HPC Parallization

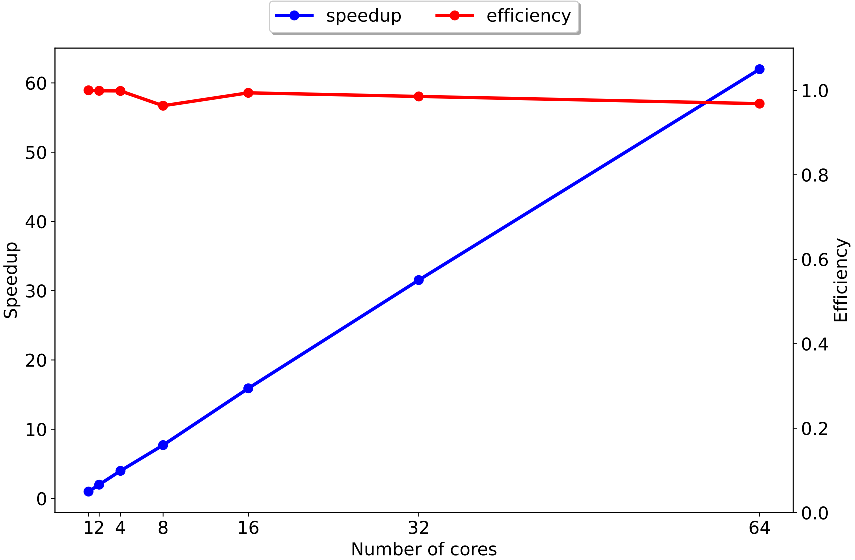
Assignment 02

Bernhard Kriesche, Josef Gallmetzer, Georg Gallmetzer, Jakob Gamper

May 4, 2023

Department of General, Inorganic and Theoretical Chemistry
University of Innsbruck

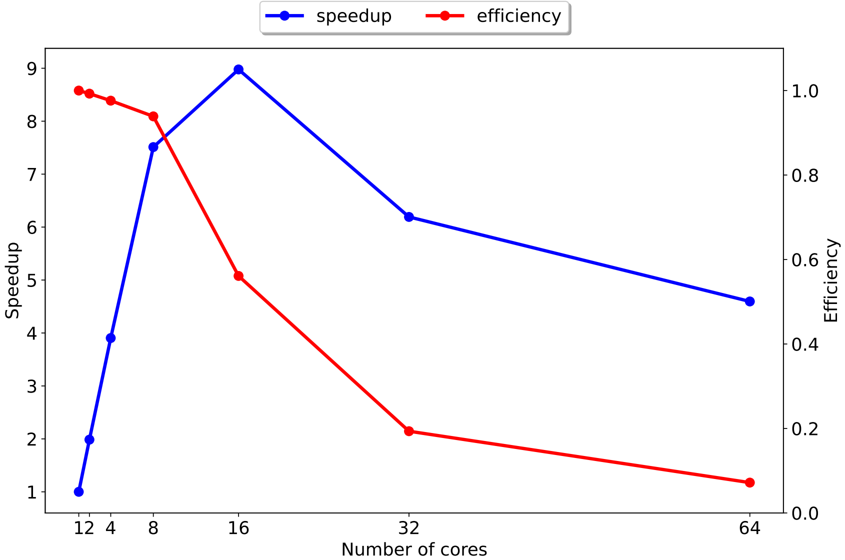
Monte-Carlo Pi Figure



Monte-Carlo Pi Data

cores	Time / s	speedup	efficiency
1	86.410	1.000	1.000
2	43.252	1.998	0.999
4	21.634	3.994	0.999
8	11.210	7.708	0.964
16	5.433	15.905	0.994
32	2.740	31.536	0.986
64	1.394	61.983	0.968

Heat Stencil



Heat stencil Data

cores	Time / s	speedup	efficiency
1	6.644	1.000	1.000
2	3.347	1.985	0.993
4	1.702	3.904	0.976
8	0.885	7.511	0.939
16	0.740	8.978	0.561
32	1.073	6.192	0.193
64	1.446	4.596	0.072

Interpretation of the results

- Compilation flag -O3 was fastest
- Monte-Calro achieves almost linear speedup until 64 cores
- In MC, there are no data-dependencies $\Rightarrow$ MC usually compute bound
- Stencil code achieves almost 80% efficiency until 8 cores
- With more cores, the efficiency plummets, even run time increases
- In heat stencil code, for every cell, neighboring cells need to accessed $\Rightarrow$ data dependencies
- Stencil codes are usually memory bound

Stability of measurements MC Pi

Measurement no.	time / s
1	1.659868451
2	1.479569032
3	1.430867489
4	1.420867624
5	1.409361594
6	1.849449042
7	1.411459893
8	1.449916326
9	1.400285332
10	1.617293296
mean = 1.51289	**sd = 0.141327**

Stability of measurements heat stencil

Measurement no.	time / s
1	1.74461
2	1.3345
3	1.38557
4	3.5005
5	1.35053
6	1.3578
7	1.41246
8	1.47691
9	1.53155
10	1.41435
mean = 1.65088	**sd = 0.627197**