

# Kreslicí nástroje v systémech CAD



Se základními grafickými prvky je možné se setkat ve všech grafických aplikacích. Grafické aplikace jsou podle principu práce s grafickými prvky rozděleny na rastrové a vektorové editory. Do skupiny vektorových editorů patří všechny CAD programy, ale také například většina textových editorů. Rastrové editory pracují vždy jen s grafickým prvkem typu bod. Informace, které jsou o bodu uchovány jsou:

barevná hloubka  
poloha bodu v kreslicí ploše

Vektorové editory používají jako základní grafický prvek úsečku. O úsečce jsou v systému uchovávány tyto informace:

- poloha počátečního a koncového bodu
- barva
- tloušťka
- typ čáry

V současnosti všechna grafická zařízení pracují s rastrovým způsobem zobrazování. Vektorové grafické prvky jsou proto z důvodu zobrazení nebo vytisknutí převáděny na rastrové zobrazení. To znamená, že úsečka je převedena na sled bodů, které se zobrazují nebo tisknou. Určování polohy a barvy bodů, kterými je úsečka nahrazena se jmenuje rasterizace.

Mezi základní dvourozměrné kreslicí prvky patří:

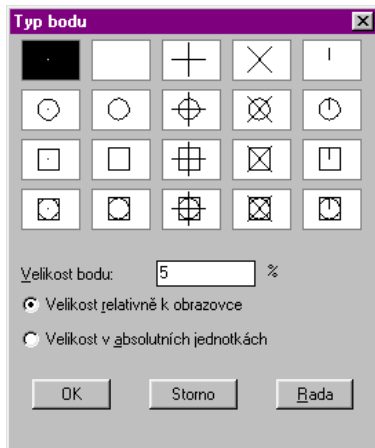
- bod
- úsečka
- kružnice
- elipsa
- oblouk
- křivka

Kreslicí prvky jsou často označovány jako entity. Sloučením entit do celků vznikají objekty. K základním objektům patří:

- obdélník
- mnohoúhelník

Objekty s širšími možnostmi nastavení jsou například tyto:

- text
- blok



Obr.1 – Dialogový panel pro definici stylu vykresleného bodu

## Kreslení bodu

Kreslení a zobrazení bodu patří k jednoduchým činnostem. Vykreslení bodu je uskutečněno na základě zadaní polohy bodu v kreslicí ploše aplikačního programu a v převedení těchto souřadnic na odpovídající instrukci výstupu. Na obrazovce dojde k vysvícení jednoho pixelu příslušnou barvou.

Vykreslení bodu v CAD systémech je ve většině případů uskutečněno prostřednictvím grafické značky, která se skládá z úseček, kružnic nebo jejich kombinací. V tomto případě se již nejedná o vykreslení bodu, ale o vykreslení jiných entit.

Vykreslení entit je popsáno v další části. Pro informaci je na obr.2 zobrazen dialogový panel z CAD programu AutoCAD, pomocí kterého se definuje typ zobrazení bodu.

## Kreslení úsečky a lomené čáry

Úsečka patří mezi základní kreslicí prvky. V CAD systému je úsečka definována dvěma způsoby:

- počátečním a koncovým bodem
- počátečním bodem, délkou a úhlem natočení vzhledem nulovému úhlu aktuálního souřadnicového systému

Kromě těchto dvou definic poloh jsou v některých systémech nastavitelné další vlastnosti:

- typ čáry
- tloušťka čáry.

*Typ čáry* je například čára čárkovaná nebo čerchovaná. Většina CAD systémů má předdefinované typy čar. Některé programy mají tyto čáry definovány jako interní nastavení, některé používají externí definice typů čar. Například instalace AutoCADu obsahuje soubor ACAD.lin a ACADISO.lin.

*Tloušťka čáry* je ve většině CAD systémů nastavena na počáteční tloušťku, která je nulová nebo rovna jedné. Některé grafické systémy, například *Corel Draw* umí úplně potlačit tloušťku čáry. Teprve nastavením hodnoty na větší než nula, je její tloušťka vidět na stínítku monitoru.

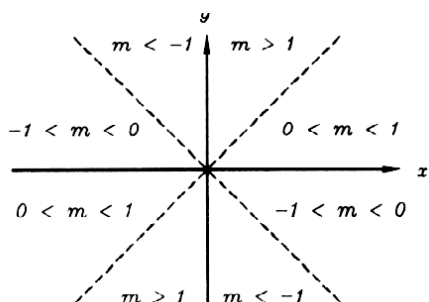
*Lomená čára (polyline)* je posloupností úseček, kterým se říká segmenty. Počáteční bod nové úsečky navazuje na koncový bod předchozí úsečky. Po dokončení kreslení lomené čáry jsou jednotlivé segmenty samostatně editovatelné.

## Převedení úsečky na rastrové zobrazení

Aby mohla být úsečka zobrazena na rastrovém zařízení, je nutné ji převést na sled bodů. Této činnosti se říká *rasterizace*. Metod rasterizace je několik. Vždy se vychází z rovnice přímky

$$\Delta y = m \cdot \Delta x$$

Rasterizace probíhá po jednotkových krocích v ose  $x$  nebo  $y$ . Vždy záleží na *směrnici přímky*  $m$ . V případě  $m \leq 0$  je prováděn jednotkový krok ve směru osy  $x$ , při  $m > 1$  je prováděn jednotkový krok ve směru osy  $y$ . Hodnoty směrnice pro různé směry je na obr.2.

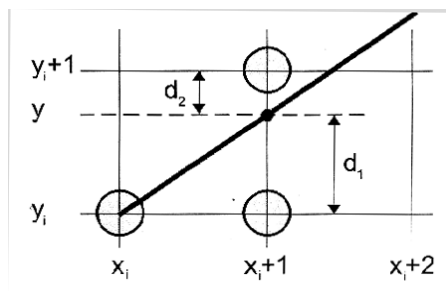


Obr.2- hodnoty směrnice  $m$

## Algoritmus DDA

Algoritmus *DDA* – *Digital Differential Analyzer*. Tento algoritmus je příkladem postupného (iteračního) výpočtu. Nové hodnoty jsou získávány z dříve vypočítaných. Velikost kroku, kterým je přímka rasterizována je stanovena na rozteč jednoho pixelu. Vždy je jeden směr řídicí. V tomto směru je přidáván jednotkový krok, druhý směr je neceločíselný, proto se musí zaokrouhlovat ne celá čísla. Postup výpočtu algoritmu je následující:

1. z koncových bodů úsečky se vypočítá směrnice a určí se řídicí osa
2. zapsání počátečního bodu
3. opakování cyklu dokud není dosaženo koncového bodu úsečky se směru řídicí osy
  - a) vykreslení bodu o souřadnicích  $x$  a zaokrouhleně  $y$
  - b) přesun o jednotkový krok se směru řídicí osy
  - c) výpočet a zaokrouhlení  $y$



Obr.3 – výběr pixelů úsečky v Bresenhamově algoritmu

## Bresenhamův algoritmus

Algoritmus vytváření úsečky je založen na principu nacházení bodů rastru, které leží nejbližší skutečné poloze bodu úsečky. K tomu je využívána celočíselná matematika. Postup je následující:

1. vypočítá se směrnice úsečky a z ní se určí, která osa je řídicí
2. nakreslí se počáteční bod úsečky daný souřadnicemi  $x$  a  $y$
3. rozhodne se o poloze dalšího bodu, zda bude mít souřadnici  $y$  stejnou jako bod předchozí nebo o jeden

pixel vyšší. K tomu se použije výpočtu rozdílu souřadnic  $\Delta d$ .

Při výpočtu se vychází z obecné rovnice přímky

$$y = mx + b$$

rozdíly souřadnic  $d_1$  a  $d_2$

$$d_1 = y - y_i$$

$$d_2 = y_i + 1 - y$$

$$\Delta d = d_1 - d_2$$

Podle proměnné  $\Delta d$  lze snadno určit, který ze dvou pixelů leží blíže skutečné úsečce. Není ani tak důležitá hodnota  $\Delta d$ , jako znaménko. Toto se využívá při převodu výpočtu do celočíselné aritmetiky.

## Zápis do obrazové paměti

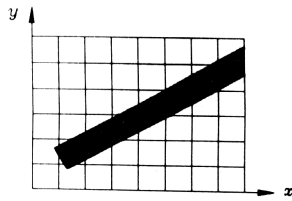
Pixel o souřadnicích  $x, y$  leží na v obrazové paměti na adrese

$$addr(x, y) = adr\_start + y(xmax + 1) + x$$

Tento výpočet obsahuje násobení. Zrychlení výpočtu spočívá na předpokladu, že sousední body přímky leží ve vzdálenosti jednoho pixelu ve směru  $x$  a  $y$ . Pak adresa následujícího pixelu se vypočítá:

$$addr(x + 1, y) = addr(x, y) + 1$$

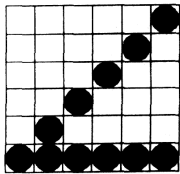
Rasterizace úsečky přináší některé nežádoucí vlivy při zobrazení. První je jev, který se nazývá *roztřepení* (*aliasing*). Roztřepení projevuje schodečkovým vzhledem šikmých úseček. Odstranění roztřepení je možné dvěma způsoby:



Obr.4 – teoretické pokrytí pixelů úsečkou

- nastavením vyššího rozlišení zobrazovacího zařízení – tato možnost není vždy dostupná, protože zařízení nemusí vždy poskytnout dostatečně velké rozlišení
- úpravou algoritmů pro vykreslení úsečky tak, aby roztřepení bylo co nejmenší – tato metoda je často v grafických systémech dostupná, většinou uživatelsky nastavitelná, to znamená zapnutí režimu vyhlazování.

Principem vyhlazování – *antialiasingu* je upravování jasů pixelů tvořících úsečku. Pixel není nekonečně malý, ale obsazuje jistou plochu na obrazovce. Plocha je daná konstrukcí monitoru. Cílem úpravy algoritmu je zjistit, jak velkou plochu pixelu překrývá plocha vykreslované úsečky (obr.x). Většinou se využívá úpravy Bresenhamova algoritmu, ve kterém je hodnoty predice  $p$  odvozena jasová úroveň.



Obr.5-rozdílné vzdálenosti bodů úsečky

Dalším nežádoucím vlivem je opticky nestejný jas horizontální a šikmé úsečky. tento jev je způsoben tím, že pixely šikmé úsečky jsou od sebe vzdáleny o  $\sqrt{2}$  krát (obr.x). V algoritmech pro kreslení úsečky se tento jev odstraňuje zvyšováním jasů šikmé úsečky podle velikosti její směrnice.

## Kružnice

Kružnice je nejčastěji definována polohou středu o souřadnicích  $x$ ,  $y$  a poloměrem  $r$ . V CAD systémech je kružnice definována ještě dalšími metodami:

- dvěma body
- třemi body
- dvěma tečnami a velikostí poloměru



Obr.6 – Ikonové menu

Příkladem je ikonové menu na obr.6.

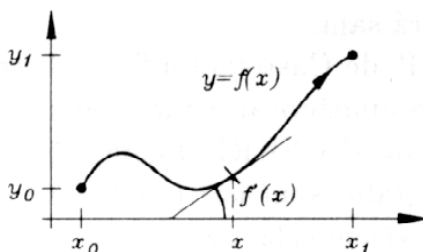
Při kresbě kružnice je možné využít existujících algoritmů pro kreslení úsečky a nahradit kružnici lomenou čarou. Jedná se nepřesný, ale rychlý postup, který se využívá při dynamickém vytváření kružnice. Po konečném umístění kružnice do kreslicí plochy se vytvoří přesným umístěním jednotlivých pixelů.

jednotlivých pixelů.

## Křivky

Křivky se používají obecně v počítačové grafice v aplikacích, které umožňují vytvářet grafické informace, v mnoha podobách. V CAD systémy umožňují například modelovat povrchy těles s využitím hladkých ploch nebo jsou použity jako obrysy při modelování trojrozměrných těles pomocí hraničního modelování. Dále jsou křivky využívány při animacích, kde se jimi definují trajektorie pohybu objektů. V textových editorech jsou křivky zastoupeny v podobě definice fontů. Požadavky na modelování křivek a později ploch v CAD systémech:

- plně řídit tvar plochy jednoduchými nástroji
- možnost vytváření ostrých hran
- zajištění hladkých přechodů mezi plochami
- zajištění spojitosti



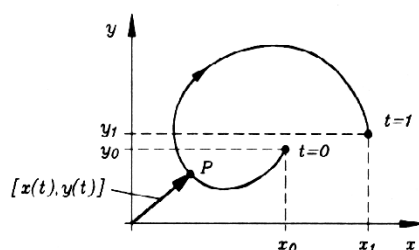
Obr.7– Explicitní vyjádření funkce

Aby bylo možné takto s křivkami pracovat, je nutné znát základní chování dvojrozměrných křivek, které jsou základem ploch.

## Vyjádření křivek v CAD systémech

Křivky jsou obvykle reprezentovány matematickými rovnicemi, ze kterých se v případě potřeby CAD program generuje zobrazení křivky na monitoru. Matematická reprezentace rovinné křivky je možná třemi způsoby:

- explicitně
- implicitně
- parametricky



Obr.8 – Parametrické vyjádření křivky

*Explicitní vyjádření* je vyjádřeno funkcí (obr.7):  $y = f(x)$   
Derivace v obecném bodě  $x$  určuje tečnu ke křivce v daném bodě.

*Implicitní vyjádření* má tvar:  $F(x,y) = 0$ .

Takovéto zadání křivky není příliš vhodné pro generování křivky v CAD systémech. Není umožněn postupný výpočet bodů křivky. Implicitní vyjádření se však používá při řešení průsečíků křivek a později i ploch v trojrozměrném prostoru.

*Parametrické vyjádření* je nejčastější způsob vyjádření křivek. Parametrický tvar je následující:  $x = x(t)$ ,  $y = y(t)$ . Častěji používán vektorový zápis  $P(t) = [x(t), y(t)]$ , kde  $P(t)$  je polohový vektor. Parametr  $t$  je zadáván v intervalu  $\langle 0,1 \rangle$ . Parametrické vyjádření je znázorněno na obr.8.

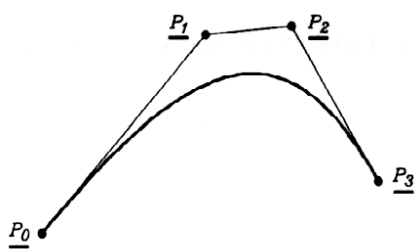
Rozdělení křivek a ploch:

- aproximační – křivka nebo plocha neprochází zadanými body, pouze se kolem nich prochází
- interpolační - vedení křivky nebo plochy zadanými body

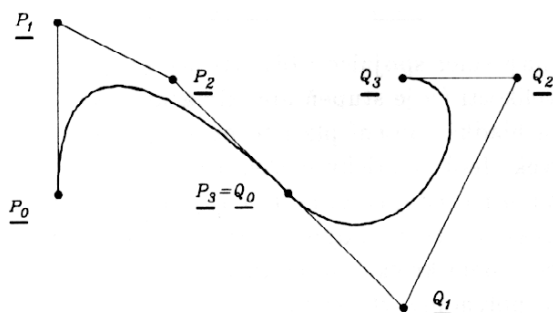
Body, kterými je křivka nebo plocha definována jsou označovány jako *uzly* nebo *uzlové body*. Například program AutoCAD dokáže proložit uzly křivku typu *spline* nebo *aproximační křivku*. Použití aproximační křivky je možné v případě kreslení rozvinutých plášťů rotačních ploch, použití spline spadá do oblasti volné tvorby designu.

Nejnámější metody a autoři z oblasti křivek a ploch:

- Ferguson – (firma Boeing) zadávání křivek a ploch pomocí okrajových podmínek (zadáním polohy bodu a tečny v krajním bodu).
- Coons – (universita M.I.T) – zobecnil Fergusonovy křivky a plochy a zavedl tzv. *spline*. Spline se bez problémů spojují.
- Bézier – (firma Renault) – zavedl aproximační plochy zadávané sítí bodů. Sít' bodů je velmi snadno intuitivně ovladatelná.
- Barsky – zobecnil *B-spliny* a zavedl *beta-spline*, které umožňují pomocí dvou intuitivních parametrů kontrolovat tvar křivky nebo plochy.



Obr.9 - Kubická Bézierova křivka



Obr.10 - Napojení Kubických Bézierových křivek

## Bézirové křivky

Bézirové křivky aproximují posloupnost alespoň dvou uzlů. V tomto případě je aproximační křivka úsečka. Základní vlastnosti bézirové křivky jsou:

- začíná a končí v prvním a posledním uzlu, těmito uzly křivka vždy prochází
- křivka se na koncích přimyká ke spojnici krajních uzlů

Typy Béziových křivek:

- obecná – je definována libovolným počtem uzlů. Nevýhodou takto zadané křivky je to, že změna polohy libovolného vnitřního bodu křivky ovlivní průběh celé křivky. Druhou nevýhodou je to, že složitý průběh křivky je nutné aproximovat křivkou vyššího řádu, například deset. To má za následek vysokou výpočtovou náročnost.

- kubická – definované třemi uzly

- kvadratická – definované čtyřmi uzly

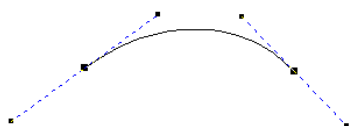
Kubická a kvadratická křivka umožňuje lokální změny průběhu křivky. Použití spočívá v tom, že jedna složitá křivka se rozloží na kratší křivkové úseky, které se prokládají křivkami menšího řádu. U takto rozložené složitě křivky

je nutné zajistit dvě podmínky:

- spojitost celé křivky
- hladkost v požadovaných uzlech

Spojitost je zajištěna tím, že je ztotožněn koncový uzel předchozí křivky s počátečním uzlem

jedné



Obr.11 – Bézierova křivka v programu CorelDraw

křivky následující.

Hladkost je zajištěna v okamžiku, kdy leží na přímce uzly:

- předposlední na předcházející křivce
- koncový a počáteční bod
- druhý uzel následující křivky

Přidání další křivky vyžaduje zadání dvou nových uzlů.

Příklad použití Béziových křivek v CAD systémech AutoCAD nebo grafickém systému Corel Draw (obr.x), dále tvorba tvarů písem v DTP (TrueType, systém PostScript)



## Coonsovy (B-spline) křivky

B-spline jsou typy aproximačních ploch. Principem zadání se podobají Béziovým křivkám. Odstraňují jejich obecnou nevýhodu při hladkém napojování.

Typy B-spline křivek:

- obecná – je definována libovolným počtem uzlů. Nevýhodou je to, že složitý průběh křivky je nutné aproximovat křivkou vyššího řádu, například deset. To má za následek vysokou výpočtovou náročnost.
- kubická – definované třemi uzly
- kvadratická – definované čtyřmi uzly

Obr.12 – Příklad vzhledu nástrojového panelu

Běžně je používána křivka kubická, která je definována čtyřmi uzly. Geometrické vlastnosti B-spline jsou následující:

- začátek křivky leží v antitěžišti trojúhelníka ABC
- směr křivky v krajním bodě je rovnoběžný se stranou trojúhelníka AC

Výhodné vlastnosti B-spline:

- při napojování další křivky stačí přidat jeden nový uzel
- složená křivka je vždy hladká
- složená křivka je vždy spojitá – spojitost druhého řádu

V případě, že je potřeba, aby křivka začínala v určitém bodě, je nutné použít vícenásobného uzlu – ztotožnění několika sousedních uzlů křivky. V případě, že leží vícenásobný uzel uvnitř křivky, je možné vytvářet ostré hrany na křivce.

## Beta-spline křivky

Beta-spline křivky jsou zobecněním kubických B-splinů. Jedná se opět o aproximační křivku. Jeden úsek křivky je zadán čtyřmi uzly, dva sousední úseky mají tři uzly společné (stejně jako u B-spline).

Tvar křivky je závislý na:

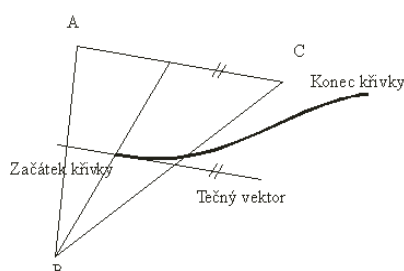
- poloze uzlů
- dvou číselných parametrech nazvaných *sklon* a *napětí*

*Sklon* – je označován jako  $\beta_1$ . Je to kladný parametr., který řídí posunutí křivky vzhledem k řídicím uzlům ( $\beta_1=1$  má B-spline). Vysokými hodnotami *sklonu* lze kontrolovat koncové body křivky.



Obr.13 – Coonsova kubika

*Napětí* - je označován jako  $\beta_2$ . Parametr ovlivňuje stupeň aproximace. Může nabývat jak kladných, tak záporných hodnot ( $\beta_2=1$  má B-spline). Vysoké hodnoty *napětí* umožňují převést křivku blíže k uzlům, to znamená, že lze vytvořit ostrou hranu.



Obr.14– Geometrická konstrukce B-spline



### **Otázky a příklady k procvičení:**

1. Jaké jsou základní kreslicí entity pro tvorbu 2D geometrie.
2. Co je antialiasing.
3. Způsob vyjádření křivek v systémech CAD.
4. Jaké vlastnosti má Bézierova křivka.



### **Literatura:**

Žára - Moderní počítačová grafika – Computer Press  
Žára - Počítačová grafika – principy a algoritmy - Grada  
Sochor – Algoritmy počítačové grafiky – ČVUT Praha