Simulation-Based Heuristic Optimization of a Traffic System

Jaroslav Sklenar
Zdislav Beranek
Pavel Popela
Department of Statistics and Operations Research
University of Malta
Msida MSD 2080, Malta
Web: http://staff.um.edu.mt/jskl1/
Faculty of Mechanical Engineering
Brno University of Technology
Technická 2, Brno, Czech Republic
E-mail: jaroslav.sklenar@um.edu.mt, beraneks@email.cz, popela@fme.vutbr.cz

KEYWORDS

Discrete Simulation, Traffic System, Optimization, Heuristics, Simulated Annealing.

ABSTRACT

This paper deals with nonlinear optimization of a stochastic system via simulation and a heuristic algorithm. These tools are used for optimization of the time parameters of the traffic lights of three junctions at Konečného square in Brno, Czech Republic, in order to reach maximum possible throughput. The objective is to minimize average waiting time in the system that might be described as an open queuing network. This is done in two steps: building a simulation model of Konečného square in Java using SSJ (Stochastic Simulation in Java - a Java library for stochastic simulation) and implementing a heuristic algorithm Simulated Annealing that is using the simulation model for optimization. After a brief description of the traffic system, its mathematical and simulation models are introduced. Then the way of getting input data is discussed as well as verification and validation of the simulation model. The results of an optimization based on Simulated Annealing are shown and interpreted in the last part of the paper. Another contribution of building the simulation model is evaluating the effect of implementing the preference given to public transportation.

TRAFFIC SYSTEM

The traffic system at Konečného square in Brno, Czech Republic consists of three complicated junctions of streets Veveří-Kotlářská, Veveří-Nerudova and Úvoz-Žižkova, controlled by traffic lights, shown in figure 1. To simplify the problem we did not consider other neighbouring junctions because they are far enough. Technical documentation was provided by Brněnské komunikace a.s. (BKOM).

Principles of controlling a signalized intersection: Traffic states are described by using phases. During each phase vehicles pass through the junction in directions that are not collisional. A periodical sequence of the phases given by the phase diagram determines how vehicles pass through the junction. Timing of the phases is determined by a signal plan that controls the traffic lights. The longer the phase lasts, the longer queues of vehicles in other collisional directions are formed. The signal plan must satisfy various constraints given by the geometry of the junction, by the obvious requirement to avoid collisions and by various timing constraint resulting from security rules. An intuitive formulation of the problem can be stated as follows:

Find the phases timing in order to maximize the junction throughput while keeping all constraints satisfied. To measure the throughput by a single scalar value, we use the average waiting time spent by a vehicle in the system.

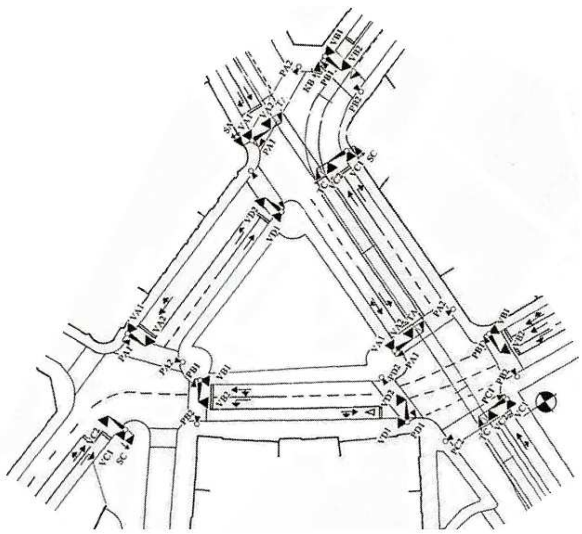

Figure 1. Cartographic Plan of the Konečného Square.

MATHEMATICAL MODEL

As mentioned above, the traffic system can be described as an open queuing network depicted in figure 2. Such systems are obviously stochastic, because some variables are not deterministically fixed. The random k-vector $\boldsymbol{\xi}$ in the objective function $f(\mathbf{x},\boldsymbol{\xi})$ represents all variables whose values cannot be found or computed exactly and are thus characterized by probability distributions. The n-vector $\mathbf{x}$ represents the timing plans of all three junctions in the system. Its size is due to different geometries of the

Proceedings 23rd European Conference on Modelling and Simulation ©ECMS Javier Otamendi, Andrzej Bargiela, José Luis Montes, Luis Miguel Doncel Pedrera (Editors)
ISBN: 978-0-9553018-8-9 / ISBN: 978-0-9553018-9-6 (CD)

junctions equal to n = 11+20+16 = 47 where the three numbers are the numbers of traffic phases in timing plans of the junctions.

If the time intervals between vehicle arrivals and times spent by vehicles in the junctions had exponential distributions, the network could be characterized as an Open Jackson Network and the problem might be solved by standard tools of Queuing Theory. But these distributions are not exponential and therefore as the first approach we substituted them by rather realistic triangular distributions that can easily be changed later. Due to this reason the only feasible method available for evaluation of particular timing plans is simulation.

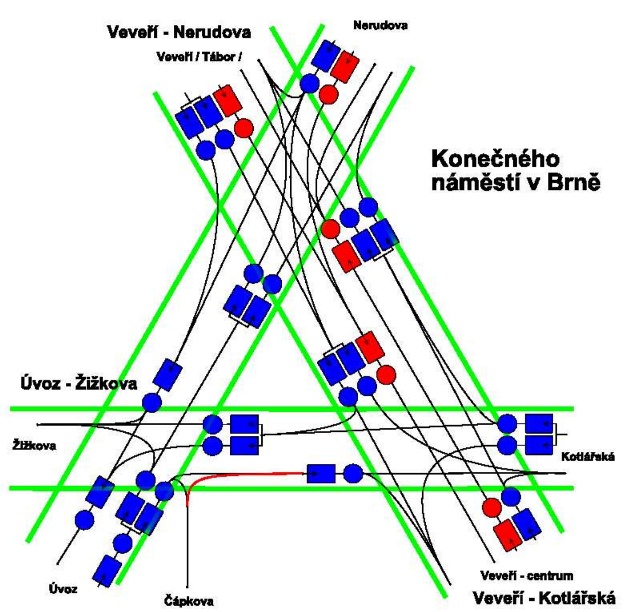

Figure 2. Queuing Network Representing the Konečného Square, (the inner lines are the trams).

Stochastic programming

To evaluate the objective function describing the performance of the traffic system, a simulation model of the Konečného square was built, since an analytical form of the function f cannot be found due to the randomness and also the complexity of the system. So using the Stochastic Programming terminology we solve the following underlying problem

$$\min_{\mathbf{x}} \{ f(\mathbf{x}, \xi) \mid \mathbf{x} \in C = \{ \mathbf{x} \in U \mid \mathbf{g}(\mathbf{x}) \circ \mathbf{0} \} \},$$

where the vector function $\mathbf{g}: \mathbb{R}^n \to \mathbb{R}^m$ represents all constrains on $\mathbf{x} \in U \subset \mathbb{R}^n$ representing time setting of the phases. C is the feasible set, U is the domain space of the decision variables $\mathbf{x}$. There are two categories of the constraints **g**. First there are timing constraints represented by lower and upper bounds on all 47 timing periods and also by lower and upper bounds on the duration of the cycles. Another obvious restriction is the stability of the queueing system. As the effective arrival and service rates depend on **x** in a way that cannot be expressed analytically, this type of feasibility can only be checked by simulation. That's why we define stability by limits on queue lengths L_Q^i in all nodes. These limits are treated differently according to the node type. Exceeding queue length limits in nodes that are entry points into the system represents system instability and it aborts the simulation. That also saves time during the optimization process. Reaching queue length limits in inner nodes represents the temporary saturation of the particular direction. Vehicles are not allowed to enter this queue and are thus forced to stay where they are, but simulation is not aborted. This mechanism also models the obvious limited capacity of road areas intended for waiting vehicles.

To solve the above problem we have to create its deterministic equivalent. We use the so-called Expected Objective (EO) deterministic equivalent where expectation is taken from all times spent by vehicles in the system queues. In particular we define the mean value W_Q^i of waiting times w_{Qj}^i in the queue of the i-th node, $W_Q^i = \frac{1}{n_i} \sum_{j=1}^{n_i} w_{Qj}^i$, where n_i is the total number of cars that went through the node i during the experiment run. These mean waiting times are then weighted by the total numbers of cars in the nodes. This is the objective function formula:

$$f\left(\boldsymbol{\kappa}(\mathbf{x}), \xi\right) = a + qs + \frac{\sum_{i=1}^{N} n_i\left(\boldsymbol{\kappa}(\mathbf{x}), \xi\right) W_Q^i\left(\boldsymbol{\kappa}(\mathbf{x}), \xi\right)}{\sum_{i=1}^{N} n_i\left(\boldsymbol{\kappa}(\mathbf{x}), \xi\right)}$$

where the function $\boldsymbol{\kappa}$ represents the configuration parameters of the simulation model and N is the number of nodes (queues) of the model.

The objective function f thus represents the weighted average of the average waiting times in the queues of the system, which is further penalized for breaking constraints (i.e. if $\mathbf{x} \in U - C$). The weights are defined as relative frequencies of vehicles detected in the queues. s is the penalization for breaking one constraint, q is the number of broken constraints and a is the penalization for aborted simulation that occurs if maximal queues lengths in entry nodes are exceeded for a generated solution $\mathbf{x}$. A big value of a is used to represent infeasibility of the particular solution.

SIMULATION MODEL

The simulation model was written in Java with the use of the SSJ (Stochastic Simulation in Java) tool. SSJ is a Java library for stochastic simulation, developed under the direction of Pierre L'Écuyer, in the Département d'Informatique et de Recherche Opérationnelle (DIRO), at the Université de Montréal. It provides facilities for generating uniform and nonuniform random numbers, computing different measures related to probability distributions, performing goodness-of-fit tests, applying quasi-Monte Carlo methods, collecting (elementary) statistics, and programming discrete-event simulations with both events and processes paradigm. For more detailed description of the tool see for example (L'Ecuyer and Buist 2005) or visit the homepage of the tool at http://www.iro.umontreal.ca/~simardr/ssj/indexe.html.

Abstract model

The abstraction is based on usual continuous time, discrete behaviour paradigm. The state of the system is changed when an event occurs, for example a vehicle arrival. This change takes no time and the state does not change between two adjacent events. As mentioned above, random variables occur in this system and thus the system is also stochastic. Vehicles movement is modelled by Java objects of the type *Event*.

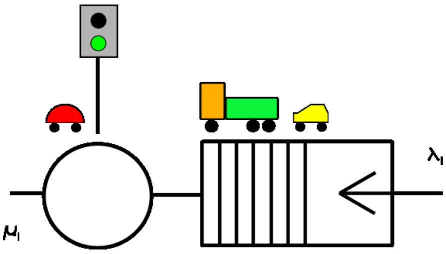

Figure 3. Node in the Traffic Network

After generating an arrival interval with mean $1/\lambda_i$, a new car is generated at system input and passed into the particular queue. If there is a green light on and the queue length is 1, the car is taken out from the queue, the waiting time is saved and queue length is updated. The car is then passed to the server, representing queue emptying, where it waits an average time $1/\mu_i$. Then it is removed and sent to the next node or out of the system. In case that the car reaches a branching node, the probability of selecting the next direction is given by statistical measurements from cartograms provided by BKOM. Parameters of triangular distributions that define the rates λ_i and μ_i for each node are based on statistical measurements performed on the real system by Zdislav Beranek.

Control of the traffic lights is modelled by SSJ objects of the type *SimProcess*. The control can run in two different modes with or without preference given to the public transport. In case of public transport preference mode there is a special phase turned on if a public transport vehicle is coming. The phase allows a requesting vehicle to pass through the junction without a too long waiting.

The more important model simplifications are these:

- Simplification of some traffic priority rules.
- No vehicles with higher priority (e.g. ambulance) are considered in the model.
- No accidental breaking of traffic rules.
- Missing traffic lights states "ready" – orange because they overlap with green and red phases. Orange before green is considered as red, orange before red is considered as green.
- Triangular distribution for generating time intervals between vehicles arrivals.
- Triangular distribution for generating time intervals that simulate passing of vehicles through the junctions.

Verification and validation of the model

The verification of the simulation model was done by comparing results with visualization software VISSIM provided by BKOM, which were obtained in two steps:

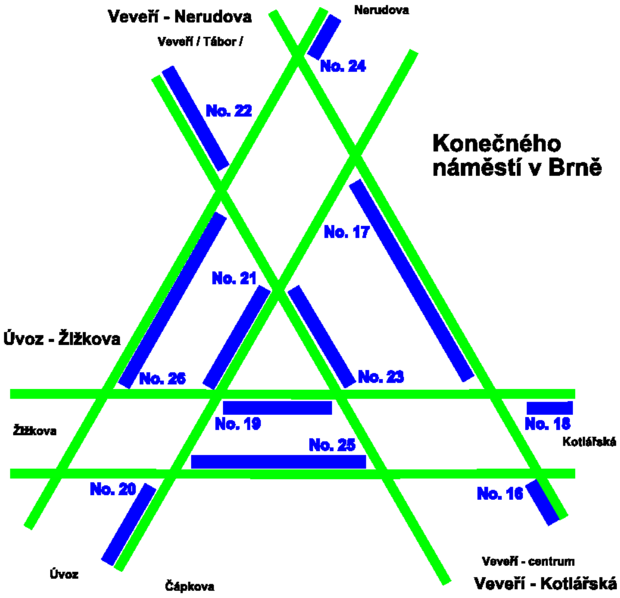

Figure 4. Measured Segments of Konečného Square

At the segments depicted in the figure 4, the maximal queues lengths and the average waiting times in these queues were measured. These waiting times from VISSIM were then compared with the ones obtained from the simulation model.

The simulation time was 72000s (20h), statistics collection started after the first 300s warming period. The used mode was without public transport preference. To compare the results see the tables 1, 2 and 3.

A significant difference occurred at segment 17, data of streets Úvoz VC, VD, VE in the table 3, are missing, because of VISSIM incomplete implementation.

With respect to effort of engineers of BKOM and that the VISSIM is working with real time, there is no such comparison for a mode implementing public transport preference. In this case we accepted an opinion of a traffic specialist from BKOM, who considers the model implementation to be correct.

To validate the model with public transport preference a new statistical measurement of the junctions would have to be performed for comparing the results. Due to high cost this was not acceptable.

Table 1. Average Waiting Times at Junction Veveří-Kotlářská

Segment	Street	Simulation model [s]	VISSIM [s]
23	Veveří VA	2.8	5.4
18	Kotlářská VB	30.6	45.2
16	Veveří VC	28.8	34.4
25	Úvoz VD	21.6	18.2

Table 2. Average Waiting Times at Junction Veveří-Nerudova

Segment	Street	Simulation model [s]	VISSIM [s]
23	Veveří VA	28.8	30.3
24	Nerudova VB	40.0	54.7
17	Veveří VC	38.1	6.9
21	Úvoz VD	21.6	18.2

Table 3. Average Waiting Times at Junction Úvoz-Žižkova

Segment	Street	Simulation model [s]	VISSIM [s]
26	Nerudova VA	10.4	9.5
19	Kotlářská VB	8.8	10.5
20	Úvoz VC	39.4	
	Úvoz VD	2.4	
	Úvoz VE	4.0	

OPTIMIZATION

As we have just numerical approximation of the objective function *f* given by the simulation model (together with checking the feasibility), we are limited in selection of the optimization algorithm. First we believe that all algorithms based on differentiability (like for example gradient based methods) are not appropriate because of two reasons. First numerical computation of gradients needs extra function evaluations that means additional simulation runs. Second, close to optima norms of gradients tend to become small, possibly much smaller than errors involved in approximation of the objective function by simulation. So we are limited to use either exact or heuristic algorithms based on function evaluation only. From these algorithms we have chosen the heuristic optimization algorithm Simulated Annealing (SA).

Simulated Annealing Algorithm

```
1.  set(c, x_current);
2.  x_current → x_best;
3.  do{
4.    update(c);
5.    do{
6.      get(x_new);
7.      if(f(x_new) < f(x_best)){
8.        x_new → x_best;
9.      }
10.     if(f(x_new) ≤ f(x_current)){
11.       x_new → x_current;
12.     }
13.     else if
        (random(0,1) ≤ exp(−(f(x_new) − f(x_current))/c)){
14.       x_new → x_current;
15.     }
16.   } while(we are not enough close to equilibrium)
17. } while(system is not frozen)
```

Decreasing control parameter c is referred to as the annealing temperature. Frozen system represents a found heuristic solution $(c \approx 0)$. The so-called Metropolis criterion (line 13) allows the algorithm to get out of a local minimum. Thermal equilibrium, for a given *c*, is determined by the probability of being in a configuration $\mathbf{x}_{i+1}$ that is characterised by the so-called energetic state E. The probability is given by Boltzmann distribution:

$$\mathbb{P}\left[E = f\left(\mathbf{x}_{i+1}\right) - f\left(\mathbf{x}_i\right)\right] = \exp\left(-\frac{f\left(\mathbf{x}_{i+1}\right) - f\left(\mathbf{x}_i\right)}{c}\right)$$

Heuristic algorithms give good solutions, they cannot guarantee reaching global optimum. However a proof of convergence of SA exists, see (Laarhoven 1989). The performance of the implemented algorithm depends on its setting, generally called cooling schedule.

Cooling schedule

- Setting of the initial value of c is done in such a way that virtually all values of $\mathbf{x}$ could be accepted. There is an acceptance ratio χ_a defined as the number of accepted $\mathbf{x}$ to all tested $\mathbf{x}$. If $\chi_a < \chi_0$, the pre-initial temperature $\tilde{c}$ is raised to $\tilde{c}_{i+1} = 2\tilde{c}_i$. Final temperature $\tilde{c}$ equals to c initial. For optimization we used $\chi_0 = 0.8$, $\tilde{c}_0 = 50$.
- The final value of c is used to stop SA if no better solution was found for a longer time. In a similar way as above there is a rejection ratio χ_r defined as the number of rejected $\mathbf{x}$ to all tested $\mathbf{x}$. If $\chi_r \geq \chi_f$, the algorithm stops. We used $\chi_f = 0.7$.
- To approximate the equilibrium for a given c_k a minimal number of perturbations ($\mathbf{x}_i \to \mathbf{x}_{i+1}$) must be accepted. But for $c_k \to 0$ this number goes to infinity. Therefore the number must be also limited by a maximal value. For optimization we used min = 5 and max = 14.
- It is recommended to decrease the control parameter c as slowly as possible. In this case it is exponential $c_{k+1} = 0.9c_k$.

Two configurations $\mathbf{x}_i$, $\mathbf{x}_j$ are statistically compared via Mann-Whitney test. A requested number of observations $\hat{f}$ of the objective function f for each of the configurations is provided via simulation and then these sets are tested. For optimization the confidence level was 0.05 and the number of observations was 7.

Optimization results

The system is optimized only in the mode without the public transport preference, because otherwise the system is not stable (see figures 5, 6) for original configuration designed by BKOM. Penalization constants in the objective function formula were $s=100$ and $a=400$. Based on the testing of stability and variability, the time horizon 172,800s of a simulation and statistics collection restart in 600s were determined. The duration of the optimization was about 7 h (real time).
Graphs in figures 5 an 6 display objective function values for various time horizon lengths.

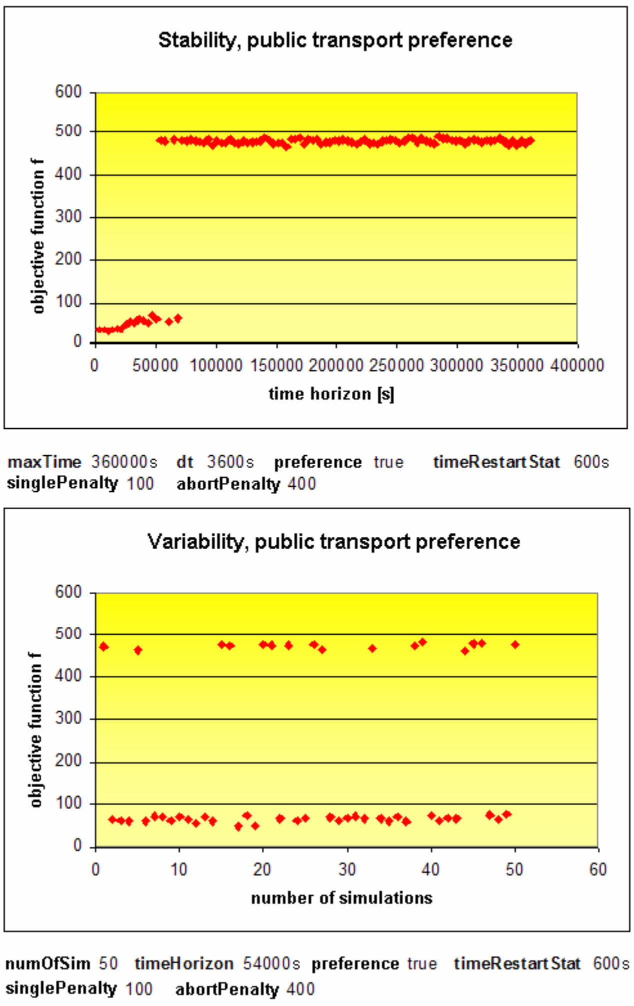

Figure 5. Stability and Variability of Simulation in Mode with Public Transport Preference.

Figure 5 is for public transport preference, in the figure 6 there are results for simulations without public transport preference. The configurations $\mathbf{x}$ are the same, designed by BKOM. The vertical axis is the objective function value, the horizontal axis is the simulation duration in seconds and the number of the simulation experiments respectively. The figure 5 shows that the mode with public transport preference is unstable because there is not enough time for other than public transport vehicles. This result corresponds with reality. The junctions are currently set to no preference to public transport mode.
Tables 4, 5, 6 contain the optimization results for the mode without public transport preference. Only the lengths of main phases (written in black frames in tables 4, 5, 6) were optimized. Other phases arrange a safe switch between the main phases. They are determined mainly by the constraints, so there is no sense in changing them. The tables 4, 5, 6 show current implementation of signal plans for which the average value of the objective function equals to 19.392 and the optimized signal plan configuration for which the average value of the objective function is 15.595.

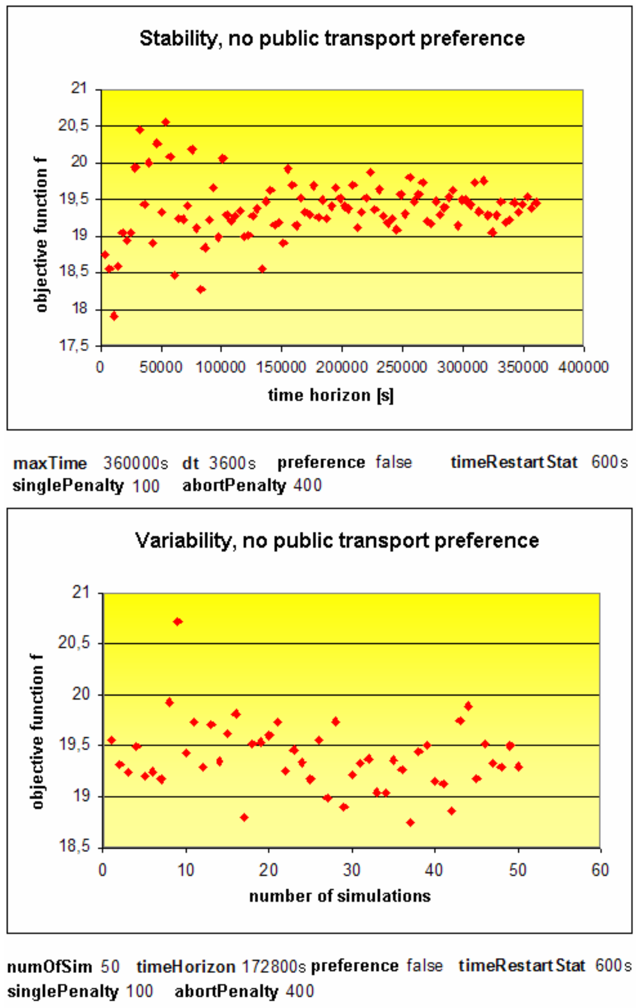

Figure 6. Stability and Variability of Simulation in Mode without Public Transport Preference.

Table 4. Optimization Results

Traffic phases timing at Veveří-Kotlářská				
	going directions	stopped directions	current	otpimized
phases	VB, VD, PA, PC	VA, VC, TA, TC, PB, PD	8	18
			7	7
			1	1
			5	5
			1	1
	VA, VC, TA, TC, PB, PD	VB, VD, PA, PC	37	37
			3	3
	VA, VC, TA, TC	VB, VD, PA, PB, PC, PD	9	4
			5	5
			1	1
	VB, VD, PA, PC	VA, VC, TA, TC, PB, PD	23	19

Table 5. Optimization Results

Traffic phases timing at Veveří-Nerudova				
	going directions	stopped directions	current	optimized
phases	VD, KD, SA, TBS, TBP, TCL	VA, VB, VC, SC, TA, TBL, TCS, TCP, PA, PB, PD	2	3
			2	2
			2	2
			3	3
			1	1
	VB, SC, TBL, TBS, TBP, TCL, TCP, PA	VA, VC, VD, KD, SA, TA, TCS, PB, PD	7	10
			8	8
			1	1
			5	5
			1	1
			1	1
			8	8
			5	5
			1	1
	VA, VC, TA, TBS, TBP, TCL, TCS, PB, PD	VB, VD, KD, SA, SC, TBL, TCP, PA	15	20
			1	1
			4	4
			1	1
			4	4
	VD, KD, SA, TBS, TBP, TCL	VA, VB, VC, SC, TA, TBL, TCS, TCP, PA, PB, PD	29	21

Table 6. Optimization Results

Traffic phases timing at Úvoz-Žižkova				
	going directions	stopped directions	current	optimized
phases	VB, PA, VD, VE	VA, VC, SC, PB, PE, PD	18	21
			5	5
			3	3
			2	2
			1	1
			2	2
	VA, PB, PE, PD	VB. VC, SC, PA, VD, VE	7	1
			10	10
	VA, PB, PE, VD, VE,	VB, VC, SC, PA, PD	6	2
			6	6
	VA, VC, PE, VD, VE	VB, SC, PA, PB, PD	27	37
			2	2
			5	5
			1	1
	VB, SC, PA, VD, VE	VA, VC, PB, PE, PD	2	3
	VB, PA, VD, VE	VA, VC, SC, PB, PE, PD	3	0

Most of directions from tables 4, 5 and 6 can be found in the figure 1. Directions starting with the letter P mark traffic lights for pedestrian crossings, Directions starting with S or K mark additional traffic directions which help to decrease the queues. The third character in TBP, TAL, TCS, etc. clarifies tram directions: right (P), left (L), and middle (S) respectively.

Tables 7, 8 and 9 compare the average queue length, the maximal number of vehicles in the queue and the average waiting time in all queues.

Table 7. Junction Veveří-Kotlářská

	Current configuration			Optimized configuration		
street	$\bar{L}_Q$	$\max L_Q$	$\bar{W}_Q$ [s]	$\bar{L}_Q$	$\max L_Q$	$\bar{W}_Q$ [s]
Veveří VA	0.47	6.00	2.81	0.77	8.00	4.56
Kotlářská VB	6.58	33.00	32.09	4.89	19.00	23.81
Veveří VC	3.79	13.00	29.01	4.02	13.00	30.69
Úvoz VD	2.86	18.00	21.14	3.71	20.00	27.31
Veveří TA	0.02	2.00	3.59	0.06	2.00	8.71
Veveří TC	0.08	1.00	11.95	0.10	1.00	15.19

Table 8. Junction Veveří-Nerudova

	Current configuration			Optimized configuration		
street	$\bar{L}_Q$	$\max L_Q$	$\bar{W}_Q$ [s]	$\bar{L}_Q$	$\max L_Q$	$\bar{W}_Q$ [s]
Veveří VA	5.89	23.00	27.62	4.50	16.00	21.09
Nerudova VB	1.96	6.00	40.03	1.80	6.00	36.98
Veveří VC	5.24	25.00	38.36	2.41	15.00	17.83
Úvoz VD	0.66	8.00	4.59	2.87	20.00	20.01
Veveří TA	0.07	1.00	21.76	0.06	1.00	18.96
Veveří TB	0.12	1.00	36.70	0.11	1.00	33.84
Veveří TC	0.09	1.00	13.38	0.07	1.00	10.33

Table 9. Junction Úvoz-Žižkova

	Current configuration			Optimized configuration		
street	$\bar{L}_Q$	$\max L_Q$	$\bar{W}_Q$ [s]	$\bar{L}_Q$	$\max L_Q$	$\bar{W}_Q$ [s]
Nerudova VA	0.93	9.00	10.50	0.96	9.00	10.93
Kotlářská VB	1.22	13.00	8.94	1.02	11.00	7.41
Úvoz VC	11.35	46.00	39.17	6.37	21.00	21.99
Úvoz VD	0.70	9.00	2.43	0.37	7.00	1.27
Úvoz VE	0.82	9.00	3.99	0.44	7.00	2.12

We can see that most of the observed values decreased in the optimized configuration. Nevertheless the traffic specialists from BKOM have to check the feasibility of the proposed optimized configuration before its possible implementation.

SUMMARY

A traffic system can be very complex and random. The only feasible way of obtaining quantitative information about modifications of such a system is simulation. The simulation model presented in this paper was built in Java with use of the SSJ simulation tool. The user of this simulation model may set a lot of data through a GUI interface and therefore it is possible to compare various settings in a user-friendly way.

It is also an example of the application of simulation methods in operation research. Simulation was used for evaluation of the objective function (weighted average of the average waiting times in the queues of the system) because analytical methods cannot be applied due to the complex and random nature of the system. The objective function was optimized by Simulated Annealing. The optimization of traffic lights configuration was performed and results are shown at the end of the paper. A considerable improvement over the current timing plan was found.

ACKNOWLEDGEMENT

This research has been supported by the Czech Ministry of Education in the frame of MSM 0021630529 Research Intention Intelligent Systems in Automation.
The valuable suggestions and support of Brněnské komunikace a.s. are gratefully acknowledged.

REFERENCES

Beránek, Z. 2007. *Heuristické optimalizační algoritmy v simulačních modelech*. (in Czech) Brno University of Technology, (M.Ing. dissertation)

Gosavi, A. 2003. *Simulation-Based Optimization*. Kluwer Academic Publishers.

L'Ecuyer, P. and Buist, E. 2005. "Simulation in Java with SSJ“. In *Proceedings of the 2005 Winter Simulation Conference* (Orlando, FL, Dec.4-7), 611-620.

Van Laarhoven, P.J.M and Aarts, E.H.L. 1989. *Simulated Annealing: Theory and Applications*. Kluwer Academic Publishers.