Metody numeryczne

Laboratorium 3

Iteracyjne rozwiązywanie układu równań liniowych metodą Jacobiego.

26.03.2020r.

Adrian Furman

1. Wstęp teoretyczny

Metody iteracyjne rozwiązywania układów liniowych w ogólności polegają na wyliczaniu kolejnych przybliżeń szukanego rozwiązania. W czasie obliczeń tworzy się ciąg kolejnych przybliżeń: $\vec{x}^{(0)}, \vec{x}^{(1)}, \vec{x}^{(2)}, \ldots, \vec{x}^{(k)}, \ldots$, gdzie za $\vec{x}^{(0)}$ przyjmujemy dowolny wektor (wartości nie mają znaczenia). Jeżeli istnieje granica tego ciągu równa $\vec{x}$:

$$\lim_{k \to \infty} \vec{x}^{(k)} = \vec{x},$$

to zarówno ciąg jak i metoda są zbieżne a wektor $\vec{x}$ jest rozwiązaniem układu równań.

Do metod iteracyjnych należy metoda Jacobiego, która opiera się na takim przekształceniu początkowo wybranego przybliżenia $\vec{x}^{(0)}$ w kolejnych iteracjach aby po k przebiegach pętli wyeliminować składowe wektora reszt:

$$(\vec{b} - A\vec{x}^{(k)})_i = 0.$$

Po odpowiednich przekształceniach oraz przyjęciu reprezentacji macierzy A jako sumy: $A = L + D + U$, gdzie L to macierz zawierająca elementy spod przekątnej macierzy A , D to macierz diagonalna z elementami przekątnej macierzy A a U to pozostała część ponad diagonalą – otrzymujemy ogólny wzór pojedynczej iteracji:

$$\vec{x}^{(k+1)} = -D^{-1}(L + U)\vec{x}^{(k)} + D^{-1}\vec{b}. \quad (1)$$

2. Zadanie do wykonania

2.1. Opis problemu

Badanym problemem był ruch ciała poddanego działaniu siły sprężystej $F_s = -\omega^2 x$ oraz siły wymuszającej $F_w = F_0 \sin(\Omega t)¿,$ z uwzględnieniem siły tarcia wprost proporcjonalnej do prędkości: $F_t = -\beta V$. Ruch ten opisuje następujące równanie różniczkowe drugiego rzędu:

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

Równanie opisuje wychylenie ciała w kolejnych chwilach czasowych:

$$t_i = h \cdot i, \quad i = 0,1,2,\ldots$$

Po wyrażeniu drugiej pochodnej oraz prędkości za pomocą ilorazów różnicowych, wstawieniu ich do równania (2) oraz po odpowiednim uporządkowaniu dochodzimy do następującej postaci równania:

$$a_1 x_{i-1} + a_2 x_i + a_3 x_{i+1} = b_i, \quad (3)$$

gdzie:

$a_1 = 1,$

$a_2 = \omega^2 h^2 - 2 - \beta h,$

$a_3 = 1 + \beta h,$

$b_i = F_0 \sin(\Omega h i) h^2.$

Aplikując do naszego równania (3) warunki początkowe: $x(t = 0) = 1$, $V(t = 0) = 0$ otrzymujemy macierz trójprzekątniową, którą możemy wyrazić za pomocą trzech wektorów:

$$\vec{d_0} = [1,\ 1,\ a_3,\ a_3,\ \ldots,\ a_3],$$

$$\vec{d_1} = [0,\ -1,\ a_2,\ a_2,\ \ldots,\ a_2],$$

$$\vec{d_2} = [0,\ 0,\ a_1,\ a_1,\ \ldots,\ a_1],$$

gdzie wektor d_0 reprezentuje przekątną macierzy, d_1 elementu tuż pod przekątną a d_2 kolejny „ukośny” rząd.

Zgodnie z metodą Jacobiego dla naszego przypadku formuła (1) dla pojedynczego elementu i-tego nowego przybliżenia w oparciu o przybliżenie z poprzedniej iteracji wygląda następująco:

$$x_n[i] = \frac{1}{d_0[i]}\left(b[i] - d_1[i]x_s[i-1] - d_2[i]x_s[i-2]\right),$$

gdzie x_n i x_s to odpowiednio nowe i stare przybliżenie.

Postawiony problem rozwiązany został dla trzech zestawów danych wejściowych:

1) $\beta = 0.0,\ F_0 = 0.0,\ \Omega = 0.8,$
2) $\beta = 0.4,\ F_0 = 0.0,\ \Omega = 0.8,$
3) $\beta = 0.4,\ F_0 = 0.1,\ \Omega = 0.8,$

za parametry przyjmując: $V_0 = 0,\ x_0 = 1,\ \omega = 1,$ liczbę kroków czasowych $n = 2000$ oraz $h = 0.01.$

2.2. Wyniki

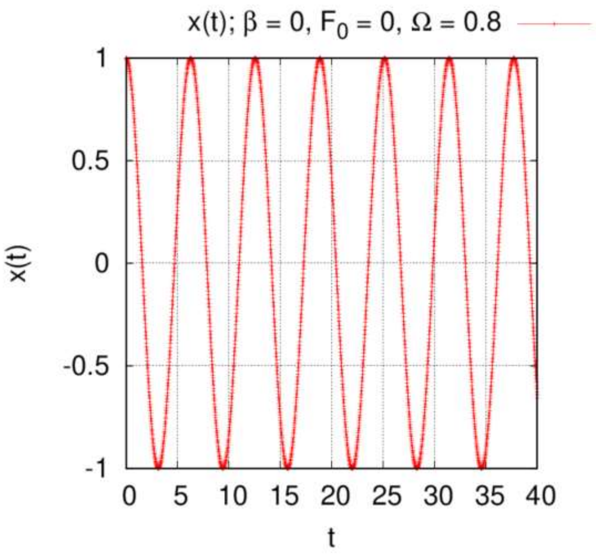
Na kolejnej stronie zamieszczone zostały wykresy wygenerowane za pomocą programu gnuplot na podstawie danych otrzymanych w każdym z trzech przypadków danych wejściowych. Za warunek zbieżności przyjęta została dokładność wartości bezwzględnej z różnicy sum kwadratów przybliżenia obecnego oraz tego z poprzedniej iteracji:

$$s_n = \sum_i \left(x_n[i]\right)^2, \quad s_s = \sum_i \left(x_s[i]\right)^2,$$

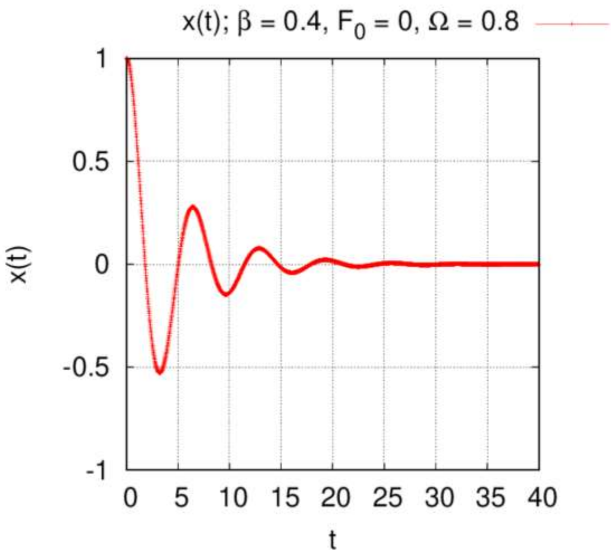
$$|s_n - s_s| < 10^{-6}.$$

Oznacza to, że zbieżność wnioskujemy na podstawie braku znaczących zmian w kolejnych przybliżeniach. Zbieżność dla każdego z przypadków uzyskano po 2002 iteracjach.

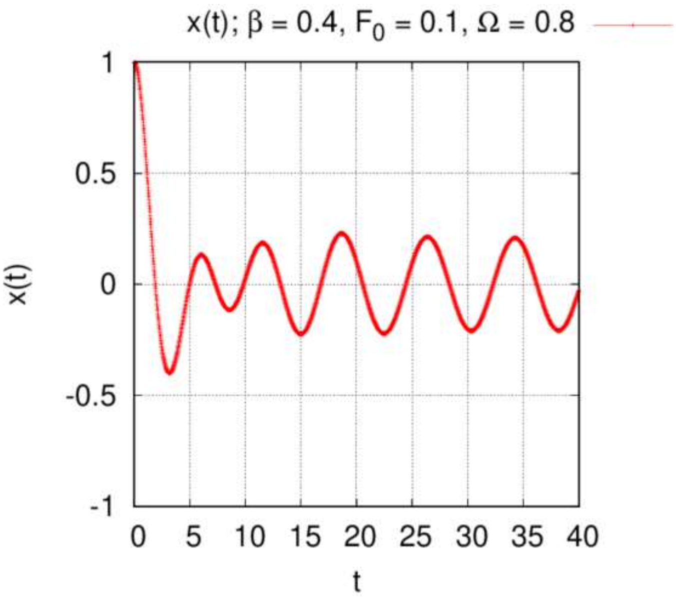
Wykresy prezentują się następująco:



Wykres 1: β = 0 (brak tłumienia), F_0 = 0 (brak wymuszenia), Ω = 0.8.



Wykres 2: β = 0.4 (tłumienie), F_0 = 0 (brak wymuszenia), Ω = 0.8.



Wykres 3: β = 0.4 (tłumienie), F_0 = 0.1 (wymuszenie), Ω = 0.8.

Otrzymane kształty pokrywają się z wzorcowymi wykresami. Na wykresie 1 widzimy typowy dla oscylatora harmonicznego kształt funkcji, ze względu na brak tłumienia oraz wymuszenia amplituda się nie zmienia. Wykres 2 przedstawia sytuację z uwzględnieniem tłumienia co przekłada się na stopniowe zmniejszanie się amplitudy drgań (zgodnie z kształtem funkcji eksponencjalnej) wraz z czasem aż to praktycznego ich zaniku. W trzecim przypadku obserwujemy początkowe słabnięcie drgań a następnie ich stabilizację spowodowaną obecnością sinusoidalnej siły wymuszającej.

3. Wnioski

Metody iteracyjne rozwiązywania układów równań są świetnym narzędziem do rozwiązywania problemów reprezentowanych przez macierze rzadkie, szczególności k-przekątniowe, czyli takie gdzie dane nie będące zerami znajdują się tylko na kilku przekątnych – są to częste przypadki podczas obliczeń naukowych i inżynierskich. Ich fenomen polega na oszczędzanej ilości pamięci i czasu – nie musimy przechowywać w pamięci całej macierzy co w wielu przypadkach wymagało by ogromnej jej ilości – często wystarczy kilka wektorów, jeśli chodzi o czas nie marnujemy go na niepotrzebne operacje na znajdujących się w macierzach rzadkich zerach.

Należąca do rodziny metody iteracyjnych metoda Jacobiego okazała się skutecznym narzędziem dając spodziewane rezultaty po 2002 iteracjach, jednakże okazuje się, że nie jest to metoda najbardziej optymalna.

Wystarczy porównać ją z metodą Gaussa-Seidla, która różni się od metody Jacobiego jedynie tym, że kolejne elementy nowego przybliżenia obliczamy na podstawie poprzednich elementów z tego wektora tym samym oszczędzając pamięć na braku drugiego wektora przechowującego poprzednie przybliżenie (pierwszą sumę kwadratów możemy obliczyć na początku pętli, a drugą na jej końcu po wykonaniu obliczeń) oraz co najważniejsze szybciej uzyskując zbieżność. W naszym przypadku warunek zbieżności spełniony został już po 2 iteracjach co jest oczywiście znacznie lepszym rezultatem.