

Projektowanie Algorytmów i Metody Sztucznej Inteligencji - Projekt 2 - Grafy

Paweł Szczepaniak (249014)

3 maja 2020r.
Termin kursu: Pt 9.15

1 Wprowadzenie

Celem projektu jest zbadanie efektywności algorytmu Bellmana-Forda w zależności od sposobu reprezentacji grafu (w postaci macierzy sąsiedztwa i listy sąsiedztwa) oraz gęstości grafu. Badania wykonano dla grafów o pięciu różnych liczbach wierzchołków (10, 50, 100, 200 i 500) oraz dla czterech różnych gęstości grafu (25%, 50%, 75% oraz dla grafu pełnego). Dla grafu o danej liczbie wierzchołków, gęstości i sposobie reprezentacji wygenerowano po sto losowych instancji, po czym czasy działania algorytmu uśredniono.

2 Opis algorytmu

Idea algorytmu Bellmana-Forda opiera się na metodzie relaksacji, polegającej na sprawdzeniu, czy przy przejściu daną krawędzią grafu z wierzchołka u do wierzchołka v , nie otrzymamy krótszej niż dotychczasowa ścieżki z wierzchołka s do wierzchołka v , gdzie u, v są dowolnymi wierzchołkami grafu połączonymi krawędzią o danej wadze, a s jest wierzchołkiem grafu połączonym z wierzchołkiem v przez wierzchołek u . Relaksacja każdej krawędzi przeprowadzana jest $|V| - 1$ razy, gdzie $|V|$ - liczba wierzchołków grafu.

W odróżnieniu od algorytmu Dijkstry, algorytm Bellmana-Forda jest w stanie obsługiwać grafu z wagami ujemnymi oraz wykrywać cykle o łącznej ujemnej wadze, które mogą zostać utworzone przez krawędzie o wagach ujemnych.

Złożoność obliczeniowa algorytmu Bellmana-Forda

- Dla reprezentacji grafu w postaci listy sąsiedztwa: $O(|V| \cdot |E|)$
- Dla reprezentacji grafu w postaci macierzy sąsiedztwa: $O(|V|^3)$

Złożoność pamięciowa algorytmu Bellmana-Forda (przy założeniu, że bierzemy pod uwagę również sam graf):

- Dla reprezentacji grafu w postaci listy sąsiedztwa: $O(|V| + |E|)$
- Dla reprezentacji grafu w postaci macierzy sąsiedztwa: $O(|V|^2)$

3 Przebieg badań

3.1 Średnie czasy działania algorytmu [ms]

3.1.1 Reprezentacja grafu w postaci listy sąsiedztwa

Liczba wierzchołków Gęstość grafu	10	50	100	200	500
25%	0.001464	0.131873	0.836543	6.623090	101.806000
50%	0.002366	0.249129	1.575270	13.661500	204.438000
75%	0.003856	0.297981	2.481280	21.113000	304.511000
100% (graf pełny)	0.003944	0.405861	3.138710	25.690000	436.405000

Tabela 1: Średnie czasy działania algorytmu Bellmana-Forda przy reprezentacji grafu w postaci listy sąsiedztwa

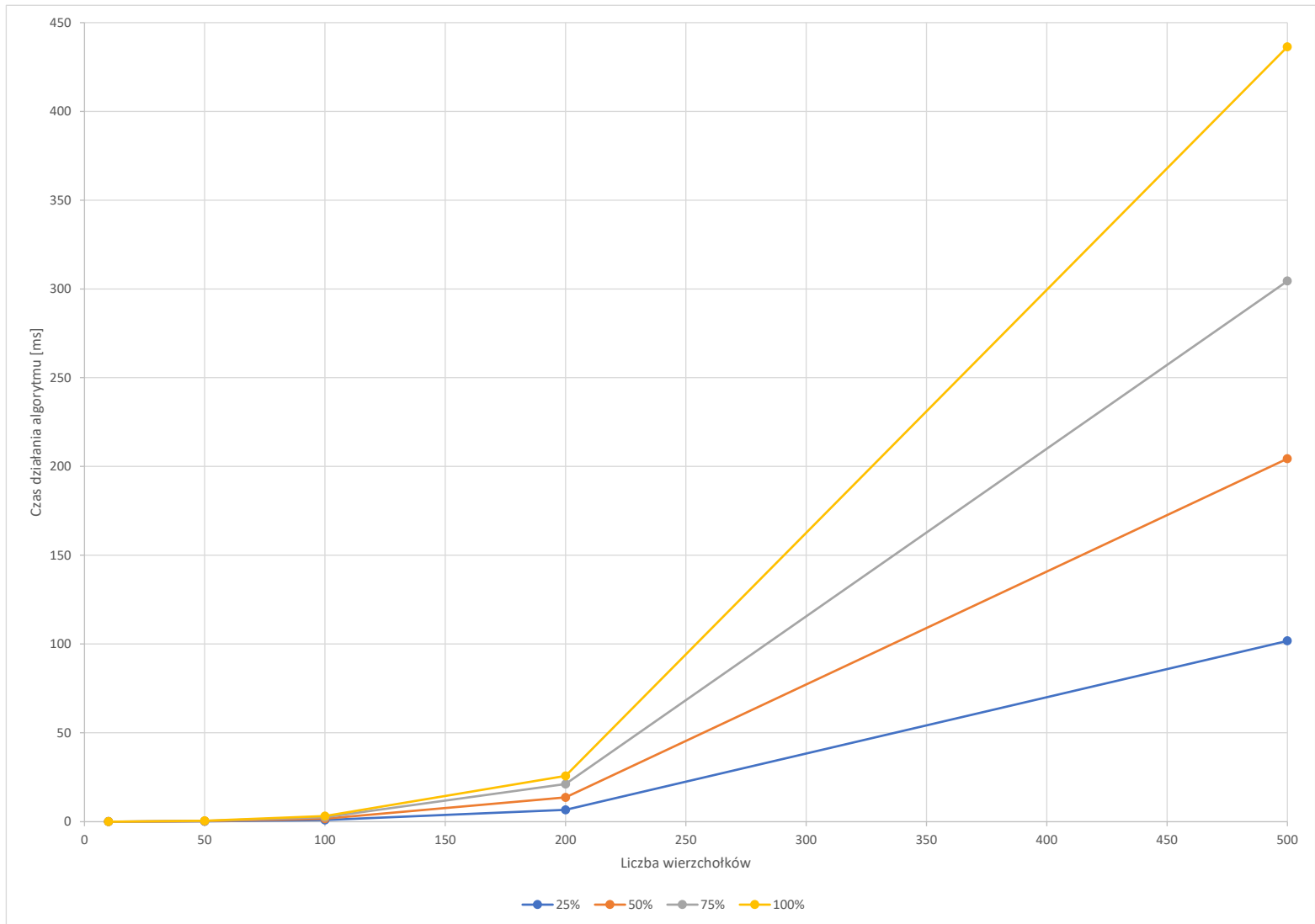
3.1.2 Reprezentacja grafu w postaci macierzy sąsiedztwa

Liczba wierzchołków Gęstość grafu	10	50	100	200	500
25%	0.004307	0.423645	2.832260	22.316700	354.687000
50%	0.003965	0.351716	2.892080	22.495900	359.083000
75%	0.004093	0.360270	2.765170	22.552500	353.161000
100% (graf pełny)	0.004407	0.359008	2.757990	21.956000	361.497000

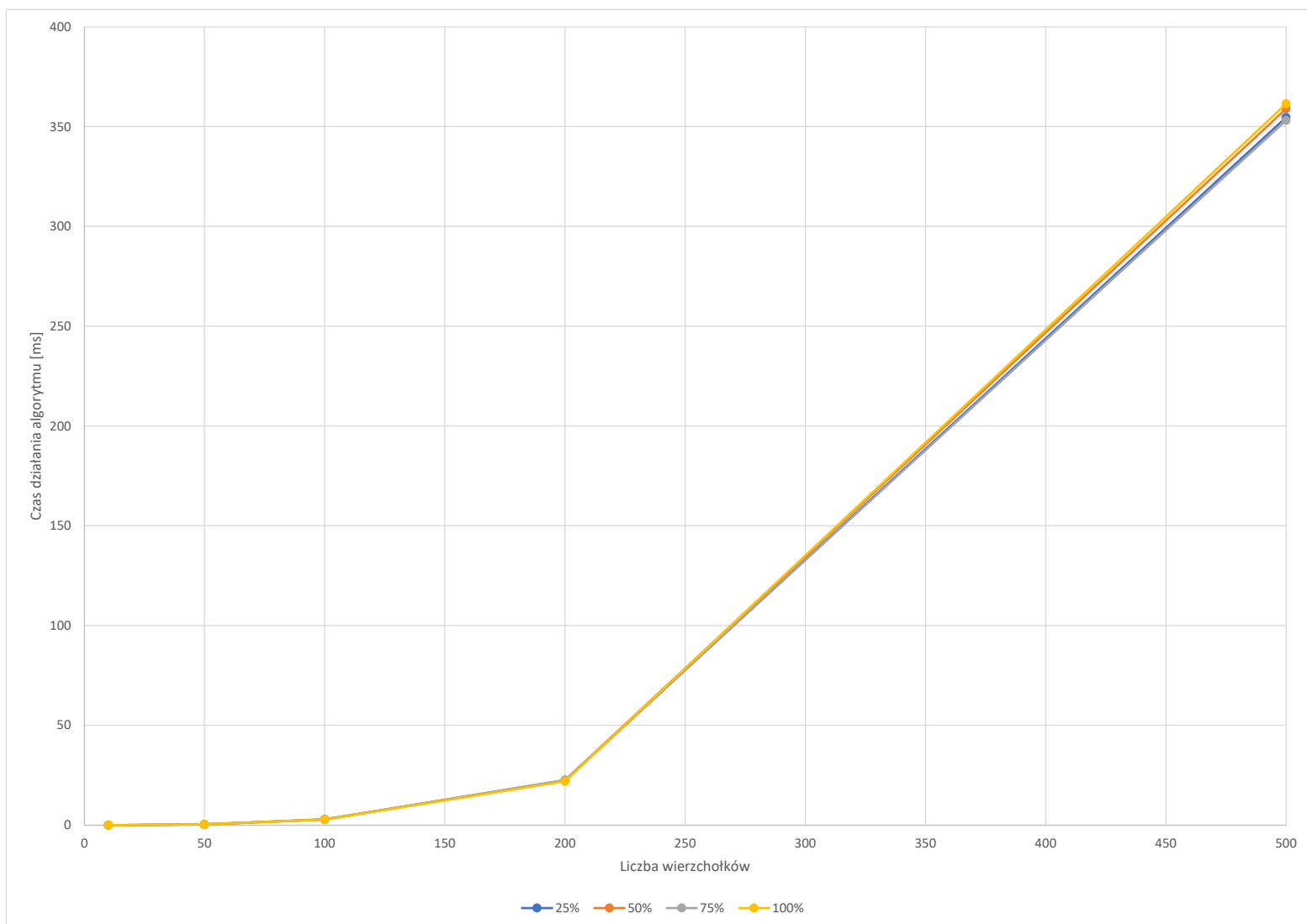
Tabela 2: Średnie czasy działania algorytmu Bellmana-Forda przy reprezentacji grafu w postaci macierzy sąsiedztwa

3.2 Wykresy zależności czasu wykonania algorytmu od liczby wierzchołków

3.2.1 Wykresy typu 1 (gęstość grafu jako parametr)

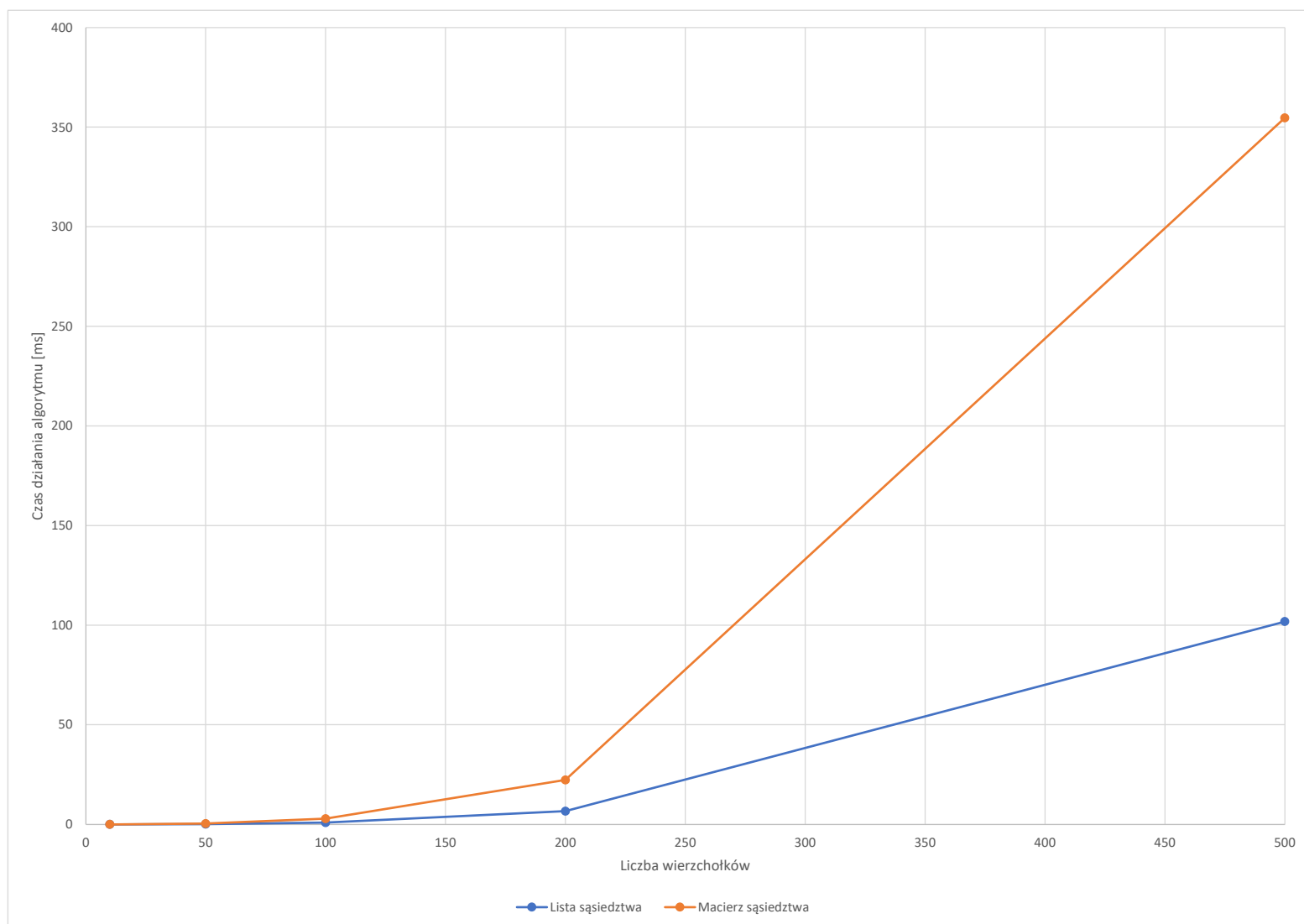


Rysunek 1: Wykres zależności średniego czasu działania algorytmu Bellmana-Forda od liczby wierzchołków dla reprezentacji grafu w postaci listy sąsiedztwa

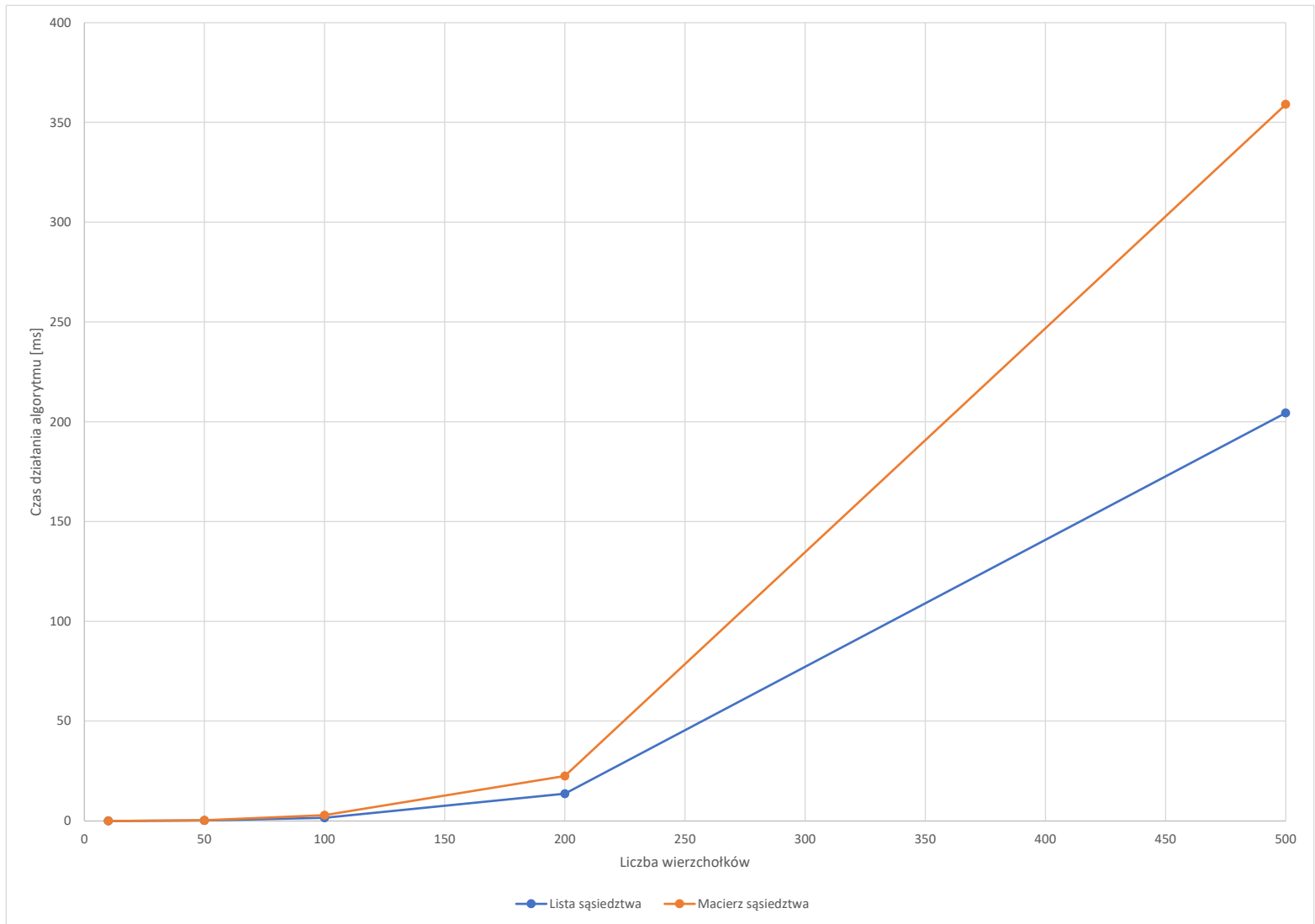


Rysunek 2: Wykres zależności średniego czasu działania algorytmu Bellmana-Forda od liczby wierzchołków dla reprezentacji grafu w postaci macierzy sąsiedztwa

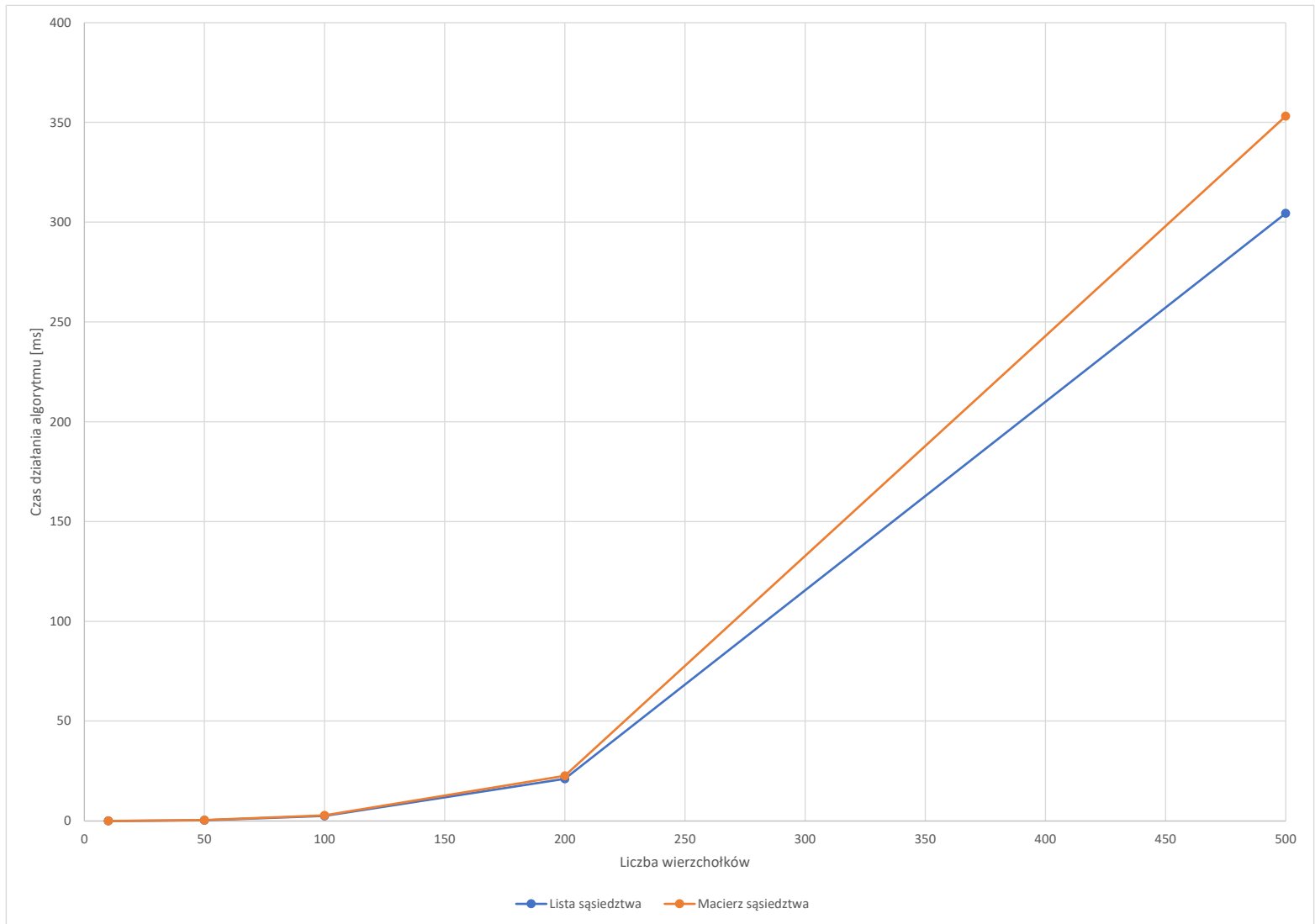
3.2.2 Wykresy typu 2 (typ reprezentacji grafu jako parametr)



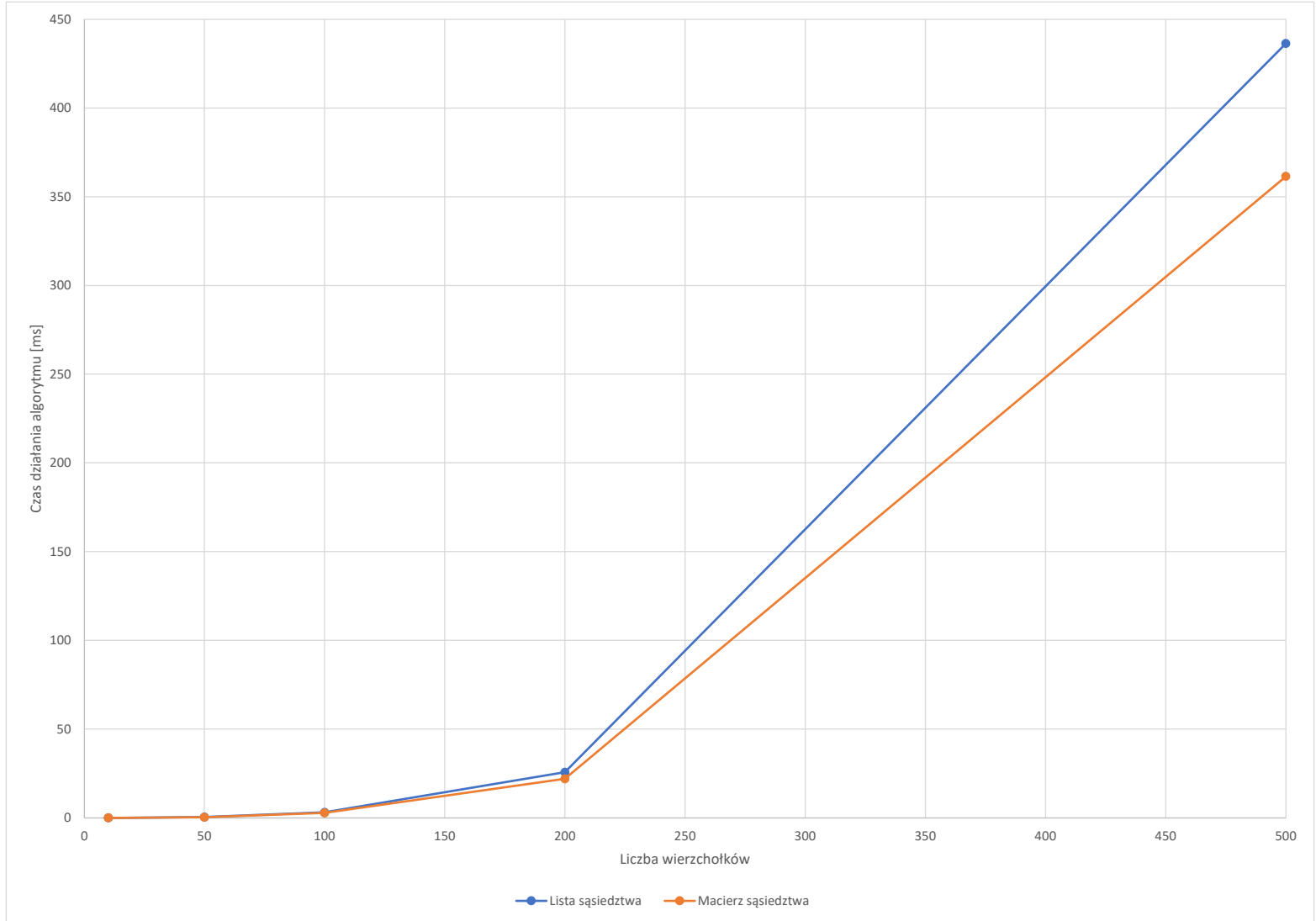
Rysunek 3: Wykres zależności średniego czasu działania algorytmu Bellmana-Forda od liczby wierzchołków dla grafu o gęstości 25%



Rysunek 4: Wykres zależności średniego czasu działania algorytmu Bellmana-Forda od liczby wierzchołków dla grafu o gęstości 50%



Rysunek 5: Wykres zależności średniego czasu działania algorytmu Bellmana-Forda od liczby wierzchołków dla grafu o gęstości 75%



Rysunek 6: Wykres zależności średniego czasu działania algorytmu Bellmana-Forda od liczby wierzchołków dla grafu pełnego

4 Wnioski

Można zauważyć, że czas wykonywania algorytmu Bellmana-Forda jest z reguły krótszy dla reprezentacji grafu w postaci listy sąsiedztwa, niż w przypadku macierzy sąsiedztwa, co zgadza się z założeniami oczekiwanej złożoności obliczeniowej algorytmu. Warto również zwrócić uwagę, że czas wykonywania algorytmu dla reprezentacji grafu w postaci listy sąsiedztwa rośnie wraz ze wzrostem gęstości grafu, w przypadku macierzy sąsiedztwa jest on niezależny od gęstości grafu. Wiąże się to ze wzrostem liczby krawędzi grafu przy zwiększaniu jego gęstości (zgodnie ze wzorem $D = \frac{|E|}{|V|(|V| - 1)}$), co przy przyjętej złożoności obliczeniowej algorytmu dla listy sąsiedztwa $O(|V| \cdot |E|)$ wpływa na czas jego wykonania. Przyjęta złożoność obliczeniowa algorytmu dla macierzy sąsiedztwa $O(|V|^3)$ nie bierze pod uwagę liczby krawędzi, jedynie liczbę wierzchołków, która przy każdej gęstości grafu pozostaje stała.

5 Bibliografia

- [Wikipedia](#)
- [cppreference.com](#)
- [stackoverflow.com](#)