

**Badanie efektywności blokowej metody  
Cholesky'ego-Banachiewicza w rozwiązywaniu  
układów równań**

Metody Numeryczne: Projekt 2

Autor: Adam Kaniasty

08.12.2023

## Spis treści

<b>1</b>	<b>Opis metody matematycznej</b>	<b>2</b>
1.1	Generowanie losowych macierzy dodatnio określonych . . . . .	2
1.2	Metoda Cholesky’ego-Banachiewicza . . . . .	2
1.3	Blokowa Metoda Choleskiego-Banachiewicza . . . . .	3
1.4	Metoda Cholesky’ego-Banachiewicza w Rozwiązaniu Układów Równań . . . . .	3
1.5	Blokowa Metoda Cholesky’ego-Banachiewicza w Rozwiązaniu Układów Równań . . . . .	4
1.6	Metoda Podstawienia w Przód . . . . .	4
1.7	Metoda Podstawienia w Tył . . . . .	4
<b>2</b>	<b>Opis działania programu</b>	<b>5</b>
2.1	Funkcja <code>normalDecomposition</code> . . . . .	5
2.2	Funkcja <code>normalSolve</code> . . . . .	5
2.3	Funkcja <code>blockDecomposition</code> . . . . .	5
2.4	Funkcja <code>blockSolve</code> . . . . .	6
2.4.1	Funkcja <code>forwardSubstitutionBlock</code> . . . . .	6
2.4.2	Funkcja <code>backwardSubstitutionBlock</code> . . . . .	6
2.5	GUI . . . . .	6
2.5.1	Examples . . . . .	6
2.5.2	CB error . . . . .	7
2.5.3	CB solve error . . . . .	7
2.5.4	CB dependency error . . . . .	7
2.5.5	Matrix error . . . . .	7
<b>3</b>	<b>Przykłady</b>	<b>7</b>
3.1	Przykład 1 . . . . .	7
3.2	Przykład 2 . . . . .	7
3.3	Przykład 3 . . . . .	8
3.4	Przykład 4 . . . . .	8
3.5	Przykład 5 . . . . .	8
3.6	Przykład 6 . . . . .	8
<b>4</b>	<b>Pomiary błędów</b>	<b>8</b>
4.1	Błędy Dekompozycji . . . . .	8
4.2	Błędy Rozwiązania Układu Równań . . . . .	9
<b>5</b>	<b>Zastosowania</b>	<b>10</b>
<b>6</b>	<b>Wnioski</b>	<b>11</b>

# 1 Opis metody matematycznej

## 1.1 Generowanie losowych macierzy dodatnio określonych

**Charakterystyka badanych macierzy** W ramach projektu analizie podlegają macierze symetryczne i dodatnio określone  $A$  o wymiarach  $n \times n$ , gdzie  $n = 3p$ . Macierze te są specjalnego typu, zwanego macierzami blokowymi, charakteryzującymi się strukturą podziału na mniejsze bloki macierzy:

$$A = \begin{bmatrix} A_{11} & A_{12} & 0 \\ A_{12}^T & A_{22} & A_{23} \\ 0 & A_{23}^T & A_{33} \end{bmatrix}$$

gdzie każdy blok  $A_{ij}$  jest dodatnio określoną macierzą  $p \times p$ .

Generowanie losowych macierzy dodatnio określonych wymaga zastosowania metod matematycznych gwarantujących, że wszystkie wartości własne macierzy będą dodatnie. W funkcji `randomMatrixGenerator`, kluczowym elementem jest zapewnienie dodatniej określoności macierzy za pomocą rozkładu spektralnego.

**Rozkład spektralny** jest metodą wykorzystywaną do przedstawienia macierzy symetrycznej  $A$  jako iloczynu macierzy ortogonalnej  $Q$ , macierzy diagonalnej  $\Lambda$ , zawierającej wartości własne, oraz transpozycji macierzy ortogonalnej  $Q^T$ , co matematycznie wyraża się jako  $A = Q\Lambda Q^T$ . Wartości własne są generowane losowo z zakresu dodatnich liczb rzeczywistych, a następnie umieszczane na przekątnej macierzy  $\Lambda$ . Macierz ortogonalna  $Q$  jest generowana poprzez zastosowanie procesu ortogonalizacji na losowej macierzy. Każdy blok diagonalny macierzy  $A$  jest w ten sposób generowany niezależnie, a wynikowy blok jest wstawiany do odpowiedniej lokalizacji w macierzy  $A$ .

## 1.2 Metoda Cholesky’ego-Banachiewicza

Metoda Cholesky’ego-Banachiewicza jest algorytmem służącym do rozkładu macierzy symetrycznej, dodatnio określonej na iloczyn macierzy trójkątnej dolnej i jej transpozycji. Jest to szczególny przypadek rozkładu LU, gdzie macierz  $L$  jest macierzą trójkątną dolną, a macierz  $U$  jest jej transpozycją. Kluczowym elementem metody jest fakt, że każda symetryczna, dodatnio określona macierz  $A$  może być przedstawiona jako  $A = LL^T$ , gdzie  $L$  jest macierzą trójkątną dolną.

Aby wyznaczyć macierz  $L$  w metodzie Cholesky’ego-Banachiewicza, należy podjąć następujące kroki. Zakładamy, że dana jest symetryczna, dodatnio określona macierz  $A$  o wymiarach  $n \times n$ .

1. Obliczanie  $L_{ii} = \sqrt{A_{ii} - \sum_{k=1}^{i-1} L_{ik}^2}$ . Jest to element diagonalny macierzy  $L$ .
2. Dla każdego  $j$  od  $i+1$  do  $n$ , obliczanie elementów pozadiagonalnych  $L_{ji} = \frac{1}{L_{ii}} \left( A_{ji} - \sum_{k=1}^{i-1} L_{jk} L_{ik} \right)$ .

Macierz  $L$  wyznaczona w ten sposób wymaga realizacji operacji pierwiastkowania i dzielenia, co podkreśla konieczność, aby macierz  $A$  była dodatnio określona.

### 1.3 Blokowa Metoda Choleskiego-Banachiewicza

Blokowa metoda Choleskiego-Banachiewicza stanowi zaawansowaną wersję standardowej metody Choleskiego. Umożliwia ona rozkładanie symetrycznych, dodatnio określonych macierzy  $A$  na iloczyn macierzy trójkątnej dolnej  $L$  i jej transpozycji  $L^T$ , przy czym operacje wykonywane są na blokach macierzy.

Założmy, że mamy macierz  $A$  o wymiarach  $n \times n$ , którą dzielimy na bloki o rozmiarach  $b \times b$ , gdzie  $b$  jest ustaloną wielkością bloku. Procedura rozkładu blokowego przedstawia się następująco:

1. Macierz  $A$  dzielona jest na bloki:

$$A = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mm} \end{pmatrix}$$

gdzie każdy  $A_{ij}$  jest blokiem macierzy o wymiarach  $b \times b$ .

2. Analogicznie, macierz  $L$  również dzielona jest na bloki:

$$L = \begin{pmatrix} L_{11} & 0 & \cdots & 0 \\ L_{21} & L_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ L_{m1} & L_{m2} & \cdots & L_{mm} \end{pmatrix}$$

3. Dla każdego bloku na przekątnej macierzy  $A$ , stosujemy standardowy algorytm rozkładu Choleskiego do obliczenia odpowiedniego bloku  $L_{ii}$  w macierzy  $L$ .
4. Następnie, dla bloków poniżej przekątnej, obliczamy  $L_{ji}$  korzystając z zależności:

$$L_{ji} = (A_{ji} - \sum_{k=1}^{i-1} L_{jk} L_{ik}^T) L_{ii}^{-T}$$

Po zakończeniu tych kroków, macierz  $L$  jest dolną macierzą trójkątną, spełniającą równanie  $A = LL^T$ .

### 1.4 Metoda Cholesky'ego-Banachiewicza w Rozwiązywaniu Układów Równań

Założmy, że naszym zadaniem jest rozwiązanie układu równań  $Ax = b$ , gdzie  $A$  jest symetryczną, dodatnio określoną macierzą. Aby rozwiązać podany układ równań należy podjąć następujące kroki:

1. Rozkład Choleskiego: Znalezienie macierzy  $L$ , dla której zachodzi  $A = LL^T$ .
2. Rozwiązanie układu  $Ly = b$  przy użyciu metody podstawienia w przód.
3. Rozwiązanie układu  $L^T x = y$  przy użyciu metody podstawienia wstecz.

### 1.5 Blokowa Metoda Cholesky'ego-Banachiewicza w Rozwiązaniu Układów Równań

Blokowa metoda Cholesky'ego rozszerza standardową metodę na przypadki, gdy macierz  $A$  podzielona jest na bloki.

1. Rozkład Blokowy Choleskiego: Znalezienie macierzy trójkątnej dolnej  $L$  w formie blokowej, dla której  $A = LL^T$ .
2. Rozwiązanie układu  $Ly = b$  przy użyciu metody podstawienia w przód z blokami.
3. Rozwiązanie układu  $L^T x = y$  przy użyciu metody podstawienia wstecz z blokami.

### 1.6 Metoda Podstawienia w Przód

Metoda podstawienia w przód stosowana jest do rozwiązywania układów równań liniowych z macierzą współczynników będącą dolną macierzą trójkątną. Rozważmy układ równań  $Lx = b$ , gdzie  $L$  jest dolną macierzą trójkątną.

1. Rozpoczęcie od pierwszego równania w postaci  $L_{11}x_1 = b_1$  i wyznaczenie  $x_1 = \frac{b_1}{L_{11}}$ .
2. Dla każdego kolejnego równania  $i$  (od 2 do  $n$ ), obliczenie  $x_i$  według wzoru:

$$x_i = \frac{b_i - \sum_{j=1}^{i-1} L_{ij}x_j}{L_{ii}}$$

gdzie  $L_{ij}$  to elementy macierzy  $L$ , a  $b_i$  to elementy wektora prawych stron.

### 1.7 Metoda Podstawienia w Tył

Metoda podstawienia w tył jest stosowana do rozwiązywania układów równań liniowych z macierzą współczynników będącą górną macierzą trójkątną. Rozpatrzmy układ równań  $Ux = y$ , gdzie  $U$  jest górną macierzą trójkątną.

1. Rozpoczęcie od ostatniego równania w formie  $U_{nn}x_n = y_n$  i wyznaczenie  $x_n = \frac{y_n}{U_{nn}}$ .
2. Dla każdego równania  $i$  (od  $n-1$  do 1, w kolejności malejącej), obliczenie  $x_i$  według wzoru:

$$x_i = \frac{y_i - \sum_{j=i+1}^n U_{ij}x_j}{U_{ii}}$$

gdzie  $U_{ij}$  to elementy macierzy  $U$ , a  $y_i$  to elementy wektora prawych stron.

## 2 Opis działania programu

Projekt realizowany jest z użyciem środowiska MATLAB. Główna funkcja programu, nazwana `decomposition`, implementuje metodę Cholesky’ego-Banachiewicza do rozkładu macierzy symetrycznych, dodatnio określonych na macierz trójkątną dolną  $L$ .

### 2.1 Funkcja `normalDecomposition`

Funkcja `normalDecomposition` wykonuje rozkład macierzy kwadratowej  $A$  na macierz trójkątną dolną  $L$  zgodnie z algorytmem standardowego rozkładu Cholesky’ego.

#### Parametry wejścia

- **Macierz  $A$ :** Macierz kwadratowa, dla której będzie przeprowadzany rozkład.

#### Parametry wyjścia

- **Macierz  $L$ :** Macierz trójkątna dolna, będąca wynikiem rozkładu macierzy  $A$ .

### 2.2 Funkcja `normalSolve`

Funkcja `normalSolve` rozwiązuje układ równań liniowych  $Ax = b$  wykorzystując podany rozkład Cholesky’ego macierzy  $A$ .

#### Parametry wejścia

- **Macierz  $A$ :** Macierz kwadratowa z układu równań  $Ax = b$ .
- **Wektor  $b$ :** Wektor prawych stron układu równań.
- **`decomposition`:** Funkcja rozkładu Cholesky’ego-Banachiewicza, służąca do przekształcenia macierzy  $A$  w macierz trójkątną dolną  $L$ .

#### Parametry wyjścia

- **Wektor  $x$ :** Rozwiązanie układu równań  $Ax = b$ .

### 2.3 Funkcja `blockDecomposition`

Funkcja `blockDecomposition` wykonuje blokowy rozkład Cholesky’ego dla symetrycznej, dodatnio określonej macierzy  $A$ . Macierz jest dzielona na bloki 9 bloków, a wynikiem jest macierz trójkątna dolna  $L$ , taka że  $LL^T = A$ .

### Parametry wejścia

- **Macierz  $A$ :** Symetryczna, dodatnio określona macierz kwadratowa do rozkładu.

### Parametry wyjścia

- **Macierz  $L$ :** Macierz trójkątna dolna wynikająca z rozkładu blokowego macierzy  $A$ .

## 2.4 Funkcja blockSolve

Funkcja `blockSolve` rozwiązuje układ równań liniowych  $Ax = b$  wykorzystując blokowy rozkład Cholesky'ego macierzy  $A$ .

### Parametry wejścia

- **Macierz  $A$ :** Macierz kwadratowa w układzie równań  $Ax = b$ .
- **Wektor  $b$ :** Wektor prawych stron w układzie równań.
- **decomposition:** Funkcja realizująca blokowy rozkład Cholesky'ego macierzy  $A$ .

### Parametry wyjścia

- **Wektor  $x$ :** Rozwiązanie układu równań  $Ax = b$ .

#### 2.4.1 Funkcja forwardSubstitutionBlock

Funkcja `forwardSubstitutionBlock` wykonuje forward substitution dla blokowej macierzy trójkątnej dolnej  $L$  i wektora  $b$ , rozwiązując  $Ly = b$  dla  $y$ .

#### 2.4.2 Funkcja backwardSubstitutionBlock

Funkcja `backwardSubstitutionBlock` wykonuje backward substitution dla transponowanej blokowej macierzy trójkątnej górnej  $U$  i wektora  $y$ , rozwiązując  $Ux = y$  dla  $x$ .

## 2.5 GUI

Aby uruchomić GUI należy przedtem wywołać komendę: `addpath(genpath("./"))`  
Poniżej znajduje się opis parametrów w poszczególnych zakładkach GUI

### 2.5.1 Examples

- Wybór przykładu: Możliwość wyboru jednego z sześciu przykładów.

### 2.5.2 CB error

- maxP: Rozmiar macierzy blokowej.
- maxScale: Rozmiar wartości w macierzy. (brana jest co dziesiąta wartość)

### 2.5.3 CB solve error

- maxP: Maksymalny rozmiar macierzy blokowej (będą przetwarzane wartości 1:maxP)
- scale: Rozmiar wartości w macierzy.
- xScale: Rozmiar wartości x podczas generowania równań
- n: Ilość powtórzeń dla każdej konfiguracji (wartość błędu jest uśredniana)

### 2.5.4 CB dependency error

- maxP: Maksymalny rozmiar macierzy blokowej (będą przetwarzane wartości 1:maxP)
- maxScale: Rozmiar wartości w macierzy.
- xScale: Rozmiar wartości x podczas generowania równań
- n: Ilość powtórzeń dla każdej konfiguracji (wartość błędu jest uśredniana)

### 2.5.5 Matrix error

- P: Rozmiar macierzy blokowej.
- Scale: Rozmiar wartości w macierzy.

## 3 Przykłady

Macierze 1-5 spełniają wcześniej opisaną własność (1.3)

### 3.1 Przykład 1

Pierwsza macierz (examples{1}) jest macierzą pełną, z różnymi wartościami w każdym elemencie.

### 3.2 Przykład 2

Druga macierz (examples{2}) jest podobna do pierwszej, ale z wartościami tylko w blokach na przekątnej.



### 3.3 Przykład 3

Trzecia macierz (examples{3}) zawiera bardzo duże wartości (1e6), w blokach na przekątnej.

### 3.4 Przykład 4

Czwarta macierz (examples{4}) składa się z bardzo małych wartości (1e-6).

### 3.5 Przykład 5

Piąta macierz (examples{5}) łączy małe i duże wartości: małe na przekątnej (1e-6) i duże w innych blokach.

### 3.6 Przykład 6

Szósta macierz (examples{6}) jest generowana losowo.

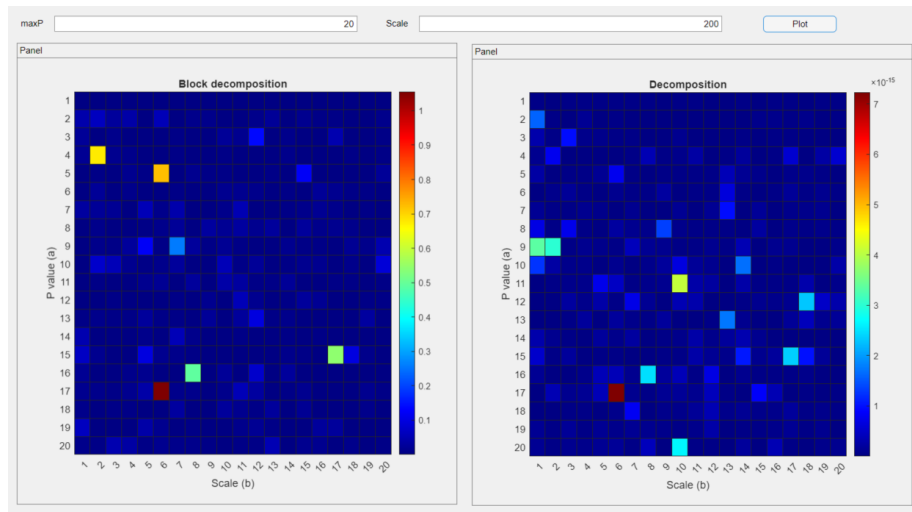
## 4 Pomiary błędów

### 4.1 Błędy Dekompozycji

Rozważamy błąd względny dekompozycji macierzy  $A$  na iloczyn  $L \cdot L^T$ , gdzie  $L$  jest macierzą trójkątną dolną. Błąd względny  $E$  dla dekompozycji jest zdefiniowany jako:

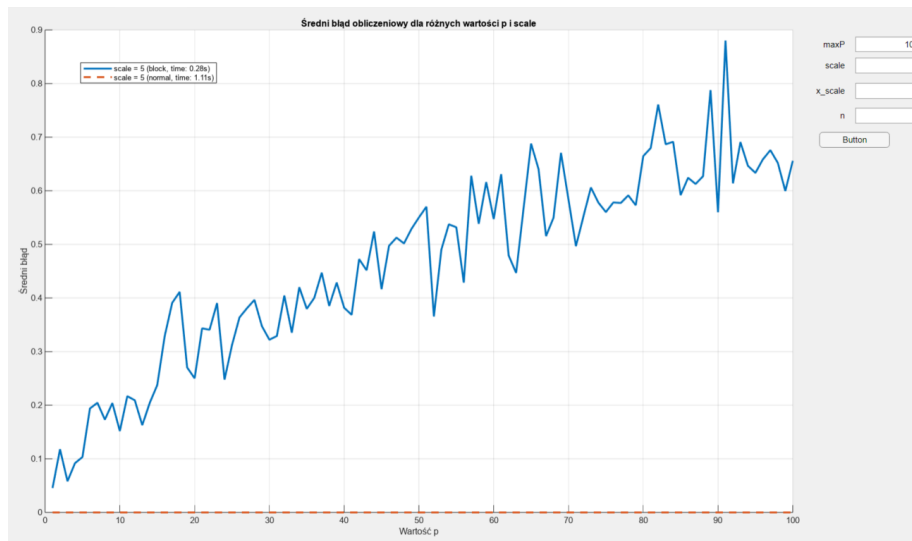
$$E = \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n \left| \frac{A_{ij} - (LL^T)_{ij}}{A_{ij}} \right|$$

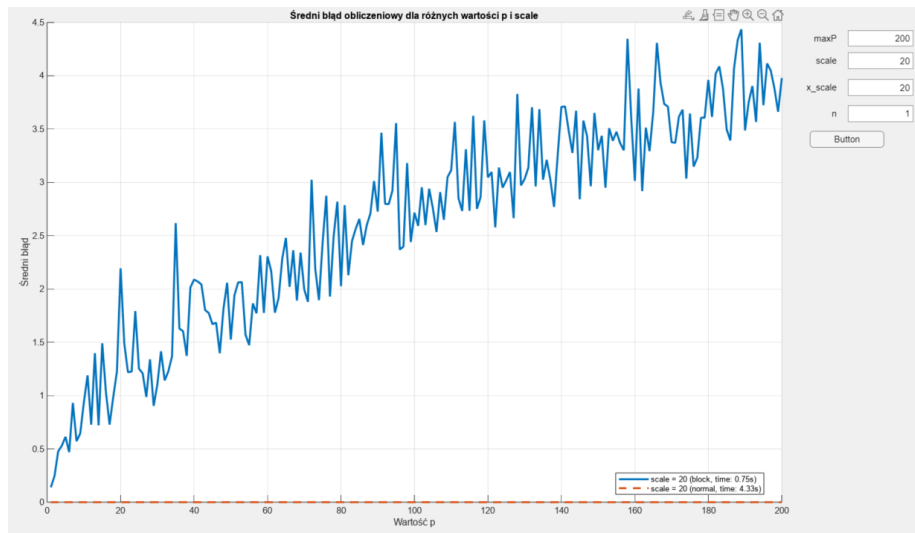
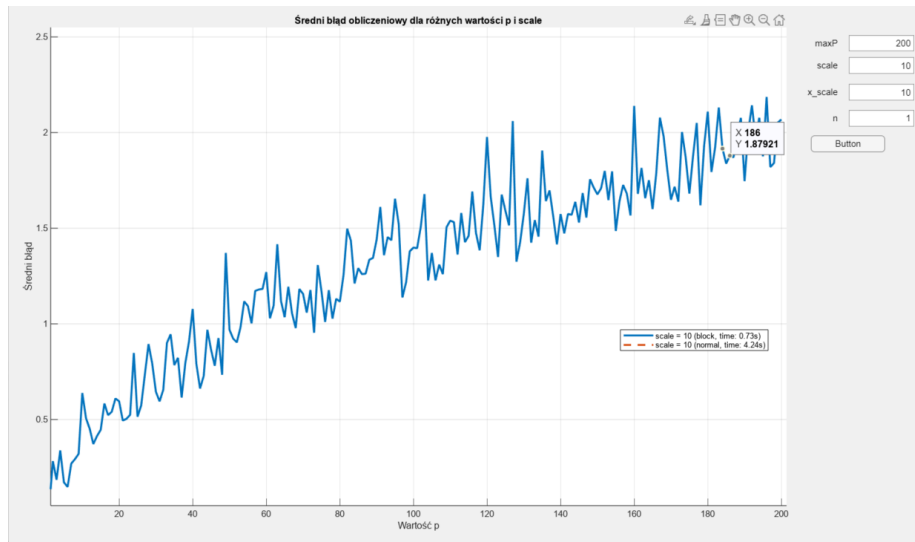
gdzie  $\|\cdot\|_1$  oznacza normę 1 macierzy. Średnia wartość błędu  $E$  jest obliczana na podstawie 10 losowo generowanych macierzy dla każdej pary wartości parametru  $P$  i skali.



## 4.2 Błędy Rozwiązania Układu Równań

Dla każdej wartości parametru  $p$  z pewnego przedziału generujemy  $n$  macierzy i  $n$  układów równań. Następnie rozwiązujemy te równania metodą Cholesky'ego-Banachiewicza, obliczając średni błąd rozwiązania. Poniżej kilka przykładów





## 5 Zastosowania

### Analiza Głównych Składowych (PCA)

Analiza Głównych Składowych (PCA) to technika statystyczna używana do redukcji wymiarowości danych poprzez przekształcenie do nowego układu współrzędnych, w którym pierwsza główna składowa ma największą wariancję, a każda kolejna składowa ma jak największą wariancję pod warunkiem bycia ortogonalną do poprzednich. Metodę blokową używa się przy okazji dekompozycji macierzy kowariancji.

## 6 Wnioski

W ramach przeprowadzonych badań porównano wydajność oraz dokładność dwóch metod rozkładu Cholesky’ego: standardowej oraz blokowej. Poniżej przedstawiono główne wnioski z przeprowadzonych analiz:

1. **Wyższa Wydajność Metody Blokowej:** Metoda blokowa wykazała znacznie wyższą wydajność w porównaniu z metodą standardową, zwłaszcza dla dużych macierzy.
2. **Większe Błędy Metody Blokowej:** Pomimo wyższej wydajności, metoda blokowa ma tendencję do generowania większych błędów numerycznych w porównaniu z metodą standardową. Może to być wynikiem zaokrągleń oraz akumulacji błędów w trakcie przetwarzania bloków.
3. **Podobna Wydajność dla Małych Macierzy:** Dla małych macierzy, różnice w wydajności między metodami blokową a standardową są marginalne, a obie metody wykazują podobne charakterystyki dokładności.
4. **Efektywne wykorzystanie zasobów komputera:** Pracując z dużymi macierzami, metoda blokowa pozwala na podzielenie zadania na podproblemy wymagające mniejszej ilości zasobów.
5. **Preferencja dla Macierzy o Blokach na Przekątnej:** Metoda blokowa jest szczególnie efektywna dla macierzy, które posiadają naturalny podział na bloki na przekątnej. W takich przypadkach, wykorzystanie struktury blokowej macierzy pozwala na znaczące przyspieszenie obliczeń.
6. **Podobne Niekorzystne Przypadki:** Zarówno metoda standardowa, jak i blokowa mają podobne scenariusze, w których ich wydajność jest ograniczona. Należą do nich macierze o szczególnie niekorzystnych właściwościach numerycznych, gdzie akumulacja błędów może znacząco wpływać na dokładność rozwiązania.

Podsumowując, wybór między metodą standardową a blokową powinien być dokonywany na podstawie rozmiaru i struktury macierzy oraz wymagań dotyczących dokładności rozwiązania.