A&QT-R 2004 (THETA 14)
2004 IEEE-TTTC - International Conference on Automation, Quality and Testing, Robotics
May 13 – 15, 2004, Cluj-Napoca, România

Price Based of Centralized Control for Urban Vehicle Traffic

T. Leția, M. Hulea, H. Vălean

Technical University of Cluj-Napoca Romania
Email: Tiberiu.Letia@aut.utcluj.ro, Mihai.Hulea@aut.utcluj.ro, Honoriu.Valean@aut.utcluj.ro

Abstract: The paper offers a centralized solution for the vehicle traffic control in agglomerated urban areas. It is based on the flow rates and the transfer coefficients between lanes. Each input flow into the system has associated a price. The prices for the inside flows are calculated using the loading of the intersections. The control system is composed by negotiators assigned to each intersection. In the first phase they request output rates for their intersections. In the second phase the negotiators grant the input rates taking into account their prices.

Key Words: traffic control,distributed control architecture.

1. INTRODUCTION

The vehicle traffic rates difficulties in agglomerated urban areas, but this is more critical in the old parts of the towns (usually the historic centers) because the density of the buildings is higher, the streets are narrow, the distances between crossroads are short and there is no possibility to intersect the roads at the different levels. So, the solution has to be based on the improvement of the control of the traffic such that to avoid the jamming or blocking and to increase the crossroad throughputs. Architecture of such a system is given by Cremer, M. X., Zhang (1993). The control could be enhanced by an information system [Astilean et al., 2002] that leads to a new level sustaining the control system.

The main characteristic of the urban vehicle traffic is the significant variation of the car rates during the day or on different days (working days or holidays) [Letia et al., 2002].

The control of vehicle traffic could be based on:

- the waiting queues of the cars for crossing the intersections that represent the states [Singh, 1982], [Porche and Lafortune, 1995]
- the flows i.e. the car rates measured on different lanes [Cormen et al., 1990]
- the density of cars distributed in the controlled area

An objective for the first case is to minimize the total delay experienced by traffic at each intersection in a given network.

Solutions of the presented problem are:

1) open loop controllers
2) closed loop controllers using the states of the local crossroad
3) closed loop controllers using the input flow rates of the local crossroad
4) closed loop controllers extending the last two controllers with feedback from the states or the flow rates of the neighbor crossroads
5) optimal systems computing the durations the green lights such that the traffic system evolutes around a steady state
6) discrete event based controllers

The first solution leads to a system which is not able to adapt to the continuous variations of the rates. Such controllers work properly only if the traffic is at a low level. The second solution adapts to the rates variations but the traffic system is prone to blocking because the lane capacities could overflow. The third and the fourth solutions are less prone to blocking but they lead to lower performances. The fifth solution solves the problem around a state which often has to be changed during a day and leads to large computation requirements. The last solution could be used only when the traffic is sporadic (usually during the night) so it does not solve the real traffic problem.

The control system could be implemented centralized or decentralized. The first solution has the advantage that has all information gathered into one computer, but increases the duration of collecting the information and also leads to large computation requirements.

Husch (2000) gives a method for evaluation of the intersection capacity utilization.

Many cars enter or exit the traffic inside the towns because they were or should be parked. In this case, the usage of the states of the system seems inadequate since the current number of cars couldn't be known with a sufficient approximation. For this reason, the usage of the car flows was chosen in the proposed control method. When the input flows of a crossroads are larger than the output flows, an accumulation of cars waiting to cross it appears. If this overflows the capacities of the lanes, the neighbor crossroads are blocked. The phenomenon extends quickly to the other crossroads, so the entire vehicle traffic system is stopped.

Another requirement of the traffic control system is given by the traffic operators which need to intervene with the aim to increase or decrease the flows on different parts of the system.

2. THE CONTROL SYSTEM ARCHITECTURE

The general structure of a real-time information and control system is given in Letia et al., 2003. Figure 1 presents the architecture of the distributed traffic control system proposed to solve the problem. It is structured on two levels with different tasks. Each crossroad has its own local controller that applies the durations of the traffic lights given by the supervisor and sends periodically information about the flows. It also, estimates the local matrix of the transfer rates of the car from one lane to another. This matrix should be evaluated periodically because (as it was mentioned) of the variations of the rates during the day. The transfer rate matrix is also sent to supervisor which uses it to improve the allocation of the green light durations to different input lanes of the crossroad. On the lanes are installed sensors which measure the rates of the cars. The controllers act on the traffic lights implementing an open loop algorithm. These add a reduced (5 seconds representing a flow of 3 cars/minute) rate applied cyclically to the lanes that were not open during previous cycle (for the safety and ethic reasons). This avoids long (or quite unlimited) waiting durations.

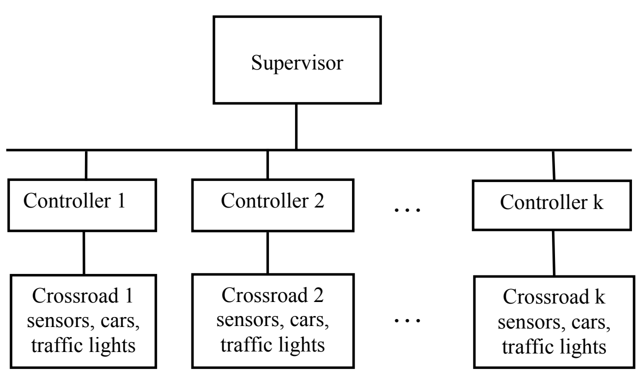

Fig. 1. The distributed traffic control system.

The paper focuses on the supervisor. It is based on a centralized implementation with the aim to avoid the large number of messages implied by a distributed solution. The supervisor has a negotiator for each crossroad. It tries to get the best duration allocation of the traffic lights to increase the crossroad performance. Each car flow (through a lane) has a price. The prices of the input lanes to the system are granted by the traffic operator. The other prices (of the lanes inside the traffic system) are calculated by the negotiators using formulas described in the supervisor algorithm.

3. THE SUPERVISOR

The algorithm of supervisor is based on the price control of the car flows. The vehicle traffic system has n lanes which connect k crossroads. The supervisor uses the table presented in fig. 2. referred to as LaneTable.

Notations	Lane name	Lane_1	Lane_2	...	Lane_n
r_i	Requested rate				
g_i	Granted rate				
p_i	Price for 1 car/min				
a_i	Available rate				

Fig. 2. The LaneTable of the vehicle traffic system.

The LaneTable contains the requested rate, the granted rate, the price and the available rate of each lane. It is used in all the phases of the algorithm.

The supervisor algorithm has three phases:

Initially, the requested rates and the offered prices for the input lanes in the systems are given by the operator. He fills them in the LaneTable. The prices of all the rest of the lanes are set to 100. The granted rates are set such that each input lane of a crossroad gets the same duration; this means the same granted output rate from the lane. The matrix of the transfer rates of each crossroad is given. Its elements are denoted by t_{xy} for the transfer from the lane l_x to lane l_y.

1) *The bid phase.* Each negotiator (of each crossroad) reads the rates requested at its input lanes and their prices. It calculates the rates and the prices of its output lanes and stories them as request rates and offered prices in LaneTable. The negotiator sums its requested input rates to calculate the crossroad load. The prices are calculated using the prices offered for the input lanes and the load of the crossroad. This phase continues until no negotiator changes their requested rates and the offered prices.
2) *The grant phase.* Each negotiator uses the granted rates of its output lanes and calculates the rates which it grants for its input lanes. For this aim it considers that the rates of its output lanes couldn't be exceeded. Also the sum of rates which it grants for its input lanes could not exceed its maximum throughput (capacity). The negotiator grants the input rates giving priority for the lanes with higher prices. The phase is finished when no negotiator changes the rates (or the prices) that it grants for its input lanes. This represents a solution of the problem.
3) *The evaluation phase.* The supervisor calculates the performance of the solution summing up for each crossroad the differences between the crossroad capacity and its current throughput. The performance is storied.

The following notations are used:

- c_i – the crossroad i
- I_i – the set of the input lanes of the crossroad i
- O_i - the set of the output lanes of the crossroad i
- cap_i – the maximum throughput (capacity) of the crossroad i

- *$load_i$* – the sum of the requested rates of the crossroad i
- l_x – the lane x of the traffic system
- r_x – the rate requested for the lane l_x
- t_{xy} – the element from the position (x,y) of the transfer rate matrix
- a_i – the available rate of the crossroad i to be distributed
- n_i – the number of input lanes of the crossroad i

The details of the *bid phase* are:

1. Initialization. The prices of all input lines in the traffic system are set by the operator (usually around 100) and the rest to 100 units i.e. p_i=100, i=1,...,n.

2. Calculus of the requested rates. For each crossroad c_i, i=1,...,k the following are performed:

- the crossroad *load* are initialized to zero, i.e. $load_i$=0;
- the *requested rates* (for each of its output lanes $l_y \in O_i$) are calculated for the crossroad c_i with the formula

$$r_y = \sum_{l_x \in I_i} r_x \cdot t_{xy}$$

- the *load* of the crossroad c_i is calculated with the formula

$$load_i = \sum_{l_x \in I_i} r_x \, ,$$

- the prices offered for the output flows are calculated with

$$p_y = \Big(\sum_{l_x \in I_i} r_x \cdot t_{xy} \cdot p_x\Big)/r_y \; - k \cdot (cap_i - load_i) + subvention_i$$

where the difference (cap_i - $load_i$) is multiplied by a factor that determines the influence of the solicitation of the crossroad on the price (it was chosen between 0.05 and 0.2) and the $subvention_i$ means the subvention added by the operator if he wants the input flows of the crossroad i to have priority for rates allocation. The rising of the price with a subvention on one crossroad has effects on the neighbor intersection too.

The requested rates and their prices are storied in the LaneTable for the usage by other negotiators or in the next iteration.

3. *The END of the bid phase.* If during the iteration all the negotiators don't change the requested rates or their prices the phase is ended; otherwise it continues from the step 2.

The details of the *grant phase* are:

1. *Initialization.* For each crossroad c_i, i=1,...,k the following are performed:

The values of the granted rates g_j, j=1,...,n_i of each input lane ($l_j \in I_i$) such that the crossroad capacity is equally distributed between them i.e. $g_j = cap_i/n_i$.

2. Calculus of granted rates.

For each crossroad c_i, i=1,...,k the following are performed:

- for each output lanes $l_y \in O_i$ do $a_y = g_y$;
- each input lane $l_x \in I_i$ is opened by controller with the minimum (safety) rate (i.e. 5 seconds that represents 3 cars/minute) cyclically so, calculates the sum of the rates that could be granted $d = cap_i - 3 \cdot n_i$ cars/minute;

  This should be distributed according to the offered prices and requested rates.

- calculates the *available* rate of each output lanes $l_y€O_i$ diminishing with the granted rate for safety, i.e. $a_y=a_y-3$;
- find the lanes with the highest price from the crossroad (for each input lane l_j, $j=1,...,n_i$) that has a request unserved yet. Let $l_x€I_i$ be that.
  Calculates the granted rate value g_x that for each $l_y€O_i$ the relations $a_y \geq g_x \cdot t_{xy}$, $r_x \geq g_x$ and $d \geq g_x$ are fulfilled. The value g_x of the lane l_x is storied.
  For all the output lanes of l_x the value *available* is calculated with $a_y=a_y-g_x \cdot t_{xy}$. The rate still available to be distributed from the crossroad i to the remained input lanes is $d=d-g_x$.
- the previous step is repeated until all the requested rates are tried to be allocated or d=0;

3. *The END of the granted phase.* If no significant changing of granted rates for all k crossroads has been performed, the phase ends; otherwise repeat the *calculus of granted rates*.

The openness of more than one input lane of a crossroad (which does not intersect their trajectories) could be taken into account also, but that does not change the essence of the algorithm. In this case, the calculus of granted rates has to consider the presented conditions simultaneously.

The detail of the *evaluation phase* is:

For each crossroad c_i, i=1,...,k the current *crossroad throughput* is calculated with the formula:

$$throughput_i = \sum_{l_x€I_i} g_x$$

This provides the medium number of cars crossing the intersection during a minute. The *system throughput* is:

$$throughput = \sum_{i=1,...,k} g_i$$

The rates allocation of crossroads is better if the system throughput is higher. Another way to appreciate the system performance is given by the formula:

$$performance = \sum_{i=1,...,k} (cap_i - throughput_i)$$

This evaluates the usage of the crossroad capacities.

The *benefit of the traffic system* is:

$$benefit = \sum_{i=1,...,k} g_i \cdot p_i$$

This should be used to appreciate the traffic system performance. It takes into account the system throughput, but also uses the prices given by operator.

The operator can impose a higher priority path through the system of streets raising the price of the corresponding input flow into the system and setting subventions for all the crossroads of the path. This manner achieves for operator an opportunity to intervene into the system behavior without the danger to overflow the lane capacities, which means to avoid the system blocking.

4. ANALYSIS OF THE SUPERVISOR ALGORITHM

The analysis of the algorithm implies the study of the behavior of the bid phase and the grant phase. It is necessary to verify if the algorithm improves always its solution at each iteration and if it stops in a finite number of iterations.

The question is how a change of a rate or a price at the crossroad input lanes propagates to the system. At the first iteration the neighbor crossroads detects the modification on its input. Further, the changes evolutes like throwing a stone in the water of a lake. Like the waves of the water, “the price or the rate waves penetrate” deeply to the system of crossroads. The effects of the change become smaller after several iterations, but even if this is not ignored, we can say that the latest effects are detected on the remotest crossroad not later than the largest number of crossroads contained on each side of the system of crossroads.

For the grant phase, the situation is the same. So, the number of iterations necessary for each phase is given by the smallest number of crossroads contained between the remotes crossroads.

5. CONCLUSIONS

The algorithm is constructed such that each input flow into a crossroad has the openness to its output lane. This avoids the system deadlock. On the other hand, the algorithm tries to improve the solution at each step without ignoring the aim to avoid the deadlock.

The control system keeps outside the system the cars until they can go through the traffic without blocking the system.

The negotiators have the aim to get higher prices for their crossroads that could defer from the maximum capacity utilization. So, a bad initially price allocation could lead to lower crossroad utilizations, but this could not be avoided if the flows should have priorities. The conclusion is that the prices in this algorithm play the role of performance or cost function in optimal systems.

6. REFERENCES

Astilean, A., Avram, C., Leția, T., Hulea, M., Vălean, H.: Using Mobile Data Services for Urban Driving Advisory Systems, *Mobile Open Society through Wireless Telecommunications Conference, Warsaw, Poland,* pp. 12-23, 2002.

Cormen, T.H., Leiserson, C.E., and Rivest, R.R.: *Introduction to Algorithms*. Massachusetts Institute of Technology, Cambridge, Massachusetts, 1990.

Cremer, M. X., Zhang: System Architectures for Complex Road Traffic Information and Control Systems, *Proc. of IFAC 12th World Congress*, Sydney, Vol. 9, p.229-232, 1993.

Husch, D.: Intersection Capacity Utilization, www.trafficware.com, 2000.

Letia, T. S., Astilean, A., Avram, C., Hulea, M., Valean, H.: Urban traffic control system, *2002 4th International Workshop on Computer and Information Technology, University of Patras, Greece*, September 2002.

Letia, T., Avram, C., Hulea, M., Valean, H.: Real-Time Information and Control System. *IFAC/IFIP Workshop on Real Time Programming*, June, 2003, Poland.

Porche, I. and Lafortune, S.: Dynamic traffic control: Descentralized and Coordinated methods using a limited look-ahead strategy, Technical Rep. CGR-95-12, College of Engineering Control Group Report Series, The University of Michigan, Dec. 1995.

Porche, I. and Lafortune, S.: Adaptive Look0ahead Optimization of Traffic Signals, The University of Michigan, 1995.

Singh, M.G. and A. Titli: Systems: Decomposition, Optimization and Control. *Pergamon Press*, Oxford, 1978.