

# Paralelný výpočtový model

- minimalizovať čas realizácie medzi začiatkom a koncom výpočtu využitím viacerých procesorov

Významné faktory:

- počet procesorov
- architektúra (zbernica, prepínaná sieť ...)
- spôsob prístupu k zdieľanej pamäti
- množstvo vymenenej informácie medzi procesmi

$T(n)$  - časová zložitosť sekvenčného výpočtu

$T_2(n)$  - s dvoma procesormi

$T_p(n)$  - časová zložitosť pri využití  $p$  procesorov

zrejme  $T_{i+1}(n) \leq T_i(n)$

$T^*(n)$  - zložitosť optimálneho sekvenčného alg.

zrejme  $T_p(n) \geq T^*(n) / p$

**Cena** (cost)  $C_p(n) = p \cdot T_p(n)$

Optimálny alg. (cost-optimal)  $C_p(n) = O(T(n))$

**Zrýchlenie** (speedup)  $S_p(n) = T(n) / T_p(n)$

❖ Zrejme  $S_p(n) \leq p$  (inak by bola možná rýchlejšia sekvenčná simulácia)

Lineárne zrýchlenie  $S_p(n) \sim p$

superlineárne zrýchlenie ?

**Efektívnosť**  $E_p(n) = T(n) / C_p(n) = S_p(n) / p \leq 1$

optimálny algoritmus - blíži sa k jednotkovej efektívnosti

Problém sčítania  $n$  čísel  $n$  procesormi (pre jednoduchosť  $n$  je mocnina 2)

1. každý párny pripočíta číslo od nepárneho suseda (podľa posledného bitu)
2. každý s predposledným bitom=0 pripočíta výsledok od procesora s opačným bitom

... Po  $\log n$  krokoch bude v procesore 0 súčet

$$S_n(n) = n / \log n \quad E_n(n) = 1 / \log n \quad C_n(n) = n \cdot \log n$$

algoritmus nie je optimálny ...

Problém sčítania  $n$  čísel  $p$  procesormi ( $1 < p < n$ )

1. každý procesor dostane segment s  $n/p$  číslami a vypočíta jeho súčet na  $n/p$  krokov
2. binárnou redukciou  $p$  procesorov spočíta súčet na  $\log p$  krokov

$$T_p(n) = n/p + \log p$$

$$S_p(n) = n.p / (n + p.\log p) \quad E_p(n) = n / (n + p.\log p)$$

$n$  bude optimálne, ak  $n = o(p.\log p)$ , teda  $p = n/\log n$

Každý algoritmus obsahuje sekvenčnú časť ( $f_S$ ) a paralelizovateľnú časť ( $f_P$ )  $f_S + f_P = 1$

Potom  $T_p(n) = f_S + f_P / p$

a tiež  $S_p(n) \leq 1 / (f_S + (1-f_S)/p) \dots \rightarrow 1 / f_S$

**Amdahlov zákon**  $S_p(n) \rightarrow 1 / f_S$

Podobne ak  $T(n) = f_S + f_P = 1$

$S_p(n) = f_S + p \cdot f_P = f_S + p \cdot (1-f_S) = p \cdot (1 - f_S + f_S/p)$

**Gustafsonov zákon**  $S_p(n) \rightarrow p \cdot (1 - f_S)$

- viacero modelov so vzájomným polynomiálnym prevodom
- základom je prístup k spoločnej zdieľanej pamäti (shared memory)
- asynchrónny a synchrónny prístup
- pre asynchrónny - dodatočná synchronizácia (bariéry ...)
- synchrónny - jednoduchší na skúmanie



- PRAM - Parallel Random Access Machine
- neohraničene veľa RAM procesorov  $P_0, P_1, \dots$  s vlastnými registrami a pamäťou
- paralelný synchronný prístup k spoločnej pamäti  $M[0], M[1], \dots$  (s ľubovoľne veľkými číslami)
- vstup je na začiatku v spoločnej pamäti a na konci je v nej výstup
- krok: prečítanie zo spoločnej pamäte, niekoľko lokálnych inštrukcií, zápis do spoločnej pamäte

- ❖ Každý problém, riešiteľný  $p$  procesorovým PRAM na  $t$  krokov, je možné riešiť  $p'$  procesorovým PRAM ( $p' < p$ ) na  $O(t \cdot p'/p)$  krokov v rovnakej pamäti.
- sekvenčný čas je teda dolným odhadom pre počet procesorov PRAM, použitých na riešenie
- ❖ Každý problém, riešiteľný  $p$  procesormi PRAM v pamäti veľkosti  $m$  v čase  $t$ , je možné riešiť v pamäti  $m' < m$ ,  $\max(p, m')$  procesormi v čase  $O(t \cdot m/m')$ .

EREW PRAM - Exclusive Read, Exclusive Write

CREW PRAM - Concurrent Read, Exclusive Write

CRCW PRAM Concurrent Read, Concurrent Write

- Common - pri zápise sa zhodujú
- Arbitrary - uspeje ľubovoľný (nedeterm.)
- Priority - uspeje procesor s najnižším indexom

 tvoria hierarchiu

- Vyhľadávanie prvku  $x$  v zozname  $n$  čísel

- simulácie PRAM modelov navzájom

- PRAM algoritmus zložitosti  $T(n)$  je možné simulovať deterministickým Turingovým strojom v čase  $T^2(n)$ .
- Deterministický Turingov stroj priestorovej zložitosti  $S(n)$  je možné simulovať CREW PRAM v čase  $O(S(n))$  pomocou  $c^{S(n)}$  procesorov.

- NC (Nick Pippenger Class) - dobre paralelizovateľné problémy -
- riešiteľné v polylogaritmickej čase ( $O(\log^k(n))$ ) pomocou polynomiálneho počtu procesorov
- $NC = P$  ??
- hierarchia NC - P-úplné problémy ...

Ďakujem za pozornosť !

