare the forecast errors of PRIAM against several competitive algorithms.

Keywords: PRIAM, Bayesian inference, BPR, neurofuzzy model, polynomials in Bernstein form.

Introduction

Data mining competitions are an effective tool for evaluating the performance of specific methods among the growing variety of approaches. Recent contests, such as those hosted on Kaggle (https://www.kaggle.com/competitions) and the Data Mining Cup (http://www.data-mining-cup.com) have demonstrated the advantages of Bayesian and support vector (SV) methods. However, despite their high performance, these methods often face challenges in seamless integration into decision support systems. In contrast, neurofuzzy modeling offers an appealing framework for knowledge representation. This work seeks to combine the strengths of Bayesian reasoning, the robustness of the SV approach, and the interpretability of neurofuzzy modeling.

Brief historical outlook. Bayesianism began with Savage's personalistic school of thought and gained strength through the objective selection of prior probabilities based on the maximum entropy principle [1]. Since then, Bayesianism has inspired a series of significant contributions. For instance, Bayesian Occam's razor was demonstrated by Gull [2] as a method to estimate the parameters of prior probabilities in regression analysis. This concept was later applied by MacKay for the regularization of artificial neural networks in the so-called Bayesian evidence framework [3]. Additionally, the theory of Gaussian Processes (GP) incorporates evidence, also referred to as marginal likelihood, as a fundamental component of Bayesian inference [4].

Brown and Harris [5] established a correspondence between associative memory networks and fuzzy logic in neurofuzzy adaptive models. These models combine the transparent knowledge representation of fuzzy systems with the analytical ability to learn from observations. The ability to describe the behavior of neurofuzzy models as a series of human-readable linguistic rules makes them particularly well-suited for expert systems. However, conventional neurofuzzy models often suffer from the curse of dimensionality. To address this, Hong and Harris [6] proposed a polynomial complexity neurofuzzy approach.

Another efficient approach to process analysis is the Statistical Learning Theory (SLT) developed by Vapnik [7]. SLT is founded on the structural risk minimization principle, which is implemented in support vector machines (SVM) for classification problems [8]. SVM has since been extended to regression problems, leading to the development of support vector regression (SVR)[**?**], and further refined into Bayesian SVR[10]. In this work, we leverage the support vector (SV) approach to enhance the robustness of our models.

General problem statement. Suppose we observe the data:

$$\mathcal{D} = \{(y_j, \mathbf{x}_j)\}_{j=1}^{N}, \quad y \in \mathbb{R}, \quad \mathbf{x} \in \mathcal{X} = \mathbb{R}^n.$$

We hypothesize the existence of a stochastic dependence that maps each $\mathbf{x}$ to some value y obtained from a random trial governed by the law $p(y|\mathbf{x})$. To determine this stochastic dependence, we aim to identify the probability density function $p(y|\mathbf{x})$. However, this inverse problem is inherently ill-posed. Using the finite training set $\mathcal{D}$, we can only estimate posterior predictive distribution $p(y|\mathbf{x}, \mathcal{D})$. This estimation depends on the confidence in the observed data and the regularization methods applied to make the problem well-posed. In this paper, we focus on finding the mean of the posterior predictive distribution along with its variance.

The paper is organized as follows. We begin with an overview of the Bayesian framework. Next, we explore neurofuzzy models in Bernstein form and introduce a procedure for searching suboptimal models. Robustness is incorporated into the model afterward. As a result, we propose PRIAM – an inductive algorithm for constructing robust BPR models in Bernstein form, capable of encoding prior knowledge and generalizing effectively. Finally, we conduct experiments with PRIAM on both synthetic and real-world datasets, comparing its performance to that of other competitive algorithms.

Bayesian Framework

Let the systematic component of the stochastic dependence be described by a latent function f of a model $\mathcal{M}$ from the model space $\mathcal{H}$. This raises the following questions: how should the model space $\mathcal{H}$ be chosen, how should the model $\mathcal{M} \in \mathcal{H}$ be selected, and how can the function $f \in \mathcal{M}$ be determined. Bayesian reasoning provides answers to the last two questions.

Selection of a model $\mathcal{M}$. According to the Bayesian approach, the model with the maximum posterior probability $P(\mathcal{M}|\mathcal{D})$ is selected. Assuming a flat prior distribution of models over the space $\mathcal{H}$ (i.e. complete ignorance), the models are ranked by their marginal likelihood $p(\mathcal{D}|\mathcal{M})$, also known as evidence. Evidence reflects the ability of the model $\mathcal{M}$ to generate the data $\mathcal{D}$ and is defined as the following Lebesgue integral:

$$p(\mathcal{D}|\mathcal{M}) = \int_{\mathcal{M}} p(\mathcal{D}|f, \mathcal{M}) d\mu(f), \qquad (1)$$

where $\mu(f)$ represents the prior probability measure on the function space $\mathcal{M}$. The likelihood $p(\mathcal{D}|f, \mathcal{M})$ reflects the ability of the function $f \in \mathcal{M}$ to generate the data $\mathcal{D}$. Assume that the random component of the stochastic dependence is represented by additive noise, so that $y_j = f(\mathbf{x}_j) + \delta_j$, where δ_j are independent and identically distributed random variables. Under this assumption, the likelihood takes the following form:

$$p(\mathcal{D}|f, \mathcal{M}) = \prod_j^N p(\delta_j|f, \mathcal{M}),$$

where $p(\delta|f, \mathcal{M})$ is a noise model. Both the noise model and the prior measure $\mu(f)$ will be selected later.

The evidence (1) can be approximated using various techniques, including expectation propagation (EP), Laplace's method, Markov chain Monte Carlo (MCMC). An overview and comparison of these methods are provided by Kuss [11].

Selection of a function f. The posterior probability measure can be derived using Bayes' rule:

$$d\mu(f|\mathcal{D}) = \frac{p(\mathcal{D}|f, \mathcal{M}) d\mu(f)}{p(\mathcal{D}|\mathcal{M})}.$$

Our goal is to determine the posterior predictive distribution, which represents the posterior beliefs about the output value y. This distribution is obtained by integrating over the posterior uncertainty of the function:

$$p(y|\mathbf{x}, \mathcal{D}, \mathcal{M}) = \int_{\mathcal{M}} p(y|\mathbf{x}, f, \mathcal{M}) d\mu(f|\mathcal{D}).$$

According to Bayesian decision theory, to obtain a single function estimate $g \in \mathcal{M}$ for regression $y(\mathbf{x})$, we minimize the Bayesian risk, defined as the expectation of a loss functional L:

$$R(g) = \mathbb{E}_f \left[L(f, g) \right] = \int_{\mathcal{M}} L(f, g) d\mu(f|\mathcal{D}).$$

The loss functional L reflects the researcher's subjective attitude toward risk. Typically, the choice of L is a point of debate among researchers. We will establish our choice of L later.

In the next section, we address the question of how to select the model space and organize the model search process.

Polynomial Regression in Bernstein Form

Hong and Harris [6] introduced neurofuzzy models based on the following truncated ANOVA decomposition for input variable $\mathbf{x} = \left\{x^i\right\}_{i=1}^n$:

$$f(\mathbf{x}) = b + \sum_{k=1}^{n} B_k^d(x^k) + \sum_{q>p}^{n} B_{pq}^d\left(x^p, x^q\right). \qquad (2)$$

B_k^d, B_{pq}^d are univariate and bivariate polynomials in Bernstein form, defined as linear combinations of Bernstein basis polynomials of degree d:

$$B_k^d(x^k) = \textstyle\sum_{j=0}^d w_j^k \phi_j^d \left[s(x^k)\right],$$
$$B_{pq}^d(x^p, x^q) = \textstyle\sum_{i+r+t=d} w_{irt}^{pq} \phi_{irt}^d \left[\mathbf{u}(x^p, x^q)\right].$$

We refer to models (2) as neurofuzzy models in Bernstein form. The Bernstein basis polynomials are defined as:

$$\phi_j^d(s) = \binom{d}{j} \cdot s^j (1-s)^{d-j},$$
$$\phi_{irt}^d(\mathbf{u}) = \binom{d}{i, r, t} u^i v^r (1-u-v)^t.$$

To determine the barycentric coordinates s and $\mathbf{u} = \{u, v\}$ we follow the approach proposed in [12], which introduces a fast inverse de Casteljau mapping based on a uniform knot layout.

After training such neurofuzzy models, dependencies can be interpreted using fuzzy logic and generate a set of fuzzy rules [5]. It is well-known that Bernstein basis polynomials are non-negative and satisfy the unity of support property: $\sum_j \phi_j = 1$. Therefore, Bernstein basis polynomials are valid fuzzy membership functions. The advantages of this approach include: transparency of the model structure, interpretation of dependencies in terms of fuzzy logic, and polynomial complexity of the resulting set of fuzzy rules.

Selection of a model space $\mathcal{H}$. Let us leverage the advantages of neurofuzzy modeling. To achieve this, we define the configuration of the model space as an upper triangular $[n \times n]$ matrix $\mathbf{C}$. Each diagonal element c_k represents the degree of a univariate polynomial in Bernstein form for the factor x^k. Each element above the diagonal, $c_{q>p}$, represents the degree of a bivariate polynomial in Bernstein form for the pair x^p and x^q. The corresponding model space is expressed as:

$$\mathcal{M}(\mathbf{w}, b, \mathbf{x}) = \mathcal{H}(\mathbf{C}, \mathbf{w}, b, \mathbf{x}) = \\ = b + \sum_{k=1}^{n} B_k^{c_k}(x^k) + \sum_{q>p}^{n} B_{pq}^{c_{pq}}(x^p, x^q), \quad (3)$$

where parameters $\mathbf{w} = \{\ldots, w_j^k, \ldots, w_{irt}^{pq}, \ldots\}$. The models in the form (3) generalize those in the form (2). Furthermore, the models (3) can also be considered linear in parameters $\mathbf{w}$ within a high-dimensional Euclidian space $\mathcal{W}$ with the canonical scalar product $\langle \cdot, \cdot \rangle$ and norm $\| \cdot \|$:

$$f(\mathbf{x}) = \mathcal{M}(\mathbf{w}, b, \mathbf{x}) = \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + b,$$

where $\Phi(\mathbf{C}) : \mathcal{X} \to \mathcal{W}$ represents the mapping:

$$\Phi : \mathbf{x} \mapsto \{\ldots, \phi_j^{c_k}\left(x^k\right), \ldots, \phi_{irt}^{c_{pq}}\left(x^p, x^q\right), \ldots\}.$$

Model search in neurofuzzy model space. Algorithm 1 demonstrates how an initial model guess can be refined using evidence-based calculations.

Algorithm 1 Model search

Input: observations $\mathcal{D}$, convergence level $\nu > 0$, initial model $\mathcal{M}^{(0)} = \mathcal{H}\left(\mathbf{C}^{(0)}\right)$
Result: suboptimal model $\mathcal{M}_{\text{opt}}$
Iterator $t \longleftarrow 0$
repeat
  $\mathcal{M}_{\text{opt}} \longleftarrow \mathcal{M}^{(t)}$
  Generate a set of candidate models: $\left\{\mathcal{M}_{ij}^{(t+1)} = \mathcal{H}\left(\mathbf{C}_{ij}^{(t+1)}\right)\right\}_{ij}$, where $\mathbf{C}_{ij}^{(t+1)} = \mathbf{C}^{(t)} \pm \mathbf{1}_{ij}, \quad 1 \leqslant i \leqslant j \leqslant n.$
  Choose the model with the maximum evidence: $\mathcal{M}^{(t+1)} = \arg\left[p^{(t+1)} = \max p\left(\mathcal{D}|\mathcal{M}_{ij}^{(t+1)}\right)\right]$
  $t \longleftarrow t + 1$
until $p^{(t)} < p^{(t-1)} + \nu$

First, we define the space $\mathcal{H}$ of models $\mathcal{M}$ in the form (3). Based on prior assumptions about the model structure, the initial configuration $\mathbf{C}^{(0)}$ is constructed. At each step, a set of candidate models is generated, each differing in the degree of one polynomial in Bernstein form. For each candidate model, the evidence is calculated. The model with the maximum evidence is selected. The new model is accepted if its evidence exceeds that of the previous model by a threshold ν. This threshold determines the linear convergence speed and reflects the degree of confidence in the prior model structure.

Robust BPR in Bernstein Form

To leverage the robustness of Support Vector Regression (SVR), we define the noise model as:

$$p(\delta_j | f, \mathcal{M}) = \frac{\beta}{2(1 + \epsilon\beta)} \exp(-\beta |\delta_j|_\epsilon), \quad (4)$$

where $|\cdot|_\epsilon$ is the ϵ-insensitive loss function (ϵ-ILF), which provides sparseness and robustness to the BPR models. The parameters ϵ and β are referred to as hyperparameters. In this work we assume flat priors for these hyperparameters.

Let the prior probability measure $\mu(f)$ have a density $p(\mathbf{w}|\mathcal{M})$ with respect to the Lebesgue measure on the parameter space $\mathcal{W}$:

$$d\mu(f) = p(\mathbf{w}|\mathcal{M}) d\mathbf{w}.$$

Let this density be a multivariate Gaussian with $\mathbf{0}$ mean and identity covariance matrix $\mathbf{I}$:

$$p(\mathbf{w}|\mathcal{M}) = \mathcal{N}(\mathbf{0}, \mathbf{I}).$$

We define the loss functional L as $L(f, g) = \{0 \text{ if } f = g; 1 \text{ if } f \neq g\}$. In this case, the

Bayesian risk is minimized at the mode of the posterior function distribution, yielding the so-called Maximum a Posteriori (MAP) estimate. The MAP estimate for BPR with the noise model (4) corresponds to the canonical SVR problem:

$$R(f) = \beta \sum_{j=1}^{N} |\delta_j|_\epsilon + \frac{1}{2}\|\mathbf{w}\|^2 \longrightarrow \min_f, \qquad (5)$$

with solution in the form:

$$f_{\text{map}}(\mathbf{x}) = \textstyle\sum_j (\alpha_j - \alpha_j^*)\langle \Phi(\mathbf{x}_j), \Phi(\mathbf{x})\rangle + b_{\text{map}},$$
$$\mathbf{w}_{\text{map}} = \textstyle\sum_j (\alpha_j - \alpha_j^*)\Phi(\mathbf{x}_j),\ \alpha_j, \alpha_j^* \in (0, \beta),$$
$$b_{\text{map}} = \text{mean}_j \left\{ \begin{array}{l} y_j - \langle \mathbf{w}, \Phi(\mathbf{x}_j)\rangle - \epsilon \\ y_j - \langle \mathbf{w}, \Phi(\mathbf{x}_j)\rangle + \epsilon \end{array} \right\},$$

where α_j, α_j^* are the Lagrange multipliers of the corresponding quadratic programming (QP) problem.

The algorithm 1, under the assumptions described above, is referred to as PRIAM. In the following subsections, we demonstrate how to estimate the evidence and error bars in PRIAM.

Evidence Estimation

For fast evidence estimation, we adopt the approach described in [13], where a locally smoothed loss function is used to approximate ϵ-ILF. This approach yields the following approximation, referred to as Bayesian Evidence Criterion (BEC), for negative logarithm of the Bayesian evidence:

$$-\ln p(\mathcal{D}|\mathcal{M}) \approx \text{BEC}(\mathcal{M}, \epsilon, \beta) = \\ = R(\mathbf{w}_{\text{map}}) - N \ln \frac{\beta}{2(1+\epsilon\beta)}. \qquad (6)$$

Although the BEC approximation is not entirely accurate, it preserves sparseness and is recognized as the fastest method for model comparison.

While evidence should be maximized, BEC should be minimized. Additionally, since BEC depends on the hyperparameters ϵ and β, it can also be minimized with respect to these hyperparameters:

$$\text{BEC}(\mathcal{M}) = \min_{\epsilon,\beta} \text{BEC}(\mathcal{M}, \epsilon, \beta). \qquad (7)$$

This is a nonlinear minimization problem. The gradient of BEC is expressed as:

$$\nabla^{\text{BEC}}_{\epsilon,\beta} = \left[\frac{N\beta}{1+\epsilon\beta} - \beta N_{\text{SV}}, N R_{\text{emp}} - \frac{N}{\beta(1+\epsilon\beta)} \right]$$

where N_{SV} is the number of support vectors, and the empiric risk is defined as $R_{\text{emp}} = \sum_{j=1}^{N} |\delta_j|_\epsilon$. To minimize (7) with respect to the hyperparameters, we employ the Interior Reflective Newton (IRN) method. IRN is known for its global and quadratic convergence properties [14].

Estimation of error bars. The variance of the noise model (4) can be easily computed as:

$$\sigma_N^2 = \frac{2}{\beta^2} + \frac{\epsilon^2(\epsilon\beta + 3)}{3(\epsilon\beta + 1)}.$$

It is well-known that, for Gaussian noise, the posterior predictive distribution is also Gaussian $p(y_*|\mathbf{x}_*, \mathcal{D}) = \mathcal{N}(y_*|f_{\text{mean}}(\mathbf{x}_*), \sigma^2)$, with variance

$$\sigma^2 = k_{**} - \mathbf{k}_*^\top (\mathbf{K} + \sigma_N^2 \mathbf{I})^{-1} \mathbf{k}_* + \sigma_N^2, \qquad (8)$$

where matrix $\mathbf{K} \sim k(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j)\rangle$, $\mathbf{k}_* = [k(\mathbf{x}_1, \mathbf{x}_*), \ldots, k(\mathbf{x}_N, \mathbf{x}_*)]^\top$ and $k_{**} = k(\mathbf{x}_*, \mathbf{x}_*)$. Although our noise model differs from the normal distribution, it can be shown that the distribution (4) is sufficiently close to a normal distribution with the same variance to justify using (8) as an acceptable approximation for the posterior variance. Furthermore, as demonstrated by Gao [15], the computation of $\mathbf{k}_*$ and $\mathbf{K}$ can be reduced to the marginal support vectors $\mathbf{X}_M = \{\mathbf{x}_i : |y_i - f(\mathbf{x}_i)| = \epsilon\}$. Finally, we use $\pm 2\sigma$ to represent the 95% confidence interval.

Pros and cons of the SVR with BEC approach. Advantages:

1. SV expansion is independent of the input space dimension, mitigating the curse of dimensionality in reconstruction problems.
2. A unique solution is obtained after training, as it is derived from solving a QP problem.
3. The SVR model exhibits robustness and sparseness.
4. The BEC provides an exceptionally fast model search method.

Disadvantages:

1. The BEC computation lacks precision, necessitating the selection of models with significantly smaller BEC values during model comparison.
2. Relying on the mode of the posterior function distribution for a given model (MAP estimation) deviates from pure Bayesian inference principles.

Experiments

For the experiments, we selected the Longley and Filip datasets from the Statistical Reference Datasets project [16]. Additionally, we included the well-known synthetic Friedman dataset. The AutoMPG dataset, which represents city cycle fuel consumption, was obtained from the UCI Machine Learning Repository [17]. The CPI and RCON datasets correspond to economic models of the Consumer Price Index and Real Consumption, respectively, as studied in [18]. The WIW dataset represents a meteorological wind-induced

wave model, while IBSS corresponds to an ecological model of macrozoobenthic biomass.

We compared PRIAM with GMDH (Group Method of Data Handling), NF-GMDH (Neurofuzzy Group Method of Data Handling, implemented in "GMDH Modeler 0.9.37"), RNN (Recurrent Neural Networks), and ANFIS (Adaptive Neuro Fuzzy Inference System), both implemented in "NeuroSolutions 5". Additionally, we evaluated GPR (Gaussian Process Regression) and XGBoost (Extreme Gradient Boosting), both available as Python packages. A brief excerpt of our experimental results is shown in Table 1.

The table also includes information about the size of the full dataset (N), the size of the learning dataset (N_{learn}), and the number of input factors (n) for each problem. A significant discrepancy between the MSEs of two different methods can be detected using Fisher statistics $F^{(80\%)}_{N-n,N-n}$.

Let us create a rating table for different algorithms. The algorithm with the smallest MSE result receives 10 points, the second 8 points, and so on. The two algorithms with the worst results receive no points. If multiple algorithms show insignificant differences in results, they share the same number of points. This way 30 points are distributed among all algorithms for each dataset.

As shown in Table 2, PRIAM achieves the highest rating among all the algorithms. Its performance is stable and never ranks among the worst. It is worth noting, however, that this rating is not absolute but instead reflects the strength of a specific algorithm in a particular implementation.

In the next subsection, we provide a detailed description of the result for the RCON dataset.

Dynamics of real consumption. The model for real consumption ($RCON$) is defined as a function of two factors: the interest rate (R) and real domestic income (RDI). The dataset consists of 24 observations, corresponding to monthly samples over a two years period. The first 14 points are used for training, while the last 10 points are reserved for forecasting and calculating the generalization error. The initial configuration of the model $\mathbf{C}^{(0)} = \mathrm{diag}\{1, 1\}$ reflects our prior belief in linear dependencies. The optimal PRIAM model is shown in Fig. 1.

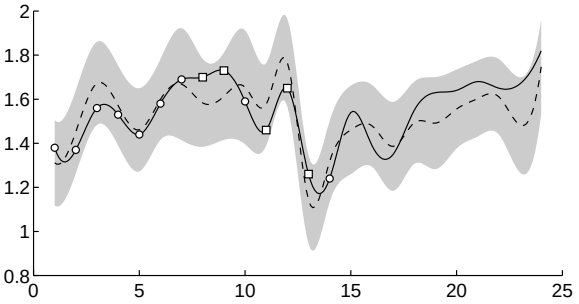
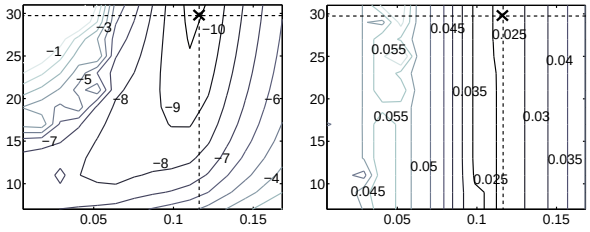


Figure 1. RCON dataset and optimal PRIAM model with 95% confidence interval. Squares stand for SVs, circles are vectors inside ϵ-tube.



(a) BEC contours (b) NMSE contours

Figure 2. The contour plots illustrate the dependencies of BEC and generalization error on the width of the ϵ-tube (abscissa axis) for different values of the hyperparameter β (ordinate axis). Crosses indicate the optimal hyperparameter values.

The corresponding optimal model configuration is given by $\mathbf{C}_{\mathrm{opt}} = \mathrm{diag}\{1, 2\}$, highlighting the reinforcing effect of RDI. The normalized model representation using the SV expansion is as follows:

$$f_{\mathrm{opt}}(\mathbf{x}) = 0.43 + 6.6k(\mathbf{x}_8, \mathbf{x}) + 25.5k(\mathbf{x}_9, \mathbf{x}) - \\ - 29.8k(\mathbf{x}_{11}, \mathbf{x}) - 2.6k(\mathbf{x}_{12}, \mathbf{x}) + 0.3k(\mathbf{x}_{13}, \mathbf{x}).$$

Dual model representation in neurofuzzy space:

$$f_{\mathrm{opt}}(\mathbf{x}) = 0.43 + 0.06\phi_0^1(x^1) - 0.06\phi_1^1(x^1) - \\ - 0.58\phi_0^2(x^2) + 0.13\phi_1^2(x^2) + 0.45\phi_2^2(x^2).$$

where $x^1 \equiv R$, $x^2 \equiv RDI$. Here, we observe a weak dependence of $RCON$ on R.

To evaluate the efficiency of BEC, we conduct a more detailed analysis of the relationship between BEC and generalization error with respect to the hyperparameters. According to the BEC contours (Fig. 2a), the optimal hyperparameter region is characterized by ϵ near 0.12, and high values of β. PRIAM successfully identifies the optimal hyperparameters, as $\beta = 29.8$ and $\epsilon = 0.12$. MSE contours (Fig. 2b) further confirm the efficiency of $\epsilon \approx 0.12$. However, they also indicate that forecasting is largely indifferent to the value of β for

Table 1. Comparison of algorithms on different datasets by normalized MSE

Algorithm	Longley	Filip	Friedman	AMPG	CPI	RCON	WIW	IBSS
$N(N_{learn})$	16(11)	82(33)	1000(500)	392(300)	24(14)	24(14)	166(100)	50(25)
n	6	1	10	4	4	2	3	2
$F^{(80\%)}_{N-n,N-n}$	2.2	1.3	1.0	1.0	1.8	1.7	1.1	1.4
PRIAM	0.017	0.004	0.009	0.022	0.003	0.027	0.033	0.076
GMDH	0.051	0.016	0.029	0.024	0.130	0.042	0.025	0.124
NF-GMDH	0.001	0.039	0.037	0.026	0.008	0.098	0.043	0.053
RNN	0.018	0.012	0.008	0.028	0.090	0.101	0.033	0.083
ANFIS	0.002	0.001	0.009	0.027	0.003	0.051	0.031	0.110
GPR	0.054	0.010	0.008	0.025	0.096	0.043	0.032	0.086
XGBoost	0.022	0.001	0.004	0.019	0.073	0.063	0.041	0.140

Table 2. Algorithm rating

Algorithm	Longley	Filip	Friedman	AMPG	CPI	RCON	WIW	IBSS	Rating
PRIAM	4	6	3	8	9	10	5	6	**51**
GMDH	0	0	0	6	0	6	10	1	**23**
NF-GMDH	9	0	0	2	6	0	0	10	**27**
RNN	4	3	7	0	2	0	5	6	**27**
ANFIS	9	9	3	0	9	6	5	1	**42**
GPR	0	3	7	4	2	6	5	6	**33**
XGBoost	4	9	10	10	2	2	0	0	**37**

wide ϵ-tubes. The significance of β becomes notable only for narrower tubes.

Generation of fuzzy rules. We demonstrate how fuzzy rules can be generated based on the model in neurofuzzy space [19]. A balanced neurofuzzy model, like the one described above, can be decomposed into two neurofuzzy submodels in the canonical form due to the unity of support property:

$$f_1(x^1) = 0.49\mu_{A_0^1}(x^1) + 0.37\mu_{A_1^1}(x^1),$$
$$f_2(x^2) = -0.15\mu_{A_0^2}(\cdot) + 0.56\mu_{A_1^2}(\cdot) + 0.88\mu_{A_2^2}(\cdot)$$

where the Bernstein basis polynomials are used as fuzzy membership functions, $\phi_j^d \equiv \mu_{A_j^d}$, with the corresponding fuzzy labels on input space: A_0^1 — *low R*, A_1^1 — *high R*, A_0^2 — *low RDI*, A_1^2 — *average RDI*, A_2^2 — *high RDI*. Each submodel generates simplified rules independently and contributes to a fuzzy knowledge base of reduced complexity.

The rules and their confidences can be easily determined if the output fuzzy membership functions are represented as B-splines. In this case, at most two adjacent coefficients are nonzero. Let us define fuzzy membership functions for the *RCON* output variable, normalized on [0; 1], using three second-order B-splines μ_{B_k} with a triangular shape. These B-splines are defined over the knots {-0.5; 0; 0.5; 1; 1.5} with peaks at {0; 0.5; 1}. They correspond to the following fuzzy labels: B_0 — *low RCON*, B_1 — *average RCON*, B_2 — *high RCON*.

The rule R_j^i produced by submodel f_i is expressed as: "if $x^i \in A_j^i$, then $y \in B_k$ with confidence c_{kj}^i", where the rule confidences c_{kj}^i are determined by converting the weights of the model in the neurofuzzy space as follows:

$$c_{kj}^i = \mu_{B_k}\left(f_i\left(\arg\max_{x^i} \mu_{A_j^i}(x^i)\right)\right).$$

Thus, we derive five rules:

R_0^1: *if R is low, then RCON is low* (0.02) *or average* (0.98)
R_1^1: *if R is high, then RCON is low* (0.26) *or average* (0.74)
R_0^2: *if RDI is low, then RCON is low* (1.0)
R_1^2: *if RDI is average, then RCON is low* (0.08) *or average* (0.92)
R_2^2: *if RDI is high, then RCON is average* (0.24) *or high* (0.76)

Although these rules are not suitable for making exact forecasts. They enable experts in the application domain to understand relationships between variables, verify the trained model, and collaborate with machine learning engineers in model fusion.

Conclusion

We have presented an inductive method for constructing robust Bayesian Polynomial Regression models in Bernstein form. This method integrates the strengths of Bayesian inference, the support vector approach, and neurofuzzy modeling. The dual model conception—combining support vector expansion with Bernstein form – enables PRIAM to remain competitive with modern machine learning algorithms while also being suitable for knowledge representation in expert systems. The use of fuzzy rules with reduced complexity allows domain experts to contribute at various stages of the modeling process, including such tasks as prior setup, model validation, and knowledge extraction.

Our experiments on real-world economic datasets demonstrate that PRIAM outperforms many modern algorithms while adhering to a parsimonious model construction logic. Notably, bivariate dependencies appear only when the true underlying function (as observed in synthetic datasets) explicitly includes a product of endogenous factors.

References

1. E. T. Jaynes, *Probability Theory: The Logic of Science* (Cambridge : Cambridge University Press, 2003).
2. S. Gull, in: Maximum Entropy and Bayesian Methods, ed. by Erickson G. J., Smith C. R. (Dordrecht: Kluwer Academic, 1988), pp. 53–74.
3. D. J. C. Mackay, Neural Computations. **4**, 448–472 (1992).
4. C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning* (Cambridge, MA: MIT Press, 2006).
5. M Brown and C. J. Harris, *Neurofuzzy adaptive modelling and control* (Hemel Hempstead: Prentice Hall, 1994).
6. X. Hong and C. J. Harris, IEEE Trans. Neural Networks. **11** (4), 889–902 (2000).
7. V. N. Vapnik, *Statistical learning theory* (New York: John Wiley and Sons Inc., 1998).
8. C. Cortes and V. Vapnik, Machine Learning. **20**, 273–297 (1995).
9. V. Vapnik, S. Golowich, and A. Smola, Advances in Neural Information Processing Systems. **9**, 281–287 (1997).
10. W. Chu, S. Keerthi, and C. J. Ong, IEEE Trans. Neural Networks. **15** (1), 29–44 (2004).
11. M. Kuss and C. E. Rasmussen, Journal of Machine Learning Research. **6**, 1679–1704 (2005).
12. O. Y. Mytnyk and P. I. Bidyuk, System Research and Information Technologies. **2**, 24–34 (2004).
13. O. Y. Mytnik, Cybernetics and Sys. Anal. **43** (4), 613–620 (2007).
14. T. F. Coleman and Y. Li, Mathematical Programming. **67** (2), 189–224 (1994).
15. J. B. Gao, S. R. Gunn, C. J. Harris, and M. Brown, Machine Learning. **46** (1–3), 71–89 (2002).
16. NIST Standard Reference Database 140, https://www.itl.nist.gov/div898/strd.
17. UCI Repository of machine learning databases, https://archive.ics.uci.edu/datasets.
18. Y. Gorodnichenko, *Effects of intergovernmental aid on fiscal behavior of local governments: the case of Ukraine : EERC MA thesis* (NaUKMA, 2001).
19. O. Y. Mytnyk, in: Proceedings of 2nd International Conference on Inductive Modelling, 15–19 Sept. 2008 (Kyiv, 2008), pp. 148–152.

Митник О. Ю.

РОБАСТНА МОДЕЛЬ БАЄСІВСЬКОЇ РЕГРЕСІЇ У ФОРМІ БЕРНШТЕЙНА

Тут представлений індуктивний метод побудови робастних моделей баєсівської поліноміальної регресії (БПР) у формі Бернштейна, що отримав назву ПРІАМ. ПРІАМ – це алгоритм, призначений для визначення стохастичної залежності між змінними. Трикомпонентна природа ПРІАМ поєднує переваги баєсівського висновку, прозорість та лінгвістичну інтерпретовність нейронечітких моделей у формі Бернштейна, робастність методу опорних векторів.

Алгоритм апробовано на відомих штучних наборах даних, а також на реальних моделях різного розміру та рівня зашумленості. Складено рейтинг, який демонструє переваги запропонованого алгоритму за більшістю метрик.

Ключові слова: ПРІАМ, баєсівський висновок, БПР, нейронечітка модель, поліноми в формі Бернштейна.

Матеріал надійшов 27.12.2024

УДК 532.5

O. Avramenko

DOI: 10.18523/2617-70807202451-56

DEVIATION OF THE INTERFACE BETWEEN TWO LIQUID HALF-SPACES WITH SURFACE TENSION: MULTISCALE APPROACH

This paper investigates the deviation of the interface between two semi-infinite liquid media under the influence of surface tension and gravity using a multiscale analysis. The initial-boundary value problem is formulated based on key dimensionless parameters, such as the density ratio and the surface tension coefficient, to describe the generation and propagation of wave packets along the interface. A weakly nonlinear model is employed to examine initial deviations of the interface, enabling the derivation of integral solutions for both linear and nonlinear approximations. The linear approximation captures the fundamental structure of forward and backward waves, while nonlinear corrections account for higher-order effects derived through multiscale expansions. These corrections describe the evolution of the wave packet envelope, highlighting the interplay between dispersion, nonlinearity, and surface tension. Integral expressions are provided for both linear and nonlinear solutions, including those illustrating the role of even and odd initial deviations of the interface. Comparisons between linear and nonlinear approximations emphasize their interconnectedness. The linear model defines the primary wave dynamics, while the nonlinear terms contribute higher harmonics, refining the solutions and facilitating stability analysis. The results reveal significant contributions from higher-order harmonics in determining the dynamics of the interface. Furthermore, the study explores the conditions under which the nonlinear envelope remains stable, including constraints on initial amplitudes to prevent instability. This research opens new perspectives for further analysis of stability and wave dynamics at fluid interfaces using symbolic computations. Potential applications include the study of wave behavior under various geometric configurations and fluid properties. The findings contribute to advancing hydrodynamic wave modeling and establish a foundation for future research in this field.

Keywords: internal waves, initial-boundary value problem, multiscale expansions, surface tension.

Introduction

The study of wave packets along the interface of two semi-infinite fluids forms the basis for solving initial-boundary value problems (IBVPs) related to the generation and evolution of internal waves. These include the transmissibility of wave harmonics and modulational stability, or the so-called Benjamin–Feir instability [1].

Benjamin-Feir instability in hydrodynamics has been widely analyzed, focusing on stabilization mechanisms and extreme wave formation. Segur et al. [2] demonstrated dissipation stabilizes instability for waves with narrow bandwidth, confirmed experimentally; Wu [3] supported this via simulations, while Onorato et al. [4] linked the Benjamin-Feir index to extreme wave probability. Zakharov and Ostrovsky [5] explored nonlinear effects from modulation instability, and El and Hoefer [6] reviewed dispersive shock waves. Armaroli et al. [7] validated wave stabilization under wind-viscosity balance through experiments.

It should be noted that wave propagation in layered fluids has been effectively studied using multiscale methods. Here, we will mention only a few studies in this field. Nayfeh [8] derived an envelope evolution equation (NLS) for waves on fluid interfaces with surface tension. Grimshaw and Pullin [9] examined modulational stability of finite-amplitude interfacial waves, while Selezov et al. [10] investigated nonlinear wave-packet propagation using higher-order multiscale expansions.

This work extends the IBVP for the deviation of the contact surface between two semi-infinite fluids under surface tension, incorporating nonlinear effects and advancing understanding of interfacial wave dynamics.

Statement of the IBVP

Problem statement. This paper investigates the IBVP based on the solutions of problem [8] concerning traveling wave packets of dispersive nature. The following parameters were introduced as the basis for dimensionless quantities: the acceleration due to gravity g, the density ρ_1, and the characteristic surface tension T_0.

The problem of wave packet propagation along

the interface $z = \eta(x,t)$ between two fluids of different densities was addressed, with the effects of surface tension T is taken into consideration

$$\begin{aligned}
&\Delta\phi_j = 0 \quad \text{in} \quad \Omega_j, \qquad (1)\\
&\eta_{,t} - \phi_{j,z} = -\alpha\eta_{,x}\phi_{j,x} \quad \text{at} \quad z = \alpha\eta(x,t),\\
&\phi_{1,t} - \rho\phi_{2,t} + (1-\rho)\eta + 0.5\alpha\left(\nabla\phi_1\right)^2 - 0.5\alpha\rho\left(\nabla\phi_2\right)^2\\
&\quad - T\left(1 + (\alpha\eta_{,x})^2\right)^{-3/2}\eta_{,xx} = 0 \quad \text{at } z = \alpha\eta(x,t),\\
&|\nabla\phi_1| \to 0 \quad \text{at} \quad z \to \pm\infty,
\end{aligned}$$

where $\Omega_1 = \{(x,z) : |x| < +\infty, -\infty < z < 0\}$, $\Omega_2 = \{(x,z) : |x| < +\infty, 0 < z < +\infty\}$, $\rho = \rho_2/\rho_1$, $\rho_i (i = 1,2)$ are the densities of fluids in Ω_i, $\alpha = a/l$ is a small parameter characterizing the steepness of the wave, a is the maximum deviation of the contact surface $\eta(x,t)$, and l is the wavelength.

Let the initial condition at $z = 0$ be given as a deviation $F(x)$ of the interface

$$\eta(x,0) = F(x). \qquad (2)$$

Preliminary results on traveling wave packets. The result presented in this study is based on previously obtained findings for traveling wave packets derived using the method of multiscale expansions [8]. The results from the aforementioned study, essential for solving the IBVP (1)-(2), are presented below.

According to the method of multiple-scale expansions, the deviation of the interface is represented as a sum of the first harmonics

$$\begin{aligned}
\eta(x,t) =& \eta_1(x_0, x_1, x_2, t_0, t_1, t_2) \qquad (3)\\
&+ \alpha\eta_2(x_0, x_1, x_2, t_0, t_1, t_2) + \mathrm{O}(\alpha^2),
\end{aligned}$$

where $x_n = \alpha^n x$, $t_n = \alpha^n t$ are the spatial and temporal scaling variables.

In the first approximation, the deviation of the contact surface caused by a forward wave η_1^+ is expressed as the sum of the product of the complex envelope $A(x_1, x_2, t_1, t_2)$ and the carrier forward wave $\exp \mathrm{i}(kx_0 - \omega t_0)$ and the product of their conjugates,

$$\eta_1^+ = A\exp(\mathrm{i}(kx_0 - \omega t_0)) + \overline{A}\exp(-\mathrm{i}(kx_0 - \omega t_0)) \qquad (4)$$

where, in the linear approximation, the envelope is considered a constant value, as it cannot depend on higher-order scales.

The solvability of the linear approximation problem determines the dispersion relation, which links ω and k; let its two solutions be denoted as $\omega_{1,2} = \pm\omega(k)$.

Here, we present the expression for the second forward harmonic, derived in [8] from the linear problem of the second approximation, in the form

$$\begin{aligned}
\eta_2^+ =& \Lambda(k,\omega)A^2\exp(2\mathrm{i}(kx_0 - \omega t_0)) \qquad (5)\\
&+ \Lambda(k,\omega)\overline{A}^2\exp(-2\mathrm{i}(kx_0 - \omega t_0))
\end{aligned}$$

where the coefficient $\Lambda(k,\omega)$ satisfies the condition $\Lambda(k,\omega) = \Lambda(k,-\omega)$.

In [8], it is also shown that the envelope A satisfies an evolution equation in the form of a NLS

$$A_{,\zeta} - 0.5\mathrm{i}\omega'' A_{,\xi\xi} = 4\mathrm{i}\alpha^2\omega^{-1}JA^2\overline{A}, \qquad (6)$$

where $\xi = x - \omega' t$ and $\zeta = t$, $J(k,\omega)$ is the Benjamin-Feir index which in this system satisfies the condition $J(k,\omega) = J(k,-\omega)$.

The IBVP solution

Linear approximation. Since the hydrodynamic system allows the propagation of only certain types of harmonics (4) and (5), with the frequency ω and wavenumber k linked by the dispersion relation, and the envelope A governed by the evolution equation (6), the problem arises of correctly specifying the initial shape $F(x)$ of the contact surface η within the framework of the weakly nonlinear model (1)-(2). Let the initial position of the contact surface $\eta(x,0)$ in the linear approximation take the form of a certain function $f(x)$

$$\eta_{lin}(x,0) = f(x). \qquad (7)$$

On the one hand, the function $f(x)$ can be represented as an integral using the Fourier expansion over the frequency spectrum, followed by synthesis based on this spectrum

$$\begin{aligned}
&f(x) = \qquad (8)\\
&\Re\left[\int_{-\infty}^{+\infty}\left(\frac{1}{2\pi}\int_{-\infty}^{+\infty} f(\xi)\exp(-\mathrm{i}k\xi)d\xi\right)\exp(\mathrm{i}kx)dk\right]
\end{aligned}$$

Considering equations (7) and (8), we have

$$\begin{aligned}
&\eta_{lin}(x,0) = \qquad (9)\\
&\int_{-\infty}^{+\infty}\Big(a_f(k)\cos(kx) - b_f(k)\sin(kx)\Big)dk,
\end{aligned}$$

where

$$a_f(k) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} f(\xi)\cos k\xi d\xi, \qquad (10)$$

$$b_f(k) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} f(\xi)\sin k\xi d\xi. \qquad (11)$$

On the other hand, taking into account the dispersion relation solution $\omega_1 = +\omega(k)$, in the linear

approximation, the oscillation of the contact surface can be represented as an integral over the wave numbers of the forward wave (4)

$$\eta^{+}_{lin}(x,t) = \int_{-\infty}^{+\infty} \Big(a_{lin}(k) \exp(\mathrm{i}(kx - \omega(k)t)) + \overline{a_{lin}}(k) \exp(-\mathrm{i}(kx - \omega(k)t)) \Big) dk, \quad (12)$$

and for the dispersion relation solution $\omega_2 = -\omega(k)$ corresponding to the backward wave

$$\eta^{-}_{lin}(x,t) = \int_{-\infty}^{+\infty} \Big(a_{lin}(k) \exp(\mathrm{i}(kx + \omega(k)t)) + \overline{a_{lin}}(k) \exp(-\mathrm{i}(kx + \omega(k)t)) \Big) dk \quad (13)$$

where $a_{\mathrm{lin}}(k)$ are the unknown coefficients of the linear approximation expansion of the interface deviation, which coincide for the forward $\eta^{+}_{lin}(x,t)$ and backward $\eta^{-}_{lin}(x,t)$ waves due to the homogeneity of liquid media in both directions of wave propagation.

It is obvious that

$$\eta_{lin}(x,t) = \eta^{+}_{lin}(x,t) + \eta^{-}_{lin}(x,t). \quad (14)$$

Next, we obtain $\eta_{lin}(x,0)$ from (14) taking into account (12) and (13)

$$\eta_{lin}(x,0) = 2 \int_{-\infty}^{+\infty} \Big(a_{lin}(k) \exp(\mathrm{i}kx) + \overline{a_{lin}}(k) \exp(-\mathrm{i}kx) \Big) dk. \quad (15)$$

Equating the expressions for the initial deviation of the contact surface $\eta_{lin}(x,0)$ from (9) and (15), we obtain

$$a_{lin}(k) = \frac{1}{4}(a_f(k) + \mathrm{i}b_f(k)). \quad (16)$$

Substituting formulas (10) and (11) into (16) we obtain

$$a_{lin}(k) = \frac{1}{8\pi} \int_{-\infty}^{+\infty} f(\xi) \exp(\mathrm{i}k\xi) d\xi, \quad (17)$$

and substituting (17) into (12)-(14) gives the linear approximation of the contact surface deviation in the following integral form

$$\eta_{lin}(x,t) = \quad (18)$$
$$\frac{1}{8\pi} \int_{-\infty}^{+\infty} \Big[\int_{-\infty}^{+\infty} f(\xi) \exp(\mathrm{i}k\xi) d\xi \exp \mathrm{i}(kx - \omega(k)t) + \int_{-\infty}^{+\infty} f(\xi) \exp(-\mathrm{i}k\xi) d\xi \exp(-\mathrm{i}(kx - \omega(k)t)) \Big] dk.$$
$$+ \frac{1}{8\pi} \int_{-\infty}^{+\infty} \Big[\int_{-\infty}^{+\infty} f(\xi) \exp(\mathrm{i}k\xi) d\xi \exp \mathrm{i}(kx + \omega(k)t) + \int_{-\infty}^{+\infty} f(\xi) \exp(-\mathrm{i}k\xi) d\xi \exp(-\mathrm{i}(kx + \omega(k)t)) \Big] dk.$$

or taking into account the formulae (10) and (11)

$$\eta_{lin}(x,t) = \eta^{+}_{lin}(x,t) + \eta^{-}_{lin}(x,t) \quad (19)$$

where

$$\eta^{\pm}_{lin}(x,t) = \frac{1}{2} \int_{-\infty}^{+\infty} \Big(a_f(k) \cos(kx \mp \omega(k)t) - b_f(k) \sin(kx \mp \omega(k)t) \Big) dk. \quad (20)$$

Nonlinear approximation. Let us proceed to derive the nonlinear approximation of the contact surface deviation. To this end, we consider one of the solutions to the evolution equation (6) and write it for both solutions of the dispersion equation

$$A_{\pm}(t,k) = \frac{1}{2} a \exp\left(\pm \mathrm{i}a^2 \frac{J(k, \pm\omega(k))}{\omega(k)} t \right), \quad (21)$$

where a is an arbitrary constant determining the amplitude of the envelope, $A_{+}(t,k)$ corresponds to the forward wave with frequency $\omega_1 = +\omega(k)$, and $A_{-}(t,k)$ to the backward wave with frequency $\omega_2 = -\omega(k)$.

Then, for wave packets traveling along the contact surface, taking into account (4) and (5) and also the fact that $J(k,\omega) = J(k,-\omega)$ and $\Lambda(k,\omega) = \Lambda(k,-\omega)$, the contact surface deviation caused by the forward $\eta^{+}_{nl}(x,t,k)$ and backward $\eta^{-}_{nl}(x,t,k)$ waves at the wave number k are expressed as the sum of the harmonics

$$\eta^{\pm}_{nl}(x,t,k) = \quad (22)$$
$$A_{\pm}(t,k) \exp(\mathrm{i}(kx \mp \omega(k)t)) + \overline{A_{\pm}}(t,k) \exp(-\mathrm{i}(kx \mp \omega(k)t)) + \alpha\Lambda(k,\omega(k)) \Big(A^2_{\pm}(t,k) \exp(2\mathrm{i}(kx \mp \omega(k)t)) + \overline{A_{+}}^2(t,k) \exp(-2\mathrm{i}(kx \mp \omega(k)t)) \Big) + \mathrm{O}(\alpha^2).$$

Assume that the coefficient $\frac{1}{2}a$ in the expressions (21) for the envelopes $A_{\pm}(t,k)$ is equal to the complex coefficient $a_{\mathrm{lin}}(k)$ and taking into account (16), it can be expressed in terms of the coefficients $a_f(k)$ and $b_f(k)$ of the function $f(x)$ expansion

$$a = \frac{1}{2} \Big(a_f(k) + \mathrm{i}b_f(k) \Big). \quad (23)$$

Let us substitute the expressions for the envelope (21) into (22) taking into account (23). After

transformations, we obtain expressions for the nonlinear approximation of the forward $\eta^{+}_{nl}(x,t,k)$ and backward $\eta^{-}_{nl}(x,t,k)$ traveling waves in the form of real-valued expressions

$$\eta^{\pm}_{nl}(x,t,k) = \tag{24}$$
$$\frac{1}{2}\big[a_f(k)\cos(kx\mp\hat{\omega}t) - b_f(k)\sin(kx\mp\hat{\omega}t)\big]$$
$$+\frac{\alpha}{8}\Lambda(k,\omega(k))\Big(\big(a_f^2(k)-b_f^2(k)\big)\cos 2(kx\mp\hat{\omega}(k)t)$$
$$-2a_f(k)b_f(k)\sin 2(kx\mp\hat{\omega}(k)t)\Big)+\mathrm{O}(\alpha^2),$$

where

$$\hat{\omega}(k) = \omega(k) - a^2\frac{J(k,\omega(k))}{\omega(k)}. \tag{25}$$

Performing the synthesis of the traveling waves (24) by the spectrum of wave numbers, we obtain expressions for the nonlinear approximation of the contact surface deviation $\eta(x,t)$ in the form

$$\eta(x,t) = \eta^{+}_1(x,t) + \eta^{-}_1(x,t) \tag{26}$$
$$+\alpha\big(\eta^{+}_2(x,t)+\eta^{-}_2(x,t)\big)+\mathrm{O}(\alpha^2),$$

where

$$\eta^{\pm}_1(x,t) = \frac{1}{2}\int_{-\infty}^{+\infty}\Big(a_f(k)\cos(kx\mp\hat{\omega}t) \tag{27}$$
$$-b_f(k)\sin(kx\mp\hat{\omega}t)\Big)dk,$$

$$\eta^{\pm}_2(x,t) = \tag{28}$$
$$\frac{1}{8}\int_{-\infty}^{+\infty}\Lambda(k,\omega(k))\Big(\big(a_f^2(k)-b_f^2(k)\big)\cos 2(kx\mp\hat{\omega}(k)t)$$
$$-2a_f(k)b_f(k)\sin 2(kx\mp\hat{\omega}(k)t)\Big)dk.$$

The expression (26), obtained for the interface deviation $\eta(x,t)$, contains the nonlinear contribution (27) of the first harmonic $\eta^{\pm}_1(x,t)$, which differs from its linear approximation $\eta^{\pm}_{lin}(x,t)$ in (20). It additionally includes (28) the contribution of the second harmonic $\eta^{\pm}_2(x,t)$. Let us introduce the terms

$$\eta_1(x,t) = \eta^{+}_1(x,t) + \eta^{-}_1(x,t),$$
$$\eta_2(x,t) = \eta^{+}_2(x,t) + \eta^{-}_2(x,t)$$

denoting the contributions of the first and second harmonics to the nonlinear solution.

It should be noted that within the framework of the nonlinear model, the synthesis operation over the spectrum of traveling waves, namely, the first and second harmonics, is mathematically valid, since each of the traveling harmonic waves represents a solution to the linear approximations of the problem at the corresponding order.

Let us now return to the question of the form of the initial contact surface deviation $F(x)$, which in the linear approximation we defined in (7) as some function $f(x)$. From (26)-(28), it is straightforward to obtain

$$F(x) \equiv \eta(x,0) = \tag{29}$$
$$\int_{-\infty}^{+\infty}\Big[a_f(k)\cos(kx) - b_f(k)\sin(kx)$$
$$+\frac{\alpha}{4}\Lambda(k,\omega(k))\Big(\big(a_f^2(k)-b_f^2(k)\big)\cos 2(kx)$$
$$-2a_f(k)b_f(k)\sin 2(kx)\Big)\Big]dk+\mathrm{O}(\alpha^2).$$

It is evident that

$$F(x) = f(x) + \mathrm{O}(\alpha),$$

i.e., the refined initial contact surface deviation differs from the specified initial deviation in the linear approximation by a small quantity.

Let us consider the stability conditions for the envelope in the form of the solution (21) discussed here. The frequency (25) in the nonlinear approximation (24) taking into account (23) is in the form

$$\hat{\omega}(k) = \omega(k) \tag{30}$$
$$-\frac{1}{4}\Big(a_f^2(k)-b_f^2(k)+2\mathrm{i}a_f(k)b_f(k)\Big)\frac{J(k,\omega(k))}{\omega(k)}.$$

Expression (30) imposes constraints on the envelope amplitude where we observe that an imaginary term appears in the exponent. The presence of this term leads to instability. This can be avoided by setting to zero either the imaginary $b_f(k)$ or real $a_f(k)$ part of a, which can be easily achieved by using an even or odd function $f(x)$, respectively.

Special cases. It is evident that in the case of an even function $f(x)$ (below, the index 'ev' indicates the values corresponding to this case), we can transition to simpler expressions with integrals over $(0,+\infty)$

$$a_f^{ev}(k) = \frac{1}{\pi}\int_0^{+\infty} f(\xi)\cos k\xi d\xi, \quad b_f^{ev}(k)\equiv 0,$$

in the linear approximation

$$a_{lin}^{ev}(k) = \frac{1}{4\pi}\int_0^{+\infty} f(\xi)\cos k\xi d\xi,$$
$$\eta_{lin}^{ev}(x,t) = \int_0^{+\infty} a_f^{ev}(k)\times$$
$$\times\Big(\cos(kx-\omega(k)t)+\cos(kx+\omega(k)t)\Big)dk,$$

and in the nonlinear approximation

$$\eta^{ev}(x,t) = \eta_1^{ev}(x,t) + \eta_2^{ev}(x,t),$$

where

$$\eta_1^{ev}(x,t) = \int_0^{+\infty} a_f^{ev}(k)\times$$
$$\times\Big(\cos(kx-\hat{\omega}^{ev}(k)t)+\cos(kx+\hat{\omega}^{ev}(k)t)\Big)dk,$$
$$\eta_2^{ev}(x,t) = \frac{1}{4}\int_0^{+\infty} \Lambda(k,\omega(k))(a_f^{ev})^2(k)\times$$
$$\times\Big(\cos 2(kx-\hat{\omega}^{ev}(k)t)+\cos 2(kx+\hat{\omega}^{ev}(k)t)\Big)dk,$$
$$\hat{\omega}^{ev}(k)=\omega(k)-(a^{ev})^2\frac{J(k,\omega(k))}{\omega(k)},\ a^{ev}=\frac{a_f^{ev}(k)}{2},$$
$$A_{\pm}^{ev}(t,k) = \frac{1}{2}a^{ev}\exp\left(\pm\mathrm{i}\big(a^{ev}\big)^2\frac{J(k,\pm\omega(k))}{\omega(k)}t\right).$$

A similar result can be easily obtained for another specific case when the function $f(x)$ is odd.

Conclusion and further developments

Linear and weakly nonlinear integral expressions for waves propagating between two liquid half-spaces, induced by the initial deviation of the contact surface, have been derived. A limitation on the initial position of the contact surface deviation within the model has been noted, stemming from the characteristics of the solution obtained using the method of multiscale expansions. The prospect of this study lies in the potential to obtain solutions for various geometric and physical properties using symbolic and numerical computation methods. In particular, based on the solutions presented here and the Benjamin-Feir stability condition, it will be possible to derive the conditions for the emergence of rogue waves.

Acknowledgments. The author expresses gratitude to the Research Council of Lithuania for the support in preparing this article.

References

1. T. B. Benjamin and J. E. Feir, Journal of Fluid Mechanics. **27** (3), 417–430 (1967).
2. H. Segur, D. Henderson, J. Carter, J. Hammack, C. Li, D. Pheiff, and K. Socha, Journal of Fluid Mechanics. **539**, 229–271 (2005).
3. G. Wu, Y. Liu, and D. K. P. Yue, Journal of Fluid Mechanics. **556**, 45–54 (2006).
4. M. Onorato, A. R. Osborne, M. Serio, L. Cavaleri, C. Brandini, and C. T. Stansberg, European Journal of Mechanics - B/Fluids. **25** (5), 586–601 (2006).
5. V. E. Zakharov and L. A. Ostrovsky, Physica D: Nonlinear Phenomena. **238** (5), 540–548 (2009).
6. G. A. El and M. A. Hoefer, Physica D: Nonlinear Phenomena. **333**, 11–65 (2016).
7. A. Armaroli, D. Eeltink, M. Brunetti, and J. Kasparian. Physics of Fluids. **30** (1), 017102 (2018).
8. A. Nayfeh, Trans. ASME, Ser. E: J. Appl. Mech. **43** (4), 584–588 (1976).
9. R. H. J. Grimshaw and D. I. Pullin, J. Fluid Mech. **160**, 297–315 (1985).
10. . I. Selezov, O. Avramenko, C. Kharif, and K. Trulsen. Comptes Rendus. Mecanique. **331** (3), 197–201 (2003).

Авраменко О. В.

ВІДХИЛЕННЯ ПОВЕРХНІ КОНТАКТУ ДВОХ РІДКИХ НАПІВПРОСТОРІВ З ПОВЕРХНЕВИМ НАТЯГОМ: БАГАТОМАСШТАБНИЙ ПІДХІД

Ця стаття присвячена дослідженню відхилення поверхні контакту між двома напівнескінченними рідинами під впливом сил поверхневого натягу та гравітації з використанням багатомасштабного аналізу. Початково-крайова задача базується на ключових безрозмірних параметрах, зокрема на відношенні густин і коефіцієнті поверхневого натягу, для опису генерації та поширення хвильових пакетів уздовж поверхні контакту. За допомогою слабко нелінійної моделі досліджують початкові відхилення поверхні контакту, що дозволяє отримати інтегральні розв'язки для як лінійного, так і нелінійного наближень. Лінійне наближення описує основну структуру прямої та зворотної хвиль, тоді як нелінійні поправки враховують ефекти вищого порядку, які виводяться за допомогою багатомасштабних розкладів. Ці поправки характеризують еволюцію обвідної хвильового пакета, виявляючи взаємодію між дисперсією, нелінійністю та поверхневим натягом. Надаються інтегральні вирази для лінійних і нелінійних розв'язків, зокрема таких, що демонструють роль парних і непарних початкових відхилень поверхні контакту. Порівняння між лінійним і нелінійним наближеннями підкреслюють їх взаємозв'язок. Лінійна модель встановлює основну динаміку хвиль, тоді як нелінійні члени додають гармоніки вищого порядку, уточнюючи розв'язки і дозволяючи проводити аналіз стійкості. Ці результати виявляють суттєві внески від гармонік вищого

порядку у визначення динаміки поверхні контакту. Крім того, у дослідженні розглянуто умови, за яких нелінійна обвідна залишається стійкою, зокрема обмеження на початкові амплітуди, щоб запобігти виникненню нестійкості. Дослідження відкриває нові перспективи для подальшого аналізу стійкості та динаміки хвиль на межі поділу рідин за допомогою символьних обчислень. Потенційні застосування передбачають подальше вивчення поведінки хвиль за різних геометричних параметрів системи та властивостей рідин. Отримані результати сприяють розвитку моделювання гідродинамічних хвиль і закладають основу для подальших досліджень у цій галузі.

Ключові слова: внутрішні хвилі, початково-крайова задача, багатомасштабні розвинення, поверхневий натяг.

Матеріал надійшов 25.11.2024

УДК 532.5

Нарадовий В. В.

DOI: 10.18523/2617-70807202457-60

НЕЛІНІЙНЕ ПОШИРЕННЯ ХВИЛЬОВИХ ПАКЕТІВ ПРИ ХВИЛЬОВИХ ЧИСЛАХ, БЛИЗЬКИХ ДО КРИТИЧНОГО, В ДВОШАРОВІЙ ГІДРОДИНАМІЧНІЙ СИСТЕМІ СКІНЧЕННОЇ ГЛИБИНИ

Розглянуто задачу про поширення слабконелінійних хвильових пакетів у двошаровій гідродинамічній системі «шар з твердим дном — шар з кришкою». Для дослідження та аналізу використано метод багатомасштабних розвинень (МБР) до третього порядку, що дає можливість отримати перші наближення досліджуваної моделі, які є лінійними відносно невдомих функцій, що є доданками у відповідних розкладах. У результаті вдається отримати еволюційне рівняння обвідної хвильових пакетів у формі нелінійного рівняння Шредінгера. Коли частота центру хвильового пакету близька до нуля, отримані з МБР результати не можуть бути використані для моделювання хвильових рухів у досліджуваній системі. В статті розглянуто граничний випадок поширення хвильових пакетів при навколокритичних хвильових числах. На основі дисперсійного співвідношення та умов розв'язуваності другого та третього наближень встановлено, що поширення хвильових пакетів при хвильових числах, близьких до критичного, описується нелінійним рівнянням Шредінгера. Отримане рівняння містить першу похідну за просторовою координатою та дві похідні за часовою координатою і може бути поширеним на всі хвильові числа. Також виведено співвідношення між хвильовим числом та малим параметром.

Ключові слова: внутрішні хвилі, нелінійне рівняння Шредінгера, хвильове число.

Вступ

Використання асимптотичних методів до задач дослідження поширення хвильових пакетів є одним із найпоширеніших способів математичного моделювання хвильових рухів у рідинах. Зокрема, застосування методу багатомасштабних розвинень (МБР) до слабконелінійних задач такого класу призводить до аналізу послідовних наближень, які є лінійними відносно невідомих функцій — доданків відповідних асимптотичних розкладів.

У дослідженні [1] проведено аналітичне моделювання поширення хвильових пакетів у двошаровій гідродинамічній системі «півпростір — півпростір». За допомогою МБР отримано перші три наближення. Виведено дисперсійне співвідношення та умови розв'язуваності другого та третього наближень. Отримано еволюційне рівняння обвідної хвильових пакетів у вигляді нелінійного рівняння Шредінгера (НРШ). Проведено аналіз модуляційної стійкості хвильових пакетів.

Роботи [2] та [3] присвячені аналізу поширення слабконелінійних хвильових пакетів у гідродинамічній системі «півпростір — шар з твердою кришкою». Зокрема, отримано еволюційне рівняння обвідної, проаналізовано форму хвильових пакетів, досліджено питання модуляційної стійкості.

У статтях [5] та [6] досліджено математичну модель хвильових рухів у гідродинамічній системі «шар з твердим дном — шар з кришкою». Отримано рівняння для обвідної хвильового пакета у формі НРШ. Проаналізовано характерні особливості поширення хвильових пакетів.

З використанням МБР у роботах [7], [8] та [9] проаналізовано питання поширення слабконелінійних хвильових пакетів у гідродинамічних системах «шар з твердим дном — шар — шар з вільною поверхнею», «півпростір — шар — шар з твердою кришкою», «шар з твердим дном — шар — шар з твердою кришкою». Отримано еволюційні рівняння обвідної хвильових пакетів у формі НРШ, проведено аналіз характеристик поширення хвиль. У вказаних роботах було проведено детальний аналіз хвильових рухів. Зазначимо, що у випадку малих частот, коли $\omega \to 0$, результати, отримані у вказаних роботах, не можуть бути застосовані для математичного моделювання хвильових рухів. Цей випадок розглянуто у [1] для моделі «півпростір — півпростір» та у [4] для моделі «півпростір — шар з твердою кришкою».

У запропонованому дослідженні ми розглянемо поширення хвильових пакетів при значенні хвильових чисел, близьких до критичного, для гідродинамічної системи «шар з твердим

дном — шар з кришкою».

Поширення хвильових пакетів у двошаровій гідродинамічній системі скінченної глибини

Розглянемо задачу про поширення двовимірних хвильових пакетів кінцевої амплітуди на поверхні контакту $\eta(x,t)$ рідких шарів $\Omega_1 = \{(x,z) : |x| < +\infty, -h_1 < z < 0\}$ та $\Omega_2 = \{(x,z) : |x| < +\infty, 0 < z < h_2\}$. Враховується сила поверхневого натягу T, сила тяжіння направлена перпендикулярно до поверхні розподілу у від'ємному z-напрямку, рідини вважаються нестисливими. Швидкості у Ω_i виражені через градієнти потенціалів $\phi_i(x,z,t)$. Математична постановка задачі в безрозмірному вигляді:

$$\begin{aligned}
&\Delta\phi_j = 0 \quad \text{in} \quad \Omega_j, \qquad (1)\\
&\eta_{,t} - \phi_{j,z} = -\alpha\eta_{,x}\phi_{j,x} \quad \text{при} \quad z = \alpha\eta(x,t),\\
&\phi_{1,t} - \rho\phi_{2,t} + (1-\rho)\eta + 0.5\alpha\left(\nabla\phi_1\right)^2 - 0.5\alpha\rho\left(\nabla\phi_2\right)^2\\
&\quad - T\left(1+(\alpha\eta_{,x})^2\right)^{-3/2}\eta_{,xx} = 0 \quad \text{при}\, z = \alpha\eta(x,t),\\
&\phi_{1,z} = 0 \quad \text{при} \quad z = -h_1,\\
&\phi_{2,z} = 0 \quad \text{при} \quad z = h_2,
\end{aligned}$$

де $\rho = \rho_2/\rho_1$, $\rho_i (i = 1,2)$ — густини шарів Ω_i, $\alpha = a/l$ — параметр нелінійності, a — максимальне відхилення поверхні контакту $\eta(x,t)$, l — довжина хвиль.

Розв'язки задачі (1) можна шукати, використовуючи метод багатомасштабних розвинень. Представимо невідомі функції $\eta(x,t)$ та $\phi_i(x,z,t)$ у вигляді

$$\begin{aligned}
\eta(x,t) &= \sum_{n=1}^{3}\alpha^{n-1}\eta_n + \mathrm{O}(\alpha^3), \qquad (2)\\
\phi_j(x,z,t) &= \sum_{n=1}^{3}\alpha^{n-1}\phi_{jn} + \mathrm{O}(\alpha^3),
\end{aligned}$$

де $\eta_n = \eta_n(x_0, x_1, x_2, t_0, t_1, t_2)$ та $\phi_{jn} = \phi_{jn}(x_0, x_1, x_2, z, t_0, t_1, t_2)$ — відповідні доданки в асимптотичних розкладах ($n = 1,2,3$), $x_j = \alpha^j x$ та $t_j = \alpha^j$ — масштабні змінні ($j = 0,1,2$).

Підставивши (2) у задачу (1), отримаємо перші три лінійні наближення відносно невідомих функцій η_n та ϕ_{jn}. Ці проблеми мають складний аналітичний вигляд, тому тут їх наводити не будемо. Зазначимо, що в результаті аналізу вказаних наближень було знайдено дисперсійне співвідношення та умови розв'язуваності другого та третього наближень, на основі яких можна отримати еволюційне рівняння обвідної у вигляді НРШ.

У роботі [5] було отримано дисперсійне співвідношення

$$\omega^2 = \frac{k\rho k + Tk^3}{\coth(kh_1) + \rho\coth(kh_2)}, \qquad (3)$$

де ω — частота центру хвильового пакета, k — хвильове число. Умову розв'язуваності для другого наближення можна записати у вигляді

$$W_{11}A_{,t_1} + W_{12}A_{,x_1} = 0, \qquad (4)$$

де A — обвідна хвильового пакета, W_{11} та W_{12} — коефіцієнти, які мають громіздкий аналітичний вигляд та залежать від параметрів (h_1, h_2, T, ρ, k). Умову розв'язуваності для третього наближення можна записати у вигляді

$$\begin{aligned}
&W_{21}A_{,t_2} + W_{22}A_{,t_2} + W_{23}A_{,t_1x_1} +\\
&W_{24}A_{,x_1x_1} + W_{25}A_{,t_1t_1} = W_{26}A^2\overline{A} \qquad (5)
\end{aligned}$$

де W_{2i} $(i = 1,6)$ — коефіцієнти, які мають громіздкий аналітичний вигляд та залежать від параметрів (h_1, h_2, T, ρ, k).

Зазначимо, що у випадку $\omega \to 0$, коли $\omega' \to \infty$, розвинення, що отримані у [5; 6] для випадку широкого спектра частот, не можуть бути застосовані. Для розгляду цього граничного випадку перепишемо умову (4) у вигляді

$$k'A_{,t_1} + A_{,x_1} = 0, \qquad (6)$$

де $k' = dk/d\omega$, отримана з (3) і має вигляд

$$\begin{aligned}
&k' = 2\omega\left(\coth(kh_1) + \rho\coth(kh_2)\right)^2 \times\\
&\{Tk^3[h_1(\coth^2(kh_1) - 1) + \rho h_2(\coth^2(kh_2) - 1)] +\\
&3Tk^2[\coth(kh_1) + \rho\coth(kh_2)] + k(1-\rho)[h_1 +\\
&+ \rho h_2 - h_1\coth^2(kh_1) - \rho h_2\coth^2(kh_2)] -\\
&-(1-\rho)[\coth(kh_1) - \rho\coth(kh_2)]\}^{-1}.
\end{aligned}$$

Продиференціюємо (6) спочатку за x_1, а потім за t_1. Отримаємо

$$k'A_{,t_1x_1} + A_{,x_1x_1} = 0, \quad k'A_{,t_1t_1} + A_{,x_1t_1}. \qquad (7)$$

Остаточно з (7) маємо

$$A_{,x_1x_1} = -k'A_{,t_1t_1}, \quad A_{,x_1x_1} = (k')^2A_{,t_1t_1}. \qquad (8)$$

Тоді, використовуючи (6), (8) та умову розв'язуваності (5), отримаємо шукане еволюційне рівняння у вигляді нелінійного рівняння Шредінгера

$$A_{,x} + k'A_{,t} - 0.5ik''A_{,tt} = 2i\alpha^2J_0A^2\overline{A}, \qquad (9)$$

де $k'' = d^2k/d\omega^2$. Вираз для k'' має складний аналітичний вигляд, тому тут його не наводимо. Рівняння (9) має розв'язок, який залежить лише від часу

$$A = 0.5a\exp(i\sigma t + cons`t), \qquad (10)$$

де a — деяка стала. Підставимо (10) у (9), отримаємо рівняння відносно σ:

$$0.5k''\sigma^{2^-}k'\sigma + 2\alpha^2a^2J_0 = 0. \qquad (11)$$

З (11) отримаємо

$$\sigma_{1,2} = \frac{k' \mp \sqrt{(k')^{2^-}2\alpha^2a^2J_0}}{k''}. \qquad (12)$$

Якщо $\omega \to 0$, тоді отримаємо, що

$$k' \to \frac{\omega(\coth(kh_1 + \rho\coth(kh_2)))}{Tk^2},$$

$$k'' \to \frac{(\coth(kh_1 + \rho\coth(kh_2)))}{Tk^2}, \qquad (13)$$

$$J_0 \to \frac{3k^3}{32}.$$

Враховуючи (13), вираз (11) для σ можна переписати у такому вигляді

$$\sigma = \omega \mp \left(\omega^{2^-}\frac{3\alpha^2a^2k^5T}{8(\coth(kh_1) + \rho coth(kh_2))}\right)^{0.5}.$$

Отже, вираз для σ набуває дійсних значень, якщо

$$\omega^2 \geq \frac{3\alpha^2a^2k^5T}{8(\coth(kh_1) + \rho coth(kh_2))}. \qquad (14)$$

З умови (14) та дисперсійного співвідношення (3) отримаємо зв'язок між хвильовим числом k та нелінійним параметром α, що відповідає критичній частоті:

$$3\alpha^2a^2k^{4^-}8k^{2^-}8\frac{1-\rho}{T} = 0,$$

звідки хвильове число визначається за формулою

$$k = \frac{4 \mp 4\sqrt{1 - \frac{3\alpha^2a^2(1-\rho)}{2T}}}{3\alpha^2a^2}. \qquad (15)$$

Отриманий результат (15) повністю узгоджується з дослідженнями, проведеними в [1] і [5]. Зазначимо, що з (15) можна отримати розвинення хвильового числа k за малим параметром α.

Висновки та перспективи подальших досліджень

У роботі розглянуто задачу про поширення слабконелінійних хвильових пакетів у двошаровій гідродинамічній системі «шар з твердим дном — шар з кришкою» з використанням методу багатомасштабних розвинень. Розглянуто граничний випадок поширення хвильових пакетів при навколокритичних хвильових числах. У цьому випадку, частота центру хвильового пакета близька до нуля, і розвинення, отримані методом багатьох масштабів, не можуть бути застосовані для моделювання хвильових рухів. В результаті встановлено, що поширення хвильових пакетів при хвильових числах, близьких до критичного, описується нелінійним рівнянням Шредінгера, що містить першу похідну за просторовою координатою та дві похідні за часовою координатою і яке може бути поширеним на всі хвильові числа. Отримано співвідношення між хвильовим числом та малим параметром. У подальшому планується дослідити поняття модуляційної стійкості (стійкості Бенджаміна — Фейра) для випадку хвильових чисел, близьких до критичного.

Список літератури

1. Nayfeh A. Nonlinear propagation of wave-packets on fluid interface. *Trans. ASME, Ser. E: J. Appl. Mech.* 1976. Vol. 43 (4). Pp. 584–588.
2. Selezov I., Avramenko O. Some features of nonlinear wave trains propagating in two-layer fluid. *Geophysical Research Abstracts.* 2001. Vol. 3. Pp. 25–30.
3. Selezov I., Avramenko O., Kharif C., and Trulsen K. High-order evolution equation for nonlinear wavepacket propagation with surface tension accounting. *Comptes Rendus. Mecanique.* 2003. Vol. 331 (3). Pp. 197–201.
4. Selezov I. T., Avramenko O. V. Nonlinear Propagation of Wave Packets for Near-Critical Wave Numbers in a Liquid that Is Piecewise Nonuniform with Depth. *Journal of Mathematical Sciences.* 2001. Vol. 103. Pp. 409–413.
5. Gurtovy Yu. V., Selezov I. T., Avramenko O. V. Features of wave-packet propagation in two-layer fluid of finite depth. *International Journal of Fluid Mechanics Research.* 2007. Vol. 34 (5). Pp. 475–491.
6. Гуртовий Ю. В., Селезов И. Т., Авраменко О. В. Нелінійна стійкість поширення хвильових пакетів в двошаровій рідині. *Прикладна гідромеханіка.* 2006. № 90 (8). С. 60-65.
7. Avramenko O. V., Naradovyi V. V., Selezov I. T. Conditions of wave propagation in a two-layer liquid with free surface. *Journal of Mathematical Sciences.* 2016. Vol. 212. Pp. 131–141.
8. Avramenko O., Lunyova M., Naradovyi V. Wave propagation in a three-layer semi-infinite hydrodynamic system with a rigid lid. *Eastern-European journal of enterprise technologies.* 2017. Vol. 5 (5). Pp. 58–66.
9. Naradovyi V. V., Kharchenko D. S. Modulation stability of wave-packets in a three-layer fluid. *Mathematical Modeling and Computing.* 2023. Vol. 10 (4). Pp. 1292–1302.

References

1. A. Nayfeh, Trans. ASME, Ser. E: J. Appl. Mech. **43** (4), 584–588 (1976).
2. I. Selezov and O. Avramenko, Geophysical Research Abstracts. **3**, 25–30 (2001).
3. I. Selezov, O. Avramenko, C. Kharif, and K. Trulsen, Comptes Rendus. Mecanique. **331** (3), 197–201 (2003).
4. I. T. Selezov and O. V. Avramenko, Journal of Mathematical Sciences. **103**, 409–413 (2001).
5. Yu. V. Gurtovy, I. T. Selezov, and O. V. Avramenko, International Journal of Fluid Mechanics Research. **34** (5), 475–491 (2007).
6. Yu. V. Hurtovyi, Y. T. Selezov, and O. V. Avramenko, Prykladna hidromekhanika. **90** (8), 60–65 (2006).
7. O. V. Avramenko, V. V. Naradovyi, and I. T. Selezov, Journal of Mathematical Sciences. **212**, 131–141 (2016).
8. O. Avramenko, M. Lunyova, and V. Naradovyi, Eastern-European journal of enterprise technologies. **5** (5), 58–66 (2017).
9. V. V. Naradovyi and D. S. Kharchenko, Mathematical Modeling and Computing. **10** (4), 1292–1302 (2023).

V. Naradovyi

NONLINEAR PROPAGATION OF WAVE PACKETS AT WAVE NUMBERS CLOSE TO THE CRITICAL ONE IN A TWO-LAYER HYDRODYNAMIC SYSTEM OF FINITE DEPTH

The problem of the propagation of weakly nonlinear wave packets in a two-layer hydrodynamic system, ≪layer with a solid bottom — layer with a lid,≫ is considered. The method of multiple-scale expansions (MSE) up to the third order is employed for investigation and analysis. This method allows one to obtain the first approximations of the studied model, which are linear with respect to the unknown functions that are the terms in the respective expansions. As a result, the evolution equation for the wave packet envelope is derived in the form of a nonlinear Schrödinger equation. When the central frequency of the wave packet is close to zero, the results obtained using MSE cannot be applied to model wave motions in the studied system. The article examines the limiting case of wave packet propagation at near-critical wave numbers. Based on the dispersion relation and solvability conditions for the second and third approximations, it is established that the propagation of wave packets at wave numbers close to the critical one is described by the nonlinear Schrödinger equation. The derived equation includes the first spatial derivative and two temporal derivatives and can be extended to all wave numbers. Additionally, a relation between the wave number and the small parameter is derived.

Keywords: internal waves, nonlinear Schrödinger equation, wave number.

Матеріал надійшов 28.12.2024

ВІДОМОСТІ ПРО АВТОРІВ

Аверкін Олександр Сергійович — випускник бакалаврської програми за спеціальністю «Прикладна математика» Національного університету «Києво-Могилянська академія». Сфера наукових інтересів: теорія графів.

Авраменко Ольга Валентинівна — доктор фіз.-мат. наук, професор кафедри математики Національного університету «Києво-Могилянська академія» та старший науковий співробітник відділу STEM, університет Вітовта Великого, Каунас, Литва. Сфера наукових інтересів: математичне моделювання, хвильовий рух у рідинах, символьне обчислення, динамічні системи.

Зубріцька Дар'я Євгенівна — випускниця бакалаврату, факультет інформатики Національного університету «Києво-Могилянська академія». Сфера наукових інтересів: дробове числення, фінансова математика, програмне моделювання.

Ліхачов Артемій Дмитрович — випускник магістерської програми «Прикладна математика» Національного університету «Києво-Могилянська академія». Сфера наукових інтересів: криптографія.

Митник Олег Юрійович — канд. техн. наук, старший викладач кафедри математики Національного університету «Києво-Могилянська академія». Сфера наукових інтересів: теорія ймовірності, машинне навчання.

Нарадовий Володимир Володимирович — кандидат тех. наук, доцент кафедри інформатики, програмування, штучного інтелекту та технологічної освіти Центральноукраїнського державного університету імені Володимира Винниченка. Сфера наукових інтересів: математичне моделювання, хвильовий рух у рідинах, символьні обчислення, чисельні методи.

Олійник Богдана Віталіївна — доктор фіз.-мат наук, професор кафедри математики Національного університету «Києво-Могилянська академія». Сфера наукових інтересів: алгебра, комбінаторика, теорія графів, теорія груп, криптографія, теорія метричних просторів.

Поляков Михайло Хельгович — аспірант, спеціальність «Прикладна математика» Національного університету «Києво-Могилянська академія». Сфера наукових інтересів: машинне навчання, комп'ютерний зір, оброблення природної мови.

Случинський Дмитро Юрійович — аспірант 1-го року навчання, спеціальність «Прикладна математика» Національного університету «Києво-Могилянська академія». Сфера наукових інтересів: фінансова математика, ризик-менеджмент, застосування машинного навчання в фінансовій математиці.

Тимошкевич Лариса Миколаївна — канд. фіз.-мат. наук, старший викладач кафедри математики Національного університету «Києво-Могилянська академія». Сфера наукових інтересів: теорія графів, функціональний аналіз, матричний аналіз.

Чорней Руслан Костянтинович — канд. фіз.-мат. наук, доцент, завідувач кафедри математики Національного університету «Києво-Могилянська академія». Сфера наукових інтересів: керовані випадкові поля, стохастичні ігри.

Швай Надія Олександрівна — доцент кафедри математики Національного університету «Києво-Могилянська академія». Сфера наукових інтересів: матричні задачі, машинне навчання, комп'ютерний зір.

Щестюк Наталія Юріївна — канд. фіз.-мат. наук, доцент кафедри математики Національного університету «Києво-Могилянська академія». Сфера наукових інтересів: фінансова математика, оцінювання опціонів, оцінки випадкових процесів і полів.

NOTES ABOUT THE AUTHORS

Oleksandr Averkin — a graduate of the bachelor's program in the specialty of "Applied Mathematics", National University of Kyiv-Mohyla Academy. Research interests: graph theory.

Olga Avramenko — Doctor of Physics and Mathematics, Professor of the Department of Mathematics, National University of Kyiv Mohyla Academy, and a Senior Researcher at the STEM Department, Vytautas Magnus University, Kaunas, Lithuania. Research interests: mathematical modelling, wave motion in fluids, symbolic computing, dynamical systems.

Ruslan Chornei — PhD in Physics and Mathematics, Associate Professor and Head of the Department of Mathematics, National University of Kyiv-Mohyla Academy. Research interests: controlled random fields, stochastic games.

Artemii Likhachov — a second-year master's degree student at the National University of Kyiv-Mohyla Academy in the specialty of applied mathematics. Research interests: cryptography.

Oleh Mytnyk — PhD (Technical Sciences), Senior Lecturer, Department of Mathematics, National University of Kyiv-Mohyla Academy. Research interests: probability theory, machine learning.

Volodymyr Naradovyi — PhD, Associate Professor of the Department of Informatics, Programming, Artificial Intelligence and Technological Education, Volodymyr Vynnychenko Central Ukrainian State University. Research interests: mathematical modelling, wave motion in fluids, symbolic computing, numerical methods.

Bogdana Oliynyk — a Doctor of Sciences and Professor, National University of Kyiv-Mohyla Academy. Research interests: algebra, combinatorics, graph theory, group theory, cryptography, and metric space theory.

Mykhailo Polyakov — a postgraduate student in the specialty of "Applied Mathematics", National University of Kyiv-Mohyla Academy. Research interests: machine learning, computer vision, natural language processing.

Nataliya Shchestyuk — Candidate of Physics and Mathematics, Associate Professor of the Department of Mathematics, National University of Kyiv-Mohyla Academy. Research interests: financial mathematics, option valuation, estimation of stochastic processes and fields.

Nadiya Shvai — Associate Professor of the Department of Mathematics, National University of Kyiv-Mohyla Academy. Research interests: matrix problems, machine learning, computer vision.

Dmytro Sluchynskyi — first-year postgraduate student in the specialty "Applied Mathematics", National University of Kyiv-Mohyla Academy. Research interests: financial mathematics, risk management, application of machine learning in financial mathematics.

Larisa Tymoshkevych — Candidate of Physics and Mathematics, a Senior Lecturer in the Department of Mathematics, National University of Kyiv-Mohyla Academy. Research interests: graph theory, functional analysis, matrix analysis.

Darya Zubritska — bachelor's degree, Faculty of Informatics, National University of Kyiv-Mohyla Academy. Research interests: fractional calculus, financial mathematics, software modelling.

ЗМІСТ

CONTENTS