

Київський національний університет імені Тараса Шевченка

Факультет комп'ютерних наук та кібернетики

ЗВІТ

з виконання завдання із некласичної оптимізації

Тема:

Диференціальна еволюція для параметричної оптимізації

Завдання 6

Виконав:

Кіщук Ярослав Ярославович

Викладач:

Джалладова Ірада Агаєвна

Зміст

| | | |
|----------|---|-----------|
| 1 | Постановка задачі | 2 |
| 2 | Математична модель | 2 |
| 3 | Алгоритм Диференціальної Еволюції | 2 |
| 3.1 | Загальний опис методу | 2 |
| 3.2 | Основні операції алгоритму | 3 |
| 3.2.1 | Мутація | 3 |
| 3.2.2 | Кросовер | 3 |
| 3.2.3 | Селекція | 3 |
| 3.3 | Налаштування параметрів | 3 |
| 4 | Програмна реалізація | 4 |
| 4.1 | Використані бібліотеки | 4 |
| 4.1.1 | NumPy | 4 |
| 4.1.2 | SciPy | 4 |
| 4.1.3 | Pyomo/Pyomoode | 4 |
| 4.2 | Структура коду | 5 |
| 5 | Результати експериментів | 5 |
| 5.1 | Візуалізація кінетичної моделі | 5 |
| 5.2 | Синтетичні дані | 6 |
| 5.3 | 3D візуалізація кінетичної моделі | 7 |
| 5.4 | Точність та збіжність | 9 |
| 5.5 | Швидкість збіжності | 10 |
| 5.6 | Відновлення параметрів | 10 |
| 6 | Висновки | 11 |

1 Постановка задачі

Метою роботи є застосування методу Диференціальної Еволюції (Differential Evolution, DE) для калібрування параметрів нелінійної регресійної моделі. Необхідно ідентифікувати 8 невідомих параметрів кінетичної моделі, що описує залежність швидкості хімічної реакції від температури та концентрації реагентів.

Основні етапи роботи:

1. Реалізація кінетичної моделі.
2. Генерація синтетичних експериментальних даних із накладанням шуму.
3. Побудова цільової функції (середньоквадратичне відхилення).
4. Застосування алгоритму DE з трьома різними стратегіями мутації.
5. Статистичний аналіз результатів за 50 незалежними запусками.

2 Математична модель

У роботі розглядається **узагальнена кінетична модель** швидкості реакції $r(T, C_A, C_B; \theta)$, яка поєднує типові елементи хімічної кінетики: експоненційну залежність від температури за законом Арреніуса, степеневі залежності від концентрацій реагентів та додатковий множник, що може описувати ефекти інгібування/активації. Модель не запозичена з конкретної публікації, а побудована як **синтетичний навчальний приклад** на основі типових формул хімічної кінетики для демонстрації роботи алгоритмів глобальної оптимізації.

Безрозмірна форма моделі має вигляд:

$$r = \theta_1 T^{\theta_2} \exp\left(-\frac{\theta_3}{RT}\right) C_A^{\theta_4} C_B^{\theta_5} (1 + \theta_6 C_A + \theta_7 C_B)^{\theta_8} \quad (1)$$

де:

- T — температура (K);
- C_A, C_B — концентрації реагентів;
- $R = 8.314$ Дж/(моль·K) — газова стала;
- $\theta = (\theta_1, \dots, \theta_8)$ — вектор невідомих параметрів.

Цільова функція $J(\theta)$ визначається як середньоквадратична помилка (MSE) між модельними значеннями та експериментальними даними:

$$J(\theta) = \frac{1}{N} \sum_{k=1}^N \left(r_{\text{model}}^{(k)} - r_{\text{exp}}^{(k)} \right)^2 \quad (2)$$

3 Алгоритм Диференціальної Еволюції

3.1 Загальний опис методу

Differential Evolution (DE) — це популяційний еволюційний алгоритм глобальної оптимізації для неперервних задач, запропонований Storn та Price у 1997 році. Алгоритм оперує популяцією з NP векторів-кандидатів у D -вимірному просторі:

$$\mathbf{x}_i^{(g)} = (x_{i,1}^{(g)}, \dots, x_{i,D}^{(g)}), \quad i = 1, \dots, NP$$

де g — номер покоління.

3.2 Основні операції алгоритму

Кожна ітерація DE складається з трьох основних етапів:

3.2.1 Мутація

Для кожного цільового вектора \mathbf{x}_i створюється донорний вектор \mathbf{v}_i за однією з стратегій:

- **DE/rand/1:** $\mathbf{v}_i = \mathbf{x}_{r1} + F(\mathbf{x}_{r2} - \mathbf{x}_{r3})$

Базова стратегія з випадковим вибором базового вектора. Забезпечує високу різноманітність популяції.

- **DE/best/1:** $\mathbf{v}_i = \mathbf{x}_{\text{best}} + F(\mathbf{x}_{r1} - \mathbf{x}_{r2})$

Використовує найкращий вектор як базовий. Забезпечує швидку збіжність, але може призводити до передчасного збігання.

- **DE/current-to-best/1:** $\mathbf{v}_i = \mathbf{x}_i + F(\mathbf{x}_{\text{best}} - \mathbf{x}_i) + F(\mathbf{x}_{r1} - \mathbf{x}_{r2})$

Комбінована стратегія, що балансує між експлорацією та експлуатацією.

Тут $F \in (0, 2)$ — коефіцієнт масштабування (у нашому випадку $F = 0.8$), а $r1, r2, r3$ — випадкові індекси, різні між собою та відмінні від i .

3.2.2 Кросовер

Формується випробувальний вектор \mathbf{u}_i шляхом біноміального схрещування:

$$u_{i,j} = \begin{cases} v_{i,j}, & \text{якщо } \text{rand}_j \leq CR \text{ або } j = j_{\text{rand}} \\ x_{i,j}, & \text{інакше} \end{cases}$$

де $CR \in [0, 1]$ — ймовірність схрещування (у нашому випадку $CR = 0.9$), j_{rand} — випадково обраний індекс, що гарантує зміну хоча б однієї координати.

3.2.3 Селекція

Застосовується жадібна селекція:

$$\mathbf{x}_i^{(g+1)} = \begin{cases} \mathbf{u}_i, & \text{якщо } f(\mathbf{u}_i) \leq f(\mathbf{x}_i^{(g)}) \\ \mathbf{x}_i^{(g)}, & \text{інакше} \end{cases}$$

Це забезпечує монотонне покращення якості популяції.

3.3 Налаштування параметрів

Для даної задачі використано наступні параметри:

- Розмір популяції: $NP = 48 \approx 50$ (близько до рекомендованого $5D$ до $10D$)
- Коефіцієнт масштабування: $F = 0.8$
- Ймовірність схрещування: $CR = 0.9$
- Критерій зупинки: 200 поколінь

4 Програмна реалізація

4.1 Використані бібліотеки

Реалізацію виконано мовою Python 3.x з використанням наступних бібліотек:

4.1.1 NumPy

Бібліотека для ефективних векторних та матричних обчислень. Використовувалась для:

- Генерації сіток даних через `np.meshgrid`
- Векторизованих обчислень кінетичної моделі
- Операцій з масивами параметрів

4.1.2 SciPy

Функція `scipy.optimize.differential_evolution` надає готову реалізацію DE з підтримкою різних стратегій мутації. Основні параметри:

```
differential_evolution(  
    func,          # цільова функція  
    bounds,        # межі параметрів  
    strategy,       # стратегія мутації  
    maxiter,        # максимальна кількість поколінь  
    popsize,        # множник розміру популяції  
    mutation,       # коефіцієнт F  
    recombination, # ймовірність CR  
    seed,           # seed для відтворюваності  
    polish=False    # вимкнути локальну оптимізацію  
)
```

У SciPy розмір популяції визначається як $NP = \text{popsize} \times D$. Для отримання $NP = 48$ при $D = 8$ використано `popsize = 6`.

4.1.3 Pymoo/PymooDE

Бібліотека `pymoo` — потужний фреймворк для багатокритеріальної оптимізації з підтримкою різних еволюційних алгоритмів. Для DE використано модуль `pymooDE`, що надає розширену реалізацію з підтримкою стратегії `DE/current-to-best/1/bin`.

Реалізація через клас задачі:

```
class KineticProblem(ElementwiseProblem):  
    def __init__(self):  
        super().__init__(  
            n_var=8,  
            n_obj=1,  
            n_constr=0,  
            xl=lower_bounds,  
            xu=upper_bounds  
        )  
  
    def _evaluate(self, x, out, *args, **kwargs):  
        out["F"] = objective_function(x)
```

Запуск оптимізації:

```
algorithm = DE(  
    pop_size=48,  
    variant="DE/current-to-best/1/bin",  
    F=0.8,  
    CR=0.9  
)  
  
res = minimize(  
    problem,  
    algorithm,  
    termination,  
    seed=seed,  
    verbose=False  
)
```

4.2 Структура коду

Програма організована в наступні основні компоненти:

1. **Модель:** Функція `rate_model()` обчислює швидкість реакції для заданих параметрів
2. **Цільова функція:** Функція `objective()` обчислює MSE між модельними та експериментальними даними
3. **Обгортки DE:** Функції для запуску різних варіантів DE з уніфікованим інтерфейсом
4. **Цикл експериментів:** 50 незалежних запусків для кожної стратегії з різними `seed`
5. **Аналіз результатів:** Збір статистики, побудова графіків, порівняння параметрів

5 Результати експериментів

Було проведено 50 незалежних запусків для кожної стратегії. Загалом використано 1000 експериментальних точок, згенерованих з істинної моделі з накладанням 5% шуму.

5.1 Візуалізація кінетичної моделі

Для кращого розуміння складності оптимізаційної задачі, на рис. 1 показано тривимірну поверхню кінетичної функції $r(T, C_A)$ при фіксованому значенні $C_B = 1.0$. Графік демонструє сильну нелінійність моделі та експоненціальне зростання швидкості реакції зі збільшенням температури.

Поверхня $r(T, C_A)$ при $C_B=1.0$

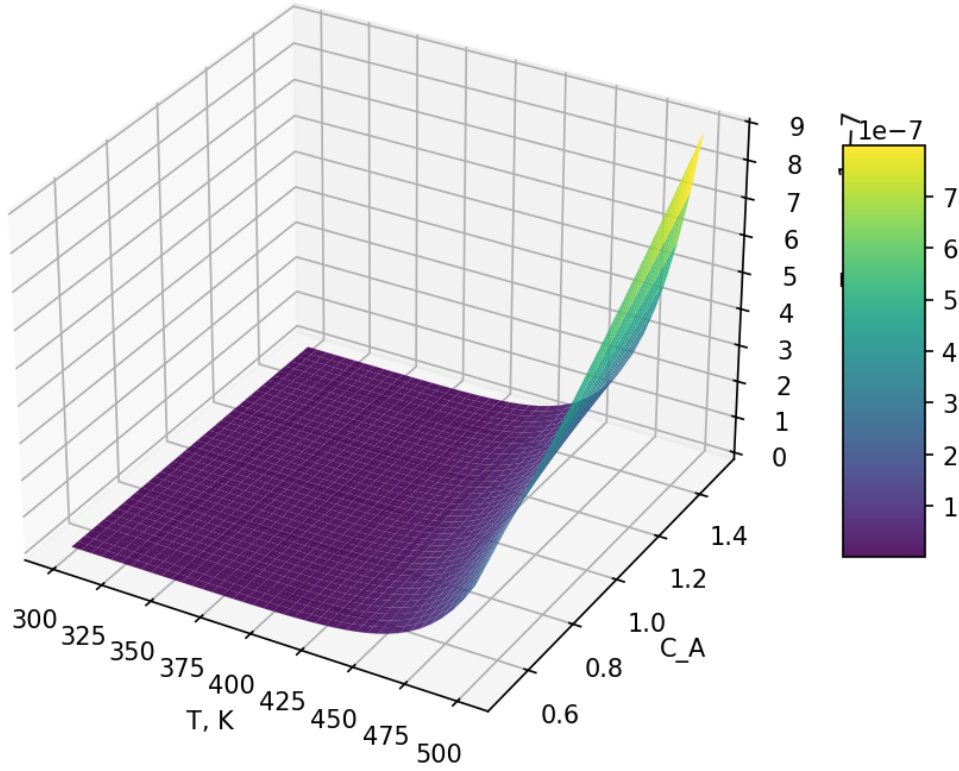


Рис. 1: 3D поверхня кінетичної функції $r(T, C_A)$ при $C_B = 1.0$

5.2 Синтетичні дані

Для тестування алгоритму DE використано метод синтетичних (штучних) даних. Цей підхід дозволяє контролювати істинні значення параметрів та оцінити якість їх відновлення.

Процес генерації даних:

1. Задано вектор **істинних параметрів** $\theta^{\text{true}} = (\theta_1 = 5.0, \theta_2 = 0.5, \dots, \theta_8 = 2.0)$.
2. Для сітки значень $T \in [300, 500]$ K, $C_A \in [0.5, 1.5]$, $C_B \in [0.4, 1.2]$ обчислено **істинні значення** швидкості реакції:

$$r_k^{\text{true}} = r(T_k, C_{A,k}, C_{B,k}; \theta^{\text{true}})$$

3. До істинних значень додано **гаусівський шум** з рівнем 5% для імітації експериментальних похибок:

$$r_k^{\text{exp}} = r_k^{\text{true}} \cdot (1 + \varepsilon_k), \quad \varepsilon_k \sim \mathcal{N}(0, 0.05^2)$$

Такий підхід моделює реальну ситуацію, коли дослідник має лише зашумлені експериментальні виміри та не знає точних значень параметрів моделі. Мета алгоритму оптимізації — відновити параметри θ , використовуючи тільки експериментальні дані r_k^{exp} .

На рис. 2 показано згенеровані експериментальні дані у порівнянні з істинними значеннями моделі. Помаранчеві точки представляють ідеальні значення без шуму, тоді як сині точки — експериментальні дані з накладеним випадковим шумом. Видно, що дані містять розкид навколо істинних значень, що ускладнює задачу ідентифікації параметрів та перевіряє робастність алгоритму DE.



Рис. 2: Порівняння експериментальних даних з істинними значеннями моделі. Помаранчеві точки — істинні значення, сині — експериментальні дані з 5% шумом.

5.3 3D візуалізація кінетичної моделі

Для кращого розуміння поведінки кінетичної моделі створено серію 3D візуалізацій з фіксуванням різних параметрів. Ці графіки дозволяють проаналізувати залежність швидкості реакції від різних комбінацій вхідних змінних.

На рис. 3 показано залежність швидкості реакції від температури T та концентрації реагенту B (C_B) при фіксованій концентрації $C_A = 1.0$. Спостерігається експоненціальне зростання швидкості реакції з підвищенням температури, що відповідає закону Арреніуса.

Поверхня $r(T, C_B)$ при $C_A=1.0$

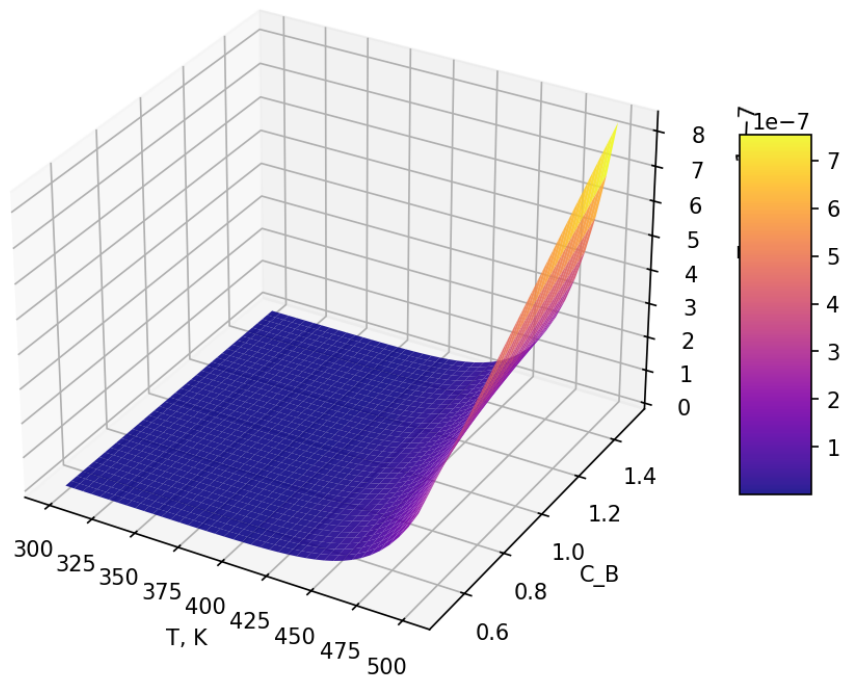


Рис. 3: 3D поверхня $r(T, C_B)$ при фіксованому $C_A = 1.0$

На рис. 4 представлено залежність від концентрацій обох реагентів при фіксованій температурі $T = 400$ К. Ця візуалізація демонструє взаємний вплив концентрацій на швидкість реакції.

Поверхня $r(C_A, C_B)$ при $T=400$ К

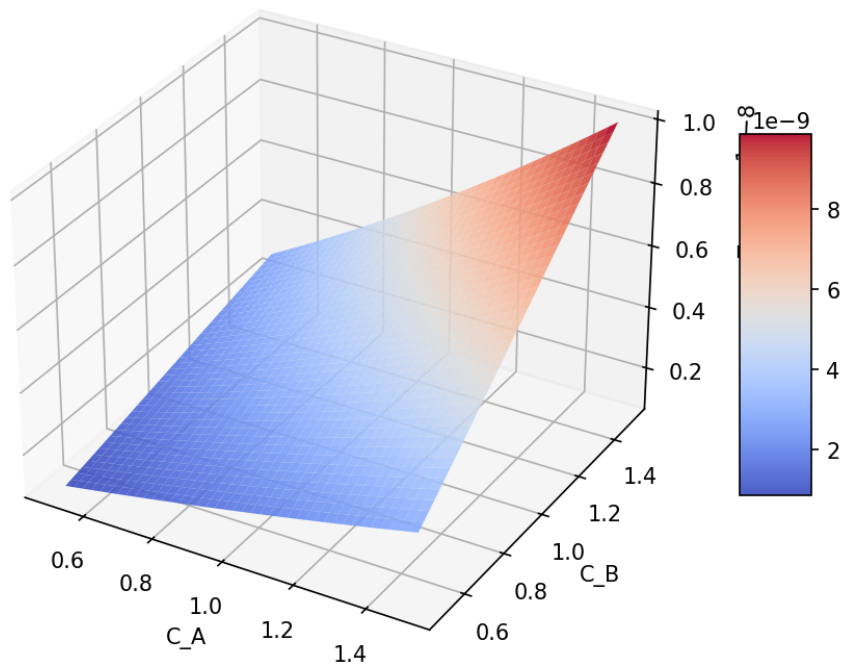


Рис. 4: 3D поверхня $r(C_A, C_B)$ при фіксованій температурі $T = 400$ К

На рис. 5 показано комбіновану візуалізацію з чотирма підграфіками, що ілюструють поведінку моделі при різних фіксованих параметрах. Це дозволяє порівняти вплив кожного параметра на форму поверхні відгуку.

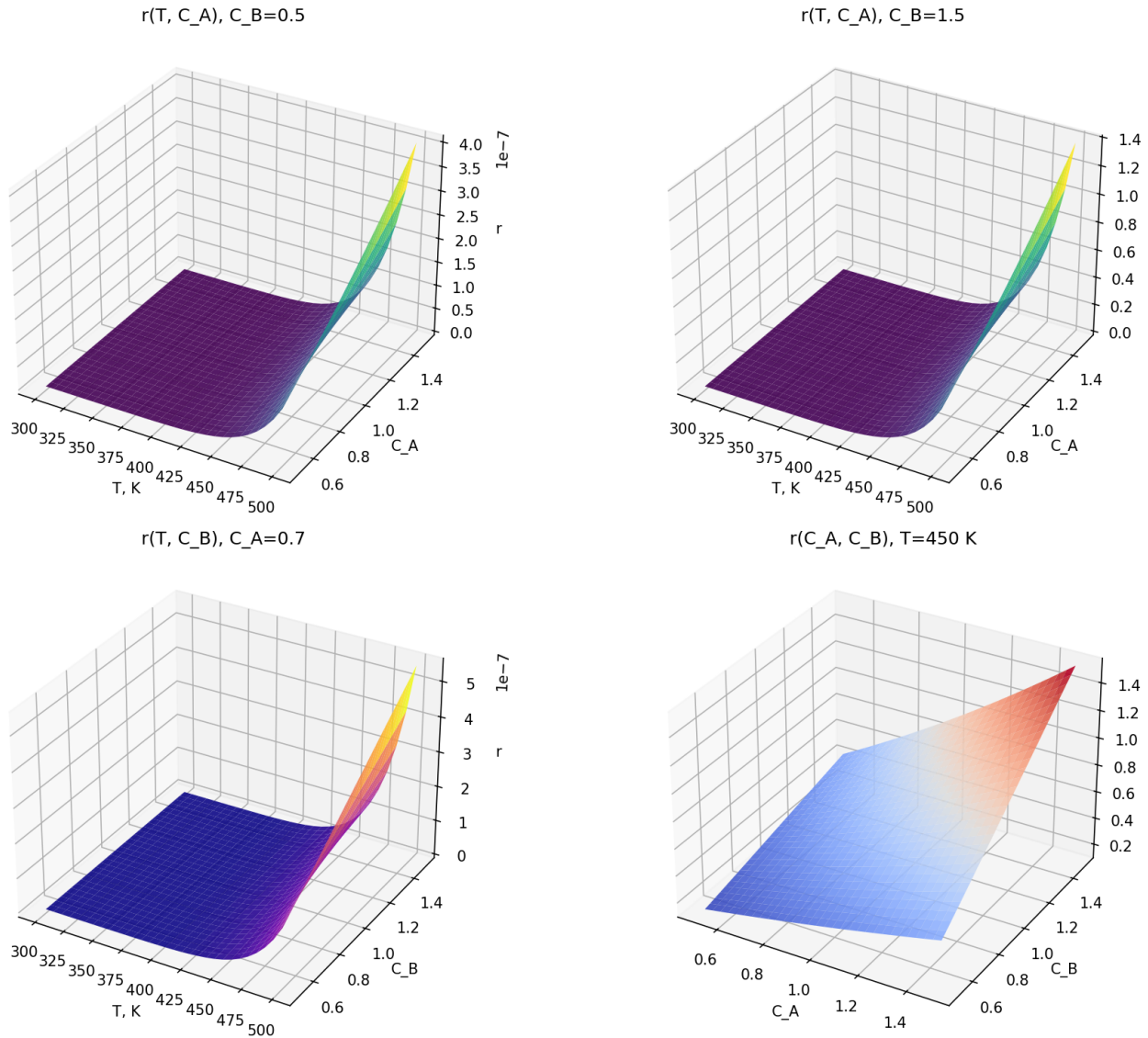


Рис. 5: Комбінована візуалізація кінетичної моделі: (верхній ряд) $r(T, C_A)$ при $C_B = 0.5$ та $C_B = 1.5$; (нижній ряд) $r(T, C_B)$ при $C_A = 0.7$ та $r(C_A, C_B)$ при $T = 450$ K

5.4 Точність та збіжність

Порівняння середнього значення функції помилки (MSE) та стандартного відхилення:

| Стратегія | Mean MSE | Std Dev |
|----------------------|-------------------------|-------------------------|
| DE/rand/1/bin | 9.146×10^{-17} | 2.781×10^{-17} |
| DE/best/1/bin | 5.400×10^{-17} | 1.421×10^{-18} |
| DE/current-to-best/1 | 5.421×10^{-17} | 2.215×10^{-18} |

Табл. 1: Статистика помилки за 50 запусків

На рис. 6 наведено діаграму розмаху (boxplot) для порівняння розподілів MSE. Видно, що стратегії **DE/best/1** та **DE/current-to-best/1** показують кращу стабільність і нижчі

значення помилки порівняно з **DE/rand/1**.

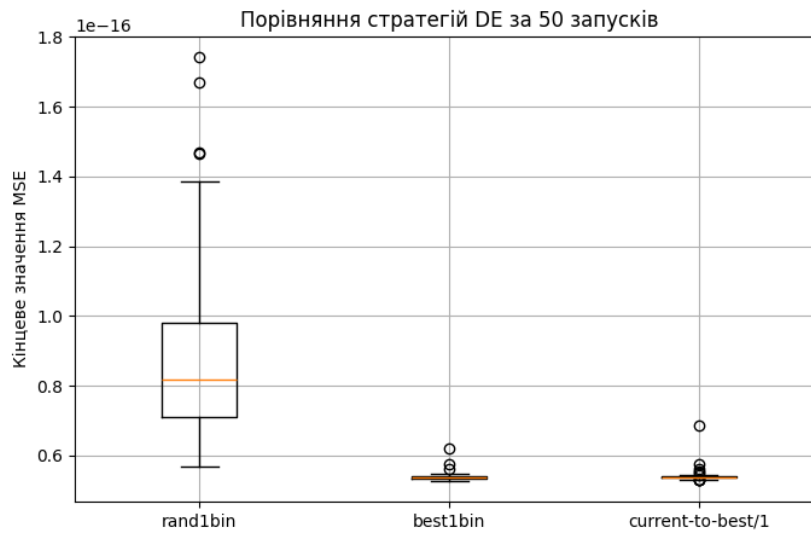


Рис. 6: Порівняння розподілу MSE для трьох стратегій

5.5 Швидкість збіжності

На рис. 7 показано динаміку збіжності для одного характерного запуску кожної стратегії. Графік демонструє, що стратегії **DE/best/1** та **DE/current-to-best/1** досягають оптимуму швидше, ніж **DE/rand/1**.

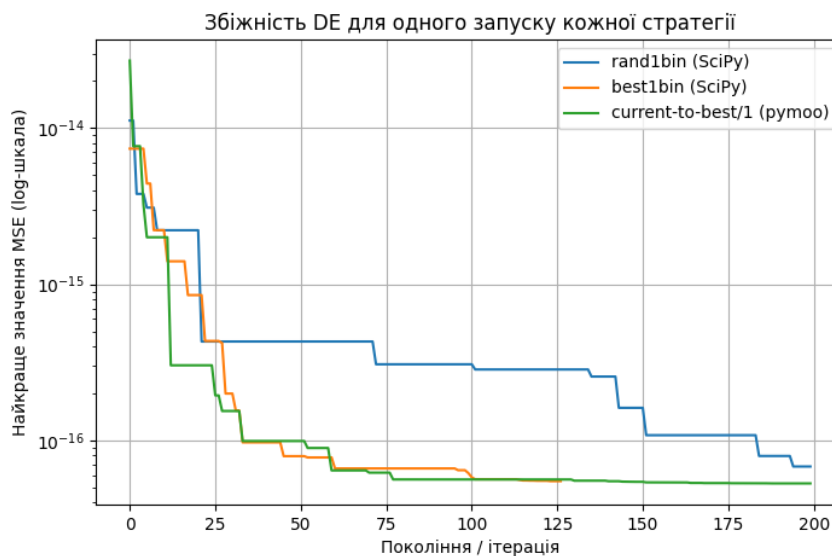


Рис. 7: Збіжність алгоритмів по поколіннях

5.6 Відновлення параметрів

Порівняння знайдених параметрів з істинними значеннями для найкращого запуску кожної стратегії:

| Параметр | Істинне | DE/rand/1 | DE/best/1 | DE/curr-to-best |
|------------|---------|-----------|-----------|-----------------|
| θ_1 | 5.0 | 841.26 | 687.28 | 429.06 |
| θ_2 | 0.5 | -0.250 | -0.115 | -0.060 |
| θ_3 | 80000.0 | 82276.5 | 83651.4 | 83214.9 |
| θ_4 | 1.0 | 1.043 | 1.120 | 1.161 |
| θ_5 | 1.2 | 1.095 | 1.245 | 1.192 |
| θ_6 | 0.1 | 1.253 | 0.015 | 0.021 |
| θ_7 | -0.05 | -0.047 | -0.044 | -0.405 |
| θ_8 | 2.0 | 0.231 | 3.728 | 0.166 |

Табл. 2: Порівняння параметрів

Найкращі значення MSE для кожної стратегії:

- DE/rand/1: 5.675×10^{-17}
- DE/best/1: 5.290×10^{-17}
- DE/current-to-best/1: 5.296×10^{-17}

6 Висновки

В ході роботи було успішно реалізовано та протестовано алгоритм Диференціальної Еволюції для задачі ідентифікації параметрів кінетичної моделі з 8 невідомими параметрами.

Основні результати дослідження:

1. **Точність оптимізації:** Всі три стратегії DE досягли дуже низьких значень MSE (порядку 10^{-17}), що свідчить про успішну мінімізацію цільової функції.
2. **Порівняння стратегій:**
 - **DE/best/1** показала найкращі результати за середнім значенням MSE (5.400×10^{-17}) та найменшу дисперсію результатів, що свідчить про високу надійність цієї стратегії.
 - **DE/current-to-best/1** продемонструвала результати, порівнянні з DE/best/1, з MSE 5.421×10^{-17} та дещо більшою варіативністю.
 - **DE/rand/1** показала найгірші результати (MSE 9.146×10^{-17}) з найбільшою дисперсією, що підтверджує більшу стохастичність базової стратегії.
3. **Швидкість збіжності:** Графіки збіжності показують, що стратегії на основі найкращого індивіда (DE/best/1 та DE/current-to-best/1) досягають оптимуму швидше, ніж випадкова стратегія DE/rand/1.
4. **Відновлення параметрів:** Незважаючи на дуже низьке значення MSE, деякі параметри моделі були відновлені з великими відхиленнями від істинних значень (наприклад, θ_1 , θ_2 , θ_8). Це вказує на наявність сильної кореляції між параметрами та існування множини локально-еквівалентних рішень, які дають практично однакові прогнози моделі при різних комбінаціях параметрів.

Практичні висновки:

- Для задач параметричної оптимізації рекомендується використовувати стратегії **DE/best/1** або **DE/current-to-best/1** через їх кращу збіжність та стабільність.

- Низьке значення цільової функції не завжди гарантує точне відновлення параметрів моделі у присутності кореляцій між параметрами.
- Налаштування гіперпараметрів ($NP = 48$, $F = 0.8$, $CR = 0.9$) виявилися ефективними для даної задачі розмірності $D = 8$.