NUMERICKÉ METÓDY LINEÁRNEJ ALGEBRY

01. Zobrazenie reálnych čísiel v počítači

Ing. Marek Macák, PhD.

Konzultácie: podľa potreby/dohody online

14. Februar 2024

Predmet numerickej lineárnej algebry

- Numerická lineárna algebra sa zaoberá návrhom algoritmov a analýzou ich vlastností pre problémy spojitej (klasickej) matematiky, pričom sa používajú nástroje lineárnej algebry (L. Trefethen, Oxford, 1997).
- Algoritmus: Návod, ako zo vstupných údajov (množiny čísiel) dospieť k výstupným údajom (iná množina čísiel). Algoritmus môže byť konečný (napr. QR rozklad matice) alebo nekonečný (napr. výpočet vlastných čísiel matice rádu n pre $n > 5$).
- Analýza vlastností: presnosť, stabilita, robustnosť, ...
- Spojitá matematika: Problém je formulovaný pomocou reálnych alebo komplexných premenných. Opakom je diskrétna matematika (napr. teória grafov), kde vystupujú celočíselné premenné.

IEEE Floating-Point Standard

- Každý počítač má ohraničenú pamäť - niektoré reálne čísla nemôžu byť reprezentované v počítači presne.
- IEEE Floating-Point Standard (1985): číslicový systém s pohyblivou rádovou čiarkou je definovaný pomocou usporiadnej štvorice celých čísiel (β,t,L,U), kde: β je základ (báza, radix), t je presnosť, L je dolná hranica exponentu, U je horná hranica exponentu.
- Nenulové normalizované číslo v systéme (β,t,L,U) má tvar:

$$\pm .d_1 d_2 ... d_t \times \beta^e = \pm(d_1\beta^{-1} + d_2\beta^{-2} + ... + d_t\beta^{-t})\beta^e \tag{1}$$

kde $\pm$ je znamienko, $d_1 d_2 ... d_t$ je mantisa a e je exponent ($L \leq e \leq U$).

IEEE Floating-Point Standard

- Normalizované binárne čísla ($\beta = 2$): v počítači sa ukladá mantisa (t bitov), znamienko a exponent s posunom tak, aby exponent bol vždy nezáporný (t.j. pričíta sa $|L| + 1$).
- Príklad:
  - Jednoduchá presnosť (IEEE): $\beta = 2, L = -126, U = 127, t = 24$ (aj so znamienkom). Číslo je reprezentované 32 bitmi (1 bit na znamienko, 8 bitov na exponent, 23 bitov na mantisu)
  - Dvojitá presnosť (IEEE): $\beta = 2, L = -1022, U = 1023, t = 53$ (aj so znamienkom). Potrebujeme 64 bitov (1 bit na znamienko, 11 bitov na exponent, 52 bitov na mantisu)
- IEEE FPS požaduje, aby exponent mal k dispozícii ešte dve skryté hodnoty:
  - L-1: používa sa na kódovanie ± 0 a denormalizovaných čísiel s $d_1 \neq= 1$ (pri $\beta = 2$).
  - U+1= používa sa na kódovanie výsledku, ktorý sa v číselnom systéme počítača nedá zobraziť (tzv. NaN = 'Not a Number', tiež sa označuje ako $\pm$inf).

IEEE Floating-Point Standard

- Označme F_t množinu normalizovaných čísiel s pohyblivou rádovou čiarkou s presnosťou t. Potom F_t nie je uzavretá vzhladom na základné aritmetické operácie: sčítanie, odčítanie, násobenie a delenie.
- Vypočítané číslo nemusí byť v F_t , pretože:
  - Exponent padne mimo interval $[L, U]$: podtečenie (underflow) resp. pretečenie (overflow).
  - Mantisa výsledku obsahuje viac ako t číslic : nutná je nejaká forma zaokrúhlenia výsledku.
- Pretečeniu a podtečeniu sa niekedy dá zabrániť reorganizáciou výpočtu.

Zaokrúhľovacie chyby

- Ak mantisa výsledku aritmetickej operácie obsahuje viac ako t číslic, potom sa dá upraviť do tvaru reprezentovateľného v počítači dvomi spôsobmi:
  - Odtrhnutie ('chopping'): číslice za d_t sa jednoduch 'zahodia'.
  - Zaokrúhlenie ('rounding'): číslica d_t sa zaokrúhli nahor (ak $d_{t+1} \geq \beta/2$) alebo nadol (ak $d_{t+1} \leq \beta/2$), a číslice za d_t sa 'zahodia'.
- Oba postupy aproximujú vypočítané číslo. Aproximácia nesie so sebou vždy chybu. Absolútna chyba aproximácie je $|\tilde{x} - x|$. Relatívna chyba $\frac{|\tilde{x}-x|}{|x|}$ berie do úvahy veľkosť aproximovaného čísla x a dáva informáciu o počte signifikantných číslic v aproximácii.

Zaokrúhľovacie chyby

- Definícia: Hovoríme, že $\tilde{x}$ aproximuje x na s signifikantných číslic, ak je s je najväčšie nezáporné číslo, pre ktoré je relatívna chyba aproximácie

$$\frac{|\tilde{x} - x|}{|x|} < 10^s \tag{2}$$

- Veta 1: Nech $fl(x)$ označuje reprezentáciu čísla x v pohyblivej rádovej čiarke. Potom:

$$\frac{|fl(x) - x|}{|x|} \leq \mu \tag{3}$$

  kde $\mu = 0.5 \times \beta^{1-t}$ pre 'rounding', resp. $\mu = \beta^{1-t}$ pre 'chopping'.

- Definícia: Číslo μ sa nazýva jednotka zaokrúhľovania ('unit roundoff') a $\mu_M = 2\mu$ je presnosť stroja ('machine precission').

Zaokrúhľovacie chyby

- Veta 1 hovorí, že pre dostatočne veľké t je množina F_t ('floating-point numbers') dostatočne "hustá". Napr. pre dvojnásobnú presnosť s $t = 53$ je $\mu = 2^{-53} \approx 1.11 \times 10^{-16}$
- Medzi susednými číslami v F_t absolútna medzera rastie s mocninou 2, ale relatívna medzera nie je nikdy väčšia ako $2^{-52} = 2\mu = \mu_m =\approx 2.22 \times 10^{-16}$
- Rozsah $y \in F_t$ (normalizované): $\beta^{L-1} \leq |y| \leq \beta^{U}(1 - \beta^{-t})$.
- Denormalizované čísla nepatria do F_t (majú menej ako t signifikantných číslic), ale rozširujú F_t.

'Floating-point' aritmetika

- Základný axióm floating-point aritmetiky splňujúcej IEEE Standard: Nech $(+,-,*,/)$ sú základné aritmetické operácie nad reálnymi číslami. Nech označenie *fl* pred niektorou z týchto operácií znamená, že je to binárna operácia nad dvomi číslami v F_t a výsledok je opäť v F_t. Potom pre každé $x, y \in F_t$ existuje $\epsilon, |\epsilon| \leq \mu$ tak, že
$$fl(x \circledast y) = (x \circledast y)(1+\epsilon), \tag{4}$$
kde $\circledast$ je jedna z operácií $(+,-,*,/)$.
- Základný axióm floating-point aritmetiky treba chápať ako požiadavku na počítač, ak chce spĺňať IEEE Standard. Týka sa to hardvéru i firmvéru, ktorý má na starosti výpočet aritmetických operácií. Niekedy (a čoraz častejšie) je podobná relatívna presnosť garantovaná aj pre výpočet druhej odmocniny (čo je unárna operácia):
$$x \in Ft, x > 0 \Rightarrow fl(\sqrt{x}) = \sqrt{x}(1+\epsilon), |\epsilon| \leq \mu. \tag{5}$$

Ochranná číslica ('Guard Digit')

- Relatívna chyba aritmetických operácií má byť najviac μ. Pokiaľ sa však sčítanie alebo odčítanie robí bez tzv. ochrannej číslice, potom výsledok nemusí splňať uvedenú presnosť.
- Definícia: Ochranná číslica je jedna číslica na mieste d_{t+1}, ktorej úlohou je zachytiť najmenšiu mocninu základu pri zarovnaní operandov, ktorá by sa inak stratila.
- Príklad ...