# Sprawozdanie z projektu 2

Wojciech Roszkowski, 320789

## Opis matematyczny

Celem projektu jest implementacja zmodyfikowanej metody Cholesky'ego-Banachiewicza rozwiązywania układu równań liniowych $Ax = b$, gdzie $A \in \mathbb{R}^{n \times n}$ jest dodatnio określoną symetryczną trójdiagonalną, a $b \in \mathbb{R}^n$.

Modyfikacja metody polega na rozkładzie macierzy $A$ na iloczyn macierzy gólnotrójkątnej i jej transpozycji, zamiast dolnotrójkątnej, tj. $A = UU^T$. Okazuje się, że jeżeli macierz $A$ jest trójdiagonalna, symetryczna i dodatnio określona, to macierz $U$ będzie postaci

$$U = \begin{pmatrix} d_1 & s_1 & & & \\ & d_2 & s_2 & & \\ & & d_3 & \ddots & \\ & & & \ddots & s_{n-1} \\ & & & & d_n \end{pmatrix}.$$

Macierz $U$ możemy więc znaleźć, rozwiązując układ równań

$$\begin{pmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{pmatrix} = \begin{pmatrix} d_1 & s_1 & & & \\ & d_2 & s_2 & & \\ & & d_3 & \ddots & \\ & & & \ddots & s_{n-1} \\ & & & & d_n \end{pmatrix} \begin{pmatrix} d_1 & & & & \\ s_1 & d_2 & & & \\ & s_2 & d_3 & & \\ & & \ddots & \ddots & \\ & & & s_{n-1} & d_n \end{pmatrix},$$

gdzie

$$A = \begin{pmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{pmatrix}$$

jest macierzą dodatnio określoną.

Wtedy współczynniki $d_i, s_i$ otrzymujemy z zależności:

$a_n = d_n^2$

$b_i = s_i d_{i+1}$

$a_i = d_i^2 + s_i^2,\ i = 1, \dots, n-1,$

co prowadzi do następującego wzoru rekurencyjnego:

$$d_n = \sqrt{a_n}$$
$$s_i = \frac{b_i}{d_{i+1}}$$
$$d_i = \sqrt{a_i - s_i^2}, i = n-1, \dots, 1.$$

## Związek z rozkładem Cholesky'ego-Banachiewicza $A = LL^T$

Zauważmy następującą zależność:

$$\begin{pmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{pmatrix} = \begin{pmatrix} d_1 & s_1 & & & \\ & d_2 & s_2 & & \\ & & d_3 & \ddots & \\ & & & \ddots & s_{n-1} \\ & & & & d_n \end{pmatrix} \begin{pmatrix} d_1 & & & & \\ s_1 & d_2 & & & \\ & s_2 & d_3 & & \\ & & \ddots & \ddots & \\ & & & s_{n-1} & d_n \end{pmatrix}$$

$$\Updownarrow$$

$$\begin{pmatrix} a_n & b_{n-1} & & & \\ b_{n-1} & a_{n-1} & b_{n-2} & & \\ & b_{n-2} & a_{n-3} & \ddots & \\ & & \ddots & \ddots & b_1 \\ & & & b_1 & a_1 \end{pmatrix} = \begin{pmatrix} d_n & & & & \\ s_{n-1} & d_{n-1} & & & \\ & s_{n-2} & d_{n-2} & & \\ & & \ddots & \ddots & \\ & & & s_1 & d_1 \end{pmatrix} \begin{pmatrix} d_n & s_{n-1} & & & \\ & d_{n-1} & s_{n-2} & & \\ & & d_{n-2} & \ddots & \\ & & & \ddots & s_1 \\ & & & & d_1 \end{pmatrix}.$$

Oznacza to, że macierz $U$ można uzyskać po odpowiedniej permutacji elementów macierzy $L$, którą uzyskamy, rozkładając macierz

$$A' \coloneqq \begin{pmatrix} a_n & b_{n-1} & & & \\ b_{n-1} & a_{n-1} & b_{n-2} & & \\ & b_{n-2} & a_{n-3} & \ddots & \\ & & \ddots & \ddots & b_1 \\ & & & b_1 & a_1 \end{pmatrix}$$

standardową metodą Cholesky'ego-Banachiewicza.

Oczywiście macierz $A'$ jest symetryczna i trójdiagonalna. Aby więc przekonać się, że obydwie metody znajdowania macierzy $U$ będą działały dla dodatnio określonej macierzy $A$, pozostaje uzasadnić, że macierz $A$ jest dodatnio określona wtedy i tylko wtedy, gdy macierz $A'$ jest dodatnio określona. Zauważmy, że miejsce ma następująca równoważność:

$$A \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{pmatrix} \Leftrightarrow A' \begin{pmatrix} c_n \\ c_{n-1} \\ \vdots \\ c_1 \end{pmatrix} = \begin{pmatrix} d_n \\ d_{n-1} \\ \vdots \\ d_1 \end{pmatrix}.$$

Wynika stąd, że macierze $A$ oraz $A'$ mają takie same wartości własne (a odpowiadające im wektory mają zamienioną kolejność elementów). Ponieważ symetryczne macierze rzeczywiste są dodatnio określone wtedy i tylko wtedy, gdy wszystkie ich wartości własne są dodatnie, to wnioskujemy, że macierze $A$ oraz $A'$ będą dodatnio określone jednocześnie.

Jeżeli więc w algorytmie szukania macierzy $U$ pod pierwiastkiem pojawi się liczba ujemna, to będzie to oznaczało, że macierz $A$ nie była dodatnio określona.

## Obliczanie wyznacznika na podstawie rozkładu $A = UU^T$

Ponieważ macierze $U, U^T$ są trójkątne, to ich wyznaczniki będą iloczynami elementów na przekątnej głównej. Dysponując więc macierzą $U$, możemy łatwo obliczyć wyznacznik macierzy $A$ za pomocą wzoru

$$\det A = \det(UU^T) = \det U \det U^T = (\det U)^2 = \left( \prod_{i=1}^{n} d_i \right)^2.$$

## Opis programu

Do implementacji posłużyły następujące funkcje:

- `[A] = pasmowa_g(a, b)` – zwraca macierz górnotrójkątną postaci

$$\begin{pmatrix} a_1 & b_1 & & & \\ & a_2 & b_2 & & \\ & & a_3 & \ddots & \\ & & & \ddots & b_{n-1} \\ & & & & a_n \end{pmatrix},$$

gdzie a = $(a_1, \dots, a_n)$, b = $(b_1, \dots, b_{n-1})$.

- `[A] = pasmowa_s(a, b)` – zwraca macierz symetryczną postaci

$$\begin{pmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{pmatrix},$$

  gdzie a = $(a_1, \dots, a_n)$, b = $(b_1, \dots, b_{n-1})$.

- `[d, s] = rozklad(a, b)` – zwraca wektory d = $(d_1, \dots, d_n)$, s = $(s_1, \dots, s_n)$, takie że

$$\begin{pmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{pmatrix} = \begin{pmatrix} d_1 & s_1 & & & \\ & d_2 & s_2 & & \\ & & d_3 & \ddots & \\ & & & \ddots & s_{n-1} \\ & & & & d_n \end{pmatrix} \begin{pmatrix} d_1 & & & & \\ s_1 & d_2 & & & \\ & s_2 & d_3 & & \\ & & \ddots & \ddots & \\ & & & s_{n-1} & d_n \end{pmatrix},$$

  gdzie a = $(a_1, \dots, a_n)$, b = $(b_1, \dots, b_{n-1})$.

  Funkcja ta oblicza wektory `d, s`, korzystając ze związku rekurencyjnego opisanego w sekcji Opis matematyczny. Jeżeli macierz zadana wektorami `a, b` nie jest dodatnio określona, to funkcja zwróci odpowiedni błąd.

- `[y] = wyznacznik(a, b)` – zwraca wyznacznik macierzy postaci

$$\begin{pmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{pmatrix}.$$

  Funkcja oblicza wyznacznik korzystając z funkcji `[d, s] = rozklad(a, b)`, wymnażając wyrazy zwróconego wektora d i podnosząc wynik do kwadratu.

Dodatkowo napisano poniższe funkcje testujące, które porównują funkcje opisane powyżej z funkcjami wbudowanymi Matlaba:

- `[y] = porownanie_czasow(dt, st)` – porównuje czasy wykonania funkcji `rozklad(a, b)` z funkcją wbudowaną `chol(A)` (gdzie analizujemy zarówno rozkład $A = LL^T$, jak i rozkład $A' = LL^T$, gdzie

$$A = \begin{pmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{pmatrix}, A' := \begin{pmatrix} a_n & b_{n-1} & & & \\ b_{n-1} & a_{n-1} & b_{n-2} & & \\ & b_{n-2} & a_{n-3} & \ddots & \\ & & \ddots & \ddots & b_1 \\ & & & b_1 & a_1 \end{pmatrix}).$$

Drugą porównywaną wielkością są czasy wykonania funkcji `wyznacznik(a, b)` z funkcją wbudowaną `det(A)`.

Argumenty `dt, st` są macierzami postaci

$$dt = \begin{pmatrix} d_{11} & \cdots & d_{1n} \\ \vdots & \ddots & \vdots \\ d_{m1} & \cdots & d_{mn} \end{pmatrix}, st = \begin{pmatrix} s_{11} & \cdots & s_{1,n-1} \\ \vdots & \ddots & \vdots \\ s_{m1} & \cdots & s_{m,n-1} \end{pmatrix},$$

a każdy wiersz zadaje dodatnio określoną macierz

$$A_j = \begin{pmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{pmatrix} = U_j U_j^T = \begin{pmatrix} d_{j1} & s_{j1} & & & \\ & d_{j2} & s_{j2} & & \\ & & d_{j3} & \ddots & \\ & & & \ddots & s_{j,n-1} \\ & & & & d_{jn} \end{pmatrix} \begin{pmatrix} d_{j1} & s_{j1} & & & \\ & d_{j2} & s_{j2} & & \\ & & d_{j3} & \ddots & \\ & & & \ddots & s_{j,n-1} \\ & & & & d_{jn} \end{pmatrix}^T,$$

która zadaje argumenty porównywanych funkcji w j-tym teście (j = 1,…,m).

Tabela, którą zwraca funkcja `porownanie_czasow(dt, st)` uwzględnia wyniki dla wszystkich $m$ wykonanych testów.

- `[y] = porownanie_wynikow(dt, st)` – zwraca dwie tabele porównujące błędy wygenerowane przez funkcje: `rozklad(a, b)`, `chol(A_odwrocona)`, `wyznacznik(a, b)`, `det(A)`, z wynikami prawidłowymi, zadanymi przez macierze `dt` i `st`. Oznacza to, że błędy kolejnych funkcji obliczane są w sposób następujący:
  - $\Delta_1 = \left\| \begin{pmatrix} d_1 & s_1 & & & \\ & d_2 & s_2 & & \\ & & d_3 & \ddots & \\ & & & \ddots & s_{n-1} \\ & & & & d_n \end{pmatrix} - \begin{pmatrix} d_1' & s_1' & & & \\ & d_2' & s_2' & & \\ & & d_3' & \ddots & \\ & & & \ddots & s_{n-1}' \\ & & & & d_n' \end{pmatrix} \right\|_2$ – błąd funkcji `rozklad(a,b)`, gdzie zwrócone przez funkcję wektory mają postać $d' = (d_1', \dots d_n'), s' = (s_1', \dots, s_{n-1}')$;
  - $\Delta_2 = \|\text{A\_odwrocona} - A''\|_2 = \left\| \begin{pmatrix} d_n & s_{n-1} & & & \\ & d_{n-2} & s_{n-2} & & \\ & & d_{n-3} & \ddots & \\ & & & \ddots & s_1 \\ & & & & d_1 \end{pmatrix} - A'' \right\|_2$ – błąd funkcji `chol(A_odwrocona)`, gdzie $A''$ - zwrócona przez funkcję macierz;
  - $\Delta_3 = |(\prod_{i=1}^n d_i)^2 - \text{wyznacznik(a, b)}|$, – błąd funkcji `wyznacznik(a, b)`;
  - $\Delta_4 = |(\prod_{i=1}^n d_i)^2 - \det(\text{A})|$, – błąd funkcji `det(A)`.

  Argumenty wejściowe `dt, st` są takie same, jak w przypadku funkcji `porownanie_czasow(dt, st)`. Pierwsza tabela zawiera błędy bezwzględne we wszystkich m testach, a druga – błędy względne.

Przykłady opisane w dalszej części zostały zaimplementowane w skrypcie `skrypt_test.m`

# Przykłady

## 1. Macierz $5 \times 5$ o współczynnikach całkowitych

Ponieważ ten przykład odnosi się do macierzy małych rozmiarów, a dobór całkowitych współczynników nie powinien generować błędów numerycznych, możemy najpierw upewnić się, że wyniki są zgodne z przytoczoną wyżej teorią.

Przykładowo, przyjmując wektory

$$d = (1, 2, 3, 4, 5), s = (-1, -2, -3, -4)$$

możemy otrzymać macierz

$$A = UU^T = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 3 & -3 & 0 \\ 0 & 0 & 0 & 4 & -4 \\ 0 & 0 & 0 & 0 & 5 \end{pmatrix} \begin{pmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 3 & -3 & 0 \\ 0 & 0 & 0 & 4 & -4 \\ 0 & 0 & 0 & 0 & 5 \end{pmatrix}^T$$
$$= \begin{pmatrix} 2 & -2 & 0 & 0 & 0 \\ -2 & 8 & -6 & 0 & 0 \\ 0 & -6 & 18 & -12 & 0 \\ 0 & 0 & -12 & 32 & -20 \\ 0 & 0 & 0 & -20 & 25 \end{pmatrix}$$

i mamy pewność, że macierz ta jest dodatnio określona i rozkłady $A = UU^T, A = LL^T$ istnieją.

Zatem argumenty funkcji `[d, s] = rozklad(a, b)` są równe

$$a = (2, 8, 18, 32, 25), b = (-2, -6, -12, -20).$$

„Odwrócona” macierz $A$, której rozkład $A' = LL^T$ zadaje „odwróconą” macierz $U$ jest postaci

$$A' = LL^T = \begin{pmatrix} 5 & -4 & 0 & 0 & 0 \\ 0 & 4 & -3 & 0 & 0 \\ 0 & 0 & 3 & -2 & 0 \\ 0 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}^T \begin{pmatrix} 5 & -4 & 0 & 0 & 0 \\ 0 & 4 & -3 & 0 & 0 \\ 0 & 0 & 3 & -2 & 0 \\ 0 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$
$$= \begin{pmatrix} 25 & -20 & 0 & 0 & 0 \\ -20 & 32 & -12 & 0 & 0 \\ 0 & -12 & 18 & -6 & 0 \\ 0 & 0 & -6 & 8 & -2 \\ 0 & 0 & 0 & -2 & 2 \end{pmatrix}.$$

Wyznacznik macierzy $A$ jest natomiast równy $\det A = 14400$.

Teraz możemy porównać wyznaczone powyżej wyniki z wynikami zwróconymi przez Matlaba. Kod wyznaczający wyniki:

```
d = [1 2 3 4 5]; s = [-1 -2 -3 -4];
U = pasmowa_g(d, s)
A = U*U'
a = diag(A)';
b = diag(A, 1)';
A_odwrocona = pasmowa_s(a(5:-1:1), b(4:-1:1));
L = chol(A_odwrocona)'
wyznacznik_funkcja_det = det(A)
wyznacznik_funkcja_wyznacznik = wyznacznik(a,b)
wyznacznik_iloczyn_wyrazow_d = prod(d)^2
```

Zwrócone dane:

```
U =

     1    -1     0     0     0
     0     2    -2     0     0
     0     0     3    -3     0
     0     0     0     4    -4
     0     0     0     0     5


A =

     2    -2     0     0     0
    -2     8    -6     0     0
     0    -6    18   -12     0
     0     0   -12    32   -20
     0     0     0   -20    25


L =

     5     0     0     0     0
    -4     4     0     0     0
     0    -3     3     0     0
     0     0    -2     2     0
     0     0     0    -1     1


wyznacznik_funkcja_det =

       14400


wyznacznik_funkcja_wyznacznik =

       14400


wyznacznik_iloczyn_wyrazow_d =

       14400
```

Widzimy więc, że funkcje działają w tym przypadku poprawnie.

Możemy też zauważyć, że rozkład $LL^T$ macierzy $A$ nie przypomina rozkładu $LL^T$ „odwróconej" macierzy $A'$ i nie zawiera już współczynników całkowitych:

```
>> chol(A)'

ans =

    1.4142         0         0         0         0
   -1.4142    2.4495         0         0         0
         0   -2.4495    3.4641         0         0
         0         0   -3.4641    4.4721         0
         0         0         0   -4.4721    2.2361
```

Następnie będziemy chcieli przeanalizować czasy wykonania i błędy generowane przez napisane funkcje w porównaniu z funkcjami wbudowanymi.

## 2. Macierze $U$ rozmiarów $10 \times 10$ o współczynnikach całkowitych z przedziału $[1, 10]$

W tym przypadku porównamy 10 macierzy, których współczynniki zadane są wektorami losowych liczb całkowitych z przedziału $[1, 10]$.

Poniższa tabela przedstawia czasy zwrócenia odpowiednich macierzy oraz ich wyznaczników wykonane przez funkcje napisane samodzielnie oraz przez funkcje wbudowane, przy czym funkcję `chol` przetestowaliśmy dla macierzy $A = UU^T$ oraz dla „odwróconej” macierzy $A'$:

| A = UU^T | A = LL^T | A' = LL^T | wyznacznik(a, b) | det(A) |
|---|---|---|---|---|
| 0.0001228 | 0.0002034 | 0.0002501 | 0.0001992 | 6.5e-06 |
| 0.0001935 | 0.0001822 | 0.0002058 | 9.8e-06 | 3.2e-06 |
| 1.3e-06 | 1.34e-05 | 1.23e-05 | 1.2e-06 | 1.7e-06 |
| 8e-07 | 1.04e-05 | 1.09e-05 | 1e-06 | 1.5e-06 |
| 8e-07 | 1e-05 | 1.09e-05 | 9e-07 | 1.2e-06 |
| 7e-07 | 1.04e-05 | 1.08e-05 | 9e-07 | 1.3e-06 |
| 8e-07 | 9.7e-06 | 1.05e-05 | 9e-07 | 1.2e-06 |
| 8e-07 | 1.01e-05 | 1.08e-05 | 9e-07 | 1.2e-06 |
| 7e-07 | 1.02e-05 | 1.04e-05 | 9e-07 | 1.2e-06 |
| 7e-07 | 1.06e-05 | 1.07e-05 | 9e-07 | 1.3e-06 |

Widzimy więc, że nawet dla macierzy o stosunkowo małych rozmiarach i współczynnikach, napisane funkcje działają nieznacznie szybciej. Nie widać za to różnicy w czasach wykonania rozkładów macierzy $A$ oraz $A'$. Różnica w czasach działania jest bardziej widoczna w przypadku rozkładu macierzy, niż w przypadku obliczenia wyznacznika macierzy $A$.

Następne dwie tabele dotyczą błędów wygenerowanych przez funkcje wbudowane oraz funkcje napisane samodzielnie. Pierwsza tabela przedstawia błędy bezwzględne, a druga – błędy względne:

| rozklad(a, b) | chol(A_odwrocona) | wyznacznik(a, b) | det(A) |
|---|---|---|---|
| 0 | 0 | 0 | 0.42542 |
| 0 | 0 | 0 | 1.4531 |
| 0 | 0 | 0 | 0.3623 |
| 0 | 0 | 0 | 0.0046387 |
| 0 | 0 | 0 | 0.25 |
| 0 | 0 | 0 | 12.586 |
| 0 | 0 | 0 | 0.03125 |
| 0 | 0 | 0 | 0.00043488 |
| 0 | 0 | 0 | 1.5 |
| 0 | 0 | 0 | 0.16406 |

| rozklad(a, b) | chol(A_odwrocona) | wyznacznik(a, b) | det(A) |
|---|---|---|---|
| 0 | 0 | 0 | 1.2508e-12 |
| 0 | 0 | 0 | 1.1121e-14 |
| 0 | 0 | 0 | 4.6233e-14 |
| 0 | 0 | 0 | 7.7402e-15 |
| 0 | 0 | 0 | 2.0417e-16 |
| 0 | 0 | 0 | 4.1162e-13 |
| 0 | 0 | 0 | 1.2618e-15 |
| 0 | 0 | 0 | 8.1922e-15 |
| 0 | 0 | 0 | 2.4206e-15 |
| 0 | 0 | 0 | 2.9889e-15 |

Wnioskujemy więc, że obydwie funkcje służące do rozkładu macierzy $A$ (w przypadku funkcji `rozklad`) oraz macierzy $A'$ = A_odwrocona (w przypadku funkcji `chol`) poradziły sobie bezbłędnie. Funkcja `wyznacznik` również zwróciła bezbłędny wynik, a funkcja `det` – bardzo niewielki błąd.

## 3. Macierze $U$ rozmiarów $10 \times 10$ o współczynnikach rzeczywistych z przedziału $(0, 10)$

Tabela przedstawiająca czasy wykonania:

| A = UU^T | A = LL^T | A' = LL^T | wyznacznik(a, b) | det(A) |
|---|---|---|---|---|
| 7.3e-06 | 3.09e-05 | 1.66e-05 | 4.1e-06 | 4.5e-06 |
| 1.1e-06 | 1.11e-05 | 1.22e-05 | 9e-07 | 1.6e-06 |
| 7e-07 | 9.8e-06 | 1.02e-05 | 9e-07 | 1.2e-06 |
| 7e-07 | 9.7e-06 | 1.31e-05 | 1e-06 | 1.1e-06 |
| 8e-07 | 1.03e-05 | 1.03e-05 | 9e-07 | 1e-06 |
| 7e-07 | 1.02e-05 | 1.85e-05 | 1e-06 | 1.1e-06 |
| 8e-07 | 1.02e-05 | 1.07e-05 | 9e-07 | 1.2e-06 |
| 7e-07 | 1.2e-05 | 1.23e-05 | 9e-07 | 1.2e-06 |
| 7e-07 | 1.01e-05 | 2.46e-05 | 9e-07 | 1e-06 |
| 7e-07 | 9.8e-06 | 1.03e-05 | 9e-07 | 8e-07 |

Fakt, że współczynniki nie są całkowite, nie wpływa na czas wykonania funkcji – wyniki są porównywalne z wcześniejszym przypadkiem.

Tabele przedstawiające błędy bezwzględne i względne:

| rozklad(a, b) | chol(A_odwrocona) | wyznacznik(a, b) | det(A) |
|---|---|---|---|
| 1.6778e-14 | 6.391e-15 | 5440 | 4416 |
| 4.7961e-15 | 6.628e-15 | 576 | 1280 |
| 8.7954e-15 | 4.4325e-15 | 60 | 42 |
| 0 | 1.4371e-15 | 0 | 1024 |
| 7.1306e-15 | 6.597e-15 | 608 | 64 |
| 2.0324e-15 | 2.9196e-15 | 224 | 480 |
| 1.3188e-14 | 1.3188e-14 | 3840 | 192 |
| 2.7214e-15 | 5.1495e-15 | 24 | 848 |
| 8.8818e-16 | 9.5113e-15 | 0 | 512 |
| 3.225e-14 | 2.9568e-14 | 8256 | 192 |

| rozklad(a, b) | chol(A_odwrocona) | wyznacznik(a, b) | det(A) |
|---|---|---|---|
| 1.0466e-15 | 3.9867e-16 | 1.14e-14 | 9.2538e-15 |
| 3.1741e-16 | 4.3865e-16 | 2.0081e-15 | 4.4625e-15 |
| 6.1944e-16 | 3.1217e-16 | 4.8984e-15 | 3.4288e-15 |
| 0 | 9.3088e-17 | 0 | 4.4105e-16 |
| 4.7549e-16 | 4.399e-16 | 6.792e-15 | 7.1495e-16 |
| 1.2303e-16 | 1.7673e-16 | 1.1384e-15 | 2.4394e-15 |
| 8.2464e-16 | 8.2464e-16 | 9.7829e-15 | 4.8915e-16 |
| 1.7488e-16 | 3.3091e-16 | 4.1701e-16 | 1.4734e-14 |
| 5.2231e-17 | 5.5933e-16 | 0 | 1.1758e-16 |
| 2.0596e-15 | 1.8884e-15 | 1.5259e-14 | 3.5486e-16 |

Widzimy więc, że w tym przypadku obydwie metody generują błędy, jednak są one podobnych, bardzo niewielkich rozmiarów.

## 4. Macierze $U$ rozmiarów $10 \times 10$ o dużych współczynnikach całkowitych

Porównamy macierze o współczynnikach z przedziałów: $[9 \cdot 10^6, 10^7], [9 \cdot 10^7, 10^8], [9 \cdot 10^{14}, 10^{15}]$.

Nie widać znaczących różnic w czasach wykonania funkcji w porównaniu z wcześniejszymi przypadkami:

| A = UU^T | A = LL^T | A' = LL^T | wyznacznik(a, b) | det(A) |
|---|---|---|---|---|
| 0.0006078 | 8.48e-05 | 1.49e-05 | 2.2e-05 | 3.6e-06 |
| 3.8e-06 | 3.39e-05 | 1.19e-05 | 1.3e-06 | 1.5e-06 |
| 1.2e-06 | 1.95e-05 | 1.27e-05 | 1.3e-06 | 1.2e-06 |
| 1e-06 | 1.1e-05 | 2.62e-05 | 1e-06 | 9e-07 |
| 9e-07 | 1.06e-05 | 1.05e-05 | 1.1e-06 | 1.1e-06 |
| 9e-07 | 1.24e-05 | 1.06e-05 | 1e-06 | 1e-06 |
| 9e-07 | 1.05e-05 | 1.05e-05 | 1e-06 | 9e-07 |
| 8e-07 | 1.08e-05 | 1.06e-05 | 1.1e-06 | 9e-07 |
| 9e-07 | 1.05e-05 | 1.04e-05 | 1e-06 | 9e-07 |
| 9e-07 | 1.08e-05 | 1.06e-05 | 1e-06 | 9e-07 |

| A = UU^T | A = LL^T | A' = LL^T | wyznacznik(a, b) | det(A) |
|---|---|---|---|---|
| 7.3e-06 | 2.29e-05 | 1.2e-05 | 3e-06 | 3.1e-06 |
| 1.3e-06 | 1.16e-05 | 1.1e-05 | 1.1e-06 | 1.4e-06 |
| 9e-07 | 1.05e-05 | 1.06e-05 | 3.14e-05 | 1.1e-06 |
| 1.2e-06 | 1.14e-05 | 1.07e-05 | 1.1e-06 | 1.2e-06 |
| 1.3e-06 | 1.07e-05 | 1.01e-05 | 1e-06 | 1e-06 |
| 9e-07 | 1.05e-05 | 1.03e-05 | 1e-06 | 1e-06 |
| 9e-07 | 1.08e-05 | 2.28e-05 | 1.6e-06 | 1.5e-06 |
| 1.1e-06 | 1.04e-05 | 1.03e-05 | 1e-06 | 1e-06 |
| 1e-06 | 1.03e-05 | 1.01e-05 | 1e-06 | 9e-07 |
| 9e-07 | 1.04e-05 | 1.09e-05 | 1e-06 | 9e-07 |

| A = UU^T | A = LL^T | A' = LL^T | wyznacznik(a, b) | det(A) |
|---|---|---|---|---|
| 6e-06 | 2.6e-05 | 1.24e-05 | 3.5e-06 | 3.3e-06 |
| 1.7e-06 | 1.21e-05 | 1.15e-05 | 1.2e-06 | 1.4e-06 |
| 9e-07 | 1.08e-05 | 1.1e-05 | 1e-06 | 1.1e-06 |
| 9e-07 | 1.05e-05 | 1.07e-05 | 1.1e-06 | 1e-06 |
| 9e-07 | 1.08e-05 | 3.03e-05 | 1.04e-05 | 1e-06 |
| 1.51e-05 | 1.11e-05 | 1.56e-05 | 1.6e-06 | 1.4e-06 |
| 1e-06 | 1.12e-05 | 1.06e-05 | 1.1e-06 | 1.1e-06 |
| 1.1e-06 | 1.05e-05 | 1.08e-05 | 1.1e-06 | 1e-06 |
| 1e-06 | 1.04e-05 | 1.03e-05 | 1.1e-06 | 9e-07 |
| 1e-06 | 1.06e-05 | 1.05e-05 | 1.1e-06 | 9e-07 |

Błędy w obliczaniu macierzy $U$ i $L$ zaczynają pojawiać się w przypadku liczb z przedziału $[9 \cdot 10^7, 10^8]$, tak samo jak błędy w obliczaniu wyznacznika funkcją `wyznacznik[a, b]`. Błędy są jednak również bardzo niewielkie. Funkcje wbudowane i napisane samodzielnie generują wówczas błędy podobnych rozmiarów. Poniższe tabele przedstawiają błędy względne we wszystkich trzech przypadkach:

| rozklad(a, b) | chol(A_odwrocona) | wyznacznik(a, b) | det(A) |
|---|---|---|---|
| 0 | 0 | 0 | 4.4371e-16 |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 7.4726e-16 |
| 0 | 0 | 0 | 6.2726e-16 |
| 0 | 0 | 0 | 3.3246e-16 |
| 0 | 0 | 0 | 3.3557e-16 |
| 0 | 0 | 0 | 7.5162e-16 |
| 0 | 0 | 0 | 5.1804e-16 |
| 0 | 0 | 0 | 3.812e-16 |
| 0 | 0 | 0 | 7.1487e-16 |

| rozklad(a, b) | chol(A_odwrocona) | wyznacznik(a, b) | det(A) |
|---|---|---|---|
| 7.8165e-17 | 7.8165e-17 | 0 | 2.3025e-15 |
| 0 | 0 | 0 | 4.6788e-15 |
| 7.902e-17 | 0 | 0 | 5.293e-15 |
| 0 | 0 | 0 | 1.1804e-15 |
| 0 | 1.3884e-16 | 0 | 4.6029e-15 |
| 0 | 0 | 0 | 2.1521e-15 |
| 2.9609e-16 | 2.9609e-16 | 2.9629e-15 | 4.7736e-15 |
| 7.8149e-17 | 7.8149e-17 | 0 | 1.2654e-15 |
| 0 | 0 | 0 | 5.928e-15 |
| 1.5349e-16 | 2.3325e-16 | 1.7202e-15 | 1.7202e-15 |

| rozklad(a, b) | chol(A_odwrocona) | wyznacznik(a, b) | det(A) |
|---|---|---|---|
| 1.6455e-16 | 4.6153e-16 | 2.0502e-15 | 1.3047e-15 |
| 2.0583e-16 | 1.2793e-16 | 2.7315e-15 | 9.5603e-16 |
| 2.6969e-16 | 2.4507e-16 | 5.1778e-15 | 3.6124e-15 |
| 5.9634e-16 | 5.3022e-16 | 7.7948e-15 | 4.9603e-15 |
| 3.4014e-16 | 4.6425e-16 | 5.2717e-15 | 2.4221e-15 |
| 1.1623e-16 | 1.6746e-16 | 2.8397e-16 | 2.4137e-15 |
| 1.295e-16 | 5.1333e-16 | 1.9311e-15 | 1.5903e-15 |
| 1.6556e-16 | 2.548e-16 | 0 | 1.2062e-15 |
| 4.8725e-16 | 6.0965e-16 | 6.9724e-15 | 6.5367e-15 |
| 3.6023e-16 | 4.3078e-16 | 3.904e-15 | 2.074e-15 |

## 5. Macierz A „ledwo” dodatnio określona

Jeżeli $A = UU^T$, gdzie macierz $U$ na głównej przekątnej ma dodatnie elementy bliskie zeru, to zarówno funkcja `rozklad(a, b)` i funkcja `chol(A_odwrocona)` nie zakwalifikują podanej macierzy jako dodatnio określoną. Przykładowo, macierz `A` zadana przez wektory $d \in \mathbb{R}^{10}, s = (1,2,3, \dots, 9)$, gdzie elementy wektora `d` nie przekraczają wartości 0.01, generuje błąd w obydwu funkcjach. Można również sprawdzić, że wartości własne takiej macierzy istotnie są błędnie obliczone (najmniejsza z nich jest bardzo niewielką liczbą ujemną, co nie powinno mieć miejsca):

```
>> eig(A)

ans =

  -0.000000000000000
   1.000043085653620
   4.000076792820334
   9.000011601407932
  15.999953817899117
  25.000014798551884
  36.000136161970843
  48.999976857029793
  63.999969004172755
  81.000358494608165
```

## 6. Macierze $U$ rozmiarów $1000 \times 1000$ o współczynnikach z przedziału $(-1,1)$

Następnie będziemy chcieli porównać macierze o dużych rozmiarach, ale małych współczynnikach. W takim przypadku różnica w czasach wykonania funkcji jest już znacząca. Funkcje `rozklad(a, b)` oraz `wyznacznik(a, b)` są zdecydowanie szybsze od funkcji wbudowanych:

| A = UU^T | A = LL^T | A' = LL^T | wyznacznik(a, b) | det(A) |
|---|---|---|---|---|
| 3.28e-05 | 0.034622 | 0.028064 | 2.45e-05 | 0.0063842 |
| 2.5e-05 | 0.041883 | 0.033568 | 3.24e-05 | 0.012802 |
| 2.44e-05 | 0.025005 | 0.02716 | 2.47e-05 | 0.00628 |
| 3.01e-05 | 0.02414 | 0.02668 | 6.19e-05 | 0.0096304 |
| 3.19e-05 | 0.023563 | 0.026682 | 2.44e-05 | 0.0062674 |
| 2.41e-05 | 0.025019 | 0.02424 | 2.46e-05 | 0.007538 |
| 2.47e-05 | 0.02508 | 0.024033 | 4.4e-05 | 0.0088807 |
| 2.4e-05 | 0.023763 | 0.026344 | 2.73e-05 | 0.0071348 |
| 2.37e-05 | 0.024209 | 0.02726 | 2.5e-05 | 0.0064296 |
| 2.66e-05 | 0.025368 | 0.027111 | 4.19e-05 | 0.0072006 |

Przewaga funkcji `rozklad` nad funkcją `chol` jest kilkukrotnie większa, niż przewaga funkcji `wyznacznik` nad funkcją `det`.

Błędy generowane przez wszystkie funkcje są jednak podobnych, małych rozmiarów. Poniższa tabela przedstawia błędy względne:

| rozklad(a, b) | chol(A_odwrocona) | wyznacznik(a, b) | det(A) |
|---|---|---|---|
| 1.055e-15 | 7.1634e-16 | 3.8882e-15 | 2.6353e-14 |
| 1.0756e-15 | 1.0756e-15 | 1.2516e-14 | 5.7765e-16 |
| 7.624e-16 | 1.3057e-15 | 7.7255e-15 | 2.2505e-14 |
| 5.7515e-16 | 6.5731e-16 | 3.7447e-15 | 9.9235e-15 |
| 1.0845e-15 | 9.2974e-16 | 5.6201e-14 | 2.3646e-14 |
| 1.702e-15 | 1.702e-15 | 1.3984e-14 | 1.0186e-14 |
| 1.1391e-15 | 1.7296e-15 | 2.2036e-14 | 2.5232e-15 |
| 9.4437e-16 | 1.1376e-15 | 1.8186e-14 | 0 |
| 1.6107e-15 | 1.6107e-15 | 3.0414e-14 | 5.1742e-15 |
| 1.0695e-15 | 7.6172e-16 | 8.5615e-15 | 1.824e-14 |

## 7. Macierz $U$ rozmiarów $10000 \times 10000$ o współczynnikach z przedziału $(-1,1)$

W tym przypadku czasy wykonania funkcji wbudowanych są bardzo duże w stosunku do poprzednich przypadków, dlatego przeanalizujemy tylko jedną macierz o tak dużych rozmiarach:

| A = UU^T | A = LL^T | A' = LL^T | wyznacznik(a, b) | det(A) |
|---|---|---|---|---|
| 0.0007407 | 4.5841 | 4.5644 | 0.0001986 | 3.2601 |

Widzimy więc, że w przypadku dużych macierzy przewaga funkcji napisanych samodzielnie jest bardzo znacząca z uwagi na czas wykonania. Funkcje zwracające wyznacznik zwracają jednak w obydwu przypadkach błędną, zerową wartość, więc tylko funkcja `rozklad` może okazać się przydatna dla tak dużych macierzy – poniższa tabela również przedstawia błędy względne porównywanych funkcji:

| rozklad(a, b) | chol(A_odwrocona) | wyznacznik(a, b) | det(A) |
|---|---|---|---|
| 4.4924e-15 | 5.9501e-15 | NaN | NaN |

## Przykład zastosowania

W analizie numerycznej macierze pasmowe często pojawiają się przy zagadnieniach takich jak **metoda elementu skończonego**. Jest ona metodą numerycznego rozwiązywania cząstkowych równań różniczkowych, polegającego na podziale rozważanego obszaru na skończoną liczbę podobszarów i w efekcie wyznaczeniu rozwiązania równania tylko w skończonej liczbie punktów.

Taka procedura prowadzi do sprowadzenia równania różniczkowego do układu równań liniowych, który charakteryzowany jest przez dużą, rzadką, dodatnio określoną macierz. Rozważane w projekcie macierze pasmowe są więc szczególnym przypadkiem takich macierzy. Zaimplementowana metoda nie tylko pozwala na bardziej efektywne przechowywanie informacji o takiej macierzy (za pomocą wektorów a i b), ale też może znacząco przyspieszyć czas rozwiązywania takich równań przy wykorzystaniu rozkładu Cholesky'ego-Banachiewicza.