# Architecture of a Grid-enabled Lattice-Boltzmann Middleware

G. Dethier[1], P. Marchot[2] and P.A. de Marneffe[1]

[1] Department of Electrical Engineering & Computer Science

[2] Department of Chemical Engineering

University of Liège, Belgium

e-mails: {*G.Dethier, P.Marchot, PA.deMarneffe*}*@ulg.ac.be*

Lattice-Boltzmann (LB) simulation methods constitute a family of computational fluid dynamics methods that can deal with complex multiphysics models and are easily parallelizable. They are based on modified lattice-gas automata [1].

The algorithm of LB simulations is quite simple. Space is discretized into a lattice. Each node of this lattice has a state. This state indicates the proportion of particules moving along fixed directions (these proportions are generally called "fields"). At each time step, the state of all lattice nodes is updated. Each node first receives fields coming from its neighbors and then "collides" them by applying a collision operator which generates the new state.

Grid computing can be defined as "coordinated resource sharing and problem solving in dynamic, multi-institutional collaborations" [2]. In practice, a Grid user (which can be a software component) submits a job composed of tasks to the Grid. The tasks are automatically run on available computational resources across organizational boundaries (i.e. clusters of multiple departments).

LaBoGrid is an application combining the concepts of LB modelling and Grid computing. It is able to run LB simulations on an arbitrary number of computational resources from a Grid. It deals with operating system and hardware performance heterogeneity. The former because LaBoGrid is written in Java. The latter thanks to load-balancing [3]. This is essential because all LaBoGrid tasks are interconnected and depend on information from one another. A slow LaBoGrid task will slow down the overall process.

LaBoGrid is based on asynchronous agents exchanging messages. The two main agents are the Controller agent (CA) and the distributed agent (DA). In a deployed LaBoGrid system, the CA exists in only one instance. It keeps track of the DAs and their topology.

A task agent running some arbitrary code can be attached to the CA (CAT) and the DAs (DAT). In LaBoGrid, LB-specialized task agents are used (this system could be adapted to other problems). A configuration file parsed by the LB CAT gives the parameters of an LB simulation. The LB CAT configures automatically the LB DATs which handle the simulation code. Currently, LB simulations are done on 3D fluids with 19 fields per state. However, the code can be adapted very easily to other fluid dimensions and different state definitions, storage and computation precisions and collision types.

## References

[1] J. Jimenez F. Higuera and S. Succi. Lattice gas dynamics with enhanced collision. *Europhys. Lett*, 9:345, 1989.

[2] J. Nabrzyski, J. Schopf, and J. Weglarz. *Grid Resource Management: State of the Art and Future Trends*. Kluwer Academic Publishers, 2003.

[3] P. Marchot G. Dethier, C. Briquet and P.A. de Marneffe. A grid-enabled lattice-boltzmann-based modelling system. In *Proc. International Conference on Parallel Processing and Applied Mathematics*, Gdansk, Poland, 2007.