PAPER

ACTAM: Cooperative Multi-Agent System Architecture for Urban Traffic Signal Control

Ruey-Shun CHEN†, Duen-Kai CHEN†a), *and* Szu-Yin LIN††, *Nonmembers*

SUMMARY The traffic congestion problem in urban areas is worsening since traditional traffic signal control systems cannot provide] efficient traffic regulation. Therefore, dynamic traffic signal control in Intelligent Transportation System (ITS) recently has received increasing attention. This study devised a multi-agent architecture, the Adaptive and Cooperative Traffic light Agent Model (ACTAM), for a decentralized traffic signal control system. The proposed architecture comprises a data storage and communication layer, a traffic regulation factor processing layer, and a decision-making layer. This study focused on utilizing the cooperation of multi-agents and the prediction mechanism of our architecture, the Forecast Module, to forecast future traffic volume in each individual intersection. The Forecast Module is designed to forecast traffic volume in an intersection via multi-agent cooperation by exchanging traffic volume information for adjacent intersections, since vehicles passing through nearby intersections were believed to significantly influence the traffic volume of specific intersections. The proposed architecture can achieve dynamic traffic signal control. Thus, total delay time of the traffic network under ACTAM can be reduced by 37% compared to the conventional fixed sequence traffic signal control strategy. Consequently, traffic congestion in urban areas can be alleviated by adopting ACTAM.

key words: *intelligent agents, multi-agent systems (MAS), decentralized control, dynamic traffic signal control*

1. Introduction

The traffic congestion problem in urban areas recently has attracted considerable attention in major cities around the world. According to research conducted by the Texas Transportation Institute (TTI), located at Texas A&M University, the United States, commuters in one third of the largest cities in the U.S.A. spend an average of over 40 hours a year, equivalent to one working week, in traffic jams. To alleviate traffic congestion in urban areas, the concept of Intelligent Transportation Systems (ITS) has been widely accepted in developed countries. ITS is a highly promising system for providing key solutions to current road congestion problems. ITS has the potential to reduce traffic jams, traffic accidents and environmental emissions by improving traffic flows and transportation efficiency [2]. Advanced Traffic Management System (ATMS), described in ITS, includes the planning of traffic signal control system. ATMS aims to use various technologies, particularly real-time traffic volume monitoring, to assist in determining traffic light control strategies. The primary issue of ATMS is how to monitor real-time traffic volume and design a real-time traffic light control strategy based on monitored traffic volume. Traditional traffic light control strategies developed during the 1970's and 1980's were inadequate for dealing with extremely heavy urban traffic.

Numerous studies have considered local optimal control of traffic signal systems, for example, PASSER II [10] and TRANSYT-7F [11]. Other studies, such as Wey in [3], employed linear optimization methods to solve traffic signal control problems.

Image processing sensor systems [5] have drawn considerable attention in the field of ITS, and the information extracted from image processing sensor systems can be applied to achieve real-time responses to the incident in question. Recently, multi-agent decentralized strategies for controlling urban traffic networks have attracted considerable attention. Roozemond and Rogier [4] proposed a prototype using agent technology to control traffic signal systems.

This study proposed a multi-agent architecture, the Adaptive and Cooperative Traffic light Agent Model (ACTAM) [1], for decentralized traffic signal control, which improved existing systems and reduced total delay time. Multi-agent systems are designed to enable problem decomposition and subcontract subproblems to different problem solving agents with their own goals and interests [9]. The proposed architecture comprises three layers, the Data layer, Processing layer and Decision layer, and these three layers oversee communication, traffic indices generation, decision-making and actuation, respectively. This study focuses on utilizing the Forecast Module to forecast the future traffic volume in each individual intersection. Furthermore, each agent has the ability to communicate, enabling each agent to exchange relevant traffic information and cooperate with multi-agents. ACTAM enables dynamic traffic signal control strategy to be achieved in a decentralized and collaborative way. Total delay time of the traffic network can be reduced. To improve current traffic signal control system adoption in urban areas requires developing a more efficient solution. The traffic signal control system based on multi-agent technology outperformed the traditional method in providing a real-time and proactive response to dynamic traffic volume. Traffic signal control agents cooperate via communication and coordination mechanisms. Analysis of the historical data enables the proposed architecture to learn and adjust its control strategy. Architecture flexibility is achieved by modular design. The simulation result supported the present hypothesis, since Multi-agent systems

Manuscript received June 16, 2003.
Manuscript revised April 20, 2004.

†The authors are with Institute of Information Management of National Chiao-Tung University, Taiwan 300, Ripublic of China.
††The author is with Department of Computer Science, National Tsing-Hua Univ., Taiwan 300, R.O.C.
a) E-mail: cdk@iim.nctu.edu.tw

can improve the efficiency of traffic signal control system.

The remainder of this paper is organized as follows. The next section describes the agent technology and its cooperative paradigm features. Section 3 then reviews previous attempts to solve the traffic signal control problem. Subsequently, Sect. 4 presents the architecture, ACTAM, proposed in this study. Section 5 then summarizes experimental results and discusses the experimental results. Finally, Sect. 6 presents some conclusions.

2. Agent Technology

This section first examines various definitions of software agents, then highlights applications of software agents, and finally briefly introduces the agent system and its common architecture.

Agent technology is a key area in [artificial intelligence research. Today, a software agent generally means a software program that accomplishes a task on behalf of its user]. According to this definition, the history of software agents can be traced back to the early 1970's, when was being conducted in the fields of software engineering, human interface research and artificial intelligence [9]. There is no generally accepted definition of software agents, and researchers involved in agent research have offered a variety of definitions.

As Jennings and Wooldridge defined, "An agent is a computer system situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives." [18]

Russell and Norvig define "An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors." [7].

Hayes-Roth defines "Intelligent agents continuously perform three functions: perception of dynamic conditions in the environment; action to affect conditions in the environment; and reasoning to interpret perceptions, solve problems, draw inferences, and determine actions" [13].

Lange and Oshima believed that "From the end-user perspective, an agent is a program that assists people and acts on their behalf. Agents function by allowing people to delegate work to them. From the system perspective, an agent is a software object that situated within an execution environment; possesses mandatory properties such as reactive, autonomous, proactive and temporally continuous; may possess any of the orthogonal properties such as communicative, mobile, learning, believable" [14].

Smith et al. defined an agent as a persistent software entity dedicated to a specific purpose. "Persistent" distinguishes agents from subroutines; agents have their own ideas about how to accomplish tasks, their own agenda. "Specific purpose" distinguishes agents from entire multifunction applications [12].

In [9], software agents are required to possess the following minimal characteristics:

- Delegation: The agent performs a set of tasks on behalf of a user (or other agents) that are explicitly approved by the user.
- Communication skills: The agent needs to be able to interact with the user to receive task delegation instructions, and inform task status and completion through an agent user interface or through an agent communication language.
- Autonomy: The agent operates without direct intervention (e.g., in the background) to the extent of the user's specified delegation. The autonomy attribute of an agent can range from being able to initiate a nightly backup to negotiating the best price of a product for the user.
- Monitoring: The agent needs to be able to monitor its environment in order to be able to perform tasks autonomously.
- Actuation: The agent needs to be able to affect its environment via an actuation mechanism for autonomous operation.
- Intelligence: The agent needs to be able to interpret the monitored events to make appropriate actuation decisions for autonomous operation.

In addition to the basic attributes mentioned above, an agent may have other attributes such as mobility, security and others.

Regarding the application of software agents, Jennings and Wooldridge note that "agent are being used in an increasingly wide variety of applications, ranging from comparatively small systems such as email filters to large, open, complex, mission critical systems such as air traffic control" [18]. According to the categorization of Jennings and Wooldridge, the application of intelligent agents can be classified into industrial, commercial, medical and entertainment applications.

Agent systems can be further classified either single-agent or multi-agent systems. Agent systems, particularly multi-agent systems, are subfields of DAI (Distributed Artificial Intelligence) research, and have existed under AI for two decades. Generally, DAI is broken into Distributed Problem Solving (DPS) and Multi-Agent Systems.

Research on designing and developing multi-agent systems focuses on the interaction between agents. Topics frequently discussed in this area of research include agent action, the relationship between agents, multi-agent system architecture and the environment in which the multi-agent systems exist; interactions among agents within the multi-agent system, and agent adaptation ability. Coordination, negotiation, cooperation are three common interactions among agents in multi-agent systems. The major difference between multi-agent systems and single-agent systems is in that multi-agent systems include several agents that are aware of one another. Figure 1 illustrates the framework of multi-agent systems.

As agents are being used in variety range of applications, efforts to standardize the implementation of agent systems are continuing. Two major efforts at standardization

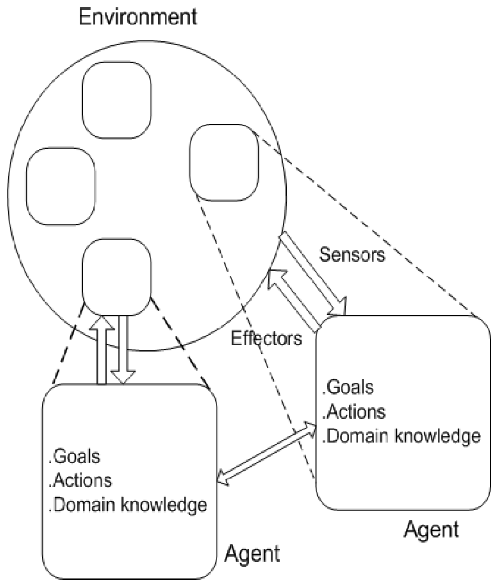


Fig. 1 Multi-agent framework [15].

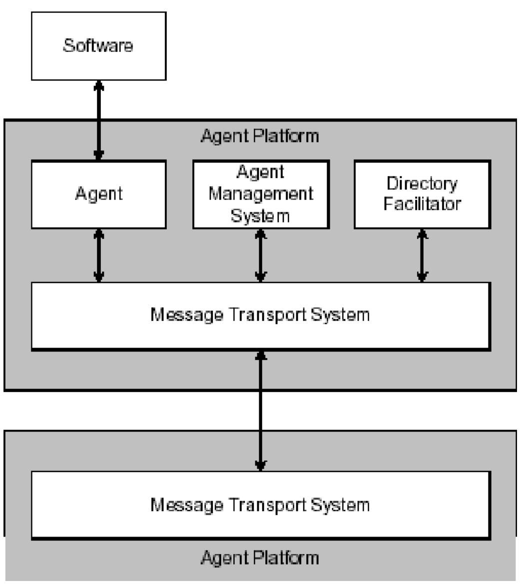


Fig. 2 The FIPA agent platform.

are the Foundation for Intelligent Physical Agents (FIPA) and the Object Management Group (OMG) Agent Working group [15]. As shown in Fig. 2, the FIPA agent platform defines the basic architecture, and provides: the Agent Management System (AMS), which controls the life cycle of the agent; the Directory Facilitator (DF), which resembles a yellow-page lookup service; and the Message Transport System (MTS), which provides inter-agent communications and messaging with other FIPA-complaint agent platforms.

3. Previous Works on Traffic Signal Control

Traffic signal control strategy manipulates three factors, cycle length, split, and offset, to regulate traffic flow. Cycle length refers to the time required for a traffic light to circulate from a particular phase (e.g. a green light) back to the same phase. In reality, cycle time is set to a range somewhere between 40 and 120 seconds, depending on traffic volume. The split is the segment of the cycle length allocated to each phase or interval that may occur. Meanwhile, the offset is the time difference for the same phase between two traffic lights at two intersections. The traffic signal system can be categorized into three classes, fixed time signal, traffic-actuated signal, traffic-adjusted signal. The fixed time signal controls the traffic light using a predefined timetable. In the traffic-actuated signal, the cycles, signal phases, signal intervals of the traffic light are defined via controllers and related devices. The new style traffic control signal system adopts the traffic-adjusted signal, which combines the advantages of both fixed time and traffic-actuated signals. Sensors placed in an artery can forward monitored traffic data to the master controller. The master controller then calculates and periodically distributes an appropriate traffic signal control strategy (represented by the signal cycle, phase and interval) to the signal controller in each individual intersection.

PASSER II [10] and TRANSYT-7F [11] aimed at local optimal control of the traffic signal system. Both studies attempted to identify optimal solutions for local traffic signal control systems at each intersection.

The domain of traffic signal control is well suited to a multi-agent based approach owing to its geographically distributed nature. Agent technology is a computer paradigm that has become a significant area in computer science. Researchers and practitioners have now realized that single-agent systems, multi-agent systems and distributed artificial intelligence are attractive because they consider the social aspects of computer systems, ranging from human-computer interaction over distributed problem solving, to the simulation of social systems [16]. Roozemond and Rogier [4] [proposed a prototype using agent technology to control a traffic signal system. Ferreira et al. [6] also presented a multi-agent decentralized strategy for controlling an urban traffic network. In the approach proposed in [6], each agent optimizes a traffic index based on its local state and sensors, and also on information from adjacent intersections.

4. ACTAM

This section describes the proposed architecture and the components contained inside ACTAM. From Fig. 3, the ACTAM architecture comprises three parts:

- IIA (Intelligent Intersection Agent): In charge of autonomous control] of the traffic lights at an intersection.
- SM (Sensor Module): Gathers real-time traffic volume information for IIA.
- Traffic Light: Regulates traffic volume.

Urban traffic relies heavily on the quick response of the control strategy for the traffic light system in each intersection. The information flow in ACTAM can be described as

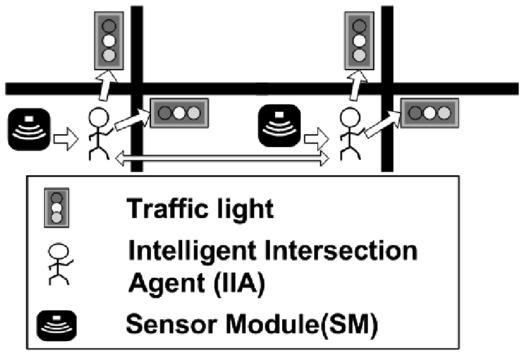


Fig. 3 Overview of ACTAM architecture.

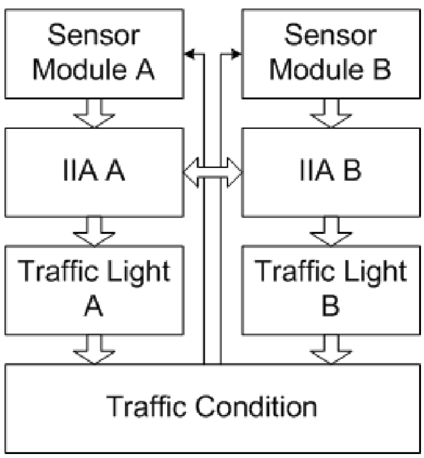


Fig. 4 Information flow of ACTAM.

follows:

- First, the sensor module is used to monitor the traffic volume and forward the traffic volume data to IIA.
- IIA then communicates and exchanges related information with the IIAs at the adjacent intersections.
- After collecting all of the required information, IIA began to generate an appropriate traffic signal control strategy for the specific traffic signal.

The information flow formed a closed loop, as shown in Fig. 4.

IIA gathers the current traffic volume situation through the Sensor Module. The data is kept in the IIA Data Process Module. IIA communicates with other IIAs in the adjacent intersection via the Communication Module. After gathering all of the necessary information, the Learning Module, Forecast Module and Weighted Module generate results according to its purpose (these components are described in detail in the following paragraph). The generated results then are delivered to Control Strategy Decision Module, where the traffic signal control strategy is devised.

Components in IIA can be further divided into three categories as shown in Fig. 5. The functions of each component in IIA are briefly summarized below.

Layer1, the Data Layer, includes the Communication Module and Data process Module. The Communication

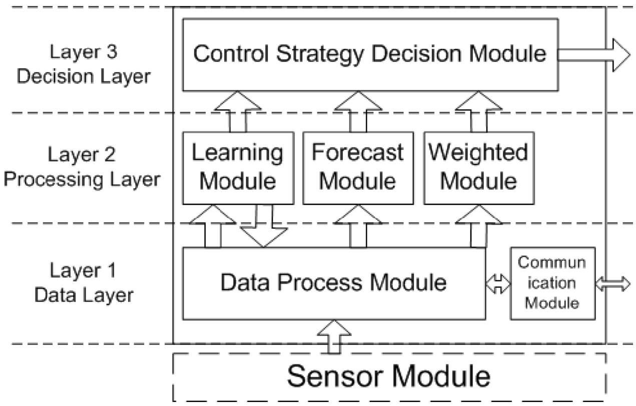


Fig. 5 Components of Intelligent Intersection Agent (IIA).

Module oversees the exchange of related traffic data with other IIAs. Restated, the Communication Module is responsible for agent interaction. The Communication Module exchanges data through KQML (Knowledge Query and Manipulation Language), KQML is a language and protocol for exchanging information and knowledge between two or more intelligent systems in support of cooperative problem solving. The Data Process Module is a database that stores the data involved in the process of IIA. Data stored in the Data Process Module includes both fixed and calculated data. Fixed data comprises the serial number of the adjacent intersections, the distance from the adjacent intersections, and so on. Meanwhile, the calculated data are from the Learning Module, and are further detailed in the paragraph explaining how the Learning Module works.

The next layer, the Processing Layer, deals with the factors that might influence the traffic signal control strategy. Traffic signal control strategy should consider three factors: first, knowledge of past traffic flow data; second, prediction of future vehicle numbers at the intersection; third, importance (judged by traffic volume) of each intersection.

The Forecast Module is designed for forecasting possible traffic volume in a specific intersection. Cooperation by exchanging traffic volume information of adjacent intersections among agents can help in forecasting future traffic volume. Cooperation is essential in Multi-agent systems. There are several points of view on cooperation. This study adopts the perspective that cooperation constitutes one of the fundamental means of resolving conflict. Agent autonomy can create situations where agents have contradictory interests, and conflict can result. Frequently, these conflict situations arise from access to limited resources. In this study, conflict situations result when vehicles pass one another at a crossroads. As mentioned before, cooperation can resolve conflict. Methods of cooperation can be classified into six categories: grouping and multiplication, communication, specialization, collaboration by sharing tasks and resources, coordination of actions, and conflict resolution by arbitration and negotiation [17]. Communication ability further extends the perceptive capacities of agents within multi-agent systems by allowing them to benefit from the information held

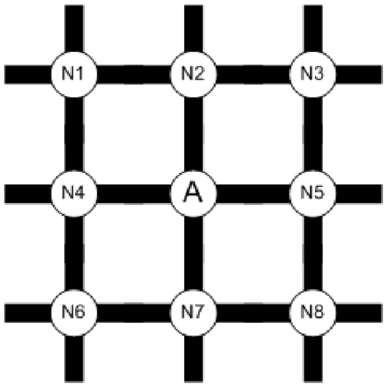


Fig. 6 Traffic network.

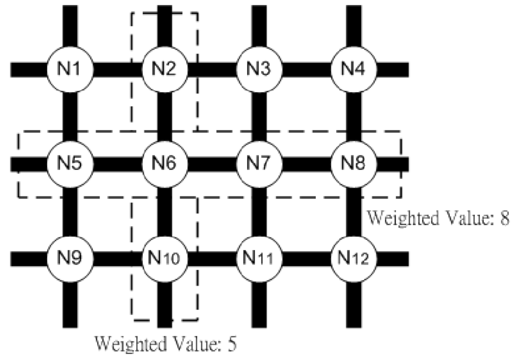


Fig. 7 Weighted Module.

by other agents. In the present architecture, agent communication ability constitutes one method of providing for the coordination of actions. As shown in Fig. 6, traffic volume of intersection A is influenced by vehicles passing through nearby intersections. Moreover, the influence of nearby intersections on intersection A is proportional to the distance between them. That is, the further two intersections are from one another, the less they influence one another.

The forecast traffic volume of each intersection can be expressed by following equations.

$$\begin{aligned} \text{Total}UN_x &= UpN_xN_1 + UpN_xN_2 \ldots + UpN_xN_n \\ &= \sum_{i=1}^{n} UpN_xN_i \qquad (1) \\ \text{Total}DN_x &= DownN_xN_1 + DownN_xN_2 \ldots + DownN_xN_n \\ &= \sum_{i=1}^{n} DownN_xN_i \qquad (2) \\ \text{Total}RN_x &= RightN_xN_1 + RightN_xN_2 \ldots + RightN_xN_n \\ &= \sum_{i=1}^{n} RightN_xN_i \qquad (3) \\ \text{Total}LN_x &= LeftN_xN_1 + LeftN_xN_2 \ldots + LeftN_xN_n \\ &= \sum_{i=1}^{n} LeftN_xN_i \qquad (4) \end{aligned}$$

Where N represents the intersections in the traffic network and $x = 1 \ldots n$.

Take Eq. (1) for example, TotalUN_x is the possible traffic volume north of intersection N_X, and $UpNXN_i$ stands for OR represents the possible influence of intersection N_i on intersection N_X. $UpNXN_i$ is determined by

$$\begin{aligned} UpN_xN_i = &\frac{1}{\text{distance}_\text{U}} \times \text{CarQueue}_\text{U} \\ &+ \frac{1}{\text{distance}_\text{D}} \times \text{CarQueue}_\text{D} \\ &+ \frac{1}{\text{distance}_\text{R}} \times \text{CarQueue}_\text{R} \\ &+ \frac{1}{\text{distance}_\text{L}} \times \text{CarQueue}_\text{L} \qquad (5) \end{aligned}$$

Where CarQueueU represents the number of vehicles queuing in front of the traffic light, and DistanceU is the distance from Nodei to Nodex.

The Learning Module is used to achieve adaptation in the proposed architecture. Two learning strategies exist:
1. Short-term Learning: Determine how recent traffic volume and waiting car queue affect traffic light control strategy.
2. Long-term Learning: Seek to discover patterns or trends in large amounts of historical data (from previous times). By analyzing historical data, the proposed architecture is capable of learning and adjusting its control strategy.

The Weighted Module in layer 2 is used to store the weighting result. Weights are assigned to different intersections with appropriate measurement according to traffic flow at each intersection as shown in Fig. 7.

The Decision Layer includes the Control Strategy Decision Module. The Control Strategy Decision Module summarizes the indices output from the Learning Module, Weighted Module, Forecast Module and alter traffic light control strategy through the modification cycle, split and offset of the traffic light controller.

5. Experiment and Performance Evaluation

The experiment aimed to determine the simulated total delay time of a specific urban traffic network achieved by the proposed decentralized ACTAM architecture, when compared to the conventional fixed sequence traffic signal control strategy. The fixed sequence traffic signal control strategy served as the base line system. In the present experiment, the control strategy minimizes the total delay time. A primary objective in signalized intersection operation is to achieve the required capacity for each flow with minimum delay [8]. In other words, total delay time is used as a criteria for performance evaluation. In the present definition, total delay time is the sum of all car waiting times in front of all traffic lights in a specific traffic network. The total delay time for a certain period of time is given by

$$T = \sum_{i=1}^{n} \sum_{j=1}^{m} W_{ij} \qquad (6)$$

Table 1 Cycle time for traffic light.

Type	Traffic condition	Cycle time
C1	R <1/45	40 seconds
C2	1/45< R <1/40	60 seconds
C3	1/40< R <1/35	80 seconds
C4	1/35< R <1/25	100 seconds
C5	1/25< R <1/20	120 seconds
C6	1/20< R <1/15	140 seconds
C7	1/15< R	160 seconds

Where T denotes the total delay time, n represents the total number of intersections within the traffic network, m indicates the number of vehicles in the network and W_{ij} is the waiting time in front of each traffic light

The parameter settings used in the present simulation are shown below:

- The traffic network used in this simulation contained 30 intersections, arranged as a 6×5 matrix.
- The distance between every intersection is 500 meters.
- Vehicle speed is set to 50 kilometers per hour
- The simulation time is set to 120 minutes
- The simulation is performed on an Intel Pentium IV PC.
- Two different car flows are implemented in the experiment to represent different traffic volumes; one vehicle entering the network every 10 seconds, and one vehicle entering the network every 2 seconds, to represent normal traffic volume and congestion traffic volume, respectively. The probabilities of a vehicle continuing straight ahead, turning left, and turning right are set to 50%, 25%, and 25%, respectively.

For the fixed sequence traffic signal control strategy, the traffic light cycle and split are predefined. In the present simulation, the cycle is set to 100 seconds per cycle, split for horizontal and vertical is set to 1 : 1. As for ACTAM, the Control Strategy Decision Module is in charge of altering the cycle and split to reduce total delay time. Traffic light cycle time should increase with crowding of traffic volume. In the present experiment, eight different types of cycle time are defined depending on level of traffic congestion. The split should be set based on the ratio of vertical and horizontal traffic volume. The tables below illustrate how cycle and split are determined in ACTAM.

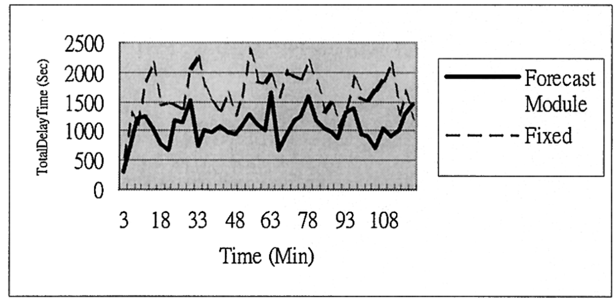
Where R can be expressed using the following equation

$$R_i = \frac{\text{Predicted upcoming traffic flow of intersection } i}{\text{Sun of predicted upcoming traffic flow for every intersection}} \tag{7}$$

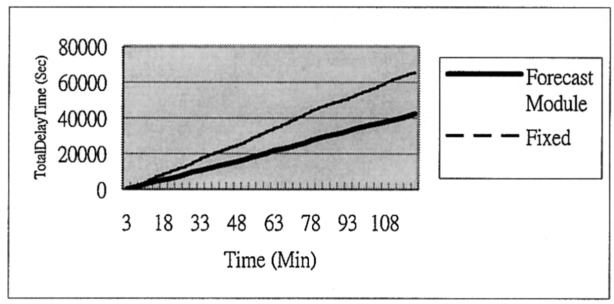
Traffic congestion is measured using R for each intersection. Table 1 listed how R is set in our simulation. Since the experiment setting involves 30 intersections, each intersection should partake of 1/30 traffic volume. Therefore, when R is between 1/35 and 1/25, the cycle time is set to 100 seconds, as the same as for the fixed sequence traffic signal control strategy. The traffic light cycle time should increase with crowding of traffic volume, and should decrease as the traffic volume becomes less crowded.

Table 2 Split for traffic light.

Split-Vertical	Cycle * (D1/(D1+D2))
Split-Horizontal	Cycle * (D2/(D1+D2))



(a)



(b)

Fig. 8 Comparison of fix sequence traffic signal control strategy and Forecast Module of ACTAM architecture under the condition of 10 seconds per vehicle enters the traffic network. (a) Total Delay Time, (b) Cumulate Total Delay Time.

As shown in Table 2, where D_1, D_2 indicate the vertical and horizontal traffic volumes, respectively. The split is set in proportion to the ratio of vertical traffic volume/horizontal traffic volume. In this study, the split is limited to 8 : 2 to 2 : 8.

Figures 8 and 9 compare the fixed sequence traffic signal control strategy and ACTAM. Each experiment was conducted over a 2 hour time frame. In Fig. 8, the frequency of vehicle entry to the traffic network was set to six vehicles per minute; that is, a vehicle enters the traffic network every 10 seconds. From Fig. 8, the performance of two different control strategies converged to two distinct levels after 9 minutes of simulation. Further investigation of the simulation data revealed an approximately 33.47% reduction in total delay time compare to fixed sequence traffic signal control strategy. The improvement rate can be obtained by

$$\text{Improvement} = \frac{\text{Fix TDT} - \text{ACTAM TDT}}{\text{Fix TDT}} \times 100 \tag{8}$$

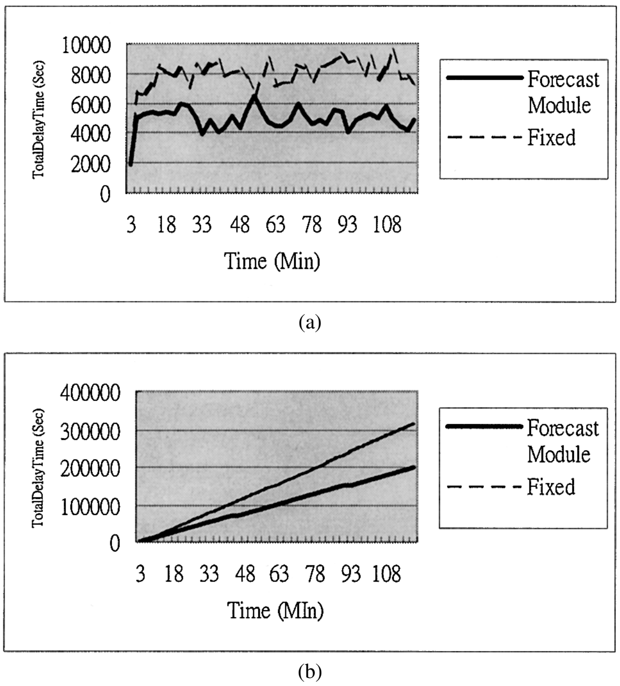


Fig. 9 Comparison of fix sequence traffic signal control strategy and Forecast Module of ACTAM architecture under the condition of 2 seconds per vehicle enters the traffic network. (a) Total Delay Time, (b) Cumulate Total Delay Time.

Where Fix TDT denotes the total delay time obtained by the fixed sequence traffic signal control strategy, and ACTAM TDT indicates the total delay time obtained by ACTAM.

In Fig. 9, the frequency with which vehicles enter the traffic network is set to 30 vehicles per minute. The use of a higher vehicle entrance frequency is designed to examine the influence of level of congestion on ACTAM. From Fig. 9, a similar result to Fig. 8 is obtained when the frequency is set to six vehicles per minute. The performance of the two different control strategies stabilized at two distinct levels following 6 minutes of simulation. The condition of 1 vehicle entering the network every 2 seconds achieved an approximately 36.96% reduction in] total delay time.

6. Conclusion

Urban traffic congestion is becoming a serious problem. The proposed ACTAM architecture utilizes multi-agent systems for decentralized traffic signal control. Adopting multi-agent systems to solve the problem of traffic congestion] can achieve advantages such as robustness and scalability. ACTAM enhances flexibility and extensibility owing to its modular design. Modular design means that the module within the architecture can be substituted using better and newer modules when necessary, thus ensuring the flexibility and extensibility of ACTAM. The Sensor Module in the proposed architecture collects traffic data, while the data layer in IIA receives data and exchanges it with other IIA in the network. The processing layer is used to generate indices representing possible influences on] the control strategy for the decision layer. The decision layer devises a control strategy based on the indices provided by the processing layer and alters cycle, split, offset for a traffic light. The present simulation implemented the Forecast Module, by forecasting possible upcoming traffic volume and arranging cooperation among agents by exchanging traffic volume information; the regulator (traffic light) can follow the decision made by IIA and reduce total delay time.

The simulation results demonstrate that ACTAM out performed conventional fixed sequence traffic signal control strategy by 37%. ACTAM aimed to minimize total delay time in specific traffic networks with decentralized control strategy decision-making. A decentralized control strategy achieves improved scalability by adopting a multi-agent system, since each IIA oversees the local optimization problem and no master controller is required. Thus, adding new intersections to the network only increases the computational loading on neighboring intersections, while that in the remainder of the network remains the same. Additionally, a decentralized control strategy enables the controller to react to incidents rapidly and proactively. Since this study attempts to minimize total delay time, the communication ability of IIAs avoids the pursuit of local optimization, and thus achieves the goal of minimizing total delay time.

References

[1] S.Y. Lin, D.K. Chen, and R.S. Chen, "Apply adaptive and cooperative MAS to urban traffic signal control," Proc. 11th Mediterranean Conference on Control and Automation (MED'03), IEEE Control System Society Press, Rhodes, Greece, 2003.
[2] N. Fujimoto, M. Moriya, A. Ishizuka, and M. Goto, "Intra- and inter-vehicle communication network using low-cost POF links," IEICE Trans. Inf. & Syst., vol.E85-D, no.11, pp.1839–1850, Nov. 2002.
[3] W.M. Wey, "A study on an urban network traffic signal control," Transportation Planning Journal, vol.29, no.4, pp.693–708, Dec. 2000.
[4] D.A. Roozemond and J.L.H. Rogier, "Agent control traffic lights," ESIT2000, Aachen, Germany, Sept. 2000.
[5] L. Andersson and Å. Rönnbom, Intelligent Agents - A New Technology for Future Distributed Sensor Systems?, Department of Informatics School of Economies and Commercial Law Göteborg University Master Thesis in Informatics, IA7400, Spring 1999.
[6] E.D. Ferreira, E. Subrahmanian, and D. Manstetten, "Intelligent agents in decentralized traffic control," Proc. IEEE Intelligent Transportation System Conference, pp.25–29, Aug. 2001.
[7] S.J. Russell and P. Norvig, Artificial intelligence/a modern approach, Prentice Hall International, Englewood Cliffs, NJ, 1995.
[8] T. Matson, W. Smith, and F. Hurd, Traffic Engineering, McGraw-Hill, 1971.
[9] H. Löthman, Intelligent Agent Platform in Distributed Software Architecture, Bachelors thesis, Oulu Polytechnic, Raahe, Institute of Computer Engineering, 2001.
[10] E.C.P. Chang, J.C. Lei, and C.J. Messer, "Arterial signal timing optimization using PASSER II-87 – microcomputer user's guide," Report TTI-2-18-86-467-1, Texas Transportation Institute, Texas A&M University System, College Station, Texas, 1988.
[11] C.E. Wallace, et al., "TRANSYT-7F user's manual," Transportation Research Center, University of Florida, Gainesville, Florida, 1988.
[12] D.C. Smith, A. Cypher, and J. Spohrer, "KidSim: Programming agents without a programming language," Communications of the ACM, vol.37, no.7, pp.55–67, 1994.

[13] B. Hayes-Roth, "An architecture for adaptive intelligent systems," Artificial Intelligence: Special Issue on Agents and Interactivity, vol.72, pp.329–365, 1995.
[14] D.B. Lange and O. Mitsura, Programming, and Deploying JAVA Mobile Agents with Aglets, Addison-Wesley, 1998.
[15] J.P. Bigus and J. Bigus, Constructing Intelligent Agents Using Java^{TM}, Second ed., John Wiley & Sons, New York, 2001.
[16] R. Schleiffer, "Intelligent agents in traffic and transportation," Transportation Research Part C: Emerging Technologies, vol.10, no.5–6, pp.325–329, Oct.–Dec. 2002.
[17] J. Ferber, Multi-Agent Systems: An introduction to distributed artificial intelligence, Addison-Wesley, 1999.
[18] N.R. Jennings and M.J. Wooldridge, "Applications of intelligent agents," in Agent Technologies: Foundations, Applications, and Markets, Springer-Verlag Telos, 1998.

Ruey-Shun Chen received M.S and Ph.D. degrees in Computer science and Engineering form National Chiao-Tung University in 1989 and 1995, research area: Internet Application and Information System. Now, he is a associate professor in Institute of Information Management of National Chiao-Tung University in Taiwan.

Duen-Kai Chen is currently pursuing Ph.D. in Institute of Information Management, National Chiao Tung University in Taiwan. His research interests focus around Intelligent Agents, Soft Computing, Decision Science, and Business Information System.

Szu-Yin Lin is a Ph.D. Student in Computer Science, National Tsing-Hua University. He received his Master degree in Institute of Information Management, National Chiao-Tung University. His professional interests focus around Artificial Intelligence, Software Agents, Internet Technology, Business Information System, and Web Programming.