# Sprawozdanie
# Metody Numeryczne 2, laboratorium 5

Grzegorz Rozdzialik (D4, grupa lab. 2)

10 grudnia 2016

## 1 Zadanie

### Temat **5**, zadanie **22**:

Stosując metodę potęgową (z normowaniem) oraz deflację $A_1 = A - \lambda x x^*$ oblicz wszystkie wartości własne macierzy trójdiagonalnej $A$, gdzie $a_{k,k} = 5, a_{k,k-1} = 2 + i, a_{k,k+1} = 2 - i$.

Niech $n \in \mathbb{N}$. Macierz $A \in \mathbb{C}^{n \times n}$ ma postać:

$$A = \begin{bmatrix} 5 & 2-i & 0 & 0 & \dots & 0 & 0 & 0 \\ 2+i & 5 & 2-i & 0 & \dots & 0 & 0 & 0 \\ 0 & 2+i & 5 & 2-i & \dots & 0 & 0 & 0 \\ 0 & 0 & 2+i & 5 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 5 & 2-i & 0 \\ 0 & 0 & 0 & 0 & \dots & 2+i & 5 & 2-i \\ 0 & 0 & 0 & 0 & \dots & 0 & 2+i & 5 \end{bmatrix}$$

Łatwo zauważyć, że macierz $A$ jest diagonalnie silnie dominująca kolumnowo i wierszowo.

Należy znaleźć wszystkie własności własne $\lambda$ macierzy $A$, czyli takie, że:

$$\exists x \in (\mathbb{C} \setminus 0) \quad Ax = \lambda x$$

Wartości własne $\lambda$ są pierwiastkami wielomianu charakterystycznego macierzy A:

$$X(\lambda) = \det(A - \lambda I)$$

Z podstawowego twierdzenia algebry wiemy, że pierwiastków wielomianu $X(\lambda)$ jest $n$, co do krotności.

# 2 Opis metody

Niech $A_0 = A$.

Ponieważ metoda potęgowa znajduje dominującą wartość własną macierzy $A_i$ ($i = 0, \ldots, n-1$), należy zastosować deflację. Po znalezieniu dominującej wartości własnej $\lambda_i$ macierzy $A_i$ oraz wektora własnego $x_i \in \mathbb{C}^n$ związanego z wartością własną $\lambda_i$ macierz $A_{i+1}$ otrzymujemy w następujący sposób:

$$A_{i+1} = A_i - \lambda_i x_i x_i^*$$

Wartość własna $\lambda_{i+1}$ macierzy $A_{i+1}$ jest mniejsza co do modułu od wartości własnej $\lambda_i$ ($|\lambda_{i+1}| < |\lambda_i|$), więc metoda potęgowa zastosowana na macierzy $A_{i+1}$ znajdzie wartość własną $\lambda_{i+1}$.

Niech $\delta \in \mathbb{R}$ będzie zadaną dokładnością przybliżenia wartości własnej $\lambda_i$. Mając przybliżenie początkowej $x_i^{(0)} \in \mathbb{C}^n$ wektora własnego odpowiadającego wartości własnej $\lambda_i$ mamy:

$$\begin{aligned} y_i^{(k)} &= A_i x_i^{(k-1)} \\ x_i^{(k)} &= \frac{y_i^{(k)}}{||y_i^{(k)}||_2} \end{aligned} \tag{1}$$

$$\lambda_i^{(k)} = \langle y_i^{(k)}, x_i^{(k-1)} \rangle \tag{2}$$

gdzie równania (1) przybliżają wektor początkowy związany z dominującą wartością własną macierzy $A_i$, a równanie (2) przybliża tą dominującą wartość własną.

Przybliżanie kontynuujemy, aż zostanie spełniony warunek stopu:

$$|\lambda_i^{(k)} - \lambda_i^{(k-1)}| < \delta$$

Obliczenia powtarzamy dla $i = 0, \ldots, n-1$.

Po znalezieniu wszystkich wartości i wektorów własnych można sprawdzić jak dokładne obliczone przybliżenia. W tym celu zdefiniujmy:

$$e_i = Ax_i - \lambda_i x_i \qquad i = 0, 1, \ldots, n-1$$

$$E = \begin{bmatrix} e_0 & e_1 & e_2 & \ldots & e_{n-1} \end{bmatrix}$$

Licząc normę z macierzy $E$ otrzymujemy konkretną wartość oznaczającą dokładność znalezionych wartości i wektorów własnych.

# 3 Implementacja metody

Implementacja metody podzielona jest na następujące kroki:

1. Konstruowanie macierzy $A$ rozmiaru $n \times n$ (funkcja `constructMatrix`)
2. Wykonanie metody potęgowej z normowaniem w celu znalezienia dominującej wartości własnej $\lambda_i$ macierzy $A_i$ (funkcja `powerIteration`)
3. Powtarzanie kroku związanego z metodą potęgową oraz deflacja po znalezieniu $\lambda_i$ w celu znalezienia wszystkich wartości własnych macierzy $A$ (funkcja `findEigenvaluesAndVectors`)
4. Obliczenie macierzy błędu $E$ (funkcja `calculateErrorMatrix`)

# 4 Poprawność metody

Biorąc wektor $x_i^{(0)} = [111\ldots1]^T$ za przybliżenie początkowe metoda jest poprawna, ponieważ wektor ten ma każdą niezerową składową, zatem daje się przybliżyć do dowolnego wektora w przestrzeni wektorów własnych.

Po wykonaniu testów stwierdzam, że metoda jest poprawna, znajduje wszystkie wartości własne macierzy $A$ oraz odpowiadające im wektory własne, a jej wyniki są zbliżone do tych z funkcji `eig` realizującej to samo zadanie, dostępnej w Matlabie.

# 5 Przykłady

Z powodu niewielkiej liczby parametrów w zadaniu przykłady pokazują porównanie metody potęgowej z normowaniem oraz metody `eig` dostępnej w Matlabie.

W każdym przykładzie przyjęty jest limit iteracji równy 100, ale w większości przypadków nie został on osiągnięty.

**Przykład 1** $n = 5, \delta = 0.1$

| | metoda potęgowa | funkcja `eig` |
|---|---|---|
| maksymalny błąd $Ax_i - \lambda_i x_i$ | 16.0 | 14.7 |
| norma macierzy błędu $E$ | 22.2 | 22.5 |
| czas przybliżania | 0.4330 ms | 0.0912 ms |

Pod względem dokładności obie metody są podobne, jednak funkcja `eig` dominuje pod względem szybkości - jest ponad czterokrotnie szybsza.

**Przykład 2** $n = 5, \delta = 10^{-5}$

Zwiększenie dokładności spowodowało zmniejszenie maksymalnego błędu $Ax_i - \lambda_i x_i$ metody potęgowej do 14.7 (wynik podobny do funkcji `eig`) przy jednoczesnym wydłużeniu czasu przybliżania do 2.61 ms (ponad sześciokrotny wzrost).

**Przykład 3** $n = 20, \delta = 0.1$

| | metoda potęgowa | funkcja `eig` |
|---|---|---|
| maksymalny błąd $Ax_i - \lambda_i x_i$ | 39.0 | 29.6 |
| norma macierzy błędu $E$ | 108.7 | 110.2 |
| czas przybliżania | 6.7451 ms | 0.9639 ms |

Maksymalny odchylenie A * x - lambda * x: 3.906204e+01 Norma wektora bledu: 1.087447e+02

Funkcja eig dostepna w Matlabie: Maksymalny odchylenie A * x - lambda * x: 2.960347e+01 Norma wektora bledu: 1.101882e+02

Znalezione wartosci wlasne znajduja sie w zmiennej "eigenvalues"

Czas obliczania wartosci wlasnych: * metoda potegowa: 6.7451 ms * funkcja eig: 0.9639 ms

# 6 Wnioski

1. **TODO**

# 7 Funkcja do testowania metody

**TODO**

# 8 Bibliografia


1. Informacje z wykładu *Metod numerycznych 2* (wydział MiNI PW, dr Iwona Wróbel), w szczególności temat dotyczący *metody potęgowej z normowaniem*, wraz z algorytmem.