



Системы и средства параллельного программирования

Отчёт

Параллельный алгоритм умножения матрицы на вектор

Работу выполнил
Евгений Кислов, 323

Москва 2020

Постановка задачи и формат данных

Задача

Разработать параллельную программу с использованием технологии MPI, реализующую алгоритм умножения плотной матрицы на вектор $Ab=c$. Тип данных – double. Провести исследование эффективности разработанной программы на системе Blue Gene/P.

Требуется:

1. Разработать параллельную программу с использованием технологии MPI. Предусмотреть равномерное распределение элементов матрицы блоками строк или столбцов, в зависимости от соотношения m и n . Вектора b и c распределены по процессам равномерно.
2. Исследовать эффективность разработанной программы в зависимости от размеров матрицы и количества используемых процессов. Построить графики времени работы, ускорения и эффективности разработанной программы. Время на ввод/вывод данных не включать.
3. Исследовать влияние мэппинга параллельной программы на время работы программы.
4. Построить таблицы: времени, ускорения, эффективности.

m	n	мэппинг	32	64	128	256	512

Для 512 процессоров рассмотреть два варианта мэппинга – стандартный, принятый по умолчанию и произвольный. Для произвольного мэппинга предусмотреть генерацию строк файла для задания случайного значения XYZT.

Ускорение (speedup), получаемое при использовании параллельного алгоритма для p процессоров, определяется величиной:

$$\text{Speedup}(n) = T1(n)/Tp(n),$$

где $T1(n)$ - время выполнения задачи на одном процессоре.

$Tp(n)$ - время параллельного выполнения задачи при использовании p процессоров.

5. Построить графики – для каждого из заданных значений размеров матрицы (512x512, 1024x1024, 2048x2048, 4096x4096, 4096x1024, 1024x4096).

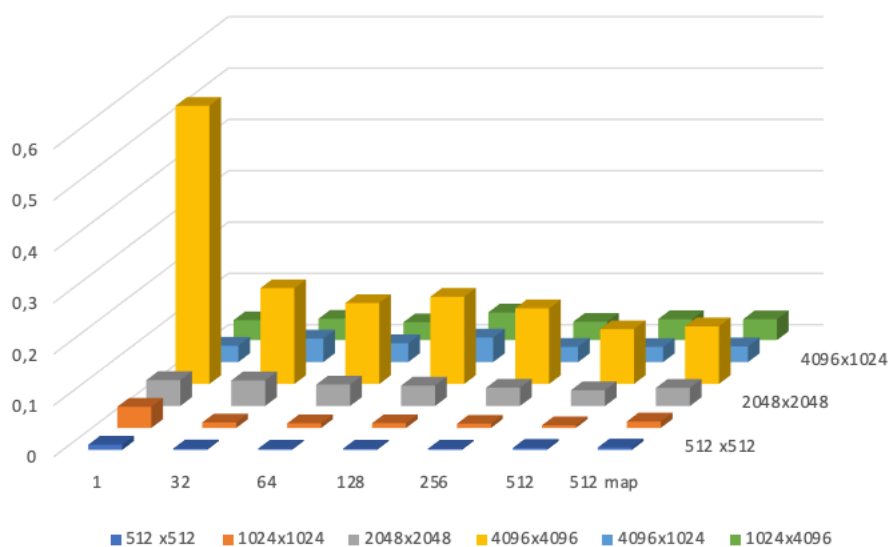
Формат командной строки

- имя файла – матрица A размером $m \times n$
- имя файла - вектор b
- имя файла – результат, вектор c

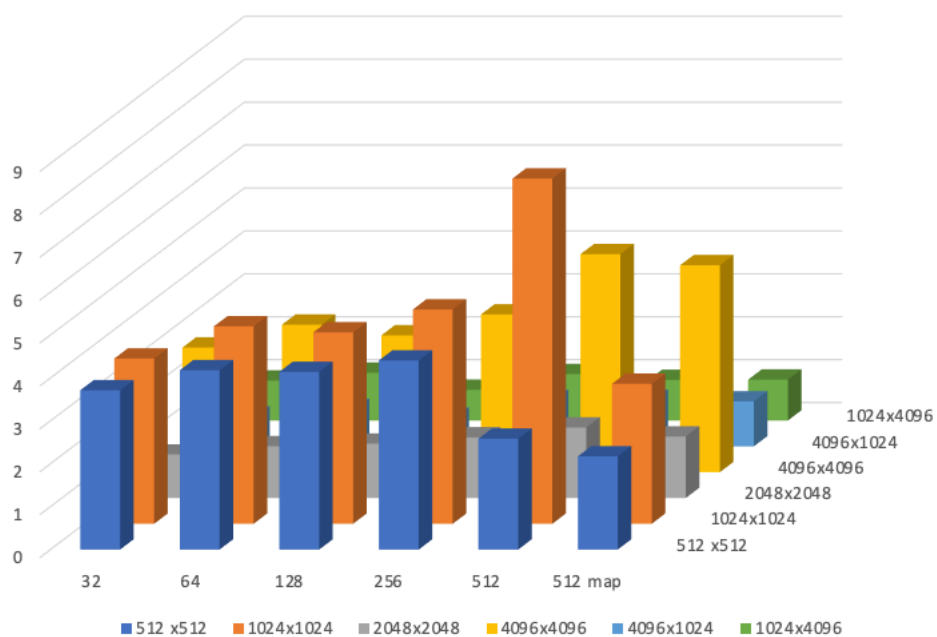
Формат задания матрицы A – как в первом задании.

Результаты

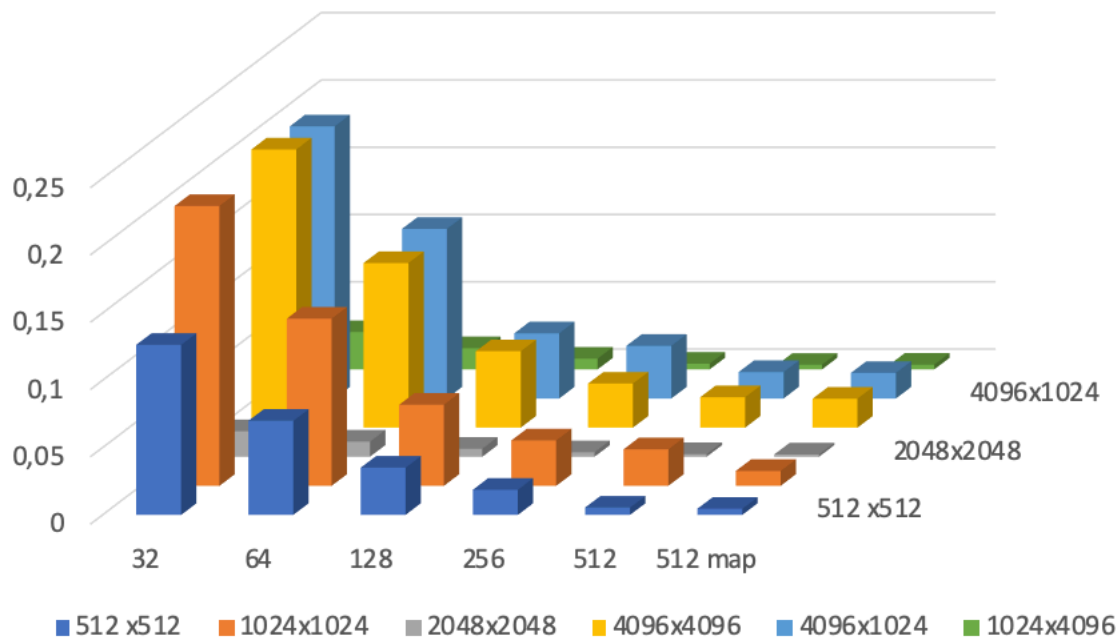
Время выполнения



Ускорение



Эффективность



Таблицы

60		512 x 512	1024x1024	2048x2048	4096x4096	4096x1024	1024x4096
61	1	0,0101542	0,04124313	0,05038302	0,54184021	0,03193103	0,0386591
62	32	0,00273534	0,01070898	0,04972847	0,18656258	0,04578827	0,04152663
63	64	0,00242834	0,00896303	0,04176783	0,15762873	0,03672624	0,03473687
64	128	0,00245158	0,00924017	0,03978272	0,16987391	0,04782672	0,0538392
65	256	0,00230368	0,00825485	0,03565123	0,14736824	0,02954832	0,03572372
66	512	0,00392944	0,00512552	0,03066261	0,10662342	0,02936873	0,04076872
67	512 map	0,00467121	0,01263767	0,03516271	0,11236101	0,03047101	0,0405611
68							
69							
70		512 x 512	1024x1024	2048x2048	4096x4096	4096x1024	1024x4096
71	32	3,712225902	3,851265947	1,01316248	2,904334889	0,6973626652	0,9309472018
72	64	4,181539653	4,601471824	1,206263768	3,437445763	0,8694336801	1,112912591
73	128	4,141900326	4,463460088	1,266454883	3,189661143	0,667639972	0,718047445
74	256	4,407817058	4,996230095	1,413219684	3,676777371	1,080637749	1,082168934
75	512	2,584134126	8,046623562	1,643141924	5,081812326	1,087245856	0,9482539555
76	512 map	2,173783666	3,263507435	1,432853725	4,822315232	1,047915051	0,9531077806
77							
78							
79							
80		512 x 512	1024x1024	2048x2048	4096x4096	4096x1024	1024x4096
81	32	0,1160070595	0,1203520608	0,03166132751	0,09076046527	0,02179258329	0,02909210006
82	64	0,06533655707	0,07189799725	0,01884787138	0,05371009004	0,01358490125	0,01738925924
83	128	0,03235859629	0,03487078194	0,009894178773	0,02491922768	0,005215937281	0,005609745664
84	256	0,01721803538	0,01951652381	0,005520389391	0,0143624116	0,004221241206	0,004227222399
85	512	0,005047136965	0,01571606164	0,003209261571	0,009925414699	0,002123527063	0,001852058507
86	512 map	0,004245671223	0,006374037958	0,002798542431	0,009418584437	0,002046709084	0,001861538634

Выводы

Параллелизм дает хорошие результаты с ростом объема данных. Ускорение программы растет до определенного предела и затем перестает увеличиваться, что связано с ростом накладных расходов на создание процессов и пересылку данных.