

Харківський національний університет імені В.Н. Каразіна
Факультет математики і інформатики
Кафедра прикладної математики

Кваліфікаційна робота
бакалавра

на тему **«Чисельне розв’язання одного класу
диференціальних рівнянь з частинними
похідними в кільці кополіномів»**

Виконав: студент групи МП41,
спеціальність
113 – Прикладна математика,
освітньо-професійна програма
«Прикладна математика»
Бабаєв Є.М.

Науковий керівник: кандидат фіз.-мат. наук,
доцент кафедри
прикладної математики
Півень О.Л.

Рецензент: доктор технічних наук,
професор кафедри
комп’ютерно-інтегрованих те-
хнологій, автоматизації та ро-
бототехніки Харківського на-
ціонального університету ра-
діоелектроніки
Ромашов Ю.В.

Харків — 2025 рік

Анотації

Бабаєв Єгор Миколайович. Чисельне розв’язання одного класу диференціальних рівнянь з частинними похідними в кільці кополіномів. У кваліфікаційній роботі досліджено задачу Коші для одного класу нелінійних диференціальних рівнянь з частинними похідними, заданих в кільці формальних степеневих рядів із кополіноміальними коефіцієнтами над комутативним кільцем у модулі кополіномів над комутативним кільцем. Надано програмну реалізацію, яка розв’язує рекурентне рівняння для коефіцієнтів розв’язку і дає результат дії розв’язку на заданий поліном.

Ключові слова: кополіном, δ -функція, диференціальне рівняння, задача Коші, перетворення Коші—Стілтєса.

Babaiev Yehor Mykolayovych. Numerical Solution of a Class of Partial Differential Equations in the Ring of Copolynomials. This qualification work investigates the Cauchy problem for a class of nonlinear partial differential equations defined in the ring of formal power series with copolynomial coefficients over a commutative ring, within the module of copolynomials over a commutative ring. A software implementation is provided that solves the recurrence relation for the coefficients of the solution and computes the action of the solution on a given polynomial.

Keywords: copolynomial, δ -function, differential equation, Cauchy problem, Cauchy–Stieltjes transform.

Зміст

Вступ	4
1. Кополіноми та їх властивості	5
1.1. Кополіноми, похідні та ряди з кополіномів	5
1.2. Перетворення Коші–Стілтєса та множення кополіномів . .	7
1.3. Формальні степеневі ряди в кільці кополіномів	8
2. Диференціальні рівняння з частинними похідними в кільці формальних степеневих рядів із кополіноміальними коефіцієнтами	10
2.1. Постановка задачі Коші і теорема існування та єдиності розв’язку	10
2.2. Теорема існування і єдиності	11
2.3. Алгоритм розв’язання рекурентних рівнянь (2.5)–(2.7) . . .	13
2.4. Чисельне розв’язання рівняння	13
Висновки	23
Список використаних джерел	24
Додаток	26

Вступ

У сучасній математичній фізиці та теорії розподілів важливу роль відіграють задачі, пов'язані з диференціальними рівняннями в узагальнених просторах. Особливий інтерес викликає дослідження нелінійних диференціальних рівнянь з частинними похідними, які задано не у класичних функціональних просторах, а в алгебраїчних конструкціях — зокрема, в кільці кополіномів над комутативним кільцем. [2] - [5]

Актуальність теми зумовлена розвитком алгебраїчних підходів до аналізу диференціальних рівнянь.

Метою цієї роботи є побудова чисельного розв'язання задачі Коші для одного класу нелінійних рівнянь з частинними похідними, визначених у модулі кополіномів. Цей клас розглянуто в [3].

У першому розділі викладено основні означення та поняття з [3], що стосуються кополіномів, δ -функції та формальних рядів з кополіноміальними коефіцієнтами. У другому розділі надано постановку задачі Коші, наведено теорему існування та єдиності розв'язку цієї задачі, а також рекурентні рівняння, до яких зводиться диференціальне рівняння [3]. Запропоновано чисельний метод, який знаходить результат дії розв'язку на заданий кополіном, заснований на розв'язанні зазначених рекурентних рівнянь.

Розділ 1

Кополіноми та їх властивості

1.1. Кополіноми, похідні та ряди з кополіномів

В цьому розділі наведемо необхідні властивості кополіномів з [3].

Означення 1.1. [3] Нехай K — комутативне цілісне кільце з одиницею, $K[x]$ — кільце многочленів з коефіцієнтами в K . **Кополіномом** над K називається K -лінійний функціонал, визначений на $K[x]$, тобто гомоморфізм K -модуля $K[x]$ у K .

Позначимо модуль кополіномів через $K[x]'$. Отже, елемент $T \in K[x]'$, якщо $T : K[x] \rightarrow K$ — K -лінійне відображення:

$$T(ap + bq) = aT(p) + bT(q), \quad (1.1)$$

для всіх $p, q \in K[x]$, $a, b \in K$. Значення T на p позначається $\langle T, p \rangle$ або (T, p) .

Означення 1.2. [3] Похідною T' від кополінома $T \in K[x]'$ називається функціонал:

$$(T', p) = -(T, p'), \quad p \in K[x]. \quad (1.2)$$

Похідна n -го порядку: [3]

$$(T^{(n)}, p) = (-1)^n (T, p^{(n)}). \quad (1.3)$$

Зокрема, [3]

$$\left(\frac{T^{(n)}}{n!}, p\right) = (-1)^n \left(T, \frac{p^{(n)}}{n!}\right), \quad p \in K[x].$$

Приклад 1.3. Нехай $\delta^{(n)}(p) = (-1)^n p^{(n)}(0)$. Тоді $\delta^{(n)} \in K[x]'$ — похідна n -ого порядку кополіноміальної δ -функції [3].

Означення 1.4. Ряд $\sum_{n=0}^{\infty} T_n$ збігається в $K[x]'$, якщо послідовність його часткових сум збігається до деякого $T \in K[x]'$ [3].

Теорема 1.5. [3] Нехай $\{a_n\}_{n=0}^{\infty} \subset K$, $T \in K[x]'$. Тоді ряд

$$\sum_{n=0}^{\infty} \frac{a_n T^{(n)}}{n!}$$

збігається в $K[x]'$.

Приклад 1.6. Обчислимо: $\sum_{n=0}^{\infty} n \left(\frac{\delta^{(n)}}{n!}, x^3\right)$. Маємо:

$$\sum_{n=0}^3 n(-1)^n \left(\delta, \frac{(x^3)^{(n)}}{n!}\right) = -3.$$

Лема 1.7. [3] Для $T \in K[x]'$ виконується:

$$T = \sum_{n=0}^{\infty} (-1)^n \frac{(T, x^n)}{n!} \delta^{(n)}. \quad (1.4)$$

Приклад 1.8. [3] Для $\delta^{(n)}$ маємо: $(\delta^{(n)}, p) = (-1)^n p^{(n)}(0)$.

Приклад 1.9. $(-1)^k \left(\frac{\delta^{(k)}}{k!}, x^m\right) = \delta_{k,m}$ — символ Кронекера.

1.2. Перетворення Коші–Стілтєса та множення кополіномів

У цьому розділі розглянемо властивість операції множення кополіномів з роботи [3], яку було введено за допомоги перетворення Коші–Стілтєса.

Означення 1.10. [3] Для $T \in K[x]'$ перетворенням Коші–Стілтєса називається ряд:

$$\mathcal{C}(T)(s) = \sum_{k=0}^{\infty} \frac{(T, x^k)}{s^{k+1}}. \quad (1.5)$$

Цей формальний степеневий ряд є елементом кільця $\frac{1}{s}K[[\frac{1}{s}]]$, де $K[[t]]$ — це кільце формальних степеневих рядів за змінною t з коефіцієнтами з K [3].

Приклад 1.11. [3] Знайдемо перетворення Коші–Стілтєса дельта-функції:

$$\mathcal{C}(\delta)(s) = \sum_{k=0}^{\infty} \frac{(\delta, x^k)}{s^{k+1}} = \frac{1}{s}.$$

Теорема 1.12. [3] Нехай $T \in K[x]'$. Тоді:

$$\mathcal{C}\left(T^{(n)}\right)(s) = (\mathcal{C}(T))^{(n)}(s), \quad n \in \mathbb{N}. \quad (1.6)$$

Означення 1.13. Добуток кополіномів $T_1, T_2 \in K[x]'$ визначається формулою:

$$T_1 T_2 = \mathcal{C}^{-1}(\mathcal{C}(T_1) \cdot \mathcal{C}(T_2)), \quad (1.7)$$

де $\mathcal{C}^{-1}$ — це обернення до перетворення Коші–Стілтєса [3].

Лема 1.14. [3] Для $T_1, T_2 \in K[x]'$ і $n \in \mathbb{N}_0$:

$$(T_1 T_2, x^n) = \begin{cases} \sum_{k=0}^{n-1} (T_1, x^k)(T_2, x^{n-1-k}), & n \geq 1, \\ 0, & n = 0, \end{cases}$$

Приклад 1.15. Нехай $(T, x^n) = 2^n$. Тоді:

$$(T \cdot \delta, x^n) = \begin{cases} 0, & n = 0, \\ 2^{n-1}, & n \geq 1, \end{cases} \quad n = 0, 1, 2, \dots$$

Встановлено зв'язок між степенями δ -функції та її похідними:

$$(-1)^n \frac{\delta^{(n)}}{n!} = \delta^{n+1}, \quad n = 0, 1, 2, \dots$$

і наслідок з цієї формули:

$$(\delta^n)' = -n\delta^{n+1}, \quad n \in \mathbb{N}.$$

1.3. Формальні степеневі ряди в кільці кополіномів

Кільце формальних степеневих рядів вигляду

$$u(t, x) = \sum_{k=0}^{\infty} u_k(x) t^k,$$

де $u_k(x) \in K[x]'$, будемо позначати через $K[x]'[[t]]$ [3].

Частинна похідна за змінною t від $u(t, x) \in K[x]'[[t]]$ визначається за формулою [3]:

$$\frac{\partial u}{\partial t} = \sum_{k=1}^{\infty} k u_k(x) t^{k-1}.$$

Частинна похідна $\partial u/\partial x$ ряду $u(t, x) \in K[x]'[[t]]$ визначається так:

$$\frac{\partial u}{\partial x} = \sum_{k=0}^{\infty} u'_k(x) t^k.$$

Позначимо через $(u(t, x), p(x))$ дію ряду $u(t, x) \in K[x]'[[t]]$ на $p(x) \in K[x]$:

$$(u(t, x), p(x)) = \sum_{k=0}^{\infty} (u_k(x), p(x)) t^k.$$

Отже, $(u(t, x), p(x)) \in K[[t]]$ [3].

Розділ 2

Диференціальні рівняння з частинними похідними в кільці формальних степеневих рядів із кополіноміальними коефіцієнтами

2.1. Постановка задачі Коші і теорема існування та єдиності розв’язку

Розглянемо задачу Коші для еволюційного рівняння в кільці формальних степеневих рядів $K[x]'[[t]]$:

$$\frac{\partial u}{\partial t} = au^{m_0} \left(\frac{\partial u}{\partial x} \right)^{m_1}, \quad (2.1)$$

$$u(0, x) = u_0 \delta(x), \quad (2.2)$$

Нехай K — комутативне цілісне кільце з одиницею, де $a, u_0 \in K$, $m_0 + m_1 \in \mathbb{N}_0$, $m_0, m_1 \neq 0$.

Оскільки $u(t, x)$, $\frac{\partial u}{\partial x}$, u^{m_0} , $\left(\frac{\partial u}{\partial x} \right)^{m_1}$ є елементами $K[x]'[[t]]$, то добуток у правій частині рівняння (2.1) визначається як добуток формальних рядів з кополіноміальними коефіцієнтами, що обчислюється згідно з операцією множення, описаною за допомогою перетворення Коші–Стілтєса.

Таким чином, задача полягає у знаходженні формального степеневого ряду $u(t, x) \in K[x]'[[t]]$, який задовольняє рівнянням (2.1) і (2.2).

2.2. Теорема існування і єдиності

В [3] доведено наступно теорему існування та єдиності розв'язку задачі Коші (2.1), (2.2).

Теорема 2.1. [3] Нехай $K \supseteq \mathbb{Q}$. Тоді задача Коші (2.1) має єдиний розв'язок у $K[x]'[[t]]$. Цей розв'язок має вигляд

$$u(t, x) = \sum_{k=0}^{\infty} u_k \delta^{nk+1} t^k = \sum_{k=0}^{\infty} u_k \frac{(-1)^{nk} \delta^{(nk)}}{(nk)!}, \quad (2.3)$$

де $u_k \in K$, а $n = m_0 + 2m_1 - 1$. Крім того, для кожного $t \in K$ ряд (2.3) збігається в топології $K[x]'$.

Розглянемо поліном

$$p(x) = \sum_{j=0}^m a_j x^j, \quad \deg p = m$$

Тоді, застосовуючи $u(t, x)$ з (2.3)

$$(u(t, x), p) = \sum_{k=0}^{\infty} t^k u_k (-1)^{nk} \sum_{j=0}^m a_j \left(\frac{\delta^{(nk)}}{(nk)!}, x^j \right)$$

Користуючись прикладом 1.9 маємо

$$\left(\frac{\delta^{(nk)}}{(nk)!}, x^j \right) = \delta_{nk,j}$$

Зокрема

$$\left(\frac{\delta^{(nk)}}{(nk)!}, x^j \right) = 0, nk > m, j = 0, \dots, m$$

Тому

$$(u(t, x), p) = \sum_{k=0}^{\left[\frac{m}{n} \right]} t^k u_k \sum_{j=0}^m a_j (-1)^{nk} \left(\frac{\delta^{(nk)}}{(nk)!}, x^j \right)$$

Отже,

$$(u(t, x), p) = \sum_{k=0}^N t^k \sum_{j=0}^m u_k a_j \delta_{nk,j} = \sum_{k=0}^N c_k t^k, \quad (2.4)$$

$$\text{де } N = \left[\frac{m}{n} \right]$$

і

$$c_k = u_k \sum_{j=0}^m a_j \delta_{nk,j}.$$

У випадку $m_0 = 0$ коефіцієнти u_k ряду (2.3) задовольняють рекурентне рівняння [3].

$$(k+1)u_{k+1} = (-1)^{m_1} a \sum_{|\beta|=k} \prod_{i=1}^{m_1} (n\beta_i + 1) u_{\beta_i}, \quad k = 0, 1, \dots \quad (2.5)$$

Тут $\beta = (\beta_1, \dots, \beta_{m_1})$ — мультиіндекс з невід'ємними цілими координатами, сума компонент якого дорівнює k , тобто $|\beta| = k$.

У випадку $m_1 = 0$ коефіцієнти u_k задовольняють рекурентне рівняння [3].

$$(k+1)u_{k+1} = a \sum_{|\alpha|=k} \prod_{i=1}^{m_0} u_{\alpha_i}, \quad k = 0, 1, \dots \quad (2.6)$$

Тут $\alpha = (\alpha_1, \dots, \alpha_{m_0})$ — мультиіндекс з невід'ємними цілими координатами, сума компонент якого дорівнює k , тобто $|\alpha| = k$.

У випадку $m_0, m_1 > 0$ коефіцієнти u_k задовольняють рекурентне рівняння [3].

$$(k+1)u_{k+1} = (-1)^{m_1} a \sum_{|\tau|=k} \sum_{|\alpha|=\tau_0} \sum_{|\beta|=\tau_1} \left(\prod_{i=1}^{m_1} (n\beta_i + 1) u_{\beta_i} \right) \left(\prod_{j=1}^{m_0} u_{\alpha_j} \right) \quad (2.7)$$

$$k = 0, 1, \dots$$

Тут $\tau = (\tau_0, \tau_1)$, $|\tau| = \tau_0 + \tau_1$, а α та β — вище введені для рівнянь (2.6)

та (2.5) відповідно мультиіндекси з невід’ємними цілими координатами.

2.3. Алгоритм розв’язання рекурентних рівнянь (2.5)–(2.7)

Розв’язання рівнянь (2.5)–(2.7) можна здійснити за допомогою одного алгоритму, оскільки ці рівняння мають схожу структуру. Коли m_0 та m_1 є натуральними числами ($m_0, m_1 \in \mathbb{N}$), ми розв’язуємо рівняння (2.7). Якщо $m_0 \in \mathbb{N}$, $m_1 = 0$, то розв’язуємо рівняння (2.6). Якщо $m_1 \in \mathbb{N}$, $m_0 = 0$, то розв’язуємо рівняння (2.5). Таким чином, при відповідному виборі параметрів m_0 та m_1 здійснюється вибір одного з трьох рівнянь (2.5)–(2.7), що дозволяє побудувати один алгоритм їх чисельного розв’язання.

2.4. Чисельне розв’язання рівняння

Всюди припускається далі, що $K \supset \mathbb{Q}$.

Опис алгоритму

Алгоритм для обчислення послідовності u_k і результату дії $(u(t, x), p)$, де p – заданий поліном, реалізований мовою Python та базується на переборі всіх можливих мультиіндексів з заданими властивостями. Наведемо основні кроки алгоритму для рівняння (2.7). Для рівнянь (2.5), (2.6) ці кроки спрощуються, бо немає необхідності генерувати мультиіндекс τ , а також мультиіндекс α при $m_0 = 0$ і мультиіндекс β при $m_1 = 0$:

Задаємо поліном $p(x) = \sum_{j=0}^m a_j x^j$

1. Для кожного значення k від 0 до $N = \left\lceil \frac{m}{n} \right\rceil$:
 - Генеруємо всі можливі пари (τ_0, τ_1) , такі що $\tau_0 + \tau_1 = k$.

2. Для кожної пари (τ_0, τ_1) :
 - Генеруємо всі можливі мультиіндекси α довжини m_0 з невід'ємними цілими координатами, сума яких дорівнює τ_0 .
 - Генеруємо всі можливі мультиіндекси β довжини m_1 з невід'ємними цілими координатами, сума яких дорівнює τ_1 .
3. Обчислення добутків:
 - Для кожного мультиіндексу α обчислюємо добуток значень u_{α_i} для всіх компонент α .
 - Для кожного мультиіндексу β :
 - (i) Обчислюємо множник як добуток виразів $(n\beta_i + 1)$ для всіх компонент β .
 - (ii) Обчислюємо добуток значень u_{β_i} для всіх компонент β .
4. Обчислення суми S :
 - Сума S обчислюється як сума всіх можливих добутків з кроку 3 для всіх мультиіндексів α та β .
5. Обчислення наступного члена послідовності u_{k+1} :

$$u_{k+1} = \frac{(-1)^{m_1} a S}{k+1}.$$

Приклад 1

Розглянемо задачу Коші (2.1), (2.2), в якій

$$a = -1, \quad m_0 = 0, \quad m_1 = 2, \quad u_0 = 1.$$

Отже, диференціальне рівняння (2.1) має вигляд

$$\frac{\partial u}{\partial t} + \left(\frac{\partial u}{\partial x} \right)^2 = 0$$

Це є рівняння типу Гамільтона-Якобі [3].

Обчислюємо:

$$n = m_0 + 2m_1 - 1 = 0 + 4 - 1 = 3.$$

Відповідне рекурентне рівняння (2.5) приймає вигляд:

$$(k+1)u_{k+1} = a \sum_{\beta_1=0}^k (3\beta_1+1)(3(k-\beta_1)+1)u_{\beta_1}u_{k-\beta_1}.$$

Для $k=0$:

Єдиний мультиіндекс $\beta = (0, 0)$, бо $|\beta| = 0$. Обчислюємо:

$$\text{beta_product} = (3 \cdot 0 + 1)(3 \cdot 0 + 1) = 1 \cdot 1 = 1,$$

$$u_\beta = u_0 u_0 = 1 \cdot 1 = 1, \quad S = 1.$$

$$u_1 = \frac{(-1)^1 \cdot a \cdot S}{1} = \frac{(-1) \cdot (-1) \cdot 1}{1} = -1.$$

Для $k=1$:

Можливі $\beta = (0, 1), (1, 0)$.

Для $(0, 1)$:

$$\text{beta_product} = (3 \cdot 0 + 1)(3 \cdot 1 + 1) = 1 \cdot 4 = 4,$$

$$u_\beta = u_0 u_1 = 1 \cdot (-1) = -1, \quad \Rightarrow S_1 = -4.$$

Для $(1, 0)$:

$$\text{beta_product} = (3 \cdot 1 + 1)(3 \cdot 0 + 1) = 4 \cdot 1 = 4,$$

$$u_\beta = u_1 u_0 = (-1) \cdot 1 = -1, \quad \Rightarrow S_2 = -4.$$

$$S = S_1 + S_2 = -8, \quad u_2 = \frac{(-1)^2 \cdot (-1) \cdot (-8)}{2} = 4.$$

Для $k = 2$:

Можливі $\beta = (0, 2), (1, 1), (2, 0)$.

$$\text{Для } \beta = (0, 2): \quad \text{beta_product} = 1 \cdot 7 = 7, \quad u_\beta = u_0 u_2 = 1 \cdot 4 = 4, \quad S_1 = 28.$$

$$\text{Для } \beta = (1, 1): \quad \text{beta_product} = 4 \cdot 4 = 16, \quad u_\beta = u_1 u_1 = (-1)^2 = 1, \quad S_2 = 16.$$

$$\text{Для } \beta = (2, 0): \quad \text{beta_product} = 7 \cdot 1 = 7, \quad u_\beta = u_2 u_0 = 4 \cdot 1 = 4, \quad S_3 = 28.$$

$$S = 28 + 16 + 28 = 72, \quad u_3 = \frac{(-1)^3 \cdot (-1) \cdot 72}{3} = -24.$$

Для $k = 3$:

Можливі $\beta = (0, 3), (1, 2), (2, 1), (3, 0)$. Після обчислення суми:

$$S = -704, \quad u_4 = \frac{(-1)^4 \cdot (-1) \cdot (-704)}{4} = 176.$$

Отримано члени послідовності $\{u_k\}_{k=0}^\infty$:

$$u_0 = 1, \quad u_1 = -1, \quad u_2 = 4, \quad u_3 = -24, \quad u_4 = 176.$$

Нехай $p(x) = x^8$. Тоді $m = 8, N = \left\lceil \frac{m}{n} \right\rceil = \left\lceil \frac{8}{3} \right\rceil = 3$.

Для полінома $p(x) = x^8$ єдиний ненульовий коефіцієнт — $a_8 = 1$ (усі $a_j = 0$ при $j = 1, \dots, 7$).

Тоді за формулою (2.4)

$$(u(t, x), x^8) = c_0 + c_1 t + c_2 t^2,$$

де $c_k = u_k \delta_{3k,8} = 0, k = 0, 1, 2 \dots$ Тому

$$(u(t, x), x^8) = 0 \quad \text{для всіх } t \in K.$$

Приклад 2

Розглянемо задачу Коші (2.1), (2.2), в якій

$$a = -1, \quad m_0 = 3, \quad m_1 = 0, \quad u_0 = 1.$$

Обчислюємо:

$$n = m_0 + 2m_1 - 1 = 3 + 0 - 1 = 2.$$

Отже, диференціальне рівняння (2.1) має вигляд

$$\frac{\partial u}{\partial t} + au^3 = 0$$

Відповідне рекурентне рівняння (2.6) має вигляд:

$$(k+1)u_{k+1} = a \sum_{|\beta|=k} ((3\beta_1+1)(3\beta_2+1)u_{\beta_1}u_{\beta_2}).$$

Для $k = 0$:

$$S = u_0 u_0 = 1, \quad u_1 = \frac{-1 \cdot 1}{1} = -1.$$

Для $k = 1$:

Можливі:

$$(0, 1), (1, 0).$$

$$S = u_0 u_1 + u_1 u_0 = 1 \cdot (-1) + (-1) \cdot 1 = -2. \quad u_2 = \frac{-1 \cdot (-2)}{2} = 1.$$

Для $k = 2$:

Можливі комбінації:

$$(2, 0), (1, 1), (0, 2).$$

Обчислення:

$$S = u_2 u_0 + u_1 u_1 + u_0 u_2 = 1 \cdot 1 + (-1)^2 + 1 \cdot 1 = 3. \quad u_3 = \frac{-1 \cdot 3}{3} = -1.$$

Для $k = 3$:

10 можливих комбінацій:

$$(3, 0), (2, 1), (1, 2), (0, 3), \dots$$

Після обчислень:

$$S = -35/2, \quad u_4 = \frac{-1 \cdot (-35/2)}{4} = \frac{35}{8}.$$

Отримано члени послідовності $\{u_k\}_{k=0}^{\infty}$:

$$u_0 = 1, \quad u_1 = -1, \quad u_2 = \frac{3}{2}, \quad u_3 = -\frac{5}{2}, \quad u_4 = \frac{35}{8}.$$

Нехай $p(x) = x^6$.

Тоді

$$m = 6, \quad N = \left\lfloor \frac{m}{n} \right\rfloor = 3.$$

Тоді за формулою (2.4)

$$(u(t, x), x^6) = c_0 + c_1 t + c_2 t^2 + c_3 t^3,$$

де

$$c_k = u_k \delta_{2k,6} = \begin{cases} 0, & k = 0, 1, 2, \\ -\frac{5}{2}, & k = 3 \end{cases}$$

Отже, $(u(t, x), x^6) = -\frac{5}{2}t^3$.

Приклад 3

Розглянемо задачу Коші (2.1), (2.2), в якій

$$a = -1, \quad m_0 = 1, \quad m_1 = 1, \quad u_0 = 1.$$

Отже, диференціальне рівняння (2.1) – це рівняння Ейлера-Хопфа.

Обчислюємо:

$$n = m_0 + 2m_1 - 1 = 1 + 2 - 1 = 2.$$

Рекурентне рівняння має вигляд:

$$(k+1)u_{k+1} = -a \sum_{|\tau|=k} \sum_{\alpha=\tau_0, \beta=\tau_1} (2\beta_1 + 1)u_\alpha u_\beta.$$

Для $k = 0$:

$$\tau_0 = \tau_1 = 0 \Rightarrow \alpha = \beta = (0).$$

$$S = (2 \cdot 0 + 1)u_0 u_0 = 1 \cdot 1 \cdot 1 = 1.$$

$$u_1 = -aS = -(-1) \cdot 1 = 1.$$

Для $k = 1$:

Можливі варіанти τ :

$$(0, 1), \quad (1, 0)$$

Для $\tau = (0, 1)$:

$$\alpha = (0), \quad \beta = (1) \Rightarrow S_1 = (2 \cdot 1 + 1)u_0 u_1 = 3 \cdot 1 \cdot 1 = 3.$$

Для $\tau = (1, 0)$:

$$\alpha = (1), \quad \beta = (0) \Rightarrow S_2 = (2 \cdot 0 + 1)u_1u_0 = 1 \cdot 1 \cdot 1 = 1.$$

$$S = S_1 + S_2 = 3 + 1 = 4, \quad u_2 = \frac{-aS}{2} = \frac{-(-1) \cdot 4}{2} = 2.$$

Для $k = 2$:

Можливі варіанти $\tau = (\tau_0, \tau_1)$, такі що $\tau_0 + \tau_1 = 2$:

$$(0, 2), \quad (1, 1), \quad (2, 0).$$

Для $\tau = (0, 2)$:

$$\alpha = (0), \quad \beta = (2) \Rightarrow S_1 = (2 \cdot 2 + 1)u_0u_2 = 5 \cdot 1 \cdot 2 = 10.$$

Для $\tau = (1, 1)$:

$$\alpha = (1), \quad \beta = (1) \Rightarrow S_2 = (2 \cdot 1 + 1)u_1u_1 = 3 \cdot 1 \cdot 1 = 3.$$

Для $\tau = (2, 0)$:

$$\alpha = (2), \quad \beta = (0) \Rightarrow S_3 = (2 \cdot 0 + 1)u_2u_0 = 1 \cdot 2 \cdot 1 = 2.$$

$$S = 10 + 3 + 2 = 15, \quad u_3 = \frac{-aS}{3} = \frac{-(-1) \cdot 15}{3} = 5.$$

Для $k = 3$:

Можливі $\tau = (0, 3), (1, 2), (2, 1), (3, 0)$

Для $\tau = (0, 3)$:

$$S_1 = (2 \cdot 3 + 1)u_0u_3 = 7 \cdot 1 \cdot 5 = 35.$$

Для $\tau = (1, 2)$:

$$S_2 = (2 \cdot 2 + 1)u_1u_2 = 5 \cdot 1 \cdot 2 = 10.$$

Для $\tau = (2, 1)$:

$$S_3 = (2 \cdot 1 + 1)u_2u_1 = 3 \cdot 2 \cdot 1 = 6.$$

Для $\tau = (3, 0)$:

$$S_4 = (2 \cdot 0 + 1)u_3u_0 = 1 \cdot 5 \cdot 1 = 5.$$

$$S = 35 + 10 + 6 + 5 = 56, \quad u_4 = \frac{-aS}{4} = \frac{-(-1) \cdot 56}{4} = 14.$$

Отримано члени послідовності $\{u_k\}_{k=0}^{\infty}$:

$$u_0 = 1, \quad u_1 = 1, \quad u_2 = 2, \quad u_3 = 5, \quad u_4 = 14.$$

Нехай $p(x) = x^8$.

Тоді за формулою (2.4)

$$(u(t, x), x^8) = c_0 + c_1t + c_2t^2 + c_3t^3 + c_4t^4,$$

де

$$c_k = u_k \delta_{2k,8} = \begin{cases} 0, & k \neq 4 \\ 14, & k = 4 \end{cases}$$

Отже, $(u(t, x), x^8) = 14t^4$.

Звернемо увагу, що в третьому прикладі виникає послідовність Каталана [3]:

$$u_k = \{1, 1, 2, 5, 14, \dots\}.$$

Числа Каталана мають явну формулу:

$$c_k = \frac{1}{k+1} \binom{2k}{k},$$

а рекурентне співвідношення має вигляд:

$$(k+1)u_{k+1} = \sum_{i=0}^k (2i+1)u_i u_{k-i},$$

яке еквівалентне класичному:

$$u_{k+1} = \sum_{i=0}^k u_i u_{k-i}.$$

Програма чисельного розв'язання (Python)

Для програмної реалізації алгоритму використаємо програму на мові Python. [\[1\]](#)

Дивитись [Додаток](#).

Висновки

Розроблено чисельний алгоритм обчислення єдиного розв'язку задачі Коші

$$\frac{\partial u}{\partial t} = au^{m_0} \left(\frac{\partial u}{\partial x} \right)^{m_1}, \quad u(0, x) = u_0 \delta(x),$$

в кільці формальних степеневих рядів із кополіноміальними коефіцієнтами. Відомо, що шуканий розв'язок цієї задачі подається у вигляді формального ряду за добутками степенів t та степенів δ -функції, коефіцієнти якого задовольняють рекурентне рівняння.

При кожному фіксованому t з основного кільця сума такого ряду утворює кополіном. Наведено програмну реалізацію засобами Python, яка обчислює коефіцієнти ряду і дає результат дії цього ряду як кополінома на заданий поліном.

Список використаних джерел

- [1] Документація Python. [Електронний ресурс]. — Режим доступу:
<https://docs.python.org/3/>
- [2] Гефтер С.Л., Півень О.Л. Диференціальні оператори нескінченного порядку в модулі формальних узагальнених функцій та у кільці формальних степеневих рядів. *Український математичний журнал*. 2022. Т. 74, №6. С. 784–799.
- [3] Gefter S.L., Piven' A.L. Some class of nonlinear partial differential equations in the ring of copolynomials over a commutative ring. *Front. Appl. Math. Stat.* 2022. Vol. 10:1466569. P. 1–9. DOI: 10.3389/fams.2023.1466569.
- [4] Gefter S.L., Piven' A.L. Linear Partial Differential Equations in Module of Formal Generalized Functions over Commutative Ring. *J. Math. Sci.* 2021. Vol. 257(5). P. 579–596.
- [5] Gefter S.L., Piven' A.L. Partial Differential Equations in Module of Copolynomials over a Commutative Ring. *J. Math. Phys. Anal. Geom.* 2025. Vol. 21(1). P. 56–83.

Додаток

```

1  from fractions import Fraction
2
3
4  3 usages
5  def compositions(n, k):
6      """Generate all k-compositions of the integer n."""
7      if k == 0:
8          if n == 0:
9              yield ()
10             return
11         if k == 1:
12             yield (n,)
13         else:
14             for i in range(n + 1):
15                 for comp in compositions(n - i, k - 1):
16                     yield (i,) + comp
17
18  1 usage
19  def compute_u(a, m0, m1, N, u0):
20      """Compute the sequence u_k up to k = N."""
21      n = m0 + 2 * m1 - 1
22      u = [Fraction(u0)]
23      for k in range(N):
24          S = Fraction(0)
25          for tau_0 in range(k + 1):
26              tau_1 = k - tau_0
27              alphas = (
28                  list(compositions(tau_0, m0)) if m0 > 0 else [] if tau_0 == 0 else []
29              )
30              betas = (
31                  list(compositions(tau_1, m1)) if m1 > 0 else [] if tau_1 == 0 else []
32              )
33              if not alphas or not betas:
34                  continue
35              for alpha in alphas:
36                  u_alpha = Fraction(1)
37                  for alpha_i in alpha:
38                      u_alpha *= u[alpha_i]
39              for beta in betas:
40                  beta_prod = Fraction(1)
41                  for beta_i in beta:
42                      beta_prod *= n * beta_i + 1
43              u_beta = Fraction(1)
44              for beta_i in beta:
45                  u_beta *= u[beta_i]
46              S += beta_prod * u_alpha * u_beta
47          u_k1 = (
48              (-1) ** m1 * a * S / Fraction(k + 1)
49              if m0 > 0 and m1 > 0
50              else a * S / Fraction(k + 1)
51          )

```

```

51         u.append(u_k1)
52     return u, n
53
54
55 1 usage
56 def compute_scalar_product(u, a_coeffs, t, n):
57     """Compute the scalar product (u(t,x), p)."""
58     m = len(a_coeffs) - 1
59     N = m // n
60     result = Fraction(0)
61     for k in range(N + 1):
62         nk = n * k
63         if nk <= m:
64             result += t ** k * u[k] * a_coeffs[nk]
65     return result
66
67 1 usage
68 def run_program():
69     print("Enter parameters for computing u_k and scalar product (u(t,x), p):")
70     a = Fraction(input("Enter the value of a (integer): "))
71     m0 = int(input("Enter the value of m0 (non-negative integer): "))
72     m1 = int(input("Enter the value of m1 (non-negative integer): "))
73     u0 = Fraction(input("Enter the initial value u0: "))
74
75     # Define the polynomial p(x)
76     m = int(input("Enter the degree m of the polynomial p(x): "))
77     print(
78         f"Enter the {m + 1} coefficients a_j for the polynomial "
79         f"p(x) = a_0 + a_1 x + ... + a_m x^m"
80     )
81     a_coeffs = []
82     for j in range(m + 1):
83         coeff = Fraction(input(f"a_{j}: "))
84         a_coeffs.append(coeff)
85
86     t = Fraction(input("Enter value of t (integer): "))
87
88     # Automatically calculate n and N
89     n = m0 + 2 * m1 - 1
90     N = m // n
91     print(f"\nComputed n = {n}, number of terms N = {N}")
92
93     # Compute the sequence u_k
94     u_sequence, _ = compute_u(a, m0, m1, N, u0)
95
96     print("\nComputed values of the sequence u_k:")
97     for idx, val in enumerate(u_sequence):
98         print(f"u_{idx} = {val}")

```

```
99         # Compute scalar product (u(t,x), p)
100         scalar_product = compute_scalar_product(u_sequence, a_coeffs, t, n)
101         print(f"\nScalar product (u(t,x), p) = {scalar_product}")
102
103
104 104 ▶ if __name__ == "__main__":
105         run_program()
```