Cycle length optimization through bi-level optimization

Yordanka Boneva$^1$

$^1$Institute of Information and Communication Technologies – Bulgarian Academy of Sciences, Sofia, Bulgaria

boneva.yordanka@gmail.com

Abstract. This paper describes the formalization of the store-and-forward model in bi-level optimization problem. The study finds the optimal duration of the cycle lengths by given green splits. The queues in front of traffic lights are minimized. Thus, the network allows higher throughput and less congestion with the resulting from this less pollution and better traffic indicators such as delay, density, speed, etc. The results for traffic indicators and pollution are obtained by Aimsun suit – a specialized traffic modelling software product. The optimization is performed by a MATLAB script based on an additional toolbox called YALMIP as well as with a state-of-the-art software product TRANSYT that is compatible with Aismun suit. Results show that it is reasonable to use the script for optimization of traffic as TRANSYT optimizes only green splits and offsets but works with the cycle value form Aimsun.

1. Introduction

Traffic lights are the main mean of control in urban areas. They separate the traffic flow and pedestrians in time. Thus a minimization of conflicts between different traffic flows and traffic flows and pedestrians is achieved.

There are several parameters related to traffic flow. Firstly, the cycle of the traffic light is defined. The cycle is a sum of green, amber and red light. Secondly, an important parameter is the green light also called green split. The amber light may vary between 2-3 seconds depending on the permitted speed in the section of the road. Usually, for urban settings the amber light is set to 3 seconds.

The optimization of these parameters for a single junction as well as for network of junctions is a prerequisite for optimal throughput of vehicles through the junction. Different optimization approaches are explored for the optimization of the cycle length and the green splits. In this paper the bi-level optimization approach is considered. It has the advantage that two objective functions can be optimized at the same time in contrast to classis optimization approach with one objective function. The goal functions are in hierarchical dependence to each other. There are an upper-level objective function and lower-level objective function. The bi-level optimization used in other areas as portfolio optimization, optimisation of railway transport, etc.[1], [2], [3].

A model of queues in front of traffic lights is used called store-and-forward model. It is a simple model based on communication networks and applied to traffic networks. Queues in front of traffic lights are calculated as sum of the cars that were on the queue at a given moment plus entering cars minus cars that are leaving the link. [4], [5], [6], [7].

Based on the bi-level optimization and the store-and-forward model a problem was formulated and solved. The problem is to optimize the duration of cycle length by given green splits. The solution was achieved through the software products MATLAB, Aimsun and TRANSYT. MATLAB uses the tool box YALMIP and the function solvebilevel() to find the cycle length [8]. Further, the results are tested
in simulations in Aimsun suit. A comparison is made between the results achieved through MATLAB and the result achieved through state-of-the-art software TRANSYT. Results show that there is a small difference in traffic indicators between the case solved with MATLAB and the one solved in TRANSYT in favour of TRANSYT. Still, because the difference is small and because the cycle length cannot be tackled in TRANSYT as it can be in MATLAB it can be concluded that the bi-level optimization is a promising approach. TRANSYT uses the Hill-Climb optimization method for optimization of green splits and offset and take the cycle length as a given constant. MATLAB uses solvebilevel() to tackle the cycle length and takes the green splits in the network as constants.

The data that are used in this study are acquired at site as the selected network is not equipped with sensors. It is though preferable when data is gathered by sensors and cameras [9].

2. Description of the simulated network
The simulation network consists of four neighboring junctions along Shipchenski prohod Blvd. The junctions are regulated by traffic lights and are situated at relatively small distance of each other which is a good prerequisite for achieving a “green wave”. The total distance of the network is 1.5 km.

There are many points of interest in the network like offices, shopping centers, schools, kindergarten, a hospital etc. that generate traffic in the area. Figure 1 is a model of the network in the simulation software Aimsun.

![Figure 1. The network of four junctions](image)

3. Description of the experiment
The experiment aims to reveal the potential of bi-level optimization for the optimization of the cycle length in a network of junctions. The function solvebilevel() in MATLAB was used for this purpose as well as the tool box YALMIP.

A script was written in MATLAB that minimizes the cycle lengths of the four connected junctions and at the same time minimizes the queues in front of traffic lights. The duration of green lights are taken as constant control parameters. Green splits are calculated as a part of the cycle length as the cycle comprises of the sum of green, yellow and red light. If the cycle is taken as 1, the yellow is taken as 0.1*cycle length, green is notated with $u_i$, red is then equal to 0.9cycle – $u_i$. These notifications are important for the understanding of the script in MATLAB and the given model.

The well-known store-and-forward model is applied. The store-and-forward model is a concept taken from computer networks and applied to transport networks. Based on the store-and-forward model the constraints for the lower level problem of the bi-level optimization were defined.
Bi-level optimization comprises of outer and inner objective. In the case of this study the outer objective is the cycle lengths that are to be minimized. The outer objective is the sum of the queues in the network to minimize. However, a higher flow is achieved than the base scenario.

The notations used by the script are as follows:
OO – outer objective
CO – outer constraints
IO – inner objective
CI – inner constraints

The variables are defined. The variables $x_1...11$ are the number of cars in front of traffic lights. The variables $c_1...4$ are the cycle lengths.

\[
\text{sdpvar } x1 \ x2 \ x3 \ x4 \ x5 \ x6 \ x7 \ x8 \ x9 \ x10 \ x11 \\
\text{sdpvar } c1 \ c2 \ c3 \ c4 
\]

The bounds for the cycle length are defined.
\[
c_{\text{min}} = 40/3600; \\
c_{\text{max}} = 120/3600; 
\]

Green split as part of the cycle length is defined.
\[
u1 = 0.422535; \\
u2 = 0.37037; \\
u3 = 0.446429; \\
u4 = 0.416667; 
\]

The flow of cars is given as measured.
\[
x_{1\text{in}} = 1368; \\
x_{2\text{in}} = 870; \\
x_{3\text{in}} = 246; \\
x_{4\text{in}} = 84; \\
x_{5\text{in}} = 150; \\
x_{6\text{in}} = 90; \\
x_{7\text{in}} = 54; \\
x_{8\text{in}} = 60; 
\]

The queue length at previous time step is given. In this case the network is accepted as empty.
\[
x_{10} = 0; \\
x_{20} = 0; \\
x_{30} = 0; \\
x_{40} = 0; \\
x_{50} = 0; \\
x_{60} = 0; \\
x_{70} = 0; \\
x_{80} = 0; \\
x_{90} = 0; \\
x_{100} = 0; \\
x_{110} = 0; 
\]

Saturation flow is given as follows:
\[
s1 = 624; \\
s2 = 744; \\
s3 = 558; 
\]
s4 = 312;
s5 = 620;
s6 = 30;
s7 = 160;
s8 = 12;
s9 = 72;
s10 = 30;
s11 = 30;
s12 = 360;
s13 = 90;
s14 = 80;
s15 = 40;
s16 = 50;
s17 = 20;
s18 = 40;
s19 = 499;
s20 = 418;
s21 = 32;
s22 = 53;
s23 = 237;
s24 = 28;
s25 = 16;

Upper level objective function and upper level constrains are given.
\[
\text{OO} = c_1 + c_2 + c_3 + c_4;
\]
\[
\text{CO} = [c_1 >= c_{\text{min}}, c_1 <= c_{\text{max}},
    c_2 >= c_{\text{min}}, c_2 <= c_{\text{max}},
    c_3 >= c_{\text{min}}, c_3 <= c_{\text{max}},
    c_4 >= c_{\text{min}}, c_4 <= c_{\text{max}}];
\]

Lower level objective function and lower level constrains based on the store-and-forward model are given:
\[
\text{OI} = x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11};
\]
\[
\text{CI} = [x_1 <= x_{10} + x_{1\text{in}} - u_1*s_2*c_1 - u_1*s_1*c_1,
    x_2 <= x_{20} + x_{2\text{in}} - (0.9 - u_1)*s_3*c_1 - (0.9 - u_1)*s_4*c_1,
    x_3 <= x_{30} + u_1*s_2*c_1 + (0.9 - u_1)*s_4*c_1 - u_2*s_5*c_2 - u_2*s_6*c_2,
    x_4 <= x_{40} + x_{3\text{in}} - (0.9 - u_2)*s_8*c_2 - (0.9 - u_2)*s_7*c_2,
    x_5 <= x_{50} + x_{4\text{in}} - (0.9 - u_2)*s_{10}*c_2 - (0.9 - u_2)*s_{11}*c_2,
    x_6 <= x_{60} + (0.9 - u_2)*s_7*c_2 + (0.9 - u_2)*s_{11}*c_2 + u_2*s_5*c_2 - u_3*s_{12}*c_3 - u_3*s_{14}*c_3 - u_3*s_{13}*c_3,
    x_7 <= x_{70} + x_{5\text{in}} - (0.9 - u_3)*s_{15}*c_3 - (0.9 - u_3)*s_{16}*c_3,
    x_8 <= x_{80} + x_{6\text{in}} - (0.9 - u_3)*s_{17}*c_3 - (0.9 - u_3)*s_{18}*c_3,
    x_9 <= x_{90} + u_3*s_{12}*c_3 + (0.9 - u_3)*s_{16}*c_3 + (0.9 - u_3)*s_{18}*c_3 - u_4*s_{19}*c_4 - u_4*s_{21}*c_4 - u_4*s_{20}*c_4,
    x_{10} <= x_{100} + x_{7\text{in}} - (0.9 - u_4)*s_{24}*c_4 - (0.9 - u_4)*s_{25}*c_4,
    x_{11} <= x_{110} + x_{8\text{in}} - (0.9 - u_4)*s_{22}*c_4 - (0.9 - u_4)*s_{23}*c_4
];
\]

x_1 >= 0,
x_2 >= 0,
x_3 >= 0,
x4 >= 0,  
x5 >= 0,  
x6 >= 0,  
x7 >= 0,  
x8 >= 0,  
x9 >= 0,  
x10 >= 0,  
x11 >= 0 ]

solvebilevel(CO,OO,CI,OI,[c1 c2 c3 c4]);

The cycle length is calculated back in seconds.

\[ c1 = c1 \times 3600; \]
\[ c2 = c2 \times 3600; \]
\[ c3 = c3 \times 3600; \]
\[ c4 = c4 \times 3600; \]

The solution of the script with approximation for c1 is 40 seconds, c2 is 55 seconds, c3 is 80 seconds c4 is 40 seconds.

4. Results and Discussion

In the previous point the MATLAB solution was given. In this point the TRANSYT solution will be presented as well as a comparison between the base case, the MATLAB case and TRANSYT case. The base case is a simulation in Aimsun suit based on data as it was measured at site.

Results show that the optimization made with TRANSYT give best results but here it should be noticed that TRANSYT optimizes green splits and offsets whereas MATLAB optimizes the cycle lengths. Although optimizing different parameters both MATLAB script and TRANSYT give better results than the simulation of the base case scenario. And comparing traffic indicators after simulation with MATLAB script input for cycle and TRANSYT input for green split and offset it is obvious that these indicators are in favor of TRANSYT but with very small difference. The results are depicted on Table 1.

| Traffic Indicators                  | Base Case | MATLAB script | TRANSYT Optimization | Units   |
|-------------------------------------|-----------|---------------|----------------------|---------|
| Delay Time                          | 51.16     | 36.4          | 35.51                | sec/km  |
| Density                             | 7.69      | 6.91          | 6.27                 | veh/km  |
| Flow                                | 4095      | 4107          | 4124                 | veh/h   |
| Fuel Consumption                    | 355.53    | 338.06        | 293.98               | l       |
| CO2                                 | 875112.07 | 843316.7      | 768464.08            | g       |
| NOx                                 | 1311.3    | 1250.91       | 1130.48              | g       |
| PM (particulate matter)             | 223.74    | 210.73        | 174.77               | g       |
| VOC (Volatile organic compounds)    | 1107.33   | 997.88        | 917.66               | g       |
| Mean Queue                          | 29.85     | 20.88         | 16.28                | veh     |
| Number of Stops                     | 0.16      | 0.14          | 0.12                 | #/veh/km |
| Speed                               | 34.25     | 37.59         | 38.8                 | km/h    |
| Stop Time                           | 38.42     | 25.48         | 25.76                | sec/km  |
| Total Number of Stops               | 8322.87   | 7404.37       | 6401.28              |         |
| Total Travel Time  | 94.76 | 85.44 | 77.96 | h    |
|-------------------|-------|-------|-------|------|
| Total Travelled Distance | 2925.17 | 2939.58 | 2932.89 | km   |
| Travel Time       | 117.76 | 102.99 | 102.13 | sec/km |

5. Conclusion
In this paper a script in MATLAB for the optimization of cycle lengths of traffic lights network along Shipchenski prohod Blvd. in Sofia was presented. The script solves bi-level problem using the store-and-forward method as a lower-level constraint. A comparison is made with the results of state-of-the-art optimization software TRANSYT. Three scenarios were simulated in Aimsun suit – a software product for modeling and simulation of vehicle traffic.

As a conclusion it can be stated base on the result that the bi-level optimization is worth exploring as an optimization approach for the purposes of traffic optimization.

Acknowledgments
This work has been partly supported by two projects: KP-06-H37/6, 6.12.2019 of the Bulgarian National Science fund: “Modelling and optimization of urban traffic in network of crossroads” and project KP-06-M27/9, 17.12.2018 of the Bulgarian National Science fund: “Contemporary digital methods and tools for exploring and modeling transport flows”.

References
[1] K. Pavlova, B. Vatchova and E Paunova, 2018. Quantitative estimation of throughput abilities in the transport graph under conditions of limited output data, Journal of Bulgarian Science, ISSN: 1314-1031, pp. 45-52.
[2] K. Pavlova, T. Stoilov, 2017. Design of State Rail and Bus Transportation Scheme with Bi-Level Optimization Model. Journal of Information technologies and control, N4, 2017, p.2-9, Print ISSN 1312-2622; Online 2367-5357, DOI: 10.1515/iti-2017-0031. http://www.aksysyst.com:8081/Sai/Journal/Docum/1-4_17.pdf
[3] T. Stoilov, Krasimira Stoilova, 2012, Portfolio Risk Management Modelling by Bi-level Optimization, chapter 5 in Handbook in decision making, vol 2 Risk management in decision making, ed. J. Lu, L. Jain, G. Zhang, Intelligent systems reference library vol.33. Springer – Verlag, Berlin, Heidelberg, ISSN1868-4394, ISBN 978-3-642-25755-1, pp.91-110
[4] D. C. Gazis and R. B. Potts, 1965, Proc. 2nd Symposium on Theory of Road Traffic Flow, O.E.C.D.), p. 221
[5] D. C. Gazis, 1964, Ops. Res., 12, 815.
[6] D.C. Gazis, 1974.Modeling and optimal control of congested transportation systems, Networks, 4, John Wiley & Sons Inc., pp. 113-124.
[7] T. Stoilov, K. Stoilova, M. Papageorgiou and I. Papamichail, 2015. Bi-Level Optimization in a Transport Network. CIT, vol.15, No 5, pp. 37-49. Print ISSN: 1311-9702; Online ISSN: 1314-4081 DOI: 10.1515/cit-2015-0023.
[8] J. Lofberg, 2004. YALMIP : A Toolbox for Modeling and Optimization in MATLAB. In Proceedings of the CACSD Conference. Taipei, Taiwan, IEEE, ISBN: 0-7803-8636-1. pp. 284 – 289, DOI: 10.1109/CACSD.2004.1393890, https://ieeexplore.ieee.org/document/1393890
[9] I. Garvanov and Vladimir Ivanov, 2019. Detection and Parameters Estimation of Moving Objects via Video Surveillance, Proceedings of the 2019 3rd International Conference on Automation, Control and Robots, Prague, Czech Republic — October 11 - 13, ICACR 2019, ACM New York, NY, USA, ISBN: 978-1-4503-7288-6, pp. 102-106, DOI: 10.1145/3365265.3366749, https://dl.acm.org/citation.cfm?id=3366749.