

Metody numeryczne

Wykład 5



Dr inż. Michał Łanczont

Instytut Elektrotechniki i Elektrotechnologii

E419, tel. 4293, m.lanczont@pollub.pl, <http://m.lanczont.pollub.pl>

Zakres wykładu

Interpolacja

- Interpolacja wielomianowa
- Wielomiany interpolacyjne:
 - Lagrange'a
 - Newtona
- Interpolacja trygonometryczna
- Interpolujące funkcje sklejjane
- Szybka transformata Fouriera - FFT

Aproksymacja

Interpolacja

Wiele zjawisk które można zaobserwować jest opisana nieokreślonymi (nieznanymi) funkcjami.

Możliwe jest jednak zmierzenie wartości tych funkcji, oraz ich pochodnych przy określonych wartościach argumentów.

Interpolacja

Interpolacją nazywamy procedurę prowadzącą do znalezienia wartości pewnej funkcji $f(x)$ w dowolnym punkcie przedziału (x_0, x_n) na podstawie znanych wartości tej funkcji w punktach $x_0, x_1, \dots, x_n$, zwanych węzłami interpolacji.

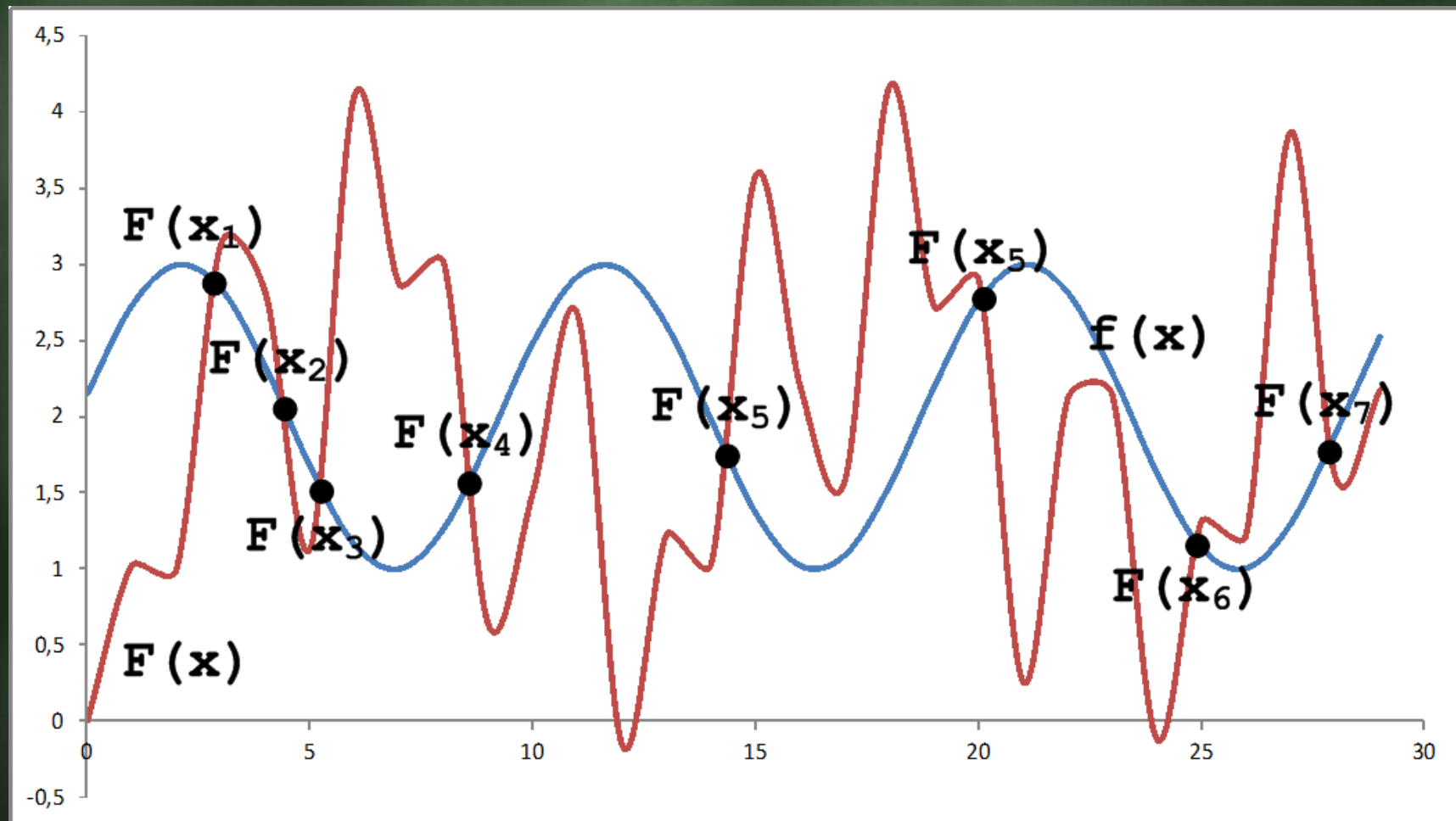
Postępowanie prowadzące do znalezienia wartości funkcji $f(x)$ w punkcie leżącym poza przedziałem (x_0, x_n) nazywamy ekstrapolacją.

Interpolacje

Interpolację lub ekstrapolację stosuje się gdy:

- Funkcja $f(x)$ nie jest znana, a posiadane są tylko informacje o jej wartościach w punktach przedziału
- Obliczanie wartości funkcji $F(x)$ jest skomplikowane (czasochłonne), wygodniej jest interpolować prostszą obliczeniowo funkcją $f(x)$ zwracającą w zadanych punktach takie same wartości jak funkcja $F(x)$

Interpolacja



Interpolacja

Funkcji interpolującej, ekstrapolującej
poszukuje się zazwyczaj w pewnej
określonej postaci:

- **Wielomianowa**
 - **Algebraiczna**
 - **Trygonometryczna**
- **Funkcjami sklejanymi**

Interpolacja wielomianowa

Interpolacja wielomianowa opiera się na wielomianie stopnia co najmniej n , gdzie n jest liczbą węzłów (punktów) interpolacji.

$$W(x_i) = y_i$$

gdzie:

$$W(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + \dots + a_nx^n$$

Interpolacja wielomianowa

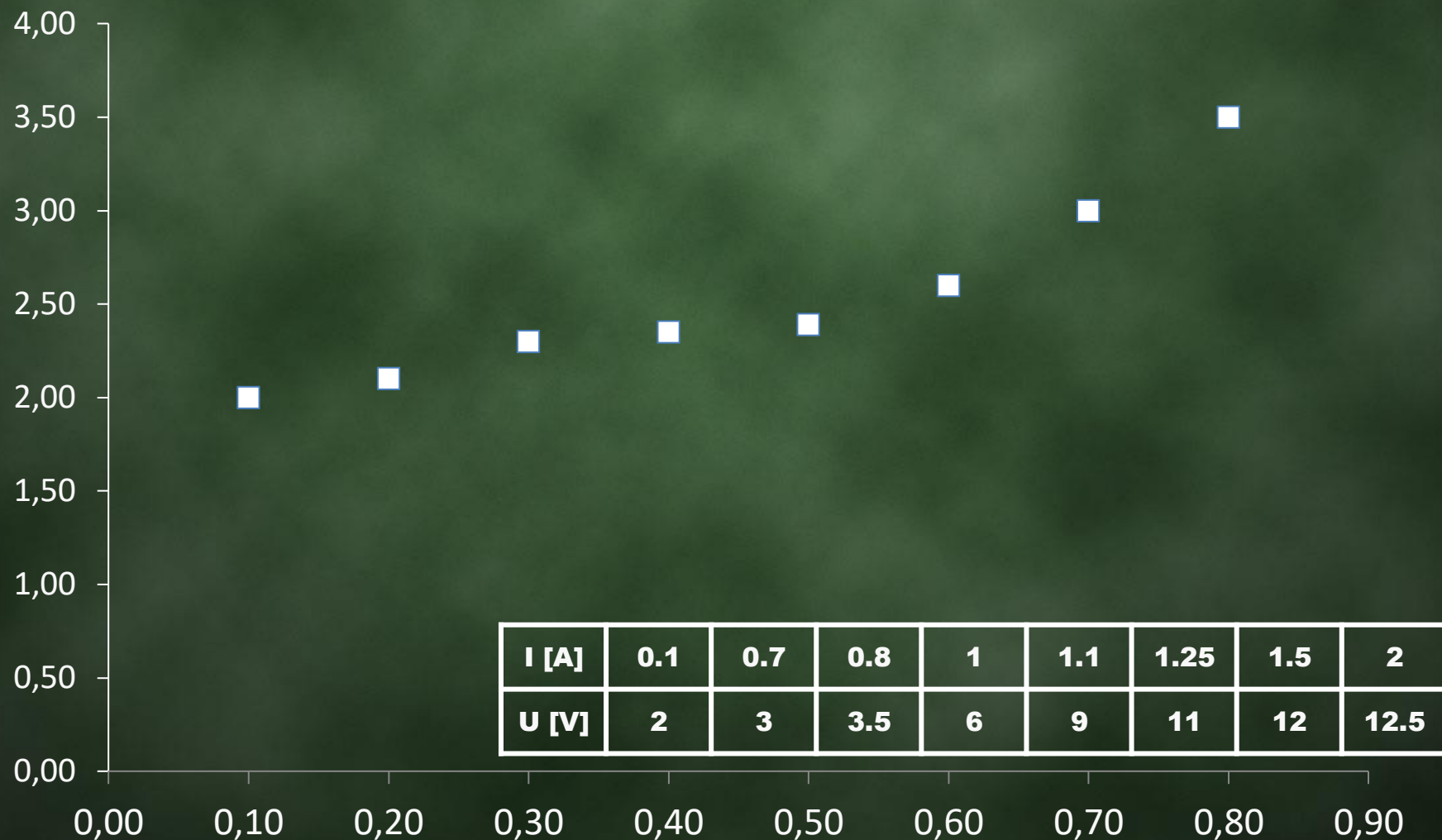
Dla n węzłów interpolacyjnych można więc zapisać równanie macierzowe opisujące warunki jakie musi spełnić wielomian interpolacyjny: **$Xa=y$**

$$\begin{vmatrix} x_0^0 & x_0^1 & x_0^2 & \cdots & x_0^n \\ x_1^0 & x_1^1 & x_1^2 & \cdots & x_1^n \\ x_2^0 & x_2^1 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_n^0 & x_n^1 & x_n^2 & \cdots & x_n^n \end{vmatrix} \begin{vmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{vmatrix} = \begin{vmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{vmatrix}$$

Interpolacja wielomianowa

- Poszukiwane wartości współczynników można policzyć z zależności $\mathbf{a} = \mathbf{X}^{-1}\mathbf{y}$
- Wyznaczenie współczynników można także zrealizować rozwiązując układ równań liniowych, funkcja $\text{linsolve}(\mathbf{X}, -\mathbf{y})$
- Warunkiem uzyskania jednego rozwiązania jest $\det \mathbf{X} \neq 0$

Interpolacja – przykład 1



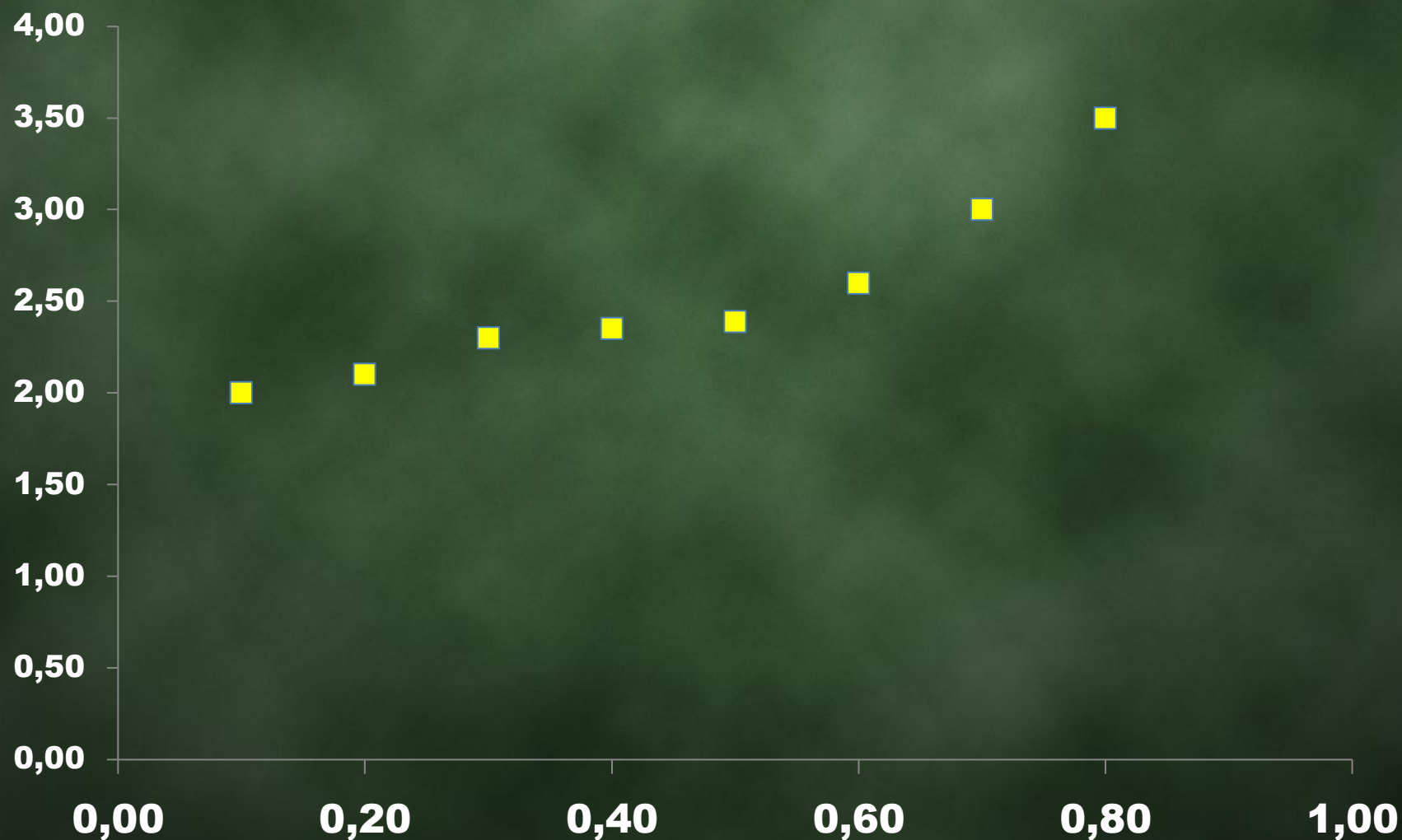
Interpolacja – przykład 1

$$X = \begin{vmatrix} 0.1^0 & 0.1^1 & 0.1^2 & 0.1^3 & 0.1^4 & 0.1^5 & 0.1^6 & 0.1^7 \\ 0.7^0 & 0.7^1 & 0.7^2 & 0.7^3 & 0.7^4 & 0.7^5 & 0.7^6 & 0.7^7 \\ 0.8^0 & 0.8^1 & 0.8^2 & 0.8^3 & 0.8^4 & 0.8^5 & 0.8^6 & 0.8^7 \\ 1^0 & 1^1 & 1^2 & 1^3 & 1^4 & 1^5 & 1^6 & 1^7 \\ 1.1^0 & 1.1^1 & 1.1^2 & 1.1^3 & 1.1^4 & 1.1^5 & 1.1^6 & 1.1^7 \\ 1.25^0 & 1.25^1 & 1.25^2 & 1.25^3 & 1.25^4 & 1.25^5 & 1.25^6 & 1.25^7 \\ 1.5^0 & 1.5^1 & 1.5^2 & 1.5^3 & 1.5^4 & 1.5^5 & 1.5^6 & 1.5^7 \\ 2^0 & 2^1 & 2^2 & 2^3 & 2^4 & 2^5 & 2^6 & 2^7 \end{vmatrix}, \quad y = \begin{vmatrix} 2 \\ 3 \\ 3.5 \\ 6 \\ 9 \\ 11 \\ 12 \\ 12.5 \end{vmatrix}$$

$$X^{-1}y = a = \begin{vmatrix} 340.31688 \\ -5293.8827 \\ 23490.778 \\ -49060.247 \\ 55645.028 \\ -35216.121 \\ 11678.743 \\ -1578.6161 \end{vmatrix}$$

$$f(x) \approx 340 - 5293x + 23490x^2 - 49060x^3 + 55645x^4 - 35216x^5 + 11678x^6 - 1578x^7$$

Interpolacja – przykład 1



Interpolacja Lagrang'a

Interpolacja wielomianem Lagrang'a jest odmianą interpolacji wielomianowej w której nieznaną funkcję interpoluje się do wielomianu iteracyjnym w postaci:

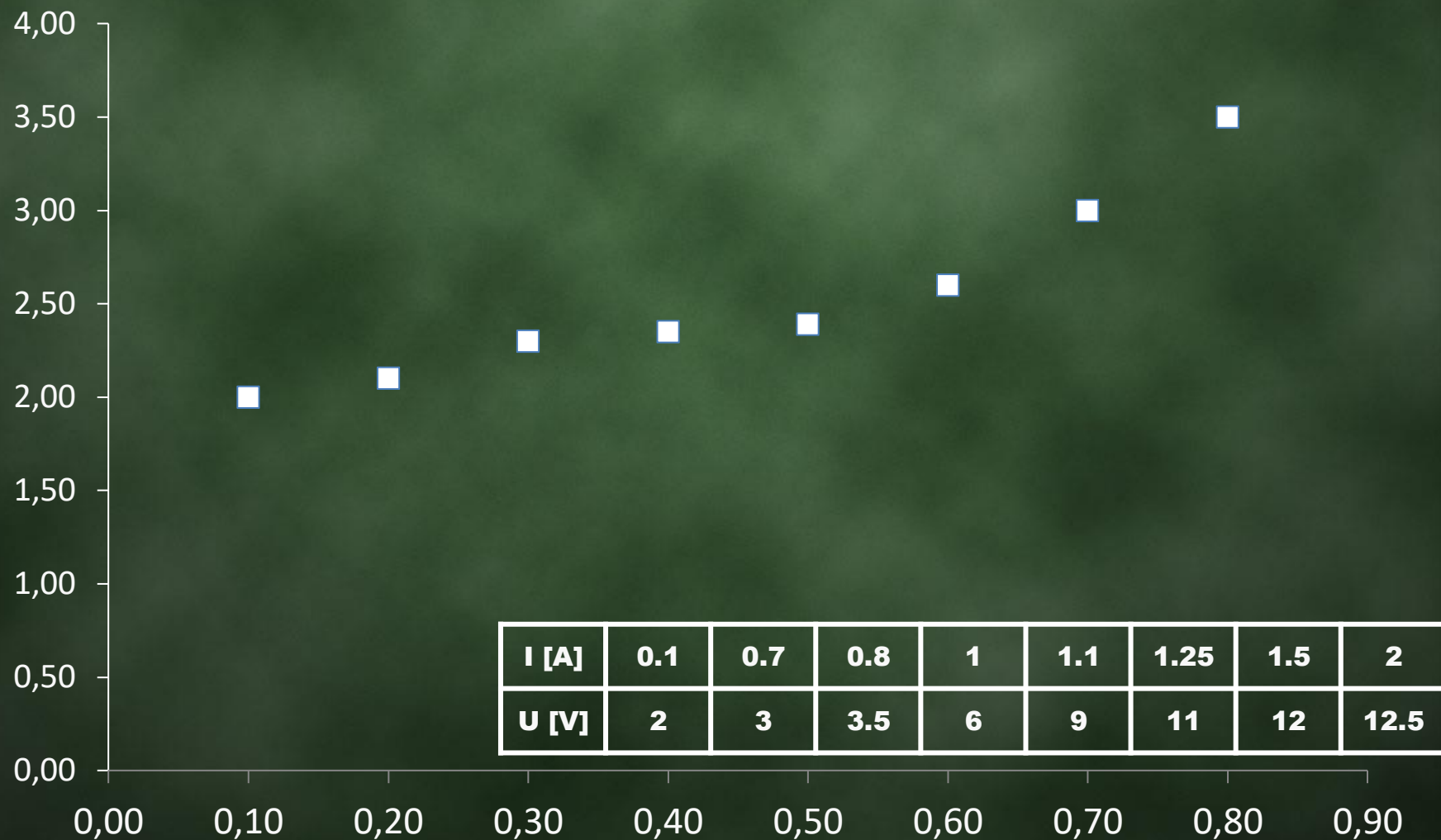
$$L(x) = \sum_{i=0}^n y_i \cdot \prod_{\substack{j=0 \\ j \neq i}}^n \frac{(x - x_j)}{(x_i - x_j)}$$

Interpolacja Lagrang'a

Rozwijając wielomian $L(x)$ otrzymujemy:

$$\begin{aligned} L(x) = & y_0 \frac{(x - x_1)(x - x_2) \cdots (x - x_n)}{(x_0 - x_1)(x_0 - x_2) \cdots (x_0 - x_n)} \\ & + y_1 \frac{(x - x_0)(x - x_2) \cdots (x - x_n)}{(x_1 - x_0)(x_1 - x_2) \cdots (x_1 - x_n)} + \cdots + \\ & + y_n \frac{(x - x_0)(x - x_1) \cdots (x - x_{n-1})}{(x_n - x_0)(x_n - x_1) \cdots (x_n - x_{n-1})} \end{aligned}$$

Lagrang– przykład 2



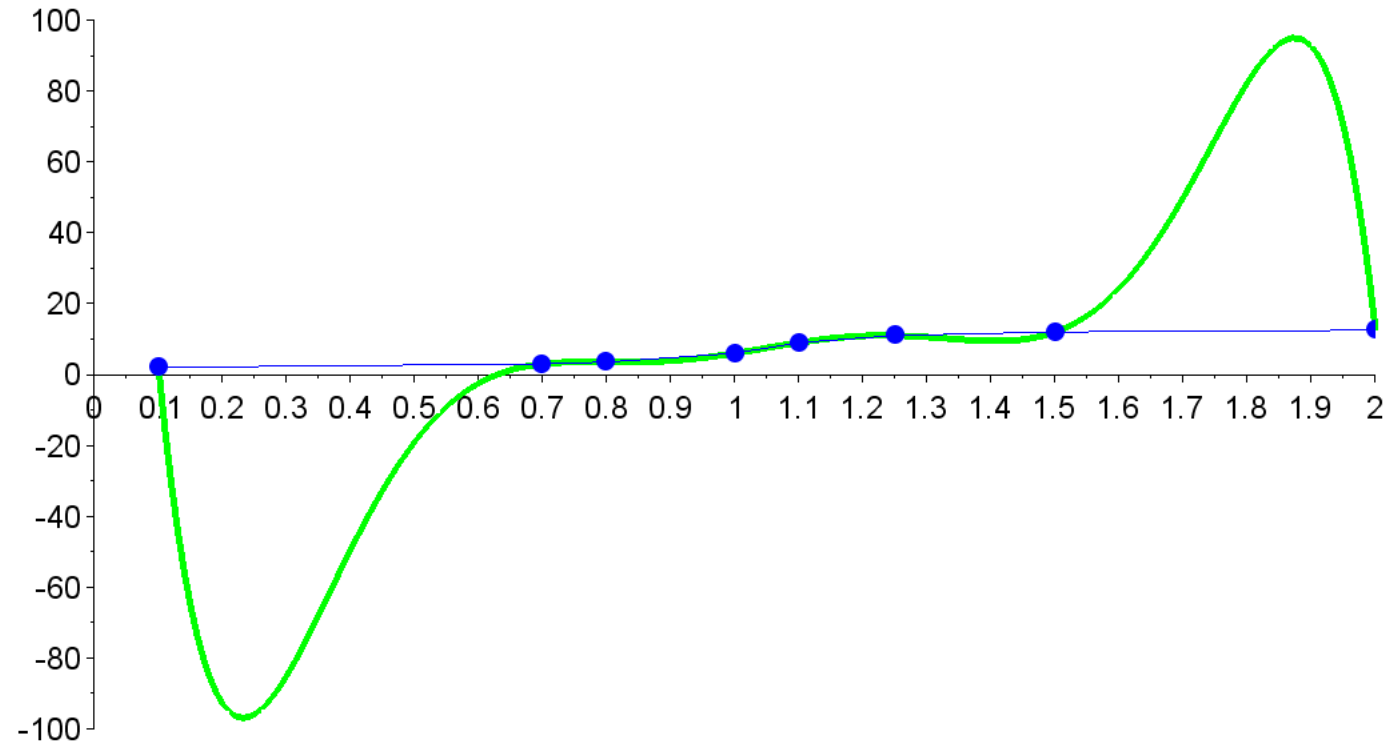
Lagrang– przykład 2

$$\begin{aligned}
 L(x) = & 2 \frac{(x - 0.7)(x - 0.8)(x - 1)(x - 1.1)(x - 1.25)(x - 1.5)(x - 2)}{(0.1 - 0.7)(0.1 - 0.8)(0.1 - 1)(0.1 - 1.1)(0.1 - 1.25)(0.1 - 1.5)(0.1 - 2)} + \\
 & +3 \frac{(x - 0.1)(x - 0.8)(x - 1)(x - 1.1)(x - 1.25)(x - 1.5)(x - 2)}{(0.7 - 0.1)(0.7 - 0.8)(0.7 - 1)(0.7 - 1.1)(0.7 - 1.25)(0.7 - 1.5)(0.7 - 2)} + \\
 & +3.5 \frac{(x - 0.1)(x - 0.7)(x - 1)(x - 1.1)(x - 1.25)(x - 1.5)(x - 2)}{(0.8 - 0.1)(0.8 - 0.7)(0.8 - 1)(0.8 - 1.1)(0.8 - 1.25)(0.8 - 1.5)(0.8 - 2)} + \\
 & +6 \frac{(x - 0.1)(x - 0.7)(x - 0.8)(x - 1.1)(x - 1.25)(x - 1.5)(x - 2)}{(1 - 0.1)(1 - 0.7)(1 - 0.8)(1 - 1.1)(1 - 1.25)(1 - 1.5)(1 - 2)} + \\
 & +9 \frac{(x - 0.1)(x - 0.7)(x - 0.8)(x - 1)(x - 1.25)(x - 1.5)(x - 2)}{(1.1 - 0.1)(1.1 - 0.7)(1.1 - 0.8)(1.1 - 1)(1.1 - 1.25)(1.1 - 1.5)(1.1 - 2)} + \\
 & +11 \frac{(x - 0.1)(x - 0.7)(x - 0.8)(x - 1)(x - 1.1)(x - 1.5)(x - 2)}{(1.25 - 0.1)(1.25 - 0.7)(1.25 - 0.8)(1.25 - 1)(1.25 - 1.1)(1.25 - 1.5)(1.25 - 2)} + \\
 & +12 \frac{(x - 0.1)(x - 0.7)(x - 0.8)(x - 1)(x - 1.1)(x - 1.25)(x - 2)}{(1.5 - 0.1)(1.5 - 0.7)(1.5 - 0.8)(1.5 - 1)(1.5 - 1.1)(1.5 - 1.25)(1.5 - 2)} + \\
 & +12.5 \frac{(x - 0.1)(x - 0.7)(x - 0.8)(x - 1)(x - 1.1)(x - 1.25)(x - 1.5)}{(2 - 0.1)(2 - 0.7)(2 - 0.8)(2 - 1)(2 - 1.1)(2 - 1.25)(2 - 1.5)}
 \end{aligned}$$

Lagrang– przykład 2

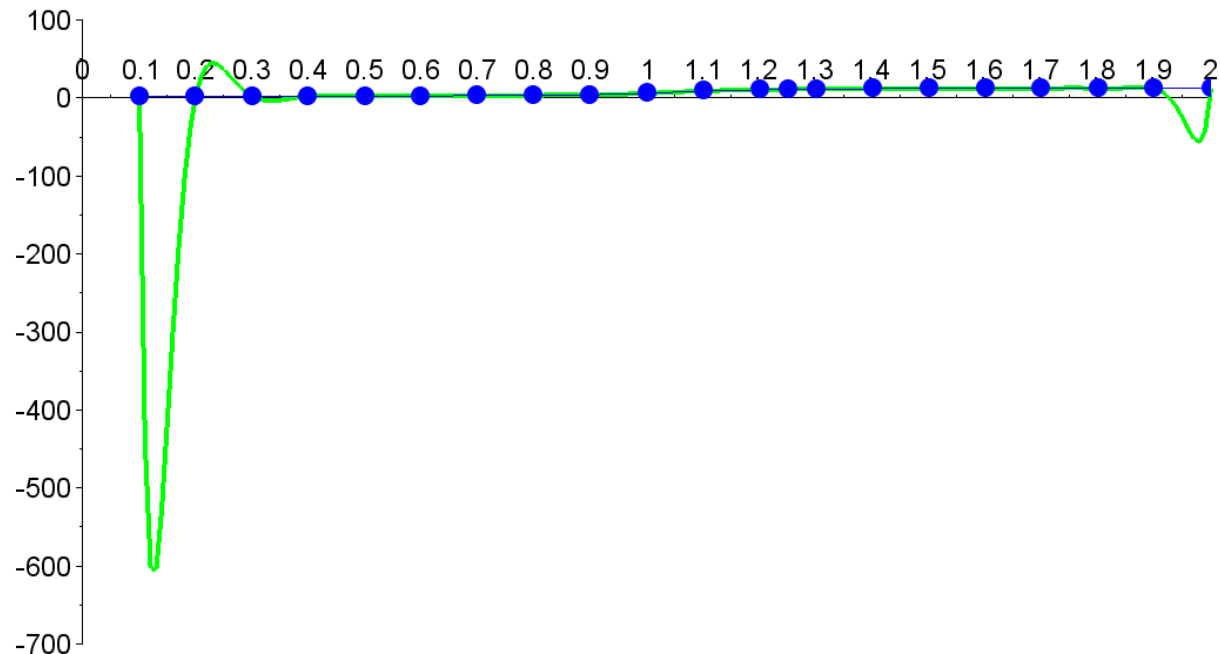
```
1 clear;
2 clc;
3 i=[0.1;0.7;0.8;1;1.1;1.25;1.5;2];
4 u=[2;3;3.5;6;9;11;12;12.5];
5 a=0.1;b=2;c=0.01;n=(b-a)/c+1;
6 xw=zeros(1,n);yw=zeros(1,n);
7 s=1;
8 for x=a:c:b
9     xw(s)=x;
10    for k=1:8
11        t=1;
12        for j=1:8
13            if (j<>k)
14                t=t*((x-i(j))/(i(k)-i(j)));
15            end
16        end
17        yw(s)=yw(s)+u(k)*t;
18    end
19    s=s+1;
20 end
21 plot2d(xw,yw,3)
22 plot2d(i,u,2)
```

Lagrang– przykład 2



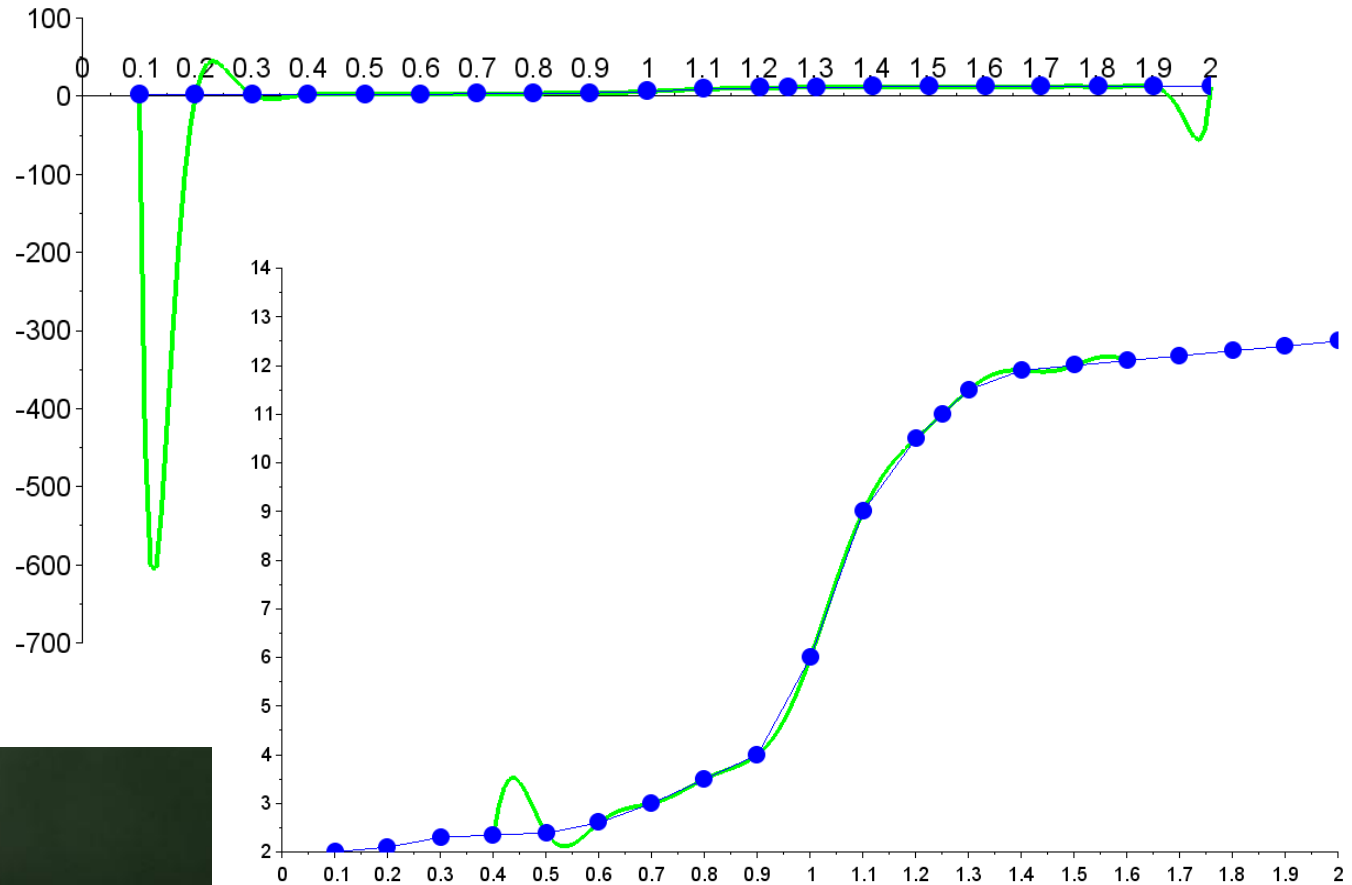
Lagrang– przykład 2

I	U
0,10	2,00
0,20	2,10
0,30	2,30
0,40	2,35
0,50	2,39
0,60	2,60
0,70	3,00
0,80	3,50
0,90	4,00
1,00	6,00
1,10	9,00
1,20	10,50
1,25	11,00
1,30	11,50
1,40	11,90
1,50	12,00
1,60	12,10
1,70	12,20
1,80	12,30
1,90	12,40
2,00	12,50



Lagrang – przykład 2

I	U
0,10	2,00
0,20	2,10
0,30	2,30
0,40	2,35
0,50	2,39
0,60	2,60
0,70	3,00
0,80	3,50
0,90	4,00
1,00	6,00
1,10	9,00
1,20	10,50
1,25	11,00
1,30	11,50
1,40	11,90
1,50	12,00
1,60	12,10
1,70	12,20
1,80	12,30
1,90	12,40
2,00	12,50



Interpolacja Newtona

11:26

Ilorazów różnicowych

Metoda Newtona opiera się na wykorzystaniu wyrażeń zwanych ilorazami różnicowymi.

Wyrazy te określa się dla $n-1$ rzędów gdzie n jest liczbą punktów pomiarowych (węzłów interpolacyjnych).

Na podstawie wyznaczonych ilorazów różnicowych tworzy się zazwyczaj tablicę.

Interpolacja Newtona

1. Pierwszego rzędu

$$[x_0, x_1] = \frac{y_1 - y_0}{x_1 - x_0}$$

$$[x_1, x_2] = \frac{y_2 - y_1}{x_2 - x_1}$$

...

$$[x_{n-1}, x_n] = \frac{y_n - y_{n-1}}{x_n - x_{n-1}}$$

2. Drugiego rzędu

$$[x_0, x_1, x_2] = \frac{[x_1, x_2] - [x_0, x_1]}{x_2 - x_0}$$

$$[x_1, x_2, x_3] = \frac{[x_2, x_3] - [x_1, x_2]}{x_3 - x_1}$$

...

$$[x_{n-2}, x_{n-1}, x_n] = \frac{[x_{n-1}, x_n] - [x_{n-2}, x_{n-1}]}{x_n - x_{n-2}}$$

3. k-tego rzędu

$$[x_i, x_{i+1}, \dots, x_{i+k}] = \frac{[x_{i+1}, x_{i+2}, \dots, x_{i+k}] - [x_i, x_{i+1}, \dots, x_{i+k-1}]}{x_{i+k} - x_i}$$

Interpolacja Newtona

Wzór ogólny iteracyjny na iloraz różnicowy można zapisać w postaci:

for $i = 1$ *to* $n - 1$

for $j = 2$ *to* n

if $(j < (i + 1))$ *then continue*

$$R_{j,i} = \frac{R_{j,i} - R_{j-1,i}}{x_j - x_{j-i}}$$

Interpolacja Newtona

Wzór interpolacyjny Newtona z różnicami ilorazowymi przyjmuje postać:

$$N_n(x) = y_0 + [x_0, x_1](x - x_0) + [x_0, x_1, x_2](x - x_0)(x - x_1) + \dots + [x_0, x_1, \dots, x_n](x - x_0)(x - x_1) \dots (x - x_{n-1})$$

$$I_n = [x_0, \dots, x_n]$$

$$N_n(x) = y_0 + \sum_{i=0}^n \left(I_n \prod_{j=0}^{i-1} (x - x_j) \right)$$

Interpolacja Newtona

Na podstawie zapisanego równania iteracyjnego wielomianu można zapisać algorytm komputerowy dla interpolacji Newtona.

for $i = 1$ to n

for $j = 0$ to $(i - 1)$

$$e = e \cdot (x - x_j)$$

$$f = l_i \cdot e$$

$$y = y_0 + f$$

Newtona – przykład 3

x	y	I	II	III	IV	V	VI	VII		
0.1	2	$N_n(x) = y_0 + \sum_{i=0}^n \left(I_n \prod_{j=0}^{i-1} (x - x_j) \right)$								
0.7	3								1.67	
0.8	3.5								5	4.76
1	6								12.5	25
1.1	9	30	58.3	83.2	60.7					
1.25	11	13.3	-66.8	-282.4	-664.7	-630.8				
1.5	12	4	-23.2	91.2	533.7	1498	1429.3			
2	12.5	1	-4	21.3	-69.9	-503	-1539.2	-1562.4		

Newtona – przykład 3

```
1 clear;  
2 clc;  
3 a=0.1;  
4 b=2;  
5 c=0.01;  
6 M=(b-a)/c+1;  
7 n=6;  
8 x=[0.1;0.7;0.8;1;1.1;1.25;1.5;2];  
9 y=[2;3;3.5;6;9;11;12;12.5];  
10 R=zeros(n,n);  
11 t=zeros(1,M);  
12 f=zeros(1,M);
```

Newtona – przykład 3

```
13 for i=1:n
14     R(i,1)=y(i);
15 end
16 for i=1:n-1
17     for j=2:n
18         if (i+1)>j continue; end
19         R(j,i+1)=(R(j,i)-R(j-1,i))/(x(j)-x(j-i))
20     end
21 end
```

Newtona – przykład 3

R =

2.	0.	0.	0.	0.	0.	0.	0.
3.	1.6666667	0.	0.	0.	0.	0.	0.
3.5	5.	4.7619048	0.	0.	0.	0.	0.
6.	12.5	25.	22.486772	0.	0.	0.	0.
9.	30.	58.333333	83.333333	60.846561	0.	0.	0.
11.	13.333333	-66.666667	-277.77778	-656.56566	-623.83671	0.	0.
12.	4.	-23.333333	86.666667	520.63492	1471.5007	1496.6696	0.
12.5	1.	-4.	21.481481	-65.185185	-488.18342	-1507.4493	-1581.1152

Newtona – przykład 3

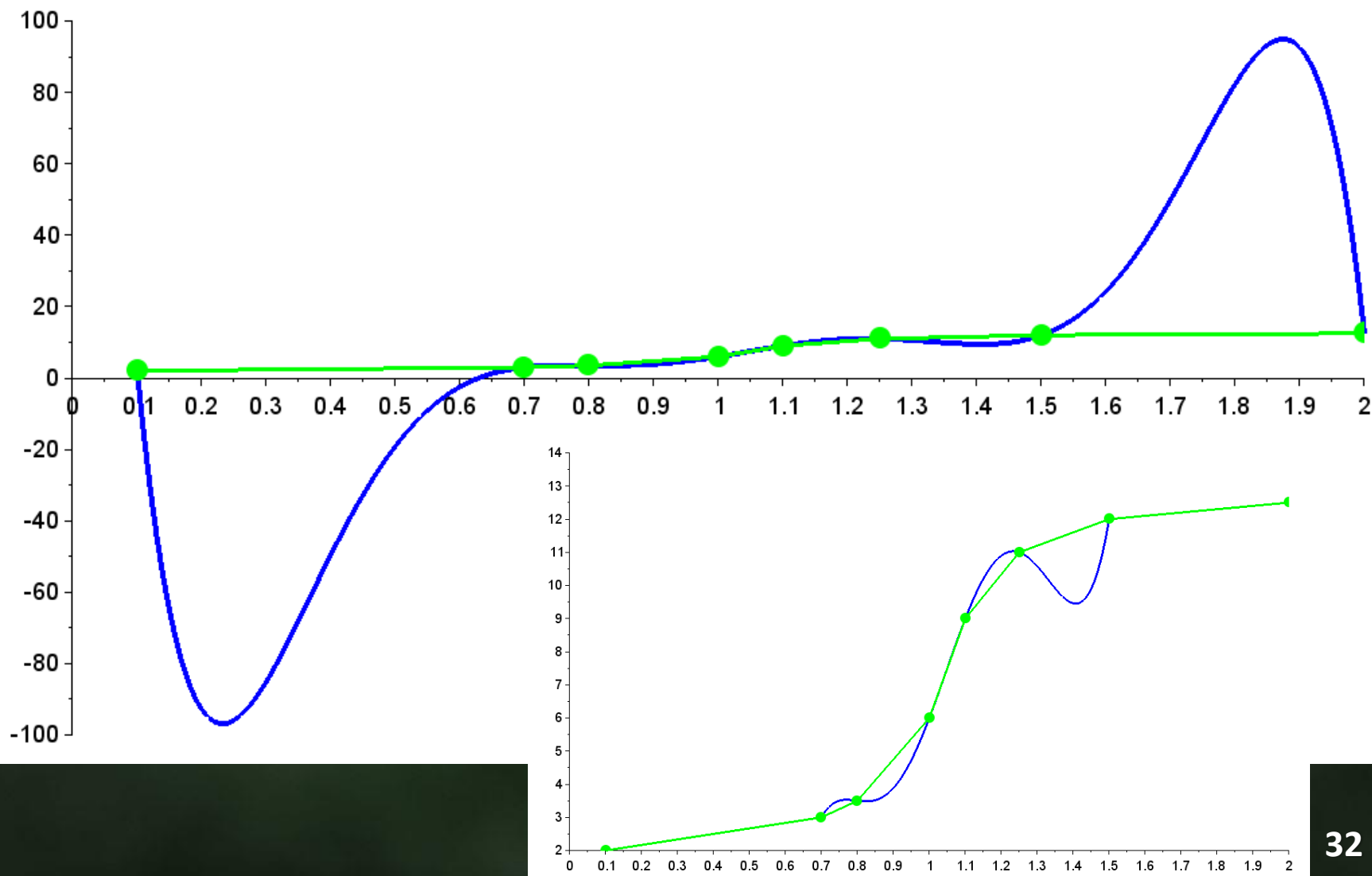
```

22 k=1;
23 for v=a:c:b
24     m=0;
25     for j=2:n
26         z=1;
27         for i=1:(j-1)
28             z=z*(v-x(i));
29         end
30         m=m+R(j,j)*z
31     end
32     t(k)=v;
33     f(k)=R(1,1)+m;
34     k=k+1;
35 end

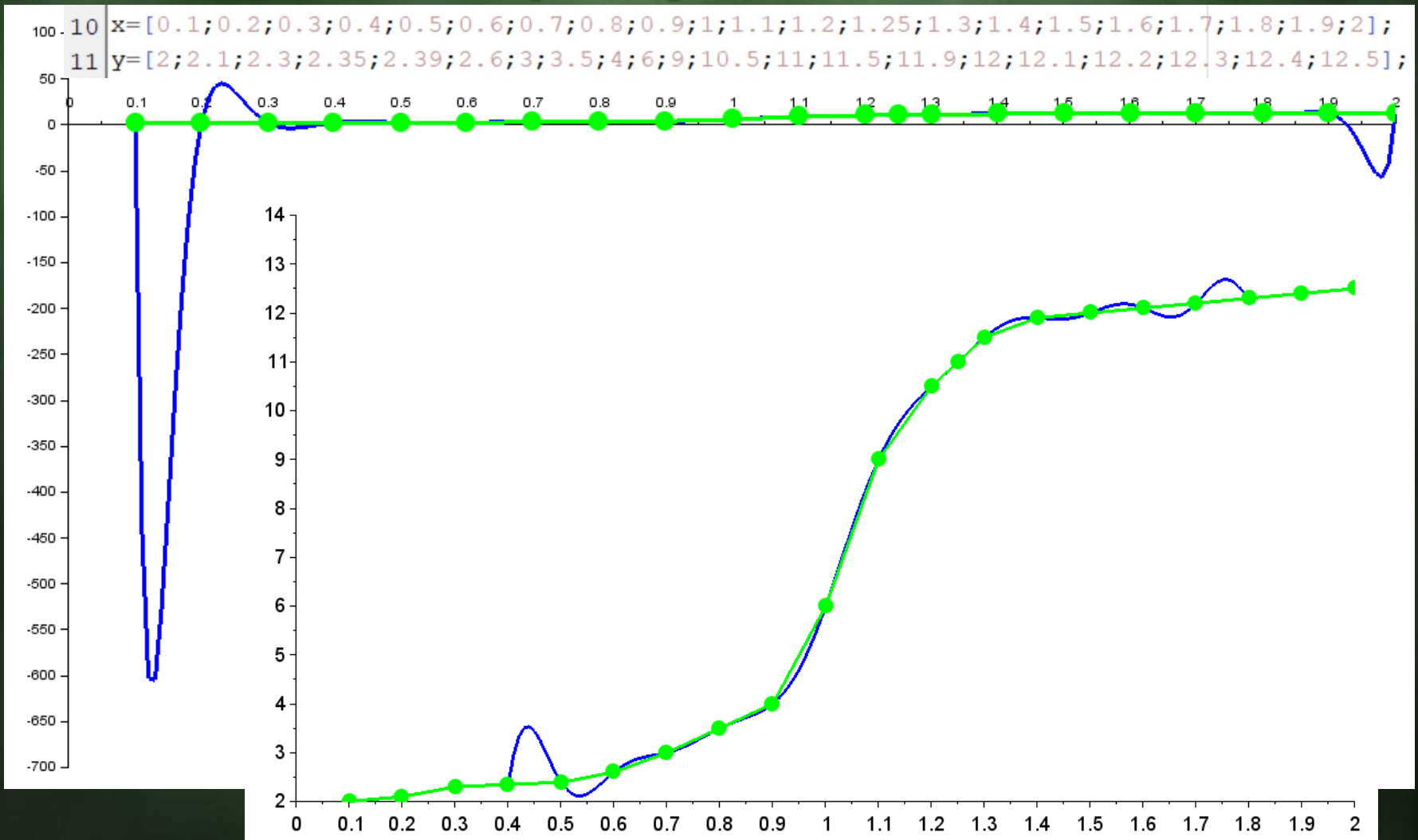
```

$$N_n(x) = y_0 + \sum_{i=0}^n \left(I_n \prod_{j=0}^{i-1} (x - x_j) \right)$$

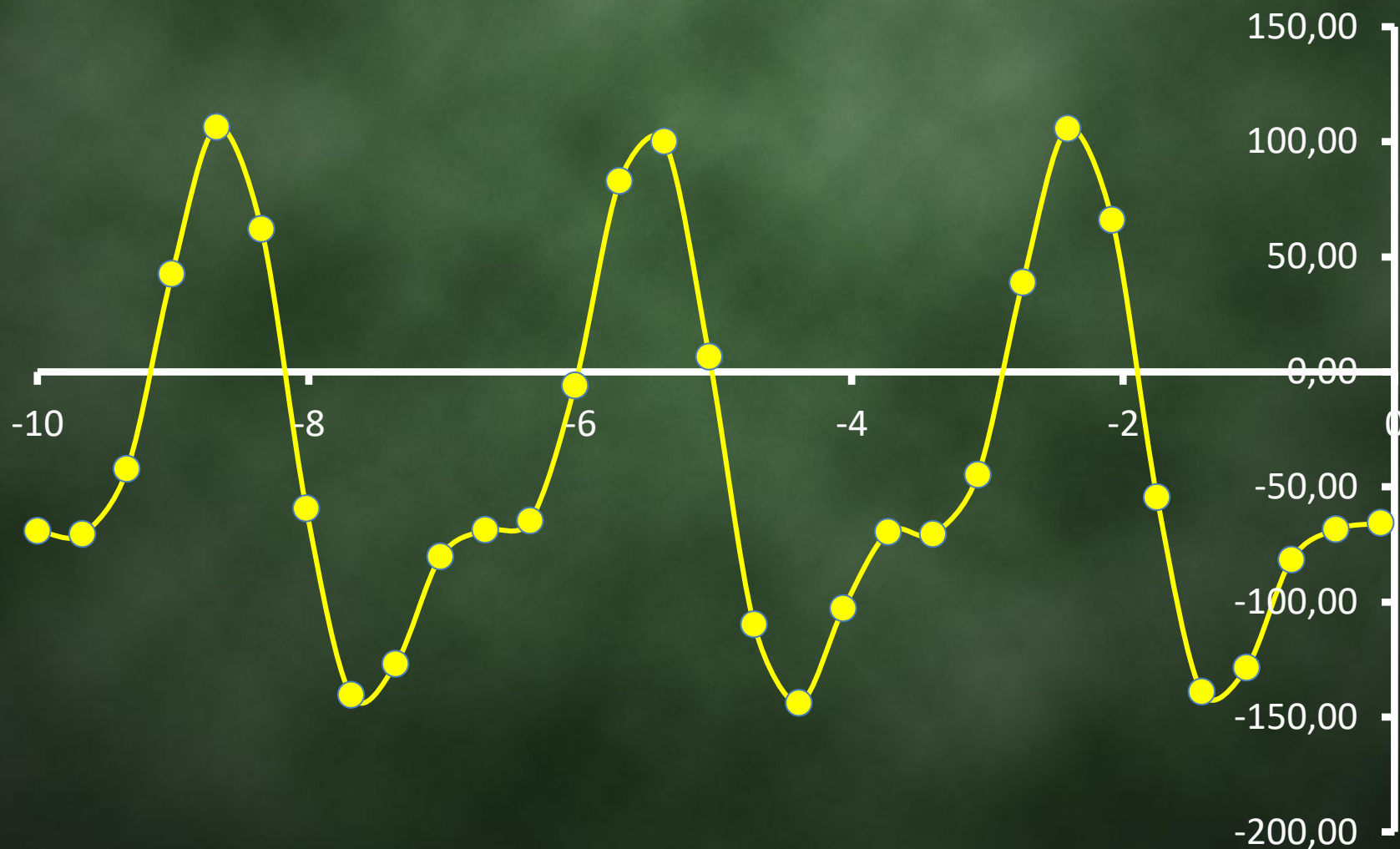
Newtona – przykład 3



Newtona – przykład 3



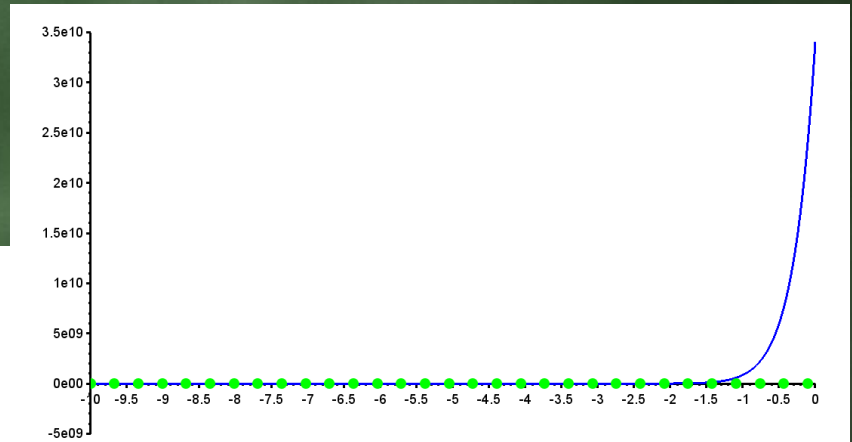
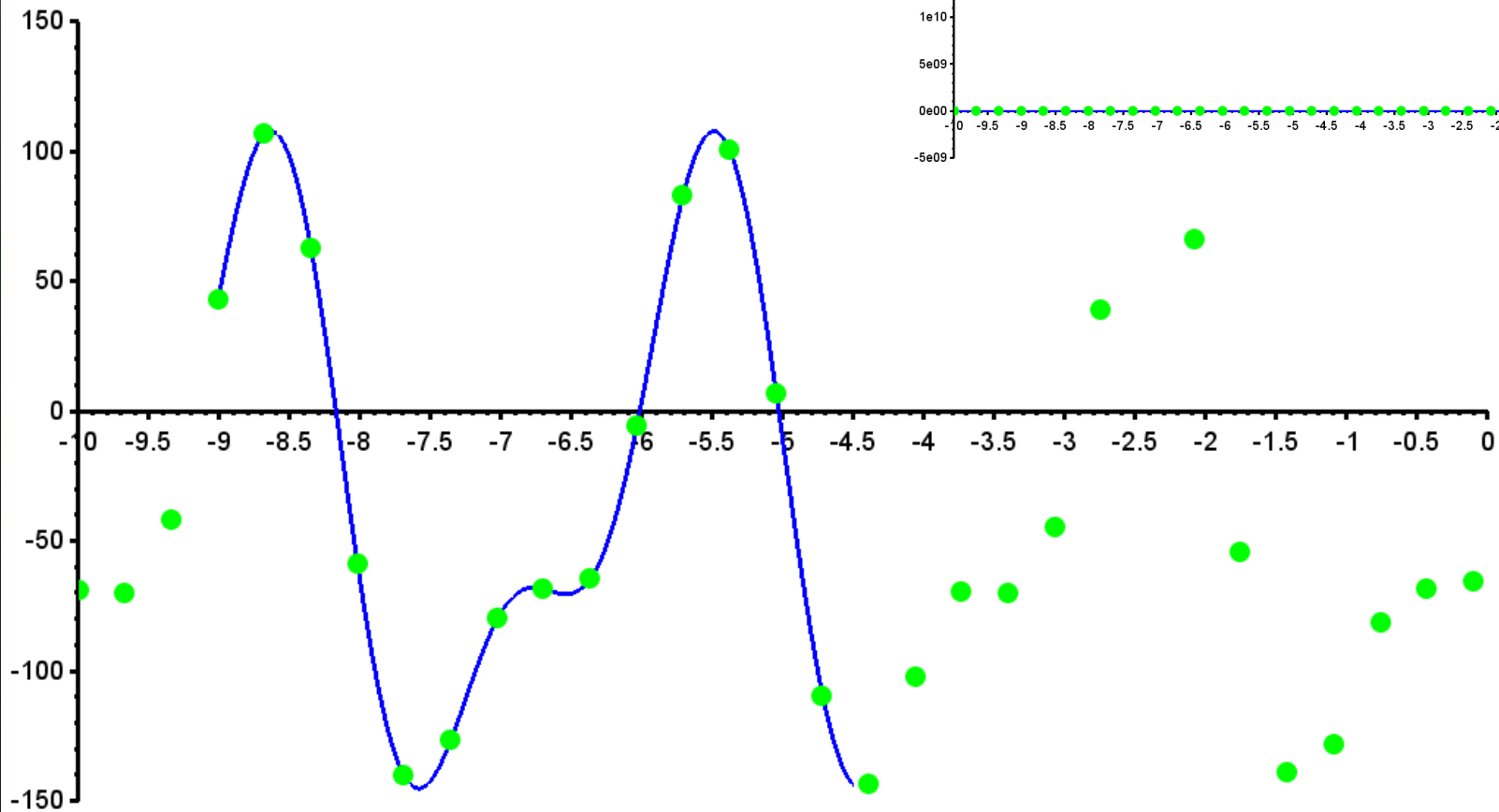
Interpolacja – sygnał okresowy



Interpolacja – Lagrang'a

sygnał okresowy

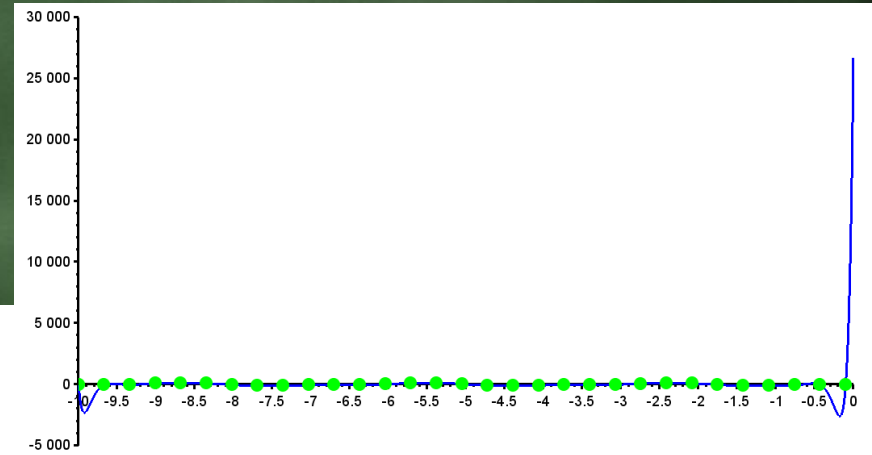
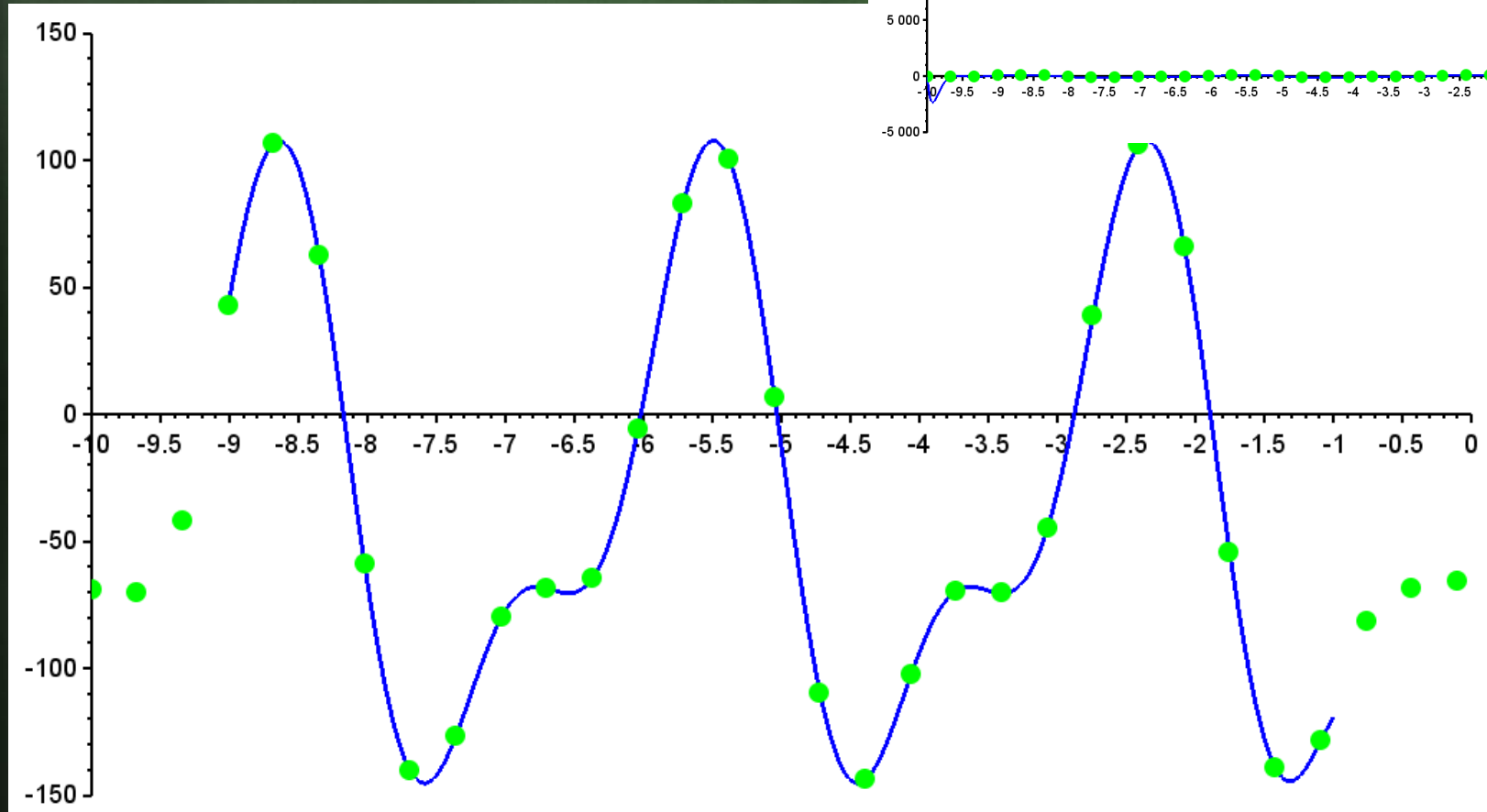
11:26



Interpolacja – Newtona

sygnał okresowy

11:26



Interpolacja trygonometryczna

- Interpolacje algebraiczne wielomianowe z racji swoich właściwości nie bardzo nadają się do opisu sygnałów okresowych
- Interpolację trygonometryczną stosuje się do wyznaczania funkcji interpolującej dla funkcji okresowych (np. sinusoidalnych)
- Dobrze nadaje się do interpolacji sygnałów elektrycznych czy drgań mechanicznych

Interpolacja trygonometryczna

Dana jest funkcja okresowa f o okresie 2π , zmiennej rzeczywistej o wartościach zespolonych:

$$f(x+2\pi)=f(x)$$

Dowolną funkcję okresową o okresie t można poprzez podstawienie $x = \frac{2\pi}{t}y$ sprowadzić do funkcji okresowej 2π

$$g(y+t) = g(y)$$

$$f(x) = g\left(\frac{xt}{2\pi}\right)$$

Można więc przy takim uogólnieniu rozpatrywać tylko funkcje o okresie 2π .

Interpolacja trygonometryczna

Interpolacja trygonometryczna sprowadza się do określenia dla nieznanej funkcji f wielomianu trygonometrycznego F przyjmującej dla danych węzłów interpolacyjnych te same wartości.

$$F_n(x) = \sum_{m=0}^n c_m e^{j \cdot m \cdot x} = \sum_{m=0}^n c_m (\sin(m \cdot x) + j \cos(m \cdot x))$$

$$F_n(x_k) = f(x_k)$$

Interpolacja trygonometryczna

- Warunki interpolacyjne można zapisać w postaci $n+1$ równań niewiadomych $c_0, c_1, \dots, c_n$

$$\sum_{m=0}^n c_m e^{jmx_k} = f(x_k)$$

- Macierz tego układu jest nieosobliwą macierzą Vandermode'a, układ ma jedno rozwiązanie.
- Wyznacznik macierzy jest różny od **0**, przy założeniu że węzły **x_k** są niepowtarzalne.

Interpolacja trygonometryczna

Jako kryterium upraszczające do dalszych rozważań przyjmuje się że węzły interpolujące są równoodległe.

$$x_k = \frac{2k\pi}{n+1}, k = 0, 1, \dots, n$$

Interpolacja trygonometryczna

Funkcje e^{jmx} , $m=0,1,2,\dots,n$ tworzą układ ortogonalny, dla iloczynu skalarnego:

$$(f, g) = \sum_{k=0}^n f(x_k) g(x_k)$$

stąd

$$(e^{jlx}, e^{jmx}) = \sum_{k=0}^n e^{jlx_k} e^{-jmx_k} = \sum_{k=0}^n e^{j(l-m)\frac{2k\pi}{n+1}} =$$

$$= \begin{cases} \frac{e^{j(l-m)2\pi} - 1}{e^{j(l-m)\frac{2k\pi}{n+1}} - 1} = 0 & \text{dla } l \neq m \\ n + 1 & \text{dla } l = m \end{cases}$$

Interpolacja trygonometryczna

- Współczynniki wielomianu trygonometrycznego c_m są równe:

$$c_m = \frac{(f, e^{jmx})}{n+1} = \frac{1}{n+1} \sum_{k=0}^n f(x_k) e^{-jmx_k}$$

- Współczynniki c_m są równe współczynnikom rozwinięcia Fouriera funkcji f .
- Wyznaczanie współczynników wielomianu trygonometrycznego funkcji f określa się mianem **dyskretnej transformaty Fouriera**.

Interpolacja trygonometryczna

Trygonometryczny wielomian
interpolacyjny funkcji f oparty na węzłach

interpolacyjnych $x_k = \frac{2k\pi}{n+1}, k = 0, 1, \dots, n$

można przedstawić w postaci:

$$F_n(x) = \frac{1}{2}a_0 + \sum_{k=1}^m (a_k \cos(kx) + b_k \sin(kx)) + \frac{\delta}{2}a_{m+1} \cos((m+1)x)$$

gdzie:

$$\begin{cases} \delta = 0 & m = \frac{n}{2} & \text{gdy } n \text{ parzyste} \\ \delta = 1 & m = \frac{n-1}{2} & \text{gdy } n \text{ nieparzyste} \end{cases}$$

Interpolacja trygonometryczna

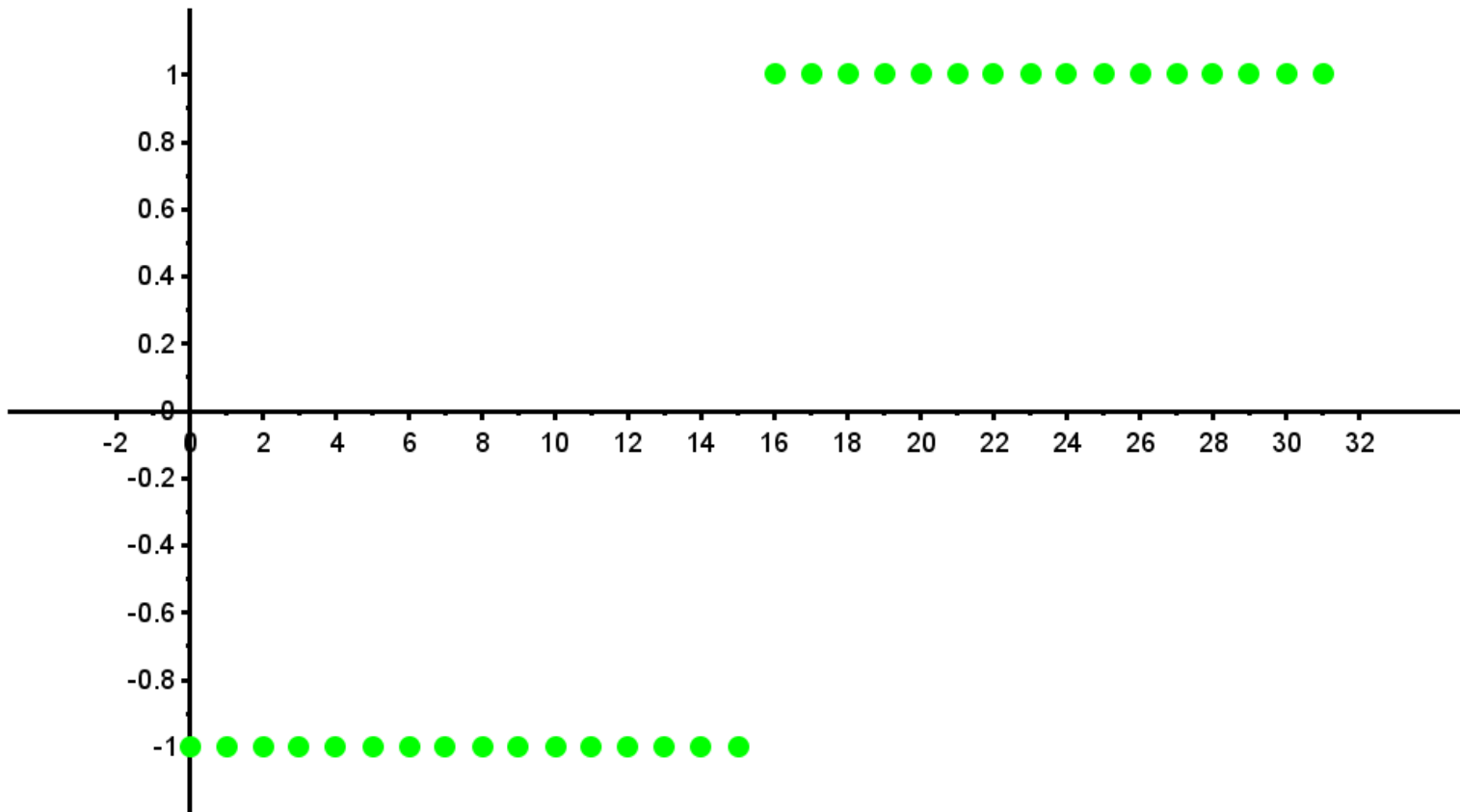
Współczynniki wielomianu wyznaczone są z zależności:

$$a_i = \frac{2}{n+1} \sum_{k=0}^n f(x_k) \cos(ix_k), \text{ dla } i = 0, 1, 2, \dots, m$$

$$b_i = \frac{2}{n+1} \sum_{k=0}^n f(x_k) \sin(ix_k), \text{ dla } i = 1, 2, \dots, m$$

Przykład 4

Określić algorytm interpolacyjny funkcji f dla podanych węzłów interpolacyjnych.



Przykład 4

$n=31$ – nieparzyste, więc $m=(n-1)/2=15$ i $\delta=1$

Na podstawie zależności: $x = \frac{2\pi}{t}y$ wyznaczane są kolejne węzły interpolacyjne.

$$a_i = \frac{2}{32} \sum_{k=0}^{31} f(x_k) \cos(ix_k), \text{ dla } i = 0, 1, 2, \dots, 16$$

$$b_i = \frac{2}{32} \sum_{k=0}^{31} f(x_k) \sin(ix_k), \text{ dla } i = 1, 2, \dots, 15$$

Przykład 4

Na podstawie wyznaczonego wielomianu trygonometrycznego wyznaczane są interpolowane wartości funkcji.

$$F_{31}(x) = \frac{a_0}{2} + \sum_{k=1}^{15} (a_k \cos(kx) + b_k \sin(kx)) + \frac{a_{16}}{2} \cos(16x)$$

Przykład 4

```
20 for i=1:m+2
21     for j=1:n
22         ak(i)=ak(i)+y(j)*cos((i-1)*xk(j));
23     end
24     ak(i)=2/n*ak(i);
25 end
26 for i=1:m
27     for j=1:n
28         bk(i)=bk(i)+y(j)*sin(i*xk(j));
29     end
30     bk(i)=2/n*bk(i);
31 end
32 for k=1:M
33     for i=1:m
34         f(k)=f(k)+(ak(i+1)*cos(i*t(k))+bk(i)*sin(i*t(k)));
35     end
36     f(k)=f(k)+ak(1)/2+(ak(m+2)/2)*cos((n+1)*t(k));
37 end
38 plot2d(x0,y,3);
39 plot2d(t2,f,2);
```

Przykład 4

$$F_{31}(x) = \frac{a_0}{2} + \sum_{k=1}^{15} (a_k \cos(kx) + b_k \sin(kx)) + \frac{a_{16}}{2} \cos(16x)$$

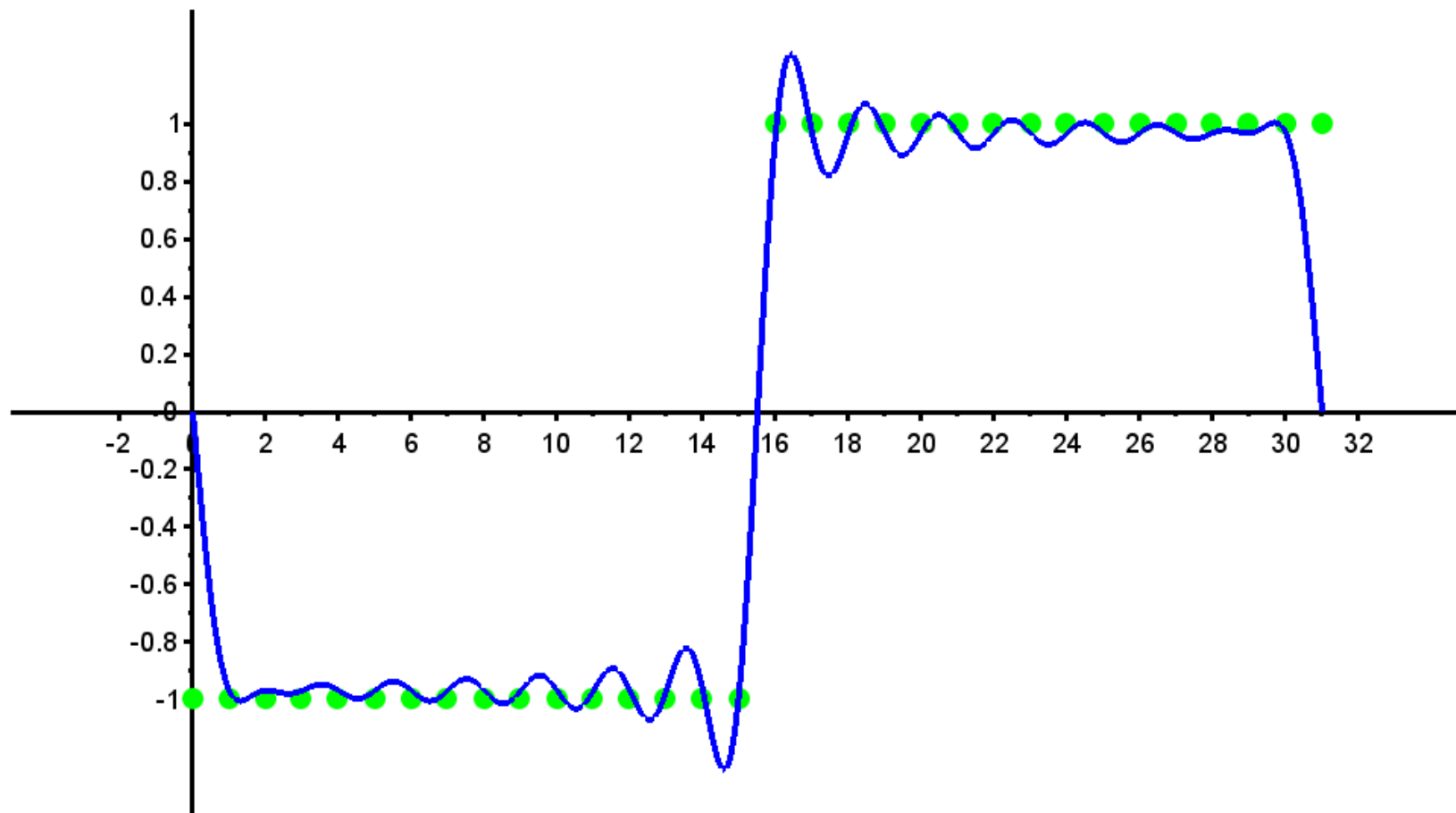
Na podstawie danych wejściowych wyznaczono współczynniki wielomianu: $a_i \approx 0$, dla $i = 0, 1, 2, \dots, 16$

Wielomian interpolacji trygonometrycznej przyjmuje postać:

$$\begin{aligned} F(x) \approx & -1.23 \sin(x) + 0.006 \sin(2x) - 0.41 \sin(3x) \\ & + 0.013 \sin(4x) - 0.24 \sin(5x) + 0.02 \sin(6x) \\ & - 0.17 \sin(7x) + 0.027 \sin(8x) - 0.13 \sin(9x) \\ & + 0.035 \sin(10x) - 0.1 \sin(11x) + 0.043 \sin(12x) \\ & - 0.081 \sin(13x) + 0.054 \sin(14x) - 0.066 \sin(15x) \end{aligned}$$

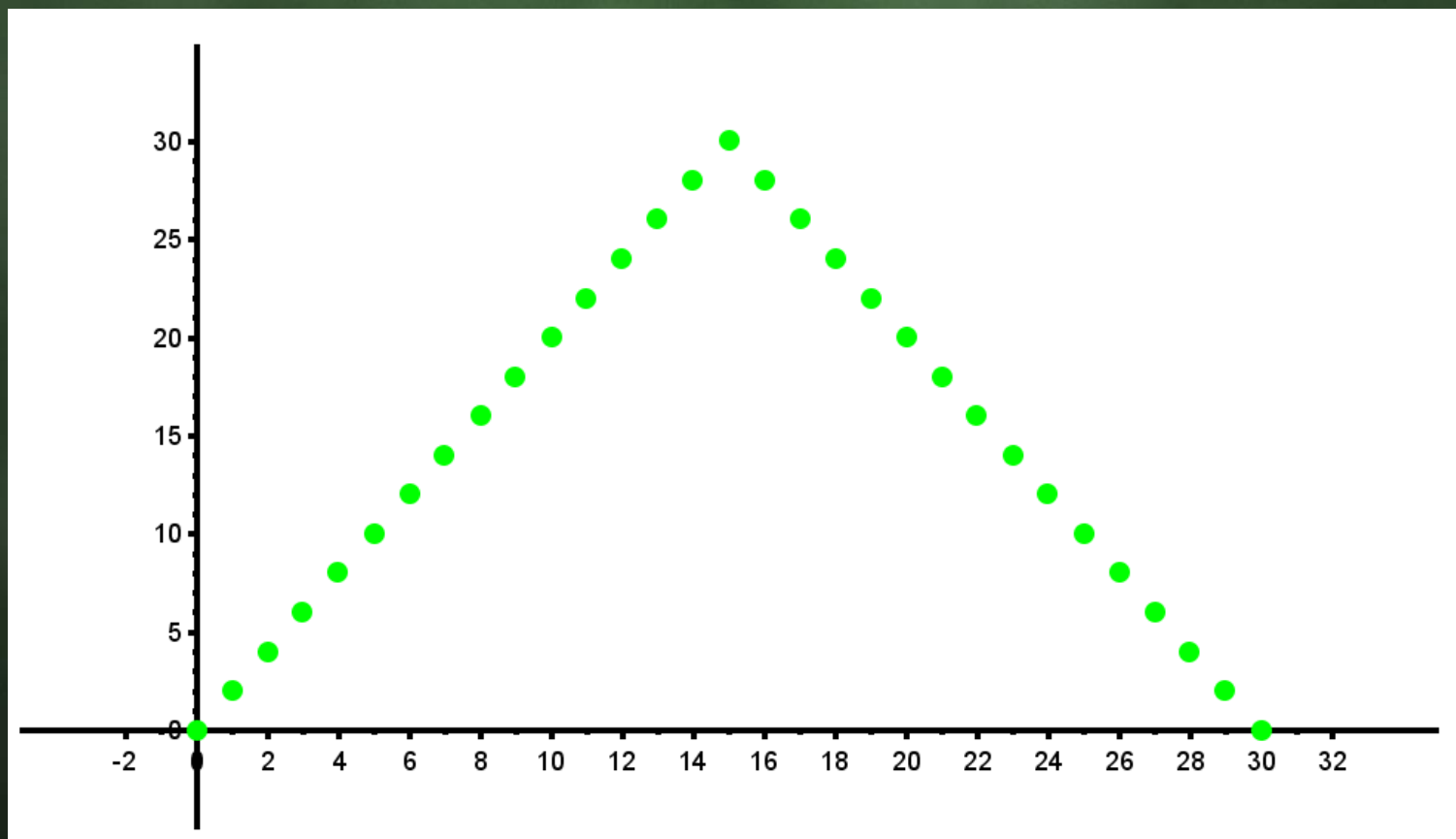
Przykład 4

11:26



Przykład 5

Wyznaczyć wielomian interpolacji trygonometrycznej zadanego sygnału:



Przykład 4

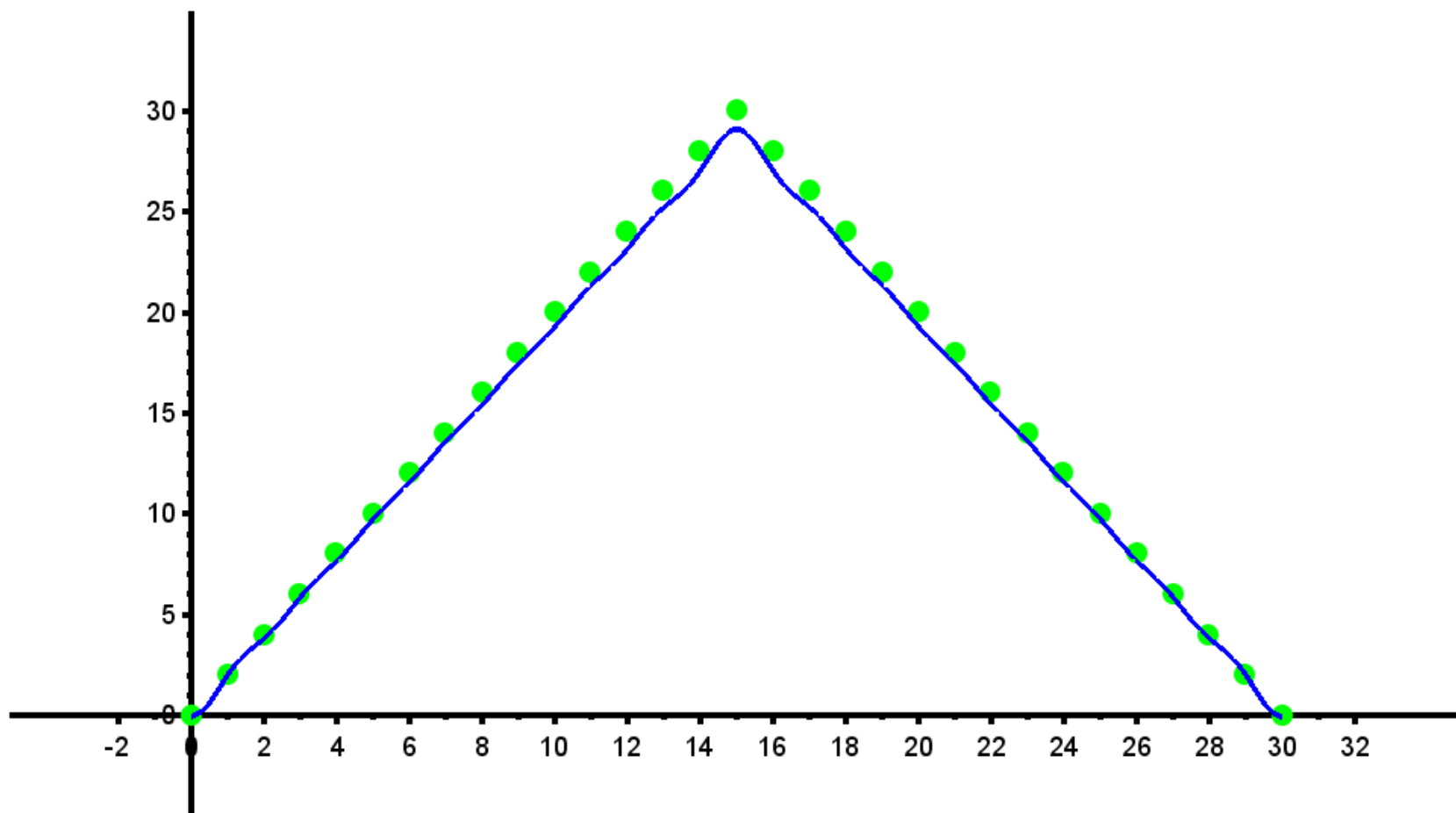
$$F_{31}(x) = \frac{a_0}{2} + \sum_{k=1}^{15} (a_k \cos(kx) + b_k \sin(kx))$$

Na podstawie danych wejściowych wyznaczono współczynniki wielomianu: $b_i \approx 0$, dla $i = 1, 2, \dots, 15$

Wielomian interpolacji trygonometrycznej przyjmuje postać:

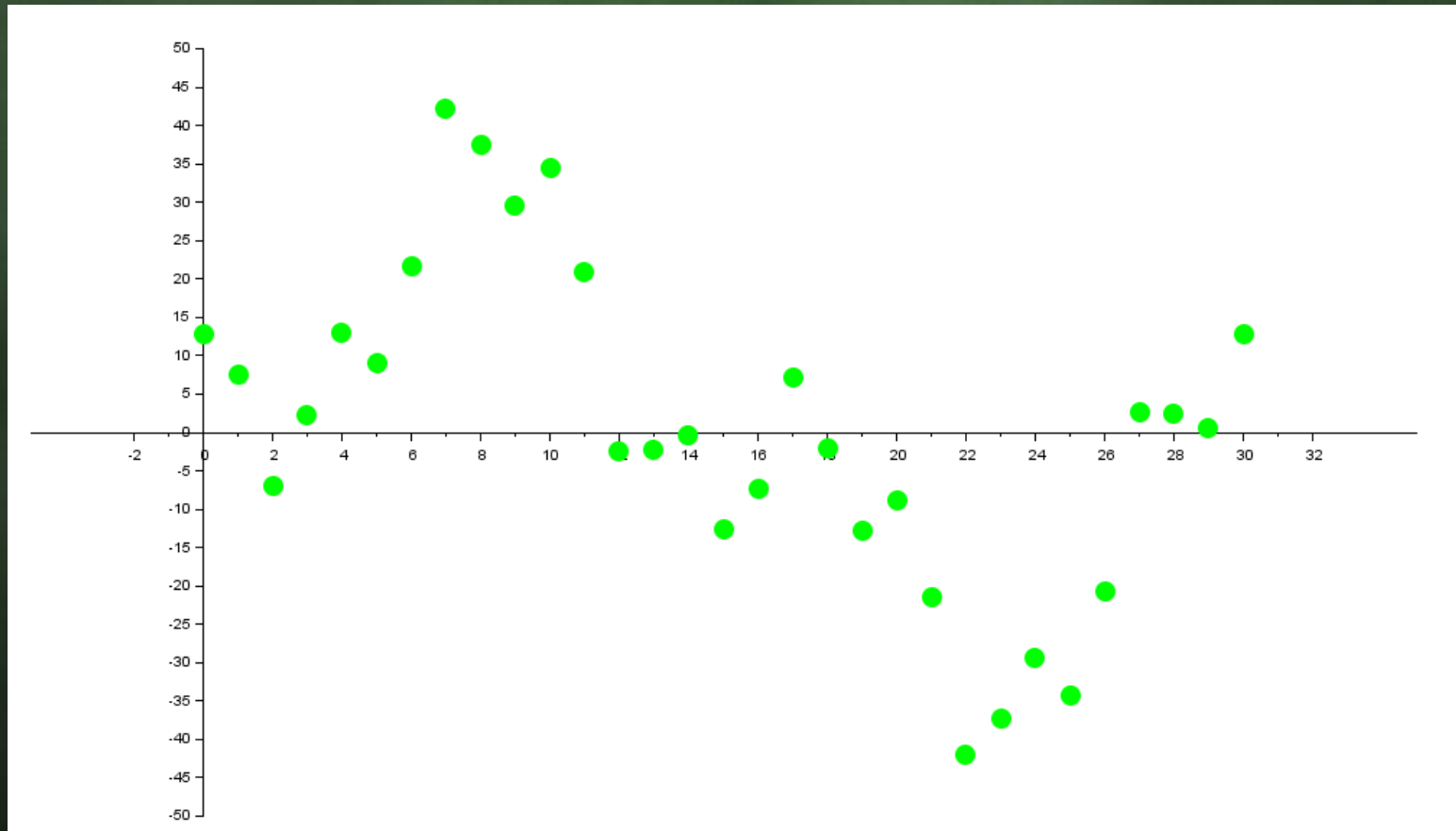
$$\begin{aligned} F(x) \approx & 14.5 - 11.8 \cos(x) - 1.35 \cos(3x) - 0.52 \cos(5x) \\ & - 0.29 \cos(7x) - 0.2 \cos(9x) - 0.15 \cos(11x) \\ & - 0.13 \cos(13x) - 0.13 \cos(15x) \end{aligned}$$

Przykład 5



Przykład 5

Wyznaczyć interpolacyjny wielomian trygonometryczny dla sygnału



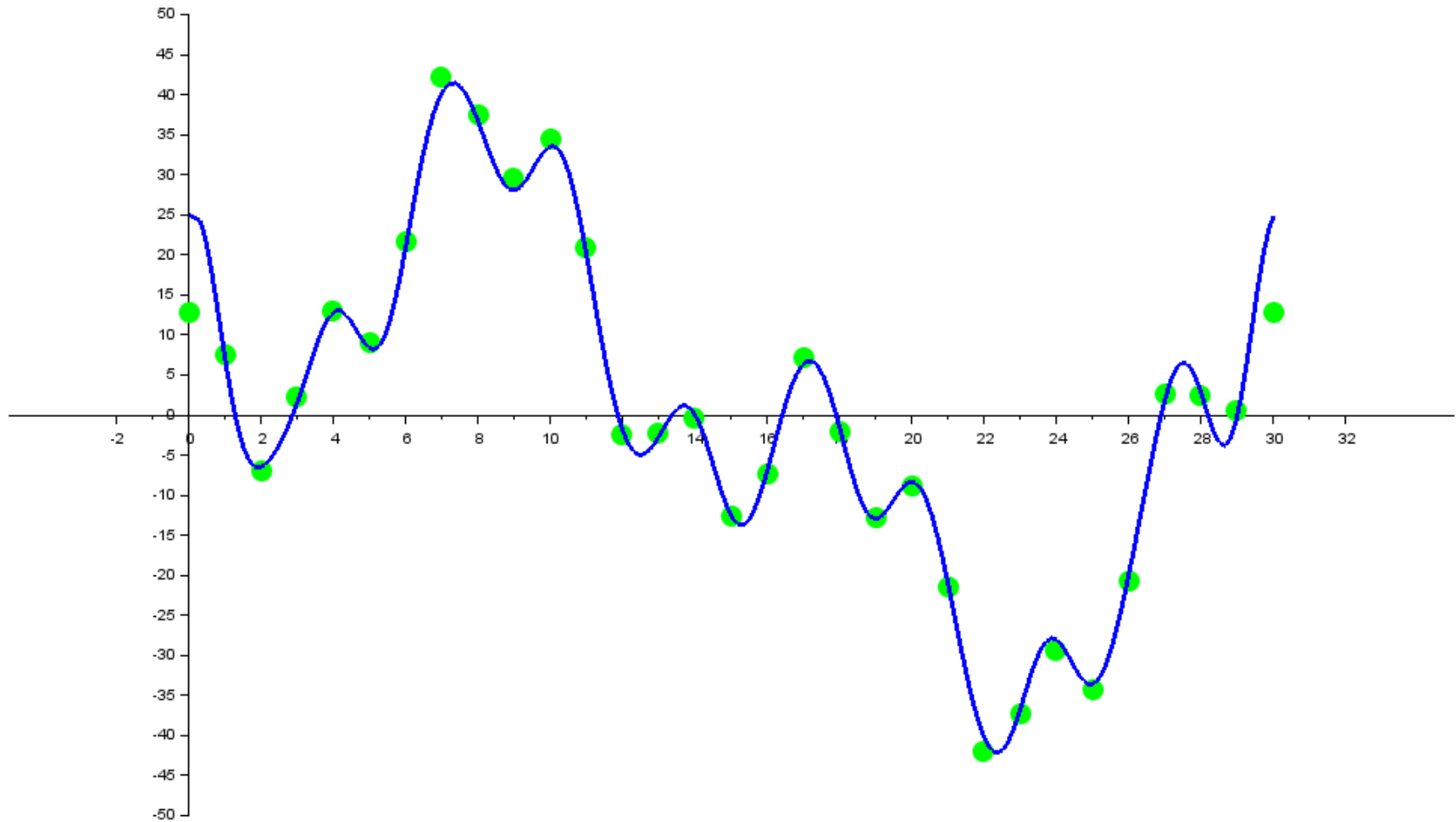
Przykład 5

$$F_{31}(x) = \frac{a_0}{2} + \sum_{k=1}^{15} (a_k \cos(kx) + b_k \sin(kx))$$

Na podstawie danych wejściowych wyznaczono współczynniki wielomianu. Wielomian interpolacji trygonometrycznej przyjmuje postać:

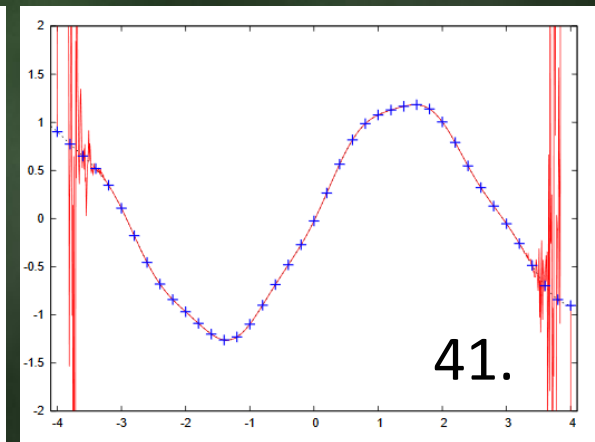
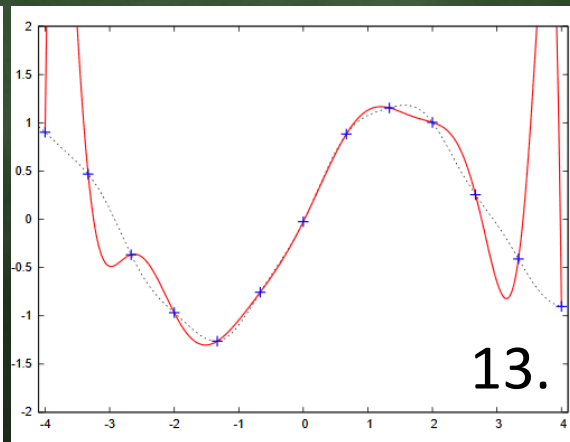
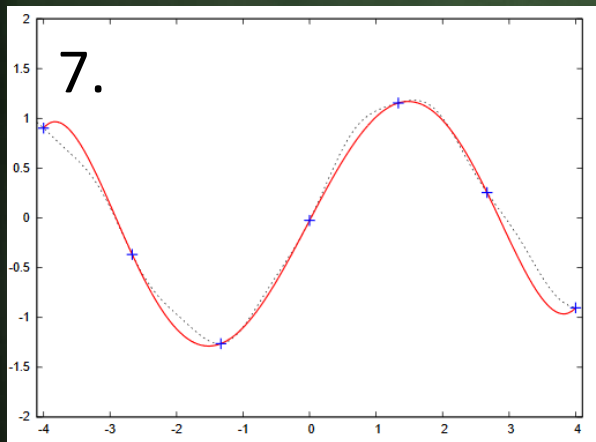
$$F(x) \approx 0.41 + 24 \sin(x) + 7.59 \cos(3x) - 11.7 \sin(3x) + 6.29 \cos(9x) + 5.47 \sin(9x)$$

Przykład 5



Interpolacje

- Interpolacje wielomianowe algebraiczne cechują się pogorszeniem jakości, szczególnie na końcach przedziału interpolacji.
- Zjawisko nasila się wraz ze wzrostem stopnia wielomianu, a więc liczby węzłów interpolacji.
- Zjawisko to nazywa się Oscylacjami Rungego



Interpolujące funkcje sklejane

- Problemy związane z interpolacjami wielomianowymi (oscylacje Rungego) powodują że wygodniejsze może być interpolowanie za pomocą funkcji sklejanych
- Funkcją sklejaną (splajnem – ang. spline) nazywamy funkcję która:
 - lokalnie jest wielomianem rzędu k
 - jest $(k-1)$ -krotnie różniczkowalna w węzłach (z czego wynika że jej pochodne rzędu $k-2$ i niższych są ciągłe).
- Najczęściej stosuje się funkcje sklejane 3 stopnia (funkcje kubiczne)

Interpolujące funkcje sklejane

Funkcja sklejana stopnia 0



Interpolujące funkcje sklejane

Funkcja sklejana stopnia 1



Interpolujące funkcje sklepane

- Interpolację funkcjami sklejanymi 3 stopnia można przeprowadzić dwoma sposobami:
 - Analizowany sygnał dzieli się na tyle podzbiorów, aby każdy z nich składał się z trzech węzłów interpolacyjnych
 - Określenie równania wielomianu 3 stopnia jest więc zagadnieniem łatwym
 - Metoda ta sprawia że mogą pojawić się na krańcach przedziałów „kanty”

Interpolujące funkcje sklepane

Zakładając, że znane są nie tylko wartości funkcji w węzłach interpolacyjnych, ale również wartości drugiej pochodnej w tych punktach można wyznaczyć interpolację funkcją sklepaną 3 stopnia bez efektu kantów.

Do dalszych rozważań przyjmujemy:

x_i	x_1	x_2	x_3	...	x_n
$f_i = f(x_i)$	f_1	f_2	f_3	...	f_n
$f_i'' = f''(x_i)$	f_1''	f_2''	f_3''	...	f_n''

Interpolujące funkcje sklepane

Dla każdego przedziału interpolacyjnego zdefiniowanego przez $[x_i; x_{i+1}]$ dla $i=1,2,\dots,n-1$ zapisujemy wielomian trzeciego stopnia:

$$y_i(x) = af_i + bf_{i+1} + cf'_i + df'_{i+1}$$

gdzie:

$$a = \frac{x_{i+1} - x}{x_{i+1} - x_i} \quad b = \frac{x - x_i}{x_{i+1} - x_i}$$

$$c = \frac{1}{6}(a^3 - a)(x_{i+1} - x_i)^2 \quad d = \frac{1}{6}(b^3 - b)(x_{i+1} - x_i)^2$$

Interpolujące funkcje sklepane

- Druga pochodna funkcji jest jednak nieznaną.
- Problem można rozwiązać zakładając ciągłość pierwszej pochodnej w węzłach
- Przyjmuje się, że pochodna lewym krańcu przedziału $y_i(x)$ jest równa pochodnej obliczanej dla prawego krańca przedziału $y_{i+1}(x)$

$$\frac{x_i - x_{i+1}}{6} f_{i-1}'' + \frac{x_{i+1} - x_{i-1}}{3} f_i'' + \frac{x_{i+1} - x_i}{6} f_{i+1}''$$

$$= \frac{f_{i+1} - f_i}{x_{i+1} - x_i} - \frac{f_i - f_{i-1}}{x_i - x_{i-1}}$$

Interpolujące funkcje sklepane

- W efekcie dany jest układ równań liniowych z niewiadomymi $\{f_i''\}$.
- Problem stanowi to, że dla n węzłów interpolacji, danych jest $n-2$ węzłów wewnętrznych i w efekcie można zapisać $n-2$ równań dla n niewiadomych
- Konieczne jest więc przyjęcie dodatkowych dwóch warunków
- Najczęściej zakłada się, że $f_1''=f_2''=0$, ale można narzucić inne warunki brzegowe

Interpolujące funkcje sklepane

Przy założeniu, że odstęp między węzłami interpolacyjnymi są równe $x_{i+1} - x_i = h$ można zapisać równanie w postaci

$$\begin{bmatrix} 4 & 1 & 0 & \dots & 0 & 0 \\ 1 & 4 & 1 & \dots & 0 & 0 \\ 0 & 1 & 4 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ 0 & 0 & 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 1 & 4 \end{bmatrix} \begin{bmatrix} f_2'' \\ f_3'' \\ f_4'' \\ \vdots \\ f_{n-2}'' \\ f_{n-1}'' \end{bmatrix} = \begin{bmatrix} 6 \cdot \frac{f_1 - 2f_2 + f_3}{h^2} \\ 6 \cdot \frac{f_2 - 2f_3 + f_4}{h^2} \\ 6 \cdot \frac{f_3 - 2f_4 + f_5}{h^2} \\ \vdots \\ 6 \cdot \frac{f_{n-3} - 2f_{n-2} + f_{n-1}}{h^2} \\ 6 \cdot \frac{f_{n-2} - 2f_{n-1} + f_n}{h^2} \end{bmatrix}$$

Interpolujące funkcje sklepane

Algorytm zastosowania interpolacji funkcjami sklejanymi 3 stopnia składa się z dwóch etapów

1. Rozwiązania układu równań w celu wyznaczenia nieznanych wartości drugich pochodnych sygnału w przedziale $i=[2;n-1]$
2. Korzystając z rodziny wielomianów $y_i(x)$ określonych dla każdego przedziału $[x_i; x_{i+1}]$, w zadanym przedziale można wyznaczyć dowolną liczbę punktów interpolujących

Interpolujące funkcje sklejane

```
1  clc;clear;  
2  x=[0;1;2;3;4;5;6;7;8;9;10];  
3  f=[7;5.5;4;5.7;7;6.6;5.6;3;5;5.5;6];  
4  h=x(2)-x(1);  
5  n=size(x);  
6  n=n(1)  
7  f2=zeros(1,n);  
8  Z=zeros(n-2,n-2);
```

Interpolujące funkcje sklejane

```
9  Z=Z+4*eye(n-2,n-2);
10 for i=1:n-2
11     for j=1:n-2
12         if (j==i+1) Z(i,j)=1; end;
13         if (i==j+1) Z(i,j)=1; end;
14     end
15 end
16 Y=zeros(1,n-2);
17 for i=1:n-2
18     Y(i)=(6/h^2)*(f(i)-2*f(i+1)+f(i+2))
19 end
20 F2=linsolve(Z,-Y');
```


Interpolujące funkcje sklejane

```
21 for i=2:n-1
22     f2(i)=F2(i-1)
23 end
24 m=500;
25 M=m*(n-1);
26 t=linspace(x(1),x(n),M)
27 ti=zeros(1,M);
28 y=zeros(1,M);
29 j=1;
30 for i=1:M
31     if ((t(i)>=x(j)) & (t(i)<=x(j+1))) then
32         ti(i)=j;
33     else
34         j=j+1;
35         ti(i)=j;
36     end
37 end
```

Interpolujące funkcje sklepane

```

38 A=zeros(1,M);B=zeros(1,M);C=zeros(1,M);D=zeros(1,M);
39 for i=1:M
40     A(i)=(x(ti(i)+1)-t(i))/(x(ti(i)+1)-x(ti(i)));
41     B(i)=(t(i)-x(ti(i)))/(x(ti(i)+1)-x(ti(i)));
42     C(i)=(1/6)*(A(i)^3-A(i))*(x(ti(i)+1)-x(ti(i)))^2;
43     D(i)=(1/6)*(B(i)^3-B(i))*(x(ti(i)+1)-x(ti(i)))^2;
44     y(i)=A(i)*f(ti(i))+B(i)*f(ti(i)+1)+C(i)*f2(ti(i))+D(i)*f2(ti(i)+1)
45 end

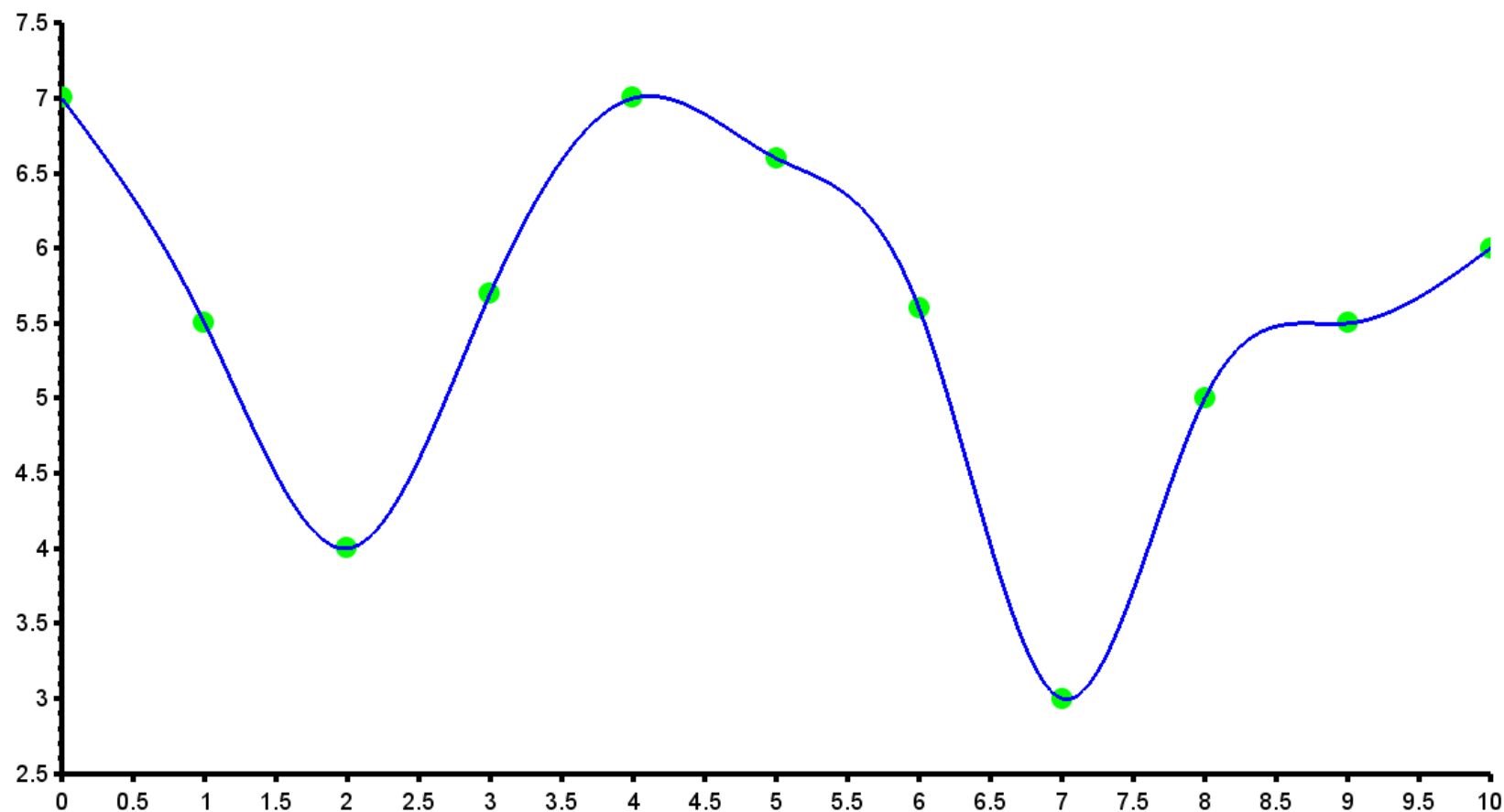
```

$$y_i(x) = af_i + bf_{i+1} + cf''_i + df''_{i+1}$$

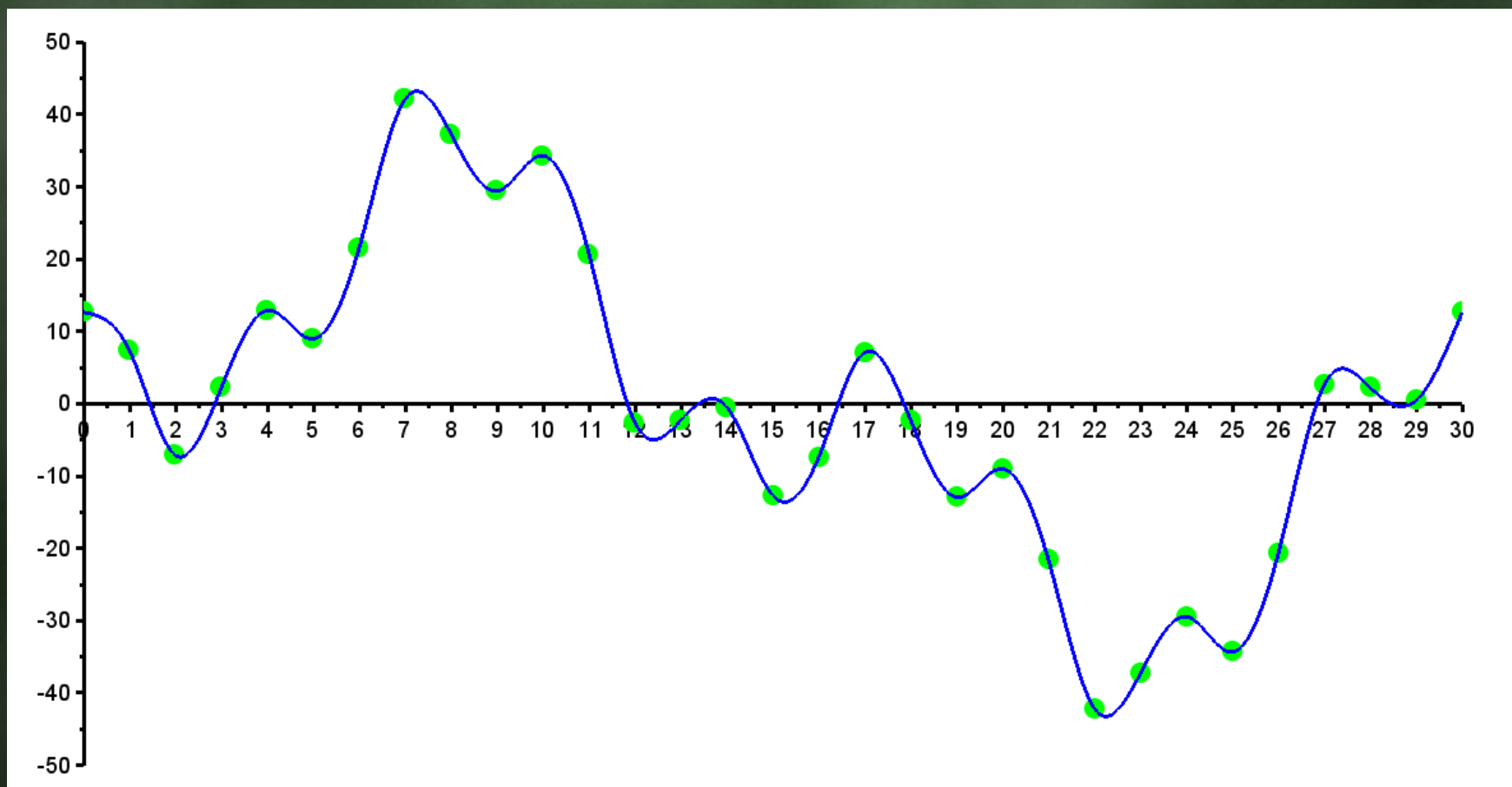
$$a = \frac{x_{i+1} - x}{x_{i+1} - x_i} \quad b = \frac{x - x_i}{x_{i+1} - x_i}$$

$$c = \frac{1}{6}(a^3 - a)(x_{i+1} - x_i)^2 \quad d = \frac{1}{6}(b^3 - b)(x_{i+1} - x_i)^2$$

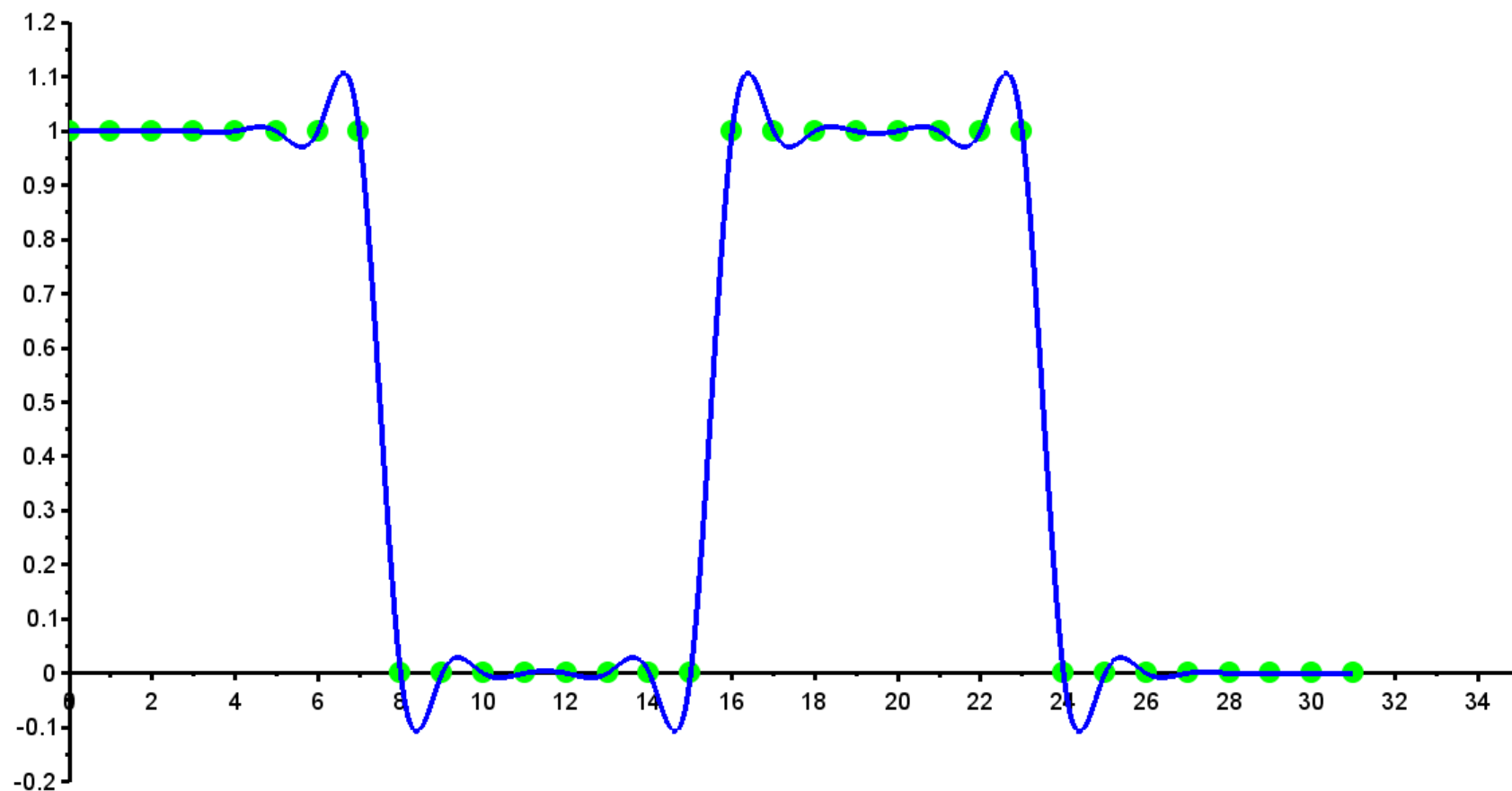
Przykład 6



Przykład 7



Przykład 7



Szybka transformata Fouriera

- Przekształcenie Fouriera (transformacja) pozwala na rozkład analizowanego sygnału na częstotliwości składowe.
- Algorytm szybkiej transformaty Fouriera FFT służy do efektywnego wyznaczania współczynnika c_m dla interpolacji trygonometrycznej

$$F_n(x) = \sum_{m=0}^n c_m e^{j \cdot m \cdot x}$$

Szybka transformata Fouriera

Dekompozycja sygnału na składowe w postaci szeregu sygnałów sinusoidalnych umożliwia interpolację sygnału w postaci szeregu trygonometrycznego poprzez identyfikację parametrów tych sygnałów:

- częstotliwości,
- amplitudy,
- fazy.

FFT znajduje szerokie zastosowanie w wielu dziedzinach: przetwarzaniu sygnałów cyfrowych, obrazowaniu medycznym, rozwiązywaniu równań różniczkowych, poszukiwaniu złóż ropy i gazu, Monitorowaniu pracy maszyn, sterowaniu samolotów i statków kosmicznych.

Szybka transformata Fouriera

- Funkcja FFT jest dostępna w większości programów obliczeniowych.
- W środowisku Scilaba szybką transformatę Fouriera można przeprowadzać dla wektora lub macierzy danych.
- Podstawowa składnia funkcji fft wymaga podania jedynie na wejście analizowanego sygnału, na wyjściu otrzymujemy odpowiednio wektor lub macierz wartości zespolonych

$X = \text{fft}(A) ;$

Przykład 8 - FFT

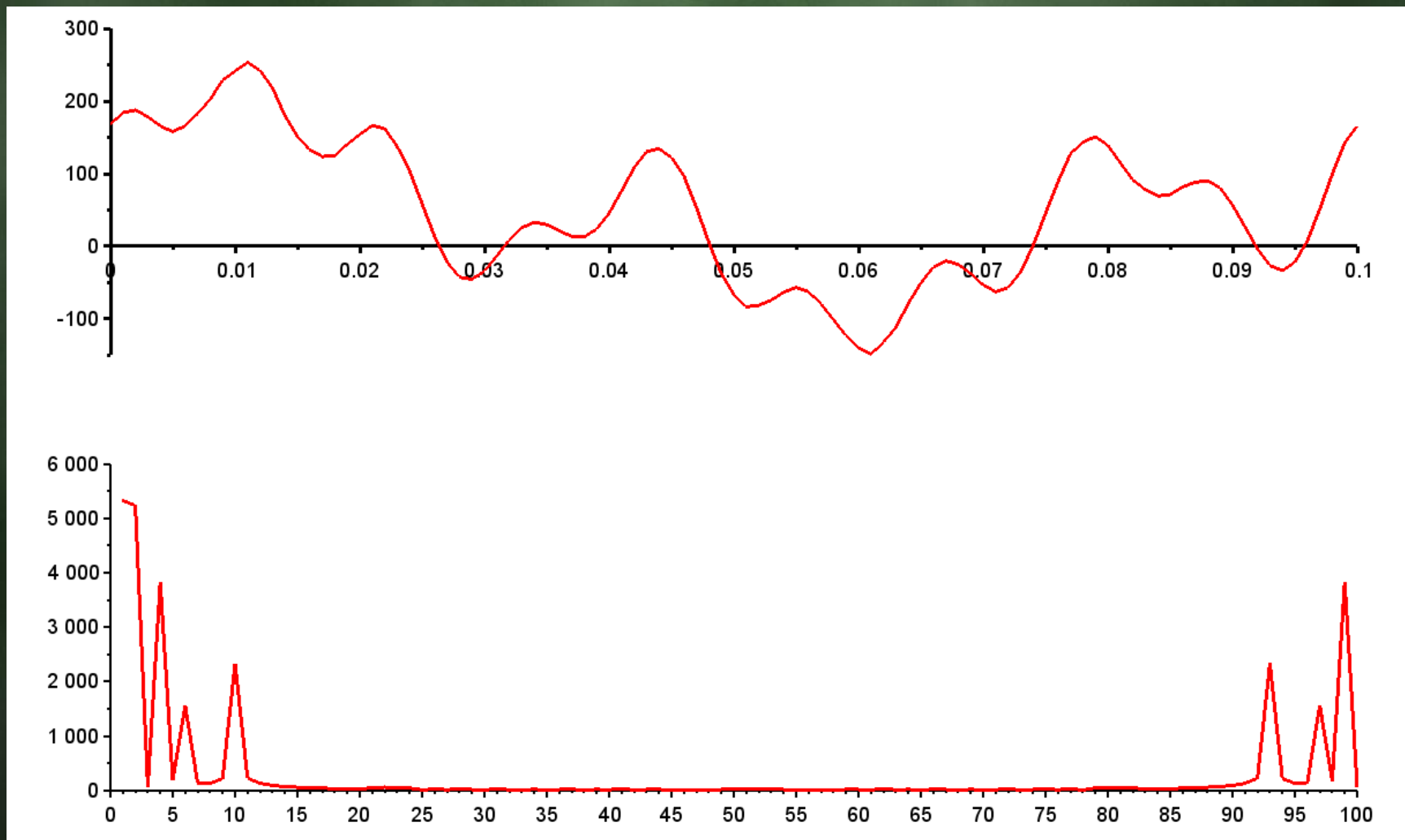
- Przeprowadzić analizę sygnału, określić parametry interpolacyjnego wielomianu trygonometrycznego (amplitudę, częstotliwość i fazę).
- Dla potrzeb przykładu analizowany przykład zawiera losowo generowany szum – odpowiadająco błędom pomiarowym

Przykład 8 - FFT

Dane wejściowe dla przykładu

```
1 clc;clear;
2 f=10;//częstotliwość podstawowa
3 n=300;//liczba próbek
4 w=2*%pi*f;
5 O=1;
6 T=1/f;//okres
7 Um=[100;75;30;45];
8 H=[1;3;5;9];
9 fi=[60;-30;45;90];
10 delta=T/n;
11 df=1/delta//częstotliwość próbkowania
12 for i=1:(n)*O;
13     t(i)=(i-1)*T/n;
14     u(i)=Um(1)/2;
15     for j=1:4
16         u(i)=u(i)+Um(j)*sin(H(j)*w*t(i)+%pi/180*fi(j));
17     end;
18     u(i)=u(i)*(1+0.05*rand());//błędy pomiarowe
19 end
```

Przykład 8 - FFT



Przykład 8 - FFT

- Poprawna analiza wymaga posiadania informacji o liczbie okresów analizowanego sygnału w próbce wejściowej
- Im większa częstotliwość próbkowania tym dokładniejsze wyniki obliczeń, ostrzejsze piki
- Liczba okresów powinna być liczbą całkowitą
- Im więcej okresów lub próbek tym amplituda większa
- Zwracany wektor **fft** ma symetryczną budowę względem środka zakresu

Przykład 8 - FFT



Przykład 8 - FFT

- Każda wartość zwracana przez funkcję `fft` jest w postaci liczby zespolonej, niosącej informacje o amplitudzie i fazie analizowanego sygnału
- Każda wartość jest powiązana z inną częstotliwością składowej sygnału wejściowego
- Informacja o częstotliwości próbkowania pozwala na wygenerowanie wektora częstotliwości
- Dodatkowo konieczna jest informacja o liczbie próbek

Przykład 8 FFT

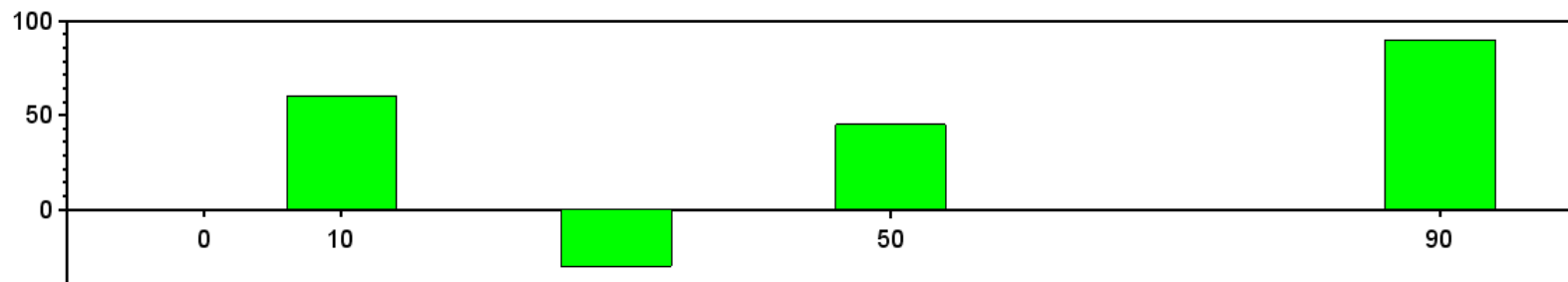
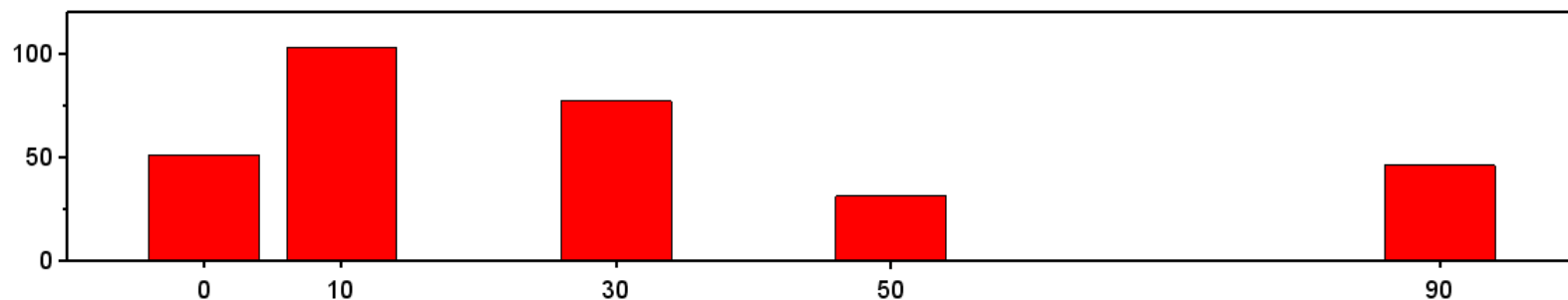
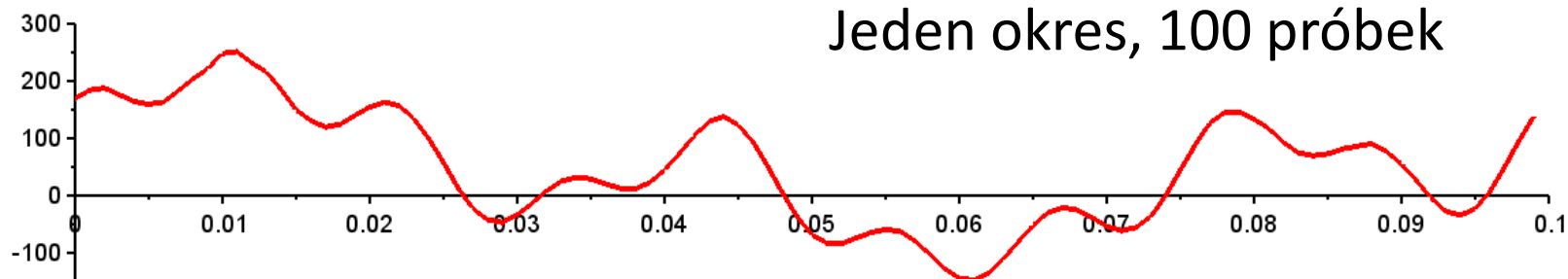
- Konieczna jest identyfikacja ekstremów które odpowiadają poszukiwanym składowym wielomianu interpolacyjnego
- Ze względu na symetrie ograniczyć się można do połowy wartości wektora zwróconego przez FFT
- Odfiltrowanie szumu można zrealizować poprzez wyszukanie wartości wektora FFT większych od średniej

Przykład 8 - FFT

- Amplituda składowych wielomianu uzyskana z funkcji FFT jest wielokrotnością iloczynu liczby próbek i okresów sygnału i liczby ekstremów związanych z częstotliwością składowej
- Częstotliwości określa poprzez iloraz częstotliwości próbkowania i iloczynu liczby próbek i okresów
- Kąty fazowe poszczególnych składowych wielomianu wyznacza się jako argument z liczby zespolonej uzyskanej z funkcji FFT dla każdej z wyznaczonych częstotliwości

Przykład 8 FFT

Jeden okres, 100 próbek



Przykład 8 FFT

Częstotliwości wyznaczonych składowych wielomianu [Hz]

0. 10. 30. 50. 90.

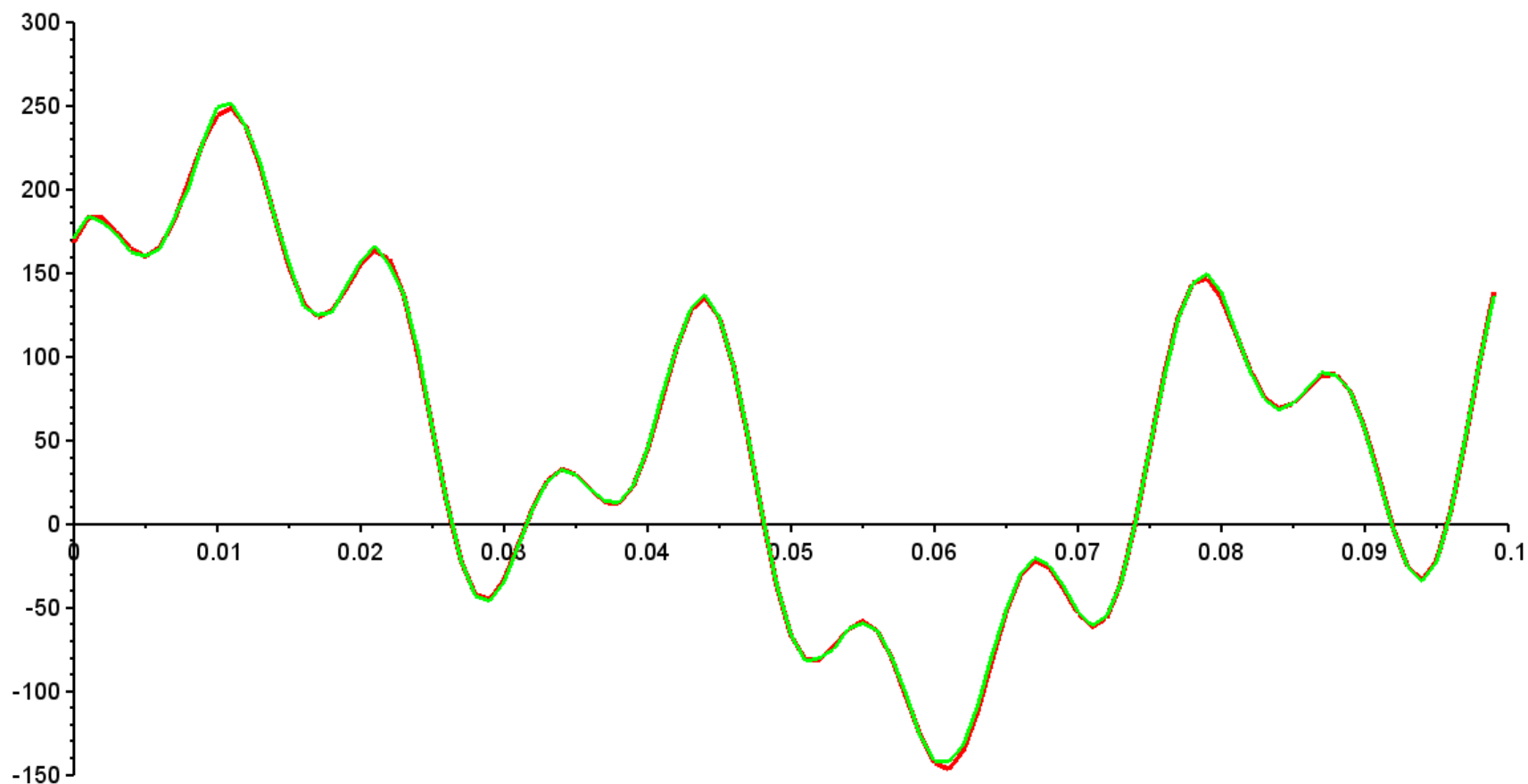
Amplituda wyznaczonych składowych wielomianu [V]

51. 102. 77. 31. 46.

Fazy wyznaczonych składowych wielomianu

0. 60. -30. 45. 90.

Przykład 8 - FFT



Przykład 9 -FFT

Częstotliwości wyznaczonych składowych wielomianu [Hz]

column 1 to 13

0. 7. 13. 20. 27. 33. 40. 47. 53. 60. 87. 93. 100.

column 14 to 15

107. 113.

Amplituda wyznaczonych składowych wielomianu [V]

column 1 to 13

68. 69. 68. 33. 59. 41. 8. 15. 23. 8. 25. 33. 13.

column 14 to 15

9. 7.

Fazy wyznaczonych składowych wielomianu

column 1 to 12

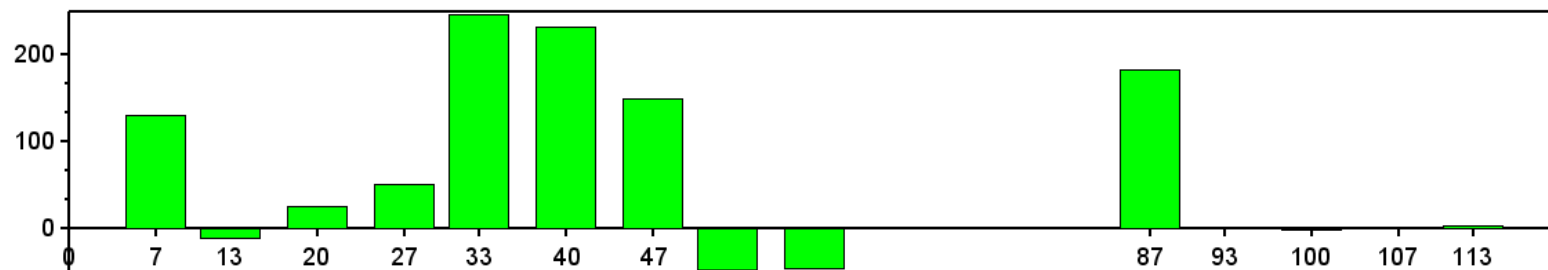
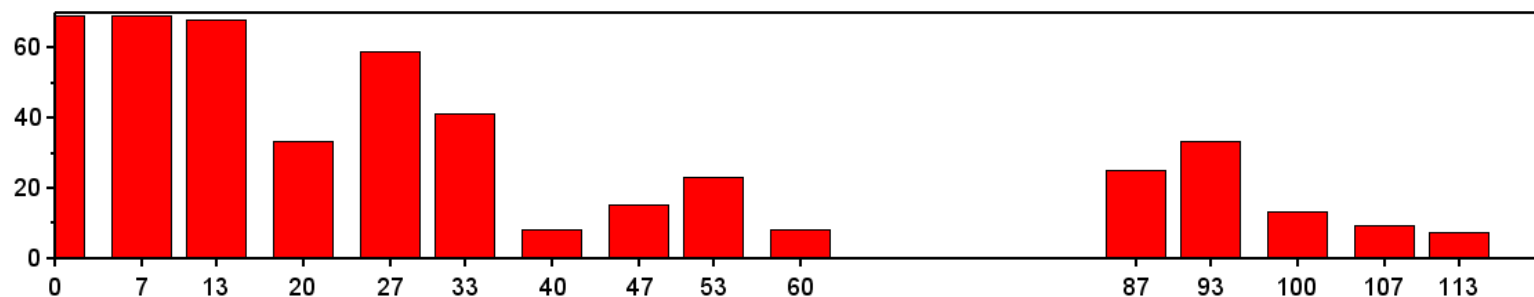
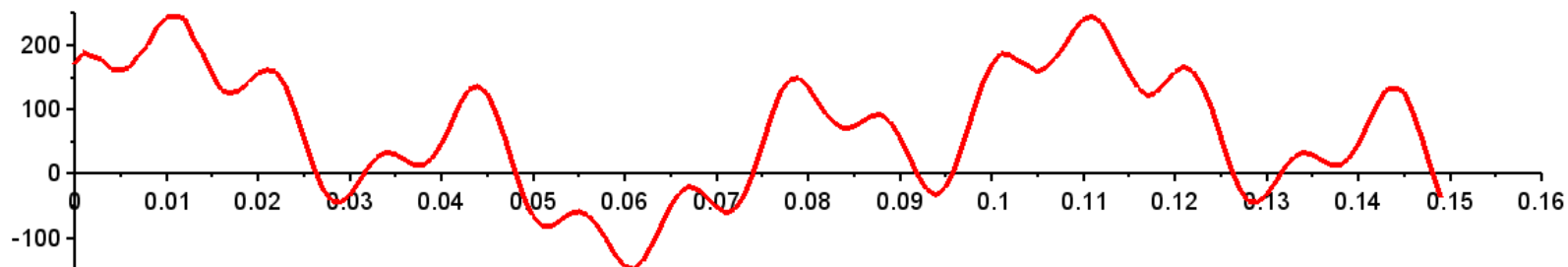
0. 130. -11. 24. 51. 246. 237. 148. -49. -52. 183. -1.

column 13 to 15

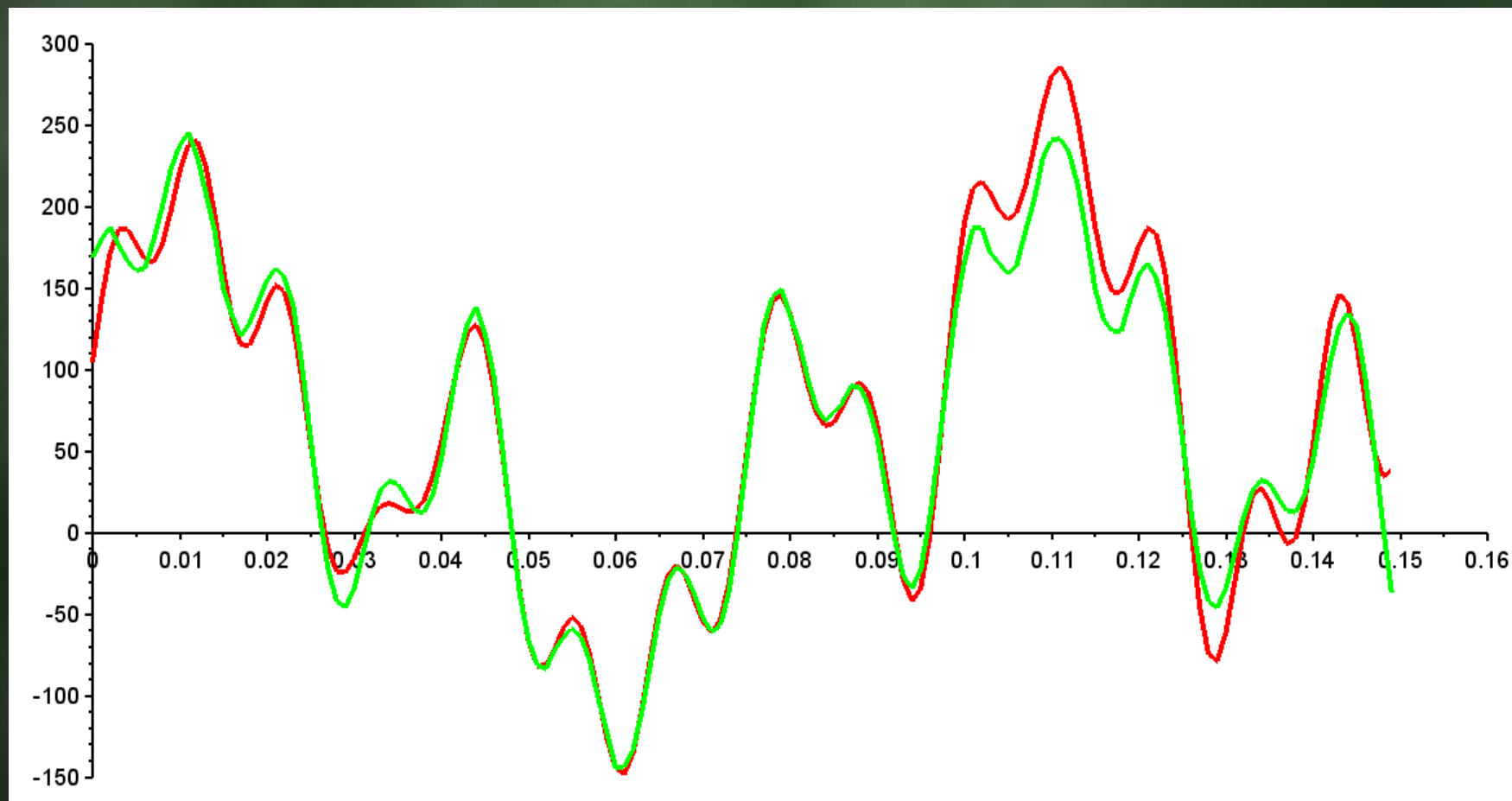
-2. 1. 1.

1.5 okresu, 100 próbek

Przykład 9 FFT



Przykład 9 FFT



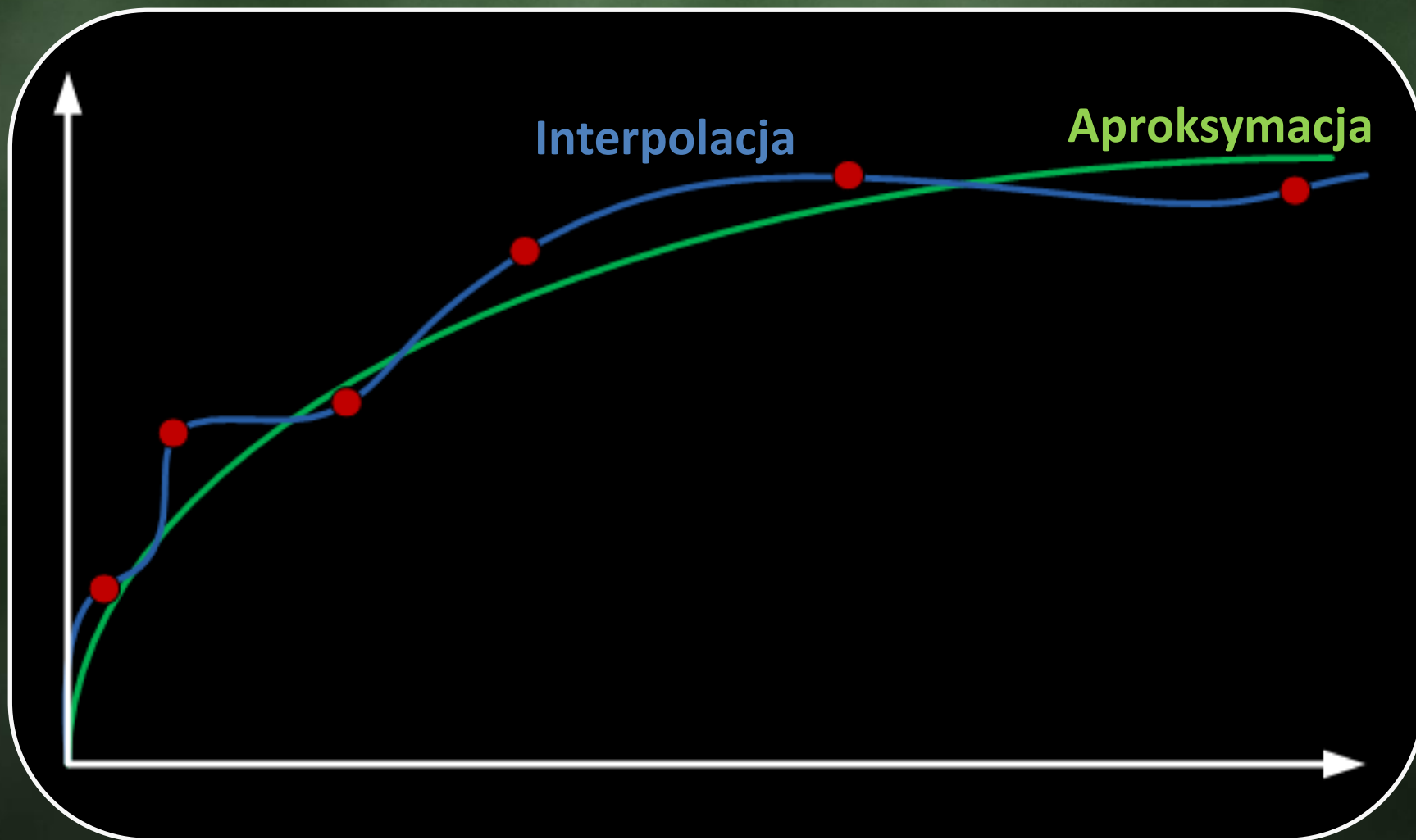
Przykład 10 - FFT

- Zwiększenie częstotliwości próbkowania przy całkowitej liczbie okresów sygnału badanego w próbce wpływa nieznacznie na poprawę dokładności interpolacji
- Zbyt duża liczba próbek może doprowadzić do błędów obliczeniowych
- Jeżeli liczba okresów nie jest całkowita, zwiększenie liczby próbek drastycznie zwiększa liczbę wyznaczonych składowych wielomianu interpolującego, nie poprawiając jednocześnie dokładności interpolacji

Aproksymacja

- Aproksymacja polega na wyznaczeniu, możliwie niskim kosztem, funkcji F przybliżającej zadaną funkcję f z żadaną dokładnością
- W odróżnieniu od interpolacji funkcja aproksymująca nie musi przechodzić przez zadane punkty funkcji aproksymowanej.
- Funkcja aproksymująca $f(x)$ ma przybliżać funkcję aproksymowaną $F(x)$ z żadaną dokładnością.

Aproksymacja



Aproksymacja

Aproksymacja funkcji $f(x)$ polega na wyznaczeniu współczynników wielomianu $F(x)$

$$F(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \cdots + a_n\varphi_n(x)$$

przy warunku osiągnięcia minimalnej wartości:

$$\|f(x) - F(x)\|$$

$\varphi_0, \varphi_1, \dots, \varphi_n$ są funkcjami bazowymi, których właściwy dobór ma decydujące znaczenie dla prawidłowości i efektywności przeprowadzenia aproksymacji

Aproksymacja

Zadanie aproksymacyjne przy wybranych funkcjach bazowych $\varphi_k(x)$ sprowadza się do wyznaczenia współczynników a_k , zapewniających osiągnięcie minimum normy:

$$\|f(x) - F(x)\|$$

Najczęściej stosowane normy to:

1. Norma jednostajna (Czebyszewa)
2. Normą średniokwadratową
3. Normą L_2

Wybór normy determinuje rodzaj aproksymacji

Aproksymacja

Funkcje bazowe $\varphi(x)$ wybierane są zazwyczaj z podprzestrzenie:

a) Funkcji trygonometrycznych z bazą:

$1, \sin(x), \cos(x), \sin(2x), \cos(2x), \dots, \sin(nx), \cos(nx)$

Dla sygnałów okresowych

b) Wielomianów co najwyżej stopnia n :

$1, x, x^2, x^3, \dots, x^n$

Pomimo prostoty działań na wielomianach, cechują się dużymi błędami przy działaniach na małych lub niewiele różniących się liczbach

c) Wielomianów Czebyszewa

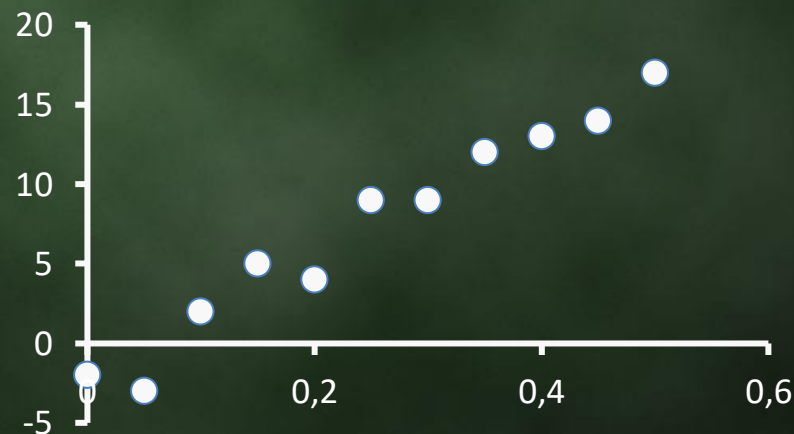
d) Wielomianów Legendre'a

Aproksymacja

- Na podstawie analizy danych pomiarowych można określić charakter – kształt funkcji wiążącej punkty pomiarowe.
- Aproksymacja średniokwadratowa znacznie lepiej „eliminuje” błędy pomiarowe od aproksymacji jednostajnej

| x | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|----|----|---|---|---|---|---|----|----|----|----|
| y | -2 | -3 | 2 | 5 | 4 | 9 | 9 | 12 | 13 | 14 | 17 |

$$y = a_1 x + a_0$$



Aproksymacja

- Zadanie sprowadza się do określenia współczynników a_0 i a_1 takich aby spełnione było najmniejsze maksimum między $F(x)$ i $f(x)$.
- Aproksymacji średniokwadratowej

$$S(a_0, a_1) = \sum_{i=1}^n (y_i - (a_1 x_i + a_0))^2$$

Aproksymacja

- Określenie wartości współczynników a_0 i a_1 spełniających warunek osiągnięcia minimum funkcji sprowadza się do rozwiązania równania dla zerowania pochodnych równania.
- Zakładając aproksymację liniową opisaną równaniem $y=a_1x+a_0$ mówi się o regresji liniowej.

Aproksymacja

Poszukując minimum funkcji należy rozwiązać układ równań

$$\frac{\partial S(a_0, a_1)}{\partial a_0} = 2 \sum_{i=1}^n (y_i - a_0 - a_1 x_i)(-1) = 0$$

$$\frac{\partial S(a_0, a_1)}{\partial a_1} = 2 \sum_{i=1}^n (y_i - a_0 - a_1 x_i)(-x_i) = 0$$

Aproksymacja

Przekształcając wzory można równanie zapisać w postaci:

$$a_0 n + a_1 \sum_{i=1}^n x_i = \sum_{i=1}^n y_i$$

$$a_0 \sum_{i=1}^n x_i + a_1 \sum_{i=1}^n x_i^2 = \sum_{i=1}^n x_i y_i$$

Aproksymacja

W zapisie macierzowym układ równań przyjmuje postać:

$$2 \begin{bmatrix} n & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_i y_i \end{bmatrix}$$

Aproksymacja

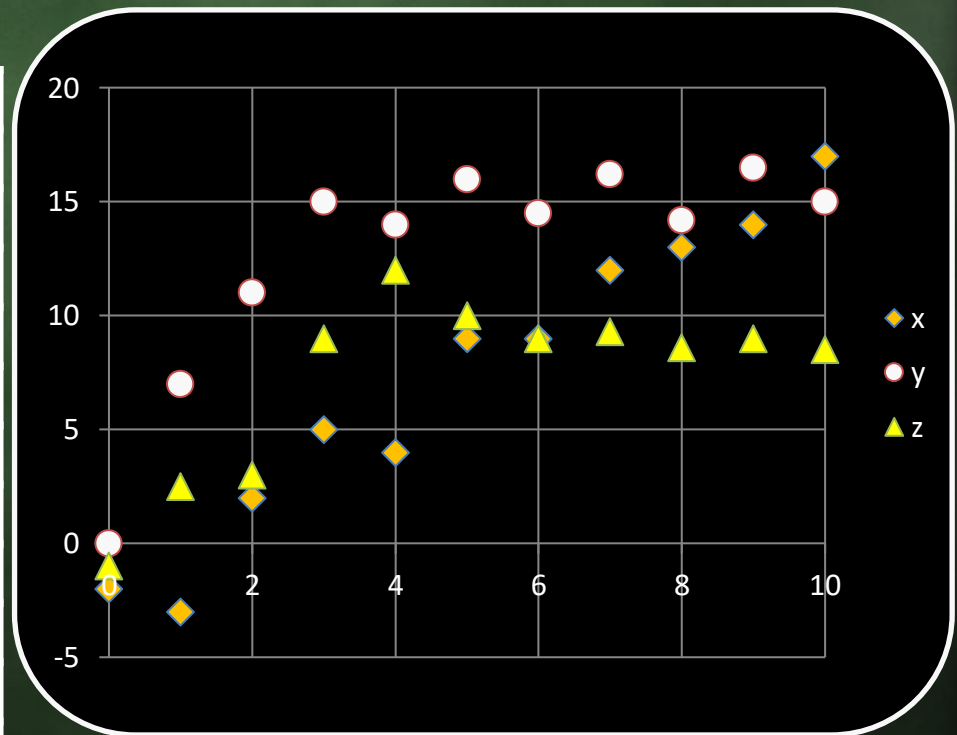
Ogólną postać dla wielomianu rzędu k można zapisać w postaci równania macierzowego:

$$(k+1) \begin{bmatrix} n & \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^1 & \dots & \sum_{i=1}^n x_i^k \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i^3 & \dots & \sum_{i=1}^n x_i^{k+1} \\ \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i^3 & \sum_{i=1}^n x_i^4 & \dots & \sum_{i=1}^n x_i^{k+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^n x_i^k & \sum_{i=1}^n x_i^{k+1} & \sum_{i=1}^n x_i^{k+2} & \dots & \sum_{i=1}^n x_i^{2k} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_k \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n x_i^2 y_i \\ \vdots \\ \sum_{i=1}^n x_i^k y_i \end{bmatrix}$$

Przykład 11

Dla zadanych sygnałów pomiarowych przeprowadzić aproksymację wielomianową:

| t | x | y | z |
|----|----|-------|-------|
| 0 | -2 | 0,00 | -1,00 |
| 1 | -3 | 7,00 | 2,50 |
| 2 | 2 | 11,00 | 3,00 |
| 3 | 5 | 15,00 | 9,00 |
| 4 | 4 | 14,00 | 12,00 |
| 5 | 9 | 16,00 | 10,00 |
| 6 | 9 | 14,50 | 9,00 |
| 7 | 12 | 16,20 | 9,30 |
| 8 | 13 | 14,20 | 8,60 |
| 9 | 14 | 16,50 | 9,00 |
| 10 | 17 | 15,00 | 8,50 |



Przykład 11

```
1 clc;clear;
2 x=[0;1;2;3;4;5;6;7;8;9;10];
3 y=[-2;-3;2;5;4;9;9;12;13;14;17];
4 //y=[0;7;11;15;14;16;14.5;16.2;14.2;16.5;15];
5 //y=[-1;2.5;3;9;12;10;9;9.3;8.6;9;8.5];
6 n=size(x) (1);
7 k=1;
8 yi=zeros(1,n); A=zeros(k+1,k+1); B=zeros(1,k+1);
```

- Każde zadanie aproksymacyjne rozwiązywane niezależnie.
- Przy aproksymacji nieliniowej dobierana jest potęga wykładnika

Przykład 11

Wyznaczanie współczynników wielomianu aproksymacyjnego

```

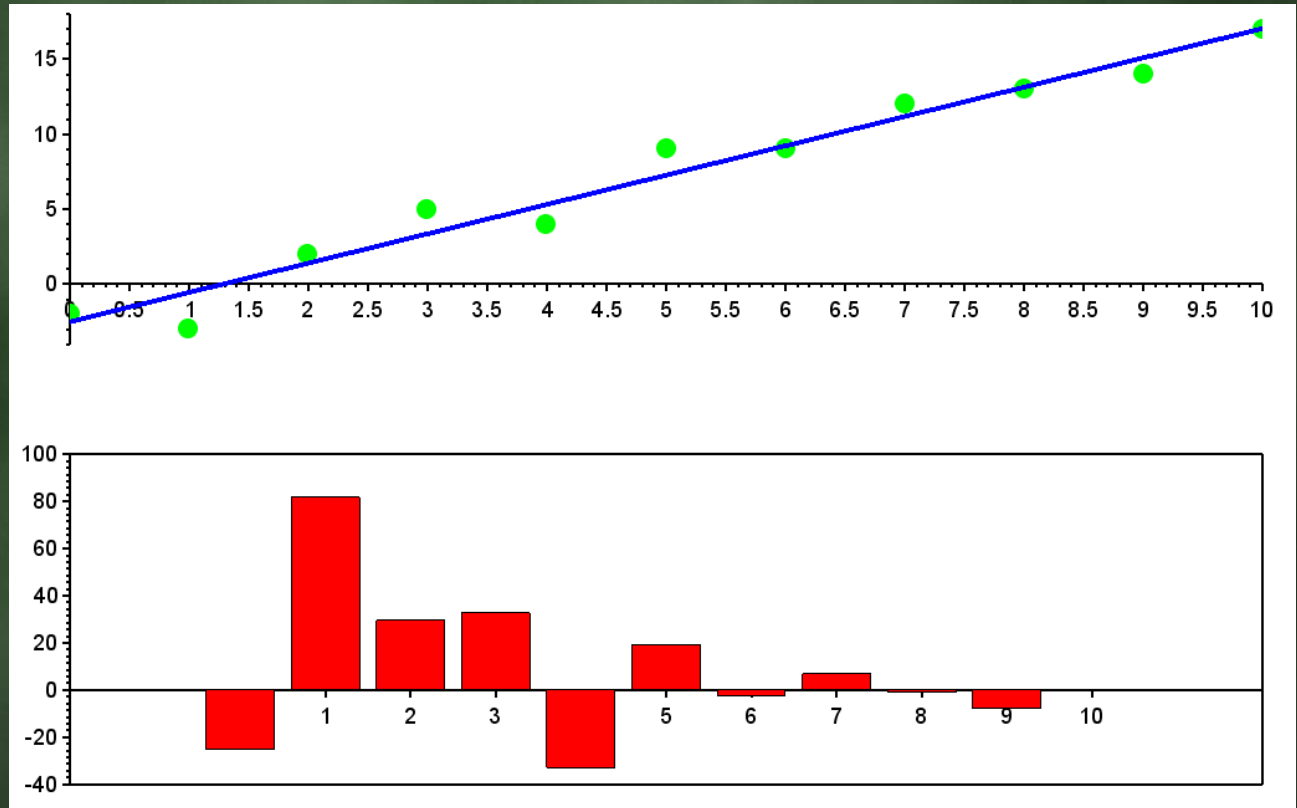
12 for l=1:k+1
13     for m=1:k+1
14         for i=1:n
15             A(l,m)=A(l,m)+x(i)^(m-1+l-1);
16         end
17         for i=1:n
18             B(m)=B(m)+y(i)*x(i)^(m-1);
19         end
20     end
21 end
22 A=(k+1)*A;
23 a=linsolve(A,-B');

24 for i=1:n
25     for m=1:k+1
26         yi(i)=yi(i)+a(m)*x(i)^(m-1);
27     end
28 end

```

Przykład 11

| t | x |
|----|----|
| 0 | -2 |
| 1 | -3 |
| 2 | 2 |
| 3 | 5 |
| 4 | 4 |
| 5 | 9 |
| 6 | 9 |
| 7 | 12 |
| 8 | 13 |
| 9 | 14 |
| 10 | 17 |

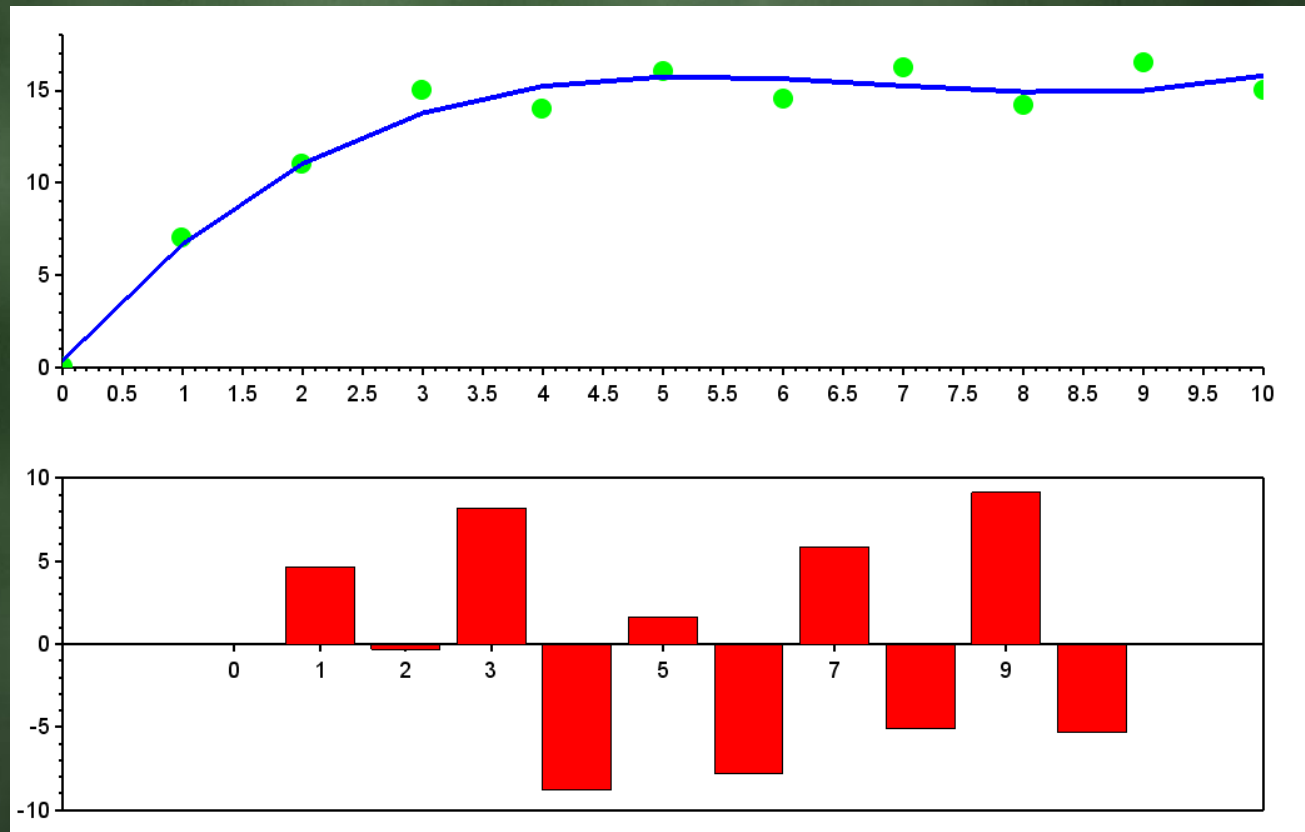


$$a_1 = 1.9545455 \quad a_0 = -2.5$$

$$x = 1.9545455t - 2.5$$

Przykład 11

| t | y |
|----|-------|
| 0 | 0,00 |
| 1 | 7,00 |
| 2 | 11,00 |
| 3 | 15,00 |
| 4 | 14,00 |
| 5 | 16,00 |
| 6 | 14,50 |
| 7 | 16,20 |
| 8 | 14,20 |
| 9 | 16,50 |
| 10 | 15,00 |

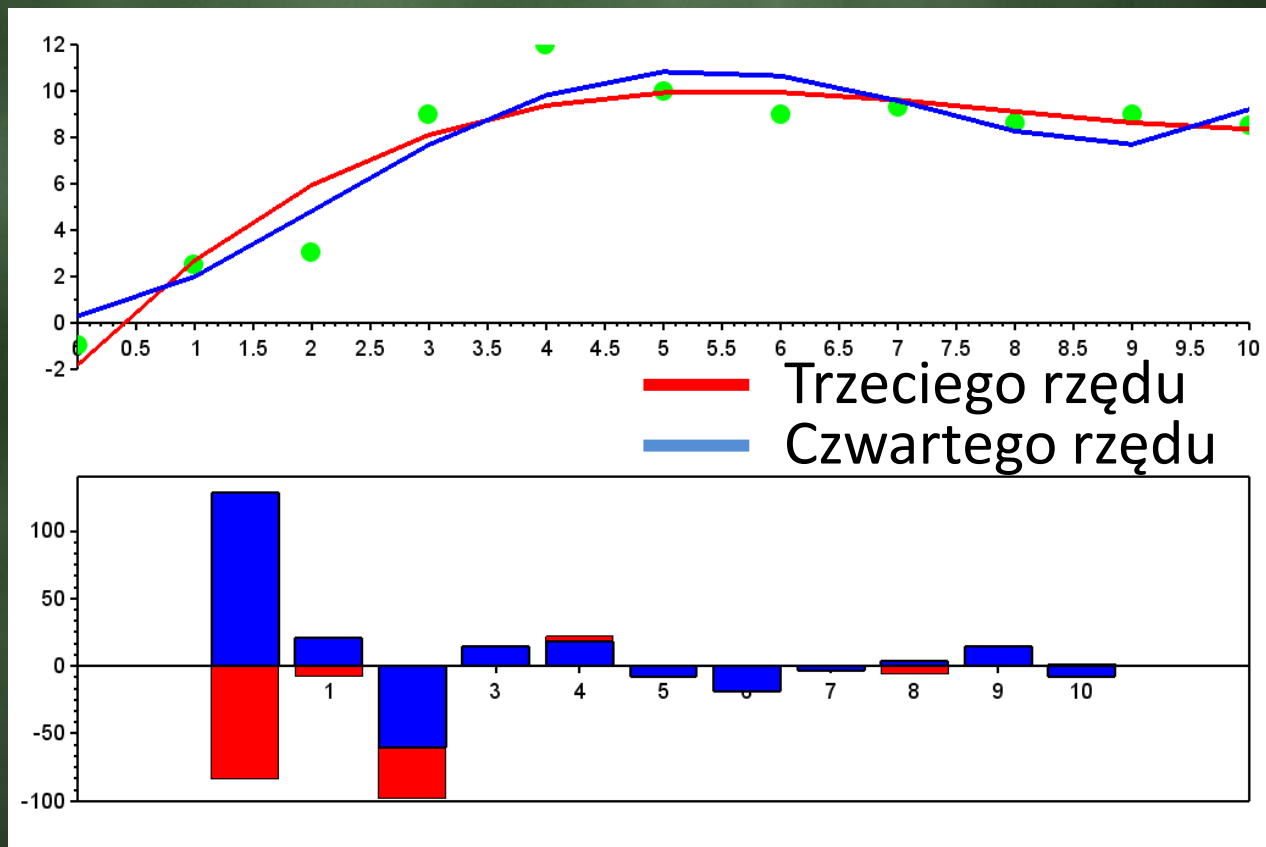


$$a_0=0.358 \quad a_1=7.405 \quad a_2=-1.146 \quad a_3=0.056$$

$$y=0.056t^3-1.146t^2+7.405t+0.358$$

Przykład 11

| t | z |
|----|-------|
| 0 | -1,00 |
| 1 | 2,50 |
| 2 | 3,00 |
| 3 | 9,00 |
| 4 | 12,00 |
| 5 | 10,00 |
| 6 | 9,00 |
| 7 | 9,30 |
| 8 | 8,60 |
| 9 | 9,00 |
| 10 | 8,50 |



$$z_3 = 0.031t^3 - 0.73t^2 + 5.22t - 1.84$$

$$z_4 = 0.014t^4 - 0.27t^3 + 1.28t^2 + 0.68t + 0.28$$

Aproksymacja

Dane pomiarowe aproksymuje się do funkcji najbardziej zbliżonej kształtem:

$$y = a_2 x^{a_1} + a_0$$

$$y = e^{a_2 x^2 + a_1 x + a_0}$$

$$y = a_2 e^{a_1 x} + a_0$$

$$y = a_2 x^2 + a_1 x + a_0$$

$$y = \sqrt{a_2 x^2 + a_1 x + a_0}$$

$$y = \frac{a_3 x + a_2}{a_1 x + a_0}$$

$$y = \frac{1}{a_2 x^2 + a_1 x + a_0}$$

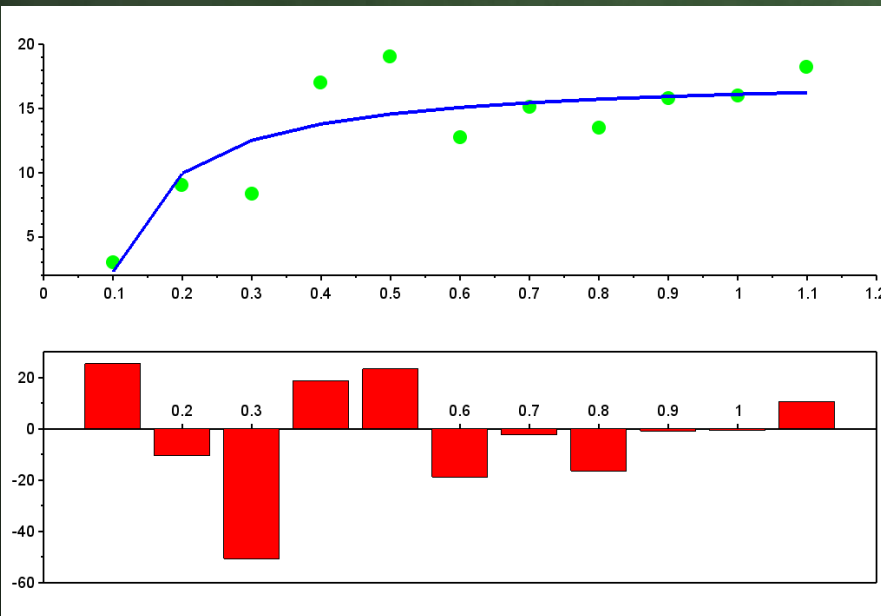
$$y = \frac{x}{a_2 x^2 + a_1 x + a_0}$$

$$y = a_2 + \frac{a_1}{x} + \frac{a_0}{x^2}$$

Aproksymacja

W wielu przypadkach można problem nieliniowy sprowadzić do zadania liniowego, stosując odpowiednie podstawienia i przekształcenia.

$$y = \frac{a_1}{x} + a_0 = a_1 \tilde{x} + a_0 \rightarrow \tilde{x} = \frac{1}{x}$$



$$y = -\frac{1.54}{x} + 17.65$$

Aproksymacja

$$y = a_0 e^{a_1 x}$$

$$\ln(y) = \ln(a_0) + \ln(e^{a_1 x})$$

$$\ln(y) = \ln(a_0) + a_1 x$$

$$\ln(y) = Y$$

$$\ln(a_0) = A$$

$$Y = A + a_1 x$$

$$y \approx 5.23e^{-0.5x}$$

