МИНОБРНАУКИ РОССИИ

Федеральное государственное автономное образовательное учреждение высшего образования «Южный федеральный университет»

Институт математики, механики и компьютерных наук им. И.И.Воровича Кафедра прикладной математики и программирования

Направление подготовки 02.04.02 – Фундаментальная информатика и информационные технологии

ОТЧЕТ ПО ТЕМЕ  
«Блочное перемножение матриц»

Выполнил:  
магистр 1 года 6 группы  
Тактаров Е.А.

Описание задачи

В основе эксперимента лежит задача блочного перемножения двух матриц вещественных чисел, где левая матрица нижне-треугольная, а правая симметричная. В памяти матрицы хранятся ненулевыми блоками в одномерном массиве по блочным столбцам и строкам для левой и правой матриц, соответственно.

С помощью программной реализации решения этой задачи, будет выяснено, как на ее производительность влияют различные размеры блоков, типы данных и компиляторы.

Характеристики компьютера

Processor: Intel® Core™ i5-3300 CPU @ 3.00GHz; 4 cores, 4 threads; no hyperthreading;

L3: 6 MB (shared);

L2: 256 kB (split);

L1: 32 kB instruction cache, 32 kB data cache (split);

RAM: DDR3, 2\*8 GB, clock speed: 668MHz.

Характеристики компиляторов

Programming language: C++;

Compiler 1: MSVC2017 14.16.27023 x64 compiler options: /c /Zi /nologo /W3 /WX- /diagnostics:classic /sdl /O2 /Ob2 /Oi /Ot /GL /D NDEBUG /D \_CONSOLE /D \_UNICODE /D UNICODE /Gm- /EHsc /MD /GS /Gy /fp:precise /permissive- /Zc:wchar\_t /Zc:forScope /Zc:inline /openmp /Fo"x64\Release\\" /Fd"x64\Release\vc141.pdb" /Gd /TP /FC /errorReport:prompt MatrixMultiplication.cpp

Compiler 2: MINGGW730 x64 compiler options: -c -fno-keep-inline-dllexport -fopenmp -O3 -Wall -W -Wextra -fexceptions -mthreads -DUNICODE -D\_UNICODE -DWIN32 -DMINGW\_HAS\_SECURE\_API=1 -DQT\_NO\_DEBUG -DQT\_GUI\_LIB -DQT\_CORE\_LIB -I..\..\matrixMultiplication-master -I. -ID:\Qt\5.13.2\mingw73\_64\include -Irelease -o release\MatrixMultiplication.o ..\MatrixMultiplication\MatrixMultiplication.cpp

OS: Windows 10 Pro v1809

Особенности реализации

Стоить отметить некоторые особенности реализации алгоритма, призванные увеличить скорость расчета.

В первую очередь, использовались матрицы смещения. Заранее посчитав смещения одномерного массива необходимые для получения каждого блока, можно поместить их в двумерный массив, где каждый элемент соответствует блоку. Таким образом, избавиться от дорогостоящих вычислений для получения каждого блока. Может показаться, что в реальности такая «уловка» не сработает, но зачастую размеры матрицы в задачах для индустриального расчета константы, поэтому смещения не нужно пересчитывать, если поменяются входные данные. Поэтому построение таких структур и другие вспомогательные операции в подсчете времени учтены не были.

Также верхний двойной цикл в перемножении блоков был заменен на ординарный цикл по задачам – парам из ***i*** и ***j***. Ординарный цикл более удобен для распараллеливания через **OpenMP**.

Время выполнения в миллисекундах

|  |  |  |  |
| --- | --- | --- | --- |
| MSVC 2017 Тип данных: Double | | |  |
| Размер блока | Без параллельности | Блочная параллельность | Матричная параллельность |
| 1 | 338534,22 | 164179,99 | 7839915,31 |
| 6 | 16105,70 | 7537,98 | 43308,21 |
| 10 | 12781,80 | 5787,66 | 12016,61 |
| 15 | 12471,95 | 5563,52 | 6465,31 |
| 20 | 11230,38 | 4992,45 | 4532,92 |
| 24 | 11854,43 | 5264,54 | 4206,28 |
| 30 | 12423,82 | 5536,21 | 4173,16 |
| 36 | 12138,87 | 5408,41 | 3809,42 |
| 40 | 12650,44 | 5606,88 | 3918,67 |
| 60 | 13635,04 | 6062,13 | 4048,79 |
| 72 | 14529,55 | 6457,59 | 4218,89 |
| 80 | 15241,71 | 6756,07 | 4323,29 |
| 96 | 16391,11 | 7227,27 | 4547,41 |
| 120 | 15981,76 | 7056,98 | 4475,55 |
| 144 | 16931,72 | 7482,94 | 4708,65 |
| 160 | 17583,34 | 7706,25 | 4872,24 |
| 180 | 17175,70 | 7540,45 | 4732,88 |
| 240 | 18269,69 | 7936,37 | 4977,25 |
| 360 | 19030,96 | 8165,16 | 5170,87 |
| 480 | 20673,83 | 8909,59 | 5551,68 |
| 720 | 84015,50 | 30683,32 | 22288,18 |

|  |  |  |  |
| --- | --- | --- | --- |
| MSVC 2017 Тип данных: Float | | | |
| Размер блока | Без параллельности | Блочная параллельность | Матричная параллельность |
| 1 | 272791,98 | 140275,68 | 7928181,82 |
| 6 | 12835,91 | 6431,58 | 49988,97 |
| 10 | 10979,73 | 5420,22 | 14586,72 |
| 15 | 10781,68 | 5318,84 | 7393,42 |
| 20 | 10609,25 | 5265,28 | 5858,75 |
| 24 | 10615,48 | 5207,33 | 5391,79 |
| 30 | 10416,11 | 5173,82 | 5439,61 |
| 36 | 11158,05 | 5627,70 | 5296,46 |
| 40 | 11481,25 | 5479,85 | 5511,57 |
| 60 | 10718,90 | 4997,85 | 5798,11 |
| 72 | 10606,48 | 4866,69 | 5910,32 |
| 80 | 10708,52 | 4907,33 | 6004,41 |
| 96 | 10715,60 | 4867,80 | 6142,31 |
| 120 | 10308,83 | 4658,98 | 6248,36 |
| 144 | 10503,96 | 4691,13 | 6342,97 |
| 160 | 10584,45 | 4711,01 | 6415,78 |
| 180 | 10208,23 | 4508,77 | 6469,98 |
| 240 | 10700,43 | 4692,01 | 6641,16 |
| 360 | 12457,33 | 5231,81 | 7004,67 |
| 480 | 15179,52 | 6296,55 | 7549,68 |
| 720 | 16739,00 | 6998,69 | 8118,04 |

|  |  |  |  |
| --- | --- | --- | --- |
| MINGW Тип данных: Double | | | |
| Размер блока | Без параллельности | Блочная параллельность | Матричная параллельность |
| 1 | 317607,53 | 152824,46 | 226045410,00 |
| 6 | 13677,38 | 6507,42 | 1425267,18 |
| 10 | 11184,13 | 5089,42 | 312424,18 |
| 15 | 10962,29 | 4937,53 | 95077,73 |
| 20 | 10890,34 | 4956,51 | 42039,32 |
| 24 | 11049,23 | 5019,47 | 25651,72 |
| 30 | 10901,33 | 4937,53 | 14659,68 |
| 36 | 12535,18 | 5638,03 | 9636,05 |
| 40 | 12497,21 | 5648,03 | 8307,15 |
| 60 | 12067,51 | 5404,20 | 4721,68 |
| 72 | 12292,35 | 5453,16 | 4184,93 |
| 80 | 12466,23 | 5582,07 | 4031,56 |
| 96 | 12735,04 | 5663,02 | 3825,02 |
| 120 | 12305,34 | 5470,15 | 3535,23 |
| 144 | 12490,21 | 5523,11 | 3512,43 |
| 160 | 13529,48 | 5964,80 | 3756,25 |
| 180 | 12612,13 | 5509,12 | 3475,26 |
| 240 | 14520,78 | 6169,66 | 3952,05 |
| 360 | 14522,78 | 6118,69 | 4069,30 |
| 480 | 17981,35 | 7770,53 | 4834,60 |
| 720 | 80144,61 | 28578,89 | 21340,98 |

|  |  |  |  |
| --- | --- | --- | --- |
| MINGW Тип данных:Float | | |  |
| Размер блока | Без параллельности | Блочная параллельность | Матричная параллельность |
| 1 | 286192,23 | 137414,14 | 233933010,00 |
| 6 | 13805,27 | 6304,55 | 1475000,27 |
| 10 | 12233,37 | 5502,12 | 320349,12 |
| 15 | 12084,48 | 5380,20 | 97163,16 |
| 20 | 11832,66 | 5272,28 | 42658,92 |
| 24 | 12026,52 | 5215,32 | 25989,66 |
| 30 | 11840,65 | 5190,34 | 14869,51 |
| 36 | 13233,67 | 5648,02 | 9727,34 |
| 40 | 13549,44 | 5920,83 | 8459,68 |
| 60 | 12713,04 | 5507,12 | 4753,46 |
| 72 | 12002,54 | 5358,22 | 4062,66 |
| 80 | 12057,50 | 5368,21 | 3835,29 |
| 96 | 12011,53 | 5317,25 | 3578,47 |
| 120 | 11589,83 | 5089,41 | 3302,25 |
| 144 | 11675,77 | 5125,38 | 3254,62 |
| 160 | 11776,69 | 5158,36 | 3262,35 |
| 180 | 11421,95 | 4973,49 | 3133,71 |
| 240 | 11856,64 | 5118,95 | 3217,63 |
| 360 | 13772,29 | 5759,94 | 3765,34 |
| 480 | 17014,00 | 7725,55 | 4630,73 |
| 720 | 18700,81 | 7788,12 | 5100,40 |

Графики зависимости времени исполнения от размера блока

Сравнительный анализ

Как видно, MINGW для блочной параллельности всегда показывает результат лучше, однако сильно проигрывает MSVC для низких размеров блоков при матричной параллельности. Вполне возможно, что это вызвано, тем что MSVC более эффективно реализует открытие и закрытие потоков в ОС Windows, которые возникают особенно часто на матричной параллельности с низким размером блоков.

Стоит отметить, что матричная параллельность оказалась самым эффективным способом перемножения матриц для обоих вещественных типов особенно на больших размерах блоков. Скорее всего, это вызвано эффективными векторными операциями на процессоре, а резкий спад эффективности работы на размерах блоков от 720 сигнализирует, от том что вектора такого и больших размеров уже не помещаются в кэш.

В сравнении типов данных быстрее оказался расчет матрицы типа Float, но при этом выбор типа не сильно влияет на зависимость времени от компилятора или размера блоков.

Вывод

Результаты проведённых экспериментов показали, что, хотя компиляторы генерируют различные по эффективности наборы инструкций, но явно превосходящего компилятора нет. Однако критичным для скорости работы оказались выбор метода параллельности и размера блока.