# Evolving Individual Behavior in a Multi-agent Traffic Simulator

Ernesto Sanchez, Giovanni Squillero, and Alberto Tonda

Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129, Torino, Italy
{ernesto.sanchez,giovanni.squillero,alberto.tonda}@polito.it

**Abstract.** In this paper, we illustrate the use of evolutionary agents in a multi-agent system designed to describe the behavior of car drivers. Each agent has the selfish objective to reach its destination in the shortest time possible, and a preference in terms of paths to take, based on the presence of other agents and on the width of the roads. Those parameters are changed with an evolutionary strategy, to mimic the adaptation of a human driver to different traffic conditions. The system proposed is then tested by giving the agents the ability to perceive the presence of other agents in a given radius. Experimental results show that knowing the position of all the car drivers in the map leads the agents to obtain a better performance, thanks to the evolution of their behavior. Even the system as a whole gains some benefits from the evolution of the agents' individual choices.

**Keywords:** Multi-agent systems, Evolution, Traffic simulation.

## 1 Introduction

Road traffic congestion is a crucial problem, the short-range consequences of which can vary from delays to decreased throughput of vehicles. Long-range consequences include reduced safety, environmental pollution, and reduced economic competitiveness. This problem is becoming more intense, not only in western cities but also in countries where the presence of cars, once scarce, is growing at an alarming rate.

From websites displaying the current traffic conditions [1] to collections of traffic control strategies [2] available online, information technology is playing a vital role in the development of new approaches to traffic control, even by simply providing the means to evaluate innovative methodologies, by means of sensors, databases and data mining.

In this context, simulations are heavily employed to test the possible outcome of new strategies, and often multi-agent systems are chosen as a simulation tool. An agent is defined as a complex software entity that is capable of acting with a certain degree of autonomy in order to accomplish tasks. A multi-agent system (MAS) is a collection of software agents that work in conjunction with each other. They may cooperate or they may compete, or some combination of the two, but there is some common infrastructure that result in the collection being a 'system', as opposed to simply being a disjoint set of autonomous agents [3]. Each agent in the MAS tries to achieve some individual or collective task.

C. Di Chio et al. (Eds.): EvoApplications 2010, Part I, LNCS 6024, pp. 11–20, 2010.

Many works on MAS have been led around road traffic issues. For instance, agents can be used for traffic management. In [4], the authors propose an application based on coordination of agents to diagnostic and inform drivers about traffic problems in a located area. Traffic-related issues are also the basis for [5], [12] and [13]. In [8] a multi-agent simulation is used to validate three different self-organizing methods aimed at optimizing the configuration of traffic lights in a city.

In [6] Doniec et al. experimented traffic simulation by modeling an intersection between two real roads, and studying the behavior of agents representing car drivers who could choose whether to abide by the traffic rules or ignore them, with the selfish objective to minimize the time they had to wait in order to cross the intersection. They demonstrated that an accurately modeled MAS simulated the traffic trend effectively going on during the day in the real world intersection.

To enhance a generic MAS traffic simulator, we propose to populate it with agents able to evolve their behavior through an evolutionary strategy, mimicking human ability to adapt to changing traffic conditions. Behaviors are optimized to provide a more realistic simulation, that is, the result of the whole evolution process is a *single* realistic scenario. The simulator is not intended to analyze evolution. We demonstrate that the agents significantly improve their performance when given information on the position of other cars, thanks to the evolution in their preference on roads to go through. Since the change of behavior of an agent may trigger the change of behavior of other agents, this interaction creates a co-evolutionary system. However the work focuses on the global properties of the system, rather on an analysis of the co-evolution itself. In section 2, we illustrate the specifications for the model we built. Agents' behavior and evolution are discussed in section 3. Section 4 shows the results we obtained through experimental evaluation of a simple MAS. Section 5 contains the conclusions we were able to draw from the experience.

## 2 Model Specifications

In this work we use a simplified MAS to simulate the traffic in the central part of a typical medium-sized European town (e. g. with a population of 20,000 – 30,000 people), which usually has different kinds of roads, from the small alleys to the main street, from one-way roads to large two-way ones, with several roundabouts but no traffic lights. Each agent in the MAS models a car driver, and it has the objective of traveling from a starting point to an arrival point in the map.

Each agent in the MAS previously described possesses various properties:

a) a starting location;
b) an ending location;
c) agent's preferences when it has to choose a path.

Starting location and ending location are determined at the beginning of each run, for each agent: the first one is randomized between all the points in the map, while the second is chosen so that the path the car driver has to travel through is at least half the size of the map (both on the horizontal axis and on the vertical axis). This means that each agent has to pass through the center of the map in order to reach its destination, thus increasing the probability of traffic jams. Each agent is started at a different time during the first seconds of the simulation.

The inclination of each agent to choose a certain road when it comes to a crossroad in the map is expressed by a series of weights associated to the agent itself. Those weights let an agent choose, for example, a longer but less crowded road over a shortest but very busy one, applying the formula that represents the hypothetical time needed to reach the next intersection on the road:

$$road\_weight = length\left(1 + \frac{w_1 agents\_num}{1 + w_2 width\_bool}\right)$$

where:

- *length* is the length of the road;
- *agents_num* is the number of agents on the road at the time when it is evaluated;
- *width_bool* is a Boolean value which express whether we are considering a two-lane or a one-lane road;
- *w1* and *w2* are the weights that lead the preference of the agent, making it choose wider roads over less crowded ones, or viceversa.

Every time an agents reaches a crossroad, it computes the value *road_weight* for each road departing from the crossroad, then for each road that road intersects, using *Dijkstra's algorithm* to find the path with minimum sum of *road_weights* based on the current situation. It is important to notice that this procedure is repeated each time an agent reaches a crossroad, because the parameter *agents_num* changes at each unit of time of the simulation as the agents are moving through the map. Thus, a car driver can change the path that it computed at the previous crossroad, as it perceives that the road he was going to take is now more busy, depending on the values associated to its weights.

Each agent has a perception of the situation of the roads around him up to a certain range, which can be set by the user of the simulation tool, expressed in the same units of length which are used for the map. For example, an agent could know how many agents are moving in all the roads in a radius of 100 m from its current position: in this situation, it would compute the *road_weight* for those roads as normal, but it could not obtain the parameter *agents_num* for streets outside that range, and thus would evaluate the remaining part of the map only on the basis of the *length* and *width* of the roads. It is important to notice that two agents may have the same starting and ending point in the map, and still follow two completely different paths, since their weights could make them choose two opposite ways from the beginning of the simulation.

The speed of an agent in the MAS is adjusted during each step of the simulation, on the basis of the number of cars that are currently traveling immediately ahead of him on the road the agent is going to take. If there are no cars, the agent will gradually increase its speed up to a maximum value, the city speed limit. If other cars are on its same path, vice versa, its speed will decrease by a value proportional to the number of cars and the width of the road, up to a complete stop.

## 3 Agents

The objective of each agent/car driver in the MAS is to travel from a starting point to an arrival point in the map. A driver is rewarded when he manages to minimize its own traveling time. It is important to notice that there is no common goal for all the agents: each one tries to reach its objective working independently from all the others. Despite this, we can have an index of how well the whole system is performing by measuring the average speed and the average time needed to reach the arrival point, considering all the agents roaming through the map.

### 3.1 Agents Structure

The genotype of each agent is a series of weights (*w1, w2*) that describe its behavior: those values represent the preferences of the agent when he is about to choose its way out of a crossroad, based on the number of other agents/cars on each street and on the number of lanes of the roads.

### 3.2 Evolution

In order to evaluate whether an optimal choice of path for a single agent could lead to an overall improvement in the mobility on all the roads, we chose to evolve the behavior of each agent, running the simulation through multiple generations.

There is, in fact, a series of problems that arise when we try to evaluate the global performance of this system, since each agent has a selfish objective and will not cooperate with other agents to reach it. Also, it is unlikely that the system itself will reach a stable state at a certain generation, because the change of behavior of an agent could potentially influence all the others: given that a car driver takes into account the presence of other cars on the roads it could choose, even a single agent modifying its path could lead to a chain-reaction of behavior modification for a great number of car drivers in the MAS. In order to avoid dramatic fluctuations of the paths and to reflect the behavior of human car drivers (that seldom change their track of choice), only a given percentage of the agents spawns a child each generation.

### 3.3 Agents Evolution

The evolution is similar to an *evolutionary strategy* (1+1): at each generation step, a single individual, which is represented by the weights associated to an agent, has a certain probability to produce a child. The child is then evaluated by making the agent behave on the basis of the new weights generated. It is important to notice that there is no such thing as a “population of all the agents”: each agent stores a population made of a single individual that spawns a single child.

The genotype of each individual is a vector of real values (*w1, w2*): the new individual is created by mutation. A random value, obtained through a Gaussian distribution with mean 0, is added to each component of the vector. The selection is deterministic: the child is compared to the parent, and if its fitness value is better, it becomes the new parent; otherwise, it is discarded. The fitness, in our case, is based on the time it takes the agent to reach its destination: the lesser, the better. The population at each step is thus composed by two individuals, parent and child. We made this choice because the fitness value can be properly compared only between individuals with the same starting and ending points on the map: since each agent represents a car driver, we are modeling the fact that a human in this situation would probably learn from the experience, changing the parameters on which he makes his choices on the basis of his past experiences.

Every agent's evolution is independent from the other agents, since car drivers, unluckily, do not cooperate with each other to minimize their travelling time. There is, however, a diffused co-evolution, since the behavior of each agent could influence the choices of all the others: one of the parameters taken into account when a car driver is selecting a path, is the number of cars on each road it could take. When even a single agent modifies its behavior, and consequentially its path, it obviously changes the number of cars that are going on a certain road: this could lead to a chain reaction where a great number of other agents would change path because of that increase/decrease in the quantity of cars on that road.

## 4 Experimental Evaluation

In order to experimentally evaluate our framework, we created a simplified model of a medium-sized downtown, to easily verify whether the results we would obtain were coherent to what we expected, while keeping the time needed for a simulation within reasonable parameters.

In our simplified model:

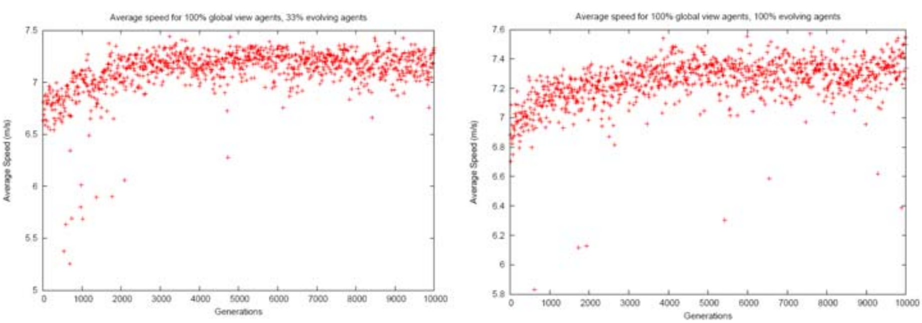
- all cars/agents are represented as points;
- each crossroad has up to four roads departing from it;
- crossroads are not regulated by traffic lights, a common occurrence in small to medium urban areas as the one in Figure 1, where roundabouts have replaced traffic lights;
- roads can only be vertical (crossing the map north to south) or horizontal (crossing the map east to west);
- the map of the city downtown is 0.16 $Km^2$, a square with an edge of 400 m, since we used meters as the unit of length to describe the roads and the cars, and comprehends 12 different roads, eight 400 m single lane roads, four 200 m single lane and two double lane 400 m ones.
- During each simulation, we chose to keep some parameters fixed, in order to easily compare the experimental results:
- the number of agents on the map at the same time is 400, which makes our city downtown quite crowded. Such a density is chosen to increase evolutionary pressure;
- at the beginning of the simulation, 10 agents are placed on their starting point every second. Each vehicle in a simulation run has the same initial and final location in every generation;
- once every 10 generations, we chose to run a simulation where no agent produced offspring, in order to check what is the best strategy, expressed as a set of weights, obtained for each agent up to that point;
- each car driver/agent can reach a maximum speed of 13 m/s, which is about 46.8 km/h or 29.1 MPH, slightly under the speed limits that are enforced in most

downtown streets. Also, when the traffic is heavy, it is very unlikely that a driver could run at a higher speed;

- when any agent reaches a crossroad and finds that its average speed computed on the last 10 seconds of simulation is under 7 m/s (about 25.2 km/h or 15.7 MPH), it will compute again the path from its current location to the destination, using an updated value for the number of agents on each road;
- each agent has a complete knowledge of the streets' map;
- depending on the simulation, a certain number of agents possesses a knowledge of the position of other agents the whole map, while the rest has a *local view*, which means that it can perceive the presence of cars only in a radius of 150 m;
- we consider that each car occupies 6 m in a certain road, because most cars are 5 m long, and when they are in a line it is wise to keep a distance of at least 1 m from the other cars. This parameter is used by the agents to detect the presence of other car drivers inside their sight radius.

### 4.1 Complete Knowledge

In the first set of runs, we tested a simulation where every agent had a complete knowledge of the position of all other agents. Only a third (33%) of the agents, selected randomly, would produce an offspring at each generation step. We let our system evolve for 10,000 generations, and then we evaluated the average speed (considering all cars in the MAS) each 10 generations, thus plotting only the simulations with all the best individuals.




**(A)** **(B)**

**Fig. 1.** On the X axis, number of generations. On the y axis, average speed (m/s). **(A)** Average speed for 100% global view agents, 33% evolving agents. **(B)** Average speed for 100% global view agents, 100% evolving agents.

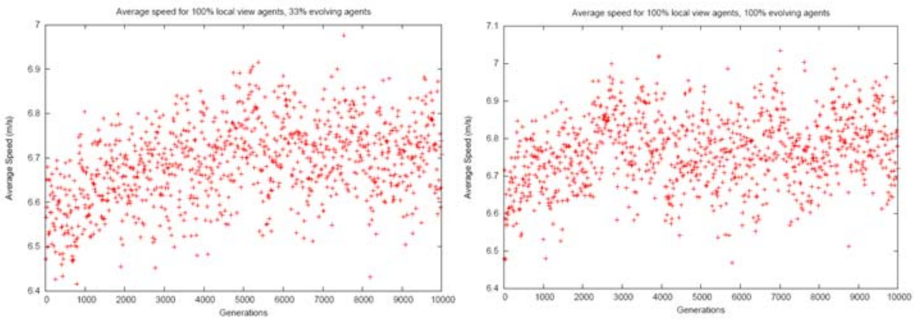
We can see that, while there is a fluctuation in the average speed (due to the agents changing their paths), it quickly increases from 6.5 m/s to over 7 m/s during the first 3,000$^{th}$ generations, then it keeps shifting between 7.25 and 7.3 m/s, with a peak just under 7.5 m/s. There are some generations, in the beginning of the simulation, where the average speed drops to values around 5.5 m/s, probably because the system is adjusting to the new paths chosen by a significant number of agents. After 5,000 generations, however, the fluctuations become smaller and smaller.

Increasing the number of agents that generate an offspring at each step could improve the performance of the system: thus, we ran another series of simulations where we raised the percentage of car drivers trying new preferences at each generation up to 100%.

As we can see the average speed rises faster during the first 3,000 steps, while the fluctuations in the later generations seem to be a little stronger than in the previous run, with drops in speed even after 5,000 generations.

## 4.2 Partial Knowledge

In a second time, we tested a set of simulations where no agent had a complete knowledge of the position of the other agents: thus, 100% of the car drivers on the map could perceive other cars only in a radius of 150 m from their current position. In this first simulation, only 33% of the agents change their preferences at each generation.




**(A)** **(B)**

**Fig. 2 .** On the X axis, number of generations. On the y axis, average speed (m/s).
**(A)** Average speed for 100% local view agents, 33% evolving agents.
**(B)** Average speed for 100% local view agents, 100% evolving agents.

As we expected, even from the first generation the average speed is lower than the values obtained from the previous experiments, where all the car drivers could perceive the presence of every car driver on every road. The average speed rises far slowly and, even if there are generations with peaks just below 7 m/s, the values are centered around 6.7 m/s. The improvement in the average speed is not so distinct as in the previous runs, and even in the last generations we have points that are at the same level of the initial value.

Increasing the number of car drivers trying new preferences at each generation proved useful when we had agents with global information on the position of other cars: running a set of simulations with 100% of partial-knowledge agents and 100% of evolving ones produced the following graphic.

While there are various peaks over 7 m/s, the shape of the graphic is very similar to the previous simulation where all the agents had only partial information.

**Table 1.** Average time for an agent to reach its destination, under different conditions

| Type of experiment | | Average on the first 5 generations | Average on the last 5 generations |
|---|---|---|---|
| 100% complete knowledge | 33% evolving | 61.50 s | 56.30 s |
| | 100% evolving | 60.09 s | 55.10 s |
| 100% partial knowledge | 33% evolving | 63.29 s | 61.34 s |
| | 100% evolving | 63.73 s | 61.33 s |

The information in Table 1 shows that the average time the agents need to reach their destination drops in a significant way during the simulations where all the agents had all the global information on the map, while the improvement in the experiments where the agents had only information on the position of the other cars in a limited radius is much smaller.

From the experimental results, we can see that information about the position of other cars in a heavy traffic situation like the one we simulated is surely useful, as we expected, both for the single agent and for the system. Even if each agent used the information it had in order to selfishly maximize his speed, without any cooperation with the other car drivers, this behavior proved functional to the whole system: the average speed of all the car drivers increased from the first generation up to a certain value, and kept fluctuating around that speed.

We can notice that the increment is present in every simulation, even those where all the agents can perceive the presence of other cars only in a radius of 150 m: we can conclude that even partial information, provided in real-time to car drivers able to evolve their behavior like humans do, could help lightning the traffic in the downtown of great cities.

### 4.3 Mixed Agents

By setting up our MAS with both local-view agents and global-view agents, we expected to find a slower increase in the average speed (thus, a slower decrease of the average time the agents need to arrive to their destination) than in the "complete knowledge" case, while on the contrary obtaining better results than in the "partial knowledge" simulation. We also assumed that, even if the presence of global-view agents would surely improve the average performance of the system, above a certain percentage of global-view agents there would be no further improvements. Thus, we ran several experiments with mixed agents, in various proportions, while keeping the other parameters fixed. The percentage of evolving agents is always 100%, and the local view radius is 150 m.

Our results confirm what we supposed: the average time the agents need to reach their arrival location drops steadily as the percentage of local-view agents drops and the percentage of global-view agents increases. This could mean that in a real situation, even if not all the drivers could have access to all the information, there could be

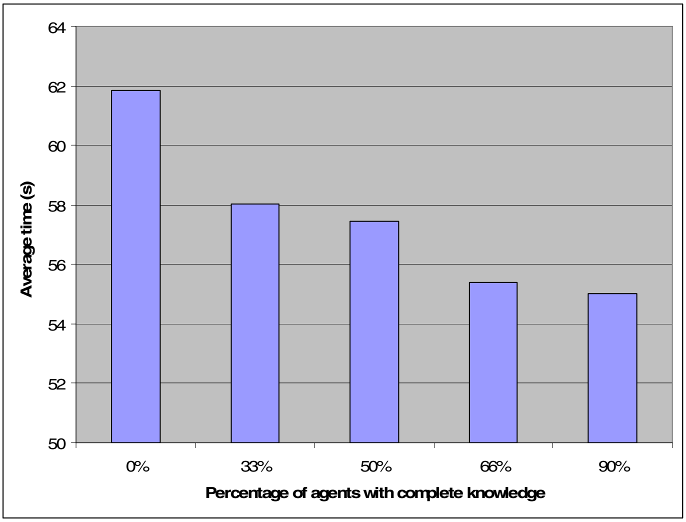



**Fig. 3.** Different arrival times with an increasing percentage of complete knowledge information agents in the population

an improvement in the traffic given as long as a certain percentage of car drivers have complete knowledge of the position of other cars.

## 5 Conclusions

We presented a MAS populated by agents able to evolve their behavior. Each agent represents a car driver starting from a random point in the map, with the objective to reach another random point, with the selfish objective of minimizing its traveling time. Every agent is able to evolve its choices through a series of generations, and it has a certain knowledge of the whole map. We tested the system with a number of agents chosen in order to model a situation where traffic is heavy, for example at the start or at the end of a business day, and providing each agent with information that could help a real-world driver, able to change its behavior. Our experimental results show that, even if there is no cooperation between agents, the average time used to arrive at their goal will gradually decrease generation after generation, finally starting to dynamically shift around a minimum value. This process is faster and the minimum reached is lower when every agent has complete information of its surroundings and of the position of every other agent. We can conclude that data obtained is consistent with the expected results, and thus the evolution of agents' behavior could provide more realistic data in traffic simulations. Future works will include large scale simulations with evolvable agents, implemented with open-source microscopic traffic simulation packages [10][11].

## Acknowledgements

Our thanks to Chiara Patané for developing the software used in the simulations.

## References

1. http://trafficinfo.lacity.org/index.html
2. Ding, W., Marchionini, G.: A Study on Video Browsing Strategies. Technical Report UMIACS-TR-97-40, University of Maryland, College Park, MD (1997)
3. http://www.agtivity.com/def/multi_agent_system.htm
4. Hernandez, J.Z., Ossowski, S., Garca-Serrano, A.: Multiagent architectures for intelligent traffic management systems. Transportation Research Part C: Emerging Technologies, 10, 473–506 (2002)
5. Bazzan, A.L.C.: A distributed approach for coordination of traffic signal agents. Journal of Autonomous Agents and Multi-Agent Systems (10), 131–164 (2005)
6. Doniec, A., Espie, S., Mandiau, R., Piechowiak, S.: Non-normative behaviour in multi-agent system: some experiments in traffic simulation. In: IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2006, December 18-22, pp. 30–36 (2006)
7. Balan, G., Sean, L.: History-based Traffic Control. In: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems, pp. 616–621 (2006)
8. Gershenson, C.: Self- Organizing Traffic Lights. Centrum Leo Apostel (2005)
9. Maroto, J., Delso, E., Félez, J., Cabanellas, J.M.: Real-time traffic simulation with a microscopic model. IEEE Trans. Intell. Transp. Syst. 7, 513 (2006)
10. Krajzewicz, D., Bonert, M., Wagner, P.: RoboCup 2006 Infrastructure Simulation Competition (2006)
11. Barceló, J., Codina, E., Casas, J., Ferrer, J.L., García, D.: Microscopic traffic simulation: A tool for the design, analysis and evaluation of intelligent transport systems. Journal of Intelligent and Robotic Systems 41(2-3) (January 2005)
12. Kesting, A., Treiber, M., Helbing, D.: Agents for Traffic Simulation, Contribution to Agents, Simulation and Applications. In: Uhrmacher, A., Weyns, D. (eds.) eprint arXiv:0805.0300 (May 2008)
13. Yang, J., Deng, W., Wang, J., Li, Q., Wang, Z.: Modeling pedestrians' road crossing behavior in traffic system micro-simulation in China. Transportation Research Part A: Policy and Practice 40(3), 280–290 (2006)