

OBLICZENIA NAUKOWE
Lista nr 5 (laboratorium)

Jednostka badawcza dużej firmy działającej w branży chemicznej prowadzi intensywne badania. Wynikiem tych badań są modele pewnych zjawisk chemii kwantowej. Rozwiązanie tych modeli, w pewnym szczególnym przypadku, sprowadza się do rozwiązania układu równań liniowych

$$\mathbf{A}\mathbf{x} = \mathbf{b}.$$

dla danej macierzy współczynników $\mathbf{A} \in \mathbb{R}^{n \times n}$ i wektora prawych stron $\mathbf{b} \in \mathbb{R}^n$. Macierz \mathbf{A} jest rzadką, tj. mającą dużą elementów zerowych, i blokową o następującej strukturze:

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 & \mathbf{C}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{B}_2 & \mathbf{A}_2 & \mathbf{C}_2 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_3 & \mathbf{A}_3 & \mathbf{C}_3 & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{B}_{v-2} & \mathbf{A}_{v-2} & \mathbf{C}_{v-2} & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{B}_{v-1} & \mathbf{A}_{v-1} & \mathbf{C}_{v-1} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B}_v & \mathbf{A}_v \end{pmatrix}, \quad (1)$$

$v = n/\ell$, zakładając, że n jest podzielne przez ℓ , gdzie ℓ jest rozmiarem wszystkich kwadratowych macierzy wewnętrznych (bloków): \mathbf{A}_k , \mathbf{B}_k i \mathbf{C}_k . Mianowicie, $\mathbf{A}_k \in \mathbb{R}^{\ell \times \ell}$, $k = 1, \dots, v$ jest macierzą gęstą, $\mathbf{0}$ jest kwadratową macierzą zerową stopnia ℓ , macierz $\mathbf{B}_k \in \mathbb{R}^{\ell \times \ell}$, $k = 2, \dots, v$ jest następującej postaci:

$$\mathbf{B}_k = \begin{pmatrix} 0 & \cdots & 0 & b_1^k \\ 0 & \cdots & 0 & b_2^k \\ \vdots & & \vdots & \vdots \\ 0 & \cdots & 0 & b_\ell^k \end{pmatrix},$$

\mathbf{B}_k ma tylko jedną, ostatnią, kolumnę niezerową. Natomiast $\mathbf{C}_k \in \mathbb{R}^{\ell \times \ell}$, $k = 1, \dots, v-1$ jest macierzą diagonalną:

$$\mathbf{C}_k = \begin{pmatrix} c_1^k & 0 & 0 & \cdots & 0 \\ 0 & c_2^k & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & c_{\ell-1}^k & 0 \\ 0 & \cdots & 0 & 0 & c_\ell^k \end{pmatrix}.$$

Firma napotkała problem z efektywnym rozwiązaniem problemu $\mathbf{A}\mathbf{x} = \mathbf{b}$, gdzie \mathbf{A} jest postaci (1). Chodzi przede wszystkim o wymagania czasowe i pamięciowe, które wynikają z bardzo dużego rozmiaru macierzy (n jest bardzo duże). W konsekwencji wyklucza to pamiętanie macierzy \mathbf{A} jako tablicę $n \times n$ oraz użycie standardowych (bibliotecznych) algorytmów dla macierzy gęstych (tj. takich, gdzie nie zakłada się dużej liczby elementów zerowych). Zatem jedynym podejściem do rozwiązania tego problemu jest specjalna struktura pamiętająca efektywnie macierz \mathbf{A} , która pamięta tylko elementy niezerowe i adaptacja standardowych algorytmów tak, aby uwzględniały postać macierzy (1), tj. jej rzadkość, regularność występowania elementów zerowych i niezerowych (jeżeli ℓ jest stałą, to czas $O(n^3)$ można zredukować do $O(n)$)

1. Napisać funkcję rozwiązującą układ $\mathbf{A}\mathbf{x} = \mathbf{b}$ metodą eliminacji Gaussa uwzględniającą specyficzną postać (1) macierzy \mathbf{A} dla dwóch wariantów:
 - (a) bez wyboru elementu głównego,
 - (b) z częściowym wyborem elementu głównego.

2. Napisać funkcję wyznaczającą rozkład \mathbf{LU} macierzy \mathbf{A} metodą eliminacji Gaussa uwzględniającą specyficzną postać (1) macierzy \mathbf{A} dla dwóch wariantów:

- (a) bez wyboru elementu głównego,
- (b) z częściowym wyborem elementu głównego.

Rozkład \mathbf{LU} musi być efektywne pamiętany!

3. Napisać funkcję rozwiązującą układ równań $\mathbf{Ax} = \mathbf{b}$ (uwzględniającą specyficzną postać (1) macierzy \mathbf{A}) jeśli wcześniej został już wyznaczony rozkład \mathbf{LU} przez funkcję z punktu 2.

Powyższe funkcje powinny być zaprogramowane w języku Julia umieszczone w module o nazwie `blocksys`. Aby efektywnie pamiętać macierze rzadkie można zapoznać się z rozdziałem o *Sparse Arrays* (zob. manual języka Julia). Możemy założyć wtedy, że dostęp do elementu macierzy (Sparse Arrays) jest w czasie stałym (powszechnie wiadomo, że tak nie jest). **Można również zaproponować własny, efektywny, sposób pamiętania tylko elementów niezerowych - uwzględniający specyfikę problemu, co jest pożądane.** Niewolno używać `x=A\b` i `lu` z modułu `LinearAlgebra`.

Wymagania dotyczące programów:

Programy powinny umożliwiać

- czytanie danej macierzy \mathbf{A} i wektora prawych stron \mathbf{b} z plików tekstowych, których format jest następujący (przykładowe dane umieszczone są na stronie kursu), dla macierzy \mathbf{A} , gdzie zadane są tylko elementy niezerowe:

```
n 1          <--- rozmiar macierzy A, rozmiar macierzy Ak, Bk, Ck
i1 j1 A[i1,j1] <--- indeksy i1,j1 i niezerowy element a[i1,j1] macierzy A
i2 j2 A[i2,j2] <--- indeksy i2,j2 i niezerowy element a[i2,j2] macierzy A
i3 j3 A[i3,j3] <--- indeksy i3,j3 i niezerowy element a[i3,j3] macierzy A
...
...
EOF
```

dla wektora prawych stron \mathbf{b} :

```
n <--- rozmiar wektora b
b[1] <--- składowa nr 1 wektora b
b[2] <--- składowa nr 2 wektora b
...
...
b[n] <--- składowa nr n wektora b
```

- czytanie danej macierzy \mathbf{A} z pliku tekstowego (format opisany wyżej) oraz obliczanie wektora prawych stron \mathbf{b} na podstawie wektora \mathbf{x} (jak na liście nr 2), tj. $\mathbf{b} = \mathbf{Ax}$, gdzie $\mathbf{x} = (1, \dots, 1)^T$. Mnożenie macierzy przez wektor, \mathbf{Ax} powinno uwzględniać postać (1) (macierz jest rzadka),
- drukowanie obliczonego rozwiązania $\hat{\mathbf{x}}$ do pliku tekstowego, jeśli macierz \mathbf{A} i wektor prawych stron \mathbf{b} były czytane z plików,

```
x[1] <--- składowa nr 1 wektora x
x[2] <--- składowa nr 2 wektora x
...
...
x[n] <--- składowa nr n wektora x
```

- drukowanie obliczonego rozwiązania $\hat{\mathbf{x}}$ do pliku tekstowego, jeśli macierz \mathbf{A} była czytana z pliku a wektor prawych stron \mathbf{b} był obliczany

```

blad <--- blad wzgledny
x[1] <--- skladowa nr 1 wektora x
x[2] <--- skladowa nr 2 wektora x
...
...
x[n] <--- skladowa nr n wektora x

```

- testowanie rozwiązywania $Ax = b$ (funkcja z punktu 1 w dwóch wariantach wyboru elementu głównego),
- testowanie rozwiązywania $Ax = b$ dwuetapowo (funkcja z punktu 2 w dwóch wariantach wyboru elementu głównego i funkcja z punktu 3), tj. $A = LU$ (funkcja z punktu 2 w dwóch wariantach wyboru elementu głównego), następnie $LUx = b$ (funkcja z punktu 3).

Dane testowe

Przykład danych testowych znajduje się na stronie kursu (macierze są dobrze uwarunkowane). Macierze A o postaci (1) zapisywanej do pliku tekstowego, zgodnego z formatem opisanym wyżej, można generować za pomocą funkcji `blockmat` z modułu `matrixgen.jl` umieszczonego na stronie kursu.

Wyniki eksperymentów

Wyniki przedstawić w sprawozdaniu, plik pdf, które powinno zawierać:

1. opis algorytmów - opis implementacji z analizą złożoności,
2. wyniki eksperymentów porównujących zaimplementowane algorytmy dla danych testowych (tabele, wykresy) oraz ich interpretację.
3. wnioski.

Do sprawozdania należy dołączyć pliki z kodem (*.jl). Pliki powinny być skomentowane: imię i nazwisko autora (**anonimowe źródła nie będą sprawdzane**), komentarze zmiennych. Spakowane pliki wraz ze sprawozdaniem (*.zip) należy przesłać e-mailem prowadzącemu.

Uwaga: Ostateczną wersję proszę programów przekompilować i przetestować pod linuxem.

Na ocenę co najwyżej dostateczny plus nie trzeba implementować punktu 2 i punktu 3 oraz testowania dwuetapowego. Ponadto, jeśli zostały zaimplementowane wszystkie punkty, ale implementacja jest oparta na Sparse Arrays wtedy można otrzymać ocenę co najwyżej dobry plus.

Literatura

- [1] A. Kiełbasiński, H. Schwetlick, *Numeryczna algebra liniowa. Wprowadzenie do obliczeń zautomatyzowanych*, wyd. 2, WNT, Warszawa 1994.