

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra matematiky

Diplomová práce

T-spline objekty a jejich aplikace

Abstract

This master thesis deals with the theory of T-spline surfaces and has only two parts. The first part reviews fundamental definitions and features of Spline, Bézier, B-spline, and NURBS curves and surfaces, and introduces the definition of T-spline surface. And the second part focuses on finding the approximation of scattered 3D data points.

Prohlášení

Prohlašuji, že jsem diplomovou práci vypracovala samostatně a výhradně s použitím citovaných pramenů, uvedených na konci.

V Plzni dne

.....
Pešková Irena

Poděkování

Tímto bych ráda poděkovala vedoucímu diplomové práce Ing. Bohumíru Bastlovi, Ph.D. za ochotu, trpělivost, hodnotné rady, připomínky a celkové vedení práce.

Obsah

1	Objekty geometrického modelování	8
1.1	Spline křivky a plochy	8
1.1.1	Fergusonova kubika	9
1.1.2	Kubické spline křivky	9
1.1.3	Spline křivky stupně s	10
1.1.4	Spline plochy	11
1.2	Bézierovy křivky a plochy	11
1.2.1	Bézierovy křivky	11
1.2.2	Bézierovy plochy	13
1.3	B-spline křivky a plochy	14
1.3.1	B-spline báze	14
1.3.2	B-spline křivky	15
1.3.3	B-spline plochy	17
1.3.4	Hierarchická B-spline plocha	18
1.4	NURBS křivky a plochy	18
1.4.1	Racionalizace	18
1.4.2	NURBS křivky	19
1.4.3	NURBS plochy	20
1.5	T-spline	20
1.5.1	Uzlový interval	22
1.5.2	PB-spline plocha	23
1.5.3	Řídící síť T-spline plochy (T-mesh)	24
1.5.4	Standardní T-spline plochy	26
2	Aproximace neuspořádané množiny bodů v prostoru	27
2.1	Shrnutí současného stavu	27
2.2	Adaptivní aproximace T-spline plochou	28
2.2.1	Příprava dat	28
2.2.1.1	Konvexní obal a oblouky	28
2.2.1.2	Metoda nejmenších čtverců	29
2.2.1.3	Výpočet řídicí sítě	31
2.2.1.4	Uzlové pravidlo	32
2.2.2	Zjemňování sítě	33
2.2.2.1	Metoda nejmenších čtverců	33
2.2.2.2	Kumulativní chyba	34
2.2.2.3	Porušení pravidel při zjemňování	34

2.2.3	Zjemnění bázové funkce	36
2.2.4	Zakončení	39
2.2.5	Výsledky	40
2.2.5.1	Kumulativní chyba	40
2.2.5.2	Tvar T-spline	41
2.2.5.3	Doba výpočtu	44
2.2.6	Balíček TsplinePlocha	44

Úvod

CAD/CAM (Computer-Aided Design / Computer-Aided Manufacturing) programy, v překladu programy pro počítačem podporované navrhování, jsou v dnešní době hojně využívány. Jejich hlavním cílem je pomoci konstruktérovi vytvářet geometrický popis navrhovaného objektu, na němž později bude provádět simulace. Kupříkladu automobily jsou nyní navrhovány, vytvářeny a testovány (simulací) s pomocí počítače, čímž se předchází nákladnému a časově náročnému vytváření fyzických prototypů, přicházejícími s každou změnou návrhu, které byly potřeba dříve.

Geometrický popis objektu v počítači je obvykle konstruován pomocí nejzákladnějších tvarů, tzv. *primitiv*, která se liší na základě dimenze. Pro dvou-dimenzionální objekty jsou primitivy přímky a křivky, a pro tří-dimenzionální to jsou roviny a plochy. S těmito primitivy konstruktér manipuluje tak dlouho, dokud nevytvoří požadovaný tvar. Takovému procesu se říká modelování a výsledný objekt se nazývá model.

V této diplomové práci se věnujeme křivkám a plochám, jejichž primitiva mají nastavitelné komponenty, zvané *řídící body*. Řídící bod je tím nejzákladnějším prvkem, kterým konstruktér může manipulovat. Změnou pozice řídicích bodů se mění tvar odpovídající části primitiva. Řídící body křivek tvoří řídicí polygon a pro plochy je to řídicí síť.

V případě, že je objektu potřeba přidat větší detaily, je do primitiva vložen nový řídicí bod. Stinnou stránkou ale je, že příliš mnoho bodů model zbytečně komplikuje. Proto existuje i opačný postup - zjednodušení - kdy se odebírají body, které nenesou žádnou důležitou informaci, či jsou prostě navíc.

Hlavní náplní diplomové práce je použití T-spline plochy k aproximaci zadaných bodů v prostoru. T-spline plochy jsou poměrně novým způsobem geometrického modelování, který zobecňuje NURBS plochy tím, že v jejich řídicích sítích povolí existenci spojů ve tvaru T , proto název T-spline. T-spline plochy jsou ideální aproximační plochou hned ze dvou důvodů.

Za prvé, dovolují lokální zjemnění, takže se nové body nepropagují po celé řádce či sloupci, jako je tomu u B-spline ploch. Tímto je zaručeno, že řídicích bodů nebude mnoho, a co nejméně bude bodů nadbytečných, tj. bodů, které jsou v řídicí síti jen kvůli splnění pravidel a nenesou žádnou důležitou informaci.

Za druhé si T-spline plochy udržují spojitost třídy C^2 , a mohou být převedeny do NURBS reprezentace, což umožňuje jejich použití v komerčních softwarových systémech.

V první kapitole se seznámíme se základními prvky standardně používaných reprezentací v geometrickém modelování, jejich vlastnostmi a popisem, jako jsou Bézierovy, spline, B-spline a NURBS křivky a plochy, a v neposlední řadě také s tématem celé diplomové práce, T-spline plochami. Druhá část se věnuje popisu navržené metody pro

vytvoření plochy, která co nejlépe aproximuje zadané body v prostoru a výsledkům, kterých je možno touto metodou dosáhnout.

Kapitola 1

Objekty geometrického modelování

Zaměřením této diplomové práce jsou T-spline plochy a jejich získání ze zadaných bodů, proto si předtím musíme vysvětlit několik základních pojmů, které jsou potřeba k pochopení této problematiky. Podrobnější informace jsou k dispozici např. v [1, 2, 3, 4, 5, 12, 13].

1.1 Spline křivky a plochy

Spline je dostatečně hladká po částech definovaná polynomiální funkce $\mathbf{S} : [a, b] \rightarrow \mathbb{R}$. Na daném intervalu $[a, b]$ je složená z uspořádaných disjunktních podintervalů $[t_{i-1}, t_i]$, pro které platí

$$a = t_0 < t_1 < \dots < t_{k-1} < t_k = b.$$

V technických oborech patří spline křivky k nejčastěji používaným interpolačním křivkám.

Poznámka: Interpolace se v počítačové grafice používá k určení křivky, pro kterou jsou zadané body, nazýváme je opěrnými body. Interpolační křivka, na rozdíl od aproximační, všemi těmito body prochází. Zadanými body ale spline křivka není určena jednoznačně, a proto je ještě nutné přidat nějaké další informace, např. tečné vektory v bodech.

V minulosti se spline křivky hojně používaly pro konstrukce lodních trupů, kde konstruktérům sloužily jako matematický model pružného lačkového křívítka. Parametrické vyjádření spline křivky s opěrnými body $\mathbf{P}_0, \dots, \mathbf{P}_n$ a hodnotami t_0, \dots, t_n parametru pro tyto body je

$$\mathbf{P} = \mathbf{P}(t), \quad t \in \langle t_0, t_n \rangle,$$

a platí

$$\mathbf{P}_i = \mathbf{P}(t_i), \quad i = 0, \dots, n.$$

Podle volby parametrů máme buď *uniformní parametrizaci* interpolační křivky, kde $t_{i+1} - t_i = \text{konst}$, nejčastěji se v tomto případě volí $t_i = i$. Nebo *parametrizaci neuniformní*, např. chordálovou, v níž krok parametru pro oblouk spline křivky je dán

vzdáleností krajních bodů oblouku a je definována vztahem $t_{i+1} = t_i + k |\mathbf{P}_{i+1} - \mathbf{P}_i|$, kde k je jistá konstanta. Neuniformní parametrizace se používá pro silně nerovnoměrně rozložené opěrné body. Interpoláčnı křivku často tvoří několik oblouků křivky, protože pokud bychom chtěli křivku popsanou jedinou vektorovou funkcí s polynomičnými složkami, byly by tyto polynomy příliš vysokého stupně. Základem pro výpočet jednotlivých oblouků je tzv. Fergusonova kubika.

1.1.1 Fergusonova kubika

Nechť jsou dány vstupní body P_i a P_{i+1} , kterým odpovídají hodnoty parametru t_i a t_{i+1} , jejich polohové vektory \mathbf{P}_i a \mathbf{P}_{i+1} , a tečné vektory \mathbf{P}'_i a \mathbf{P}'_{i+1} v těchto bodech. Fergusonovu kubiku potom definujeme vztahem

$$\mathbf{P}(t) = H_0(t - t_i) \mathbf{P}_i + H_1(t - t_i) \mathbf{P}_{i+1} + H_2(t - t_i) \mathbf{P}'_i + H_3(t - t_i) \mathbf{P}'_{i+1}, \quad (1.1.1)$$

kde $H_i(s)$ jsou polynomy třetího stupně, které vyplývají z okrajových podmınek

$$\mathbf{P}(t_i) = \mathbf{P}_i, \quad \mathbf{P}(t_{i+1}) = \mathbf{P}_{i+1}, \quad \mathbf{P}'(t_i) = \mathbf{P}'_i, \quad \mathbf{P}'(t_{i+1}) = \mathbf{P}'_{i+1}. \quad (1.1.2)$$

$H_i(s)$ při značení ${}^i k = t_{i+1} - t_i$ a $s = t - t_i$ vypadají následovně (zdroj [1]):

$$\begin{aligned} H_0(s) &= \frac{2}{{}^i k^3} s^3 - \frac{3}{{}^i k^2} s^2 + 1 \\ H_1(s) &= -\frac{2}{{}^i k^3} s^3 + \frac{3}{{}^i k^2} s^2 \\ H_2(s) &= \frac{1}{{}^i k^2} s^3 - \frac{2}{{}^i k} s^2 + s \\ H_3(s) &= \frac{1}{{}^i k^2} s^3 - \frac{1}{{}^i k} s^2. \end{aligned}$$

1.1.2 Kubické spline křivky

Ze všech typů spline křivek mají kubické spline křivky největší význam. Kubickou spline křivkou pro opěrné body P_0, P_1, \dots, P_n , a hodnoty parametru $t \in \langle t_0, t_n \rangle$, $n = 0, 1, \dots, n$ rozumíme křivku $\mathbf{P} = \mathbf{P}(t)$, $t \in \langle t_0, t_n \rangle$, kde každá složka vektorové funkce $\mathbf{P}(t)$ je kubickou spline funkcí.

Jen opěrnými body ale kubická spline funkce není zadána jednoznačně, neboť každý z n kubických polynomů má 4 vektorové koeficienty, dohromady tedy $4n$, a k dispozici máme jen $4n - 2$ podmınek. $n + 1$ z interpolačních podmınek, $n - 1$ z C^0 spojitosti křivky, $n - 1$ ze spojitosti první a dalších $n - 1$ ze spojitosti druhé derivace. Proto je třeba ještě další dvě podmıinky k určení kubické spline funkce. Nejčastěji se jako tyto dodatečné podmıinky zadávají tečné vektory v počátečním a koncovém bodě křivky, nebo hodnoty druhých derivací v těchto bodech (speciální podmıinkou je „podmıinka volného konce“, kde v krajních bodech je druhá derivace nulová), nebo podmıinka uzavřnosti spline křivky.

Samotný výpočet kubické spline křivky se pak provádí ve dvou krocích: nejdříve určíme tečné vektory křivky v opěrných bodech a poté vypočteme jednotlivé oblouky křivky jako Fergusonovy kubiky podle vztahů (1.1.1) a (1.1.2). Pro provedení prvního kroku si označíme hledané tečné vektory v opěrných bodech jako \mathbf{P}'_i , $i = 0, \dots, n$. Z požadavku spojitosti druhé derivace získáme ze vztahů (1.1.1) a (1.1.2) po úpravách rovnici

$$\frac{1}{i_k} \mathbf{P}'_i + \left(\frac{2}{i_k} + \frac{2}{i+1_k} \right) \mathbf{P}'_{i+1} + \frac{1}{i+1_k} \mathbf{P}'_{i+2} = \frac{3}{i+1_k^2} \mathbf{P}_{i+2} + \left(\frac{3}{i_k^2} - \frac{3}{i+1_k^2} \right) \mathbf{P}_{i+1} - \frac{3}{i_k^2} \mathbf{P}_i,$$

$$i = 0, \dots, n-2, \quad i_k = t_{i+1} - t_i. \quad (1.1.3)$$

- Pokud dodatečnou podmínkou jsou vektory v počátečním a koncovém bodě, dostáváme $n-1$ rovnic pro $n-1$ neznámých, tudíž lineární soustavu rovnic o jednom řešení.
- Jsou-li dány vektory druhých derivací v počátečním a koncovém bodě, musíme doplnit soustavu (1.1.3) o dvě rovnice

$$2\mathbf{P}'_0 + \mathbf{P}'_1 = \frac{3}{0_k} (\mathbf{P}_1 - \mathbf{P}_0) - \frac{0_k}{2} \mathbf{A}$$

$$\mathbf{P}'_{n-1} + 2\mathbf{P}'_n = \frac{3}{n-1_k} (\mathbf{P}_n - \mathbf{P}_{n-1}) - \frac{n-1_k}{2} \mathbf{B}.$$

Speciálním případem je $\mathbf{A} = 0$, $\mathbf{B} = 0$, kdy dostáváme tzv. *kubický přirozený spline*.

- Podmínka uzavřenosti spline křivky rovněž přidává další dvě rovnice k soustavě (1.1.3). Rovnice získáme použitím obecné formule (1.1.3) pro $i = n-1$ a $i = n$ tak, že od indexů větších než n odečteme $n+1$.

K vyřešení získané lineární soustavy o $n+1$ rovnicích s $n+1$ neznámými používáme Gaussovu eliminační metodu.

1.1.3 Spline křivky stupně s

Spline křivky stupně s , $s \geq 1$, jsou chápány jako po částech polynomické funkce stupně s se spojitou derivací do stupně $s-1$. Z důvodu symetrie okrajových podmínek, což zaručuje nezávislost na orientaci křivky, se v praxi nejčastěji používají spline křivky lichého stupně. Každý segment křivky je určen sudým počtem podmínek, protože k jeho jednoznačnému určení jsou potřeba nejen krajní body, ale i vektory derivací v každém z nich.

Máme zadáno $n+1$ opěrných bodů $\mathbf{P}_0, \dots, \mathbf{P}_n$ pro n oblouků spline křivky, a potřebujeme tedy určit n polynomů stupně s . Každý polynom (oblouk křivky) je určen $s+1$ koeficienty, takže celkem potřebujeme $n(s+1)$ podmínek abychom pro dané body určili spline křivku stupně s . Interpoláčnící podmínky nám dávají $n+1$ rovnic a ze spojitosti nulté až $(s-1)$ -té derivace plyne $s(n-1)$ rovnic. Celkem tedy máme $n+1+s(n-1) = n+1+sn-s = n(s+1)+1-s$ rovnic. Je tedy zřejmé, že je třeba doplnit ještě $s-1$ okrajových podmínek.

Pokud je spline křivka stupně s uzavřená, je už jednoznačně určena svými opěrnými body a parametrizací.

1.1.4 Spline plochy

Spline plocha se řadí mezi plochy určené sítí bodů, tzn. je určena maticí bodů a okrajovými podmínkami, jimiž jsou příčné derivace v okrajových bodech a twisty v rohových bodech.

Pro výpočet spline plochy je třeba najít oba tečné vektory pro každý bod řídicí sítě, což se dá najít jako tečné vektory spline křivky dané sloupcem nebo řádkem matice bodů, a twisty neboli zkruty, což jsou smíšené parciální derivace v rohových bodech zadané sítě bodů.

1.2 Bézierovy křivky a plochy

1.2.1 Bézierovy křivky

Počátek Bézierových křivek se datuje na přelom 50. a 60. let dvacátého století, kdy dva francouzští muži, pracující ve dvou různých automobilkách, nezávisle na sobě publikovali své práce. Prvním byl matematik a fyzik Paul de Casteljaou pracující u automobilky Citroën, který roku 1959 vyvinul algoritmus, později pojmenovaný po něm algoritmus de Casteljaou, s nímž byl schopen spočítat Bézierovy křivky. Krátce poté, roku 1962, technik pro automobilku Renault Pierre Bézier přišel se svou studií křivek. Ta byla uznána širokou veřejností, načež po něm byly křivky pojmenovány. Bézier se snažil popsat tvarově složité plochy, na které v automobilovém průmyslu neustále narážel, co možná nejjednodušeji a aby se daly snadno modifikovat. Tímto položil základní kameny teorie těchto křivek.

Definice 1. Mějme lomenou čáru, nazývanou řídicí polygon, která se skládá z vrcholů V_0, \dots, V_n a jejich příslušných polohových vektorů $\mathbf{V}_0, \dots, \mathbf{V}_n$, kde $n \geq 1$ a $n \in \mathbb{Z}$. Pak vztah

$$\mathbf{P}(t) = \sum_{i=0}^n \mathbf{V}_i B_i^n(t), \quad t \in \langle 0, 1 \rangle,$$

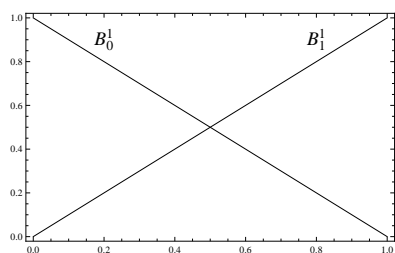
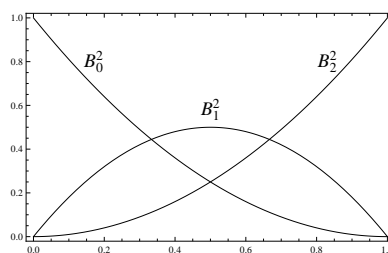
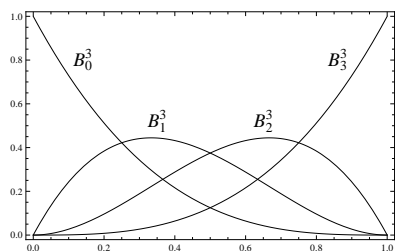
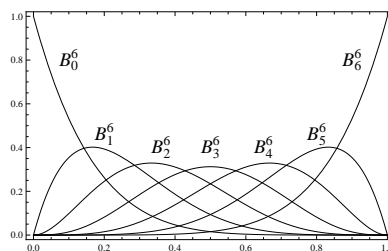
kde $B_i^n(t)$ jsou Bernsteinovy polynomy dané vztahem

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}, \quad i = 0, \dots, n,$$

definuje Bézierovu křivku. Pokud v $B_i^n(t)$ vznikne výraz $\binom{0}{0}$ nebo $\binom{n}{0}$ položíme ho rovný nule. Na obrázcích 1.2.1 (a), (b), (c) a (d) jsou zobrazeny úplné systémy Bernsteinových polynomů pro $n = 1, 2, 3, 6$.

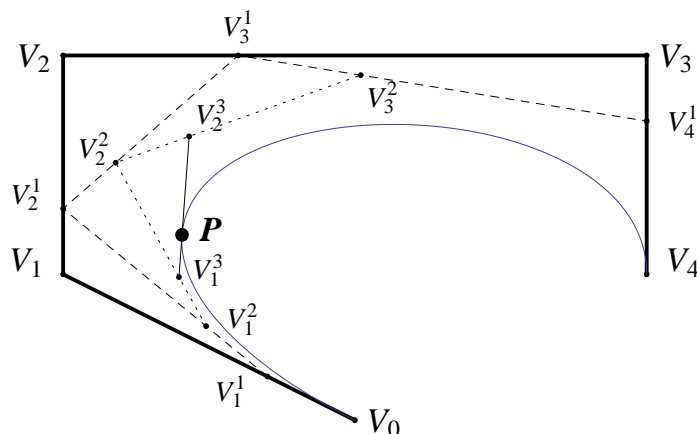
Vlastnosti Bézierových křivek

- Bézierova křivka začíná bodem V_0 , jehož polohový vektor je \mathbf{V}_0 a končí bodem V_n , který má polohový vektor \mathbf{V}_n .
- Tečnou Bézierovy křivky v počátečním bodě je první strana řídicího polygonu a obdobně v koncovém bodě je tečnou poslední strana řídicího polygonu.

(a) Bernsteinovy polynomy pro $n = 1$ (b) Bernsteinovy polynomy pro $n = 2$ (c) Bernsteinovy polynomy pro $n = 3$ (d) Bernsteinovy polynomy pro $n = 6$

Obrázek 1.2.1: Úplné systémy Bernsteinových polynomů

- Pokud jsou body řídicího polygonu kolineární, Bézierovou křivkou je úsečka V_0V_n .
- Neboť součet hodnot Bernsteinových polynomů se pro libovolné $t \in \langle 0, 1 \rangle$ rovná jedné, tak celá Bézierova křivka leží v konvexním obalu svého řídicího polygonu.
- Ze vztahu $B_i^n(t) = B_{n-i}^n(1-t)$, což vyjadřuje symetričnost Bernsteinových polynomů, plyne, že Bézierova křivka je nezávislá na orientaci řídicího polygonu.
- Bézierova křivka je invariantní vůči afinní transformaci, to znamená, že výsledná transformovaná Bézierova křivka je totožná s křivkou vzniklou z transformovaného řídicího polygonu.
- Počet průsečíků libovolné přímky a Bézierovy křivky je nejvýše roven počtu průsečíků přímky a řídicího polygonu křivky - variation diminishing property.
- Jestliže přidáme bod do řídicího polygonu nebo změním polohu některého bodu polygonu, změní se tvar celé křivky.
- Pokud aplikujeme postup výpočtu Bernsteinových polynomů na výpočet bodů Bézierovy křivky, dostaneme algoritmus de Casteljaeu. Tento algoritmus je založen na postupném dělení úseček řídicího polygonu v daném poměru (daném parametrem t_0), čímž vznikají nové body. Algoritmus opakujeme až do doby než nám vznikne jediný bod - bod křivky. Na obr. 1.2.2 je pro daný polygon tvořený body V_0, V_1, V_2, V_3, V_4 určen bod \mathbf{P} pro $t = 0.3$. Abychom získali všechny body Bézierovy křivky, je nutné provést algoritmus pro všechna $t \in \langle 0, 1 \rangle$.
 - Algoritmus de Casteljaeu lze také použít pro získání tečny Bézierovy křivky v daném bodě. Obecně, tečna v bodě $\mathbf{P}(t)$ je určena body $V_1^{n-1}(t)$ a $V_2^{n-1}(t)$.



Obrázek 1.2.2: Algoritmus de Casteljau

1.2.2 Bézierovy plochy

Bézierovy plochy se řadí mezi plochy určené sítí bodů.

Definice 2. Necht' je dána množina $(m + 1) \times (n + 1)$ navzájem různých bodů s polohovými vektory $\mathbf{V}_{0,0}, \mathbf{V}_{0,1}, \dots, \mathbf{V}_{m,n}$, kterou nazýváme řídicí síť a dá se popsat maticí

$$\mathbf{V} = \begin{pmatrix} \mathbf{V}_{0,0} & \cdots & \mathbf{V}_{0,n} \\ \vdots & \ddots & \vdots \\ \mathbf{V}_{m,0} & \cdots & \mathbf{V}_{m,n} \end{pmatrix}.$$

Bézierovu plochu pak definujeme předpisem

$$\mathbf{P}(u, v) = \sum_{i=0}^m \sum_{j=0}^n \mathbf{V}_{i,j} B_i^m(u) B_j^n(v), \quad u, v \in \langle 0, 1 \rangle,$$

kde $B_i^m(u)$ a $B_j^n(v)$ jsou Bernsteinovy polynomy.

Vlastnosti Bézierových ploch

- Bézierova plocha prochází rohovými body řídicí sítě (body $\mathbf{V}_{0,0}, \mathbf{V}_{0,n}, \mathbf{V}_{m,0}, \mathbf{V}_{m,n}$), a její okrajové křivky jsou Bézierovy křivky dané okrajovými lomenými čarami sítě.
- Stejně jako Bézierovy křivky i Bézierovy plochy jsou afinně invariantní a leží v konvexním obalu své řídicí sítě.
- Pro Bézierovy plochy už ovšem neplatí „variation diminishing property“, tj. počet průsečíků libovolné přímky a Bézierovy plochy není nikterak omezen počtem průsečíků této přímky s řídicí sítí plochy.

- Pro tečnou rovinu v rohových bodech Bézierovy plochy platí, že je určena tímto bodem a jeho sousedy. Tedy tečná rovina v rohovém bodě $\mathbf{V}_{m,n}$ je určena body $\mathbf{V}_{m-1,n}$, $\mathbf{V}_{m,n}$ a $\mathbf{V}_{m,n-1}$.
- Algoritmus de Casteljau se u ploch realizuje buď po plochách (*přímý algoritmus* de Casteljau), což je založeno na postupném generování bodů hyperbolických paraboloidů a na rekurentním vztahu

$$\mathbf{V}_{i,j}^{k+1} = (1-u, u) \begin{pmatrix} \mathbf{V}_{i,j}^k & \mathbf{V}_{i,j+1}^k \\ \mathbf{V}_{i+1,j}^k & \mathbf{V}_{i+1,j+1}^k \end{pmatrix} (1-v, v)^T,$$

$$k = 0, \dots, n-1, \quad i, j = 0, \dots, n-k,$$

či *po křivkách*, kde se využívá algoritmus de Casteljau pro křivky. Algoritmus aplikujeme nejdříve na sloupce matice \mathbf{V} , čímž získáme polygon bodů, jako kdybychom počítali $\mathbf{P}(u, v_0)$, na který opět aplikujeme algoritmus de Casteljau. Tímto jsme získali určení bodu $\mathbf{P}(u_0, v_0)$. Algoritmus se samozřejmě může nejprve aplikovat na řádky matice, výsledek bude stejný.

1.3 B-spline křivky a plochy

1.3.1 B-spline báze

Základem pro Bézierovy křivky a plochy byly Bernsteinovy polynomy, zato pro B-spline křivky se definují báze funkce po částech s tím, že tyto funkce jsou spline funkcemi, tj. jsou to po částech polynomické funkce se „spojitou derivací co do nejvyššího řádu“.

Označme $\mathbf{T} = (t_0, \dots, t_m)$ tzv. *vektor parametrizace* (uzlový vektor) neklesající posloupnost parametrů t_i , tj. $t_0 \leq t_1 \leq \dots \leq t_m$. Hodnoty t_i nazýváme uzly.

B-spline báze funkce N_i^k stupně k pro vektor parametrizace $\mathbf{T} = (t_0, \dots, t_m)$ jsou definovány rekurentním předpisem:

- pro $k = 0$

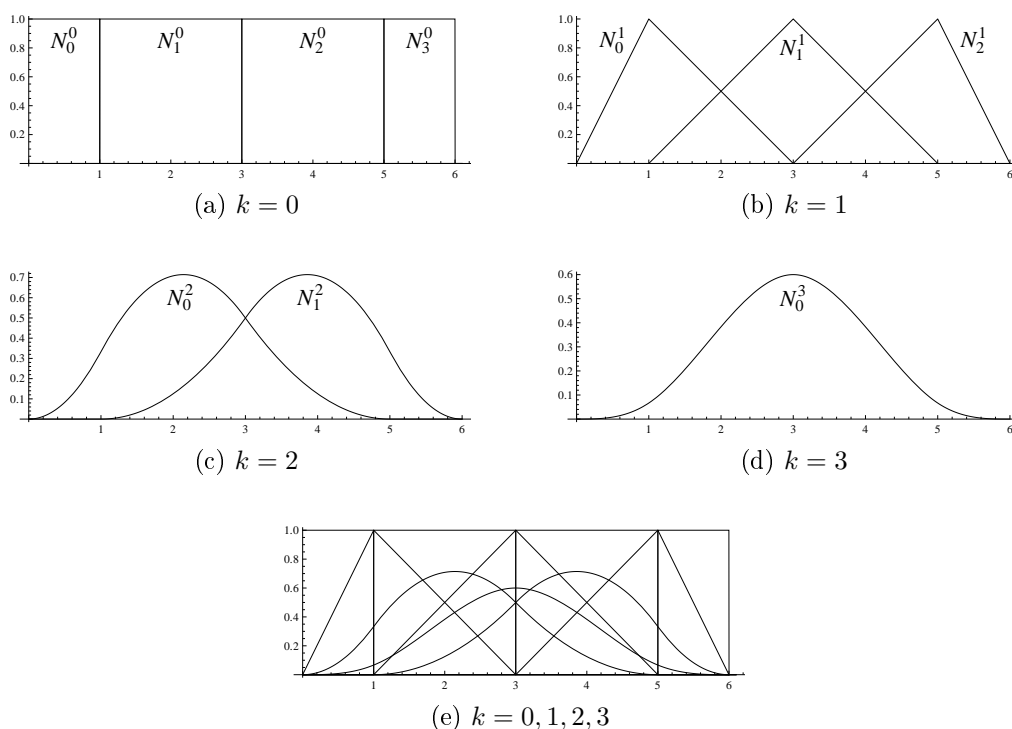
$$N_i^0(t) = \begin{cases} 1 & \text{pro } t_i \leq t < t_{i+1} \\ 0 & \text{jinde} \end{cases}$$

- pro $k > 0$

$$N_i^k(t) = \frac{t - t_i}{t_{i+k} - t_i} N_i^{k-1}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} N_{i+1}^{k-1}(t), \quad (1.3.1)$$

kde $t \in \langle t_0, t_m \rangle$. V případě, že vzniknou výrazy typu $\frac{a}{0}$, položíme je rovny nule. Na obrázku 1.3.1 (a), (b), (c), (d) je zobrazeno, jak vypadají báze funkce pro vektor parametrizace $\mathbf{T} = (0, 1, 3, 5, 6)$ a stupně $k = 0, 1, 2, 3$, a na (e) jsou pro porovnání všechny dohromady.

B-spline báze je tedy charakterizována: stupněm k polynomů, vektorem parametrizace, který má $m+1$ uzlů $t_0 \leq t_1 \leq \dots \leq t_m$, a číslem j , které znázorňuje počet funkcí tvořících bázi. Ze vztahu (1.3.1) plyne, že pro funkci $N_i^k(t)$ je potřeba mít ve vektoru



Obrázek 1.3.1: B-spline bázové funkce

parametrizace až složku s indexem $i + k + 1$. Jelikož i nabývá maximální hodnoty j , platí vazba $m = j + k + 1$, nebo-li $m + 1 = j + k + 2$, což znamená, že počet složek vektoru parametrizace ($m + 1$) je roven součtu stupně B-spline báze (k), počtu funkcí tvořících bázi ($j + 1$) a jedničky.

Pokud by složky parametrického vektoru byly tvaru $t_0 = t_1 = \dots = t_n = 0$ a $t_{n+1} = t_{n+2} = \dots = t_{2n+1} = 1$, tedy vektor parametrizace by měl tvar $\mathbf{T} = (0, \dots, 0, 1, \dots, 1)$, pak B-spline bázi jsou Bernsteinovy polynomy.

Vlastnosti bázových funkcí

- Pro všechna t, i, k jsou bázové funkce nezáporné: $N_i^k(t) \geq 0$.
- Součet všech bázových funkcí je roven jedné: $\sum_{i=0}^p N_i^k(t) = 1, t \in \langle t_0, t_m \rangle$.

1.3.2 B-spline křivky

B-spline křivka stupně k s řídicím polygonem $\mathbf{V}_0, \dots, \mathbf{V}_n, k \leq n$ a vektorem parametrizace $\mathbf{T} = (t_0, \dots, t_m)$ je definována vztahem

$$\mathbf{P}(t) = \sum_{i=0}^n \mathbf{V}_i N_i^k(t). \quad (1.3.2)$$

Vektor parametrizace

Obecně rozlišujeme periodický a neperiodický tvar vektoru parametrizace. Zatímco periodický vektor nemá žádnou speciální vnitřní strukturu, pro neperiodický vektor parametrizace platí

$$\mathbf{T} = \left(\underbrace{a, \dots, a}_{k+1}, t_{k+1}, \dots, t_{m-k-1}, \underbrace{b, \dots, b}_{k+1} \right).$$

Jak už z názvu vypovídá, uzavřené křivky mají periodický vektor parametrizace a otevřené neperiodický.

Parametrizace může být buď *uniformní*, a to pokud pro dva sousední uzly platí $t_{i+1} - t_i = konst$, nebo *neuniformní*, což je např. parametrizace, která respektuje poměry délek stran řídicího polygonu.

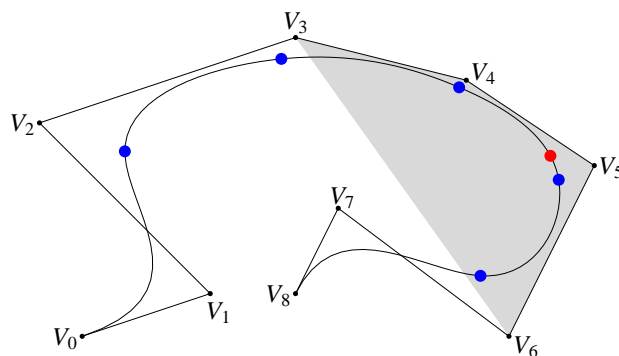
Vlastnosti B-spline křivek

- B-spline křivka stupně k je spline křivkou, tzn. obecně je třídy C^{k-1} a tedy je spojitá až do $(k-1)$ -ní derivace. Třídou spojitosti lze ve vybraných bodech i snížit, a to uvedením násobných hodnot ve vektoru parametrizace. Pro takové křivky pak platí, že jsou třídy C^{k-q} , kde q je násobnost uzlu.
- Všechny body B-spline křivky leží v konvexním obalu zadaného řídicího polygonu a také platí, že pokud t je z intervalu $\langle t_i, t_{i+1} \rangle$, pak bod křivky $\mathbf{P}(t)$ leží v konvexním obalu $(k+1)$ po sobě jdoucích vrcholů $\mathbf{V}_{i-k}, \mathbf{V}_{i-k+1}, \dots, \mathbf{V}_i$, zdroj [13].

– Příklad:

Na obr. 1.3.2 je kubická B-spline křivka s řídicím polygonem o devíti bodech $\mathbf{V}_0, \dots, \mathbf{V}_9$ a parametrickém vektoru $\mathbf{T} = (0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 6, 6, 6)$. Jelikož parametrický vektor obsahuje šest skoků uzlových hodnot, skládá se křivka ze šesti oblouků. Uzly jsou označeny modře. Bod s $t = 3.8$ (označen červeně) spadá do uzlového intervalu $\langle t_i, t_{i+1} \rangle = \langle 3, 4 \rangle$, takže $(k+1) = 4$ B-spline báze funkce, které jsou na tomto intervalu nenulové jsou $N_3^3, N_4^3, N_5^3, N_6^3$ a ze vztahu (1.3.2) je zřejmé, že to jsou koeficienty bodů $\mathbf{V}_3, \mathbf{V}_4, \mathbf{V}_5, \mathbf{V}_6$. Z vlastnosti součtu B-spline báze funkcí rovnému jedné plyne, že jejich „vážený“ průměr $\mathbf{P}(t)$ leží v konvexní oblasti definované těmito řídicími body.

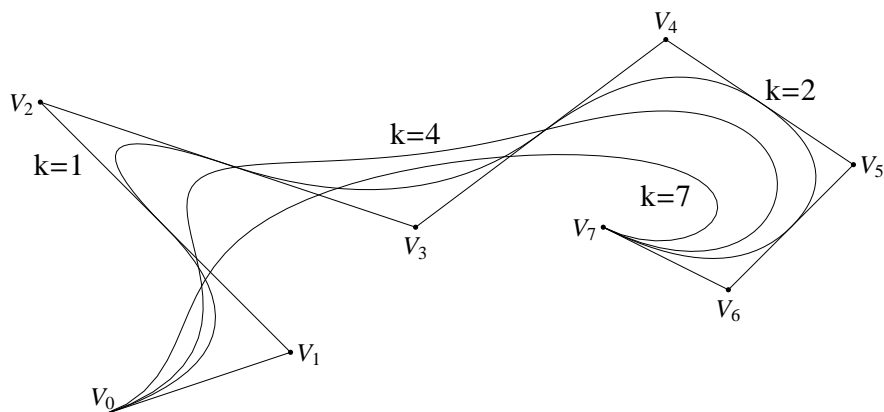
- Obdobně jako u Bézierových křivek i zde platí afinní invariantnost, tzn. výsledná transformovaná křivka je totožná s křivkou vytvořenou z transformovaného řídicího polygonu.
- Vliv změny polohy řídicích bodů je *lokalizován*, tzn. poloha bodu \mathbf{V}_s ovlivňuje tvar křivky pouze v intervalu $\langle t_s, t_{s+k} \rangle$ a nedochází tedy ke změně celé křivky.
- Algoritmus de Casteljau má pro B-spline křivky obdobu v *de Boorově algoritmu*. Ten ale neprovádí dělení stran řídicího polygonu v konstantním poměru jako je



Obrázek 1.3.2: Konvexní obal

tomu u algoritmu de Casteljau, ale je proměnný a závisí na rozdílu hodnot ve vektoru parametrizace.

Na obrázku 1.3.3 jsou pro daný řídicí polygon o osmi bodech vykresleny B-spline křivky o stupních 1, 2, 4 a 7. Povšimněme si, že pro $k = 1$ je B-spline křivkou samotný řídicí polygon a s rostoucím stupněm se křivka více vyhlazuje. Pokud se stupeň křivky rovná počtu stran řídicího polygonu, v tomto případě pro $k = 7$, je výslednou B-spline křivkou Bézierova křivka pro stejný řídicí polygon.

Obrázek 1.3.3: B-spline křivky o stupních $k = 1, 2, 4, 7$

B-spline křivkami se dají modelovat jen objekty, které mají po částech polynomiální vyjádření, což znamená, že kružnice, hyperboly a jejich oblouky nelze pomocí B-spline reprezentovat. Proto se zavádí racionální specializace (viz. odstavec 1.4.1).

1.3.3 B-spline plochy

Nechť máme řídicí síť $(m + 1) \times (n + 1)$ bodů, které jsou popsány polohovým vektorem $\mathbf{V}_{i,j}$, a vektory parametrizace \mathbf{U}, \mathbf{V} pro parametry u a v . Pro obecnou B-spline plochu stupně (k, l) , kde k je stupeň polynomů B-spline báze pro parametr u , a l je stupeň polynomů báze pro parametr v , potom platí vztah

$$\mathbf{P}(u, v) = \sum_{i=0}^m \sum_{j=0}^n \mathbf{V}_{i,j} N_i^k(u) N_j^l(v), \quad (1.3.3)$$

kde $N_i^k(u)$, resp. $N_j^l(v)$ jsou B-spline bázové funkce stupně k pro parametr u , resp. stupně l pro parametr v .

Vlastnosti B-spline ploch

- Hraničními křivkami jsou B-spline křivky dané lomenou čarou hraničních bodů.
- I zde platí podmínka konvexního obalu.
- De Boorův algoritmus aplikovaný po křivkách dokáže generovat body B-spline plochy.

1.3.4 Hierarchická B-spline plocha

Pro B-spline křivky platí, že jemnější detaily lze získat vložení uzlu do vektoru parametrizace, což vyvolá pouze lokální změnu křivky. Naproti tomu, pokud vložíme uzel do vektorů parametrizace B-spline plochy, způsobí to vytvoření celé nové řádky nebo sloupce řídicích bodů. Aby se překonala tato nedokonalost, Forsey a Bartels v článku [9] představili pojem hierarchických B-spline ploch. Hierarchické B-spline plochy jsou tedy zobecněním B-spline ploch, které umožňují lokální zjemňování.

Hierarchická B-spline plocha se skládá ze série *úrovní*, které jsou složeny z několika B-spline plátů. Nejspodnější úroveň 0 má nejnížší rozlišení a popisuje pouze základní tvar plochy. Každý jeho další potomek, zvaný *překrytí*, popisuje určitou vlastnost plochy.

Nechť je dáno překrytí úrovně k s řídicími body $\mathbf{P}^{(k)}$, potom každý bod překrytí je popsán tvarem „reference plus offset“, tj. $\mathbf{P}^{(k)} = \mathbf{R}^{(k-1)} + \mathbf{O}^{(k)}$, kde reference $\mathbf{R}^{(k-1)}$ je získána z rodičovské plochy na úrovni $k - 1$, a $\mathbf{O}^{(k)}$ je vektor posunu (offset). Z toho plyne, že jakákoliv změna na nižších úrovních je ihned propagována do všech úrovní nad ní.

Hierarchickou B-spline plochu lze transformovat do T-spline plochy, a to spojením všech jejích úrovní do jedné plochy, jejíž řídicí síť obsahuje pouze pár nadbytečných bodů a zároveň zachovává tvar výsledné plochy.

1.4 NURBS křivky a plochy

1.4.1 Racionalizace

Protože se Bézierovými a B-spline křivkami a plochami nedají reprezentovat některé objekty např. kuželosečky a kvadriky, je třeba definovat postup, který nám to umožní. Tomuto postupu se říká racionalizace. Podstatou racionalizace v našem podání je využití projektivního rozšíření Euklidovského prostoru, tedy prostoru využívajícího homogenních souřadnic. Tato homogenizující složka souřadnic bodů se nazývá *váha*. Po

projektivním rozšíření prostoru je navíc možné mít v řídicím polygonu či síti nevlastní body.

Nechť máme bod $\mathbf{P} = (x, y, z)$, potom jeho souřadnice po přechodu do projektivně rozšířeného prostoru jsou $\mathbf{P}^w = (x', y', z', w)$, kde w je daná váha a pro původní souřadnice platí převody $x = \frac{x'}{w}$, $y = \frac{y'}{w}$ a $z = \frac{z'}{w}$.

1.4.2 NURBS křivky

NURBS (Non-Uniform Rational B-Spline) je definován jako B-spline v projektivním rozšíření Euklidovského prostoru. V počítačovém modelování se těchto objektů používá nejčastěji (CAD systémy).

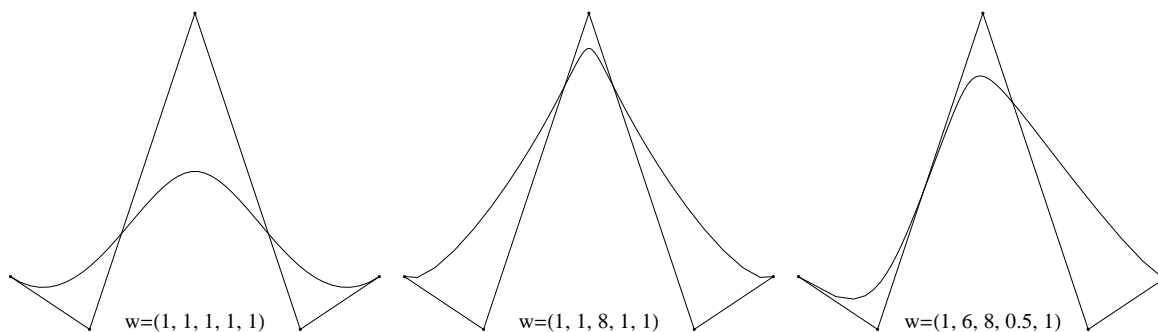
Mějme polohové vektory pro body řídicího polygonu $\mathbf{P}_0, \mathbf{P}_1, \dots, \mathbf{P}_m$, kde $\mathbf{P}_i = (x_i, y_i)$, jejich váhy w_0, w_1, \dots, w_m , kde $w_i \neq 0$, a vektor parametrizace $\mathbf{T} = (t_0, \dots, t_q)$. NURBS křivku stupně k potom definujeme s pomocí B-spline báze funkcí $N_i^k(t)$ jako

$$\mathbf{P}(t) = \frac{\sum_{i=0}^m \mathbf{P}_i w_i N_i^k(t)}{\sum_{i=0}^m w_i N_i^k(t)}.$$

Vlastnosti NURBS křivek

- Možnost konstrukce kuželoseček.
- Stejně jako B-spline křivka i NURBS křivky procházejí prvním a posledním bodem řídicího polygonu.
- Jsou projektivně invariantní.
- Lokalizace změn. Při změně polohy bodu \mathbf{P}_s je ovlivněn tvar křivky pouze v intervalu $\langle t_s, t_{s+k} \rangle$ a ne celé křivky.
- Platí „variation diminishing property” - počet průsečíků libovolné přímky a NURBS křivky je nejvýše roven počtu průsečíků přímky a řídicího polygonu křivky.
- Přidáním parametru váhy se zvyšuje počet možností jak křivku modifikovat.
- Pokud jsou váhy $w_i \forall i$ kladné, tak NURBS křivka leží v konvexním obalu svého řídicího polygonu, je-li ale jediná váha záporná, už to neplatí.
- Pro konstrukci se používá de Boorův algoritmus.

Na obrázku 1.4.1 jsou vidět NURBS křivky s různými vektory vah. Pokud $w_i = 1 \forall i$ je křivkou B-spline křivka. Při $w_i > 1$ se křivka k bodu \mathbf{V}_i přibližuje, a při $w_i \in (0, 1)$ se oddaluje.



Obrázek 1.4.1: Různé váhy bodů

1.4.3 NURBS plochy

Mějme řídicí síť o $(m + 1) \times (n + 1)$ bodech, které jsou popsány polohovým vektorem $\mathbf{V}_{i,j}$, jejich váhy $w_{i,j}$ a vektory parametrizace \mathbf{U}, \mathbf{V} pro parametry u a v . NURBS plocha je definována vztahem

$$\mathbf{P}(u, v) = \frac{\sum_{i=0}^m \sum_{j=0}^n \mathbf{V}_{i,j} w_{i,j} N_i^k(u) N_j^l(v)}{\sum_{i=0}^m \sum_{j=0}^n w_{i,j} N_i^k(u) N_j^l(v)},$$

kde $N_i^k(u)$, resp. $N_j^l(v)$ jsou B-spline báze funkce stupně k pro parametr u , resp. stupně l pro parametr v .

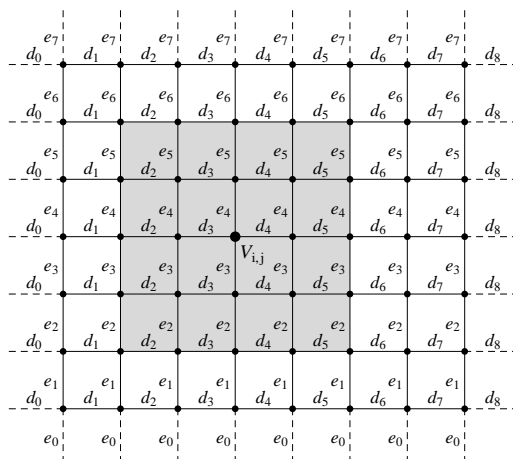
Vlastnosti NURBS ploch

- Jelikož NURBS plochy mají za základ B-spline báze plochy, sdílí s nimi mnoho vlastností.
- Váha řídicího bodu $\mathbf{V}_{i,j}$ ovlivňuje tvar plochy v oblasti $\langle u_i, u_{i+k+1} \rangle \times \langle v_j, v_{j+l+1} \rangle$.
- Obdobně jako u NURBS křivek platí lokalizace změn. Bod $\mathbf{V}_{i,j}$ ovlivňuje tvar plochy pouze v konečné oblasti řídicí sítě. Příklad, jaká oblast to je pro řídicí síť kubické NURBS plochy, je na obrázku 1.4.2. Báze funkce $N_i^3(u)$ závisí na uzlových intervalech (e_2, e_3, e_4, e_5) a báze funkce $N_j^3(v)$ závisí na (d_2, d_3, d_4, d_5) . Co vlastně takovéto označení intervalů je, a jak se zjišťuje, je popsáno v odstavci 1.5.1.

1.5 T-spline

NURBS plochy mají dvě hlavní nevýhody:

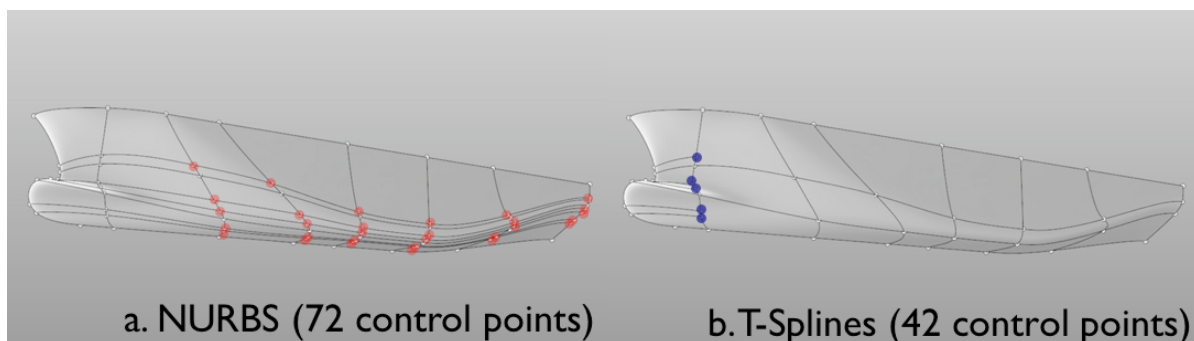
- Topologii řídicí sítě. Sice je snadné ji matematicky popsat, ale může vést k vzniku mnoha nepotřebných bodů, které nenesou žádnou důležitou informaci a existují jen kvůli splnění topologických vazeb NURBS plochy.

Obrázek 1.4.2: Oblast vlivu bodu $V_{i,j}$

- Parametrizace NURBS ploch nedovoluje popsání některých běžných tvarů, a pokud už, tak často ne jen jedinou plochou. NURBS plochy jsou homeomorfní (homeomorfismus je vzájemně jednoznačné zobrazení mezi topologickými prostory, které zachovává topologické vlastnosti) k rovinám, otevřeným válcovým plochám a anuloidům, takže například NURBS reprezentace kulové plochy je ve skutečnosti válec se sbalenými konci a ne uzavřená plocha.

T-spline plochy, představeny široké veřejnosti ve článku [5], jsou novou matematickou formulací ploch, které tato omezení NURBS ploch odbourávají, protože nejen že, nemusí být zadané pouze na obdélníkových oblastech, ale hlavně výrazně snižují počet nadbytečných řídicích bodů, což zlepšuje následnou manipulaci s modelem. To, co nejvíce odlišuje řídicí síť T-spline plochy (zvanou T-mesh) a řídicí síť pro NURBS je fakt, že T-mesh dovoluje zakončení po sobě jdoucích řídicích bodů v řádku či sloupci, což znamená vznik T-spoje (T-junction).

Na obr. 1.5.1 je vidět rozdíl mezi plochou trupu lodi modelovanou jako NURBS plocha a T-spline plocha. Červeně jsou označeny nadbytečné body a modrou jsou zvýrazněny T-spoje.



Obrázek 1.5.1: Rozdíl mezi NURBS a T-spline plochami, zdroj [8]

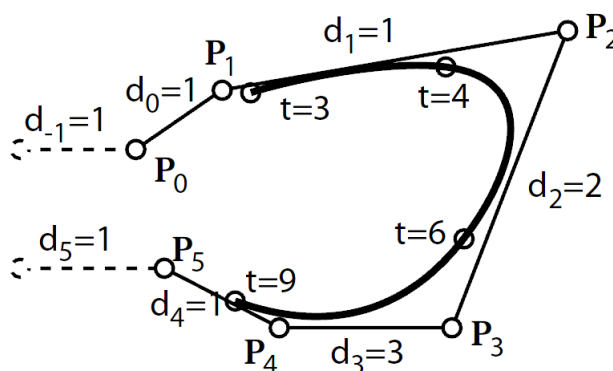
Další vlastnosti T-spline ploch

- Jelikož je T-spline plocha založena na stejných básových funkcích jako NURBS plochy, sdílí s nimi užitečné vlastnosti jako
 - lokalizace změn,
 - variation diminishing property,
 - vlastnosti týkající se spojitosti.
- Všechny vnitřní body řídicí sítě NURBS plochy mají valenci 4, tzn. že z daného bodu v kontrolní síti vycházejí čtyři hrany. T-mesh narozdíl od toho umožňuje i valenci 3, tzn. T-spoj.

1.5.1 Uzlový interval

Jak již víme, složky vektoru parametrizace, jinak zvaného uzlového vektoru, tedy uzly, se používají k určení B-spline básových funkcí. Rozdíl hodnot dvou po sobě jdoucích uzlů definuje nový pojem, uzlový interval, který je zároveň délkou parametru Bézierovy křivky, která se nachází mezi těmito dvěma uzly. Toto nezáporné číslo je přiděleno ke každé hraně v řídicí síti T-spline plochy. Pojem uzlový interval (knot interval) byl představen ve článku [4].

Na obrázku 1.5.2 z článku [5] je příklad kubické B-spline křivky o šesti řídicích bodech. Uzly jsou značeny t a hodnoty d_i podél řídicího polygonu znázorňují uzlové intervaly. Vektor parametrizace pro tuto křivku má tvar $\mathbf{T} = (1, 2, 3, 4, 6, 9, 10, 11)$. Hodnota prvního a posledního uzlového intervalu pro neuzavřené křivky se přiřazuje imaginárním hranám řídicího polygonu, na obrázku to jsou d_{-1} a d_5 .



Obrázek 1.5.2: Kubický B-spline, zdroj [5]

Jde provést i opačný proces, a to určit složky vektoru parametrizace na základě uzlových intervalů. Jednoduše přiřadíme prvnímu uzlu nějakou hodnotu, a abychom dostali hodnotu druhého uzlu, k ní připočteme první uzlový interval, pro třetí uzel připočteme k hodnotě druhého uzlu druhý uzlový interval atd.

1.5.2 PB-spline plocha

Tato plocha je založena výhradně jen na svých řídicích bodech, tzn. body mezi sebou nemají žádný topologický vztah. Odsud je odvozen název **P**oint **B**ased.

K definici PB-spline plochy je tedy potřeba množina řídicích bodů a pro každý bod dvojice uzlových vektorů. Uzlový vektor v proměnné s je pro bod \mathbf{P}_i definován jako $s_i = [s_{i0}, s_{i1}, s_{i2}, s_{i3}, s_{i4}]$, a v proměnné t jako $t_i = [t_{i0}, t_{i1}, t_{i2}, t_{i3}, t_{i4}]$. Jejich získání je zobrazeno na obrázku 1.5.3.

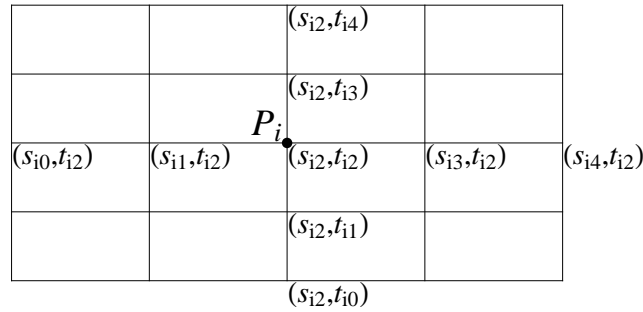
Nechť \mathbf{P}_i jsou zadané řídicí body a $\mathbf{D}_i = (s_{i0}, s_{i4}) \times (t_{i0}, t_{i4})$ je oblast vlivu bodu \mathbf{P}_i . Pro PB-spline plochu potom platí vztah

$$\mathbf{P}(s, t) = \frac{\sum_{i=1}^n \mathbf{P}_i B_i(s, t)}{\sum_{i=1}^n B_i(s, t)}, \quad (s, t) \in \mathbf{D}, \quad (1.5.1)$$

kde $B_i(s, t)$ jsou bázové funkce definované jako

$$B_i(s, t) = N_{i0}^3(s) N_{i0}^3(t), \quad (1.5.2)$$

kde $N_{i0}^3(s)$, resp. $N_{i0}^3(t)$ jsou kubické B-spline bázové funkce pro vektor $s_i = [s_{i0}, s_{i1}, s_{i2}, s_{i3}, s_{i4}]$, resp. $t_i = [t_{i0}, t_{i1}, t_{i2}, t_{i3}, t_{i4}]$ dané (1.3.1).



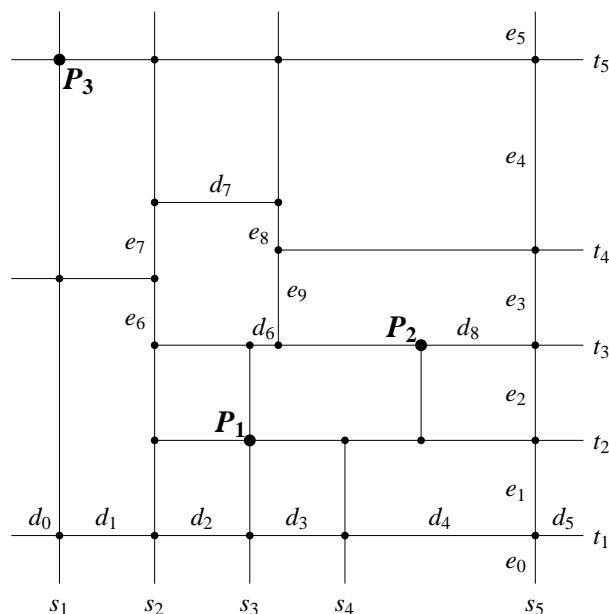
Obrázek 1.5.3: Sestavení uzlového vektoru, zdroj [5]

Vlastnosti PB-spline plochy

- Splňuje vlastnost konvexního obalu, pokud si označíme $C(s, t) = \{\mathbf{P}_i | (s, t) \in \mathbf{D}_i\}$ pak $\mathbf{P}(s, t)$ leží v konvexním obalu $C(s, t)$.
- \mathbf{D} v rovnici 1.5.1 je oblast vlivu, kde je definovaná celá PB-spline plocha. To znamená, že $\mathbf{D} \subset \{\mathbf{D}_1 \cup \mathbf{D}_2 \cup \dots \cup \mathbf{D}_n\}$, \mathbf{D} ale nemusí být obdélníková oblast. Z vlastnosti konvexního obalu plyne následující: Pokud $(s, t) \in \mathbf{D}$ leží pouze v jedné oblasti vlivu \mathbf{D}_i , pak $\mathbf{P}(s, t) = \mathbf{P}_i$, pokud $(s, t) \in \mathbf{D}$ leží pouze ve dvou oblastech vlivu \mathbf{D}_i a \mathbf{D}_j , pak $\mathbf{P}(s, t)$ je úsečka spojující \mathbf{P}_i a \mathbf{P}_j , atd. Tudíž je vhodné, aby každý bod v \mathbf{D} ležel v alespoň třech oblastech vlivu \mathbf{D}_i .
- $N_{i0}^3(s)$ a $N_{i0}^3(t)$ jsou neměnicí se části bázových funkcí PB-spline plochy. Jelikož je PB-spline plocha založená na bodech a jejich uzlových vektorech, jsou $N_{i0}^3(s)$ a $N_{i0}^3(t)$ jedinečné pro každý bod. Naproti tomu u NURBS ploch platí, že kubická B-spline bázová funkce $N_{i0}^3(s)$ je stejná pro všechny řídicí body v sloupci, a $N_{i0}^3(t)$ je stejná pro všechny body v řádku.

1.5.3 Řídící síť T-spline plochy (T-mesh)

Teď, když už máme definovaný pojem PB-spline plochy, lze říci, že T-spline plocha je ve skutečnosti PB-spline plochou, jejíž řídící síť splňuje určitá pravidla. Takovému řídící síti budeme říkat řídící síť T-spline plochy neboli T-mesh.



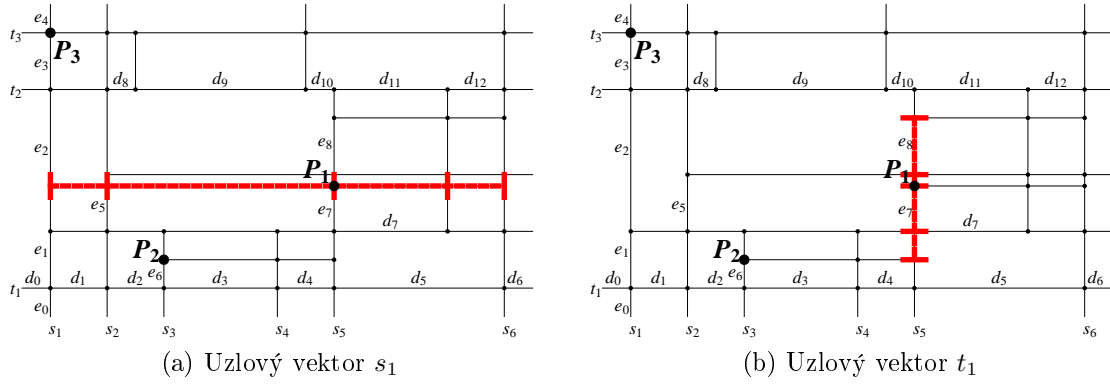
Obrázek 1.5.4: T-mesh, zdroj [5]

T-mesh je definován množinou hran a řídicích bodů. Každá hrana spojuje dva řídicí body, a je jí přidělena hodnota uzlového intervalu. Množina všech hran vytváří síť, jejíž stěny se skládají ze čtyř stran a zároveň každá strana stěny obsahuje více hran, klidně i spojených T-spojem. Na obrázku 1.5.4 lze vidět možné zobrazení T-meshe v parametrickém prostoru (s, t) . Hodnoty s_i označují s souřadnice, hodnoty t_i označují t souřadnice, a d_i, e_i představují uzlové intervaly. Z obrázku je dále vidět, že platí $s_4 = s_3 + d_3$, $t_5 = t_4 + e_4$, apod, a T-spoj je například v bodu \mathbf{P}_2 .

Každá hrana v řídicí síti T-spline plochy je úsečkou s konstantním s , resp. t , což nazýváme s -hrana, resp. t -hrana. Nyní lze lépe definovat T-spoj: T-spoj je vrchol společný buď pro jednu s -hranu a dvě t -hrany nebo naopak, pro jednu t -hranu a dvě s -hrany. Každá hrana je v řídicí síti označena uzlovým intervalem, který musí splňovat dvě následující pravidla:

- Součet uzlových intervalů na protilehlých stranách každé stěny musí být stejný.
 - Takže např. v obr. 1.5.4 platí $d_7 = d_2 + d_6$ a $e_6 + e_7 = e_8 + e_9$.
- Pokud T-spoj na jedné hraně stěny lze spojit s T-spojem na protější hraně bez porušení prvního pravidla, tak tato spojovací hrana musí být zahrnuta do řídicí sítě. Jinými slovy, spojovací hrana rozdělí zkoumanou stěnu na dvě menší.
 - Pokud by v obrázku 1.5.4 platilo $e_6 = e_9 = e_3$ a $e_7 = e_8$, vodorovná hrana dané stěny by musela být zahrnuta do řídicí sítě, a tedy t -hrana t_4 by pokračovala přes celý T-mesh, kde by vytvářela průsečík s s -hranou s_1 .

Pokud je T-mesh obdélníková síť bez T-spojů, T-spline plocha se stává B-spline plochou.



Obrázek 1.5.5: Nalezení uzlových vektorů

Lokální uzlový souřadnicový systém se dá vyvodit z uzlových intervalů v řídicí síti T-mesh. K jeho zavedení je třeba nejprve zvolit řídicí bod, který bude sloužit jako počátek parametrického prostoru $(s, t) = (0, 0)$. Např. v řídicí síti na obrázcích 1.5.5 (a), (b) označíme (s_1, t_1) za uzel počátku. Jakmile je počátek vybrán, můžeme začít přidělovat s uzlovou hodnotu ke každé vertikální hraně, a t uzlovou hodnotu ke každé horizontální hraně. Tyto hodnoty jsou v obrázcích 1.5.5 označeny jako s_i a t_i . Vzhledem k naší volbě počátečního uzlu, jsme dostali $s_1 = t_1 = 0$, $s_2 = d_1$, $s_3 = d_1 + d_2$, $t_2 = e_1 + e_2$, $t_3 = e_1 + e_2 + e_3$, atd. Obdobně každý řídicí bod má uzlové souřadnice. Např. $\mathbf{P}_1 = (s_5, t_1 + e_1 + e_5)$, $\mathbf{P}_2 = (s_3, t_1 + e_6)$ a $\mathbf{P}_3 = (s_1, t_3)$.

Teď, když už máme popsány všechny body a hrany v řídicí síti T-mesh, můžeme určit uzlové vektory pro každý bod. Právě touto možností vyvodit uzlové vektory z řídicí sítě, se T-spline plochy liší od B-spline ploch.

Uzlové souřadnice bodu \mathbf{P}_i jsou (s_{i2}, t_{i2}) , a uzlové vektory, které hledáme, jsou tvaru $s_i = [s_{i0}, s_{i1}, s_{i2}, s_{i3}, s_{i4}]$ a $t_i = [t_{i0}, t_{i1}, t_{i2}, t_{i3}, t_{i4}]$. Nechť je dán paprsek $\mathbf{R}(\alpha) = (s_{i2} + \alpha, t_{i2})$, $\alpha > 0$, tzn. vycházející z bodu $\mathbf{P}_i = (s_{i2}, t_{i2})$ směrem doprava. Uzly s_{i3} a s_{i4} jsou potom s souřadnice prvních dvou s -hran (vertikální hrany), které jsou paprskem protnuty, přičemž s -hrana, na které bod \mathbf{P}_i leží, se nezahrnuje jelikož ta už je ve vektoru obsažená jako jeho třetí složka, proto $\alpha > 0$. Pro uzly s_{i1} a s_{i0} vysíláme paprsek doleva, $\mathbf{R}(\alpha) = (s_{i2} - \alpha, t_{i2})$, $\alpha > 0$, a získáváme celý vektor $s_i = [s_{i0}, s_{i1}, s_{i2}, s_{i3}, s_{i4}]$. K získání vektoru $t_i = [t_{i0}, t_{i1}, t_{i2}, t_{i3}, t_{i4}]$ potřebujeme zjistit uzly t_{i0} , t_{i1} , t_{i3} a t_{i4} a proto vysíláme paprsek dolů $\mathbf{R}(\alpha) = (s_{i2}, t_{i2} - \alpha)$, $\alpha > 0$ a nahoru $\mathbf{R}(\alpha) = (s_{i2}, t_{i2} + \alpha)$, $\alpha > 0$ a sledujeme jeho průtnutí s horizontálními hranami. Vše je znázorněno na obrázku 1.5.5.

Tedy např. pro \mathbf{P}_1 je $s = [s_1, s_2, s_5, s_5 + d_7, s_6]$ a $t = [t_1 + e_6, t_1 + e_1, t_1 + e_1 + e_7, t_1 + e_1 + e_5, t_1 + e_1 + e_5 + e_8]$ a podobně pro \mathbf{P}_2 je $s = [s_1, s_2, s_3, s_4, s_5]$ a $t = [t_1 - e_0, t_1, t_1 + e_6, t_1 + e_1, t_1 + e_1 + e_5]$. \mathbf{P}_3 je hraniční řídicí bod. V závislosti na které hranici se nachází, se v uzlových vektorech hraničního řídicího bodu zduplikují složky. V případě z obrázku nezáleží na složkách $s_{3,0}$, resp. $t_{3,4}$ vektorů s_3 , resp. t_3 , takže vektory mají tvar: $s_3 = [s_1 - d_0, s_1 - d_0, s_1, s_2, s_2 + d_8]$ a $t_3 = [t_2 - e_2, t_2, t_3, t_3 + e_4, t_3 + e_4]$.

Jakmile máme určeny uzlové vektory pro všechny body, můžeme definovat T-spline

plochu pomocí rovnic pro PB-spline plochu (1.5.1).

$$\mathbf{P}(s, t) = \frac{\sum_{i=1}^n \mathbf{P}_i B_i(s, t)}{\sum_{i=1}^n B_i(s, t)}, \quad (s, t) \in \mathbf{D}$$

1.5.4 Standardní T-spline plochy

Pojem standardní T-spline plocha je definován jako T-spline plocha, kde součet básových funkcí $B_i(s, t)$ z rovnice (1.5.1) daných vztahem (1.5.2) je roven jedné pro $\forall (s, t) \in \mathbf{D}$. Tudiž i B-spline plocha je standardní T-spline plochou.

Důležitou vlastností je, že i po vložení uzlu do básových funkcí standardní T-spline plochy, tato plocha zůstává standardní T-spline plochou. Navíc T-spline plocha vytvořená spojením dvou B-spline ploch je také standardní T-spline plocha (postup popsán v [5]).

Standardní T-spline plocha lze rozložit do polynomických bikubických plátů. Ne-standardní T-spline plocha je také rozložitelná do bikubických plátů, ale už to jsou racionální pláty.

Kapitola 2

Aproximace neuspořádané množiny bodů v prostoru

2.1 Shrnutí současného stavu

Nejtěžším úkolem aproximace neuspořádané množiny bodů, je vytvoření řídicí sítě, která co nejlépe vystihuje vlastnosti hledané plochy, a zároveň jejíž počet řídicích bodů je rozumný. I v dnešní době ještě stále existují případy, kdy manuální zásah do řídicí sítě není na škodu, jelikož výsledná plocha vytvořená z upravené řídicí sítě, je mnohonásobně lepší aproximací zadaných bodů.

Jedna z metod, které kombinují jak automatické vytvoření, tak manuální zásahy do řídicí sítě je prezentována v článku [10]. Metoda je použitelná pro vytváření široké škály reprezentací, jako B-spline, NURBS nebo T-spline plochy, což nás zajímá nejvíce.

Metoda je složená z těchto kroků:

- Prvotní vytvoření sítě se provede nejmodernějšími metodami, jako např. PGP (Periodic Global Parameterization).

Poznámka: Periodic Global Parameterization, článek [11]

Metoda parametrizace trojúhelníkových ploch libovolné topologie představená v roce 2006. Vstupem jsou dvě navzájem kolmá po částech lineární vektorová pole definovaná nad vstupní sítí a výstupem dvě po částech lineární periodické funkce.

- Dále je možné manuálně zasáhnout do sítě, pokud si uživatel přeje vytvořit určité vlastnosti budoucí plochy. Je možné buď ovlivnit síť již při jejím vytváření, což se provede změnou určitých parametrů v metodě PGP, nebo si lze po vytvoření počáteční sítě pohrát s hranami, například některé hrany vyměnit za geodetiky, vložit nové hrany nebo naopak hrany vymazat.
- Převést síť na řídicí síť T-spline plochy, tím že vynutíme splnění pravidel řídicí sítě T-mesh. Platnost pravidel se pokusíme vynutit všude, kde jsou nějaká porušení. Pokud se vyskytnou případy, že vložení hran nikterak nepomůže, vložíme do sítě tzv. *mimořádné body* a okolí těchto bodů bude ve výsledné ploše nahrazeno T-NURCC plochou [5]. Rozlišujeme tři druhy mimořádných bodů:

- Do každé oblasti o N hranách, kde N je sudé, se vloží nový bod tak, že se spojí s každým druhým vrcholem oblasti. Toto vytvoří $N/2$ nových čtyřúhelníků a jeden mimořádný bod.
 - Do každé oblasti o N hranách, kde N je liché, se vložený bod spojí se středem každé její hrany, čímž se nejenže vytvoří jeden mimořádný bod, ale i N nových T-spojů a N nových čtyřúhelníků.
 - Obdobně pro oblasti o N hranách, kde N je liché a jeden její vrchol má valenci 3, se vložený bod spojí se středem každé hrany oblasti.
- Po tomto kroku je již síť platnou řídicí sítí pro T-spline plochu (kromě okolí mimořádných bodů) a je možné vypočítat T-spline plochu vztahem (1.5.1).

2.2 Adaptivní aproximace T-spline plochou

Úkolem metody, která je zde přestavena, je konstrukce hladkých povrchů objektů ze zadaných bodů v \mathbf{E}_3 , nejlépe geodat. Metoda je založena na postupné konstrukci T-spline řídicí sítě. Začíná se s hrubou aproximací bodů pomocí B-spline plochy s pravidelnou řídicí sítí, která se postupně zjemňuje vkládáním nových hran tam, kde je to třeba.

2.2.1 Příprava dat

2.2.1.1 Konvexní obal a oblouky

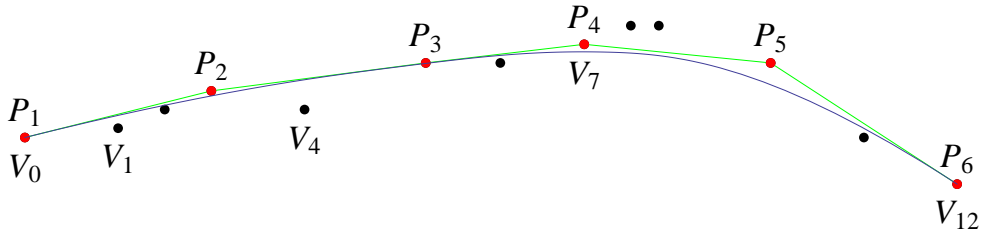
Metoda je navrhovaná pro vypočítání T-spline plochy ze zadaných geodat terénu, tudíž vstupem je list reálných souřadnic bodů v \mathbf{E}_3 . Nejdříve zjistíme, jak jsou body v prostoru umístěny a jaký zhruba má množina bodů tvar. Za tímto účelem body promítneme do roviny a zjistíme konvexní obal jejich bodů v rovině.

Poznámka: Konvexní obal konečné množiny M v \mathbf{E}_2 je nejmenší konvexní mnohoúhelník, který obsahuje všechny body z M , tzn.

$$\forall a, b \in M, t \in \langle 0, 1 \rangle : ta + (1 - t)b \in M.$$

Abychom získali prvotní aproximaci B-spline plochou potřebujeme čtyři okrajové křivky, které ohraničují množinu promítnutých bodů do roviny. Protože body nejsou uspořádané do žádné sítě, je třeba zjistit „rohové body“ množiny promítnutých bodů. Čtyři body, jež považujeme za rohy množiny bodů v rovině, jsou body konvexního obalu, jejichž dvě hrany spojující je se sousedními body z konvexního obalu, svírají úhel, který se co nejvíce blíží úhlu pravému.

Řídicí polygon B-spline křivky k -tého stupně musí obsahovat minimálně $k + 1$ bodů, tudíž aby B-spline křivky, které po nalezení rohů mezi nimi definujeme byly kubické, musí se jejich polygony skládat minimálně ze čtyř řídicích bodů, tzn. rohy mezi sebou mít minimálně dva body. Pro zjednodušení bude odteď situace vysvětlována na příkladu bodů jen mezi dvěma rohy, tzn. na jedné „straně“ množiny promítnutých bodů do roviny, protože postup na zbývajících třech stranách je stejný. Tyto strany budeme nazývat oblouky.

Obrázek 2.2.1: Body oblouku \mathbf{V}_i , řídicí body \mathbf{P}_i a počáteční B-spline

Teď, když víme, které body konvexního obalu jsou rohy, snadno zjistíme body ležící mezi nimi. Rohy a body mezi nimi označíme je $\mathbf{V}_0, \mathbf{V}_2, \dots, \mathbf{V}_n$ a vybereme z nich body \mathbf{P}_j , $j \in \langle 0, m \rangle$, $m \leq n$ tak, že mezi rohovými body jsou minimálně dva \mathbf{P}_j body (opět kvůli zaručení kubické B-spline křivky), a \mathbf{P}_j jsou rozděleny mezi rohy přibližně ekvidistantně. Je zřejmé, že kdyby nastala situace, že se mezi rohy množiny bodů promítnutých do roviny nacházejí jen dva body \mathbf{V}_i , tak platí $\mathbf{V}_i = \mathbf{P}_j$. Body \mathbf{P}_j jsou řídicími body hledané počáteční B-spline křivky (viz. obrázek 2.2.1), měněním jejich polohy zajistíme lepší aproximaci bodů oblouku touto křivkou.

Poznámka: Takto bylo původně počítáno s konstrukcí počáteční B-spline křivky.

Často se ale stalo, že počet bodů \mathbf{P}_j byl menší než čtyři, ať už z důvodu, že bodů oblouku \mathbf{V}_i bylo málo nebo byly špatně rozložené. Proto je počet \mathbf{P}_j natvrdo zvolen na základě počtu \mathbf{V}_i , a ekvidistanční rozložení sice zůstalo zachováno, ale nevybírání se jen z bodů \mathbf{V}_i , ale i z vnitřních bodů blízkých oblouku.

Na obrázku 2.2.2 je pro množinu bodů v rovině k vidění její konvexní obal, rohy konvexního obalu (červené body) a body \mathbf{P}_j (modré body).

Jak již bylo řečeno z bodů oblouku \mathbf{P}_j vytvoříme B-spline křivku $\mathbf{P}(t)$ pomocí vztahu (1.3.2). Tato křivka je jen počáteční a pomůže nám nalézt lepší B-spline křivku, která bude body obsažené v oblouku aproximovat lépe.

Tento úkol provedeme díky metodě nejmenších čtverců.

2.2.1.2 Metoda nejmenších čtverců

Poznámka: Metoda nejmenších čtverců

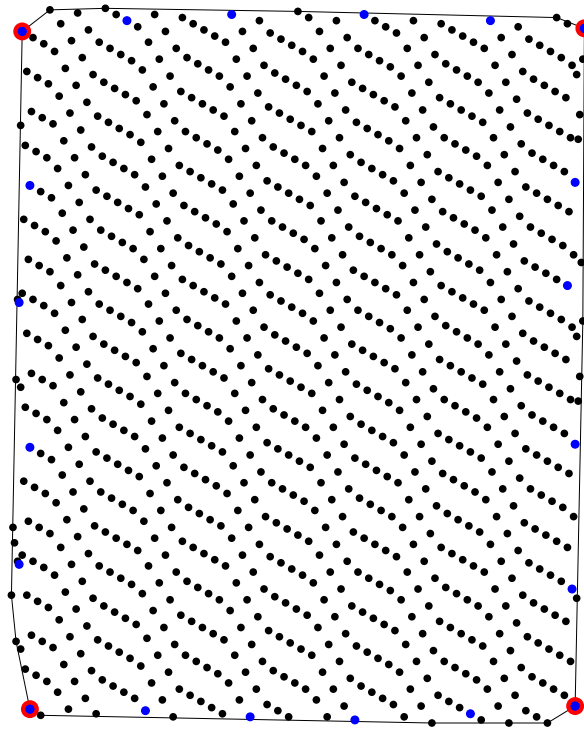
Aproximační metoda spočívající v tom, že hledáme takové parametry funkce f , pro které je součet čtverců odchylek vypočtených hodnot od hodnot naměřených minimální [14].

V našem případě hledáme parametry bodů B-spline křivky $\mathbf{P}(t)$, které mají nejmenší vzdálenost od daných bodů oblouku \mathbf{V}_i , tzn.

$$t_i = \min \|\mathbf{V}_i - \mathbf{P}(t)\|, \quad \forall t \in \langle 0, 1 \rangle, \quad \forall i \in \langle 0, n \rangle. \quad (2.2.1)$$

Nalezené parametry dosadíme do rovnice pro B-spline křivku, která se rovná bodu oblouku ve tvaru

$$\sum_{j=0}^m \mathbf{Q}_j N_i^3(t_i) = \mathbf{V}_i, \quad \forall i \in \langle 0, n \rangle, \quad (2.2.2)$$

Obrázek 2.2.2: Konvexní obal, rohy a body \mathbf{P}_j

kde $\mathbf{Q}_j = \mathbf{P}_j + (x_j, y_j)$ jsou posunuté řídicí body \mathbf{P}_j , pro které nová B-spline křivka lépe aproximuje zadané body oblouku \mathbf{V}_i . Vyjde nám tedy $(n + 1)$ rovnic pro $(m + 1)$ neznámých. Jelikož platí $m \leq n$, máme víc rovnic než neznámých, tj. přeúřčenou soustavu rovnic.

Poznámka: Necht' soustava lineárních rovnic $\mathbf{A}\mathbf{x} = \mathbf{b}$ je přeúřčenou soustavou, tzn. má více rovnic než neznámých. Pokud vektor pravých stran \mathbf{b} není lineární kombinací sloupců matice \mathbf{A} platí, že řešení \mathbf{x} neexistuje, jinak existuje právě jedno řešení, pro které platí

$$\mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}. \quad (2.2.3)$$

Rovnice podle vztahu (2.2.3) vyřešíme a hodnoty (x_j, y_j) přičteme k \mathbf{P}_j . Z těchto nových bodů \mathbf{Q}_j opět vypočteme B-spline křivku a provedeme metodu nejmenších čtverců a hledání nových bodů $\mathbf{R}_j = \mathbf{Q}_j + (x_j, y_j)$ ještě jednou. Celý postup je znázorněn v algoritmu 1.

Algoritmus 1: Metoda nejmenších čtverců pro B-spline křivku

vstup: body oblouku \mathbf{V}_i a body \mathbf{P}_j

výstup: posunuté body \mathbf{P}_j , B-spline křivka

begin

 vypočítat B-spline křivku $\mathbf{P}(t)$

 For (2x)

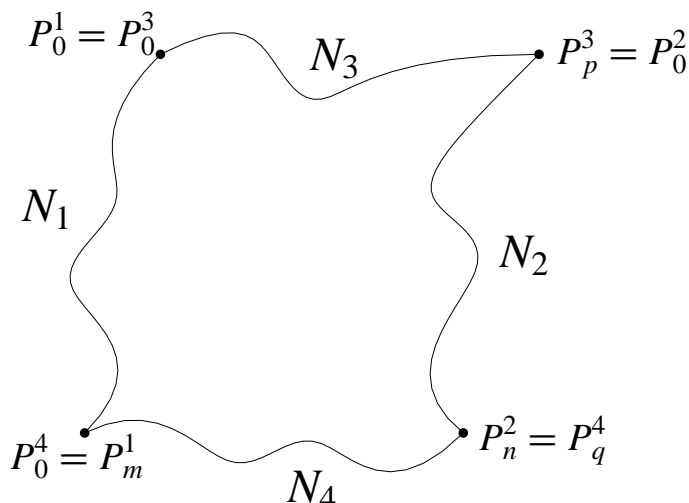
 pro každý bod oblouku \mathbf{V}_i najít parametr bodu na křivce, aby platilo (2.2.1)

parametr dosadit do (2.2.2)
 vyřešit přeurčenou soustavu
 nalezený posun připočíst k bodům \mathbf{P}_j
 vypočítat B-spline křivku z \mathbf{P}_j
return ($\mathbf{P}_j, \mathbf{P}(t)$)

Výsledkem jsou dvakrát posunuté body a výslednou B-spline křivka. Jelikož máme čtyři oblouky konvexního obalu promítnutých bodů do roviny, máme i čtyři B-spline křivky, které co nejlépe aproximují body těchto oblouků.

2.2.1.3 Výpočet řídicí sítě

Na tyto křivky, respektive na jejich řídicí body, použijeme metodu NURBS4 z balíčku bakalářské práce Petra Anděla [6] a získáme síť bodů, kterou použijeme jako počáteční řídicí síť T-mesh.



Obrázek 2.2.3: Rovinná oblast určená čtyřmi okrajovými křivkami splňující podmínku kompatibility

Poznámka: Rovinná oblast určená čtyřmi okrajovými křivkami [6].

Vstupem jsou čtyři okrajové NURBS křivky $\mathbf{N}^1 := [\mathbf{P}^1, \mathbf{W}^1, \mathbf{U}^1]$, $\mathbf{N}^2 := [\mathbf{P}^2, \mathbf{W}^2, \mathbf{U}^2]$, $\mathbf{N}^3 := [\mathbf{P}^3, \mathbf{W}^3, \mathbf{V}^1]$, $\mathbf{N}^4 := [\mathbf{P}^4, \mathbf{W}^4, \mathbf{V}^2]$, jež mají řídicí polygony $\mathbf{P}^1 = (\mathbf{P}_0^1, \dots, \mathbf{P}_m^1)$, $\mathbf{P}^2 = (\mathbf{P}_0^2, \dots, \mathbf{P}_n^2)$, $\mathbf{P}^3 = (\mathbf{P}_0^3, \dots, \mathbf{P}_p^3)$, $\mathbf{P}^4 = (\mathbf{P}_0^4, \dots, \mathbf{P}_q^4)$, příslušné váhy \mathbf{W}^1 , \mathbf{W}^2 , \mathbf{W}^3 , \mathbf{W}^4 a vektory parametrizace \mathbf{U}^1 , \mathbf{U}^2 a \mathbf{V}^1 , \mathbf{V}^2 . Křivky, musí splňovat podmínku kompatibility, tj. platí rovnosti $\mathbf{P}_0^1 = \mathbf{P}_0^3$, $\mathbf{P}_0^4 = \mathbf{P}_m^1$, $\mathbf{P}_0^2 = \mathbf{P}_p^3$ a $\mathbf{P}_q^4 = \mathbf{P}_n^2$ (viz. obrázek 2.2.3), a totéž pro jejich váhy.

Metoda proběhne ve dvou fázích. Nejprve se sjednotí vektory parametrizace křivek, které jsou naproti sobě, tj. křivek \mathbf{N}^1 , \mathbf{N}^2 a \mathbf{N}^3 , \mathbf{N}^4 .

- Pokud po převodu na normalizovaný tvar $(0, \dots, 0, \dots, 1, \dots, 1)$ jsou vektory shodné, tzn. křivky jsou stejného stupně, nic se s vektory nedělá.

- Pokud se stupně křivek rovnají, ale vektory se buď liší v normalizovaných tvarech, či mají rozdílnou velikost, tak se vektory upraví pomocí vkládání uzlů, aby byly shodné.
- Jsou-li křivky rozdílného stupně, použije se algoritmus na jejich srovnání a případně se i upraví vektory vkládáním uzlů.

Poté se dopočte řídicí síť a matice vah.

- Nejprve se naleznou vnitřní body sítě pro okrajové křivky \mathbf{N}^1 , \mathbf{N}^2 a dostaneme matici bodů $\mathbf{A} = (\mathbf{A}_{ij})$.
- Následně se vnitřní body naleznou pro okrajové křivky \mathbf{N}^3 , \mathbf{N}^4 , z čehož dostaneme matici $\mathbf{B} = (\mathbf{B}_{ij})$.
- Vnitřní řídicí body, které hledáme jsou středy úseček daných body \mathbf{A}_{ij} a \mathbf{B}_{ij} . To samé platí pro váhy.

Nyní máme určenu řídicí síť pro čtyři okrajové křivky, která je tvaru

$$\begin{pmatrix} \hat{\mathbf{P}}_0^1 = \hat{\mathbf{P}}_0^3 & \cdots & \cdots & \cdots & \hat{\mathbf{P}}_m^3 = \hat{\mathbf{P}}_0^2 \\ \vdots & \mathbf{Q}_1^1 & \cdots & \mathbf{Q}_1^{m-1} & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \mathbf{Q}_{p-1}^1 & \cdots & \mathbf{Q}_{p-1}^{m-1} & \vdots \\ \hat{\mathbf{P}}_0^4 = \hat{\mathbf{P}}_p^1 & \cdots & \cdots & \cdots & \hat{\mathbf{P}}_p^2 = \hat{\mathbf{P}}_m^4 \end{pmatrix}.$$

Řídicí síť budeme odteď značit jako $\mathbf{M} = (\mathbf{M}_{ij})$, $i \in \langle 0, p \rangle$, $j \in \langle 0, m \rangle$.

2.2.1.4 Uzlové pravidlo

Jelikož zatím máme obdélníkovou síť, kde všechny vnitřní body mají valenci 4, plocha vytvořená z této sítě by byla B-spline plochou. Za účelem vytvoření T-spline plochy potřebujeme T-mesh a ten dostaneme zjemňováním řídicí sítě.

Pro to je třeba zjistit uzlové vektory pro každý řídicí bod. Hledání uzlových vektorů z článku [5], které je popsáno v odstavci 1.5.3, je prováděno paprskem v parametrickém prostoru (s, t) , který je vyslán z bodu, pro nějž uzlový vektor hledáme, do čtyř stran: doleva, doprava, nahoru a dolů. Uzly pro uzlový vektor v souřadnici s , resp. souřadnici t jsou potom první dvě s -hrany zleva a zprava, resp. t -hrany shora a zdola.

Navrhovaná metoda toto řeší přes tzv. *matici sousedů*. Tuto matici si vytvoříme z matice aktuální řídicí sítě. Matice sousedů má jednoduchou podobu

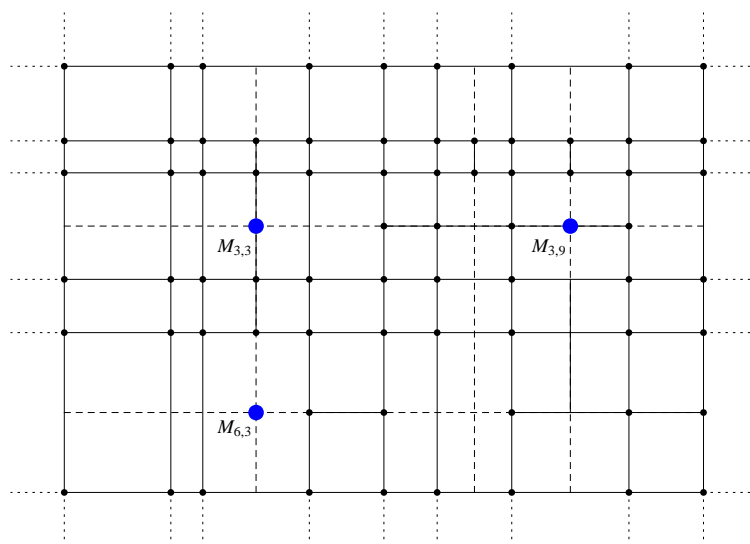
$$\mathbf{M}_S = \begin{pmatrix} (c, \mathbf{M}_{00}) & \cdots & (c, \mathbf{M}_{0m}) \\ \vdots & \ddots & \vdots \\ (c, \mathbf{M}_{p0}) & \cdots & (c, \mathbf{M}_{pm}) \end{pmatrix},$$

kde c je rovno:

- 1, pokud \mathbf{M}_{ij} je reálný bod, $\mathbf{M}_{ij} = (s_{ij}, t_{ij})$.
- 2.1, pokud \mathbf{M}_{ij} je virtuální bod na horizontální hraně, $\mathbf{M}_{ij} = (0)$.

- 2.2, pokud \mathbf{M}_{ij} je virtuální bod na vertikální hraně, $\mathbf{M}_{ij} = (0)$.
- 3, pokud \mathbf{M}_{ij} je kompletně virtuální, $\mathbf{M}_{ij} = (0)$.

Při prvním vytvoření budou všechna c samozřejmě rovná jedné, protože všechny body jsou reálné, ale po několika zjemněních, může řídicí síť vypadat jako na obrázku 2.2.4. Konstanta c má například pro $\mathbf{M}_{3,3}$ hodnotu 2.2, pro $\mathbf{M}_{3,9}$ hodnotu 2.1, a pro $\mathbf{M}_{6,3}$ hodnotu 3, jelikož bod je ryze virtuální a nenachází se na žádné reálné hraně.



Obrázek 2.2.4: Zjemněná řídicí síť

Z matice sousedů už je jednoduché zjistit uzlové vektory pro každý reálný bod řídicí sítě. Jak už bylo uvedeno v kapitole 1.5.3, uzlové vektory mají tvar $s = [s_{i0}, s_{i1}, s_{i2}, s_{i3}, s_{i4}]$ a $t = [t_{i0}, t_{i1}, t_{i2}, t_{i3}, t_{i4}]$.

Pro nalezení uzlů s_{i0} a s_{i1} , je třeba se podívat na sousední body po levé straně bodu. Pokud je bod reálný, nebo je virtuální a nachází se na reálné vertikální hraně, uzlem je hodnota s -hrany, na které leží. Když hledáme uzly s_{i3} a s_{i4} , prozkoumáváme sousední body na opačné straně. Složky uzlového vektoru v parametru t hledáme nahoře a dole, a zajímají nás reálné sousední body nebo virtuální body na reálné horizontální t -hraně.

Tomuto procesu hledání uzlových vektorů budeme říkat *uzlové pravidlo*.

2.2.2 Zjemňování sítě

Řídicí body se nacházejí zatím jen v rovině, takže jim přiřadíme průměrnou výšku vstupních bodů \mathbf{V}_{ij} abychom získali body v \mathbf{E}_3 . Další možností jak řídicí body „pozvednout“ je, pro každý řídicí bod najít vstupní bod, který je mu nejbližší a pak řídicímu bodu přiřadit výšku tohoto bodu.

2.2.2.1 Metoda nejmenších čtverců

Tyto už prostorové body řídicí sítě aproximujeme B-spline plochou $\mathbf{P}(u, v)$ dle vztahu (1.3.3). I zde dvakrát aplikujeme, tentokrát na všechny vstupní body, metodu nejmenších čtverců, kde hledáme parametry bodů vytvořené B-spline plochy, které mají nejmenší

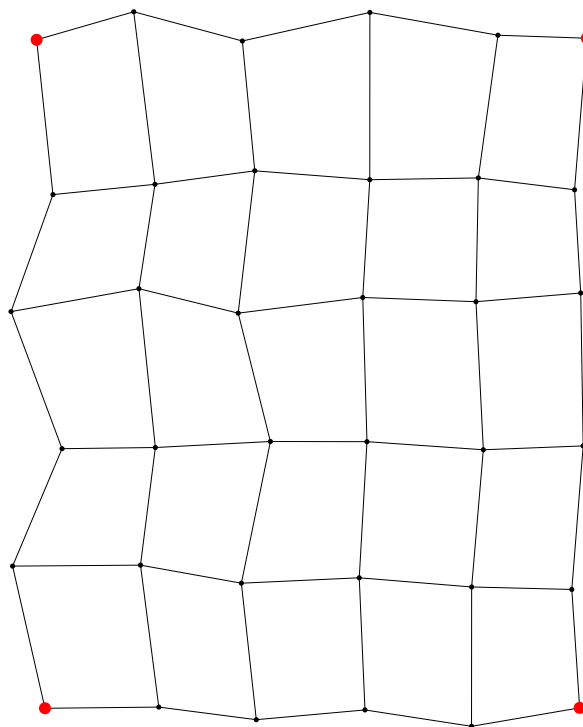
vzdálenost od vstupních bodů, tzn.

$$(u_{ij}, v_{ij}) = \min \|\mathbf{V}_{ij} - \mathbf{P}(u, v)\|, \quad \forall u, v \in \langle 0, 1 \rangle, \quad \forall \mathbf{V}_{ij}$$

Nalezený posun řídicích bodů k nim přičteme, čímž zajistíme lepší aproximaci vstupních bodů \mathbf{V}_{ij} B-spline plochou $\mathbf{P}(u, v)$.

2.2.2.2 Kumulativní chyba

Řídicí síť nám rozděluje B-spline plochu na $m \times p$ oblastí. Ve zjednodušeném případě pro 6×6 oblastí, lze samotnou řídicí síť promítnutou do roviny vidět na obrázku 2.2.5 a plochu s řídicí sítí na obrázku 2.2.6.

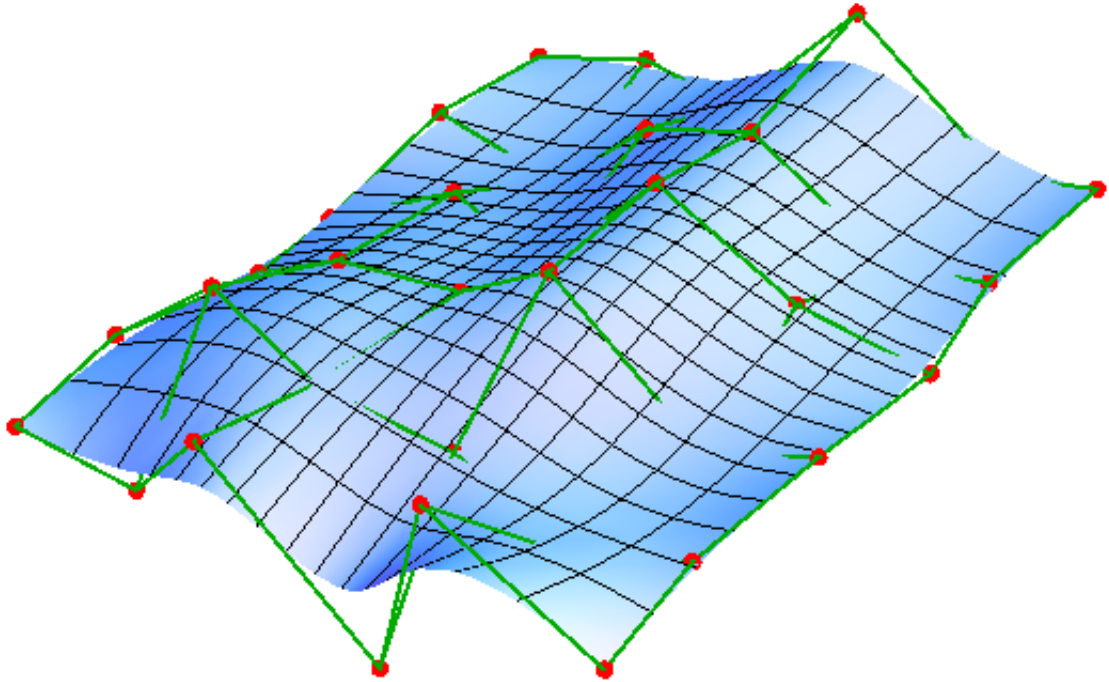


Obrázek 2.2.5: Oblasti B-spline plochy v rovině

Jádro navrhované metody leží ve zjemňování řídicí sítě tam, kde je chyba aproximace největší, takže si v každé oblasti zjistíme kumulativní chybu vzdáleností vstupních bodů od plochy, které spadají do této oblasti. Do oblasti, kde je chyba největší, budeme vkládat novou hranu a to tak, že její krajové body se umístí do středů delších stran oblasti [7].

2.2.2.3 Porušení pravidel při zjemňování

Je potřeba si uvědomit, jak spolu báze funkce řídicích bodů a řídicí síť T-mesh blíže souvisejí. Pro každý bod řídicí sítě \mathbf{P}_i existuje báze funkce $B_i(s, t)$ (1.5.2), a uzlové vektory pro každou báze funkce jsou nalezeny v řídicí síti uzlovým pravidlem. V průběhu zjemňování ale toto spojení dočasně rozdělíme, tzn. dovolíme existenci báze funkce $B_i(s, t)$, která porušuje uzlové pravidlo, a existenci řídicích bodů, ke kterým



Obrázek 2.2.6: Řídící síť s B-spline plochou

není přiřazena žádná báze funkce. Což samozřejmě není ve výsledku žádoucí a proto se tato porušení musí v metodě napravit.

Rozlišujeme tři druhy porušení, která se během procesu zjemňování mohou vyskytnout:

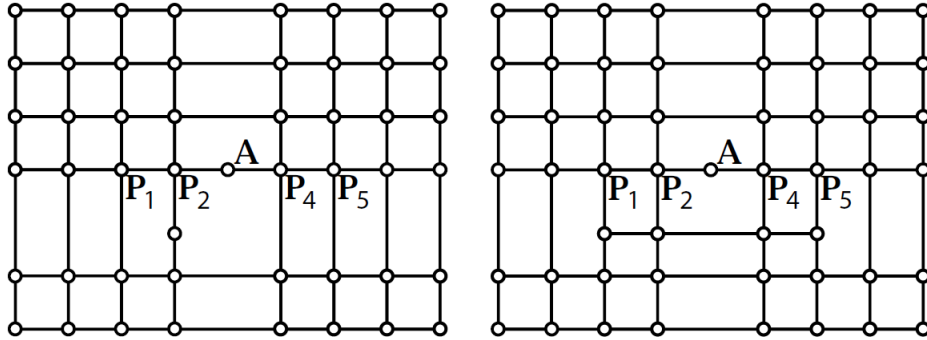
- Porušení 1 - Ve vektorech báze funkce chybí uzel, který by tam podle uzlového pravidla pro aktuální řídicí síť být měl.
- Porušení 2 - Ve vektorech báze funkce je uzel, který není uzlovým pravidlem předepsán.
- Porušení 3 - Existuje řídicí bod, k němuž není přiřazena žádná báze funkce.

Pokud se vyskytnou jakákoliv porušení, je třeba je jedno po druhém řešit, až už žádná neexistují. Vyřešením všech případů porušení 1 a 2 se automaticky vyřeší i případy 3.

Tento přístup a řešení z článku [7], které je popsáno v odstavci 2.2.3 se lehce liší od původního přístupu v článku [5]. Tam bylo ke dvěma pravidlům, které jsou popsány v odstavci 1.5.3, pro T-spline plochy přidáno ještě třetí pravidlo:

- Bod může být na t -hranu, resp. s -hranu přidán pouze v případě, že okolní čtyři body na dané t -hraně mají stejný uzlový vektor v parametru t , resp. okolní čtyři body na dané s -hraně mají stejný uzlový vektor v parametru s . Například na

obrázku 2.2.7 může být bod **A** přidán na horizontální hranu v řídicí síti pouze v případě, že $t_1 = t_2 = t_4 = t_5$.



Obrázek 2.2.7: Třetí pravidlo, zdroj [5]

Protože, ale uzlový vektor v t pro bod \mathbf{P}_2 je odlišný od uzlových vektorů v t pro body \mathbf{P}_1 , \mathbf{P}_4 a \mathbf{P}_5 , není možné ihned přidat nový bod do řídicí sítě. Jakmile se přidají nové řídicí body pod body \mathbf{P}_1 , \mathbf{P}_4 a \mathbf{P}_5 , jako je tomu na obrázku 2.2.7 vpravo, je již vložení bodu **A** povoleno.

Zároveň jsou ještě dány rovnice pro přepočítání pozic zúčastněných bodů. Například pro vložení bodu \mathbf{P}'_3 , jako je zobrazeno na obrázku 2.2.8 mají rovnice podobu:

$$\mathbf{P}'_1 = \mathbf{P}_1, \quad \mathbf{P}'_5 = \mathbf{P}_5$$

$$\mathbf{P}'_2 = \frac{d_4 \mathbf{P}_1 + (d_1 + d_2 + d_3) \mathbf{P}_2}{d_1 + d_2 + d_3 + d_4} \quad (2.2.4)$$

$$\mathbf{P}'_4 = \frac{(d_4 + d_5 + d_6) \mathbf{P}_4 + d_3 \mathbf{P}_5}{d_3 + d_4 + d_5 + d_6} \quad (2.2.5)$$

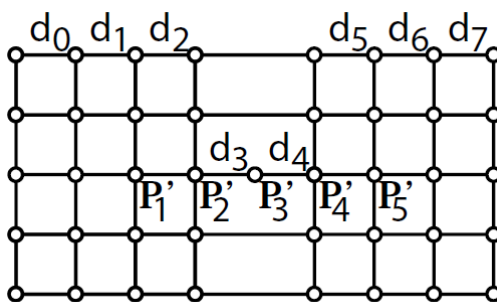
$$\mathbf{P}'_3 = \frac{(d_4 + d_5) \mathbf{P}_2 + (d_2 + d_3) \mathbf{P}_3}{d_2 + d_3 + d_4 + d_5} \quad (2.2.6)$$

V článku [5] je sice v rovnici (2.2.5) koeficient u bodu \mathbf{P}_4 uveden jako $(d_4 + d_4 + d_4)$, ale kvůli zachování konzistence s rovnicí (2.2.4) si jsem jista, že je tam chyba a správně je to tak, jak je uvedeno zde.

2.2.3 Zjemnění bazové funkce

Zjemňovat T-spline plochu znamená vložit do řídicí sítě dva body, čímž definujeme novou hranu, nebo jeden bod pokud se napojujeme na již existující hranu. Jelikož jsou s řídicí sítí úzce spjaty bazové funkce a jejich uzlové vektory, vložení nového řídicího bodu musí být spojeno s vložením nových uzlů do bazových funkcí. Proces zjemňování je realizován v těchto krocích:

1. Vložení požadovaného bodu do řídicí sítě.



Obrázek 2.2.8: Třetí pravidlo, zdroj [5]

2. Pokud se některá báze funkce dopustila porušení 1 (viz. odstavec 2.2.2.3), je třeba do uzlových vektorů této báze funkce provést vložení nového uzlu.
3. Pokud se některá báze funkce dopustila porušení 2, vložení příslušného bodu do řídicí sítě se to napraví.
4. Opakuj kroky 2 a 3 dokud nejsou žádná další porušení.

Vložení nového uzlu do uzlových vektorů báze funkce je definováno následujícím způsobem. Necht' $s = [s_0, s_1, s_2, s_3, s_4]$ je uzlový vektor a \tilde{s} je uzlový vektor o m složkách, který s obsahuje. Potom je možné vyjádřit funkci $N[s_0, s_1, s_2, s_3, s_4](s)$, což je kubická B-spline báze funkce $N_0^3(s)$ spojená s uzlovým vektorem $s = [s_0, s_1, s_2, s_3, s_4]$, jako lineární kombinaci $(m - 4)$ B-spline báze funkcí definovanými nad podmnožinami o pěti složkách uzlového vektoru \tilde{s} .

Pro $m = 6$ vypadají rovnice zjemenění takto (pro $m > 6$ jsou rovnice nalezeny opakovanou aplikací těchto rovnic):

- Pokud $\tilde{s} = [s_0, k, s_1, s_2, s_3, s_4]$,
pak $N(s) = c_0 N[s_0, k, s_1, s_2, s_3](s) + d_0 N[k, s_1, s_2, s_3, s_4](s)$,
kde $c_0 = \frac{k-s_0}{s_3-s_0}$ a $d_0 = 1$.
- Pokud $\tilde{s} = [s_0, s_1, k, s_2, s_3, s_4]$,
pak $N(s) = c_1 N[s_0, s_1, k, s_2, s_3](s) + d_1 N[s_1, k, s_2, s_3, s_4](s)$,
kde $c_1 = \frac{k-s_0}{s_3-s_0}$ a $d_1 = \frac{s_4-k}{s_4-s_1}$.
- Pokud $\tilde{s} = [s_0, s_1, s_2, k, s_3, s_4]$,
pak $N(s) = c_2 N[s_0, s_1, s_2, k, s_3](s) + d_2 N[s_1, s_2, k, s_3, s_4](s)$,
kde $c_2 = \frac{k-s_0}{s_3-s_0}$ a $d_2 = \frac{s_4-k}{s_4-s_1}$.
- Pokud $\tilde{s} = [s_0, s_1, s_2, s_3, k, s_4]$,
pak $N(s) = c_3 N[s_0, s_1, s_2, s_3, k](s) + d_3 N[s_1, s_2, s_3, k, s_4](s)$,
kde $c_3 = 1$ a $d_3 = \frac{s_4-k}{s_4-s_1}$.
- Pokud $k \leq s_0$ nebo $k \geq s_4$, potom se $N(s)$ nemění.

Báze funkce $B_i(s, t)$ může projít metodou vložení bodu, jak v parametru s , tak v parametru t . V algoritmu 2 je vidět stručný popis postupu.

Algoritmus 2: Metoda zjemňování řídicí sítě

vstup: řídicí síť, vkládaný bod

výstup: řídicí síť

begin

vložení nového řídicího bodu do sítě

While (existuje porušení)

Kontrola

If (existuje porušení 1)

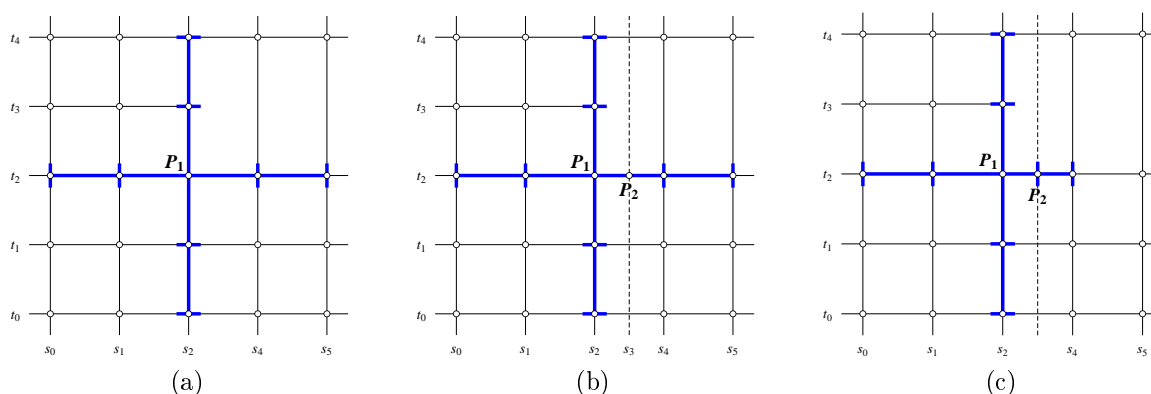
vložit nový uzel s využitím rovnic z odstavce 2.2.3

If (existuje porušení 2)

vložit nový bod do řídicí sítě

return (řídicí síť)

Příklad: Postup vložení bodu do řídicí sítě a následné zjemnění bazových funkcí budeme ilustrovat na příkladu. Na obrázku 2.2.9 (a) je zobrazena platná řídicí síť T-mesh bez jakýchkoliv porušení. Jakmile však vložíme do sítě bod P_2 (obr. 2.2.9 (b)), vyvstane hned několik porušení.



Obrázek 2.2.9: Ilustrační příklad vložení bodu do řídicí sítě

Jelikož má P_2 uzlové souřadnice (s_3, t_2) , bazové funkce pro čtyři nejbližší body na hraně t_2 nabývají porušení 1, jsou to bazové funkce umístěné v (s_2, t_2) , (s_1, t_2) , (s_4, t_2) a (s_5, t_2) . Jak bylo definováno na začátku tohoto odstavce, nápravou je vložení nového uzlu, zde s_3 , do každé z těchto funkcí.

Bázová funkce umístěná v (s_2, t_2) má před zjemněním tvar

$$N[s_0, s_1, s_2, s_4, s_5](s) N[t_0, t_1, t_2, t_3, t_4](t),$$

a po zjemnění, kde vkládáme uzel s_3 , se dle třetího bodu v odstavci 2.2.3 rozpůlí do dvou bazových funkcí,

$$c_2 N[s_0, s_1, s_2, s_3, s_4](s) N[t_0, t_1, t_2, t_3, t_4](t),$$

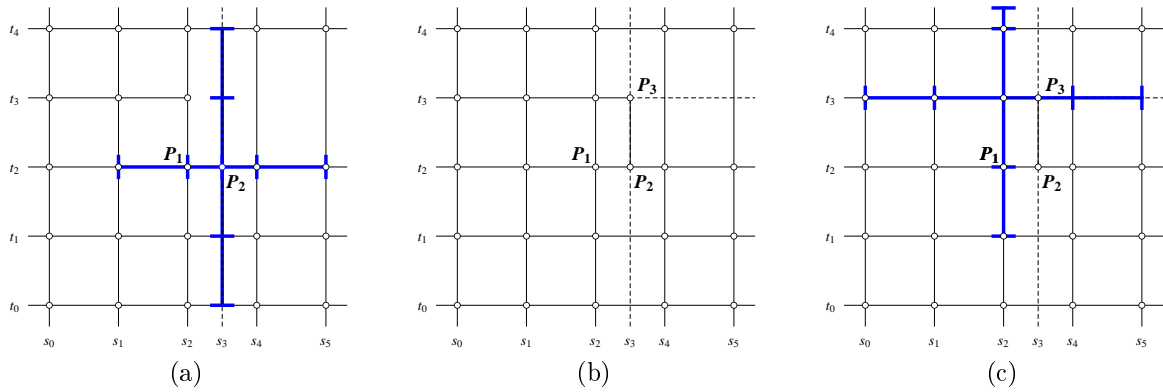
k vidění na obrázku 2.2.9 (c), a

$$d_2 N [s_1, s_2, s_3, s_4, s_5] (s) N [t_0, t_1, t_2, t_3, t_4] (t),$$

k vidění na obrázku 2.2.10 (a).

Bázová funkce $c_2 N [s_0, s_1, s_2, s_3, s_4] (s) N [t_0, t_1, t_2, t_3, t_4] (t)$ splňuje uzlové pravidlo, a stejně tak i zjemnění bázových funkcí umístěných v (s_1, t_2) , (s_4, t_2) a (s_5, t_2) .

Co však podléhá porušení 2, je uzlový vektor v parametru t bázové funkce $d_2 N [s_1, s_2, s_3, s_4, s_5] (s) N [t_0, t_1, t_2, t_3, t_4] (t)$, na obrázku 2.2.10 (a), protože vektor definovaný zjemněním je $[t_0, t_1, t_2, t_3, t_4]$, kdežto uzlové pravidlo uzel t_3 nenajde. Tento problém je možné napravit jedině vložením nového řídicího bodu do řídicí sítě. Bodem, který potřebujeme, je bod P_3 , zobrazen na obrázku 2.2.10 (b). Jeho vložením se odstraní porušení 2, ale zároveň vzniká porušení 1, protože jak je vidět na obrázku 2.2.10 (c), bázová funkce umístěná v (s_2, t_3) má uzlový vektor v parametru s , který neobsahuje uzel s_3 , jak je uzlovým pravidlem požadováno.



Obrázek 2.2.10: Pokračování ilustračního příkladu

Porušení 1 se opět vyřeší vložením chybějícího uzlu do uzlového vektoru. Tímto vložením se vyřeší všechna porušení 1 a 2 a tím zároveň i porušení 3.

Jelikož zjemňování bázových funkcí a vkládání nových řídicích bodů do řídicí sítě vyžaduje uzlové hodnoty, které už v řídicí síti T-mesh existují, algoritmus popsany tímto odstavcem, se provede v konečném počtu kroků. Nejhorší případ, který může nastat je, že by algoritmus protáhl částečnou řadu řídicích bodů, přes celý parametr.

Oproti algoritmu zjemňování ze staršího článku [5], který je popsán ve druhé polovině odstavce 2.2.2, má toto zjemňování dvě výhody: je zaručeno, že bude vždy fungovat a obvykle požaduje vložení o mnoho méně dodatečných řídicích bodů.

2.2.4 Zakončení

Jakmile máme vyřešena všechna porušení pro vloženou hranu, opět zjistíme posun bodů řídicí sítě pro lepší aproximaci metodou nejmenších čtverců.

Poté spočítáme kumulativní chybu vzdálenosti vstupních bodů od plochy v každé oblasti (používáme nové oblasti vytvořené vložením hrany a dodatečnými body), a do oblasti, kde je chyba největší opět vložíme hranu. Toto provádíme tak dlouho, dokud maximální vzdálenost vstupních bodů od plochy, neklesne pod zadanou přesnost.

Hledanou výslednou plochou, která co nejlépe aproximuje vstupní body, je tato T-spline plocha, již jsme vytvořili na začátku na základě B-spline plochy a jejíž řídicí síť byla střídavě zjemňována a poloha bodů upravována.

2.2.5 Výsledky

Se smutkem musím konstatovat, že implementovaná metoda nepůsobí tak, jak od ní bylo očekáváno.

2.2.5.1 Kumulativní chyba

Součet vzdáleností vstupních bodů od T-spline plochy se nezmenšuje, naopak někdy i zvětšuje.

Ukázka Výpis vzdáleností (chyb) patnácti průchodů zjemňováním:

Pruchod cislo 1: největší chyba = 6952.04, součet chyb = 96749.2

Pruchod cislo 2: největší chyba = 6150.67, součet chyb = 83967.7

Pruchod cislo 3: největší chyba = 6091.47, součet chyb = 85850.2

Pruchod cislo 4: největší chyba = 5698.26, součet chyb = 85356.

Pruchod cislo 5: největší chyba = 4908.49, součet chyb = 81340.4

Pruchod cislo 6: největší chyba = 4839.67, součet chyb = 93800.8

Pruchod cislo 7: největší chyba = 14265.5, součet chyb = 105820.

Pruchod cislo 8: největší chyba = 5858.13, součet chyb = 96066.6

Pruchod cislo 9: největší chyba = 4871.02, součet chyb = 93668.9

Pruchod cislo 10: největší chyba = 4088.11, součet chyb = 95795.8

Pruchod cislo 11: největší chyba = 3268.44, součet chyb = 86671.4

Pruchod cislo 12: největší chyba = 3855.65, součet chyb = 93393.1

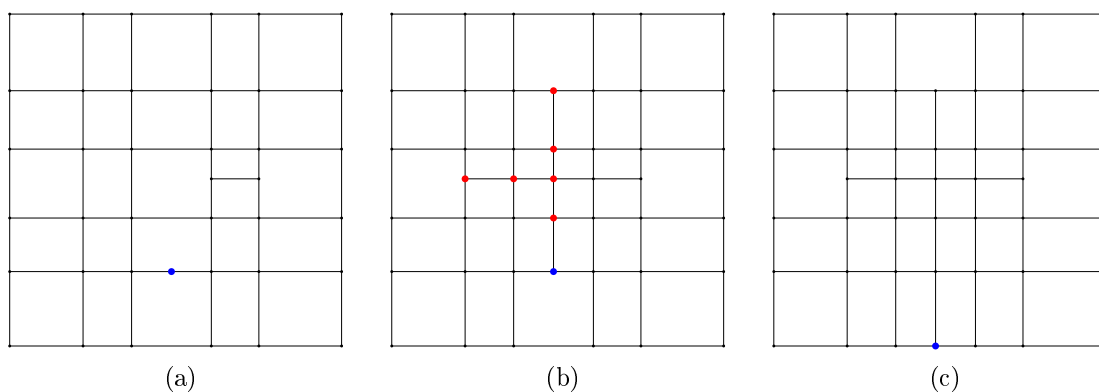
Pruchod cislo 13: největší chyba = 5948.51, součet chyb = 91031.2

Pruchod cislo 14: největší chyba = 10836.5, součet chyb = 120837.

Pruchod cislo 15: největší chyba = 5312.2, součet chyb = 117644.

Druhou vlastností, která nebyla splněna je nepropagace bodů po vložení bodu do řídicí sítě. Body se naopak během jednoho zjemnění vloží skoro po celé délce řídicí sítě. Například velice brzy po začátku metody se mi stala následující situace znázorněná na obrázcích 2.2.11.

Bylo třeba vložit vertikální hranu do poslední řady hran. Na obrázku 2.2.11 (a) je modře označen vložený první bod hrany. Kontrolou porušení spojení bazových funkcí a uzlových vektorů došlo několikrát ke zjemnění některých bazových funkcí, což mělo za následek postupné vložení dalších šesti bodů, které jsou na obrázku 2.2.11 (b) označeny červeně. Poté konečně mohlo dojít ke vložení druhého bodu hrany, který už žádné další body negeneroval, viz. obrázek 2.2.11 (c).

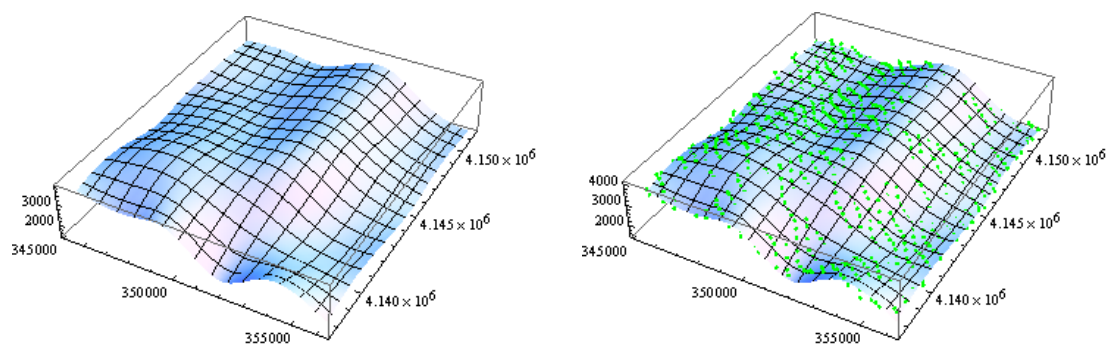


Obrázek 2.2.11: Propagace nových bodů

2.2.5.2 Tvar T-spline

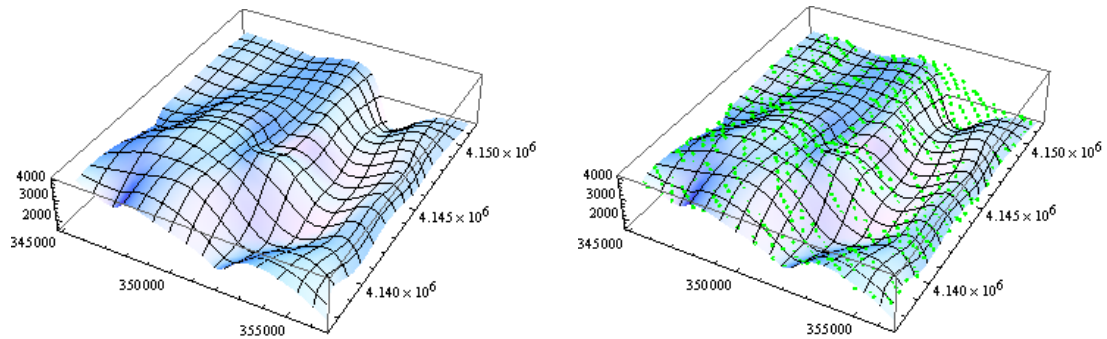
Není jasné zda-li špatným vkládáním, či propagací bodů, ale plocha se s větším počtem zjemnění čím dál tím víc odchyľuje od původního tvaru množiny vstupních bodů. Ukázka na obrázcích 2.2.12, 2.2.13, 2.2.14 a 2.2.15, k porovnání je k dispozici plocha a plocha se vstupními body.

- Počáteční T-spline plocha. Plocha je totožná s B-spline plochou. Aproximace není nejlepší, ale dobře vystihuje hrubý tvar množiny bodů.



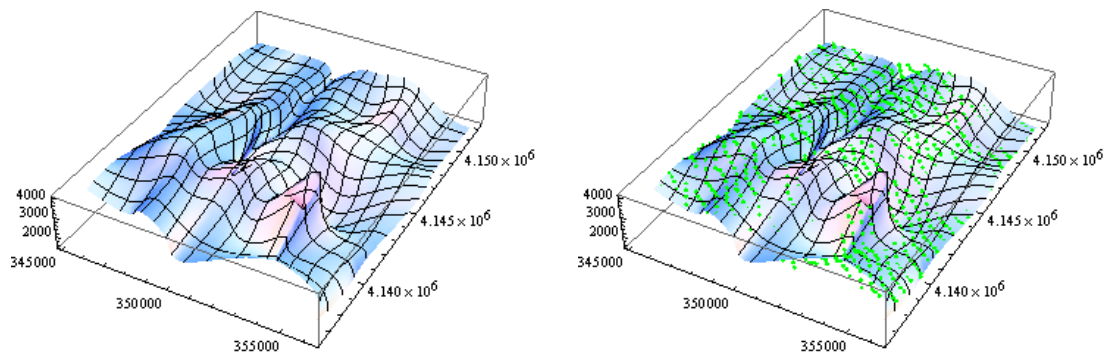
Obrázek 2.2.12: Počáteční T-spline plocha a plocha s body

- T-spline plocha po pěti zjemněních. Plocha začíná mít jasnější rysy, ale objevují se zvláštní odtažení plochy od kraje.



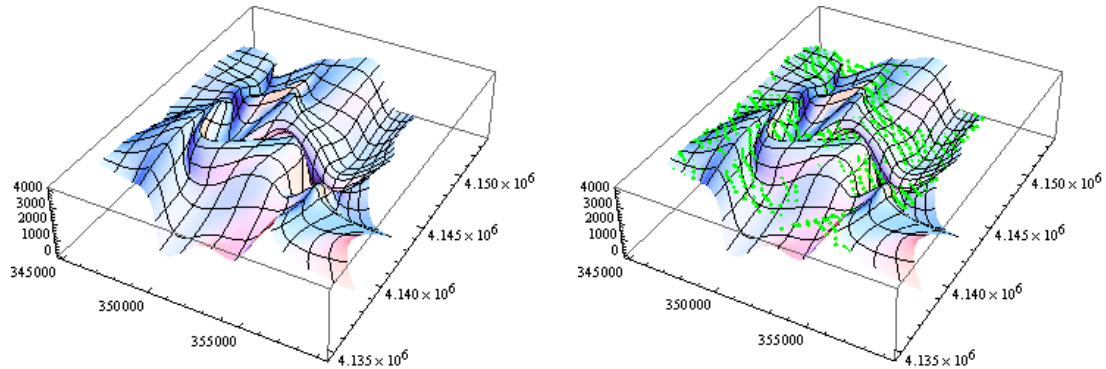
Obrázek 2.2.13: T-spline plocha a plocha s body po pěti zjemněních

- T-spline plocha po čtrnácti zjemněních. Plocha obsahuje vyvýšeniny a prohlubně, které vstupní množina bodů neobsahuje.



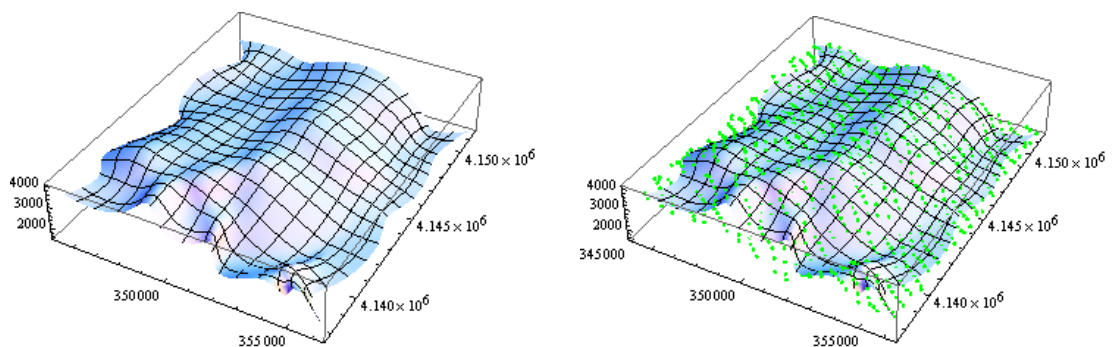
Obrázek 2.2.14: T-spline plocha a plocha s body po čtrnácti zjemněních

- T-spline plocha po osmnácti zjemněních. Plocha je občas divně smrštěná občas natáhnutá a rozhodně již nemá nic společného s tvarem množiny vstupních bodů.



Obrázek 2.2.15: T-spline plocha a plocha s body po osmnácti zjemněních

Vyzkoušela jsem i omezení posunu bodu v metodě nejmenších čtverců jen na posun v z -ové souřadnici. Podivným odskokům kraje plochy to ale stejně nezabránilo, viz obrázek 2.2.16, maximální chyba v oblastech se sice snižovala, celková chyba aproximace však zůstávala na podobné úrovni.



Obrázek 2.2.16: T-spline plocha a plocha s body po dvaceti zjemněních

Ukázka Výpis vzdáleností (chyb) průchodů zjemňováním:

- pruchod cislo 1: nejvetsi chyba = 6952.04, soucet chyb = 96749.2
- pruchod cislo 2: nejvetsi chyba = 5655.63, soucet chyb = 85287.7
- pruchod cislo 3: nejvetsi chyba = 5090.55, soucet chyb = 85477.1
- pruchod cislo 4: nejvetsi chyba = 5143.89, soucet chyb = 84487.4
- pruchod cislo 5: nejvetsi chyba = 4404.02, soucet chyb = 83448.1
- pruchod cislo 10: nejvetsi chyba = 3577.8, soucet chyb = 89404.7
- pruchod cislo 15: nejvetsi chyba = 3034.34, soucet chyb = 85655.8
- pruchod cislo 20: nejvetsi chyba = 2611.63, soucet chyb = 86653.4

2.2.5.3 Doba výpočtu

Vzhledem k tomu, že k jednomu zjemnění je potřeba třikrát vypočítat vzdálenosti vstupních bodů od plochy (jednou k určení kumulativní chyby a dvakrát v metodě nejmenších čtverců), trvá výpočet zjemnění poměrně dlouhou dobu.

Při počtu 1000 vstupních bodů se dvacet zjemnění provede asi za tři hodiny.

2.2.6 Balíček TsplinePlocha

Balíček TsplinePlocha byl vytvořen v programu Wolfram Mathematica 8.0. Načtení balíčku, což načte všechny funkce obsažené v něm, se provádí příkazem `<<TsplinePlocha``. Vlastní funkce se pak spouští příkazem TSP.

- Vstupem této funkce je pole bodů $\mathbf{V} = \{\{x_0, y_0, z_0\}, \dots, \{x_n, y_n, z_n\}\}$ a počet iterací k .
 - Čím více obdélníkový tvar bude mít množina bodů, tím lépe se naleznou rohové body a tím lepší bude aproximace.
 - Jelikož je nepravděpodobné, že by hodnota vzdálenosti vstupních bodů od plochy klesla pod určitou hranici, je jako zastavovací podmínka použit počet iterací (přechodů zjemněním) místo minimální hodnoty vzdálenosti.
- Výstupem je matice řídicích bodů \mathbf{P} a rovnice T-spline plochy $P(s, t) = \{P_x(s, t), P_y(s, t), P_z(s, t)\}$.

Volání funkce TSP má následující tvar: `TSP[{V, k}]`.

Z důvodu dlouhého čekání na výsledek, jsem přiložila jak balíček, který tiskne jen

T-spline plochy - balíček TsplinePlocha, tak i verzi s výpisy, aby bylo vidět, co právě program počítá - balíček TsplinePlochaVypisy.

Závěr

Cílem této diplomové práce bylo vymyslet a implementovat algoritmus pro aproximaci bodů v prostoru T-spline plochou.

Začátek je věnovaný základům standardně používaných reprezentací v geometrickém modelování, tudíž Bézierovým křivkám a plochám, obecným spline křivkám a plochám, B-spline křivkám a plochám, hierarchickým B-spline plochám, NURBS křivkám a plochám a nakonec tématu celé diplomové práce T-spline plochám.

Druhá část se věnuje aproximaci neuspořádaných bodů v prostoru. Nejdříve je popsán existující algoritmus: hrubý postup nalezení řídicí sítě neuspořádané množiny bodů a její následná konverze na řídicí síť T-mesh. Zbytek této části se věnuje popisu navrženému algoritmu pro aproximaci zadaných bodů v prostoru. Algoritmus je více méně složením dvou článků ([5] a [7]), s předefinováním některých postupů.

Prvním krokem je získat aproximaci bodů promítnutých do roviny, takže zadané body promítneme, zjistíme jejich konvexní obal a „rohy“ konvexního obalu. Jakmile máme rohy množiny bodů v prostoru, můžeme definovat B-spline křivky mezi nimi. Následně na křivky použijeme metodu nejmenších čtverců, pomocí které vypočítáme posuny řídicích bodů křivky, aby křivka ještě lépe aproximovala zadané body oblouku. Výsledné křivky použijeme jako vstup do vypůjčené metody z Bakalářské práce [6]. Výstupem metody je řídicí síť.

Protože stále operujeme jen v rovinné oblasti, je třeba bodům této sítě přiřadit výšku a přesunout se tak do prostoru. Tyto prostorové body řídicí sítě aproximujeme B-spline plochou. Opět využijeme metodu nejmenších čtverců a vypočítáme posuny bodů řídicí sítě, aby B-spline plocha lépe aproximovala zadané body v prostoru. Řídicí síť rozděluje B-spline plochu na několik oblastí, v každé oblasti vypočítáme kumulativní chybu vzdálenosti zadaných bodů od B-spline plochy. V oblasti, kde je chyba největší provedeme zjemnění řídicí sítě, a to vložím nové hrany do sítě, čímž vytváříme T-mesh.

Jelikož jsou s řídicí sítí úzce spjaty báze funkce vytvořené T-spline plochy, nelze jen tak vložit hranu do řídicí sítě, je třeba zkontrolovat, zda-li nenastala porušení této spojitosti. Pokud ve vektorech báze funkce chybí uzel, který by tam dle řídicí sítě být měl, je třeba do uzlových vektorů této báze funkce provést vložení nového uzlu. Pokud ale je ve vektorech báze funkce uzel, který nelze v řídicí síti dohledat, vložím nového bodu do řídicí sítě se to napraví. Zmíněné uzlové vektory se dají vyčíst z řídicí sítě pomocí matice sousedů.

Toto vkládání nových hran a následné přepočítání sítě metodou nejmenších čtverců provádíme tak dlouho, dokud maximální vzdálenost vstupních bodů od plochy, neklesne pod zadanou přesnost.

Takto byl algoritmus navržen a implementován. Bohužel výsledné plochy neapro-

ximují vstupní body dobře a dokonce se s postupem času více a více zhoršují. Příčina může být ve špatném výběru kam vložit hranu, či přímo ve způsobu vložení.

Literatura

- [1] JEŽEK, František. *Geometrické a počítačové modelování*. Pomocný učební text. ZČU, 2009.
- [2] FARIN, Gerald - HOSCHEK, Josef - KIM, Myung-Soo. *Handbook of computer aided geometric design*. 1st edition Amsterdam: Elsevier, 2002. ISBN 0-444-51104-0.
- [3] PIEGL, Les - TILLER, Wayne. *The NURBS Book*. Second edition. Berlin, Heidelberg, New York Springer-Verlag, 1997. ISBN 3-540-61545-8.
- [4] SEDERBERG, Thomas W. - ZHENG, J. - SEWELL, D. - SABIN, M. *Non-uniform recursive subdivision surfaces*. Proceedings of SIGGRAPH 98 (July), pp. 387-394, 1998. ISBN 0-89791-999-8.
- [5] SEDERBERG, Thomas W. - ZHENG, Jianmin. - BAKENOV, Almaz - NASRI, Ahmand. *T-spline and T-NURCCs*. Journal ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2003, Vol. 22, No. 3, pp. 477-484, 2003.
- [6] ANĎĚL, Petr. *NURBS reprezentace rovinných oblastí*, Plzeň: Západočeská univerzita. Fakulta aplikovaných věd. Katedra matematiky, 2010. 47 s. Vedoucí diplomové práce Ing. Bohumír Bastl, Ph.D.
- [7] SEDERBERG, Thomas W. - CARDON, David L. - FINNIGAN, G. Thomas - NORTH, Nicholas S. - ZHENG, Jianmin - LYCHE, Tom. *T-spline Simplification and Local Refinement*. Journal ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2004, Vol. 23, No. 3, pp. 276-283, 2004.
- [8] SEDERBERG, Matthew T. - SEDERBERG, Thomas W. *T-Splines: A Technology for Marine Design with Minimal Control Points*.
URL: <<http://www.tsplines.com/m/T-SplinesChesapeakepaper.pdf>> [cit. 2015-07-10]
- [9] FORSEY, David R. - BARTELS, Richard H. *Hierarchical B-spline refinement*. Proceeding SIGGRAPH 88 Proceedings of the 15th annual conference on Computer graphics and interactive techniques, pp. 205-212 , 1988.
- [10] LI, Wan-Chiu - RAY, Nicolas - LÉVY, Bruno. *Automatic and Interactive Mesh to T-Spline Conversion*. Eurographics Symposium on Geometry and Processing, 2006.

- [11] RAY, Nicolas - LI, Wan-Chiu - LÉVY, Bruno - SHEFFER, Alla - ALLIEZ, Pierre. *Periodic Global Parameterization*. Journal ACM Transactions on Graphics (TOG), Volume 25 Issue 4, pp. 1460-1485, 2006.
- [12] *Wikipedia, The Free Encyclopedia*. URL: <<http://en.wikipedia.org/>> [cit. 2015-07-10]
- [13] Dr. SHENE, C.-K. *CS3621 Introduction to Computing with Geometry Notes*. URL: <<http://www.cs.mtu.edu/~shene/COURSES/cs3621/NOTES/>> [cit. 2015-07-10]
- [14] *Laboratorní cvičení z fyziky: Metoda nejmenších čtverců*. České vysoké učení technické. Fakulta elektrotechnická. URL: <<http://herodes.feld.cvut.cz/mereni/mnc/mnc.php>> [cit. 2015-07-10]