

Exercise Sheet 2

Exercise A) External CPU load

Experiment setup:

Überordner mit:

- ordner "build", der die kompilierten small samples sowie den problems ordner für qap
- kompiliertes loadgen tool (buildfiles direkt im ordner)
- config file (yaml) mit struktur wie "benchmark_config.yaml"

Um die Messungen am Lcc3 durchzuführen, muss man nur das start_benchmarks.sh mit sbatch starten. Dies führt das dispatch_benchmarks.py script aus, welches jobscripts für die in der config definierten Programme generiert, sowie run_benchmark.py für jedes Programm (via sbatch) startet. Dieses subscript Falls das Confidenceinterval für alle metriken erreicht wurde, dann werden die Iterationen früher abgebrochen. Die Ergebnisse werden für jedes definierte Programm in ein seperates JSON file geschrieben. Der workload generator kann in der config mit dem feld 'load' ein- und ausgeschaltet werden. Will man die benchmarks lokal laufen lassen, dann muss man einfach im YAML file den mode string auf "local" setzen ansonsten "lcc3", das fuktioniert allerdings nur ohne extra workload.

Messungen

Exercise	Parameter	Mean Wall Clock Time	Mean CPU Time	Mean System Time	Mean Max Memory Usage	Variance Wall Clock Time	Variance CPU Time	Variance System Time	Variance Max Memory Usage
delannoy	12	3.79	0.785	0.0	1379.2	3.0731	0.0096056	0.0	674.8444
filegen	15 15 1024000 1048576	26.541	4.127	0.068	2487.2	14.007	0.0558678	0.00004	3615.2889
filesearch		0.151	0.0	0.001	1440.0	0.0875656	0.0	0.00001	2709.3333
mmul		18.959	3.489	0.005	24584.4	211.2275	6.0161211	0.00005	3806.0444
nbody		15.632	2.794	0.0	1850.0	8.5394	0.00076	0.0	6671.1111
qap	'./build/problems/chr15a.dat'	25.487	4.538	0.0	1498.8	43.6469	0.1047956	0.0	316.6222