Slovak University of Technology in Bratislava
Institute of Information Engineering, Automation, and Mathematics

PROCEEDINGS

of the 18$^{\mathrm{th}}$ International Conference on Process Control

Hotel Titris, Tatranská Lomnica, Slovakia, June 14 – 17, 2011

ISBN 978-80-227-3517-9

http://www.kirp.chtf.stuba.sk/pc11

Editors: M. Fikar and M. Kvasnica

Makýš, M., Kozák, Š.: Hybrid Methods for Traffic Lights Control, Editors: Fikar, M., Kvasnica, M., In *Proceedings of the 18th International Conference on Process Control*, Tatranská Lomnica, Slovakia, 64–71, 2011.

Full paper online: http://www.kirp.chtf.stuba.sk/pc11/data/abstracts/036.html

Hybrid methods for traffic lights control

Miroslav Makys*. Stefan Kozak**

** Slovak University of Technology, Faculty of Informatics and Information Technologies, Ilkovicova 3, 84216 Bratislava, Slovakia (e-mail: makys@fiit.stuba.sk)*
*** Slovak University of Technology, Faculty of Electrical Engineering and Information Technology, Ilkovicova 3, 84216 Bratislava, Slovakia (e-mail: stefan.kozak@ stuba.sk)*

Abstract: The paper deals with modeling and design of hybrid control using HYSDEL modeling language and MPT toolbox for MATLAB, respectively. We created optimal structure model using hybrid systems theory and designed predictive control of traffic lights of three interconnected intersections. Considered intersections are in detail described in proposed article. A model has been built describing the evolution of the queue based on number of incoming and outgoing cars and traffic lights. A realistic intersections model has been built and achieving optimal traffic lights control was successfully verified on many model variations.

1. INTRODUCTION

During last tenths of year traffic in towns has become serious problem and traffic jam everyday occurrence. As traffic lights are tool to control traffic we need to control the lights optimally. Many authors dealt this problem and various approaches were examined. For example in (Zhang, 2008) traffic light control system was considered to be hybrid system and hybrid Petri nets were used to examine this problem. Timed coloured Petri nets were used in (Huang, 2010) and in (Schutter, 1998) the problem was solved as minimization of special performance function.

In this paper we created model of intersection as a hybrid system and use hybrid systems control theory to control traffic lights. The main aim is to design effective and fast control of traffic lights of three interconnected intersections in optimal way.

The paper has the following structure. In the second section examined system structure is described and simplified yet precise model is derived. The section three deals with control problem definition and obtained results and finally in section four is conclusion of this paper.

2. DESCRIPTION AND MODEL OF INTERSECTIONS

2.1 Description of intersections

In this study, we work with three interconnected intersections. It means that cars outgoing from one of arms of intersection 1 are coming to arms of intersections 2 and 3 and vice versa. Scheme of this system is depicted in Figure 1. First intersection depicted in Figure. 2. consists of 4 two-way arms labelled *A, B, C, D* second intersection depicted in Figure 3. consists of 3 two-way arms labelled *E, F, G* and third intersection depicted in Figure 4. consists of 3 two-way arm labelled *H, I, J*.

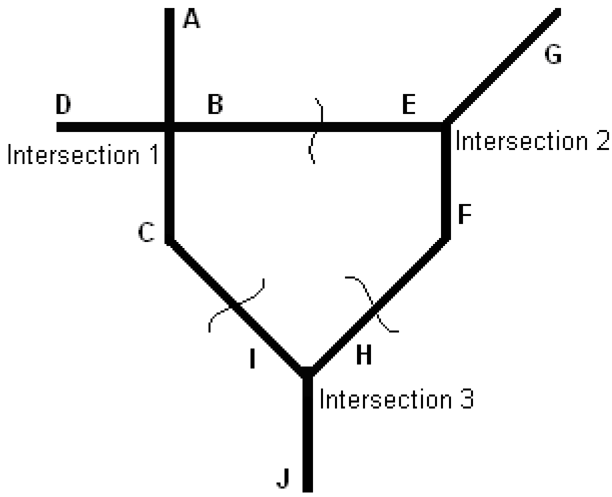

Fig. 1. System of interconnected intersections

Now we are going to describe first intersection. Streams of cars entering the intersection are labelled *A1, B1, B2, C1, C2, D1, D2* and controlled by traffic lights *SA1, SB1, SB2, SC1, SC2, SD1, SD2*, respectively. Each of traffic lights has 3 phases: green, amber and red.

Intersection is depicted in Figure 2. from which it is possible to identify which directions are allowed for cars entering the intersection. Cars in *A1, C1* and *D1* streams can only drive straight, cars in *B2, C2* and *D2* streams can drive to the right and from B1 stream straight and left.

The amount of cars coming into the intersection for each of the directions is denoted as λ_i, where $i \in \{A1, B1, B2, C1, C2, D1, D2\}$. Function λ_i is formed by a series of Dirac pulses, and for each of the stream is different. One Dirac pulse represents coming of one car. When the traffic light is green or amber, amount of outgoing cars for each of the streams is denoted as μ_i or κ_i, respectively. Functions μi and κi are also formed by a series of Dirac pulses.

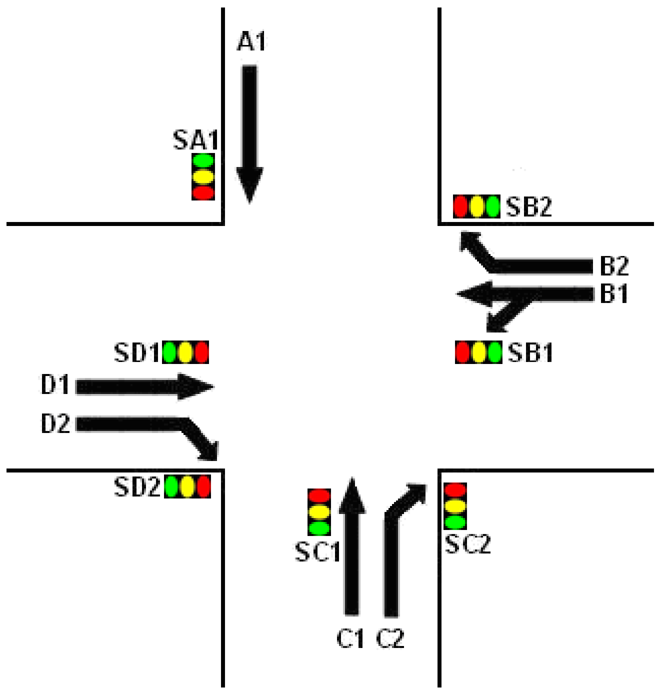

Fig. 2. Scheme of intersection 1

Intersection 2 which is depicted in Figure 3. can be described accordingly to intersection 1. Streams of intersection 2 are labelled *E1*, *E2*, *F1*, *F2*, *G1* and *G2* and are controlled by traffic lights *SE1*, *SE2*, *SF1*, *SF2*, *SG1* and *SG2* respectively.

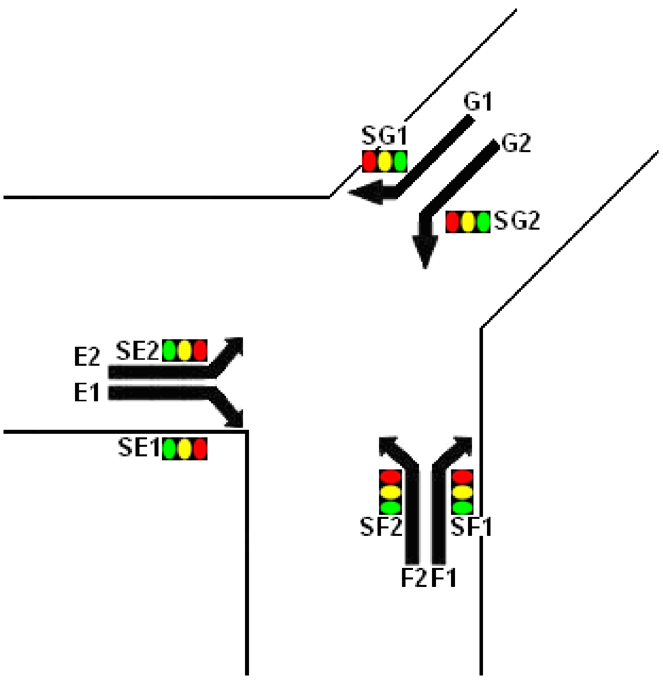

Fig. 3. Scheme of intersection 2

Intersection 3 is similar to intersection 2. Streams of intersection 3 are labelled *H1*, *H2*, *I1*, *I2*, *J1* ans *J2* and are controlled by traffic lights *SH1*, *SH2*, *SI1*, *SI2*, *SJ1* ans *SJ2* respectively.

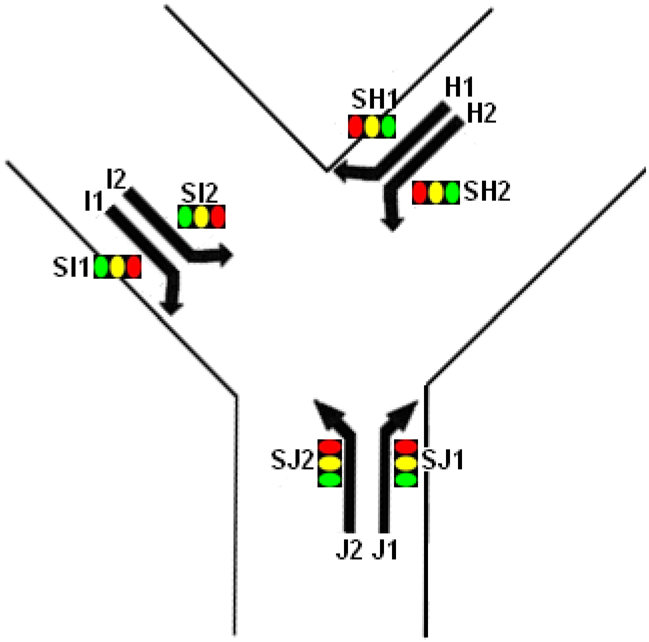

Fig. 4. Scheme of intersection 3

Cars outgoing from arm *B* of intersection 1 are coming to arm *E* of intersection 2, cars outgoing from arm *C* or intersection 1 are coming to arm *I* of intersection 3, cars outgoing from arm *F* of intersection 1 are coming to arm *H* of intersection 3 and wice versa. Time needed car to come from one intersection to another is 15 seconds.

2.2 Model of intersections

Proposal of intersection model was based on ideas in (Schutter, 1998). We again describe just model of intersection 1, models is of intersection 2 and intersection 3 were done accordingly. Let us denote length of waiting car queues as L_i, where $i \in \{A1, B1, B2, C1, C2, D1, D2\}$. When traffic light *Si* is red cars are just coming to intersection when it is green or amber cars are coming and outgoing. Difference of queue length is determined by equation (1):

$$\frac{dL_i}{dt} = \begin{cases} \lambda_i(t) & \text{if Si} = \text{"red"} \\ \lambda_i(t) - \mu_i(t) & \text{if Si} = \text{"}\mathit{green}\text{"} \\ \lambda_i(t) - \kappa_i(t) & \text{if Si} = \text{"amber"} \end{cases} \qquad (1)$$

where $i \in \{A1, B1, B2, C1, C2, D1, D2\}$.

Stream 1	Stream 2
A1	D1
A1	D2
A1	B1
D1	C1
D1	C2
D1	B1
D2	B1
C1	B1

Table 1. List of stream pairs which can not enter intersection 1 at the same time

For a realistic model of intersection it is important to define a constraint to the queue length: $Li \geq 0$. Due to avoid collisions in the intersection it is necessary to impose restrictions on the concurrent green color for determined pairs of traffic lights. These were determined on the basis of intersection specification. The list is in Table 1. Since the B2 stream is not in conflict with any other stream, it is not necessary to control it and we miss it.

Similar restrictions related to intersection 2 are listed in Table 2, restrictions for intersection 3 are listed in Table 3.

Stream 1	Stream 2
E1	G2
E2	F1
E2	F2
E2	G2
F2	G1
F2	G2

Table 2. List of stream pairs which can not enter intersection 2 at the same time

Stream 1	Stream 2
H1	J2
H2	I1
H2	I2
H2	J2
I2	J1
I2	J2

Table 3. List of stream pairs which can not enter intersection 3 at the same time

2.3 Simplified intersection model

Previous intersection model is too complicated for mathematical analysis, therefore, in this section we simplify the model so that it is easier to work with it while still precise enough. It includes following changes:

- lengths of queues are continuous variables,
- comings and outgoings of cars from intersection are represented by constant function,
- amber phase is missing.

Working with intersection 1 let us denote amount of incoming cars as π_i and amount of outgoing cars as τ_i for each stream where $i \in \{A1, B1, B2, C1, C2, D1, D2\}$.

Difference of queue length is determined by equation (2):

$$\frac{dL_i}{dt} = \begin{cases} \pi_i(t) & \text{if Si} = \text{"}red\text{"} \\ \pi_i(t) - \tau_i(t) & \text{if Si} = \text{"}green\text{"} \end{cases} \qquad (2)$$

Intersection model is thus simplified, so that when traffic light is green, cars are coming and outgoing from intersection in constant rate when traffic lights is red cars are just outgoing in constant rate. Amber phase is omitted. The model was created by HYSDEL modelling language described in (Bemporad, 2004), (Bemporad, 2007) and (Bemporad, 2009) which was designed for modelling of hybrid systems. HYSDEL model is than translated into MLD model using HYSDEL compiler which is one of most used hybrid models. Disadvantage of hybrid models is rapid complexity increase in dependance on number of binary variables which from control point of view results to very difficult and time consuming optimisation problem. Next factor which increases model complexity are constraints on state, input and output variables. That is why we try to create intersection model as simple as possible – with minimum number of binary variables and constraints.

Since the generated model is designed to simulate changes in the length of streams of cars on each of intersection arms, state of system is defined as the number of cars waiting in individual streams thus it is vector of length 6. The basic idea is simple: a queue of cars waiting before the intersection is increasing when the light is red and decreasing when the light is green.

Since the intersection is controlled by 6 traffic lights, intersection model will have 6 input binary control signals one for each of traffic lights. Traffic light can be green if it does not violate the restrictions in Table 1. For example, if *SC1* light is green also *SC2* and *SD2* lights or *SC2* and *SA1* lights can be green. *SB1* and *SD1* lights have to be red. We need not therefore to subject to each of the traffic lights in particular. With this feature it is possible to reduce the number of control signals, thus simplifying the intersection model and hence the problem of control. Our aim is to determine the minimum number of control signals. Task is therefore to determine minimum normal disjunctive form (MNDF) on the basis of Karnaugh map. Table 4 shows Karnaugh map for the intersection 1.

MNDF for given map is:

(!A1 & !B1 & !D1) + (!B1 & !D1 & !D2) + (!A1 & !C1 & !D1 & !D2) + (!A1 & !B1 & !C1 & !C2)

where ! denotes operator of negation.

		D2				D2				D2				D2		
			D1								D1					
					C2											
	C1															
A1	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1
B2	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1
B1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
B2	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
	1	1	0	0	0	0	1	1	1	1	0	0	1	1	1	1
	1	1	0	0	0	0	1	1	1	1	0	0	1	1	1	1

Table 4. Karnaugh map for intersection 1

It flows from MNDF that to control the intersection we need 4 signals which control the traffic lights following way:

S1 = !A1 & !B1 & !D1 – green light for *B2, C1, C2, D2* streams

S2 = !B1 & !D1 & !D2 – green light for *A1, B2, C1, C2* streams

S3 = !A1 & !C1 & !D1 & !D2 – green light for *B1, B2, C2* streams

S4 = !A1 & !B1 & !C1 & !C2 – green light for *B2, D1, D2* streams

Instead of restrictions listed in the Table 1. we get a new restriction: at most one of the signals can be set to TRUE. Because of maximum intersection throughput we can modify this restriction so that it is: just one of the signals must be set to TRUE.

Let us define vector *X* as system state and vector *U* as system input:

$$\boldsymbol{x} = \begin{bmatrix} L_{A1} \\ L_{B1} \\ L_{C1} \\ L_{C2} \\ L_{D1} \\ L_{D2} \end{bmatrix}, \ \boldsymbol{u} = \begin{bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{bmatrix} \tag{3}$$

Model of system is then defined by (4):

$$x(k+1) = Ax(k) + Bu(k) + f \tag{4}$$

where:

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \ \boldsymbol{f} = \begin{bmatrix} \pi_{A1} \\ \pi_{B1} \\ \pi_{C1} \\ \pi_{C2} \\ \pi_{D1} \\ \pi_{D2} \end{bmatrix}$$

$$\boldsymbol{B} = \begin{bmatrix} -\tau_{A1} & 0 & 0 & 0 \\ 0 & 0 & -\tau_{B1} & 0 \\ -\tau_{C1} & -\tau_{C1} & 0 & 0 \\ -\tau_{C2} & -\tau_{C2} & -\tau_{C2} & 0 \\ 0 & 0 & 0 & -\tau_{D1} \\ -\tau_{D2} & 0 & 0 & -\tau_{D2} \end{bmatrix}$$

Table 5 shows parameters used to create intersection 1 model. Row *Inc.* contains number of cars incoming to assigned stream per 1 second. Row *Out.* contains number of cars outgoing from assigned stream if traffic lights is green.

	A1	B1	B2	C1	C2	D1	D2
Inc.	0.4	0.4	0.4	0.4	0.4	0.4	0.4
Out.	1.8	2	1.2	2.2	0.8	1.8	1.4

Table 5. List of intersection 1 parameters

Using similar methodology we get following results for intersection 2. Based of restrictions in Table 2, we create Karanugh map shown in Table 6.

MNDF of this map is:

(!E2 & !G1 & !G2) | (!E1 & !E2 & !F2) | (!F1 & !F2 & !G2) | (!E2 & !F2 & !G2)

Based on MNDF we have 4 control signals:

S5 = !E2 & !G1 & !G2 – green light for *E1, F1, F2*

S6 = !E1 & !E2 & !F2 – green light for *F1, G1, G2*

S7 = !F1 & !F2 & !G2 – green light for *E1, E2, G1*

S8 = !E2 & !F2 & !G2 – green light for *E1, F1, G1*

And we get similar intersection model.

		F1						
			E2					
	E1							
F2	1	1	0	0	0	0	1	1
G2	0	0	0	0	0	0	0	0
G1	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	1	1	0	1	1	0	1	1
	0	0	0	0	0	0	1	1
	0	0	0	0	0	0	1	1
	1	1	0	1	1	0	1	1

Table 6. Karnaugh map for intersection 2

Table 7 shows parameters used to create intersection 2 model. Row *Inc.* contains number of cars incoming to assigned stream per 1 second. Row *Out.* contains number of cars outgoing from assigned stream if traffic lights is green.

	E1	E2	F1	F2	G1	G2
Inc.	0.4	0.6	0.4	0.2	0.4	0.4
Out.	1	2	1.2	0.6	1.2	1.8

Table 7. List of intersection 2 parameters

		I1						
			H2					
	H1							
I2	1	1	0	0	0	0	1	1
J2	0	0	0	0	0	0	0	0
J1	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	1	1	0	1	1	0	1	1
	0	0	0	0	0	0	1	1
	0	0	0	0	0	0	1	1
	1	1	0	1	1	0	1	1

Table 8. Karnaugh map for intersection 3

Using similar approach we get Karnaugh map for intersection 3 shown in Table 8.

MNDF of this map is:

(!H2 & !J1 & !J2) | (!H1 & !H2 & !I2) | (!I1 & !I2 & !J2) | (!H2 & !I2 & !J2)

Based on MNDF we have 4 control signals:

S9 = !H2 & !J1 & !J2 – green light for *H1, I1, I2*

S10 = !H1 & !H2 & !I2 – green light for *I1, J1, J2*

S11 = !I1 & !I2 & !J2 – green light for *H1, H2, J1*

S12 = !H2 & !I2 & !J2 – green light for *H1, I1, J1*

And we get similar intersection model.

Table 9 shows parameters used to create intersection 3 model. Row *Inc.* contains number of cars incoming to assigned stream per 1 second. Row *Out.* contains number of cars outgoing from assigned stream if traffic lights is green.

	H1	H2	I1	I2	J1	J2
Inc.	0.3	0.4	0.6	0.6	0.4	0.2
Out.	1	2	1.2	1	1.2	1.4

Table 9. List of intersection 3 parameters

3. CONTROL DESIGN

Design of control was made using MPT toolbox for Matlab described in (Kvasnica, 2004) and (Kvasnica, 2009). The aim of control is to set traffic lights so that throughput of intersection is maximum, while the cars on less busy streams do not wait too long to get to turn. Theoretical maximum throughput is achieved when all the lights are green, which of course due to collisions is not possible. Restrictions are summarized in Table 1., Table 2. and Table 3.

Aim of approach used in this paper is to minimize the number of cars facing red lights. To avoid permanent switching of traffic lights performance function penalizes also changes in traffic lights. The aim is to minimize performance function (5):

$$J = \sum_{k=1}^{N} \left\| Q_x x(k) \right\|_1 - \sum_{k=1}^{N} \left\| Q_L u \right\|_1 + \\ + \sum_{k+1}^{N} \left\| Q_U (u(k-1) - u(k)) \right\|_1 \tag{5}$$

subject to:

$$x(k+1) = Ax(k) + Bu(k) + f$$

$$\left\| \boldsymbol{u} \right\|_1 = 1$$

$$\boldsymbol{x}(t+k) \geq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \text{ for } k \in \{1,2,\ldots,N\}$$

where: N is length of prediction horizon,

QX, QL, QU are penalty matrices,

$$Q_L = \begin{bmatrix} L_{A1} & 0 & 0 & 0 \\ 0 & 0 & L_{B1} & 0 \\ L_{C1} & L_{C1} & 0 & 0 \\ L_{C2} & L_{C2} & L_{C2} & 0 \\ 0 & 0 & 0 & L_{D1} \\ L_{D2} & 0 & 0 & L_{D2} \end{bmatrix}.$$

The sence of $||u||_1=1$ constraint is to ensure that during all control time just one of input singnals is set to TRUE.

Using this approach we get three similar control laws – one for each intersection. Relation between these three systems is presented by "road" which connects two arms of intersections.

During testing of this method we created temporary bigger amount of cars coming to stream *B1* of intersection 1. In resulting graphs we can see that designed control algorithm is able to adapt to changed conditions.

Results of this method are depicted in Figures 5-7 for intersection 1, Figures 8-10 for intersection 2 and Figures 11-13 for intersection 3.

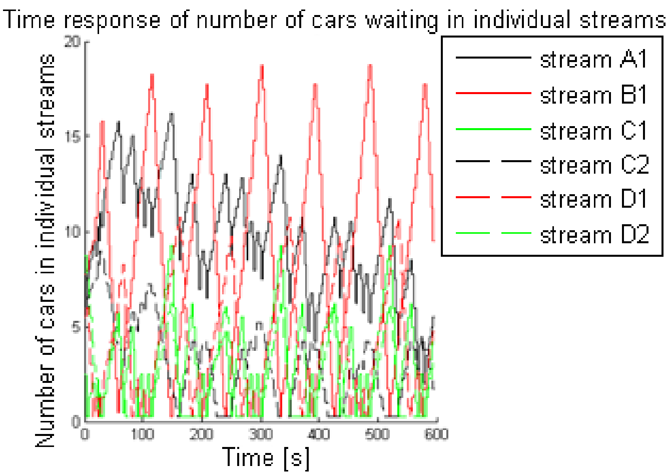

Fig. 5. Time response of number of cars waiting in individual streams in intersection 1

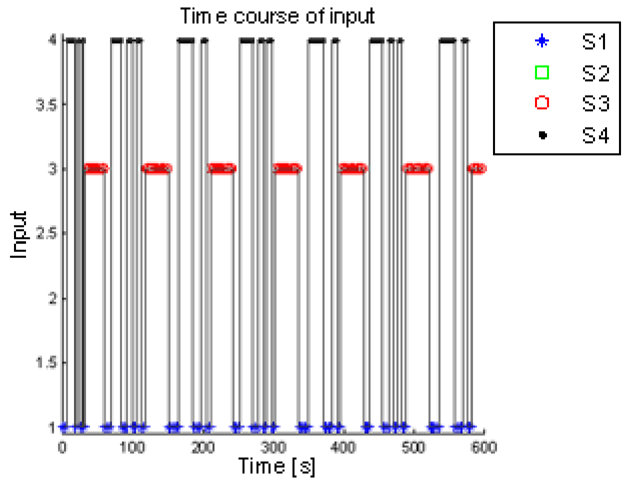

Fig. 6. Input signal S in intersection 1 is set to TRUE when value of function in graph is *i*

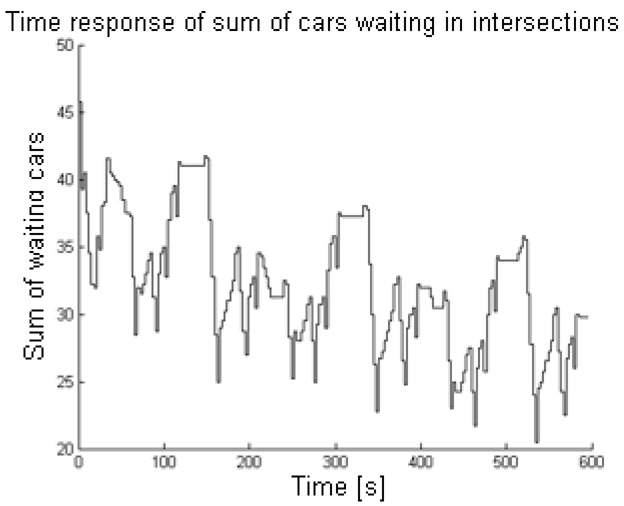

Fig. 7. Time response of total number of cars waiting in streams of intersection 2

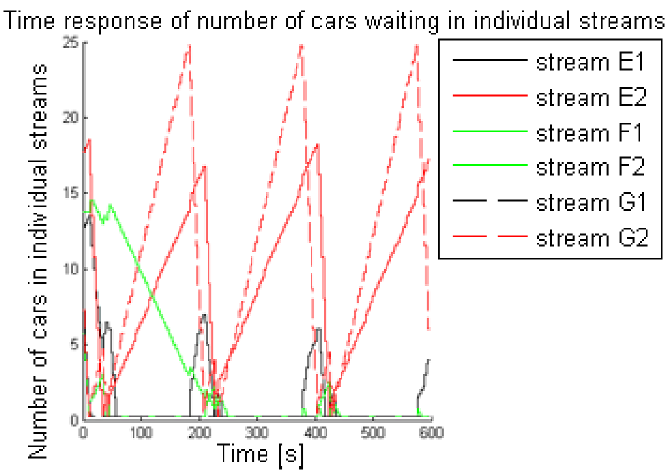

Fig. 8. Time response of number of cars waiting in individual streams in intersection 2

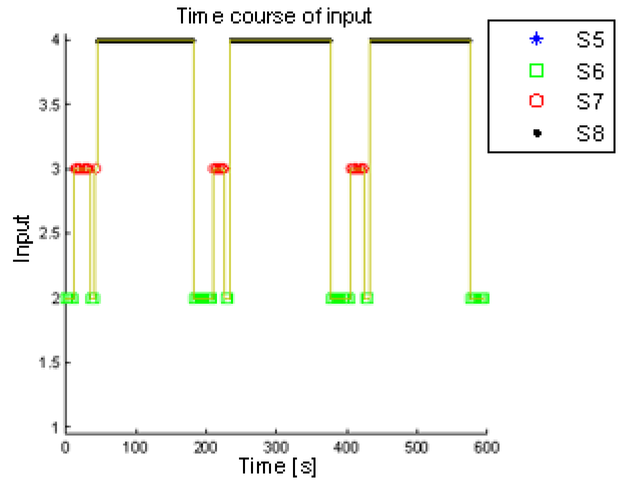

Fig. 9. Input signal S in intersection 2 is set to TRUE when value of function in graph is *i*

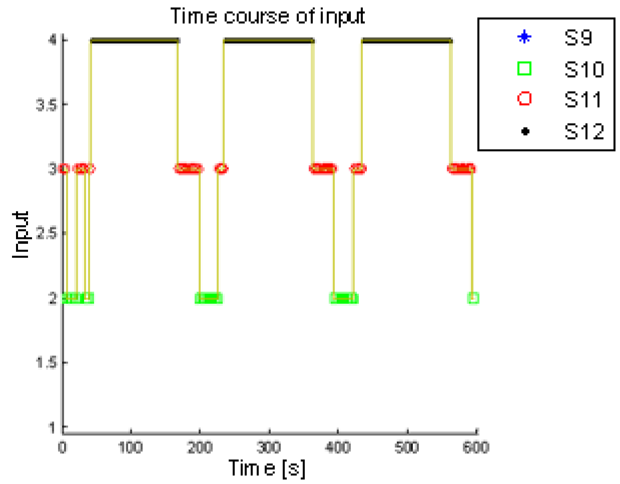

Fig. 12. Input signal S in intersection 3 is set to TRUE when value of function in graph is *i*

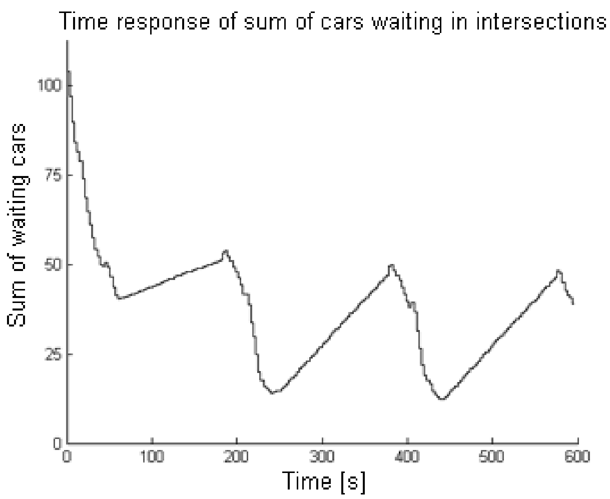

Fig. 10. Time response of total number of cars waiting in streams of intersection 2

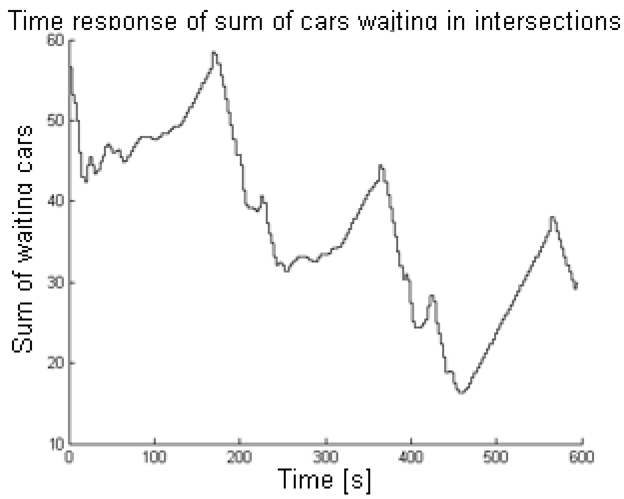

Fig. 13. Time response of total number of cars waiting in streams of intersection 3

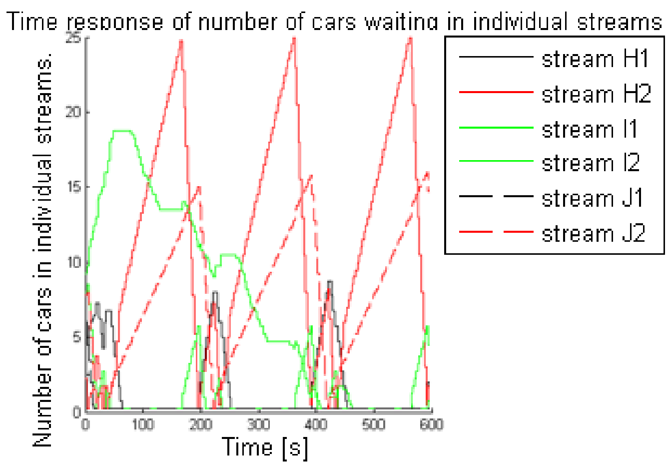

Fig. 11. Time response of number of cars waiting in individual streams in intersection 3

4. CONCLUSION

The paper deals with the design of optimal model structure and innovative hybrid predictive control of typical hybrid dynamic real traffic systems. The main result of the paper is proposal of two interconnected intersections control based on hybrid predictive control. The main objective for traffic lights setting is the number of cars faced to red light. This is a “fair” approach because we let pass through cars which are in longest queue.

To get information about other approaches to modelling and control of transport systems we refer reader to (Kvasnica, 2009), (Saez, 2007), (Cortes, 2009), (Zhang, 2008) and (Huang, 2010).

Advantage of this approach is that it computes with actual lengths of car queues so it is able to adapt to changes in amount of cars coming to intersection as it was shown.

Modelling and simulation obtained results proved that the proposed approach is suitable for real intersections.

ACKNOWLEDGEMENTS

This project was supported by grant of VG 1/1105/11 (VEGA) project and APVV project No VMSP-P-0059-09.

REFERENCES

Bemporad. A. (2004). *Hybrid toolbox – user's guide*. http://www.ing.unitn.it/~bemporad/hybrid/toolbox

Bemporad. A. (2007). *Model Predictive Control of Hybrid Systems*. http://www.dii.unisi.it/hybrid/school07/pdf/08.Bemporad.pdf

Bemporad. A. (2009). Modeling and control of hybrid dynamical systems: The Hybrid toolbox for MATLAB. in Proc. MATHMOD Conference, I. Troch and F. Breitenecker, Eds, *ARGESIM Reports*, 82–100. Vienna, Austria

Cortés. C.E. (2009). Hybrid predictive control for real-time optimization of public transport systems' operations based on evolutionary multi-objective optimization. *Transportation Research Part C: Emerging Technologies*, 757 – 769.

Huang. Y., Chung. T. (2010). Modelling and analysis of traffic light control systems using timed coloured Petri nets. In Pawel Pawlewski, *Petri Nets Applications*. INTECH

Kvasnica. M., Grieder. P., and Baotic. M. (2004). *Multi-parametric toolbox (MPT).* http://control.ee.ethz.ch/~mpt/

Kvasnica. M. (2009). *Real-time model predictive control via multi-parametric programming: theory and tools*. VDM Verlag, Saarbruecken

Sáez. D. et. Al. (2007). Hybrid predictive control strategy for a public transport system with uncertain demand. In *Proceedings of Sixth Triennial Symposium on Transportation Analysis* , Phuket Island, Thailand