Complex Transport Simulations*

K. Gallmeister[1]

[1]Institut für Theoretische Physik, Johann Wolfgang Goethe - Universität Frankfurt, Max-von-Laue-Str. 1, D-60438 Frankfurt am Main, Germany

In order to understand and analyze current and future experiments in modern nuclear and hadron physics, a number of very complex computer programs based on transport theoretical models have been developed in the last years.

One prominent example is the program BAMPS[1], which is a important tool for the understanding of the dynamics of the hot and dense quark gluon phase in high energetic heavy ion collisions. This description is primarily relevant for the understanding of RHIC and LHC experiments. Here quarks and gluons are the relevant degrees of freedom. On the other side, hadronic models try to describe the same topic in terms of hadronic degrees of freedom, as, e.g. UrQMD[2, 3] or GiBUU[4].

Unfortunately, all these programs reside in different stages of programming techniques. The used programming languages range from Fortran 77 up to C++. Accordingly, the underlying program structures as modularity or object orientation differ drastically.

In addition, due to the different used languages, the use of standardized (and well tested) libraries is implemented in different levels. A very strong deficiency of these programs is the heterogeneous input/output strategy. It is not possible, to import immediately the results of one program into another in order to process them further.

Also the aspect of maintainability is diversified. A concept of version control is implemented at the moment only partially.

A very important aspect is the development of computer hardware during the last years. In order to cope with this, new programming techniques are necessary. Thus an adaption of the existing software and their development for multi and many-core architectures is very important.

At the moment we are using the program BAMPS as a test field for possible improvements.

It has shown up, that some cleaning up of the code, including restructuring of some of its elements was necessary to enhance the readability and maintainability. Especially, a clean implementation of the 4D vectors for position and momentum is essential. In addition, performing Lorentz transformations is one of the bottlenecks of all transport codes. Here an implementation in terms of SSE intrinsics has been proven to be very successful (cf. fig. 1). A speedup in the overall computation time up to 10% was achieved contrary to standard implementations, combined with a tremendous increase of readability of the code.

The transport codes mentioned above have some underlying structure, which contradicts in general some (naive) parallelization. Thus we are currently working to use the modern hardware architectures in the framework of GPGPU programming. Here first steps have been done, to parallelize the calculation of complex particle-particle cross sections for the BAMPS simulation in the framework of OpenCL [5].

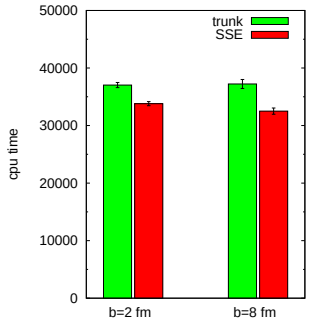

Figure 1: Comparison of the CPU time (with arbitrary normalization) of some sample configurations ($\mathrm{Pb}+\mathrm{Pb}$, RHIC conditions) for two impact parameter b for the original implementation ("trunk") and the new implementation using SSE3 instructions ("SSE") on the LOEWE CSC cluster.

While already some essential improvements have been reached, we are still in a learning phase to really figure out, where and how we can put the different codes on some common footing.

To summarize we state, that it is very important to bring the computer programs used by physicists in order to understand modern experiments to a stage, that corresponds modern programming knowledge and eliminate old coding techniques and languages. This will facilitate the use of the programs and allow their direct use by the experimentalist as a tool for planning, understanding and analyzing their experiments. We have started this effort successfully for the transport theoretical programs BAMPS, UrQMD and GiBUU.

References

[1] Z. Xu and C. Greiner, Phys. Rev. C **71**, 064901 (2005).

[2] S. A. Bass, M. Belkacem, M. Bleicher, M. Brandstetter, L. Bravina, C. Ernst, L. Gerland and M. Hofmann *et al.*, Prog. Part. Nucl. Phys. **41**, 255 (1998) [Prog. Part. Nucl. Phys. **41**, 225 (1998)].

[3] H. Petersen, J. Steinheimer, G. Burau, M. Bleicher and H. Stocker, Phys. Rev. C **78**, 044901 (2008).

[4] O. Buss, T. Gaitanos, K. Gallmeister, H. van Hees, M. Kaskulov, O. Lalakulich, A. B. Larionov, T. Leitner, J. Weil and U. Mosel, Phys. Rept. **512**, 1 (2012).

[5] http://www.khronos.org/opencl

* Work supported by GSI F&E.