

Vysoká škola báňská – Technická univerzita Ostrava
Fakulta stavební



Přednáška z předmětu: Algoritmizace inženýrských výpočtů

Téma č.9: Interpolace a aproximace

doc. Ing. Martin Krejsa, Ph.D.

Obsah

1. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

Obsah

1 Úvod do interpolace a aproximace	3
2 Interpolace	5
2.1 Lineární interpolace	5
2.2 Lagrangeova interpolace	6
2.3 Newtonova interpolace	14
3 Aproximace	24
3.1 Aproximace metodou nejmenších čtverců	24
3.1.1 Aproximace přímkou	27
3.1.2 Aproximace polynomem m -tého stupně	29
Literatura	35



Obsah

2. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno



Kapitola 1

Úvod do interpolace a aproximace

Cíle

Kapitola by měla sloužit:

- k vysvětlení pojmů interpolace a aproximace,
- k uplatnění algoritmů pro interpolaci a aproximaci v inženýrských úlohách.

Úlohou **interpolace** je například:

- najít k funkci f_x mnohočlen $\Phi_n(x)$ n -tého stupně, který nabývá pro $n + 1$ argumentů x_k , kde $k = 0, 1, \dots, n$ týchž hodnot jako funkce f_x .
- počítat z tabulky funkce f_x sestavené pro $x = x_k$, přibližné hodnoty f_x pomocí mnohočlenu $\Phi_n(x)$ pro body x , které jsou různé od uzlových bodů x_i .

Obsah

3. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

Nejsou-li dané hodnoty funkce y_i ($i = 0, 1, \dots, n$) v uzlových bodech x_0, x_1, \dots, x_n dány přesně (jsou získány například měřením, které je vždy zatíženo chybou), nemá význam, aby se hledaná funkce ztotožnila s funkcí přesně v uzlových bodech, jako v případě interpolace. Úlohou **aproximace** je tedy nalezení jednodušší a matematicky přesně definované spojitě aproximační funkce F_x v intervalu $\langle a, b \rangle$, která by co nejlépe přiléhala k empirickým bodům x_0, x_1, \dots, x_n .

[Obsah](#)

4. strana ze 35

[Zavřít dokument](#)[Konec](#)[Celá obrazovka/Okno](#)



Kapitola 2

Interpolace

2.1. Lineární interpolace

Lineární interpolace umožňuje nahradit průběh funkce mezi dvěma body o souřadnicích x_k, y_k a x_{k+1}, y_{k+1} úsečkou, která je definovaná rovnicí přímky:

$$\frac{y(x) - y_k}{x - x_k} = \frac{y_{k+1} - y_k}{x_{k+1} - x_k}. \quad (2.1)$$

Pro úpravě (2.1) lze získat rovnici pro výpočet $y(x)$ s parametrem x :

$$y(x) = \frac{y_k \cdot (x - x_{k+1}) - y_{k+1} \cdot (x - x_k)}{x_k - x_{k+1}}. \quad (2.2)$$

[Obsah](#)[5. strana ze 35](#)[Zavřít dokument](#)[Konec](#)[Celá obrazovka/Okno](#)

Příklad 2.1. S využitím lineární interpolace pro dva body o souřadnicích $[x_0, y_0] = [1, 1.8]$ a $[x_1, y_1] = [2, 2.27]$ stanovte hodnotu interpolační funkce $y(x = 1.5)$.

Řešení. Funkce pro lineární interpolaci v souboru `lin_interpol.m` může vypadat např. takto:

```
function y=lin_interpol(x,xy2)
y=(xy2(1,2)*(x-xy2(2,1))-xy2(2,2)*(x-xy2(1,1)))/(xy2(1,1)-xy2(2,1));
```

Při vyvolání této funkce `y=lin_interpol(x,sour_xy_2b)` s parametry `x=1.5` a `sour_xy_2b=[1 1.8; 2 2.27]` lze získat výsledek:

```
y =
    2.0350
```

Lze rovněž znázornit graficky - viz obr. 2.1.



2.2. Lagrangeova interpolace

Pokud $f(x)$ je reálná funkce definovaná v intervalu $\langle a, b \rangle$, lze uvažovat také o funkci:

$$\Phi(x) = a_0 \cdot \varphi_0(x) + a_1 \cdot \varphi_1(x) + a_2 \cdot \varphi_2(x) + \dots + a_i \cdot \varphi_i(x) + \dots + a_n \cdot \varphi_n(x), \quad (2.3)$$



Obsah

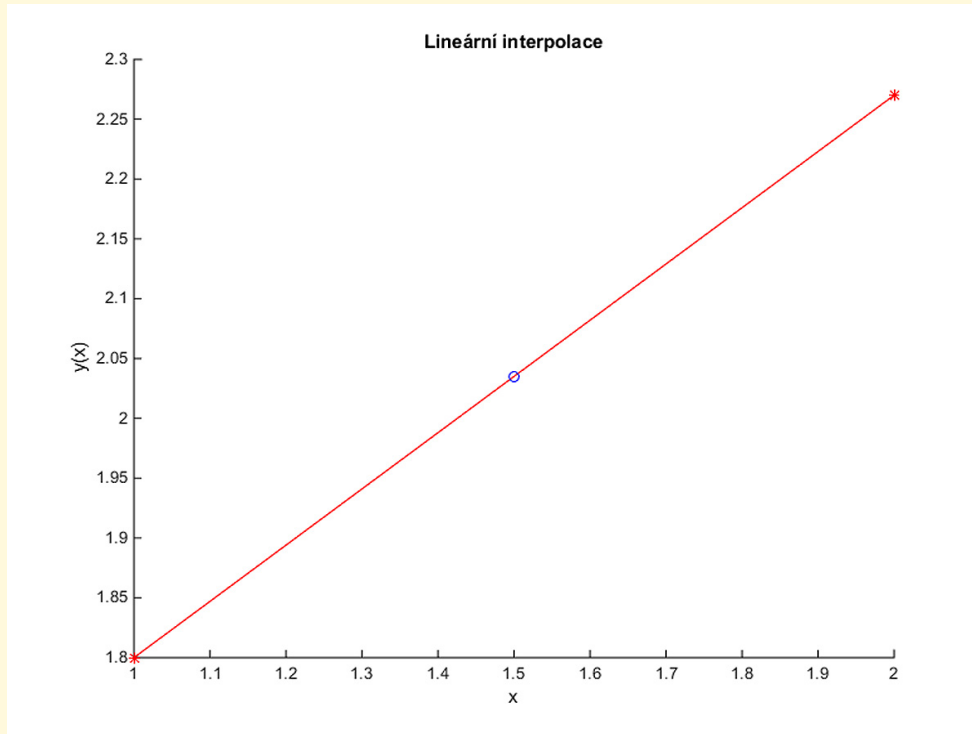
6. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno



Obr. 2.1 Výsledná lineární interpolace pro bod se souřadnicí $x = 1.5$, který je označen kroužkem



Obsah

7. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

kde a_i jsou reálné koeficienty a $\varphi_i(x)$ je rovna x^i pro $i = 0, 1, \dots, n$. Řešením je pak nalezení interpolačního polynomu $\Phi(x)$, pro který platí:

$$\Phi(x_i) = f(x_i), \quad (2.4)$$

kde x_i se nachází v intervalu $\langle a, b \rangle$ pro $i = 0, 1, 2, \dots, n$. Znamená to, že hledaná funkce interpolačního polynomu $\Phi(x)$ by měla mít totožné hodnoty s danou funkcí $f(x)$ pro $n + 1$ vstupních parametrů $x_0, x_1, x_2, \dots, x_n$.

Uvedený problém lze řešit např. postupným dosazováním $x = x_i, i = 0, 1, 2, \dots, n$ do rovnice (2.4), čímž lze získat soustavu $n + 1$ lineárních rovnic s neznámými koeficienty a_i :

$$\begin{aligned} a_0 \cdot \varphi(x_0) + a_1 \cdot \varphi(x_0) + \dots + a_n \cdot \varphi(x_0) &= f(x_0) \\ a_0 \cdot \varphi(x_1) + a_1 \cdot \varphi(x_1) + \dots + a_n \cdot \varphi(x_1) &= f(x_1) \\ &\vdots \\ a_0 \cdot \varphi(x_n) + a_1 \cdot \varphi(x_n) + \dots + a_n \cdot \varphi(x_n) &= f(x_n) \end{aligned} \quad (2.5)$$

Jeden ze způsobů, jak se při určování interpolačního polynomu $\Phi(x_i)$ řešení zmiňované soustavy lineárních rovnic (2.5) vyhnout, je Lagrangeova metoda.

Poznámka 2.2. I když je metoda pojmenovaná podle Josepha Louise Lagrange, který ji publikoval v roce 1795, byla poprvé objevena v roce 1779 Edwardem Waringem a její důsledky částečně zveřejněny v roce 1783 Leonhardem Eulerem.

Pokud je v intervalu $\langle a, b \rangle$ dáno $n+1$ různých uzlových bodů $x_0, x_1, x_2, \dots, x_n$ a hodnoty funkce $y_i = f(x_i)$ pro $i = 0, 1, \dots, n$, pak lze sestavit interpolační polynom stupně nejvýše n -tého, pro který bude platit:

$$\Phi_n(x) = P_0(x) + P_1(x) + \dots + P_n(x) = y_0 \cdot L_0(x) + y_1 \cdot L_1(x) + \dots + y_n \cdot L_n(x). \quad (2.6)$$



Obsah

8. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

Pro $L_i(x)$ platí:

$$L_i(x_j) = \begin{cases} 1 & \text{pro } i = j \\ 0 & \text{pro } i \neq j \end{cases}. \quad (2.7)$$

Tuto podmínku splňuje polynom:

$$\begin{aligned} L_i(x) &= \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x - x_j}{x_i - x_j} = \\ &= \frac{x - x_0}{x_i - x_0} \cdot \frac{x - x_1}{x_i - x_1} \cdot \dots \cdot \frac{x - x_{i-1}}{x_i - x_{i-1}} \cdot \frac{x - x_{i+1}}{x_i - x_{i+1}} \cdot \dots \cdot \frac{x - x_n}{x_i - x_n}. \end{aligned} \quad (2.8)$$

Výsledný tvar Lagrangeova interpolačního polynomu je pak:

$$\begin{aligned} L_n(x) &= y_0 \cdot \left(\frac{x - x_1}{x_0 - x_1} \cdot \frac{x - x_2}{x_0 - x_2} \cdot \dots \cdot \frac{x - x_n}{x_0 - x_n} \right) + \\ &+ y_1 \cdot \left(\frac{x - x_0}{x_1 - x_0} \cdot \frac{x - x_2}{x_1 - x_2} \cdot \dots \cdot \frac{x - x_n}{x_1 - x_n} \right) + \dots \\ &+ y_i \cdot \left(\frac{x - x_0}{x_i - x_0} \cdot \frac{x - x_1}{x_i - x_1} \cdot \dots \cdot \frac{x - x_{i-1}}{x_i - x_{i-1}} \cdot \frac{x - x_{i+1}}{x_i - x_{i+1}} \cdot \dots \cdot \frac{x - x_n}{x_i - x_n} \right) + \dots \\ &+ y_n \cdot \left(\frac{x - x_0}{x_n - x_0} \cdot \frac{x - x_1}{x_n - x_1} \cdot \dots \cdot \frac{x - x_{n-1}}{x_n - x_{n-1}} \right). \end{aligned} \quad (2.9)$$

Funkci, která stanoví pro zadaný bod se souřadnicí x ve vstupním parametru par hodnotu Lagrangeova interpolačního polynomu, sestaveného pro zadanou množinu bodů se



Obsah

9. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

souřadnicemi x a y uloženou ve vstupních parametrech s vektory x a y , lze naprogramovat v MATLABU např. pomocí skriptu `lagrange.m`:

```
function s=lagrange(x,y,par)
n=length(x);
s=0;
for i=1:n
    m=y(i);
    for j=1:n
        if ~(i==j)
            m=m*(par-x(j))/(x(i)-x(j));
        end
    end
    s=s+m;
end
```

Příklad 2.3. S využitím Lagrangeova interpolačního polynomu pro tři body o souřadnicích $[x_0, y_0] = [0, 1]$, $[x_1, y_1] = [2, 2]$ a $[x_2, y_2] = [3, 4]$ stanovte rovnici interpolační funkce $y(x)$.

Řešení. Uvedený příklad lze řešit obecně dosazením zadaných souřadnic tří bodů do obecné rovnice interpolačního polynomu (2.9):

[Obsah](#)[10. strana ze 35](#)[Zavřít dokument](#)[Konec](#)[Celá obrazovka/Okno](#)

$$\begin{aligned}
L_2(x) &= y_0 \cdot \frac{(x-x_1) \cdot (x-x_2)}{(x_0-x_1) \cdot (x_0-x_2)} + y_1 \cdot \frac{(x-x_0) \cdot (x-x_2)}{(x_1-x_0) \cdot (x_1-x_2)} + \\
&+ y_2 \cdot \frac{(x-x_0) \cdot (x-x_1)}{(x_2-x_0) \cdot (x_2-x_1)} = \\
&= 1 \cdot \frac{(x-2) \cdot (x-3)}{(0-2) \cdot (0-3)} + 2 \cdot \frac{(x-0) \cdot (x-3)}{(2-0) \cdot (2-3)} + 4 \cdot \frac{(x-0) \cdot (x-2)}{(3-0) \cdot (3-2)} = \quad (2.10) \\
&= \frac{1}{6} \cdot (x^2 - 5 \cdot x + 6) + 2 \cdot -\frac{1}{2} \cdot (x^2 - 3 \cdot x) + 4 \cdot \frac{1}{3} \cdot (x^2 - 2 \cdot x) = \\
&= \frac{1}{2} \cdot x^2 - \frac{1}{2} \cdot x + 1 .
\end{aligned}$$

O správnosti odvozeného interpolačního polynomu se lze přesvědčit dosazením souřadnic zadaných bodů:

$$L_2(x_0) = \frac{1}{2} \cdot x_0^2 - \frac{1}{2} \cdot x_0 + 1 = \frac{1}{2} \cdot 0^2 - \frac{1}{2} \cdot 0 + 1 = 1 , \quad (2.11)$$

$$L_2(x_1) = \frac{1}{2} \cdot x_1^2 - \frac{1}{2} \cdot x_1 + 1 = \frac{1}{2} \cdot 2^2 - \frac{1}{2} \cdot 2 + 1 = 2 , \quad (2.12)$$

$$L_2(x_2) = \frac{1}{2} \cdot x_2^2 - \frac{1}{2} \cdot x_2 + 1 = \frac{1}{2} \cdot 3^2 - \frac{1}{2} \cdot 3 + 1 = 4 . \quad (2.13)$$

Sestrojený Lagrangeův interpolační polynom lze zobrazit rovněž graficky - viz obr. 2.2.



Obsah

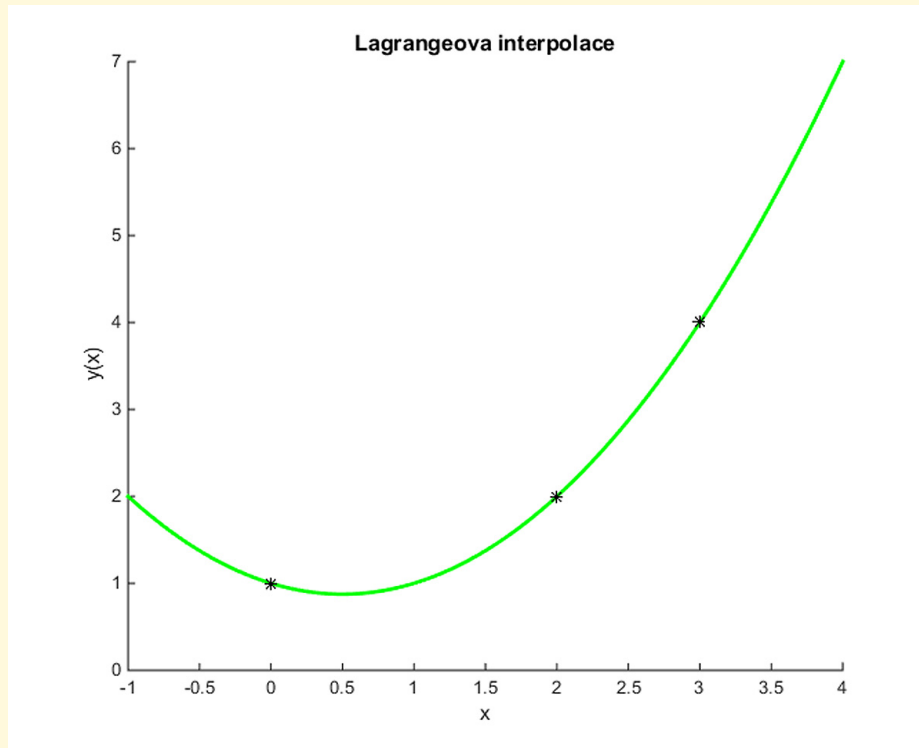
11. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno



Obr. 2.2 Výsledná interpolace s využitím Lagrangeova interpolačního polynomu pro body o souřadnicích $[x_0, y_0] = [0, 1]$, $[x_1, y_1] = [2, 2]$ a $[x_2, y_2] = [3, 4]$



Obsah

12. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

Příklad 2.4. S využitím Lagrangeova interpolačního polynomu stanovte hodnotu ohybového momentu konstrukce popsané v příkladu 2.1 pro bod se souřadnicí $x = l/5 = 1.2$ m. Pro sestavení Lagrangeova interpolačního polynomu využijte hodnoty skutečných ohybových momentů ve třech bodech o souřadnicích $[x_0, x_1, x_2] = [0, l/2, l] = [0, 3, 6]$ m.

Řešení. Nejprve je samozřejmě nutné stanovit v zadaných bodech hodnoty skutečných ohybových momentů, které pro dané zadání nabývají hodnot $M_y(x_0 = 0) = 0$ kNm, $M_y(x_1 = 3) = 9$ kNm a $M_y(x_2 = 6) = -18$ kNm. Vytvoření Lagrangeova interpolačního polynomu je možné již vytvořeným a dříve popsáním skriptem `lagrange.m`. Celý výpočet může být proveden např. následujícím sledem příkazů:

```
clc; clear; format short;
qz=4000;
l=6;
M=[0 3/8*qz*l -qz/2]/1000;
x=[0 1/2 l]; % 0 m, 3 m, 6 m
y=[horner(2,M,x(1)) horner(2,M,x(2)) horner(2,M,x(3))];
par=l/5;
res=lagrange(x,y,par)
```

Výsledkem celého řešení je pak hodnota v bodu o souřadnici $x = l/5 = 1.2$ m:

```
res =
    7.9200
```



Obsah

13. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

Vzhledem ke skutečnosti, že výsledný Lagrangeův interpolační polynom tvoří stejně jako průběh ohybových momentů polynom 2. stupně, lze pozorovat naprostou shodu mezi touto dvojicí funkcí - viz obr. 2.3.



2.3. Newtonova interpolace

Interpolační polynom $\Phi_n(x)$ lze vyjádřit také funkcí obsahující diference:

$$\begin{aligned}\Phi_n(x) = & a_0 + a_1 \cdot (x - x_0) + a_2 \cdot (x - x_0) \cdot (x - x_1) + \\ & + \dots + a_n \cdot (x - x_0) \cdot (x - x_1) \cdot \dots \cdot (x - x_{n-1}).\end{aligned}\quad (2.14)$$

Požadovaný interpolační polynom $\Phi_n(x)$ musí v bodech x_i nabývat hodnot:

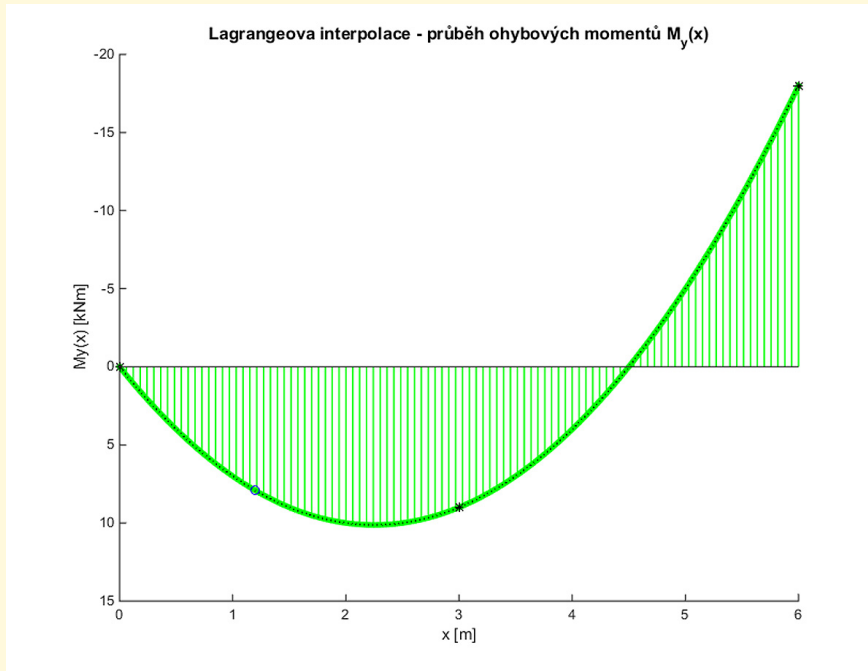
$$\Phi_n(x_i) = f(x_i) \text{ pro } i = 0, 1, \dots, n. \quad (2.15)$$

Řešením úlohy je stanovení neznámých koeficientů a_k pro $k = 0, 1, \dots, n$, jenž jsou obsaženy ve vztahu 2.14. Postupným dosazováním $x = x_i$ pro $i = 0, 1, \dots, n$ do polynomu $\Phi_n(x)$ z 2.14 lze získat následující podmínky:

[Obsah](#)

14. strana ze 35

[Zavřít dokument](#)[Konec](#)[Celá obrazovka/Okno](#)



Obr. 2.3 Výsledná interpolace průběhu ohybových momentů na nosníku z příkladu 2.1 s využitím Lagrangeova interpolačního polynomu pro body $M_y(x_0 = 0) = 0$ kNm, $M_y(x_1 = 3) = 9$ kNm a $M_y(x_2 = 6) = -18$ kNm označené hvězdičkou. Hodnota ohybového momentu v zadaném bodu o souřadnici $x = l/5 = 1.2$ m je pak naznačena kroužkem.



Obsah

15. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

$$\begin{aligned}
\Phi_0(x_0) &= f(x_0) = a_0 \\
\Phi_1(x_1) &= f(x_1) = a_0 + a_1 \cdot (x_1 - x_0) \\
\Phi_2(x_2) &= f(x_2) = a_0 + a_1 \cdot (x_1 - x_0) + a_2 \cdot (x_1 - x_0) \cdot (x_2 - x_0) \\
&\vdots \\
\Phi_n(x_n) &= f(x_n) = a_0 + a_1 \cdot (x_1 - x_0) + a_2 \cdot (x_1 - x_0) \cdot (x_2 - x_0) + \\
&\quad + \dots + a_n \cdot (x_1 - x_0) \cdot (x_2 - x_0) \cdot \dots \cdot (x_n - x_0) .
\end{aligned} \tag{2.16}$$

Hledané koeficienty a_k lze z uvedených rovnic postupně určit, např.:

$$\begin{aligned}
a_0 &= f(x_0) \\
a_1 &= \frac{f(x_1) - a_0}{x_1 - x_0} \\
a_2 &= \frac{f(x_2) - a_0 - a_1 \cdot (x_1 - x_0)}{(x_1 - x_0) \cdot (x_2 - x_0)} \\
&\vdots \\
a_n &= \frac{f(x_n) - a_0 - a_1 \cdot (x_1 - x_0) - \dots - a_{n-1} \cdot (x_1 - x_0) \cdot \dots \cdot (x_{n-1} - x_0)}{(x_1 - x_0) \cdot (x_2 - x_0) \cdot \dots \cdot (x_n - x_0)} .
\end{aligned} \tag{2.17}$$

Rovnice (2.16) lze pro $k = 1, \dots, n$ vyjádřit rovněž pomocí následujícího vztahu:

$$\Phi_k(x_k) = f(x_k) = \Phi_{k-1}(x_k) + a_k \cdot (x_1 - x_0) \cdot (x_2 - x_0) \cdot \dots \cdot (x_k - x_0) , \tag{2.18}$$



Obsah

16. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

čehož lze využít i zobecnění výpočtu neznámých koeficientů a_k :

$$a_k = \frac{f(x_k) - \Phi_{k-1}(x_k)}{(x_1 - x_0) \cdot (x_2 - x_0) \cdot \dots \cdot (x_k - x_0)}. \quad (2.19)$$

Uvedený problém lze popsat také s využitím tzv. dělených diferencí:

$$\begin{aligned} f[x_k] &= f(x_k) \\ f[x_k \ x_{k+1}] &= \frac{f[x_{k+1}] - f[x_k]}{x_{k+1} - x_k} \\ f[x_k \ x_{k+1} \ x_{k+2}] &= \frac{f[x_{k+1} \ x_{k+2}] - f[x_k \ x_{k+1}]}{x_{k+2} - x_k} \\ f[x_k \ x_{k+1} \ x_{k+2} \ x_{k+3}] &= \frac{f[x_{k+1} \ x_{k+2} \ x_{k+3}] - f[x_k \ x_{k+1} \ x_{k+2}]}{x_{k+3} - x_k} \\ &\vdots \end{aligned} \quad (2.20)$$

Tato čísla odpovídají koeficientům a_k pro $k = 0, 1, \dots, n$ Newtonova interpolačního polynomu, který lze definovat ve finální podobě:

$$\begin{aligned} N_n(x) &= f[x_1] + f[x_1 \ x_2] \cdot (x - x_1) \\ &\quad + f[x_1 \ x_2 \ x_3] \cdot (x - x_1) \cdot (x - x_2) \\ &\quad + f[x_1 \ x_2 \ x_3 \ x_4] \cdot (x - x_1) \cdot (x - x_2) \cdot (x - x_3) \\ &\quad + \dots + \\ &\quad + f[x_1 \ \dots \ x_n] \cdot (x - x_1) \cdot \dots \cdot (x - x_{n-1}), \end{aligned} \quad (2.21)$$



Obsah

17. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

resp. pro $k = 1, \dots, n$:

$$N_k(x) = N_{k-1}(x) + f[x_1 \ \dots \ x_k] \cdot (x - x_1) \cdot \dots \cdot (x - x_{k-1}) . \quad (2.22)$$

Výpočet Newtonova interpolačního polynomu lze algoritmicky vyjádřit např. pomocí algoritmu 1.

Vstup : $x = [x_1 \ \dots \ x_n]$, $y = [y_1 \ \dots \ y_n]$, z

Výstup: $N_n(z)$

for $j \leftarrow 1, 2, \dots, n$ **do**

 | $f[x_j] \leftarrow y_j$

end

for $i \leftarrow 2, 3, \dots, n$ **do**

 | **for** $j \leftarrow 1, 2, \dots, n + 1 - i$ **do**

$$| \quad | \quad f[x_j \ \dots \ x_{j+i-1}] \leftarrow \frac{f[x_{j+1} \ \dots \ x_{j+i-1}] - f[x_j \ \dots \ x_{j+i-2}]}{x_{j+i-1} - x_j}$$

 | **end**

end

$$N_n(z) \leftarrow \sum_{i=1}^n f[x_1 \ \dots \ x_i] \cdot (z - x_1) \cdot \dots \cdot (z - x_{i-1})$$

Algoritmus 1: Stanovení hodnoty Newtonova interpolačního polynomu $N_n(x)$

Pro rekurzivní vyjádření dělených diferencí Newtonova interpolačního polynomu se používá tabulkového vyjádření (pro tři body viz tab. 2.1). Koefficienty Newtonova interpolačního polynomu (2.22) pak lze odečíst z horní hrany zobrazeného trojúhelníka.



Obsah

18. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

x_1	$f[x_1]$		
x_2	$f[x_2]$	$f[x_1 \ x_2]$	
x_3	$f[x_3]$	$f[x_2 \ x_3]$	$f[x_1 \ x_2 \ x_3]$

Tab. 2.1 Dělené diference Newtonova interpolačního polynomu pro tři body

Příklad 2.5. S využitím Newtonova interpolačního polynomu stanovte rovnici interpolační funkce $y(x)$ pro tři body z příkladu 2.3 o souřadnicích $[x_0, y_0] = [0, 1]$, $[x_1, y_1] = [2, 2]$ a $[x_2, y_2] = [3, 4]$.

Řešení. S využitím postupu (2.21) pro sestavení Newtonova interpolačního polynomu lze sestavit tabulku 2.2, ve které se jednotlivé členy určí s pomocí (2.20):

$$f[x_1 \ x_2] = \frac{f[x_2] - f[x_1]}{x_2 - x_1} = \frac{2 - 1}{2 - 0} = \frac{1}{2}, \quad (2.23)$$

$$f[x_1 \ x_2 \ x_3] = \frac{f[x_2 \ x_3] - f[x_1 \ x_2]}{x_3 - x_1} = \frac{2 - \frac{1}{2}}{3 - 0} = \frac{1}{2}, \quad (2.24)$$

$$f[x_2 \ x_3] = \frac{f[x_3] - f[x_2]}{x_3 - x_2} = \frac{4 - 2}{3 - 2} = 2. \quad (2.25)$$



Obsah

19. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

$$\begin{array}{l|l}
 x_1 = 0 & f[x_1] = 1 \\
 & f[x_1 \ x_2] = \frac{1}{2} \\
 x_2 = 2 & f[x_2] = 2 \\
 & f[x_2 \ x_3] = 2 \\
 x_3 = 3 & f[x_3] = 4 \\
 & f[x_1 \ x_2 \ x_3] = \frac{1}{2}
 \end{array}$$

Tab. 2.2 Dělené diference Newtonova interpolačního polynomu pro tři body z příkladu 2.3

Jak již bylo řečeno, koeficienty hledaného Newtonova interpolačního polynomu definovaného (2.22) pak lze odečíst z tabulky 2.2 z horní hrany zobrazeného trojúhelníka:

$$\begin{aligned}
 N_3(x) &= f[x_1] + f[x_1 \ x_2] \cdot (x - x_1) + f[x_1 \ x_2 \ x_3] \cdot (x - x_1) \cdot (x - x_2) = \\
 &= 1 + \frac{1}{2} \cdot (x - 0) + \frac{1}{2} \cdot (x - 0) \cdot (x - 2) = \frac{1}{2} \cdot x^2 - \frac{1}{2} \cdot x + 1.
 \end{aligned} \tag{2.26}$$

Z výsledné rovnice vztahu Newtonova interpolačního polynomu (2.26) je zřejmé, že bylo dosaženo stejného polynomu druhého řádu jako v případě příkladu (2.3). ▲

Funkci, která stanoví pro zadaný bod se souřadnicí x ve vstupním parametru **par** hodnotu Newtonova interpolačního polynomu, sestaveného pro zadanou množinu bodů se souřadnicemi x a y uloženou ve vstupních parametrech s vektory \mathbf{x} a \mathbf{y} , lze naprogramovat v MATLABU např. pomocí skriptu `newton.m`:

```
function s=newton(x,y,par)
```



Obsah

20. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

```
n=length(x);
for j=1:n
    tab(j,1)=y(j);
end
for i=2:n
    for j=1:n+1-i
        tab(j,i)=(tab(j+1,i-1)-tab(j,i-1))/(x(j+i-1)-x(j));
    end
end
s=tab(1,1);
for i=2:n
    m=tab(1,i);
    for j=1:i-1
        m=m*(par-x(j));
    end
    s=s+m;
end
```

Skript lze nepatrně upravit tak, aby bylo možno hodnoty Newtonova interpolačního polynomu efektivně určit i pro vektor, obsahující ve vstupním parametru **par** souřadnice x více bodů (skript je funkční i pro jednu souřadnici).

```
function s=newton(x,y,par)
n=length(x);
for j=1:n
    tab(j,1)=y(j);
end
```

[Obsah](#)[21. strana ze 35](#)[Zavřít dokument](#)[Konec](#)[Celá obrazovka/Okno](#)

```
for i=2:n
    for j=1:n+1-i
        tab(j,i)=(tab(j+1,i-1)-tab(j,i-1))/(x(j+i-1)-x(j));
    end
end
num=length(par);
for k=1:num
    tot=tab(1,1);
    for i=2:n
        m=tab(1,i);
        for j=1:i-1
            m=m*(par(k)-x(j));
        end
        tot=tot+m;
    end
    s(k)=tot;
end
```

Příklad 2.6. S využitím Newtonova interpolačního polynomu stanovte hodnotu ohybového momentu podle zadání v příkladu 2.6.

Poznámka 2.7. Pro konstrukci Newtonova interpolačního polynomu lze použít i velmi zajímavý skript - viz dále, který umožňuje zadávat jednotlivé body potřebné k sestavení interpolačního polynomu přímo z grafu klikáním levým tlačítkem myši. Je zajímavé sledovat,

[Obsah](#)[22. strana ze 35](#)[Zavřít dokument](#)[Konec](#)[Celá obrazovka/Okno](#)

jak se s přibývajícimi body zvyšuje řád interpolačního polynomu. Výpočet se ukončí po kliknutí pravým tlačítkem myši.

```
xmin=-3; xmax=3;
x_p=xmin:.01:xmax;
ymin=-3; ymax=3;
plot([xmin xmax],[0 0],'k',[0 0],[ymin ymax],'k');
grid on;
x=[]; y=[];
tlac=1; k=0;
while ~(tlac==3)
    [x_novy,y_novy,tlac]=ginput(1);
    if tlac==1
        k=k+1; x(k)=x_novy; y(k)=y_novy;
        y_p=newton(x,y,x_p);
        plot(x,y,'o',x_p,y_p,[xmin xmax],[0,0],'k',[0 0],[ymin ymax],'k');
        axis([xmin xmax ymin ymax]);
        grid on;
    end
end
```

[Obsah](#)

23. strana ze 35

[Zavřít dokument](#)[Konec](#)[Celá obrazovka/Okno](#)



Kapitola 3

Aproximace

3.1. Aproximace metodou nejmenších čtverců

Při interpolaci některou z předchozích metod se předpokládalo, že interpolovaná funkce je zadaná tabulkou s hodnotami x_i a $f(x_i) = y_i$, kde $i = 0, 1, \dots, n$. V případě aproximace není úkolem najít funkci, která se ztotožní v zadaných bodech s hledanou funkcí, nýbrž určit aproximační funkci $F(x)$, která by co nejlépe přiléhala k $n + 1$ zadaným empirickým bodům $[x_0, y_0]$, $[x_1, y_1]$ až $[x_n, y_n]$.

V metodě nejmenších čtverců se jako kritérium přiléhavosti využívá součet druhých mocnin (čtverců) rozdílů mezi hodnotami aproximační funkce $F(x_i)$ a naměřenými hodnotami y_i :

[Obsah](#)

24. strana ze 35

[Zavřít dokument](#)[Konec](#)[Celá obrazovka/Okno](#)

$$Q = \sum_{i=0}^n (F(x_i) - y_i)^2. \quad (3.1)$$

Funkce $F(x)$ může být obecně dána jako:

$$F(x) = a_0 \cdot f_0(x) + a_1 \cdot f_1(x) + \dots + a_m \cdot f_m(x), \quad (3.2)$$

kde f_0, f_1, \dots, f_m jsou vhodně zvolené lineárně nezávislé funkce a a_0, a_1, \dots, a_m neznámé reálné koeficienty, které se určí tak, aby hodnota Q ve vztahu (3.1) byla minimální. Musí tedy platit:

$$\frac{\partial Q}{\partial a_k} = 2 \cdot \sum_{i=0}^n (F(x_i) - y_i) \cdot \frac{\partial F(x_i)}{\partial a_k} = 0, \quad (3.3)$$

kde $k = 0, 1, \dots, m$.

Při volbě

$$\frac{\partial F(x_i)}{\partial a_k} = f_k(x_i), \quad (3.4)$$

musí platit:

$$\frac{\partial Q}{\partial a_k} = 2 \cdot \sum_{i=0}^n [a_0 \cdot f_0(x_i) + a_1 \cdot f_1(x_i) + \dots + a_m \cdot f_m(x_i) - y_i] \cdot f_k(x_i) = 0. \quad (3.5)$$

Vztah (3.5) lze dále upravit:



Obsah

25. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

$$\begin{aligned} \sum_{i=0}^n [a_0 \cdot f_k(x_i) \cdot f_0(x_i) + a_1 \cdot f_k(x_i) \cdot f_1(x_i) + \dots + a_m \cdot f_k(x_i) \cdot f_m(x_i)] = \\ = \sum_{i=0}^n f_k(x_i) \cdot y_i, \end{aligned} \quad (3.6)$$

resp.

$$\begin{aligned} a_0 \cdot \sum_{i=0}^n f_k(x_i) \cdot f_0(x_i) + a_1 \cdot \sum_{i=0}^n f_k(x_i) \cdot f_1(x_i) + \dots + a_m \cdot \sum_{i=0}^n f_k(x_i) \cdot f_m(x_i) = \\ = \sum_{i=0}^n f_k(x_i) \cdot y_i, \end{aligned} \quad (3.7)$$

kde $k = 0, 1, \dots, m$.

Vztah (3.7) lze vyjádřit i v maticovém tvaru:



Obsah

26. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

$$\begin{bmatrix} \sum_{i=0}^n f_0^2(x_i) & \sum_{i=0}^n f_0(x_i) \cdot f_1(x_i) & \dots & \sum_{i=0}^n f_0(x_i) \cdot f_m(x_i) \\ \sum_{i=0}^n f_1(x_i) \cdot f_0(x_i) & \sum_{i=0}^n f_1^2(x_i) & \dots & \sum_{i=0}^n f_1(x_i) \cdot f_m(x_i) \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=0}^n f_m(x_i) \cdot f_0(x_i) & \sum_{i=0}^n f_m(x_i) \cdot f_1(x_i) & \dots & \sum_{i=0}^n f_m^2(x_i) \end{bmatrix} \cdot \begin{Bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{Bmatrix} = \begin{Bmatrix} \sum_{i=0}^n f_0(x_i) \cdot y_i \\ \sum_{i=0}^n f_1(x_i) \cdot y_i \\ \vdots \\ \sum_{i=0}^n f_m(x_i) \cdot y_i \end{Bmatrix} . \quad (3.8)$$

3.1.1. Aproximace přímkou

Při lineární aproximaci platí mezi proměnnými x a y vztah:

$$F(x) = a \cdot x + b , \quad (3.9)$$

kde a, b jsou neznámé parametry, jež lze určit z podmínky podle (3.1):



Obsah

27. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

$$Q = \sum_{i=0}^n (a \cdot x_i + b - y_i)^2 = \min . \quad (3.10)$$

Řešení úlohy dané vztahem (3.2) vede k soustavě dvou rovnic:

$$\frac{\partial Q}{\partial a} = 0 \quad (3.11)$$

a

$$\frac{\partial Q}{\partial b} = 0 . \quad (3.12)$$

Po úpravě obou rovnic podle (3.5) až (3.7) lze získat jejich výsledný tvar:

$$\begin{aligned} n \cdot b + \left(\sum_{i=0}^n x_i \right) \cdot a &= \sum_{i=0}^n y_i \\ \left(\sum_{i=0}^n x_i \right) \cdot b + \left(\sum_{i=0}^n x_i^2 \right) \cdot a &= \sum_{i=0}^n x_i \cdot y_i , \end{aligned} \quad (3.13)$$

jež lze vyjádřit maticově:

$$\begin{bmatrix} n & \sum_{i=0}^n x_i \\ \sum_{i=0}^n x_i & \sum_{i=0}^n x_i^2 \end{bmatrix} \cdot \begin{Bmatrix} b \\ a \end{Bmatrix} = \begin{Bmatrix} \sum_{i=0}^n y_i \\ \sum_{i=0}^n x_i \cdot y_i \end{Bmatrix} . \quad (3.14)$$



Obsah

28. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

3.1.2. Aproximace polynomem m -tého stupně

Pokud bude zvolen za aproximující funkci polynom m -tého stupně:

$$F_m(x) = a_0 \cdot x^0 + a_1 \cdot x^1 + a_2 \cdot x^2 + \dots + a_m \cdot x^m, \quad (3.15)$$

po úpravách (3.3) až (3.7) lze po dosazení za $f_k(x) = x^k$, $k = 0, 1, \dots, m$ získat soustavu $m + 1$ rovnic:

$$\begin{bmatrix} \sum_{i=0}^n (x_i^0)^2 & \sum_{i=0}^n x_i^0 \cdot x_i^1 & \dots & \sum_{i=0}^n x_i^0 \cdot x_i^m \\ \sum_{i=0}^n x_i^1 \cdot x_i^0 & \sum_{i=0}^n (x_i^1)^2 & \dots & \sum_{i=0}^n x_i^1 \cdot x_i^m \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=0}^n x_i^m \cdot x_i^0 & \sum_{i=0}^n x_i^m \cdot x_i^1 & \dots & \sum_{i=0}^n (x_i^m)^2 \end{bmatrix} \cdot \begin{Bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{Bmatrix} = \begin{Bmatrix} \sum_{i=0}^n x_i^0 \cdot y_i \\ \sum_{i=0}^n x_i^1 \cdot y_i \\ \vdots \\ \sum_{i=0}^n x_i^m \cdot y_i \end{Bmatrix}. \quad (3.16)$$

Soustavu rovnic (3.16) s neznámými koeficienty a_0, a_1, \dots, a_m pak lze dále upravit na tvar:

$$\begin{bmatrix} n+1 & \sum_{i=0}^n x_i & \sum_{i=0}^n x_i^2 & \dots & \sum_{i=0}^n x_i^m \\ \sum_{i=0}^n x_i & \sum_{i=0}^n x_i^2 & \sum_{i=0}^n x_i^3 & \dots & \sum_{i=0}^n x_i^{m+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \sum_{i=0}^n x_i^m & \sum_{i=0}^n x_i^{m+1} & \sum_{i=0}^n x_i^{m+2} & \dots & \sum_{i=0}^n x_i^{2 \cdot m} \end{bmatrix} \cdot \begin{Bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{Bmatrix} = \begin{Bmatrix} \sum_{i=0}^n y_i \\ \sum_{i=0}^n x_i \cdot y_i \\ \vdots \\ \sum_{i=0}^n x_i^m \cdot y_i \end{Bmatrix}. \quad (3.17)$$



Obsah

29. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

Příklad 3.1. Proveďte lineární aproximaci i aproximaci polynomem 2.stupně pro data obsažená v tabulce 3.1. U obou případů stanovte součet druhých mocnin (čtverců) rozdílů mezi hodnotami aproximační funkce $F(x_i)$ a naměřenými hodnotami y_i .

x	1	2	3	4	5
y	0	2	2	5	4

Tab. 3.1 Vstupní údaje pro výpočet aproximace v příkladu 3.1

Řešení. Výpočet lineární aproximace i polynomem m -tého stupně včetně součtu druhých mocnin (čtverců) rozdílů mezi hodnotami příslušné aproximační funkce $F(x_i)$ a naměřenými hodnotami y_i lze provést následujícím sledem příkazů:

```
clear; clc;
x0=[1 2 3 4 5];
y0=[0 2 2 5 4];
m=1;
for i=1:m+1
    for j=i:m+1
        A(i,j)=sum(x0.^((i-1)+(j-1)));
        if ~(i==j)
            A(j,i)=A(i,j);
        end
    end
end
b(i)=sum((x0.^(i-1)).*y0);
```



Obsah

30. strana ze 35



Zavřít dokument

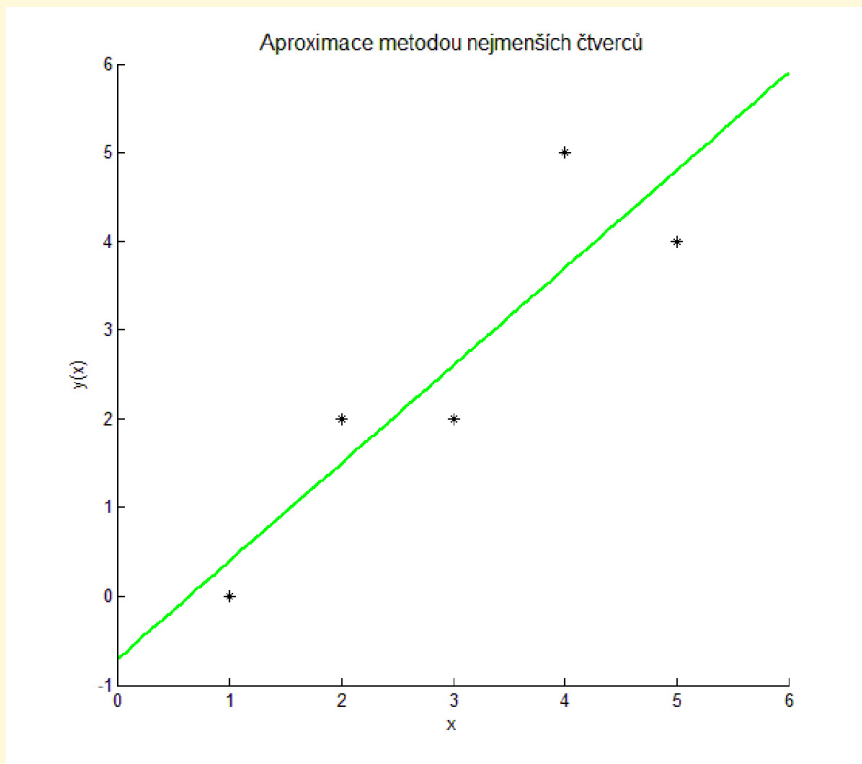
Konec

Celá obrazovka/Okno

```
end
c=A\b';
x=0:.1:6;
for j=1:length(x)
    s=c(1);
    for i=1:m
        s=s+c(i+1)*x(j)^(i);
    end
    y(j)=s;
end
plot(x0,y0,'o',x,y);
soucet_ctvercu=0;
for j=1:length(x0)
    s=c(1);
    for i=1:m
        s=s+c(i+1)*x0(j)^(i);
    end
    soucet_ctvercu=soucet_ctvercu+(s-y0(j))^2;
end
soucet_ctvercu
```

Pro případ lineární aproximace lze získat přímkou, zobrazenou na obr. 3.1, se součtem čtverců rozdílů mezi hodnotami příslušné aproximační funkce $F(x_i)$ a naměřenými hodnotami y_i rovném 3.1. V případě aproximace polynomem 2. stupně - viz obr. 3.2 je součet čtverců rozdílů mezi hodnotami příslušné aproximační funkce $F(x_i)$ a naměřenými hodnotami y_i roven hodnotě 2.4571.

[Obsah](#)[31. strana ze 35](#)[Zavřít dokument](#)[Konec](#)[Celá obrazovka/Okno](#)



Obr. 3.1 Lineární aproximace pro body, zadané v příkladu 3.1



Obsah

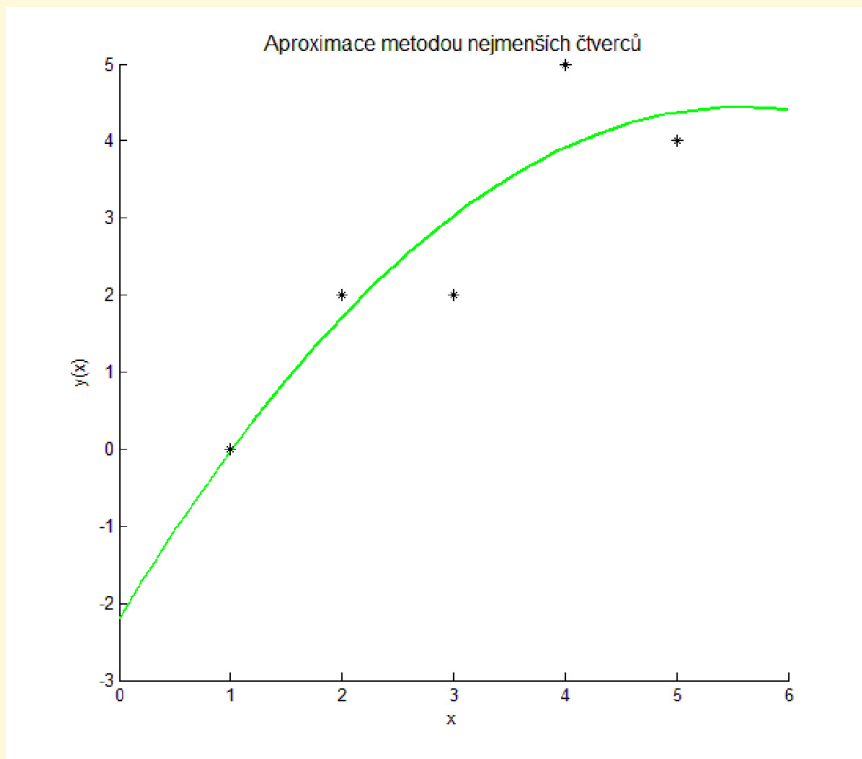
32. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno



Obr. 3.2 Aproximace polynomem 2. stupně pro body, zadané v příkladu 3.1



Obsah

33. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno



Příklad 3.2. Vyberte nejvhodnější stupeň polynomu pro aproximaci naměřených hodnot krychelné pevnosti betonu v tlaku v závislosti na dnech zrání betonové směsi, které jsou zobrazeny v tabulce 3.2. Jako kritérium nejlepší přiléhavosti použijte součet druhých mocnin (čtverců) rozdílů mezi hodnotami aproximační funkce $F(x_i)$ a naměřenými hodnotami y_i .

x [dny]	0	0	0	7	7	7	14	14	14	28	28	28
y [MPa]	0	0	0	21.5	22.2	21.2	30.7	31.4	30.5	40.1	43.4	41.5

Tab. 3.2 Vstupní údaje pro výpočet aproximační funkce v příkladu 3.2

Obsah

34. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno

Literatura

- [1] Olehla, M. — Tišer, J. *Praktické použití Fortranu*. 2. upravené vydání. Nakladatelství dopravy a spojů, Praha, 1979. (432 s).
- [2] Sauer T. *Numerical Analysis*. George Mason University. Pearson Education, Inc., 2006. (669 s). ISBN 0-321-26898-9.



Obsah

35. strana ze 35



Zavřít dokument

Konec

Celá obrazovka/Okno