Optimization of a complex urban intersection using discrete event simulation and evolutionary algorithms

Adriana Simona Mihăiţă * **Mauricio Camargo** *
Pascal Lhoste *

* *Université de Lorraine, ERPI, EA6737, 8 rue Bastien Lepage, 54010 Nancy, France*

Abstract: Dealing with traffic management for complex crossroads is a challenging problem for traffic control planners. As a contribution to solve this problem, the present paper develops a mesoscopic simulation model for detecting the most suited fire plan for a complex road intersection, using a discrete event simulation tool and an evolutionary algorithm optimization. The modeling goal is to eliminate congestion by choosing an appropriate fire plan which will be adapted to the actual configuration of the intersection, as well as to a future reconfiguration meant to accept a higher inflow of vehicles. The proposed model is applied to a down-town crossroads from Nancy, France. Four different configurations of the input data flow were studied under the proposed simulation-optimization approach, and an optimal fire plan is proposed.

Keywords: mesoscopic road traffic simulation, evolutionary algorithms, optimization

1. INTRODUCTION

In a modern world concerned by global warming and environmental crisis, efficient transportation systems become a major preoccupation when trying to solve mandatory problems such as : congestion, fluidity in transport, rapidity to reach the working place, pollution, etc. The continuous development of urban agglomerations requires new infrastructure, reconfiguration and rethinking of the transportation systems. Often, the extension of the actual infrastructure is impossible due to the large costs this action would imply or to the lack of space. Studies have shown that the simple expansion of the traffic infrastructure will not solve the congestion problems, but moreover will induce a bigger demand for traveling and rapid depletion of the additional capacity.

Therefore a traffic optimization becomes mandatory. This would improve mobility, safety, congestion and of course, the time spent in traffic. In the last decade, efforts have been made towards the implementation of Intelligent Traffic Systems (ITS), which would provide drivers with useful traffic information or traffic forecasts via electronic panels, Internet or radio. For example, Hafstein et al. (2004) proposed a freeway Traffic Information System based on high resolution cellular automata; by running a java applet in a web page they provide users with useful information after simulating current traffic zones every 30 and 60 minutes.

As the feedback information strategy is very important in ITS, studies are also interested in evaluating, for example, the impact of the travel time feedback strategies (in Wahle et al. (2000) the new entering vehicles in the intersection will chose the route will less passing time) or the vacancy length feedback strategy (the traffic control displays as each time step the distance from the last vehicle to the entrance of each road, leading the drivers to choose the road with longer vacant distance). Multiple comparisons and simulation results of the above methods can be found in (Chen et al. (2012), Wang et al. (2005)).

The simulation of the traffic flow becomes a powerful tool for the analysis, the reproduction and the foresight of a wide variety of problems, which would be difficult to analyze with real traffic tests. The main challenge remains of course the optimization and improvement of the traffic simulation in order to obtain accurate and realistic results. But one of the most important problems in traffic optimization is choosing the right traffic light plan, as it has a strong impact on the traffic flow results: Brockfeld et al. (2001). This is a combinatorial problem which is difficult to solve by deterministic methods. Regarding the fact that various works combine traffic simulation tools and genetic algorithms (GAs), and obtain encouraging results, we concentrate our attention on bio-inspired optimization methods such as evolutionary algorithms (EAs) which we present in this article, in combination with a new traffic simulation tool, FlexSim.

Our main challenge is to find an optimal traffic plan for a complex crossroads, knowing that many optimal and local solutions for the problem may exist. The method has been tested on the intersection C129 from downtown Nancy France, containing three main junctions. The intersection is a part of the new ecological quarter "Nancy Grand Cœur" currently under projection for future reconfiguration. The main goal of the future reconfiguration is to be able to absorb a bigger traffic inflow, and therefore to choose an appropriate fire plan.

* The authors of this work are grateful to the data and support provided by the Urban Community of Grand Nancy (CUGN).

The current paper is organized as follows. In Section 2 we present the state of the art concerning different traffic simulation tools, followed by the introduction of a new simulation tool: FlexSim. Section 3 is reserved to the discussion of different optimization approaches in traffic control, followed by the presentation of our evolutionary algorithm in subsection 3.1. In Section 4 we present the C129 intersection from Nancy, by explaining the implementation in FlexSim (subsection 4.1), the statistics and the scenarios we have tested (subsection 4.2), as well as the optimization results (subsection 4.3). The model has been built using the data from the Urban Community of Grand Nancy (CUGN) [1]. The last part of the article presents the interpretation of the results and the future perspectives of this work.

2. TRAFFIC SIMULATION TOOLS

During the last decades, a wide range of traffic flow models and theories have been developed in order to respond to the traffic congestion problems. These models have been designed according to either the scale of the application (networks, links, intersections), the representation of the process (deterministic, stochastic) or the scale of independent variables (continuous, discrete). But one of the most popular classification criterion is the level of details, or more explicitly, the level of description of the traffic entities, as in Hoogendoorn and Bovy (2001). In the following present a brief summary of some of the most popular traffic simulators according to the level of detail.

2.1 Microscopic or sub-microscopic simulators

The microscopic traffic models describe in high detail the behavior of the simulated entities entering the system (vehicles, drivers), and calculates at each time step the position, the speed and the acceleration of each entity, or the behavior of the drivers. Consequently, they are more suited for small urban areas or special transportation facility studies, needing fast computers with powerful resources. Nowadays some of the most popular microsimulation tools are CORSIM: Fellendorf and Vortisch (2010), VISSIM: Hidas (2005), PARAMICS: Cameron and Duncan (1996), and DYNASIM: Nishimoto et al. (2002). If CORSIM and VISSIM are widely used in China, PARAMICS and DYNASIM have been more popular in Europe. However the great diversity of microsimulation models can raise difficulty when choosing the appropriate simulation platform. For a technical and comparative analysis between some of the above mentioned micro-simulators, the reader is directed to recent studies of Sun et al. (2013).

2.2 Macroscopic simulators

The macroscopic simulation models describe the traffic at a high level of aggregation, as a flow, using characteristics such as density, velocity or flow-rate. They offer a global description of the traffic, using for example, differential equations, instead of modeling the movement of each vehicle in the simulation. For example, METACOR: Salem and Papageorgiou (1998), is a macro-simulation model in which the traffic network is represented as a graph having the entrances (or exits) as nodes. Other macroscopic simulation models can be found in the works of (Zegeye et al. (2013), Li et al. (2011)).

2.3 Mesoscopic simulators

The mesoscopic flow simulators reproduce the traffic flow at an intermediate level between the microscopic and the macroscopic level. They do not distinguish nor trace the individual behavior of the vehicles in the system, but specify in terms of probabilities the behavior of small groups of traffic entities moving together.

Every connection inside the model influences the next computation, as the average passage time in a segment is influenced by the flow, the occupation of the segment or its capacity. Some examples of mesoscopic simulators are: TransModeler [2], DynaSmart [3] and Metropolis from Palma and Marchal (2002), which offers the possibility of modeling the departure time choice and a day-to-day dynamics. Although interesting simulation features can be found in the previous examples, new mesoscopic simulators need to be taken into consideration. An efficient discrete event simulator is FlexSim which we present in the next section.

2.4 FlexSim simulation

FlexSim is a powerful analysis and simulation tool which allows to model, visualize and optimize real-life process, from manufacturing to supply chains [4]. With FlexSim the users can interact with common spreadsheet database applications in order to build realistic 3-dimensional models with model charts and graphs which can dynamically display output statistics during the simulation. Using the integrated built-in experimenter, we can optimize and test many scenarios over a chosen number of replications, stock results in either global tables or export the final visual representations for later use.

The facility of 3-D testing and visualization made FlexSIm popular for optimization problems in distribution centers, Hou (2013), production and assembly plants or even hospitals. In Cimino et al. (2010), the authors present a detailed comparison and evaluation between some of the most popular software used for the discrete event simulation : Arena, Witness, Promodel, Anylogic, Automod, Emplant and FlexSim. In addition to the presented simulation software, FlexSim allows to create its own libraries and classes of objects, graphical user interfaces, applications. Although Arena can be as well used for the traffic flow simulation: Wen (2008), the flowchart for a simple crossroads becomes complex and difficult to follow, especially when congestion problems occur. For this reason a 3D graphical representation of the crossroads will offer a more realistic perception of real-life situations, such as detection or visualization of collisions.

All the above features have led us to test and use FlexSim for the mesoscopic traffic simulation model that we present in this paper. The results of the FlexSim simulation will be presented in Section 4.3.

[1] http://www.grand-nancy.org/

[2] www.caliper.com/transmodeler/

[3] www.its.uci.edu/ paramics/Models.html

[4] www.flexsim.com/

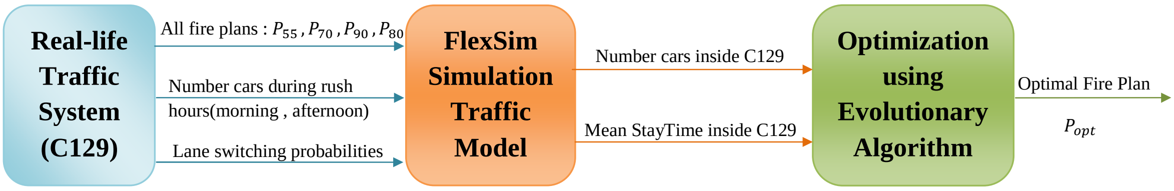

Fig. 1. Block diagram of the traffic simulation and optimization procedure.

3. TRAFFIC OPTIMIZATION SYSTEMS

Although a traffic flow simulation is a very good reproduction of the real-life situation, it remains a method of testing scenarios and compare the results in order to take the best decision. Once the simulation model has been defined, optimization could be easily reached through an iterative procedure, in the case of a simple system, but this becomes hard as the system complexity rises.

With the progressive growth of the urban traffic networks, the decentralization of the traffic control and decision making has become inherent. Several techniques based on artificial intelligence, such as multi-agent systems, Fuzzy logic or artificial neural networks (ANNs), have raised as important simulation and optimization techniques, which respond to the needs of a distributed control with the goal of reaching an optimum traffic state. A detailed comparative analysis of each of the above techniques can be found in Liu (2007), or more recently in Qureshi and Abdullah (2013). Some other interesting approaches are the ones using Petri nets as a simulation tool combined either with PLC or Matlab Simulink: Voinescu et al. (2009); Lin et al. (2003).

In the urban traffic control, an important aspect is the optimization of the fire plans. As this can be regarded as a non-convex nonlinear programming problem, finding a global optimal solution is difficult to achieve by traditional mathematical methods. Evolutionary algorithms (EA) have the advantage of finding solutions for such problems, as proved by the works of Medina: Sanchez Medina et al. (2008), in which they combine a Genetic Algorithm (GA) as an optimization technique, with a traffic microscopic simulator, and apply it to traffic junctions from Santa Cruz de Tenerife. Other interesting combinations which motivate our choice of combining EA and traffic simulation can be found in (Zhiyong (2006), Anfilets and Shuts (2012)).

One of the advantages of using the evolutionary optimization procedures (EO), is the fact that EOs use stochastic operators, without gradient information in the search process, and use more than one solution in an iteration (a population approach), unlike most classical optimization algorithms, which update one solution at each iteration: Deb and Kalyanmoy (2001). The evolutionary algorithm which we adopt in this paper is presented in the next section.

3.1 The evolutionary algorithm (EA)

During the traffic simulation, the vehicles are being generated at the main entrances inside the model, and recuperated once they leave the network. The number of vehicles leaving the network, as well as the mean average time spent inside the network give us the optimization criteria which is tested using the evolutionary algorithm we present in this section. Our objective is to decrease the mean time that the vehicles spend inside the traffic model, in order to increase the number of vehicles exiting the model, and thus increase the traffic flow. This would allow to choose an adapted fire plan for the intersection, from the existing available fire plans to be tested.

In Figure 1 we represent the logical schema of the traffic simulation model we propose and the optimization procedure. Based on the information we receive from the real-world traffic intersection: the current available fire plans for testing (P_{55}, P_{70}, P_{90}, P_{80}, see Section 4.1), the number of total cars entering the intersection during rush hours, and the probability of lane switching, we build the simulation model in FlexSim. The simulation model offers the possibility of determining the mean number of cars and the average staytime inside the intersection, which will be used in the EA optimization procedure. The proposed EA will then determine the optimal fire plan which is best adapted for the C129 intersection.

The algorithm we present in this paper is based on the general approach of evolutionary algorithms given in Deb and Kalyanmoy (2001). This algorithm has been selected due to a good compromise between the execution time and the computational precision, as shown in Perrin et al. (1997). The complete outline of the algorithm is given in Algorithm 1, containing the following steps:

(1) The current EA is an iterative optimization process, starting from an initial population of *nind* individuals, which are supplied by the traffic simulation model, and which are characterized by two variables: the mean number of cars and the average stay-time inside the intersection. The function **initialise_population** is responsible for the initialization of all the individuals inside the algorithm. By $P^{(ngen)}$ we denote the whole population we are creating at each generation of individuals.
(2) The next step is the evaluation of the population, by computing the objective criteria we have defined, inside the function **calculate_objective_fct**.
(3) Once the objective criteria has been computed for all the individuals of the current population, the next step is to sort and select the best individuals which we call: *survivors*, using the function **select_best_indiv**. This step is usually known as a sorting of solutions from best to worst, and can be also achieved by computing a domination score: Halsall-Whitney and Thibault (2006).
(4) At this point, we have a selection of best individuals. Now, we randomly generate the mutants of the popu-

Algorithm 1 Outline of the evolutionary algorithm.

Require: *nind* (the number of individuals in a population), *ngmax* (the number of maximum generations to be created);
Ensure: P - the optimized population ;
Parameters: *nsurv* (number of survivors), *nmut* (number of mutants), *ngen* (number of populations)
//Step 1: construct initial population from simulation
$ngen = 0$;
$P^{(0)} \leftarrow$ **initialise_population()**;
while do($ngen \leq ngmax$)
//Step 2: Compute the objective criteria
for all $ind \in P^{(ngen)}$ **do**
calculate_objective_fct(ind)
end for
//Step 3: select best individuals(survivors)
$P_{surv} \leftarrow$ **select_best_indiv**($P^{(ngen)}$, *nsurv*)
//Step 4: generate mutants
$P_{mut} \leftarrow \emptyset$
for $i = 1 : nmut$ **do**
$mutant \leftarrow$ **generate_mutant()**;
$P_{mut} \leftarrow P_{mut} \bigcup mutant$
end for
//Step 5: generate children
$P_{child} \leftarrow \emptyset$
for $i = 1 : (nind - nsurv - nmut)$ **do**
$(p_1, p_2) \leftarrow$ **select_parents**(P_{surv});
$child \leftarrow$ **create_child**(p_1,p_2);
$P_{child} \leftarrow P_{child} \bigcup child$
end for
//Step 6: create the whole new population
$P^{ngen+1} \leftarrow P_{surv} \bigcup P_{mut} \bigcup P_{child}$
$ngen++$; *// increase the population counter*
end while

lation, inside the domain definition of the population, using the function **generate_mutant**.

(5) The main part of the algorithm is the creation of new individuals (children), by randomly choosing two different parents (I_{p1} and I_{p2}) from the population of survivors (function **select_parents**). The combination of these two individuals inside the function **create_child**, is made according to the equation :

$$I_{child} = D_p I_{p1} + (1 - D_p) I_{p2},$$

where D_p is a randomly selected real number between 0 and 1, each time an input I_{child} has been determined.

(6) Steps (2) to (5) will be repeated until we generate a predetermined maximal number of generations (*mgmax*), where *ngmax* is chosen by considering the expected precision of the results.

To resume, the EA we present here is a population-based stochastic search procedure, which selects the best members of a population, and uses them to recombine and perturb locally, in order to create new and better populations until the predefined goal was reached. Overall the EA offers the possibility of having a flexible optimization procedure for the traffic flow problem we are trying to solve.

4. CASE STUDY

As stated in the introduction, the first aim of this paper is to show that our proposition is able to model the traffic flow of a real-life complex intersection (C129) which is based in downtown of Nancy France (Figure 2).

Fig. 2. Aerial view of the C129 intersection from Nancy, France (Google Maps).

One of the main interests is to know which areas of the intersection are more crowded and which traffic plan would be more adapted to the new configurations, meant to receive an increased number of vehicles each day.

When analyzing C129, we can observe that the vehicles enter the intersection either from the bridge *Pont des Fusillés* which is the main artery (passing over the railway tracks), the *Joffre Boulevard* which also receives the vehicles from *Boulevard Ghetto Varsovie* (passing under the bridge) and the *Grand Rabin Haguenaeur* street which first intersects the *Cyfflé* road. The main roads to exit the C129 junction are *Abbé Didelot*, *Cyfflé* and *Joffre Boulevard* as well.

4.1 FlexSim simulation model

When building the mesoscopic simulation model for the above intersection, various elements need to be considered: the main structure and configuration of the intersection (static), the entering and exiting objects (vehicles, pedestrians, buses, etc.) as well as the traffic light plans. FlexSim offers a powerful and scalable 3D simulation environment which allows the modeling and simulation of various objects inside the junction, using simulation objects such as : conveyors for the streets, FlowBin Items for vehicles and pedestrians, Visual Tools for traffic lights, sources and sinks for generating, respectively for disposing vehicles, processors for the random insertion of vehicles, AutoCAD drawings for background, etc. A snapshot of the 3D FlexSim Simulation can be seen in Figure 3.

Our work has been done in collaboration with the CUGN which provided for us the structure of the intersection, the fire plans which would be tested, along with the directional metering of the vehicles during the rush hours: morning (07:30 - 08:30) and evenings (16:30 - 18:00), for a whole

Fig. 3. Simulation of the C129 intersection in FlexSIm.

week. We will give a short description of these parameters in the following.

We consider that after the vehicles randomly enter C129, they follow directional probabilities for switching the lanes. These probabilities are built from the metered data we have received. For example, in Figure 4, the 220 cars entering the Grand Rabin Haguenauer street, will either turn right (in proportion of 4.54%), turn left (20.46%) but most of them will continue to enter the intersection (75%).

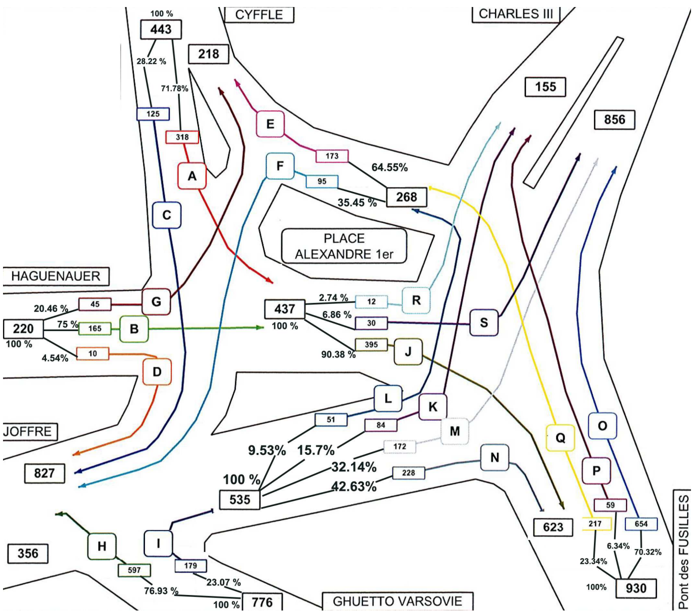

Fig. 4. Probabilities of lane changing in C129.

As stated earlier, the main objective of the simulation is to choose a suited fire plan which would ease the traffic flow during rush hours when a bigger number of vehicles would enter the intersection. By fire plan we denote the planning of the red-yellow-green cycles for all the traffic lights of the C129. Four fire plans have been implemented and tested in the simulation (lasting respectively 55, 70, 80 and 90 seconds), which give us the possibility to test different scenarios in the simulation. We note these plans : P_{55}, P_{70}, P_{80} and P_{90}, respectively.

The random characteristics of the system demands a certain number of replications to be made, in order to obtain accurate results. The method suggested by Archer and Hgskolan (2005) is to run successive simulations until the average mean and standard variation of the average stay-time (or the mean number of cars) fall within an acceptable confidence interval calculated in relation to the standard t-distribution. Using this procedure in accordance with a confidence interval of 95 per cent, the number of runs indicated approximately 10-12 runs per scenario. Given the importance of the accuracy in the results, we decided to conduct 15 simulation runs for each time-period scenario. An important aspect of FlexSim is that it can run parallel replications of the simulation model according to the number of available processors. The simulations have been made using an Intel Quad Core i7 (2.4 GHz) computer having 8 GB DDR3 SDRAM memory.

4.2 Statistics in FlexSim

The first step of the result interpretation is to compare and analyze the mean number of cars inside C129 (N_{cars}) when each traffic plan is applied, as well as the average stay-time (T_{avg}) needed to pass C129.

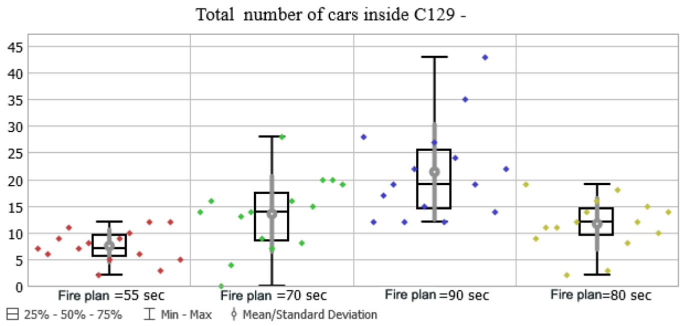

Fig. 5. The total number of cars inside C129.

Figure 5 shows the variation of the number of cars inside C129, during the morning rush hours. We can notice that the P_{90} plan seems to allow a bigger number of cars to pass the intersection and thus to be the one suitable for bigger inflow; the next step would be to verify if the plan is also suitable in terms of average stay-time.

Figure 6 shows a comparison of the T_{avg} on two different streets from C129. Although we would tend to confirm that the P_{90} plan gives the smallest stay-time on the Pont des Fusillés street: Figure 6a), we can observe that this plan would dramatically increase the waiting time on the Joffre Boulevard: Figure 6b).

This observation made us questioning the behavior of the intersection during each fire plan, when a considerably big number of vehicles will enter C129, as well as the variation of the T_{avg} versus the Nr_{cars}. We denote by D_2 the total number of vehicles entering C129 according to the configuration we have received from the CUGN (Figure 4). We will then test each fire plan (P_{55}, P_{70}, P_{90}, P_{80}) within the D_2 scenario, but as well with one the following scenarios: $D_1 = D_2/2$, $D_3 = D_2 * 2$, $D_4 = D_2 * 3$. Figure 7a) present a full representation of these experiments.

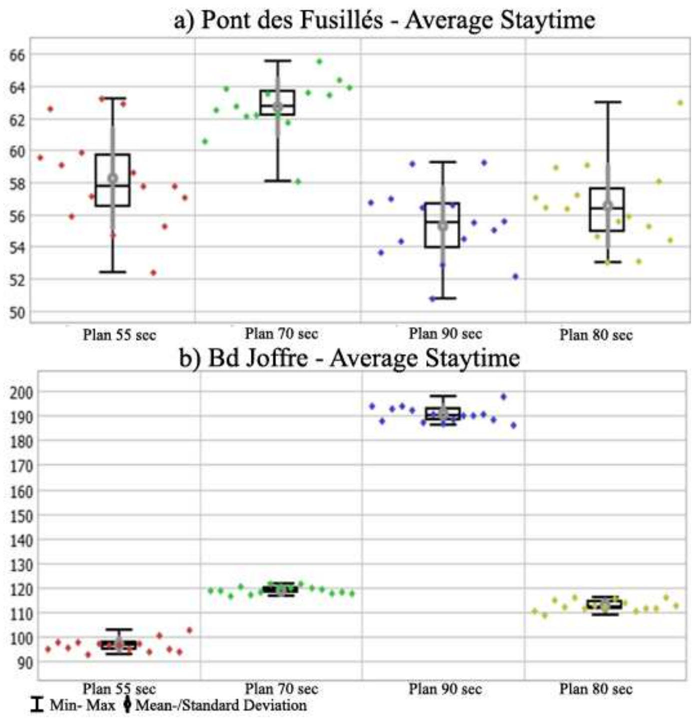

Fig. 6. T_{avg} on a) Pont des Fusillés and b) Joffre Boulevard.

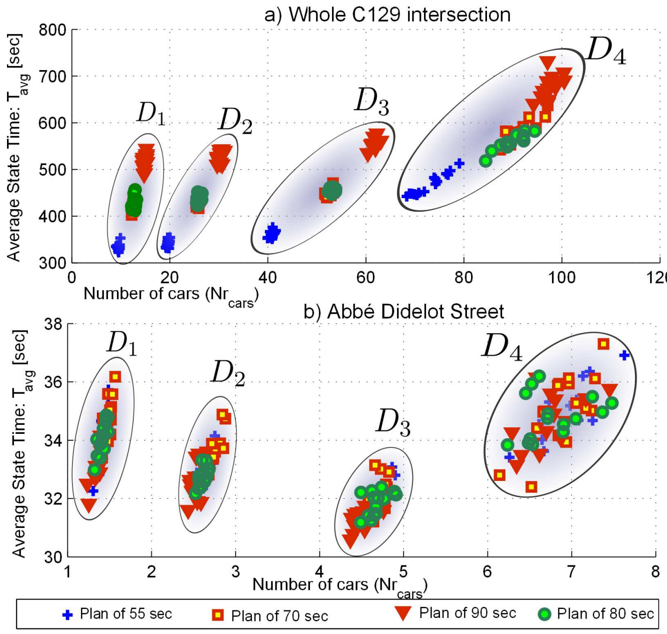

Fig. 7. Data variations on a) C129 b) Abbé Didelot Street

Based on Figure 7a), we would tend to make the following remarks: a) the average stay-time inside C129 will grow as the number of vehicles entering grows, b) the best adapted plan even when a bigger number of vehicles enter the intersection, seems to be the P_{55}, which is in a total contradiction with the previous observations. For example, when representing the same variations on the Abbé Didelot street (Figure 7b)), we can notice that there is a mix of possible plans which would better be adapted to a bigger number of vehicles (see the set $D3$).

4.3 Optimization problem

The observations from the previous section show us that we have a heterogeneous and a complex system, for which a particular optimization technique would be necessary. We therefore search to maximize the output flow of C129 (Q) during T_{avg}. In other words, we search the minimal average stay-time which would allow a maximal number of cars to pass the C129 intersection:

Maximize $\qquad Q = \frac{N_{cars}}{T_{avg}}$

subject to $\qquad T_{avg} \geq 0$ and $Nr_{cars} \geq 0$.

This would be the objective criteria we want to optimize using the EA from Section 3.1. The number of individuals are the total number of points resulted from each simulation, following the experimental plan as in Figure 7, while the number of survivors and mutants have been set according to the input data. The optimal result we obtain in Figure 8 shows us that the best fire plan which would better manage a big number of input vehicles is the P_{90} (given by the closest point to the horizontal axis of the optimum). This plan is currently being used in the intersection but for all the possible situations (bigger or lower inflow, rush hours or relaxed periods).

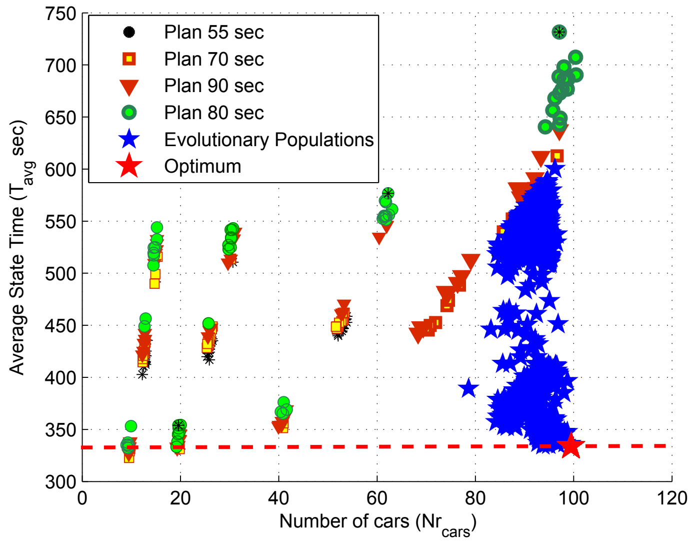

Fig. 8. Algorithm 1 applied to the whole C129

But when analyzing the results using the D_2 data set, as shown in Figure 9, we have to state that the optimal plan which would be nowadays adapted to most of the streets of C129 is the P_{70} plan.

This aspect shows us the complexity of the system and the need to conduct a deeper analysis in a systemic way, especially when proposing an optimal traffic light plan of an urban intersection.

5. CONCLUSION

In this paper we have presented a mesoscopic traffic simulation model for a complex road intersection from Nancy, France (C129). The first part of the paper presents the traffic simulation model, while the second part is

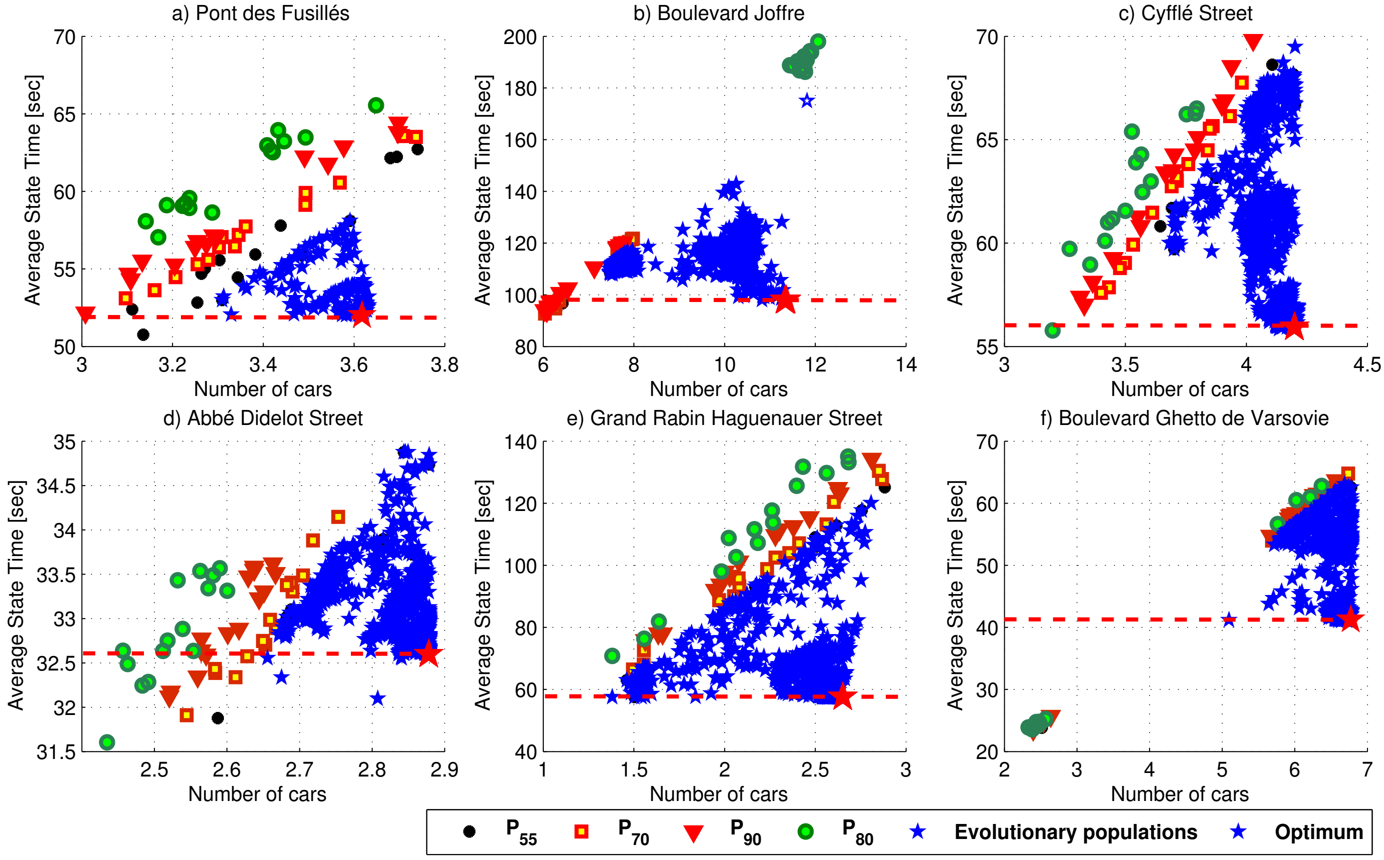

Fig. 9. A comparison of the optimal solutions obtained for each street of the C129 intersection using the EA 1.

focused on the evolutionary algorithm we have applied for the optimization problem. The results indicate which fire plan should be used with the actual configuration of the system, chosen from the current existing plans. A further perspective is to be able to optimize and choose the best adapted fire plan from all the possible fire plans we can conceive for this intersection. The C129 intersection is part of a reorganization plan of the ecological central quarter of Nancy, therefore an extension of the actual traffic model is enhanced.

ACKNOWLEDGEMENTS

The traffic simulation model was conceived at ENSGSI Nancy France [5], and has received the 1st prize in the international simulation tournament organized by FlexSim in 2013 [6].

REFERENCES

Anfilets, S. and Shuts, V. (2012). The use of natural optimization algorithms for the implementation of adaptive control at the crossroad. In *12th International Conference Reliability and Statistics in Transportation and Communication*, 227–233. Riga, Latvia.

Archer, J. and Hgskolan, K. (2005). *Indicators for Traffic Safety Assessment and Prediction and Their Application in Micro-simulation Modelling: A Study of Urban and Suburban Intersections*. Royal Institute of Technology.

Brockfeld, E., Barlovic, R., Schadschneider, A., and Schreckenberg, M. (2001). Optimizing traffic lights in a cellular automaton model for city traffic. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 64(5 Pt 2), 056132.

Cameron, G. and Duncan, G. (1996). Paramics: Parallel microscopic simulation of road traffic. *The Journal of Supercomputing*, 10(1), 25–53.

Chen, B., Xie, Y., Tong, W., Dong, C., Shi, D., and Wang, B. (2012). A comprehensive study of advanced information feedbacks in real-time intelligent traffic systems. *Physica A: Statistical Mechanics and its Applications*, 391(8), 2730–2739.

Cimino, A., Longo, F., and Mirabelli, G. (2010). A general simulation framework for supply chain modeling: State of the art and case study.

Deb, K. and Kalyanmoy, D. (2001). *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Inc., New York, NY, USA.

Fellendorf, M. and Vortisch, P. (2010). Microscopic traffic flow simulator vissim. In J. Barcelo (ed.), *Fundamentals of Traffic Simulation*, volume 145 of *International Series in Operations Research & Management Science*, 63–93. Springer New York.

Hafstein, S.F., Chrobok, R., Pottmeier, A., Schreckenberg, M., and C. 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*, 19(5), 338–350.

[5] Ecole Nationale Supérieure en Génie des Systèmes Industriels

[6] www.flexsim.com/blog/flexsim-student-tournament-in-france-and-morocco/

Halsall-Whitney, H. and Thibault, J. (2006). Multi-objective optimization for chemical processes and controller design: Approximating and classifying the pareto domain. *Computers & Chemical Engineering*, 30(6-7), 1155–1168.

Hidas, P. (2005). A functional evaluation of the AIMSUN, PARAMICS and VISSIM microsimulation models. *Road and Transport Research*, 14(4).

Hoogendoorn, S. and Bovy, P. (2001). State of the art of vehicular traffic flow modelling. In *The Delft University of Technology*, 283303.

Hou, S. (2013). Distribution center logistics optimization based on simulation.

Li, J., Zhang, B., Liu, W., and Tan, Z. (2011). Research on OREMS-based large-scale emergency evacuation using vehicles. *Process Safety and Environmental Protection*, 89(5), 300–309.

Lin, L., Nan, T., Xiangyang, M., and Fubing, S. (2003). Implementation of traffic lights control based on petri nets. In *IEEE Intelligent Transportation Systems*, volume 2, 1087–1090 vol.2.

Liu, Z. (2007). A survey of intelligence methods in urban traffic signal control. *International Journal of Computer Science and Network Security*, 7(7), 105–112.

Nishimoto, K., Fucatu, C.H., and Masetti, I.Q. (2002). DynasimA time domain simulator of anchored FPSO. *Journal of Offshore Mechanics and Arctic Engineering*, 124(4), 203–211.

Palma, A.d. and Marchal, F. (2002). Real cases applications of the fully dynamic METROPOLIS tool-box: An advocacy for large-scale mesoscopic transportation systems. *Networks and Spatial Economics*, 2(4), 347–369.

Perrin, E., Mandrille, A., Oumoun, M., Fonteix, C., and Marc, I. (1997). Optimisation globale par stratégie d'évolution : Technique utilisant la génétique des individus diploïdes. *RAIRO. Recherche opérationnelle*, 31(2), 161–201.

Qureshi, K.N. and Abdullah, A.H. (2013). A survey on intelligent transportation systems. *Middle-East Journal of Scientific Research 1*, 629–642.

Salem, H.H. and Papageorgiou, M. (1998). Metacor : a dynamic macroscopic simulation tool for corridor traffic. *CESA 1998-IEEE*.

Sanchez Medina, J., Galn Moreno, M., and Rubio, E. (2008). Evolutionary computation applied to urban traffic optimization. In W. Kosinski (ed.), *Advances in Evolutionary Algorithms*. InTech.

Sun, D., Zhang, L., and Chen, F. (2013). Comparative study on simulation performances of CORSIM and VISSIM for urban street network. *Simulation Modelling Practice and Theory*, 37, 18–29.

Voinescu, M., Udrea, A., and Caramihai, S. (2009). On urban traffic modelling and control. *Journal of Control Engineering and Applied Informatics*, 11(1), 10–18.

Wahle, J., Bazzan, A., Klugl, F., and Schreckenberg, M. (2000). Decision dynamics in a traffic scenario. *Physica A*, 287(3), 669–681.

Wang, W., Wang, B., Zheng, W., Yin, C., and Zhou, T. (2005). Advanced information feedback in intelligent traffic systems. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 72(6 Pt 2), 066702.

Wen, W. (2008). A dynamic and automatic traffic light control expert system for solving the road congestion problem. *Expert Systems with Applications*, 34(4), 2370–2381.

Zegeye, S., Schutter, B.D., Hellendoorn, J., Breunesse, E., and Hegyi, A. (2013). Integrated macroscopic traffic flow, emission, and fuel consumption model for control purposes. *Transportation Research Part C: Emerging Technologies*, 31, 158–171.

Zhiyong, L. (2006). Immunity genetic algorithms based adaptive control method for urban network signal. *Control Theory & Applications*, 23(1), 119–125.