PARALLEL IMPLEMENTATION OF PARTON STRING MODEL EVENT GENERATOR

S. Nemnyugin*, A. Asryan, G. Feofilov, P. Naumenko, V. Solodkov, V. Vechernin, A. Zarochencev, V. Zolotarev, Saint Petersburg State University, Ulyanovskaya str., 1, Saint Petersburg, Petrodvorets, Russian Federation 198504

Abstract

We report the results of parallelization and performance tests of the Parton String Model Monte Carlo event generator on the parallel "ALICE" cluster of Saint Petersburg State University. Two methods of parallelization were studied for different possible parallel cluster architectures. Some problems were successfully solved.

PSM MONTE CARLO EVENT GENERATOR

The Parton String Model event generator (PSM) is being successfully applied to the description of the large amount of experimental data on relativistic nucleus collisions [1]. It is based on the picture of the formation of color strings between the partons of the colliding nuclei and the following decay [2, 3]. With the increase of energy or mass of colliding nuclei the string density grows and the possible interaction of color strings should be taken into account by the introduction of string fusion or percolation. The PSM generator includes also the account of interaction of the final state particles (re-scattering) that is important for the proper description of the transverse momentum spectra and strange particles yields. Color strings are the objects extended in the rapidity space. It gives a possibility to observe the string decay products and study the correlations between various observables relevant to some sufficiently separated in rapidity windows (a so called long range correlations [4]). Study of long range correlations between such observables as mean transverse momentum and multiplicity of charged particles were recently proposed for the ALICE experiment at the future LHC [5] as a method of observation of color string fusion phenomenon.

The extensive model event-by-event PSM simulations require good statistics and are time consuming, especially with the account of re-scattering. That is why the parallel implementation of the PSM code was applied in this work. The modifications of the PSM code that were performed previously are also included in order to allow some additional possibilities: i.e. to select impact parameter windows, to include the experimental acceptance and trigger selection data if necessary, and to calculate various long range correlations between such observables as mean transverse momentum and charged particles multiplicity.

Below we present the main results of studies of two schemes of PSM event code parallelization. Comparison and benefits are described. Some problems of parallelization are also discussed. Results of the tests of parallel PSM code are presented.

Simulation with the PSM event generator consists of two stages. First one is getting "raw" output data with the PSM event generator. Then statistical processing of the output file should be done.

A lot of statistics (10^4-10^6 events) is required to get statistically reliable results, but simulation is time consuming (13.5 sec/event on 600 MHz processor) if all options of the model are turned on:

- hard gluon rescattering;
- string fusion;
- resonances decay;
- rescattering of secondaries sourced from strings breaking as well as secondaries and spectators.

Also it should be mentioned that file with results of simulation may be large: 1 GByte for moderate statistics of 25 000 events and 20 GByte for 500 000 simulated events.

PARALLEL PSM EVENT GENERATOR

Efficient parallel algorithm have to be tuned in the architecture of multiprocessor computing system, so we developed two parallel versions of the PSM event generator: (1) for symmetrical cluster and (2) for asymmetrical configuration. "Symmetry" here is symmetry on size of hard disks. The parallel codes were implemented with MPICH 1.2.4 library.

Parallel PSM event generator for clusters with large local disks

Parallel PSM event generator for symmetrical cluster configuration with local hard disks large enough to save files with results of simulation is developed following master-slave scheme [6] with distribution of events to be simulated between nodes (fig. 1). Master process coordinates work of slave processes, broadcasts input data. Results of simulation are saved in local files.

Benefit of this approach is that communication network is not overloaded, maximum scalability may be reached approaching theoretical limit of Amdahl's law. On the

*nemnugin@mph.phys.spbu.ru

other hand a lot of additional work should be done when postprocessing local output files.

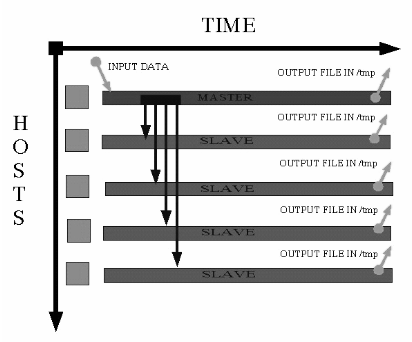

Figure 1: "Hosts-time" diagram of the parallel PSM event generator for a cluster without file server.

Parallel PSM event generator for clusters with file server

Parallel PSM event generator for a cluster with file server may be realized in different ways. One approach is based on the master-slave model with parallel write operations in the shared file. Statistical processing stage in that case is simplified, but communication network may be overloaded so scalability of the program may be significantly decreased.

In a compromise approach more efficient gathering operations are used instead of parallel output operations. “Hosts-time” diagram of the program is given on fig. 2.

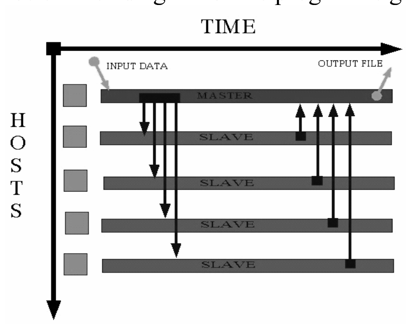

Figure 2: “Hosts-time” diagram of the parallel PSM event generator for a cluster with file server (upper host).

The compromise approach was taken for implementation.

Performance tests

Performance tests have been performed on “ALICE” cluster in Saint Petersburg State University. It consists of 7 two-processor hosts (600 MHz, 512 MB RAM) and server (1200 MHz, 256 MB RAM). Results of tests are given in fig. 3 and tables 1 and 2.

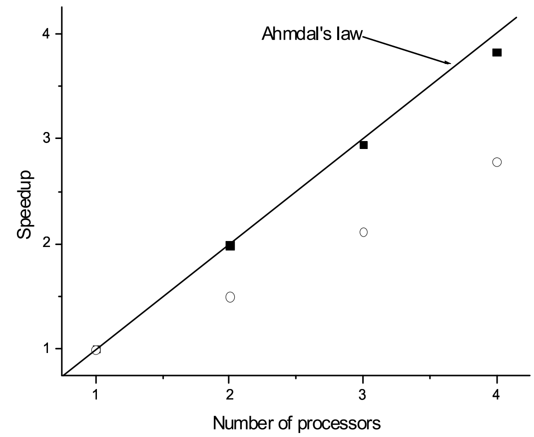

Figure 3: Speedup vs. number of processors diagram: black squares correspond to the first approach to parallelization and open circles to the second.

Speedup is defined by the formula:

$$Speedup = \frac{T_{exec}(1)}{T_{exec}(N)}$$

where $T_{exec}(N)$ is execution time of the program on N hosts.

Execution times are given in table 1 for the first approach to parallelization of the PSM generator for three sets of options of the Parton String model. Notation is following. Options (left to right): A) hard gluon rescattering; B) string fusion; C) resonances decay; D) rescattering. Letter **T** means that corresponding option is used and **F** – not used. Times are given in minutes and seconds.

Table 1: Performance test

Hosts	**TTTT**	**TTTF**	**TFFF**
1	224m21s	14m46s	4m35s
2	114m13s	7m59s	2m21s
3	87m59s	5m24s	1m35s
4	62m01s	4m16s	1m11s
5	55m07s	3m35s	0m58s
6	49m11s	3m09s	0m46s
7	44m32s	3m09s	0m42s

Table 2 presents execution times for the (1) first approach to parallelization and (2) second one. It may be seen that performance of the second version is less approximately by 30 percents.

Table 2: Performance test

Hosts	Time (1)	Time (2)
1	224m21s	225m28s
2	114m13s	149m56s
3	87m59s	105m26s
4	62m01s	81m14s

CONCLUSIONS

Parallelization and tests of the Parton String Model event generator are performed at the parallel cluster of Saint Petersburg State University communication center. Two schemes of parallelization were studied: (1) with the master process coordinating the uniform distribution of jobs at the cluster processors; (2) with the organization of the parallel processes writing the results into the same single file.

The modified parallel version of the PSM code includes a variety of additional possibilities that could be used for the direct comparison of results with the existing experimental data: selection of impact parameter windows, account of the acceptance of the experimental setup and trigger selection, long range correlations between such observables as mean transverse momentum and charged particles multiplicity.

The results could be used for the parallelization of other event generators and applied to the future GRID-based calculations.

ACKNOWLEDGEMENTS

This job was partially supported by the grants #1649 and #1286 of the Ministry of Education, Russian Federation.

REFERENCES

[1] N. S. Amelin, N. Armesto, C. Pajares and D. Sousa, Eur. J. Phys. C22 (2001) 149.

[2] M. Nardi and H. Satz, Phys. Lett. B442 (1998) 14; H. Satz, Nucl. Phys. A661 (2000) 104.

[3] M. A. Braun and C. Pajares, Eur. Phys. J. C16 (2000) 349; N. Armesto and C. Pajares, Int. J. Mod. Phys. A 15 (2000) 2019.

[4] M. A. Braun, C. Pajares and V. V. Vechernin. "On the forward-backward correlations in a two-stage scenario." Phys. Lett. B493 (2000) 54.

[5] P. Bolokhov, M. A. Braun, G. A. Feofilov, V. P. Kondratiev, V. V. Vechernin, "Long Range Forward-Backward pt and Multiplicity Correlations Studies in ALICE", ALICE Internal Note/PHY, ALICE-INT-2002-20, 24th June,2002, CERN.

[6] S. Nemnyugin, O. Stesik, Parallel Programming for Multiprocessor Computing Systems; BHV (2002), in Russian.