Application of Genetic Algorithms and High-Performance Computing to the Traffic Signal Setting Problem*

Pawel Gora[1] and Przemyslaw W. Pardel[2]

[1] Faculty of Mathematics, Informatics and Mechanics
University of Warsaw
p.gora@mimuw.edu.pl

[2] Interdisciplinary Centre for Computational Modelling
University of Rzeszow
ppardel@ur.edu.pl

Abstract. The paper presents results of our research on application of genetic algorithms to the problem of finding good configurations of traffic signals offsets in a road network (Traffic Signal Setting problem). We tested algorithms on a large road network - realistic map of Warsaw acquired from the OpenStreetMap project. The main research tool was the software Traffic Simulation Framework developed by the first author. To speed up experiments we employed a high-performance computing cluster at the University of Rzeszów - sessions of experiments were supervised by the second author.

Key words: traffic optimization, traffic modelling, traffic simulation, Traffic Signal Setting problem, metaheuristics, high-performance computing

1 Introduction

Large traffic congestion is an important civilizational and commercial problem, especially in urban areas. It causes travel delays, noise, stress of drivers, increase of air pollution, fuel and energy consumption, problems in organizing public transport and detours etc. It is estimated that drivers in 7 largest polish cities lose yearly approximately 3.5 billion PLN because of traffic jams [DEL]. The situation is similar in other countries, e.g., drivers in US lose yearly 5.5 billion hours and 2.9 billion gallons of fuel [PDSG], it is forecasted that the cost of traffic gridlocks will be 293.1 billion dollars by 2030 (almost a 50% increase from 2013) [CEBR]. Thus, it is important to conduct a research on how to improve the traffic flow, especially in urban, densely inhabited areas.

* Experiments were carried out in the Interdisciplinary Centre for Computational Modelling. The research was partially supported by the computational grant nr 1/2015 at the University of Rzeszów and by the European Social Fund from the European Union. Authors would like to express their gratitude to Andrzej Skowron and Jan Bazan for their support during conducting research presented in the paper.

In the research traffic optimization methods based on adaptive traffic signal control are developed, with focus on the Traffic Signal Setting problem [CYP], [GOR3] in which the goal is to find the best possible configuration of offsets for a given traffic situation. For this problem a genetic algorithm (GA) is applied, in which fitness functions take into account average speed, total stopping time and total time of drive with low speed (below 20 km/h). Values of quality functions (fitness functions) of traffic signal settings are computed using microscopic and mesoscopic traffic simulation models implemented in the Traffic Simulation Framework software, developed by P.Gora ([GOR1], [GOR2]). Experiments were conducted on a realistic road network of Warsaw originating from the OpenStreetMap project [OSM]. The main novelty of the research, in comparison to previous results [GOR3], is introducing mesoscopic traffic model, calculating fitness function and modifying traffic signal settings on smaller areas, computing values of fitness functions using high-performance computing cluster.

The rest of the paper is organized as follows: Sect. 2 presents the software Traffic Simulation Framework which was used as a research tool to conduct experiments and calculate fitness function. Section 3 describes a problem of traffic optimization which we tackle, defined as the Traffic Signal Setting problem, outlines our approach to solve it and summarizes results of conducted experiments. Section 4 concludes the paper and outlines plans for the future work and possible improvements.

2 Traffic Simulation Framework

In the research, a special version of the Traffic Simulation Framework software (TSF, in short), was used for the purpose of computing values of fitness functions. TSF is an advanced tool for simulating and investigating vehicular traffic in cities, being developed at the University of Warsaw by P. Gora. It uses realistic maps acquired from the OpenStreetMap project [OSM] and is able to perform realistic, large-scale traffic simulations. The tool currently implements a microscopic traffic simulation model being extension of the Nagel-Scheckenberg model [NS]. Also, a new mesoscopic model (Sect. 2.2) has been recently developed in order to conduct even faster simulations. TSF is developed in C#.NET [NET], it possesses Graphical User Interface presenting maps and enabling editing many simulation parameters, as well as modifying maps (e.g., editing locations and configurations of traffic signals, specifying Origin-Destination (OD) matrix). It can record 5 types of output data: avarage speeds, congestion, instantaneous positions and velocities, locations of traffic jams, structure of the road network. The software was described in more details in the previous works [GOR1], [GOR2], which contain description of the underlying microscopic model, being extension of the Nagel-Schreckenberg model [NS]. It is important to notice that TSF has already found a few scientific applications and is still under development, e.g., recently, thanks to cooperation with Wojciech Chmiel, few new modules were implemented [CHMIEL]:

- module for reading transportation zones and OD matrices constructed using these zones, defined in the format provided by urban roads authorities in Warsaw, ZDM [ZDM];
- module for estimating OD matrix from traffic counts;

– module for calculating routes by taking into account drivers' responses to the traffic situation.

2.1 Microscopic Model

The TSF's microscopic simulation model is an extension of the Nagel-Schreckenberg model [NS] (Na-Sch model) based on a probabilistic cellular automaton, in which a 1-dimensional road is represented as an infinite tape divided into cells (corresponding to the area of a length 7.5 meters), time and space are discrete. At each step each cell may be empty or occupied by a single car. Cars are indistinguishable and move according to rules imitating realistic drive. The model was extensively studied and extended to more sophisticated cases, e.g. 2-dimensional road network [CS], multiple lanes [NWWS].

Na-Sch model was also extended to the case of a realistic road network, represented as a directed graph, implemented in the TSF software [GOR1], [GOR2]. The most important extensions are: many lanes, traffic signals, different profiles of drivers (influencing default maximal speed), different types of roads, speed reduction before crossroads and traffic signals, positions of cars within a single cell. In TSF routes are calculated using extension of the A* algorithm - weights of edges are estimated times of drive, based on geographical distance and default maximal speeds of drive, randomization models possible deviations from expected times.

2.2 Mesoscopic Model

Recently, a mesoscopic model was implemented in TSF, as a second traffic model (beside microscopic model). The model is not time-based, but event-based, which means that it doesn't specify position and speed of cars at each time step, instead, it estimates time of drive (T_{EST}) on road between 2 neighbouring nodes on the path, based on the geographical distance between nodes (D) and default maximal speed on this road (V_{MAX}): $T_{EST} = 1.2 \cdot \frac{D}{V_{MAX}}$. Then, if a car has to wait on a red signal, the total time of drive is increased by a time to the next switch from the red signal state to the green signal state.

This is a very simple mesoscopic model, constructed for the purpose of traffic optimization, to assess quality of traffic signal configurations (e.g. to minimize the total waiting time, because of a red signal). It doesn't take into account a speed reduction caused by large traffic density (in fact, it is not straightforward to estimate traffic density on every road segment at arbitrary moment of time), queueing models and capacity models. These improvements are in plans for the further research.

3 Traffic Optimization

There are many approaches to traffic optimization and reducing impact of large traffic gridlocks, for example: adaptive traffic signal control, adaptive control of speed limits, proposing alternative routes to drivers, building new roads, introducing tolls for driving in the city center, introducing intelligent parking systems. The approach presented in the paper is focused on the first mentioned method: adaptive traffic signal control.

There already exist traffic management systems deployed in many cities, for example ZSZR [ZSZR], UTMS [UTMS], TRISTAR [TRI]. These systems implement some of the mentioned traffic optimization methods. Among adaptive traffic control mechanisms implemented in traffic management sytems the most popular are: SCATS [SCA], SCOOT [SCO], OPAC [OPAC], RHODES [RHO], MARLIN [MAR], MOTION [MOTION]. These systems collect traffic data from detectors installed close to crossroads (usually: inductive loops, cameras) and adaptively change configurations of traffic signals based on measured values of congestion, lengths of queues etc. Some of these systems perform short-term traffic prediction, based on historical and actual data, to propose even better traffic control strategies. The efficiency of such traffic managemenet systems nowadays is already quite good, but the large and even increasing traffic congestion creates demand for designing even better approaches. The rapid progress in computer science, especially in fields of machine learning, artificial intelligence, Big Data processing, and high-performance computing makes it possible to develop new traffic management solutions, potentially more effective than existing strategies.

The existing traffic management systems could be improved, for example, by collecting traffic data (positions, speeds and, potentially, also routes) using GNSS [GNSS] and V2X (V2I - vehicle-to-infrastructure or V2V - vehicle-to-vehicle) communication [V2X] and performing advanced traffic prediction based on data analysis methods (e.g., Bayesian networks [ABG]) or traffic simulations calibrated using real-world data [GOR1], [GOR2]. In response to local (predicted) gridlocks, traffic optimization methods could adapt traffic signal settings globally, or, at least, on much larger area than it is realized in most of existing traffic management systems. Traffic signal control systems on neighbouring crossroads could potentially collaborate as a self-organizing system and exchange information in order to make traffic control even better [MAR].

In the paper, we focus on one traffic optimization method, being the initial step in the process of building an advanced traffic management system of a new generation. Namely, we propose a methodology based on a genetic algorithm supported by high-performance computing cluster (to speed up computations) to find suboptimal configurations of traffic signals (relative to the so called Traffic Signal Setting problem) for a given traffic situation. The method assumes that the traffic management system possesses a complete knowledge about the traffic, i.e., initial positions, speeds and routes of all cars. This assumption is not fulfilled nowadays, but there already exist technical means to make it possible: vehicles may be equipped with devices acquiring precise location from GNSS [GNSS], e.g., Galileo system [GALILEO] supported by EGNOS [EGNOS]. Such devices may be integrated with an onboard computer or a mobile device registering acquired data, calculating parameters of drive, such as instantaneous speed, time of drive etc. In addition, the navigation system may be used to calculate the most optimal route and guide driver through this route. All these data (positions, speeds, routes etc.) could be sent to the traffic management system using dedicated short-range communication (DSRC) technology [DSRC] realizing V2X communication [V2X]. In response, the traffic management system may send to vehicles information about the current traffic state or propose optimal routes for drive. In this research information about real traffic conditions was not taken into account in calculating routes of cars, but this approach in the routing process is also investigated by us and may be potentially used in traffic optimization.

It seems that it is already technically feasible to acquire all data required for implementing methods proposed by our research in the real-world scenario. Nowadays the realization may be still difficult and expensive, but in the future it may be just a standard.

3.1 The Traffic Signal Setting Problem

The Traffic Signal Setting problem (TSS problem) is one of problems related to adaptive traffic optimization of vehicular traffic in cities. It can be mathematically defined as follows:

- Given is a directed graph of a road network with traffic signals located in some vertices. Traffic signals are objects with attributes: duration of a red signal phase (T_R), duration of a green signal phase (T_G), offset (T_S) - values of these attributes may be modified.
- **Traffic Signal Setting** (TSS) is a set of values (T_G, T_R, T_S) for all traffic signals in a road network.
- Given are cars with starting positions in some vertices of the graph road network, static routes, rules of drive on edges.
- Given is a function **F** which calculates quality of a traffic signal setting.
- **Goal:** Find a traffic signal setting for which the quality is optimal.

Usually it is assumed that T_G, T_R, T_S may take only nonnegative integer values, interpreted as seconds (less than 1 second of difference usually doesn't give noticeable difference in the traffic), from a finite set. In the paper we consider a case in which it is assumed that values of T_G and T_R are constant and equal to 58 seconds and 62 seconds, respectively (duration of a red signal phase is a bit longer to ensure safety, there is always a short period in which all signals at the crossroad are in the red phase), so the total cycle length is 120 and the only values, that may be modified, are offsets (these values may be from the set $\{0, 1, 2, 3, \ldots, 119\}$).

To ensure safety it may be assumed that traffic signals located at the same crossroad are synchronized, traffic signals at opposite entries have the same values of all attributes, while values of attributes of traffic signals at transverse entries are set complementary, so if the state of traffic signals in one direction is **green**, then the state of signals in all transverse directions should be **red**. Thus, it is sufficient to consider only 1 traffic signal per crossroad - all other should be configured in a deterministic way to ensure traffic safety. However, even in such case the space of possible solutions is large, of the size 120^n, where n is the number of crossroads with traffic signals. In case of a large city, such as Warsaw (this is our experimental case), n is approximately 800 (we don't have a full description of the real road network, since maps used for experiments contain information about only 291 crossroads with traffic signals). It is impossible to evaluate all configurations.

What may be the quality function F? It depends on what is the goal of traffic optimization. Some possible functions are defined by:

– total waiting time - times spent with a speed $0\frac{km}{h}$, summed up over all cars;
– total time spent with low speed, below $20\frac{km}{h}$;
– average speed of all cars;
– delay - the total wasted time, in comparison to the free flow case;
– total emission of fumes;
– total fuel consumption;
– total length of queues at crossroads;
– total number of stops;

Values of these functions could be calculated by running computer simulations of the model evolution. However, realistic traffic models are often nondeterministic, so computed values in such models may come, in fact, from random variable distributions, which are usually not known explicitly. In such case, the natural choice for values of quality functions are mean values, estimated by averaging outcomes of few computer simulations (Monte Carlo method).

Some of mentioned quality functions are correlated, i.e., if values of one function increase, it usually means that values of another function also increase or decrease, but it is not straightforward to give a precise correlation. Since some quality functions may require more complicated models (fumes emission, fuel consumption) than other quality functions (stopping times, number of stops), it is better to focus on simpler functions, which might give better approximations of real-world values. The presented research has been focusing so far on 3 quality functions: total waiting time, total time spent with speed lower than $20\frac{km}{h}$, average speed of all cars - denoted as T_0, T_{20}, $AvgSpeed$, respectively.

A natural question is whether investigated quality functions can be calculated without running computer simulations? It turns out that traffic simulation models are usually too complicated to calculate quality of a given traffic signal setting explicitly, without running a computer simulation, i.e., it is impossible to give a compact formula for calculating the quality in a constant time, or to design algorithm which will calculate the quality in a shorter time than by running traffic simulation. This property is called *computational irreducibility* and was introduced by Steven Wolfram [WOL]. The property is typical for most nontrivial cellular automata. In fact, even for very simple models, such as Nagel-Schreckenberg model [NS], it is impossible to obtain all properties without running computer simulation (with the exception of $V_{MAX} = 1$ [SHA]). Thus, running simulations is currently the only known option to compute values of mentioned quality functions in realistic traffic models.

On the other hand, quality functions could be approximated by functions which values could be calculated in a simpler way than by running time-consuming computer simulations. If such approximations are *monotonic*, in a sense that traffic signal settings with lower values of quality functions have also lower values of approximations, then computing approximations is sufficient to find optimal traffic signal settings. However, it is very difficult to find monotonic approximations. But even if approximation is not monotonic, it may be still useful. Since the space of possible solutions of the Traffic Signal Setting problem is large, it may be difficult to find the most optimum setting even if a quality function could be computed very fast. However, quality functions are just approximations of corresponding values in the real-world traffic, the optimal value of a quality function and optimal TSS in the given traffic model may not be optimal in case of a real-world traffic. In addition, in case of a real-world traffic and nondeterministic models it may not be proved that the given setting is optimal, it may be even difficult to define what optimality actually means. Instead, it may be sufficient to find settings which are suboptimal, for which values of quality functions may be a bit worse than for (unknown) optimal configuration, but should be good enough in the model, can be accepted in reality and could be found relatively easy. Thus, it is sufficient to have approximation functions of quality functions, which will be computed fast and will guarantee that traffic signal settings with good (suboptimal) values of the approximation function will have also good (suboptimal) values of the corresponding quality function in the model and in the real-world scenario. In case of quality functions computed in the microscopic traffic simulation model, a good approximation may be the relevent function in the mesoscopic model. This is a reason why the mesoscopic traffic model (Sect. 2.2) approximating microscopic model may be considered as potentially good model for running traffic simulations and computing quality functions in case of our research. This is one of important innovations introduced in the research in comparison to the previous results [GOR3].

3.2 A Genetic Algorithm for the TSS Problem

For the Traffic Signal Setting problem we propose a genetic algorithm being extension of the algorithm introduced by [CYP]. The extensions are related to:

– More realistic microscopic traffic model (rules of drive based on the Na-Sch model, crossroads with more than 2 traffic signals, realistic Origin-Destination matrices, routes are known at the moment of start of drive, different profiles of drivers, different types of roads) and mesoscopic model
– Possibility to encode not only offsets, but also duration of a red signal phase and green signal phase;
– Different selection procedure - square root selection (however, other methods are also allowed)
– Many possible fitness functions, calculating fitness function on a selected area (potentially smaller than the whole road network)
– Modifying traffic signal settings on a selected area (potentially smaller than the whole road network)

3.3 Genetic Algorithm Specification

In the proposed algorithm TSS is represented as a genotype - vector of genes, in which each gene (vector position) corresponds to a single traffic signalization (representant of a single crossroad with traffic signals, other traffic signals on that crossroad should be deterministically synchronized to ensure traffic safety (see Sect. 3.1)). The value of a single gene represents traffic signal offset - time (in seconds) from the reference point in time to the next phase switch from the **red** signal state to the **green** signal state. This value should be from the set $\{0, 1, 2, \ldots, T_{CYCLE} - 1\}$, where T_{CYCLE} is a duration of a traffic ligth cycle equal to $T_G + T_R = 120$ (we assume $T_G = 58$, $T_R = 62$, see Sect. 3.1).

The length of a genotype depends on the number of crossroads considered in the experiment. We investigated 2 areas for modifying traffic signal settings and calculating fitness functions:

– A - The whole road network (291 crossroads with traffic signals)
– B - Smaller area corresponding to the *Stara Ochota* district in Warsaw (15 crossroads with traffic signals)

In experiments the initial population consisted of 100 or 400 genotypes, generated randomly, i.e., every gene at each position had a value being random number from the set $S_1 = \{0, 1, 2, \ldots, 119\}$. However, in some experiments sets of possible values were sparse: $S_5 = \{0, 5, 10, 15, \ldots, 115\}$, $S_{10} = \{0, 10, 20, \ldots, 110\}$.

3 different fitness functions were considered: $Time0$ (total stopping time - time with speed $0\frac{km}{h}$), $Time20$ (total time of drive with speed below $20\frac{km}{h}$), $AvgSpeed$ (average speed of all cars). Values of fitness functions were calculated by running traffic simulations in the Traffic Simulation Framework. 2 different simulation models were applied: the original microsopic model and the new mesoscopic model (Sect. 2.2).

As a selection operator, the Square-root operator was chosen: from a given population with n genotypes only $\sqrt{n}$ were selected to take part in the reproduction. The reason for that was to get rid of poor traffic signal settings as soon as possible and enhance TSSs which are better and may lead to better solutions.

As a crossover operator the Uniform Crossover operator was chosen: $\sqrt{n}$ genotypes were crossed with each other to obtain a new set of n genotypes, and for each pair of crossing genotypes and each vector position a gene value was selected from a randomly chosen genotype.

As a mutation operator the Uniform Mutation operator was chosen: for each gene with a given probability (3 values were investigated: $\{\frac{1}{100}, \frac{1}{20}, \frac{1}{5}\}$) the gene value was set to a value selected randomly (with a uniform distribution) from the set of possible values.

3.4 Experiments

We prepared and performed many series of experiments. In the paper, we present only experiments being milestones in the research (there were many more auxiliary experiments conducted, which led to conclusions taken into account in designing next experiments) and focus on the last experiment, which gave the best outcome so far.

1. TSS modified on area A, microscopic stochastic simulation model, fitness functions computed on area A (without HPC cluster) 5 times and averaged
2. TSS modified on area B, microscopic stochastic simulation model, fitness functions computed on area A once (without HPC cluster)
3. TSS modified on area A, microscopic stochastic simulation model, fitness functions computed on area B once (on HPC cluster)
4. TSS modified on area B, microscopic stochastic simulation model, fitness functions computed on area B once (on HPC cluster)

5. TSS modified on area B, microscopic stochastic simulation model, fitness functions computed on area B (on HPC cluster), 5 times and averaged
6. TSS modified on areas A and B, mesoscopic deterministic simulation model, fitness functions computed on areas A and B (on HPC cluster)

In each experiment, we calculated a relative difference between values of the best genotype in the first iteration and the best genotype found in all iterations. First 2 sessions of experiments were conducted on a few machines in the computer laboratory at the University of Warsaw. Because of large computational complexity of traffic simulations and genetic algorithms, we decided to use a high-performance computing cluster at the University of Rzeszów (Sect. 3.5).

Table 1. Simulation parameters used in the 6th session of experiments

Name of the parameter	**Possible values**
Duration of traffic simulation	600 seconds
Initial number of cars in the simulation	10000, 30000, 50000, 70000
Number of cars which start drive at each step	10
Interval between possible values of offsets	1 (Set S_1), 10 (Set S_{10})
Optimized value	Time0
Number of iterations	50
Population size	100, 400
Area on which TSSs are modified	A (the whole road network), B (smaller region, "Stara Ochota" district)
Area on which fitness function is computed	A (the whole road network), B (smaller region, "Stara Ochota" district)
Probability of mutating a single gene	0.01, 0.05, 0.2

The first session of experiments and its results are described in the paper [GOR3]. One of conclusions after the first experiment was that GA brings promising results - 3.11% improvement of the $Time0$ fitness function and 1.82% improvement of the $Time20$ fitness function, after only 9 iterations of the algorithm, but computations should be speed up in order to compute more iterations. This was one of goals of 4 next sessions of experiments in which we investigated different configurations of areas for calculating fitness functions and modifying traffic signal settings. However, improvements were not as good as expected, giving still only up to $5-6\%$ of improvement (usually much less), despite of running much more iterations of algorithms (up to 50 populations). The reason of not satisfactory performance of GA in such models is still investigated (one of potential reasons may be influence of nondeterminism of the model on values of fitness functions). In the meanwhile, we decided to implement a new traffic model, mesoscopic model described in Sect. 2.2.

In the 6-the session we ran in parallel in the cluster 192 experiments with fitness functions computed using a mesoscopic model from Sect. 2.2 and results were significantly better than before. The Table 1 presents values of parameters used in our experiments (in this session we used only a subset of values of parameters used in all session, e.g., the only calculated fitness function is $Time0$, because in the mesoscopic model $AvgSpeed$ should lead to the same TSSs as $Time0$, while $Time20$ cannot be computed). We drew the following conclusions:

1. Calculating values of the fitness function *Time0* in the mesoscopic model takes few seconds which is about 100 times faster than in the microscopic model. It can be even improved by introducing concurrency in computing waiting times of cars (the speed up may depend on number of cars, traffic model, road network topology, architecture of the cluster, number of cores etc.) and by parallelizing computation of a single fitness function (the speed up may depend on number of genotypes in a population, architecture of the cluster, number of cores etc.). We estimate that thanks to such parallelism execution time of the presented genetic algorithm may be reduced from few hours to few minutes on the experimental cluster (Sect. 3.5) and to maximum few seconds on larger HPC clusters, making possible realtime applications.
2. If signals are modified on smaller area (B) and fitness function is calculated on the whole road network (area A), the improvement is small, in the range $0,4657\% - 0,7182\%$. The main reason is that modifying traffic signals on small area has small impact on global traffic parameters.
3. If signals are modified on smaller area (B) and fitness function is computed on smaller area (B), the improvement is significant, in the range $6,8567\% - 15,5888\%$, depending on values of other parameters.
4. If signals are modified on the whole road network (area A) and fitness function is computed on the whole road network (A), the improvement is also significant, in the range $5,7101\% - 18,1204\%$, depending on parameters.
5. If signals are modified on the whole road network (A) and fitness function is computed on area B, the improvement is in the range $26,6716\% - 51,4472\%$ (much better than in case of reconfiguring traffic signals only on the area B). However, it may cause larger delays on the rest of the road network, thus this method alone cannot be applied, but should be further investigated.

3.5 Computational Cluster

Experiments on a high-performance computing cluster "PEGAZ" (produced by Hewlett-Packard) were carried out in the Interdisciplinary Centre for Computer Modelling at the University of Rzeszów from January to June 2015 and were supervised by the second author (Dr. P. Pardel). The cluster consists of 40 computational nodes with 2 processors (INTEL Xeon E5-2620, 6 cores, 2.0 GHz, 15MB cache) per node. The computational power is 7.5 TeraFLOPS, 1TB RAM (1032GB, 258x 4GB DDR3 1333MHz).

4 Conclusions and Future Work

In the paper, new results of applications of genetic algorithms for the Traffic Signal Setting problem were presented. In comparison to previous results [GOR3] the major improvement in computation time came from applying a high-performance computing cluster (Sect. 3.5) and introducing mesoscopic traffic simulation model, instead of microscopic model. The latter change brought also much better results of GA: more than 10% reduction of the total waiting time on the whole road network of Warsaw and on a smaller district.

As for the future research, the mesoscopic model implemented in the TSF software should be extended in order to take into account traffic congestion, queueing model and capacity model. For ensuring realism of results, a detailed calibration and validation of the mesoscopic model should be performed, by comparison to the microscopic model and using real-world traffic data, which have been recently acquired thanks to collaboration with ZDM [ZDM].

References

[ABG] Adamczyk M., Betliński P., Gora P., *"Combined Bayesian Networks and Rough-Granular Approaches for Discovery of Process Models Based on Vehicular Traffic Simulation"*, in *"Communications in Computer and Information Science"*, 2010, Vol. 80, Part 2, pp. 278-287.

[CEBR] Prediction of a future cost of traffic gridlocks, `http://www.cebr.com/reports/the-future-economic-and-environmental-costs-of-gridlock`.

[CHMIEL] Chmiel W., *"Origin-Destination matrix estimation based on real traffic data"*, Master thesis at the University of Warsaw, 2015.

[CS] Chowdhury D., Schadschneider A., *"Self-organization of traffic jams in cities: effects of stochastic dynamics and signal periods"*, in *"Physical Review E (Statistical Physics, Plasmas, Fluids, and Related Interd. Topics)"*, Vol. 59, I. 2, pp. 1311-1314.

[CYP] Chen S.W., Yang C.B., Peng Y.H., *"Algorithms for the Traffic signal Setting Problem on the Graph Model"*, in *"Proc. of the 12th Conference on Artificial Intelligence and Applications"*, TAAI 2007.

[DEL] Report on the cost of traffic jams in 7 largest polish cities `http://korkometr.targeo.pl/Raport_Korki_2014.pdf`.

[DSRC] Kenney J.B., *"Dedicated Short-Range Communications (DSRC) Standards in the United States"*, Proceedings of the IEEE, vol. 99, Issue 7, 2011, pp. 1162-1182.

[EGNOS] EGNOS, `http://www.gsa.europa.eu/egnos/what-egnos`.

[GALILEO] GALILEO, `http://www.esa.int/Our_Activities/Navigation/The_future_-_Galileo/What_is_Galileo`.

[GNSS] Hossam-E-Haider M., Tabassum A., Shihab R. H., Hasan C. M., *"Comparative analysis of GNSS reliability: GPS, GALILEO and combined GPS-GALILEO"*, in *"2013 International Conference on Electrical Information and Communication Technology (EICT)"*, 2014, pp. 1-6.

[GOO] Google Self-Driving Cars project, `http://www.google.com/selfdrivingcar`.

[GOR1] Gora P., *"Traffic Simulation Framework - a Cellular Automaton based tool for simulating and investigating real city traffic"*, in *"Recent Advances in Intelligent Information Systems"*, 2009, pp. 641-653, ISBN: 978-83-60434-59-8.

[GOR2] Gora P., *"Traffic Simulation Framework"*, in *"14th International Conference on Modelling and Simulation"*, 2012, pp. 345-349, ISBN: 978-0-7695-4682-7.

[GOR3] Gora P., *"A genetic algorithm approach to optimization of vehicular traffic in cities by means of configuring traffic signals"*, in *"Emergent Intelligent Technologies in the Industry"*, 2011, pp. 1-10, ISBN: 978-3-642-22731-8.

[MAC] Maciejewski M., *"A comparison of microscopic traffic flow simulation systems for an urban area"*, Transport Problems, VOl. 4, Issue 5, 2010, pp. 27-38.

[MAR] El-Tantawy S., Abdulhai B., Abdelgawad H., *"Multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC): methodology and large-scale application on downtown Toronto"*, *"IEEE Transactions on Intelligent Transportation Systems"*, 2013, Vol. 14, Issue 3, pp. 1140-1150.

[MOTION] SIEMENS MOTION - traffic management system, `http://www.signalsystems.org.vt.edu/documents/July1998AnnualMeeting/mot_usa1308.pdf`.

[NET] C#.NET, `http://msdn.microsoft.com/en-us/library/z1zx9t92.aspx`

[NS] Nagel K., Schreckenberg M., *"A cellular automaton model for freeway traffic"*, in *"Journal de Physique"*, 1992, pp.: 2221-2229.

[NWWS] Nagel K., Wolf D. E., Wagner P., Simon P., *"Two-lane traffic rules for cellular automata: A systematic approach"*, in *"Physical Review E (Statistical Physics, Plasmas, Fluids, and Related Interdis. Topics)"*, Vol. 58, Issue 2, 1998, pp. 1425-1437.

[OPAC] Gartner, N, H., Pooran, F.J., Andrews, C.M., *"Implementation of the OPAC adaptive control strategy in a traffic signal network"*, in *"2001 IEEE Intelligent Transportation Systems, 2001. Proceedings."*, 2001, pp. 195-200.

[OSM] OpenStreetMap project, `http://www.openstreetmap.org`.

[PDSG] B. Pan, U. Demiryurek, C. Shahabi, C. Gupta, *"Forecasting Spatiotemporal Impact of Traffic Incidents on Road Networks"*, ICDM 2013, pp. 587-596.

[RHO] Mirchandani P., Wang F-Y., *"RHODES to Intelligent Transportation Systems"*, in *"Journal IEEE Intelligent Systems"*, Vol. 20, Issue 1, 2005, pp. 10-15.

[SCA] SCATS - traffic management system: `http://www.scats.com.au`.

[SCO] SCOOT - traffic management system: `http://www.scoot-utc.com`.

[SHA] Schadschneider A., *"The Nagel-Schreckenberg model revisited"*, in *"The European Physical Journal B"*, Volume 10, Issue 3, 1999, pp. 573-582.

[TRI] TRISTAR - traffic management system, `http://www.gdansk.pl/biznes,1105.html`.

[UTMS] UTMS - traffic management system, `https://szr.szczecin.pl/utms/index`.

[V2X] Vehicle-to-infrastructure communication, solutions proposed by Kapsch, `https://www.kapsch.net/ca/ktc/its-solutions/V2X-Cooperative-Systems`.

[WOL] Wolfram S., *"A New Kind of Science"*, Wolfram Media, Inc., 2002.

[ZDM] ZDM, The Authority of Urban Roads in Warsaw, `http://www.zdm.waw.pl`.

[ZSZR] Integrated Traffic Management System in Warsaw, `http://zszr.zdm.waw.pl`.