

Задание 7

Отчёт

по реализации и анализу алгоритмов работы с графом в формате CSR

Ши Хуэй shihuicollapsor@gmail.com

1. Постановка задачи

Реализовать параллельное решение двумерного уравнения Лапласа (при $g = 0$) методом Якоби, запуская его на кластере Polus. Для этого необходимо построить квадратную сетку с размером N , являющимся степенью двойки, и разделить её на горизонтальные полосы, где каждый процесс будет обрабатывать свою часть, равную по ширине всей сетке. Начальные значения функции f на сетке задаются случайными числами.

Выполнить заданное количество итераций n_iter метода Якоби и на последней итерации рассчитать норму разности решений между двумя соседними шагами времени на каждом процессе.

Требования к реализации: массив, представляющий всю сетку, не должен храниться на одном процессе (за исключением случаев запуска на одном процессе). Для коммуникации следует использовать только блокирующие point-to-point методы MPI. Программу необходимо запустить с помощью `mpisubmit.pl` на кластере Polus для числа процессов $P = \{1, 2, 4, 8, 12, 16, 24, 32, 36\}$, а также собрать статистику по вызовам MPI с помощью инструмента `mpiP`.

После выполнения программы для фиксированного большого размера сетки требуется построить графики зависимости времени выполнения $T(P)$, ускорения $S(P)$ и параллельной эффективности $E(P)$ от числа процессов. На основе данных `mpiP` необходимо кратко прокомментировать полученные результаты и отметить интересные наблюдения.

2. Формат командной строки

```
mpicc -g -Wl,-rpath=/home_edu/edu-cmc-sqi22/edu-cmc-sqi22-29/sem07/mpiP-3.5 -L /home_edu/edu-cmc-sqi22/edu-cmc-sqi22-29/sem07/mpiP-3.5 -lmpiP work7.c -o work7  
mpisubmit.pl -p 4 -w 00:05 ./work7
```

3. Спецификация системы

- Operating system : polus

4. Записи экспериментов и результаты

P	T	Norm	E	P
1	22.209154	29.194801	1	1
2	11.109417	20.605502	1.99912866714788	0.99956433357394
4	10.326312	14.425558	2.15073435704829	0.537683589262073
8	2.959509	10.522345	7.50433737488212	0.938042171860265
12	2.118885	8.615898	10.481528728553	0.873460727379417
16	1.633433	7.538623	13.5966115537032	0.84978822210645
24	2.165094	6.041378	10.2578243715977	0.427409348816571
32	1.264377	5.301516	17.5652942120902	0.548915444127819
36	1.064949	4.640325	20.8546644017695	0.579296233382486

P=1

Iteration 0, Norm: 331.264034

Iteration 100, Norm: 36.268891

Iteration 200, Norm: 32.455353

Iteration 300, Norm: 31.106265

Iteration 400, Norm: 30.419350

Iteration 500, Norm: 30.005285

Iteration 600, Norm: 29.730193

Iteration 700, Norm: 29.535451

Iteration 800, Norm: 29.391247

Iteration 900, Norm: 29.280809

Total execution time (T): 22.209154 seconds

Max norm after 1000 iterations: 29.194801

P=2

Iteration 0, Norm: 234.137034

Iteration 100, Norm: 25.379712

Iteration 200, Norm: 22.804827

Iteration 300, Norm: 21.905295

Iteration 400, Norm: 21.446129

Iteration 500, Norm: 21.166637

Iteration 600, Norm: 20.978767

Iteration 700, Norm: 20.844305

Iteration 800, Norm: 20.743812

Iteration 900, Norm: 20.666268

Total execution time (T): 11.109417 seconds

Max norm after 1000 iterations: 20.605502

P=4

[edu-cmc-sqi22-29@polus-ib sem07]\$ export MPIP="-t 10.0 -k 4"

[edu-cmc-sqi22-29@polus-ib sem07]\$ mpisubmit.pl -p 4 -w 00:05 ./work7

Job <1205998> is submitted to default queue <short>.

[edu-cmc-sqi22-29@polus-ib sem07]\$ cat work7.1205998.out

Sender: LSF System <lsfadmin@polus-c4-ib.bmc.hpc.cs.msu.ru>

Subject: Job 1205998: <# this file was automatically created by mpisubmit.pl script for edu-cmc-sqi22-29 #;source /polusfs/setenv/setup.SMPI;#BSUB -n 4;#BSUB -W 00:05;#BSUB -o work7.%J.out;#BSUB -e work7.%J.err;OMP_NUM_THREADS=1 mpiexec ./work7> in cluster <MSUCluster> Done

Job <# this file was automatically created by mpisubmit.pl script for edu-cmc-sqi22-29 #;source /polusfs/setenv/setup.SMPI;#BSUB -n 4;#BSUB -W 00:05;#BSUB -o work7.%J.out;#BSUB -e work7.%J.err;OMP_NUM_THREADS=1 mpiexec ./work7> was submitted from host <polus-ib.bmc.hpc.cs.msu.ru> by user <edu-cmc-sqi22-29> in cluster <MSUCluster> at Thu Nov 7 13:41:43 2024

*Job was executed on host(s) <4*polus-c4-ib.bmc.hpc.cs.msu.ru>, in queue <short>, as user <edu-cmc-sqi22-29> in cluster <MSUCluster> at Thu Nov 7 13:41:44 2024*

</home_edu/edu-cmc-sqi22/edu-cmc-sqi22-29> was used as the home directory.

</home_edu/edu-cmc-sqi22/edu-cmc-sqi22-29/sem07> was used as the working directory.

Started at Thu Nov 7 13:41:44 2024

Terminated at Thu Nov 7 13:41:55 2024

Results reported at Thu Nov 7 13:41:55 2024

Your job looked like:

#LSBATCH: User input

this file was automatically created by mpisubmit.pl script for edu-cmc-sqi22-29
source /polusfs/setenv/setup.SMPI

#BSUB -n 4

#BSUB -W 00:05

#BSUB -o work7.%J.out

#BSUB -e work7.%J.err

OMP_NUM_THREADS=1 mpiexec ./work7

Successfully completed.

Resource usage summary:

<i>CPU time :</i>	<i>41.81 sec.</i>
<i>Max Memory :</i>	<i>252 MB</i>
<i>Average Memory :</i>	<i>189.25 MB</i>
<i>Total Requested Memory :</i>	<i>-</i>
<i>Delta Memory :</i>	<i>-</i>
<i>Max Swap :</i>	<i>-</i>
<i>Max Processes :</i>	<i>8</i>
<i>Max Threads :</i>	<i>21</i>
<i>Run time :</i>	<i>13 sec.</i>
<i>Turnaround time :</i>	<i>12 sec.</i>

The output (if any) follows:

mpiP: Found MPIP environment variable [-t 10.0 -k 4]

mpiP: Set the report print threshold to [10.00%].

mpiP: Set the callsite stack traceback depth to [4].

mpiP:

mpiP: mpiP V3.5.0 (Build Nov 6 2024/21:55:54)

mpiP:

Iteration 0, Norm: 165.467168

Iteration 100, Norm: 17.744697

Iteration 200, Norm: 15.895827

Iteration 300, Norm: 15.274757

Iteration 400, Norm: 14.968647

Iteration 500, Norm: 14.786226

Iteration 600, Norm: 14.664949

Iteration 700, Norm: 14.578590

Iteration 800, Norm: 14.514174

Iteration 900, Norm: 14.464490

Total execution time (T): 10.326312 seconds

Max norm after 1000 iterations: 14.425558

mpiP:

mpiP: Storing mpiP output in [./work7.4.120339.1.mpiP].

mpiP:

PS:

Read file <work7.1205998.err> for stderr output of this job.

[edu-cmc-sqi22-29@polus-ib sem07]\$

P=8

Iteration 0, Norm: 117.466912

Iteration 100, Norm: 12.846925

Iteration 200, Norm: 11.542179

Iteration 300, Norm: 11.095953

Iteration 400, Norm: 10.879834

Iteration 500, Norm: 10.754660

Iteration 600, Norm: 10.673772

Iteration 700, Norm: 10.617641

Iteration 800, Norm: 10.576751

Iteration 900, Norm: 10.545930

Total execution time (T): 2.959509 seconds

Max norm after 1000 iterations: 10.522345

P=12

Iteration 0, Norm: 96.273998

Iteration 100, Norm: 10.518011

Iteration 200, Norm: 9.434790

Iteration 300, Norm: 9.066519

Iteration 400, Norm: 8.892407

Iteration 500, Norm: 8.793427

Iteration 600, Norm: 8.730321

Iteration 700, Norm: 8.687108
Iteration 800, Norm: 8.656117
Iteration 900, Norm: 8.633160
Total execution time (T): 2.118885 seconds
Max norm after 1000 iterations: 8.615898

P=16

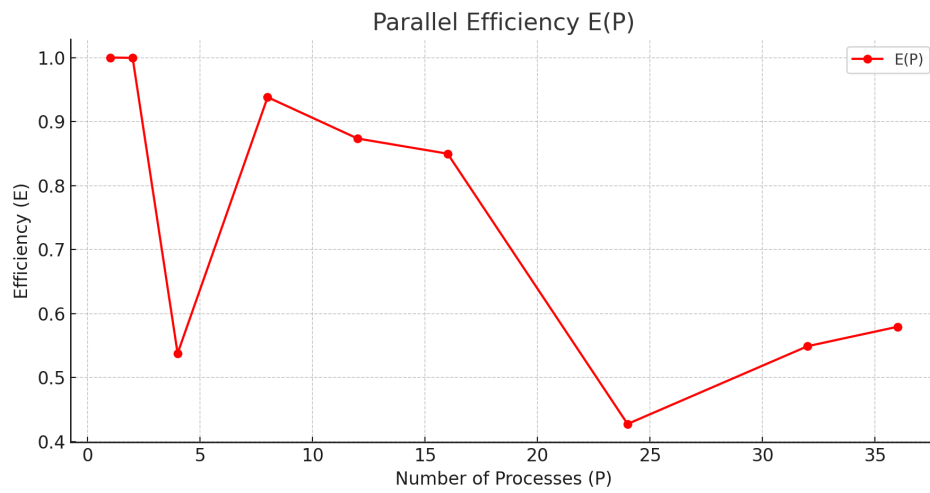
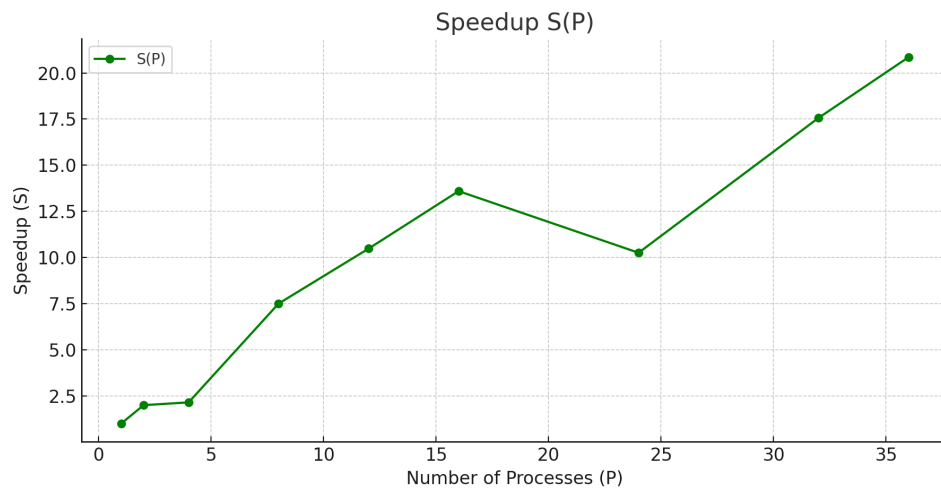
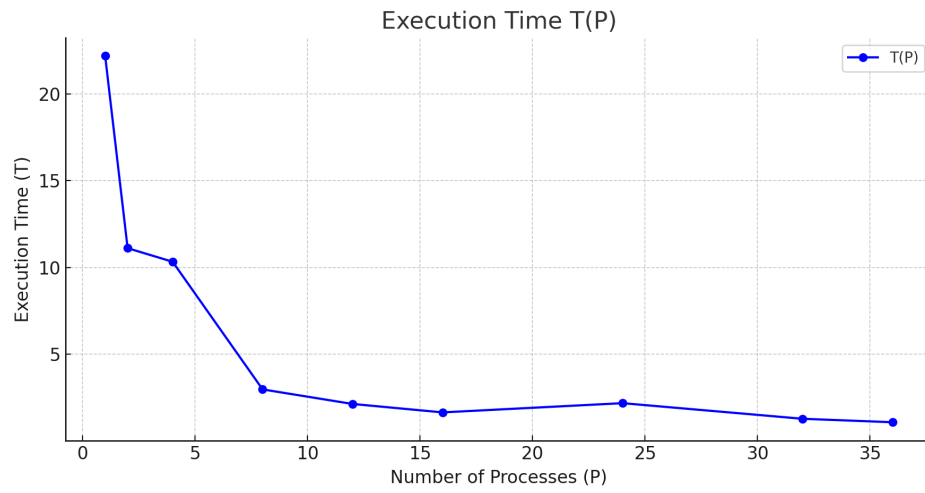
Iteration 0, Norm: 83.160239
Iteration 100, Norm: 9.032222
Iteration 200, Norm: 8.170505
Iteration 300, Norm: 7.890680
Iteration 400, Norm: 7.760372
Iteration 500, Norm: 7.685184
Iteration 600, Norm: 7.635591
Iteration 700, Norm: 7.600252
Iteration 800, Norm: 7.573962
Iteration 900, Norm: 7.553934
Total execution time (T): 1.633433 seconds
Max norm after 1000 iterations: 7.538623

P=24

Iteration 0, Norm: 68.301926
Iteration 100, Norm: 7.245498
Iteration 200, Norm: 6.562285
Iteration 300, Norm: 6.340974
Iteration 400, Norm: 6.238502
Iteration 500, Norm: 6.178467
Iteration 600, Norm: 6.137202
Iteration 700, Norm: 6.105859
Iteration 800, Norm: 6.080564
Iteration 900, Norm: 6.059375
Total execution time (T): 2.165094 seconds
Max norm after 1000 iterations: 6.041378

p=32

Iteration 0, Norm: 59.026604



Iteration 100, Norm: 6.147985

Iteration 200, Norm: 5.632218

Iteration 300, Norm: 5.478070

Iteration 400, Norm: 5.409192

Iteration 500, Norm: 5.370706

Iteration 600, Norm: 5.346143

Iteration 700, Norm: 5.329205

Iteration 800, Norm: 5.317016

Iteration 900, Norm: 5.308064
Total execution time (T): 1.264377 seconds
Max norm after 1000 iterations: 5.301516

P=36

Iteration 0, Norm: 56.206832
Iteration 100, Norm: 5.647714
Iteration 200, Norm: 5.053677
Iteration 300, Norm: 4.851214
Iteration 400, Norm: 4.758592
Iteration 500, Norm: 4.709112
Iteration 600, Norm: 4.680301
Iteration 700, Norm: 4.662726
Iteration 800, Norm: 4.651731
Iteration 900, Norm: 4.644751
Total execution time (T): 1.064949 seconds
Max norm after 1000 iterations: 4.640325

6. Заключение

С увеличением числа процессов время выполнения уменьшается, ускорение растет, но параллельная эффективность демонстрирует тенденцию к снижению. Это соответствует типичному явлению в параллельных вычислениях — эффекту убывающей отдачи от параллелизма.

При увеличении числа процессов коммуникационные затраты и время ожидания синхронизации могут возрасти, что приводит к снижению прироста эффективности и увеличению времени выполнения. Особенно заметен аномальный скачок времени выполнения при 24 процессах, что, возможно, связано с дополнительными затратами на коммуникацию.

При использовании более 16 процессов параллельная эффективность начинает значительно снижаться, что указывает на ограниченность эффективности параллельного вычисления для данной задачи при увеличении числа процессов. Это означает, что при большом числе процессов параллелизация может перестать давать существенные преимущества.