A Case Study of Evaluating Traffic Signal Control Systems Using Computational Experiments

Fenghua Zhu, Guoxi Li, Zhenjiang Li, Cheng Chen, and Ding Wen

***Abstract*—A new traffic signal control system (TSCS) evaluation method that uses computational experiments based on artificial transportation systems (ATSs) is proposed in this paper. Some basic ideas of the method are discussed, i.e., generating reasonable travel demand, modeling the influence of environment, and designing communication interface. Using a 30-day computational experiment on ATSs, a case study is carried out to evaluate three TSCSs, which are implemented using fixed-time (FT), queue-based responsive (QBR), and adaptive dynamic program (ADP) algorithms, respectively. Aside from normal weather, three types of adverse weather, i.e., rain, wind, and fog, are modeled in the computational experiment. After analyzing aggregate data and detailed operating record, reliable evaluation results are obtained from this case study. Furthermore, several interesting phenomena are observed in this case study, which have yet to be noticed by previous work.**

***Index Terms*—Artificial transportation systems (ATSs), communication interface, computational experiment, evaluation, traffic signal control system (TSCS).**

I. Introduction

THE traffic signal control system (TSCS) has been widely applied in the real world, and it plays an important part in traffic management, particularly in large cities. However, there is still no effective way to evaluate its performance and reliability. The main difficulty lies on the ability to reproduce an authentic transportation environment within the laboratory, as real-world traffic scenarios are both too huge and too complex to be described by traditional simulation methods.

Currently, the evaluations of TSCS are mainly carried out through field operational tests (FoTs). Moor *et al.* evaluated the Split Cycle Offset Optimization Technique (SCOOT) in Anaheim, CA, through FoTs from 1994 to 1998 [1]. The evaluation was carried out in 12 scenes, by comparing the traffic flows before and after SCOOT was installed. In addition to evaluating in peak and off-peak periods, the experiments before and after a National Hockey League hockey game were also conducted to assess SCOOT's capacity to adapt to a sudden change of traffic flow. The most famous evaluation of TSCS was the FoTs of Traffic-responsive Urban Control (TUC) performed by Kosmatopoulos *et al.* in Europe [2]. The evaluation was carried out in three traffic networks with quite different traffic and control infrastructure characteristics: Chania, Greece (23 intersections); Southampton, U.K. (53 intersections); and Munich, Germany (25 intersections), where TUC was compared with the respective resident TSCS TASS, SCOOT, and BALANCE. The main disadvantage of FoTs is that they can only be carried out after the system is deployed in the field. This means that problems cannot be identified before the installation; therefore, it is inevitable that resources are wasted.

Traffic simulation has been considered as one significant innovation in transportation research and development [3]–[6]. Theoretically, traffic simulation software can be used in the evaluation much more widely than FoTs. However, traffic simulation software still faces many challenges, which cause its applications to be very limited in this area. One challenge is generating individual travel demand for each person in the simulation [7]–[9]. Most traffic simulation software use aggregating methods and require historical origin-destination (OD) data as input. Not only is it very costly to collect OD data in a wide area, but it is also very difficult—if not impossible—to transfer OD data to individual travel plans. Another challenge is modeling transportation systems in various scenarios. The urban congestion problem is increasingly becoming a major issue in social, economic, and environmental concerns around the world, from developed countries to emerging new powers. All these make the top–down reductionism method of traditional simulation very ineffective in building up transportation scenarios [10]. Furthermore, there is still no standard communication interface between the TSCS and simulation software. Although much research has been conducted in this area, their applications are very limited. The main reason is that most works are based on a private communication protocol and cannot be reused by others.

Artificial transportation systems (ATSs), which are based on concepts and methods in artificial societies, upgrade traffic simulation to a higher level and a wider perspective [11]–[15]. The main idea of ATSs is to obtain a deeper insight of vehicle movement and traffic evolution by extracting the basic rules of individual vehicles and local traffic behavior, as well as observing and analyzing the complex phenomena that emerge from interaction between individuals. Furthermore, ATSs grow artificial systems to "substitute" for real traffic systems for use in the laboratory. ATSs aim to explore feasible approaches

Manuscript received April 16, 2010; revised December 21, 2010; accepted April 23, 2011. Date of publication July 22, 2011; date of current version December 5, 2011. This work was supported in part by the National Natural Science Foundation of China under Projects 60921061, 70890084, 90920305, 90924302, 60904057, and 60974065. The Associate Editor for this paper was L. Li.

F. Zhu, Z. Li, and C. Chen are with the State Key Laboratory for Intelligent Control and Management of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China (e-mail: fenghua.zhu@ia.ac.cn; zhenjiang.li@ia.ac.cn; cheng.chen@ia.ac.cn).

G. Li and D. Wen are with the National University of Defense Technology, Changsha, Hunan 410073, China (e-mail: lgx2020@sina.com; wending2010@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2011.2157691

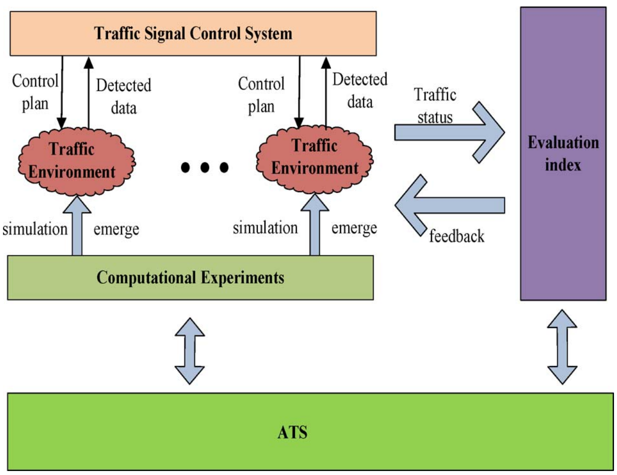


Fig. 1. Evaluating TSCS through experimenting based on an ATS.

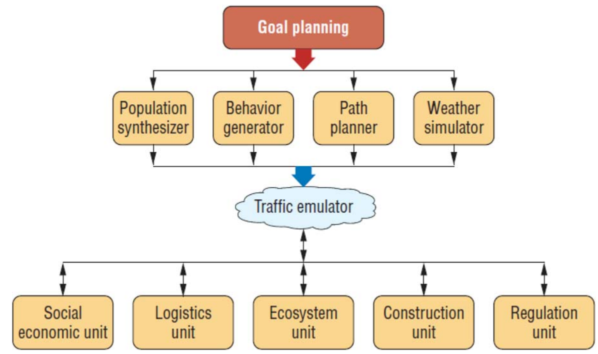


Fig. 2. Designing computational experiments based on ATSs.

to reproduce traffic environments in the laboratory; thus, they provide us with a new way to evaluate the TSCS.

The emphasis of this paper is to introduce our works in carrying out one case study of evaluating TSCS using computational experiments on one specific ATS. The rest of this paper is organized as follows: Section II first gives a brief introduction of the evaluation method. Then, Section III describes the communication interface between ATS and TSCS. More details about the evaluation method are discussed in Section IV, where one specific evaluation platform is built on ATS. Computational experiments are carried out in Section V, and the results are analyzed. Conclusions and future work are discussed in Section VI.

II. Evaluation Method

Essentially, the proposed evaluation method consists of two steps: First, construct one specific ATS and then evaluate one TSCS by directly applying to the virtual traffic environment provided by the ATS, as shown in Fig. 1. Using this evaluation method, we get more comprehensive and reliable results because we are able to design evaluation experiments under various traffic scenarios and, if necessary, repeat the experiments time after time, which is very hard—if not impossible—to perform using traditional methods.

This evaluation method differs from others in two main aspects [16]–[18]. First, travel demands in ATSs are generated from an individual's activity plan, instead of OD data. After constructing activity plans for each member of a population, travel demand can be derived from the fact that consecutive activities at different locations need to be connected by travel. This activity-based travel demand generation method fits well into the paradigm of multiagent simulation and provides us with a feasible approach to generate individual travel demand. Second, ATSs use the simple-is-consistent method to model transportation systems in various environments. As simple-is-consistent is one basic idea in the evaluation process, we will explain it more here.

To obtain reliable result, the evaluation needs to be carried out in various scenarios. In addition to traffic subsystems, the scenarios will also cover the social and economic aspects, as transportation is tightly connected with the environment. From microactivities, such as an individual's psychology and driving behavior, to microphenomena, such as travel gross and travel distribution, all are influenced by environment. The mechanisms by which the environment influences traffic status are very complex, and there are still many disputes about how to model the influences using the top–down reductionism method [19], [20]. However, for simple objects, such as individual vehicle and local traffic behavior, most of the current conclusions about the influences that they receive from the environment are consentaneous. Thus, if simple microobjects and local behavior are modeled using these widely approved conclusions, the macrocomplex phenomena that emerged are also expected to be understandable and agreeable. The idea for modeling transportation system can be abstracted as simple-is-consistent and has been widely used in designing computational experiments based on ATSs (see Fig. 2). It has been proved that, using this idea, ATSs cannot only model direct traffic-related activities but generate their traffic processes from various indirect facilities and activities as well, such as weather, legal, and social involvements [11]–[14].

For example, it is hard—if not impossible—to generate transportation scenarios in adverse weather from a macro perspective. However, for microobjects, the influences that they received in adverse weather can be easily measured. For example, the influences of adverse weather on one person can be modeled in two aspects. One aspect is about his experience and behavior in traveling. As road conditions and visibility get worse, the performance of vehicles heavily decreases, and drivers feel depressed. There are more traffic violations, and the chance of accidents increases as a consequence. In another aspect, the individual will adjust his travel plans according to the adverse weather. For example, the individual will try to reduce unnecessary travel by canceling or postponing shopping and entertainment activities. Artificial intelligence methods and computational-intelligence algorithms are effective tools in modeling one's decision process during adverse weather. We have explored this area for a long time, and many exiting results have been achieved. For example, we have proved that the mechanisms by which individuals adjust their activities can be expressed by simple rules.

Artificial transportation scenarios during adverse weather have been established in our case study, and their effectiveness in evaluating TSCS has been verified. More details will

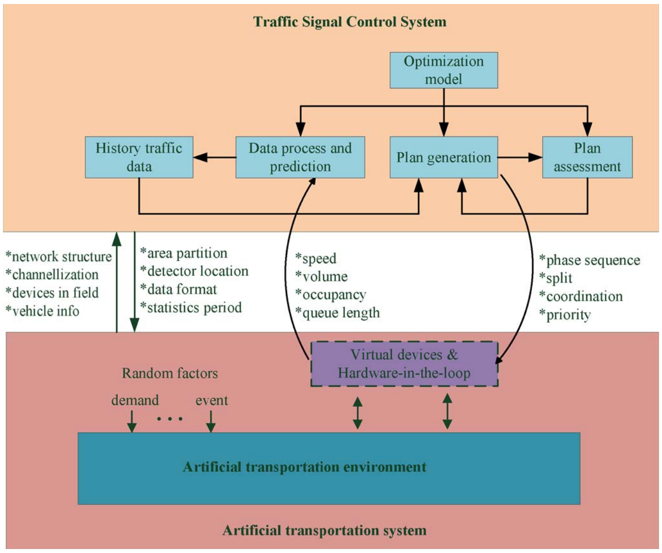


Fig. 3. Communications between TSCS and ATS.

be shown in the case study section of this paper. It should be pointed out that, although current simulation software, including TSIS and the PARAllel MICroscopic Traffic Simulator (PARAMICS), can provide interfaces for inputting adverse weather conditions, their considerations of adverse weather are limited in the moving process of vehicles, such as decreasing speed and acceleration, i.e., the adjustments of an individual's travel demand in adverse weather are completely neglected.

III. Communication Interface

To guarantee the generality of the evaluation method, we also designed an open interface for the communication between TSCS and ATS. There are many TSCSs that are quite different in their implementations, particularly the optimization models and the execution platforms. The well-designed interface hides the implementation details of TSCSs and forms the basis for establishing a standard evaluation platform in the future. Here, we will briefly analyze the working procedure of one TSCS and introduce the main ideas of the interface.

The working procedure of the TSCS can be represented by Fig. 3. TSCS computes the control plan using the optimization model and sends it to traffic signal controllers, which are installed at the intersections and regulate traffic by controlling traffic lamps. At the same time, traffic signal controllers collect traffic information using various sensors and send them to the TSCS through communication network. When the TSCS receives the uploaded real-time information, it will continue to optimize the control plan using historical data (stored in database), current data (collected in real time), and future data (predicted by itself). These steps form a closed loop, and there are mainly two types of factors that can influence its operating status. One type includes the uncontrollable random factors, which are described by experimental conditions, such as accidents and traffic demands. The other type is composed of the control plans of TSCS. To keep the fluctuations of traffic status in a controllable range, TSCS optimizes control

TABLE I
Some Functions of TSCS

Function	Input	Output
Cycle adjusting to decrease the maximum saturation level	✓ Speed ✓ Occupancy ✓ Volume	✓ Cycle time ✓ Splits ✓ Offset
Phase sequence adjusting for quicker servicing of emergency vehicles and reducing stop and delay	✓ Volume ✓ Queue length ✓ Preempt signal	✓ Phase sequence ✓ Transition mode
Phase time adjusting to minimization of risk of over saturation and queue spill back	✓ Queue length ✓ Link capacity	✓ Extend current phase or not ✓ The time of next phase
Provide priority to public transport vehicles (PTV)	✓ PTV priority ✓ PTV schedule ✓ Current signal status	✓ Green extension ✓ Red omit ✓ New phase

TABLE II
Weather Conditions in Computational Experiment

Weather	Type	Level	Lasting time
W_0	normal	-	-
W_1	Rainy	1 (Precipitation=5mm)	5:00-8:00, 18:00-20:00
W_2	Rainy	2 (Precipitation=15mm)	5:00-8:00, 18:00-20:00
W_3	Rainy	3 (Precipitation=40mm)	5:00-8:00, 18:00-20:00
W_4	Windy	1 (Force=2m/s)	5:00-24:00
W_5	Windy	2 (Force=6m/s)	5:00-24:00
W_6	Windy	3(Force=10m/s)	5:00-24:00
W_7	Foggy	1 (visibility is 300m)	6:00-10:00
W_8	Foggy	2 (visibility is 200m)	6:00-10:00
W_9	Foggy	3 (visibility is 100m)	6:00-10:00

plans continuously and, whenever necessary, sends the results to signal controllers to update the current operating parameters.

Functional decomposition is the first step in designing universal interfaces. Many optimization models have been proposed by different vendors. Examples of the models include the hierarchical model in SCOOT, REALBANDS in The Real-Time Hierarchical Optimized Distributed Effective System (ROHDES), rolling horizon in Optimization Policies for Adaptive Control (OPAC), etc. Although the internals of these models are very different, they all have similar input/output parameters and can be divided into small standard functions. Some examples of the functions are listed in Table I.

Communication protocol is another problem in designing the interface. Web services are preferred over other middleware technologies for reasons of interoperability and portability. We have defined the web service interfaces for all the functions in Table II using the Simple Object Access Protocol (SOAP) [21] and integrated them into ATSs. Based on the communication interface, distributed evaluation architecture has been implemented. Generally, the architecture is composed of three parts, i.e., ATS (the service requestor), TSCS (the service provider), and the service registry. TSCS advertises its services in a service registry. ATS finds a suitable service from the service registry and subsequently interacts with the associated TSCS. This architecture brings us great convenience in evaluation as TSCSs that have been developed on any platform and deployed anywhere on the Internet can be easily accessed.

Fig. 4. Zhongguancun area in Beijing, China.

Using the universal communication interface, real TSCS, instead of virtual systems, can be integrated during the evaluation process. It can not only extend the application scope but can also improve the reliability of the evaluation result. Although many models that include emulation for actuated signals and basic signal coordination have been implemented, some types of advanced signal control may be difficult or impossible to implement. Furthermore, the problems that could not be foreseen during the design of the system but would otherwise become evident during the first field implementation cannot be identified using only virtual systems [22], [23]. All these can be resolved by evaluating the real system. The process is also called hardware-in-the-loop evaluation, which can help us bridge the gap between the evaluated system and the real system.

IV. Evaluation Environment

To verify the practicability and effectiveness of this evaluation method, we have built one evaluation platform in our laboratory. The platform is based on one specific ATS that is established to model the transportation system of the Zhongguanchun area in Beijing, China (see Fig. 4). The activity places that were modeled in this ATS include 40 residential areas, 74 office buildings, 47 restaurants, 12 schools, seven hospitals, 27 shopping centers, 11 leisure centers, and eight sports centers. The road network in this area, which is composed of four expressways, three arterials, and two secondary arterials, is represented using 97 nodes and 224 links. According to the census statistics of this area, more than 200 000 individuals are modeled in this specific ATS. As we described before, while plenty of simple individuals interact with each other in this platform, macrocomplex phenomena can naturally emerge.

In this environment, we have evaluated three TSCSs, which are implemented using three control algorithms, respectively.

FT Algorithm: Fixed-time (FT) control, which is the most well-known algorithm, is usually used as a reference model in the evaluation. In our implementation, one day is first divided into several intervals, among which 6:30–9:00 and 17:00–19:00 are peak periods, and the other intervals are off-peak periods. Then, cycle time and splits, which are kept fixed in each interval, can be computed using historical traffic data according to the following two equations:

$$c = \frac{1.5L + 5}{1 - \sum_i y_i} \tag{1}$$

$$g_j = \frac{y_j}{\sum_i y_i}(c - L) \tag{2}$$

where $y_i = q_{ci}/S_i$ is the maximum ratio of volume to capacity in phase i, L is the lost time in one cycle, c is the cycle time, and g_j is the effective green time of phase j.

QBR Algorithm: Queue-based responsive (QBR) and the flowing adaptive dynamic program (ADP) are all hierarchical algorithms, which are implemented using a multilevel structure. For the sake of clarity, we will use a two-level structure as an example. The control center software locates at the top level and optimizes the cycle times of all the controllers in the network according to some indices (such as the degree of saturation). Traffic controllers are located at the bottom level and adjust the phase times responding to real-time traffic flow.

Consider an intersection with m incoming directions and a Φ-phase light. The controller finds the optimal solution to the problem characterized by the following cost function:

$$\min \int_{\tau_0}^{\tau_0+T} \left\{ \sum_{j\in\{1,\ldots,m\}} \max_{k\in\phi_j} q_{jk}^2(\tau) + \sum_{i\in\{1,\ldots,m\}} \sum_{j\in\{1,\ldots,m\}, j\neq i} \left[\max_{k\in\phi_j} q_{jk}(\tau) - \max_{k\in\phi_j} q_{ik}(\tau) \right]^2 \right\} d\tau \tag{3}$$

where the first term represents the number of vehicles waiting in the queues to be minimized, whereas the second term forces an equalization among all the queue lengths.

ADP Algorithm: ADP is one of the most advanced traffic control algorithms and provides a feasible and effective way to achieve optimal control performance based on traditional or intelligent control methods. It combines the theories of dynamic programming and neural networks, trying to solve the curse of dimensionality in dynamic programming problems using the approximating characteristic of neural networks [24], [25].

Fig. 5 shows a schematic diagram for implementations of ADP. The inputs to the action network are traffic states $x(t)$, which are the queue length in our implementation. The action network outputs a control variable $u(t)$, such as stopping or extending current phases. The inputs to the critic network are $x(t)$ and $u(t)$. ADP uses the critic network's output to estimate the discounted cost-to-go $J(t)$. During the training process, ADP tries to decrease the critic network's training errors to zero, so that the critic network will accurately evaluate the action network's optimal traffic control performance. To achieve better convergence of the neural networks, γ $(0 < \gamma < 1)$, which is a discount factor for infinite-horizon problems, and $r(t)$, which is a reward or reinforcement signal for $u(t)$, must be carefully selected.

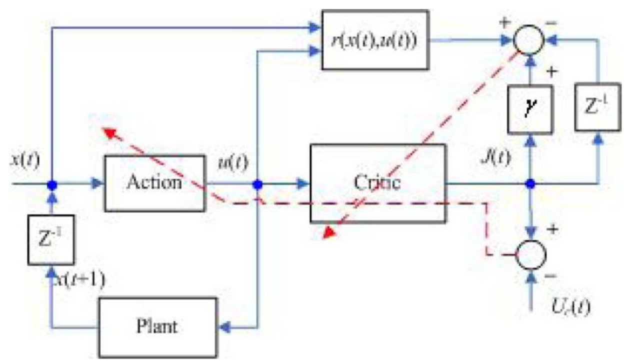


Fig. 5. Schematic diagram for implementations of ADP.

TABLE III
DISTRIBUTION OF WEATHER CONDITIONS IN RANDOM DAYS

D	1	2	3	4	5	6	7	8	9	10
W	W_0	W_2	W_1	W_6	W_9	W_5	W_0	W_9	W_5	W_4
D	11	12	13	14	15	16	17	18	19	20
W	W_8	W_7	W_1	W_3	W_8	W_8	W_6	W_4	W_6	W_3
D	21	22	23	24	25	26	27	28	29	30
W	W_2	W_1	W_0	W_2	W_4	W_5	W_3	W_7	W_9	W_7

V. COMPUTATIONAL EXPERIMENTS AND RESULT

In this case study, a 30-day computational experiment is designed on ATS to model different weather conditions. In addition to normal condition, three types of adverse weather, i.e., rain, wind, and fog, are implemented. Every adverse weather condition is further divided into three levels, which are light (Level 1), medium (level 2), and heavy (level 3). All these weather levels are very typical in Beijing, and each of them is implemented in a computational experiment using one representative quantitative measurement, as shown in Table II. As mentioned in Section II, these weather conditions do not directly affect traffic flow, speed, and traffic demand. Instead, they influence the individual's behavior in two ways. Traffic flow, speed, and traffic demand are the macrophenomena that have emerged naturally.

In the process of designing computational experiments, many principles of traditional experiment were adopted to facilitate the analysis of experimental result. For example, each weather experiment is repeated three times, as replication is one effective method to alleviate the influence of random errors. All weather are distributed in 30 days using randomization (see Table III), which is another basic principle of experiment design.

Three algorithms that are introduced in Section IV are evaluated using the 30-day computational experiment on the ATS. All these experiments are carried out on a normal personal computer, which is equipped with Intel Core 2 4400 at 2.00 GHz and second-generation double-data-rate memory. Because the ATS simulated the detailed traveling process of each individual in computational experiments, extensive evaluation indices can be generated. In reality, the average speed is the most important indicator to represent traffic congestion status and is widely used in urban traffic control and management. We will also use it here to demonstrate the analysis process.

The result is shown in Fig. 6, where each point represents one day's result. In this figure, the x and y coordinates are the

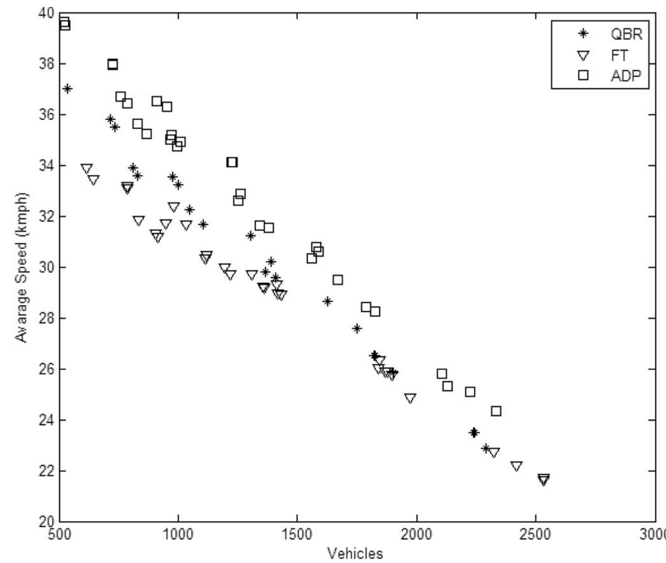


Fig. 6. Average speed versus average vehicles in the network.

TABLE IV
ANALYSIS OF VARIANCE (ANOVA)

Source	SS	DF	MS	F	Prob>F
Algorithms	263.70	2	131.85	8.03	6.32e-04
Error	1429.00	87	16.425		
Total	1692.70	89			

average vehicle number in the network and the average speed of all vehicles in one day, respectively. We can see from this figure that the average vehicle number and average speed are quite different in different days, i.e., traffic conditions dramatically change under various weather conditions. No matter how traffic conditions vary, the performance of ADP always remains better than FT and QBR. However, if we only consider QBR and FT, their performances are very close, and the differences can be neglected when the number of vehicles is more than 1500.

Table IV presents the analysis of variance of the data in Fig. 6. We can see from this table that the differences of algorithms are statistically significant, i.e., we can reject the hypothesis $H_0 : AV_{\mathrm{FT}} = AV_{\mathrm{QBR}} = AV_{\mathrm{ADP}}$ at a confidence level of 0.99. Based on Table IV, we can further compare the performances of any two algorithms using interval estimations. At a confidence level of 0.95, the estimation results are

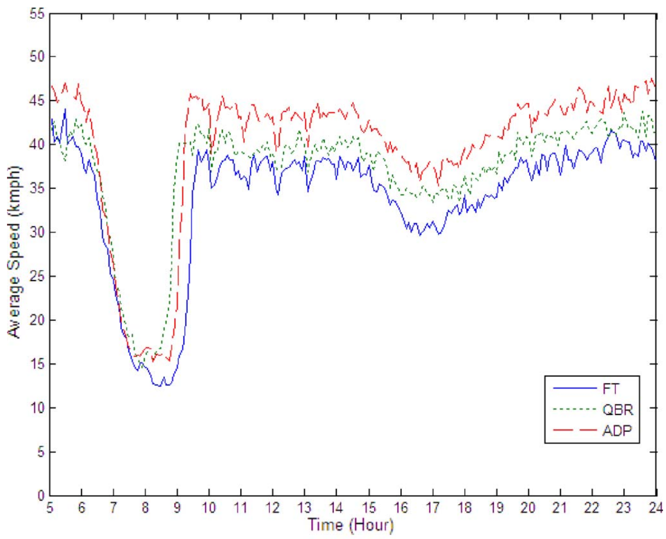
$$AV_{\mathrm{QBR}} - AV_{\mathrm{FT}} : 1.534 \pm 1.026$$

$$AV_{\mathrm{ADP}} - AV_{\mathrm{QBR}} : 2.612 \pm 1.026$$

$$AV_{\mathrm{ADP}} - AV_{\mathrm{FT}} : 4.146 \pm 1.026.$$

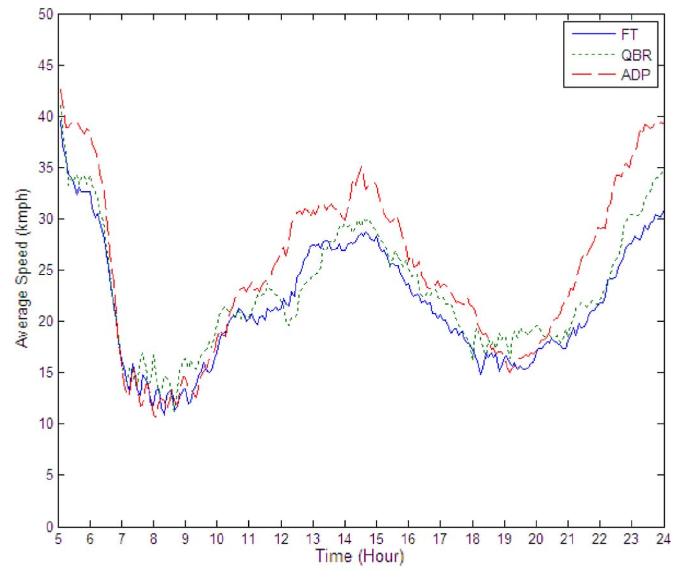
Because the second and third intervals do not contain zero and are all on the right side of zero, the performance of ADP is better than QBR and FT, which also confirms the conclusions drawn from Fig. 6.

In addition to generating aggregate data for statistical analysis, the ATS simulated the whole traveling process of each individual in computational experiments. Fig. 7 shows the operating record of the 23rd day, which is in normal weather and the best case for the ADP algorithm. From this figure, we



	Min	Max	Mean	Std
FT	12.43	44.09	33.07	7.62
QBR	14.64	43.78	37.01	6.70
ADP	15.28	47.51	39.84	8.40

Fig. 7. Operating record of the 23th day (best case).



	Min	Max	Mean	Std
FT	10.92	39.53	21.63	5.95
QBR	10.94	41	22.88	5.88
ADP	10.88	42.83	25.1	8.91

Fig. 8. Operating record of the 14th day (worst case).

can see that, although the performance of ADP is superior to QBR and FT most of the time, there are also some intervals where ADP is not the best. One interval is around 9 A.M., which is the end of morning peak, when traffic conditions change from saturation to free flow. This phenomenon is caused by the logging character of ADP. Because ADP needs some time to learn new parameters, the adjustment of its parameters may log behind real-time traffic status when traffic conditions are changing sharply. Some statistics of the samples on this day are listed at the bottom of Fig. 7, and they give us quantitative comparisons of the three algorithm's performances on this day. For example, compared with QBR and FT, the mean of average speed on this day when using the ADP algorithm is improved by 7.6% and 12%, respectively. It should be noted that the sample standard deviation of ADP is biggest among the three algorithms, which means that there are more fluctuations in traffic flow when ADP is deployed.

Fig. 8 shows another operating record of the 14th day, which is in heavy rain and is the worst case for the ADP algorithm. Compared with normal weather, the traffic condition significantly deteriorates on this day, particularly during peak hours. This phenomenon is consistent with our experience that heavy rain during peak hours usually causes serious congestion. The performance of ADP is still better than the other two algorithms most of the time, but their differences are not very obvious, and there are more intervals where ADP's performance is not the best. It seems ADP does not work very well when serious congestion occurs. To the authors' knowledge, there is no report about this phenomenon, which may be one topic for our future research. Statistics of the samples are also listed at the bottom of Fig. 8. Although ADP still has the biggest mean value among three algorithms, it may not be the best choice as its standard deviation (std) is much greater than that of the other two algorithms.

VI. Conclusion

The main difficulty of evaluating TSCS lies in the ability to reproduce an authentic transportation environment within the laboratory, as traffic scenarios are both too huge and too complex to be described by traditional simulation methods. ATS aims to explore feasible approaches to reproducing traffic environments in the laboratory; thus, it provides us with a new road to evaluate TSCS. In addition to discussing some basic ideas of evaluating TSCS using computational experiments based on ATS, we have introduced the evaluation's platform and illustrated the evaluation method using a case study. Three algorithms, i.e., FT, QBR, and ADP, are evaluated in this case study using one 30-day computational experiment. In addition to normal weather, three types of adverse weather, i.e., rain, wind, and fog, are modeled in random days. After analyzing aggregate data and the detailed operating record, which are both generated by computational experiment, reliable evaluation results are obtained from this case study. Furthermore, several interesting phenomena are observed in the evaluation, which have not been noticed by others' work.

The work presented in this paper is the first step in our plan to set up the evaluation theory and method using computational experiments based on ATS. Future work involves designing and carrying out abundant computational experiments in various environments, such as other adverse weather conditions and economic activity conditions, as well as setting up the evaluation index system for TSCS by combining tradeoff comprehensive metrics.

Acknowledgment

The authors would like to thank Prof. F.-Y. Wang for his advice and great help in designing the evaluation method and carrying out the experiments.

REFERENCES

[1] J. E. Moore, R. Jayakrishnan, M. G. McNally, and C. A. MacCarley, "Evaluation of the Anaheim advanced traffic control system field operational test: Introduction and task A: Evaluation of SCOOT performance," Fed. Hwy. Admin., Washington, DC, Calif. PATH Res. Rep. UCB-ITS-PRR-99-27, 1999.

[2] E. Kosmatopoulos, M. Papageorgiou, C. Bielefeldt, V. Dinopoulou, R. Morrisc, J. Mueck, A. Richards, and F. Weichenmeier, "International comparative field evaluation of a traffic-responsive signal control strategy in three cities," *Transp. Res. Part A: Policy Pract.*, vol. 40, no. 5, pp. 399–413, Jun. 2006.

[3] P. Mirchandani and L. Head, "A real-time traffic signal control system: Architecture, algorithms, and analysis," *Transp. Res. Part C, Emerging Technol.*, vol. 9, no. 6, pp. 415–432, Dec. 2001.

[4] H. Zhou, "Evaluation of route diversion strategies using computer simulation," *J. Transp. Syst. Eng. Inf. Technol.*, vol. 8, no. 1, pp. 61–67, Feb. 2008.

[5] J. F. Tate and M. C. Bell, "Evaluation of a traffic demand management strategy to improve air quality in urban areas," in *Proc. 10th Int. Conf. Road Transp. Inf. Control*, 2000, pp. 158–162.

[6] C. Wilson, G. Millar, and R. Tudge, "Microsimulation evaluation of the benefits of SCATS coordinated traffic control signals," in *Proc. Transp. Res. Board Meeting*, Washington, DC, Jan. 2006, pp. 1–15.

[7] C. R. Bhat and F. S. Koppelman, *Activity-Based Modeling of Travel Demand*. New York: Springer-Verlag, 1999.

[8] W. Davidson, R. Donnelly, P. Vovsha, J. Freedman, S. Ruegg, J. Hicks, J. Castiglione, and R. Picado, "Synthesis of first practices and operational research approaches in activity-based travel demand modeling," *Transp. Res. Part A, Policy Pract.*, vol. 41, no. 5, pp. 464–488, Jun. 2007.

[9] R. Kitamura and S. Fujii, "Two computational process models of activity-travel behavior," in *Proc. Theor. Found. Travel Choice Model.*, 1998, pp. 251–279.

[10] "Metropolitan Travel Forecasting: Current Practice and Future Direction," Nat. Acad., Washington, DC, Special Rep. 288, 2007. Committee for Determination of the State of the Practice in Metropolitan Area Travel Forecasting.

[11] F.-Y. Wang, "Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 630–638, Sep. 2010.

[12] F.-Y. Wang and S. Tang, "Artificial societies for integrated and sustainable development of metropolitan systems," *IEEE Intell. Syst.*, vol. 19, no. 4, pp. 82–87, Jul./Aug. 2004.

[13] F.-Y. Wang, "Toward a paradigm shift in social computing: The ACP approach," *IEEE Intell. Syst.*, vol. 22, no. 5, pp. 65–67, Sep./Oct. 2007.

[14] F.-Y. Wang, K. M. Carley, D. Zeng, and W. Mao, "Social computing: From social informatics to social intelligence," *IEEE Intell. Syst.*, vol. 22, no. 2, pp. 79–83, Mar./Apr. 2007.

[15] G. Xiong, K. Wang, F. Zhu, C. Cheng, X. An, and Z. Xie, "Parallel traffic management for the 2010 Asian games," *IEEE Intell. Syst.*, vol. 25, no. 3, pp. 81–85, May/Jun. 2010.

[16] F.-Y. Wang and S. Tang, "Concept and framework of artificial transportation system," *J. Complex Syst. Complexity Sci.*, vol. 1, no. 2, pp. 52–57, 2004.

[17] F.-Y. Wang, "Toward a revolution in transportation operations: AI for complex systems," *IEEE Intell. Syst.*, vol. 23, no. 6, pp. 8–13, Nov./Dec. 2008.

[18] N. Zhang, F.-Y. Wang, F. Zhu, D. Zhao, and S. Tang, "DynaCAS: Computational experiments and decision support for ITS," *IEEE Intell. Syst.*, vol. 23, no. 6, pp. 19–23, Nov./Dec. 2008.

[19] R. Hranac, E. Sterzin, M. Farzaneh, M. Arafeh, R. Hranac, and D. Krechmer, "Empirical studies on traffic flow in inclement weather," Virginia Tech. Transp. Inst., Washington, DC, Jun. 2006.

[20] M. J. Koetse and P. Rietveld, "The impact of climate change and weather on transport: An overview of empirical findings," *Transp. Res. Part D, Transp. Environ.*, vol. 14, no. 3, pp. 205–221, May 2009.

[21] World Wide Web Consortium (W3C), *W3C. Soap Version 1.2 Part 2: Adjuncts*, Jan. 2007.

[22] E. R. Poelof, C. M. Poe, and K. N. Balke, "Development of a distributed hardware-in-the-loop simulation system for transportation networks," in *Proc. Transp. Res. Board Annu. Meeting*, Washington, DC, 1999, pp. 1–25.

[23] D. Bullock, B. Johnson, B. Wells, M. Kyte, and Z. Li, "Hardware-in-the-loop simulation," *Transp. Res. Part C, Emerging Technol.*, vol. 12, no. 1, pp. 73–89, 2004.

[24] T. Li, D. Zhao, and J. Yi, "Application of ADP to intelligent traffic control," in *Proc. 4th ISNN*, vol. 4491, *Lecture Notes in Computer Science*, Nanjing, China, Jun. 2007, pp. 374–379.

[25] F.-Y. Wang, N. Jin, D. Liu, and Q. Wei, "Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε-error bound," *IEEE Trans. Neural Netw.*, vol. 22, no. 1, pp. 24–36, Jan. 2011.

[26] M. Yang, Y. Liu, and Z. You, "The reliability of travel time forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 162–171, Mar. 2010.

[27] J. J. Sanchez-Medina, M. J. Galán-Moreno, and E. Rubio-Royo, "Traffic signal optimization in "La Almozara" district in Saragossa under congestion conditions, using genetic algorithms, traffic microsimulation, and cluster computing," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 132–141, Mar. 2010.

[28] H. Zhao, S. Tang, and Y. Lv, "Generating artificial populations for traffic microsimulation," *IEEE Intell. Transp. Syst. Mag.*, vol. 1, no. 3, pp. 22–28, Fall 2009.

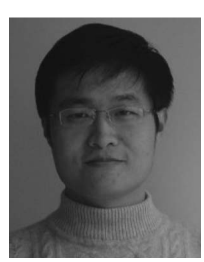
Fenghua Zhu was born in 1976. He received the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China. His thesis was about the evaluation of traffic signal control systems based on artificial transportation systems.

After working as an Assistant Researcher, in 2003, he started his career in intelligent transportation systems with the Institute of Automation, Chinese Academy of Sciences, where he is currently an Associate Researcher with the State Key Laboratory for Intelligent Control and Management of Complex Systems. His research interests are artificial transportation systems and parallel transportation management systems.

Guoxi Li is currently a Professor with the College of Electronic Science and Engineering, National University of Defense Technology, Changsha, China, where he is also a Professor with the Center for Military Computational Experiments and Parallel Systems. His research interests include Artificial societies, Computational experiments, and Parallel execution (ACE) theory and parallel systems technology.

Zhenjiang Li received the B.Eng. degree in electrical engineering and automation from Beijing Jiaotong University, Beijing, China, in 2002 and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, in 2007.

He is currently an Assistant Professor with the State Key Laboratory for Intelligent Control and Management of Complex Systems, Institute of Automation, Chinese Academy of Sciences. His research interests include agent-based control, parallel control, and management for transportation.



Cheng Chen received the B.Eng. degree in measuring and control technology and instrumentation from HuaZhong University of Science and Technology, Wuhan, China, in 2008. He is currently working toward the Ph.D. degree in control theory and control engineering with the State Key Laboratory for Intelligent Control and Management of Complex Systems, Chinese Academy of Sciences, Beijing, China.

His research interests include multiagent systems and distributed artificial intelligence and its applications.

Ding Wen is currently a Professor with the Center for Military Computational Experiments and Parallel Systems, National University of Defense Technology, Changsha, China. His research interests include algebra of communicating processes theory and parallel control and management.