Optimization of PMetrics

To enable large and complex population pharmacokinetic modelling.

SIGMA2 AUS Final Report: Optimization of PMetrics (NN9736K)

Date: 28 May 2020

Sabry Razick (SR) *Ph.D.*, Department for Research Infrastructure Services, University of Oslo
Frode Strømsvåg (FS) *Cand.Scient*, Department for Research Support Services, University of Oslo
Markus Herberg Hovd (MH) *M.Sc. Pharm*, Department of Pharmacy, University of Oslo

---

Overall summary

The Optimization of PMetrics, Advanced User Support (AUS) project successfully achieved the stated objectives in four stages, allowing for larger and more complex pharmacokinetic models to be run on the HPC infrastructure. During the first stage, wrappers were developed for the software to be executed in headless-batch mode. In the second stage, the pipeline was modified to accept larger Input data, which was the main limitation faced by the requesters. Third stage the processing time was reduced by using Intel® Math Kernel Library and making parallel execution possible using OpenMP technology. In the fourth stage scalability was evaluated and optimal resource allocation was decided for the Saga computer cluster. An overall 15 times speed improvement over the previous version was recorded in this final version.

The work was carried out over a little more than 4 months and with 1 PMs effort, during a time of transitioning from Abel to Saga and impacted by the recent pandemic. The AUS provides the end users with improved infrastructure, shifting the workload from personal workstations and laptops to a HPC-cluster.

Initial request

(By MH). PMetrics (http://www.lapk.org/pmetrics.php) is R package that is easy to use, somewhat customizable, and integrates with R. It is usually run on a personal or UiO-computer. PMetrics mostly uses gfortran to compile its packages, but is also compatible with ifort, g95, and more. Recently our models have gotten quite complicated, and we have more patient data than we are able to process in PMetrics due to size limitations. The biggest showstopper for us is this error;

“relocation truncated to fit: R_X86_64_PC32 against symbol …”.
which I assume is due to creating an array >2GB.

PMetrics has support for inputting custom fortran statements, but as I stated, fortran or fortran environments are not one of our key strengths. I have attempted to install and configure ifort (in hope that it would work differently), and use freebee.abel.uio.no with module ifort loaded, but to no success.

Is this somewhat intangible problem something you could possibly assist us with?

Initial observations and recommendations.

Pmetrics(http://www.lapk.org/pmetrics.php) is an R package for parametric and non-parametric modeling and simulation of pharmacokinetic and pharmacodynamic systems. The software is designed for a system with graphical user interfaces. So the procedures included user interactions and pop-ups. This had to be adopted to a head-less-batch mode, for the investigation to begin.
The Pmetrics workflow has several sequential steps.

1. Install the R package and perform an initial setup process. If a fortran compiler is not installed, this must be installed as well. PMetrics recommends gfortran. The setup includes calling the R function PMbuild(). This compiles Fortran code distributed along with the R package. This supports several fortran-compilers.
2. The main function used in the analysis is NPrun. This takes several inputs:
    a. A data file
    b. A model file
    c. Number of cycles
    d. Indpts (Index of starting grid point number)
3. The function fabricates a model in fortran then compiles it
4. Use data, a model file and number of cycles as inputs and fabricate a Fortran model file using R code.
5. Compile the fabricated model
6. Run the model

7. PMetrics creates a report on the results, and allows for the inspection and manipulation of output in R.

Initial suggestion was to rewrite or modify the fortran part of the software so that it can handle larger files.

Objectives of AUS application:

1. Optimize PMetrics' fortran routines in order to allow for larger and more complex pharmacokinetic models to be run.
2. Optimize PMetrics parallel processing capabilities in order to increase the speed of modelling.

Deliverables

Work hours	Modification	Improvement achieved	Date
1 (SR)	Create Github repository for collaborative development	Collaborative development and issue tracking. https://github.com/Sabryr/Pmetrics_test	Jan 3, 2020
20 (SR, MH)	Set up headless mode on Abel, create test data and testing.	Perform analysis without user intervention and pop-ups	Jan 7, 2020
20 (FS)	Analyse Fortran part of the software to identify a method to overcome limitations	Found compiler options and how to inject them to the software	Jan 17, 2020
10 (SR, FS)	Overcome 2Gb, 32 bit limitation. Fix "relocation truncated to fit: R_X86_64_PC32	Data consists of pharmacokinetic observations for multiple subjects, with the aim of modelling population pharmacokinetic models. Models have begun to include many patients, and grow increasingly complex. Thus, more optimized tools are necessary. Indicators for improved optimization are increased amount of maximum indpts, and running time. Improvement will lead to a significant increase in research output.	Jan 20, 2020
5 (SR, MH)	Set up environment on SAGA and testing	Requirement, Abel decommissioned,	Feb 10, 2020
30 (SR)	Solve issues related to newer compilers on SAGA by setting up a conda environment. This was not anticipated during AUS agreement.	Ability to process large files on SAGA (XXX include number of subjects etc..). Provide conda setup to provide the exact compiler versions and other libraries. Manually list all libraries so that they can be compiled with exact compiler options. https://github.uio.no/sabryr/Pmetrics_test/blob/master/environment.yml	Mar 18, 2020

5 (FS)	Locate the issue that broke the software when using Indpts=100	Avoid that indpts value	April 16, 2020
20 (SR)	Create a fork of original software to make the changes yet to be included in main release, available for testing	https://github.com/Sabryr/Pmetrics	Apr 23, 2020
5 (SR)	Introduce Intel® Math Kernel Library	Reduce processing time by half, all other variables kept intact	April 20, 2020
5 (SR)	Contribute to original code base through a pull request to make it possible to use intel compiler	The intel compilers on SAGA failed to compile the code due to old conventions used in the code.	Not delivered
10 (SR)	Introduce OPenMP	Enable parallel processing. Reduce runtime by 6 times, compared to step 6	May 2, 2020
10 (SR, MH)	Test scalability	Found that the performance increase with number of cores provided up to a limit and then start to decline (see plot xx)	May 4, 2020
5 (SR)	Produce HTML report	The original code failed the report generation on SAGA as the Latex generation part failed. The issue was solved by providing the HTML version and provide a Latex version where the table alignment is not optimal.	May 12, 2020
10 (SR, MH)	Package software to be easily usable by end user and document usage	User adaptability https://github.uio.no/sabryr/Pmetrics_test/blob/master/README	May 15, 2020
5 (SR, FS, MH)	Reporting	This report	May 22, 2020
161 Total work hours			

Details of optimizations

1. The code was mainly designed for a system with graphical user interfaces. So the procedure included user interactions and pop ups. This had to be adopted to a head-less-batch-mode for the work to begin. SR, disassembled the R code and made this possible. Subsequently he separated the Fortran code from R and provided a test bed for FS so he can investigate the Fortran code to overcome the limitations the user is facing.
2. FS, performed a deep investigation of the Fortran part of the code and came up with compiler flags that could make it possible to overcome the data size limitations the user was facing .
3. SR repackaged the R and Fortran components and created bash scripts to manually inject the compiler options
4. SR created a repository by forking the original repository and provided access to changes yet to be accepted by the original authors. The main changes are to make it possible to manually compile the generated Fortran code and to make the analysis parallel.
5. FS investigated and found solutions for components that broke the workflow when moving from Abel to Saga. Added the bug triggering cycle values to known errors.
6. SR created a conda based runtime environment on SAGA to provide a consistent and reproducible runtime environment
7. SR optimized the runtime performance by introducing Intel® Math Kernel Library (Intel® MKL).
8. SR introduced OpenMP (libgomp) to enable parallel processing. Perform benchmarks to verify the parallel code execution.

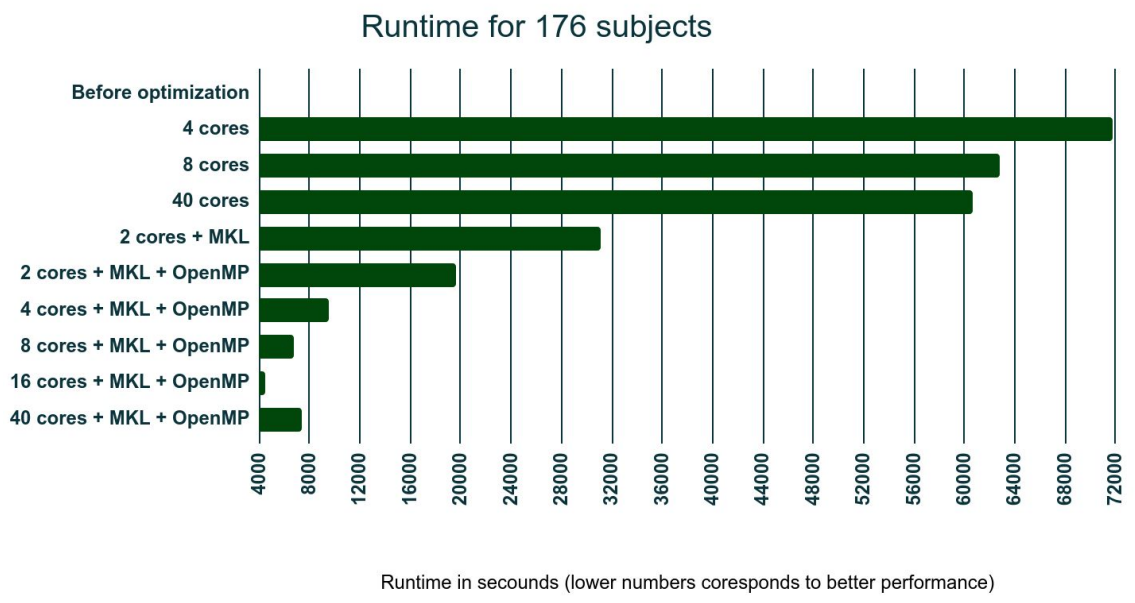


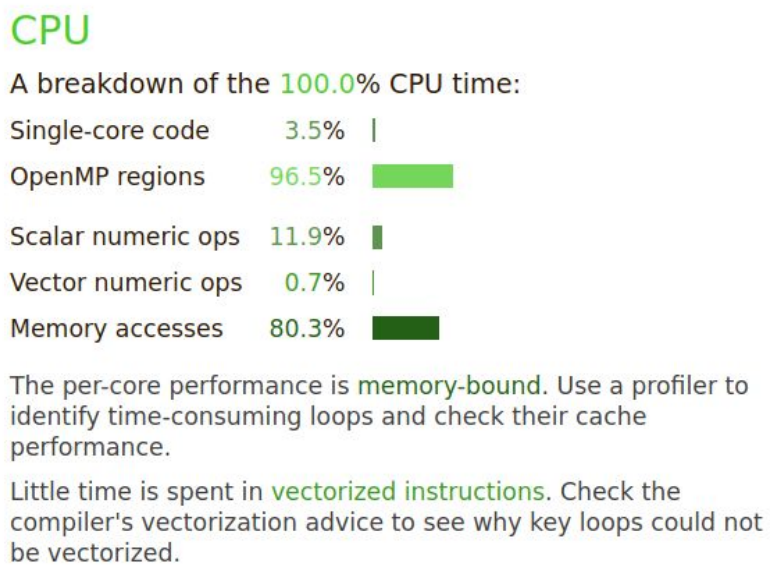
Fig 1: Plot showing the performance increase (lower numbers are better), with different levels of optimization. There are no numbers before the optimization as for the provided data set, the software would not complete the processing on Abel or Saga compute clusters. After the data set limit was removed the software showed very little performance increase with additional cores. Introduction of MKL (Intel® Math Kernel Library) half the runtime, but there was still no further performance increase with additional cores. Therefore OpenMP parallelization was included to achieve scalability. There was a performance increase up to 16 cores for the provided model with 176 subject data set and then a decline with additional cores due to the overhead of provisioning additional cores exceeding the performance gain.

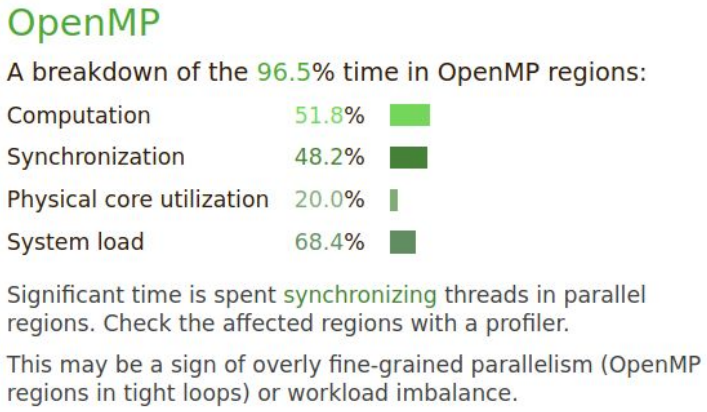
ARM Performance Reports

(full report as appendix)

From the report it shows that the current software setup is Compute-bound and capable of using multiple cores efficiently. Software was not tested using MPI so there is no evaluation of multi node configuration. The IO and memory footprint is minimal. As drawbacks, the software is accessing main memory frequently and does not use cache memory efficiently. In addition, compiler level vectorization of loops is not optimal. Improving those shortcomings will not be addressed as part of this AUS.

As things stand, authors wish to propose this software as suited for a SAGA compute cluster using a single compute node. Suitability to be deployed in Fram or Betzy compute clusters is minimal, and no attempt will be taken to improve that as part of this AUS.





User experience

Objective 1 - Overcome size limitation
This objective was the primary goal of the Advanced User Support. This was successfully accomplished, and will enable our research group to employ larger and more complex models on the supercomputer.

Objective 2 - Decrease run times
Model run times seem to be drastically lowered, and a suggestion for optimal slurm options was provided by the AUS team. This allows us to spend less time figuring out which settings to tweak, and spend more time on research.

Usability
The modified version is easily used, and provides a simple framework for installation, which is much appreciated.
Most of the parameters available in R are accessible through the supplied job-script, allowing for easy use and possibly automation of different models.

Remarks
The AUS has been of great value to our research group, and has allowed us to overcome obstacles that we previously couldn't. We are now able to run multiple, more complex models at a HPC, which is a drastic upgrade to our infrastructure.

Summary
Overall we are very grateful for the results from this AUS, which as previously mentioned has drastically increased our research capabilities due to improved infrastructure.

Future aspects
In order to facilitate usage of the HPC for users that are less familiar with UNIX-based systems and slurm, a wrapper that allows for submissions from personal laptops to the HPC-cluster would be of value. This could be written by the research group, due to the ease of use of the supplied job-scripts.

Another possible addition would be the automation of model testing, e.g. based on select criteria, the models would be modified and re-run. However, this is far outside of the scope of this AUS and very complex, but is potentially possible with the new infrastructure and setup.