Transportation Research Part C 64 (2016) 164–182

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Personalized real-time traffic information provision: Agent-based optimization model and solution framework

Jiaqi Ma [a], Brian L. Smith [b], Xuesong Zhou [c,*]

[a] Leidos, Inc., 11251 Roger Bacon Drive, Reston, VA 20190, USA
[b] Department of Civil and Environmental Engineering, University of Virginia, Thornton Hall, 351 McCormick Road, Charlottesville, VA 22904, USA
[c] School of Sustainable Engineering and the Built Environment, Arizona State University, Tempe, AZ 85287, USA

ARTICLE INFO

Article history:
Received 16 May 2014
Received in revised form 3 March 2015
Accepted 3 March 2015
Available online 21 March 2015

Keywords:
Agent-based modeling
Network modeling
Traveler information provision
Dynamic traffic management

ABSTRACT

The advancement of information and communication technology allows the use of more sophisticated information provision strategies for real-time congested traffic management in a congested network. This paper proposes an agent-based optimization modeling framework to provide personalized traffic information for heterogeneous travelers. Based on a space–time network, a time-dependent link flow based integer programming model is first formulated to optimize various information strategies, including elements of where and when to provide the information, to whom the information is given, and what alternative route information should be suggested. The analytical model can be solved efficiently using off-the-shelf commercial solvers for small-scale network. A Lagrangian Relaxation-based heuristic solution approach is developed for medium to large networks via the use of a mesoscopic dynamic traffic simulator.

1. Introduction

Advanced Traveler Information Systems (ATIS) aim to enable drivers and commuters to adapt to changing traffic conditions and make informed routing decisions. In particular, Dynamic Message signs (DMS/VMS) and Highway Advisory Radio (HAR), as well as 511 systems and emerging social media such as twitter, are widely used to deliver information on major traffic events (e.g., incidents, congestion) by public agencies, however, they are usually spatially and/or temporally limited, and constrained in the amount of information delivered. Meanwhile, private-sector information provision vendors can provide the user optimal but uncoordinated routes, through in-vehicle route guidance system, to equipped drivers.

Many empirical studies (e.g., Peeta et al., 2000) reveal that drivers prefer detailed traffic information, including incident/congestion location, expected delay and available alternative routes, and more detailed information can yield higher diversion rates. The proliferation of mobile communication technology and devices such as smartphones and on-board units of connected vehicles provide an accessible and cost-effective platform for public-sector Traffic Operation Centers to deliver location-based and personalized traveler information that is timelier and seamlessly integrated with system-wide transportation management goals (e.g., Williams, 2010). The emergence of social networks has enabled direct access to people's mobility patterns and the ability to interact with them, thus presenting an opportunity to incentivize behavior change (either through a social group or the social network). Under major incident or severe weather conditions, a systematical integration of (1)

* Corresponding author. Tel.: +1 480 9655827.
E-mail addresses: Jiaqi.ma@leidos.com (J. Ma), briansmith@virginia.edu (B.L. Smith), xzhou74@asu.edu (X. Zhou).

http://dx.doi.org/10.1016/j.trc.2015.03.004
0968-090X/

publishing area-wide or corridor-level information through DMS and HAR and (2) delivering personalized guidance to individual drivers can greatly help transportation system operators not only reduce the cost and complexity of maintaining ATIS infrastructure but also effectively use bi-directional communications to manage vehicular networks.

All of these advances have created the need for new modeling approaches (in particular to encompass the new data), new estimation, inference and filtering methods and are leading to the development of new paradigms for control. This paper aims to develop an agent based optimization framework to use personalized information provision (PIP) to reduce system-wide delay. We are interested in addressing many practically important but theoretically challenging questions regarding a successful PIP deployment under non-recurring congestion conditions such as: when and where to inform travelers, which drivers to be informed and what alternative route to be suggested for different drivers.

2. Literature review

2.1. Modeling of traveler information provision

As the most common approach of traffic management, the use of VMS to manage real-time traffic have been studied by many researchers from different aspects. Peeta et al. (2000) conducted a stated preference survey along a freeway corridor and examined various driver diversion behavior with different VMS messages contents, including location of accident, expected delay, detour strategy and various combinations of the above. Their study shows that the maximum of willingness to divert occurs when all three information contents are provided. A later study (Peeta and Gedela, 2001) incorporated the driver response in a three-step VMS control heuristic, including activation, message display and update. The required diversion rate is determined by the different system-optimal assignment proportions (prepared offline using mean OD demand) and multiple user class situation (no-information user and real-time information user). Xu et al. (2012) incorporated an aggregated driver behavior model with attributes that can be obtained on-line (traffic message, traffic flow, weather and incident). The behavior model is calibrated on-line and was used in a feedback control to decide optimal traffic information. These studies focused on "optimal" VMS contents considering entire passing vehicles as a whole, while not explicitly considering which drivers or what proportion of them need to or should receive real time information.

The VMS location problem is closely related to the information provision problem in this paper. Since VMS is a fixed asset requiring a large amount of capital investment, most of the studies were to determine where to install the VMS so that the long-term system performance was optimal. A study by Abbas and McCoy (1999) sought to determine the locations for VMS to maximize potential benefits, defined as the sum of changes in delay and accidents on the freeway upstream and downstream of the incident and those on the alternative routes. The reduction in delay as a result of diversion was computed by the demand-capacity analysis procedure in the Highway Capacity Manual. Huynh et al. (2003) and Chiu and Huynh (2007) proposed novel dynamic network flow optimization models in which a set of the VMS locations are selected in the planning stage to minimize the expected network travel time given possible random incident realizations. Based on a simulation based Dynamic Traffic Assignment (DTA) framework, greedy heuristics and Tabu search algorithms are developed to find a near-optimal solution from a limited number of candidate links due to the exponentially increasing amount of combinations.

In the broad field of real-time traffic management modeling there are multiple subcategories that focus on vehicular flow management, for example simulation-based traffic assignment models, reactive feedback control, mathematical programming and dynamic programming. A dynamic programming approach (e.g., Charbonnier et al., 1991) produces optimal solutions and many insights because of its analytical nature, while it can only be applied to small size problems due to the huge computational burden associated with the large number of required modeling states in the formulation. The simulation-based approach can better describe complex traffic flow dynamics required for real-world traffic flow control applications. A number of studies along this line (Mahmassani and Jayakrishnan, 1999; Paz and Peeta, 2009; Bolelli et al., 1991; Hawas and Mahmassani, 1995; Hawas, 2012) strike to seamlessly integrate DTA simulators into general optimization or feedback control frameworks. The mathematical programming approach has received continuous attention from the research community, as its analytical nature offers many insights into the problem. Various optimization models have been proposed in the literature to produce simple and compact formulations, such as linear programming, so that the resulting problems are easy to solve by standard optimization solvers. On the other hand, dynamic network structures introduced by DTA optimization models are still quite complex and most of the existing models targets only simple test networks, such as single destination cell transition model based formulation by Ziliaskopoulos (2000). It should be also remarked that most of the optimization models in the literature are based on time-dependent origin–destination flow, with a typical departure time interval of 5 or 15 min. This flow-based modeling paradigm is adequate for finding the optimal dynamic system through flow-based traffic management strategies such as signal controls or aggregated information provision. In contrast, the PIP represents a fundamentally new approach for real-time ATIS system modeling under ubiquitous communication. This allows traffic system operators to fully optimize and coordinate individuals' trip plans according to the personal value of time, allowable budgets for congestion tolling and willingness to taking detours.

2.2. Agent-based modeling and optimization

Closely related to the PIP problem under consideration in this paper, the agent-based modeling approach has received increasing attention by transportation researchers to capture personal characteristics in traveler's daily activities, such as route choice, departure time choice, en-route diversion, as well as the interactions between individual agents. Early research by Dia (2002) models driver's route choice behavior based on real-time traffic information provided by dynamically changing road signs using an agent-based simulation approach. Adler and Blue (2002) suggested a solution concept for the efficient usage of network capacity with the following agents: network managers, information service providers, and individual drivers negotiating for network resources. Zhang and Levinson (2004) developed an agent-based travel demand model in which travel demand emerges from the interactions of three types of agents in the transportation system: node, arc, and traveler. Simple local rules were efficiently used to solve complicated transportation problems, such as traffic distribution and assignment. Zhang et al. (2008) explored the welfare consequences of product differentiation using heterogeneous user agents on a congested mixed ownership network. To evaluate the impacts of different integrated multimodal corridor management strategies, Zhou et al. (2008) developed a dynamic micro-assignment modeling approach to simulate the route, mode and departure time choices of traveling agents. Zhu et al. (2008) proposed an Agent-based Route Choice (ARC) model to track choices of each individual decision-maker in a road network over time and map individual choices into a macroscopic flow pattern. Li et al. (2011) constructed a multi-day vehicle-based simulation framework to evaluate the response and benefits of traffic information provision under stochastic demand and capacity conditions.

While the agent-based approach shows its effectiveness in capturing individual behavior and system dynamics, its use in system-wide traffic management and optimization is still limited by several intrinsically challenging modeling characteristics. Agents are commonly modeled through rule-based mechanism and in a spatially distributed manner, which makes it difficult to construct a global traffic state representation and thus difficult to guide agents' behavioral choice to reach a global optimal solutions. In order to optimize agent-based systems, researchers proposed various combinations of agent-based models with mathematical optimization tools, to name a few (Persson et al., 2005; Davidsson et al., 2007; Gaffuri, 2006; Gjerdrum et al., 2001). In the specific area of transportation control and management, Chen et al. (2014) recently introduced a simulation based optimization approach to improve the system-wide performance represented by an agent-based DTA simulator.

2.3. Overview of contributions

Researchers have been advocating hybrid models to combine agent-based approaches and optimization techniques to better reflect system reality and also achieve better controllability for the target system. This paper aims to build an agent-based linear integer programming model to produce optimal real-time, en-route PIP strategies. While there are many studies in the literature on VMS-type route guidance or route guidance from a private-sector service provider, this study offers the following distinct modeling features.

First, compared to VMSs with fixed locations, it offers geometrically scalable and adaptive ATIS solutions and only makes suggestions when necessary to optimize system performance. Compared to existing personalized but uncoordinated route guidance systems, our proposed approach can deliver personalized, en-route information to a minimum set of individuals to achieve system-wide performance in an extremely cost-effective manner.

Second, in addition to capturing the essential characteristics of individual travelers, the proposed integer programming model can comprehensively optimize various dimensions of real-time PIP decisions, including where and when to provide the information, to whom the information is given, and what alternative route information should be suggested.

Third, the proposed analytical model is computationally tractable for standard optimization solvers to handle small to medium-scale networks. Its decomposable problem structure allows a systematic integration of Lagrangian Relaxation-based heuristics and a mesoscopic DTA simulator to provide practically useful solutions for medium and large scale networks.

Considering the many related important studies, we further summarize the differences between this paper compared with the most relevant literature. The significant work of real time information provision by Peeta and Mahmassani (1995) built a quasi-real time multi-user class assignment problem in a rolling horizon framework based on a simulation DTA. This methodology relies highly on simulation and the computational burden when analyzing large networks, such as searching for UE/SO equilibrium in each steps. Usually we can only trade-off solution accuracy with computational efficiency. The agent-based model in this paper avoids the search for equilibrium and the input to the model is real-time probe data, such as GPS traces, thereby saving large amount of computational time. Some researchers built very complex optimization frameworks that cannot be solved by traditional optimization algorithms, and instead applied heuristics, like Tabu search, to find near optimal solution. These models can only be used to find solution for discrete variables (e.g., location) of limited amount (e.g., 2–3 VMS) (Chiu and Huynh, 2007; Huynh et al., 2003). Our paper builds a mathematical programming model using simplified model (point queue model) for network flow representation. The model can be efficiently solved by standalone commercial optimization solvers. We also introduce a Lagrangian Relaxation-based heuristics and efficiently solve the optimization problem using a simulator for large scale problems. While some work (e.g., Ziliaskopoulos, 2000; Sawaya et al., 2001) also realized the benefits to formulate an analytical linear mathematical programming model, these models used complex dynamic network loading models, such as cell transmission model (Daganzo, 1994, 1995), and their model can only solve single destination in their formulations, which significantly reduced the applicability of such models. Instead, we develop a special representation of network, multi-layered, space–time expanded network, which allows us a compact formulation, not only considering multi-OD but also more details for information provision strategies. In addition, based on the traffic management requirements (fourth column in Table 1), we use agent-based models for

Table 1
Comparison of optimization models, agent-based approaches and real world systems.

Attributes	Optimization approach	Agent-based modeling approach	Traffic management requirements (to be addressed by our approach)
Problem size	Able to solve small to medium sized problems	Scalable since only local behavior is considered	Small to large size, depending on problem nature and network topology
Decision level	Centralized decisions made for all travelers	Distributed decisions made by each traveler	Centralized decisions should be made for vehicles, while considering vehicle characteristics, behavior and response
Resolution level	Global view with simplified system description without considering detailed traveler characteristics	Detailed description of characteristics and behavior of different agents, but lack of global view	Need traveler details consideration but also need to trade it off for control ability due to incurred computational burden
Solution quality	Solution of global optimum; consider final realized system state (e.g., route travel time) with predicative ability	No guarantee of system optimum since only local rules considered; only consider current system state for each step without predicative ability	Need system optimum solution and also consider individual characteristics and preferences; Need to be predictive to consider realized system state for better strategy provision

developing personalized traffic information strategies. Table 1 briefly compares our approach with traditional optimization approach and simple agent-based approaches.

Our paper is organized as follows. Based on a space–time network representation, we first offer the problem statement in session 3, and describe key variables and constraints for the proposed mathematical programming model in session 4. After discussing several distinctive modeling features in detail, a Lagrangian Relaxation-based heuristics is introduced in session 5, followed by numerical experiments for various test networks in Section 6.

3. Space–time network formulation with agents on multiple origin–destination pairs

The use of space–time expanded network representation (introduced in detail in this section) is advantageous and can facilitate the problem formulation in many ways, such as ability to capture complex time-dependent flow conservation with multiple commodity and convenient consideration of time-dependent link capacity and travel time as they are direct input to our model. For example, Yang and Zhou (2014) adopted the space–time network representation for a expected shortest path problem, but only for single path and OD. This paper builds upon this concept, and incorporate more elements, such as multiple OD and different commodities, for modeling real time information provision. Table 2 lists the notations used in this paper for our multi-OD information provision model.

3.1. Space–time network representation

Consider a directed, connected traffic network (N,E), where N is a finite set of nodes, and E is a finite set of traffic links between different adjacent nodes. The planning time horizon is discretized into a set of small time slots, denoted by $T = \{t_0,\ t_0 + \sigma,\ t_0 + 2\sigma, \ldots, t_0 + M\sigma\}$. Symbol t_0 specifies the given departure time from the origin node O, and σ represents a short time interval (e.g., 6 s) during which no perceptible changes of travel times are assumed to take place in a transportation network. M is a sufficiently large positive integer so that the time period from t_0 to $t_0 + M\sigma$ covers the entire planning horizon.

For a given physical network, one can construct a corresponding space–time expanded network, denoted by (V, A), expanded from the physical network (N, E) and time-varying link travel time. Specifically, $V = \{(i,t)|i \in N, t \in T\}$ represents the set of time-dependent nodes, where $(i,t) \in V$ indicate the state of node i at time stamp t and each state will be treated as a separate node. The set of time dependent arcs is represented as $V = \{(i,j,t,s)|(i,j) \in E,\ t_0 \leqslant t \leqslant s \leqslant t_0 + M\sigma\}$, where time dependent arc (i,j,t,s) occur in the space–time network when one can travel from physical node i at timestamp t and arrive at physical node j at timestamp s. As shown in Fig. 1, the plot on the left hand side exemplifies a physical network with assumed 1-min travel time for each link, while the right hand side depicts its corresponding space–time network with a horizontal time dimension. Waiting arcs are introduced to model the situation of traveling agents staying at a node from one timestamp to the next, represented by dash lines in Fig. 1. We also added source/sink node as well as super arcs from the super source and to the super sink (represented by dot-sash lines). For simplicity, our model first considers a point-queue model where each traveling agent is assumed to travel through a link at free-flow speed and waiting at the end node of the link if the inflow capacity of the subsequent link is unavailable. The point queue model can be relatively easy extended to the spatial queue and kinematic wave model, as shown by Zhou and Taylor (2014). A similar space–time representation was adopted by Yang and Zhou (2014) for an expected least travel time path problem where multiple days of observed link travel times are used to represent different random scenarios. This space–time structure can be also easily incorporated in the implementation for a dynamic network loading problem using agents.

Table 2
Notations for sets, parameters and variables.

Symbol	Definition
Superscripts, subscripts and parameters used in mathematical formulation	
N	Set of nodes in the physical traffic network
E	Set of physical road links
V	Set of nodes in the space–time network
A	Set of arcs in the space–time network, including waiting arcs $(i,i,t,t+1)$
i	Index of different time stamps, $t \in \{t_0, t_0+\sigma, t_0+2\sigma, ..., t_0+M\sigma\}$
i,j	Indices of nodes, $i,j \in N$
(i,j)	Index of road links between nodes i and j, $(i,j) \in E$
(i,j,t,s)	Index of space–time network arcs between physical node i and j entering at time t and leaving at time s, including waiting arcs $(i,i,t,t+1)$
O_k	Set of travel origins corresponding to each vehicle k departing at time t, $(i,k,t) \in O_k$
D_k	Set of travel destinations corresponding to each vehicle k, $(i,k) \in D_k$
IN	Set of intermediate nodes (all notes other than origins and destinations) labeled with time stamp t, $(i,t) \in \mathrm{IN}$
c_{ijts}	Free flow travel time on road link (i,j) from time stamp t to s
K	Set of vehicles
k	Index of vehicles
cap_{ij}^{t}	(entering) capacity of link (i,j) at time stamp t
α	Generalized unit cost of providing information to a user
h_{ij}^{k}	1 if link (i,j) is used by vehicle k as part of historical route
H^{k}	Set of physical links on historical route of vehicle k
g_{ij}^{k}	1 if link (i,j) is used by vehicle k as part of real time information route
ω	Incident detection time stamp (start of information provision)
β	Percentage tolerance to diversion routes
c_{hist}^{k}	Historical travel time of vehicle k
B	Information budget constraint (the amount of travelers who can receive detailed information)
Decision variables used in mathematical formulation	
y_{ijts}^{k}	1 if traffic link (i,j) is used from time stamp t to s by historical information vehicle k; 0 otherwise
z_{ijts}^{k}	1 if traffic link (i,j) is used from time stamp t to s by real-time information vehicle k; 0 otherwise
x_{js}	Number of vehicles that receive real time information at node j and time stamp s

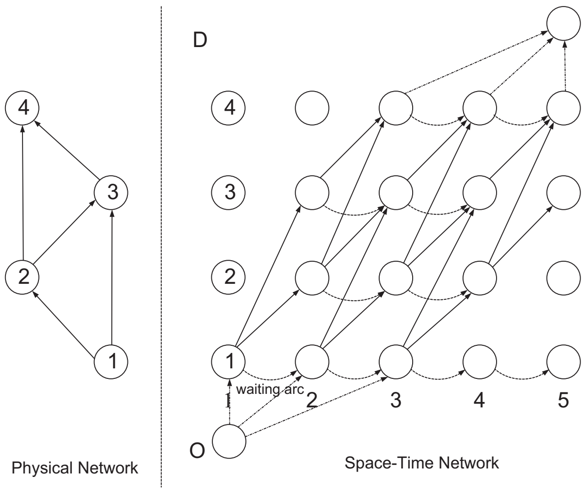

Fig. 1. An illustration of physical network and space–time expanded network.

3.2. Illustrative example and key decision variables

All the decision variables and parameters of the proposed model are listed in Table 1. We further explain a few important model concepts here. Two most important decision variables in the agent-based optimization formulation, $y_{ij,ts}^{k}$ and $z_{ij,ts}^{k}$, indicate whether agent k uses physical link (i,j) from time t to time s, along its Historical Info (HI) route or real-time Personalized Info (PI) route, respectively. Agents with PI are assumed to follow the navigation guidance optimized by the traffic system operator.

In Fig. 2 (with the same structure as Fig. 1), we assume that three agents travel from node 1 to node 4 with a HI path as 1–2–4, with agent "1" and "2" starting at minute 1 and agent "3" starting at minute 2. An incident occurs on link (2,4) at minute 2 and last for 2 min. Without receiving any information updates, agent 1 still strictly follows the HI route and has to wait at node 2 for 2 min. Agent 2 receives real-time PI instructions at node 2 and minute 2, and then he/she follows

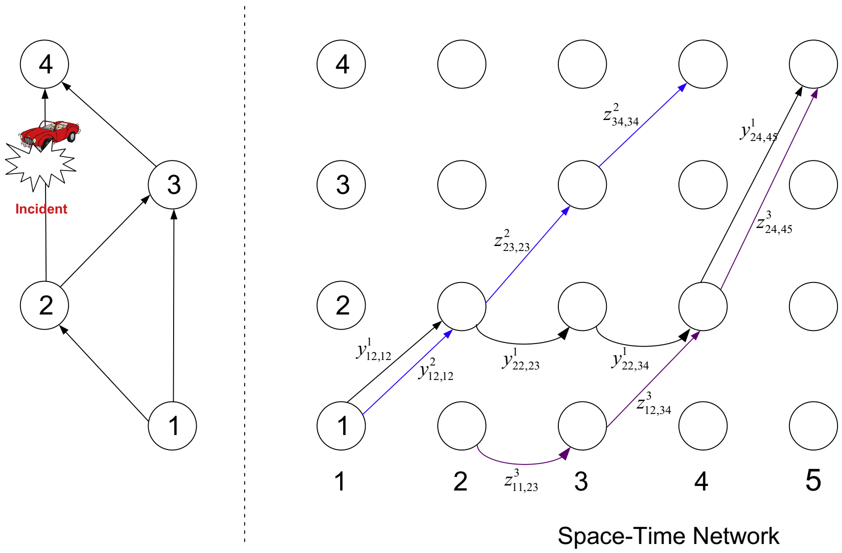

Fig. 2. Illustration of decision variable y and multi-OD flow in a space–time network.

the alternative route of 2–3–4 to avoid traffic congestion on link (2,4). With the same OD and departure time, agent can save 1 min in travel. Agent "3" receives real-time PI instructions at home before departure and then he/she waits for another 2 min at home. Although the planned departure time of agent "3" is 1 min later than agent "1", it is able to arrive at the destination 1 min earlier than agent "1". Agent "3" is suggested to wait at origin (home) instead of at nodes of road network, since the generalized cost of waiting at home is much lower than waiting on the road. In terms of variable representation, agent 1 sets binary variable y to 1 along its traveling space–time trajectory, while agent 2 has $y = 1$ on link (1,2) and then its variable y is converted to variable z thanks to information provision.

Another decision variable x_{js} represents the number of agents that receive PI information at node j and time stamp s. The sum of the variables x over different nodes j and different times s is equivalent to the total number of PI equipped travelers to be informed in the entire network.

3.3. Constraints on time-dependent network flows

The network flow balance constraints are required for both static and time-expanded networks in order to represent flow propagation correctly. While flow balance constraints based on physical links are quite common in the literature, it is non-trivial to consider the flow balance in a space–time expanded network with user class transition (from HI to PI). In the proposed model, constraints (1)–(3) are flow balance constraints at origin, destination and intermediate nodes, respectively.

First, at the origin node, shown in Eq. (1), each agent is loaded from his path origin node and the travel time from the origin to the first node is zero. Since no pre-trip information is considered in this study, all agents departing the origins follow the historically best route on the first link. Note that set O_k represents the origin for each agent and each element contains three entries, origin node ID of physical network, agent ID and departure time stamp.

3.3.1. Flow balance constraints on origin node

$$\sum_s \sum_j y^k_{ijts} = 1, \quad \forall (i,k,t) \in O_k \tag{1}$$

Next, the flow balance on destination node constraint is presented in Eq. (2), which dictates that every agent will flow into destination nodes from one of the intermediate nodes at different times. We consider the possibility of the existence of information provision on this layer by allowing different user classes. Note that set D_k represents the origins for each agent and each element contains two entries, including destination node ID of physical network and agent ID. There is no time stamp entry compared with O_k since there is no set time for agent arrival because of complex conditions during the travel, such as incident, congestion and diversions.

3.3.2. Flow balance constraints on destination nodes

$$\sum_t \sum_s \sum_i [y^k_{ijts} + z^k_{ijts}] = 1, \quad \forall (j,k) \in D_k \tag{2}$$

Then, flow balance on intermediate node constraint is formulated in Eq. (3). The intermediate nodes here are not necessarily on the historical path when considering the possibility of diversion. This constraint applies to every node on the layer other than origins and destinations for each agent. Also, waiting arcs $(i, i, t, t+1)$ are allowed in the flow balance constraint for intermediate nodes (not for origins and destinations), to represent the condition where downstream link capacity is not enough for all incoming agents and queues build up.

3.3.3. Flow balance constraints on intermediate node

$$\sum_t \sum_i [y^k_{ijts} + z^k_{ijts}] = \sum_{t'} \sum_{i'} [y^k_{ji'st'} + z^k_{ji'st'}], \quad \forall j \in IN, s \in T, k \in K \tag{3}$$

This constraint (Eq. (3)) is also very crucial for modeling information provision. Intermediate nodes allow activity on them, which means information provision is realized at nodes. As seen in the example in Fig. 3, two agents with historical information arrive at node j at time stamp t. Assuming they do not need to wait at node j, agent 1 continues on the historical route while agent 2 receives real-time information and is converted to en-route information agent (dashed line) and starts to use the suggested alternative route. Constraint 3 requires total volume constraint, but allows transfers of user information. In this way, the modeling of information provision is reflected by the conversion of network user groups at intermediate nodes. Under different circumstances, Eq. (3) may reduce to simpler forms.

Condition 1: for agents that haven't received any information, $z^k_{ijts} = 0$, Eq. (3) reduces to:

$$\sum_t \sum_i y^k_{ijts} = \sum_{t'} \sum_{i'} y^k_{ji'st'}, \quad \forall j \in IN, s \in T, k \in K$$

Condition 2: At an information provision point (diversion point), Eq. (3) reduces to:

$$\sum_t \sum_i y^k_{ijts} = \sum_{t'} \sum_{i'} z^k_{ji'st'}, \quad \forall j \in IN, s \in T, k \in K$$

Condition 3: for already diverted flow. Eq. (3) reduces to:

$$\sum_t \sum_i z^k_{ijts} = \sum_{t'} \sum_{i'} z^k_{ji'st'}, \quad \forall j \in IN, s \in T, k \in K$$

3.4. Constraints on network propagation

This paper adopts concepts of the bottleneck model (Vickrey, 1963) for dynamic network flow propagation, as shown in Fig. 4. In this model, agents always move along a link with the free flow speed until they arrive at the exit point, where they form a queue if the outflow rate (or inflow rate of next link) they induce exceeds the maximum value (capacity flow) of the link. Since the bottleneck model ignores the physical length of agents and assumes that a queue occupies a point, it is also known as the vertical queue or point queue (P–Q) model. Other dynamic network loading models can also be adopted in the optimization framework we proposed here with some modification, such as Newell's simplified kinetic wave model (Newell, 1993; Lu et al., 2013) and cell transmission model (Daganzo, 1994, 1995). However, using more complex models may add to the complexity of the optimization framework and increases computation burdens. In this paper, we consider link capacity as inflow capacity, which means only a certain number of agents are allowed to enter the link at a time stamp, as shown in Eq. (4).

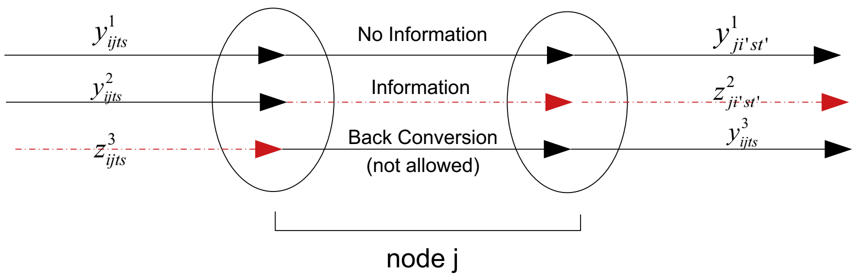

Fig. 3. Illustration of user information group conversion process.

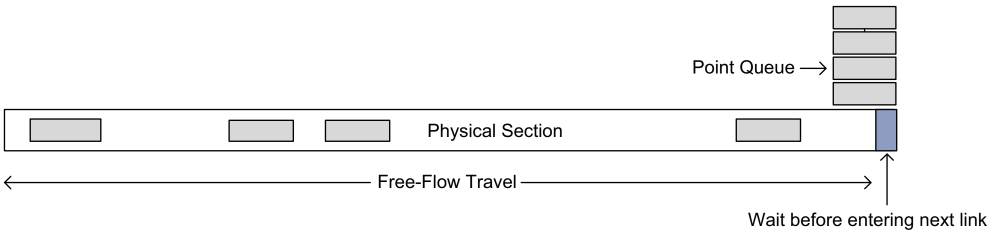

Fig. 4. Illustration of point queue model.

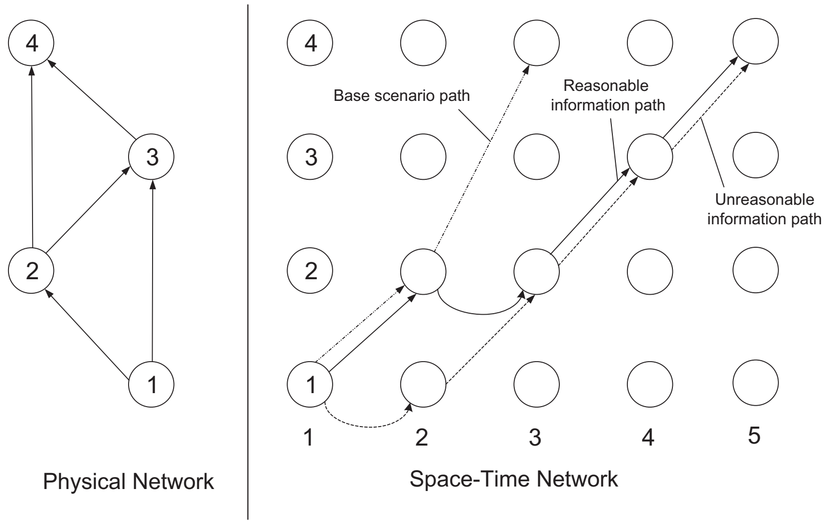

Fig. 5. Illustration of continuous flow constraint.

3.4.1. Inflow capacity constraints

$$\sum_{k}\sum_{s}[y_{ijts}^{k}+z_{ijts}^{k}] \leqslant cap_{ijt}, \quad \forall (i,j,t) \in A \tag{4}$$

If there is information provision, and an agent has to wait at or before a diverting node, System Optimum (SO) mathematical models may produce results allowing agents to wait at earlier nodes (instead of bottleneck nodes) while there is still unused capacity of next link along the historical path. This may be problematic in some cases. One example is shown in Fig. 5. An agent wanted to travel form node 1 to node 4 via path 1–2–4. The dot-dash line is the base case scenario path. When link (2,4) is closed and information is provided only from time 3, the agent would wait at node 2 from time 2 to 3 (given the travel time of each link is 1) and then follow the suggested route 2–3–4. However, without extra constraints added, the program may let agent wait at node 1 from time 1 to 2 and then follow suggested route 1–2–3–4 without waiting in-between. This is not realistic since agents will travel along the historical path continuously and not stop unless no capacity is left in the next link. This becomes a problem if an agent stays on an earlier node too long, and the program may provide them information at that node, which is not possible in reality. Therefore, we add "continuous flow constraints" to prevent such unrealistic situations.

3.4.2. Continuous flow constraints

$$y_{jjs,s+1}^{k} \leqslant \left[\sum_{k'}\sum_{t}(y_{jist}^{k'}+z_{jist}^{k'})\right] \Big/ cap_{jis}, \quad \forall (j,i,s) \in A, (i,j) \in H^{k} \tag{5}$$

where H^k is the historical link set of agent k. This means that, along the historical path, each agent must proceed unless there is not enough capacity ahead. So, if the there is no capacity left for the next link, the right-hand-side (RHS) is 1 and then it is possible that left-hand-side (LHS) term equals 1. Otherwise, when there is no capacity left in the next link (RHS = 0), then LHS must equal 0. Note that waiting arcs are not physical links and thus not included in set H^k.

It is known that mathematical formulations of system optimum (SO) problems are problematic in meeting First-In-First-Out requirements and in generating holding-back phenomenon. Generally, they would require the introduction of additional constraints that yields a non-convex constraint set, destroying many nice properties of the formulation and severely increasing the computational requirements. Readers can refer to the literature (e.g.,Carey and Subrahmanian, 2000) for some introduction. This paper aims to build a mathematical model which can be solved efficiently and used in traffic management. We will leave this problem for future exploration. Also, this paper adopts an integrated framework combining a mathematical programming model and a simulator. We will use our formulation to decide an optimal information strategy and use a traffic simulator to evaluate the information strategy.

3.5. Constraints on information provision

In addition to Eq. (3), multiple other constraints are needed to model real-time information provision. Information activation constraints (Eq. (6)) states that the diverted agents have already received real-time information and cannot be converted back to historical information again.

3.5.1. Information activation constraints

$$\sum_{i}\sum_{t} y^k_{ijts} \geqslant \sum_{i'}\sum_{t'} y^k_{ji'st'}, \quad \forall (i,s),(i',s) \in A, j \in IN, k \in K \tag{6}$$

Two other important constraints, historical information user constraint (Eq. (7)) and real time information provision constraint (Eq. (8)) decide where information can be located (candidate information locations). Eq. (7) states that historical information users will only use links that are historically used, with h^k_{ij} a pre-specified binary parameter. For example, as in the network in Fig. 3, a historical path of 1–2–4 for agent k can be represented by $h^k_{12} = 1$ and $h^k_{24} = 1$ while other $h^k_{ij} = 0$. Thus, this constraint will force all y^k_{ijts} values except y^k_{12ts} and y^k_{24ts} equal zero. The historical links and h^k_{ij} value can be determined by many methods, such as GPS probe data or a traffic assignment equilibrium run.

3.5.2. Historical information provision constraints

$$y^k_{ijts} \leqslant h^k_{ij}, \quad \forall (i,j,t,s) \in A, k \in K \tag{7}$$

Real time information provision constraint (Eq. (8)) prescribes candidate information links that can be suggested to each agent/traveler via parameter g^k_{ij}. One the one hand, this parameter can be set by TOC to avoid traffic being diverted to unsuitable routes. One the other hand, this parameter can be set for each agent and make information provision more personalized. Without the loss of generality, we set all g^k_{ij} value to be 1 for the personalized traveler information in this paper. Note that g^k_{ij} is only for physical links and this is fully consistent with actual real-world conditions, where only physical paths can be suggested, though travelers who follow the suggested route might have different space–time trajectories.

3.5.3. Real time information provision constraints

$$z^k_{ijts} \leqslant g^k_{ij}, \quad (i,j,t,s) \in A, k \in K \tag{8}$$

Information provision amount constraints (Eq. (9)) define whether a certain agent k receives real time information at a certain node j and time stamp s.

3.5.4. Information provision constraints

$$r^k_{js} = \sum_{t}\sum_{i} [y^k_{ijts}] - \sum_{t'}\sum_{i'} [y^k_{ji'st'}], \quad \forall j \in IN, s \in T, k \in K \tag{9}$$

Node information provision constraints (Eq. (10)) define the total number of agents that receive real time information at a certain node j and time stamp s. Note that x_{js} is also a variable in this formulation to control the number of agent that can receive information.

3.5.5. Node information provision constraints

$$\sum_{k} r^k_{js} \leqslant x_{js}, \quad \forall j \in IN, s \in T \tag{10}$$

3.5.6. Budget constraints

Information provision usually comes with a cost. Traditional VMS incurs a lot of infrastructure and maintenance investment. The proposed system in this paper requires capacities of commutation and user devices. Also, receiving and comprehending information (especially complex ones) is distraction to them and a type of invisible cost. Thus, we introduce a parameter B, information provision budget. Parameter α is the cost to provide information to each travelers. Parameter B should be decided based on available resources, engineering judgment and sensitivity analysis like in the case study section of this paper. If we assume α equal to 1, the parameter B can also be considered total number of agents that receive real time information.

$$\alpha \cdot \sum_j \sum_s x_{js} \leqslant B \tag{11}$$

Information start time constraint (Eq. (12)) states that information will only be provided after an incident is detected. We achieve this by restricting that all x_{js} is zero until incident detection time ω. This differentiates this study with others on point-to-point route guidance, and it indicates that information is only provided when incident happens, such as a crash or severe congestion.

3.5.7. Information start time constraints

$$\sum_j \sum_{s \leqslant \omega} x_{js} = 0 \tag{12}$$

In order to provide a deeper insight of the concept of information provision in this model, we further discuss here via comparison with VMS and radio information. For VMS with alternative information, g_{ij}^k needs to be the same for all agents and VMS has no control on the provision amount x_{js}. It is possible to have either over-diversion or under-diversion. While it would be interesting to investigate users' response rate and information contents (Peeta et al., 2000; Peeta and Gedela, 2001; Paz and Peeta, 2009), one can still find there is a great level of uncertainty about the actual impact of the VMS information provisions. In comparison, the agent-based information provision in our model can be much more flexible as different agents can be fed with different route suggestions with different values of g_{ij}^k to fully utilize road capacity, which is a more controllable and desirable scenario for traffic management.

For roadside variable messages signs without alternative information, it is unknown to the system manager what alternative they will use. To model it, we also need to further apply complex user equilibrium constraints (non-linear) or using instantaneous (not predicted) travel time from a traffic simulator (e.g., set $g_{ij}^k = 1$ for shortest path using prevailing travel time, which is not the scope of this paper), one can capture the greedy behavior and potential drawback due to uncoordinated routing. In comparison, it should be noted that information provision strategy from this model uses predicted travel time (realized travel time after the trip is completed) in the optimization problem while many simulation-based studies used only instantaneous travel time from the simulator. In simulators, implementation of real-time information provision can be generalized as following steps: (1) find agent k; (2) fetch current travel time for each link at time t; (3) calculate shortest path from routing current node to the destination node; (4) set $g_{ij}^k = 1$ for links on the shortest path, $g_{ij}^k = 0$ otherwise; and (5) agent k will follow the suggested route based on $g_{ij}^k = 1$, its actual trajectory is dependent on the simulation results based on road capacity in the future. The model in this paper, instead, uses the concept of predictive travel time directly when solving the optimization model.

The system optimum solution has long been criticized because it may discriminate against some users in favor of others and sometimes recommending very long detours. From the perspective of user experience, this type of solution is unacceptable. Therefore, we introduce detour constraints to prevent long detours for every traveler. Jahn et al. (2005) proposed several different constraints of this type, such as the ones based free-flow travel time, user equilibrium travel time. Here we constrain the detour travel time is no more than $(1 + \beta)$ time of the historical travel time, which can be obtained from real world probe data or results from dynamic network loading.

3.5.8. Detour constraints

$$\sum_{i,j,t,s} c_{ijts} \cdot (y_{ijts}^k + z_{ijts}^k) \leqslant (1+\beta) \cdot c_{hist}^k \quad \forall k \in K \tag{13}$$

3.6. Objective function

The objective function (Eq. (13)) is the total system cost, representing the total system travel time by all travelers in the system.

$$F(x,y,z) = \sum_{i,j,t,s} \sum_k \left\{ c_{ijts} \times [y_{ijts}^k + z_{ijts}^k] \right\} \tag{14}$$

where $c_{ijts} = FFTT(i, j)$, free flow travel time from node i to node j from time t to time $s = t + FFTT(i,j)$; $c_{i,i,t,t+1} = 1$ for waiting arcs at node i from time t to time $t + 1$.

With the objective to minimize the objective function $F(x, y, z)$, total system cost, the optimization problem can be formulated as a mixed integer programming model, as in Eq. (15). Note we also add a binary variable constraint and nonnegative constraint of the decision variables at the end.

$$\min_{x,y,z} \sum_{i,j,t,s} \sum_{k} \{c_{ijts} \times [y^k_{ijts} + z^k_{ijts}]\}$$
$$\text{subject to}: \quad (15)$$
$$\text{Eqs. } (1)-(13)$$

4. Lagrangian Relaxation based heuristics approach

The agent-based optimization model proposed in Section 3 describes the problem analytically and captures complex flow balance relationships for travelers with different types of information. It is able to provide a global view for the problem and obtained a system optimum solution. However, analytical models like this usually suffer from their inability to solve large-scale problems. In order to deal with this problem and make our framework more generalized, in this section, we propose a Lagrangian Relaxation-based heuristic approach and implement it with a simulator, DTALite (http://code.google.com/p/nexta/), for solving the agent-based optimization problem.

One of the advantages of using a simulator to solve such problems is that some constraints of analytical models can be easily handled by a simulator. For example, the network flow balance constraints on origins, intermediate nodes and destinations (Eqs. (1)–(3)) and continuous flow constraint (Eq. (5)) are met automatically in a simulator. Also, as discussed earlier, the limit of a simulator is that agents only follow local rules without a global view and impact the system through their own behavior and interactions with other agents. It is difficult to ensure system improvements by considering only agents' local rules and behavior. Thus, for a traffic management application, it is of critical importance to introduce an optimization process to optimize critical values and achieve system optimum.

4.1. Lagrangian Relaxation based heuristic approach

Although many simulation-based DTA models such as DYNASMART (Jayakrishnan et al., 1994) and DTALite (Zhou and Taylor, 2014) are agent-based, many of its applications (e.g., Lu et al., 2008, 2009) are still using flow-based framework and these tools are used for dynamic network loading. This paper uses an agent-based framework, considering individual agent's characteristics in developing real time information and routing strategy.

As in the analytical model, it is assumed that we know the agent's historical routes, departure time and also the location of the agent when an incident is detected. In real applications, the agents' location can be obtained using real-time map matching algorithms based on probe data such as GPS traces sent to the server by each agent. This technology has been studied extensively and applied widely in the past decades. Each agent's historical route and departure time can be either input by users themselves or obtained by using machine learning approaches that have already been embedded in systems of personal devices, such as IOS 7. These approaches could learn from the data of agent's day-to-day travel patterns and generalize the routes and departure time of largest likelihood.

Since all other constraints can be easily met or handled in a simulation, the analytical optimization model in Section 3 can be reduced to the following form, as in Eq. (16). As stated above, flow balance constraints (Eqs. (1)–(3)) and continuous flow constraints (Eq. (5)) are met automatically in simulation. Information activation constraints (Eq. (6)) are also not necessary since once one of a simulated agent's characteristics is changed, the characteristic will not be changed back unless otherwise forced to. Constraints such as historical information provision constraints (Eq. (7)) or real time information provision constraints (Eq. (8)) can be easily handled in a simulation by considering behavioral characteristics or local rules for each agent. Without loss of generality, we do not consider the budget constraints (Eq. (11)). The detour constraints (Eq. (13)) can be checked every time when we find a new route for an agent. If the new route meets the constraint, accept it. Otherwise, stay with the original route.

Note that, for simplicity, we use only one variable y^k_{ijts} to indicate whether agent k uses link (i,j) from time stamp t to s. For the simulation implementation, the model reduces to:

$$\min_{y} \sum_{i,j,t,s} \sum_{k} \{c_{ijts} \times y^k_{ijts}\}$$
$$\text{subject to}: \quad (16)$$
$$\sum_{k} \sum_{s} y^k_{ijts} \leqslant cap_{ijt}, (i,j,t) \in A \quad y^k_{ijts} \in \{0,1\}$$

We dualize the capacity constraints (hard constraints) and the Lagrangian Relaxation reformulation of (16) is as follows:

$$\min_{y,\lambda} \quad L(y,\lambda) = \sum_{i,j,t,s}\sum_{k}(c_{ijts} \times y^k_{ijts}) + \sum_{i,j,t}\lambda_{ijt}\left[\sum_{k}\sum_{s}(y^k_{ijts} - cap_{ijt})\right] = \sum_{i,j,t,s}\sum_{k}[(c_{ijts} + \lambda_{ijt}) \cdot y^k_{ijts}] - KS\sum_{i,j,t}[\lambda_{ijt} \cdot cap_{ijt}] \tag{17}$$
$$\text{subject to} \quad y^k_{ijts} \in \{0,1\}, \lambda_{ijt} \geqslant 0$$

where λ is the Lagrangian multiplier, and K and S are the total number of agents/agents and potential link head node arrival time stamps. Eq. (17) is a Lagrangian relaxed problem and its optimal solution provides a lower bound to problem (16). Thus, solving problem (17) involves the identification of optimal value λ that produces the tightest or largest lower bound to the primal problem (16). The resulting problem is called a Lagrangian dual problem (18).

$$\max_{\lambda \geqslant 0} \quad L^D(\lambda) = \inf\left\{\sum_{i,j,t,s}\sum_{k}(c_{ijts} \times y^k_{ijts}) + \sum_{i,j,t}\lambda_{ijt}\left[\sum_{k}\sum_{s}(y^k_{ijts} - cap_{ijt})\right]\right\} \tag{18}$$

In this Lagrangian Relaxation framework, in each outer loop iteration n, the solution procedure for problem (16) consists of two major algorithmic steps: given a Lagrangian multiplier $\lambda^{(n)}$, find an optimal y value by solving the Lagrangian lower bound problem (19), and given y value and current network state, update the Lagrangian multiplier by using the subgradient optimization method which is discussed later.

$$\begin{aligned} &\min_{y} \quad L(y) = \sum_{i,j,t,s}\sum_{k}[(c_{ijts} + \lambda^{(n)}_{ijt}) \cdot y^k_{ijts}] - KS\sum_{i,j,t}[\lambda^{(n)}_{ijt} \cdot cap_{ijt}] \\ &\Longleftrightarrow \min_{y} \quad L(y) = \sum_{i,j,t,s}\sum_{k}[(c_{ijts} + \lambda^{(n)}_{ijt}) \cdot y^k_{ijts}] \end{aligned} \tag{19}$$

subject to $y^k_{ijts} \in \{0,1\}$

In Eq. (19), only links along the route taken by each agent k are considered. For each physical link (i,j), the term c_{ijts} represents the link delay encountered by agent itself, with queuing considered by waiting arcs $(i,i,t,t+1)$. The term $\lambda^{(n)}_{ijt}$ can be considered as marginal delay to the whole system caused by the agent's arrival at an arc, either a physical link or a waiting arc. This insight is similar to those from many previous study (e.g., Ghali and Smith, 1995). Then, the generalized time-dependent link cost for an agent arriving at link (i,j), at time t, denoted by g_{ijt}, can be calculated by:

$$g_{ijt} = c_{ijts} + \lambda^{(n)}_{ijt} \tag{20}$$

Then generalized time-dependent path cost used by agent k, denoted by g_k, can be calculated by:

$$g_k = \sum_{i,j,t,s}(c_{ijts} + \lambda^{(n)}_{ijt}) \cdot y^k_{ijts} \tag{21}$$

Eqs. (19)–(21) can be further explained by borrowing the concept of link marginal delay of Ghali and Smith (1995), as in Fig. 6, where the agent arrival curve $A(t)$ and agent departure curve $D(t)$ is the traffic condition at the end of a link and insufficient downstream link capacity make this point a bottleneck. At time t_2, the agent arrival rate is greater than departure rate and a queue forms. When the one unit of agent arrives at time t_2, the queue length is q and he will be able to departure at time t_2, and the total link delay he would encounter is $d = t_3 - t_2$. However, the arrival of this agent also caused delay for other agents arriving here from time t_2 to t_4 (queue dissipation time), and the additional delay imposed on others is $t_4 - t_3$, which is similar to the concept of $\lambda^{(n)}_{ijt}$ in our model. Eq. (19) means that for each agent, $\sum_{i,j,t,s}\left[(c_{ijts} + \lambda^{(n)}_{ijt}) \cdot y^k_{ijts}\right]$ is the path cost for this system optimum problem and the total for each agent should be minimized.

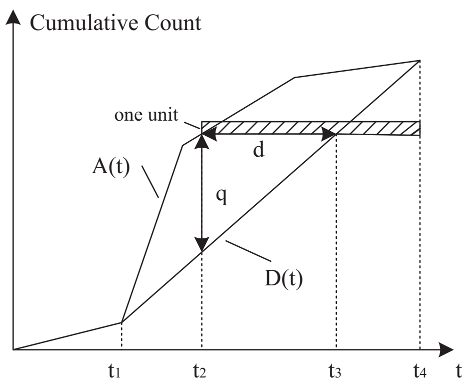

Fig. 6. Illustration the concepts of link and path marginal delay, adapted from Ghali and Smith (1995).

Based on Bellman's principle of optimality, some modified label correcting algorithms (e.g., Ziliaskopoulos and Mahmassani, 1992) can be adopted to search for optimal time dependent least cost path for model (19). In the searching procedure, an $M+1$-dimensional vector-based label, denoted by $\Gamma_j = (\eta_j(t_0), \eta_j(t_0+\sigma), \ldots, \eta_j(t_0+M\sigma))$, can be used to show the least travel cost from origin O to current node j at each time stamp $t' \in T$, where $\eta_j(t')$ is computed by

$$t' = t + c_{ijts} : \eta_j(t') = \begin{cases} \min\{\eta_j(t'), g_{ijt} + \eta_j(t)\}, & j \in N \setminus i, t' \in T \\ 0, & j \in O, t' \in T \end{cases} \tag{22}$$

Note that in Ghali and Smith (1995), the marginal delay of a link or path is calculated using the realized network condition, while we use updated Lagrangian Multiplier in each iteration. Although it would be interesting to know the relationship between both concepts, it is out of this paper's scope and will be left for future investigation.

4.2. Heuristic descent direction method

Since we are considering each agent in calculating the shortest path, we may encounter significant system performance deterioration if we divert all agents which can find a route with lower generalized cost. It is necessary to make sure the decent direction to avoid over-diverting agent in each iteration. Ghali and Smith (1995) presented a simulation-based approach to evaluate local link marginal travel time or delay on a congested link, based on cumulative flow curves. Their approach can be adapted to solve the problem of this paper, as in Algorithm DEC.

4.2.1. Algorithm DEC

Step 1: Obtain historical routes, departure time, and real-time location at the time of incident detection for each agent. For each agent, do Step 2 and Step 3.
Step 2: Determine the agent new route with smallest marginal delay, calculated by summing the local marginal delay of each link on the route.
Step 3: If a new route is found in Step 2, determine the new value of total system travel time and accept the new route if it reduces total system travel time. Retain old route for the agent if total travel is not reduced.
Step 4: If new routes for agents are found in Step 2 and Step 3, then go to Step 2 and Step 3 for another iteration. Otherwise, terminate.

4.3. Subgradient method

The Lagrangian Relaxation model (17) provides a lower bound for the primal problem. We adopt a subgradient method to update the Lagrangian multipliers λ iteratively. We want the lower bound as close to optimal value as possible, while the Lagrangian dual problem (18) needs to be solved. Meanwhile, the upper bound will be updated using newly available, better feasible path solutions (with a lower objective function value of problem (16)) to minimize the dual gap.

To iteratively update the Lagrangian multipliers λ for reaching tighter lower bounds, at each iteration, the direction of update can be calculated as:

$$\nabla L_{\lambda_{ijt}}(\lambda) = \sum_k \sum_s (y^k_{ijts} - cap_{ijt}) \tag{23}$$

Then, the following equation can be used to update the Lagrangian multiplier λ_{ijt} for iteration $n+1$.

$$\lambda^{n+1}_{ijt} = \lambda^n_{ijt} + \theta_n \left[\sum_k \sum_s (y^k_{ijts} - cap_{ijt}) \right] \tag{24}$$

where the parameter θ_n is the step size for updating Lagrangian multiplier. There are multiple ways of determine the step size. Usually in optimization theory, an upper bound will be calculated at each iteration and then used to update step size. However, since evaluating the upper bound is expensive in a simulator, the heuristic approach proposed in this paper adopts a moving average method for updating the step size, as below:

$$\theta_n = 1/(n+1) \tag{25}$$

Once the objective function of Lagrangian Relaxation reformulation problem (Eq. (17)) between two consecutive iterations are less than a pre-determined error ε, or when the iteration number reaches a pre-specified maximum number $N_{\max}$, the algorithm is terminated.

4.4. Solution algorithm

As illustrated in Fig. 7, we now present the complete solution procedure as follows:

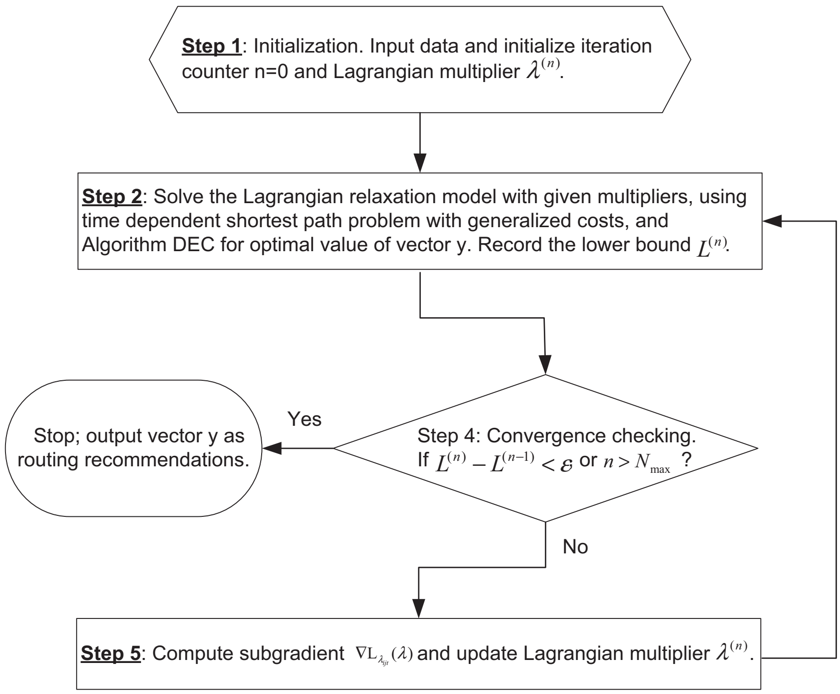

Fig. 7. Solution procedure for Lagrangian Relaxation based heuristics.

Step 1: (Initialization) Set $n = 0$; Perform dynamic network loading using DTALite with initial path of each agent; initialize the Lagrangian multiplier $\lambda^{(n)}$, evaluate the objective function value $L^{(n)}$ of Lagrangian Relaxation problem (17).

Go to Step 2.

Step 2: (Solve the relaxed model)

Given Lagrangian multiplier $\lambda^{(n)}$ and current results from network loading, solve the problem (19) using a modified label correcting algorithm for time dependent least cost path problem, and Algorithm DEC, to find current optimal value for vector y.

Update the objective function value $L^{(n)}$ of Lagrangian Relaxation problem (17).

Go to Step 3.

Step 3: (Convergence checking) If $L^{(n)} - L^{(n-1)} < \varepsilon$ or $n > N_{\max}$, terminate the iteration and output current solution of vector y. Otherwise, go to step 4.

Step 4: Compute subgradient with Eq. (22) and update Lagrangian multiplier using Eq. (23). Go to step 2.

5. Numerical examples

5.1. Simplified network

The first set of numerical experiments considers a simple transportation network shown in Fig. 8. This simple network can reflect many real world conditions, where, for example, most of the travelers are taking an Interstate to a major city while another US route can serve as an alternative. In this case, we need to figure out information strategy when an incident occurs on the Interstate.

The travel time on each link is assumed to be time-invariant, shown in Table 3. Node 5 is origin and node 6 and 7 are destinations. We have two path demands in this network. They are path 5-1-2-4-6 (with 8 and 50 agents departing at time stamp 1 for two small examples) and path 5-1-2-3-7 (with 8 and 50 agents departing at time stamp 1). Link (5,1), (4,6) and (3,7) are virtual links associated with origins/destinations with travel time of 0 and infinite capacity. Note that we are using point queue model for network flow representation and we consider link entering capacity, the limit of agents entering into a link at a certain time.

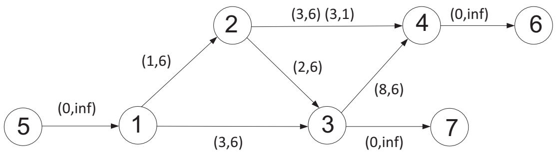

Fig. 8. Simplified network as an example.

Table 3
Travel and entering capacity of each link.

Link	Travel time	Capacity (8 agent)	Capacity (100 agent)
(5,1)	0	inf	inf
(1,2)	1	6	70
(1,3)	3	6	70
(2,3)	2	6	70
(2,4)	3	6 (0 at time 2–10 for scenario 2)	70 (1 at time 2–10 for scenario 2)
(3,4)	8	6	70
(4,6)	0	inf	inf
(3,7)	0	inf	inf

Two scenarios are tested with low demand case (8 agents) using the simplified network. The first is base case scenario as in Table 1, while the second is incident scenario, where the capacity of link (2,4) reduced to 1 from time 2 to 10. The mixed integer program is solved using GAMS optimization software on the Windows 7.0 platform with a personal computer with a 2.90 GHz CPU and 4 GB memory. The computation time for the 8 agent incident scenario is less than 0.2 s, and the computation time for the 8 agent incident scenario) about 15 s. Based on the calculation, all the agents in the base case scenario followed historical path and no information is provided, while in the incident scenario, all x values are zero except $x_{23} = 2$, indicating two information units are provided at node 2 and time stamp 3 (equivalent to that two travelers is provided with information at node 2 and time stamp 3 in this model). The z variables of value 1 are as follows: z^2_{2335}, z^4_{2335}, $z^2_{345,13}$, $z^4_{345,13}$, $z^2_{46,13,20}$ and $z^4_{46,13,20}$. These variables indicate that agents 2 and 4 are provided information at node 2 at time 3 and then follows the suggested route in the information until the destination node 6. Since the model has an "Information activation constraints" (Eq. (6)), which prevent agents being converted back to historical information travelers, the agent will follow suggested route once agent is provided with the information until their destination. Also, since "Flow balance on destination node constraints" sums over leaving time s as in Eq. (2), the last link travel time will always be from the time the agent entering the virtual link to end of time horizon. However, objective function (Eq. (13)) does not include travel time over the virtual links.

Sensitivity analyses are also conducted against parameter information budget B using incident scenarios, as in Table 4 and Fig. 9. In the 8-agent incident scenario, the objective function value stops to decrease when we provide 4 units of information, which means 4 units of information will be enough to make the system perform best under the specific incident scenario. In reality, if we provide more than four units of information, over-diversion will be caused and thus the network capacity is not fully used. The same result is obtained from the 100 agent incident scenario, where 45 units of information will be adequate and best for traffic management. In both cases, the optimum number of information units is less than one half of the total agent amounts, and this further exemplifies the importance of providing detailed information only to a selective proportion of travelers, instead of a one-for-all solution such as roadside DMS.

It is noted that there is fluctuation of objective function values when the budget value is more than 45, while in reality the value should stay constant. It is due to the solution algorithm of Branch and Cut used by Cplex to solve the proposed Mixed Integer Program. The algorithm keeps on searching for optimal solution until current solution satisfies tolerances. For relatively larger examples containing large amount of agents, this fluctuation will be less obvious for the same algorithm tolerance value.

Table 4
Sensitivity analysis of information budget.

8-agent incident scenario										
B	0	1	2	3	4	5	6	7	8	
Obj	64	63	62	61	60	60	60	60	60	
100-agent incident scenario										
B	0	5	10	15	20	25	30	35	40	45
Obj	755	750	745	740	735	730	732	720	715	714
B	55	60	65	70	75	80	85	90	95	100
Obj	729	726	723	734	717	720	714	714	717	714

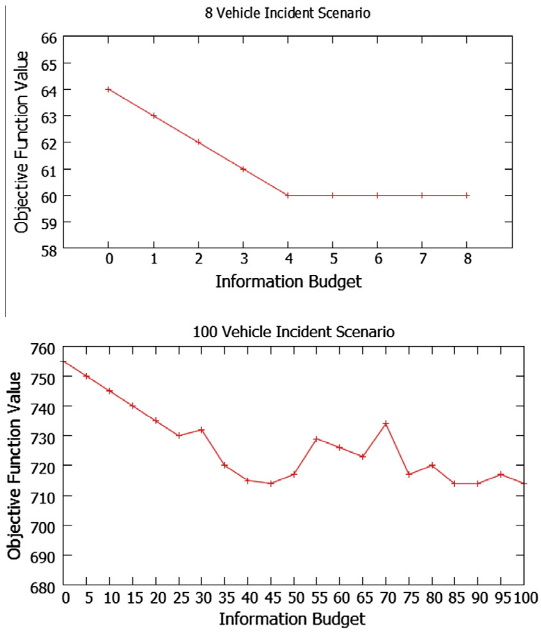

Fig. 9. Sensitivity analysis of information budget for both 8 and 100 agents.

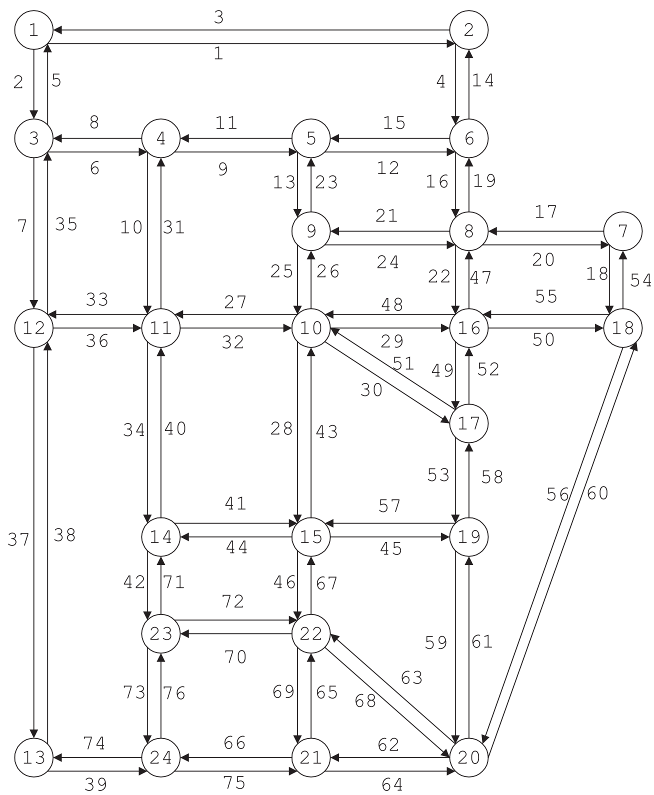

Fig. 10. Simplified Sioux Falls network.

5.2. Median scale network

In the second set of experiments, we consider a simplified Sioux Falls network as shown in Fig. 10. The sample problem is that agents travel from node 1 to node 20 and the free flow travel time of each link is 1 time units. The total time horizon is 20 time units. Table 5 shows the Problem size and solution time of models with different number of agents.

5.3. Large scale network

The next numerical experiment is performed on a real-world subarea network within the Portland, Oregon metropolitan area, which includes 858 nodes, 2000 links, and 208 OD zones shown in Fig. 11 (KAI, 2010; Jia et al., 2011). There are a total of 212,000 trips/agents for a 4-h typical evening peak hour period. The experiments were performed on a PC with 16 GB memory and 8-core processors running at 2.70 GHz. We implemented the proposed Lagrangian Relaxation-based heuristics in the open-source dynamic traffic assignment package DTALite. As the DTA simulator is implemented in C++, which can provide more efficient algorithms to simulate the traffic congestion as a dynamic network loading program. Each iteration of simulation run takes about 3 min CPU time.

Clearly, the effectiveness of the Lagrangian Relaxation based lower bound rule depends on many settings in subgradient updating schemes and shortest path algorithms. In this study, we first fine-tune all the related algorithm parameters through five randomly generated seeds in the same network, and apply the best settings for the remaining experiments. The quality of lower bounds, in this paper, is measured by the percentage gap between a lower bound estimate and the corresponding optimal value for the total system-wide travel times.

We first construct a base line case without incident, and simulate traffic conditions with a major incident on Highway 26, also known as the Sunset Highway, which forms the northern boundary of the study area. Fig. 12 shows the solution quality of our proposed lower bound compared to the best available upper bound generated through the Lagrangian Relaxation iterative process. As expected, the Lagrangian Relaxation based lower bound shown in Eq. (19) iteratively increases the estimation value, and in general marginal improvements become insignificant after 5 iterations. After 20 iterations, the maximum achievable Lagrangian lower bound is 330.7 K, which is equivalent to additional travel delay of 1.55 min due to incidents for all travelers. To quickly generate a feasible solution, this research users the DTALite to simulate the optimized routing plan. The resulting solution offers a feasible solution with an upper bound of the optimal total system-wide (additional) travel delay at 340 K minutes, corresponding to a very small relative solution quality gap of (340–330.7 K)/330.7 K = 2.8%.

Table 5
Problem size and solution time of models with different number of agents.

	10 agents	100 agents	300 agents
Problem size	30 Mb	373 Mb	2238 MB
# of variables	38,901	384,681	1,153,081
# of equations	51,974	498,104	1,489,504
# of non-zero elements	301,219	5,757,109	35,629,309
Solution time/Cplex time (s)	0.31	12.23	448.27

Fig. 11. Portland Beaverton subarea network adapted from SHRP II C05 project, rectangles on the right represent sensor locations for demand calibration.

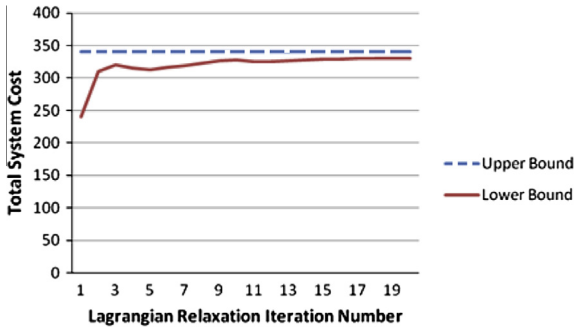

Fig. 12. Upper bound and lower bound series of optimal solution for Lagrangian Relaxation problem, total system cost unit: additional delay (1000 min) compared to the base line condition without incident.

6. Conclusions and future research

In order to develop advanced personalized information strategies for dynamic traffic management, this paper proposed an agent-based optimization framework to consider heterogeneous travelers in terms of characteristics such as behavior, origin–destination and historical routes. The information strategy includes elements, such as where and when to provide the information, to whom the information is given, and what alternative route information should be suggested.

An agent-based optimization formulation with multiple origin–destination pairs (mixed integer programming) is first built on a space–time expanded network and it can be solved efficiently using off-the-shelf commercial solvers for small scale network. In addition, a Lagrangian Relaxation-based heuristic solution approach, combined with a subgradient method and moving average step size update, is then introduced and the agent-based model to solve medium to large networks via the use of a mesoscopic traffic simulator.

Future research will focus on the following aspects. The proposed model can be further extended and incorporated in a multi-period rolling horizon approach. We can also consider robust information provision by considering variables such as random incident duration and compliance rate, by using techniques such as stochastic programming. The current model simply minimizes total system travel time, while future models can could be multi-objective, such as considering minimizing environmental pollution.

Acknowledgements

We would like to thank our colleagues Jay Przybyla and Jeffrey Taylor for their constructive comments. The Portland sub-area network data set is made available through TRB SHRPII C05 project titled "Understanding the Contribution of Operations, Technology, and Design to Meeting Highway Capacity Needs". The work presented in this paper remains the sole responsibility of the authors.

References

Abbas, M., McCoy, P., 1999. Optimizing variable message sign locations on freeways using genetic algorithms. In: 78th Annual Meeting of the Transportation Research Board, Washington, DC.
Adler, J.L., Blue, V.J., 2002. A cooperative multi-agent transportation management and route guidance system. Transport. Res. Part C: Emerg. Technol. 10 (5), 433–454.
Bolelli, A., Mauro, V., Perono, E., 1991. Models and strategies for dynamic route guidance. Part B, a decentralized, fully dynamic, infrastructure supported route guidance. In: DRIVE Conference: Advanced Telematics in Road Transport, Brussels, Belgium.
Carey, M., Subrahmanian, E., 2000. An approach to modelling time-varying flows on congested networks. Transport. Res. B-Meth 34 (3), 157–183.
Charbonnier, C., Farges, J.L., Henry, J.J., 1991. Models and strategies for dynamic route guidance – Part C: optimal control approach. In: Proceedings of the DRIVE Conference, Brussels, Belgium, pp. 106–112.
Chen, X.M., Zhang, L., He, X., Xiong, C., Li, Z., 2014. Surrogate-Based Optimization of Expensive-to-Evaluate Objective for Optimal Highway Toll Charges in Transportation Network. Comput-Aided. Civ. Inf 29 (5), 359–381.
Chiu, Y.C., Huynh, N., 2007. Location configuration design for Dynamic Message Signs under stochastic incident and ATIS scenarios. Transport. Res. Part C: Emerg. Technol. 15 (1), 33–50.
Daganzo, C.F., 1994. The cell transmission model: a dynamic representation of highway traffic consistent with the hydrodynamic theory. Transport. Res. Part B 28 (4), 269–287.
Daganzo, C.F., 1995. The cell transmission model, Part II: Network traffic. Transport. Res. Part B 29 (2), 79–93.
Davidsson, P., Persson, J.A., Holmgren, J., 2007. On the integration of agent-based and mathematical optimization techniques. In: Agent and Multi-agent Systems: Technologies and Applications. Springer, Berlin, Heidelberg, pp. 1–10.
Dia, H., 2002. An agent-based approach to modelling driver route choice behaviour under the influence of real-time information. Transport. Res. Part C: Emerg. Technol. 10 (5), 331–349.
Gaffuri, J., 2006. How to merge optimization and agent-based techniques in a single generalization model. In: Workshop on Generalisation and Multiple Representation, Vancouver, United States.
Ghali, M.O., Smith, M.J., 1995. A model for the dynamic system optimum traffic assignment problem. Transport. Res. Part B: Method. 29 (3), 155–170.

Gjerdrum, J., Shah, N., Papageorgiou, L.G., 2001. A combined optimization and agent-based approach to supply chain modelling and performance assessment. Prod. Plann. Control 12 (1), 81–88.

Hawas, Y.E., 2012. A cooperative distributed system for real-time route guidance. J. Transport. Technol. 2 (03), 230.

Hawas, Y., Mahmassani, H.S., 1995. A decentralized scheme for real-time route guidance in vehicular traffic networks. In: Steps Forward. Intelligent Transport Systems World Congress (No. vol. 4).

Huynh, N., Chiu, Y.C., Mahmassani, H.S., 2003. Finding near-optimal locations for variable message signs for real-time network traffic management. Transport. Res. Rec.: J. Transport. Res. Board 1856 (1), 34–53.

Jahn, O., Möhring, R.H., Schulz, A.S., Stier-Moses, N.E., 2005. System-optimal routing of traffic flows with user constraints in networks with congestion. Oper. Res. 53 (4), 600–616.

Jayakrishnan, R., Mahmassani, H.S., Hu, T.Y., 1994. An evaluation tool for advanced traffic information and management systems in urban networks. Transport. Res. Part C: Emerg. Technol. 2 (3), 129–147.

Jia, A., Zhou, X., Li, M., Rouphail, N.M., Williams, B.M., 2011. Incorporating stochastic road capacity into day-to-day traffic simulation and traveler learning framework. Transport. Res. Rec.: J. Transport. Res. Board 2254 (1), 112–121.

Kittelson & Associates, Inc., 2010. Understanding the Contributions of Operations, Technology, and Design to Meeting Highway Capacity Needs. Transportation Research Board, Washington, DC. <http://onlinepubs.trb.org/onlinepubs/shrp2/SHRP2_S2-C05-RW-1.pdf>.

Li, M., Zhou, X., Rouphail, N.M., 2011. Quantifying benefits of traffic information provision under stochastic demand and capacity conditions: a multi-day traffic equilibrium approach. In: IEEE Intelligent Transportation Systems Conference (ITSC), Washington, DC, 2011.

Lu, C.C., Mahmassani, H.S., Zhou, X., 2008. A bi-criterion dynamic user equilibrium traffic assignment model and solution algorithm for evaluating dynamic road pricing strategies. Transport. Res. Part C: Emerg. Technol. 16 (4), 371–389.

Lu, C.C., Mahmassani, H.S., Zhou, X., 2009. Equivalent gap function-based reformulation and solution algorithm for the dynamic user equilibrium problem. Transport. Res. Part B: Method. 43 (3), 345–364.

Lu, C.C., Zhou, X., Zhang, K., 2013. Dynamic origin–destination demand flow estimation under congested traffic conditions. Transport. Res. Part C: Emerg. Technol. 34, 16–37.

Mahmassani, H.S., Jayakrishnan, R., 1999. System performance and user response under real-time information in a congested traffic corridor. Transport. Res. Part A 25 (5), 293–308.

Newell, G.F., 1993. A simplified theory on kinematic waves in highway traffic, Part I: General theory. Transport. Res. Part B 27 (4), 281–287.

Paz, A., Peeta, S., 2009. Behavior-consistent real-time traffic routing under information provision. Transport. Res. Part C: Emerg. Technol. 17 (6), 642–661.

Peeta, S., Gedela, S., 2001. Real-time variable message sign-based route guidance consistent with driver behavior. Transport. Res. Rec.: J. Transport. Res. Board 1752 (1), 117–125.

Peeta, S., Mahmassani, H.S., 1995. Multiple user classes real-time traffic assignment for on-line operations: a rolling horizon solution framework. Transport. Res. 3C (2), 83–98.

Peeta, S., Ramos, J.L., Pasupathy, R., 2000. Content of variable message signs and on-line driver behavior. Transport. Res. Rec.: J. Transport. Res. Board 1725 (1), 102–108.

Persson, J.A., Davidsson, P., Johansson, S.J., Wernstedt, F., 2005. Combining Agent-Based Approaches and Classical Optimization Techniques. EUMAS, pp. 260–269.

Sawaya, O.B., Doan, D.L., Ziliaskopoulos, A.K., Fourer, R., 2001. Multistage stochastic system optimum dynamic traffic assignment program with recourse for incident traffic management. Transport. Res. Rec.: J. Transport. Res. Board 1748 (1), 116–124.

Vickrey, W., 1963. Pricing in urban and suburban transport. Am. Econ. Rev.: Pap. Proc. 53 (2), 452–465.

Williams, C.A., 2010. A Data Mining Approach to Rapidly Learning Traveler Activity Patterns for Mobile Applications. University of Illinois at Chicago.

Xu, T.D., Hao, Y., Peng, Z.R., Sun, L.J., 2012. Real-time travel time predictor for route guidance consistent with driver behavior. Can. J. Civ. Eng. 39 (10), 1113–1124.

Yang, L., Zhou, X., 2014. Constraint reformulation and a Lagrangian relaxation-based solution algorithm for a least expected time path problem. Transport. Res. Part B: Method. 59, 22–44.

Zhang, L., Levinson, D., 2004. Agent-based approach to travel demand modeling: exploratory analysis. Transport. Res. Rec.: J. Transport. Res. Board 1898 (1), 28–36.

Zhang, L., Zhu, S., Levinson, D., 2008. Agent-based modeling of price competition, capacity choice, and product differentiation in congested networks. J. Transp. Econ. Policy 42 (3), 435–461.

Zhou, X., Taylor, J., 2014. DTALite: a queue-based mesoscopic traffic simulator for fast model evaluation and calibration. Cogent Eng. 1 (1), 1–19.

Zhou, X., Mahmassani, H.S., Zhang, K., 2008. Dynamic micro-assignment modeling approach for integrated multimodal urban corridor management. Transport. Res. Part C: Emerg. Technol. 16 (2), 167–186.

Zhu, S., Levinson, D.M., Zhang, L., 2008. Agent-based route choice with learning and exchange of information. In: Transportation Research Board 87th Annual Meeting (No. 08-2152).

Ziliaskopoulos, A.K., 2000. A linear programming model for the single destination system optimum dynamic traffic assignment problem. Transport. Sci. 34 (1), 37–49.

Ziliaskopoulos, A., Mahmassani, H.S., 1992. Design and implementation of a shortest path algorithm with time-dependent arc costs. In: Proceedings of the 5th Advanced Technology Conference, Washington, DC, pp. 221–242.