Towards Traffic Light Control through a Multiagent Cooperative System: A Simulation-Based Study

Journal:	*Spring Simulation Multiconference*
Manuscript ID:	SMC-2005-ADS-038
Topic Area Conference:	Agent-Directed Simulation
Submission Type:	Full Paper for Review
Date Submitted by the Author:	01-Dec-2004
Complete List of Authors:	Guzman, Francisco; Monterrey Institute of Technology, Center for Intelligent Systems Garrido, Leonardo; Monterrey Institute of Technology, Center for Intelligent Systems
Keywords:	Agent-Based System, Multiagent System, Cooperative System, Traffic Light Control

Towards Traffic Light Control through a Multiagent Cooperative System: A Simulation-Based Study

Francisco Guzmán and Leonardo Garrido
Tecnológico de Monterrey
Centro de Sistemas Inteligentes
Monterrey, N.L., México
guzmanhe@exatec.itesm.mx and leonardo.garrido@itesm.mx

KEYWORDS
Agent-Based System, Multiagent System, Cooperative System, Traffic Light Control.

ABSTRACT

Present day traffic networks are unable to efficiently handle the daily car traffic through urban areas. We think that multiagent systems are an excellent way of doing microscopic simulation and thus provide possible solutions to the traffic control problem. In this paper, we present our simulation-based study to simulate traffic networks and optimize them via a multiagent cooperative system for traffic light control. This system simulates the traffic on an intersection, minimizing the time that each car has to wait in order to be served. Light agents can communicate each other in order to negotiate and share their light times. Our experimental results have shown how our approach can decrease the average car delay while the spawn probability is increased varying the service time and the number of traffic lights sets at a specific intersection. These results show important improvements using our multiagent light control system.

1. MOTIVATION

The 20th century witnessed the worldwide adoption of the automobile as a primary mode of transportation. Coupled with an expanding population, present-day traffic networks are unable to efficiently handle the daily movements of traffic through urban areas. At present, traffic lights may possess sensors to provide basic information relating to their immediate environment. The use of such sensors provides greater flexibility within traffic lights, since more appropriate patterns can be calculated for the current situation.

Traffic simulation has been in existence for many years. They all had to use simplified models of traffic flow in order to produce results within practical timescales, as discussed in [8]. A typical assumption is to represent traffic flow in a particular road as a single quantity. Such models are called macroscopic models. Microscopic simulation enables more accurate study of congestion formation/dispersion and emphasizes the insight into the nature of road traffic flow. During each time step, the vehicles move towards their destinations, as in real-life.

Current methods for enabling traffic to flow through intersections include building overpasses and installing traffic lights. However, the former is only worth the cost at the most congested intersections, and the latter can be quite inefficient. Improvements to urban traffic congestion must focus on reducing internal bottlenecks to the network, rather than replacing the network itself. Of primary concern is the optimization of the traffic lights, which regulate the movement of traffic through the various intersections within the environment. Multiagent systems are an excellent way of doing microscopic simulation and thus provide possible solutions to the traffic problem. Agents are expected to work within a real-time, non-terminating environment. As well, agents can handle dynamically occurring events and may posses several processes to recognize and handle a variety of traffic patterns.

There have had several approaches to developing multiagent traffic simulations. From those approaches which are concerned with modeling human behaviors and psychological issues, such as [5] and [7], to those who are more interested in modeling traffic and transportation systems.

Balmer et. al. [1] proposed an interesting system for simulating traffic and people transportation.

Their main concern was to simulate real-world scenarios with millions of travelers.

Penner et. al. [6] presented SuRJE which is a tool for testing and optimizing traffic light sequences. Their approach is a combination of ant-based ideas developed in the area of Swarm Intelligence and a evolutionary approach.

France and Ghorbani [3] presented a system for optimizing urban traffic not only in one intersection but in several. Their approach, however, may not represent properly real traffic behavior in congested situations, since they use the concept of traffic flow as a single quantity.

Dresner and Stone [2] proposed this year a reservation system which stands upon the assumption that a single car is completely autonomous and computer controlled. The fact is that such assumption cannot be considered as an every day life's scenario.

In this paper, we propose a new system which simulates the traffic on an intersection and optimizes it, plugging the information that can be obtained from sensors into our traffic light cooperative multiagent system. The remainder of this document is organized as follows: Section 2 describes our multiagent-based simulator that has been implemented; In section 3, we describe the basis over which the proposed cooperative multiagent system states; In section 4, we explain the variety of experiments that have been done using this simulator and the experimental results; Finally, we discuss, in section 5, our conclusions, current, and future work.

2. SIMULATOR

In order to offer a possible improvement to the traffic problem, it was required to build a simulator that permitted the analysis of different traffic patterns. Therefore the simulator that has been built is as customizable as possible. It allows wide variety of scenarios to be setup on an $n \times m$ grid. Figure 1 shows a screenshot of the graphical display of our simulator. We have developed this simulator using the MadKit Multiagent platform [4].

The model under which our simulator is ruled is simple. We do not have a quantitative approach but a qualitative one, meaning that we simulate the behavior of cars and traffic light not under typical units (e.g. meters, seconds) but rather in our own environment's states (e.g. time steps, position on the grid). Nevertheless, the resulting behavior is comparable to the behavior produced by other simulators using typical units.

Now, we give here a brief description of the most important agents of our simulator:

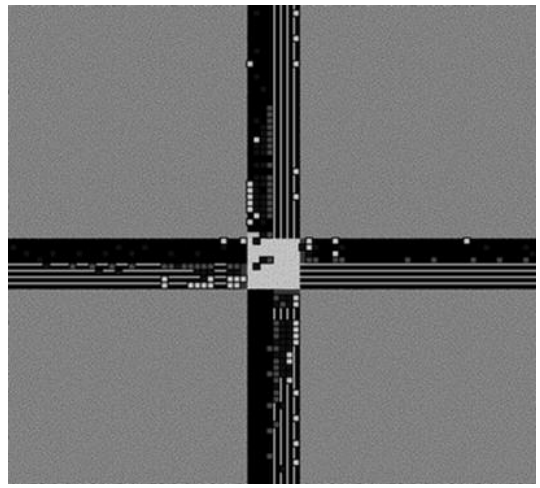

Figure 1. A screenshot of the graphical display of our simulator configured with 4 lanes in each direction.

- **Car:** This agent is perhaps the most important agent in the simulation. It is purely reactive. It obtains its data by communicating with the environment. It is aware of other cars, crossways and traffic lights.
- **Traffic Light:** This agent has several internal states that represent the different light colors in a traffic light.
- **LightSet:** Every traffic light agent belongs to a single traffic LightSet. This set is in charge of changing the state of its aggregated traffic lights. It has a start and finish service times, which represent the times when the traffic lights are going to be on green state. It also keeps track of the current state of traffic flow. If a lane that a Traffic Set is managing is very congested, it will trigger the cooperative system.

- **Traffic Manager:** The simulator has a special agent which is in charge of coordinating the different traffic LightSets in the simulation.
- **Source:** This agent spawns cars with a predefined probability λ into a predefined position. It allows us to determine whether a car spawned will be able to turn, and if so, to which direction. Even if this agent may not exist in real life, it is comparable to sensors placed on streets from which we are able of getting information.

For each cycle of the simulator, the following events occur:

1. On each street, a SOURCE agent spawns cars with a predefined probability. Once the vehicle is spawned, it is placed at a predefined position (where the SOURCE agent is located). If that position is occupied by any other vehicle, the will not be spawned.
2. The driver of each car possesses a “vision” limit. It is able to look forward and check for the state of a traffic light which is in within the “vision” range. If it sees a red light, it will stop at a given distance. If not, it will avoid colliding with other cars ahead by slowing down if another car is too close. Since the simulator won’t allow cars to change lanes, this sole restriction is enough to prevent accidents.
3. The driver can take several actions depending on the state of its environment: STOPAT (to stop the car at a given distance), ACCELERATE (to increase speed by a fixed acceleration), DECELERATE (to decrease speed), TURN (to change the direction to which the car is leading).
4. Any vehicle leaving the environment is removed from the simulation. Before “dying”, the vehicle outputs its important data in to a log file.

Finally, as we have said, our simulator is very customizable. The system allows its user to determine the number of streets, the direction of each, the position of the traffic lights, their operating times; the position of the sources (car spawning agents), their car spawning rate, and the destination of cars.

3. COOPERATIVE SYSTEM

In this traffic system, it is of interest to minimize the time that a car has to wait in order to be served (green light). Therefore, we need indicators to be able of measuring the performance of an intersection. The system developed is based on the M/M/1 queuing system. For which, we consider each traffic light as a server with an arrival rate λ and a departure rate μ. For this to hold true, we made the following assumptions:

1. The arrival process of cars corresponds to a Poisson process with parameter λt.
2. The server has an exponential service time distribution with a mean service rate μ.
3. There is a single traffic light per lane.
4. The size of the queue is infinite.
5. The number of potential cars to be served is infinite.
6. All traffic lights belonging to the same LightSet have the same mean service rate.
7. The worst case congestion scenario for LightSet will be the one for which λ is the greatest.

If we recall the mean waiting time for an M/M/1 queuing system we obtain:

$$W = \frac{1}{\mu - \lambda} = \frac{1}{\mu(1-\rho)}$$

Here, λ is a parameter of the simulation (the spawn probability of each source). On the contrary, the way μ is determined it’s more complicated. Let’s suppose a departure rate distribution over time from one light as the shown in Figure 2. **T** is the total light cycle time, *M(t)* represents the quantity of cars that are served by a light by time step, μc is the maximal amount of cars that one light can serve on a time step (simulator cycle) and τ is the amount of time a light remains on green per cycle (ser-vice time).

Then, average depart rate can be calculated as:

$$\bar{\mu} = \frac{1}{T}\int_0^T M(t)dt \approx \frac{1}{T}\int_{start}^{finish} \mu_c dt = \frac{\mu_c \tau}{T}$$

Then we can write the utilization rate as:

$$\rho = \frac{T\lambda}{\mu_c \tau}$$

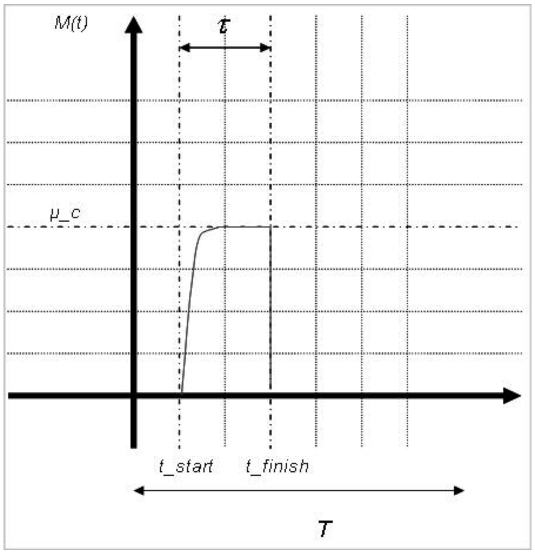

Figure 2. Graph showing the service distribution. We assume that a traffic light serves μc cars per simulator cycle

From this equation we can observe that the utilization rate of a light will be smaller as the greater is its service time τ.

In order to diminish the time a car has to wait to be served, we introduced a cooperative multiagent system, in which each LightSet is continuously checking its utilization rate ρ. The way it works is the following:

1. The LightSet checks for the worst case of congestion among its aggregated lights and obtains its λ.
2. It calculates the actual utilization rate. If this rate is greater than a "tolerance utilization rate" ρc (determined by the user) then it sends an `open_proposal` message to neighbor LightSets. It initiates a call for proposals, asking for time that neighbors may not need. It also creates a cooperative group called `coop_group`, for which it is the manager and has the role of *leader*.
3. When the other LightSets receive a message pointing that there is a call for new proposals, they subscribe to `coop_group` with the role of *providers*. They propose a time quota (Δτ), in function of their actual utilization state. That quota needs to be long enough to be considered a good offer by the *leader* but small enough so that LightSet remains operational (that means ρ<ρc). The way a time quota is calculated is the following:

$$\Delta\tau = \alpha\left[\frac{T\lambda}{\mu_u}\left(\frac{1}{\rho} - \frac{1}{\rho_c}\right)\right]$$

   Where α is a "security factor" that will prevent a provider of going into critical state because of lending too much time.
4. After a LightSet has calculated the proposed time (time quota) it sends the message `reply_proposal` to the *leader*, in which the amount of time proposed and the LightSet own address is contained. This is for the owner to keep track of the proposals.
5. The *leader* makes a contest to determine which the best of propositions is.
6. A Light cycle T passed, the *leader* closes the call for proposals, announcing to the whole group the *winne*r, the amount of time proposed by the winner and increases its ser-vice time the amount of Δτ .
7. The *winner* decreases its service time by Δτ .

4. EXPERIMENTS AND RESULTS

The configuration of the scenario used to test the efficiency of the cooperative multiagent system consist on a crossway with one lane for each direction, four traffic lights (one for each lane), an initial spawn probability of 0.01 for all sources with no turning cars. The test done increases the spawn probability in steps of 0.005 for one lane, which will result in its gradual congestion. As the congestion increases, the utilization of that lane also increases, triggering the cooperative multiagent system. Two different parameters of simulation were varied, in order to get a wider range of responses. These parameters are the initial duration (τ) of service time for every Traffic Set and the number of Traffic Sets present in the system.

4.1. Service time variation

For this experiment, the number of Traffic Sets was set to 4, meaning that each lane's traffic light is controlled in-dependently and has an initial duration of τ. For each experiment, we varied τ from 10 to 40 in steps of 10. The behavior of the most congested lane (the one with increasing spawn probability) is shown in Figure 3.

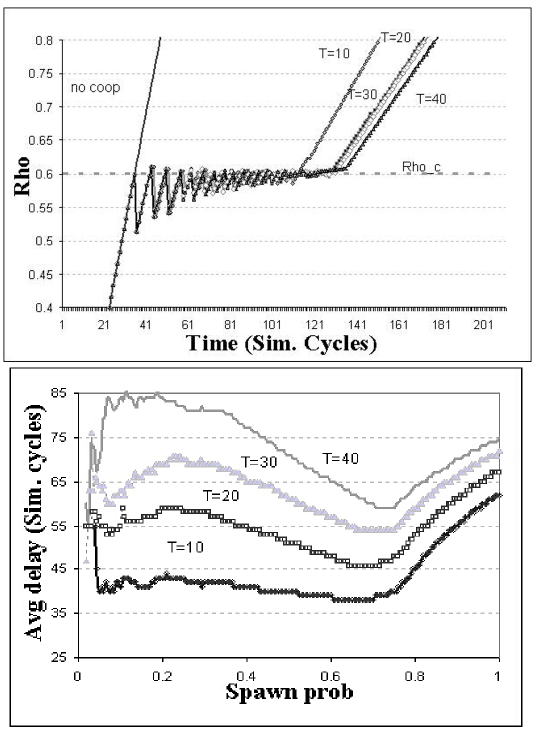

Figure 3. Graphs showing the response of the cooperative multiagent system varying service time, under an increasing spawn probability regime.

Notice that in the first graphic (top), all of the different configurations behave the same way (as if it didn't exist any cooperative multiagent system) before they reach the point where $\rho > \rho c$. Then the cooperative multiagent system is triggered and starts regulating the traffic in the congested lane. It is important to re-mark that the greater the service time is, the more time passes before the cooperative multiagent system saturates. Once the bid-ding system saturated, ρ keeps growing but to a smaller rate (its slope has reduced).

In the second graphic (bottom), observe that the aver-age delay for cars traveling in the most congested lane keeps increasing as the spawn probability increases. Around the spawn value of 0.2, the delays start decreasing (when the cooperative multiagent system goes on) until they reach a minimum (which corresponds to the saturation point). Then increase again. Notice that as the service time increases, the aver-age delay also increases. It is important to point out that even if we may observe that for a greater initial service time τ, the total bidding range will be wider, the average delay will be also greater, resulting in cars that have to wait longer to be served. Nevertheless, if we take the middle values such as $\tau = 30, 20$, we may observe that it is possible of getting a better performance, since the cooperative multiagent system takes longer before going into saturation, and also the average delay times are the optimal.

4.2. Number of Traffic Sets

For this experiment, the service time was set to 40 while the number of traffic sets varied between 2 (meaning that traffic lights from north and south directions share the same service time, as well as east-west) and 4 (each light has its own service time). The results of experiments are shown in Figure 4

Notice that for the first graphic (top) the greater the number of sets, the longer the cooperative multiagent system last without saturation. For the second graphic (bottom), observe that the gains on performance are more substantial with 4 sets. It is also important to remark that as the spawn probability (λ) increases the average delay has an asymptotic behavior. For instance, in the 4 sets with no bid line, the aver-age delay would go asymptotic to a value near to 210 simulator cycles. This may not be as expected from equation 1 from where we may assume that as λ goes bigger (greater than μ) the waiting time trends to reach the infinity. Nevertheless, this behavior may be attributed to a fail on assumption 4, which states that the size of the queue is infinite while in our scenario, this condition is difficult to accomplish as sources won't produce more cars if the place over which they must spawn cars is already occupied. Yet, in real life situations, the queue length can be considered as infinite, since there is no limit for the amount of cars that can be waiting in an intersection to cross.

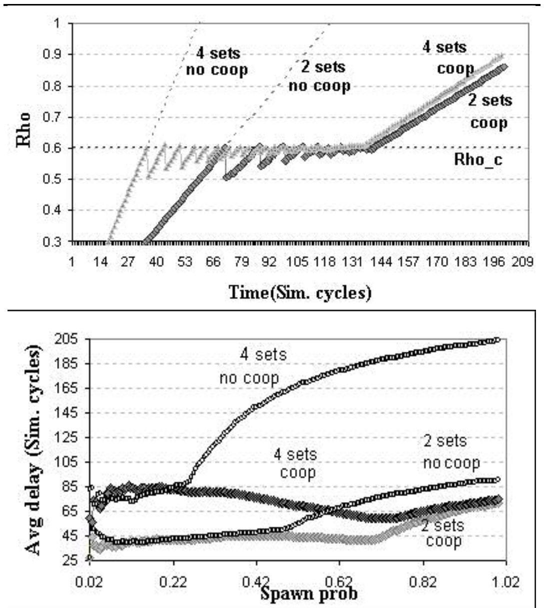

Figure 4. Graphs showing the response of the cooperative multiagent system varying the number of light sets, under an increasing spawn probability regime.

4.3. Cooperative Multiagent System efficiency

Another quantity used as reference is the mean improvement which is calculated by the difference of the value of the average time with no cooperative multiagent system and the average time with the cooperative multiagent system.

$$\frac{1}{N}\sum_{n=1}^{N}\left(\bar{t}_{nobid}(n) - \bar{t}_{bid}(n)\right)$$

The results of the performance analysis is summarized on table 1.

Sets \ τ	10	20	30	40
2	7.767	9.035	10.636	12.828
4	74.237	75.136	85.252	75.510

Table 1. Table showing the performance of different runs

From these results, we may observe that the greatest improvement resulting from using the cooperative multiagent system is more related to the number of sets that may be present in the system, rather than to the amount of time they possess. For instance, improvements for the 4-set configuration are remarkably greater than those for a 2 set configuration. In the other hand, we would expect that the greater the initial τ is, the greater the improvements. Nevertheless, the best results for all simulation were obtained with τ = 30.

Notice that the empirical data obtained shows that the improvements that can be done to the traffic light time distribution using a cooperative multiagent system are substantial. Besides, it is important to remember that the results shown in this study were obtained with a configuration where only one of the lanes is congested. Also, we should remark that the cooperative multiagent system is useful when and only when there is at least one lane that is not congested. If the system is completely congested, the cooperative multiagent system will not be able to improve the actual state of an intersection since there will not be any bidding time that may "circulate" among traffic lights.

5. CONCLUSIONS AND FUTURE WORK

Basically, we have presented the first empirical results of a simulation-based study on traffic light control through a multiagent cooperative system. The main goal of this new system is to coordinate and optimize the time distribution for traffic lights at each intersection.

In order to conduct this simulation-based study, we have designed and developed a multiagent traffic simulator. This simulator has been built is as customizable as possible. It allows wide variety of scenarios.

We have also proposed a new multiagent cooperative system for traffic light control. This system simulates the traffic on an intersection, minimizing the time that each car has to wait in order to be served. Light agents can communicate each other in order to negotiate and share their light times.

Our experimental results have shown how our approach can decrease the average car delay while the spawn probability is increased varying the service time and the number of traffic lights sets at a specific intersection. These results show important improvements using our multiagent light control system.

Currently, we are working on an extension to several intersections. The idea is to continue using the cooperative system for local optimization at each intersection but now enabling communication among intersection agents. In this way, lights at each intersection will be able to predict the traffic flow and try to optimize it now from a global point of view.

REFERENCES

[1] Balmer, Michael, Nurhan Cetin, Kai Nagel, Bryan Raney. "Toward truly agent-based traffic and mobility simulations"

[2] Dresner, Kurt and Peter Stone. "Multiagent traffic management: A reservation-based intersection control mechanism". In *The Third International Joint Conference on Autonomous Agents and Multiagent Systems*, 2004.

[3] France, John and Ali A. Ghorbani. "A multiagent system for optimizing urban traffic". In *IEEE/WIC International Conference on Intelligent Agent Technology*, 2003.

[4] Gutknecht, Oliver and Jacques Ferber. "The MadKit Project". http://www.madkit.org/

[5] Paruchuni, Praveen, Alok R. Pullalarevu, and Kamalakar Karlapalem. "Multi Agent Simulation of Unorganized Traffic". In *the 1st international joint conference on Autonomous Agents and Multiagent Systems*, 2002.

[6] Penner, Joanne, R. Hoar, and C. Jacob. "Swarm-Based Traffic Simulation with Evolutionary Traffic Light Adaptation". In the *International Conference Applied Simulation and Modeling*, 2002.

[7] Shculze, Thomas and Thomas Fliess. "Urban Traffic Simulation with Psycho-physical vehicle-following models". In *the 29th Conference on Winter Simulation*, 1997.

[8] Tibaut, Andrej. "Parallel traffic simulation with an algorithm for adaptive load-balancing". In *Internationales Kolloquim ber An-wendungen der Informatik und Mathematik in Architectur und Bauwesen*, 1997.

BIOGRAPHY

Francisco Guzmán

Francisco Guzmán is currently participating in the Washington Center's NAFTA Leaders Program. Through the program, he secured an internship with Counterpart International, Inc., where he provides support to the Food Security and Sustainable Agriculture Divisions with IT projects. He is also volunteering as IT consultant for the Environment Office at the Mexican Embassy in Washington D.C. Prior to his involvement with the NAFTA Leader's program, Francisco worked at the ITESM in Monterrey, Mexico, as a research assistant, where he coauthored a research paper discussing the use of Cooperative Multiagent Systems to optimize the flow of urban traffic. In Paris (2003), he took part in an industrial software architecture project focused on Production Processes Management. He was an intern for Synergie (2003), Travail Temporaire in Nantes, France, where he was in charge of a preliminary study of Workflow technologies implementation. He worked as a teaching assistant for migrant children in the NOMAD bilateral (USA- MEX) program in Kalamazoo, MI and he also worked as an English teacher for children. He has volunteered in several CCIVS (UNESCO, 2003) work camps, in Noto, Italy and in Liège, Belgium. He completed his undergraduate studies with a French-Mexican double-degree in Industrial Physics Engineering (ITESM, Mexico) and Computer Science (EPF, France).

Leonardo Garrido

Leonardo Garrido is professor and researcher at the Center for Intelligent Systems at the Monterrey Institute of Technology in Monterrey, N.L. México. He got his Doctoral Degree in Artificial Intelligence. His doctoral research was about modeling other agents in multiagent systems, doing a simulation-based study on the performance of modeler agents using different approaches such as Bayesian and reinforcement learning techniques. He has conducted several research projects on multiagent systems and artificial intelligence since 1994 at the Center for Intelligent Systems.