Technical Report CoSBi 08/2008

Study on an Optimistic Reaction-Diffusion Simulator based on Gillespie SSA

Lorenzo Dematté

The Microsoft Research - University of Trento
Centre for Computational and Systems Biology
DISI - Universitá degli studi di Trento

dematte@cosbi.eu

Tommaso Mazza

The Microsoft Research - University of Trento
Centre for Computational and Systems Biology

mazza@cosbi.eu

This is the preliminary version of a paper that will appear in proceedings of the sixth international conference on ***Computational Methods in Systems Biology (CMSB2008), LNBI 5307, pp. 191 – 210****, available at*
http://www.springerlink.com/content/w341771112012560/

Study on an Optimistic Reaction-Diffusion Simulator based on Gillespie SSA

Lorenzo Dematté Tommaso Mazza

September 28, 2009

Abstract

The parallel simulation of biochemical reactions is a very interesting problem: biochemical systems are inherently parallel, yet the majority of the algorithms to simulate them, including the well-known and widespread Gillespie SSA, are strictly sequential. Here we investigate, in a general way, how to characterize the simulation of biochemical systems in terms of Discrete Event Simulation. We dissect their inherent parallelism in order both to exploit the work done in this area and to speed-up their simulation. We study the peculiar characteristics of discrete biological simulations in order to select the parallelization technique which provides the greater benefits, as well as to touch its limits. We then focus on reaction-diffusion systems: we design and implement an efficient parallel algorithm for simulating such systems that include both reactions between entities and movements throughout the space.

Keywords: Parallel and distributed simulation, reaction-diffusion systems, Gillespie SSA

1 Introduction

In computational biology, the interest on multi-processor computing is growing over the years, even if ubiquitous and parallel computing require deep knowledge both on the bio-reality and on the tools in charge of handling and interpreting it. Indeed, the correct parallel computation of whatever problem must take into account four milestones: (i) the best computational splitting policy; (ii) how to handle synchronization among the computational workers, (iii) the more suitable hardware architecture and software packages to use and (iv) the nature of the inherent parallelism.

There are problems naturally parallelizable and others purely serial. According to the case, the additional computing power afforded by new machines can be used to advantage of one or of the other. To enhance the efficiency of Monte Carlo simulations, Single Replication in Parallel (SRIP) and Multiple Replications in Parallel (MRIP) computational paradigms have been widely contemplated in the past and deemed to be appropriate.

Single Replication in Parallel. The SRIP approach is based on the decomposition of a stochastic trajectory into logical processes, running on different processors and communicating by means of message passing protocols [4]. For naturally divisible problems, it shows elevated performances in speed-up and scale-up benchmarks. Significant drawbacks originate from the necessity for warranty of synchronism.

Multiple Replications in Parallel. The method speeds up simulation by launching independent replications on multiple computers and using different random seeds in such a way that the processes are approximatively uncorrelated. Therefore, more observations can be collected during a given time interval than running a single replication on one computer within the same period of time. Traditionally, one runs a simulation for a fixed time and then performs the data analysis [8]. When the accuracy defined by the user is reached, the simulation stops and a confidence interval is generated. If the numer of processes, the length of each replication and the deletion period are carefully chosen, the statistics will be valid [3]. In contrast to SRIP, MRIP can be easily applicable to any system, independent of the inherent system parallelism. However, the fact that a single replication cannot be executed on a unique processor and that outputs (or pieces of them) almost deterministic are identical when replicated, make the use of MRIP approaches sometimes inappropriate [6]. The MRIP and SRIP approaches are not exclusive, i.e., it is possible to use MRIP and SRIP in the same simulation program.

In biology, whereas the MRIP policy, well understood and investigated for a long time [3], [13], [5], [6], [7], [11], [17], [9], [1], [15], finds straightforward application to real case-studies [16], [2], the SRIP policy has a rather vague characterization. SRIP methods can be further divided into two opposite subcategories which include: (a) methods that exploit *data-parallelism* (or *loop-level parallelism*), namely that exemplify simulation of interacting particles on a finite grid in which individual processors are in charge of simulating the state of each site [14]; (b) methods that exploit *task-parallelism* (or *functional parallelism*), namely that divide the computation of a realization into a set of sub-computations among cooperative processors by computational dependency criteria [4], [12] (See Fig. 1 for a compact view of the parallel paradigms just described). To date, the research in distributed-parallel processing has successfully solved many related problems; however, it has not led yet to a portable and efficient tool for distributing stochastic simulation in the field of computational biology. We aim to move the attention of the reader toward our target by going through the theoretical basis and strategic decisions which configure our insight.

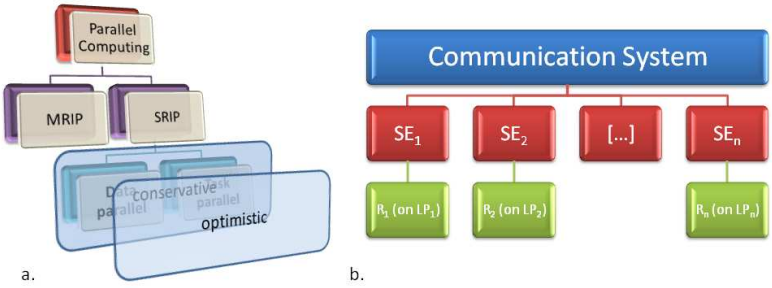

Figure 1: a. Parallel paradigms hierarchy. b. Model partitioning structure into Logical Processes and Simulation Engines

For the full content of this paper, please refer to its published version: *On Parallel Stochastic Simulation of Diffusive Systems*, In proceedings of the sixth International Conference on Computational Methods in Systems Biology (CMSB2008), LNBI 5307, pp. 191 – 210.

2 Conclusion and Future Work

One of the obstacles on the way of systems biology is the scalability of current approaches, i.e. their ability to deal with bigger and more complex models; these complex models are necessary to understand higher level behaviours, but need for both powerful modelling tools and efficient simulation engines to analyse them.

In this paper we tackled the problem of designing a parallel simulator for biochemical systems, based on the theory developed by Gillespie, from both a theoretical and a practical point of view. The design of parallel and distributed algorithms requires indeed both a strong theoretical background, in order to guarantee that the designed algorithm is equivalent to the serial one, and a good deal of practical tricks and experience in order to make it really scalable and efficient.

Here we presented some first steps in this direction; although the results we obtained so far are promising, a lot of work needs to be done. In particular, Jeschke et. al. [10] conducted a parallel research on the same topic, focusing on the analysis of communication costs and on sizing of the window for optimistic execution in a distributed grid environment. It will be interesting to incorporate their studies and analysis of the window size to our framework, to see which are the differences between their grid-based and our HPC based approaches. Other problems we need to face are the analysis of the obtained data, whose dimension grows at an impressive rate when dealing with spatial simulations, load-balancing techniques for workload subdivision and analysis of the rollback mechanisms on different biochemical systems. Finally, we would like to perform an in-depth study of the performances, with different checkpoint frequencies, different number of nodes, different policy of cell allocation between nodes and different state saving strategies.

References

[1] L. Bononi, M. Bracuto, G. D'Angelo, and L. Donatiello. Concurrent replication of parallel and distributed simulation. In *Proceedings of the 19th ACM/IEEE/SCS PADS Workshop*, pages 430–436, 2005.

[2] K. Burrage, P. M. Burrage, N. Hamilton, and T. Tian. Computer-intensive simulations for cellular models. In *Parallel Computing in Bioinformatics and Computational Biology*, pages 79–119, 2006.

[3] G. C. Ewing, D. McNickle, and L. Pawlikowski. Multiple replications in parallel: Distributed generation of data for speeding up quantitative stochastic simulation. In *Proceedings of the 15th Congress of Int. Association for Matemathics and Computer in Simulation*, pages 397–402, 1997.

[4] R. M. Fujimoto. Parallel discrete event simulation. *Comm. ACM*, 33(10):30–53, 1990.

[5] P. W. Glynn and P. Heidelberger. Analysis of parallel replicated simulations under a completion time constraint. *ACM TOMACS*, 1(1):3–23, 1991.

[6] P. W. Glynn and P. Heidelberger. Analysis of initial transient deletion for parallel steady-state simulations. *SIAM J. Scientific Stat. Computing*, 13(4):904–922, 1992.

[7] P. W. Glynn and P. Heidelberger. Experiments with initial transient deletion for parallel, replicated steady-state simulations. *Management Science*, 38(3):400–418, 1992.

[8] P. Heidelberger. Discrete event simulations and parallel processing: statistical properties. *SIAM Journal Stat. Comput.*, 9:1114–1132, 1988.

[9] M. Hybinette and R. M. Fujimoto. Cloning parallel simulations. *ACM TOMACS*, 11(4):378–407, 2001.

[10] M Jeschke, R Ewald, A Park, R Fujimoto, and AM Uhrmacher. Parallel and distributed spatial simulation of chemical reactions. In *Proceedings of the 22nd ACM/IEEE/SCS PADS Workshop*, 2008.

[11] Y-B. Lin. Parallel independent replicated simulation on a network of workstations. *ACM SIGSIM Simulation Digest*, 24(1):73–80, 1994.

[12] T. Mazza and R. Guido. Guidelines for parallel simulation of biological reactive systems. In *Proceedings of NETTAB 2008, Bioinformatics Methods for Biomedical Complex System Applications*, pages 83–85, 2008.

[13] M. E. J. Newman and G. T Barkema. *Monte Carlo Methods in Statistical Physics*. Oxford University Press, 2000.

[14] M. Schwehm. Parallel stochastic simulation of whole-cell models. In *Proceedings of ICSB*, pages 333–341, 2001.

[15] S. Streltsov and P. Vakili. Parallel replicated simulation of markov chains: implementation and variance reduction. In *Proceedings of the 25th conference on Winter simulation*, pages 430–436, 1993.

[16] T. Tian and K. Burrage. Parallel implementation of stochastic simulation for large-scale cellular processes. In *Proceedings of of Eighth International Conference on High-Performance Computing in Asia-Pacific Region*, pages 621–626, 2005.

[17] V. Yau. Automating parallel simulation using parallel time streams. *ACM TOMACS*, 9(2):171–201, 1999.