

# SPRAWOZDANIE - LABORATORIUM 3

## Iteracyjne rozwiązywanie układu równań liniowych metodą Jacobiego

Marta Dychała

25 marca 2021

### 1 Wstęp teoretyczny

Metoda Jacobiego jest jedną z metod iteracyjnych rozwiązywania układów równań liniowych. Oznacza to, że w tej metodzie dany układ równań rozwiązuje się wielokrotnie w celu znalezienia jak najlepszego przybliżenia rozwiązania. Każde kolejne rozwiązanie powstaje na podstawie poprzedniego, stając się w ten sposób coraz bardziej dopasowanym do rozwiązania analitycznego. Idąc tym tokiem rozumowania, można dojść do wniosku, że im więcej stosuje się iteracji, tym rozwiązanie powinno być bliższe teorii. Omawiana własność metody Jacobiego czyni ją bardzo przydatną przy rozwiązywaniu układów równań, gdzie macierz współczynników jest macierzą rzadką<sup>1</sup>.

Przykładowo dany jest poniższy układ równań liniowych o  $N$  niewiadomych:

$$\underbrace{\begin{bmatrix} a_{11} & 0 & 0 & 0 & 0 & \cdots & 0 \\ a_{21} & a_{22} & 0 & 0 & 0 & \cdots & 0 \\ a_{31} & a_{32} & a_{33} & 0 & 0 & \cdots & 0 \\ 0 & a_{42} & a_{43} & a_{44} & 0 & \cdots & 0 \\ 0 & 0 & a_{53} & a_{54} & a_{55} & \cdots & 0 \end{bmatrix}}_{\mathbf{A}} \cdot \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_N \end{bmatrix}}_{\mathbf{b}} \quad (1)$$

gdzie  $\mathbf{A}$  to macierz współczynników,  $x$  to macierz niewiadomych, a  $b$  to macierz wyrazów wolnych. W tym przypadku macierz  $\mathbf{A}$  jest macierzą rzadką i jednocześnie trójdziagonalną<sup>2</sup>. Rozwiązywanie układu równań (1) za pomocą metod bezpośrednich byłoby bardzo kosztowne pod względem pamięci - wszak należałoby alokować pamięć dla całej macierzy nawet jeśli ta ma ogromne wymiary i jest macierzą rzadką. W metodach iteracyjnych, takich jak metoda Jacobiego wystarczy alokować pamięć jedynie na niezerowe diagonale w postaci wektorów  $d_k$ , gdzie  $k$  to numer kolejnej diagonal. Na przykład w celu rozwiązania układu równań (1) wystarczy alokować pamięć dla trzech wektorów, które odpowiadają określonym przekątnym macierzy  $\mathbf{A}$ :

$$\begin{aligned} d_0 &= [a_{11}, a_{22}, a_{33}, a_{44}, a_{55}], \\ d_1 &= [0, a_{21}, a_{32}, a_{43}, a_{54}], \\ d_2 &= [0, 0, a_{31}, a_{42}, a_{53}] \end{aligned}$$

Na podstawie rozmieszczenia elementów  $a_{ij}$  na diagonalach macierzy  $n$ -przekątniowej można opracować przepis na wektory  $d_k$  odpowiadającym diagonalom. Należy przy tym pamiętać, że każdy z wektorów  $d_k$  musi mieć długość  $N$  dla układu równań z  $N$  niewiadomymi; dlatego też jeżeli:

- $i - j > 0$ , to na początku wektora  $d_k$  dodajemy  $i - j$  zer,
- $i - j = 0$ , to do wektora  $d_k$  przepisujemy przekątną (mamy do czynienia z przekątną główną, która już jest rozmiaru  $N$ ),
- $i - j < 0$ , dopisujemy  $j - i$  zer na końcu wektora  $d_k$ .

<sup>1</sup>Macierz rzadka - macierz, której większość elementów ma wartość zero

<sup>2</sup>Macierz  $n$ -diagonalna ( $n$ -przekątniowa) - macierz, która posiada niezerowe elementy jedynie na  $n$  diagonalach

Aby w metodzie Jacobiego odnaleźć rozwiązanie układu równań z macierzą n-diagonalną, potrzebujemy znać wartości  $n - 1$  początkowych jego niewiadomych, chociażby poprzez warunki początkowe. Na tej podstawie tworzymy dwie macierze niewiadomych: pierwsza z nich to macierz poprzedniego przybliżenia  $x_s$ ; druga natomiast to macierz bieżącego przybliżenia  $x_n$ , której elementy w każdej iteracji przyjmują wartości kolejnych elementów wektora  $x_s$ . Znając  $n - 1$  wartości początkowych wektora  $x_s$  oraz przyjmując, że nie występują żadne niezerowe diagonale powyżej diagonali głównej, można znaleźć pozostałe wartości wektora niewiadomych korzystając z algorytmu na wartość i-tego elementu wektora niewiadomych  $x_n$ :

$$x_n[i] = \frac{1}{d_0} (b[i] - \sum_{k=1}^n d_k x_s[i - k]), \quad (2)$$

gdzie  $b[i]$  to i-ty element wektora wyrazów wolnych.

## 2 Zadanie do wykonania

### 2.1 Opis problemu

Celem laboratorium było znalezienie za metody Jacobiego rozwiązania równania różniczkowego:

$$\frac{d^2 x}{dt^2} = -\omega^2 x - \beta v + F_0 \sin(\Omega t) \quad (3)$$

Opisuje ono ruch ciała poddany działaniu sił: sprężystej ( $-\omega^2 x$ ), tarcia zależnej od prędkości ( $-\beta v$ ) oraz wymuszającej ruch ( $F_0 \sin(\Omega t)$ ). Ponieważ równanie (3) zmienia się w czasie, można więc potraktować czas jako iloczyn pojedynczego kroku czasowego  $h$ :

$$t = i \cdot h, \quad i = 0, 1, 2, \dots$$

Drugą pochodną można przedstawić jako symetryczny trójpunktowy iloraz różnicowy. W ten sposób otrzymuje się równanie:

$$\frac{d^2 x}{dt^2} = \frac{x_{i-1} - 2x_i + x_{i+1}}{h^2} = -\omega^2 x - \beta v + F_0 \sin(\Omega t)$$

Ponieważ prędkość jest pierwszą pochodną położenia po czasie, również może być zamieniona na iloraz różnicowy:

$$v_i = \frac{x_{i+1} - x_i}{h}$$

W ten sposób przenosząc wyrazy z niewiadomymi  $x_i$  na lewą stronę, a na prawej pozostawiając wyraz wolny otrzymujemy:

$$\begin{aligned} x_{i-1} - 2x_i + x_{i+1} \omega^2 h^2 x_i + \beta h(x_{i+1} - x_i) &= F_0 \sin(\Omega h i) h^2 \\ x_{i-1} - (\omega^2 h^2 - 2 - \beta h) x_i + (1 + \beta h) x_{i+1} &= F_0 \sin(\Omega h i) h^2 \end{aligned}$$

Co można zapisać symbolicznie jako:

$$a_1 x_{i-1} + a_2 x_i + a_3 x_{i+1} = b_i, \quad (4)$$

gdzie  $a_1 = 1$ ,  $a_2 = \omega^2 h^2 - 2 - \beta h$ ,  $a_3 = 1 + \beta h$ ,  $b_i = F_0 \sin(\Omega h i) h^2$ . W ten sposób otrzymuje się iteracyjny przepis na rozwiązanie układu równań liniowych, który jest prawdziwy dla  $i = 2, 3, 4, \dots$ . Wartości niewiadomych  $x_0$  oraz  $x_1$  uzyskujemy poprzez warunki początkowe na wychylenie ( $x_0 = 1$ ) oraz na prędkość początkową:  $v_0 = \frac{x_{i+1} - x_i}{h} = 0$ . Na podstawie warunków początkowych oraz równania (4), otrzymujemy układ równań liniowy postaci:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ a_1 & a_2 & a_3 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & a_1 & a_2 & a_3 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & a_1 & a_2 & a_3 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & a_1 & a_2 & a_3 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & a_1 & a_2 & a_3 & \cdots & 0 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ \cdots \\ x_n \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ b_1 \\ b_2 \\ b_3 \\ \cdots \\ b_n \end{bmatrix} \quad (5)$$

W trakcie zajęć laboratoryjnych należało rozwiązać układ równań (5) za pomocą metody Jacobiego. Ponieważ macierz układu równań (5) jest macierzą trójdziagonalną, nie jest wskazane przechowywanie jej całej w pamięci. Dlatego też lepszym rozwiązaniem pod względem optymalizacji zużycia pamięci jest przechowanie niezerowych przekątnych w postaci trzech tablic jednowymiarowych o rozmiarze  $N + 1$  (w naszym przypadku  $N = 2000$ ):

$$\begin{aligned}d_0 &= [1, 1, a_3, a_3, \dots, a_3] \\d_1 &= [0, -1, a_2, a_2, \dots, a_2] \\d_2 &= [0, 0, a_1, a_1, \dots, a_1]\end{aligned}$$

$i$ -ty element nowego przybliżenia ( $x_n[i]$ ) ma wtedy postać:

$$x_n[i] = \frac{1}{d_0[i]}(b[i] - d_1[i]x_s[i-1] - d_2[i]x_s[i-2]), \quad i = 2, 3, 4, \dots \quad (6)$$

gdzie  $x_s[i]$  to  $i$ -ty element wektora poprzedniego przybliżenia. Dla pojedynczej iteracji należy użyć wzoru (6) na każdym z elementów wektora  $x_n$  nieokreślonym przez warunki początkowe. Na końcu każdej iteracji zapisuje się do wektora  $x_s$  tablicę  $x_n$ , tak aby wektor  $x_s$  w kolejnej iteracji był poprzednim przybliżeniem. Zawartość tablicy  $x_n$  do  $x_s$  przekopiuje się tyle razy, dopóki dopóki poniższy warunek nie będzie spełniony:

$$\left| \sum_{i=0}^n (x_n[i])^2 - \sum_{i=0}^n (x_s[i])^2 \right| = 0. \quad (7)$$

Jednak ze względu na pojawiające się w programie błędy numeryczne, należy zmodyfikować warunek opisany w równaniu (7) przyjmując pewną tolerancję przybliżenia (w naszym przypadku będzie to  $10^{-6}$ ):

$$\left| \sum_{i=0}^n (x_n[i])^2 - \sum_{i=0}^n (x_s[i])^2 \right| < 10^{-6}. \quad (8)$$

Kiedy warunek opisany nierównością (8) zostanie spełniony, zakończy się iterowanie po elementach macierzy, a wektor  $x_n$  stanie się rozwiązaniem układu równań (5).

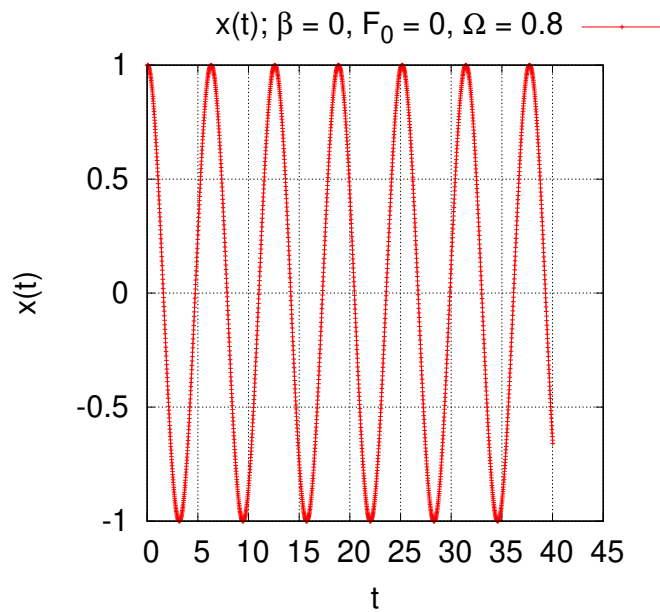
## 2.2 Wyniki

Ustalono poniższe warunki początkowe:

- prędkość początkowa  $v_0 = 0$ ,
- położenie początkowe  $x_0 = 1$ ,
- częstość  $\omega = 1$ ,
- krok czasowy  $h = 0,02$ .
- liczba kroków czasowych  $N = 2000$

Równanie (3) rozwiązano dla trzech przypadków. Na podstawie obliczonego rozwiązania dla każdego z nich sporządzono wykres zależności położenia  $x(t)$  od czasu. Za każdym razem obliczanie przybliżenia zakończyło się po 2000 iteracjach:

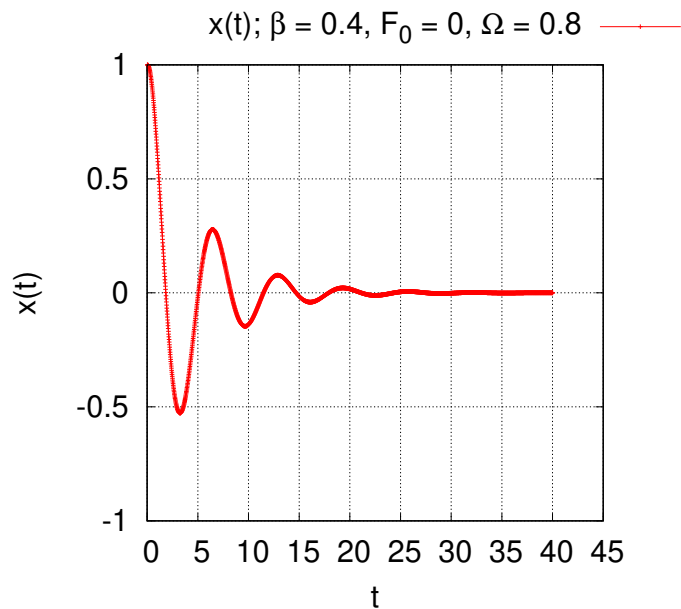
(a)  $F_0 = 0.0$ ,  $\beta = 0.0$ ,  $\Omega = 0.8$



Rysunek 1: Wykres zależności wychylenia od czasu dla przypadku a).

Na wykresie z Rysunku 1. widać, że rozwiązanie ma przebieg sinusoidalny. Jest to przypadek drgań nietłumionych przez żadne siły oporu.

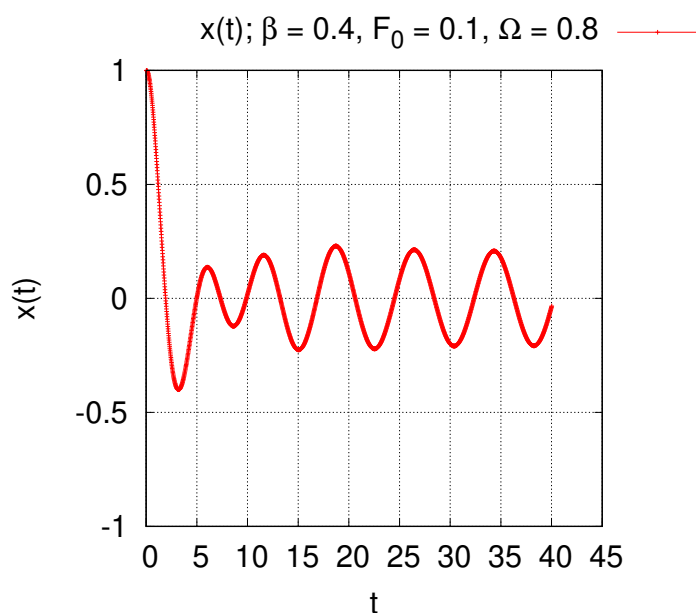
(b)  $F_0 = 0.0$ ,  $\beta = 0.4$ ,  $\Omega = 0.8$



Rysunek 2: Wykres zależności wychylenia  $x(t)$  od czasu dla przypadku b).

Przypadek b) opisuje równanie ruchu, kiedy na ciało działa niezerowa siła tarcia ( $\beta \neq 0$ ). Na wykresie powyżej łatwo zauważyć tłumienie w jego końcowej części.

(c)  $F_0 = 0.1$ ,  $\beta = 0.4$ ,  $\Omega = 0.8$



Rysunek 3: Wykres zależności wychylenia  $x(t)$  od czasu dla przypadku c).

Przypadek c) opisuje równanie ruchu ciała poddanego zarówno sile tarcia jak i sile wymuszającej ruch ( $F_0 \neq 0$ ). Można zauważyć, że po czasie około 15 s tłumienie ustępuje.

### 3 Wnioski

Metoda Jacobiego jest bardzo wydajną metodą przy rozwiązywaniu układów równań z macierzami rzadkimi. Dla metod bezpośrednich rozwiązywanie macierzy o "dużych" wymiarach (np.  $2000 \times 2000$ ) jest bardzo czasochłonne ze względu na mnogość operacji wiążących się z wymiarami macierzy. W metodzie Jacobiego czas rozwiązywania danego układu równań zależy od wymiaru macierzy współczynników. Z drugiej strony jeśli macierz współczynników układu równań nie jest macierzą rzadką, a jej elementy nie są związane ze sobą pewną zależnością, to metoda Jacobiego nie będzie zbyt praktyczna, bowiem w programie o wiele łatwiej jest zarządzać jedną macierzą niż wieloma wektorami.