Міністерство освіти і науки України
Харківський національний університет імені В.Н. Каразіна
Факультет математики і інформатики
Кафедра прикладної математики

До захисту допущено
кафедрою прикладної математики, протокол № 5 від 12 червня 2026 р.

завідувач кафедри
прикладної математики
доктор фіз.-мат. наук, професор *Валерій КОРОБОВ*

КВАЛІФІКАЦІЙНА РОБОТА

здобувача першого (бакалаврського) рівня вищої освіти

«Поліноміальні розв'язки систем лінійних диференціальних рівнянь нескінченного порядку»

Спеціальність 113 Прикладна математика
Освітня програма Прикладна математика

Здобувач Дмитро ТИМОЩУК

Науковий керівник кандидат фіз.-мат. наук, доцент
кафедри прикладної математики
Олексій ПІВЕНЬ

Харків – 2026

Анотація

Роботу присвячено опису поліноміальних розв'язків системи лінійних диференціальних рівнянь нескінченного порядку в скінченновимірному просторі. З такою системою пов'язується лінійний диференціальний оператор нескінченного порядку з квадратними матричними коефіцієнтами. Показано, що необхідною і достатньою умовою існування та єдиності розв'язку цієї системи при будь-якій поліноміальній правій частині є оборотність вільного члену цього оператору. Для побудови відповідного єдиного розв'язку застосовується метод італійського математика У. Броджи, який він запропонував для побудови розв'язків скалярних лінійних диференціальних рівнянь скінченного порядку. У випадку необоротності вільного члену лінійного диференціального оператору система або взагалі не має поліноміальних розв'язків, або має нескінченну кількість таких розв'язків. В цьому випадку виникає задача побудови загального поліноміального розв'язку цієї системи. При додаткових умовах на матричні коефіцієнти для опису загального поліноміального розв'язку також можна використати метод У. Броджи. Зроблено програмну реалізацію цього методу в середовищі Python для побудови загального поліноміального розв'язку системи. В загальній ситуації при необоротному вільному члені оператора запропоновано метод невизначених коефіцієнтів, який зводить задачу знаходження поліноміальних розв'язків системи до розв'язання деякої системи лінійних алгебраїчних рівнянь. Створена програмна реалізація методу невизначенних коефіцієнтів дозволяє описати поліноміальні розв'язки степеню не вище заданого числа. Надано оцінку знизу степеню поліноміального розв'язку системи. Результати роботи також частково узагальнюють відповідні результаті робот С.Л. Гефтера, А.Б. Гончарук та О.Л. Півня для скалярних лінійних диференціальних рівнянь скінченного і нескінченного порядку над комутативними кільцями з одиницею.

Ключові слова: система лінійних диференціальних рівнянь нескінченного порядку, поліноміальний розв'язок, диференціальний оператор, матричні коефіцієнти, програмна реалізація

Dmytro TYMOSHCHUK, Polynomial solutions of infinite order systems of linear differential equations

Abstract

This study is devoted to a description of the polynomial solutions of the system of infinite order linear differential equations in the finite-dimensional space. The infinite order linear differential operator with square matrix coefficients is connected with this system. It is shown that the invertibility of the constant matrix term is a necessary and sufficient condition for the existence and uniqueness of the solution of this system for an arbitrary polynomial right-hand side. To construct the corresponding unique solution it is applied the method of Italian mathematician U. Broggi, which was originally used to construct the solutions of scalar finite order linear differential equations. In the case of non-invertibility of the constant matrix term of the linear differential operator, the system either has infinitely many polynomial solutions or has no such solutions. In this case, the problem of the constructing of the general polynomial solution of this system appears. Under additional conditions on matrix coefficients for the description general polynomial solution can be used the U. Broggi's method. The software implementation has been developed for this method in a Python environment for the construction of a general polynomial solution of the system. In the general case, under non-invertibility of the constant matrix term of the operator, an undetermined coefficients method is proposed that reduces the problem of finding polynomial solutions of the system to the problem of solving some system of linear algebraic equations. The software implementation has been developed of the undetermined coefficients method, making it possible to describe polynomial solutions of degree less than or equal to a given integer. A lower bound has been established for the degree of the solution of the system. The results of this study partially generalize the results of S.L. Gefter, A.B. Goncharuk, A.L. Piven' for scalar linear differential equations of finite and infinite order over a commutative ring with identity.

Keywords: system of linear differential equations of infinite order, polynomial solution, differential operator, matrix coefficient, software implementation.

Зміст

Вступ **5**

1 Поліноміальні розв'язки систем лінійних диференціальних рівнянь нескінченного порядку у випадку невиродженої матриці A_0 **7**

1.1 Теорема існування та єдиності та використання методу У. Броджи 7

1.2 Програмна реалізація методу 11

1.3 Приклади 13

2 Поліноміальні розв'язки систем лінійних диференціальних рівнянь нескінченного порядку у випадку виродженості матриці A_0 **15**

2.1 Використання методу У. Броджи для випадку $A_0 = A_1 = \cdots = A_{k-1} = 0, \ \det A_k \neq 0$ 16

2.1.1 Програмна реалізація методу 20

2.1.2 Приклади 22

2.2 Метод невизначених коефіцієнтів 24

2.2.1 Програмна реалізація методу 27

2.2.2 Приклад ілюстрації роботи алгоритму 29

3 Висновки **34**

Список використаних джерел **35**

Вступ

Як відомо (див., наприклад, [1, Теорема 2.3.4]), лінійне диференціальне рівняння зі сталими коефіцієнтами

$$\sum_{j=0}^{n} a_j u^{(j)}(x) = P(x) \tag{0.1}$$

завжди має поліноміальний розв'язок при правій поліноміальній частині $P(x)$.

В 1930 році італійським математиком У. Броджи вперше було запропоновано метод побудови поліноміального розв'язку диференціального рівняння (0.1) за допомоги формального степеневого ряду, що є оборотним до характеристичного многочлену $\sum\limits_{j=0}^{n} a_j \lambda^j$ за умови, що $a_0 \neq 0$ (див. [2, §5, 22.2]). Узагальнення цього результату на випадок рівняння (0.1) над комутативним кільцем з одиницею при умові оборотності елементу a_0, було виконано в статті [3] з доведенням єдиності розв'язку. З теореми [4, Теорема 3, с. 21] випливає існування неєдиного поліноміального розв'язку диференціального рівняння нескінченного порядку $\sum\limits_{j=0}^{\infty} a_j u^{(j)}(x) = P(x)$ з дійсними коефіцієнтами a_j $(a_0 = 0, a_n \neq 0$ для деякого $n \in \mathbb{N})$ і будь-якою правою поліноміальною правою частиною $P(x) \in \mathbb{R}[x]$. При цьому цей розв'язок є єдиний тоді і тільки тоді, коли $a_0 \neq 0$ [4, Теорема 1, с. 20].

В даній роботі розглядається система лінійних диференціальних рівнянь нескінченного порядку

$$\sum_{j=0}^{\infty} A_j u^{(j)}(x) = P(x), \tag{0.2}$$

з квадратними матричними коефіцієнтами A_j з дійсними елементами і поліноміальною вектор-функцією $P(x)$ Систему диференціальних рівнянь (0.2) скінченного порядку k $(A_j = 0, j > k, A_k \neq 0)$ називають ще диференціально-алгебраїчним рівнянням. Такі рівняння досліджувались в багаточисельних роботах (див., наприклад, [5, 6, 7] і бібліографію в цих роботах). Зокрема, в [7] доведено існування та єдиність поліноміального розв'язку диференціально-алгебраїчного рівняння $A_1 u'(x) + A_0 u(x) = P(x)$ у випадку оборотності матриці A_0.

В даній роботі показано, що умова оборотності матричного коефіцієнту

A_0 є необхідною і достатньою умовою існування та єдиності поліноміального розв'язку системи (0.2) нескінченного порядку (Наслідок 1.1). Для побудови розв'язку використовується метод У. Броджи. Цей метод також застосовується і для побудови загального розв'язку системи (0.2) у випадку $A_0 = A_1 = \cdots = A_{k-1} = 0$, $\det A_k \neq 0$ (Теорема 2.1). В загальному випадку для побудови поліноміальних розв'язків системи (0.2) використовується метод невизначених коефіцієнтів. Як і для класичних систем класичних диференціальних рівнянь [1, п. 2.4.7], задача зводиться до розв'язання системи лінійних алгебраїчних рівнянь відносно коефіцієнтів поліноміального розв'язку (Теорема 2.2).

Теоретичні результати роботи ілюструються прикладами. Розроблено програмне забезпечення на мові Python, яке реалізує вказані вище методи. Реалізація методу невизначених коефіцієнтів дозволяє описати поліноміальні розв'язки степеню не вище заданого числа.

1 Поліноміальні розв'язки систем лінійних диференціальних рівнянь нескінченного порядку у випадку невиродженої матриці A_0

Нехай $\mathbb{R}^{n\times n}$ – сукупність квадратних матриць розмірності $n \times n$ з дійсними елементами, $\mathbb{R}^n[x]$ – множина поліномів змінної x з коефіцієнтами з n–вимірного простору $\mathbb{R}^n$. Множина $\mathbb{R}^n[x]$ утворює векторний простір над полем дійсних чисел $\mathbb{R}$, а у частковому випадку $n = 1$ простір $\mathbb{R}^1[x]$ співпадає з комутативним кільцем поліномів $\mathbb{R}[x]$ з дійсними коефіцієнтами.

Розглянемо наступну систему лінійних диференціальних рівнянь:

$$\sum_{j=0}^{\infty} A_j u^{(j)}(x) = P(x), \tag{1.1}$$

де $A_j \in \mathbb{R}^{n\times n}$–задані матричні коефіцієнти, $P(x) \in \mathbb{R}^n[x]$. Будемо шукати поліноміальні розв'язки $u \in \mathbb{R}^n[x]$ рівняння (1.1).

1.1 Теорема існування та єдиності та використання методу У. Броджи

Розглянемо лінійний диференціальний оператор нескінченного порядку

$$F = \sum_{j=0}^{\infty} A_j \frac{d^j}{dx^j} : \mathbb{R}^n[x] \to \mathbb{R}^n[x],$$

який визначено на всьому векторному просторі поліномів $\mathbb{R}^n[x]$. Тоді рівняння (1.1) записується в наступному еквівалентному вигляді

$$F(u(x)) = P(x). \tag{1.2}$$

В наступній теоремі встановлюється критерій бієктивності оператора F, звідки випливають необхідні і достатні умови існування єдиного поліноміального розв'язку рівняння (1.1) при будь-якій поліноміальній правій частині $P(x)$.

Теорема 1.1. *Оператор F бієктивний тоді і тільки тоді, коли* $\det A_0 \neq 0$.

Доведення достатності.

Спочатку доведемо ін'єктивність оператора F. Для цього достатньо показати, що однорідне рівняння

$$F(v(x)) = \sum_{j=0}^{\infty} A_j v^{(j)}(x) = 0. \tag{1.3}$$

має тільки нульовий розв'язок. Припустимо від супротивного, що існує ненульовий розв'язок диференціального рівняння (1.3) у вигляді:

$$v(x) = V_s x^s + V_{s-1} x^{s-1} + \cdots + V_0,$$

де $\deg(v(x)) = s$, а $V_k \in \mathbb{R}^n$ для $k = \overline{0, s}$ та $V_s \neq 0$.

Під дією операції диференціювання степінь полінома зменшується: $\deg(v^{(j)}(x)) < \deg(v(x))$ для $j \geq 1$. Тоді коефіцієнт при старшому степені x^s полінома $\sum_{j=0}^{\infty} A_j v^{(j)}(x)$ дорівнює $A_0 V_s$. Оскільки за умовою матриця A_0 є невиродженою, а за припущенням $V_s \neq 0$, то $A_0 V_s \neq 0$. Ми отримали суперечність з рівнянням (1.3). Отже, наше припущення про існування ненульового розв'язку однорідного рівняння (1.3) було хибним, тому $v(x) \equiv 0$. Таким чином, оператор F є ін'єктивним.

Тепер доведемо сюр'єктивність оператора F. За означенням, оператор F є сюр'єктивним тоді і тільки тоді, коли образ оператора збігається з усім простором $\mathbb{R}^n[x]$. Іншими словами, для будь-якого $P(x) \in \mathbb{R}^n[x]$, існує розв'язок $u(x) \in \mathbb{R}^n[x]$ рівняння (1.2).

Будемо шукати $u(x)$ у вигляді дії деякого лінійного оператора на функцію $P(x)$:

$$u(x) = Q(P(x)) = \sum_{j=0}^{\infty} B_j P^{(j)}(x), \tag{1.4}$$

де B_j — шукані матриці, а оператор Q задовольняє умову $F \cdot Q = I$, де $I : \mathbb{R}^n[x] \to \mathbb{R}^n[x]$ — тотожний оператор. Тобто Q–правообернений оператор.

Для знаходження коефіцієнтів B_j скористаємося операторною рівністю:

$$\left(\sum_{k=0}^{\infty} A_k \frac{d^k}{dx^k} \right) \cdot \left(\sum_{l=0}^{\infty} B_l \frac{d^l}{dx^l} \right) = I,$$

Перемножимо ряди:
$$\sum_{k=0}^{\infty}\sum_{l=0}^{\infty} A_k B_l \frac{d^{l+k}}{dx^{l+k}} = I.$$
Згрупувавши доданки, що мають однаковий порядок похідної, і провівши заміну індексу сумування $j = l + k$, маємо наступну рівність:
$$\sum_{j=0}^{\infty}\sum_{k=0}^{j} A_k B_{j-k} \frac{d^{j}}{dx^{j}} = I.$$
Прирівняємо коефіцієнти при однакових степенях оператора диференціювання. Отримаємо:
$$\begin{cases} A_0 B_0 = I \\ \sum\limits_{k=0}^{j} A_k B_{j-k} = 0, j \in \mathbb{N} \end{cases}$$
Тоді
$$\begin{cases} B_0 = A_0^{-1}, \\ B_j = -A_0^{-1} \sum\limits_{k=1}^{j} A_k B_{j-k},\ j \in \mathbb{N}. \end{cases} \tag{1.5}$$
Отже, коефіцієнти B_j визначені однозначно. Тепер покажемо, що функція (1.4) задовольняє рівність (1.2). Справді,
$$F(u(x)) = \sum_{j=0}^{\infty} A_j \left(\sum_{k=0}^{\infty} B_k P^{(k)}(x)\right)^{(j)} = \sum_{j=0}^{\infty}\sum_{k=0}^{\infty} A_j B_k P^{(k+j)}(x).$$
Згрупуємо доданки при однакових похідних ($l = k + j$):
$$\sum_{l=0}^{\infty}\sum_{j=0}^{l} A_j B_{l-j} P^{(l)}(x) = A_0 B_0 P^{(0)}(x) + \sum_{l=1}^{\infty}\left(\sum_{j=0}^{l} A_j B_{l-j}\right) P^{(l)}(x).$$
Використовуючи систему (1.5), отримаємо:
$$F(u(x)) = P(x) + \sum_{l=1}^{\infty} 0 \cdot P^{(l)}(x) = P(x).$$
Отже побудована функція u задовольняє рівняння (1.2). Таким чином, опе-

ратор F є сюр'єктивним. Враховуючи доведену вище ін'єктивність, оператор F є бієктивним.

Необхідність. Нехай F–бієктивний оператор. Припустимо, що матриця A_0 є необоротною. Тоді існує ненульовий вектор $u_0 \in Ker(A_0)$ таки, що поліном $v(x) \equiv u_0$ є нетривіальним поліноміальним розв'язком однорідного рівняння (1.3). Це суперечить бієктивності оператору F. Теорему доведено повністю. $\square$

Оскільки F — бієктивний оператор, то правооберненний оператор Q (див. (1.4)) є також лівооберненим, тобто:

$$B_0 A_0 = I, \quad \sum_{j=0}^{k} B_j A_{k-j} = 0, \text{ для } k \in \mathbb{N}.$$

Тому з Теореми 1.1 випливає наступний критерій існування єдиного поліноміального розв'язку системи (1.1), а також формула для явного вигляду цього розв'язку.

Наслідок 1.1. *Для того, щоб система* (1.1) *мала єдиний поліноміальний розв'язок $u(x)$ при будь-якій поліноміальній правій частині $P(x) \in \mathbb{R}^n[x]$, необхідно і достатньо, щоб $det\ A_0 \neq 0$. При цьому, єдиний поліноміальний розв'язок системи* (1.1) *має вигляд:*

$$u(x) = F^{-1}(P) = \sum_{j=0}^{\infty} B_j P^{(j)}(x) = \sum_{j=0}^{s} B_j P^{(j)}(x), \tag{1.6}$$

де $\deg u(x) = \deg P(x) = s$ і матричні коефіцієнти B_j визначаються з рекурентних співвідношень:

$$B_j = \begin{cases} A_0^{-1}, \ j = 0, \\ A_0^{-1} \sum\limits_{k=1}^{j} A_k B_{j-k}, \ j \in \mathbb{N}. \end{cases} \tag{1.7}$$

Метод побудови розв'язку системи (1.1), заснований на формулах (1.6), (1.7), і є матричним аналогом методу У. Броджи.

Зауваження 1.1. Якщо $\deg P(x) = s$, то для побудови розв'язку системи (1.1) за формулою (1.6) достатньо знайти за формулами (1.7) скінченну кількість членів послідовності матриць B_j, а саме $B_0, ..., B_s$.

З Теореми 1.1 випливає наступне твердження про розв'язність скалярного лінійного диференціального рівняння нескінченного порядку [4,теорема 1, с. 20–21].

Наслідок 1.2. *Нехай $a_j \in \mathbb{R}$, $j = 0, 1, 2, \ldots$. Для того, щоб скалярне диференціальне рівняння нескінченного порядку $\sum_{j=0}^{\infty} a_j u^{(j)}(x) = P(x)$ мало єдиний поліноміальний розв'язок $u(x) \in \mathbb{R}[x]$ при будь-якій поліноміальній правій частині $P(x) \in \mathbb{R}[x]$, необхідно і достатньо, щоб $a_0 \neq 0$. При цьому $\deg u(x) = \deg P(x) = s$ і розв'язок $u(x)$ має наступний вигляд:*

$$u(x) = \sum_{j=0}^{s} b_j P^{(j)}(x), \tag{1.8}$$

де коефіцієнти b_j визначаються з рекурентних співвідношень:

$$b_j = \begin{cases} a_0^{-1}, \ j = 0, \\ -a_0^{-1} \sum\limits_{k=1}^{j} a_k b_{j-k}, \ j \in \mathbb{N}. \end{cases} \tag{1.9}$$

У частковому випадку рівняння (0.1) скінченного порядку над комутативним кільцем це твердження було доведено раніше в [3].

1.2 Програмна реалізація методу

Для програмної реалізації методів розв'язання системи (1.1) був побудований клас `InfiniteOrderDiffSystem`.

Логіку розв'язання системи (1.1) за умови $\det A_0 \neq 0$ з Наслідку 1.1 містить метод `Broggi_method()` класу `InfiniteOrderDiffSystem`.

1. Перевірка умови бієктивності

Відповідно до Теореми 1.1, першим кроком алгоритму є перевірка невиродженості матриці A_0. Якщо вона не виконується, то метод `Broggi_method()` повертає `None`. У програмному коді за це відповідає метод
`is_bijective_operator()`:

```
def is_bijective_operator(self):
    if np.linalg.det(self.A_func(0)) != 0:
        return True
    return False
```

2. Обчислення матричних коефіцієнтів B_j

Для знаходження матриць B_j використовується рекурентна формула (1.7), наведена в Наслідку 1.1. Програмно вона реалізована у вигляді окремої функції `find_inverse_coefficients()`. Оскільки права частина $P(x)$ є поліномом степеня s, ряд у загальному розв'язку обривається, і нам достатньо знайти матриці B_j лише для $j = 0, \ldots, s$ (див. Зауваження 1.1).

```
def find_inverse_coefficients(A, B, j, n):
    res = np.zeros((n,n))
    if j == 0:
        return np.linalg.inv(A[0,:,:]) # B_0 = A_0^{-1}
    i = 1
    while(i <= j):
        res += A[i,:,:] @ B[j-i,:,:] # Sum A_k * B_{j-k}
        i += 1
    return -B[0,:,:] @ res # Multiply on -A_0^{-1}
```

3. Матричне представлення оператора диференціювання

Програмно поліном $P(x) = (p_1(x), \ldots, p_n(x))^T$ ($p_i(x) \in \mathbb{R}[x]$, $i = 1, \ldots, n, \deg P(x) = s$) представлений, як масив розмірності $n \times s$. Значення кожного i−го рядка масиву відповідають коефіцієнтам полінома $p_i(x)$ при степенях x. Тому для обчислення похідних використана формула $P^{(j)}(x) = P(x)D^j$, де D - матричний оператор диференціювання розмірності $s \times s$, вигляду:

$$D = \begin{pmatrix} 0 & 0 & \ldots & 0 & 0 \\ 1 & 0 & \ldots & 0 & 0 \\ 0 & 2 & \ddots & 0 & 0 \\ \ldots & \ldots & \ddots & \ddots & \ldots \\ 0 & 0 & \ldots & s-1 & 0 \end{pmatrix}$$

Матрицю D формує функція `build_diff_matrix()`:

```
def build_diff_matrix(n):
    return np.diag(np.arange(1, n), k=-1)
```

4. Формування розв'язку

Фінальна збірка розв'язку $u(x) = \sum_{j=0}^{s} B_j P^{(j)}(x)$ відбувається в основному циклі методу `Broggi_method`. Алгоритм послідовно обчислює матриці A_j та матриці

B_j, після чого додає їхні добутки на похідну $P^{(j)}(x)$ до загального результату:

```
def Broggi_method(self):
    if not self.is_bijective_operator():
        return None

    s = self.degree
    n = self.dimension
    # ... Initialisation of arrays A, B, res, P_dx ...
    D = build_diff_matrix(s)

    for i in range(s):
        A[i, :, :] = np.array(A_j(i)).astype(np.float64)
        B[i, :, :] = find_inverse_coefficients(A, B, i, n)

    res += B[0, :, :] @ P
    for i in range(1,s):
        res += B[i, :, :] @ (P@np.linalg.matrix_power(D,i))

    return res
```

1.3 Приклади

Приклад 1.1. Нехай в (1.1)

$$A_j = \begin{pmatrix} j & j+1 \\ j+2 & j \end{pmatrix}, P(x) = \begin{pmatrix} x^3 \\ x^2+1 \end{pmatrix}.$$

Розглянемо матрицю

$$A_0 = \begin{pmatrix} 0 & 1 \\ 2 & 0 \end{pmatrix},$$

її детермінант $\det A_0 = -2 \neq 0$. Згідно з Наслідком 1.1 для заданої системи існує єдиний поліноміальний розв'язок вигляду (1.6), де коефіцієнти B_j знаходяться з рекурентних співвідношень (1.7).

Степінь правої частини $\deg P(x) = 3$, це означає що згідно із Зауваженням 1.1 нам достатньо знайти лише матричні коефіцієнти B_j при $j = 0, 1, 2, 3, 4$. Для цього запишемо матриці A_j $(j = 1, 2, 3)$:

$$A_1 = \begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 2 & 3 \\ 4 & 2 \end{pmatrix}, \quad A_3 = \begin{pmatrix} 3 & 4 \\ 5 & 3 \end{pmatrix}$$

Обчислимо матриці B_j за рекурентними формулами (1.7):

$$B_0 = A_0^{-1} = \begin{pmatrix} 0 & 1/2 \\ 1 & 0 \end{pmatrix}$$

$$B_1 = -A_0^{-1}(A_1 B_0) = \begin{pmatrix} -1/2 & -3/4 \\ -2 & -1/2 \end{pmatrix}$$

$$B_2 = -A_0^{-1}(A_1 B_1 + A_2 B_0) = \begin{pmatrix} 3/4 & 3/8 \\ 3/2 & 3/4 \end{pmatrix}$$

$$B_3 = -A_0^{-1}(A_1 B_2 + A_2 B_1 + A_3 B_0) = \begin{pmatrix} -3/8 & -3/16 \\ -3/4 & -3/8 \end{pmatrix}$$

Знайдемо розв'язок $u(x)$ за формулою (1.6):

$$u(x) = \begin{pmatrix} u_1(x) \\ u_2(x) \end{pmatrix} = \begin{pmatrix} 0 & 1/2 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x^3 \\ x^2 + 1 \end{pmatrix} + \begin{pmatrix} -1/2 & -3/4 \\ -2 & -1/2 \end{pmatrix} \begin{pmatrix} 3x^2 \\ 2x \end{pmatrix} +$$

$$+ \begin{pmatrix} 3/4 & 3/8 \\ 3/2 & 3/4 \end{pmatrix} \begin{pmatrix} 6x \\ 2 \end{pmatrix} + \begin{pmatrix} -3/8 & -3/16 \\ -3/4 & -3/8 \end{pmatrix} \begin{pmatrix} 6 \\ 0 \end{pmatrix} =$$

$$= \begin{pmatrix} -x^2 + 3x - 1 \\ x^3 - 6x^2 + 8x - 3 \end{pmatrix}$$

Вигляд розв'язку наданого програмою:

```
u_1 = -x**2 + 3*x - 1
u_2 = x**3 - 6*x**2 + 8*x - 3
```

Приклад 1.2. Розглянемо наступне скалярне диференціальне рівняння

$$\sum_{j=0}^{\infty} 2^j u^{(j)}(x) = x^2. \tag{1.10}$$

Оскільки $a_0 \neq 0$, то за Наслідком 1.2 існує єдиний поліноміальний розв'язок цього рівняння. Цей розв'язок має вигляд (1.8), де коефіцієнти b_j знаходяться з системи (1.9).

Знайдемо коефіцієнти b_j, $j = 0, 1, 2$ з системи (1.9), при $a_j = 2^j$.

$$
\begin{aligned}
j = 0: &\quad b_0 = a_0^{-1} = 1 \\
j = 1: &\quad b_1 = -a_0^{-1} a_1 b_0 = -2 \\
j = 2: &\quad b_2 = -a_0^{-1}(a_1 b_1 + a_2 b_0) = 0
\end{aligned}
$$

Отримуємо єдиний поліноміальний розв'язок рівняння (1.10) у вигляді:

$$u(x) = b_0 P(x) + b_1 P'(x) + b_2 P''(x) = x^2 - 4x.$$

Вигляд розв'язку наданого програмою:

```
u_1 = x**2 - 4*x
```

2 Поліноміальні розв'язки систем лінійних диференціальних рівнянь нескінченного порядку у випадку виродженості матриці A_0

В даному розділі досліджується система (1.1) за умови $\det A_0 = 0$. Відомо, що система диференціальних рівнянь першого порядку

$$A_1 u'(x) + A_0 u(x) = P(x) \tag{2.1}$$

може не мати розв'язків лише у випадку сингулярного жмутка матриць $\lambda A_1 + A_0$ [5, теорема 2.3.1], тобто у випадку $\det(\lambda A_1 + A_0) \equiv 0$. Зокрема, як показано в [6, Приклад 2.15], не для будь-якого поліному $P(x)$ система (2.1) має поліноміальний розв'язок. Наступний приклад показує, що у випадку $\det A_0 = 0$ системи лінійних диференціальних рівнянь (1.1) нескінченного порядку може взагалі не мати поліноміальних розв'язків.

Приклад 2.1. Нехай в (1.1)

$$A_j = \begin{pmatrix} j & 0 \\ j & 0 \end{pmatrix}, \; j = 0, 1, 2, ..., P(x) = \begin{pmatrix} 0 \\ x \end{pmatrix}.$$

Тоді будь-який поліноміальний розв'язок $u = \begin{pmatrix} u_1(x) \\ u_2(x) \end{pmatrix} \in \mathbb{R}^2[x]$ повинен задовольняти наступну систему рівнянь:

$$\begin{cases} u_1(x) + u_2(x) + u_1'(x) + u_2'(x) + \cdots = 0, \\ 0 = x \end{cases}$$

Друге рівняння цієї системи є суперечним, тому відповідна система диференціальних рівнянь (1.1) не має поліноміальних розв'язків.

2.1 Використання методу У. Броджи для випадку $A_0 = A_1 = \cdots = A_{k-1} = 0, \; \det A_k \neq 0$

Спочатку розглянемо випадок, коли $A_0 = A_1 = \cdots = A_{k-1} = 0$. Наступна теорема показує, що у цьому випадку задача знаходження поліноміальних розв'язків системи (1.1) зводиться шляхом заміни $v(x) = u^{(k)}(x)$ до аналогічної системи з оборотним матричним коефіцієнтом при $v(x)$. Також вже в цьому випадку порушується умова єдиності поліноміального розв'язку системи (1.1).

Теорема 2.1. *Нехай* $A_0 = A_1 = \cdots = A_{k-1} = 0, \; \det A_k \neq 0$ *і* $P(x) = \sum\limits_{j=0}^{s} P_j x^j$, *причому* $\deg P(x) = s$. *Тоді система*

$$\sum_{j=k}^{\infty} A_j u^{(j)}(x) = P(x), \tag{2.2}$$

має загальний поліноміальний розв'язок $u(x)$ *вигляду:*

$$u(x) = \sum_{i=0}^{s} x^{k+i} \left(\sum_{j=0}^{s-i} \frac{(i+j)!}{(i+k)!} B_j P_{i+j} \right) + \sum_{j=0}^{k-1} C_j \frac{x^j}{j!}, \tag{2.3}$$

де $C_j \in \mathbb{R}^n$ *– довільні сталі вектори. Матричні коефіцієнти* B_j *визначаються* *з наступних співвідношень:*

$$B_j = \begin{cases} A_k^{-1}, & j = 0, \\ -A_k^{-1} \sum_{i=1}^{j} A_{i+k} B_{j-i}, & j \in \mathbb{N}. \end{cases} \tag{2.4}$$

Доведення.

Змінимо в (2.2) індекс підсумовування $j' = j - k$. Система (1.1) переписується в наступному вигляді:

$$\sum_{j=0}^{\infty} A_{j+k} u^{(j+k)}(x) = P(x).$$

Використовуючи заміну

$$v(x) = u^{(k)}(x), \tag{2.5}$$

приходимо до наступної системи диференціальних рівнянь:

$$\sum_{j=0}^{\infty} A_{j+k} v^{(j)}(x) = P(x). \tag{2.6}$$

За Наслідком 1.1 система (2.6) має єдиний розв'язок, причому цей розв'язок має вигляд:

$$v(x) = \sum_{j=0}^{\infty} B_j P^{(j)}(x), \tag{2.7}$$

де коефіцієнти B_j знаходяться з рекурентних рівнянь (2.4). Оскільки $P(x)$ — векторний поліном степеня s, то $P^{(j)}(x) = 0$ для $j \in \mathbb{N},\ j > s$. Відповідно, розв'язок (2.7) набуде наступної форми:

$$v(x) = \sum_{j=0}^{s} B_j P^{(j)}(x). \tag{2.8}$$

Похідну $P^{(j)}(x)$ можна представити в наступному вигляді:

$$P^{(j)}(x) = \sum_{i=0}^{s-j} \frac{(i+j)!}{i!} P_{i+j} x^i, \quad j = 0, \ldots, s.$$

Підставимо цей вираз в (2.8). Отримаємо

$$v(x) = \sum_{j=0}^{s} B_j \sum_{i=0}^{s-j} \frac{(i+j)!}{i!} P_{i+j} x^i.$$

Змінимо порядок підсумування в цій формулі. Отримаємо:

$$v(x) = \sum_{i=0}^{s} \left(\sum_{j=0}^{s-i} \frac{(i+j)!}{i!} B_j P_{i+j} \right) x^i.$$

Для зручності позначимо

$$V_i = \left(\sum_{j=0}^{s-i} \frac{(i+j)!}{i!} B_j P_{i+j} \right), \ i = 0, \ldots, s. \tag{2.9}$$

Тоді

$$v(x) = \sum_{i=0}^{s} V_i x^i. \tag{2.10}$$

Нехай $J : \mathbb{R}^n[x] \to \mathbb{R}^n[x]$ - оператор інтегрування, який визначається за наступним правилом:

$$J(f(x)) = \int_0^x f(z) dz, \quad f \in \mathbb{R}^n[x].$$

Враховуючи заміну (2.5), отримаємо загальний поліноміальний розв'язок системи (1.1):

$$u(x) = J^k v(x) + \sum_{j=0}^{k-1} C_j \frac{x^j}{j!}, \tag{2.11}$$

де J^k має вигляд:

$$J^k v(x) = \frac{1}{(k-1)!} \int_0^x (x-z)^{k-1} v(z)\, dz. \tag{2.12}$$

Зауважимо, що

$$\int_0^x \frac{(x-z)^{k-1}}{(k-1)!} z^i \, dz = \frac{i!}{(i+k)!} x^{i+k}, \quad i = 0, \ldots, s. \tag{2.13}$$

Тепер, з урахуванням рівностей (2.9)–(2.13) маємо наступне зображення:

$$
\begin{aligned}
u(x) = &\frac{1}{(k-1)!}\int_0^x (x-z)^{k-1} v(z)\,dz + \sum_{j=0}^{k-1} C_j \frac{x^j}{j!} = \\
=&\frac{1}{(k-1)!}\int_0^x (x-z)^{k-1} \sum_{i=0}^{s} V_i z^i\,dz \; + \sum_{j=0}^{k-1} C_j \frac{x^j}{j!} = \\
=&\sum_{i=0}^{s} V_i \int_0^x \frac{(x-z)^{k-1}}{(k-1)!} z^i\,dz + \sum_{j=0}^{k-1} C_j \frac{x^j}{j!} = \\
=&\sum_{i=0}^{s} V_i \frac{i!}{(i+k)!} x^{i+k} + C_j \frac{x^j}{j!} = \\
=&\sum_{i=0}^{s} \left(\sum_{j=0}^{s-i} \frac{(i+j)!}{(i+k)!} B_j P_{i+j} \right) x^{i+k} \; + \sum_{j=0}^{k-1} C_j \frac{x^j}{j!}.
\end{aligned}
$$

Таким чином, загальний поліноміальний розв'язок системи (2.2) має вигляд:

$$
u(x) = \sum_{i=0}^{s} \left(\sum_{j=0}^{s-i} \frac{(i+j)!}{(k+i)!} B_j P_{i+j} \right) x^{k+i} \; + \sum_{j=0}^{k-1} C_j \frac{x^j}{j!},
$$

де $C_j \in \mathbb{R}^n$ - довільні сталі вектори. Теорему доведено. □

З Теореми 2.1 випливає наступне твердження про розв'язність скалярного диференціального рівняння нескінченного порядку.

Наслідок 2.1. *Нехай* $P(x) = \sum\limits_{j=0}^{s} P_j x^j$, *де* $P_j \in \mathbb{R}$, $P_s \neq 0$, $a_j \in \mathbb{R}$ *та* $a_k \neq 0$ *для деякого* $k \in \mathbb{N}$. *Тоді загальний поліноміальний розв'язок рівняння* $\sum\limits_{j=k}^{\infty} a_j u^{(j)}(x) = P(x)$ *має вигляд:*

$$
u(x) = \sum_{i=0}^{s} \left(\sum_{j=0}^{s-i} \frac{(i+j)!}{(i+k)!} b_j P_{i+j} \right) x^{k+i} + \sum_{j=0}^{k-1} c_j \frac{x^j}{j!}. \tag{2.14}
$$

де c_i *— довільні сталі, а коефіцієнти* b_j *— визначаються з наступної системи:*

$$
b_j = \begin{cases} a_k^{-1}, & j = 0, \\ -a_k^{-1} \sum\limits_{i=1}^{j} a_{i+k} b_{j-i}, & j \in \mathbb{N}. \end{cases} \tag{2.15}
$$

2.1.1 Програмна реалізація методу

На основі Теореми 2.1 було програмно реалізовано метод `degenerate_Broggi_method()` для класу `InfiniteOrderDiffSystem`.

1. Знаходження індексу k

Першочергово для застосування Теореми 2.1 необхідно встановити індекс першої невиродженої матриці A_j. Для визначення цього індексу, програма знаходить всі цілі корені рівняння $\det A_j = 0$ відносно параметра j. Після чого шукає мінімальне натуральне число, яке не належить множині цілих розв'язків. Воно і стає індексом k. У програмі за це відповідає метод `first_non_degenerate_index()`:

```
def first_non_degenerate_index(self):

    det_expr = self.A_j.det()

    roots = sp.solve(det_expr, j)

    integer_roots = [r for r in roots if r.is_integer and r >= 0]

    d = 0
    while d in integer_roots:
        d += 1

    if self.A_func(d).det() == 0 and d == 0:
        return None

    return d
```

2. Перевірка умови застосовності методу

Під час виклику метода `degenerate_Broggi_method()` відбувається перевірка перших вироджених матриць на виконання умови $A_0 = A_1 = \cdots = A_{k-1} = 0$. За цю перевірку відповідає наступна частина коду:

```
def degenerate_Broggi_method(self):

    A_j = self.A_func
    s = self.degree
    n = self.dimension
    P = self.P
    k = self.first_non_degenerate_index()

```

```
    for j in range(k):
        if not np.array_equal(sp.matrix2numpy(A_j(j)),
                                            np.zeros_like(A_j(j))):
            return sp.Matrix(np.zeros((n,s+k)))
```

3. Формування загального розв'язку

Побудова загального розв'язку відбувається в методі degenerate_Broggi_method, який розділений на два етапи. Перший етап майже ідентичний алгоритму наведеному в Broggi_method. Єдина відмінність полягає в тому, що обчислення коефіцієнтів B_j відбувається на основі матриць A_{j+k}. Другий етап відповідає за формування коефіцієнтів розв'язку за формулою (2.3).

```
def degenerate_Broggi_method(self):
    #... Checking conditions and Initialisation of arrays A , B
                                                     , res , P_dx ...
    D = build_diff_matrix(s)

    for i in range(s):
        A[i, :, :] = np.array(A_j(i+k)).astype(np.float64)
        B[i, :, :] = find_inverse_coefficients(A, B, i, n)

    for i in range(s):
        for j in range(s-i):
            sol_system[:,i] += B[j, :, :] @ P[:,i+j] *
            math.factorial(i+j)/math.factorial(i+k)

    res = np.zeros((n,s+k))
    res[:,k:] =  sol_system

    c_vars = sp.Matrix([
              [sp.Symbol(f"c{j}{i+1}/{math.factorial(j)}")
              for i in range(n)] for j in range(k) ])
    output = sp.Matrix(res)
    output[:,:k] = sp.Matrix(c_vars).T
    return sp.nsimplify(output)
```

2.1.2 Приклади

Приклад 2.2. Розглянемо систему диференціальних рівнянь (1.1), де

$$A_j = \begin{pmatrix} 0 & j \\ j & 0 \end{pmatrix}, P(x) = \begin{pmatrix} x^2 \\ x \end{pmatrix}$$

Тут $k = 1$, а $s = 2$. Отже ми маємо тільки одну вироджену матрицю:

$$A_0 = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Знайдемо загальний поліноміальний розв'язок, користуючись Теоремою 2.1.

Обчислимо коефіцієнти B_j, з системи (2.4). Для цього запишемо всі необхідні матриці A_j:

$$A_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 0 & 2 \\ 2 & 0 \end{pmatrix}, \quad A_3 = \begin{pmatrix} 0 & 3 \\ 3 & 0 \end{pmatrix}.$$

Після цього отримаємо:

$$B_0 = A_1^{-1} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$B_1 = -A_1^{-1}(A_2 B_0) = \begin{pmatrix} 0 & -2 \\ -2 & 0 \end{pmatrix}$$

$$B_2 = -A_1^{-1}(A_2 B_1 + A_3 B_0) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Користуючись Теоремою 2.1, знайдемо розв'язок $u(x)$ (враховуючи, що $k =$

1):

$$
\begin{aligned}
u(x) = &\sum_{i=0}^{2} x^{1+i}\left(\sum_{j=0}^{2-i}\frac{(i+j)!}{(i+1)!}B_jP_{i+j}\right)+\sum_{j=0}^{0}C_j\frac{x^j}{j!}=\\
=&x\left(B_0P_0+B_1P_1+2B_2P_2\right)+x^2\left(\frac{1}{2}B_0P_1+B_1P_2\right)+x^3\left(\frac{1}{3}B_0P_2\right)+C_0=\\
=&x\left[\begin{pmatrix}0\\0\end{pmatrix}+\begin{pmatrix}-2\\0\end{pmatrix}+\begin{pmatrix}0\\2\end{pmatrix}\right]+x^2\left[\begin{pmatrix}1/2\\0\end{pmatrix}+\begin{pmatrix}0\\-2\end{pmatrix}\right]+x^3\begin{pmatrix}0\\1/3\end{pmatrix}+C_0=\\
=&x\begin{pmatrix}-2\\2\end{pmatrix}+x^2\begin{pmatrix}1/2\\-2\end{pmatrix}+x^3\begin{pmatrix}0\\1/3\end{pmatrix}+\begin{pmatrix}c_{01}\\c_{02}\end{pmatrix}=\begin{pmatrix}x^2/2-2x+c_{01}\\x^3/3-2x^2+2x+c_{02}\end{pmatrix},
\end{aligned}
$$

де $C_0=\begin{pmatrix}c_{01}\\c_{02}\end{pmatrix}$ — довільний вектор з $\mathbb{R}^2$.

В результаті отримали наступний загальний поліноміальний розв'язок системи диференціальних рівнянь (1.1):

$$
u(x)=\begin{pmatrix}\frac{x^2}{2}-2x+c_{01}\\\frac{x^3}{3}-2x^2+2x+c_{02}\end{pmatrix},
$$

де c_{01}, c_{02} – довільні дійсні сталі.

Вигляд загального розв'язку наданого програмою:

```
u_1 = c01 + x**2/2 - 2*x
u_2 = c02 + x**3/3 - 2*x**2 + 2*x
```

Приклад 2.3. Розглянемо диференціальне рівняння

$$
\sum_{j=0}^{\infty} ju^{(j)}(x)=x^2+3x+4. \tag{2.16}
$$

тут $k=1, s=2, a_j=j$. Знайдемо загальний поліноміальний розв'язок цього рівняння за Наслідком 2.1.

Знайдемо коефіцієнти b_j, $j = 0, 1, 2$ з системи (2.15), при $a_j = j$, $j \in \mathbb{N}$.

$$
\begin{aligned}
j = 0: &\quad b_0 = a_1^{-1} = 1\\
j = 1: &\quad b_1 = -a_1^{-1} a_2 b_0 = -2\\
j = 2: &\quad b_2 = -a_1^{-1}(a_2 b_1 + a_3 b_0) = 1
\end{aligned}
$$

Знайдемо за формулою (2.14) загальний поліноміальний розв'язок рівняння (2.16).

$$
u(x) = \sum_{i=0}^{2}\left(\sum_{j=0}^{2-i}\frac{(i+j)!}{(i+1)!}b_j P_{i+j}\right)x^{1+i} + c_{01} =
$$

$$
= c_{01} + \left(\sum_{j=0}^{2}\frac{(j)!}{(1)!}b_j P_j\right)x + \left(\sum_{j=0}^{1}\frac{(1+j)!}{(2)!}b_j P_{j+1}\right)x^2 +
$$

$$
+\left(\sum_{j=0}^{0}\frac{(2+j)!}{(3)!}b_j P_{j+2}\right)x^3 = c_{01} - \frac{1}{2}x^2 + \frac{1}{3}x^3,
$$

де $c_{01} \in \mathbb{R}$ - довільна стала.

Отже, отримали такий загальний поліноміальний розв'язок диференціального рівняння (2.16):

$$
u(x) = c_{01} - \frac{1}{2}x^2 + \frac{1}{3}x^3,
$$

де $c_{01} \in \mathbb{R}$ - довільна стала.

Вигляд загального розв'язку наданого програмою:

```
u_1 = c01 + x**3/3 - x**2/2
```

2.2 Метод невизначених коефіцієнтів

Як показує Приклад 1.1, рівняння (1.1) може взагалі не мати поліноміальних розв'язків. Наступна лема дає оцінку знизу степеню поліноміального розв'язку рівняння (1.1) через степінь правої частини.

Лема 2.1. *Нехай $u(x)$ – поліноміальний розв'язок системи* (1.1) *і* $\deg P(x) = s$. *Тоді степінь поліноміального розв'язку обмежений знизу:* $\deg u(x) \geq s$.

Доведення. Оскільки $\deg u^{(j)}(x) < \deg u(x)$ для $j \in \mathbb{N}$, то

$$s = \deg F(u(x)) \leq \deg u(x).$$

Лему доведено. $\square$

Наступна теорема показує, що задача знаходження поліноміальних розв'язків степеню не вище t зводиться до системи лінійних алгебраїчних рівнянь (СЛАР) відносно невідомих коефіцієнтів цього розв'язку.

Теорема 2.2. *Для того, щоб поліноміальна вектор-функція* $u(x) = \sum_{j=0}^{t} U_j x^j \in \mathbb{R}^n[x]$ *була поліноміальним розв'язком системи* (1.1) *при* $P(x) = \sum_{j=0}^{s} P_j x^j$, *необхідно і достатньо, щоб* $t \geq \deg u(x) \geq s = \deg P(x)$ *і векторні коефіцієнти* U_j $(j = 0, \ldots, t)$ *задовольняли наступну СЛАР:*

$$\begin{cases} \sum_{j=0}^{t-i} \frac{(i+j)!}{i!} A_j U_{i+j} = 0, & \text{для } i = s+1, \ldots, t. \\ \sum_{j=0}^{t-i} \frac{(i+j)!}{i!} A_j U_{i+j} = P_i, & \text{для } i = 0, \ldots, s. \end{cases} \tag{2.17}$$

Доведення. **Необхідність.**

За Лемою 2.1 $t \geq \deg u \geq s$. Розпишемо $F(u(x))$:

$$F(u(x)) = \sum_{j=0}^{\infty} A_j u^{(j)}(x) = P(x). \tag{2.18}$$

Розв'язок $u(x)$ — це поліном степеня $\deg u(x) \leq t$. Відповідно, $u^{(j)}(x) = 0$ для $j > t$. Тоді рівність (2.18) набуває наступного вигляду:

$$\sum_{j=0}^{t} A_j u^{(j)}(x) = P(x). \tag{2.19}$$

У випадку $t > s$ подамо поліном $P(x)$ у наступному вигляді:

$$P(x) = \sum_{j=0}^{t} P_j x^j,$$

позначаючи $P_j = 0$, для $j \in \mathbb{N},\ j = s+1, .., t$.

Тоді (2.19) переписується у наступному вигляді:

$$\sum_{j=0}^{t} A_j \left(\sum_{i=0}^{t} U_i x^i\right)^{(j)} = \sum_{i=0}^{t} P_i x^i. \tag{2.20}$$

Оскільки

$$\left(\sum_{i=0}^{t} U_i x^i\right)^{(j)} = \sum_{i=0}^{t-j} \frac{(i+j)!}{i!} U_{i+j} x^i,$$

то (2.20) переписується в вигляді

$$\sum_{j=0}^{t} A_j \sum_{i=0}^{t-j} \frac{(i+j)!}{i!} U_{i+j} x^i = \sum_{i=0}^{t} P_i x^i.$$

Змінимо в цій рівності порядок підсумування, одержимо наступну рівність:

$$\sum_{i=0}^{t} x^i \left(\sum_{j=0}^{t-i} \frac{(i+j)!}{i!} A_j U_{i+j}\right) = \sum_{i=0}^{t} P_i x^i. \tag{2.21}$$

Для виконання рівності (2.21) необхідно і достатньо, щоб коефіцієнти при однакових степенях x лівої та правої частини були рівні, тобто:

$$\sum_{j=0}^{t-i} \frac{(i+j)!}{i!} A_j U_{i+j} = P_i, \quad \text{для } i = 0, \ldots, t. \tag{2.22}$$

Таким чином, коефіцієнти U_j повинні задовольняти систему лінійних алгебраїчних рівнянь (2.17).

Достатність.

Нехай коефіцієнти U_i $(i = 0, \ldots, t)$ поліноміальної вектор-функції $u(x) = \sum\limits_{i=0}^{t} U_i x^i$ задовольняють систему (2.17). Тоді

$$\sum_{j=0}^{\infty} A_j \left(\sum_{i=0}^{t} U_i x^i\right)^{(j)} = \sum_{j=0}^{t} A_j \left(\sum_{i=0}^{t} U_i x^i\right)^{(j)} = \sum_{j=0}^{t} A_j \sum_{i=0}^{t-j} \frac{(i+j)!}{i!} U_{i+j} x^i =$$

$$= \sum_{i=0}^{t} x^i \left(\sum_{j=0}^{t-i} \frac{(i+j)!}{i!} A_j U_{i+j} \right).$$

Користуючись СЛАР (2.17), маємо:

$$\sum_{i=0}^{t} x^i \left(\sum_{j=0}^{t-i} \frac{(i+j)!}{i!} A_j U_{i+j} \right) = \sum_{i=0}^{t} P_i x^i$$

Таким чином $u(x)$ є поліноміальним розв'язком системи (1.1). □

2.2.1 Програмна реалізація методу

За програмну реалізацію методу невизначених коефіцієнтів відповідає метод `undetermined_coefficients_method()`, класу `InfiniteOrderDiffSystem`. Під час виклику цього метода, задається число $m \geq s$, для якого алгоритм буде шукати поліноміальні розв'язки системи (1.1) степені не вище m. Отже, з урахуванням Леми 2.1 цей метод знаходить поліноміальні розв'язки степенів $t = s, ..., m$.

1. Побудова системи лінійних алгебраїчних рівнянь.
Для знаходження невідомих коефіцієнтів U_j поліноміального розв'язку $u(x) = \sum_{j=0}^{t} U_j x^j$ системи (1.1) степеню t, на кожній ітерації циклу будується відповідна СЛАР (2.17). За побудову цієї системи відповідає наступна частина коду:

```
def undetermined_coefficients_method(self,m):
    s = self.degree - 1
    n = self.dimension
    A_func = self.A_func
    deg_max = m

    for t in range(s, deg_max + 1):
        P_current = self.P.T.tolist()

        try:
            u_vars = [
                [sp.Symbol(f"c{j}{i+1}") for i in range(n)] for j
                                                in range(t + 1)
            ]
            all_symbols = [sym for sublist in u_vars for sym
                                                in sublist]
```

```
        U = [sp.Matrix(u_vars[i]) for i in range(t + 1)]

        A = [A_func(j) for j in range(t + 1)]

        while len(P_current) <= t:
            P_current.append([0] * n)

        all_equations = []

        for k in range(t + 1):
            lhs_degree_k = -sp.Matrix(P_current[k])

            for j in range(t - k + 1):
                coeff = math.perm(k + j, j)
                lhs_degree_k += coeff * A[j] * U[k + j]

            all_equations.extend(lhs_degree_k[:])
```

2. Розв'язання СЛАР.

Для розв'язання СЛАР використовується функція `linsolve` з бібліотеки SymPy. Наприкінці кожної ітерації циклу програма перевіряє існування розв'язку для поточного степеня t (де $s \leq t \leq m$). Якщо розв'язок існує, функція формує та повертає відповідну матрицю коефіцієнтів. У випадку, якщо після перевірки всіх допустимих степенів до m включно розв'язку не знайдено, алгоритм завершує роботу та повертає `None`.

```
    solution_set = sp.linsolve(all_equations, all_symbols)

    if solution_set and solution_set != sp.EmptySet:
        solution = list(solution_set)[0]
        return sp.Matrix(
            [[solution[j * n + i] for j in range(t + 1)] for
                                                i in range(n)]
        )
    else:
        continue

except NonlinearError as e:
    continue
except Exception as e:
    continue
```

```
return None
```

Наприклад, для системи диференціальних рівнянь (1.1), де

$$A_j = \begin{pmatrix} j & e^j & \sin(j) & j^2 & \cos(j) \\ 0 & j-1 & e^j & \sin(j) & j^2 \\ 0 & 0 & j-2 & e^j & \sin(j) \\ 0 & 0 & 0 & j-3 & e^j \\ 0 & 0 & 0 & 0 & j-4 \end{pmatrix}, \; P(x) = \begin{pmatrix} 1+2x^2 \\ 3x \\ 4+x+x^2 \\ 5x^2 \\ 2+2x \end{pmatrix}.$$

програма видає всі поліноміальні розв'язки степеню не вище $m = 10$ наступного вигляду:

```
u_1 = c01 + 10*x**3/9 + x**2*(-301 + 216*sin(1) + 252*E)/72 +
+ x*(-7128*E - 6768*sin(1) + 216*cos(1) + 2592*sin(2) +
+ 2160*E*sin(1) + 6641 + 5616*exp(2))/432
u_2 = -4*x**2/3 - x*(41 + 120*sin(1) + 156*E)/36 - 6*exp(2) -
- 2699/432 - 10*sin(2)/3 + 65*sin(1)/36 + 131*E/36
u_3 = -4*x**2/3 - 5*E*x/3 + 67*x/36 - 5*exp(2)/3 - 1331/432 -
- sin(1)/4 + 16*E/9
u_4 = -5*x**2/3 + 37*x/18 - E/6 - 65/216
u_5 = -x/2 - 1/8
```

Результат підстановки знайденого розв'язку у систему надану програмою:

```
    p_1 = 2*x**2 + 1
    p_2 = 3*x
    p_3 = x**2 + x + 4
    p_4 = 5*x**2
    p_5 = 2*x + 2
```

2.2.2 Приклад ілюстрації роботи алгоритму

Приклад 2.4. Розглянемо систему диференціальних рівнянь (1.1), де

$$A_j = \begin{pmatrix} 1 & j(j-1) \\ j(j-1) & 0 \end{pmatrix}, \; P(x) = \begin{pmatrix} x \\ 1 \end{pmatrix}.$$

Опишемо поліноміальні розв'язки степеня не вище $m = 4$, вигляду:

$$u(x) = \sum_{j=0}^{4} U_j x^j, \text{де } U_j = \begin{pmatrix} u_{j1} \\ u_{j2} \end{pmatrix}.$$

Запишемо значення для перших п'яти матриць A_j:

$$A_0 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad A_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 1 & 2 \\ 2 & 0 \end{pmatrix}, \quad A_3 = \begin{pmatrix} 1 & 6 \\ 6 & 0 \end{pmatrix}, \quad A_4 = \begin{pmatrix} 1 & 12 \\ 12 & 0 \end{pmatrix}$$

Складаємо відповідну СЛАР (2.17) при $t = 1,\ s = 1$:

$$\begin{cases} A_0 U_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ A_0 U_0 + A_1 U_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{cases} \tag{2.23}$$

Розкриємо СЛАР (2.23):

$$\begin{cases} u_{11} = 1 \\ u_{11} + u_{01} = 0 \\ 0 = 1 \end{cases}$$

Для $\deg u(x) = 1$ ця система несумісна.

Складаємо відповідну СЛАР (2.17) при $t = 2,\ s = 1$:

$$\begin{cases} A_0 U_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ A_0 U_1 + 2A_1 U_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ A_0 U_0 + A_1 U_1 + 2A_2 U_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{cases} \tag{2.24}$$

Розкриємо СЛАР (2.24):

$$\begin{cases} u_{21} = 0 \\ 2u_{21} + u_{11} = 1 \\ 2u_{21} + 4u_{22} + u_{11} + u_{01} = 0 \\ 4u_{21} = 1 \end{cases}$$

Для $\deg u(x) = 2$ ця система несумісна.
Складаємо відповідну СЛАР (2.17) при $t = 3,\ s = 1$:

$$\begin{cases} A_0U_3 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ A_0U_2 + 3A_1U_3 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ A_0U_1 + 2A_1U_2 + 6A_2U_3 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ A_0U_0 + A_1U_1 + 2A_2U_2 + 6A_3U_3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{cases} \tag{2.25}$$

Розкриємо СЛАР (2.25):

$$\begin{cases} u_{31} = 0 \\ u_{21} + 3u_{31} = 0 \\ u_{11} + 2u_{21} + 6u_{31} + 12u_{32} = 1 \\ u_{01} + u_{11} + 2u_{21} + 4u_{22} + 6u_{31} + 36u_{32} = 0 \\ 4u_{21} + 36u_{31} = 1 \end{cases}$$

Для $\deg u(x) = 3$ ця система несумісна.
Складаємо відповідну СЛАР (2.17) при $t = 4,\ s = 1$:

$$\begin{cases} A_0U_4 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ A_0U_3 + 4A_1U_4 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ A_0U_2 + 3A_1U_3 + 12A_2U_4 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ A_0U_1 + 2A_1U_2 + 6A_2U_3 + 24A_3U_4 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ A_0U_0 + A_1U_1 + 2A_2U_2 + 6A_3U_3 + 24A_4U_4 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{cases} \tag{2.26}$$

Розкриємо СЛАР (2.26):

$$\begin{cases} u_{41} = 0 \\ u_{31} = 0 \\ u_{21} + 3u_{31} + 12u_{41} + 24u_{42} = 0 \\ u_{11} + 2u_{21} + 6u_{31} + 12u_{32} + 24u_{41} + 144u_{42} = 1 \\ u_{01} + u_{11} + 2u_{21} + 4u_{22} + 6u_{31} + 36u_{32} + 24u_{41} + 288u_{42} = 0 \\ 4u_{21} + 36u_{31} + 288u_{41} = 1 \end{cases}$$

Отже отримали наступні значення для невідомих коефіцієнтів:

$$\begin{cases} u_{41} = 0 \\ u_{42} = -\frac{1}{96} \\ u_{31} = 0 \\ u_{32} = c_{32} \\ u_{21} = \frac{1}{4} \\ u_{22} = c_{22} \\ u_{11} = 2 - 12c_{32} \\ u_{12} = c_{12} \\ u_{01} = \frac{1}{2} - 24c_{32} - 4c_{22} \\ u_{02} = c_{02} \end{cases},$$

де $c_{32}, c_{22}, c_{12}, c_{02}$ - довільні сталі.

Маємо наступний вигляд поліноміального розв'язку:

$$u(x) = \begin{pmatrix} \frac{x^2}{4} + (2 - 12c_{32})x + \frac{1}{2} - 24c_{32} - 4c_{22} \\ -\frac{x^4}{96} + c_{32}x^3 + c_{22}x^2 + c_{12}x + c_{02} \end{pmatrix}$$

Вигляд розв'язку наданого програмою:

```
u_1 = -4*c22 - 24*c32 + x^2/4 - 2*x*(6*c32 - 1) + 1/2
u_2 = c02 + c12*x + c22*x^2 + c32*x^3 - x^4/96
```

3 Висновки

Досліджено систему лінійних диференціальних рівнянь нескінченного порядку (1.1) в скінченновимірному просторі. Така система узагальнює так звані лінійні диференціально - алгебраїчні рівняння. Показано, що необхідною і достатньою умовою існування та єдиності розв'язку цієї системи при будь-якій поліноміальній правій частині є оборотність матричного коефіцієнту A_0. Для побудови відповідного єдиного розв'язку застосовується метод італійського математика У. Броджи. У випадку необоротності матричного коефіцієнту A_0 система (1.1) або може взагалі не мати поліноміального розв'язків, а якщо має, то нескінченну кількість розв'язків. В цьому випадку виникає задача побудови загального поліноміального розв'язку цієї системи. В роботі показано, що метод У. Броджи можна використати для побудови загального розв'язку системи (1.1) при виконанні умови $A_0 = A_1 = \cdots = A_{k-1} = 0,\ \det A_k \neq 0$ для деякого $k \in \mathbb{N}$. Зроблено програмну реалізацію цього методу в середовищі Python. В загальній ситуації при $\det A_0 = 0$ запропоновано метод невизначених коефіцієнтів, який зводить задачу знаходження поліноміальних розв'язків системи до розв'язання деякої системи лінійних алгебраїчних рівнянь. Створена програмна реалізація методу невизначенних коефіцієнтів дозволяє описати поліноміальні розв'язки степеню не вище заданого числа. Надано оцінку знизу степеню поліноміального розв'язку системи (1.1). Надалі для удосконалення методу невизначених коефіцієнтів планується оцінити зверху степінь поліноміального розв'язку цієї системи.

Список використаних джерел

[1] Фардигола Л.В. Курс звичайних диференціальних рівнянь. Навчальний посібник. Наукова думка, Київ. 2022.

[2] Kamke E. Differentialgleichungen Lösungsmethoden und Lösungen, Springer-Verlag. 2013.

[3] Гефтер, С. Л., Гончарук А. Б. Лінійне диференціальне рівняння з неоднорідністю у вигляді формального степеневого ряду над кільцем із неархімедовим нормуванням, Український математичний журнал. 2022. Т. 74(11). С. 1463–1477.

[4] Gefter S., Piven' A. Polynomial and copolynomial solutions of infinite order linear partial differential equations P.20-21. 7-th International Conference “Differential Equations and Control Theory” DECT 2025. Book of abstracts. P.20–21. https://appmath.karazin.ua/pdf/DECT2025/DECT2025_Book.pdf

[5] Brenan K.E., Campbell S.L., Petzold L.R. Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations. Revised and corrected reprint of the 1989 original, with an additional chapter and additional references. Classics in Applied Mathematics, 14. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA. 1996.

[6] Kunkel T., Mehrmann V. Differential-algebraic equations, analysis and numerical solution. European Mathematical Society. 2006.

[7] Gefter S., Stulova T. On Holomorphic Solutions of Some Implicit Linear Differential Equations in a Banach Space. In: Adamyan V.M. et al. (eds) Modern Analysis and Applications. Operator Theory: Advances and Applications, vol 191. Birkhauser Basel. 2009. P. 331–340.