Numerical Simulation and Natural Computing applied to a Real World Traffic Optimization Case under Stress Conditions: "La Almozara" District in Saragossa

M.J. Galán Moreno, J.J. Sánchez Medina, L. Álvarez Álvarez
E. Rubio Royo[1]

Abstract: Urban traffic is a key factor for the development of a city. There exist many different approaches facing traffic optimization. In our case we have focused on traffic lights optimization. We have designed and tested a new architecture to optimize traffic light cycle times. The purpose of this research is to demonstrate the good performance of our architecture in a congested scenario. We have simulated several congestion situations for a very large real world traffic network – "La Almozara" in Zaragoza, Spain. Our results seem encouraging in this extreme situation. As we increase the load in the network we get a the better traffic behavior of our architecture. Finally, new research directions are presented.

Keywords: Intelligent Transportation Systems, Traffic Modelling, Genetic Algorithms, Traffic Congestion.

1   Introduction

At the present time we live a global energy and environmental crisis. The scientific community argues that the global warming process is, at least partially, a consequence of modern societies development. A key area in that situation is the citizens mobility. World economies seem to require fast and efficient transportation infrastructures for a significant fraction of the population.

The non-stopping overload process that traffic networks are suffering calls for new solutions. In the vast majority of cases it is not viable to extend that infrastructures due to costs, lack of available space, and environmental impacts. Thus, traffic departments all around the world are very interested in optimizing the existing infrastructures to obtain the very best service they can provide.

---

[1] M.J. Galán Moreno, J.J. Sánchez Medina, L. Álvarez and E. Rubio Royo are members of CICEI, ULPGC, SPAIN.

In the last decade many initiatives have been developed to give the traffic network new management facilities for its better exploitation. They are grouped in the so called Intelligent Transportation Systems.

Examples of these approaches are the Advanced Traveler Information Systems (ATIS) and Advanced Traffic Management Systems (ATMS). Most of them provide drivers or traffic engineers the current traffic real/simulated situation or traffic forecasts. They may even suggest actions to improve the traffic flow.

To do so, researchers have done a lot of work improving traffic simulations, specially through the development of accurate microscopic simulators. In the last decades the application of that family of simulators was restricted to small test cases due to its high computing requirements. Currently, the availability of cheap faster computers has changed this situation.

Some famous microsimulators are MITSIM(Yang (1997)), INTEGRATION (Rakha, Van Aerde, Bloomberg, and Huang (1998)), AIMSUN2 (Barcelo, J.L., Garcia, Florian, and Le Saux (1996)), TRANSIMS (Nagel and Barrett (1997)), etc. They will be briefly explained in the following section.

Although traffic research is mainly targeted at obtaining accurate simulations there are few groups focused at the optimization or improvement of traffic in an automatic manner – not dependent on traffic engineers experience and "art".

One of the most important problems in traffic optimization is traffic light cycles[1] optimization. This is a hard Combinatorial Problem which seems not to have a known deterministic solution at the present time.

In our group we have been working on the optimization of traffic lights cycles for the better performance of urban traffic networks. As shown in Brockfeld, Barlovic, Schadschneider, and Schreckenberg (2001), traffic light cycles have a strong influence in traffic flow results. For that reason we decided to focused on that problem. We have combined a Genetic Algorithm (GA) as optimization technique with a traffic microscopic simulator running on a scalable MIMD multicomputer[2].

We have tested the forementioned three pillar model with some works (Sánchez, Galán, and Rubio (2004), Sánchez, Galán, and Rubio (2005b), Sánchez, Galán, and Rubio (2005a), Sánchez, Galán, and Rubio (2006), Sánchez, Galán, and Rubio (2007) and Sanchez-Medina, Galan-Moreno, and Rubio-Royo (2008)). By means of this work our aim is to study how it works in a congested big network. We have

---

[1] Traffic light cycle: the finite sequence of states– e.g. green, orange, etc. – that a traffic light runs iteratively.

[2] MIMD: Multiple Instruction Multiple Data: A type of parallel computing architecture where many functional units perform different operations on different data. For example a network of PC's working in parallel.

used the traffic data supplied by the local government of Zaragoza – Spain. We have simulated the traffic at this network with the supplied statistics of traffic inflow. In a second phase we have increased the traffic input far beyond the current statistics and then we have optimized the traffic lights times for the new situation. The results show our model to be suitable for optimizing such times also in a congestion scenario.

The rest of this article is organized as follows. In the following subsection we give a wide survey of the current State of the Art. In 1.1.3 we briefly expose our own contribution to the matter. In section 2 we explain with some detail the proposed methodology. In section 3 the experiments performed, the restrictions assumed and the results obtained are shown. In section 4 we discuss the results and their implications. Some final concussions and future work ideas are given in the section 5.

1.1 State of the Art

In this subsection we want to give a survey of some significant works in the area. We have used Genetic Algorithms as non-deterministic optimization technique. Focusing now on our research area, we have categorized works in three classes: those mostly related to Advanced Traveler Information Services (ATIS); those mainly about Advanced Traffic Management Systems (ATMS), and in a third subset we have called Advanced Traffic Optimization Systems (ATOS), those where traffic is not just managed but optimized – or tried to be optimized – in an automatic manner, without human interaction.

1.1.1 Advanced Traveler Information Services

Advanced Traveler Information Services are those services that can potentially help drivers to make better decisions in order to reduce their travel time. There are many initiatives in this area. Here we show some examples.

In Florian (2004), this thesis provides an empirical study of the impact of ATIS on transportation network quality of service using an application of DynaMIT (Dynamic network assignment for the Management of Information to Travelers). The main results are that the provision of dynamic route guidance can simultaneously benefit the individual performance of drivers, both guided and unguided, as well as the system performance of existing transportation infrastructure.

In Hafstein, Chrobok, Pottmeier, Schreckenberg, and Mazur (2004) a high resolution cellular automata freeway traffic simulation model applied to a Traffic Information System. They provide a simulation for current traffic zones without loop detectors, and 30 min. and 60 min. future traffic forecasts. They run a java applet

in a web page in order to give the network users this useful information.

1.1.2 Advanced Traffic Management Systems

Advanced Traffic Management Systems are those systems that help engineers to better manage traffic networks. There are many works around this topic, most of them focused on traffic simulation. Some examples are the following.

The INTEGRATION model has been used to simulate traffic for the Salt Lake Metropolitan Area (Rakha, Van Aerde, Bloomberg, and Huang (1998)).The objective of this paper is threefold. First, the feasibility of modeling a large-scale network at a microscopic level of detail is presented. Second, the unique data collection challenges that are involved in constructing and calibrating a large-scale network microscopically are described. Third, the unique opportunities and applications from the use of a microscopic as opposed to a macroscopic simulation tool are described.

The MITSIM model (Yang (1997)) has been used to evaluate aspects of both the traffic control system and the ramp configurations of the Central Artery/Tunnel project in Boston. It explicitly incorporates traffic prediction, time variant traffic information, and dynamic route choice.

AIMSUN2 has been used to simulate the Rings Roads of Barcelona Barcelo, J.L., Garcia, Florian, and Le Saux (1996). Uses parallel computers to shorten the execution time.

Traffic simulation using CA models has also been performed on vector supercomputers to simulate traffic in shortest possible time (Nagel and Schleicher (1994a)).

The INTELSIM model is used in Aycin and Benekohal (1998) and Aycin and Benekohal (1999). In those works a linear acceleration car-following model has been developed for realistic simulation of traffic flow in intelligent transportation systems (ITS) applications. The authors argue that the new model provides continuous acceleration profiles instead of the stepwise profiles that are currently used. The brake reaction times and chain reaction times of drivers are simulated. As a consequence, they say that the good performance of the system in car-following and in stop-and-go conditions make this model suitable to be used in ITS.

Moreover, in Aycin and Benekohal (1999) they compare many car-following methods with their proposed method, and with field data.

In Bham and Benekohal (2004) they proposed a "high fidelity" model for simulation of high volume of traffic at the regional level. Their model uses concepts of Cellular Automata and Car-Following models. They propose the concept of Space Occupancy (SOC) used to measure the traffic congestion. Their aim is to simulate high volume of traffic with shorter execution time using efficient algorithms

on a personal computer. Like in our case, they based their simulator on Cellular Automata concepts. Although their model could be more accurate than the one of ourselves, in our work we go further using our simulator inside a GA for optimizing the traffic – not just for simulating traffic.

In Tveit (2003), Dr. Tveit, a senior researcher with *SINTEF*[3], explains that a common cycle time[4] for a set of intersections is a worse approach than a distributed and individualized one. His conclusions appear sound and convincing, so we consider them in our approach. In our system every intersection has independent cycles.

In Smith (1988) the use of responsive signals[5], with network capacity (rather than total travel cost) as a control criterion is argued. The capacity of the network is maximized if the signals operate to equalize traffic density on the most occupied parts of the network. This is another example of multiple local optimizations instead of a global optimization, like the one of ours.

In Logi and Ritchie (2001) a knowledge based system is presented for traffic congestion management. The proposed model comprises a data fusion algorithm, an algorithm for selection the suitable control plan, and it presents the proposed plan with an explanation of the reasoning process for helping the traffic operators decisions. They presented also a validation example for displaying the ability of their system to reduce congestion. From our point of view, although this seems a very interesting approach to the matter, both the selection of control strategies and the estimation of future traffic are based on the experience of traffic engineers. In spite of this, in our methodology we use the combination of two widely accepted and trusted techniques. We use a more accurate estimation of future traffic – thought a microsimulator – and a genetic algorithm for the optimization of the traffic flow.

1.1.3 Advanced Traffic Optimization Systems

TRANSIMS project used CA models to simulate traffic for the city of Fortworth-Dallas using parallel computers (Nagel and Barrett (1997)). This paper presents a day-to-day re-routing relaxation approach for traffic simulations. Starting from an initial plan-set for the routes, the route-based microsimulation is executed. The result of the microsimulation is fed into a re-router, which re-routes a certain percentage of all trips.

---

[3] SINTEF means The Foundation for Scientific and Industrial Research at the Norwegian Institute of Technology.

[4] Common cycle time: This is a very simple way of programming traffic lights in an intersection or groups of intersections. All the traffic lights share a cycle length. The starting point of each one of the states or *stages* in the particular cycle of every traffic light may be different, but the cycle period is the same for all of them.

[5] Responsive signals: Traffic signals capable of adapting their state to the current traffic situation near them.

In Wey (2001), an isolated intersection is controlled applying techniques based on linear systems control theory to solve the linear traffic model problem. The main contribution of this research is the development of a methodology for alleviating the recurrent isolated intersection congestion caused by high transportation demand using existing technology. Again this work deals with very small scale traffic networks — one intersection.

In Schutter and Moor (1997) the authors present a single intersection – two two ways streets – model describing the evolution of the queue lengths in each lane as a function of time, and how (sub)optimal traffic switching schemes for this system can be determined.

In Febbraro, Giglio, and Sacco (2002) Petri Nets are applied to provide a modular representation of urban traffic networks. An interesting feature of this model is the possibility of representing the offsets among different traffic light cycles as embedded in the structure of the model itself. Even though it is a very interesting work, the authors only optimize the coordination among different traffic light cycles. Our cycle optimization methodology is a complete flexible one because we implicitly optimize not only traffic light offsets but also every *stage length*.

Another interesting work using Petri Nets is Li, Tang, Mu, and Shi (2004) where they are applied to control a single intersection by means of programmable logic controllers (PLCs). They compare three methods for modeling the traffic lights at an intersection and found out that the more suitable is the one that combines Petri nets with PLCs. Again, in this research just one intersection is optimized, and not a whole traffic network.

In Spall and Chin (1994) the authors presented a neural network (NN) approach for optimizing traffic light cycles. A neural network is used to implement the traffic lights control function. The training process of the NN is fed exclusively with real data. This being so, it would only be useful in systems with an on-line data acquisition module installed. However, so far such systems are not common at all.

The "offset-time"[6] between two traffic lights is optimized using Artificial Neural Networks (ANNs) at López, Hernandez, Hernandez, and Garcia (1999). Although our system does not treat explicitly the offset time parameter we think that our system faces traffic optimization in a much more flexible manner.

In GiYoung, JeongJin, and YouSik (2001) a real-time local optimization of one intersection technique is proposed. It is based on fuzzy logic. Although an adaptive optimization may be very interesting – we checked out this in Sánchez, Galán, and Rubio (2004) – we believe that a global optimization is a more complete approach

---

[6] Offset-time: the time since a traffic light turns green until the next traffic light – for example, in a boulevard – turns also green.

to the problem.

In You-Sik, Hyunsoo, and Chong-Kug (1999) authors present a fuzzy control system for extending or shortening the fixed traffic light cycle. By means of electrosensitive traffic lights they can extend the traffic cycle when many vehicles are passing on the road or reduce the cycle if there are few vehicles passing. Through simulation they presented efficiency improvement results. This work performs a local adaptation for a single traffic light instead of a global optimization.

In Rouphail, Park, and Sacks (2000) an "ad hoc" architecture is used to optimize a 9 intersection traffic network. It uses Genetic Algorithms as an optimization technique running on a single machine. The CORSIM[7] model is used within the evaluation function of the GA. In this work scalability is not addressed. Authors recognize that it is a customized non scalable system. Our system has the scalability feature thanks to the intrinsic scalability of the Beowulf Cluster and the parallel execution of the evaluation function within the GA.

In Hong, Kim, Kwangson, and Park (1999) the concept of the optimal green time algorithm is proposed, which reduces average vehicle waiting time while improving average vehicle speed using fuzzy rules and neural networks. Through computer simulation, this method has been proven to be much more efficient than using fixed time cycle signals. The fuzzy neural network will consistently improve average waiting time, vehicle speed, and fuel consumption. This work only considers a very small amount of traffic signals — two near intersections — in the cycle optimization. We do agree with them about the non-suitability of fixed cycles.

An interesting combination of Genetic Algorithms and Traffic Simulation is published in Taniguchi and Shimamoto (2004). In this work a routing and scheduling system for freight carrier vehicles is presented. They use Genetic Algorithms as optimization technique. The objective of the GA is the minimization of the costs of travel. A dynamic vehicle routing algorithm is proposed and tested with a test road network. The implemented traffic simulation model is macroscopic.

Another very interesting work is presented in Varia and Dhingra (2004). A dynamic system-optimal (DSO) traffic assignment model is formulated for a congested urban network with a number of signalized intersections. They also combine traffic simulation with Genetic Algorithms. The aim of this work is to assign any traveler a route. A GA is used to minimize the users total travel time. A macroscopic model is used for the estimation of traffic delays. The DSO problem is solved with fixed signal timings, and with the optimization of signal timings.

In Vogel, Goerick, and von Seelen (2000) every intersection is optimized considering only local information. Moreover, it can be adapted to short and long term

---

[7] CORSIM: Corridor Traffic Simulation Model (Halati, Lieu, and Walker (1997)).

traffic fluctuations. In our case we perform a global optimization instead of multiple local optimizations. We think that our approach may be a more efficient exploitation of the traffic infrastructure.

A very interesting work is published in Wiering, Vreeken, van Veenen, and Koopman (2004). In this work, traffic is regarded as formed by a set of intersections to be optimized in a stand alone manner. They proposed to use reinforcement learning algorithms to optimize what they consider a multi-agent decision problem. We do not agree with them. Although a local optimization can obviously reduce average waiting times of cars – as it seems to happen with simulated tests at this work – we think that a global optimization taking into account every intersection in a zone should be more profitable.

Another very interesting work (Yoshimura (2006)) is published in the journal Computer Modeling in Engineering and Sciences. In that work a multi-agent traffic simulator is proposed. They argue considering elements like car (driver), traffic signal, and pedestrian. Therefore, the system is very complex. Finally MATES is applied to simulate city traffic in Kashiwa city in Japan, employing various real world data as input.

Own Contribution   In this subsection we have included our contribution to the art. In Sánchez, Galán, and Rubio (2004) we presented our methodology for the optimization of Traffic Light Cycles in a Traffic Network. The very good results of a parallel speed-up study convinced us that it was advisable to use a "Beowulf Cluster" as parallel computing system.

In OPTDES IV[8] we shared a scalability study on that architecture. We ran tests using four networks from 80 up to 1176 cells. In that work we found out that our system had a very good performance for all cases.

In Sánchez, Galán, and Rubio (2005b) we compared two versions of our microscopic traffic simulator: a stochastic versus a deterministic traffic simulator. There were three differences between the stochastic and the deterministic version: The cells updating order; the new vehicle creation time and the acceleration probability. From that work we realized that the stochastic simulator is a suitable – convergent – statistical process to compare with; and we demonstrated that the deterministic simulator outputs are highly linearly correlated with the stochastic ones. Therefore, our deterministic simulator can arrange the population ranking in order of fitness at least as well as the stochastic simulator, but with a remarkably lower computing time.

In the research presented for CIMCA2005 (Sánchez, Galán, and Rubio (2005a)) we

---

[8] Optimization and Design in Industry IV, Tokyo, Japan, (September, 26-30th, 2004)

described the difference between two sorts of encoding, yielding different crossover and mutation strategies. The main achievement in that work was to demonstrate – by means of a wide set of tests – that, at least for our particular case, a bit level crossover combined with a variable mutation probability means a great saving of computing time. Besides, we noticed how that choice lets the algorithm cover the solution space faster due to a bigger gene variability between generations. This combination seems to avoid premature convergence.

In ECT2006 we delivered a research (Sánchez, Galán, and Rubio (2006)) that included two goals. First, we introduced a new methodology – such a visual one – helping those practitioners occupied tuning a GA by giving them much deeper knowledge of how the GA is doing than they had before. Furthermore, we tried this new methodology with a wide set of tests. We used it for tuning the genetic algorithm within our traffic optimization architecture applied to a particular network.

We presented another research in Eurocast 2007 (Sánchez, Galán, and Rubio (2007)). In that communication we shared a study considering three candidate criteria as a first step toward extending our fitness function towards a multicriteria one. The criteria where related to the total number of vehicles that left the network, the occupancy of the network and greenhouse gases emissions. We performed a correlation study and, although conclusions where not definitive, we obtained some interesting conclusions about the relationship among those parameters.

Finally, soon we will publish an optimization research (Sanchez-Medina, Galan-Moreno, and Rubio-Royo (2008)) for another traffic network situated in Santa Cruz de Tenerife, Spain. Although the scale of that network is not as large as the one treated for the current paper, results are promising.

2 Methodology

2.1 Optimization Model

The architecture of our system comprises three items, namely a Genetic Algorithm (GA) as Non-Deterministic Optimization Technique, a Cellular Automata (CA) based Traffic Simulator inside the evaluation routine of the GA, and a Beowulf Cluster as MIMD multicomputer. Through this section we will give a wide description for the GA and the CA based Traffic Simulator used in our methodology. Finally, a brief description of the Beowulf Cluster sill also be provided.

2.1.1 Genetic Algorithm

Genetic Algorithms (GA) have become a widely used technique in many different science fields as versatile non deterministic optimization technique. For instance they are used in Rao, Rao, and Dattaguru (2004) for accelerating the computational
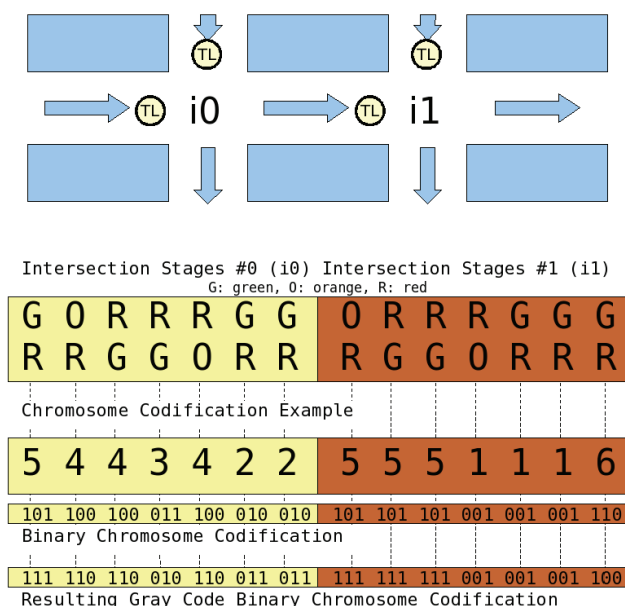
Figure 1: Chromosome Encoding

process required for automatic partitioning of unstructured meshes for parallel finite element computations.

In Sinha and Ch (2008) it is published another application of GA. A multi objective binary coded elitist non-dominated sorting genetics algorithm (NSGA-II) is used to obtain optimal solution regarding the modeling and optimization of Fluid catalytic cracking units – a key element in modern refineries.

Other interesting example is in de Lacerda and da Silva (2006) to identify polarization curves of buried slender structures for a described two-dimensional boundary element formulation.

In Jimenez-Octavio, Lopez-Garcia, Pilo, and Carnicero (2008) genetic algorithms are applied to mechanical, electrical and electromechanical optimization problems concerning the design and optimization of power transmission lines.

Another example of the use of Genetic Algorithms is in Harris, Mustata, Elliott, Ingham, and Lesnic (2008). In this paper, the steady flow of a single liquid phase through a rectangular, composite specimen, composed of two anisotropic materials with a plane contact surface, is analysed. To measure the full hydraulic conductivity tensor in rocks or soils, it is used an inverse boundary element method within a genetic algorithm maximisation procedure.

In this subsection we will describe the genetic algorithm utilized.

Optimization Criterion. Fitness Function. After testing several criteria we found out that we obtained the better results just by using the absolute number of vehicles that left the traffic network once the simulation finishes.

During the traffic simulation many new vehicles are created as if they were arriving at the inputs of the network. Furthermore, during the simulation many vehicles reach their destination point and leave the network. The number of vehicles that reach their destination point easily illustrates how the simulation was, and consequently helps us to compare a particular cycle combination with another.

Other optimization criteria tested are the following:

- Mean time at the network – Mean Elapsed Time, MET. During the simulation, the arrival and departure time of every vehicle is stored. With these values we can easily calculate the number of iterations (or seconds) it takes any vehicle to leave the network. Once the simulation finishes the average time at the network is calculated.

- Standard Deviation values of vehicle times at the network.

- A linear combination between the MET and the Standard Deviation of vehicle times at the network.

- A linear combination between the MET and the total number of vehicles that have left the network during the simulation.

- The traffic network mean occupancy density. To calculate this parameter we divided the network into small sections and counted the number of vehicles inside every section.

As we search the optimization criteria for our system we encountered an unexpected problem. If we included the minimization of the MET in a multicriteria evaluation function we provoked a very undesirable effect. The chromosomes that blocked the network faster were the best marked. That is because only a few vehicles were able to leave the network (in a small amount of iterations) before it collapsed. Hence, we obtained very "good" values but caused by "false" optimal cycle combinations. Therefore, we resigned to include that criterion in our fitness function.

Chromosome Encoding   In figure 1 we present the chosen encoding used in our methodology. In this figure we represent a chromosome example for a very simple traffic network. It consists of only two intersections and two traffic lights for each intersection.

Below the traffic network we have put the stages[9] of each traffic light separated in two different color regions, one for each one of the two intersections. The traffic light state at each *stage* may be green (G), orange (O) or red (R).

This *stages* sequence is preestablished, and will cycle *ad infinitum* – or until we stop the corresponding simulation. The objective of our system is to optimize the duration of each *stage* (in seconds) in order to get the very best traffic behavior from the network under study.

In figure 1 a chromosome encoding example is included. It can be seen that through several translation steps we obtained a binary Gray Code encoding (Black (2005)). We have proven out this methodology to be very efficient for our case in Sánchez, Galán, and Rubio (2005a).

We use Gray Code because it is designed in such a manner that when a bit changes its value – when mutation occurs – the *stage length* value only increases or decreases one unit. This is a desirable feature because it makes the search space to *conform* with the "Hamming Distance Metric".

Initial Population   Before the GA starts we created an initial population. Initially we set a time range for every preestablished *stage*. Each individual is created by choosing a random value within its corresponding range.

For this research we included within the initial population another individual that was not randomly created. That individual is the currently used cycle times combination, provided by the Zaragoza Traffic Department.

Random Number Generation   For the random number generation we have employed the MT19937 generator of Makoto Matsumoto and Takuji Nishimura, known as the "Mersenne Twister" generator. It has passed the DIEHARD statistical tests (Matsumoto and Nishimura (1998)). The seeds for that algorithm were obtained from the "/dev/urandom" device provided by the Red Hat 9 operating system.

Selection Strategy   We have chosen a Truncation and Elitism combination as selection strategy. It means that at every generation a little group of individuals — the best two individuals in our case — is cloned to the next generation. The remainder

---

[9] Stage: Every one of the states associated to an intersection, that contains a set of traffic lights.

of the next generation is created by crossovering the individuals from a best fitness subset – usually a 66 percent of the whole population.

This combination seems to be the most fitted to our problem among a set of selection strategies tested. However, we do not discard to change it if better results seem attainable.

Other selection strategies previously tested – and discarded – for this problem are succinctly explained as follows:

- Elitism: The population is ordered by fitness and a small set with the best individuals (elite) is cloned to the next generation.

- Truncation: The population is ordered by fitness. Then the population is divided into two sets, one to survive and the another one is simply discarded.

- Tournament: Small groups of individuals are chosen at random. The best fitness individual of each one of them is selected.

- Random Tournament: Like the Tournament Selection but the best individual is not always selected. It will depend on a probability value.

- Roulette Linear Selection: Every individual has a survival probability proportional to its fitness value.

- Elitism plus Random Tournament.

Crossover Operator We have tested some different crossover operators: Uniform Crossover, Two Points Crossover at fixed points and Two Points Crossover at random points. We reached the conclusion that for our case the better one was the third one.

For a couple of parents, it simply chooses two random points at each one of the two chromosomes, cut them into three pieces and then interchanges the central chunk of them.

Mutation Operator The value of a randomly chosen bit in the chromosome is just flipped.

The mutation probability is not fixed. It starts with a very high mutation probability that will decrease multiplied by a factor value in the range (0,1) until it reaches probability values near to the inverse of the population size as approaching the end of the planned number of generations.

2.1.2   Traffic Simulator

Traffic Simulation is known to be a very difficult task. There are mainly two different traffic simulations paradigms. The first one is the Macroscopic model. Macroscopic simulators are based on Fluid Dynamics, since they consider traffic flow as a continuous fluid. The second paradigm is the one that includes Microscopic simulators. For them, traffic is considered as a collection of discrete particles following some rules about their interaction. In the last decade there has been a common belief about the better performance of Microscopic simulators to do Traffic Modeling. One Microscopic model widely used is the Cellular Automata Model.

There has been a large tradition of macroscopic approaches for traffic modeling. In the 50's some "first order" continuum theories of highway traffic appeared. In the 70's and later on some other "second order" models were developed in order to correct the formers' deficiencies. References Helbing (1995); Kerner and Konhäuser (1994); Kühne and Rödiger (1991)); Kühne (1991); Payne (1979) and Witham (1974) may illustrate some of these models. However, in Daganzo (1995) "second order" models are questioned due to some serious problems like negative flows predictions and negative speeds under certain conditions.

Nowadays the microscopic simulators are widely used. One reason for this fact is that macroscopic simulators can not model the discrete dynamics that arises from the interaction among individual vehicles (Benjaafar, Dooley, and Setyawan (1997)). Cellular Automata are usually faster than any other traffic microsimulator (Nagel and Schleicher (1994b)), and, as said in Cremer and Ludwig (1986) "the computational requirements are rather low with respect to both storage and computation time making it possible to simulate large traffic networks on personal computers"

The Cellular Automata as Inspiring Model   Cellular Automata Simulators are based on the Cellular Automata Theory developed by John Von Neumann (von Neumann (1963)) at the end of the forties at the Logic of Computers Group of the University of Michigan. Cellular Automata are discrete dynamical systems whose behavior is specified in terms of local relation. Space is sampled into a grid, with each cell containing a few bits of data. As time advances, each cell decides its next state depending on the neighbors state and following a small set of rules.

In the Cellular Automata model not only space is sampled into a set of points, but also time and speed. Time becomes iterations. A relationship between time and iterations is set. For instance, $1(sec.) \equiv 1(iteration)$. Consequently, speed turns into "cells over iterations".

In Brockfeld, Kühne, Skabardonis, and Wagner (2003) we can find a well described

list of microscopic models and a comparative study of them. Although conclusions are not definitive, this work seems to demonstrate that models using less parameters have a better performance.

We have developed a traffic model based on the SK[10] model (Krauss, Wagner, and Gawron (1997)) and the SchCh[11] model (Schadschneider, Chowdhury, Brockfeld, Klauck, Santen, and Zittartz (1999)). The SchCh model is a combination of a highway traffic model (Nagel and Schreckenberg (1992)) and a very simple city traffic model (Biham, Middleton, and Levine (1992)). The SK model adds the "smooth braking" for avoiding abrupt speed changes. We decided to base our model in the SK model due to its better results for all the tests shown in Brockfeld, Kühne, Skabardonis, and Wagner (2003).
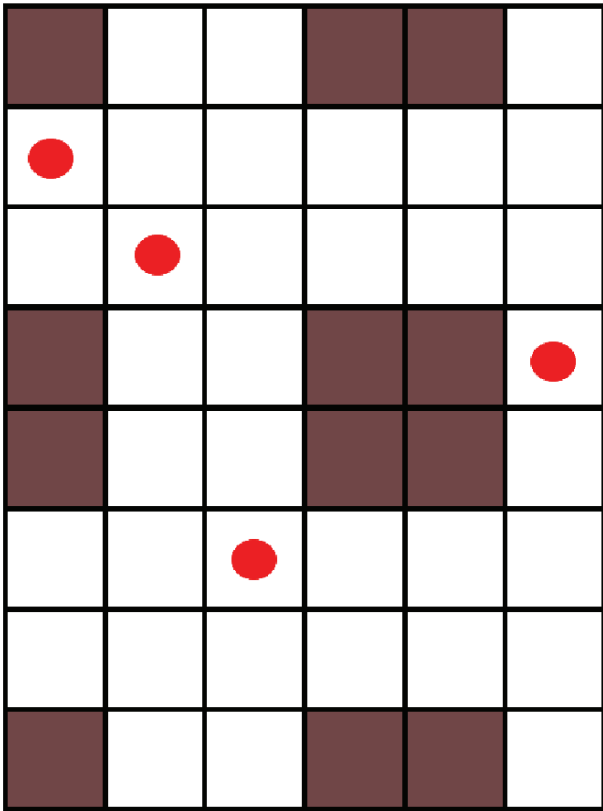

Figure 2: Paths in our Improved Cellular Automata Model

---

[10] Stephan Krauss, the author.

[11] Andreas Schadschneider and Debashish Chowdhury, the authors.

Our Improved Cellular Automata Model   Based on the Cellular Automata Model we have developed a non-linear model for simulating traffic behavior. The basic structure is the same as the one used in Cellular Automata. However, in our case we add two new levels of complexity by creating two new abstractions: "Paths" and "Vehicles".

"Paths" are overlapping subsets included in the Cellular Automata set. There is one "Path" for every origin-destination pair. To do this, every "Path" has a collection of positions and, for each one of them, there exists an array of allowed next positions. In figure 2 we try to illustrate this idea.

"Vehicles" consists of an array of structures, each one of them having the following properties:

1. Position: the Cellular Automaton where it is situated. Note that every cell may be occupied by one and only one vehicle.

2. Speed: the current speed of a vehicle. It means the number of cells it moves over every iteration.

3. Path: In our model, every vehicle is related to a "path".

These are the rules applied to every vehicle:

1. A vehicle ought to accelerate up to the maximum speed allowed. If it has no obstacle in its way (another vehicle, or a red traffic light), it will accelerate at a pace of 1 point per iteration, every iteration.

2. If a vehicle can reach an occupied position, it will reduce its speed and will occupy the free position just behind the preceding.

3. If a vehicle has a red traffic light in front of, it will stop.

4. Smooth Braking: Once the vehicle position is updated, then the vehicle speed is updated too. To do this, the number of free positions from the current position ahead is taken into account. If there is not enough free space for the vehicle to move forward on the next iteration going at its current speed (hypothetically, since in the next iteration the traffic situation may change), it will reduce its speed in one unit.

5. Multi-lane Traffic: When a vehicle is trying to move on, or update its speed, it is allowed to consider positions on other parallel lanes. For every origin-destination couple (path), at every point there exists a list of possible next positions. The first considered is the one straight forward. If this one is

not free, there may be more possible positions in parallel lanes that will be considered. Of course, this list of possible next positions is created taking the basic driving rules into account.

By means of these rules we can have lots of different path vehicles running in the same network. This model may be seen as a set of N_{paths} traditional Cellular Automata networks working in parallel over the same physical points.

Note that, so far, we are not considering a different behavior for the green and the orange state. However, our architecture is designed in such a manner that we can modify this whenever we want to, with a small effort.
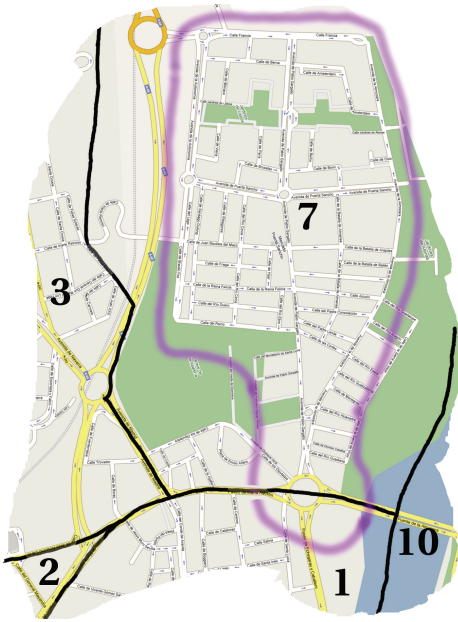

Figure 3: Urban Districts in "La Almozara"

Figure 4: Bird-eye Picture of "La Almozara"

2.1.3 Beowulf Cluster

The Architecture of our system is based on a five node Beowulf Cluster, due to its very interesting price/performance relationship and the possibility of employing Open Software on it. On the other hand, this is a very scalable MIMD computer, a very desirable feature in order to solve all sort — and scales — of traffic problems.

Every cluster node consists of a Pentium IV processor at 3.06 GHz with 1 GB DDR RAM and 80GB HDD. The nodes are connected through a Gigabit Ethernet Backbone. Every node has the same hardware, except the master node having an extra Gigabit Ethernet network card for "out world" connection.

Every node has installed Red Hat 9 on it — Kernel 2.4.20-28.9, glibc ver. 2.3.2 and gcc ver. 3.3.2. It was also necessary for parallel programming the installation of LAM/MPI (LAM 6.5.8, MPI 2).

In our application there are two kinds of processes, namely *master* and *slave* processed. There is only one master process running on each test. At every generation it sends the chromosomes (MPI_Send) to slave processes, receives the evaluation results (MPI_Recv) and creates the next population. Slave processes are inside an endless loop, waiting to receive a new chromosome (MPI_Recv). Then they evaluate it and send the evaluation result (MPI_Send).

3   Results

3.1   Performed Tests

In figures 3 and 4 we present the studied zone. "La Almozara" is the seventh district in Zaragoza - Spain.

Figure 5: Treated Zone in "La Almozara" – Zaragoza, Spain

The scale of this zone is large. In figure 5 one may note this fact. In appendix Appendix A: we enumerate the streets included and withdrawn from the model.

We have performed a wide set of tests with this zone. In this research we wanted to optimize the traffic light cycles with the current traffic situation.

Moreover, we were also interested in seeing how our system worked in congestion situations. To do so, we have created 10 hypothetical test cases varying the traffic input to the network. In table 1 we represent the current situation statistics for the new vehicle creation period for each traffic input cell in the treated network. Each integer means the number of simulation iterations between two consecutive vehicle creation instants. These values are proportional to the real traffic input flow at each input cell.

In table 2 we represent the 10 test cases periods. Note that for situation #0 there is a new vehicle at each input cell every simulation iteration. This is the worst case of congestion.

Table 1: Current Situation Traffic
Input Periods

Input	Periods
0	403
1	403
2	403
3	403
4	29
5	72
6	201
7	171
8	171
9	171
10	266
11	266
12	266
13	87
14	87
15	87

Table 2: Traffic Input Periods Tested

Situation	Periods
#0	0
#1	1
#2	2
#3	3
#4	5
#5	10
#6	20
#7	30
#8	50
#9	100

All through this research we have used a population of 200 individuals for the GA. The genetic algorithm convergence raised at different generation numbers depending on the traffic situation. In table 3 we enumerated the chosen values for that parameter.

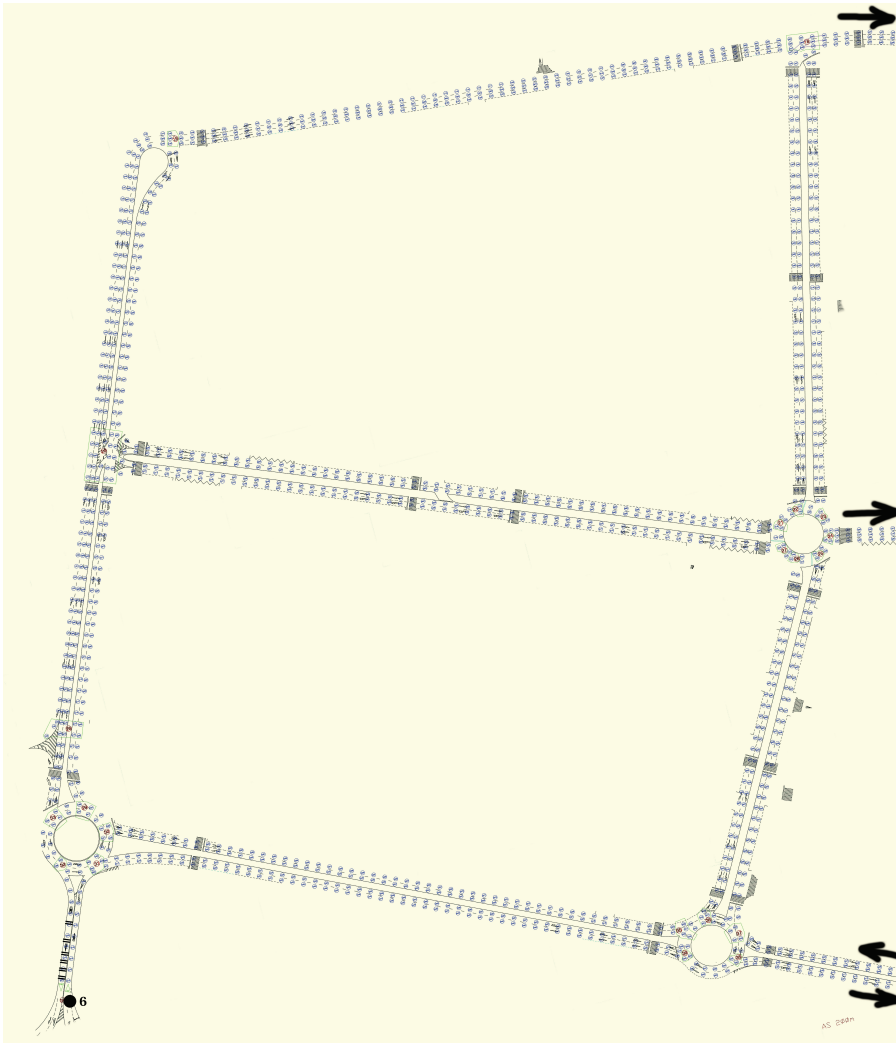The Selection Operator chosen was a combination between Elitism and Truncation.

Figure 6: "La Almozara" Discretized – #1 Third

Moreover, we have employed a variable mutation probability. It starts with a hy-permutation probability (0.999) which is decreased generation by generation, by a factor of 0.9875.

About the Cellular Automata based Traffic Simulator, we have employed a 4000 iterations simulation. In figures 6, 7 and 8 we represent the zone under study, once discretized. The whole network includes 2753 cells.
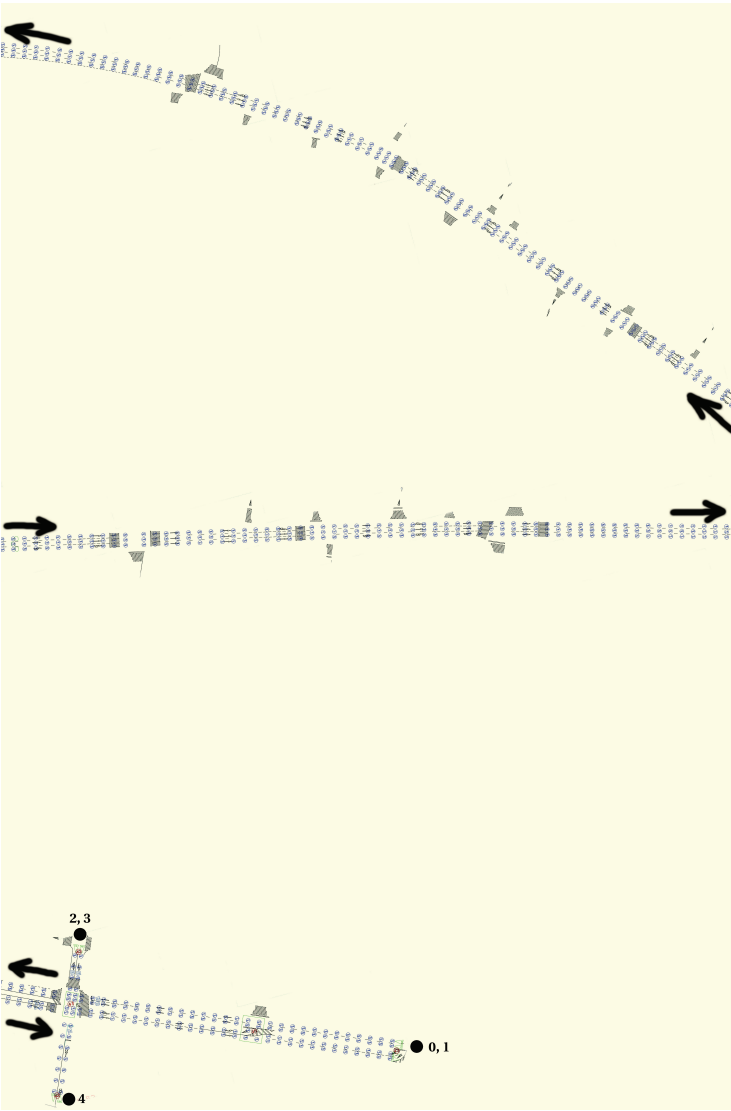
Figure 7: "La Almozara" Discretized – #2 Third

3.2 Parameters Sampled

So far, we have used a single optimization criterion. In this research it has been the maximization of the number of vehicles that left the network before the simulation finished. We are considering to include more criteria in a multiobjective optimization, but so far, this is the one more profitable we have found.
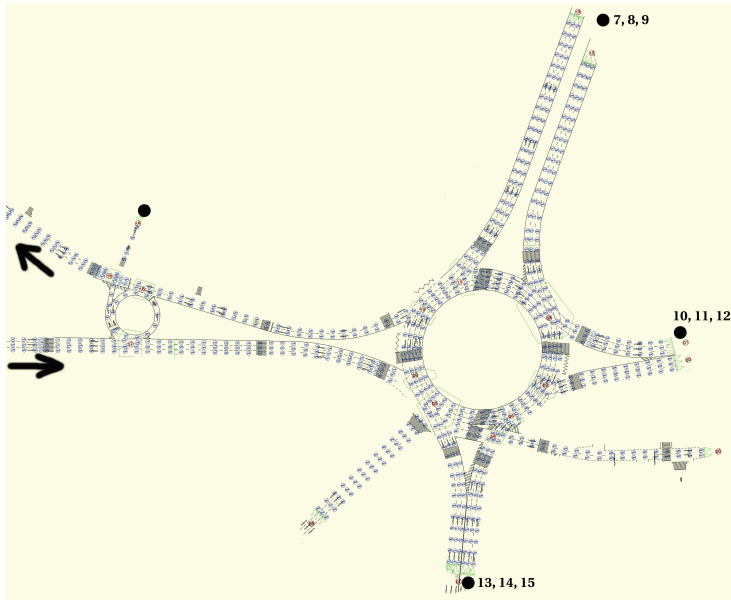
Figure 8: "La Almozara" Discretized – #3 Third

During the optimization we have stored not only the fitness values, but other parameters for every individual. We have obtained the average "State of Congestion"(SOC) and the average "Time of Occupancy" (TOC). SOC was defined in Bham and Benekohal (2004). TOC was defined in May (1990).

In equations 1 and 2 we represent both parameters.

In equation 1, N_c^o means the number of cells occupied by a vehicle, and N_c^T means the total number of cells in the treated network.

About equation 2, N_{it}^o means the number of simulation iterations that a particular cell is occupied by any vehicle, and N_{it}^T means the whole number of iterations that the simulation lasts.

As one may infer from equations 3 and 4, that the average SOC across all the simulation iterations and the average TOC across all the cells in the network are the same. In other words, the mean value of the average occupied cell ratio across all the simulation iterations and the mean value of the average number of occupied iterations for a particular cell across all the cells in the traffic network, is the same value.

Hence, for this research we have used just the average SOC. From here on we will call this statistical moment as TOC/SOC.

Table 3: Test Cases GA Generations

T. Situation	#Generations	Execution Time (s)
Current Input	400	18605
#0	400	30803
#1	300	24439
#2	250	20558
#3	250	19241
#4	250	18657
#5	200	13423
#6	200	11290
#7	200	10710
#8	200	10471
#9	200	9739

$$SOC = \frac{N_c^o}{N_c^T} \tag{1}$$

$$TOC = \frac{N_{it}^o}{N_{it}^T} \tag{2}$$

$$S\bar{O}C = \frac{\sum_{i=0}^{N_{it}^T} \frac{N_c^o(i)}{N_c^T}}{N_{it}^T} = \frac{\sum_{i=0}^{N_{it}^T} N_c^o(i)}{N_c^T * N_{it}^T} \tag{3}$$

$$T\bar{O}C = \frac{\sum_{c=0}^{N_c^T} \frac{N_{it}^o(c)}{N_{it}^T}}{N_c^T} = \frac{\sum_{c=0}^{N_c^T} N_{it}^o(c)}{N_c^T * N_{it}^T} \tag{4}$$

3.3 Restrictions Assumed

As said before, in Appendix A: we enumerate the streets included and withdrawn from the model. Moreover, in Appendix B: we listed the assumptions accomplished for the Origin-Destination probability matrix shown in Appendix B:.1.

3.4 Results

In figure 9 it is represented the maximum and average fitness obtained for each one of the traffic situations from table 2 once the genetic algorithm ends its work. Moreover, it is displayed the simulated fitness value obtained when we feed our simulator with the currently used traffic lights times and the current traffic statistics. 495 vehicles left the network for that case.

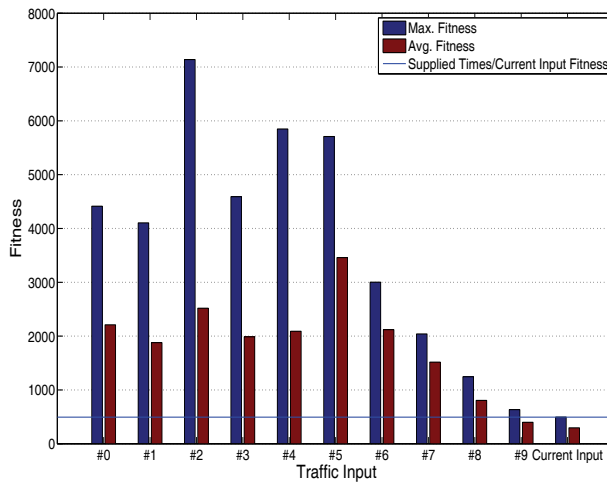In table 3 we include the mean execution time for each case.

Figure 9: Max. and Average Fitness Obtained for the 11 Situations

In figure 10 we have plotted the improvement of fitness values in percentage, using a logarithmic scale. It may be observed that the maximum fitness improvement keeps bigger than a 40 percent except for the last two cases on the right. For the #9 situation the maximum improvement is about a 7%. For the current traffic situation the improvement is over a 10%.

In figure 11 it is represented the maximum and average value of the TOC/SOC parameter (defined in 3.2). This parameter reflects the occupancy of the traffic network. A clear relationship between the fitness value (number of vehicles that left the network) and the occupancy may be seen in that figure. This is a foreseeable and desirable effect for a traffic network.

Finally, in figure 12 is shown how the better results are obtained for the first 9 traffic situations. For the situation #9 and the current traffic input modest rates are achieved, just like for the fitness parameter.
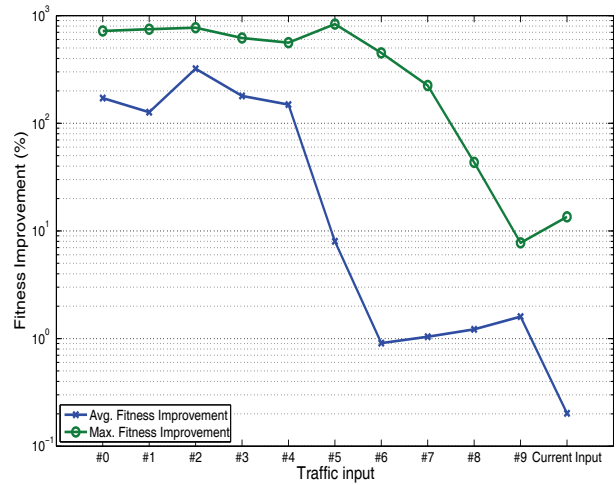
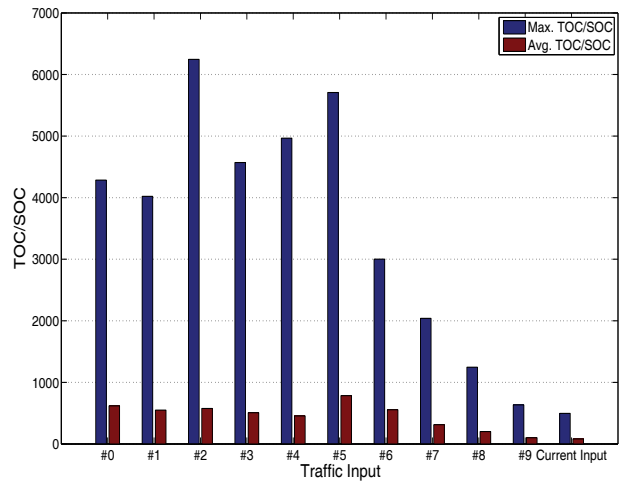Figure 10: Max. and Average Fitness Improvement for the 11 Situations

Figure 11: Max. and Average TOC/SOC Obtained for the 11 Situations

4 Discussion

Figures 9 and 10 are enlightening. In our simulated environment we obtained better fitness results using the more demanding traffic inputs. In such figure, traffic
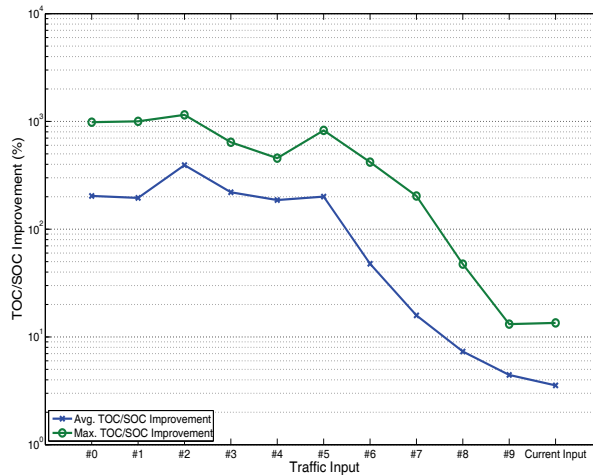
Figure 12: Max. and Average TOC/SOC Improvement for the 11 Situations

situations are ordered from high to low traffic input. For the first – more loaded – situations our systems behaves very well, improving the fitness a lot.

For the current traffic situation modest results are obtained. Everything seems to indicate that this is because that when the traffic input is so low that, no matter the traffic light cycles combination, the network would have similar outputs. In short, an empty traffic network is not likely to be improved just by optimizing the traffic lights times.

From figures 11 and 12, the main information we found out is that it seems to exist some kind of correlation between the total number of vehicles that left the network – our optimization criterion – and the occupancy of this. This conforms to common sense, and, from our point of view, is a good sign indicating a correct behavior of our traffic simulator. We plan to do more research focused on this possible correlation among the two fore mentioned parameters.

Once we focus our attention in the 6 more congested test cases in figures 9 and 11 we can give the last reflection about the presented results. Although results are very good for situations #0 to #5, for traffic situations #0 and #1 we obtained slightly worse results. For these two extreme cases the system worked a little worse.

5 Conclusions and Future Work Aims

Throughout this paper we have shared our experience with the optimization of traffic light cycles for a real world zone. Using the supplied maps and data we have simulated the current traffic behavior and moreover, we have optimized the traffic lights times in order to achieve better traffic statistics.

We want to remark that the scale of the network is really big. This research confirms previous works conclusions which indicated that our architecture is a very scalable one due to the intrinsic parallelism of genetic algorithms and the easy extendability of Beowulf clusters.

Another remarkable goal of this research is a forced congestion study. There is not much research treating this topic. In this research we have defined 10 hypothetical congestion situations and have run our system using them. The results seem promising since we obtain very good fitness rates for all cases. The tougher cases were the more improved. This is a key feature of our system, since congested networks are the most in need of optimization.

Further research will be done about the correlation between TOC/SOC parameters and our optimization criterion – total number of vehicles that left.

Finally, we plan to test this methodology using more realistic scenarios – currently congested networks – and see how it does.

Appendix A: Streets In/Out of Our Model

Appendix A:.1   Streets Taken into Account

In this subsection we enumerate the streets under consideration in this work.

- Avenida de Francia

- Avenida de la Almozara

- Avenida de Pablo Gargallo

- Avenida de la Autonomía

- Avenida Puerta de Sancho

- Calle del Lago

- Calle de Braulio Foz

- Calle de Iriarte de Reinoso (a fragment)

- Calle de Juan Bautista del Mazo (a fragment)

- Calle del Río Alcanadre

- Calle de los Diputados (a fragment)

- Paseo de María Agustín (a fragment)

- Paseo de Echegaray y Caballero (a fragment)

- Calle de Santa Lucía

- Puente de la Almozara (a fragment)

Appendix A:.2   Streets Withdrawn from Our Model

In this subsection the streets withdrawn for simplifying purposes are enumerated.

- Calle de Mónaco

- Calle de Berna

- Calle Jardines de Lisboa

- Calle de Viena

- Calle de Bruselas

- Calle de París

- Calle de Berlín

- Calle de Bohn

- Calle de Amsterdam

- Calle Jardines de Atenas

- Calle de Oslo

- Calle de la Batalla de Almansa

- Calle de la Batalla de Arapiles
- Calle de la Batalla de Bailén
- Calle Ainzón
- Calle del Padre Consolación
- Calle del Padre Landa
- Calle de las Cortes
- Calle del Río Guadalope
- Calle del Río Aragón
- Calle de la Sierra de Vicor
- Calle del Río Esera
- Calle del Río Guatizalema
- Calle de Monegros
- Calle de Dionisio Casañal
- Calle del Río Guadiana
- Calle de Santiago Dulong
- Calle de Ribagorza
- Calle del Río Cinca
- Calle de Fraga
- Calle de Hijar
- Calle de la Reina Felicia
- Calle del Río Ebro
- Calle del Río Duero
- Calle del Reino
- Calle del Monasterio de Santa Lucía

Appendix B: Approximations for the Origin-Destination Matrix

On this appendix we present the approximations assumed for the origin-destination probability matrix calculation.

- Traffic leaving district #7 to district #1 divides up into two equal parts going by Paseo de Echegaray y Caballero, and Calle de Santa Lucía.

- Traffic leaving district #7 to district #3 divides up into three equal parts going by Calle de Iriarte de Reinoso, Calle de los Diputados and Avenida de Francia.

- All vehicles leaving district #7 by Avenida de Francia go to district #3.

Appendix B:.1   Origin-Destination Probability Matrix for this Work

In this appendix we are showing the Origin-Destination Probability Matrix employed. For generating this matrix we have used Average Daily Traffic[12] statistics between districts for a weekday and the approximations listed in appendix Appendix B:.

Table 4: Origin-Destination Probability Matrix Employed

Outputs	00	05	07	08	12	34	42	43
	3.83	3.83	33.10	10.68	19.51	10.68	10.68	7.68

References

Aycin, M. F.; Benekohal, R. F. (1998):   Linear acceleration car-following model development and validation. *Transportation research record*, , no. 1644, pp. 10–19.

Aycin, M. F.; Benekohal, R. F. (1999):   Comparison of car-following models for simulation. *Transportation research record*, , no. 1678, pp. 116–127.

Barcelo, J.; J.L., F.; Garcia, D.; Florian, M.; Le Saux, E. (1996):   The Parallelization of AIMSUN2 microscopic simulator for ITS applications. In *Proceedings of The 3rd World Congress on Itelligent Transportation Systems*.

---

[12] Average Daily Traffic (ADT): The total volume during a given time period in whole days greater than one day and less than one year divided by the number of days in that time period

Benjaafar, S.; Dooley, K.; Setyawan, W. (1997): Cellular Automata for Traffic Flow Modeling. Technical Report CTS 97-09, Intelligent Transportation Systems Institute, 1997.

Bham, G. H.; Benekohal, R. F. (2004): A high fidelity traffic simulation model based on cellular automata and car-following concepts. *Transportation Research Part C*, , no. 12, pp. 1–32.

Biham, O.; Middleton, A. A.; Levine, D. (1992): Self-organization and a dynamical transition in traffic-flow models. *Phys. Rev. A*, vol. 46, no. 10, pp. R6124–R6127.

Black, P. E. (2005): "Gray code", from Dictionary of Algorithms and Data Structures, Paul E. Black, ed., NIST, 2005.

Brockfeld, E.; Barlovic, R.; Schadschneider, A.; Schreckenberg, M. (2001): Optimizing traffic lights in a cellular automaton model for city traffic. *Phys. Rev. E*, vol. 64, no. 5, pp. 056132.

Brockfeld, E.; Kühne, R.; Skabardonis, A.; Wagner, P. (2003): Towards Benchmarking Microscopic Traffic Flow Models. *Transportation Research Record*, , no. 1852, pp. 124–129.

Cremer, M.; Ludwig, J. (1986): A fast simulation model for traffic flow on the basis of Boolean operations. *Mathematics and Computers in Simulation*, , no. 28, pp. 297–303.

Daganzo, C. (1995): Requiem for second order fluid approximations of traffic flow. *Transportation Research B*.

de Lacerda, L.; da Silva, J. M. (2006): A dual bem genetic algorithm scheme for the identification of polarization curves of buried slender structures. *CMES: Computer Modeling in Engineering and Sciences*, vol. 14, no. 3, pp. 153–160.

Febbraro, A. D.; Giglio, D.; Sacco, N. (2002): On applying Petri nets to determine optimal offsets for coordinated traffic light timings. In *The IEEE 5th International Conference on Intelligent Transportation Systems*, pp. 773 – 778.

Florian, D. G. (2004): Simulation-based evaluation of Advanced Traveler Information Services, 2004. Thesis.

GiYoung, L.; JeongJin, K.; YouSik, H. (2001): The optimization of traffic signal light using artificial intelligence. In *Fuzzy Systems, 2001. The 10th IEEE International Conference on*, volume 3, pp. 1279–1282.

Hafstein, S. F.; Chrobok, R.; Pottmeier, A.; Schreckenberg, M.; Mazur, F. (2004): A High-Resolution Cellular Automata Traffic Simulation Model with Application in a Freeway Traffic Information System. *Computer-Aided Civil and Infrastructure Engineering*, , no. 19, pp. 338–350.

Halati, A.; Lieu, H.; Walker, S. (1997): CORSIM - Corridor Traffic Simulation Model. In *The 76th Annual Meeting of the Transportation Research Board*.

Harris, S. D.; Mustata, R.; Elliott, L.; Ingham, D. B.; Lesnic, D. (2008): Numerical identification of the hydraulic conductivity of composite anisotropic materials. *CMES: Computer Modeling in Engineering and Sciences*.

Helbing, D. (1995): An improved fluid dynamical model for vehicular traffic. *Physical Review. E*, , no. 51, pp. 3164.

Hong, Y. S.; Kim, J.; Kwangson, J.; Park, C. (1999): Estimation of optimal green time simulation using fuzzy neural network. In *Fuzzy Systems Conference Proceedings, 1999. FUZZ-IEEE '99*, pp. 761 – 766.

Jimenez-Octavio, J.; Lopez-Garcia, O.; Pilo, E.; Carnicero, A. (2008): Coupled electromechanical optimization of power transmission lines. *CMES: Computer Modeling in Engineering and Sciences*.

Kerner, B.; Konhäuser, P. (1994): Structure and Parameters of Clusters in Traffic Flow. *Physical Review E*, , no. 50:54.

Krauss, S.; Wagner, P.; Gawron, C. (1997): Metastable states in a microscopic model of traffic flow. *Phys. Rev. E*, vol. 55, pp. 5597 – 5605.

Kühne, R. (1991): Verkehrsablauf auf Fernstraï¿½n. *Phys. Bl.*, vol. 47, no. 3:201.

Kühne, R. D.; Rödiger, M. B. (1991): Macroscopic simulation model for freeway traffic with jams and stop-start waves. In *WSC '91: Proceedings of the 23rd conference on Winter simulation*, pp. 762–770, Washington, DC, USA. IEEE Computer Society.

Li, L.; Tang, N.; Mu, X.; Shi, F. (2004): Implementation of Traffic Lights Control Based on Petri Nets. In *Intelligent Transportations Systems, IEEE 2003*, volume 3, pp. 1749–1752.

Logi, F.; Ritchie, S. (2001): Development and evaluation of a knowledge-based system for traffic congestion management and control. *Transportation Research Part C: Emerging Technologies*, vol. 9, no. 6, pp. 433–459.

López, S.; Hernandez, P.; Hernandez, A.; Garcia, M. (1999): Artificial Neural Networks as useful tools for the optimization of the relative offset between two consecutive sets of traffic lights. *LNCS. Foundations and Tools for Neural Modeling. Springer-Verlag*, pp. 795–804.

Matsumoto, M.; Nishimura, T. (1998): Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator. *ACM Transactions on Modeling and Computer Simulation*, vol. 8, pp. 3–30.

May, A. (1990): . Prentice Hall.

Nagel, K.; Barrett, C. (1997): Using microsimulation feedback for trip adaptation for realistic traffic in Dallas. Technical Report LA-UR 97-1334, 1997.

Nagel, K.; Schleicher, A. (1994): Microscopic traffic modeling on parallel high performance computers. *Parallel Comput.*, vol. 20, no. 1, pp. 125–146.

Nagel, K.; Schleicher, A. (1994): Microscopic traffic modeling on parallel high performance computers. *Parallel Computing*, vol. 20, no. 1, pp. 125–146.

Nagel, K.; Schreckenberg, M. (1992): A Cellular Automaton Model for Freeway Traffic. *Journal de Physique I France*, vol. 2, no. 33, pp. 2221–2229.

Payne, H. (1979): Freflo: A macroscopic simulation model of freeway traffic. *Transportation Research Record*, , no. 722:68.

Rakha, H.; Van Aerde, M.; Bloomberg, L.; Huang, X. (1998): Construction and calibration of a large scale microsimulation model of the salt lake area. Technical Report 1644, 1998.

Rao, A. R. M.; Rao, T. A.; Dattaguru, B. (2004): Generating optimised partitions for parallel finite element computations employing float-encoded genetic algorithms. *CMES: Computer Modeling in Engineering and Sciences*, vol. 5, no. 3, pp. 213–234.

Rouphail, N.; Park, B.; Sacks, J. (2000): Direct signal timing optimization: Strategy development and results, 2000.

Sánchez, J.; Galán, M.; Rubio, E. (2004): Genetic Algorithms and Cellular Automata: A New Architecture for Traffic Light Cycles Optimization. In *Proceedings of The Congress on Evolutionary Computation 2004 (CEC2004)*, volume 2, pp. 1668–1674.

Sánchez, J.; Galán, M.; Rubio, E. (2005):   Bit Level Versus Gene Level Crossover in a Traffic Modeling Environment. In *International Conference on Computational Intelligence for Modelling Control and Automation - CIMCA'2005 (In press)*.

Sánchez, J.; Galán, M.; Rubio, E. (2005):   Stochastic Vs Deterministic Traffic Simulator. Comparative Study for its Use within a Traffic Light Cycles Optimization Architecture. In *Proceedings of The International Work-conference on the Interplay between Natural and Artificial Computation (IWINAC)*, volume 2, pp. 622–631.

Sánchez, J. J.; Galán, M. J.; Rubio, E. (2006):   A visual and statistical study of a real world traffic optimization problem. In Topping, B. H. V.; Montero, G.; Montenegro, R.(Eds): *Proceedings of the Fifth International Conference on Engineering Computational Technology*, Stirlingshire, United Kingdom. Civil-Comp Press. paper 147.

Sánchez, J. J.; Galán, M. J.; Rubio, E. (2007):   Study of correlation among several traffic parameters using evolutionary algorithms: Traffic flow, greenhouse emissions and network ocuppancy. In *Proceedings of the EUROCAST 2007 conference*.

Sanchez-Medina, J. J.; Galan-Moreno, M. J.; Rubio-Royo, E. (2008):   Applying a traffic lights evolutionary optimization technique to a real case: "las ramblas" area in santa cruz de tenerife. *IEEE Transactions on Evolutionary Computation*.

Schadschneider, A.; Chowdhury, D.; Brockfeld, E.; Klauck, K.; Santen, L.; Zittartz, J. (1999):   A new cellular automata model for city traffic. In *Traffic and Granular Flow '99: Social, Traffic, and Granular Dynamics*. Springer.

Schutter, B. D.; Moor, B. D. (1997):   Optimal Traffic Light Control for a Single Intersection. In *Proceedings of the 1997 International Symposium on Nonlinear Theory and its Applications (NOLTA'97)*, pp. 1085–1088.

Sinha, S.; Ch, P. (2008):   Optimization of industrial fluid catalytic cracking unit having five lump kinetic scheme using genetic algorithm. *CMES: Computer Modeling in Engineering and Sciences*, vol. 32, no. 2, pp. 85–102.

Smith, M. (1988):   Optimum network control using traffic signals. In *UK Developments in Road Traffic Signalling, IEE Colloquium on*, pp. 8/1 – 8/3.

Spall, J.; Chin, D. (1994):   A model-free approach to optimal signal light timing for system-wide traffic control. In *33rd IEEE Conference on Decision and Control, 1994*, pp. 1868 – 1875.

Taniguchi, E.; Shimamoto, H. (2004): Intelligent transportation system based dynamic vehicle routing and scheduling with variable travel times. *Transportation Research Part C: Emerging Technologies*, vol. 12, no. 3-4, pp. 235–250.

Tveit, O. (2003): Common cycle time - a strength or barrier in traffic light signaling. *Traffic Engineering and Control (TEC) Magazine*, vol. 1, no. 44, pp. 19–21.

Varia, H.; Dhingra, S. (2004): Dynamic Optimal Traffic Assignment and Signal Time Optimization Using Genetic Algorithms. *Computer-Aided Civil and Infrastructure Engineering*, , no. 19, pp. 260–273.

Vogel, A.; Goerick, C.; von Seelen, W. (2000): Evolutionary Algorithms for Optimizing Traffic Signal Operation. In *ESIT 2000*, pp. 83–91.

von Neumann, J. (1963): *The General and Logical Theory of Automata*, volume 5. A. H. Taub (ed.), Macmillan, New York.

Wey, W.-M. (2001): Applications of linear systems controller to a cycle-based traffic signal control. In *Intelligent Transportation Systems, 2001*, pp. 179 – 184.

Wiering, M.; Vreeken, J.; van Veenen, J.; Koopman, A. (2004): Simulation and Optimization of Traffic in a City. In *Intelligent Vehicles Symposium, IEEE 2004*, pp. 453–458.

Witham, G. (1974): Linear and Nonlinear Waves. *Wiley, New York*.

Yang, Q. (1997): A Simulation Laboratory for Evaluation of Dynamic Traffic Management Systems, 1997.

Yoshimura, S. (2006): Mates : Multi-agent based traffic and environment simulator – theory, implementation and practical application. *CMES: Computer Modeling in Engineering and Sciences*, vol. 11, no. 1, pp. 17–26.

You-Sik, H.; Hyunsoo, J.; Chong-Kug, P. (1999): New Electrosensitive Traffic Light Using Fuzzy Neural Network. In *IEEE Transactions on Fuzzy Systems*, volume VII, pp. 759–767.