

Raport z Analizy Efektywności Wielowątkowości

Igor Tomkowicz
nnpdev@gmail.com

1 kwietnia 2025

Spis treści

1	Wprowadzenie	2
2	Środowisko testowe	2
3	Metodologia	2
3.1	Dane wejściowe i konfiguracja symulacji Monte Carlo	2
3.2	Przebieg testów	3
4	Opis logiki przetwarzania i zastosowania wielowątkowości	3
5	Dodatkowe uwagi i analiza czasów przetwarzania	3
5.1	Czas przetwarzania pojedynczych elementów	3
5.2	Analiza rozkładu wyników symulacji Monte Carlo	3
6	Wizualizacja wyników	4
6.1	Wykres zależności czasu wykonania od liczby wątków	4
6.2	Wykresy rozkładu wyników Monte Carlo	4
7	Porównanie wyników dla różnych konfiguracji drzew	7
7.1	Opis konfiguracji	7
7.2	Wykres porównawczy	7
8	Wnioski	8
9	Podsumowanie	8

1 Wprowadzenie

W niniejszym raporcie przedstawiono analizę efektywności wielowątkowości w programie obliczającym tzw. **legacy index** – wskaźnik interpretowany jako miara doświadczenia przekazywanego z pokolenia na pokolenie. Wynik ten uzyskiwany jest poprzez obliczenie ważonej średniej wieku z wykorzystaniem symulacji Monte Carlo, której dokładność wzrasta wraz ze wzrostem liczby iteracji. Program operuje na generowanym zbiorze danych reprezentującym rodzinę, gdzie kolejne pokolenia charakteryzują się zmniejszonym wiekiem. Drugim argumentem programu jest liczba wątków, która wpływa na równoległość przetwarzania danych. Program umożliwia również dodawanie własnych zadań (osób) do przetworzenia, co jednak nie zostało wykorzystane w prezentowanych testach. Bazowo program przetwarza dwa pierwsze poziomy drzewa (root i dzieci).

2 Środowisko testowe

Testy przeprowadzono na następującej konfiguracji:

- **Procesor:** Intel i5-11400H (6 rdzeni / 12 wątków)
- **Pamięć RAM:** 16 GB
- **System operacyjny:** Windows 11
- **Uwagi:** Pomiar czasu realizowany był przy użyciu wbudowanej funkcji `System.currentTimeMillis()`.

3 Metodologia

3.1 Dane wejściowe i konfiguracja symulacji Monte Carlo

Program generuje strukturę drzewa reprezentującą zbiór rodzinny, gdzie:

- Liczba rootów: 15,
- Liczba dzieci dla każdego roota: 15,
- Liczba wnuków dla każdego roota: 15.

Obliczenie **legacy index** przebiega w dwóch etapach:

1. Rekurencyjne obliczenie ważonej średniej wieku. Każdy kolejny poziom (dzieci, wnuki) ma zmniejszoną wagę.
2. Korekta wyniku poprzez symulację Monte Carlo, która wykorzystuje następujące parametry:
 - Liczba iteracji (np. 5, 100, 500),
 - Liczba dni w roku: 365,
 - Liczba zdarzeń na dzień: 5,
 - Prawdopodobieństwo zdarzenia pozytywnego: 0.1,
 - Wpływ zdarzenia pozytywnego: +0.045,
 - Wpływ zdarzenia negatywnego: -0.005.

3.2 Przebieg testów

Testy wykonano dla przetwarzania sekwencyjnego oraz równoległego. W trybie równoległym zadania dodawane są do kolejki (listy) przez klasę `TaskManager`, a wątki pracownicze (instancje `CalculationThread`) pobierają je i przetwarzają. Drugi argument programu określa liczbę wątków, co umożliwia analizę wydajności dla różnych konfiguracji (np. 2, 3, 4, ..., X wątków). W naszym przykładzie nie wykorzystano dodatkowych zadań – przetwarzany był ten sam zbiór danych, odpowiednio okrojony dla konfiguracji 10x10x10.

4 Opis logiki przetwarzania i zastosowania wielowątkowości

Wielowątkowość zastosowano w celu przyspieszenia symulacji Monte Carlo, która jest intensywna obliczeniowo. Kluczowe elementy:

- **TaskManager:** Zarządza kolejką zadań (lista), synchronizując dostęp do obiektów `Person`.
- **CalculationThread:** Każdy wątek oblicza **legacy index** dla przypisanego obiektu, wykorzystując symulację Monte Carlo.
- **ResultManager i ResultFormatter:** Odpowiadają za gromadzenie i formatowanie wyników obliczeń.

5 Dodatkowe uwagi i analiza czasów przetwarzania

5.1 Czas przetwarzania pojedynczych elementów

W trakcie testów zauważono, że czas przetwarzania obiektu `Person` zależy od jego **legacy index**, czyli od **averageAge**. Kilka przykładów:

- **Person: Root724 Johnson** — *averageAge: 11,78, Processing Time: 232 ms*
- **Person: Child645 Johnson** — *averageAge: 17,82, Processing Time: 592 ms*
- **Person: Child2636 Johnson** — *averageAge: 14,47, Processing Time: 309 ms*

Wyższy czas przetwarzania wynika z faktu, że obiekt o wyższym **averageAge** (a więc wyższym **legacy index**) powoduje, że symulacja Monte Carlo musi przeprowadzić większą liczbę operacji. W szczególności:

$$D = \text{averageAge} \times 365$$

oznacza liczbę dni symulacji, a całkowita liczba operacji wynosi:

$$N = \text{averageAge} \times 365 \times \text{EVENTS_PER_DAY} \times \text{MONTE_CARLO_RUNS}.$$

Przy założeniu, że pojedyncza operacja trwa stały czas α , czas przetwarzania T można oszacować jako:

$$T \sim \alpha \times \text{averageAge} \times 365 \times \text{EVENTS_PER_DAY} \times \text{MONTE_CARLO_RUNS}.$$

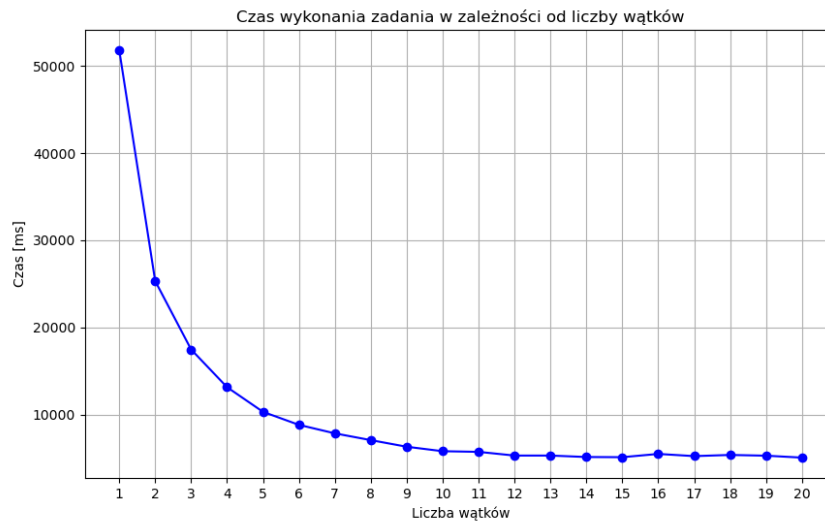
5.2 Analiza rozkładu wyników symulacji Monte Carlo

Przy wykorzystaniu 500 iteracji symulacji wyniki mają rozkład o wyraźnie skoncentrowanej wartości średniej, co wskazuje na wysoką stabilność obliczeń. Graficznie rozkład ten przypomina jednolite, symetryczne skupienie wyników wokół średniej – charakterystyczny, zaokrąglony kształt, który odzwierciedla tendencję do koncentracji wyników przy większej liczbie iteracji. Oznacza to, że im więcej iteracji, tym wyniki pojedynczych symulacji stają się bardziej zbliżone do siebie.

6 Wizualizacja wyników

6.1 Wykres zależności czasu wykonania od liczby wątków

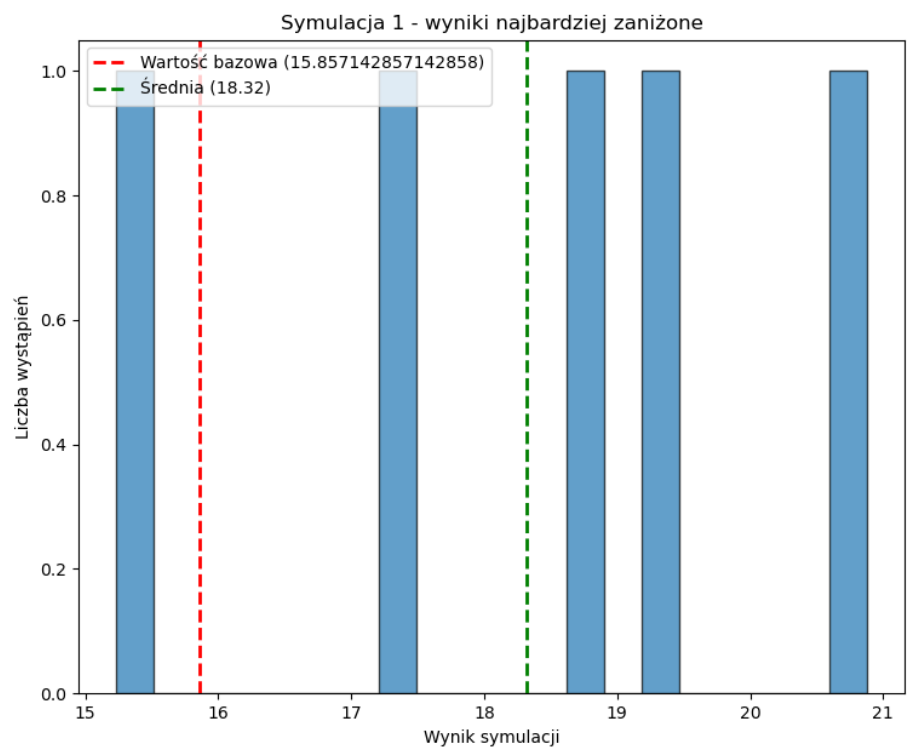
Poniżej przedstawiono wykres ilustrujący czas wykonania zadania w zależności od liczby wątków. Każdy wykres odpowiada przetwarzaniu sekwencyjnemu (1 wątek) oraz równoległemu (2,3, ..., X wątków) dla różnych konfiguracji wątków.



Rysunek 1: Czas wykonania zadania w zależności od liczby wątków.

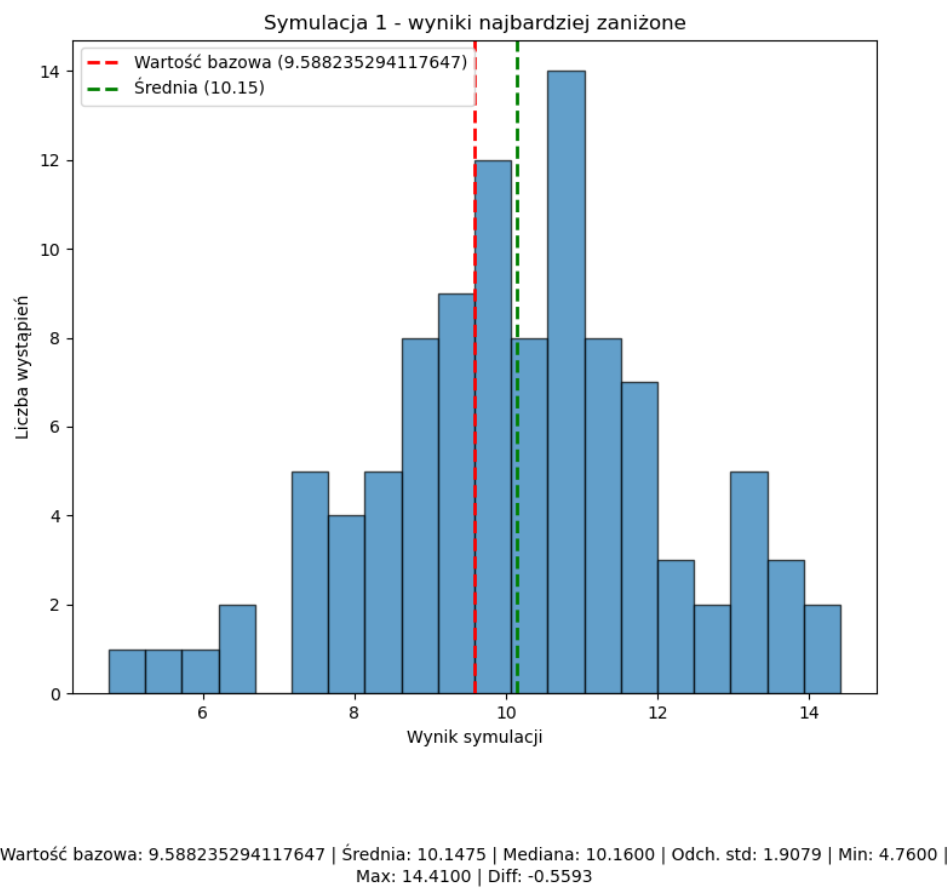
6.2 Wykresy rozkładu wyników Monte Carlo

Dla zobrazowania wpływu liczby iteracji symulacji przygotowano następujące wykresy:

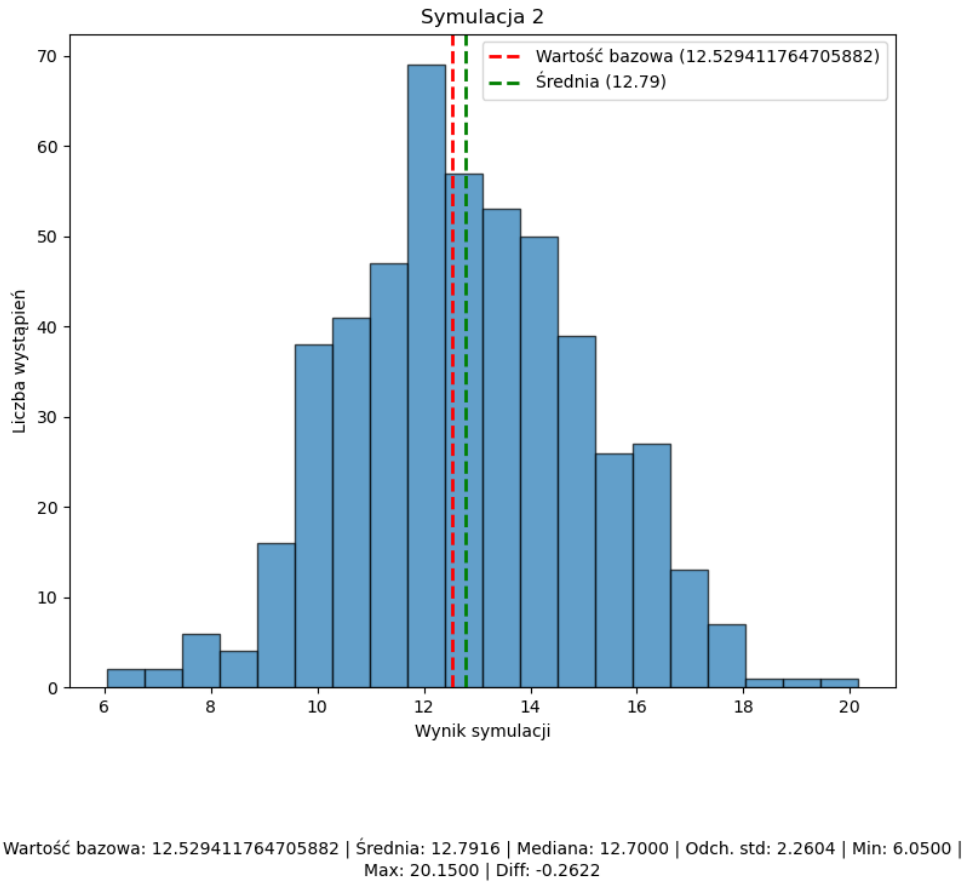


Wartość bazowa: 15.857142857142858 | Średnia: 18.3200 | Mediana: 18.7800 | Odch. std: 1.8988 | Min: 15.2300
 | Max: 20.8800 | Diff: -2.4629

Rysunek 2: Rozkład wyników symulacji Monte Carlo dla 5 iteracji.



Rysunek 3: Rozkład wyników symulacji Monte Carlo dla 100 iteracji.



Rysunek 4: Rozkład wyników symulacji Monte Carlo dla 500 iteracji.

7 Porównanie wyników dla różnych konfiguracji drzew

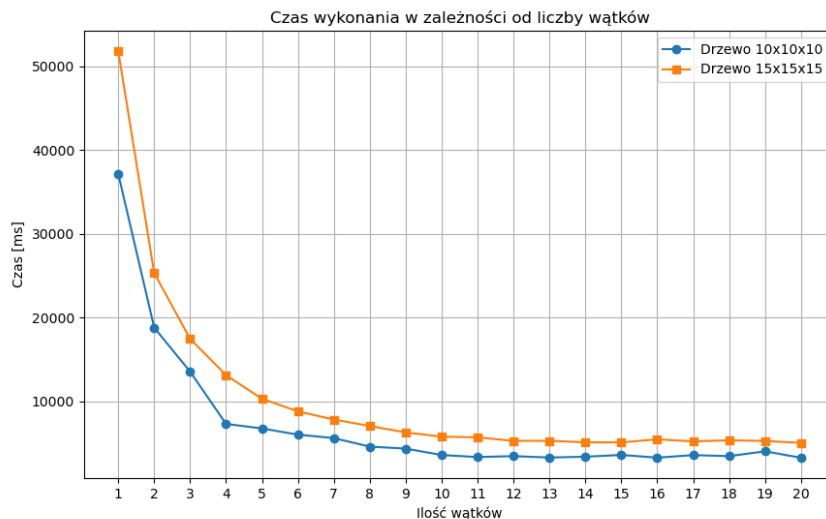
W niniejszej sekcji przedstawiono porównanie wyników symulacji Monte Carlo dla dwóch konfiguracji drzewa: 10x10x10 oraz 15x15x15. Celem tego porównania jest ocena wpływu rozmiaru drzewa na czas przetwarzania.

7.1 Opis konfiguracji

- **Drzewo 10x10x10:** Struktura drzewa zawiera 10 rootów, każdy z 10 dziećmi i 10 wnukami.
- **Drzewo 15x15x15:** Struktura drzewa zawiera 15 rootów, każdy z 15 dziećmi i 15 wnukami.

7.2 Wykres porównawczy

Na wykresie widoczne jest, że niezależnie od wielkości drzewa czas przetwarzania maleje przy zwiększającej się liczbie wątków (aż do granicy dostępnych wątków procesora). Jednocześnie większe drzewo generuje więcej operacji symulacyjnych, co wynika z większej liczby elementów.



Rysunek 5: Porównanie czasów wykonania dla drzew 10x10x10 oraz 15x15x15 w zależności od liczby wątków.

8 Wnioski

Na podstawie przeprowadzonych testów oraz analizy wyników można wyciągnąć następujące wnioski:

- **Skalowalność:** Zwiększenie liczby wątków znacząco skraca czas przetwarzania. Optymalny efekt uzyskuje się przy około 12 wątkach (zgodnie z liczbą dostępnych wątków sprzętowych – 6 rdzeni z hyper-threadingiem). Przekroczenie tej liczby nie przynosi znaczących korzyści, a może wprowadzać dodatkowe narzuty.
- **Dokładność symulacji:** Większa liczba iteracji Monte Carlo poprawia precyzję wyników, co widać po skoncentrowanym rozkładzie wyników przy 500 iteracjach.
- **Wpływ legacy index na czas przetwarzania:** Obiekty o wyższym `averageAge` wymagają więcej operacji symulacyjnych, co przekłada się na dłuższy czas przetwarzania.
- **Porównanie konfiguracji drzew:** Przejście z konfiguracji 10x10x10 do 15x15x15 skutkuje znacznym wzrostem czasu przetwarzania (38 s vs 52 s przy 1 wątku) ze względu na większą liczbę elementów.

9 Podsumowanie

- Wielowątkowość znacząco przyspiesza obliczenia przy zastosowaniu intensywnych symulacji, takich jak metoda Monte Carlo.
- W przypadku ograniczeń sprzętowych zaleca się dostosowanie liczby wątków do liczby dostępnych wątków sprzętowych.
- Dla jeszcze dokładniejszych wyników należy rozważyć zwiększenie liczby iteracji symulacji Monte Carlo.
- Przy rozbudowie struktury drzewa należy pamiętać, że wzrost liczby elementów zwiększa czas przetwarzania.

- W przyszłości można rozszerzyć klasę **Person** o dodatkowe atrybuty, na przykład stan zdrowia, cechy osobowości czy predyspozycje, oraz wprowadzić nowe parametry symulacji Monte Carlo, które pozwolą na lepszą selekcję wpływu zdarzeń pozytywnych i negatywnych na legacy index. Takie modyfikacje mogłyby znaleźć zastosowanie w symulacjach strategicznych lub grach symulacyjnych, gdzie analiza rodów i ich cech odgrywa istotną rolę.