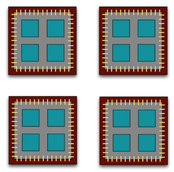
Parallelization in Multiple Imputation

Sven Nekula, Joshua Simon and Eva Wolf
Otto Friedrich University, Bamberg

What is Parallelization?

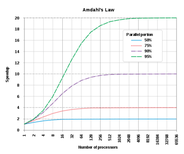


Parallelization is a technique to fasten time-consuming computations. It uses all the cores on a CPU (Central Processing Unit) parallely and splits up the computational work on them. Afterwords, the results are merged. This can reduce the time needed for a task.

Theory

Parallel Computing

In parallel programming, the multiple cores of a computational system want to be used best to decrease computation time. Gene **Amdahl** was the first one to desribe the boundaries of that project: Every parallel process also requires additional workload, so called "data management housekeeping". The speed up through parallel processing will tend to 0 at a certain amount of processing units involved, as this **overhead** workload exceeds the capacity of the computational unit which it is assigned to.



Multiple Imputation

Multiple imputation is a method to complete a dataset with missing information. It relies on the estimation of the missing values through different methods. What is common to them is that we use not a single imputation run, but several. The results of all imputation runs are then merged and lead to realistic uncertainty of the estimators given the missing data.

Methodology

data generation We used a simple data generator of normally distributed random variables. The data set created contains 10 variables, of which some depend on each other. The sample size created was n 10000. Expermients were compared to euqal settings with real data, and the results did not change the direction of the results. More complex data favor the mechanism of parallelization.

time measurement was done with the *system.time* function, which returns 3 values: User CPU-, User System- and Elapsed time. User CPU is the time needed by the current task such as an execution in R. System CPU describes the time needed by the operating system to organize that task such as opening folders or asking for the System time. Elapsed time is the Wall Clock time that passed while the function was running.

mice method The method of the mice algorithm was set on defaut, pmm.

speed up The speed_up value is calculated by the serial time (runtime without parallelization) divided by the runtime of the current parallelization implementation [**cite:chapple2016mastering**]. Values below one show that the parallelized run took longer than the serial run.

Results

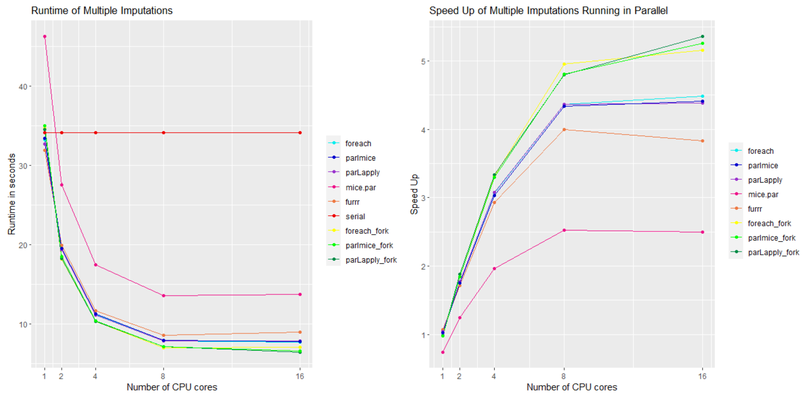


Fig. 1: Runtime and Speedup from 1 up to 8 cores

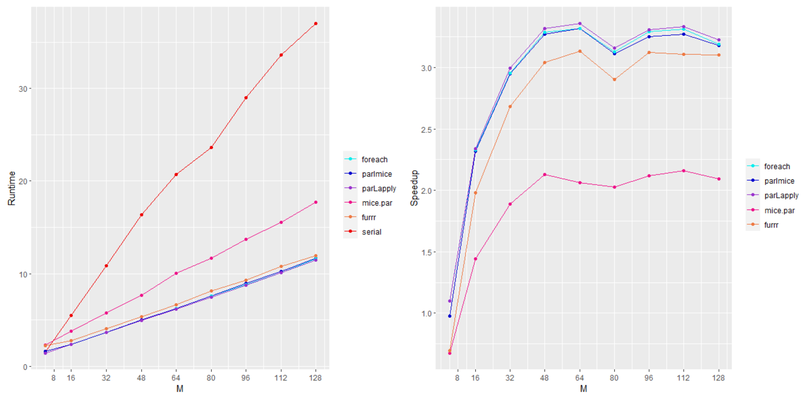


Fig. 2: Runtime and Speedup from 1 up to 8 cores

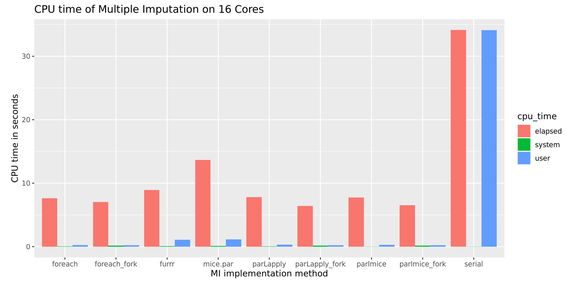


Fig. 3: Runtime and Speedup from 1 up to 8 cores

Comparison of Implementations

foreach::foreach usability ★★★★★ runtime★★★★★

mice::parlmice usability ★★★★★

micemd::mice.par usability ★★★★★

parallel::parLapply usability ★★★★★

purrr::future_map usability ★★★★★

Application

For **disk framing**, parallelization is a useful tool to decrease computation time. Disk framing seperated larger-than-RAM datsets into several chunks of data, which are then treated seperately for the task to perform. Afterwords, results are merged again. With the help of parallelization, chunks of data can be processed parallely and so calculation time is reduced.

References