# Large scale multi-agent simulations of transportation systems

N. Cetin[1], C. Gloor[1], K. Nagel[1], B. Raney[1], and A. Voellmy[1]

[1] Dept. of Computer Science, ETH Zürich, CH-8092 Zürich, Switzerland
{cetin@inf, chgloor@inf, nagel@inf, raney@inf, res@vis}.ethz.ch

On a certain level of abstraction, human transportation systems do not look very different from ant hills: Agent with different and mostly uncoordinated goals move through the system, they interact where this is necessary for their goals and where resources (e.g. walking or driving space) becomes scarce, and somehow they contribute to make the system "function".

For transportation systems, it is now possible to do direct agent-based simulations of such systems with many millions of travelers. Such a simulation does not only move the individual travelers through the system according to their mode of transportation (walk, car, etc.), but also integrates modules which simulate the strategic decisions of the agents, such as the generation of their daily schedule, or the selection of a route.

Typical such modules of a transportation simulation package are:

- **Traffic simulation module –** This is where travelers move through the street network by walking, car, bus, train, etc. (Fig. 1)

- **Modal choice and route generation module –** The travelers in the traffic simulation usually know where they are headed; it is the task of this module to decide which mode they take (walk, bus, car, bicycle, ...) and which route.

- **Activity generation module –** The standard cause why travelers are headed toward a certain destination is that they want to perform a specific activity at that location, for example work, eat, shop, pick someone up, etc. The activity generation module generates synthetic daily plans for the travelers.

- **Life style, housing, land use, freight, etc. –** The above list is not complete; it reflects only the most prominent modules. For example, the whole important issue of freight traffic is completely left out. Also, at the land use/housing level, there will probably be many modules specializing into different aspects.

In addition, there need to be initialization modules, such as the **synthetic population generation module**, which takes census data and generates disaggregated populations of individual people and households. Similarly, it will probably be necessary to generate good default layouts for intersections etc. without always knowing the exact details.

Since we run large scenarios (a typical metropolitan region has several million inhabitants), it is necessary to use parallel computers. We use a Linux-based Beowulf cluster of 32 Pentium PCs. The most compute-intensive part, and also the most difficult to parallelize, is the traffic simulation. When using domain decomposition, i.e. giving a different piece of the geographical

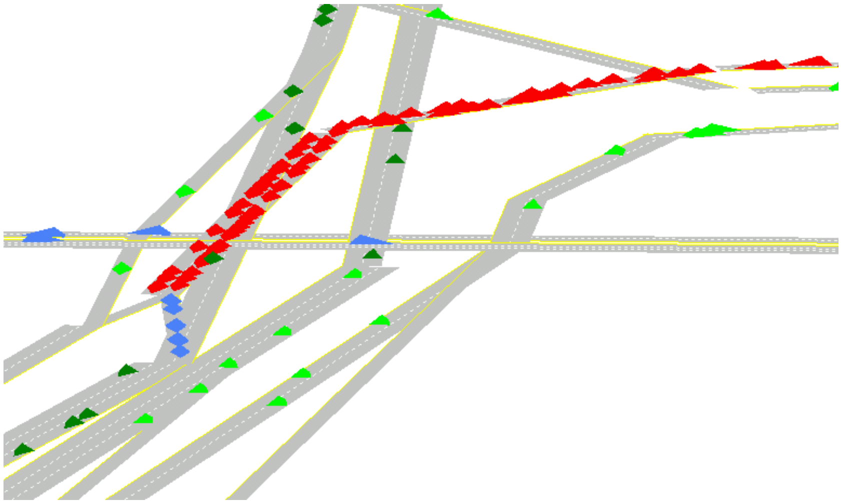

Figure 1: Traffic jam in Portland

area to each CPU, we are able to reach computing speeds of 100 times faster than real time. Because of some peculiarities of parallel speedup, this number is predicted to be valid for nearly arbitrarily large scenario sizes.

The above modules interact, and the interaction goes in both directions: activities and routes generate congestion, yet (the expectation of) congestion influences activities and routes. This is typically solved via a relaxation method, i.e. modules are run sequentially assuming that the others remain fixed, until the results are consistent.

A typical way to implement this is a “best reply” strategy: Between runs of the traffic simulation, a randomly selected 10% of the agents will obtain new dayplans that would have been optimal for the previous run. These agents will go into the next traffic simulation run using the new plans, while all other agents keep the old plans.

Fig. 2 demonstrates a typical result for a test case. The test scenario is that 50 000 randomly distributed agents all want to travel to Lugano (indicated by the circle). Initially (left figure), they all attempt to use the freeways, since they all have planned their trips not thinking about the others). After 49 iterations as described above, traffic is much more spread out (right figure).

The advantage of the agent-based approach is that it can be made more realistic. We are currently working in two directions:

- **Agent database** – Instead of having to accept the new plan from the system, the agents keep old strategies and their performance scores in a database. This has the advantage that the plans generation mechanism can be considerably more creative, since a badly performing plan will be ignored. Also, it is now possible to have arbitrary individual performance measures. In particular, it is possible to have performance measures for which finding an optimal solution is difficult or impossible, but behavioral heuristics (such as genetic algorithms) can be used.

- **Within-day replanning** – Real world travelers do not only replan over night, but they change their plans during the day or even during a trip, for example when they are caught

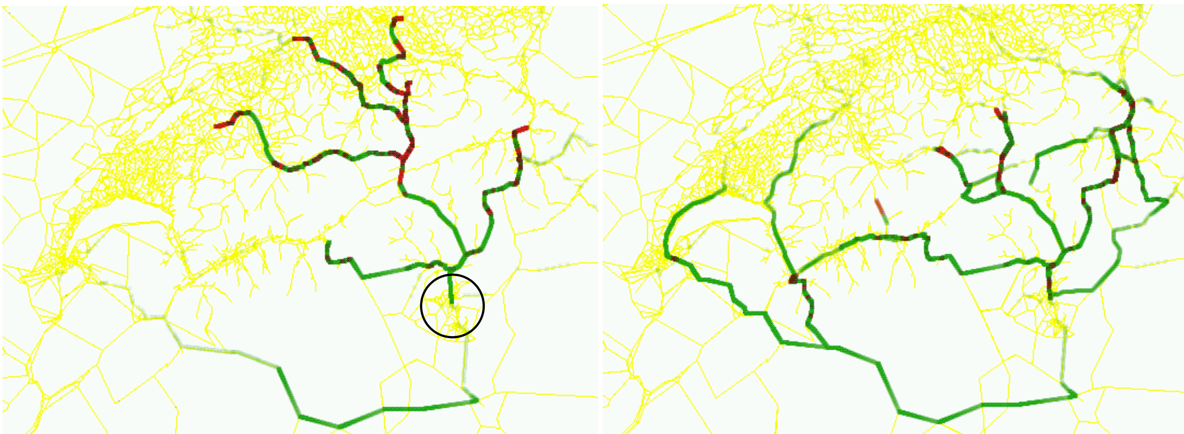

Figure 2: Result of day-to-day learning in a test example. The scenario is a test case, using a network of Switzerland, where 50 000 vehicles from all over the system start between 6am and 7am and simultaneously attempt to reach a destination within the circle. Green shows free flowing traffic; red shows traffic jams. LEFT: Situation at 9:00am in the zeroth iteration. RIGHT: Situation at 9:00am in the 49th iteration. Clearly, traffic has spread out and the system is using more different routes in the 49th iteration.

in unexpected congestion.

Within-day replanning becomes a challenging problem if one wants to maintain parallel computing efficiency, and if one attempts to structure the design so that different programming teams can independently work on the different modules. We are currently implementing this by having only "low level intelligence" of the agents (such as collision avoidance, lane changing, reaction to traffic lights) directly in the traffic simulation and do all the strategic computation outside. The simulation will be coupled to the strategic modules via messages.

We are currently working on a simulation of all of Switzerland, that is on a full, realistic traffic simulation of 24 hours of a typical workday. In a first step, demand will come from traditional origin-destination-matrices, which are provided by the Swiss transportation authorities. Besides the traffic simulation, our system will do the route learning. Once this is working, we will include the learning of activities and daily plans into the framework. Fig. 2 is an intermediate test result from this project.

For more information, please see http://www.inf.ethz.ch/˜nagel/ .