Алгоритм будує обернену матрицю за допомогою LU-розкладання, застосовуючи алгоритм Штрассена для оптимізованого множення матриць. Це надійний метод для квадратних, невироджених матриць і пропонує переваги у продуктивності для великих матриць, балансуючи між складністю між $O(n^3)$ і O(n^2.81).

Опис алгоритму:

Алгоритм побудови оберненої матриці за допомогою LU-розкладу базується на розкладанні початкової квадратної матриці A на дві трикутні матриці L (нижня трикутна) та U (верхня трикутна), де $A = L \cdot U$. Після цього для кожного стовпця одиничної матриці вирішується система рівнянь $L \cdot Y = E_i$ (пряма підстановка) та $U \cdot X = Y$ (зворотна підстановка). Отримані вектори X формують стовпці оберненої матриці A^-1.

Для підвищення ефективності алгоритм використовує алгоритм Штрассена для множення матриць, що знижує кількість операцій порівняно зі стандартним методом.

Кроки алгоритму:

1. **Введення розмірності та елементів матриці**
   - На першому етапі користувач вводить розмірність квадратної матриці n та її елементи. Якщо розмірність матриці n є непозитивною (тобто $n \leq 0$), алгоритм завершує виконання.

2. **Доповнення розмірності до степеня двійки (за потреби)**
   - Щоб забезпечити коректну роботу алгоритму Штрассена, розмірність матриці може бути доповнена до найближчого більшого степеня двійки. Це дозволяє уникнути проблем з ефективністю під час множення. У випадку доповнення додаються нульові елементи, а після виконання основних операцій зайві елементи видаляються.

3. **LU-розкладання**
   - Мета: Розкласти початкову матрицю A на добуток двох матриць: нижньої трикутної L та верхньої трикутної U, де $A = L \cdot U$.
   - Процес:
     - Матриця L має одиниці на головній діагоналі.
     - Під час розкладання елементи діагоналі матриці U не повинні бути нульовими, щоб уникнути виродженості.
     - На кожному кроці виконується підстановка для обчислення елементів матриць L та U з заданих елементів початкової матриці A.

4. Використання алгоритму Штрассена для множення матриць
   - Мета: Прискорити множення матриць у процесі обчислень.
   - Алгоритм Штрассена:
     - Виконує розбиття матриці на підматриці розміром 2 на 2.
     - Замість восьми операцій множення, алгоритм виконує сім, що знижує асимптотичну складність множення.

- У випадку невеликих матриць (наприклад, 2 на 2) використовується стандартне множення, що є більш оптимальним для таких розмірів.

5. **Знаходження оберненої матриці через систему підстановок**
   - **Крок 1**: Розв'язання системи рівнянь L・Y = E_i, де E_i є i-тим стовпцем одиничної матриці. Цей етап називається **прямою підстановкою** і знаходить вектор Y.
   - **Крок 2**: Розв'язання системи рівнянь U・X = Y для отримання вектора X. Цей етап називається **зворотною підстановкою**.
   - Всі отримані вектори X формують стовпці оберненої матриці A^-1.

Приклад роботи алгоритму (умовний):
1. Дано квадратну матрицю A.
2. Виконується LU-розкладання, отримуючи матриці L та U.
3. Для кожного стовпця одиничної матриці:
   - Виконується пряма підстановка L・Y = E_i
   - Виконується зворотна підстановка U・X = Y
   - Вектор X додається до оберненої матриці.

Додаткові аспекти:
- Алгоритм Штрассена застосовується лише до великих матриць для прискорення обчислень.
- Перевіряється, чи не є елементи діагоналі матриці U близькими до нуля, що є важливим для уникнення чисельних похибок.

Висновок:
Алгоритм обчислення оберненої матриці методом LU-розкладу з використанням алгоритму Штрассена дозволяє ефективно працювати з великими матрицями, забезпечуючи значний приріст продуктивності для розмірів, що є степенем двійки.

Сфера застосування:

Метод LU-розкладу широко застосовується для розв'язання систем лінійних рівнянь, обчислення визначників, обернення квадратних матриць. Його ефективність зростає для великих матриць, особливо коли розмір є степенем двійки. Використання алгоритму Штрассена дозволяє прискорити обчислення для великих розмірів матриць.

Оцінювання складності:

1. LU-розкладання:
   - Стандартна складність для квадратної матриці розміром \(n \times n \) становить $O(n^3)$.

2. Алгоритм Штрассена:
   - Має складність приблизно O(n^log2 7)≈O(n^2.81), що значно швидше для великих розмірів.

3. Підстановки:
   - Пряма та зворотна підстановка мають лінійну складність $O(n^2)$.

Загальна оцінка:

Загальна складність обчислення оберненої матриці із використанням алгоритму Штрассена наближається до $O(n^{2.81})$. У випадку класичного множення загальна складність дорівнює $O(n^3)$. Алгоритм ефективний для великих матриць і демонструє виграш у швидкодії завдяки алгоритму Штрассена.