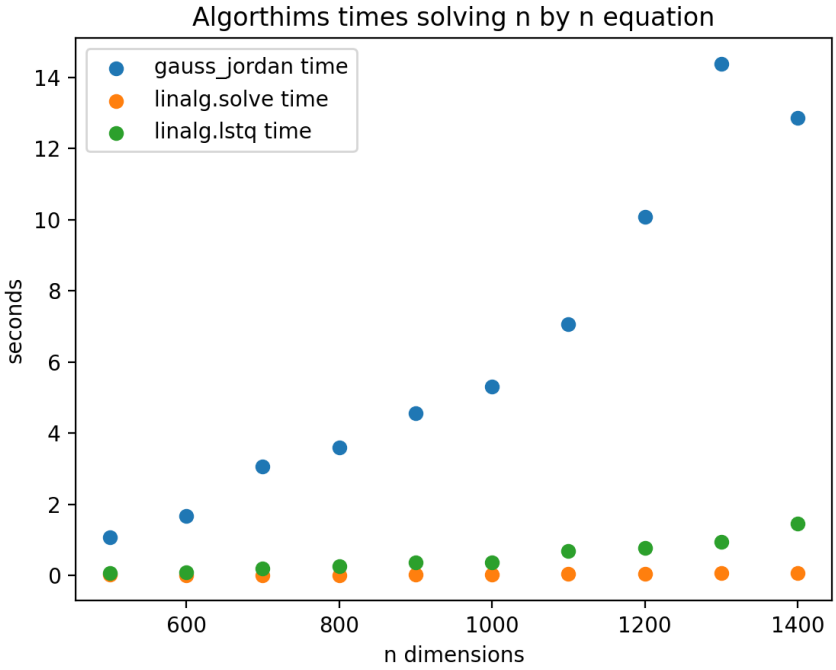
Metody obliczeniowe w nauce i technice
Laboratorium 2
Rozwiązywanie układów równań liniowych

Adam Dyda

Kwiecień 2021

1 Metoda Gaussa-Jordana

Pierwszym zadaniem było zaimplementowanie funkcji rozwiązującej układ równań liniowych metodą Gaussa-Jordana z częściowym poszukiwaniem elementu wiodącego. Zaimplementowaną funkcje porównałem z funkcjami bibliotecznymi z pakietu numpy linalg.solve oraz linalg.lstq wyniki widać na poniższym wykresie, jak można było się spodziewać własna implementacja jest zdecydowanie wolniejsza od funkcji bibliotecznych.



2 Faktoryzacja LU

W tej cześci zająłem się implementacją funkcji dokonującej faktoryzacji LU macierzy bez poszukiwania elementu wiodącego, poprawność działania funkcji sprawdziłem obliczając $||A - LU||$ i porównując wynik do zera. Funkcja działa poprawnie i co warto dodać implementacja funkcji działa w miejscu czyli bez wykorzystywania dodatkowej pamięci.

3 Analiza obwodu elektrycznego

W ostatniej cześci zająłem sie napisaniem programu który znajduje natężenia prądu w obwodzie elektrycznym przy danej rezystancji każdej gałęzi oraz przyłożonej sile elektromotorycznej E. Do rozwiązywania wykorzystałem metode praw Kirchhoffa oraz/i metode potencjałów węzłowych oraz dokonałem wizualizacji wynikow. Warto dodać ze przy implementacji bardzo przydatna okazała się biblioteka networkx.

3.1 Metoda praw Kirchhoffa

Na początku algorytmu znajduje cykle proste w grafie, ja robiłem to za pomocą funkcji nx.cycle_basis z biblioteki networkx. Następnie dla każdego cyklu przechodze po gałęziach znajdujących się w nim, i wykorzystując drugie prawo Kirchhoffa zapisuje odpowiednie równania do macierzy którą będe rozwiązywał oraz ustalam kierunek płynącego prądu. Kolejnym etapem jest przejście po wszystkich węzłach obwodu i za pomocą pierwszego prawa kirchhofa zapisanie odpowiednich równań do macierzy. Na koniec rozwiązuje równanie macierzowe i otrzymuje w wyniku wektor prądów dla odpowiednich gałęzi.

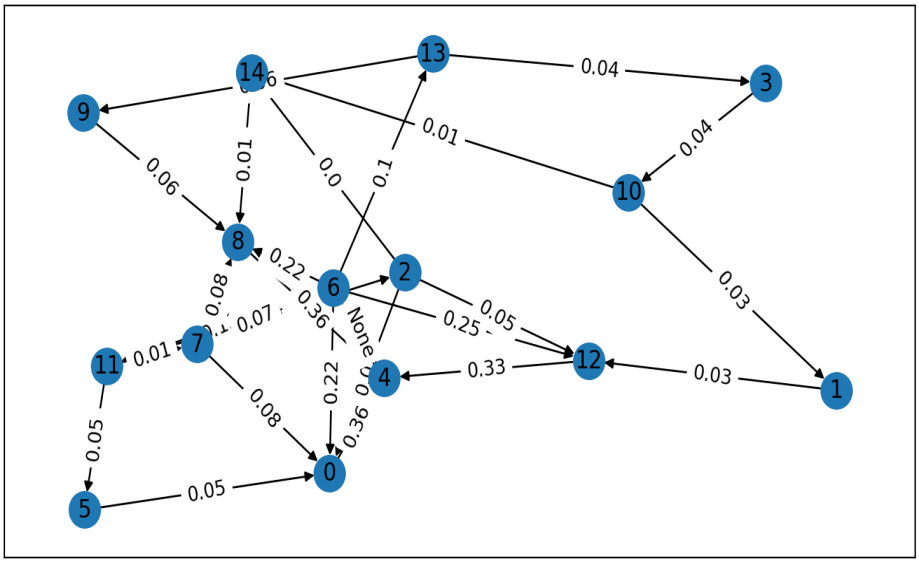
3.2 Testy i poprawnośc rozwiązania

Wykorzystując funkcję do generowania grafów wymaganych w treści zadania przeprowadziłem testy i postarałem się sprawdzić poprawnośc rozwiązania poprzez porównanie wyników obu metod. Jednak wyniki nie w każdym przypadku się zgadzają i niestety nie udało mi się ustalić powodu. Poniżej przedstawiam rozpiske wyników czyli ilośc zgadzających się prądów w obu rozwiązaniach dla różnej wielkości grafów (warto zaznaczyć że wyniki są bardzo różne w zależnosci od wygenerowanego grafu).

ERDOS GRAPH TEST
Number of currents mismatched: 68 for 20 nodes
Number of currents mismatched: 0 for 40 nodes
Number of currents mismatched: 539 for 60 nodes
Number of currents mismatched: 0 for 80 nodes
Number of currents mismatched: 0 for 100 nodes
Number of currents mismatched: 0 for 120 nodes

Number of currents mismatched: 0 for 140 nodes
Number of currents mismatched: 0 for 160 nodes
Number of currents mismatched: 0 for 180 nodes
Number of currents mismatched: 0 for 200 nodes
CUBIC GRAPH TEST
Number of currents mismatched: 0 for 20 nodes
Number of currents mismatched: 57 for 40 nodes
Number of currents mismatched: 0 for 60 nodes
Number of currents mismatched: 0 for 80 nodes
Number of currents mismatched: 0 for 100 nodes
Number of currents mismatched: 0 for 120 nodes
Number of currents mismatched: 0 for 140 nodes
Number of currents mismatched: 0 for 160 nodes
Number of currents mismatched: 0 for 180 nodes
Number of currents mismatched: 0 for 200 nodes
2D GRID GRAPH TEST
Number of currents mismatched: 12 for 9 nodes
Number of currents mismatched: 0 for 5 nodes
Number of currents mismatched: 0 for 7 nodes
Number of currents mismatched: 0 for 9 nodes
Number of currents mismatched: 0 for 11 nodes
Number of currents mismatched: 0 for 13 nodes
SMALL WORLD GRAPH TEST
Number of currents mismatched: 0 for 20 nodes
Number of currents mismatched: 192 for 40 nodes
Number of currents mismatched: 0 for 60 nodes
Number of currents mismatched: 0 for 80 nodes
Number of currents mismatched: 0 for 100 nodes
Number of currents mismatched: 0 for 120 nodes
Number of currents mismatched: 0 for 140 nodes
Number of currents mismatched: 0 for 160 nodes
Number of currents mismatched: 0 for 180 nodes
Number of currents mismatched: 0 for 200 nodes

Dołączam jeszcze obrazek z przykładową wizualizacją rozwiązania.



Poprawnośc wyników oraz wizualizacje można sprawdzić odplając plik test.py znajdujący się w folderze zad3. Warto dodać że program obsługuje również ładowanie grafow z plików txt.