UNIVERZITA PALACKÉHO V OLOMOUCI
PŘÍRODOVĚDECKÁ FAKULTA

BAKALÁŘSKÁ PRÁCE

Interpolace v R^2

Katedra matematické analýzy a aplikací matematiky
Vedoucí bakalářské práce: **RNDr. Jitka Machalová, Ph.D.**
Vypracovala: **Hana Zedková**
Studijní program: B1103 Aplikovaná matematika
Studijní obor: Matematika–ekonomie se zaměřením na bankovnictví/pojišťovnictví
Forma studia: prezenční
Rok odevzdání: 2017

BIBLIOGRAFICKÁ IDENTIFIKACE

Autor: Hana Zedková

Název práce: Interpolace v R^2

Typ práce: Bakalářská práce

Pracoviště: Katedra matematické analýzy a aplikací matematiky

Vedoucí práce: RNDr. Jitka Machalová, Ph.D.

Rok obhajoby práce: 2017

Abstrakt: V této práci se budeme zabývat problematikou interpolace v $\mathbb{R}^2$, kterou si ukážeme na příkladech. Dále budou předvedeny v matematickém software Matlab kódy pro výpočet koeficientů interpolačního polynomu $P_{nm}(x, y)$.

Klíčová slova: Lagrangeův interpolační polynom, fundamentální polynom, podmínky interpolace, uzly interpolace, interpolační polynom $P_{nm}(x, y)$, interpolační polynom $P_{11}(x, y)$ na trojúhelníku.

Počet stran: 42

Počet příloh: 0

Jazyk: Český

BIBLIOGRAPHICAL IDENTIFICATION

Author: Hana Zedková

Title: Interpolation in R^2

Type of thesis: Bachelor's

Department:
Department of Mathematical Analysis and Application of Mathematics

Supervisor: RNDr. Jitka Machalová, Ph.D.

The year of presentation: 2017

Abstract: In this bachelor thesis we study the problem of interpolation in $\mathbb{R}^2$, which is shown it in the examples. Then in software Matlab are demonstrated mathematical codes to calculate the interpolation polynomial coefficients $P_{nm}(x, y)$.

Key words: Lagrange interpolation polynomial, fundamental polynomial, interpolation conditions, interpolation nodes, interpolation polynomial $P_{nm}(x, y)$, interpolation polynomial $P_{11}(x, y)$ on a triangle.

Number of pages: 42

Number of appendices: 0

Language: Czech

Prohlášení

Prohlašuji, že jsem vypracovala tuto bakalářskou práci samostatně pod vedením RNDr. Jitky Machalové, Ph.D., za použití uvedené literatury.

V Olomouci dne 5.5.2017 ....

podpis

Poděkování

Ráda bych na tomto místě poděkovala především vedoucí mé bakalářské práce RNDr. Jitce Machalové, Ph.D. za odborné vedení mé bakalářské práce, za cenné rady a za čas, který mi věnovala při konzultacích. Také bych chtěla poděkovat mé rodině za podporu při studiu.

Obsah

Úvod

Cílem této bakalářské práce je seznámit čtenáře s teorií polynomiální interpolace v $\mathbb{R}^2$. Následně na příkladech ukázat její využití a nastínit, jak se pomocí matematického softwaru Matlab dají vypočítat koeficienty interpolačního polynomu a jaké funkce pro interpolaci Matlab nabízí. Tato práce je rozdělena do dvou kapitol, přičemž první se dělí na tři podkapitoly.

V úvodu mé práce se čtenář seznamuje s úlohou interpolace. Pro lepší pochopení problematiky je nejprve zmíněna teorie interpolace v $\mathbb{R}$. Je zde zadefinován Lagrangeův interpolačního polynom, který je dále využit v interpolaci $\mathbb{R}^2$. Poté je ukázán příklad vhodný pro ruční počítání a mnou vytvořený m-soubor v Matlabu, který počítá koeficienty Lagrangeova interpolačního polynomu a také vykreslí graf tohoto polynomu. Následně je práce věnována interpolaci v $\mathbb{R}^2$. Nejprve na obdélníkové síti, kde je uvedena konstrukce interpolačního polynomu, která je odvozena z teorie pro interpolaci v $\mathbb{R}$. Konstrukce interpolačního polynomu v $\mathbb{R}^2$ je opět ukázána na příkladu pro ruční počítání a také je uveden mnou vytvořený m-soubor. V závěru první kapitoly se práce věnuje interpolaci v $\mathbb{R}^2$ na trojúhelníku, kde je nejdříve uveden interpolační polynom pro referenční trojúhelník a pomocí transformací je uveden interpolační polynom pro obecný trojúhelník. Transformace jsou v celé práci ukázány vždy na příkladech.

Druhá kapitola čtenáře seznamuje s funkcemi, které se dají použít pro interpolaci v matematickém softwaru Matlab. Ve funkcích jsou pro srovnání vloženy data z příkladů, které se počítaly v první kapitole.

1. Úloha interpolace

Zavedené pojmy v úvodu této kapitoly poskytují lepší pochopení celé problematiky a byly čerpány z [2]. První podkapitola se věnuje interpolaci v $\mathbb{R}$, kde je interpolační polynom konstruován dle Lagrengeova interpolačního polynomu pomocí fundamentálních polynomů. Teorii jsem nastudovala z [2] a [4]. Druhá podkapitola se týká interpolace v $\mathbb{R}^2$, kde byla problematika čerpána z [3]. Poslední podkapitolu jsem konzultovala s vedoucí své bakalářské práce.

Interpolace je jedním z možných způsobů aproximace funkce, kdy hledáme funkci, která v daných bodech nabývá předepsaných hodnot. Tuto funkci nazýváme interpolační funkce.

Jedním z typů interpolace je lineární, kde hledaná funkce je ve tvaru

$$\psi(x; a_0, \dots, a_n) = a_0\psi_0(x) + \cdots + a_n\psi_n(x),$$

$\psi(x; a_0, \dots, a_n)$ je třída funkcí jedné proměnné, kde $\psi_0(x), \dots, \psi_n(x)$ jsou dané funkce a $a_0, \dots, a_n$ jsou parametry. Tyto parametry popisují jednotlivé funkce této třídy.

Úlohu interpolace vnímáme ve dvou případech u kterých máme dány body $x_0, x_1, \dots, x_n$, $x_i \neq x_k, i \neq k, i, k = 0, 1, \dots, n$.

- Můžeme mít funkci $f(x)$ zadanou složitější formou. Tuto funkci $f(x)$ aproximujeme jednodušší funkcí $\psi(x)$ tak, že najdeme hodnoty parametrů $a_0, .., a_n$ tak, aby

  $$\psi(x_i; a_0, \dots, a_n) = f(x_i), \qquad \forall i = 0, 1, \dots, n.$$

  Dále budeme používat značení $f_i = f(x_i), \forall i = 0, 1, \dots, n$.

- V druhém případě máme pro každý bod x_i danou hodnotu f_i a hledáme pro funkci $\psi(x; a_0, \dots, a_n)$ parametry $a_0, \dots, a_n$ aby platilo

$$\psi(x_i; a_0, \ldots, a_n) = f_i, \qquad \forall i = 0, 1, \ldots, n.$$

Vidíme, že úloha interpolace je dána tím, že hledáme funkci $\psi(x)$ tak, aby byly splněny tzv. podmínky interpolace:

$$\psi(x_i; a_0, \ldots, a_n) = f_i, \qquad \forall i = 0, 1, \ldots, n.$$

V práci se budeme zabývat polynomiální interpolací, která patří mezi nejjednodušší úlohy lineární interpolace.

1.1. Polynomiální interpolace v $\mathbb{R}$

U polynomiální interpolace v $\mathbb{R}$ předpokládáme, že máme dány body $x_i \in \mathbb{R}$, pro $i \neq j$, $i, j = 0, 1, ..., n$, které nazýváme uzly interpolace. Tyto hodnoty jsou na ose x a k nim máme dané funkční hodnoty f_i, popřípadě si funkční hodnoty dopočítáme pomocí zadané funkce $f(x)$. Hodnoty $[x_i, f_i]$ lze zapsat do tabulky.

x_i	x_0	$\ldots$	x_n
f_i	f_0	$\ldots$	f_n

Definice 1.1. *Pro dané body* $[x_i, f_i], i = 0, 1, ..., n$ *polynom* $P_n(x)$ *takový, že* $P_n(x_i) = f_i, \forall i = 0, 1, ..., n$ *nazýváme interpolační polynom.*

Třídu všech polynomů stupně nejvýše n budeme značit symbolem π_n. Interpolační polynom musíme hledat tak, aby splňoval podmínky interpolace:

$$P_n(x_i) = f_i \qquad \forall i = 0, 1, ..., n. \tag{1}$$

Naším úkolem je najít interpolační polynom ve tvaru

$$P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0. \tag{2}$$

Interpolační polynom existuje právě jeden za předpokladu $x_i \neq x_k$ pro $i \neq k$, $i, k = 0, 1, ..., n$, viz následující věta.

Věta 1.1. *Pro* $(n+1)$ *daných dvojic čísel* $(x_i, f_i), x_i \neq x_k$, *pro* $i \neq k$, $i, k = 0, 1, ..., n$ *existuje právě jeden polynom* $P_n(x) \in \pi_n$ *takový, že*

$$P_n(x_i) = f_i, \qquad \forall i = 0, 1, ..., n.$$

Důkaz: viz [2] strana 160.

Důkaz předchozí věty je veden tak, že se zkonstruuje Lagrangeův interpolační polynom, který má tvar

$$P_n(x) = l_0(x)f_0 + l_1(x)f_1 + ... + l_n(x)f_n = \sum_{i=0}^{n} l_i(x)f_i \tag{3}$$

a ukáže se, že tento polynom splňuje podmínky interpolace (1).
Polynomy $l_i(x), i = 0, 1, ..., n$ se nazývají Lagrangeovy fundamentální polynomy a jsou definované vztahem

$$l_i(x) = \frac{(x - x_0)\cdots(x - x_{i-1})(x - x_{i+1})\cdots(x - x_n)}{(x_i - x_0)\cdots(x_i - x_{i-1})(x_i - x_{i+1})\cdots(x_i - x_n)}. \tag{4}$$

Tyto polynomy lze po roznásobení psát ve tvaru:

$$l_i(x) = a_n^i x^n + \cdots + a_1^i x + a_0^i. \tag{5}$$

Způsobů, jak najít interpolační polynom, je více. Kromě Lagrangeovy interpolace existuje například i metoda neurčitých koeficientů, Newtonův interpolační polynom, iterovaná interpolace...

Na následujícím příkladu si ukážeme, jak nalézt Lagrangeův interpolační polynom.

Příklad 1.1 Pro data v tabulce nalezněte Lagrangeův interpolační polynom.

x_i	0	1	-1
f_i	-1	1	2

Nejprve se podíváme na stupeň polynomu. Z tabulky je zřejmé, že $i = 0, 1, 2$, tedy $n = 2$. Proto hledáme interpolační polynom $P_2(x)$. Dále určíme fundamentální polynomy podle vztahu (4).

$$l_0(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} = \frac{(x-1)(x+1)}{(0-1)(0+1)} = \frac{x^2-1}{(-1)} = -(x^2-1),$$
$$l_1(x) = \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} = \frac{(x-0)(x+1)}{(1-0)(1+1)} = \frac{x(x+1)}{2} = \frac{x^2+x}{2},$$
$$l_2(x) = \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} = \frac{(x-0)(x-1)}{(-1-0)(-1-1)} = \frac{x(x-1)}{2} = \frac{x^2-x}{2}.$$

Podle vzorce (3) sestavíme Lagrangeův interpolační polynom

$$\begin{aligned} P_2(x) &= f(x_0)l_0(x) + f(x_1)l_1(x) + f(x_2)l_2(x) = (-1)\left[-(x^2-1)\right] + \\ &\quad +1\cdot\frac{x^2+x}{2} + 2\cdot\frac{x^2-x}{2} = x^2+1+\frac{1}{2}\cdot x^2 + \frac{1}{2}\cdot x + x^2 - x = \\ &= \frac{5}{2}\cdot x^2 - \frac{1}{2}\cdot x - 1. \end{aligned}$$

Nyní si ověříme, zda jsou splněny podmínky interpolace (1). Ověření se provede jednoduchým dosazením zadaných uzlů do předpisu $P_2(x)$.

$$\begin{aligned} P_2(x_0) &= \frac{5}{2}\cdot 0^2 - \frac{1}{2}\cdot 0 - 1 = -1 \\ P_2(x_1) &= \frac{5}{2}\cdot 1^2 - \frac{1}{2}\cdot 1 - 1 = 1 \\ P_2(x_2) &= \frac{5}{2}\cdot (-1)^2 - \frac{1}{2}\cdot (-1) - 1 = 2. \end{aligned}$$

Interpolační podmínky jsou splněny.

S využitím matematického softwaru Matlab lze zkonstruovat m-soubor, který pro zadaná data najde Lagrangeův interpolační polynom. V následujícím textu se podíváme na m-soubor, který jsem sama zkonstruovala. Nejdříve vysvětlím, jaké funkce jsem v něm použila. Dále do tohoto m-souboru aplikuji výše uvedený příklad k porovnání výsledku. Na vstupu se zadávají uzly x_i do vektoru x. Funkční hodnoty f_i do sloupcového vektoru f. Ve výstupu bude řádkový vektor koeficientů

Lagrangeova interpolačního polynomu. Tento vektor bude koeficienty seřazovat sestupně dle tvaru (2), tj. $[a_n, \ldots, a_0]$.

Následně si představíme, jak kód v m-souboru vypadá.

```
function [a]=lagrange(x,f)

% vstup      x ... vektor uzlu interpolace
%            f ... vektor funkcnich hodnot
% vystup     a ... vektor koeficientu
%                  interpolacniho polynomu

%overeni dat
n=length(x);

if n~= length(f)
    error('Vstupni vektory nejsou stejne dlouhe.')
end

for j=1:n
    for i=j+1:n
        if x(j)==x(i);
    error('Uzly musi byt vzajemne ruzne.')
        end
    end
end

%vypocet vektoru koeficientu
X=zeros(n,n);
for i=1:n;
    v=1;
    for j=1:n;
        if i~=j;
            z=poly(x(j))
            v = conv(v,z)/(x(i)-x(j))
        end
    end
    X(i,:)=v
end
a = f' * X
```

```
%vykresleni grafu
rozsah = max(x) - min(x);
krok = rozsah/100;
xx=[min(x) - krok :krok: max(x) + krok];
yy=polyval(a,xx);
plot(x,f,'ro',xx,yy,'b')
```

Na začátku této kapitoly bylo představeno, že pro konstrukci Lagrangeova interpolačního polynomu $P_n(x)$ je třeba výpočet fundamentálních polynomů $l_i(x)$, $i = 0, 1, ..., n$ daných vztahem (4), resp. (5) .

V Matlabu se pro nalezení koeficientů $a_n^i, \ldots, a_0^i$ fundamentálního polynomu $l_i(x)$ musí pracovat s dvěma funkcemi. Nejdříve si představíme funkci $poly(x)$:

$$z = poly(x(j))$$

Tato funkce v daném m-souboru pracuje ve for cyklu, ve kterém je daná podmínka $x_i \neq x_j$, kde $i \neq j, i, j = 0, 1, ..., n$. Funkce $poly(x(j))$ vytváří koeficienty polynomu, jehož kořeny jsou zadané ve vektoru x. V našem případě postupně vytvoří čitatele fundamentálního polynomu $l_i(x)$, respektive koeficienty polynomu $(x - x_0) \cdots (x - x_{i-1})(x - x_{i+1}) \cdots \ (x - x_n)$. Pro objasnění si vezměme z Příkladu 1.1 fundamentální polynom $l_0(x)$, na kterém si předvedeme výpočet, který nám funkcí $poly(x(j))$ vznikne. Pro uzel x_0 máme fundamentální polynom ve tvaru

$$l_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} = \frac{(x - 1)(x + 1)}{(0 - 1)(0 + 1)}.$$

Funkce $poly(x(j))$ nejdříve vezme uzel $x_1 = 1$, tedy $poly(1)$, a do vektoru z vloží sestupně koeficienty polynomu $(x - x_1) = (x - 1)$, tedy

$z =$ 1 -1,

poté pro uzel $x_2 = -1$, $poly(-1)$, vytvoří koeficienty polynomu $(x-x_2) = (x+1)$, tedy

$z =$ 1 1.

Dále v tomto for cyklu pracuje funkce $conv(v, z)$, která umožňuje součin dvou polynomů v, z, kde v je na začátku rovno jedné.

$$v = conv(v, z)/(x(i) - x(j)),$$

Jedná se tedy o součin polynomů $(x - x_0) \cdots (x - x_{i-1})(x - x_{i+1}) \cdots\ (x - x_n)$ a pak o vydělení konstantou $(x_i - x_0) \cdots (x_i - x_{i-1})(x_i - x_{i+1}) \cdots (x_i - x_n)$, která představuje jmenovatele fundamentálního polynomu $l_i(x)$.

Pro $l_0(x)$ z Příkladu 1.1. je konstanta $(x_0 - x_1)(x_0 - x_2) = (0 - 1)(0 + 1) = -1$ a m-soubor pracuje následovně:

$v = [1\,]$
$z = [1 \quad -1\,]$ %koeficienty polynomu (x-1)
$v = [-1 \quad 1\,]$ %koefcienty polynomu 1(x-1)/(-1)
$z = [1 \quad 1\,]$ %koeficienty polynomu (x + 1)
$v = [-1 \quad 0 \quad 1\,]$ %koeficienty polynomu (x-1)(x+1)/(-1).

Vektory v jsou postupně vytvářeny pro každý uzel $x_i, i = 0, 1, ..., n$. Poté jsou zadávány do matice X po řádcích. Tedy dle (5) jsou v $i-tém$ řádku $i = 0, 1, \quad , n$ matice X koeficienty lagrangeova interpolačního polynomu $l_i(x)$, tj.

$$X = \begin{pmatrix} a_n^0 & a_{n-1}^0 & \dots & a_1^0 & a_0^0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_n^n & a_{n-1}^n & \dots & a_1^n & a_0^n \end{pmatrix}.$$

K vytvoření koeficientů Lagrangeova interpolačního polynom $P_n(x)$, je třeba podle tvaru (3) sloupcový vektor $f = \begin{pmatrix} f_0 & f_1 & \dots & f_n \end{pmatrix}^T$, který představuje předepsané funkční hodnoty, vynásobit maticí X.

$$\begin{aligned} a_n &= f_0 \cdot a_n^0 + f_1 \cdot a_n^1 + \cdots + f_n \cdot a_n^n \\ &\vdots \\ a_0 &= f_0 \cdot a_0^0 + f_1 \cdot a_0^1 + \cdots + f_n \cdot a_0^n. \end{aligned}$$

Celkový součin vypadá následovně:

$$a = f^T \cdot X = (f_0 \qquad f_1 \qquad \dots \qquad f_n\,) \cdot \begin{pmatrix} a_n^0 & a_{n-1}^0 & \dots & a_1^0 & a_0^0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_n^n & a_{n-1}^n & \dots & a_1^n & a_0^n \end{pmatrix}.$$

V následujícím příkladu uvidíme daný m-soubor pracovat a porovnáme, zda se nám výsledky shodují s ručním výpočtem.

Příklad 1.2 Pro stejná data jako v Příkladu 1.1 nalezněte pomocí uvedeného m-souboru Lagrangeův interpolační polynom. Data byla dána následující tabulkou

x_i	0	1	-1
f_i	-1	1	2

Do příkazového okna Matlabu zadáme vstupy:

```
x=[0; 1; -1];
f=[-1; 1; 2];
[a]=lagrange(x,f)
```

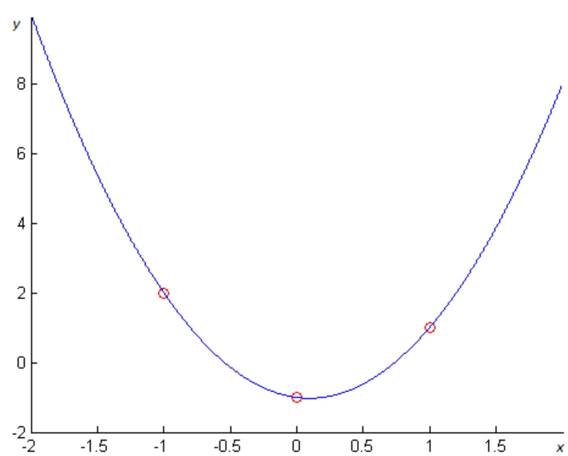
a zavoláním této funkce získáme

```
a =

    2.5000    -0.5000    -1.0000
```

čímž jsme získali stejný výsledek jako při ručním výpočtu.

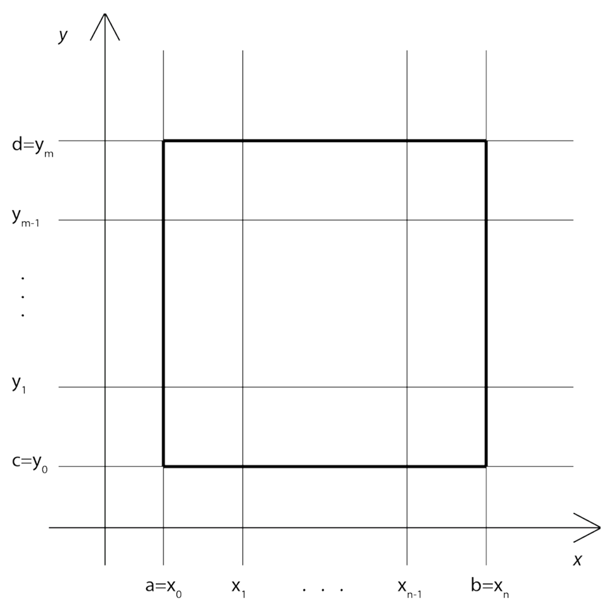
Graf polynomu spolu s body interpolace, které jsou značené červeným kroužkem, je zobrazen na obrázku 1.



Obrázek 1: Graf interpolačního polynomu $P_2(x)$

1.2. Polynomiální interpolace v $\mathbb{R}^2$ na obdélníkové síti

V problematice dvourozměrné interpolace pomocí Lagrangeova interpolačního polynomu vyjdeme z poznatků o interpolaci funkce jedné proměnné. Máme funkci $f(x, y)$, která je spojitá na obdélníku $D = \{[x, y]\,; x \in [a, b]\,, y \in [c, d]\}, D \in \mathbb{R}^2$. V intervalu $[a, b]$ si zvolíme uzly x_i, pro $i = 0, 1, ..., n$, takové, že $a = x_0 < x_1 < ... < x_n = b$, kde $n \in \mathbb{N}$. Analogicky v intervalu $[c, d]$ si zvolíme uzly y_j, kde $j = 0, 1, ..., m$ tak, že $c = y_0 < y_1 < ... < y_m = d$, kde $m \in \mathbb{N}$. Tím se na obdélníku D vytvoří pravoúhlá síť uzlů (x_i, y_j).



Obrázek 2: Pravoúhlá síť uzlů

Úloha interpolace v $\mathbb{R}^2$ na obdélníkové síti je obdobná jako v $\mathbb{R}$. Úkolem je najít interpolační polynom, který v předepsaných uzlech (x_i, y_j) bude nabývat předepsaných hodnot $f(x_i, y_j)$, tj. musí být splněny podmínky interpolace

$$P_{nm}(x_i, y_j) = f(x_i, y_j) \qquad \forall i = 0, \ldots, n, \forall j = 0, 1, \ldots, m. \tag{6}$$

Dále budeme značit $f_{ij} = f(x_i, y_j), i = 0, 1, ..., n, j = 0, 1, ..., m$. Interpolační polynom $P_{nm}(x, y)$ má v proměnné x nejvýše n-tý stupeň a v proměnné y nejvýše m-tý stupeň. V následujících příkladech budeme zadávat hodnoty způsobem, který je uveden v následující tabulce.

	y_0	y_1	$\dots$	y_m
x_0	f_{00}	f_{01}	$\dots$	f_{0m}
x_1	f_{10}	f_{11}	$\dots$	f_{1m}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
x_n	f_{n0}	f_{n1}	$\dots$	f_{nm}

Polynom $P_{nm}(x, y)$ budeme zapisovat následovně

$$P_{nm}(x, y) = a_{nm}x^n y^m + \dots + a_{1m}x^1 y^m + a_{0m}y^m + \dots + a_{n1}x^n y + \dots + a_{11}xy + \\ + a_{01}y + a_{n0}x^n + \dots + a_{10}x + a_{00}. \tag{7}$$

Systém uspořádání členů polynomu $P_{nm}(x, y)$ je odvozen právě ze zadání funkčních hodnot f_{ij}, které je uvedeno v tabulce níže

a_{00}	a_{01}	$\dots$	a_{0m}
a_{10}	a_{11}	$\dots$	a_{1m}
$\vdots$	$\vdots$	$\ddots$	$\vdots$
a_{n0}	a_{n1}	$\dots$	a_{nm}

a tedy vektor koeficientů

$$a = \begin{pmatrix} a_{nm} & \dots & a_{1m} & a_{0m} & \dots & a_{n1} & \dots & a_{11} & a_{01} & a_{n0} & \dots & a_{10} & a_{00} \end{pmatrix}^T. \tag{8}$$

Princip konstrukce interpolačního polynomu $P_{nm}(x, y)$ spočívá v tom, že si nejdříve zkonstruujeme Lagrangeovy interpolační polynomy $P(x, y_j)$ na přímkách $y = y_j, j = 0, 1, \dots, m$. Lagrangeův interpolační polynom vypadá takto

$$P(x, y_j) = \sum_{i=0}^{n} l_i(x) f_{ij}. \tag{9}$$

Potom pomocí funkcí $P(x, y_j)$, $j = 0, 1, \dots, m$ zkonstruujeme $P_{nm}(x, y)$, tedy

$$P_{nm}(x, y) = \sum_{j=0}^{m} l_j(y) P(x, y_j) = \sum_{j=0}^{m} l_j(y) \sum_{i=0}^{n} l_i(x) f_{ij} = \\ = \sum_{i=0}^{n} \sum_{j=0}^{m} l_i(x) l_j(y) f_{ij}. \tag{10}$$

Dále budeme značit $l_{ij}(x,y) = l_i(x)l_j(y)$.

Na následujícím příkladu si konstrukci polynomu $P_{nm}(x,y)$ přiblížíme.

Příklad 1.3 Pro zadaná data na oblasti $D = \{[x, y\,]\colon x \in [0, 1\,], y \in [2, 3\,]\}$ nalezněte interpolační polynom v $\mathbb{R}^2$.

	2	3
0	-1	2
1	3	-2

Z tabulky lze snadno určit maximální stupeň polynomu. Vidíme, že jak u proměnné x, tak u proměnné y jsou dva body, tudíž $n = 1$ a $m = 1$. Proto budeme hledat interpolační polynom nejvýše $P_{11}(x,y)$, tedy lineární v obou proměnných.
Pro upřesnění uvedeme tvar interpolačního polynomu, který hledáme

$$P_{11}(x,y) = a_{11}xy + a_{01}y + a_{10}x + a_{00}.$$

Pomocí fundamentálních polynomů, vzorec (4), vypočítáme $l_0(x)$, $l_1(x)$, $l_0(y)$, $l_1(y)$:

$$\begin{aligned}
l_0(x) &= \frac{x - x_1}{x_0 - x_1} = \frac{x-1}{(-1)} = -(x-1) = -x+1,\\
l_1(x) &= \frac{x - x_0}{x_1 - x_0} = x,\\
l_0(y) &= \frac{y - y_1}{y_0 - y_1} = \frac{y-3}{(-1)} = -(y-3) = -y+3,\\
l_1(y) &= \frac{y - y_0}{y_1 - y_0} = \frac{y-2}{(1)} = y-2.
\end{aligned}$$

Dále podle vztahů (9) a (10) je

$$\begin{aligned}
P_{11}(x,y) &= \sum_{i=0}^{1} l_j(y)P(x,y_j) = l_0(y)P(x,y_0) + l_0(y)P(x,y_1) =\\
&= l_0(y)l_0(x)f_{00} + l_0(y)l_1(x)f_{10} + l_1(y)l_0(x)f_{01} + l_1(y)l_1(x)f_{11}
\end{aligned}$$

$$
\begin{aligned}
&= (-y+3)(-x+1)(-1) + (-y+3)x3 + (y-2)(-x+1)\\
&\quad +(y-2)x(-2) = (xy-y-3x+3)(-1) - 3xy +\\
&\quad +9x + (-xy+y+2x-2) - 2y + 4 = -xy + y + 3x - 3 +\\
&\quad -3xy + 9x - 2xy + 2y + 4x - 4 - 2xy + 4x = -8xy + 3y + 20x - 7.
\end{aligned}
$$

Tím jsme našli hledaný interpolační polynom a opět můžeme ověřit, zda jsou splněny podmínky interpolace (6).

$$
\begin{aligned}
P_{11}(x_0, y_0) &= -8 \cdot 0 \cdot 2 + 3 \cdot 2 + 20 \cdot 0 - 7 = -1\\
P_{11}(x_1, y_0) &= -8 \cdot 1 \cdot 2 + 3 \cdot 2 + 20 \cdot 1 - 7 = 3\\
P_{11}(x_0, y_1) &= -8 \cdot 0 \cdot 3 + 3 \cdot 3 + 20 \cdot 0 - 7 = 2\\
P_{11}(x_1, y_1) &= -8 \cdot 1 \cdot 3 + 3 \cdot 3 + 20 \cdot 1 - 7 = -2.
\end{aligned}
$$

Podmínky interpolace jsou splněny.

Ukázali jsme si postup výpočtu interpolačního polynomu, který spíše vyhovuje při ručním počítání. Tento přístup není vhodný do m-souboru v softwaru Matlab. Výpočet se konstruuje pomocí matic. V kódu se objevují stejné funkce, které byly použity při výpočtu interpolačního polynomu s jednou proměnnou. Abychom porozuměli, jak soubor s maticemi pracuje, tak si matice nejdříve obecně zadefinujeme. Poté si na Příkladu 1.3 ukážeme, jak m-soubor pracuje a porovnáme výsledky. Na vstupu zadáme pro každou proměnnou vektor uzlů interpolace. Dále zadáme matici funkčních hodnot. Do řádků se zapisují funkční hodnoty po přímkách x_i, tzn.

$$
F = \begin{pmatrix} f_{00} & f_{01} & \dots & f_{0m}\\ f_{10} & f_{11} & \dots & f_{1m}\\ \vdots & \vdots & \ddots & \vdots\\ f_{n0} & f_{n1} & \dots & f_{nm} \end{pmatrix} \tag{11}
$$

Výstupem bude vektor koeficientů interpolačního polynomu. Tento vektor bude koeficienty seřazovat sestupně dle tvaru (8),
tj. $(a_{nm}, \dots, a_{1m}, a_{0m}, \dots, a_{n1}, \dots, a_{11}, a_{01}, a_{n0}, \dots, a_{10}, a_{00})$.

Nyní si představíme m-soubor, který má ve vstupu dvě proměnné.

Text m-souboru v MATLABu:

```
function [a]=lagrange_n(x,y,F)

% vstup      x ... vektor hodnot na ose x
%            y ... vektor hodnot na ose y
%            F ... matice funkcnich hodnot
% vystup     a ... vektor koeficientu interpolacniho
%                  polynomu

%overeni dat
n=length(x);
m=length(y);
[b] = size(F);
if b(:,1)~=length(x)
   error('Velikost matice neodpovida vstupnim vektorum.');
end

if b(:,2)~= length(y)
   error('Velikost matice neodpovida vstupnim vektorum.');
end

for j=1:n
    for i=j+1:n
        if x(j)==x(i);
    error('Uzly musi byt vzajemne ruzne.')
        end
    end
end

for s=1:m
    for r=s+1:m
        if y(s)==y(r);
    error('Uzly musi byt vzajemne ruzne.')
        end
    end
end

%vypocet vektoru koeficientu
X=zeros(n,n);
for i=1:n;
```

```
        v=1;
        for j=1:n;
            if i~=j;
                z=poly(x(j));
                v = conv(v,z)/(x(i)-x(j));
            end
        end
        X(i,:)=v;
end
Y=zeros(m,m);
for s=1:m
        t=1;
        for r=1:m
            if  s~=r
                u=poly(y(r));
                t = conv(t,u)/(y(s)-y(r));
            end
        end
        Y(s,:)=t;

end
P = F.' * X;
IP = P.';
A = IP * Y;
a = A(:)'

%vykresleni grafu
X = min(x):0.1:max(x);
Y = min(y):0.1:max(y);
[xx,yy] = meshgrid(X,Y);
ex=linspace(n-1,0,n);
dx=xx(1,:);
for i=1:length(dx);
        vx=dx(i);
        vxx=vx.^ex;
        Vx(i,:)=vxx;
end
ey=linspace(m-1,0,m);
dy=yy(:,1);
for s=1:length(dy);
        vy=dy(s);
        vyy=vy.^ey;
        Vy(:,s)=vyy;
```

```
end
P=Vx*A*Vy;
plot(x,y,F,'ko')
surf(xx,yy,P')
```

V m-souboru se nejdříve zkonstruují matice X a Y, jejichž řádky reprezentují koeficienty fundamentálních polynomů $l_i(x), i = 0, 1, \cdots, n$ a $l_j(y), j = 0, 1, \cdots, m$. Tyto koeficienty jsme si představili v kapitole 1.1 na straně 11, tedy $a_n^i, \cdots, a_0^i$ jsou koeficienty polynomu $l_i(x)$ a analogicky $b_m^j, \cdots, b_0^j$ koeficienty polynomu $l_j(y)$, tedy

$$X = \begin{pmatrix} a_n^0 & \dots & a_0^0 \\ \vdots & \ddots & \vdots \\ a_n^n & \dots & a_0^n \end{pmatrix}, \quad Y = \begin{pmatrix} b_m^0 & \dots & b_0^0 \\ \vdots & \ddots & \vdots \\ b_m^m & \dots & b_0^m \end{pmatrix}.$$

Poté budeme potřebovat matici F danou vztahem (11), která obsahuje funkční hodnoty f_{ij}.

Na příkladu si ukážeme, jak funguje m-soubor, který budeme v následujícím textu používat. Mějme stejné zadání jako u Příkladu 1.3. Nejdříve si vytvoříme matice X a Y, které představují koeficienty fundamentálních polynomů $l_i(x)$ a $l_j(y)$. Fundamentální polynomy pro proměnné x a y jsme si v Příkladu 1.3. již spočítali a tedy

$$\begin{aligned} l_0(x) &= -(x-1) = -x+1, & l_0(y) &= -y+3, \\ l_1(x) &= x, & l_1(y) &= y-2. \end{aligned}$$

Pro tyto polynomy jsou matice X a Y ve tvaru

$$X = \begin{pmatrix} -1 & 1 \\ 1 & 0 \end{pmatrix} \quad Y = \begin{pmatrix} -1 & 3 \\ 1 & -2 \end{pmatrix}.$$

Matice F je tvaru

$$F = \begin{pmatrix} -1 & 2 \\ 3 & -2 \end{pmatrix}.$$

Dle vztahu (9) potřebujeme vytvořit polynomy $P(x, y_j)$.

Matice P bude představovat koeficienty $p_n^j, \cdots, p_0^j$ polynomů $P(x, y_j)$, kde $j = 0, 1, \cdots, m$

$$P = \begin{pmatrix} p_n^0 & p_{n-1}^0 & \cdots & p_0^0 \\ p_n^1 & p_{n-1}^1 & \cdots & p_0^1 \\ \vdots & \vdots & \ddots & \vdots \\ p_n^m & p_{n-1}^m & \dots & p_0^m \end{pmatrix}$$

Tuto matici vytvoříme vztahem $P = F^T \cdot X$, tj.

$$\begin{pmatrix} p_n^0 & p_{n-1}^0 & \cdots & p_0^0 \\ p_n^1 & p_{n-1}^1 & \cdots & p_0^1 \\ \vdots & \vdots & \ddots & \vdots \\ p_n^m & p_{n-1}^m & \dots & p_0^m \end{pmatrix} = \begin{pmatrix} f_{00} & f_{01} & \cdots & f_{0m} \\ f_{10} & f_{11} & \cdots & f_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ f_{n0} & f_{n1} & \cdots & f_{nm} \end{pmatrix}^T \begin{pmatrix} a_n^0 & a_{n-1}^0 & \cdots & a_0^0 \\ a_n^1 & a_{n-1}^1 & \cdots & a_0^1 \\ \vdots & \vdots & \ddots & \vdots \\ a_n^n & a_{n-1}^n & \cdots & a_0^n \end{pmatrix}$$

a tedy

$$\begin{aligned} p_n^0 &= f_{00} * a_n^0 + f_{10} * a_n^1 + \cdots + f_{n0} * a_n^n \\ &\vdots \\ p_0^m &= f_{0m} * a_0^0 + f_{1m} * a_0^1 + \cdots + f_{nm} * a_0^n. \end{aligned}$$

Pro data z Příkladu 1.3. bude matice P vypadat následovně:

$$P = \begin{pmatrix} 4 & -1 \\ -4 & 2 \end{pmatrix}.$$

Interpolační polynom $P_{mn}(x, y)$ si odvodíme podle vztahu (10), tj. $P_{nm}(x, y) = \sum_{j=0}^{m} l_j(y) P(x, y_j)$. Vynásobíme transponovanou matici P s maticí Y.

$$P^T Y = \begin{pmatrix} p_n^0 & p_{n-1}^0 & \cdots & p_0^0 \\ p_n^1 & p_{n-1}^1 & \cdots & p_0^1 \\ \vdots & \vdots & \ddots & \vdots \\ p_n^m & p_{n-1}^m & \dots & p_0^m \end{pmatrix}^T \begin{pmatrix} b_m^0 & b_{m-1}^0 & \cdots & b_0^0 \\ b_m^1 & b_{m-1}^1 & \cdots & b_0^1 \\ \vdots & \vdots & \ddots & \vdots \\ b_m^m & b_{m-1}^m & \dots & b_0^m \end{pmatrix} = \begin{pmatrix} a_{00} & a_{01} & \cdots & a_{0m} \\ a_{10} & a_{11} & \cdots & a_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n0} & a_{n1} & \dots & a_{nm} \end{pmatrix} = A$$

Výsledná matice A je tvořena prvky koeficientů $a_{00}, \cdots, a_{nm}$ hledaného interpolačního polynomu $P_{nm}(x, y)$. Princip tohoto násobení je odvozen z principu pro interpolaci v $\mathbb{R}$, tedy

$$\begin{aligned} a_{00} &= p_n^0 \cdot b_m^0 + p_n^1 \cdot b_m^1 + \cdots + p_n^m \cdot b_m^m \\ &\vdots \\ a_{nm} &= p_0^0 \cdot b_0^0 + p_0^1 \cdot b_0^1 + \cdots + p_0^m \cdot b_0^m. \end{aligned}$$

Matice A pro data z Příkladu 1.3. vypadá následovně

$$A = P^T Y = \begin{pmatrix} 4 & -4 \\ -1 & 2 \end{pmatrix} \cdot \begin{pmatrix} -1 & 3 \\ 1 & -2 \end{pmatrix} = \begin{pmatrix} -8 & 20 \\ 3 & -7 \end{pmatrix}$$

Koeficienty interpolačního polynomu jsou v matici A seřazeny dle sloupců. Vektor a bude ve tvaru

$$a = (-8 \qquad 3 \qquad 20 \qquad -7)\,.$$

Příklad 1.4 Pro stejná data jako v Příkladu 1.3 nalezněte pomocí uvedeného m-souboru interpolační polynom. Data byla dána na oblasti $D = \{[x, y]: x \in [0, 1], y \in [2, 3]\}$ následující tabulkou

	2	3
0	-1	2
1	3	-2

Do příkazového okna Matlabu zadáme vstupy:

```
x=[0; 1];
y=[2; 3];
F=[−1 2; 3 −2];
[a]=lagrange_n(x,y,F)
```

Po potvrzení těchto vstupů získáme

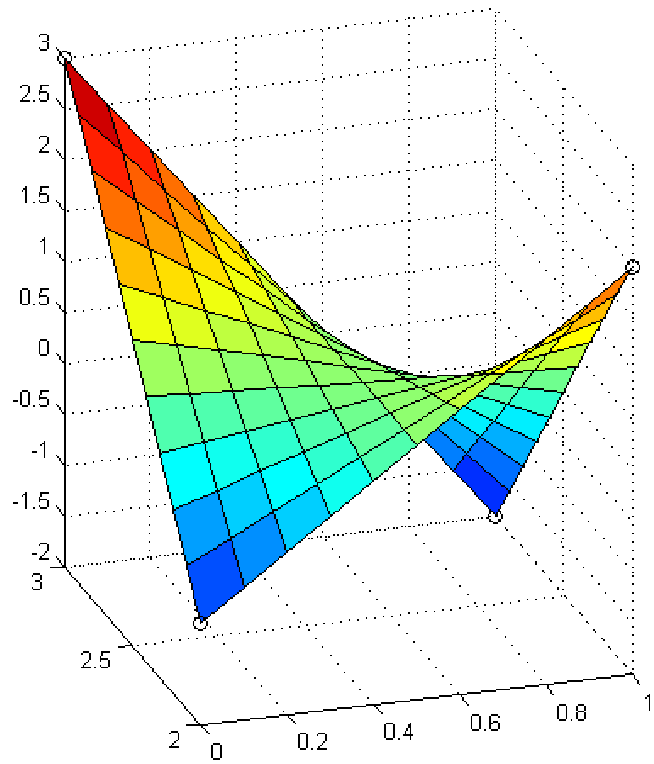
```
a =

   −8     3    20    −7
```

čímž jsme získali stejný výsledek jako při ručním výpočtu. Kód vykreslí graf polynomu, který jsme vypočítali u příkladu 1.4., je zobrazen na obrázku 3.

Příklad 1.5 Pro zadaná data na oblasti $D = \{[x, y]: x \in [-2, 4], y \in [-3, 1]\}$ nalezněte interpolační polynom v R^2.

	-3	-2	1
-2	0	-2	-1
0	1	5	1
1	0	1	1
3	1	-1	-3
4	2	0	0



Obrázek 3: Graf interpolačního polynomu $P_{11}(x)$ pro data z Příkladu 1.4

Nejdříve z tabulky určíme stupeň polynomu. Vidíme, že u proměnné x je pět bodů, tudíž $n = 4$. U proměnné y tři, tedy $m = 2$. Proto budeme hledat interpolační polynom $P_{42}(x, y)$.

Uvedeme tvar interpolačního polynomu, který hledáme

$$P_{42}(x,y) = a_{42}x^4y^2 + a_{32}x^3y^2 + a_{22}x^2y^2 + a_{12}x^1y^2 + a_{02}y^2 + a_{41}x^4y^1 + a_{31}x^3y^1 + \\ +a_{21}x^2y^1 + a_{11}xy + a_{01}y + a_{40}x^4 + a_{30}x^3 + a_{20}x^2 + a_{10}x + a_{00}.$$

Do příkazového okna Matlabu zadáme vstupy:

```
x=[−2 0 1 3 4];
y=[−3 −2 1];
F=[0 −2 −1; 1 5 1; 0 1 1; 1 −1 −3; 2 0 0];
[a]=lagrange_n(x,y,F)
```

Po potvrzení těchto vstupů získáme vektor a dle tvaru (8), tedy koeficienty interpolačního polynomu $P_{42}(x, y)$

a =

0.0407	−0.2704	0.2880	1.0250	−1.3333
0.1204	−0.6852	0.3565	2.6250	−2.6667
−0.0528	0.6722	−1.5861	−2.5333	5.0000

Interpolační polynom bude ve tvaru

$$
\begin{aligned}
P_{42}(x,y) =&0.0407x^4y^2 - 0.2704x^3y^2 + 0.2880x^2y^2 + 1.0250x^1y^2 - 1.3333y^2 + \\
&+ 0.1204x^4y^1 - 0.6852x^3y^1 + 0.3565x^2y^1 + 2.6250xy - 2.6667y - \\
&+ 0.0528x^4 + 0.6722x^3 - 1.5861x^2 - 2.5333x + 5.0000.
\end{aligned}
$$

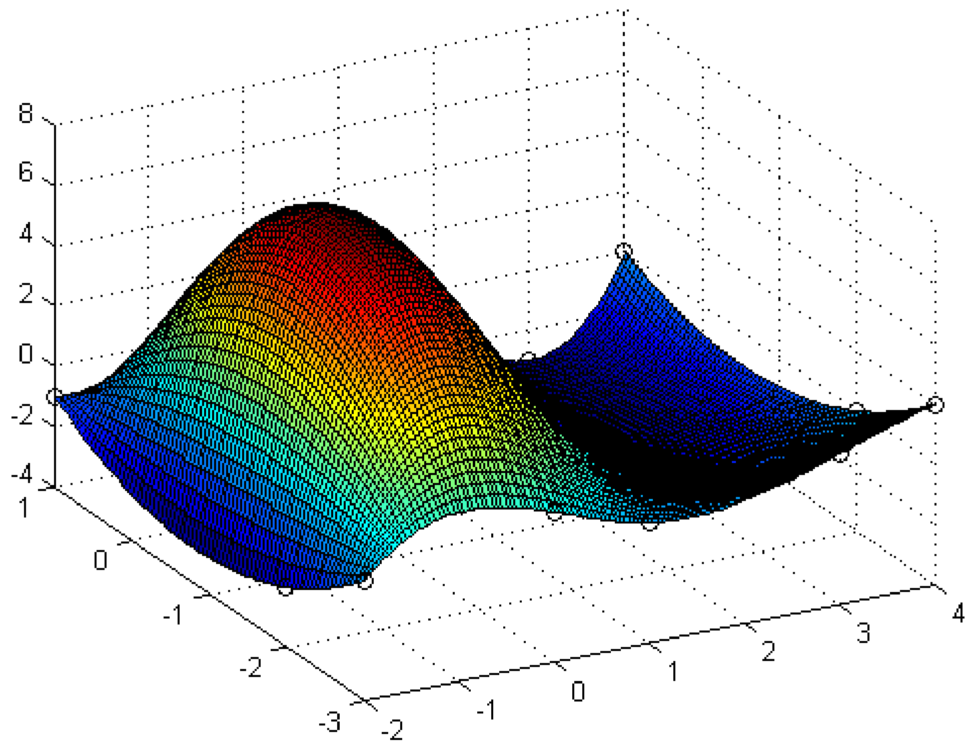
Opět musíme ověřit, zda jsou splněny podmínky interpolace (6). Ověření se provede dosazením do předpisu $P_{42}(x,y)$. Pro ukázku jsou uvedeny některé z nich.

$$
\begin{aligned}
P_{42}(x_0,y_0) = &\ 0.0407(-2)^4(-3)^2 - 0.2704(-2)^3(-3)^2 + 0.2880(-2)^2(-3)^2 + \\
&+1.0250(-2)^1(-3)^2 - 1.3333(-3)^2 + 0.1204(-2)^4(-3)^1 - \\
&+0.6852(-2)^3(-3)^1 + 0.3565(-2)^2(-3)^1 + 2.6250(-2)(-3) - \\
&+2.6667 - 0.0528(-2)^4 + 0.6722(-2)^3 - 1.5861(-2)^2 - 2.5333(-2) + \\
&+5.0000 = 0
\end{aligned}
$$

$$
\begin{aligned}
P_{42}(x_1,y_0) = &\ 0.0407(0)^4(-3)^2 - 0.2704(0)^3(-3)^2 + 0.2880(0)^2(-3)^2 + \\
&+1.0250(0)^1(-3)^2 - 1.3333(-3)^2 + 0.1204(0)^4(-3)^1 - \\
&+0.6852(0)^3(-3)^1 + 0.3565(0)^2(-3)^1 + 2.6250(0)(-3) - \\
&+2.6667 - 0.0528(0)^4 + 0.6722(0)^3 - 1.5861(0)^2 - 2.5333(0) + \\
&+5.0000 = 1
\end{aligned}
$$

Podmínky interpolace jsou splněny.

Graf polynomu z příkladu 1.5 je zobrazen na obrázku 4.



Obrázek 4: Graf interpolačního polynomu $P_{42}(x, y)$ z Příkladu 1.5

1.3. Polynomiální interpolace v $\mathbb{R}^2$ na trojúhelníku

Rozdělení oblasti, na které chceme funkci dvou proměnných interpolovat, může být nejen obdélníkové, ale i trojúhelníkové. Plochu je snadnější pokrýt pomocí trojúhelníků. Máme trojúhelník T se třemi vrcholy $T_k = (x_k, y_k), k = 0, 1, 2$, k nim předepsané funkční hodnoty f_k. Proto hledáme interpolační polynom $P_{11}(x, y)$ ve tvaru

$$P_{11}(x, y) = \alpha x + \beta y + \gamma,$$

kde α, β, γ jsou jeho koeficienty a přitom musí být splněny podmínky interpolace, tj. $P_{11}(x_k, y_k) = f_k$.

Nejdříve si ukážeme interpolační polynom $P_{11}(x, y)$ pro nejjednodušší situaci a to pro referenční trojúhelník, který si označíme T. Referenční trojúhelník T má vrcholy $T_0 = (x_0, y_0) = (0,0)$, $T_1 = (x_1, y_1) = (1,0)$, $T_2 = (x_2, y_2) = (0,1)$, kde platí $x_0 = x_2 = 0, y_0 = y_1 = 0$ a předepsané funkční hodnoty fk v těchto vrcholech.

Chceme mít $P_{11}(x,y)$ tak, aby platilo $P_{11}(x_0, y_0) = f_0$, $P_{11}(x_1, y_1) = f_1$, $P_{11}(x_2, y_2) = f_2$:

$$\begin{array}{lclcl} P_{11}(0,0) & = & \alpha \cdot 0 + \beta \cdot 0 + \gamma = \gamma & = & f_0 \\ P_{11}(1,0) & = & \alpha \cdot 1 + \beta \cdot 0 + \gamma = \alpha + \gamma & = & f_1 \\ P_{11}(0,1) & = & \alpha \cdot 0 + \beta \cdot 1 + \gamma = \beta + \gamma & = & f_2, \end{array}$$

tedy:

$$\begin{aligned} \gamma &= f_0 \\ \alpha &= f_1 - \gamma = f_1 - f_0 \\ \beta &= f_2 - \gamma = f_2 - f_0. \end{aligned}$$

Interpolační polynom $P_{11}(x,y)$ pro referenční trojúhelník T v závislosti na daných f_k bude ve tvaru

$$P_{11}(x,y) = (f_1 - f_0)x + (f_2 - f_0)y + f_0. \tag{12}$$

Na následujícím příkladu si ukážeme, jak nalézt interpolační polynom $P_{11}(x,y)$ pro referenční trojúhelník T.

Příklad 1.6 Nechť je dán referenční trojúhelník T s funkčními hodnotami $f_0 = -3$, $f_1 = 2$, $f_2 = 1$. Pro tyto data nalezněte interpolační polynom $P_{11}(x,y)$.

Koeficienty interpolační polynom jdou vyřešit jednoduchým dosazením do vztahu (12), tedy

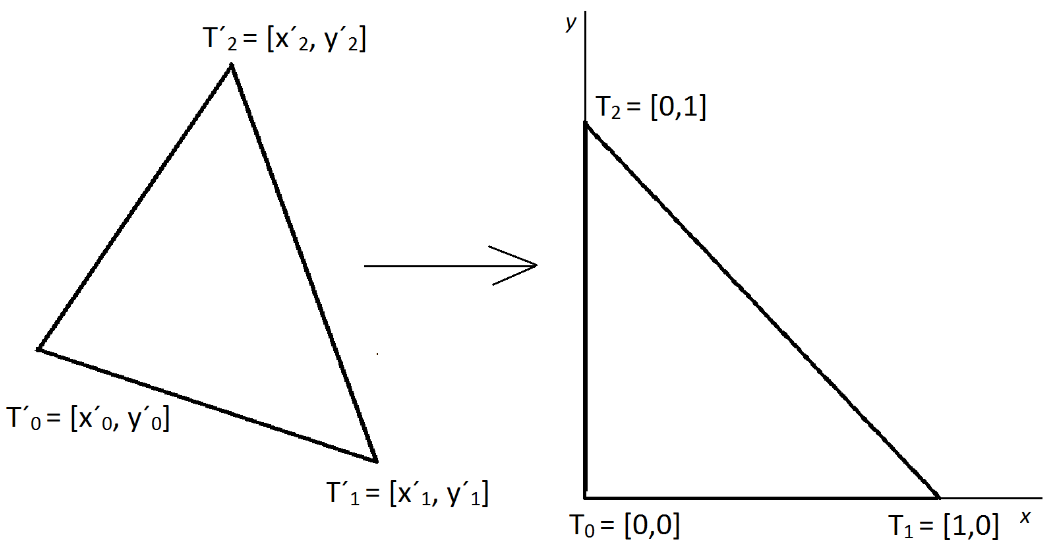
$$\begin{aligned} P_{11}(x,y) &= (f_1 - f_0)x + (f_2 - f_0)y + f_0 = (2+3)x + (1+3)y - 3 = \\ &= 5x + 4y - 3. \end{aligned}$$

Musíme ověřit podmínky interpolace, platí:

$$P_{11}(0,0) = 5 \cdot 0 + 4 \cdot 0 - 3 = -3$$
$$P_{11}(1,0) = 5 \cdot 1 + 4 \cdot 0 - 3 = 2$$
$$P_{11}(0,1) = 5 \cdot 0 + 4 \cdot 1 - 3 = 1.$$

Podmínky interpolace jsou splněny.

V praxi se obvykle vyskytuje obecný trojúhelník $T^{'}$ s vrcholy $T_k^{'} = (x_k^{'}, y_k^{'}), k = 0, 1, 2$, které se dají transformovat na vrcholy $T_0 = (0,0)$, $T_1 = (1,0)$, $T_2 = (0,1)$ referenčního trojúhelníku T, viz. obrázek 5.



Obrázek 5: Transformace obecného trojúhelníku

Pro lineární transformaci použijeme vztah $x^{'} = Ax + b$, který lze maticově zapsat

$$\begin{pmatrix} x^{'} \\ y^{'} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}. \tag{13}$$

Vrcholy referenčního trojúhelníku T dosadíme dle (13) do $\begin{pmatrix} x \\ y \end{pmatrix}$. Za $\begin{pmatrix} x' \\ y' \end{pmatrix}$ dosadíme vrcholy $T_k' = (x_k', y_k')$ obecného trojúhelníka T'.
Dostáváme tedy pro:

$$T_0' = (x_0', y_0'), T_0 = (0,0): \quad \begin{pmatrix} x_0' \\ y_0' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \Rightarrow$$

$$x_0' = b_1$$
$$y_0' = b_2$$

$$T_1' = (x_1', y_1'), T_1 = (1,0): \quad \begin{pmatrix} x_1' \\ y_1' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \Rightarrow$$

$$a_{11} = x_1' - x_0'$$
$$a_{21} = y_1' - y_0'$$

$$T_2' = (x_2', y_2'), T_2 = (0,1): \quad \begin{pmatrix} x_2' \\ y_2' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \Rightarrow$$

$$a_{12} = x_2' - x_0'$$
$$a_{22} = y_2' - y_0'.$$

Pro matici A a vektor b tedy platí

$$A = \begin{pmatrix} x_1' - x_0' & x_2' - x_0' \\ y_1' - y_0' & y_2' - y_0' \end{pmatrix}, \quad b = \begin{pmatrix} x_0' \\ y_0' \end{pmatrix}. \tag{14}$$

Můžeme psát

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x_1' - x_0' & x_2' - x_0' \\ y_1' - y_0' & y_2' - y_0' \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} x_0' \\ y_0' \end{pmatrix}, \tag{15}$$

a tedy:

$$\begin{aligned} x' &= (x_1' - x_0')x + (x_2' - x_0')y + x_0' \\ y' &= (y_1' - y_0')x + (y_2' - y_0')y + y_0' \end{aligned}$$

Transformaci si ukážeme na následujícím příkladu.

Příklad 1.7 Trojúhelník $T^{'}$ s vrcholy $T_0^{'} = (x_0^{'}, y_0^{'}) = (3,1)$, $T_1^{'} = (x_1^{'}, y_1^{'}) = (2,-1)$, $T_2^{'} = (x_2^{'}, y_2^{'}) = (-1,4)$ převeď na referenční trojúhelník T.

Transformaci provedeme jednoduchým dosazením do vztahu (15) a dostáváme:

$$\begin{pmatrix} x^{'} \\ y^{'} \end{pmatrix} = \begin{pmatrix} -1 & -4 \\ -2 & 3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 3 \\ 1 \end{pmatrix}.$$

Z čehož vyplývá

$$\begin{aligned} x^{'} &= -x - 4y + 3 \\ y^{'} &= -2x + 3y + 1. \end{aligned}$$

Ověření získáme dosazením vrcholů referenčního trojúhelníku T za $\begin{pmatrix} x \\ y \end{pmatrix}$, tedy:

$$\begin{aligned} T_0 = (0,0) : x_0^{'} &= 0 - 0 + 3 = 3, \\ y_0^{'} &= 0 + 0 + 1 = 1, \end{aligned}$$

$$\begin{aligned} T_1 = (1,0) : x_1^{'} &= -1 + 0 + 3 = 2, \\ y_1^{'} &= -2 + 0 + 1 = -1, \end{aligned}$$

$$\begin{aligned} T_2 = (0,1) : x_2^{'} &= 0 - 4 + 3 = -1, \\ y_2^{'} &= + + 3 + 1 = 4. \end{aligned}$$

Vidíme stejné hodnoty souřadnic vrcholů pro zadaný trojúhelník $T^{'}$.

Transformace se dá provést v obou směrech, tj. vrcholy referenčního trojúhelníku T do vrcholů obecného trojúhelníku $T^{'}$. Vztah $x^{'} = Ax + b$ můžeme upravit na

$$x = A^{-1}x^{'} - A^{-1}b = A^{-1}(x^{'} - b), \tag{16}$$

přičemž

$$A^{-1} = \frac{1}{a_{11}a_{22}-a_{12}a_{21}} \begin{pmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{pmatrix},$$

Matici A a vektor b jsme si odvodili výše dle (14). Máme tedy

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{(x_1^{'}-x_0^{'})(y_2^{'}-y_0^{'})-(x_2^{'}-x_0^{'})(y_1^{'}-y_0^{'})} \begin{pmatrix} y_2^{'}-y_0^{'} & -x_2^{'}+x_0^{'} \\ -y_1^{'}+y_0^{'} & x_1^{'}-x_0^{'} \end{pmatrix} \left[\begin{pmatrix} x^{'} \\ y^{'} \end{pmatrix} - \begin{pmatrix} x_0^{'} \\ y_0^{'} \end{pmatrix}\right], \tag{17}$$

z čehož vyplývá:

$$\begin{aligned} x &= \frac{(y_2^{'}-y_0^{'})(x'-x_0^{'})-(x_2^{'}-x_0^{'})(y'-y_0^{'})}{(y_2^{'}-y_0^{'})(x_1^{'}-x_0^{'})-(x_2^{'}-x_0^{'})(y_1^{'}-y_0^{'})} = \\ &= \frac{x'y_2^{'}-x_0^{'}y_2^{'}-x'y_0^{'}-x_2^{'}y^{'}+x_2^{'}y_0^{'}+x_0^{'}y^{'}}{x_1^{'}y_2^{'}-x_0^{'}y_2^{'}-x_1^{'}y_0^{'}-x_2^{'}y_1^{'}+x_2^{'}y_1^{'}+x_0^{'}y_1^{'}} = \\ &= \frac{x'(y_2^{'}-y_0^{'})+y^{'}(x_0^{'}-x_2^{'})-x_0^{'}y_2^{'}+x_2^{'}y_0^{'}}{x_1^{'}y_2^{'}-x_0^{'}y_2^{'}-x_1^{'}y_0^{'}-x_2^{'}y_1^{'}+x_2^{'}y_0^{'}+x_0^{'}y_1^{'}}, \end{aligned}$$

$$\begin{aligned} y &= \frac{(x_1^{'}-x_0^{'})(y'-y_0^{'})-(y_1^{'}-y_0^{'})(x'-x_0^{'})}{(y_2^{'}-y_0^{'})(x_1^{'}-x_0^{'})-(x_2^{'}-x_0^{'})(y_1^{'}-y_2^{'})} = \\ &= \frac{x_1^{'}y'-x_1^{'}y_0^{'}-x_0^{'}y'-x^{'}y_1^{'}+x_0^{'}y_1^{'}+x^{'}y_0^{'}}{x_1^{'}y_2^{'}-x_0^{'}y_2^{'}-x_1^{'}y_0^{'}-x_2^{'}y_1^{'}+x_2^{'}y_0^{'}+x_0^{'}y_1^{'}} = \\ &= \frac{x'(y_0^{'}-y_1^{'})+y^{'}(x_1^{'}-x_0^{'})-x_1^{'}y_0^{'}+x_0^{'}y_1^{'}}{x_1^{'}y_2^{'}-x_0^{'}y_2^{'}-x_1^{'}y_0^{'}-x_2^{'}y_1^{'}+x_2^{'}y_0^{'}+x_0^{'}y_1^{'}}. \end{aligned}$$

Transformaci si opět ukážeme na příkladu. Vezmeme si data z Příkladu 1.7.

Příklad 1.8 Referenční trojúhelník T převeď na trojúhelník T^1 s vrcholy $T_0^{'} = (x_0^{'}, y_0^{'}) = (3, 1)$, $T_1^{'} = (x_1^{'}, y_1^{'}) = (2, -1)$, $T_2^{'} = (x_2^{'}, y_2^{'}) = (-1, 4)$.
Transformaci provedeme jednoduchým dosazením do vztahu (17)

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{(2-3)(4-1)-(-1-3)(-1-1)} \begin{pmatrix} 3 & 2 \\ 4 & -1 \end{pmatrix} \left[\begin{pmatrix} x^{'} \\ y^{'} \end{pmatrix} - \begin{pmatrix} 3 \\ 1 \end{pmatrix}\right],$$

z čehož vyplývá:

$$x = \frac{3x' + 4y^{'} - 13}{-11},$$

$$y = \frac{2x' - y^{'} - 5}{-11}.$$

Ověření dostaneme dosazením vrcholů obecného trojúhelníku, tedy:

$$T_0^{'} = (3,1) : x_0 = \frac{3 \cdot 3 + 4 \cdot 1 - 13}{-11} = -\frac{0}{11} = 0,$$

$$y_0 = \frac{2 \cdot 3 - 1 - 5}{-11} = -\frac{0}{11} = 0,$$

$$T_1^{'} = (2,-1) : x_1 = \frac{3 \cdot 2 + 4 \cdot (-1) - 13}{-11} = \frac{11}{11} = 1,$$

$$y_0 = \frac{2 \cdot 2 + 1 - 5}{-11} = -\frac{0}{11} = 0,$$

$$T_2^{'} = (-1,4) : x_0 = \frac{3 \cdot (-1) + 4 \cdot 4 - 13}{-11} = -\frac{0}{11} = 0,$$

$$y_1 = \frac{2 \cdot (-1) - 4 - 5}{-11} = \frac{11}{11} = 1.$$

Čímž jsme získaly vrcholy referenčního trojúhelníku T.

Nyní si ukážeme, jak bude vypadat interpolační polynom $P_{11}(x,y)$ pro obecný trojúhelník $T^{'}$. Musí platit $P_{11}(x_k,y_k) = f_k$, tedy

$$\begin{array}{rcccl} P_{11}(x_0,y_0) & = & \alpha x_0 + \beta y_0 + \gamma & = & f_0 \\ P_{11}(x_1,y_1) & = & \alpha x_1 + \beta y_1 + \gamma & = & f_1 \\ P_{11}(x_2,y_2) & = & \alpha x_2 + \beta y_2 + \gamma & = & f_2, \end{array}$$

Máme soustavu tří rovnic o třech neznámých, kterou lze zapsat maticově $Op = f$ a platí $\mathrm{O} = \begin{pmatrix} x_0 & y_0 & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{pmatrix}, p = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}, f = \begin{pmatrix} f_0 \\ f_1 \\ f_2 \end{pmatrix}.$

Vidíme, že je třeba vypočítat vektor p. $Op = f$ upravíme na $p = O^{-1}f$, kde O^{-1} vyjádříme pomocí reciproké matice $adjO$, tedy

$$O^{-1} = \frac{1}{detO} adjO,$$

kde

$$detO = x_0y_1 - x_0y_2 + x_1y_2 - x_1y_0 + x_2y0 - x_2y_1$$

$$adjO = \begin{pmatrix} y_1 - y_2 & y_2 - y_1 & y_0 - y_1 \\ x_2 - x_1 & x_0 - x_2 & x_1 - x_0 \\ x_1y_2 - x_2y_1 & x_2y_0 - x_0y_2 & x_0y_1 - x_1y_0 \end{pmatrix}.$$

Můžeme psát

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \frac{1}{x_0y_1 - x_0y_2 + x_1y_2 - x_1y_0 + x_2y_0 - x_2y_1} \cdot \\ \cdot \begin{pmatrix} y_1 - y_2 & y_2 - y_1 & y_0 - y_1 \\ x_2 - x_1 & x_0 - x_2 & x_1 - x_0 \\ x_1y_2 - x_2y_1 & x_2y_0 - x_0y_2 & x_0y_1 - x_1y_0 \end{pmatrix} \begin{pmatrix} f_0 \\ f_1 \\ f_2 \end{pmatrix}.$$

Koeficienty interpolačního polynomu $P_{11}(x, y)$ lze vyjádřit jako

$$\alpha = \frac{(y_1 - y_2)f_{00} + (y_2 - y_1)f_{11} + (y_0 - y_1)f_{22}}{x_0y_1 - x_0y_2 + x_1y_2 - x_1y_0 + x_2y_0 - x_2y_1}$$
$$\beta = \frac{(x_2 - x_1)f_{00} + (x_0 - x_2)f_{11} + (x_1 - x_0)f_{22}}{x_0y_1 - x_0y_2 + x_1y_2 - x_1y_0 + x_2y_0 - x_2y_1}$$
$$\gamma = \frac{(x_1y_2 - x_2y_1)f_{00} + (x_2y_0 - x_0y_2)f_{11} + (x_0y_1 - x_1y_0)f_{22}}{x_0y_1 - x_0y_2 + x_1y_2 - x_1y_0 + x_2y_0 - x_2y_1}$$

a polynom $P_{11}(x, y)$ pro obecný trojúhelník $T^{'}$ je ve tvaru

$$\begin{aligned} P_{11}(x, y) = \; & \frac{(y_1 - y_2)f_{00} + (y_2 - y_1)f_{11} + (y_0 - y_1)f_{22}}{x_0y_1 - x_0y_2 + x_1y_2 - x_1y_0 + x_2y_0 - x_2y_1} x + \\ & + \frac{(x_2 - x_1)f_{00} + (x_0 - x_2)f_{11} + (x_1 - x_0)f_{22}}{x_0y_1 - x_0y_2 + x_1y_2 - x_1y_0 + x_2y_0 - x_2y_1} y + \\ & + \frac{(x_1y_2 - x_2y_1)f_{00} + (x_2y_0 - x_0y_2)f_{11} + (x_0y_1 - x_1y_0)f_{22}}{x_0y_1 - x_0y_2 + x_1y_2 - x_1y_0 + x_2y_0 - x_2y_1}. \end{aligned} \quad (18)$$

V následujícím příkladu si ukážeme, jak může vypadat interpolační polynom $P_{11}(x, y)$ pro obecný trojúhelník $T^{'}$.

Příklad 1.9 Pro trojúhelník T^1 s vrcholy $T_0^{'} = (x_0^{'}, y_0^{'}) = (3,1)$, $T_1^{'} = (x_1^{'}, y_1^{'}) = (2,-1)$, $T_2^{'} = (x_2^{'}, y_2^{'}) = (-1,4)$ a funkčními hodnotami $f_0 = -3, f_1 = 2, f_2 = 1$ zkonstruuj interpolační polynom $P_{11}(x,y)$.

Data dosadíme do vztahu (18) a dostaneme interpolační polynom ve tvaru

$$\begin{aligned}
P_{11}(x,y) = & \quad \frac{(-1-4)(-3)+(4-1)2+(1+1)(-1)}{-3+8-1-1-12-2}x + \\
& + \frac{(-1-2)(-3)+(3+1)2+(2-3)1}{-3+8-1-1-12-2}y + \\
& + \frac{(8-1)(-3)+(-1-12)2+(-3-2)1}{-3+8-1-1-12-2} = \\
= & -\frac{23}{11}x - \frac{16}{11}y + \frac{52}{11} = -2,0909x - 1,4545y + 4,7273.
\end{aligned}$$

Ověříme podmínky interpolace, tedy

$$\begin{aligned}
P_{11}(3,1) &= -\frac{23}{11}x - \frac{16}{11}y + \frac{52}{11} = -\frac{23}{11}3 - \frac{16}{11}1 + \frac{52}{11} = -\frac{33}{11} = -3 \\
P_{11}(2,-1) &= -\frac{23}{11}x - \frac{16}{11}y + \frac{52}{11} = -\frac{23}{11}2 - \frac{16}{11}(-1) + \frac{52}{11} = \frac{22}{11} = 2 \\
P_{11}(-1,4) &= -\frac{23}{11}x - \frac{16}{11}y + \frac{52}{11} = -\frac{23}{11}(-1) - \frac{16}{11}4 + \frac{52}{11} = \frac{11}{11} = 1.
\end{aligned}$$

Podmínky interpolace jsou splněny.

Nyní si představíme m-soubor, jehož výstupem budou koeficienty interpolačního polynomu pro obecný trojúhelník $T^{'}$. M-soubor je odvozen podle předchozí teorie, dle $Op = f$. Na vstupu zadáme souřadnice bodů $[x_k, y_k]$ a funkční hodnoty $f_k, k = 0,1,2$. Ve výstupu bude řádkový vektor koeficientů intepolačního polynomu $P_{11}(x,y)$.

Text m-souboru v MATLABu vypadá následovně:

```
function [p]=polynom(x,y,f)

% vstup     x ... x-ova souradnice bodu
%           y ... y-ova souradnice bodu
%           f ... predepsane funkcni hodnoty
% vystup    p ... vektor koeficientu interpolacniho
%                 polynomu

O=[x(1) y(1) 1;
   x(2) y(2) 1;
   x(3) y(3) 1];
```

```
P = inv(O)*f;
p = P(:)';
```

Do příkazového okna Matlabu zadáme vstupy:

```
x=[3; 2; -1];
y=[1; -1; 4];
f=[-3; 2; 1];
[p]=polynom(x,y,f)
```

Po potvrzení těchto vstupů získáme

```
p =   -2.0909     -1.4545      4.7273
```

čímž jsme získali stejné koeficienty, jak při ručním výpočtu.

2. Interpolace v matematickém softwaru MATLAB

Kapitola se týká funkcí pro interpolaci v matematickém softwaru MATLAB. Teorii jsem studovala z [5].

Matematický software MATLAB pro interpolaci ve dvourozměrném prostoru na obdélníkové síti nabízí pouze funkci $interp2$, kterou lze volat různými způsoby. V následujícím textu je ukázán jeden způsob volání, další jsou uvedeny v [5]. Syntaxe funkce je $ZZ = interp2(X, Y, Z, XX, YY, method)$, kde $method$ určuje jakým typem chceme interpolaci provést a může být:

nearest	metoda nejbližšího souseda
linear	metoda lineární interpolace
spline	po částech kubický splajn
cubic	kubická Hermitovská interpolace

Funkce $interp2$ vytvoří interpolační polynom $P_{nm}(x, y)$ v námi zvolených uzlech interpolace $[x_i, y_j]$ a předepsaných funkčních hodnotách, které píšeme do matice Z. Uzly $[x_i, y_j]$ píšeme do vektorů x a y, ze kterých je nutné vytvořit matice tj.

$$[X, Y] = meshgrid(x, y).$$

Výstupem jsou funkční hodnoty tohoto polynomu $P_{nm}(x, y)$ pro námi zvolené body, které píšeme do vektorů xx, yy, ze kterých musíme opět udělat matice pomocí funkce $meshgrid$, tzn.

$$[XX, YY] = meshgrid(xx, yy).$$

Pro interpolaci dvou proměnných na nepravidelné síti nabízí software Matlab funkci $griddata$. Na nepravidelné síti nejsou uzly ekvidistantní, tedy rozestup mezi jednotlivými uzly je rozdílný. Syntaxe je stejná jako u funkce $interp2$,
tj. $ZZ = griddata(X, Y, Z, XX, YY)$.

V následujícím příkladu si ukážeme rozdíl mezi funkcí $interp2$ a $griddata$. Data vezmeme z Příkladu 1.3.

Příklad 1.10 Pro stejná data jako v Příkladu 1.3 nalezněte pomocí uvedeného m-souboru interpolační polynom. Data byla dána na oblasti $D = \{[x, y]: x \in [0, 1], y \in [2, 3]\}$ následující tabulkou

	2	3
0	-1	2
1	3	-2

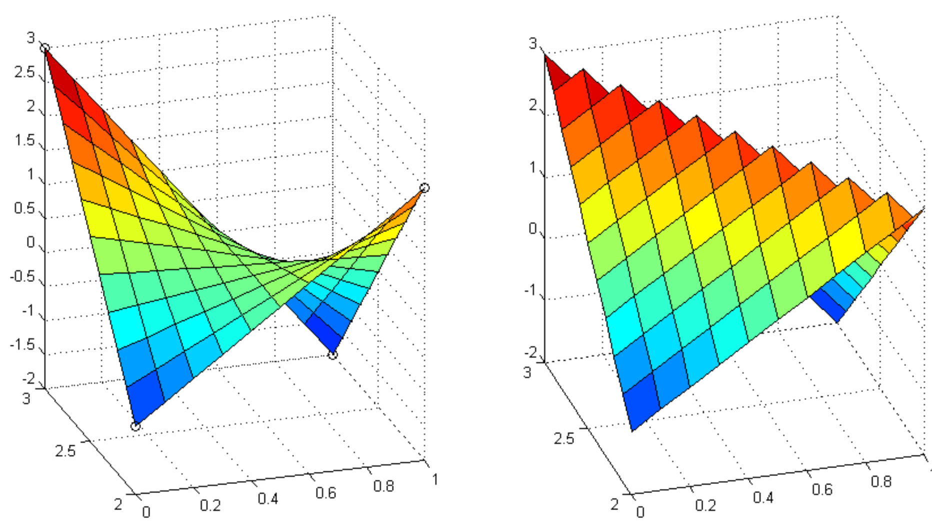
Příkazy v Matlabu budou vypadat následovně:

```
x=[0;  1];
y=[2;3];
[X,Y]=meshgrd(x,y)
Z=[-1  2;  3  -2];
xx=[0:0.1:1];
yy=[2:0.1:3];
ZZ=interp2(X,Y,Z,XX,YY,  'linear');
ZZ=griddata(X,Y,Z,XX,YY,  'linear');

%vykresleni  grafu
plot3(X,Y,Z,'ko')
ZZ=interp2(X,Y,Z,XX,YY,'linear');
subplot(1,2,1);
surf(XX,YY,ZZ);
hold on;

plot3(X,Y,Z,'ko')
ZZ1=griddata(X,Y,Z,XX,YY,'linear');
subplot(1,2,2)
surf(XX,YY,ZZ1);
```

Graf interpolační funkce dvou proměnných může vypadat následovně.



Obrázek 6: Interpolační polynom dvou proměnných. Vlevo na obdélníkové síti, vpravo na nepravidelné.

Vidíme rozdíl grafů, tedy funkce se nedají použít pro stejná data.

Pro interpolaci jedné proměnné Matlab nabízí více možností. V této bakalářské práci jsou uvedené jen některé z nich.
Funkce $polyfit$ nalezne koeficienty výsledného polynomu $P(x)$. Syntaxe této funkce je $a = polyfit(x, f, n)$, kde x je vektor bodů na ose x a k nim předepsané funkční hodnoty obsahuje vektor f. Stupeň polynomu, jimž chceme interpolovat body x značí n. Na Příkladu 1.11. si ověříme, zda Příklad 1.1. bude shodný s funkcí $polyfit$.

Příklad 1.11 Pro data v tabulce nalezněte Lagrangeův interpolační polynom.

x_i	0	1	-1
$f(x_i)$	-1	1	2

Do příkazového okna zadáme vstupy:

```
x=[0 1 -1];
f=[-1 1 2];
a=polyfit(x,f,2)
```

Po potvrzení vstupů získáme sestupné uspořádání koeficientů, tedy

```
a =

    2.5000    -0.5000    -1.0000
```

Vidíme shodnost výsledků.

Pokud by nás zajímaly hodnoty polynomu $P_n(x)$ ve všech prvcích vektoru x, lze použít funkce *polyval*:

$$y = polyval(a, x),$$

kde a je vektor do kterého se zadávají koeficienty polynomu, který chceme aproximovat. Vektor x je vektor uzlů interpolace. Pro představu si funkci *polyval* ukážeme opět na Příkladu 1.1.
Do příkazového okna zadáme vstupy:

```
p = [2.5000    -0.5000    -1.0000];
x=[0 1 -1];
f=polyval(p,x)
```

Po potvrzení vstupů získáme

```
f =

    -1     1     2
```

Vidíme shodnost.

Další funkci, kterou můžeme pro interpolace použít je $interp1$, která je obdobou funkce $interp2$. Základní skladba této funkce je $y_i = interp1(x, y, xx)$, kde vektor x obsahuje uzly x_i, do vektoru y se zapisují funkční hodnoty těchto bodů a vektor xx obsahuje body na ose x v nichž má být vypočítána hodnota interpolačního polynomu. Funkce $interp1$ může být volána různými způsoby a zadána také s metodou, kterou chceme použít (více v [5]).

Závěr

Cílem mé bakalářské práce bylo nastudování teorie interpolace v $\mathbb{R}^2$. Čtenářům jsem chtěla ukázat, že se příklady dají řešit nejen pro ruční počítání, ale i v matematickém softwaru Matlab. Vytvořila jsem ke každému tématu kód, který byl vysvětlen, jakým způsobem funguje a následně v něm ukázán řešený příklad.

Nejdříve jsem se zaměřila na seznámení úlohy interpolace a také na interpolační podmínky . V první podkapitole byl řešen interpolační polynom v $\mathbb{R}$ pomocí Lagrangeova interpolačního polynomu, který se dále využil pro odvození interpolačního polynomu v $\mathbb{R}^2$ na obdélníkové síti v druhé podkapitole. Stěžejní částí byla třetí podkapitola, kde jsem čerpala z konzultací s vedoucí své bakalářské práce. Interpolační polynom pro tuto situaci byl odvozen ze soustavy tří rovnic o třech neznámých, která se řešila maticově. Druhá kapitola se věnovala matematickému softwaru Matlab. Čtenář byl informován jaké funkce Matlab nabízí a blíže s němi seznámen.

Díky své bakalářské práci jsem si rozšířila teorii interpolace. V předmětu Numerické metody jsem studovala interpolaci pro $\mathbb{R}$ a tak interpolace v $\mathbb{R}^2$ pro mne byla novým učivem. Dále jsem si také zlepšila své dovednosti v matematickým softwaru Matlab. Navíc psaní této bakalářské práce mě naučilo lépe formulovat teorii tak, aby čtenář co nejsrozumitelněji pochopil danou problematiku.

Literatura

[1] Demidovič, B. P., Maron, I. A.: *Základy numerické matematiky.* Státní nakladatelství technické literatury, Praha, 1966.

[2] Horová, I., Zelinka, J.: *Numerické metody.* Masarykova univerzita v Brně, Brno, 2004.

[3] Isaacson, E., Keller, H. B.: *Analysis of numerical methods.* Dover Publications, Mineola, 1994.

[4] Kobza, J.: *Numerické metody.* Skriptum UP, Olomouc, 1993.

[5] Ženčák, P: *Matlab pro začátečníky i mírně pokročilé.* Skriptum UP, Olomouc, 2013.