Cooperative Control for Urban Vehicle Traffic

T. S. Letia * **O. Cuibus** **

* *Technical University of Cluj-Napoca, Romania (e-mail: Tiberiu.Letia@aut.utcluj.ro).*
** *Technical University of Cluj-Napoca, Romania (e-mail: ocuibus@yahoo.com)*

Abstract: The paper presents a cooperative control approach applied to an urban vehicle traffic system that is implemented through a set of communicating agents. The distributed controllers use fuzzy logic rules to get the control decisions. To improve the local control performances the agents send each other their contexts. Improving the control involves more terms in the fuzzy logic control rules. This has the effect of strongly increasing the number of rules. Hierarchical architectures are chosen to diminish the number of logic rules for different methods. To obtain the unknown fuzzy logic control rules the generic genetic approach is used. The control rule sets, the simulations results and their comparison are given for some cooperative methods.

1. INTRODUCTION

Urban Vehicle Traffic (UVT) is a very complex and dynamic system, involving many vehicles that travel within a network of roads linked by crossroads. The vehicles follow some routes that can be either pre-established or not. The states of traffic lights are assigned to phases that govern the crossing through intersections. Each phase has associated one or more input lanes and this permits the crossing towards the output lanes without involving conflicts.

Vehicle traffic incidents and the exceeding of the intersection capacities can lead to traffic congestions. Papageorgiou et al. (2003) explain the congestion appearance on freeways. The congestions of UVT can be seen (detected) when some vehicles, that start the crossing on green light, remain inside intersections even when they loose the right of crossing (signaled by red light) because the output lanes are overloaded. The congestions diminish the intersections throughputs and as consequence, these diminish the whole UVT system transfer capacity.

The performance of the UVT system can be evaluated using several factors such as the vehicle waiting times, the throughput and the system transfer capacity. As total, current or mean value, the waiting time is often used as a measure of performance. The throughput of the system is defined as the total number of vehicles that cross through all the intersection in a period. An intersection has a transfer capacity estimated as the maximum throughput. The maximum possible total system throughput is the system transfer capacity. A particular evaluation is the sum of transfer capacities of some specified paths of the urban road system.

⋆ This research is funded by the European Union in the framework of the SEE initiative Transnational Cooperation Programme by the I3E project (code SEE/A/219/1.1/X)

Bazzan (2005) stresses out that the main goals of an UVT control are:

- to maximize the overall capacity of the network,
- to maximize the capacity of critical routes and intersections that represent the traffic bottleneck,
- to maximize the negative impact on the environment and on energy consumption,
- to minimize travel times and
- to increase traffic safety.

In the current paper, the main control goal is to avoid congestions, because they have a negative and relevant influence on the above-mentioned goals, by evolving a so-called stop and go move. Another goal is to obtain the highest possible system throughput that avoids the congestions.

The centralized control methods use an actor to control the whole system. The distributed control approach splits the system into some controlled subsystems where each has its own controller. The main advantage is when a controller falls or the communication is interrupted the rest of the system can continue to behave correctly. It is possible for a controlled subsystem that fails or fails to communicate with the rest of the system, to influence other controlled subsystems. In this case, the influence of the subsystem that quits the cooperation is seen as a disturbance. Other advantages of the distributed approaches are a reduced communication volume and an increase of the control system reactivity.

Typically, intersections were/are controlled independently using fixed phases and fixed cycle lengths. Better performance is achieved when the phases of neighbor intersections are correlated. Moreover, the phases of the traffic lights must be correlated with the variable traffic demand and hence, the large volume of calculations makes the centralized control strategy unfeasible. The old solution that avoids the centralized control drawback used fixed time phases. Unfortunately, the dynamic independent controllers have their own local goals that can be in conflict when the system is highly loaded. Local goals typically include maximizing intersection throughput, minimizing intersection congestion times, minimizing waiting queues lengths and vehicle waiting times, etc. High throughput of one intersection can result in long waiting queues of other intersections that in turn can lead to congestion of the first intersection. This represents a conflict of goals.

Distributed control for UVT is proposed by Bazzan (2005) to avoid the long times needed to collect the distributed information, the large amount of calculus required for the centralized control and to improve the reliability. The dynamic distributed control can be adapted to the current throughput demands, but it has to solve the main problem related to the conflicts of local controllers.

A solution to avoid the local controller conflicts is given by the cooperative control, in which some agents exchange information and control decisions in order to reach a certain established common goal.

The distribution of the UVT control can be successfully approached by cooperative control, since it allows the development of complex behavior based on several controllers combined to achieve some common goals as desired results.

2. STATE OF THE ART

The main problems that one faces when dealing with UVT systems are:

- design a control network that fulfils some specified requirements controlling the traffic system using current, previous or predicted traffic demand values, while also concerning system throughput, waiting times and avoiding or diminishing congestions,
- finding a route in the traffic network (from departure point to destination) that fulfils certain requirements: minimize a given evaluation function, avoid certain areas (high vehicle density areas), include certain checkpoints, etc.,
- guiding a vehicle from the departure point to the destination such that some requirements are fulfilled: reach the destination before a given moment, reduce fuel consumption, diminish traffic system fault effects, etc.

The problems of UVT can be divided into structure problems and strategy problems. Some of the structure problems are:

- using fixed or variable cycle durations for crossroad traffic light;
- if the cycle durations are fixed, the periods have to be set;
- if the cycle durations are fixed, the offsets between neighbor intersections have to be set;
- set the number and order of phases for each intersection and assign lanes to phases.

Some of the most used approaches of UVT control problem include:

- operations research (linear, nonlinear, integer, and dynamic programming, queuing networks, decision analysis),
- linear or non linear control (optimal, adaptive, model predictive),
- artificial intelligence (fuzzy logic and rule based control generally comprised in expert systems, game theory, planning),
- software engineering (static or dynamic resource allocation),
- real-time system approaches (synchronous or asynchronous).

A possible solution to the UVT problem is to find the best route to travel from the departure point to the destination (Avella et al, 2004). Yue and Shao (2007) present a shortest path search algorithm in dynamic UVT. The use of ant colony systems to solve the problem of vehicle routing is a possible approach (Donati et al, 2008).

Papageorgiou et al. (2003) present a remarkable review of methods for local, centralized or distributed control of UVT. The use of stochastic dynamic programming for the development of implementable heuristics is applied to the dynamic control of queuing networks (George and Harrison, 2001).

The application of expert system concepts in the UVT control implies the use of given rules for deciding the controllers actions. A network of roads connected by traffic light-based expert systems is presented by Findler et al. (1992). The expert systems are synchronized and their set of rules can be optimized by considering the rules frequency of usage and their rates of success. An expert system to vehicle traffic control is used by Kirschfink, et al, (2000). For this aim, intelligent analysis and predictions monitor the current traffic situations in the relevant granularity.

Several studies show that applying Fuzzy Logic Control (FLC) in the UVT leads to superior results compared to the classical traffic light control strategies. Tan et al. (1995) propose a fuzzy logic controller that should mimic human intelligence and applied it to a single crossroad. The use of fuzzy logic for controlling a set of crossroads is studied by Lee et al. (1995). A hierarchical fuzzy logic traffic controller for a real intersection is developed by Hu et al. (2009). The fuzzy rule base is obtained by using an evolutionary algorithm. The comparison of the results obtained with the FLC and the controller currently employed in the intersection shows that the FLC clearly shortens the vehicle waiting time.

Henry et al (1998) use neuro-fuzzy techniques for controlling the traffic lights. A new hybrid neural network model is used by Min et all (2006) to design local traffic signal controllers. The neural networkbased local controllers need to adapt to the traffic network changes generated by the random fluctuation of the traffic volumes.

Wiering (2000) proposes multi-agent reinforcement learning algorithms applied to the traffic light controllers that learn how to minimize the overall waiting time of cars in the city. The reinforcement learning systems learn value functions estimating expected waiting times for cars given different setting of traffic lights. An approach that uses reinforcement learning for controlling a single traffic light and coordination between the neighboring traffic lights is presented by Kuyer et al (2008). The coordination is based on the max-plus algorithm which estimates the optimal joint action by sending locally optimized messages among the connected agents. Steingrver et al. (2005) describe the optimization of traffic light controllers using a multi-agent, model-based reinforcement learning approach.

Cooperative control of distributed multi-agent systems involves: distributed control and computation, adversarial interactions, uncertain evolution and complexity management.

Roozemond (2001) proposes a system that can autonomously adapt itself, based upon internal rules and its environment, at changing of UVT parameters.

Fan et al. (2008) propose a mutual beneficial cooperation model for UVT control based on a Lotka-Volterra model that is used for simulating population dynamic relation in ecology. The study points out the evolvement directions of the cooperation system and shows how the system can be developed rapidly by the magnification of the cooperative effects.

Zhao et al. (2005) outline the importance of cooperation between the traffic control and route guidance subsystems. An interesting de-congestioning strategy is also described.

Yang et al. (2005) propose an intelligent cooperation algorithm based on genetic reinforcement learning. The described system is capable of managing ever changing traffic conditions in real-time.

3. UVT CONTROL

The control system can control the vehicle flows or the vehicle waiting queues.

As already mentioned, UVT control can use fixed or variable intersection periods. Taking this into account, the solution of the control problem provides for each intersection the phase durations $d_1, \cdots, d_p$. Denoting by:

- p is the number of phases,
- T is he cycle period,
- T_y is the total duration of yellow color of traffic lights during a cycle,
- T_p is the total duration of green color for pedestrian during a cycle and
- T_r is the duration of the clearance red color,

one can observe that the constraint (1) is fulfilled.

$$d_1 + \cdots + d_p + T_y + T_p + T_r = T \quad (1)$$

The current approach uses for the UVT control fixed intersection periods, fixed order of intersection phases and fixed offsets between intersection periods. These parameters are calculated and set offline and do not represent objectives of the current study.

The following assumptions are considered fulfilled:

- Each intersection has a control subsystem that contains a controller and an agent.
- The controllers provide the information related to traffic demands for each lanes and lane occupancies to the associated agents .

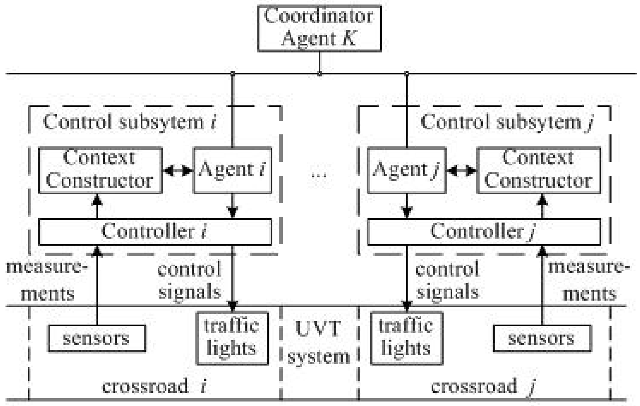

Fig. 1. Cooperative system architecture

- The controllers apply the phase durations (received from associated agents) using the intersections traffic lights.
- There is a duration O equal to the least common multiple of all intersection periods.
- Each agent works at the end of the intersection period and provides specified information to neighbor agents.
- Each agent sends to its neighbor agents the specified information.
- The agents calculate the phase durations at the end O period and provide them to associated local controllers.
- Each lane is opened at least a small duration (usually, 5 seconds) during any period.
- Each driver fully respects the driving rules and the colors of the traffic lights.

4. COOPERATIVE CONTROL

The cooperative control strategies can be organized in three main types of methods:

- implicit cooperative control,
- coordinated cooperation and
- explicit cooperative control.

These methods differ regarding the information that is used for the calculation of the control durations and whether an agreement between agents is achieved about the control decisions.

Figure 1 shows the general architecture of the UVT control system using cooperation between agents. Context constructor builds the control context taking the information provided by the local controller and the operator. The control context is composed of:

- state measurements,
- constraints,
- measured or estimated disturbances,
- predictions,
- operator recommendations or requirements,
- inference information from the past,
- context information communicated by other agents and
- neighbor control decisions.

The agents are able to communicate with the purpose of improving their control decisions.

In the case of independent controllers the architecture lacks the communication facilities. Each agent takes the control decision individually, based on its own context. The calculation of intersection i phase durations $(d_1^i, \cdots, d_n^i)$ should minimize the performance function listed below.

$$J = \sum_{k=1}^{H} (a^i \cdot x^i(k) + b^i \cdot c^i(k)) \quad (2)$$

with the constraints:

$$x^i(k+1) = f^i(x^i(k), u^i(k), d^i(k), y^i(k), c^i(k)) \quad (3)$$

$$y^i(k) = g^i(x^i(k), d^i(k), c^i(k)) \quad (4)$$

$$c^i(k) = h^i(x^i(k), d^i(k), z^i(k), l^i) \quad (5)$$

where

- a^i is the coefficient vector that penalizes the lengths of waiting queues,
- $u^i(k)$ is the the vector input flows (cars lining up at the queue),
- $y^i(k)$ is the output flow vector,
- $x^i(k)$ is the state vector representing the queue lengths of the input lanes at the time k
- b^i is the coefficient vector that penalizes the congestions of the intersection
- $c^i(k)$ is the congestion vector of intersection i at the time k
- $f^i(\cdots)$,$g^i(\cdots)$ and $h^i(\cdots)$, are the vector functions that model the intersection behavior,
- $d^i(k)$ is a vector with the element values 1 if the corresponding phase is opened during the time k,
- $z^i(k)$ is a state vector representing the queue lengths of the output lanes at the time k,
- l^i is capacity vector of the output lanes of intersection i.

Instead of on-line optimization of the presented function J_i that often takes a long time, the current research used the FLC.

The general form of the inference rules is presented in (6).

$$IF(x_1 \in X_1) \wedge \cdots \wedge (z_1 \in Z_1) \cdots \wedge (z_m \in Z_m)$$
$$THEN(d_i \in D_1); \cdots (d_p \in D_p); (d_r \in D_r) \quad (6)$$

The first part of the relation introduces by the values x_i the condition of the loading of the input lanes and by z_i the condition of the loading of the output lanes. The notations X_i, Z_i and D_i correspond to the logical domain of the fuzzy logical variables. The consequence concludes about relative phase durations and the total durations of phases during of a period. Different implementations of the formula (6) lead to different kinds of control systems.

4.1 Independent phase control

Each phase of each intersection is controlled by an independent controller that uses only the queue lengths of the allocated input lanes. The maximum queue length or a usual statistically known lane with the maximum queue can be used with the purpose of reducing of the number of variables.

The form of logical inference formulae is (7).

Table 1. **Inference rules for independent phase control**

x_i	L	M	H
d_i	L	M	H

Table 2. **Inference rules (phases 1 and 3)**

$x_1 \backslash x_3$	L	M	H
L	L,L	L, M	L,H
M	M,L	M,M	L,H
H	H,L	H,L	M,M

Table 3. **Rule sets for implicit control**

$x_1 \backslash x_3$	L	M	H
L	L	L	L
M	L	M	M
H	L	M	H

$$IF(x_i \in X_i)THEN(d_i \in D_i) \quad (7)$$

Table 1 contains the inference rules.

4.2 Independent intersection control

Better results are expected when all the phase durations of an intersection are calculated together using information regarding all the queue lengths. By grouping two phases the inference rules have the form (8) and (9) for an intersection with four phases.

$$IF(x_1 \in X_1) \wedge (x_3 \in X_3)THEN(d_1 \in D_1); (d_3 \in D_3) \quad (8)$$

$$IF(x_2 \in X_2) \wedge (x_4 \in X_4)THEN(d_2 \in D_2); (d_4 \in D_4) \quad (9)$$

The rules are included in Table 2 for the case when phase 1 and phase 3 are grouped together. Identical rules are obtained for the phase 2 and 4.

The extra time that is not used for a phase due to the inference can be added to its pair phase. The mentioned rules were determined using a genetic algorithm.

An improvement can use all the phases but the number of rules is increased significantly. Due to the fact that the independent controllers do not have the values z_i corresponding to the information about the output lane states; these can be estimated and set off-line and included in the premise. The expected results have to be lower that in the case when this information is actuated and sent on-line.

4.3 Implicit cooperative control

In implicit cooperative control the agents send each other the control context. The agents can determine better phase durations using the values z_i and that improves the system general behavior. The rules of the form (10) can be used for each phase of an intersection.

$$IF(x_i \in X_i) \wedge (z_i \in Z_i)THEN(d_i \in D_i) \quad (10)$$

Using genetic algorithms the rule set presented in Table 3 is obtained.

Obviously, the rules become more complex when the information about all the phases is included in the same rule. To diminish the problem complexity a hierarchical structure of the rule set implementation can be used.

Table 4. **Coordinator rule set**

$x_i \backslash z_i$	L	M	H
L	L	L	M
M	L	M	H
H	M	H	H

Table 5. **Agent rule set**

$x_i \backslash e_i$	L	M	H
L	L	L	L
M	L	M	M
H	L	M	H

4.4 Coordinated cooperative control

To implement this, the agents send their information (containing their context and their intended phase durations) to a coordinator. The role of the coordinator is to constrain the phase durations that could lead to congestions. To avoid these, the coordinator uses for each intersection a rule set with the form (11).

$$IF(d_1 \in D_1) \wedge \cdots \wedge (z_1 \in Z_1) \cdots \wedge (z_m \in Z_m)$$
$$THEN(e_i \in E_1); \cdots (e_p \in E_p) \quad (11)$$

where e_i (i=1,... ,p) represents the higher bound accepted for the phase durations.

Each agent calculates the phase durations using the values ei received form the coordinator, instead of the values z_i. For one phase, the simplified rules for agents and coordinator are of the form (12) and (13).

$$coordinator : IF(x_i \in X_i) \wedge (z_i \in Z_i) THEN(e_i \in E_i) \quad (12)$$

Finally, each agent applies (here for a phase) the constraints given by the rules of the form (13).

$$agent : IF(x_i \in X_i) \wedge (e_i \in D_i) THEN(d_i \in D_i) \quad (13)$$

The rule sets presented in Table 4 and Table 5 were obtained using genetic algorithms.

4.5 Explicit cooperative control

The cooperative agent algorithm is presented below.

(1) Send to each other the context.
(2) Receive the neighbors context and estimate the z_i values.
(3) Calculate the phase durations using rule sets of the form (13)
(4) Repeat steps (1) to (3) until no change in phase durations or queue lengths is obtained.

The algorithm is bounded because an agent changes the y_i vector only if it gets an improvement (i.e. to increase the value) for some elements. Because the lane input buffer is bounded, the number of iterations is limited.

5. TESTS AND RESULTS

The described methods were used to control an urban vehicle traffic system implemented by software simulation. The system has 4 crossroads and each direction has 3 input lanes with fixed flow splits, and one output lane. The intersection cycle durations are fixed (76 sec.), equal, and the offset is 0. Traffic demand is considered random, with new vehicles entering each external road at a rate of 0.8 vehicles/second. Vehicles pass by the intersections at a rate of 1 vehicle/second. Standard membership functions were used for fuzzification and defuzzification.

Table 6. **Simulation results**

Control strategy	Throughput	Congestion	GA exec. time
Fixed phase	532	1797	-
Indep. phase	1649	2551	15 sec.
Indep. int.	2574	6259	10 min.
Implicit c.	2059	2268	2 min.
Coordinated	1160	938	20 min.
Explicit c.	2833	41	15 min.

Table 6 shows the results obtained through simulations. The last column contains the execution durations of GA used to determine the rule sets. The second column presents the measurements of the system throughput (the number of vehicles leaving each intersection) in 1000 seconds. The third column contains the measurement of the system congestions (the total number of cars that remain in each intersection having lost the right to cross).

6. CONCLUSIONS

The on-line use of FLC is appropriate for local controller implementations due to the short execution times. When more information is used, the performances of the control system are improved, but more complex rule sets are necessary.

The GA is used off-line to determine the rule sets. They need longer times for execution when the rule lengths are increased. The hierarchical structures for the rule sets can be used to diminish the rule lengths and so to decrease the GA execution times.

REFERENCES

M. Papageorgiou, C. Diakaki, V. Dinopolou, A. Kotsialos, and Y. Wang. Review of road traffic control strategies. *Proceedings of the IEEE*, 91: No. 12,pages 2043-2067, 2003.

A. L. C. Bazzan. A distributed approach for coordination of traffic signal agents. *Autonomous and Multi-Agent Systems, Springer Science+Bussines Media*, 10:131-164, 2005.

P. Avella, M. Boccia, and A. Sforza. Resource constrained shortest path problems in path planning for fleet management. *Journal of Mathematical Modelling and Algorithms 3*, Kluwer Academic Publishers. 135:pages 1-17, 2004.

H. Yue, C. Shao. Study on the application of a shortest path search algorithm in dynamic urban traffic. *Third International Conference on Natural Computation*, 2007.

A. Donati, R. Montemanni, N. Casagrande, A. E. Rizzoli and L. M.Gambardella. Time dependent vehicle routing problem with a multi ant colony system. *European Journal of Operational Research*, 185:pages 1174 -1191, 2008.

J. M. George, J. M. Harrison. Dynamic control of a queue with adjustable service rate. *Operations Research*, 49(5):pages 720-731, 2001.

K. K. Tan, M. Khalid, R. Yusof. Intelligent traffic lights control by fuzzy logic. *Malaysian Journal of Computer Science*, 9-2, 1995.

N. Findler, and J. Stapp. A Dstributed Approach to Optimized Control of Street Traffic Signals. *Journal of Transportation Engineering*, 118-1:pages 99-110, 1992.

H. Kirschfink, J. Hernndez, and M. Boero. Intelligent traffic management models. *Proceedings of ESIT*, Aachen, Germany, pages 36-44, 2000.

J. Lee, K. Lee, K. Seong, C. Kim, and H. Lee-Kwang. Traffic control of intersection group based on fuzzy logic. *Proceedings of the 6th International Fuzzy Systems Association World Congress*, pages 465-468, 1995.

Y. Hu, A. Chiou and Q. Han. Hierarchical fuzzy logic control for multiphase traffic intersection using evolutionary algorithms. *IEEE International Conference on Industrial Technology*, pages 1-6, 2009.

J. J. Henry, J.L. Farges and J.L., Gallego. Neuro-fuzzy techniques for traffic control. *Control Engineering Practice*, 6:pages 755-761, 1998.

D. Min, C. C. Srinivasan and R. L. Cheu. Neural network for continuous online learning and control. *IEEE Transactions on Neural Networks*, volume 17, No. 6, pages 1511-1531, 2006.

M. Wiering. Multi-agent reinforcement learning for traffic light control. *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 1151-1158, 2000.

L. Kuyer, S. Whiteson, B. Bakker and N. Vlassis. Multiagent reinforcement learning for urban traffic control using coordination graphs. *Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases*, Part I,pages 656-671, 2008.

M. Steingrover, R. Schouten, S. Peelen, S. Nijhuis and B. Bakker. Reinforcement learning of traffic light controllers adapting to traffic congestion. *Proceedings of the Belgium-Netherlands Artificial Intelligence Conference*, 2005.

D. A. Roozemond, Using intelligent agent for pro-active, real-time urban intersection control. *European Journal of Operational Research*, volume 131, issue 2 :pages 293-301, 2001.

W. T. Fan, R. Y. Wang and Z. X. Huang. The analysis of mutual beneficial cooperation among urban traffic modes. *Proceedings of the International Conference on Intelligent Computation Technology and Automation*, ISBN 978-0-7695-3357-5, pages 516-520, 2008.

S. Y. Zhao, C. Xin, S. T. Yang and P. S.Jian. Intelligent cooperation control of urban traffic networks. *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics*, Guangzhou,pages 1482-1486, 2005.