

Geometrická zobrazení v rovině a jejich aplikace ve vektorové grafice

JAKUB MAZUCH

PedF UK Praha – SPŠE Ječná – Jednota školských informatiků

V rámci středoškolské výuky matematiky se žák gymnázia krom jiného seznámí s geometrickými zobrazeními v rovině. V souladu RVP G platnými od 1. září 2022 je žák seznámen se shodnými zobrazeními (osová a středová souměrnost, posunutí, otočení) a se stejnolehlostí. Středoškolské učivo je omezeno na zobrazení z roviny do roviny. V každém takovém zobrazení přiřazujeme tedy libovolnému bodu X roviny ρ jako jeho obraz právě jeden bod X' téže roviny ρ (značíme $X \mapsto X'$). Jestliže speciálně obraz X' bodu X s ním splývá, pak bod $X = X'$ se nazývá samodružný bod.

Ke geometrickém zobrazením lze přistupovat syntetickou a analytickou (výpočetní) metodou. Analytická vyjádření jednotlivých typů zobrazení lze vyjádřit maticí. Zobrazení rozlišujeme na:

- shodnosti
- podobnosti

Shodnosti

Prosté zobrazení v rovině nazýváme shodným zobrazením (shodností), právě když pro každé dva body X, Y roviny a jejich obrazy X', Y' ($X \mapsto X', Y \mapsto Y'$) v tomto zobrazení platí

$$|X'Y'| = |XY|.$$

Zvláštním případem shodnosti je identita, jež každému bodu X dané roviny přiřazuje přiřazuje jako obraz též bod $X' = X$.

Přímá shodnost je každá shodnost, ve které libovolný $\triangle ABC$ a jeho obraz $\triangle A'B'C'$ jsou přímo shodné trojúhelníky, tedy při „obíhání“ jejich hranic od bodu A přes bod B k bodu C a od bodu A' přes bod B' do bodu C' postupujeme ve stejném smyslu otáčení. Nepřímá shodnost je každá taková shodnost, v níž libovolný $\triangle ABC$ a jeho obraz $\triangle A'B'C'$ jsou nepřímá shodné trojúhelníky, tedy při „obíhání“ jejich hranic od bodu A přes bod B k bodu C a od bodu A' přes bod B' do bodu C' postupujeme v opačných směrech otáčení.

Afinní zobrazení lze vyjádřit maticí:

$$A = \begin{pmatrix} a & b & m \\ c & d & n \\ 0 & 0 & 1 \end{pmatrix}$$

Bod $X' = [x'_1; x'_2]$, který je obrazem bodu $X = [x_1; x_2]$ lze analyticky zobrazit:

$$X' = AX$$

$$\begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \end{bmatrix} = \begin{pmatrix} a & b & m \\ c & d & n \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix}$$

Druhy přímých a nepřímých zobrazení [3]

Přímé shodnosti jsou:

a) Identita \mathcal{I}

$$\mathcal{I} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

b) Posunutí (translace) $\mathcal{T}(\mathbf{u})$

Je určeno vektorem posunutí $\mathbf{u} = (u_1; u_2)$, resp. maticí:

$$\mathcal{T}(\mathbf{u}) = \begin{pmatrix} 1 & 0 & u_1 \\ 0 & 1 & u_2 \\ 0 & 0 & 1 \end{pmatrix}$$

c) Otočení (rotace) $\mathcal{R}(S; \alpha)$

Je určeno středem rotace $S = [s_1; s_2]$ a (orientovaným) úhlem otočení α o velikosti $0^\circ < \alpha \leq 360^\circ$, resp. maticí:

$$\mathcal{R}(S; \alpha) = \begin{pmatrix} \cos \alpha & -\sin \alpha & s_1(1 - \cos \alpha) + s_2 \sin \alpha \\ \sin \alpha & \cos \alpha & -s_1 \sin \alpha + s_2(1 - \cos \alpha) \\ 0 & 0 & 1 \end{pmatrix}$$

d) Středová souměrnost $\mathcal{S}(S)$

Jedná se o speciální případ rotace – otočení kolem bodu $S = [s_1; s_2]$ (středu souměrnosti) o úhel $\alpha = 180^\circ$ (S je střed úsečky XX').

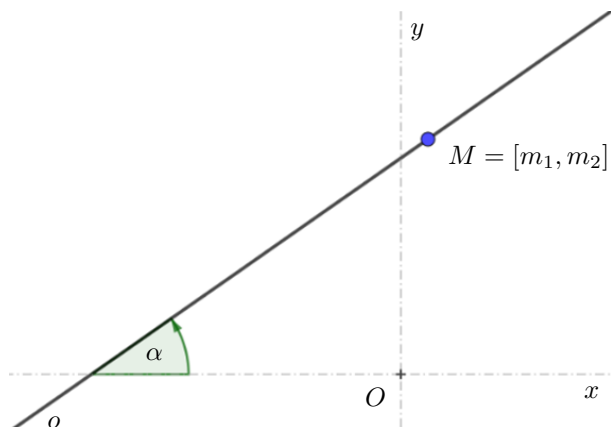
$$\mathcal{S}(S) = \begin{pmatrix} -1 & 0 & 2s_1 \\ 0 & -1 & 2s_2 \\ 0 & 0 & 1 \end{pmatrix}$$

Nepřímé shodnosti jsou:

e) Osová souměrnost s_o

Osová souměrnost s_o s osou o je zobrazení v rovině, ve kterém se zobrazí každý bod $X \in o$ na bod $X' = X$ a každý bod $X \notin o$ na X' tak, že o je osa úsečky XX' . Tedy $|XS| = |SX'|$, kde $S \in o$ je střed úsečky XX' . Přímku o nazýváme osou souměrnosti [1].

Osová souměrnost je určena přímkou o , jdoucí bodem $M = [m_1; m_2]$, a směrovým úhlem α . Značíme $s_o(m_1; m_2; \alpha)$.



$$s_o(u, v, \alpha) = \begin{pmatrix} \cos 2\alpha & \sin 2\alpha & u(1 - \cos 2\alpha) - v \sin 2\alpha \\ \sin 2\alpha & \cos 2\alpha & v(1 - \sin 2\alpha) + u \cos 2\alpha \\ 0 & 0 & 1 \end{pmatrix}$$

f) Posunutá osová souměrnost (posunuté zrcadlení) \mathcal{P}_S

$$\mathcal{P}_S = s_o(u, v, \alpha) \circ \mathcal{T}(\mathbf{u}) = \mathcal{T}(\mathbf{u}) \circ s_o(u, v, \alpha)$$

Podobnost a stejnolehlost

Podobné zobrazení (podobnost) $\mathcal{Z}: X \mapsto X'$ je geometrické zobrazení v rovině, pro které existuje kladné reálné číslo k tak, že pro každé dvě dvojice bodů A, A' a B, B' vzoru a obrazu je splněn vztah $|A'B'| = k \cdot |AB|$. Číslo k se nazývá poměr podobnosti.

Každá podobnost je prosté zobrazení. Podobně jako u shodnosti lze definovat podobnost přímou a nepřímou. Shodnost je zvláštní případ podobnosti pro $k = 1$.

Je dán bod S a reálné číslo κ , $\kappa \neq 0$. *Stejnolehlost (homotetie)* se středem S a koeficientem κ je zobrazení $\mathcal{H}(S, \kappa)$, které přiřazuje:

1. každému bodu $X \neq S$ bod X' tak, že platí $|SX'| = |\kappa| \cdot |SX|$; přitom pro $\kappa > 0$ leží bod X' na polopřímce SX , pro $\kappa < 0$ leží bod X' na polopřímce opačné k polopřímce SX ,
2. bodu S bod $S' = S$.

Jestliže $\kappa = 1$, je každý bod samodružný a zobrazení je identita. Je-li $\kappa = -1$, je stejnolehlost středovou souměrností. Je-li $\kappa = 0$, je obrazem každého bodu střed stejnolehlosti, proto je tento případ v definici vyloučen. Jediným samodružným bodem ve stejnolehlosti, která není identitou, je střed stejnolehlosti. Samodružné přímky jsou všechny přímky, které prochází středem stejnolehlosti [2].

Pro stejnolehlost platí:

- a) Přímka a její obraz jsou rovnoběžné.
- b) Úsečka a její stejnolehlý obraz jsou rovnoběžné. Pokud je koeficient stejnolehlosti kladný (resp. záporný), jsou úsečka a její obraz souhlasně (resp. nesouhlasně) orientovány.

- c) Poměr délek obrazu úsečky a jejího obrazu se rovná absolutní hodnotě koeficientu stejnolehlosti.
- d) Obrazem úhlu je úhel s ním shodný [1].

Inverzním zobrazení ke stejnolehlosti $\mathcal{H}(S, \kappa)$ je $\mathcal{H}(S, 1/\kappa)$. Ukázka je v učebnici [1].

Stejnolehlost (homotetie) $\mathcal{H}(S, \kappa)$ je určena středem stejnolehlosti $S = [s_1; s_2]$ a koeficientem κ nebo maticí

$$\mathcal{H}(S, \kappa) = \begin{pmatrix} \kappa & 0 & (1 - \kappa) \cdot s_1 \\ 0 & \kappa & (1 - \kappa) \cdot s_2 \\ 0 & 0 & 1 \end{pmatrix}$$

Středoškolské učivo geometrických zobrazení je však ukončeno vyjmenováním shodností, podobností a jejím určením. Žáci tak nejsou běžně seznámeni s analytickým popisem těchto zobrazení, s možnostmi vyjádření zobrazení pomocí matic (a vektorů). Ve velmi výjimečných případech jsou žáci okrajově s těmito možnostmi seznamováni, a to pouze ve výběrových seminářích. Samotné práci s maticemi je ve výuce matematiky věnováno minimum času – též spíše v rozšiřujících – volitelných předmětech.

Analytický popis zobrazení pomocí matic může otevřít nové přístupy k tvorbě webových stránek a designu. Matice mají důležitou roli i v jiných oblastech matematiky a informatiky, ale podle současného kurikula není vyučování matic součástí matematického vzdělání na středních školách.

Primárním nástrojem tvorby webových stránek je značkovací jazyk HTML. Tím přiřazujeme obsahu webu sémantiku. Design a způsob rozmístění prvků určuje soubor CSS pravidel. Výše zmíněný analytický popis zobrazení můžeme jednoduše aplikovat i v rámci vývoje webové grafiky. Specifické postavení ve vývoji webu, a hlavně webové grafice má technologie SVG.

SVG je značkovací jazyk a formát pro dvoudimenzionální vektorovou grafiku založený na XML. Podporuje tři typy grafických objektů – vektorové tvary, bitmapové obrázky a text. Díky bohatým grafickým funkcím se hodí pro použití na webu i v tisku. Formát podporuje metadata pro lepší vyhledávání a indexaci obsahu. Ačkoli byla jeho implementace omezená v minulosti z důvodu chybějící podpory v některých prohlížečích, v současnosti podporuje SVG většina desktopových a mobilních prohlížečů. SVG soubory jsou uloženy v XML textových souborech a mohou být editovány textovými editory nebo vektorovými editory

Syntaxe SVG se řídí zápisem XML, který popisuje jednotlivé grafické prvky a jejich vlastnosti. Každý element je uzavřen mezi značkami, které udávají jeho název. Vlastnosti elementu jsou definovány pomocí atributů, které jsou uvedeny uvnitř počáteční značky. Hodnoty atributů jsou odděleny mezerou. Kromě toho může být obsah elementu tvořen dalšími elementy, textem nebo odkazy na jiné soubory. SVG také podporuje inline styly a kaskádové styly pomocí CSS.

Příklad 1

Mějme dva čtverce definované v SVG:

```
<svg width="200" height="100" style="border:1px solid black">
  <rect width="100" height="100" x="0" y="0" class="obdelnik" />
</svg>
```

obr. 1

```
<svg width="200" height="100" style="border:1px solid black">
  <rect width="100" height="100" x="0" y="0" class="obdelnik"
    transform="rotate(30, 100, 100)"
  />
</svg>
```

obr. 2

Tento kód vytvoří dva obdélníky pomocí SVG. Oba obdélníky mají stejnou šířku a výšku, 100 a 100 pixelů, a jsou umístěny v levém horním rohu SVG kanvasu pomocí atributů $x="0"$ a $y="0"$. Oba obdélníky jsou ohraničeny černou čarou, což je nastaveno pomocí CSS vlastnosti $style="border: 1px solid black"$. Celý kód $\langle\text{svg}\rangle\dots\langle\text{svg}\rangle$ lze vložit do HTML či do jakéhokoliv SVG editoru.

První obdélník nemá žádnou transformaci a je tedy zobrazen standardně. Druhý obdélník má transformaci pomocí atributu $transform="rotate(30, 100, 100)"$. Tato transformace se středem v bodě $[100;100]$ (v pixelech) a s úhlem otočení $\alpha = 30^\circ$. Pro formálnější přehlednost je nutno dodat, že SVG využívá repéru $\langle O, (1; 0), (0; -1) \rangle$.



Výše užitou rotaci je možné matematicky definovat maticí. Pro zjednodušení uvedeme v kódu aproximovanou matici:

$$\mathcal{R}([100; 100]; 30) = \begin{pmatrix} \frac{\sqrt{3}}{2} & -\frac{1}{2} & -50(\sqrt{3} - 3) \\ \frac{1}{2} & \frac{\sqrt{3}}{2} & -50(\sqrt{3} - 1) \\ 0 & 0 & 1 \end{pmatrix} \doteq \begin{pmatrix} 0,866 & -0,5 & 63,397 \\ 0,5 & 0,866 & -36,603 \\ 0 & 0 & 1 \end{pmatrix}.$$

Matrice zobrazení $\text{matrix}(a,b,c,d,e,f)$ je funkcí určující zobrazení ve formě matice zobrazení šesti hodnot. SVG zápis $\text{transform}=\text{matrix}(a,b,c,d,e,f)$ je ekvivalentní s maticí zobrazení:

$$\begin{pmatrix} a & c & e \\ b & d & f \\ 0 & 0 & 1 \end{pmatrix}.$$

Deklarace `transform` bude mít hodnotu `matrix(0.866,0.5,-0.5,0.866,63.397,-36.603)`. Výsledný kód ilustrovaný obr. 3 bude mít následující podobu:

```
<svg width="200" height="100" style="border:1px solid black">
  <rect width="100" height="100" x="0" y="0" class="obdelnik"
    transform="matrix(0.866,0.5,-0.5,0.866,63.397,-36.603)"
  />
</svg>
```

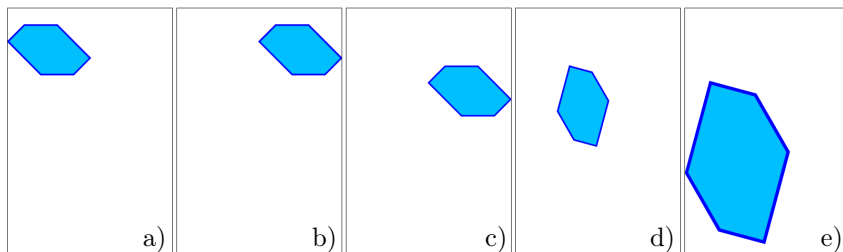
obr. 3



Hodnota deklarace `transform="rotate(30, 100, 100)"` je ekvivalentní s hodnotou `matrix(0.866,0.5,-0.5,0.866,63.397,-36.603)`.

Příklad 2

Je dán polygon, který je v SVG vyjádřen (viz obr. 4):



Obr. 4

```
<svg width="200" height="300" style="border:1px solid black">  
  <polygon points="20,20 60,20 100,60 80,80 40,80 0,40"  
    class="polygon"  
  />  
</svg>
```

Postupně na tento obrázek aplikujeme posunutí $\mathcal{T}(100, 0)$, $\mathcal{T}(0, 50)$, otočení $\mathcal{R}(60, 0, 0)$ a stejnoolehlost $\mathcal{H}(0, 0, 2)$.

Obrázek 4b) vznikl posunutím $\mathcal{T}(100, 0)$:

```
<svg width="200" height="300" style="border:1px solid black">  
  <polygon points="20,20 60,20 100,60 80,80 40,80 0,40"  
    class="polygon"  
    transform="translate(100,0)"  
  />  
</svg>
```

Obrázek 4c) vznikl posunutím $\mathcal{T}(0, 50)$:

```
<svg width="200" height="300" style="border:1px solid black">  
  <polygon points="20,20 60,20 100,60 80,80 40,80 0,40"  
    class="polygon"  
    transform="translate(100,0) translate(0,50)"  
  />  
</svg>
```

Zde vidíme ukázky možností skládání zobrazení i v SVG.

Tentýž obrázek 4c) mohl však vzniknout i složením $\mathcal{T}(100, 0) \circ \mathcal{T}(0, 50)$:

$$\mathcal{T}(100, 0) \circ \mathcal{T}(0, 50) = \begin{pmatrix} 1 & 0 & 100 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 50 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 100 \\ 0 & 1 & 50 \\ 0 & 0 & 1 \end{pmatrix}.$$

Tedy deklarací `transform="matrix(1,0,0,1,100,50)"`.

Obrázek 4d) může vzniknout různými způsoby:

(1)

```
<svg width="200" height="300" style="border:1px solid black">
  <polygon points="20,20 60,20 100,60 80,80 40,80 0,40"
    class="polygon"
    transform="
      translate(100,0)
      translate(0,50)
      rotate(60,0,0)"
  />
</svg>
```

(2)

```
<svg width="200" height="300" style="border:1px solid black">
  <polygon points="20,20 60,20 100,60 80,80 40,80 0,40"
    class="polygon"
    transform="matrix(1,0,0,1,100,50) rotate(60,0,0)"
  />
</svg>
```

(3) Nebo opět složením:

$$\begin{aligned} \mathcal{T}(100, 50) \circ \mathcal{R}(60, 0, 0) &= \\ &= \begin{pmatrix} 1 & 0 & 100 \\ 0 & 1 & 50 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \frac{1}{2} & -\frac{\sqrt{3}}{2} & 0 \\ \frac{\sqrt{3}}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & -\frac{\sqrt{3}}{2} & 100 \\ \frac{\sqrt{3}}{2} & \frac{1}{2} & 50 \\ 0 & 0 & 1 \end{pmatrix} \doteq \begin{pmatrix} 0,5 & -0,866 & 100 \\ 0,866 & 0,5 & 50 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

Vyjádření ve zdrojovém kódu SVG:

```
<svg width="200" height="300" style="border:1px solid black">
  <polygon points="20,20 60,20 100,60 80,80 40,80 0,40"
    class="polygon"
    transform="matrix(0.5,0.866,-0.866,0.5,100,50)"
  />
</svg>
```

Obrázek 4e) je výsledkem operace: $\mathcal{T}(100, 0) \circ \mathcal{T}(0, 50) \circ \mathcal{R}(60, 0, 0) \circ \mathcal{H}(0, 0, 2)$. Možností vyjádření v SVG je několik. Pro jednoduchost uvedeme jen dva extrémní:

1. Postupným vyjádřením všech transformací, přičemž stejnolehlost lze vyjádřit `transform="... matrix(2,0,0,2,0,0)"`

```
<svg width="200" height="300" style="border:1px solid black">
  <polygon points="20,20 60,20 100,60 80,80 40,80 0,40"
    class="polygon"
    transform="
      translate(100,0)
      translate(0,50)
      rotate(60,0,0)
      matrix(2,0,0,2,0,0)"
  />
</svg>
```

2. Uvedením výsledné matice:

$$\mathcal{T}(100, 50) \circ \mathcal{R}(60, 0, 0) \circ \mathcal{H}(0, 0, 2) =$$

$$= \begin{pmatrix} \frac{1}{2} & -\frac{\sqrt{3}}{2} & 100 \\ \frac{\sqrt{3}}{2} & \frac{1}{2} & 50 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & -\sqrt{3} & 100 \\ \sqrt{3} & 1 & 50 \\ 0 & 0 & 1 \end{pmatrix} \doteq \begin{pmatrix} 1 & -1,732 & 100 \\ 1,732 & 1 & 50 \\ 0 & 0 & 1 \end{pmatrix}$$

```
<svg width="200" height="300" style="border:1px solid black">
  <polygon points="20,20 60,20 100,60 80,80 40,80 0,40"
    class="polygon"
    transform="matrix(1,1.732,-1.732,1,100,50)"
  />
</svg>
```

V tomto článku jsem postupně ukázal aplikace běžné středoškolské matematiky ve webové grafice. Aktuální středoškolské kurikulum však není vzdělávání na středních školách v oblasti matematiky dostatečně zaměřeno na matice, které jsou v informatice klíčovým prvkem. Pokud by žáci zvládli matice, umožnilo by jim to snadněji pracovat s grafickými prvky v SVG formátu. Kromě webové grafiky se matice využívají třeba i v teorii grafů, nebo elektrotechnice.

Žáci všeobecných gymnázií jsou s maticemi seznamovány okrajově ve vyšších ročnících ve volitelných předmětech. Na jejich aplikace, ať už ve webové grafice, nebo teorii grafů již pak nezbývá čas. Domnívám se však, že na odborných školách, zaměřených na informační technologie či elektrotechniku by žáci měli být s maticemi seznámeni již na počátku studia. Navazující odborné předměty pak na těchto znalostech mohou dále stavět.

Literatura

- [1] *Pomykalová, E.*: Matematika pro gymnázia: planimetrie. Učebnice pro střední školy, 4. upr. vyd., Prometheus, Praha, 2000.
- [2] *Boček, L., Zhouf, J.*: Planimetrie. 2. rozš. vyd., Pedagogická fakulta, Univerzita Karlova, Praha, 2012.
- [3] *Hejný, M., Jirotková, D., Vondrová, N.*: Geometrické transformace: (metoda analytická). Univerzita Karlova, Praha, 1997.

Rozsvícené lampy (Úlohy z MO kategorie P, 46. část)

PAVEL TÖPFER

Matematicko-fyzikální fakulta UK, Praha

Pro dnešní pokračování dlouhodobého seriálu věnovaného vybraným problémům z Matematické olympiády kategorie P (programování) jsme zvolili jednu starší praktickou soutěžní úlohu. Jedná se o úlohu z celostátního kola 45. ročníku MO (školní rok 1995/96), tedy ze soutěže konané před téměř třiceti lety. Začneme jako obvykle původním zadáním úlohy,