

PORR DOKUMENTACJA ADDENDA

Michał Przyłuski

10 lutego 2010

1 Zrównoleglenie OpenMP

Poniższa tabela prezentuje czasy wykonania wersji sekwencyjnej zdekomponowanej na sun16.

	200k	700k	2M	4M
2 pod.	30.37s	1m47s	5m44s	12m36s
4 pod.	30.64s	1m54s	5m22s	14m57s
8 pod.	26.96s	1m39s	7m02s	16m35s

Tabela 1: Wyniki gs-extros-b w wersji sekwencyjnej zdekomponowanej na sun16

Porównując ją z tabelą 6 z oryginalnego sprawozdania zauważamy:

- 200k — najlepsza dekompozycja: 26.96s — odpowiadające jej OMP: 6.43s — prawie 4.2 razy szybciej (8 wątków)
- 700k — najlepsza dekompozycja: 1m39s — odpowiadające jej OMP: 23.86s — prawie 4.15 razy szybciej (8 wątków)
- 2M — najlepsza dekompozycja: 5m22s — odpowiadające jej OMP: 2m28s — prawie 2.2 razy szybciej (4 wątki)
- 4M — najlepsza dekompozycja: 12m36s — odpowiadające jej OMP: 8m45s — prawie 1.5 razy szybciej (2 wątki)
- 4M — najlepsze OMP: 3m59s — dekompozycja na 8: 16m35s — prawie 4.2 razy szybciej (8 wątków)

A zatem przy wykorzystaniu 8 wątków (i jednocześnie dekompozycji problemu na 8) udało się uzyskać nawet ponad 4-krotny spadek czasu wykonania programu.

Warto dodać, że dla mniejszej liczby wątków uzyskano lepsze („pełniejsze”) wykorzystanie wielu wątków, tj. przykładowo wersja 700k zdekomponowana na 4 przyspieszyła z 1m54s do 34.80s, a zatem prawie 3.3 razy.

2 Zrównoleglenie MPI

Poniższa tabela prezentuje czasy wykonania wersji sekwencyjnej zdekomponowanej na ux2 (czyli na jednym rdzeniu). Ze względu na to, że rdzenie ux2 są taktowane zegarem 3GHz czasy wykonania są zauważalnie krótsze niż na sun16 (por. tabela 1).

	200k	700k	2M	4M
2 pod.	14.27s	56.29s	3m42s	8m7s
4 pod.	13.37s	54.87s	3m27s	9m30s
8 pod.	13.67s	56.80s	3m28s	8m53s

Tabela 2: Wyniki gs-extros-b w wersji sekwencyjnej zdekomponowanej na ux2

Warto zauważyć, że pewne rozmiary poditeracji są bardziej korzystne niż inne, chociaż w przypadku małych zadań (< 4M) różnice te są znikome.

Kolejna tabela zawiera czasy wykonania tej samej wersji po zrównolegleniu OpenMPI na klastrze ux1-12.

	200k	700k	2M	4M
np=2	12.36s	49.57s	3m18s	7m14s
np=4	7.22s	25.65s	1m58s	4m42s
np=8	5.05s	16.50s	1m11s	2m34s

Tabela 3: Wyniki gs-extros-b w wersji MPI na klastrze ux1-12

W powyższym przypadku, liczba podziałów dziedziny była równa liczbie procesów (np). Ponieważ czasy wykonania wersji zdekomponowanej dla problemów 200k, 700k i 2M są zbliżone, poniżej porównano najlepszą dekompozycję do **najlepszego MPI**.

- 200k — najlepsza dekompozycja: 13.37s — MPI: 5.05s — 2.65 razy szybciej (8 wątków)
- 700k — najlepsza dekompozycja: 54.87s — MPI: 16.50s — prawie 3.35 razy szybciej (8 wątków)
- 2M — najlepsza dekompozycja: 3m27s — MPI: 1m11s — prawie 2.92 razy szybciej (8 wątków)
- 4M — najlepsza dekompozycja: 8m7s — MPI: 7m14s — prawie 1.15 razy szybciej (2 wątki)
- 4M — najlepsze MPI: 2m34s — dekompozycja na 8: 8m53s — prawie 3.5 razy szybciej (8 wątków)

Niestety nie można tych wyników (bezwzględnych czasów wykonania) odnosić bezpośrednio do czasów wykonania wersji sekwencyjnej zdekomponowanej na sun16, ze względu na znaczną różnicę w wydajności pojedynczego rdzenia na sun16 i ux2. Z

tego błędnego porównania mogły wynikać przedwczesne wnioski o prawie 8-krotnym przyspieszeniu wykonania problemu 700k między równoległym wykonaniem na ux1-12, a sekwencyjnym sun16. Przykładowa maszyna ux2, przy wykorzystaniu jednego rdzenia, jest prawie 2 razy szybsza niż pojedynczy rdzeń sun16.

3 Wnioski

Udało się uzyskać dobre przyspieszenie działania programu dzięki równoległemu wykonaniu wersji zdekomponowanej na wielu jednostkach wykonawczych. Rozwiązania OpenMP oraz MPI oferują porównywalną wydajność, chociaż zazwyczaj OpenMP jest szybsze. Wynikać to może ze znacznie mniejszych narzutów na komunikację.

Ponad 4-krotne przyspieszenie (w skrajnych przypadkach) jest zadowalającym wynikiem. Wydaje się, że dalszej poprawy wydajności można szukać w usprawnieniu alokacji pamięci, która w przypadku wykonania na maszynie z pamięcią wspólną (sun16) może ograniczać wydajność dla dużych zadań.