Warszawa, 26.03-17.04.2020

PROJEKT WNUM

ZADANIE 1 #25

Politechnika Warszawska

Wydział Elektroniki i Technik Informacyjnych

Przedmiot: WNUM

Prowadzący projekt: dr inż. Andrzej Miękina

Wykonawca: Maciej Kaczkowski

1. WSTĘP TEORETYCZNY, CELE DOŚWIADCZEŃ

Celem doświadczeń była analiza dokładności obliczeń komputerowych. Wykorzystano do tego szacowanie błędów za pomocą liczby epsilon. Zbadano również propagacje błędów za pomocą rachunku epsilonów oraz liniowego modelu propagacji błędów. Rachunki sprawdzono za pomocą programu Matlab. Posłużyło to do oszacowania całkowitego błędu jaki może wprowadzić badana funkcja

$$y=\cos(x^2+2)\exp(x^3+2)\, dla\, x\in[0,1]$$

Następnie ten błąd oszacowano również metodą symulacyjną, czyli sprawdzając wszystkie możliwe kombinacje w jakich mogą wystąpić błędy w wyrażeniu

$$\tilde{y}=\cos(((x(1+\epsilon))^2(1+\eta_p)+2)(1+\eta_s))(1+\eta_{\cos})\exp(((x(1+\epsilon))^2(1+\eta_p)x(1+\epsilon)(1+\eta_m)+2)(1+\eta_s))(1+\eta_{\exp})(1+\eta_m)$$

Ostatnim doświadczeniem było wykorzystanie metody symulacji statystycznej do oszacowania niepewności rozwiązania układu równań typu

$$\boldsymbol{A}\cdot\boldsymbol{x}=\boldsymbol{b}$$

Gdzie ***A*** jest macierzą 5x5

2. PUNKT 1

$$y=\cos(x^2+2)\exp(x^3+2)\, dla\, x\in[0,1]$$

Obliczono współczynniki przenoszenia względnych błędów zmiennopozycyjnej reprezentacji liczb oraz zaokrągleń operacji zmiennopozycyjnych korzystając z rachunku epsilonów.

$$\tilde{y}=\cos(((x(1+\epsilon))^2(1+\eta_p)+2)(1+\eta_s))(1+\eta_{\cos})\exp(((x(1+\epsilon))^2(1+\eta_p)x(1+\epsilon)(1+\eta_m)+2)(1+\eta_s))(1+\eta_{\exp})(1+\eta_m)$$

$$T_x=\frac{x}{y}\frac{dy}{dx}$$

Dla czytelności wykonano podstawienie:

$$u=\cos(x^2+2)$$

$$v=\exp(x^3+2)$$

$$y=u\,v$$

Zatem

$$\tilde{y}=\tilde{u}\tilde{v}(1+\eta_m)$$

$$\tilde{u}=\cos(((x(1+\epsilon))^2(1+\eta_p)+2)(1+\eta_s))(1+\eta_{\cos})$$

$$\tilde{u}\approx\cos(x^2(2\epsilon+\eta_p+eta_s)+2\eta_s+x^2+2)(1+\eta_{\cos})$$

$$\tilde{u}=\cos((x^2+2)(1+\frac{(x^2(2\epsilon+\eta_p+\eta_s)+2\eta_s)}{x^2}+2))(1+\eta_{\cos})$$

$$\tilde{u}=\cos(x^2+2)(1+T_{\cos}\frac{(x^2(2\epsilon+\eta_p+\eta_s)+2\eta_s)}{x^2}+2)(1+\eta_{\cos})$$

$$T_{\cos}=\frac{z}{\cos}(z)(-\sin(z))=-z\tan(z)=-(x^2+2)\tan(x^2+2)$$

Zatem

$$\tilde{u}=\cos(x^2+2)(1+\eta_{\cos}-\tan(x^2+2)(x^2(2\epsilon+\eta_p+\eta_s)+2\eta_s))$$

$$\tilde{v}=\exp(((x(1+\epsilon))^2(1+\eta_p)x(1+\epsilon)(1+\eta_m)+2)(1+\eta_s))(1+\eta_{\exp})$$

$$\tilde{v}=\exp((x^3(1+\eta_p+3\epsilon+\eta_m)+2)(1+\eta_s))(1+\eta_{\exp})$$

$$\tilde{v}=\exp(x^3(1+\eta_p+3\epsilon+\eta_m+\eta_s)+2\eta_s+x^3+2)(1+\eta_{\exp})$$

$$\tilde{v}=\exp((x^3+2)(1+\frac{(x^3(1+\eta_p+3\epsilon+\eta_m+\eta_s)+2\eta_s)}{(x^3+2)}))(1+\eta_{\exp})$$

$$\tilde{v}=\exp(x^3+2)(1+T_{\exp}\frac{(x^3(1+\eta_p+3\epsilon+\eta_m+\eta_s)+2\eta_s)}{(x^3+2)})(1+\eta_{\exp})$$

$$T_{\cos}=(\frac{z}{e^z})e^z=z=x^3+2$$

Zatem

$$\tilde{v}=\exp(x^3+2)(1+\eta_{\exp}+x^3(1+\eta_p+3\epsilon+\eta_m+\eta_s)+2\eta_s)$$

Wracając z podstawieniem otrzymano

$$\tilde{y}=\cos(x^2+2)\exp(x^3+2)(1+\eta_{\exp}+\eta_{\cos}+x^3(1+\eta_p+3\epsilon+\eta_m+\eta_s)+2\eta_s-\tan(x^2+2)(x^2(2\epsilon+\eta_p+\eta_s)+2\eta_s))$$

Z powyższego wyniku można odczytać wartości poszczególnych współczynników przenoszenia błędów zgodnie ze schematem

$$\breve{y}=y(1+f_1(x)\epsilon+f_2(x)\eta_s+\ldots)$$

$$T_x=f_1(x)$$

$$K_s=f_2(x)$$

…

Zatem

$$K_{\cos}=K_{\exp}=1$$

$$K_m=1+x^3$$

$$K_s=x^3+2-\tan(x^2+2)(x^2+2)$$

$$K_p=x^3-x^2\tan(x^2+2)$$

$$T_x=3x^3-2x^2\tan(x^2+2)$$

Wyniki otrzymane metodą analityczną (różniczkowanie) są zgodne z wynikami otrzymanymi metodą rachunku epsilonów. Dowodem na to są poniższe wykresy

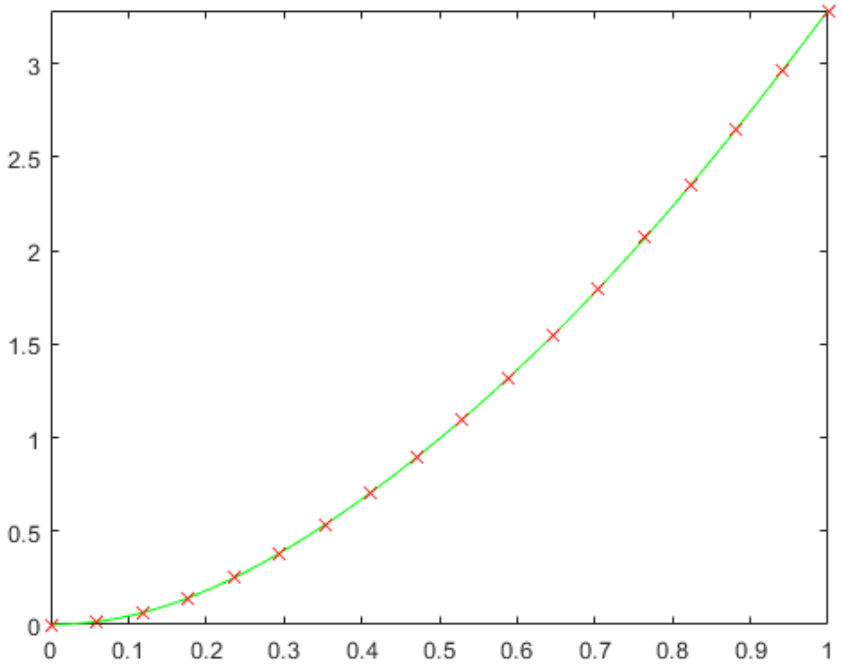

Fig. 1

$T_x(x)$

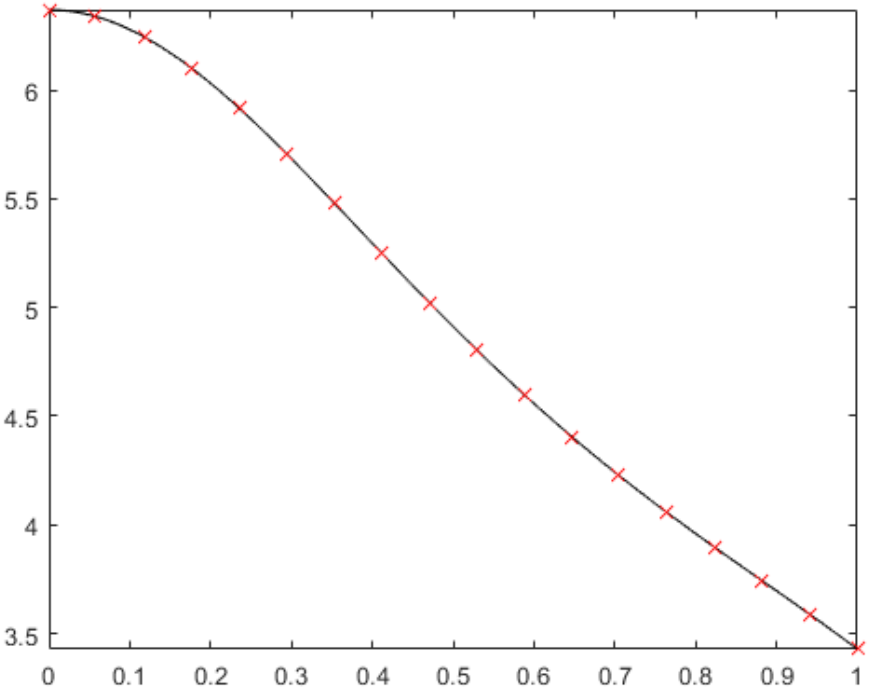

Fig. 2

$K_s(x)$

3. PUNKT 2

Ze względu na to, że poszczególne współczynniki osiągają maksimum w różnych punktach konieczne było oszacowanie błędu całkowitego za pomocą sumowania całych wektorów błędów pochodzących od współczynników w przedziale [0,1]. Działania wykonano za pomocą Matlaba przyjmując

$$\epsilon = 5 \cdot 10^{-12}$$

Uzyskano

$$\delta y_{supremum} = supremum\left(\left|\left(T_x(x)\right)\right| + \left|\left(K_p(x)\right)\right| + \left|\left(K_m(x)\right)\right| + \left|\left(K_s(x)\right)\right| + \left|\left(K_{\exp}(x)\right)\right| + \left|\left(K_{\cos}(x)\right)\right|\right) \approx 5{,}9276 \cdot 10^{-11}$$

Jest to sensowna i zadowalająca dokładność, gorsza o 1 rząd wielkości od przyjętego epsilona.

4. PUNKT 3

Używając metody symulacyjnej, czyli sprawdzenia za pomocą Matlaba wszystkich kombinacji błędów jakie mogą wystąpić uzyskano wynik

$\delta y_{supremum} = 5{,}3989 \cdot 10^{-11}$

Ze względu na założenie, że błędy przyjmują tylko wartości -eps lub +eps i na fakt, że we wzorze z punktu 1 jest 11 możliwości wystąpienia błędu użyto macierzy o 11 kolumnach i 2^{11} wierszach. Została ona wygenerowana za pomocą funkcji de2bi z toolboxa Matlab Comunications i posłużyła, po małych zmianach, jako macierz zawierająca wszystkie możliwe kombinacje występujących błędów. W każdej iteracji pętli parfor błąd wyliczenia wartości y był zapisywany do wektora, z którego następnie wybrano największą wartość.

Rozbieżność pomiędzy wynikami z punktu 2 i 3 jest rzędu przyjętego epsilona. Można ją wyjaśnić niedokładnościami numerycznymi, powstałymi np. podczas obliczania wartości funkcji tangens lub odejmowania.

5. PUNKT 4

Cel, czyli oszacowanie niepewności rozwiązania układu równań algebraicznych, osiągnięto w następujących krokach:

1. Obliczenie dokładnej wartości wektora b
2. Dodanie losowego zaburzenia na poziomie [-eps, eps] do macierzy A
3. Obliczenie zaburzonego wektora x za pomocą lewostronnego dzielenia A\b
4. Oszacowanie niepewności za pomocą wyrażenia

$$\Delta = \frac{(\|(\tilde{x} - \hat{x})\|)_2}{|\hat{x}|}$$

i zapisanie jej do wektora

5. Powtórzenie kroków od 2 do 5 milion razy w pętli parfor
6. Wybranie największego elementu wektora niepewności rozwiązań, który osiągał w zależności od wywołania wartości rzędu

10^{-10} *na przykład* $1{,}9624 \cdot 10^{-10}$

Zastosowana w punkcie 3 metoda polegająca na stworzeniu macierzy wszystkich możliwych kombinacji błędów w tym przypadku okazała się niemożliwa do przeprowadzenia ze względu na ograniczenie pamięci operacyjnej laptopa. Metoda ta wymagała alokacji pamięci na macierz zawierającą 25 kolumn i 2^{25} wierszy.

6. UWAGI, OBSERWACJE, WNIOSKI

Przyjęta w doświadczeniach wartość epsilona była identyczna jak faktyczna wartość, przyjmowana automatycznie przez Matlaba.

Przyjęta w punkcie 4 metoda rozwiązywania układu równań – bezpośrednie dzielenie macierzy w praktyce jest rzadko stosowana. Wynika to z tego, że dzielenie macierzy jest operacją bardzo źle uwarunkowaną numerycznie. W przypadku układu 5 równań, a zatem macierzy 5x5 nie ma to decydującego wpływu na dokładność rozwiązania, jednak być może, przy użyciu lepszej metody jest możliwość uzyskania mniejszej niepewności.

7. UŻYTY KOD MATLABA

```
clc
clear all

%deklaracja zmiennych i niektórych funkcji
syms x v w;
eps = 5e-12;
y = cos(x.^2+2).*exp(x.^3+2);
p = linspace(0,1,1000);

%----------------------------------------
%------------PODPUNKT 1-----------------
%----------------------------------------

%wspolczynnik transmisji obliczony
analitycznie
Tx = x/y*diff(y,x);

%wspolczynnik transmisji obliczony za pomoca
rachunku epsilonow "na kartce"
Tx2 = (3*x.^3)-((2*x.^2)*tan(x.^2+2));

figure(1);
title('T(x)');
fplot (Tx, [0,1], '-g');
hold on;
fplot (Tx2, [0,1], 'x r');
hold off;

fTx = @(x) (3*x.^3)-((2*x.^2).*tan(x.^2+2));
%maks w 1

%pozostale wspolczynniki obliczone za pomoca
rachunku epsilonow "na kartce"
Kcos = 1;
Kexp = 1;

Km2 = @(x) 1+x.^3;
%maks w 1

Ks2 = @(x) x.^3+2-(x.^2+2).*tan(x.^2+2);
%maks w 0
Kp2 = @(x) x.^3 - (x.^2).*tan(x.^2+2);
%maks w 1


%pozostale wspolczynniki obliczone
analitycznie

%sumowanie
y = cos(x.^2+2).*exp(x.^3+2);
ys = subs(y, x^2+2, v);
Ksprim = v/ys*diff(ys,v);
Ksprim = subs(Ksprim, v, x^2+2);
ys = subs(y, x^3+2, w);
Ksbis = w/ys*diff(ys,w);
Ksbis = subs(Ksbis, w, x^3+2);
Ks = Ksprim + Ksbis;

figure(2);
title('Ks(x)');
fplot (Ks, [0,1], '-k');
hold on;
fplot (Ks2, [0,1], 'x r');
hold off;


%----------------------------------------
%------------PODPUNKT 2-----------------
%----------------------------------------

delta1 =
(fTx(p)+Kcos+Kexp+Km2(p)+Ks2(p)+Kp2(p)).*eps;
dMAX1 = max(abs(delta1));
display (dMAX1);


%----------------------------------------
%------------PODPUNKT 3-----------------
```

```
%---------------------------------------

epsy = de2bi(0:4096);
epsy(epsy==0) = -1;
epsy = epsy*eps;

delta2 = zeros(1, 4096);
y = cos(p.^2+2).*exp(p.^3+2);

parfor n = 1:4096
    ep = epsy(n,:);
    yd =
cos((((p.*(1+ep(1))).^2).*(1+ep(2))+2).*(1+ep(
3))).*(1+ep(4)).*(1+ep(5)).*exp((((p.*(1+ep(6)
)).^2).*(p.*(1+ep(7))).*(1+ep(8)).*(1+ep(9))+2
).*(1+ep(10)))*(1+ep(11));
    delta2(1, n) = abs((yd-y)/y);
end

dMAX2 = max(delta2);
display (dMAX2);

%---------------------------------------
%------------PODPUNKT 4-----------------
%---------------------------------------

x =[1; -1; 0; 1; 1];
xprim = zeros(5, 1);

A = [42 -50 -160 -4 378;
    -44 46 154 20 -390;
    -37 25 114 26 -297;
    -43 25 120 38 -333;
    -25 21 82 14 -209];

b = A*x;
deltaX = zeros(1, 1000000);
Aprim = zeros (5);

parfor n=1:1000000
    dist = (2*rand(5)-1)*eps;
    Aprim = A + dist;
    xprim = Aprim\b;
    deltaX(n) = (norm(xprim - x, 2))/(norm(x,
2));
end

dXMAX = max(deltaX);
display(dXMAX);
```