IL NUOVO CIMENTO Vol. 32 C, N. 2 Marzo-Aprile 2009
DOI 10.1393/ncc/i2009-10384-9

Colloquia: CSFI 2008

The Distrimobs approach for parallelization of pedestrian mobility computations

M. Brambilla and L. Cattelani

Physics of the City Laboratory(*), *Alma Mater Studiorum, Università di Bologna*
Via Zamboni 72, 40126, Bologna, Italy

(ricevuto il 22 Giugno 2009; pubblicato online il 3 Settembre 2009)

Summary. — Simulating pedestrian mobility is a typical centralized problem where each agent must interact with a plurality of other agents in order to make decisions about its local path planning. Distrimobs is a parallel- and distributed-agent–based pedestrian mobility simulator able to represent thousands of agents while keeping a good scalability. The aim of the Distrimobs simulator is to simulate the whole Carnival of Venice. In this work we present the Distrimobs approach for parallelization of the computations and some experimental results of performance intensive scenarios. These results highlight the scalability and the computational complexity of the simulator.

PACS 07.05.Tp – Computer modeling and simulation.
PACS 89.20.Ff – Computer science and technology.

1. – Pedestrian-mobility simulation

Simulation of pedestrian mobility is an innovative field of research involving several computer science subjects. The most advanced simulators model each pedestrian with an abstraction, typically with an agent, allowing representing single persons and not just crowds. The main problem caused by the punctual representation of pedestrians is the computational cost: each agent must interact with a plurality of other agents in order to make decisions about its local path planning (see fig. 1A). Furthermore in order to provide an output describing the positions of all the agents, punctual simulators have a computational cost lower bounded by the number of agents.

Distrimobs [1, 2] allows simulating and studying pedestrian mobility inside environments. Furthermore such a kind of tool makes it possible to experiment virtual improvements to already existing environments and/or to new and "in-design" ones [3]. These experiments allow urbanists, architects and so on to predict the consequences of

(*) www.physicsofthecitylab.unibo.it

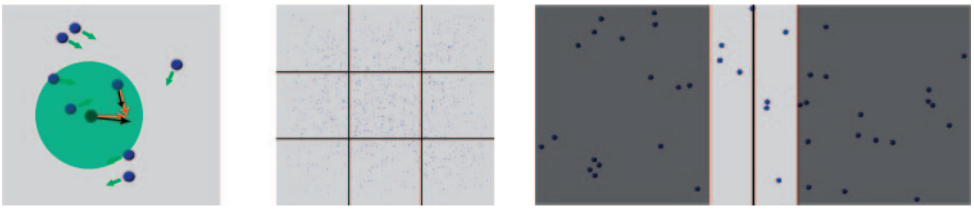

Fig. 1. – (A) Interactions of a mobber with an incoming one and corrections of the trajectories. (B) Subdivision of the map in zones. (C) Particular on the border zones.

their choices. It is easy to imagine that there are many other interesting applications of the Distrimobs simulator (security, commercial info and so on) and it is probable that pedestrian mobility simulators will become very important tools in the next years.

Recent researches have been made on modeling pedestrian mobility using least effort algorithm [4], social force model [5], gas kinetic [6], chronotopical [7] and cellular automata [8, 9]. The computational cost of the least effort algorithm is unknown being the algorithm itself unknown. Physics-based models (social force, gas kinetic and chronotopical) have a quadratic computational cost unless cuts of the force influences are made. Finally cellular automata based models usually have a lower computational cost but do not provide a punctual representation.

Distrimobs has been designed in order to simulate the whole Carnival of Venice that involves, lasting three days, about one million of persons. However, no particular optimizations using the Venice environment as input has been made on the simulator. This means that Distrimobs is perfectly modular and cities, environments, fluxes of pedestrians and their behaviors are just input parameters. In order to support thousands and thousands of agents simultaneously, many efforts have been dedicated to the Distrimobs parallelization engine. Here we present some theoretical aspects about simulating pedestrian mobility and, overall, show some results about the performances and the scalability of the parallelism of the Distrimobs simulator.

2. – Parallelization

In simulation, there are many techniques to subdivide computation among different processors. Three of the main methods are based on subdividing by: algorithms (different processing units take charge of different algorithms), agents (different processing units take charge of running different agents) and spatial areas (every processing unit takes care of a part of the map).

The distribution of the computations of pedestrian mobility simulation in Distrimobs is resolved adopting an approach based on two main concepts:

– the spatial subdivision;

– the information propagation.

The simulator subdivides the map where agents (mobbers) lie in a set of zones (see fig. 1B). This set is then partitioned between the processing units; each processing unit has to simulate the pedestrian mobility inside its zones. The main problem is represented by the interactions of mobbers near the borders of the zones. Mobbers near the borders interact with other zones in many cases. For example when a mobber may want to move from a zone to another or when a mobber sees neigh mobbers and so on.

All these kinds of problems can be solved starting from the information propagation principle. The information propagation principle exists in the Distrimobs model because the interactions between mobbers are bounded by two physical values: maximal sight and speed of mobbers. The information propagation principle of the Distrimobs model allows the sharing of a small amount of information to be sufficient in order to resolve the above-cited kinds of problems. This means that problems caused by interactions between mobbers belonging to different zones are solved if every zone shares the information about mobbers near its borders with the zones neighboring with it. In order to propagate information about mobbers near borders a further subdivision of the map is mandatory. The simulator creates for each border of each zone a border zone (see fig. 1C). The part of the zone uncovered by border zones is called internal zone. The extension of the border zones depends on the information propagation principle (maximal sight and speed of mobbers). All the information about mobbers lying in a border zone is shared with near zones. This means that information is shared between processing units too.

Spatial subdivision in zones and border zones allows solving the main problem of conceiving a scalable subdividing algorithm for pedestrian mobility computations. However, many other important problems arise from the parallelization of the simulation; some of these are

- load balancing: what happens if a multitude of mobbers lies in the same zone?
- mobbers spawning: what happens when mobbers are spawned in a border zone?
- determinism: parallelization and distribution usually introduce elements of indeterminism. How to handle RNG, parallelism and keep the simulator deterministic?
- serialization: mobbers may move between processing units (moving between zones). How to allow them to move although being featured with memory and AI?

The explanation to how problem like these are solved in Distrimobs is out of the scope of this paper but we deem important to consider that parallelization of an agent-based simulator involves a large amount of problems.

3. – Experiments and analysis of the results

Distrimobs has been benchmarked with many simulations in a special environment represented by an unobstructed square. Mobbers are spawned in south and east borders. Their goals were respectively the north and the west borders. The two big fluxes cross themselves through the entire environment. Managed experiments are:

- Analysis of the computational cost augmenting the simultaneously existing mobbers in the simulation keeping constant density. The results confirmed the expected theoretical computational cost: a linear trend (see fig. 2 in the upper-left quadrant).
- Analysis of the computational cost augmenting the simultaneously existing mobbers in the simulation (with variable density) with different processing units. The results, represented in fig. 2 (in the upper-right and lower-left quadrants), show a meaningful speed gain using multiple processing units. The more mobbers exist in the simulations the more multiple processing units utilization gains speed.

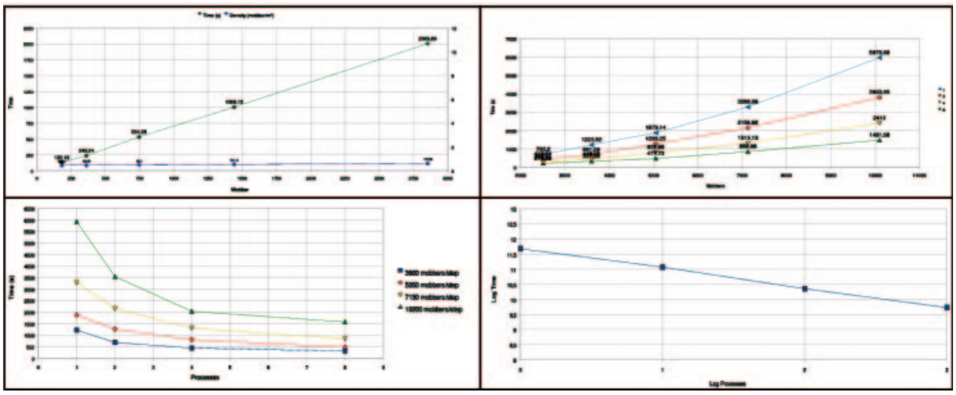

Fig. 2. – Graphs showing results of the benchmarks.

– Analysis of the scalability of the multiple processing units utilization. Results of simulations containing an average presence of twelve thousands of mobbers are analyzed. The analysis of the scalability is realized through a linear regression of logarithms of number of processing units and of execution time (see the lower-right quadrant of fig. 2).

$$\text{Regression time} \approx \frac{2^{11.7}}{|\text{CPU}|^{0.6567}} \text{ with an accuracy } R^2 = 0.9991.$$

Results are good: the regression time curve, having an exponent of 0.66, is not too far from the unreachable theoretical limit of 1.0. These results allow us to believe in being able to simulate the Carnival of Venice using a distributed and parallel architecture.

REFERENCES

[1] Brambilla M., *Distrimobs, simulatore di mobilità pedonale: studio e progettazione dell'architettura distribuita ed adattiva su cluster di blades*, Master Thesis, Alma Mater Studiorum - University of Bologna, Bologna (2005).

[2] Cattelani L., *Distrimobs, simulatore di mobilità pedonale: studio e progettazione dell'intelligenza artificiale degli agenti*, Master Thesis, Alma Mater Studiorum - University of Bologna, Bologna (2005).

[3] Brambilla M. and Cattelani L., *Build. Environ.*, **44** (2009) 595.

[4] Keith Still G., *Crowd dynamics*, PhD thesis, University of Warwick, Warwick (2000).

[5] Helbing D. and Molnar P., *Phys. Rev. E*, **51** (1995) 4282.

[6] Hoogendoorn S. and Bovy P. H. L., *Transp. Res. Board*, **1710** (2000) 28.

[7] Bazzani A., Giorgini B., Servizi G. and Turchetti G., *Phys. A*, **325** (2003) 517.

[8] Batty M., *Cities and Complexity Understanding Cities with Cellular Automata, Agent-Based models, and Fractals* (MIT Press, MA, Cambridge) 2005.

[9] Bandini S., Manzoni S. and Vizzari G., *IEICE Trans. Inf. Systems*, Vol. **E87-D**, p. 669.