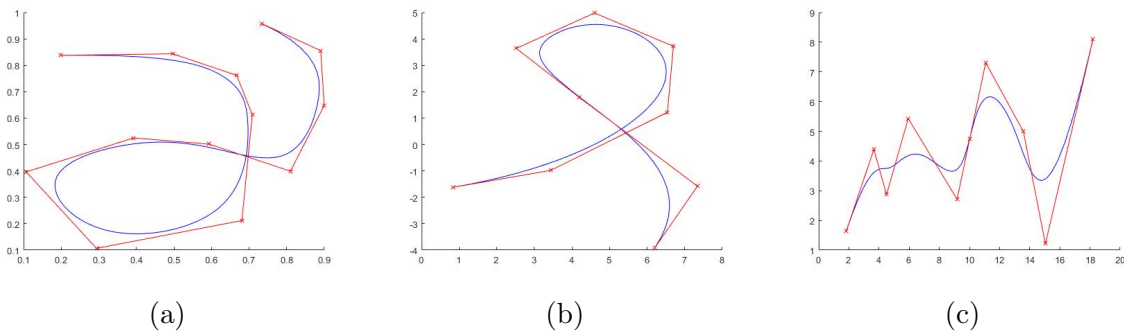


# 1 Teoretická časť

## 1.1 B-spline krivky

Nasledujúca kapitola vychádza zo zdrojov [1, 3, 5, 6]. Na Bézierove krivky nadväzuje teória B-splinov. Názov spline predstavuje prúžok dreva alebo kovu, ktorý sme schopný ohýbať. B predstavuje slovo bázový, a teda B-spliny sú vlastne bázové spliny. Nevýhoda kriviek, ktoré sú definované jedným polynómom je, že prispôsobenie niektorým komplexným tvarom vyžaduje vysoký stupeň polynómu. Na druhú stranu, príliš vysoký stupeň negatívne ovplyvní stabilitu numerického výpočtu. To je jeden z dôvodov zavedenia B-splinu. B-spline krivka je zložená z viacerých Bézierov v jednotlivých úsekoch.

Existujú ďalšie krivky zložené z Bézierových, napríklad kubický Hermitov spline a tak tiež Catmull-Romov spline. Podobne ako pri Bézierových krivkách, aj B-spliny môžeme rozdeliť na racionálne a neracionálne. Výhoda B-splinov je, že ich algoritmus je pomerne jednoduchý, rýchly a prípadne sa dajú ľahko lokálne upraviť. Sú teda výhodnejšie v praxi.



Obr. 1: B-spline krivky

### 1.1.1 Uzlový vektor

Ak by sme chceli napriamo rozdeliť B-spline krivku, bolo by to náročné. Je preto výhodnejšie, aby sme rozdelili jej oblasti. Ak je oblasť napríklad interval  $< 0, 1 >$ , rozdelíme ju na takzvané uzly  $0 \leq u_0 \leq u_1 \leq \dots \leq u_m \leq 1$ . Táto neklesajúca postupnosť  $U = \{u_0, u_1, \dots, u_m\}$  sa nazýva uzlový vektor, jednotlivé  $u_i$  sú uzly a je to potrebný vstupný parameter na vytvorenie B-splinu. Keďže je to neklesajúca postupnosť, uzly  $u_i$  sa môžu opakovať. Maximálna násobnosť pre vnútorné uzly je  $p$  a pre koncové uzly  $p + 1$ , kde  $p$  je stupeň krivky.

### 1.1.2 Bázové funkcie

**Definícia 1.1.** Nech  $U = \{u_0, u_1, \dots, u_m\}$  je uzlový vektor a  $u \in < u_0, u_m >$ . Potom bázové funkcie stupňa  $p$  definujeme takto:

$$N_{i,0}(u) = \begin{cases} 1 & u \in < u_i, u_{i+1} > \\ 0 & \text{inak} \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \quad (1.1)$$

*Poznámka.* Na definícii 1.1 si môžeme všimnúť, že  $N_{i,p}(u)$  využíva rekurziu.  $N_{i,p}(u)$  je nezáporná po častiach spojitá polynomiálna funkcia. Pri viacnásobných uzloch vznikajú zlomky  $\frac{0}{0}$ . Tento problém odstránime, ak ich položíme rovné nule.

**Definícia 1.2.** Majme  $N_{i,p}$  báзовú funkciu stupňa  $p$  z definície 1.1,  $n+1$  navzájom rôznych riadiacich bodov  $P_i$ , uzlový vektor  $U = \{u_0, u_1, \dots, u_m\}$  a číslo  $u \in \langle u_0, u_m \rangle$ . Definujme B-spline krivku stupňa  $p$  ako lineárnu kombináciu báзовých funkcií a riadiacich bodov:

$$C(u) = \sum_{i=0}^p N_{i,p}(u) P_i, \quad u \in \langle u_0, u_m \rangle \quad (1.2)$$

*Poznámka.* Pre B-spline stupňa  $p$  s  $n+1$  riadiacimi bodmi bude mať uzlový vektor dĺžku  $m+1 = n+p+2$  a bude mať tento tvar:

$$U = \{a_0, \dots, a_p, u_{p+1}, \dots, u_{m-p-1}, b_0, \dots, b_p\} \quad (1.3)$$

Pomocou uzlového vektoru sa posúvame po jednotlivých sekvenciách B-splinu, ktoré sú definované dĺžkou uzlu – dĺžkou intervalu  $\langle u_i, u_{i+1} \rangle$ .

*Poznámka.* Pokiaľ platí, že  $n = p$  a  $U = \{0, \dots, 0, 1, \dots, 1\}$ , potom báзовé funkcie majú rovnaký tvar ako pri Bézierovej krivke a  $C(u)$  je vlastne Bézier.

**Definícia 1.3.** Uzlový vektor sa nazýva uniformný, ak platí  $u_{i+1} - u_i = konst$  pre  $i \in \langle p+1, m-p-1 \rangle$ .

### 1.1.3 Násobné uzly

Pokiaľ parameter  $u$  nie je uzol v uzlovom vektore  $U$ ,  $C(u)$  v strede segmentu stupňa  $p$  je nekonečne diferencovateľná. Pokiaľ toto neplatí a  $u$  je uzol s násobnosťou  $k$  pri nenulovej báзовой funkcii,  $C(u)$  je iba  $C^{p-k}$  spojitá na danom segmente.

Zmena spojitosti pri násobných uzloch nie je ich jediný dopad na B-spline. Ďalší dôsledok, ktorý je dôležitý pri výpočtových algoritmoch B-spline kriviek, je, že každý uzol s multiplicitou  $k$  zníži počet nenulových báзовých funkcií v tomto uzle. Uzol s multiplicitou  $k$  má najviac  $p - k + 1$  nenulových báзовých funkcií. Pre uzol s násobnosťou  $k = p$  je iba jedna nenulová báзовая funkcia.

### 1.1.4 Typy B-spline kriviek

B-spline krivky môžeme rozdeliť na viacero druhov. Nech  $C(u)$  je B-spline krivka z definície (1.2).

1. **Uzavreté:** Ak by sme chceli z  $C(u)$  skonštruovať uzavretú B-spline krivku  $p$ -tého stupňa, jedným zo spôsobov je vziať prvých  $p$  bodov a zaradiť ich za posledný bod  $P_n$ . Uzlový vektor musí byť uniformný. Krivka bude potom začínať v bode  $P_0$  a končiť v  $P_n$ .

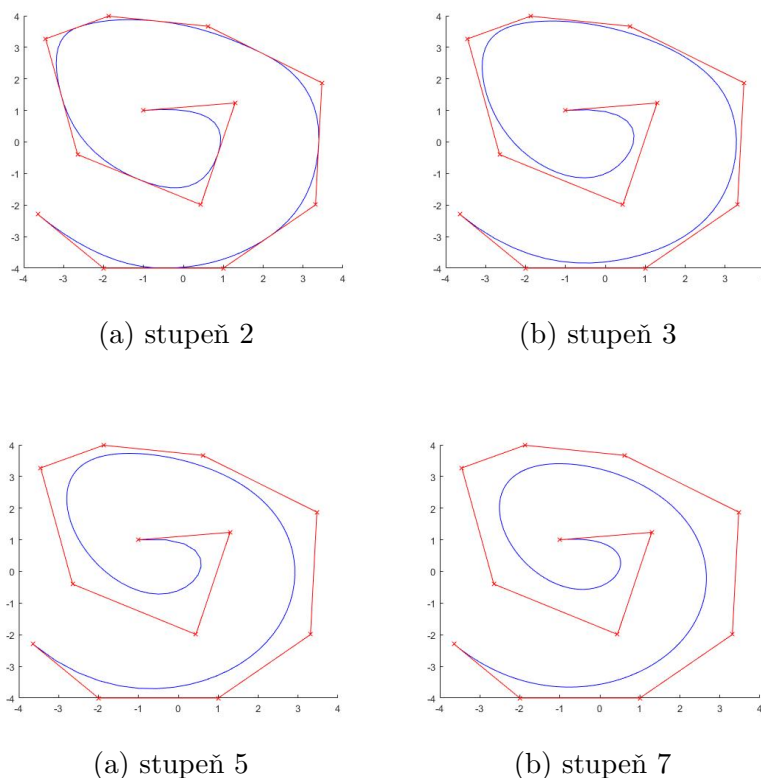
2. **Vnorené<sup>1</sup>**: Na vytvorenie vnorenej B-spline krivky potrebujeme, aby prvý a posledný uzol v uzlovom vektore mali multiplicitu  $p + 1$ .
3. **Otvorené**: Otvorená B-spline krivka je taká, ktorej uzlový vektor nemá štruktúru násobnosti prvého a posledného uzlu ako pri vnorenej. Uniformnosť nie je nutná podmienka.

### 1.1.5 Vlastnosti B-spline kriviek

Najprv sa pozrieme na rozdiely medzi Bézierovými a B-spline krivkami. Zmenou aspoň jedného riadiaceho bodu v Bézierovej krivke dostaneme iný tvar krivky. Toto však neplatí pri B-spline – zmena  $i$ -tého riadiaceho bodu zmení  $C(u)$  iba na intervale  $[u_i, u_{i+p+1})$ , kde  $p$  predstavuje stupeň krivky. Ďalším rozdielom je, že pri B-spline je stupeň krivky vstupným parametrom. Toto nám umožňuje, aby sme zostrojili B-spline krivku komplexného tvaru pri nízkom stupni, čo je pri Bézierovej krivke samotnej nemožné. Pokiaľ má uzlový vektor tvar  $U = \{0, \dots, 0, 1, \dots, 1\}$  a platí  $n = p$ , tak sa B-spline zredukuje na Bézierovu krivku.

B-spline krivka teda zdedila vlastnosti Bézierovej, spomínané v predošlej kapitole, medzi nimi afinnú invariantnosť. Vďaka nej nám stačí použiť afinnú transformáciu na riadiace body a zostrojiť pre ne B-spline. Nemusíme transformovať samotnú krivku. Použitie B-spline kriviek v počítačovej grafike má mnoho výhod, avšak sú to polynomiálne krivky, a tie sa nedajú použiť na vykreslenie kruhov, elíps a podobných tvarov.

Na nasledujúcich obrázkoch sa nachádzajú B-spline krivky rôznych stupňov pri rovnakých riadiacich bodoch.



Obr. 3: B-spline krivky rôznych stupňov

<sup>1</sup>Preložené z anglického *clamped* – upnutý, vnorený. V českém jazyku se používá výraz *plovoucí*.

## 1.2 B-spline plochy

Literatúra použitá v tejto kapitole je rovnaká ako v predošlej.

**Definícia 1.4.** Nech  $p$  a  $q$  sú stupne kriviek. Majme  $m + 1$  riadkov a v každom riadku  $n + 1$  riadiacich bodov  $P_{i,j}$ , kde  $0 \leq i \leq m$  a  $0 \leq j \leq n$  a uzlové vektory  $U = \{u_0, \dots, u_r\}$   $V = \{v_0, \dots, v_s\}$  v nasledujúcom tvare:

$$\begin{aligned} U &= \{0, \dots, 0, u_{p+1}, \dots, u_{r-p-1}, 1, \dots, 1\} \\ V &= \{0, \dots, 0, v_{q+1}, \dots, v_{s-q-1}, 1, \dots, 1\}, \end{aligned} \quad (1.4)$$

(krajné uzly majú násobnosť  $p + 1$  v  $U$  a  $q + 1$  vo  $V$ ).

Nech  $\{u, v\}$  je dvojica čísel, pre ktoré platí:  $u \in \langle u_0, u_r \rangle$  a  $v \in \langle v_0, v_s \rangle$  a nech  $N_{i,n}(u)$  a  $N_{j,q}(v)$  sú bázové funkcie. Potom definujeme B-spline plochu takto:

$$S(u, v) = \sum_{i=0}^n \sum_{j=0}^m N_{i,p}(u) N_{j,q}(v) P_{i,j} \quad (1.5)$$

$U$  má  $r + 1$  uzlov a  $V$  má  $s + 1$  uzlov. Indexy teda musia spĺňať nasledovné rovnice:

$$\begin{aligned} r &= n + p + 1 \\ s &= m + q + 1 \end{aligned} \quad (1.6)$$

V časti Programové spracovanie a rozbor sa dozvieme, ako zostrojiť B-spline krivku. Zameriame sa konkrétne na De Boorov algoritmus, ktorý vychádza z riadiacich bodov, z nich spočíta ďalšie, až sa prepracuje k bodom, ktoré sa nachádzajú na hľadanej krivke.

### 1.2.1 Vlastnosti B-spline plôch

#### 1. Nezápornosť bázových funkcií

$$\forall u, v, i, j, p, q : N_{i,p}(u) N_{j,q}(v) \geq 0$$

#### 2. Bézierova plocha je špeciálny prípad B-spline plochy

pokiaľ  $n = p$ ,  $m = q$ ,  $U = \{0, \dots, 0, 1, \dots, 1\}$  a  $V = \{0, \dots, 0, 1, \dots, 1\}$ , bázové funkcie sú Bernsteinove polynómy a výsledná plocha je Bézierova plocha.

#### 3. Nulový prípad súčinu bázových funkcií

Pokiaľ sa  $\{u, v\}$  nachádza mimo štvorca  $\langle u_i, u_{i+p+1} \rangle \times \langle v_j, v_{j+q+1} \rangle$ , súčin  $N_{i,p}(u) N_{j,q}(v)$  je nulový.

#### 4. Afínná invariantnosť

Platí ako aj pri krivkách. Aplikovaním afinného zobrazenia na riadiace body aplikujeme afinné zobrazenie na celú plochu.

#### 5. Maximum plochy

Pri kladných stupňoch  $p, q$  B-spline plocha dosiahne práve jedno maximum.

#### 6. Konvexný obal

Pri dvojici  $\{u, v\}$  nachádzajúcej sa vo vnútri štvorca  $\langle u_{i^*}, u_{i^*+p+1} \rangle \times \langle v_{j^*}, v_{j^*+q+1} \rangle$  platí, že plocha  $S(u, v)$  je v konvexnom obale tvorenom riadiacimi bodmi  $P_{i,j}$ , kde  $(i^* - p) \leq i \leq i^*$  a  $(j^* - q) \leq j \leq j^*$ .

#### 7. Rohové body

B-spline plocha interpoluje štyri rohové body:  $S(0, 0) = P_{0,0}$ ,  $S(1, 1) = P_{1,1}$ ,  $S(1, 0) = P_{n,0}$ ,  $S(0, 1) = P_{0,m}$ .

## 2 Programové spracovanie a rozbor

### 2.1 B-spline

V tejto časti práce sa pozrieme na vytvorenie B-spline plochy. Aby sme ju vytvorili, musíme najprv vedieť vytvoriť B-spline krivku z riadiacich bodov. Zopakujeme si jej definíciu:

**Definícia 2.1.** Nech  $p$  je stupeň krivky,  $P_i$  sú riadiace body s počtom  $n + 1$  a  $U = \{u_0, \dots, u_p, u_{p+1}, \dots, u_{m-p-1}, u_{m-p}, \dots, u_m\}$  je uniformný uzlový vektor, kde prvých  $p+1$  uzlov sú nuly a posledných  $p + 1$  uzlov sú jednotky. Ďalej nech  $N_{i,p}(u)$  je bázová funkcia. B-spline krivka má potom tento tvar:

$$C(u) = \sum_{i=0}^p N_{i,p}(u)P_i, \quad u \in \langle u_0, u_m \rangle \quad (2.1)$$

Na jej výpočet môžeme použiť viacero algoritmov. Jedným z nich je de Casteljau algoritmus, ktorý využíva rekurzívnu vlastnosť B-spline bázových funkcií  $N_{i,p}(u)$ . Ďalším z nich, De Boorov algoritmus, je zovšeobecnením de Casteljau algoritmu, a naň sa v tejto práci zameriame.

#### 2.1.1 De Boorov algoritmus

De Boorov algoritmus je rýchly a numericky stabilný algoritmus, ktorý slúži na nájdenie bodu B-spline krivky. Tento bod vypočíta pomocou riadiacich bodov.

vstupné hodnoty: parameter  $u \in \langle u_0, u_m \rangle$ , uzlový vektor  $U$ , riadiace body  $P_i$   
výstupná hodnota: bod na krivke B-spline  $C(u)$

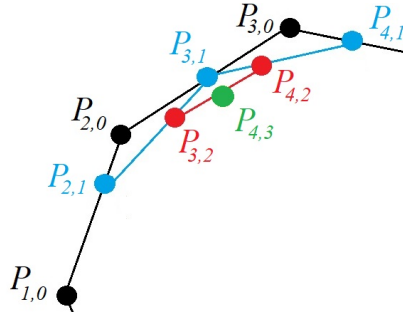
---

**Algorithm 1:** De Boorov algoritmus

---

```
Ak  $u \in \langle u_k, u_{k+1} \rangle \wedge u \neq u_k$ , potom  $h = p$  a  $s = 0$ 
Ak  $u = u_k$  a  $u_k$  je uzol multiplicity  $s$ , potom  $h = p - s$ 
Afektované riadiace body  $P_{k-s}, P_{k-s-1}, \dots, P_{k-p}$  skopírujeme do nového poľa a
premenujeme na  $P_{k-s,0}, P_{k-s-1,0}, \dots, P_{k-p,0}$ .
for  $r=1$  to  $h-s$  do
    for  $i=k-p+r$  to  $k-s$  do
         $a_{i,r} = (u - u_i) / (u_{i+p-r+1} - u_i)$ 
         $P_{i,r} = (1 - a_{i,r})P_{i-1,r-1} + a_{i,r}P_{i,r-1}$ 
    end
end
 $P_{k-s,p-s}$  je bod  $C(u)$ 
```

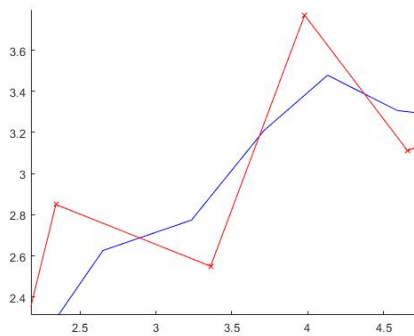
---



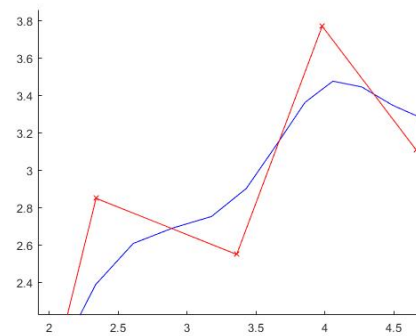
Obr. 4: Výpočet bodov De Boorovým algoritmom

Takto získame jeden bod B-spline krivky pre konkrétnu hodnotu parametra  $u$ . Na získanie celej krivky vypočítame jednotlivé  $P_{k-s,p-s}$  pre delenie intervalu  $\langle u_0, u_m \rangle$ , ktoré si sami zvolíme. Čím jemnejšie delenie zvolíme, tým budú body krivky hustejšie a krivka môže nadobudnúť takmer spojitý tvar pre ľudské oko.

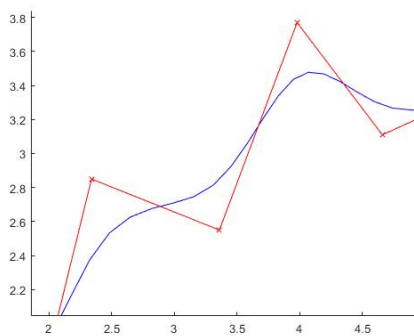
Na nasledujúcich obrázkoch vidíme B-spline krivky stupňa 2 s rovnakými riadiacimi bodmi pri rôznom delení intervalu  $\langle u_0, u_m \rangle$ . Pohľad je zväčšený, aby sme mohli lepšie vidieť rozdiely.



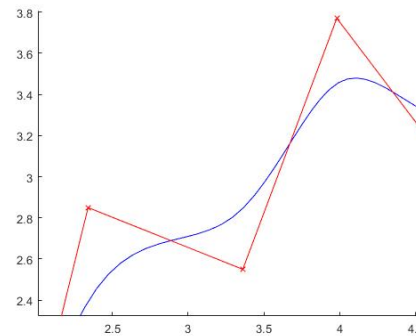
(a) 15 deliacich sekvencií



(b) 30 deliacich sekvencií



(a) 50 deliacich sekvencií



(b) 100 deliacich sekvencií

Obr. 6: B-spline krivky s rôznym počtom deliacich sekvencií

Teraz si vysvetlíme funkcie naprogramované v MATLABE.

Treba podotknúť, že zatiaľ sme pri B-spline funkciách indexovali všetko od 0. MATLAB však indexuje od 1. Výpočty tomu boli prispôbené.

Prvá funkcia spočíta uzlový vektor, multiplicitu parametru  $u$  a tiež index intervalu alebo uzlu, kde sa  $u$  nachádza v uzlovom vektore.

Uniformný uzlový vektor sa počítal jednoducho: najprv sme do prvých  $p+1$  prvkov vektoru uložili nuly a do posledných  $p+1$  prvkov jednotky. Prostredné hodnoty  $u_{p+1}, \dots, u_{m-p-1}$  sme spočítali tak, aby boli ekvidistantné spolu s prvkami  $u_p$  a  $u_{m-p}$ .

Hodnotu, o ktorú sa v prostredných hodnotách uzlového vektoru posúvame, spočítame takto:

$$krok = \frac{1}{n - p + 1} \quad (2.2)$$

**Príklad 2.2.** Máme B-spline krivku druhého stupňa s ôsmimi riadiacimi bodmi. Chceme vypočítať jej uzlový vektor  $U$ :

$$\begin{aligned} p &= 2 \\ n &= 8 \end{aligned} \quad (2.3)$$

Dĺžku uzlového vektoru vypočítame:

$$m + 1 = p + n + 1 = 11 \quad (2.4)$$

Teraz zistíme krok, o ktorý sa budeme v prostredných hodnotách vektoru posúvať:

$$krok = \frac{1}{n - p + 1} = \frac{1}{6} \quad (2.5)$$

Krajných  $p+1$  uzlov naplníme na začiatku nulami a na konci jednotkami. Prostredné hodnoty budeme posúvať po  $\frac{1}{6}$ . Uniformný uzlový vektor v našom prípade vyzerá takto:

$$U = \{0, 0, 0, \frac{1}{6}, \frac{2}{6}, \frac{3}{6}, \frac{4}{6}, \frac{5}{6}, 1, 1, 1\} \quad (2.6)$$

Ďalší zo vstupných parametrov De Boorovej funkcie je multiplicita a index parametra. Pre každý prvok delenia intervalu  $u_i$  si uložíme jeho pozíciu a násobnosť:

Ak  $u_i \geq U_j \wedge u_i \leq U_{j+1}$  potom  $k_i = j$ ,  $s_i = 0$   
 Ak  $u_i = U_j$  potom  $k_i = j$  a  $s_i$  získame v cykle

Vstupné hodnoty do MATLAB funkcie:

n	počet riadiacich bodov
p	stupeň krivky
sekv	počet sekvencií delenia intervalu

Funkcia na výpočet uzlového vektoru, multiplicity a indexu:

```

1 function [U,s,k]=knotvektor(n,p,sekv)
2
3     m=2; % dimenzia riadiacich bodov
4     uk=0;
5
6     % naplnenie vektoru nulami na začiatku a jednotkami na konci
7     U(1:(p+1))=0;
8     U((n+1):(p+n+1))=1;
9
10    % prostredné hodnoty u dopočítame ekvidistantne
11    krok=1/(n-p);
12
13    for i=1:(n-p-1)
14        uk=uk+krok;
15        U((p+1+i):n)=uk;
16    end
17
18    % vytvoríme delenie intervalu podľa počtu sekvencií
19    u = linspace(0, 1, sekv);
20
21    k=zeros(size(u));
22    s=zeros(size(u));
23
24    % algoritmus
25    % pre každú sekvenciu vypočítame jej multiplicitu v uzlovom vektore a
    index, kde sa v ňom nachádza
26
27    for i=1:numel(u)
28
29        h=0;
30
31        for j=1:(numel(U))
32
33            % pokiaľ sa nachádza sekvencia u(i) medzi U(j) a U(j+1) uzlami,
            jej multiplicita je nulová a uložíme si jej index
34
35            if (U(j)<u(i)) && (U(j+1)>u(i))
36                k(i)=j;
37                s(i)=0;
38                break; % našli sme interval, v ktorom sa nachádza u(i)
39
40            % pokiaľ sa sekvencia rovná uzlu, spočítame jej multiplicitu a
            uložíme index prvého zhodného uzlu
41
42            elseif u(i)==U(j)
43                k(i)=j;
44                s(i)=s(i)+1;
45            end
46        end
47    end
48 end

```



Vstupné hodnoty do funkcie De Boor:

$P$  matica, ktorej jednotlivé riadky predstavujú riadiace body  
 $p$  stupeň krivky  
 $U$  uzlový vektor  
 $k$  index sekvencie  
 $s$  multiplicita sekvencie v uzlovom vektore  
 $u$  sekvencia

Nasledujúcu funkciu zavoláme pre každú sekvenciu delenia intervalu  $u_i$ , jej násobnosť  $s_i$  v uzlovom vektore, index  $k_i$  a potom parametre riadiace body  $P$ , uzlový vektor  $U$  a stupeň krivky  $p$ , ktoré sú rovnaké pre všetky  $u_i$ . B-spline bude potom aproximovať všetky riadiace body.

```
1 function [C]=deboor(P,p,U,k,s,u)
2
3 m=size(P,1); % dimenzia riadiacich bodov
4 n=size(P,2); % pocet riadiacich bodov
5
6 Pk = zeros(m,n,p);
7
8 % algoritmus
9
10 h=p-s;
11
12 if h>0
13
14     c= k-p;
15     b= k-s;
16     Pk(:, c:b, 1) = P(:, c:b); % vyberieme afektované body zo vstupných
17
18     for r=2:(h+1)
19         for j=(k-1-p+r):(k-s)
20
21             a(j,r)=(u-U(j))/(U(j+p-r+1)-U(j));
22             Pk(:,j,r)=(1-a(j,r))*Pk(:,j-1,r-1)+a(j,r)*Pk(:,j,r-1);
23
24         end
25     end
26
27     C = Pk(:,k-s,p-s+1); % získaný bod na krivke
28
29 elseif k == numel(U)
30
31     C = P(:,end); % posledný bod
32
33 else
34
35     C = P(:,1); % prvý bod
36
37 end
38 end
```

### 2.1.2 B-spline plocha

Výpočet jedného bodu na B-spline ploche vyžaduje dvojnásobne použitie De Boorovho algoritmu. V tejto časti si to priblížime. Zopakujme si definíciu B-spline plochy:

**Definícia 2.3.** Nech  $p$  a  $q$  sú stupne krivky. Majme  $m + 1$  riadkov a v každom riadku  $n + 1$  riadiacich bodov  $P_{i,j}$ , kde  $0 \leq i \leq m$  a  $0 \leq j \leq n$  a uzlové vektory  $U = \{u_0, \dots, u_r\}$   $V = \{v_0, \dots, v_s\}$  v nasledujúcom tvare:

$$U = \{0, \dots, 0, u_{p+1}, \dots, u_{r-p-1}, 1, \dots, 1\}$$

$$V = \{0, \dots, 0, v_{q+1}, \dots, v_{s-q-1}, 1, \dots, 1\},$$

(krajné uzly majú násobnosť  $p + 1$  v  $U$  a  $q + 1$  vo  $V$ ).

Nech  $\{u, v\}$  je dvojica čísel, pre ktoré platí:  $u \in \langle u_0, u_r \rangle$  a  $v \in \langle v_0, v_s \rangle$  a nech  $N_{i,n}(u)$  a  $N_{j,q}(v)$  sú bázové funkcie. Potom definujeme B-spline plochu takto:

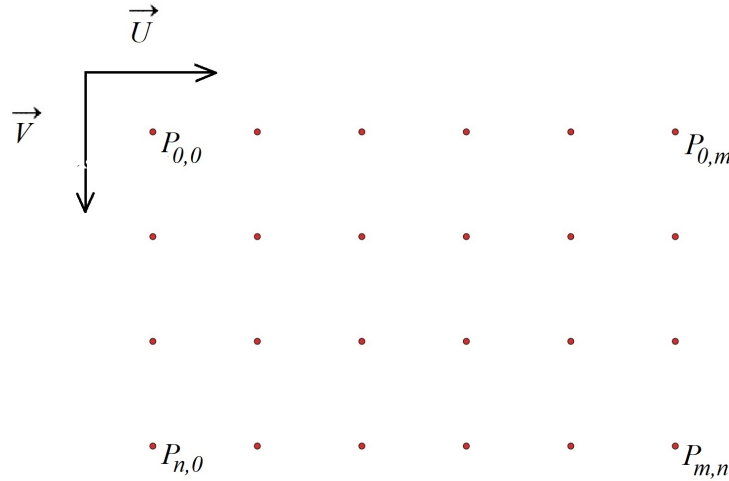
$$S(u, v) = \sum_{i=0}^n \sum_{j=0}^m N_{i,p}(u) N_{j,q}(v) P_{i,j} \quad (2.7)$$

$U$  má  $r + 1$  uzlov a  $V$  má  $s + 1$  uzlov. Indexy teda musia spĺňať nasledovné rovnice:

$$r = n + p + 1$$

$$s = m + q + 1 \quad (2.8)$$

*Poznámka.* Vstupné riadiace body  $P_{i,j}$  musia mať pri kolmom pohľade nasledujúcu štruktúru:



Obr. 7: Sieť bodov  $P_{i,j}$

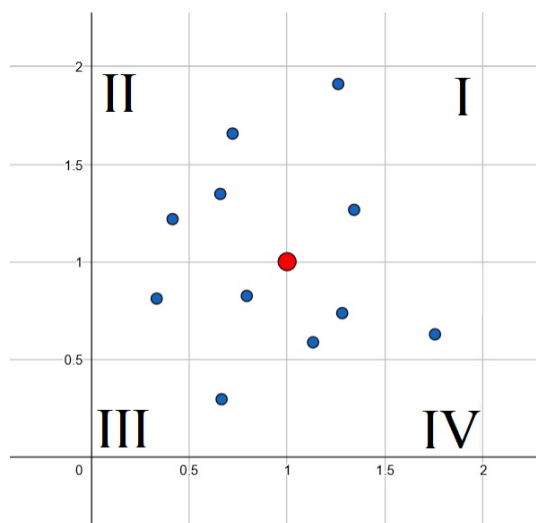
### 2.1.3 Mriežková štruktúra

Riadiace body, ktoré vstupujú do výpočtu B-spline plochy, musia mať požadovanú mriežkovú štruktúru. Namerané body sú väčšinou nepravidelne usporiadané, a preto ich potrebujeme upraviť do mriežky ako na obrázku obr.7. Výšku bodov mriežky spočítame štatistickou metódou najbližšieho suseda.

## Metóda najbližšieho suseda

Nasledujúci text je čerpaný zo zdroja [4]. Metóda najbližšieho suseda je jednoduchá interpolačná metóda, ktorá odhaduje neznámu hodnotu na základe hodnôt najbližšieho miesta, v ktorom údaj poznáme. Za hľadaný údaj teda budeme považovať známy údaj z najbližšieho miesta. Metódu si rozšírime tak, že budeme uvažovať najbližší údaj z každého kvadrantu.

Červený bod je bod mriežky a modré body sú namerané údaje. V každom kvadrante vyberieme modrý bod, ktorý je najbližšie k červenému a vypočítame ich vzdialenosť.



Obr. 8: Metóda najbližšieho suseda

Čím je modrý bod bližšie k červenému, tým je podstatnejší pre výpočet. Jednotlivým  $n$  minimálnym vzdialenostiam  $d_i$  pridáme ďalšiu hodnotu, a to váhu  $\lambda_i$ , pričom platí:

$$\sum_{i=1}^n \lambda_i = 1 \quad (2.9)$$

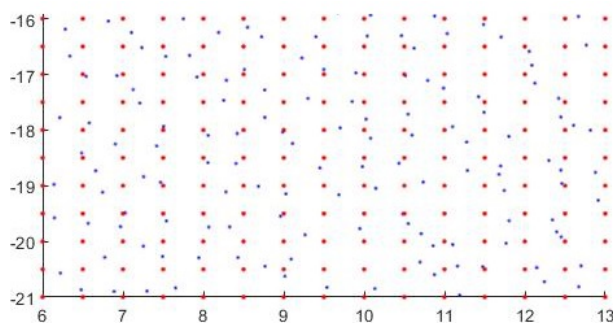
Váhu  $\lambda_i$  minimálnej vzdialenosti  $d_i$  vypočítame nasledovne:

$$\lambda_i = \frac{\frac{\sum_{k=1}^n d_k}{d_i}}{\sum_{j=1}^n \frac{\sum_{k=1}^n d_k}{d_j}} \quad (2.10)$$

Nakoniec hľadanú výšku  $v$  vypočítame ako lineárnu kombináciu váh  $\lambda_i$  jednotlivých vstupných bodov a ich výšok  $v_i$ :

$$v = \sum_{i=1}^n \lambda_i v_i \quad (2.11)$$

Upravená štruktúra vstupných bodov má tento tvar:



Obr. 9: Sieť upravených červených bodov  $P_{i,j}$

Pred vypočítaním mriežky pre vstupné body je nutné určiť, v ktorom smere ju budeme počítať. V tejto práci sme mriežku počítali v rovine  $xz$ , tým pádom výšky riadiacich bodov  $v$  boli ich  $y$ -ové súradnice.

Pokiaľ by sa mal výpočet mriežky zovšeobecniť, program by bol veľmi náročný, preto sme ho upravili podľa zvolených vstupov.

Prejdime na výpočet samotných bodov B-spline plochy. Vezmeme prvý riadok a vypočítame body B-spline krivky De Boorovým algoritmom pre zvolený počet deliacich sekvencií v tomto smere. Následne tento výpočet zopakujeme pre všetky vzniknuté stĺpce, tiež pre zvolený počet sekvencií v stĺpcovom smere. Takto získame body  $S(u, v)$  pre všetky dvojice  $\{u, v\}$ , kde  $u \in \langle u_0, u_r \rangle$  a  $v \in \langle v_0, v_s \rangle$ .

Vstupné hodnoty do funkcie:

P	body mriežky $P_{i,j}$
p	stupeň krivky v smere riadkov
q	stupeň krivky v smere stĺpcov
sekv1	počet deliacich sekvencií v smere riadkov
sekv2	počet deliacich sekvencií v smere stĺpcov

Funkcia B-spline plochy:

```

1 function [deb_col]=bsplineplocha(P,p,q,sekv1,sekv2)
2
3 ps=size(P,2); % počet stĺpcov
4 pr=size(P,3); % počet riadkov
5
6 [U,s,c]=knotvektor(pr,p,sekv2); % uzlový vektor a vektor multiplicity
   pre stĺpce
7 [V,t,d]=knotvektor(ps,q,sekv1); % uzlový vektor a vektor multiplicity
   pre riadky
8
9 u=linspace(0,1,sekv2); % parameter u
10 v=linspace(0,1,sekv1); % parameter v
11
12 deb_row=zeros(3,sekv2,pr);

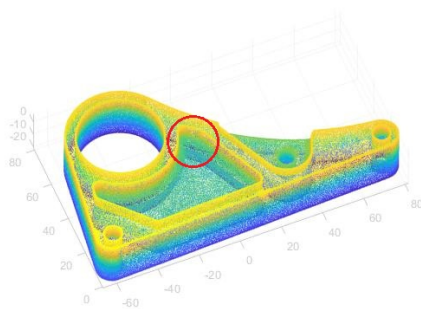
```

```

13
14 P_z=P(3,1,1:pr); % uložíme si súradnice z, pre jednotlivé riadky sú
    rovnaké, do De Boorovho algoritmu nevstupujú
15
16 % DEBOOROV ALGORITMUS PO RIADKOCH
17 for k=1:sekv1 %indexovanie podľa sekvencií
18
19     for i=1:pr % riadky
20         Pdyn=P(1:2, :, i); % do Pdyn si uložíme i-ty riadok
21         deb_row(1:2,k,i)=deboor(P(1:2, :, i),p,V,d(k),t(k),v(k)); % na i-ty
        riadok aplikujeme De Boorov algoritmus
22         deb_row(3,k,i)=P_z(i); % dopíšeme chýbajúcu súradnicu z
23     end
24
25 end
26
27 P_x=deb_row(1,1:sekv1,1); % uložíme si súradnice x, pre jednotlivé
    stĺpce sú rovnaké, do De Boorovho algoritmu nevstupujú
28
29 % DEBOOROV ALGORITMUS PO STĽPCOCH
30 for k=1:sekv1
31
32     for i=1:pr
33         Pdyn2(:,i)=deb_row(:,k,i); % do Pdyn2 si uložíme i-ty stĺpec
34     end
35
36     for i=1:sekv2 %indexovanie podľa sekvencií
37         deb2(2:3,i)=deboor(Pdyn2(2:3, :),q,U,c(i),s(i),u(i)); % na i-ty
        stĺpec aplikujeme De Boorov algoritmus
38     end
39
40     deb2(1,1:i)=P_x(k); % dopíšeme chýbajúcu súradnicu x
41     deb_col(:, :, k)=deb2; % v deb_col sú uložené hľadané body B-spline
        plochy
42 end

```

Vstupné body, ktoré boli použité na výpočet mriežky a potom B-spline plochy sme získali orezaním 3D skenu. Na nasledujúcich obrázkoch sa nachádza 3D sken.



(a)

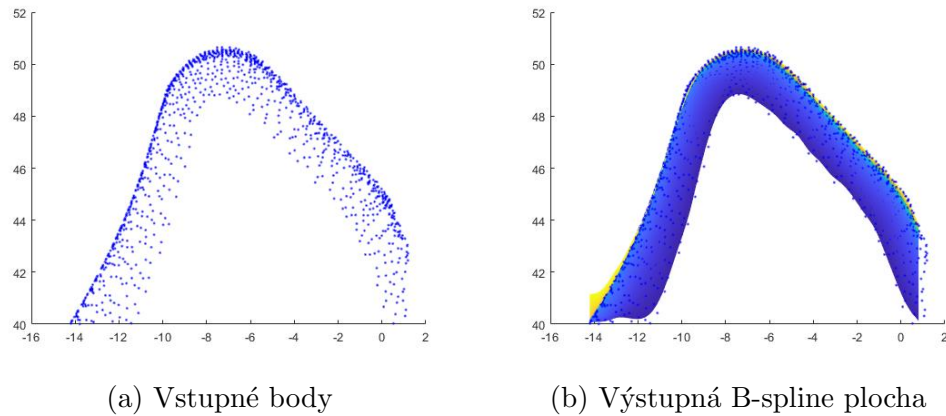


(b)

Obr. 10: 3D sken

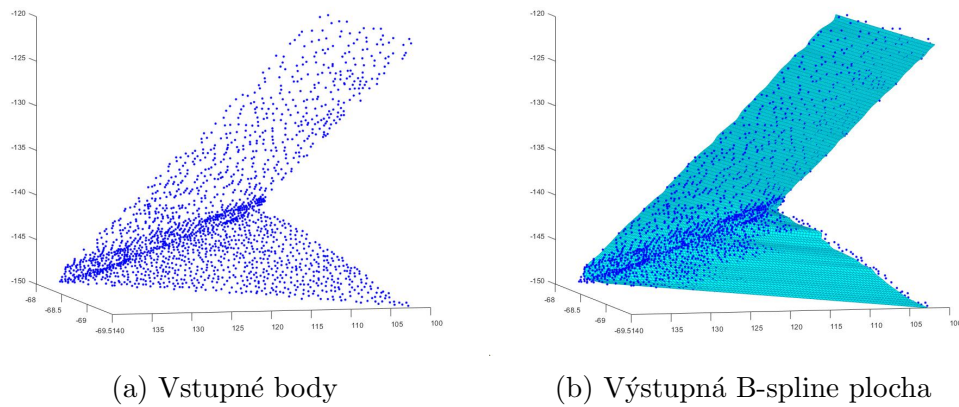
Červeným krúžkom je z neho označená zvolená oblasť, ktorá bola použitá na aproximáciu B-spline plochou.

Aproximovali sme B-spline plochu medzi bodmi z 3D skenu:



Obr. 11: Aproximácia vstupných bodov pomocou B-spline plochy

Funkcia sa dá použiť aj na iné vstupné body, pokiaľ sú vhodne otočené. Výšky bodov musia byť ich ypsilonové súradnice. Na nasledujúcich obrázkoch aproximujeme iné vstupné body B-spline plochou:



Obr. 12: Aproximácia vstupných bodov pomocou B-spline plochy

## Literatúra

- [1] SOBOTA, Branislav. *NURBS -NeUniformné Racionálne B-Spline* [online]. In: . 2015 [cit. 2021-04-11]. Dostupné z: [https://hornad.fei.tuke.sk/predmety/svr/doc/slides2015/SVR0715\\_NURBS.pdf](https://hornad.fei.tuke.sk/predmety/svr/doc/slides2015/SVR0715_NURBS.pdf)
- [2] SHENE, C.-K. *Introduction to Computing with Geometry Notes* [online]. [cit. 2021-04-11]. Dostupné z: <https://pages.mtu.edu/~shene/COURSES/cs3621/NOTES/>
- [3] THORNE, Tom. *Computer Graphics 17 - Curves and Surfaces 2* [online]. In: . [cit. 2021-04-11]. Dostupné z: <http://www.inf.ed.ac.uk/teaching/courses/cg/lectures/slides17.pdf>
- [4] KAŇUK, Ján. *Priestorové analýzy a MODELOVANIE*. Košice: Univerzita Pavla Jozefa Šafárika v Košiciach, 2015. ISBN 978-80-8152-290-1.
- [5] MARKOVÁ, Hana. *B-spline křivky*. Praha, 2007. Bakalářská práce. Univerzita Karlova v Praze. Vedoucí práce Doc. RNDr. Karel Najzar, CSc.
- [6] PIEGL, Les a Wayne TILLER. *The NURBS book*. 2nd ed. Berlin: Springer-Verlag, 1997. ISBN 35-406-1545-8.

## Zoznam príloh

### B-spline plocha

Funkcia *knotvektor* počíta uzlový vektor, multiplicitu a index sekvencie

Funkcia *deboor* počíta bod nachádzajúci sa na B-spline krivke

Funkcia *makegrid* vyrobí zo vstupných bodov pravidelnú mriežku

Funkcia *bsplineplocha* vyrobí z mriežky B-spline plochu

Súbor *vstupy\_bsplineplocha* otestuje funkciu *bsplineplocha* na vstupoch