



Implementacja różnych metod rozwiązywania układów równań liniowych

IMPLEMENTACJA SEKWENCYJNA, WSPÓŁBIEŻNA ORAZ RÓWNOLEGŁA

Cel projektu

- ▶ Celem projektu jest implementacja oraz analiza wydajności różnych metod rozwiązywania układów równań liniowych w wersji sekwencyjnej, współbieżnej i równoległej, w celu określenia najbardziej efektywnego podejścia pod kątem szybkości i skalowalności.

Wstęp teoretyczny

- ▶ **Metoda eliminacji Gaussa-Jordana:**

Przekształca macierz do postaci jednostkowej ($O(n^3)$) oraz umożliwia równoległość przy przetwarzaniu wierszy

- ▶ **Metoda Gaussa-Seidela:**

Jest to metoda iteracyjna, która szybciej konverguje dla układów rzadkich ($O(n^2)$). Metoda ta umożliwia zrównoleglenia dla różnych wierszy w iteracji

- ▶ **Metoda Jacobiego:**

Jest to również metoda iteracyjna z tą różnicą, że ma inny sposób aktualizacji wartości zmiennych w kolejnych iteracjach (na koniec iteracji, gdzie metoda Gaussa-Seidela robi to od razu)

Obszar stosowalności

► Obliczenia inżynierskie i naukowe

- ◆ **Symulacje fizyczne** – np. w mechanice płynów, analizie strukturalnej, modelowaniu zachowań materiałów
- ◆ **Analiza obwodów elektrycznych** – układy równań występują w analizie sieci elektrycznych
- ◆ **Mechanika i inżynieria konstrukcyjna** – analiza naprężeń i odkształceń

► Uczenie maszynowe i analiza danych

- ◆ **Przetwarzanie danych wielowymiarowych** – metody numeryczne są wykorzystywane np. w analizie regresji i modelach predykcyjnych
- ◆ **Sztuczna inteligencja (AI)** – np. w algorytmach związanych z optymalizacją i trenowaniem modeli
- ◆ **Wizja komputerowa i grafika 3D** – np. transformacje geometryczne, analiza obrazu

► Grafika komputerowa i przetwarzanie obrazu

- ◆ **Renderowanie scen 3D** – operacje macierzowe są wykorzystywane do transformacji obiektów
- ◆ **Efekty wizualne i filtrowanie obrazu** – np. filtry wideo i przekształcenia przestrzenne

► Kryptografia i kompresja danych

Koncepcja zrównoleglenia

Każda metoda będzie miała trzy wersje:

- ▶ Sekwencyjna (C++) – podstawowa implementacja
- ▶ Współbieżna (`std::thread`) – przydział obliczeń do kilku wątków
- ▶ Równoległa (OpenMP) – podział operacji na wiele rdzeni

Które metody lepiej nadają się do równoległości?

- ▶ Gauss-Jordan średnio nadaje się do równoległości (zależności między wierszami)
- ▶ Gauss-Seidel lepiej nadaje się do współbieżności (ale wymaga synchronizacji)
- ▶ Jacobi idealnie nadaje się do równoległości, bo iteracje są niezależne

Wstępne pomiary dla implementacji sekwencyjnej

Testy wydajnościowe (dla sekwencyjnej wersji w C++):

Metoda	Wymiar (n)	Czas obliczeń (sekundy)
Gauss-Jordan	1000	1.5231
Gauss-Seidel	1000	0.7320
Jacobi	1000	0.6543

Obserwacje:

- Gauss-Jordan rośnie najszybciej ($O(n^3)$) → idealny do dużych macierzy
- Gauss-Seidel i Jacobi lepsze dla dużych układów → mogą szybciej konwergować

Założenia projektowe

- ▶ Zaimplementowanie trzech metod numerycznych do rozwiązywania układów równań liniowych.
- ▶ Pomiar czasu wykonania dla różnych rozmiarów macierzy
- ▶ Analiza skalowalności w zależności od liczby wątków
- ▶ Wykresy i porównanie wydajności implementacji

1. Cel końcowy:
Celem końcowym projektu jest znalezienie najbardziej efektywnej metody rozwiązywania układów równań liniowych pod względem wydajności i możliwości zrównoleglenia
2. Odpowiedź na kluczowe pytania:
 - Która metoda jest najszybsza dla dużych układów równań?
 - Czy równoległość rzeczywiście przyspiesza każdą metodę?
 - Ile wątków daje największe przyspieszenie, a kiedy osiągamy limit skalowalności?



Dziękuję za uwagę

Wykonał: Dawid Garncarek