Pokroky matematiky, fyziky a astronomie

Jan Zítko

Nový pohled na numerický výpočet největšího společného dělitele dvou polynomů

Pokroky matematiky, fyziky a astronomie, Vol. 55 (2010), No. 3, 231–242

Persistent URL: http://dml.cz/dmlcz/141962

Nový pohled na numerický výpočet největšího společného dělitele dvou polynomů

Jan Zítko, Praha

Úvod

Mějme dány dva polynomy f a g a spočítejme jejich největšího společného dělitele. Nechť koeficienty obou polynomů jsou takové, jaké se zadávaly ve škole, to znamená, že s tužkou mohu na papíře s přesnými čísly až do konce provést známý Eukleidův algoritmus a napsat největšího společného dělitele polynomů f a g. V tomto případě řekneme, že jsme výpočet provedli symbolicky. Pokud ovšem koeficienty polynomů budou uloženy v počítači a programem v konečné aritmetice provedeme bez jakýchkoliv dalších úprav Eukleidův algoritmus, pak výsledkem vůbec nemusí být největší společný dělitel. Tady se dostáváme do situace, kdy je postup ukončení algoritmu ovlivněn rozhodováním, zdali testované číslo je nebo není nula. V případě Eukleidova algoritmu se zvolí tolerance *tol* a pokud jsou koeficienty některého z vypočítaných polynomů v absolutní hodnotě menší než *tol*, považujeme polynom za nulový, jinak pokračujeme v dělení. Čtenář z tohoto vidí, že získání nesprávného výsledku může mít dva hlavní důvody. Buď přijmeme za nulu to, co nula není, a skončíme předčasně, nebo naopak odmítneme za nulu číslo, které nula je a ve výpočtu pokračujeme. V obou případech dostaneme špatný výsledek. Toto riziko je všeobecně známé. Použití násobné aritmetiky může být cesta, jak provést správné rozhodnutí o ukončení procesu dělení, což jsme si ověřili na mnoha spočítaných příkladech. Po tomto vysvětlení je čtenáři jasné, proč nemusíme dostat správný výsledek. Cílem článku je pojednat o matematické formulaci problémů technické praxe pro matematiky, které souvisí s Eukleidovým algoritmem, a napsat pro čtenáře informativní přehled. Tato úloha je v poslední době důležitá při řešení technických problémů a má uplatnění v těchto oborech:

1. Geometrické návrhy pomocí počítačů (Computer Aided Geometric Design) [4], [5].
2. Počítačová grafika (Computer Graphics) [6], [10].
3. Zpracování signálu (Signal Processing) [11].
4. Teorie řízení (Control) [12].

Článek má následující kapitoly. Ve druhé kapitole vyjádříme nejprve první krok Eukleidova algoritmu pomocí elementárních transformací Sylvestrovy matice. Uká-

Doc. RNDr. Jan Zítko, CSc., Katedra numerické matematiky, Matematicko-fyzikální fakulta, Univerzita Karlova, Sokolovská 83, 186 75 Praha 8. (e-mail: zitko@karlin.mff.cuni.cz)

Grant/contract sponzor: Článek je součástí výzkumného projektu MSM 0021620839 financovaného MŠMT.

žeme souvislost Eukleidova algoritmu s teorií matic a rozvinutý aparát této teorie nám pomůže při řešení problému. O úplném Eukleidově algoritmu není v další části druhé kapitoly pojednáno již tak podrobně. Eukleidovým algoritmem dostaneme konečnou posloupnost polynomů. Každým dvěma po sobě jdoucím polynomům odpovídá Sylvestrova matice a úplný Eukleidův algoritmus v maticové formulaci představuje transformaci všech výše uvedených Sylvestrových matic na dolní trojúhelníkový tvar. Pokud tyto transformace provedeme za sebou a pojmeme jako jednu výslednou transformaci, pak při troše fantazie každý uvidí, že úplný Eukleidův algoritmus odpovídá regulární transformaci původní Sylvestrovy matice na dolní trojúhelníkovou matici, a dále uvidí, jak lze z tohoto tvaru vyčíst koeficienty největšího společného dělitele.

Ve třetí kapitole se zmíníme o algoritmu STLN, který pro zadané a obecně i nesoudělné polynomy f a g, pro přirozené číslo k a zadanou toleranci tol počítá polynomy δf a δg tak, aby $\max(\|\delta f\|/\|f\|, \|\delta g\|/\|g\|) \leqq tol$ a aby stupeň největšího společného dělitele polynomů $f+\delta f$ a $g+\delta g$ byl roven alespoň k, pokud takové polynomy existují. Symbol $\|h\|$ značí eukleidovskou normu vektoru sestaveného z koeficientů polynomu h. Celý postup vyústí v úlohu na použití metody nejmenších čtverců [2], [7], [13], [14]. Ukážeme jednu numerickou realizaci na počítači. Na tomto místě již uvidíme výhody maticové formulace Eukleidova algoritmu.

V poslední kapitole připojíme krátký historický přehled podle Barnettovy knihy [1]. Budeme pracovat s polynomy s komplexními koeficienty, to jest s prvky v $\mathbb{C}$. Symbol $\mathbb{C}^{p\times q}$ bude značit množinu všech $p \times q$ komplexních matic.

Eukleidův algoritmus a transformace Sylvestrovy matice

Nechť symbol $\deg(f)$ značí stupeň polynomu f a $\mathrm{GCD}(f,g)$ největšího společného dělitele polynomů f a g. Nechť

$$f(x) = a_0x^m + a_1x^{m-1} + \cdots + a_{m-1}x + a_m, \tag{1}$$
$$g(x) = b_0x^n + b_1x^{n-1} + \cdots + b_{n-1}x + b_n, \tag{2}$$

přičemž předpokládáme bez újmy na obecnosti, že $m \geqq n,\ a_0a_m \neq 0,\ b_0b_n \neq 0$.
Nejprve zavedeme pojem Sylvestrovy matice pro polynomy (1) a (2). Sylvestrova matice, kterou označíme $S(f,g) \in \mathbb{C}^{(m+n)\times(m+n)}$, je matice

$$S(f,g) = \underbrace{\begin{bmatrix} a_0 & & & & b_0 & & & \\ a_1 & a_0 & & & b_1 & b_0 & & \\ \cdot & a_1 & \cdot & & \cdot & b_1 & \cdot & \\ \cdot & \cdot & \cdot & a_0 & \cdot & \cdot & \cdot & b_0 \\ a_m & \cdot & \cdot & a_1 & b_n & \cdot & \cdot & b_1 \\ & a_m & \cdot & \cdot & & b_n & \cdot & \cdot \\ & & \cdot & \cdot & & & \cdot & \cdot \\ & & & a_m & & & & b_n \end{bmatrix}}_{n \text{ sloupců}} \underbrace{\qquad\qquad}_{m \text{ sloupců}} . \tag{3}$$

Připomeňme ještě, že vektor e_i značí i-tý sloupec jednotkové matice I, a matice

$$E_{i,j}(\sigma) = I - \sigma e_i e_j^\top, \quad \sigma \in \mathbb{C}, \tag{4}$$

známá z Gaussovy eliminace, je v literatuře často uváděna pod názvem elementární trojúhelníková matice. Dimenzi u jednotkové matice I budeme vyznačovat jenom v případě, že by mohlo dojít k omylu. Totéž bude platit při zápisu vektorů.
Pro pohodlí položíme $f_0 := f$ a $f_1 := g$. Eukleidův algoritmus je definován rekurencí

$$f_j(x) = f_{j+1}(x)q_j(x) + f_{j+2}(x), \quad j = 0, 1, 2, \ldots, \tag{5}$$

kde $\deg(f_{j+2}) < \deg(f_{j+1})$. Je-li $f_{t+1} \equiv 0$ a $f_j \neq 0$ pro všechna $j \leqq t$, pak $f_t = \mathrm{GCD}(f_0, f_1)$.

Nejprve provedeme první krok Eukleidova algoritmu, tj. proces (5) pro $j = 0$, a paralelně s tím odpovídající transformace Sylvestrovy matice. První krok se může sestávat z více dělení. Pro lepší pochopení celého postupu uvažujme stupně polynomů f a g rovné $m = 4$ a $n = 2$. Pro snadnější zápis algoritmu zaveďme ještě $h_4(x) := f_0(x)$ a $\sigma_1 := a_0/b_0$. Eukleidův algoritmus začneme dělením, které popíšeme podrobně:

$$\begin{aligned}
&\overbrace{\overbrace{(a_0x^4 + a_1x^3 + a_2x^2 + a_3x + a_4)}^{f_0(x)=h_4(x)} - \overbrace{(b_0x^2 + b_1x + b_2)}^{f_1(x)}\overbrace{(a_0/b_0)x^2}^{\sigma_1 x^2}}^{} \\
&= \underbrace{0 + \underbrace{\left(a_1 - \frac{a_0b_1}{b_0}\right)}_{a_1^{(1)}}x^3 + \underbrace{\left(a_2 - \frac{a_0b_2}{b_0}\right)}_{a_2^{(1)}}x^2 + \underbrace{a_3}_{a_3^{(1)}}x + \underbrace{a_4}_{a_4^{(1)}}}_{a_1^{(1)}x^3 + a_2^{(1)}x^2 + a_3^{(1)}x + a_4^{(1)}}.
\end{aligned} \tag{6}$$

Horní index $^{(1)}$ u koeficientů polynomů (6) označuje koeficienty po prvním dělení. Nechť přirozené číslo i_1 značí dolní index u prvního nenulového členu v posloupnosti $a_1^{(1)}$, $a_2^{(1)}$, $a_3^{(1)}$, $a_4^{(1)}$. V našem konkrétním příkladě předpokládejme pro ilustraci, že $i_1 = 1$, takže stupeň m_1 výsledného polynomu h_3 po prvním dělení se rovná $m - i_1 = 3$. Polynom h_3 je napsán pod vztahem (6).

Ukážeme si nyní, jak získáme koeficienty polynomu h_3 z transformované Sylvestrovy matice. Sylvestrova matice $S(f_0, f_1)$ pro polynomy f_0 a f_1 má tvar

$$S(f_0, f_1) = \begin{bmatrix}
a_0 & & b_0 & & & \\
a_1 & a_0 & b_1 & b_0 & & \\
a_2 & a_1 & b_2 & b_1 & b_0 & \\
a_3 & a_2 & & b_2 & b_1 & b_0 \\
a_4 & a_3 & & & b_2 & b_1 \\
 & a_4 & & & & b_2
\end{bmatrix}. \tag{7}$$

Odečtěme třetí, resp. čtvrtý sloupec matice $S(f_0, f_1)$ vynásobený číslem $\sigma_1 = a_0/b_0$, od prvního, resp. druhého sloupce, což se realizuje vynásobením matice $S(f_0, f_1)$ maticemi $E_{3,1}(\sigma_1)$ a $E_{4,2}(\sigma_1)$ zprava. Zapíšeme násobení ve tvaru

$$S^{(1)}(f_0, f_1) := S(f_0, f_1)E_{3,1}(\sigma_1)E_{4,2}(\sigma_1).$$

Zápis matice $S^{(1)}(f_0, f_1)$ obdržíme ze (7) tak, že v prvních dvou sloupcích dosadíme na pozice prvků a_0, a_1, a_2, a_3, a_4 prvky $0, a_1^{(1)}$, $a_2^{(1)}$, $a_3^{(1)}$, $a_4^{(1)}$. Ostatní prvky zůstanou beze změny. Vidíme, že právě v prvních dvou sloupcích matice $S^{(1)}(f_0, f_1)$ jsou koeficienty polynomu h_3. Vraťme se k zápisu (6) a opišme první dělení v prvním kroku ve tvaru

$$h_4(x) - f_1(x)(\sigma_1 x^2) = h_3(x). \tag{8}$$

Protože stupeň polynomu h_3 je větší než stupeň f_1, který se rovná $n = 2$, pokračujeme v prvním kroku algoritmu a dělíme polynom h_3 polynomem f_1. Položme $\sigma_2 = a_1^{(1)}/b_0$. Mocnina x u koeficientu σ_2 je zřejmě rovna $1 = \deg(h_3) - \deg(f_1)$. Napišme si dělení podrobně:

$$\begin{aligned} &\underbrace{(a_1^{(1)}x^3 + a_2^{(1)}x^2 + a_3^{(1)}x + a_4^{(1)}}_{h_3(x)} - \underbrace{(b_0x^2 + b_1x + b_2)}_{f_1(x)}\underbrace{(a_1^{(1)}/b_0)x}_{\sigma_2 x} \\ &= \underbrace{0 + \underbrace{\left(a_2^{(1)} - \frac{a_1^{(1)}b_1}{b_0}\right)}_{a_2^{(2)}}x^2 + \underbrace{\left(a_3^{(1)} - \frac{a_1^{(1)}b_2}{b_0}\right)}_{a_3^{(2)}}x + \underbrace{a_4^{(1)}}_{a_4^{(2)}}}_{a_2^{(2)}x^2 + a_3^{(2)}x + a_4^{(2)}}. \end{aligned} \tag{9}$$

Nechť $a_{i_2}^{(2)}$ je první nenulový prvek v posloupnosti $a_2^{(2)}, a_3^{(2)}, a_4^{(2)}$. Začněme od konce. Je-li $a_2^{(2)} = 0$ a $a_3^{(2)} \neq 0$, takže dolní index i_2 prvního nenulového koeficientu se rovná 3, pak polynom (9) má tvar $a_3^{(2)}x + a_4^{(2)} =: h_1(x)$. Jeho stupeň je menší než stupeň polynomu f_1, první krok Eukleidova algoritmu končí a ve shodě s označením ve vztahu (5) položíme $f_2 := h_1$. Jestliže $a_2^{(2)} \neq 0$, takže $i_2 = 2$, položíme $h_2(x) = a_2^{(2)}x^2 + a_3^{(2)}x + a_4^{(2)}$. Druhé dělení v prvním kroku Eukleidova algoritmu můžeme pro oba případy napsat ve tvaru

$$h_3(x) - f_1(x)(\sigma_2 x) = h_{4-i_2}(x). \tag{10}$$

Koeficienty polynomu h_{4-i_2} obdržíme následující transformací matice $S_1(f_0, f_1)$: odečteme čtvrtý, resp. pátý sloupec matice $S^{(1)}(f_0, f_1)$, vynásobený číslem $\sigma_2 = a_1^{(1)}/b_0$, od prvního, resp. druhého sloupce. To se realizuje tak, že se matice $S^{(1)}(f_0, f_1)$ vynásobí zprava maticemi $E_{4,1}(\sigma_2)$ a $E_{5,2}(\sigma_2)$. Obdržíme matici

$$S^{(2)}(f_0, f_1) := S^{(1)}(f_0, f_1)E_{4,1}(\sigma_2)E_{5,2}(\sigma_2).$$

Její tvar získáme z matice $S(f_0, f_1)$ tak, že v prvních dvou sloupcích dosadíme na pozice prvků a_0, a_1, a_2, a_3, a_4 prvky $0, 0, a_2^{(2)}, a_3^{(2)}, a_4^{(2)}$. V prvních dvou sloupcích máme tudíž koeficienty polynomu h_2. Pro demonstraci předpokládejme ještě, že $a_2^{(2)} \neq 0$. První krok Eukleidova algoritmu pokračuje tedy ještě dělením polynomu h_2 polynomem f_1, to jest

$$h_2(x) - f_1(x)\overbrace{(a_2^{(2)}/b_0)}^{\sigma_3} = \underbrace{a_3^{(3)}x + a_4^{(3)}}_{h_1(x)}. \tag{11}$$

Dokončeme výpočet koeficientů polynomu h_1 pomocí transformací Sylvestrovy matice. Koeficienty polynomu h_1 obdržíme následovně: odečteme pátý sloupec matice $S^{(2)}(f_0, f_1)$, násobený $\sigma_3 = a_2^{(2)}/b_0$, od prvního a šestý sloupec, násobený σ_3, od druhého. Tato úprava je realizována násobením matice $S^{(2)}(f_0, f_1)$ zprava maticí $E_{5,1}(\sigma_3)E_{6,2}(\sigma_3)$. Maticově si zapišme tento krok ve tvaru

$$S^{(3)}(f_0, f_1) := S^{(2)}(f_0, f_1)E_{5,1}(\sigma_3)E_{6,2}(\sigma_3).$$

Tvar matice $S^{(3)}(f_0, f_1)$ obdržíme z matice $S(f_0, f_1)$ tak, že v prvních dvou sloupcích dosadíme na pozice prvků a_0, a_1, a_2, a_3, a_4 prvky $0, 0, 0, a_3^{(3)}, a_4^{(3)}$, a je vidět, že první dva sloupce obsahují koeficienty polynomu h_1, což je polynom f_2 z rekurence (5), který získáme po prvním kroku Eukleidova algoritmu.

Shrneme-li první krok Eukleidova algoritmu, to jest vzorce (8), (10), (11), dostaneme

$$\begin{aligned}(\underbrace{a_0x^4 + a_1x^3 + a_2x^2 + a_3x + a_4}_{h_4(x)=f_0(x)=f(x)} \\ = \underbrace{\left(b_0x^2 + b_1x + b_2\right)}_{f_1(x)=g(x)} \underbrace{\left(\sigma_1x^2 + \sigma_2x + \sigma_3\right)}_{q_0(x)} + \underbrace{\left(a_3^{(3)}x + a_4^{(3)}\right)}_{h_1(x)=f_2(x)}.\end{aligned} \tag{12}$$

Je-li matice Q rovna součinu všech výše uvedených šesti elementárních trojúhelníkových matic, přičemž je jedno v jakém pořadí, protože jsou navzájem komutativní, a je-li $P \in \mathbb{R}^{6\times 6}$ permutační matice ve tvaru

$$P = \begin{bmatrix} e_3 & e_4 & e_5 & e_6 & e_1 & e_2 \end{bmatrix},$$

pak

$$S(f_0, f_1)QP = \left[\begin{array}{ccc|ccc} b_0 & & & & & \\ b_1 & b_0 & & & & \\ b_2 & b_1 & b_0 & & & \\ - & - & - & + & - & - & - \\ & b_2 & b_1 & & b_0 & a_3^{(3)} & \\ & & b_2 & & b_1 & a_4^{(3)} & a_3^{(3)} \\ & & & & b_2 & & a_4^{(3)} \end{array}\right], \tag{13}$$

přičemž blok na místě (2,2) je Sylvestrova matice pro polynomy $f_1 = g$ a f_2. Na tu opět aplikujeme celý výše uvedený postup. Tím provádíme druhý krok Eukleidova algoritmu, to jest rekurenci (5) pro $j = 1$. A tento postup opakujeme tak dlouho, dokud nedostaneme při dělení nulový zbytek. Podrobnější výklad s dalšími vlastnostmi je možné najít v článku [15] nebo v knize [1]. Všimněme si ještě ze vztahu (13), že pro hodnost matice $S(f_0, f_1)$ platí rovnost

$$\mathrm{h}(S(f_0, f_1)) = \deg\,(f_0) - \deg\,(f_2) + \mathrm{h}(S(f_1, f_2)). \tag{14}$$

Dříve, než popíšeme úplný Eukleidův algoritmus, učiňme jednu poznámku. Abychom čtenáři co nejvíce přiblížili souvislost mezi Eukleidovým algoritmem a manipulací se

Sylvestrovou maticí, násobili jsme Sylvestrovu matici zprava elementárními trojúhelníkovými maticemi E_{ij}, což jinak řečeno znamená, že jsme prováděli Gaussovu eliminaci. Poznamenejme ještě, že místo násobení maticemi E_{ij} je možné provést i jiné elementární transformace. Na příklad k nulování prvku a_0 v demonstrační matici $S(f_0, f_1) \in \mathbb{C}^{6\times 6}$ můžeme zprava násobit 6×6 maticí

$$G_1(c,s) = \begin{bmatrix} c & 0 & 0 & . & \\ 0 & c & 0 & . & \\ s & 0 & 1 & . & \\ 0 & s & 0 & 1 & \\ . & . & & & I_2 \end{bmatrix}, \quad \text{kde} \quad c = \frac{b_0}{\sqrt{a_0^2 + b_0^2}}, \quad s = -\frac{a_0}{\sqrt{a_0^2 + b_0^2}},$$

a analogicky při eliminaci dalších koeficientů. Tato druhá volba dává numericky mnohem přesnější výsledky než násobení elementárními trojúhelníkovými maticemi, které jsme zvolili pro ilustraci, protože přesně kopírují Eukleidův algoritmus.

Uvažujme nyní submatici na místě (2,2) v matici (13). Je-li $a_3^{(3)} = a_4^{(3)} = 0$, pak polynom f_1 dělí polynom f_0. V prvním sloupci uvedené submatice jsou koeficienty polynomu f_1, což je největší společný dělitel polynomů f_0 a f_1 a zbývající dva sloupce jsou nulové.

Je-li $a_3^{(3)} = 0$ a $a_4^{(3)} \neq 0$, je matice (13) dolní trojúhelníková s nenulovou diagonálou a polynomy f_0 a f_1 jsou nesoudělné, protože ve zbytku Eukleidova algoritmu vyšla nenulová konstanta $a_4^{(3)} = \mathrm{GCD}(f_0, f_1)$, což je polynom f_2 a při dělení polynomu f_1 polynomem f_2 vyjde nulový zbytek. V tomto případě není blokové rozdělení matice (13) takové, jak je nakresleno, ale blok na místě (2,2) je Sylvestrova matice $S(f_1, f_2) = \mathrm{diag}\,[a_4^{(3)}, a_4^{(3)}]$ velikosti 2×2. Ostatní bloky se upraví podle bloku (2,2).

Je-li $a_3^{(3)} \neq 0$ a $a_3^{(4)} \neq 0$, pak submatice na místě (2,2) je Sylvestrova matice $S(f_1, f_2)$ velikosti 3×3, se kterou budeme provádět transformace analogické těm, které jsme aplikovali na matici $S(f_0, f_1)$. Čtenář si nyní snadno domyslí, že při větší matici bude proces transformace na dolní trojúhelníkový tvar pokračovat dále a že analogicky nám vyjdou Sylvestrovy matice $S(f_2, f_3)$, $S(f_3, f_4)$, ... , dokud nedostaneme regulární dolní trojúhelníkovou matici nebo nulové sloupce. Při troše fantazie nebude nyní těžké si představit zobecnění konkrétního příkladu na polynomy stupňů m a n.

Vraťme se k rekurenci (5):

Nechť $f_{t+1} = 0$ a $f_j \neq 0$ pro všechna $j \leqq t$, takže f_t je největším společným dělitelem polynomů f_0 a f_1. Složením všech transformací aplikovaných postupně na matice $S(f_0, f_1)$, $S(f_1, f_2)$, $S(f_2, f_3)$, ... podle výše popsaného postupu dostaneme výslednou transformaci, kterou označme T a kterou se násobí matice $S(f_0, f_1)$. Matice T je součinem elementárních dolních trojúhelníkových a permutačních matic konstruovaných tak, jak bylo ukázáno na konkrétním příkladě. Ten jsme uvedli podrobně pro první krok Eukleidova algoritmu, a tedy nebude obtížné pochopit následující tvrzení. Pro pohodlí si označme na chvíli $n_j = \deg(f_j)$ pro $j \in \{0, 1, \ldots, t+1\}$.

Existuje regulární matice T řádu $n_0 + n_1$ taková, že matice $S(f_0, f_1)T$ má tvar

$$S(f_0, f_1)T = \left[\begin{array}{c|c} L_{1,1} & 0 \\ \hline L_{2,1} & L_{2,2} \end{array}\right],$$

kde $L_{1,1}$ je čtvercová dolní trojúhelníková matice s nenulovými diagonálními prvky. Matice $L_{2,2}$, která je výsledkem transformace Sylvestrovy matice $S(f_{t-1}, f_t)$ řádu $(n_{t-1} + n_t)$ na čtvercovou dolní trojúhelníkovou matici, má následující tvar:

(a) Je-li $n_t > 0$, pak prvních n_{t-1} prvků na diagonále matice $L_{2,2}$ je nenulových. Každý z prvních n_{t-1} sloupců obsahuje $n_t + 1$ koeficientů polynomu f_t, což je největší společný dělitel polynomů f_0 a f_1. Dalších n_t sloupců matice $L_{2,2}$ je nulových. Prvních n_{t-1} sloupců matice $L_{2,2}$ je totožných s posledními n_{t-1} sloupci matice $S(f_{t-1}, f_t)$.

(b) Je-li $n_t = 0$, pak f_t je nenulová konstanta a $L_{2,2} = f_t I_{n_{t-1}}$ (diagonální matice s hodnotou f_t na diagonále). V tomto případě je $\mathrm{h}(S(f_0, f_1)) = n_0 + n_1$ a polynomy f_0 a f_1 jsou nesoudělné.

(c) Označme $k := n_t$. Napišme si rovnost (14) postupně pro matice $S(f_j, f_{j+1})$, $j = 1, 2, \ldots, t-2$, sečtěme všechny rovnosti a dostaneme

$$\mathrm{h}(S(f_0, f_1)) = n_0 + n_1 - n_t = n_0 + n_1 - k. \tag{15}$$

Vraťme se k původnímu označení, tj. pišme pro polynomy f a g místo f_0 a f_1 a pro stupně polynomů m a n místo n_0 a n_1.
Z právě provedených úvah snadno plyne, že

$$\mathrm{h}(S(f, g)) = m + n - k \quad \Longleftrightarrow \quad \deg(\mathrm{GCD}(f, g)) = k. \tag{16}$$

Jiný důkaz vztahu (16) je možné najít v [8]. Na závěr kapitoly uveďme ještě jednu úvahu pro k-tý Sylvestrův subresultant, který označíme S_k a který se vytvoří z matice $S(f, g)$ škrtnutím $(k-1)$ posledních řádků, $(k-1)$ posledních sloupců s koeficienty, které patří polynomu f, a $(k-1)$ posledních sloupců s koeficienty, které patří polynomu g.

Nechť k je stupeň největšího společného dělitele polynomů (1) a (2). Pak existují polynomy p_{m_k} a q_{n_k} stupňů $m-k$ a $n-k$ tak, že

$$f(x) q_{n_k}(x) - g(x) p_{m_k}(x) = 0. \tag{17}$$

Položíme-li

$$\begin{aligned} p_{m_k}(x) &= p_0 x^{m-k} + p_1 x^{m-k-1} + \ldots + p_{m-k}, \\ q_{n_k}(x) &= q_0 x^{n-k} + q_1 x^{n-k-1} + \ldots + q_{n-k}, \\ v &= [\, q_0, \, q_1, \, \ldots, \, q_{n-k}, \, -p_0, \, -p_1, \, \ldots, \, -p_{m-k}]^T, \end{aligned}$$

pak rovnici (17) můžeme přepsat na homogenní soustavu

$$S_k v = 0, \tag{18}$$

jejíž netriviální řešení tvoří prostor dimenze 1. Protože $S_k \in \mathbb{C}^{(m+n-k+1)\times(m+n-2k+2)}$, je $\mathrm{h}(S_k) = m + n - 2k + 1$. Dále podle (15) je $\mathrm{h}(S(f, g)) = m + n - k$. A tady se opět vytvořila bohatá teorie v literatuře nazývaná low rank approximation, která numericky

vyústila v následující postup: najít poruchu pro prvky Sylvestrovy matice $S(f,g)$ v zadané toleranci tak, aby porušená matice měla co nejmenší hodnost, neboli aby stupeň největšího společného dělitele polynomů f a g byl co největší. O tom se můžeme dočíst na příklad v [7], [14] a v dalších článcích. V tomto a v předchozích odstavcích jsme si připomněli některé známé věci z algebry. Pro pochopení následujícího výkladu to však bylo nutné.

Jednoduchá úloha pro nepřesně zadané polynomy

Nyní přicházíme k jedné praktické úloze. Chceme, aby dva polynomy, jejichž koeficienty porušíme v zadané toleranci (na příklad 10^{-7}) měly největšího společného dělitele co největšího stupně. Přejděme k přesné matematické formulaci. Nechť je dáno přirozené číslo k, $1 \leqq k \leqq \min(m,n)$ a tolerance tol. Chceme spočítat poruchy δf a δg polynomů f a g,

$$\begin{aligned}\delta f(x) =& \delta a_0 x^m + \delta a_1 x^{m-1} + \cdots + \delta a_{m-1} x + \delta a_m,\\ \delta g(x) =& \delta b_0 x^n + \delta b_1 x^{n-1} + \cdots + \delta b_{n-1} x + \delta b_n\end{aligned}$$

tak, aby platilo

$$\deg\left(\mathrm{GCD}(f+\delta f, g+\delta g)\right) \geqq k \quad \text{a} \tag{19}$$

$$\max(\|\delta f\|/\|f\|, \|\delta g\|/\|g\|) \leqq tol. \tag{20}$$

V tomto odstavci označíme symboly f a g polynomy s nepřesně zadanými koeficienty, které si pro naši demonstraci vyrobíme. Označme přesné polynomy $\hat{f}$ a $\hat{g}$:

$$\begin{aligned}\hat{f}(x) &= \hat{a}_0 x^m + \hat{a}_1 x^{m-1} + \cdots + \hat{a}_{m-1} x + \hat{a}_m,\\ \hat{g}(x) &= \hat{b}_0 x^n + \hat{b}_1 x^{n-1} + \cdots + \hat{b}_{n-1} x + \hat{b}_n.\end{aligned}$$

Nechť $c_f \in \mathbb{R}^{m+1}$ a $c_g \in \mathbb{R}^{n+1}$ jsou vektory, jejichž složky jsou náhodná čísla z intervalu $[-1,\ 1]$. Nechť písmeno $\varepsilon > 0$ značí velikost poruchy. Pro demonstraci si definujme koeficienty porušených polynomů následujícím způsobem:

$$\delta\hat{a}_i = \varepsilon \frac{\|\hat{f}(x)\|}{\|c_f\|} c_{f,\,i}\,, \quad \delta\hat{b}_i = \varepsilon \frac{\|\hat{g}(x)\|}{\|c_g\|} c_{g,\,i}\,,$$

kde $c_{f,\,i}$ značí i-tou složku vektoru c_f, $i = 0, 1, \ldots, m$, a analogický význam má symbol $c_{g,\,i}$. Definujme tedy porušené polynomy (a ty jsou dány, proto je značíme f a g) následovně:

$$f(x) = \sum_{i=0}^{m} (\hat{a}_i + \delta\hat{a}_i) x^{m-i} =: \sum_{i=0}^{m} a_i x^{m-i}, \qquad g(x) = \sum_{i=0}^{n} (\hat{b}_i + \delta\hat{b}_i) x^{n-i} =: \sum_{i=0}^{n} b_i x^{n-i}.$$

A nyní uvedene jednoduchý příklad, který počítal student Matematicko-fyzikální fakulty Ján Eliaš ve své bakalářské práci, která byla vysoko hodnocena (viz [3]). Zvolme $\varepsilon = 10^{-6}$. Přesné polynomy nechť jsou následující:

$$\hat{f}(x) = (x - 1.2)^4(x + 2)^5(x - 0.5)^4,$$
$$\hat{g}(x) = (x - 1.4)^2(x + 2)^3(x - 0.5)^4.$$

Je ihned vidět, že

$$\mathrm{GCD}(\hat{f}, \hat{g}) = (x + 2)^3(x - 0.5)^4 =$$
$$= x^7 + 4x^6 + 1.5x^5 + 7.5x^4 - 0.9375x^3 + 6.375x^2 - 3.25x + 0.5.$$

Podle uvedeného postupu pro upravené polynomy f a g student na počítači stanovil hodnost Sylvestrovy matice $S(f, g)$, která se rovnala $m + n$, což znamená podle toho, co bylo řečeno, že stupeň největšího společného dělitele je roven nule, a tedy polynomy f a g jsou nesoudělné. Podle toho, jak jsme pro demonstraci zvolili ε, se domníváme, že v epsilonovém okolí polynomů f a g najdeme polynomy $\widetilde{f}$ a $\widetilde{g}$ tak, že koeficienty polynomů $\mathrm{GCD}(\hat{f}, \hat{g})$ a $\mathrm{GCD}(\widetilde{f}, \widetilde{g})$ se liší na příklad na čtvrtém nebo na některém z dalších desetinných míst. Pro poslední polynom se používá názvu přibližný největší společný dělitel. Pro výpočet $\widetilde{f}$ a $\widetilde{g}$ se používá metoda STLN (structure total least norm). Pro informaci naznačme myšlenku algoritmu, podle kterého se počítá. Podrobně je vše popsáno na příklad v [7] nebo [9].

Vraťme se na začátek tohoto odstavce a formulujme si následující jednodušší úlohu. Chceme spočítat poruchy δf a δg polynomů f a g tak, aby v nerovnosti (19) platila rovnost. Nechť $S_k(f, g)$ resp. $S_k(\delta f, \delta g)$ značí k-tý Sylvestrův subresultant sestrojený pro polynomy f a g, resp. δf a δg. Položme

$$S_k(f, g) = [u_k, A_k], \quad S_k(\delta f, \delta g) = [h_k, E_k],$$

kde u_k a h_k značí první sloupce již uvedených Sylvestrových subresultantů. V článku [7] se ukazuje, že podmínka (19) je ekvivalentní podmínce, že soustava

$$(A_k + E_k)y = u_k + h_k \tag{21}$$

má právě jedno netriviální řešení. Podmínky (20) a (21) se složitějšími úpravami převedou na úlohu nejmenších čtverců s lineárními omezeními. Řešení nejmenších čtverců je podrobně popsáno v knize [2] v kapitole 5. K numerickému řešení byly použity metody navržené v [13].

Napišme si ještě schéma práce a numerické hodnoty koeficientů právě uvedených polynomů včetně koeficientů přibližného největšího společného dělitele. Postup je následující:

$$\left\{ \begin{matrix} \hat{f}(x) \\ \hat{g}(x) \end{matrix} \right\} \overset{\text{porucha}}{- - - - \rightarrow} \left\{ \begin{matrix} f(x) \\ g(x) \end{matrix} \right\} \overset{\text{STLN}}{- - - - \rightarrow} \left\{ \begin{matrix} \widetilde{f}(x) \\ \widetilde{g}(x) \end{matrix} \right\}.$$

Nechť $m = 13$, $n = 9$ a $k = 7$. Pro kontrolu se na počítači stanovila hodnost Sylvestrovy matice pro přesné polynomy. Vyšlo $\mathrm{h}(S(\tilde{f}, \tilde{g})) = 15$, což je $m + n - k$ a

$\mathrm{h}(S(f,g)) = 22$. V tabulkách otištěných z grafických důvodů na konci článku uvádíme příslušné koeficienty.
V tomto jednoduchém příkladu dostáváme dobrý souhlas s teorií.

V praxi ovšem polynomy $\widetilde{f}(x)$ a $\widetilde{g}(x)$ neznáme a výpočet se proto soustřeďuje na to, abychom užitím teoreticky odvozených postupů spočetli přibližného největšího společného dělitele co nejvyššího stupně při malých poruchách, které se mohou pohybovat až na hranici strojové přesnosti.

Několik poznámek na závěr

Literatura na výpočet největšího společného dělitele je bohatá, byly napsány stovky publikací. Díky internetu a domovským stránkám řady autorů je možné se dnes docela dobře seznámit s touto problematikou, i když ne vždy také proto, že řada časopisů s touto tematikou v našich knihovnách není a odpovídající stránky na internetu jsou nedostupné. Rád bych ještě poznamenal, že kromě polynomů v jedné proměnné se podobné úlohy řeší i pro polynomy více proměnných. To bychom ovšem zašli daleko za rámec tohoto článku.

Na závěr bych rád připojil pár poznámek z historie. Sylvestrova matice se původně uvažovala v transponovaném tvaru, to jest $S(f,g)^\top$ a transformovala se tudíž na horní trojúhelníkový tvar [1], [8]. Pro počítání koeficientů $\mathrm{GCD}(f,g)$ se používal QR-rozklad. Zde se již jasně projeví výhodnost maticové formulace Eukleidova algorimu. Použitím klasického zápisu Eukleidova algoritmu bychom těžko prováděli QR-rozklad. Připomeňme, že QR-rozklad nezachovává strukturu Sylvestrovy matice. V posledním nenulovém řádku horní trojúhelníkové matice se pak obdržely koeficienty největšího společného dělitele. Podrobně a s mnoha příklady se o tom může čtenář dočíst v článku [16] nebo v knize [1]. Poznamenejme, že všechny uvažované transformace jsou regulární a nemění tudíž hodnost matice. Zajímavé je, jak velkou roli má v důkazech Frobeniova matice. V angličtině se užívá názvu companion matrix. Připomeňme jenom několik zajímavostí. Frobeniova matice svázaná s polynomem (1), u kterého se předpokládá, že $a_0 = 1$, je matice tvaru

$$F = \begin{bmatrix} 0 & 1 & 0 & . & . & . & 0 \\ 0 & 0 & 1 & 0 & . & . & 0 \\ 0 & 0 & 0 & 1 & . & & 0 \\ . & . & . & . & . & . & 0 \\ . & . & . & . & . & . & 0 \\ . & . & . & . & . & . & 1 \\ -a_m & -a_{m-1} & -a_{m-2} & & & -a_2 & -a_1 \end{bmatrix}.$$

Pod pojmem Frobeniova matice se také často uvažuje matice transponovaná. Nechť podle (2) je polynom $g(x) = b_0x^n + b_1x^{n-1} + \cdots + b_{n-1}x + b_n$. Spočítáme $M = J_m g(F) J_m$, kde $J_m = [e_m, e_{m-1}, \ldots, e_1]$ je matice $m \times m$. Provedeme-li QR-rozklad matice M, pak v posledním nenulovém řádku spočítané horní trojúhelníkové matice dostaneme koeficienty největšího společného dělitele polynomů f a g. Toto vše

s podrobně propracovanými důkazy a velkým množstvím příkladů nalezneme v [1]. A opět máme jiný postup z lineární algebry na výpočet největšího společného dělitele. Vidíme, že i v této oblasti spojené s Eukleidovým algoritmem je hodně zajímavých otázek i aplikací. Těmto otázkám se u nás věnuje malá pozornost a mnohdy se bohužel přecházejí poznámkou, že Eukleidův algoritmus je nestabilní v tom smyslu, jak bylo zmíněno v úvodu. Tento článek má za cíl ukázat další problémy, které s Eukleidovým algoritmem souvisí, jsou zajímavé a jejichž řešení potřebuje fundamentální znalosti z lineární algebry. Dále jsme se snažili ukázat, že maticová formulace je nejenom elegantnější, ale rozšiřuje celou teorii právě o možnost použít prostředků lineární algebry zejména v případě porušených koeficientů polynomů. V posledním případě nám není známo, jak by se klasického zápisu Eukleidova algoritmu dalo vůbec použít.

Na závěr tohoto pojednání uvádíme tabulky koeficientů polynomů z jednoduchého příkladu ve třetí kapitole.

	$f(x)$	$g(x)$
x^{13}	1	
x^{12}	3.20025	
x^{11}	−8.26093	
x^{10}	−26.49540	
x^9	38.00476	1
x^8	85.59627	1.199981
x^7	−121.21627	−7.739988
x^6	−109.89824	−3.859967
x^5	223.97294	23.002372
x^4	−17.51887	−5.699975
x^3	−156.15339	−22.937378
x^2	120.28351	22.094884
x^1	−36.63814	−7.769948
x^0	4.14757	0.979989

	$\widetilde{f}(x)$	$\widetilde{g}(x)$
x^{13}	1	
x^{12}	3.19998	
x^{11}	−8.26007	
x^{10}	−26.49212	
x^9	38.00016	1
x^8	85.58606	1.199982
x^7	−121.20177	−7.739994
x^6	−109.88508	−3.859969
x^5	223.94605	23.002392
x^4	−17.51684	−5.699980
x^3	−156.13476	−22.937396
x^2	120.26907	22.094900
x^1	−36.63364	−7.769959
x^0	4.14719	0.979990

	$GCD(\hat{f},\hat{g})$	$GCD(\widetilde{f},\widetilde{g})$
x^7	1	1
x^6	4	3.999978
x^5	1.5	1.499947
x^4	−7.5	−7.500006
x^3	−0.9375	−0.937463
x^2	6.375	6.375001
x^1	−3.25	−3.250011
x^0	0.5	0.499999

Poděkování. Autor děkuje prof. RNDr. Michalu Křížkovi, DrSc., Bc. Jánovi Eliašovi a neznámemu recenzentovi za pečlivé přečtení celého textu a za řadu cenných připomínek. Dále děkuje dr. Joabovi Winklerovi z university v Sheffieldu v Anglii za připomínky a literaturu, která se týká praktických aplikací teorie vyložené v článku. Všichni výše uvedeni přispěli ke zlepšení kvality článku.

Autor děkuje výzkumnému projektu MSM0021620839 financovanému MŠMT za podporu.

L i t e r a t u r a

[1] Barnett, S.: *Polynomial and linear control system.* Marcel Dekker, New York, USA, 1983.

[2] Björk, Å: *Numerical methods for least squares problems.* SIAM Publications, Philadelphia, 1996.

[3] Eliaš, J.: *Problémy spojené s výpočtem největšího společného dělitele.* Bakalářská práce, 2009.

[4] Farouki, R. T., Rajan, V. T.: *On the numerical condition of algebraic curves and surfaces, 1 Implicit equations.* Computer Aided Geometric Design *5* (1988), 215–252.

[5] Goldman, R. N., Sederberg, T. W., Anderson, D. C.: *Vector elimination: A technique for the implicitization, inversion and intersection of planar parametric rational polynomial curves.* Computer Aided Geometric Design *1* (1984), 327–356.

[6] Kajiya, J. T.: *Ray tracing parametric patches.* Computer Graphics *16* (1982), 245–254.

[7] Kaltofen, E., Yang, Z., Zhi. L.: *Structured low rank approximation of a Sylvester matrix.* Preprint, 2005.

[8] Laidacker, M. A.: *Another theorem relating Sylvester's matrix and the greatest common divisor.* Math. Magazine *42* (May 1969), No. 3, 126–128.

[9] Li, B., Yang, Z., Zhi, L.: *Fast low rank approximation of a Sylvester matrix by structured total least norm.* J. Japan Soc. Symbolic and Algebraic Comp. *11* (2005), 165–174.

[10] Manocha, D., Demmel, J.: *Algorithms for intersecting parametric and algebraic curves, I: Simple intersections.* ACM Transactions on Graphics *13* (1994), No. 1, 73–100.

[11] Sitton, G. A., Burrus, C. S., Fox, J. W., Treitel, S.: *Factoring very high degree polynomials.* IEEE Signal Processing Magazine *20* (2003), No. 6, 27–42.

[12] Stoica, P., Söderström, T.: *Common factor detection and estimation.* Automatica *33* (1977), No. 5, 985–989.

[13] Van Loan, C.: *On the method of weighting for equality-constrained least-squares problems.* SIAM J. Numer. Anal. *22* (1985), No. 5, 851– 864.

[14] Winkler, J. R., Allan, J. D.: *Structured total least norm and approximate GCDs of inexact polynomials.* J. Comput. Appl. Math. *215* (2008), 1–13.

[15] Winkler, J., Zítko, J.: *Some questions associated with the calculation of the GCD of two univariate polynomials.* Proceedings SNA'07-Seminar on Numerical Analysis, Institute of Geonics AS ČR, Ostrava, (2007), 133–137.

[16] Zarowski, C. J., Ma, X., Fairman, F. W.: *QR- factorization method for computing the greatest common divisor of two polynomials with inexact coeffixients.* IEEE Trans. Signal Processing *48* (2000), No. 11, 3042–3051.