

МГУ им. М. В. Ломоносова, факультет ВМК

Задание 1

Решение СЛАУ методом отражений

Арбузов Николай Романович
группа 323

Постановка задачи

Требуется написать параллельную программу с использованием технологии OpenMP для решения системы линейных уравнений $Ax = b$ методом отражений.

Алгоритм

1. Приводим матрицу A к верхне-треугольному виду с помощью матрицы отражения $U(x) = I - 2xx^T$; при $\|x\| = 1$.

- a. За вектор x берется вектор, построенный с использованием элементов матрицы A :

$$x^{(i)} = \pm \frac{a_i - \|a_i\|e_i}{\|a_i - \|a_i\|e_i\|}$$

- b. При умножении матриц $U(x^{(i)})A$ получается матрица A^* , в которой i -й столбец имеет все нули после i -ого элемента.

2. С помощью метода Гаусса находим значения элементов вектора x .
3. Система решена.

Компиляция и запуск

Все вычисления производились на машине Polus.

Сама программа написана на языке C++ и состоит из файлов:

- main.cpp
- reflection_method.cpp
- reflection_method.h
- matrix.cpp
- matrix.h

Компилировалась с использованием Makefile:

```
all: main

main: *.cpp *.h
    g++ *.cpp -o prog -std=c++17 -fopenmp

omp_polus: *.cpp *.h
    xlc++ *.cpp -o prog -Wall -std=c++11 -qsmp=omp -fopenmp

clean:
    rm -rf ./prog
```

Запуск производился постановкой в очередь с помощью lsf-файлов вида:

Для 1 и 2 потоков:

```
#BSUB -n 1
#BSUB -W 00:15
#BSUB -o \"./out_files/j/i.out\"
#BSUB -e \"./err_files/j/i.err\"
#BSUB -R \"span[hosts=1]\"
OMP_NUM_THREADS=i ./prog j
```

Для 4, 8, 16 и 32 потоков:

```
#BSUB -W 00:15
#BSUB -o \"./out_files/j/i.out\"
#BSUB -e \"./err_files/j/i.err\"
#BSUB -R \"affinity[core(M)]\"
OMP_NUM_THREADS=i
/polusfs/lsf/openmp/launchOpenMP.py ./prog j
```

Где i – количество потоков, на которых будет запускаться программа, M – количество ядер ($M = i / 2$), j – размер матрицы, на которой будут производиться вычисления.

Результаты

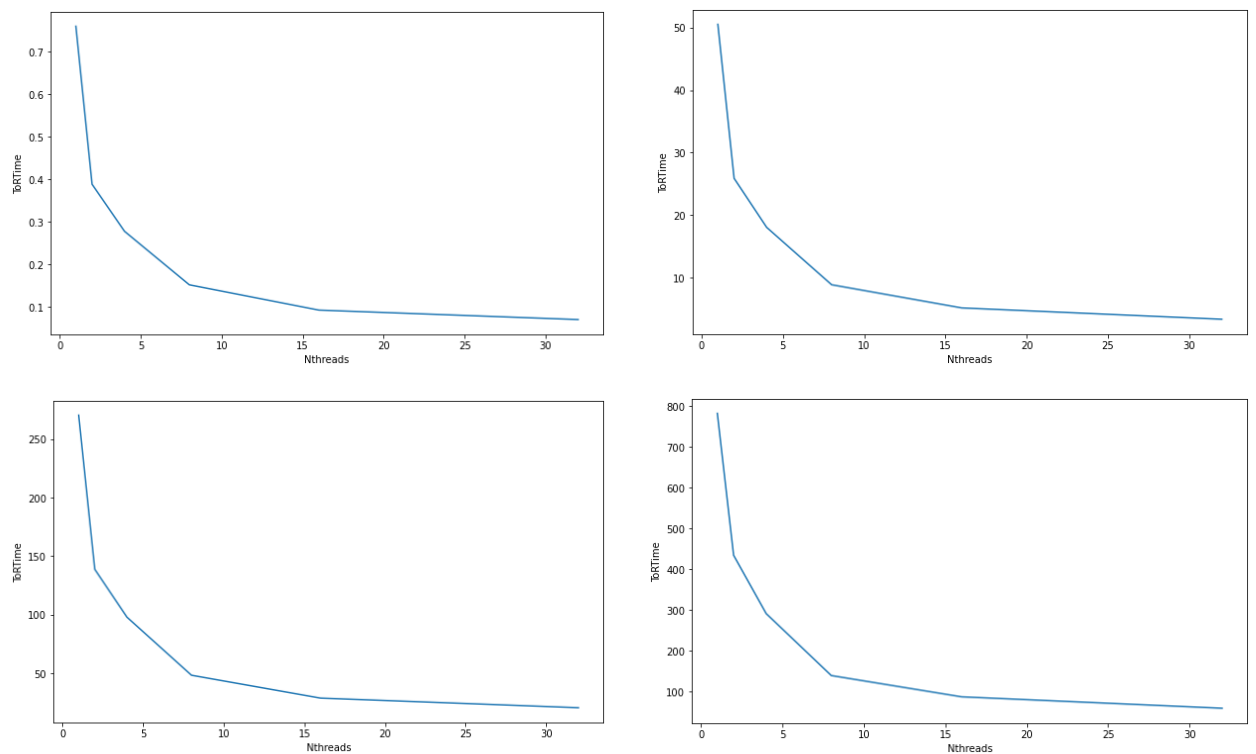


Рисунок 1. На первом графике время для матрицы размерности 1000x1000, на втором – 4000x4000, на третьем – 7000x7000, на четвертом – 10000x10000.

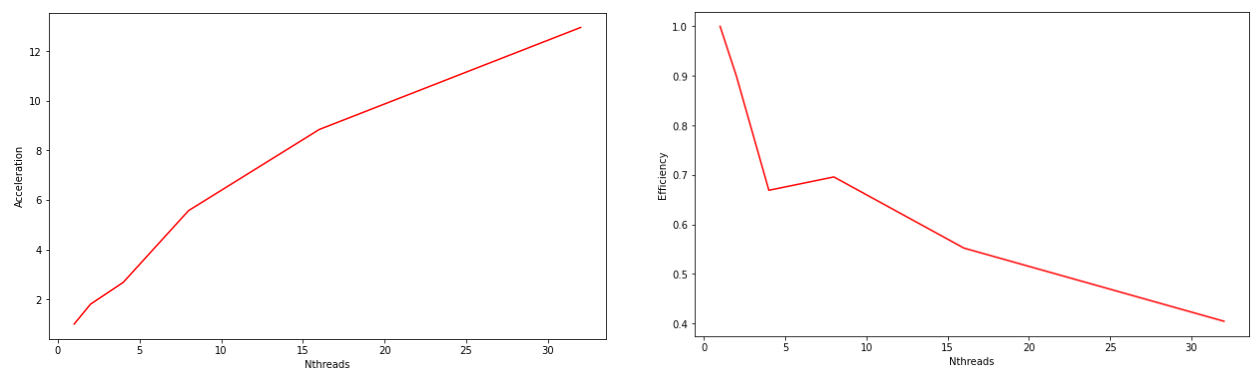


Рисунок 2. Первый график – зависимость ускорения от количества потоков. Второй – зависимость эффективности от количества потоков.

Полная таблица результатов:

FullTime	ToRTime	GaussTime	Nthreads	MatrixSize	Residual	Acceleration	Efficiency
0.761482	0.760371	0.001111	1	1000	2.345460e-12	1.000000	1.000000
0.389845	0.388836	0.001010	2	1000	2.345460e-12	1.953294	0.976647
0.279615	0.278227	0.001387	4	1000	2.345460e-12	2.723323	0.680831
0.153823	0.152454	0.001370	8	1000	2.345460e-12	4.950378	0.618797
0.094356	0.092899	0.001457	16	1000	2.345460e-12	8.070317	0.504395
0.072182	0.070668	0.001514	32	1000	2.345460e-12	10.549472	0.329671
50.594600	50.527500	0.067072	1	4000	2.693120e-11	1.000000	1.000000
25.942700	25.888700	0.054017	2	4000	2.693120e-11	1.950244	0.975122
18.103600	18.054600	0.049069	4	4000	2.693120e-11	2.794726	0.698681
8.916270	8.867300	0.048971	8	4000	2.693120e-11	5.674413	0.709302
5.193050	5.156930	0.036120	16	4000	2.693120e-11	9.742752	0.608922
3.397200	3.346840	0.050368	32	4000	2.693120e-11	14.893030	0.465407
270.753000	270.449000	0.303913	1	7000	1.320350e-11	1.000000	1.000000
139.119000	138.808000	0.310710	2	7000	1.320350e-11	1.946197	0.973099
98.429800	97.873000	0.556792	4	7000	1.320350e-11	2.750722	0.687680
48.812000	48.274900	0.537101	8	7000	1.320350e-11	5.546853	0.693357
29.264100	28.779900	0.484259	16	7000	1.320350e-11	9.252053	0.578253
21.133800	20.548000	0.585798	32	7000	1.320350e-11	12.811373	0.400355
783.245000	782.488000	0.756235	1	10000	4.143470e-11	1.000000	1.000000
435.173000	434.405000	0.768188	2	10000	4.143470e-11	1.799847	0.899924
292.648000	291.326000	1.322390	4	10000	4.143470e-11	2.676406	0.669102
140.675000	139.434000	1.241150	8	10000	4.143470e-11	5.567763	0.695970
88.619100	87.187700	1.431450	16	10000	4.143470e-11	8.838332	0.552396
60.484600	59.176600	1.308040	32	10000	4.143470e-11	12.949495	0.404672