



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA POČÍTAČOVÝCH SYSTÉMŮ

Závěrečná zpráva

Semestrální práce z MI-PAR

Úloha NSR: nejkratší společný řetězec

Tomáš Srna
Petr Hodač

1 Definice problému a popis sekvenčního algoritmu

1.1 Zadání

Vstupní data n = přirozené číslo představující počet řetězců, $n \geq 10$

z = přirozené číslo představující maximální délku řetězců, $z > \log n$

$R = (r_1, \dots, r_n)$ = množina n různých binárních řetězců r_i o délkách $\log n \leq |r_i| \leq z$

Úkol Pro zadané n nejdříve vygenerujte množinu řetězců R délek nejméně $\log n$ a nejvýše z . Délka nejdelšího vygenerovaného řetězce nechť se značí m . Pak zvolte celé číslo k větší než m a současně menší než poloviční součet délek všech řetězců z R . Čili k je z otevřeného intervalu $(m/2, m)$. Nalezněte nejkratší binární řetězec w o délce nejvýše k takový, že každý řetězec z množiny R je jeho podřetězcem. Čili každý řetězec r z R může být doplněn na řetězec $w = w_0 r w_1$, kde w_0 a w_1 jsou binární řetězce (mohou být i prázdné). Takový řetězec w se nazývá nejkratší k -nad řetězec množiny R .

Výstup algoritmu Konstatování, zda existuje k -nad řetězec množiny R a pokud ano, pak výpis nejkratšího z nich a rozpis jeho n rozkladu $w_0 r_i w_1$, kde $r_i \in R$.

1.2 Popis

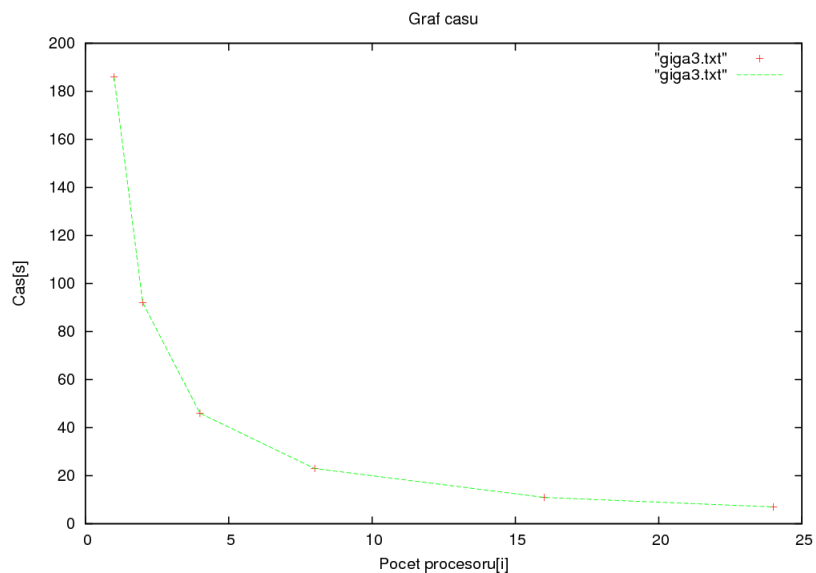
Sekvenční řešení je implementováno jako rekurzivní prohledávání stavového prostoru který je tvořený permutacemi 0 a 1 délky 1 až k . Proměna k je omezením hloubky stavového prostoru.

2 Popis paralelního algoritmu a jeho implementace v MPI

- Paralelní algoritmus má na rozdíl od sekvenčního řešení vlastní zásobník který je možné rozdělit, protože lze přistoupit jak k vrcholu tak ke spodku.
- Řešení problému BB_DFS, prohledávání **úplného** stavového prostoru pro nalezení nejlepšího řešení.
- Algoritmus dělení zásobníku = D_AHD viz [1]
- Algoritmus hledání dárce = ACŽ_AHD (asynchronní cyklické žádosti) viz [1]
- Algoritmus ukončení ADUV viz [1]

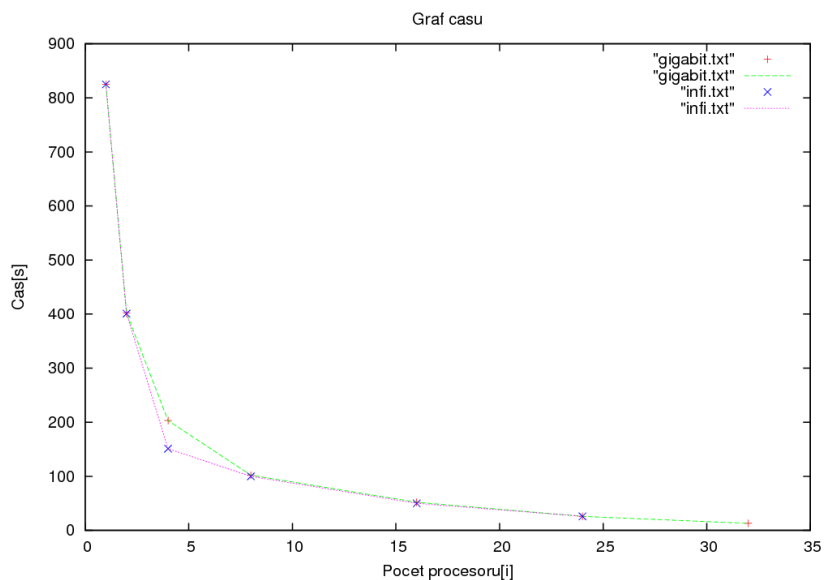
3 Naměřené výsledky a vyhodnocení

3.1 Měření pro instanci 3min sekvenčního řešení



Obrázek 1: Měření pro instanci $n = 10, z = 10, m = 10, k = 25$

3.2 Měření pro instanci 15 min sekvenčního řešení



Obrázek 2: Měření pro instanci $n = 10, z = 10, m = 10, k = 27$

4 Závěr

Podařilo se nám navrhnout a implementovat jednoduché sekvenční řešení, které bylo jednoduché paralizovat. Z grafu je patrné že jsme dosáhli téměř lineárního zrychlení. Dále jsme zjistili že síť InfiniBand je o trochu rychlejší než GigabitE ale čekali jsme lepší výsledek. Výraznější zrychlení pomocí InfiniBandu by se asi projevilo až při větších instancích problému a více komunikace.

Reference

- [1] *Slide z předmětu MI-PAR [online]* (Říjen 2011)
https://edux.fit.cvut.cz/courses/MI-PAR/_media/lectures/mi-par2011-prednaska1.pdf
- [2] *Reference knihovny boost [online]* (Listopad 2011)
<http://www.boost.org/>
- [3] *C++ reference* (Listopad 2011)
<http://www.cplusplus.com/reference/>