



FAKULTA APLIKOVANÝCH VĚD
ZÁPADOČESKÉ UNIVERZITY
V PLZNI

KATEDRA
MATEMATIKY

KMA/GKP GEOMETRIE KŘIVEK A PLOCH

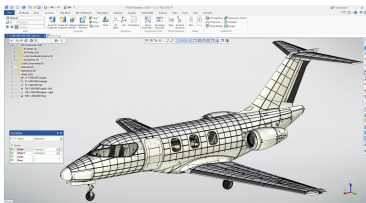
Geometrické modelování – křivky

Osnova

- 1 CAGD
- 2 Bézierovy křivky
- 3 B-spline křivky
- 4 Racionální techniky
- 5 Subdivizní křivky

Geometrické modelování – CAGD

- ▶ **CAGD (Computer Aided Geometric Design)** je oblast na pomezí matematiky, počítačové grafiky a inženýrského návrhu, která se zabývá konstrukcí a reprezentací křivek, ploch a objemů volných tvarů.
- ▶ Termín **CAGD** byl zaveden **R. Barnhillem** a **R. Riesenfeldem** při konferenci na University of Utah (1974).
- ▶ První časopis: **Computer Aided Geometric Design**, založený 1984.
- ▶ Hlavní stavební kameny CAGD:
 - ▶ **Křivky:** Bézierovy, B-spline, NURBS, subdivision křivky.
 - ▶ **Plochy:** Coonsovy, Bézierovy, B-spline a NURBS plochy.
 - ▶ **Algoritmy:** interpolace, aproximace, subdivize, optimalizace tvaru.
- ▶ CAGD tvoří teoretický základ moderních systémů **CAD/CAE/CAM** používaných v automobilovém, leteckém a strojírenském průmyslu.



Digitální řetězec průmyslového návrhu

- ▶ CAGD poskytuje matematický základ (křivky, plochy), na kterém stojí celý průmyslový návrhový proces.
- ▶ CAD (Computer Aided Design)
 - ▶ Návrh tvaru výrobku pomocí křivek, ploch a těles.
 - ▶ Výstupem je **přesná geometrická reprezentace**.
- ▶ CAE (Computer Aided Engineering)
 - ▶ Fyzikální simulace navržené geometrie (pevnost, deformace, proudění, teplo).
 - ▶ Výstupem je **ověření funkčnosti a bezpečnosti návrhu**.
- ▶ CAM (Computer Aided Manufacturing)
 - ▶ Převod geometrie na **dráhy nástroje** pro CNC stroje.
 - ▶ Výstupem je **program pro výrobu** (např. G-kód – seznam pohybů a příkazů pro stroj).



Stručná historie CAGD

- ▶ Počátky CAGD sahají do 40.–50. let 20. století, kdy v leteckém a automobilovém průmyslu vznikla potřeba **matematického popisu křivek a ploch volných tvarů**.
- ▶ Hlavní motivací bylo:
 - ▶ **Digitálně uložit** návrh plochy,
 - ▶ **Přenést geometrii na číslicově řízené obráběcí stroje (NC, CNC)**.
- ▶ Tyto praktické úkoly vedly k vývoji algoritmů, které bylo možné implementovat na počítačích – tím vznikl základ **Computer Aided Geometric Design**.
- ▶ Průkopníci ploch volných tvarů:
 - ▶ **R. Liming** a **J. Ferguson** (Boeing, USA) – první numerické metody pro návrh leteckých ploch.
 - ▶ **S. Coons** (MIT) – Coonsovy plochy, první matematický model plochy volného tvaru.
 - ▶ **M. Sabin** (British Aircraft Corporation) – matematické podmínky spojitosti mezi plochami a vývoj raných B-spline technik.
 - ▶ **P. de Casteljaou** (Citroën) – algoritmus pro konstrukci Bézierových křivek.
 - ▶ **P. Bézier** (Renault) – průmyslové využití křivek a ploch ve výrobě (UNISURF).
- ▶ Požadavky průmyslu **formovaly matematiku**: vznikly nástroje, které dnes tvoří základ všech moderních CAD systémů.
- ▶ CAGD se tak stalo **mostem mezi matematikou, grafikou a průmyslovým designem**.

Paul de Casteljau a Pierre Bézier

- ▶ **Paul de Casteljau** (Citroën, od 1959):
 - ▶ Vyvinul metody pro návrh křivek a ploch.
 - ▶ Použil Bernsteinovy polynomy a zavedl tzv. **de Casteljauův algoritmus**.
 - ▶ Přelom: používání řídicích polygonů – návrh tvaru křivky pomocí bodů, které na křivce neleží.
 - ▶ Citroën dlouho výsledky tajil; veřejné uznání přišlo až koncem 70. let.
- ▶ **Pierre Bézier** (Renault, 1960s):
 - ▶ Vyvíjel podobné metody nezávisle na Citroënu.
 - ▶ Renaultův systém UNISURF byl založen na Bézierových křivkách a plochách.
 - ▶ Ovlivnil vývoj CAD systémů, např. CATIA (Computer Aided Three-dimensional Interactive Application).



Křivky volných tvarů

- ▶ Historicky se při návrhu používaly kružnice a kuželosečky.
- ▶ Hladké **křivky volných tvarů**, řízené malým počtem řídicích bodů, jsou moderní nástroje, rozvíjené od 50. let 20. století.

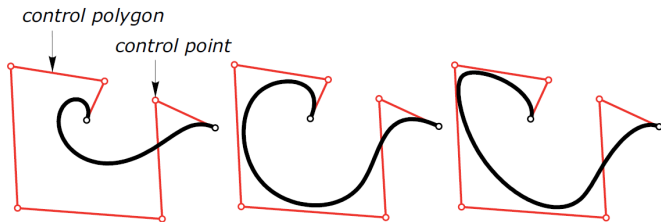
Hlavní typy křivek volných tvarů:

- ▶ **Bézierovy křivky** – základní konstrukce, de Casteljauův algoritmus.
- ▶ **B-spline křivky** – lokální kontrola tvaru, generování pomocí dělení (subdivision).
- ▶ **NURBS křivky** – jemné doladění pomocí vah; umí popsat i kuželosečky.

Bézier curve

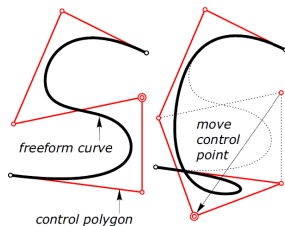
B-spline curve

NURBS curve



Jak navrhujeme křivky volných tvarů?

- ▶ Při ručním kreslení závisí kvalita křivky na dovednosti kreslíře a případně na mechanických pomůckách (např. pružné lišty – **spline**).
- ▶ V CAD softwaru tento přístup napodobujeme:
 - ▶ Bézierovy, B-spline a NURBS křivky jsou určeny několika řídicími body spojenými řídicím polygonem.
 - ▶ Hladká křivka je pak automaticky dopočítána geometrickým algoritmem.
- ▶ Výhody:
 - ▶ Rychlý vstup (pár řídicích bodů místo stovek ručně kreslených).
 - ▶ Snadná úprava tvaru přes posun řídicích bodů.



Dva přístupy k návrhu křivek

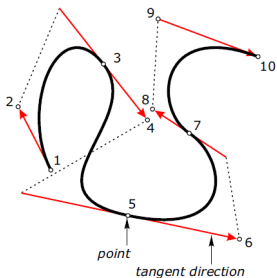
Interpolace

- ▶ Návrhář zadá několik bodů, kterými má křivka přesně procházet.
- ▶ Dále se mohou doplnit další geometrické podmínky, např.
 - ▶ tečné směry v interpolačních bodech,
 - ▶ křivosti v interpolačních bodech.

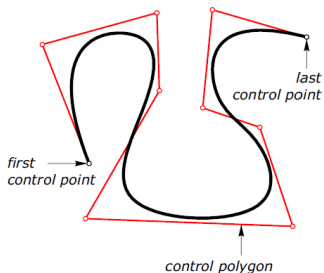
Aproximace

- ▶ Návrhář určí hrubý „lomený“ řídicí polygon.
- ▶ Algoritmus spočítá hladkou křivku, která se polygonu přibližuje, ale obecně jím neprochází.
- ▶ Křivka interpoluje první a poslední řídicí bod.

interpolating curve with tangent directions



approximating curve



Osnova

- 1 CAGD
- 2 Bézierovy křivky
- 3 B-spline křivky
- 4 Racionální techniky
- 5 Subdivizní křivky

Bézierovy techniky

▶ Co jsou Bézierovy techniky?

- ▶ Geometricky intuitivní metoda pro popis a manipulaci s polynomiálními křivkami a plochami.
- ▶ Základní nástroj v 3D modelování a CAGD.

▶ Proč jsou důležité?

- ▶ Umožňují kreativní návrh díky práci s **řídícími body**.
- ▶ Mají výbornou **numerickou stabilitu**, tedy odolnost vůči zaokrouhlovacím chybám.

▶ Co zahrnují?

- ▶ Křivky, obdélníkové plochy, trojúhelníkové plochy.
- ▶ Základ pro pokročilejší metody jako B-splíny a NURBS křivky a plochy.

Motivace: kvadratická Bézierova křivka

Příklad

- Uvažujme parametrickou křivku ve 2D:

$$\mathbf{x}(t) = [x(t), y(t)] = [t, 1 - t + t^2].$$

- Ekvivalentní zápis:

$$\mathbf{x}(t) = \mathbf{a}_0 + \mathbf{a}_1 t + \mathbf{a}_2 t^2,$$

kde

$$\mathbf{a}_0 = [0, 1], \quad \mathbf{a}_1 = [1, -1], \quad \mathbf{a}_2 = [0, 1].$$

jsou **koeficienty** křivky a $1, t, t^2$ tvoří kvadratickou **monomickou bázi**.

- Polynomiální křivku lze vyjádřit vzhledem k různým bázím, např.

$$\mathbf{x}(t) = \mathbf{b}_0(1 - t)^2 + 2\mathbf{b}_1 t(1 - t) + \mathbf{b}_2 t^2,$$

kde

$$\mathbf{b}_0 = [0, 1], \quad \mathbf{b}_1 = \left[\frac{1}{2}, \frac{1}{2} \right], \quad \mathbf{b}_2 = [1, 1].$$

jsou **Bézierovy řídicí body** řídicího polygonu a

$$B_0(t) = (1 - t)^2, \quad B_1(t) = 2t(1 - t), \quad B_2(t) = t^2$$

jsou kvadratické **Bernsteinovy polynomy** (bázové funkce).

Definice: Bézierova křivka

Definice (Bézierova křivka)

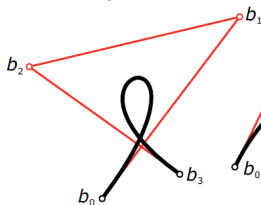
Bézierova křivka stupně n má tvar:

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

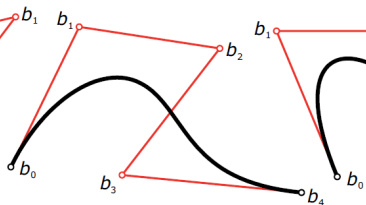
kde \mathbf{b}_i jsou řídicí body a $B_i^n(t)$ jsou Bernsteinovy polynomy stupně n :

$$B_i^n(t) = \binom{n}{i} (1-t)^{n-i} t^i, \quad \binom{n}{i} = \frac{n!}{(n-i)!i!}.$$

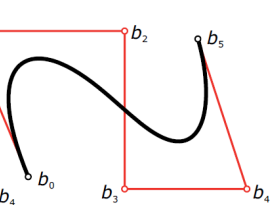
cubic Bézier curve
with a loop



Bézier curve of degree 4



Bézier curve of degree 5



Vlastnosti Bézierových křivek

- ▶ **Interpolace krajních bodů:** Bézierova křivka prochází prvním a posledním řídicím bodem.
- ▶ **Symetrie:** Křivka je invariantní vůči obrácení pořadí řídicích bodů.
- ▶ **Afinní invariance:** Křivka se transformuje stejně jako řídicí polygon.
- ▶ **Konvexní obal:** Křivka leží v konvexním obalu řídicího polygonu.
- ▶ **Útlum oscilací:** Křivka se nevlne více než řídicí polygon.
- ▶ **Lineární přesnost:** Pokud řídicí body leží rovnoměrně na přímce, křivka je úsečka.
- ▶ **Pseudolokální kontrola:** Posun bodu \mathbf{b}_i nejvíce ovlivní křivku v okolí $t = \frac{i}{n}$.

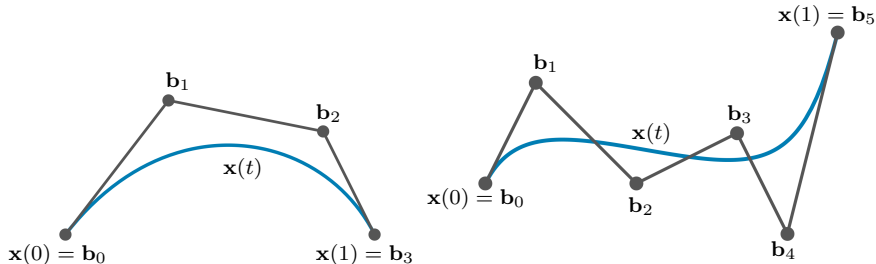
Vlastnosti Bézierových křivek (1)

Interpolace krajních bodů

Bézierova křivka stupně n vždy prochází prvními a posledními řídicími body:

$$\mathbf{x}(0) = \mathbf{b}_0, \quad \mathbf{x}(1) = \mathbf{b}_n.$$

- ▶ Křivka vždy prochází prvním a posledním kontrolním bodem.
- ▶ V krajních bodech je křivka tečná k první a poslední hraně kontrolního polygonu.
- ▶ Ostatní kontrolní body křivka neinterpoluje, pouze se řídí tvarem polygonu.

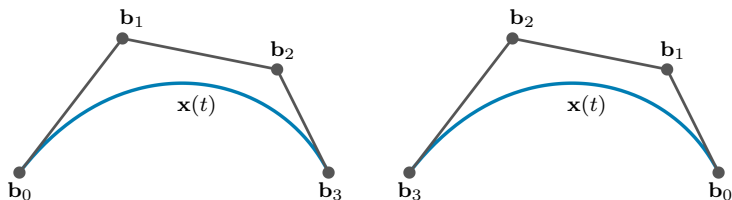


Vlastnosti Bézierových křivek (2)

Symetrie

Bézierova křivka definovaná řídicími body $\mathbf{b}_0, \mathbf{b}_1, \dots, \mathbf{b}_n$ je stejná jako křivka definovaná obráceným polygonem $\mathbf{b}_n, \mathbf{b}_{n-1}, \dots, \mathbf{b}_0$; mění se jen směr parametrizace.

- ▶ Křivka nezávisí na pořadí zápisu řídicího polygonu, pouze na průběhu parametru t .
- ▶ Pokud projdeme polygon pozpátku, dostaneme stejný tvar, ale s obráceným směrem parametrizace ($t \rightarrow 1 - t$).



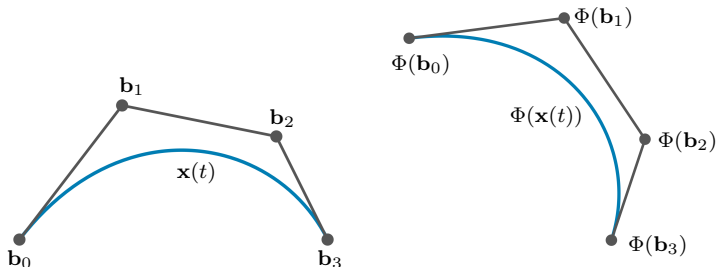
Vlastnosti Bézierových křivek (3)

Afinní invariance

Pokud na řídicí polygon $\mathbf{b}_0, \dots, \mathbf{b}_n$ aplikujeme afinní zobrazení f , pak výsledná Bézierova křivka je obrazem původní křivky pod tímto zobrazením:

$$f\left(\sum_{i=0}^n \mathbf{b}_i B_i^n(t)\right) = \sum_{i=0}^n f(\mathbf{b}_i) B_i^n(t).$$

- ▶ Bézierova křivka se transformuje „spolu“ se svým řídicím polygonem.
- ▶ To znamená, že můžeme transformovat buď křivku, nebo řídicí body – výsledek je stejný.

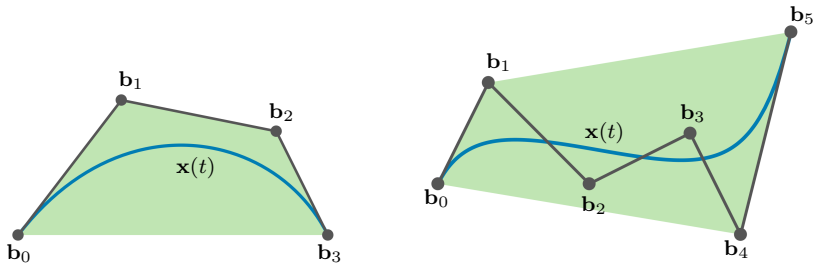


Vlastnosti Bézierových křivek (4)

Konvexní obal

Každý bod $x(t)$ na Bézierově křivce pro $t \in \langle 0, 1 \rangle$ leží uvnitř konvexního obalu řídicího polygonu b_0, \dots, b_n .

- ▶ Křivka nemůže „opustit“ oblast vymezenou řídicími body.
- ▶ Tato vlastnost poskytuje praktickou geometrickou kontrolu nad tvarem.

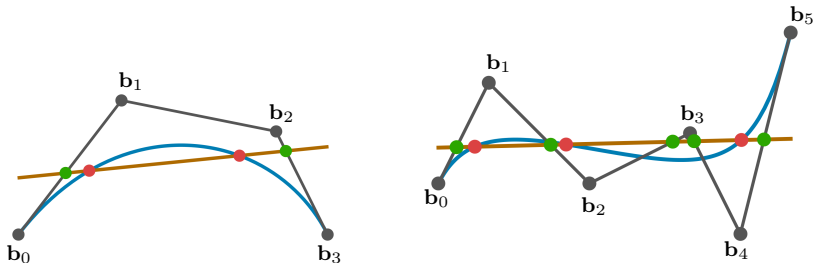


Vlastnosti Bézierových křivek (5)

Útlum oscilací

Pokud přímka protíná rovinný řídicí polygon m -krát, pak protne Bézierovu křivku nejvýše m -krát.

- ▶ Křivka „neosciluje“ víc než její řídicí polygon.
- ▶ Prakticky: žádné zbytečné vlňení mezi řídicími body.
- ▶ Tato vlastnost zajišťuje predikovatelné chování křivky při modelování.

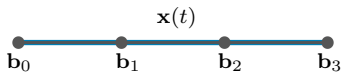


Vlastnosti Bézierových křivek (6)

Lineární přesnost

Pokud řídicí body $\mathbf{b}_0, \dots, \mathbf{b}_n$ leží rovnoměrně na úsečce mezi \mathbf{b}_0 a \mathbf{b}_n , pak Bézierova křivka stupně n je lineární interpolace mezi \mathbf{b}_0 a \mathbf{b}_n .

- ▶ Křivka se zjednoduší na úsečku mezi krajními body.
- ▶ Mezilehlé řídicí body neurčují nový tvar, pouze podporují přímkovost.

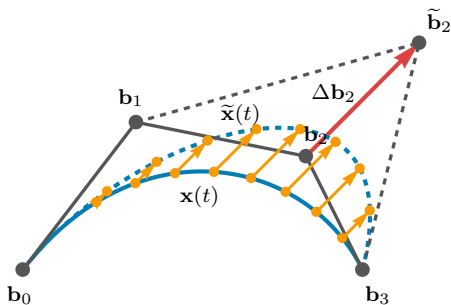


Vlastnosti Bézierových křivek (7)

Pseudolokální kontrola

Pokud posuneme i -tý řídicí bod \mathbf{b}_i , křivka se nejvíce změní v okolí parametru $t = \frac{i}{n}$.

- ▶ Úprava kontrolního bodu má největší vliv v jeho okolí, ale mírně ovlivní celou křivku.
- ▶ Všechny body na křivce se posunou rovnoběžně se změnou $\Delta \mathbf{b}_i$.



De Casteljauův algoritmus

- ▶ **Cíl:** Pro dané řídicí body $\mathbf{b}_0, \dots, \mathbf{b}_n$ a konkrétní volbu parametru $t \in \langle 0, 1 \rangle$ spočítat bod $\mathbf{x}(t)$ na Bézierově křivce.

- ▶ Vycházíme z řídicích bodů:

$$\mathbf{b}_i^0 = \mathbf{b}_i.$$

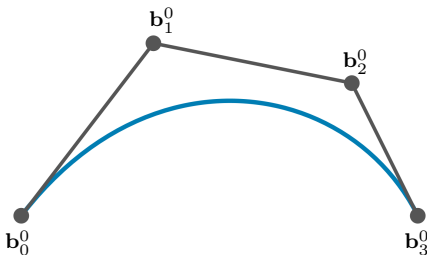
- ▶ V každém kroku $r = 1, 2, \dots, n$ spočítáme nové body:

$$\mathbf{b}_i^r(t) = (1-t)\mathbf{b}_i^{r-1} + t\mathbf{b}_{i+1}^{r-1}, \quad i = 0, \dots, n-r.$$

- ▶ Výsledný bod na křivce je:

$$\mathbf{x}(t) = \mathbf{b}_0^n(t).$$

- ▶ Každý krok provádí lineární interpolaci mezi body z předchozího kroku.



De Casteljauův algoritmus

- ▶ **Cíl:** Pro dané řídicí body $\mathbf{b}_0, \dots, \mathbf{b}_n$ a konkrétní volbu parametru $t \in \langle 0, 1 \rangle$ spočítat bod $\mathbf{x}(t)$ na Bézierově křivce.

- ▶ Vycházíme z řídicích bodů:

$$\mathbf{b}_i^0 = \mathbf{b}_i.$$

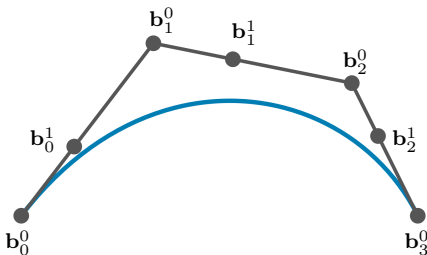
- ▶ V každém kroku $r = 1, 2, \dots, n$ spočítáme nové body:

$$\mathbf{b}_i^r(t) = (1-t)\mathbf{b}_i^{r-1} + t\mathbf{b}_{i+1}^{r-1}, \quad i = 0, \dots, n-r.$$

- ▶ Výsledný bod na křivce je:

$$\mathbf{x}(t) = \mathbf{b}_0^n(t).$$

- ▶ Každý krok provádí lineární interpolaci mezi body z předchozího kroku.



De Casteljauův algoritmus

- ▶ **Cíl:** Pro dané řídicí body $\mathbf{b}_0, \dots, \mathbf{b}_n$ a konkrétní volbu parametru $t \in \langle 0, 1 \rangle$ spočítat bod $\mathbf{x}(t)$ na Bézierově křivce.

- ▶ Vycházíme z řídicích bodů:

$$\mathbf{b}_i^0 = \mathbf{b}_i.$$

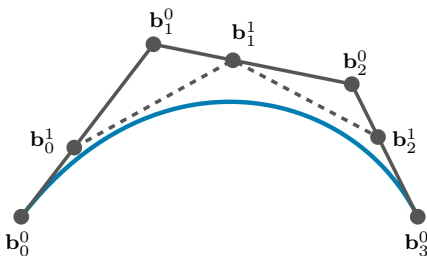
- ▶ V každém kroku $r = 1, 2, \dots, n$ spočítáme nové body:

$$\mathbf{b}_i^r(t) = (1-t)\mathbf{b}_i^{r-1} + t\mathbf{b}_{i+1}^{r-1}, \quad i = 0, \dots, n-r.$$

- ▶ Výsledný bod na křivce je:

$$\mathbf{x}(t) = \mathbf{b}_0^n(t).$$

- ▶ Každý krok provádí lineární interpolaci mezi body z předchozího kroku.



De Casteljauův algoritmus

- ▶ **Cíl:** Pro dané řídicí body $\mathbf{b}_0, \dots, \mathbf{b}_n$ a konkrétní volbu parametru $t \in \langle 0, 1 \rangle$ spočítat bod $\mathbf{x}(t)$ na Bézierově křivce.

- ▶ Vycházíme z řídicích bodů:

$$\mathbf{b}_i^0 = \mathbf{b}_i.$$

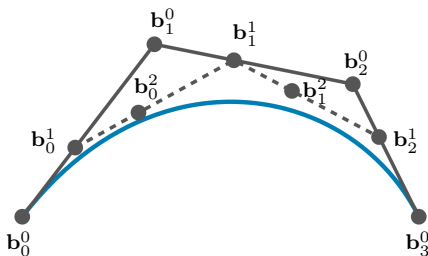
- ▶ V každém kroku $r = 1, 2, \dots, n$ spočítáme nové body:

$$\mathbf{b}_i^r(t) = (1-t)\mathbf{b}_i^{r-1} + t\mathbf{b}_{i+1}^{r-1}, \quad i = 0, \dots, n-r.$$

- ▶ Výsledný bod na křivce je:

$$\mathbf{x}(t) = \mathbf{b}_0^n(t).$$

- ▶ Každý krok provádí lineární interpolaci mezi body z předchozího kroku.



De Casteljauův algoritmus

- ▶ **Cíl:** Pro dané řídicí body $\mathbf{b}_0, \dots, \mathbf{b}_n$ a konkrétní volbu parametru $t \in \langle 0, 1 \rangle$ spočítat bod $\mathbf{x}(t)$ na Bézierově křivce.

- ▶ Vycházíme z řídicích bodů:

$$\mathbf{b}_i^0 = \mathbf{b}_i.$$

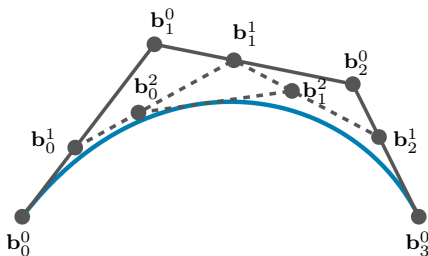
- ▶ V každém kroku $r = 1, 2, \dots, n$ spočítáme nové body:

$$\mathbf{b}_i^r(t) = (1-t)\mathbf{b}_i^{r-1} + t\mathbf{b}_{i+1}^{r-1}, \quad i = 0, \dots, n-r.$$

- ▶ Výsledný bod na křivce je:

$$\mathbf{x}(t) = \mathbf{b}_0^n(t).$$

- ▶ Každý krok provádí lineární interpolaci mezi body z předchozího kroku.



De Casteljauův algoritmus

- ▶ **Cíl:** Pro dané řídicí body $\mathbf{b}_0, \dots, \mathbf{b}_n$ a konkrétní volbu parametru $t \in \langle 0, 1 \rangle$ spočítat bod $\mathbf{x}(t)$ na Bézierově křivce.

- ▶ Vycházíme z řídicích bodů:

$$\mathbf{b}_i^0 = \mathbf{b}_i.$$

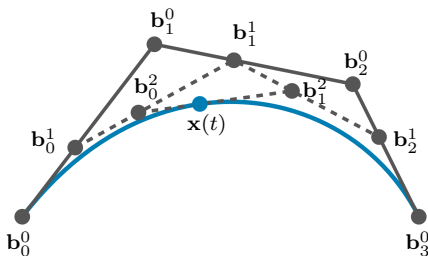
- ▶ V každém kroku $r = 1, 2, \dots, n$ spočítáme nové body:

$$\mathbf{b}_i^r(t) = (1-t)\mathbf{b}_i^{r-1} + t\mathbf{b}_{i+1}^{r-1}, \quad i = 0, \dots, n-r.$$

- ▶ Výsledný bod na křivce je:

$$\mathbf{x}(t) = \mathbf{b}_0^n(t).$$

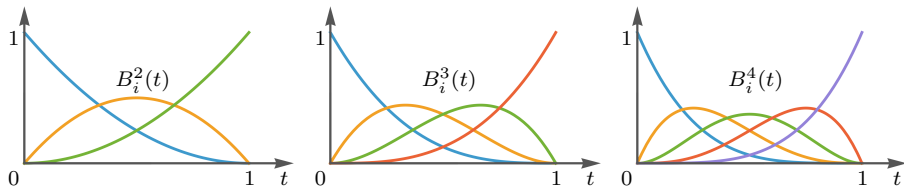
- ▶ Každý krok provádí lineární interpolaci mezi body z předchozího kroku.



Bernsteinovy polynomy

- ▶ Bernsteinovy polynomy stupně n tvoří bázové funkce prostoru všech polynomů stupně nejvýše n na intervalu $t \in \langle 0, 1 \rangle$ a mají tvar

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



Vlastnosti Bernsteinových polynomů

- ▶ **Rozklad jednotky:** Pro libovolné $t \in \langle 0, 1 \rangle$ platí

$$\sum_{i=0}^n B_i^n(t) = 1.$$

Tato vlastnost zajišťuje, že Bézierova křivka je barycentrická kombinace řídicích bodů.

- ▶ **Nezápornost:** Pro $t \in \langle 0, 1 \rangle$ je každý Bernsteinův polynom nezáporný:

$$B_i^n(t) \geq 0.$$

Spolu s rozkladem jednotky to vede ke konvexnímu obalu.

- ▶ **Symetrie:**

$$B_i^n(t) = B_{n-i}^n(1-t).$$

Tato vlastnost se odráží v symetrii Bézierových křivek.

- ▶ **Rekurze:**

$$B_i^n(t) = (1-t) B_i^{n-1}(t) + t B_{i-1}^{n-1}(t).$$

Umožňuje efektivní výpočet např. pomocí de Casteljauova algoritmu.

Vlastnosti Bernsteinových polynomů

▶ **Hodnoty na krajích:**

$$B_i^n(0) = \delta_{i,0}, \quad B_i^n(1) = \delta_{i,n},$$

kde

$$\delta_{i,n} = \begin{cases} 1, & \text{pro } i = n, \\ 0, & \text{pro } i \neq n. \end{cases}$$

Odpovídá interpolaci krajních bodů Bézierových křivek.

▶ **Lineární přesnost:**

$$\sum_{i=0}^n \frac{i}{n} B_i^n(t) = t.$$

Bézierovy křivky přesně interpolují úsečku mezi krajními body.

▶ **Jediné maximum:** Každý $B_i^n(t)$ má maximum v

$$t = \frac{i}{n},$$

což souvisí s pseudolokální kontrolou tvaru.

Konverze mezi monomickou a Bernsteinovou bází

- ▶ Nechť $p(t)$ je polynom stupně n ve tvaru:

$$p(t) = a_0 + a_1t + a_2t^2 + \cdots + a_nt^n = \sum_{i=0}^n a_it^i.$$

- ▶ Ten lze také vyjádřit pomocí **Bernsteinovy báze**:

$$p(t) = \sum_{i=0}^n b_i B_i^n(t), \quad B_i^n(t) = \binom{n}{i} (1-t)^{n-i} t^i.$$

- ▶ Existuje **matice přechodu** \mathbf{T}_n , která převádí koeficienty:

$$\begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{pmatrix} = \mathbf{T}_n \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix}.$$

- ▶ Pro $n = 2, 3$:

$$\underbrace{\begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix}}_{n=2} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 2 & 0 \\ 1 & -2 & 1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix}, \quad \underbrace{\begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix}}_{n=3} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix}.$$

Fergusonova kubika (kubická Hermitova křivka)

- ▶ **James C. Ferguson**, působící v 60. letech ve společnosti Boeing (Seattle, USA), publikoval v roce 1964 koncept **Hermitovské kubické interpolace**, známý jako **Fergusonova kubika**.
- ▶ **Fergusonova kubika** je definována dvěma body $\mathbf{p}_0, \mathbf{p}_1$ a dvěma tečnými vektory $\mathbf{t}_0, \mathbf{t}_1$:

$$\mathbf{x}(t) = \mathbf{p}_0 H_0(t) + \mathbf{t}_0 H_1(t) + \mathbf{t}_1 H_2(t) + \mathbf{p}_1 H_3(t), \quad t \in \langle 0, 1 \rangle,$$

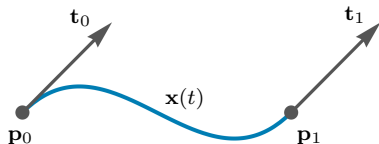
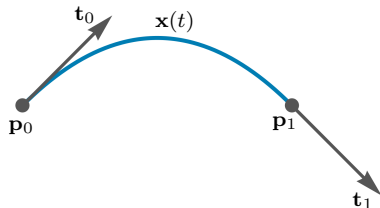
kde $H_i(t)$ jsou **Hermitovy** **bázové** funkce:

$$H_0(t) = B_0^3(t) + B_1^3(t) = 2t^3 - 3t^2 + 1,$$

$$H_1(t) = \frac{1}{3}B_1^3(t) = t^3 - 2t^2 + t,$$

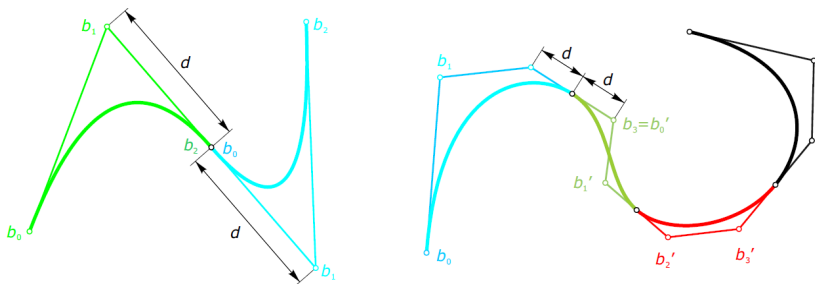
$$H_2(t) = \frac{1}{3}B_2^3(t) = -t^3 + t^2,$$

$$H_3(t) = B_2^3(t) + B_3^3(t) = -2t^3 + 3t^2.$$



Skládání Bézierových křivek

- ▶ Pro delší volné křivky lze spojit více nízkostupňových Bézierových křivek (stupně 2 nebo 3).
- ▶ V napojovacích bodech chceme zajistit:
 - ▶ stejnou tečnu (kvadratické křivky),
 - ▶ případně i stejnou křivost (kubické a vyšší křivky).
- ▶ U složených Bézierových křivek lze zajistit tečnou spojitost, ale dosažení stejné křivosti v napojovacích bodech je už komplikované.
- ▶ Omezení Bézierových a složených Bézierových křivek vedlo k zavedení **B-spline křivek**.

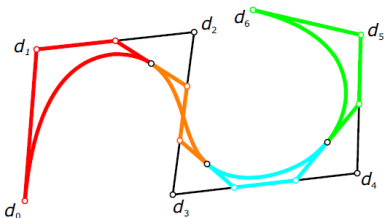
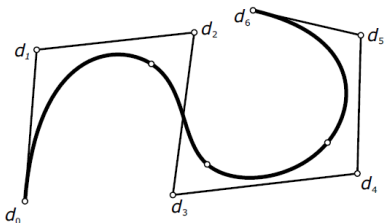


Osnova

- 1 CAGD
- 2 Bézierovy křivky
- 3 B-spline křivky
- 4 Racionální techniky
- 5 Subdivizní křivky

B-spline křivky – pojem a historie

- ▶ **B-spline křivky** = křivky složené z Bézierových segmentů stejného stupně.
 - ▶ Segmenty jsou spojeny automaticky s maximální možnou spojitostí (stejná tečna, stejná křivost, pokud je to možné).
- ▶ **Výhody:**
 - ▶ Snadnější manipulace a editace než u složených Bézierových křivek,
 - ▶ Lépe sledují tvar kontrolního polygonu,
 - ▶ Sdílí užitečné vlastnosti Bézierových křivek (afinní invariance, konvexní obal, atd.).
- ▶ **Historie:**
 - ▶ „Spline“ = pružná lišta používaná lodními staviteli pro kreslení hladkých křivek,
 - ▶ **B-spline** = **basis spline**, pojem zaveden I. J. Schoenbergem,
 - ▶ Bézierova křivka je speciálním případem B-spline křivky.



B-spline bázové funkce

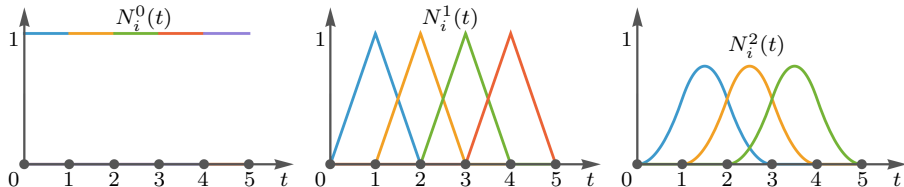
Definice (B-spline báze)

Nechť $U = \{t_0, \dots, t_m\}$ je **uzlový vektor**, tj. neklesající posloupnost reálných čísel. i -tá B-spline funkce stupně p je definována rekurzí:

$$N_{i,0}(u) = \begin{cases} 1, & t_i \leq u < t_{i+1}, \\ 0, & \text{jinak,} \end{cases}$$

$$N_{i,p}(u) = \frac{u - t_i}{t_{i+p} - t_i} N_{i,p-1}(u) + \frac{t_{i+p+1} - u}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(u).$$

- ▶ **Nezápornost a kompaktní nosič** (funkce je nenulová pouze) na intervalu $\langle t_i, t_{i+p+1} \rangle$.
- ▶ Na každém intervalu $\langle t_j, t_{j+1} \rangle$ je nenulových nejvýše $p + 1$ funkcí.
- ▶ Platí $\sum_i N_{i,p}(u) = 1$.



bázové funkce $N_i^p(t)$ pro $p = 0, 1, 2$ a uniformní uzlový vektor

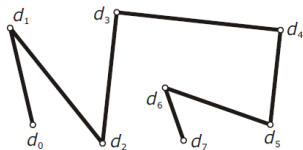
Definice B-spline křivky

Definice (B-spline křivka)

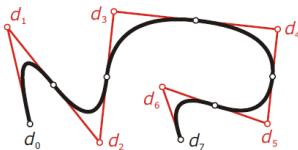
Nechť máme $n+1$ kontrolních bodů $\mathbf{d}_0, \dots, \mathbf{d}_n$, uzlový vektor $U = \{t_0, \dots, t_m\}$ a stupeň p , kde $m = n + p + 1$. **B-spline křivka** stupně p je

$$\mathbf{c}(u) = \sum_{i=0}^n N_{i,p}(u) \mathbf{d}_i, \quad u \in \langle t_p, t_{m-p} \rangle.$$

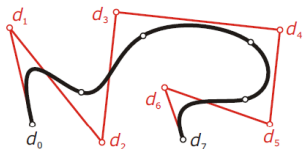
$n=1$



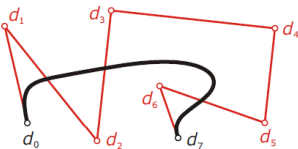
$n=2$



$n=3$



$n=7$



Otevřená B-spline křivka

- ▶ Pokud jsou všechny uzly **různé**, tj.

$$t_0 < t_1 < \dots < t_m \quad (\text{open knot vector}),$$

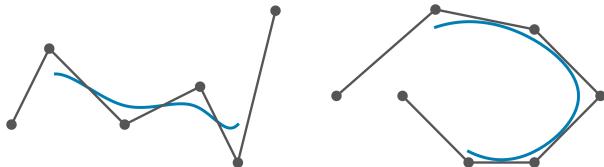
dostáváme tzv. **otevřenou B-spline křivku**.

- ▶ Otevřená B-spline křivka **neprochází** prvním ani posledním kontrolním bodem.
- ▶ Používá se např. **uniformní uzlový vektor**, kde jsou rozdíly mezi sousedními uzly konstantní:

$$t_{i+1} - t_i = h = \text{konst.}$$

Uzly tak tvoří aritmetickou posloupnost, např.

$$U = \{0, 1, 2, 3, 4, 5, 6\}.$$

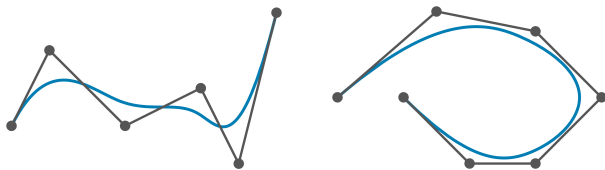


Clamped B-spline křivka

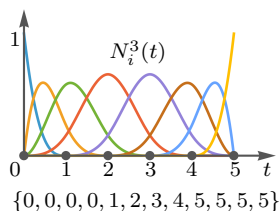
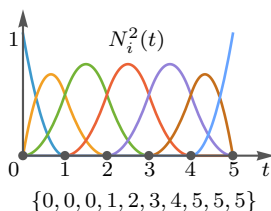
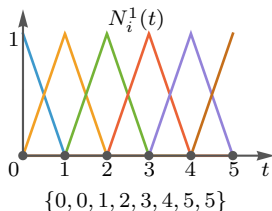
- Pokud mají koncové uzly násobnosti $p + 1$, tj.

$$t_0 = \dots = t_p, \quad t_{m-p} = \dots = t_m \quad (\text{clamped}),$$

dostáváme tzv. **clamped B-spline křivku**.



- clamped B-spline křivka **prochází** prvním a posledním kontrolním bodem a **je tečná** k první a poslední hraně kontrolního polygonu.



Uzavřená B-spline křivka

- Pokud jsou první a poslední segment B-spline křivky spojeny tak, že

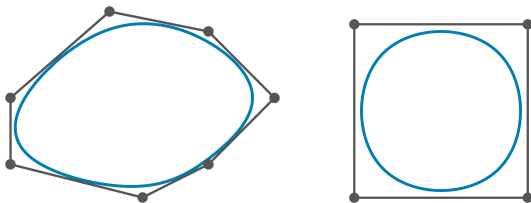
$$\mathbf{c}^{(k)}(t_0) = \mathbf{c}^{(k)}(t_m) \quad \text{pro } k = 0, 1, \dots, p - 1,$$

dostáváme tzv. **uzavřenou (closed) B-spline křivku**. Křivka i její derivace až do řádu $p - 1$ jsou spojité v místě spojení.

- Uzavřenost se dosahuje **periodickým rozšířením** uzlového vektoru a kontrolních bodů:

$$\mathbf{d}_{n+i} = \mathbf{d}_i, \quad t_{m+i} = t_i + (t_m - t_0),$$

tedy opakováním prvních p kontrolních bodů na konci.



Spojítost B-spline křivky v uzlech

- ▶ Spojitost B-spline křivky v uzlových bodech závisí na **násobnosti uzlu** r .
- ▶ Pokud má uzel t_i násobnost r , pak je křivka v tomto bodě spojitá řádu C^{p-r} .
- ▶ **Důsledky:**
 - ▶ všechny uzly jednoduché ($r = 1$) \Rightarrow spojitost C^{p-1} ,
 - ▶ uzel dvojnásobný ($r = 2$) \Rightarrow spojitost C^{p-2} ,
 - ▶ uzel s násobností $r = p \Rightarrow$ pouze spojitost C^0 (spojitá, ale nehladká).
- ▶ Zvyšování násobnosti uzlu tedy **snižuje hladkost** křivky v daném místě.

kvadratická clamped B-spline křivka:

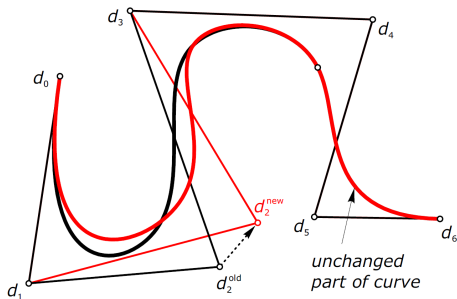


$$U = \{0, 0, 0, 1, 2, 3, 4, 5, 5, 5\}$$

$$U = \{0, 0, 0, 1, 2, 2, 3, 4, 4, 4\}$$

Vlastnosti B-spline křivek

- ▶ **Lokální kontrola:** změna d_i ovlivní jen úsek $\langle t_i, t_{i+p+1} \rangle$.
- ▶ **Silný konvexní obal:** na intervalu $\langle t_j, t_{j+1} \rangle$ leží $c(u)$ v konvexním obalu $p + 1$ bodů.
- ▶ **Spojitost:** řízena násobností uzlů.
- ▶ **Afinní invariance:** afinní transformace kontrolních bodů = afinní transformace křivky.
- ▶ **Útlum oscilací:** křivka nemá více průsečíků s přímkou než kontrolní polygon.
- ▶ **Bézierovy křivky** jsou speciálním případem B-splinů ($n = p$).



Motivace: kružnice jako racionální křivka

- ▶ Polynomiální křivky (např. Bézierovy nebo B-spline) **nemohou přesně popsat kružnici.**

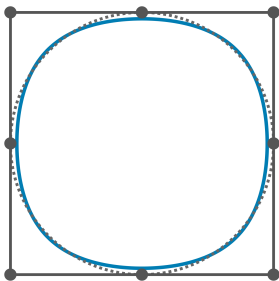
- ▶ Rovnice kružnice

$$x^2 + y^2 = 1$$

není splnitelná pro žádné polynomiální funkce $x(u), y(u)$.

- ▶ Pomocí polynomiálních segmentů lze kružnici **pouze aproximovat.**
- ▶ Potřebujeme **racionální křivky.**

Uzavřená B-spline křivka stupně 8:



Osnova

- 1 CAGD
- 2 Bézierovy křivky
- 3 B-spline křivky
- 4 Racionální techniky**
- 5 Subdivizní křivky

Racionální techniky – úvod

- ▶ **Racionální parametrické křivky a plochy** jsou klíčové v počítačovém geometrickém modelování.
- ▶ Historicky: počátky CAD systémů stavěly na **kuželosečkách** a **kvadratických plochách**.
- ▶ Kuželosečky se používaly např. v **leteckém průmyslu** (Ford, Boeing).
- ▶ Potřeba jednotného formátu pro výměnu dat vedla k vývoji **NURBS** (Non-Uniform Rational B-Splines).
- ▶ **NURBS** byly zavedeny ve Versprillově disertaci a standardizovány v softwaru **I-DEAS** (Integrated Design and Engineering Analysis Software), což byl jeden z prvních integrovaných CAD/CAE systémů.
- ▶ Dnes jsou NURBS nedílnou součástí formátů **IGES** a **STEP**, a standardem v komerčních CAD balících.
- ▶ **IGES** (Initial Graphics Exchange Specification) je starší standardní formát pro výměnu geometrických dat mezi CAD systémy.
- ▶ **STEP** (ISO 10303, Standard for the Exchange of Product model data) je modernější a obecnější standard pro výměnu CAD modelů včetně topologie a struktury.
- ▶ Algoritmy pro Bézierovy křivky a plochy lze snadno zobecnit na racionální případ.

NURBS křivky – definice

Definice (NURBS křivka)

NURBS křivka (Non-Uniform Rational B-Spline) stupně p je parametrická křivka definovaná kontrolními body $\mathbf{d}_i \in \mathbb{R}^n$, $n = 2, 3$, vahami $w_i > 0$ a uzlovým vektorem $U = \{t_0, t_1, \dots, t_m\}$, kde $m = n + p + 1$:

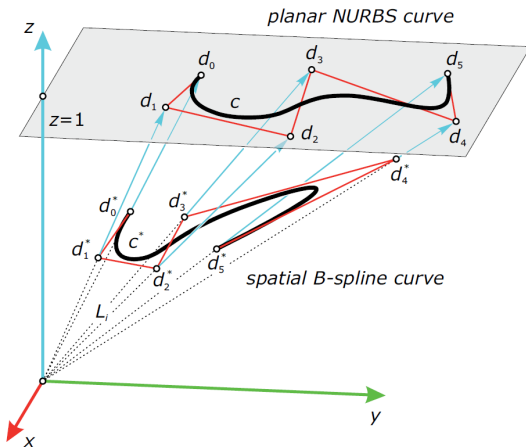
$$\mathbf{c}(u) = \frac{\sum_{i=0}^n N_{i,p}(u) w_i \mathbf{d}_i}{\sum_{i=0}^n N_{i,p}(u) w_i}, \quad u \in \langle t_p, t_{m-p} \rangle.$$

kde $N_{i,p}(u)$ jsou B-spline bázové funkce stupně p .

- ▶ „Non-Uniform“ znamená, že uzlový vektor nemusí být rovnoměrný (ale může být).
- ▶ „Rational“ označuje, že jde o zobecnění polynomiálních B-spline křivek – díky vahám w_i .

NURBS křivky – geometrická interpretace

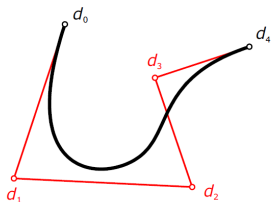
- ▶ Mějme kontrolní body $\mathbf{d}_0, \dots, \mathbf{d}_m$, které vložíme do roviny $z = 1$.
- ▶ Z počátku vedeme přímky L_i k bodům \mathbf{d}_i a posuneme po nich body do nových pozic \mathbf{d}_i^* .
- ▶ Body \mathbf{d}_i^* tvoří kontrolní polygon **prostorové** B-spline křivky $\mathbf{c}^*(u)$.
- ▶ **Centrální projekcí** této křivky do roviny $z = 1$ vznikne **rovinná NURBS** křivka $\mathbf{c}(u)$.



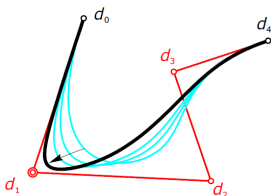
NURBS křivky – váhy

- ▶ Zvýšení váhy w_i **přitahuje** křivku k odpovídajícímu kontrolnímu bodu \mathbf{d}_i .
- ▶ Snížení váhy w_i naopak **oddaluje** křivku od bodu \mathbf{d}_i .
- ▶ Váhy jsou tedy **smysluplný tvarový parametr** – umožňují lokální řízení tvaru křivky.
- ▶ Z geometrické konstrukce plyne, že změna váhy w_i odpovídá posunu bodu \mathbf{d}_i^* prostorové B-spline křivky po přímce L_i .
- ▶ Posun jednoho kontrolního bodu ovlivňuje pouze lokální část křivky – tedy i změna jedné váhy má **lokální účinek**.

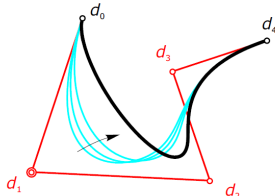
uniform weights



increasing weights in d_1



decreasing weights in d_1

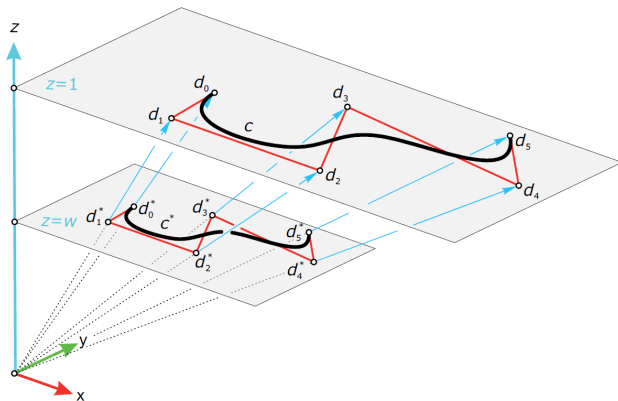


B-spline křivky jako speciální NURBS

- ▶ **B-spline křivka** je zvláštní případ NURBS křivky, kde všechny váhy jsou stejné:

$$w_0 = w_1 = \dots = w_n = w.$$

- ▶ Skutečnou NURBS křivku dostaneme pouze tehdy, když alespoň jedna váha w_i je **odlišná** od ostatních.



NURBS křivky – designové parametry

- ▶ Každá Bézierova i B-spline křivka je **speciálním případem** NURBS křivky.
- ▶ Následující přehled shrnuje, jaké **designové parametry** může uživatel nastavovat:

Typ křivky	Kontrolní body	Stupeň	Uzlový vektor	Váhy
Bézier	✓			
B-spline	✓	✓	✓	
NURBS	✓	✓	✓	✓

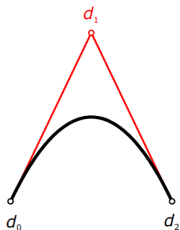
- ▶ U Bézierovy křivky lze měnit pouze kontrolní body, stupeň a uzlový vektor jsou implicitně pevně dány a váhy jsou všechny rovny 1.
- ▶ U B-spline křivky lze volit kontrolní body, stupeň a uzlový vektor, ale váhy jsou opět $w_i = 1$.
- ▶ Pouze u **obecné NURBS** křivky lze nastavovat **všechny čtyři** parametry: kontrolní body, stupeň, uzlový vektor i váhy.
- ▶ NURBS křivky zároveň dědí vlastnosti B-spline křivek: **lokální řízení tvaru** a **konvexní obal**.

NURBS křivky – kuželosečky jako speciální případy

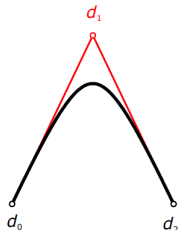
- ▶ Kuželosečky lze reprezentovat jako **speciální NURBS křivky** se třemi kontrolními body $\mathbf{d}_0, \mathbf{d}_1, \mathbf{d}_2$ a odpovídajícími vahami w_0, w_1, w_2 .
- ▶ Volbou vah získáme oblouky paraboly, hyperboly, elipsy i kružnice:

Kuželosečka	w_0	w_1	w_2
Parabola	1	1	1
Hyperbola	1	> 1	1
Elipsa	1	< 1	1
Kružnice	1	$\sin(\varphi/2)$	1

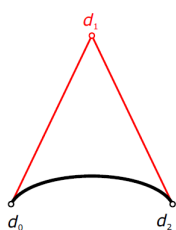
parabolic arc



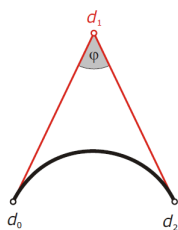
hyperbolic arc



elliptical arc



circular arc

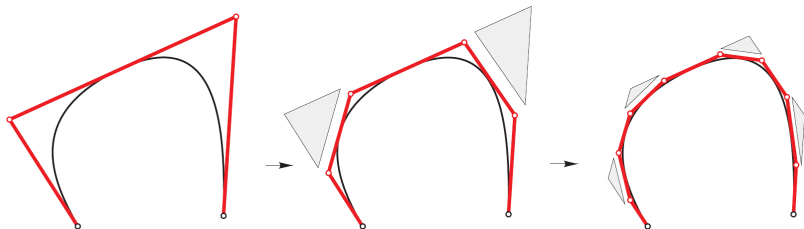


Osnova

- 1 CAGD
- 2 Bézierovy křivky
- 3 B-spline křivky
- 4 Racionální techniky
- 5 Subdivizní křivky

Subdivizní křivky – základní princip

- ▶ **Subdivizní křivky** vznikají **iterativním zpřesňováním** (refinací) zadaného hrubého kontrolního polygonu, dokud v limitě nedostaneme hladkou křivku.
- ▶ S pojmem subdivize jsme se setkali již u **de Casteljaouva algoritmu** pro generování Bézierovy křivky.
- ▶ Vlastnost subdivize znamená, že kontrolní polygon lze opakovaně zpřesňovat tak, že sekvence těchto polygonů rychle konverguje k Bézierově křivce.
- ▶ Proces lze chápat jako „**ořezávání rohů**“ – v každé iteraci odřezáváme vrcholy aktuálního polygonu, čímž se jeho tvar postupně vyhlazuje.

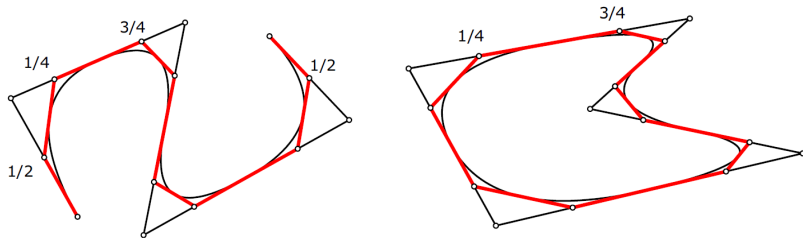


Subdivizní křivky – definice a historie

- ▶ **Subdivizní křivka** je polygon určený dvěma entitami:
 - ▶ kontrolními body a
 - ▶ úrovní subdivize (refinace) k .
- ▶ Mluvíme tedy o „**subdivizní křivce úrovně k dané kontrolními body**“.
- ▶ Podle typu algoritmu rozlišujeme **aproximační** a **interpolující** subdivize.
- ▶ Příklady algoritmů:
 - ▶ **Chaikinův algoritmus** – generuje kvadratické B-spline křivky.
 - ▶ **Lane–Riesenfeldův algoritmus** – zobecnění Chaikina, v limitě dává uniformní B-spline křivku stupně n .
 - ▶ **Čtyřbodové schéma** – interpolační subdivize (křivka prochází kontrolními body).
- ▶ Historie:
 - ▶ Myšlenka „**ořezávání rohů**“ (*corner cutting*) sahá až k G. de Rahmovi (1947), který popsal tzv. „třetinové“ a později „čtvrtinové“ dělení (v $1/4$ a $3/4$).
 - ▶ G. Chaikin nezávisle znovuobjevil roku 1974 algoritmus s dělením v $1/4$ a $3/4$ pro rychlé generování hladkých křivek na počítači.
 - ▶ Tento postup se stal známým jako **Chaikinův algoritmus**.

Chaikinův algoritmus – aproximační subdivize

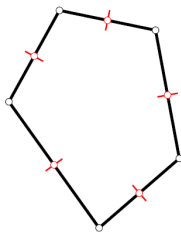
- ▶ VSTUP: kontrolní polygon.
- ▶ V každém kroku subdivize provedeme **ořezání rohů** (*corner cutting*):
 - ▶ každou hranu rozdělíme v poměrech $1/4$ a $3/4$ pomocí lineární interpolace,
 - ▶ nové body propojíme a vytvoříme tak nový polygon.
- ▶ Opakováním procesu získáme posloupnost polygonů, která v limitě konverguje ke **kvadratické uniformní B-spline křivce**.
- ▶ Chaikinův algoritmus je příklad **aproximační subdivize** – výsledná křivka neprochází vstupními kontrolními body.
- ▶ Pro otevřený kontrolní polygon je nutná modifikace: první a poslední hrana se dělí pouze jednou – v jejím středu.



Chaikinův algoritmus – metoda „split and average“

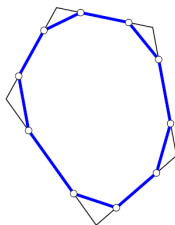
- ▶ Chaikinův **ořezávací algoritmus** lze chápat také jako **postup rozděl a zprůměruj**.
- ▶ Každý krok subdivize se skládá ze dvou fází:
 - 1 **Rozdělení (split)**: do každé hrany vložíme její střed, čímž vznikne nový polygon s dvojnásobným počtem vrcholů.
 - 2 **Průměrování (average)**: pro všechny hrany nového polygonu vypočteme jejich středy (průměry sousedních bodů).
- ▶ Spojením těchto nových vrcholů získáme polygon pro další krok subdivize.
- ▶ V limitě tento proces vede opět ke **kvadratické uniformní B-spline křivce**.

step 1

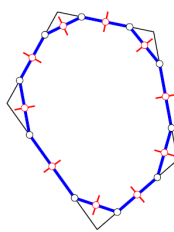


split

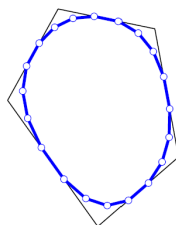
step 2



1 x average



split

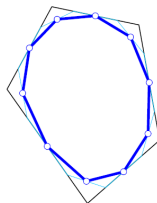
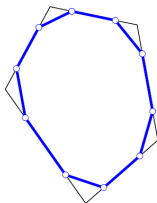
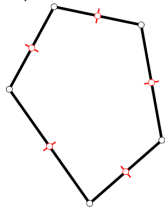


1 x average

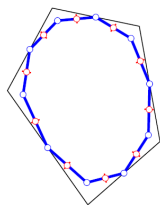
Lane–Riesenfeldův algoritmus

- ▶ Roku 1980 Lane a Riesenfeld zobecnili Chaikinův postup na „split and $n \times$ average“.
- ▶ V limitě vzniká **uniformní B-spline** křivka stupně $n + 1$.

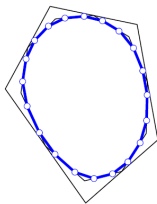
step 1



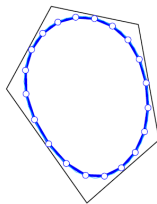
step 2



split



1 x average



2 x average

Čtyřbodové schéma – interpolační subdivize

- ▶ Na rozdíl od Chaikinova algoritmu má výsledná křivka body **interpolovat**.
- ▶ Dubuc (1986) ukázal, že z bodů $\mathbf{p}_{i-1}, \mathbf{p}_i, \mathbf{p}_{i+1}, \mathbf{p}_{i+2}$ lze vytvořit nový bod podle vztahu:

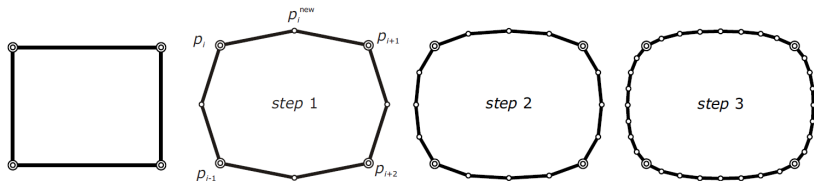
$$\mathbf{p}_i^{\text{new}} = -\frac{1}{16} \mathbf{p}_{i-1} + \frac{9}{16} \mathbf{p}_i + \frac{9}{16} \mathbf{p}_{i+1} - \frac{1}{16} \mathbf{p}_{i+2}.$$

- ▶ Součet koeficientů

$$(-1 + 9 + 9 - 1)/16 = 1$$

zaručuje **geometrický smysl** schématu.

- ▶ Koeficienty byly **odvozeny matematicky** z jednoznačně určené kubické křivky interpolující čtyři dané body.



Čtyřbodové schéma – obecná forma

- ▶ Obecnější tvar čtyřbodového schématu je dán vztahem

$$\mathbf{p}_i^{\text{new}} = -w \mathbf{p}_{i-1} + \left(\frac{1}{2} + w\right) \mathbf{p}_i + \left(\frac{1}{2} + w\right) \mathbf{p}_{i+1} - w \mathbf{p}_{i+2}.$$

- ▶ Pro

$$w = \frac{1}{16}$$

dostáváme původní Dubucovo čtyřbodové schéma.

- ▶ Ne všechny hodnoty parametru w však vedou k **hladké limitní křivce**.

