Assignment Report #3
Running Jobs on Shabyt HPC Cluster

Student Name: Dias Suleimenov (ID: 202158836)
Course: PHYS 421 Parallel Computing

Submitted: October 2, 2025

AI tools used:

Gemini 2.5 Pro	Generation of longtable

Abstract

Parallel Monte Carlo simulation on HPC Cluster Shabyt was made. Speedup efficiency was measured. With all 64 cores produced 10.5 times speedup with 16.4% efficiency.

1 Introduction

The previous parallelization assignment was performed on HPC cluster Shabyt. The SLURM was used to run the parallelized Monte Carlo simulation.

2 Results

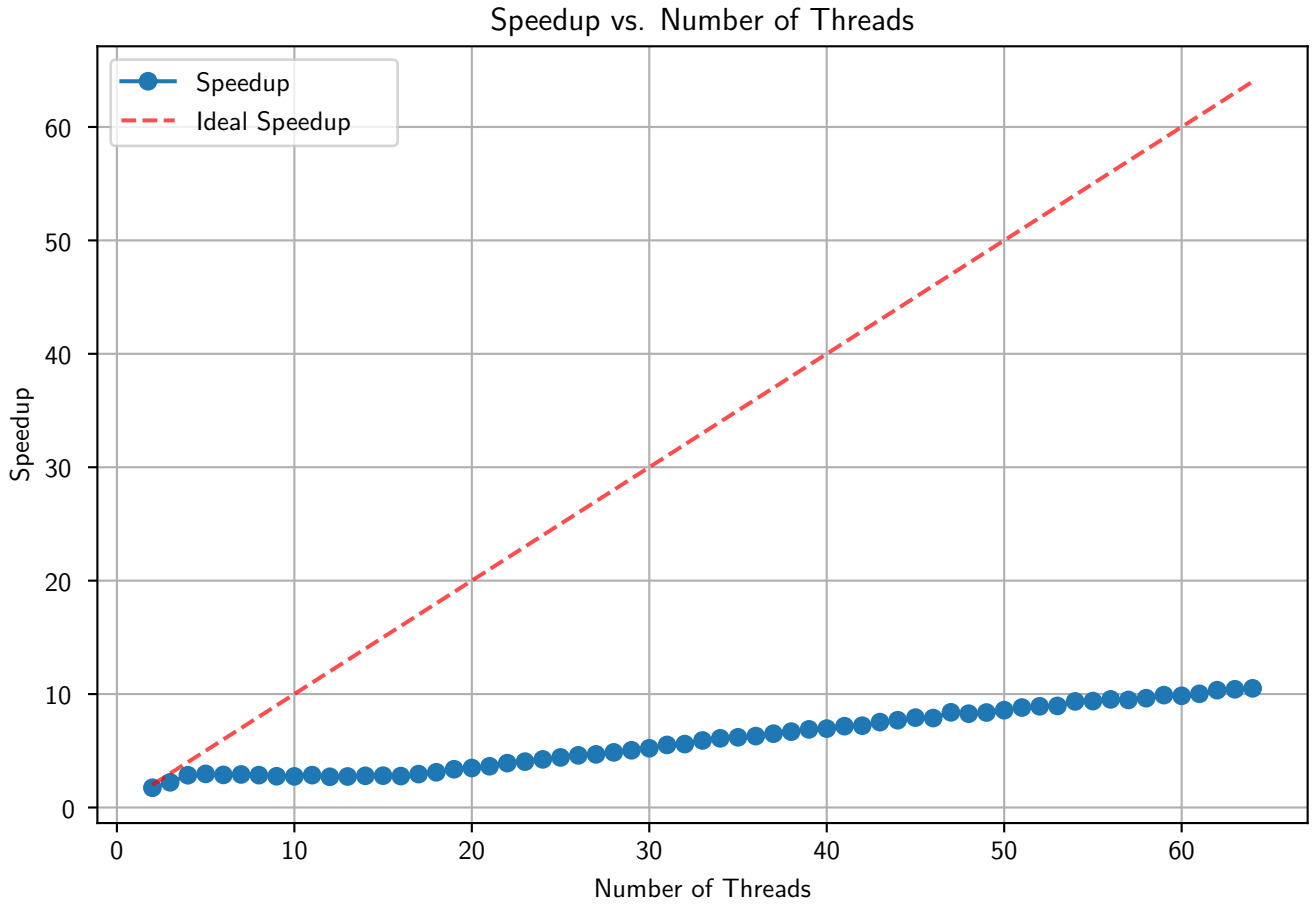

Figure 1: The speedup factor versus the number of threads for sphere with dimension $n = 10$, norm $p = 4$ and radius $R = 1$, number of samples $N = 4 \times 10^7$ and number of threads varying from 1 to 64.

Table 1: The speedup factor and efficiency in relation to number of threads for sphere with dimension $n = 10$, norm $p = 4$ and radius $R = 1$, number of samples $N = 4 \times 10^7$

Threads	**Serial Time (s)**	**Parallel Time (s)**	**Speedup**	**Efficiency**
2	25.146751	14.453392	1.739851	0.869926
3	25.146751	11.337518	2.218012	0.739337
4	25.146751	8.846620	2.842526	0.710632
5	25.146751	8.490713	2.961677	0.592335
6	25.146751	8.752003	2.873257	0.478876
7	25.146751	8.638105	2.911142	0.415877
8	25.146751	8.819301	2.851331	0.356416
9	25.146751	9.134567	2.752922	0.305880
10	25.146751	9.172859	2.741430	0.274143
11	25.146751	8.816778	2.852148	0.259286

Continued on next page

Table 1 – continued from previous page

Threads	Serial Time (s)	Parallel Time (s)	Speedup	Efficiency
12	25.146751	9.289585	2.706983	0.225582
13	25.146751	9.199840	2.733390	0.210261
14	25.146751	9.006661	2.792017	0.199430
15	25.146751	8.964772	2.805063	0.187004
16	25.146751	9.080043	2.769453	0.173091
17	25.146751	8.541177	2.944179	0.173187
18	25.146751	8.095705	3.106184	0.172566
19	25.146751	7.459109	3.371281	0.177436
20	25.146751	7.226864	3.479622	0.173981
21	25.146751	6.918348	3.634791	0.173085
22	25.146751	6.430593	3.910487	0.177749
23	25.146751	6.216476	4.045178	0.175877
24	25.146751	5.931276	4.239687	0.176654
25	25.146751	5.697653	4.413528	0.176541
26	25.146751	5.470770	4.596565	0.176791
27	25.146751	5.369477	4.683277	0.173455
28	25.146751	5.176837	4.857551	0.173484
29	25.146751	4.986053	5.043418	0.173911
30	25.146751	4.816356	5.221115	0.174037
31	25.146751	4.559502	5.515241	0.177911
32	25.146751	4.500491	5.587557	0.174611
33	25.146751	4.256846	5.907367	0.179011
34	25.146751	4.124094	6.097521	0.179339
35	25.146751	4.064498	6.186926	0.176769
36	25.146751	3.995402	6.293923	0.174831
37	25.146751	3.864555	6.507024	0.175866
38	25.146751	3.760271	6.687483	0.175986
39	25.146751	3.651326	6.887019	0.176590
40	25.146751	3.616014	6.954274	0.173857
41	25.146751	3.507699	7.169017	0.174854
42	25.146751	3.487416	7.210712	0.171684
43	25.146751	3.340689	7.527415	0.175056
44	25.146751	3.268198	7.694378	0.174872
45	25.146751	3.174569	7.921312	0.176029
46	25.146751	3.189499	7.884232	0.171396
47	25.146751	2.996316	8.392557	0.178565
48	25.146751	3.040494	8.270612	0.172304
49	25.146751	3.003876	8.371435	0.170846
50	25.146751	2.931042	8.579457	0.171589

Continued on next page

Table 1 – continued from previous page

Threads	Serial Time (s)	Parallel Time (s)	Speedup	Efficiency
51	25.146751	2.853859	8.811489	0.172774
52	25.146751	2.819148	8.919982	0.171538
53	25.146751	2.808205	8.954743	0.168957
54	25.146751	2.688688	9.352796	0.173200
55	25.146751	2.680779	9.380389	0.170553
56	25.146751	2.638331	9.531311	0.170202
57	25.146751	2.652672	9.479782	0.166312
58	25.146751	2.609536	9.636482	0.166146
59	25.146751	2.538349	9.906734	0.167911
60	25.146751	2.554390	9.844524	0.164075
61	25.146751	2.509214	10.021763	0.164291
62	25.146751	2.431288	10.342976	0.166822
63	25.146751	2.413230	10.420371	0.165403
64	25.146751	2.393391	10.506744	0.164168

```
115.318963200000 115.327953463827 25.146750914864 0.008990263827
115.340032000000 115.327953463827 14.453392296098 0.012078536173
```

Listing 1: Example output from the parallel Monte Carlo simulation run.

3 Discussion

As can be see from the Figure 1 and Table 1, the speedup efficiency is suboptimal. The maximal speedup was utilizing all 64 cores producing 10.5 times speedup with 16.4% efficiency. The inefficiency might possibly come from the parallelization overhead, memory bandwidth limitations and cache competition.

4 Conclusion

The Monte-Carlo simulation was repeated on Shabyt cluster and the speedup factors of parallelization were measured. The best method was using all 64 cores producing 10.5 times speedup with 16.4% efficiency.