Schedule-based transit assignment with online bus arrival information

Pramesh Kumar[*,a,b], Alireza Khani[a]

[a] *Department of Civil, Environmental and Geo-Engineering, University of Minnesota, Twin Cities, MN, U.S.A.*
[b] *Department of Civil Engineering, Indian Institute of Technology Madras, Chennai, TN, India*

Abstract

Most public transportation services deviate from their published schedule. To cope with the delay caused by the unreliable service, passengers use online information about the bus arrival time which affects their route choice behavior. Current schedule-based transit assignment models fail to capture the passengers' adaptive response to unreliable service, resulting in an inaccurate estimation of passenger wait time and passenger loads on various transit routes. The current study proposes schedule-based transit assignment models that incorporate online bus arrival information when modeling the passenger route choice in a stochastic and time-dependent transit network. The authors propose that passengers employ strategies when traveling between different origin-destination pairs not only due to the limited capacity of vehicles but also to cope with the transit delay. The passenger routing problem is modeled as a Markov Decision Process, and efficient algorithms are developed to solve this problem. Depending on the vehicle capacity, two types of assignment models are presented, namely, uncapacitated and capacitated assignments. When penalties for arriving at the destination outside the desired arrival time window are not applied, the uncapacitated assignment problem is formulated as a linear program. On the other hand, the capacitated assignment is formulated as a variational inequality problem for which an efficient Method of Successive Averages-based heuristic solution algorithm is proposed. Computational experiments are presented for a small and a large schedule-based transit network. The results show that denied boarding in an unreliable network leads to higher expected costs to passengers compared to the reliable and uncongested network. Furthermore, the analysis shows that the strategies evaluated with reliable schedule assumption lead to unreliable paths in the network and produces more transferring flow than should happen in practice. The application of our method to a subnetwork of the Twin Cities transit network with artificial demand reveals that passengers traveling from a residential area to the University of Minnesota campus may prefer taking a path with transfer in the event of highly unreliable transit service on the direct routes.

Keywords: schedule-based, transit assignment, online information, congestion, equilibrium, variational inequality

---

[*]Corresponding author
[1]Email: kumar372@umn.edu
[2]Tel: (612) 461-1643
[3]Web: http://umntransit.weebly.com/

1. Introduction

Transit network assignment is a discipline whereby network models are conceptualized, designed, and calibrated to reflect the system-side and user-side behavior within a transit network. There are two classes of transit assignment models, namely, frequency-based (FB) (Spiess and Florian 1989) and schedule-based (SB) (Hamdouch and Lawphongpanich 2008) models. The FB models assume a static representation of a transit network and are useful for long-term planning operations such as frequency design, etc. On the other hand, SB models assume a dynamic representation of a transit network and are useful for short-term planning operations such as time-tabling, vehicle scheduling, etc. The SB assignment is the topic of the current study.

Current SB assignment models assume the timely arrival of buses at stops. However, in reality, bus travel time is subject to uncertainty due to road congestion (since buses use the same right of way as cars), traffic signals, inclement weather, varying dwell times, and maintenance disruptions. This uncertainty causes early/late arrival of buses at stops, which results in the possibility of missing transfers by passengers flowing in the network. For example, in Minneapolis-St. Paul, on a typical day, around 10% of transfers failed due to either early/late transit arrival at stops during peak-hours (Kumar and Khani 2019). Moreover, to avoid extra waiting time caused by early/late arrival of buses, it is common for passengers to use online information about the bus arrival time at different stops to make adaptive decisions *en-route* in this stochastic and dynamic system (Webb et al. 2020). For example, Islam and Fonzone 2021 surveyed passengers in Edinburgh, UK and found that more than 56% passengers used real-time bus arrival information, half of which changed at least one aspect of their trip, including changing their departure time from the origin (29%), boarding time (21%), boarding stop (13%), bus route (15%), and alighting stop (5%). Similar findings were also reported by Fonzone 2015. A literature review on the benefits of providing real-time bus arrival information by Brakewood and Watkins 2019 reveals a reduction in passenger wait times, a decrease in overall travel time due to changes in path choice, and increased satisfaction with public transit usage and security. However, the current SB assignment models fail to capture the adaptive passenger response to unreliable service, which causes an inaccurate estimation of passenger wait time and passenger loads on various transit routes (Fonzone and Schmöcker 2014).

The current research proposes SB transit assignment models that incorporates online bus arrival information when modeling the passenger route choice in a stochastic and time-dependent (STD) transit network. We propose that passengers employ strategies when traveling between an origin-destination pair. These strategies are not only motivated by the limited capacity of vehicles (as in the current SB assignment models) but also due to the use of online information for making adaptive decisions *en-route* in the STD transit network. We formulate the route choice problem as a *stochastic shortest path* (SSP) problem whose solution give us optimal strategy/policy that describe the adaptive passenger behavior in the network. The passenger assignment uses these strategies to predict flow on various transit routes in the network. Based on the capacity limits of transit vehicles, the current study proposes two types of assignment models, namely, uncapacitated and capacitated assignment models. The uncapacitated assignment assumes the unlimited capacity of vehicles, whereas capacitated assignment enforces the capacity of transit vehicles when assigning passengers. Through this research, we envisage a more realistic SB transit assignment.

2. Related work

Transit assignment has attracted a lot of attention since the 1970s, and various models have emerged over the years. Chriqui and Robillard 1975 posed the decision problem, also known as *common lines problem*, faced by a passenger traveling between two stops served by several transit routes. Spiess and Florian 1989 proposed that passengers adopt *strategies* when traveling, which is defined as a set of rules that help a passenger to move from an origin to a destination in a transit network. They proposed the first FB transit assignment model formulated as a linear program. Further, Nguyen and Pallottino 1988 formalized any strategy as a sub-network between two nodes in the transit network, known as a *hyperpath* and proposed a greedy algorithm to find the shortest hyperpaths in the network. It was soon realized that current models could not predict passenger behavior in congested FB networks. Therefore, several approaches are proposed by the researchers to model congestion in the network. This includes asymmetric BPR-type function of waiting by Wu et al. 1994 and De Cea and Fernández 1993, effective frequency function by Cominetti and Correa 2001 and Cepeda et al. 2006, and failure-to-board probabilities by Kurauchi et al. 2003. A multi-modal FB assignment model was proposed by Kumar and Khani 2022. The FB models consider single vehicle runs, which results in an approximation of accurate vehicle loads for a time-dependent transit service (Nuzzolo et al. 2012). Therefore, *schedule-based or dynamic transit assignment models* emerged in the literature. Nguyen et al. 2001 presented a graph-theoretic framework for the SB transit network, Tong and Wong 1999 proposed a SB transit assignment model based on the schedule-based transit shortest path algorithm developed by Tong and Richardson 1984, and Poon et al. 2004 proposed a simulation-based assignment model with FIFO queuing discipline. To model congestion into SB models, studies have used a BPR-type discomfort function for in-vehicle links (Crisalli 1970, Nielsen 2004, Nuzzolo et al. 2001). The main drawback of this approach is that discomfort is applied to all the passengers in a bus (both seating and standing), and the assignment results may not satisfy the strict capacity of transit vehicles. Hamdouch et al. 2004 and Hamdouch and Lawphongpanich 2008 proposed that passengers adopt strategies in a SB network when competing with other passengers for limited vehicle capacity. They proposed an assignment model based on the User Equilibrium principle. The logit-based strategic SB transit assignment was proposed by Nuzzolo et al. 2012, Noh et al. 2012, and Khani et al. 2015. Various studies have also used strategy-based models for the capacitated traffic assignment problem (Marcotte et al. 2004, Zimmermann et al. 2021). As seating and standing passengers have different comfort costs, SB assignment models have also incorporated the effect of discomfort on strategies (Sumalee et al. 2009, Hamdouch et al. 2011, Binder et al. 2017).

There has been a significant amount of work on the adaptive traveler routing problem in a road network with random travel times. In this problem, the traveler has to make online choices based on the sum of current costs and future expected costs. Hall 1986 showed that the online shortest path in a network with random travel times cannot be found using the conventional shortest path methods and requires the computation of adaptive decision rules. Therefore, efficient algorithms have been developed for various variants of this problem based on different assumptions on modeling the stochasticity in travel time. This includes recursive algorithms by Polychronopoulos and Tsitsiklis 1996, Andreatta and Romeo 1988 and Gao et al. 2008, label setting algorithms by Miller-Hooks and Mahmassani 2003, label correcting algorithms by Cheung 1998 and Waller and Ziliaskopoulos 2002, a primal-dual algorithm by Provan 2003, and value iteration, linear programming, and policy iteration algorithms by Polychronopoulos and Tsitsiklis 1996 and Kumar and Khani 2021. It is common for passengers to use adaptive decision rules for navigating in the transit network. A passenger waiting at a bus stop served by several bus routes employs strategies to minimize the travel cost (Chriqui and Robillard 1975, Spiess and Florian 1989). The strategies are affected by the online information, and various studies have proposed FB assignment models incorporating online information (Gentile et al. 2005, Billi et al. 2004, Chen and Nie 2015, Oliker and Bekhor 2018). In the case of SB assignment, Hamdouch et al. 2014 developed an assignment model that incorporates the passengers' response to unreliable service by finding the strategies that minimize the sum of mean and variance of overall travel cost. Zhang et al. 2010 models the risk-taking behavior of passengers in SB networks with random arc travel time using chance constraints. Gardner et al. 2021 presented an estimation method for evaluating passenger travel time distributions in unreliable transit networks using phase-type distributed Markov chains. Rambha et al. 2016 formulated the transit shortest path problem with online information as a finite horizon Markov Decision Process and presented several procedures based on variants of the time-dependent shortest path problem to decrease the computational time of evaluating routing strategies. Hickman and Wilson 1995 and Hickman and Bernstein 1997 proposed path choice models for modeling passenger behavior of declining a bus route in favor of a faster bus route arriving at a stop based on online information. Khani 2019 proposed an efficient labeling algorithm to find strategies in a trip-based dynamic transit network considering the reliability of transfers. Other approaches include model-free reinforcement learning-based SB assignment model by Wahba and Shalaby 2009, and simulation-based models incorporating real-time information by Nuzzolo et al. 2016 and Cats and Gkioulou 2017.

The above studies have made valuable contributions to modeling the effect of real-time information on transit passenger routing. However, several gaps motivate us to pursue the current research. They are discussed below:

1. A common approach for modeling passenger response to unreliable transit service is through evaluating strategies with the least mean-variance cost. However, this approach does not model the complexity associated with missing transfers. If buses are late, passengers miss

transfers and might take alternative bus routes. Therefore, the shortest path with recourse problem in SB transit networks needs to be solved (Hall 1986, Andreatta and Romeo 1988).

2. There is no efficient method that solves the current problem in a reasonable amount of time and is scalable for large-scale SB transit networks.

To address the above issues, we develop a transit assignment model that employs strategies to describe the online routing behavior of passengers in a network with random bus arrival times at various stops and a limited capacity of vehicles. For this purpose, we use the traffic equilibrium framework proposed by Marcotte et al. 2004 and Baillon and Cominetti 2008 that formulates passenger route choice as a sequential decision-making problem. As their framework only deals with uncertainties associated with link availability and passenger perception of travel time in a static auto network, the current research proposes a generalized model that incorporates uncertainties due to both unreliable bus service and limited vehicle capacity in a time-dependent transit network. We define a stochastic shortest path (SSP) framework with a state incorporating the passenger's current location, time, and information about the bus arrival time and capacity of vehicles to evaluate their strategy. The structure of the problem allows us to solve the SSP efficiently. We pose the capacitated assignment problem as a variational inequality problem for which an efficient MSA-based solution heuristic is proposed that runs SSP and a dynamic network loading algorithm to reach an equilibrium solution. Unlike previous studies on schedule-based transit assignment that maintains the flow vector based on a specific strategy for an origin-destination pair, the current research maintains the link flow vector based on local choice probabilities for each destination. During the passenger assignment phase, it takes the convex combination of local probabilities rather than shifting the flow from various strategies of an origin-destination pair to the shortest expected cost strategy. In this way, we do not have to maintain the list of active strategies (Hamdouch et al. 2014). This difference is akin to a path-based versus link-based algorithm for solving the traffic assignment problem.

The rest of the article is structured as follows. The next section (Section 3) introduces the notations and concepts used throughout the paper. After that, Section 4 formulates the assignment problem for uncapacitated networks, which is followed by the capacitated assignment model in Section 5. Then, numerical experiments are presented in Section 6. Finally, the conclusions and directions for future research are discussed in Section 7.

3. Preliminaries

We start by discussing the creation of a SB (read as "schedule-based") transit network and introducing a few notations that we use throughout the paper. A SB transit network is characterized by a digraph $G(N, A)$, where N denotes the set of nodes and A denotes the set of links in the network. We use the trip-based representation of the transit network (Khani et al. 2014), which is created using the General Transit Feed Specification data (Google 2005). The probability distributions of link travel times are calibrated using Automatic Vehicle Location (AVL) data, which provides historical bus arrival times at various stops recorded using GPS devices installed in transit vehicles (Riter and McCoy 1977).

For the dynamic representation of the network, let us consider T as the set of integer-valued time intervals during the study period. In transit schedule data, we denote the set of transit stops/stations as $\mathfrak{B}$, set of bus routes as R, and set of transit trips as K. Each trip $k \in K$ is characterized by a bus route $r_k \in R$, set of nodes[1] $B_k \subset \mathfrak{B} \times K$, sequence $\gamma_k : B_k \mapsto \mathbb{N}$ in which various stops are visited, scheduled arrival/departure time $\hat{t}_k : B_k \mapsto T$ at various stops in the itinerary, and a set of possible (actual) arrival time at those stops $\tilde{t}_k : B_k \mapsto 2^T$, which is obtained from the AVL data. The probability of a bus associated to trip k arriving at node $i \in B_k$ at time $t \in \tilde{t}_k(i)$ is denoted by $\tilde{p}_i(t)$. For a well-defined probability distribution, we must have $\tilde{p}_i(t) \geq 0, \forall t \in \tilde{t}_k(i), \forall i \in B_k, \forall k \in K$ and $\sum_{t \in \tilde{t}_k(i)} \tilde{p}_i(t) = 1, \forall i \in B_k, \forall k \in K$. Overall, the set of nodes in the network $N = O \cup D \cup B$ can be partitioned into three subsets, namely, the set of origins O (from where passenger trips start), the set of destinations D (where passenger trip ends), and the set of transit nodes $B = \cup_{k \in K} B_k$. Let $k(i)$ and $r(i)$ be the trip and route resp. associated to transit node $i \in B$ and $w : N \times N \mapsto \mathbb{R}$ be the walking time between two nodes in the network. The passenger demand is assumed to be distributed among groups G. Each group of passengers $g \in G$ is characterized by an origin $o_g \in O$, a destination $d_g \in D$, the earliest departure time from the origin t_g^{ED}, the earliest arrival time at the destination t_g^{EA}, and the latest arrival time at the destination t_g^{LA}. Let $\{d_g^{od}\}_{(o,d) \in O \times D, g \in G}$ be the demand matrix from origin to destination for different groups. There are three types of links $A = A_a \cup A_v \cup A_t$ in the network, namely, the access/egress links A_a, the in-vehicle links A_v, and the walking/waiting transfer links A_t. The access/egress links are used to access/egress transit nodes in the network, i.e., $A_a = \{(i,j) \in O \times B \mid w(i,j) \leq \delta_0\} \cup \{(i,j) \in B \times D \mid w(i,j) \leq \delta_1\}$, where δ_0 and δ_1 are the acceptable walking times to access and egress a transit stop. The in-vehicle links are transit vehicle links created using the itinerary of a transit trip, i.e., $A_v = \{(i,j) \in B \times B \mid k(i) = k(j), \gamma_{k(j)}(j) = \gamma_{k(i)}(i) + 1\}$. Finally, the waiting and walking transfer links are created between two nodes $i, j \in B$ if they satisfy the following conditions:

1. Routes associated to both nodes are different, i.e., $r_{k(i)} \neq r_{k(j)}$.
2. The stop associated to node i is not the first stop in $k(i)$'s itinerary, i.e., $\gamma_{k(i)}(i) \neq 1$ and the stop associated to node j is not the last stop in $k(j)$'s itinerary, i.e., $\gamma_{k(j)}(j) \neq \max_{l \in B_{k(j)}} \gamma_{k(j)}(l)$.
3. Walking time between i and j is less than or equal to an acceptable walking time limit δ_2, i.e., $w(i,j) \leq \delta_2$.

[1]Here, a node is characterized by a stop and bus trip serving it.

Let us denote $A_t^{'}$ as the set of links that satisfy the above conditions. Sometimes, condition 3 needs to be relaxed when a bus (for transfer) is not available at stop i (e.g., after midnight) or so late that walking to the respective destination becomes a better choice. The above conditions create unlikely transfer links, which can be further reduced using some criterion specific to the type of assignment (capacitated or uncapacitated). The criteria are based on maximum acceptable wait time δ_3 and probability of making a successful transfer. δ_3 is the maximum waiting time that a passenger is willing to spend to access transit service. Moreover, we assume that if the waiting time exceeds δ_3 for a passenger, then the optimal choice for the passenger is to walk to her destination. The value of δ_3 should not be too low to exclude any reasonable choice and should not be too high to create enormous transfer links. The passenger survey data can help us calibrate this value. To avoid interruption in the main focus of this article, we continue this discussion in Appendix A, where a pseudo code is provided to obtain the final transfer links. Let us denote the forward and backward stars of node $i \in N$ as $FS(i)$ and $BS(i)$ respectively. Before describing the model, we make the following assumptions:

3.1. Assumptions

1. The vehicle arrival and departure times at stops are assumed to be the same, i.e., no dwell time is assumed.

2. The walking time on access, egress, and walking transfer links is integer-valued and constant.

3. The travel and wait time associated with in-vehicle and transfer links respectively are assumed to be time-varying discrete random variables with finite support.

4. The travel time or wait time on various links is assumed to be independent across time periods. Further, the travel time is assumed to be independent across trips and routes, and bus bunching is ignored.

5. Passengers are provided with online information about the bus arrival time at every node of the network.

6. At a node, passengers use online information about only those bus routes which are accessible from that node by an acceptable walking distance.

7. The online information about the bus arrivals provided to any passenger is one of the realizations obtained from the historical data.

8. Passengers decide which bus route to board as soon as they arrive at a particular node.

9. Passengers are expected-cost minimizers. The "optimal" policy/strategy minimizes the expected cost of traveling between an origin-destination pair.

10. The walking, waiting, and in-vehicle travel times are equally weighed in the cost to passengers.

11. Passengers can coordinate the departure time from their origin based on the online information about the bus arrival at various stops to avoid waiting time. Therefore, no latest departure time penalty is assumed.

12. The transit network is connected.

13. The assignment models compute the average flow of passengers on every link of the schedule-based transit network.

Some of the assumptions are non-restrictive and can be easily relaxed. Assumption 1 can be relaxed by including dwell time in the cost of in-vehicle links. Assumption 5 is also not a concern as nodes without online information can have average deterministic costs for outgoing links. Assumptions 4 and 6 are needed to avoid enormous state space in the stochastic shortest path problem. The correlations in travel time can be considered by assuming realizations of the travel time on every link in the network. However, the corresponding stochastic shortest path problem is NP-hard (e.g., see Polychronopoulos and Tsitsiklis 1996, Provan 2003). One can relax Assumption 6 by including online information about all the bus routes in the state space and developing a solution algorithm that evaluates more intelligent routing policies in the network at the expense of computational time (e.g., see Rambha et al. 2016). However, such algorithms are more suited for providing routing policies to passengers through cellphone applications. For assignment purposes, we believe this is a reasonable assumption and can aid in developing faster algorithms. Since the assignment uses historical bus arrival data to calibrate the link travel time distributions, it is necessary to state Assumption 7 because the model would not be able to explain the adaptive response of passengers for the bus arrival times realizations other than what is recorded in the historical data. The relaxation of Assumption 8 would require formulating a dynamic path choice problem similar to Hickman and Bernstein 1997, which we leave for us to explore in future work. Assumptions 2-3, 7, and 9-12 are required for modeling purposes. Finally, the Assumption 13 reveals the objective of the passenger assignment in the current study. It computes the average passenger flow on each link of the SB transit network rather than the flow for a particular realization of the network. For example, if we know the state of the network on a particular day, then we can perform the assignment of passengers in a deterministic fashion (Paulsen et al. 2021). We believe that an average flow of passengers computed based on the historical states of the network will aid in evaluating the long-term congestion in the network.

3.2. Characterization of online information

The random bus arrival time at various nodes induces a node-dependent stochasticity as when a passenger arrives at node $i \in N$ at time $t \in T$, an online information vector θ is revealed to them. This information vector consists of travel cost $\{c_{ij}^{\theta}\}_{j \in FS(i)}$ of outgoing links from node i (Gao and Huang 2012, Boyles and Rambha 2016). Let $\Theta_i(t)$ be the set of possible information sets at node i and time t. For an information vector $\theta \in \Theta_i(t)$ associated to the head node i of a transfer or access link, the travel cost of a transfer link (i, j) for a possible arrival of bus at node $j \in FS(i)$ at $t_j \in \tilde{t}_{k(j)}(j)$ is given as:

$$c_{ij}^{\theta} = \begin{cases} t_j - t, & \text{if } t + w_{ij} \le t_j \\ \infty, & \text{otherwise} \end{cases} \tag{1}$$

The probability of observing $\theta \in \Theta_i(t)$ is denoted by p^{θ}. For this probability distribution, we must have, $p^{\theta} \ge 0, \forall \theta \in \Theta_i(t), \forall t \in \tilde{t}_{k(i)}(i), \forall i \in N$ and $\sum_{\theta \in \Theta_i(t)} p^{\theta} = 1, \forall t \in \tilde{t}_{k(i)}(i), \forall i \in N$. Let us denote $\Theta = \cup_{i \in N} \cup_{t \in T} \Theta_i(t)$ as the union of all node-time information sets.

3.3. Example

To better understand the problem setting, let us consider an illustrative example provided in Figure 1. It shows two trips $K = \{1, 2\}$ of different transit routes going from stop A to C and E to C respectively. There are three in-vehicle nodes $B = \{A_1, B_1, C_1, E_2, D_2, C_2\}$, one origin node $O = \{o\}$, and one destination node $D = \{d\}$. There are four in-vehicle links $A_v = \{(A_1, B_1), (B_1, C_1), (E_2, D_2), (D_2, C_2)\}$, four access/egress links $A_a = \{(o, A_1), (o, E_2), (C_1, d), (C_2, d)\}$, and one transfer link $A_t = \{(B_1, D_2)\}$. The random link travel time of in-vehicle links or walk time of access/egress/transfer links is shown by the links in the figure. Assume that buses of trips 1 and 2 begin their journey at their commencing stop at time $t = 0$. Then, the possible arrival times of different trips at different nodes with their probabilities are given as: $\tilde{t}_1(A_1) = 0 \ w.p. \ 1.0$, $\tilde{t}_2(E_2) = 0 \ w.p. \ 1.0$, $\tilde{t}_1(B_1) = \left\{\begin{matrix} 2, & w.p. \ 0.6 \\ 8, & w.p. \ 0.4 \end{matrix}\right\}$ (which means that trip 1 arrives at node B_1 at time 2 with probability 0.6 and at time 8 with probability 0.4), $\tilde{t}_2(D_2) = \left\{\begin{matrix} 3, & w.p. \ 0.2 \\ 5, & w.p. \ 0.3 \\ 10, & w.p. \ 0.5 \end{matrix}\right\}$, $\tilde{t}_1(C_1) = \left\{\begin{matrix} 17, & w.p. \ 0.6 \\ 23, & w.p. \ 0.4 \end{matrix}\right\}$, $\tilde{t}_2(C_2) = \left\{\begin{matrix} 16, & w.p. \ 0.2 \\ 18, & w.p. \ 0.3 \\ 23, & w.p. \ 0.5 \end{matrix}\right\}$, where $w.p.$ means *with probability*. Using the arrival time information, the set of online information vectors at various nodes can be written as: $\Theta_o(0) = \begin{bmatrix} \{0, 0\}, & w.p. \ 1.0 \end{bmatrix}$, $\Theta_{A_1}(0) = \begin{bmatrix} \{2\}, & w.p. \ 0.6 \\ \{8\}, & w.p. \ 0.4 \end{bmatrix}$, $\Theta_{E_2}(0) = \begin{bmatrix} \{3\}, & w.p. \ 0.2 \\ \{5\}, & w.p. \ 0.3 \\ \{10\}, & w.p. \ 0.5 \end{bmatrix}$. If the trip 1 arrived at B_1 at $t = 2$, then possible information a traveler can receive are $\Theta_{B_1}(2) = \begin{bmatrix} \{15, 1\}, & w.p. \ 0.2 \\ \{15, 3\}, & w.p. \ 0.3 \\ \{15, 8\}, & w.p. \ 0.5 \end{bmatrix}$. This is because if trip 1 arrives at time 2, the cost of outgoing link (B_1, C_1) will always be 15, however, the cost of link (B_1, D_2) depends on the time trip 2 arrives at D_2, which is 1 $w.p.$ 0.2 , $5 - 2$ $w.p.$ 0.3, and $10 - 2$ $w.p.$ 0.5. Similarly, $\Theta_{B_1}(8) = \begin{bmatrix} \{15, \infty\}, & w.p. \ 0.5 \\ \{15, 2\}, & w.p. \ 0.5 \end{bmatrix}$, $\Theta_{D_2}(3) = \begin{bmatrix} \{13\}, & w.p. \ 1.0 \end{bmatrix}$, $\Theta_{D_2}(5) = \begin{bmatrix} \{13\}, & w.p. \ 1.0 \end{bmatrix}$, and $\Theta_{D_2}(10) = \begin{bmatrix} \{13\}, & w.p. \ 1.0 \end{bmatrix}$. At C_1 and C_2, for any possible arrival time, the information set will have a single link with cost of

1 *w.p.* 1.0.

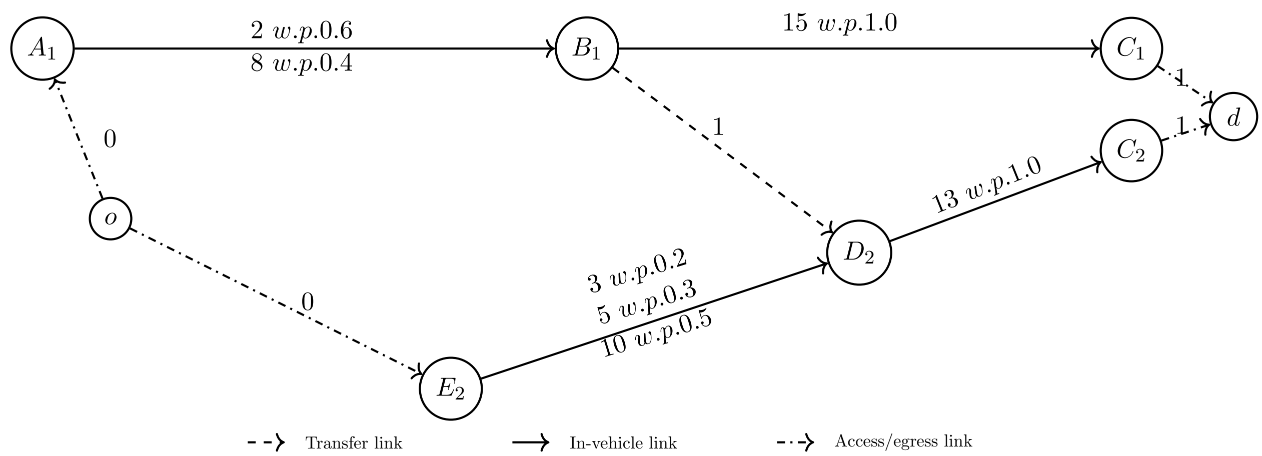


Figure 1: An illustrative example to show the stochastic transit network

In what follows, we present two different assignment models, namely, uncapacitated and capacitated assignment models. For each case, we describe algorithms for evaluating optimal strategies and assigning passengers to the network.

4. Uncapaciated assignment

In this case, we assume that transit vehicles have unlimited capacity. This assignment model is applicable for transit systems with low ridership and for which denied boarding due to limited capacity is a rare phenomenon.

4.1. Hyperpaths

Hall 1986 showed that the least expected cost route in a stochastic and time-dependent transit network is not a "simple" route but a "strategy/policy" in which arcs are selected based on an adaptive rule. Such policy can be evaluated based on the online information about bus arrival time and helps passengers make a cleverer choice and improve their overall journey time. For example, in case of missed transfers, a passenger can consider an alternative route in their policy that provides the least expected cost to her destination. A policy induces a subgraph in the network known as *hyperpath*. A hyperpath, commonly used in FB models, is a collection of paths in the network that passenger travels on with positive probability. The current problem of finding an optimal hyperpath is formulated as the stochastic shortest path (SSP) problem (Bertsekas 2012). These various components characterizing the SSP for a specific destination $d \in D$ in case of the uncapacitated assignment are described below:

1. *State space*: We use the state space description that has been used for static and dynamic stochastic shortest path problems by various researchers in the past (Andreatta and Romeo

1988, Polychronopoulos and Tsitsiklis 1996, Gao et al. 2008, Boyles and Rambha 2016, Kumar and Khani 2021). The state space $S \subseteq N \times T \times \Theta$ describe the possible positions of a passenger in space and time and bus arrival information. Each state $s \in S$ associated to transit node is characterized by a tuple $s = (i, t, \theta)$, where $i \in B$ represents the node in the network, $t \in \tilde{t}_{k(i)}(i)$ represents the possible arrival time at node i, and $\theta \in \Theta_i(t)$ represents the online information about the cost of links in $FS(i)$ obtained at node i and time t. A state s associated to origin node is characterized by $s = (o, t, \theta)$, where $o \in O$ is the origin node, $t \in T$ is the possible departure time from the origin, and $\theta \in \Theta_o(t)$ is the online information vector about the cost of outgoing links $FS(o)$ at time t. We also consider one special state known as the destination state d, which is an absorbing state.

2. *Action space*: When the passenger arrives at a node, they consider the current travel cost and the online information about the travel cost on downstream links to move forward. For example, at every transfer node, a passenger receives information about the wait time of transferring nodes and whether a transfer is missed. Then, she has to decide which available action to take next. Therefore, the set of actions for each state (i, t, θ) are given by $u(i, t, \theta) = \{j \in FS(i) : c^{\theta}_{ij} \neq \infty\}$ i.e., the set of outgoing links. A stationary policy $\mu : S \mapsto N$ assigns the action to each state. Here, $\mu(s) \in u(s), \forall s \in S$.

3. *Transition Function*: The transition function $\mathbb{P}_\mu : S \times S \mapsto \mathbb{R}$ corresponding to policy μ is defined as $\mathbb{P}_\mu[(i, t, \theta), (\mu(i, t, \theta), t + c^{\theta}_{i\mu(i,t,\theta)}, \theta')] = p^{\theta'}, \theta' \in \Theta_{\mu(i,t,\theta)}(t + c_{i\mu(i,t,\theta)}), \theta \in \Theta_i(t), t \in T, i \in N$. The probability of transitioning from d to itself, by taking any action is 1.

4. *One-step costs*: The cost of choosing a link (action) $j = \mu(i, t, \theta)$ at state (i, t, θ) is denoted by c^{θ}_{ij}, where $\theta \in \Theta_i(t)$.

5. *Expected cost function*: Let $J : S \mapsto \mathbb{R}$ be the expected cost function representing the expected cost incurred by a passenger to reach her destination from a possible state.

The value of optimal cost function J^* can be obtained by solving the Bellman equation (2) (Bertsekas 2012).

$$J^*(i, t, \theta) = \min_{j \in u(i,t,\theta)} \{c^{\theta}_{ij} + \sum_{\theta' \in \Theta_j(t + c^{\theta}_{ij})} p^{\theta'} J^*(j, t + c^{\theta}_{ij}, \theta')\}, \forall (i, t, \theta) \in S \tag{2}$$

4.2. Finding the optimal policy

It becomes challenging to solve the Bellman equation (2) when the state space is large. To solve it efficiently, we reduce the size of the state space by averaging the uncontrollable components. In this case, θ is an uncontrollable component of the state space that does not depend on the action taken by the passenger. Therefore, we formulate the problem only on the space of node and time, i.e., $\hat{S} = (N \times T) \cup \{d\}$ with online information vector θ being averaged out.

$$\hat{J}^*(i,t) = \sum_{\theta \in \Theta_i(t)} p^\theta J^*(i,t,\theta) \tag{3}$$

$$= \sum_{\theta \in \Theta_i(t)} p^\theta \min_{j \in u(i,t,\theta)} \{c_{ij}^\theta + \sum_{\theta' \in \Theta_j(t + c_{ij}^\theta)} p^{\theta'} J^*(j, t + c_{ij}^\theta, \theta')\} \tag{4}$$

$$\hat{J}^*(i,t) = \sum_{\theta \in \Theta_i(t)} p^\theta \min_{j \in u(i,t,\theta)} \{c_{ij}^\theta + \hat{J}^*(j, t + c_{ij}^\theta)\}, \forall (i,t) \in N \times T \tag{5}$$

The standard methods designed for solving SSP, such as the value iteration (VI), policy iteration (PI), etc. can be used for solving the Bellman equation (5). For example, the worst-case complexity of running the value iteration algorithm is $\mathcal{O}(|\hat{S}||N||A|)$ (Kumar and Khani 2021). However, the structure of the current problem allows us to solve the Bellman equation using a more efficient label correcting algorithm (Cheung 1998). Kumar and Khani 2021 proved that for any stationary policy μ, the associated transition graph is acyclic. The acyclicity property allows us to develop a label correcting algorithm for finding the optimal policy, the steps of which are summarized in Algorithm 1. The worst-case time computational complexity of running Algorithm 1 is $\mathcal{O}(|\hat{S}||A|)$ (Kumar and Khani 2021).

Algorithm 1 Label correcting algorithm for uncapacitated assignment

```
procedure ULC(d)                                               ▷ Input: destination d
    (Initialize) Ĵ(i,t) ← ∞, ∀(i,t) ∈ Ŝ\{d} and Ĵ(d) ← 0
    SE ← BS(d)                                                  ▷ Scan Eligible List
    while SE ≠ φ do
        Remove an element i from SE
        for t ∈ t̃_{k(i)}(i) do
            tempJ ← 0
            for θ ∈ Θ_i(t) do
                tempJ += p^θ min_{j∈u(i,t,θ)} {c^θ_ij + Ĵ(j, t + c^θ_ij)}
            if tempJ < Ĵ(i,t) then
                Ĵ(i,t) ← tempJ; SE ← SE ∪ BS(i)
    μ*(i,t,θ) ← argmin_{j∈u(i,t,θ)} {c^θ_ij + Ĵ*(j, t + c^θ_ij)}, ∀(i,t,θ) ∈ S\{d}   ▷ Computing optimal policy
    return Ĵ*, μ*
```

The algorithm is similar to the one used for the deterministic shortest path, but in this case, we compare the average costs to deal with the uncertainty. It starts by initializing the expected cost associated with various states as ∞, except the destination state, for which it is assumed to be 0. The scan eligible list SE is initialized as a list containing the neighbors of the destination node. Then, the algorithm scans elements in the backward direction updating the label of every node for every time interval. It computes a temporary label *tempJ* (Lines 7-9) and checks if it is less than the current expected cost $\hat{J}(i,t)$ (Line 10). Then, it possibly updates the expected cost of the state. After scanning all the states and computing their optimal expected cost, it evaluates the optimal policy for a given destination d (Line 12). Finally, the algorithm returns the optimal expected costs J^* and optimal policy μ^*.

4.3. Assignment of passengers

The computation of optimal policy and expected costs are performed for individual destinations. After this, for every group of passengers, we still need to figure out the optimal departure time from their origin. We assume that passengers are rational and select the departure time, which provides them the least expected cost to their destination. We present two different approaches to finding the optimal departure time:

1. *If early and late arrival penalties are included*: In this case, penalties are used to avoid arriving outside the desired travel time window. For example, commuters want to arrive at their destination within a certain time frame. We consider two different types of penalties, i.e., early (η_1 / time units) and late (η_2 / time units) arrival time penalties. Based on these penalties, the optimal departure t_g^* for group $g \in G$ is given as:

$$t_g^* \in \underset{t_g^{ED} \leq t \leq t_g^{ED}+\delta_3}{\operatorname{argmin}} \{\hat{J}^*(o_g,t)+\eta_1 * \max(0, t_g^{EA}-(t+\hat{J}^*(o_g,t)))+\eta_2 \max(0,(t+\hat{J}^*(o_g,t))-t_g^{LA})\} \tag{6}$$

   In equation (6), t_g^* is searched within the time interval $\left[t_g^{ED}, t_g^{ED}+\delta_3\right]$ for the least expected cost with associated penalties. The passenger assignment, in this case, can performed using Algorithm 2.

2. *If early and late arrival penalties are not included*: In this case, optimal departure t_g^* for group $g \in G$ is given as:

$$t_g^* \in \underset{t_g^{ED} \leq t \leq t_g^{ED}+\delta_3}{\operatorname{argmin}} \{\hat{J}^*(o_g,t)\} \tag{7}$$

   In equation (7), t_g^* is searched within the time interval $\left[t_g^{ED}, t_g^{ED}+\delta_3\right]$ for the least expected cost, where δ_3 is the maximum acceptable wait time. This case is applicable for trips such as shopping when passengers want to get to their destination in the least amount of time. The passenger assignment, in this case, can be performed using Algorithm 2 or the linear program (8).

If we do not include penalties for passenger arrival outside the desired window, then we can derive a linear program for the assignment of passengers on optimal policies. To do so, let us denote $v^d(i,t,\theta,j)$ be the number of passengers arriving at state $(i,t,\theta) \in S$ and choosing control $j \in u(i,t,\theta)$. Furthermore, let V_{gt}^d be the number of passengers from group g, departing at time

$t \in \left[t_g^{ED}, t_g^{ED} + \delta_3\right]$ from their origin o_g to destination $d \in D$. Then, we have the following assignment LP:

$$\min_{\mathbf{V},\mathbf{v}} \quad \sum_{d \in D} \sum_{(i,t,\theta) \in S} \sum_{j \in u(i,t,\theta)} v^d(i,t,\theta,j) c_{ij}^{\theta} \tag{8a}$$

$$\text{s.t.} \quad \sum_{j \in u(i,t,\theta)} v^d(i,t,\theta,j) - p^{\theta} \sum_{\substack{(k,t',\theta') \in S \setminus \{d\}: i \in u(k,t',\theta') \\ \& \ t = t' + c_{ki}^{\theta'}}} v^d(k,t',\theta',i) = 0, \forall \theta \in \Theta_i(t), \forall t \in \tilde{t}_{k(i)}(i), \forall i \in N, \forall d \in D \tag{8b}$$

$$\sum_{j \in u(o,t,\theta)} v^d(o,t,\theta,j) - p^{\theta} \sum_{\substack{g \in G: o_g = o \ \& \\ t \in \left[t_g^{ED}, t_g^{ED} + \delta_3\right]}} V_{gt}^d = 0, \forall \theta \in \Theta_o(t), \forall t \in T, \forall o \in O, \forall d \in D \tag{8c}$$

$$\sum_{t \in \left[t_g^{ED}, t_g^{ED} + \delta_3\right]} V_{gt}^d = d_g^{o_g d}, \forall g \in G : d_g = d, \forall d \in D \tag{8d}$$

$$\sum_{(k,t',\theta') \in S \setminus \{d\}: d \in u(k,t',\theta')} v^d(k,t',\theta',d) = \sum_{g \in G: d_g = d} d^{o_g d}, \forall d \in D \tag{8e}$$

$$v^d(i,t,\theta,j) \geq 0, \forall j \in u(i,t,\theta), \forall (i,t,\theta) \in S, \forall d \in D \tag{8f}$$

$$V_{gt}^d \geq 0, \forall t \in \left[t_g^{ED}, t_g^{ED} + \delta_3\right], \forall g \in G, \forall d \in D \tag{8g}$$

In the above optimization program (8), we minimize the total expected travel time given by (8a). Constraints (8b)-(8e) describe the conservation of flow for every destination. For any state, (8b) shows that the sum of passenger flow going out of state (i,t,θ) is equal to the portion of the sum of flow coming into it from other states at the time t and observing θ. (8c) describes the conservation constraint for states associated with origin nodes, i.e., the sum of passenger flow going out of origin state (o,t,θ) is equal to the sum of passenger flow from different groups that have the same origin o departing at time t and experiencing the real-time information θ. Equation (8d) describes that the total sum of flow from a group at different departure times to their destination d should be equal to the demand associated with that group. Then, for every destination $d \in D$, the total flow coming into the destination state should be equal to the total demand of groups going to d. Finally, (8f)-(8g) represent the non-negativity constraints for the flow variables.

Lemma 1. *The optimal solution of* (8) *assigns passenger flow to the optimal policy corresponding to each destination.*

Proof. See Appendix B □

The dual variables $J^d(i,t,\theta)$ of (8b)-(8c) represent the optimal cost to go from state (i,t,θ) to d. Similarly, the dual variables $J^d(g)$ of (8d) represent the optimal cost incurred by group g to go from its origin to destination. In fact, the dual program of (8) is the linear programming formulation for solving the Bellman equations (2) and (7). One of the advantages of assignment LP

(8) is that it is decomposable for every destination $d \in D$ and side constraints related to the flow of passengers can be used in this formulation. However, the number of variables can still be large due to the cardinality of state space. Therefore, it is much easier to use Algorithm 2 presented in the next paragraph for sole assignment purposes.

Algorithm 2 starts with the initialization of the transitioning flows $v^d(i,t,\theta,j)$ as 0. Then, for every destination $d \in D$, it computes the optimal cost functions $\hat{J}^{d*}$ and optimal policy μ^{d*} using Algorithm 1. After that, for every group, we find the optimal departure time(s) from their origin using (6) or (7). Then, for every possible real-time information vector a group g could receive for a given optimal departure time $t \in t_g^*$, we assign a portion of the group demand to the transitioning flow variable $v^d(o_g,t,\theta,j)$. We repeat this for every group. After this, the total demand has originated in the transition graph corresponding to μ^{d*}. Using the topological order of nodes and processing the times in chronological order, we assign the transitioning flow using Line 18. Note that after assignment, we can calculate the average flow on a link $(i,j) \in A$ as below:

$$v_{ij} = \sum_{d \in D} \sum_{t \in \tilde{t}_{k(i)}(i)} \sum_{\theta \in \Theta_i(t)} v^d(i,t,\theta,j), \forall (i,j) \in A \tag{9}$$

Algorithm 2 Uncapacitated transit assignment

(Initialization) $v^d(i,t,\theta,j) \leftarrow 0, \forall j \in u(i,t,\theta), \forall(i,t,\theta) \in S, \forall d \in D$
for every $d \in D$ **do**
  $\hat{J}^{d*}, \mu^{d*} \leftarrow ULC(d)$ ▷ Computing optimal expected costs and policy for destination d
  if arrival penalties are included **then**
    $t_g^* = \underset{t_g^{ED} \le t \le t_g^{ED}+\delta_3}{\operatorname{argmin}} \{\hat{J}^*(o_g,t) + \eta_1 * \max(0, t_g^{EA} - (t + \hat{J}^*(o_g,t))) + \eta_2 \max(0, (t + \hat{J}^*(o_g,t)) - t_g^{LA})\}$
  else
    $t_g^* = \underset{t_g^{ED} \le t \le t_g^{ED}+\delta_3}{\operatorname{argmin}} \{\hat{J}^*(o_g,t)\}$
  for every $g \in G : d_g = d$ **do**
    for $t \in t_g^*$ **do** ▷ Optimal departure times
      for $\theta \in \Theta_{o_g}(t)$ **do** ▷ Possible wait time information
        $j \leftarrow \mu^{d*}(o_g,t,\theta)$
        $v^d(o_g,t,\theta,j) \mathrel{+}= p^\theta * d_g^{o_g d} * \frac{1}{|t_g^*|}$ ▷ Passengers observing state (o_g,t,θ) and taking action j
  Find the topological order of nodes
  for $i \in$ topological order **do**:
    for $t \in \tilde{t}_{k(i)}(i)$ in chronological order **do**
      for $\theta \in \Theta_i(t)$ **do**
        $j \leftarrow \mu^{d*}(i,t,\theta)$ ▷ Optimal action at state (i,t,θ) for destination d
        $$v^d(i,t,\theta,j) \mathrel{+}= p^\theta \left(\sum_{\substack{(k,t',\theta') \in S \setminus \{d\} : \mu^{d*}(k,t',\theta') = i \\ \& \ t = t' + c_{ki}^{\theta'}}} v^d(k,t',\theta',i) \right)$$

An example problem for the uncapacitated assignment is solved in Appendix C.

5. Capacitated Assignment

The uncapacitated assignment model may produce unrealistic passenger flows on various transit routes. This is because of the limited capacity of vehicles, due to which some arcs may become saturated and cannot be accessed by some passengers depending on how other passengers make their route choice. If we assume that passengers mingle at nodes and have an equal probability to access outgoing links, then one can include the following capacity constraint in the assignment program (8) to produce capacity-feasible flows:

$$\sum_{d \in D} \sum_{t \in \tilde{t}_{k(i)}(i)} \sum_{\theta \in \Theta_i(t)} v^d(i,t,\theta,j) \le u_a, \forall (i,j) \in A_v \tag{10}$$

where, u_a is the capacity associated with transit vehicle used to serve link $a \in A_v$. However, doing so would just limit the number of passengers on each arc without explaining the strategic behavior induced by failure-to-board a congested route. Moreover, passengers on-board have continuance priority over other passengers. The above constraint would not be able to capture such behavior. To model such behavior, previous studies have proposed to use *failure-to-board* probabilities or *access* probabilities. They evaluate the probability with which a passenger waiting at a bus stop can access an outgoing link. Such access probabilities result in multiple paths that a traveler can take with positive probability. The collection of such paths is known as "hyperpath." In the capacitated assignment, the hyperpaths/strategies are induced by both risks of denied boarding due to limited capacity and missing transfers due to unreliable service. A strategy helps passengers minimize their expected costs under various types of uncertainties. When passengers employ strategies to move between various origin-destination pairs and compete for the limited capacity, the strategic equilibrium occurs when no passenger can improve her expected cost by unilaterally switching to a different strategy.

5.1. Hyperpaths

To incorporate the access/availability probabilities and find a strategy that minimizes the expected cost of travel in a capacitated network, we require augmenting the state space. For that purpose, let us define $X_i^\theta(t)$ as the random variable supported on $\{0,1\}^{|u(i,t,\theta)|}$ indicating the availability of arcs in $FS(i)$, when arriving at node $i \in N \backslash \{d\}$ at time $t \in T$, and receiving information θ. To be more precise, the component j of vector $x \in X_i^\theta(t)$ will indicate whether link $(i,j) \in A$ is available to board or not. Let π^x be the probability of observing the availability vector $x \in X_i^\theta(t)$ and $X = \cup_{(i,t,\theta)\in S} X_i^\theta(t)$ be the collection of such availability vectors. The use of π^x is akin to "access" probabilities in the previous literature, as the former describes the node-based availability of outgoing links and the later describes the availability of individual links. It is assumed that passengers do not know about the availability vector x and information vector θ in advance and realize them when reaching a particular node at a particular time. To find an optimal strategy/policy in this case, we need to solve the corresponding stochastic shortest path problem. These various components characterizing the SSP for a specific destination $d \in D$ in case of the capacitated assignment are described below:

1. *State space*: The state space $S_C \subseteq N \times T \times \Theta \times X$ describes the possible positions of a passenger in space and time, information about bus arrival, and availability of links. Each state $s \in S_C$ is characterized by a tuple $s = (i, t, \theta, x)$, where $i \in N$ represents the node in the network, t represents the possible arrival time at node i, $\theta \in \Theta_i(t)$ represents the online information about the cost of links in $FS(i)$, and $x \in X_i^\theta(t)$ represents the availability of outgoing links. Similar to the uncapacitated case, destination $d \in D$ is considered as an absorbing state.

2. *Action space*: When the passenger arrives at a node, they consider the current travel cost and future information about the cost and the availability of downstream links to decide which

arc to take next. For example, at every transfer node, the passenger receives information about the wait time of transferring nodes and whether a link is available or not. A link may be unavailable due to missed transfer or the vehicle associated with it being full. Then, she has to decide which available action to take next. Therefore, the set of actions for each state (i,t,θ,x) are given by $u_C(i,t,\theta,x) = \{j \in u(i,t,\theta) : x[j] \neq 0\}$. Note that due to Assumption 12 and the presence of walking links from transfer nodes, there is no state $s = (i,t,\theta,x)$ such that $u_C(i,t,\theta,x) = \phi$.

3. *Policy*: A policy/strategy specifies the subset of actions that can be taken at a state. To be precise, $\mu_C : S_C \mapsto 2^{|\sum_{s\in S_C} u_C(s)|}$ maps every state to a subset of controls that provide equal expected cost to destination.

4. *Transition Functions*: The transition function $\mathbb{P}_\mu : S_C \times S_C \mapsto \mathbb{R}$ corresponding policy μ is defined as $\mathbb{P}_\mu[(i,t,\theta,x),(\mu(i,t,\theta,x), t + c^{\theta}_{i\mu(i,t,\theta,x)}, \theta^{'}, x^{'})] = p^{\theta^{'}} \pi^{x^{'}}$. The probability of transitioning from d to itself, by taking any action $j \in u_C(d)$ is 1. The value of π^x depends on the route choice of other passengers, and it is obtained from the network loading procedure.

Using the components defined above, we can formulate the Bellman equation for finding the optimal strategy as below:

$$J^*_C(i,t,\theta,x) = \min_{j \in u_C(i,t,\theta,x)} \left\{ c^{\theta}_{ij} + \sum_{\theta' \in \Theta_j(t+c^{\theta}_{ij})} \sum_{x' \in X^{\theta}_j(t+c^{\theta}_{ij})} p^{\theta^{'}} \pi^{x^{'}} J^*_C(j, t + c^{\theta}_{ij}, \theta^{'}, x^{'}) \right\}, \forall (i,t,\theta,x) \in S_C \tag{11}$$

where, $J^*_C(i,t,\theta,x)$ denotes the optimal cost-to-go from state (i,t,θ,x) to the destination in case of capacitated assignment. Similar to uncapacitated assignment, one can reduce the state space and define the Bellman equation only on controllable components by averaging the uncontrollable components.

$$\hat{J}^*_C(i,t) = \sum_{\theta' \in \Theta_j(t+c^{\theta}_{ij})} \sum_{x' \in X^{\theta}_j(t+c^{\theta}_{ij})} p^{\theta} \pi^{x} \min_{j \in u_C(i,t,\theta,x)} \left\{ c^{\theta}_{ij} + \hat{J}^*_C(j, t + c^{\theta}_{ij}) \right\}, \forall (i,t) \in \hat{S} \tag{12}$$

The above Bellman equation can also be solved using a label correcting algorithm, the steps which are summarized in Algorithm 3 from Line 1-12. The worst-case complexity remains the same as $\mathcal{O}(|S_C||A|)$. The algorithm starts by initializing the expected cost of various states as ∞, except the destination state, for which it is assumed as 0. The scan eligible list SE is initialized as a list containing the neighbors of the destination node. Then, the algorithm scans elements in the backward direction updating the label of every node for every time interval. It computes a temporary label *tempJ* (Lines 7-10) using both the online information probability p^{θ} and availability probability π^x and checks if it is less than the current expected cost $\hat{J}(i,t)$ (Line 10). Then, it possibly updates the expected cost of the state. After scanning all the nodes and finalizing their expected costs, it evaluates the optimal policy μ_C^* for a given destination d (Line 13). Further, the optimal cost of taking a certain action at any state Q_C^* is evaluated in line 14.

Algorithm 3 Label correcting algorithm for capacitated assignment

1: **procedure** CLC(d) ▷ Input: destination d
2: *(Initialize)* $\hat{J}_C(i,t) \leftarrow \infty, \forall (i,t) \in \hat{S} \backslash \{d\}$ and $\hat{J}(d) \leftarrow 0$
3: $SE \leftarrow BS(d)$
4: **while** $SE \neq \phi$ **do** ▷ Input: Scan Eligible List
5: Remove an element i from SE
6: **for** $t \in \tilde{t}_{k(i)}(i)$ **do**
7: $tempJ \leftarrow 0$
8: **for** $\theta \in \Theta_i(t)$ **do** ▷ Information vector
9: **for** $x \in X_i^\theta(t)$ **do** ▷ Availability vector
10: $tempJ \mathrel{+}= p^\theta \pi^x \min_{j \in u_C(i,t,\theta,x)} \left\{ c_{ij}^\theta + \hat{J}_C(j, t + c_{ij}^\theta) \right\}$
11: **if** $tempJ < \hat{J}_C(i,t)$ **then**
12: $\hat{J}_C(i,t) \leftarrow tempJ$; $SE \leftarrow SE \cup BS(i)$
13: $\mu_C^*(i,t,\theta,x) \leftarrow \underset{j \in u_C(i,t,\theta,x)}{\text{argmin}} \{c_{ij}^\theta + \hat{J}_C^*(j, t + c_{ij}^\theta)\}, \forall (i,t,\theta,x) \in S_C \backslash \{d\}$ ▷ Computing optimal policy
14: $Q_C^*(s,j) \leftarrow c_{ij}^\theta + \hat{J}_C^*(j, t + c_{ij}^\theta), \forall j \in u_C(s), \forall s = (i,t,\theta,x) \in S_C$ ▷ Cost of taking action j at state s
15: $P_{s,j} \leftarrow 1.0/|\mu_C^* s(s)|, \forall j \in \mu^*(s), \forall s = (i,t,\theta,x) \in S_C$ ▷ Probability of taking action j at state s
16: **for** every $g \in G : d_g = d$ **do**
17: **if** arrival penalties are included **then**
18: $t_g^* \leftarrow \underset{t_g^{ED} \leq t \leq t_g^{ED} + \delta_3}{\text{argmin}} \{\hat{J}^*(o_g,t) + \eta_1 * \max(0, t_g^{EA} - (t + \hat{J}^*(o_g,t))) + \eta_2 \max(0, (t + \hat{J}^*(o_g,t)) - t_g^{LA})\}$
19: **else**
20: $t_g^* \leftarrow \underset{t_g^{ED} \leq t \leq t_g^{ED} + \delta_3}{\text{argmin}} \{\hat{J}^*(o_g,t)\}$
21: $R_{g,t} \leftarrow 1.0/|t_g^*|, \forall t \in t_g^*$ ▷ Departure time choice probability
22: **return** $\hat{J}_C^*, Q_C^*, \mu^*, P, R$

The route choice of passengers is characterized by the probability of taking an action at a particular state. Therefore, we introduce $P = \{P_{s,a}^d\}$ as the probability of taking an action $a \in u_C(s), \forall s \in S$, when going to destination $d \in D$ and $R = \{R_{g,t}\}$ as the probability of group $g \in G$ departing at time $t \in \left[t_g^{ED}, t_g^{ED} + \delta_3\right]$. These route choice probabilities are calculated in Algorithm 3 from lines 15-21. We can observe that when there are multiple actions $j \in \mu^*(i,t,\theta,x)$ at state (i,t,θ,x) that achieve optimal expected cost, we assign equal probability to each optimal action. Similarly, if multiple departure times provide the same optimal expected cost for group $g \in G$, we assign equal probabilities to these departure times. However, if there is only one action that achieves minimum, then we assign the probability 1.0 to that action. The use of route choice probabilities allows us to use a flexible choice of selecting actions at various states. For example, one can use the logit-based route choice probabilities.

5.2. Network loading

The computation of optimal policy/strategy for individual destinations reveals the number of passengers using a specific strategy (since we know the number of passengers in a group, their departure time probabilities, and their route choice probabilities). In this section, we describe a NetworkLoading procedure that converts these strategic flows into link flows. The loading of passengers follows some behavioral rules that are described below:

1. At a transfer node, if a passenger according to her strategy decides to continue on the same route r rather than taking a transfer or ending her trip, then that passenger should get the priority over other passengers who either want to transfer to r or begin their journey with r. Such priority is known as *continuance priority* (Hamdouch and Lawphongpanich 2008). At any node, the passengers with continuance priority are loaded first onto the outgoing links.

2. We assume that passengers without continuance priority have equal access to the outgoing links, and they are processed in random and a uniformly distributed single queue. Such loading of passengers is also known as *random loading*. One could try other loading approaches such as *First-Come-First-Serve*, *Regret*, etc. Binder et al. 2017 discusses such exogenous priority rules for transit assignment, which we leave for future research to explore.

The NetworkLoading procedure is summarized in Algorithm 4. It takes passenger route choice probabilities for various destinations $\{\hat{P}^d\}_{d\in D}$ and departure time probabilities for individual groups $\{\hat{R}_g\}_{g\in G}$ as inputs and outputs link flows $\mathbf{v}$ and availability probabilities π. The procedure starts by initializing the state-action passenger flows $\mathbf{v}^d$ for various destinations, node-time priority passenger flows $\mathbf{V}_p$ and non-priority passenger flows $\mathbf{V}_n$. After this, we originate the group flows at various departure times according to the departure time choice probabilities $\hat{\mathbf{R}}$ (Lines 4-6). Then, we process various nodes in topological order to load the passenger demand on outgoing links. For each node $i \in N\backslash\{d\}$, we assume an availability vector, where all the outgoing links $u(i,t,\theta)$ are available, i.e., we assign $\pi^x = 1, \forall x = (i,t,\theta,\{1\}^{|u(i,t,\theta)|}) \in S_C,\ 0$, otherwise. Then, we perform the loading of the demand that reached node i onto outgoing links, which is divided into two phases. In the first phase, we assign the priority flows (Lines 11-23). Depending on the strategy at various states (i,t,θ,x) for destination d, a fraction of flow *tempFlow* is assigned to outgoing link $(i,j) : j \in u_C(i,t,\theta,x)$ according to route choice probabilities $\hat{P}^d_{(i,t,\theta),j}$. Then, a fraction of *tempFlow* is further assigned to node j and transitioning time t' either as priority or non-priority flow depending on the strategy and route choice probabilities. Of course, for the origin nodes, there will be no priority flow to be assigned. The second phase of the loading procedure at node i is the loading of non-priority flows. This loading is performed using a single-queue processing procedure described by Marcotte et al. 2004 and Zimmermann et al. 2021 for static auto networks. We first calculate the residual capacity $\tilde{\mathbf{u}}$ of outgoing links after the loading of priority flows. Then, based on the route choice probabilities, we evaluate $\tilde{\mathbf{v}}$, which describes the number of passengers trying to access outgoing links. The flow trying to access a particular link $\tilde{v}_{ij}$ may exceed the residual capacity $\tilde{u}_{ij}$. Assuming that all the non-priority passengers waiting at that node have an equal probability of accessing an outgoing link, we evaluate the access probability $(\frac{\tilde{u}_{ij}}{\tilde{v}_{ij}})$ of that link. Then, a minimum access probability β of any outgoing link is calculated using the expression given in Line 34. β describes the proportion of passengers that can be loaded before one or more outgoing links get saturated. If the accessing flow of any outgoing link does not exceed its residual capacity, then $\beta = 1$, which means all the waiting passengers can access their optimal choice. For assigning the appropriate number of passengers onto outgoing links, we repeat a similar procedure as priority flows, where some of the passengers reach the outgoing node as a priority and some as non-priority flow. We update the residual capacity and the number of passengers to be loaded at various times $U(i,t)$. If $\beta < 1$, we evaluate the saturated outgoing links, prepare the availability vector, and update its probability π^x using β. The availability probabilities are updated based on the principle that only the β proportion of passengers will observe the current state, and the rest of the passengers $(1-\beta)$ will observe a different state. We continue updating the state availability probabilities in this manner until all the accessing flow is assigned. Note that due to Assumption 12 and the presence of walking links from transfer nodes, we will never observe the availability vector, where all the outgoing links get saturated and are not available. This procedure will evaluate π's, which will be further used in the label correcting algorithm for updating the strategies in the assignment algorithm. An example problem showing the execution of the NetworkLoading algorithm is provided in Appendix C.

Algorithm 4 Network loading

1: **procedure** NetworkLoading($\hat{\mathbf{P}}, \hat{\mathbf{R}}$)
2: $\quad v^d(s,j) \leftarrow 0, \forall j \in u_C(s), \forall s = (i,t,\theta,x) \in S_C, \forall d \in D$
3: $\quad V_p(i,t) \leftarrow 0, V_n(i,t) \leftarrow 0, \forall t \in \tilde{t}_{k(i)}(i), \forall i \in N$
4: $\quad$ **for** $g \in G$ **do**
5: $\quad\quad$ **for** $t \in \left[t_g^{ED}, t_g^{ED} + \delta_3\right]$ **do**
6: $\quad\quad\quad V_n^d(o_g,t) \mathrel{+}= \hat{R}_{g,t} * d_g^{o_g d}$
7: $\quad$ Find the topological order of nodes in N
8: $\quad$ **for** $i \in$ topological order **do**
9: $\quad\quad$ stop$\leftarrow$ FALSE; $U_n(i,t) \leftarrow V_n(i,t); \forall t \in \tilde{t}_{k(i)}(i)$
10: $\quad\quad x_{(i,t,\theta)} \leftarrow \{1\}^{|u(i,t,\theta)|}, \pi^x \leftarrow 1, \text{ if } x = x_{(i,t,\theta)}, 0, \text{otherwise}, \forall (i,t,\theta,x) \in S_C$
11: $\quad\quad$ **for** $d \in D$ **do**
12: $\quad\quad\quad$ **for** $t \in \tilde{t}_{k(i)}(i)$ **do**
13: $\quad\quad\quad\quad$ **for** $\theta \in \Theta_i(t)$ **do**
14: $\quad\quad\quad\quad\quad$ **for** $j \in u_C(i,t,\theta,x_{(i,t,\theta)}) : k(i) == k(j)$ **do**
15: $\quad\quad\quad\quad\quad\quad tempFlow \leftarrow p^\theta * \pi^{x_{(i,t,\theta)}} * \hat{P}^d_{(i,t,\theta,x_{(i,t,\theta)}),j} * V_p^d(i,t);\ t' = t + c_{ij}^\theta$
16: $\quad\quad\quad\quad\quad\quad v^d(i,t,\theta,x_{(i,t,\theta)},j) \mathrel{+}= tempFlow$
17: $\quad\quad\quad\quad\quad\quad$ **for** $\theta' \in \Theta_j(t')$ **do**

$x_{(j,t',\theta')} \leftarrow \{1\}^{|u(j,t',\theta')|}$
for $l \in u_C(j,t',\theta',x_{(j,t',\theta')})$ **do**
  if $k(j) == k(l)$ **then**
    $V_p^d(j,t') \mathrel{+}= p^{\theta'} * \pi^{x_{(j,t',\theta')}} * \hat{P}^d_{(j,t',\theta',x_{(j,t',\theta')}),l} * tempFlow$
  else
    $V_n^d(j,t') \mathrel{+}= p^{\theta'} * \pi^{x_{(j,t',\theta')}} * \hat{P}^d_{(j,t',\theta',x_{(j,t',\theta')}),l} * tempFlow$

for $j \in FS(i)$ **do**
  $\tilde{u}_{ij} \leftarrow \mathfrak{C}(k(j)) - \sum_{d\in D}\sum_{t\in\tilde{t}_{k(i)}(i)}\sum_{x\in X_i^\theta(t)}\sum_{\theta\in\Theta_i(t)} v^d(i,t,\theta,x,j)$

while not stop **do**
  for $d \in D$ **do**
    $\tilde{v}^d_{ij} \leftarrow 0, \forall j \in FS(i), \forall d \in D$
    for $t \in \tilde{t}_{k(i)}(i)$ **do**
      for $\theta \in \Theta_i(t)$ **do**
        for $j \in u_C(i,t,\theta,x_{(i,t,\theta)})$ **do**
          $\tilde{v}^d_{ij} \mathrel{+}= p^\theta * \pi^{x_{(i,t,\theta)}} * \hat{P}^d_{(i,t,\theta,x_{(i,t,\theta)}),j} * U_n(i,t)$

  $\tilde{v}_{ij} = \sum_{d\in D}\tilde{v}^d_{ij}$ ▷ Flow that'll be competing to access (i,j)
  $\beta \leftarrow \min\left\{1, \min_{j\in FS(i)}\left(\frac{\tilde{u}_{ij}}{\tilde{v}_{ij}}\right)\right\}$
  for $j \in FS(i)$ **do**
    $\tilde{u}_{ij} = \tilde{u}_{ij} - \beta\tilde{v}_{ij}$

  for $d \in D$ **do**
    for $t \in \tilde{t}_{k(i)}(i)$ **do**
      for $\theta \in \Theta_i(t)$ **do**
        for $j \in u_C(i,t,\theta,x_{i,t,\theta})$ **do**
          $tempFlow \leftarrow \beta * p^\theta * \pi^{x_{(i,t,\theta)}} * \hat{P}^d_{(i,t,\theta,x_{(i,t,\theta)}),j} * U_n^d(i,t)$; $t' = t + c^\theta_{ij}$
          $v^d(i,t,\theta,x_{(i,t,\theta)},j) \mathrel{+}= tempFlow$
          for $\theta' \in \Theta_j(t')$ **do**
            for $l \in u_C(j,t',\theta',x_{(j,t',\theta')})$ **do**
              if $k(j) == k(l)$ **then**
                $V_p^d(j,t') \mathrel{+}= p^{\theta'} * \pi^{x_{(j,t',\theta')}} * \hat{P}_{(j,t',\theta',x_{(j,t',\theta')}),l} * tempFlow$
              else
                $V_n^d(j,t') \mathrel{+}= p^{\theta'} * \pi^{x_{(j,t',\theta')}} * \hat{P}_{(j,t',\theta',x_{(j,t',\theta')}),l} * tempFlow$

      $U_n^d(i,t) \leftarrow (1-\beta)V_n^d(i,t)$
  if $\beta < 1$ **then**
    for $j' \in \text{argmin}_{j\in FS(i)}\left(\frac{\tilde{u}_{ij}}{\tilde{v}_{ij}}\right)$ **do**
      for $t \in \tilde{t}_{k(i)}(i)$ **do**
        for $\theta \in \Theta_i(t)$ **do**

$p \leftarrow \pi^{x_{(i,t,\theta)}}$
$\pi^{x_{(i,t,\theta)}} \leftarrow \beta p$
$x_{(i,t,\theta)}[j'] \leftarrow 0$
$\pi^{x_{(i,t,\theta)}} \leftarrow (1-\beta)p$
else
stop $\leftarrow$ TRUE
return $\pi, \mathbf{v}$

5.3. *Assignment of passengers*

The optimal strategy computed using Algorithm 3 helps evaluate the route choice $\mathbf{P}$ and departure time choice probabilities $\mathbf{R}$ using the probability of availability vectors π. Then, the NETWORKLOADING procedure in Algorithm 4 will update the values of π based on $\mathbf{P}$ and $\mathbf{R}$. When no passenger can improve their expected cost of travel by altering the probability of taking any action, then the equilibrium is achieved. This means that, in equilibrium, all non-null choice probabilities $P^d_{s,a}$ and $R^d_{g,t}$ associated to a state and group resp. will have the same expected costs $Q^d_{s,a}$ and $\hat{J}^d_{o_g,t}$. To characterize equilibrium, let us define the feasible set of route choice and departure time choice probabilities $\mathfrak{P}$ as below:

$$\mathfrak{P} = \left\{ \mathbf{P} \times \mathbf{R} \in \mathbb{R}^{|D| \times \sum_{s \in S_C} |u_C(s)|} \times \mathbb{R}^{|G| \times |T|} : \sum_{j \in u_C(s)} P^d_{s,j} = 1, \forall s \in S_C, \forall d \in D, \text{ and } \sum_{t \in \left[t^{ED}_g, t^{ED}_g + \delta_3\right]} R_{g,t} = 1, \forall g \in G \right\} \tag{13}$$

Further, the expected cost of choice probability vector $(\mathbf{P}, \mathbf{R})$ denoted by $C(\mathbf{P}, \mathbf{R})$ can be evaluated using the following equation:

$$C(\mathbf{P}, \mathbf{R}) = \sum_{d \in D} \sum_{s \in S_C} \sum_{j \in u_C(s)} Q^d(s,j) \times P^d_{s,j} + \sum_{g \in G} \sum_{t \in \left[t^{ED}_g, t^{ED}_g + \delta_3\right]} \hat{J}^d(o_g, t) \times P_{g,t} \tag{14}$$

where, $Q^d(s,j)$ is the cost of taking action j in state s when going to destination d. We call $(\mathbf{P}^*, \mathbf{R}^*)$ as the equilibrium probabilities if they satisfy the variational inequality given as:

$$\left\langle C(\mathbf{P}^*, \mathbf{R}^*), \begin{Bmatrix} \mathbf{P}^* - \mathbf{P} \\ \mathbf{R}^* - \mathbf{R} \end{Bmatrix} \right\rangle \leq 0, \forall (\mathbf{P}, \mathbf{R}) \in \mathfrak{P} \tag{15}$$

Since the expected cost of mapping $C(\mathbf{P}, \mathbf{R})$ cannot be evaluated in closed form as it depends on the availability probabilities π through the loading procedure, we cannot formulate the above VI problem into an equivalent optimization problem. However, there exists at least one solution to this VI problem because the set $\mathfrak{P}$ is compact, and mapping $C(\mathbf{P}, \mathbf{R})$ is continuous since it depends on the availability probabilities π, which is a function of continuous $(\mathbf{P}, \mathbf{R})$ (Zimmermann et al. 2021). Moreover, we cannot show that there exists a unique solution to the given variational inequality. To solve the assignment problem, we use an MSA-based heuristic approach. We start by initializing the entries of the initial $(\hat{\mathbf{P}}, \hat{\mathbf{R}})$ as zero. Before running the Algorithm 3, we assume that

$\pi^{(i,t,\theta,x)} = 1,$ if $x = \{1\}^{|u(i,t,\theta)|}, 0$ otherwise $, \forall x \in X_i^\theta(t), \forall (i,t,\theta) \in S$. Then, we evaluate the best response choice probabilities $(\mathbf{P}, \mathbf{R})$ using Algorithm 3, which are used for updating the current $(\hat{\mathbf{P}}, \hat{\mathbf{R}})$ based on the values of $(\hat{\mathbf{P}}, \hat{\mathbf{R}})$ and $(\mathbf{P}, \mathbf{R})$ using $\alpha = \frac{1}{k+1}$, where k is the iteration number. Then, the updated $(\hat{\mathbf{P}}, \hat{\mathbf{R}})$ is used for the NetworkLoading procedure that further updates the availability probabilities π. We continue this procedure until the $gap(\hat{\mathbf{P}}, \hat{\mathbf{R}}, \mathbf{P}, \mathbf{R})$ calculated using (16) reaches below the tolerance level ϵ. The gap function is similar to the one used in the traffic assignment studies based on the user equilibrium principle. However, they use link flow vectors but the current study uses the link choice probabilities. The overall MSA algorithm is summarized in Algorithm 5. The converged average link flow values can be calculated using (17).

$$gap(\hat{\mathbf{P}}, \hat{\mathbf{R}}, \mathbf{P}, \mathbf{R}) = \frac{\sum_{d \in D} \sum_{s \in S_C} \sum_{j \in u_C(s)} Q^d(s,j) \times (\hat{P}^d_{s,j} - P^d_{s,j}) + \sum_{g \in G} \sum_{t \in \left[t_g^{ED}, t_g^{ED} + \delta_3\right]} \hat{J}^d(o_g, t) \times (\hat{R}_{g,t} - R_{g,t})}{\sum_{d \in D} \sum_{s \in S_C} \sum_{j \in u_C(s)} Q^d(s,j) \times P^d_{s,j} + \sum_{g \in G} \sum_{t \in \left[t_g^{ED}, t_g^{ED} + \delta_3\right]} J^d(o_g, t) \times R_{g,t}} \tag{16}$$

$$v_{ij} = \sum_{d \in D} \sum_{t \in \tilde{t}_{k(i)}(i)} \sum_{\theta \in \Theta_i(t)} \sum_{x \in X_i^\theta(t)} v^d(i,t,\theta,x,j), \forall (i,j) \in A \tag{17}$$

Algorithm 5 Method of successive averages for capacitated assignment

1: **procedure** MSA(ϵ)
2: *(Initialization)* $\hat{P}^d_{s,j} \leftarrow 0, \forall j \in u_C(s), \forall s \in S_C, \forall d \in D$
3: $\hat{R}_{g,t} \leftarrow 0, \forall t \in \left[t^{ED}, t^{ED} + \delta_3\right], \forall g \in G$
4: $k \leftarrow 0; gap \leftarrow \infty$
5: **while** $gap > \epsilon$ **do**
6: $\alpha = \frac{1}{k+1}$
7: $\hat{J}^d, Q^d, \mu^d, P^d, R^d \leftarrow CLC(d), \forall d \in D$
8: $\hat{\mathbf{P}} \leftarrow \alpha\hat{\mathbf{P}} + (1-\alpha)\hat{\mathbf{P}}$; $\hat{\mathbf{R}} \leftarrow \alpha\hat{\mathbf{R}} + (1-\alpha)\hat{\mathbf{R}}$
9: $\pi, \mathbf{v} \leftarrow$NetworkLoading$(\hat{\mathbf{P}}, \hat{\mathbf{R}})$
10: Calculate gap using the equation (16)
11: $k \leftarrow k+1$
12:

6. Numerical experiments

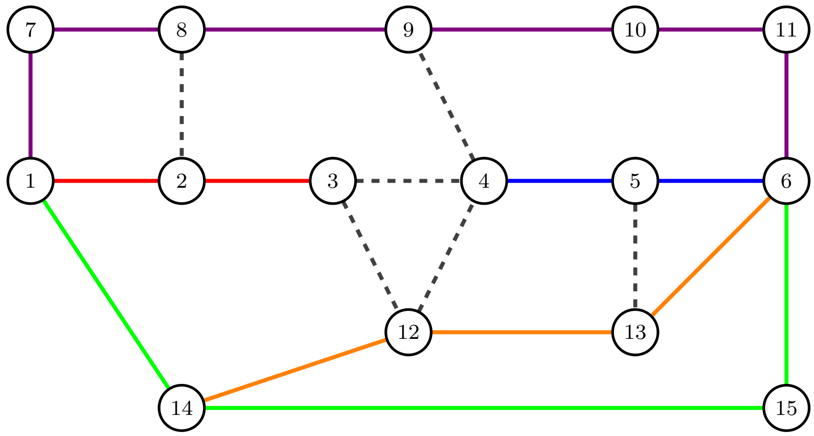
In this section, we show the application of the proposed schedule-based assignment models. For the first experiment, the network, schedule, and demand table are given in Figure 2. There are fifteen stops and five color-coded transit routes in the network. The original network has only three walking transfer links, namely, 3-4, 3-12, and 12-4. To better understand transfer behavior in the presence of online information, we created more walking transfer links in the network. They are given as 2-8, 9-4, and 13-5. Stop 14 is the only stop that provides a waiting transfer from one route to another in the network. There are four trips of each route whose complete schedule is shown in Figure 2(b). There are six origin-destination pairs in the network. A synthetic demand table is created for the assignment, which is shown in Figure 2(c). It has 24 groups with different origins, destinations, earliest departure, and earliest and latest arrival times, with a total demand of 128 passengers. The network has multiple routes, trips, transfers, O-D pairs, and passenger groups, which makes it a suitable candidate for testing our transit assignment models.

The support of random travel times of in-vehicle links is given as $\{0.9\bar{c}_{ij}, \bar{c}_{ij}, 1.2\bar{c}_{ij}, 1.5\bar{c}_{ij}\}$, where $\bar{c}_{ij}$ is the scheduled travel time of link $(i, j) \in A_v$. All trips are assumed to have a capacity of 20 passengers. The early and late arrival penalties are assumed to be $\eta_1 = \eta_2 = 0.5$. The acceptable waiting and walking times are assumed as $\delta_0 = \delta_1 = \delta_2 = \delta_3 = 15$ minutes. Overall, there are 89 nodes and 173 links in the schedule-based transit network. It has 24 access, 20 egress, and 64 in-vehicle links. In what follows, we present the assignment results for the uncapacitated and capacitated transit assignment in separate subsections.

6.1. Uncapacitated assignment

We start by creating the transfer links using Algorithm 6. For the uncapacitated assignment, it creates only four waiting transfer links and twenty five walking transfer links in the current schedule-based network. The number of generated states is 4,720. After this, we solve the Bellman equation (2) for individual destinations. It took a fraction of a second to solve the current problem using both value iteration and label correcting algorithms. Figure 3 shows the expected cost of travel between various origin-destination pairs for varying departure times. We can observe that for various origins, the expected cost to the respective destination decreases with time until we reach the time when a bus trip departs from that origin. Then, similar behavior is observed for the passengers waiting for the next trip to arrive. Further, we see that the average cost to destination 6 is lower than the average cost to destination 5. This is because destination 6 can be reached from various origins without transferring to a different route. On the other hand, to reach destination 5, one must transfer to a different route, which sometimes causes longer expected cost.

We employ the Monte-Carlo simulation to estimate the reliability of optimal paths in the network when the schedule is perfectly reliable. We begin by evaluating the optimal paths between various O-D pairs for a perfectly reliable network. Then, for every O-D pair (o, d) and departure



(a) Network

RouteID	TripID	Schedule
Red	1001	10:00, 10:02, 10:04
	1002	10:10, 10:12, 10:14
	1003	10:20, 10:22, 10:24
	1004	10:30, 10:32, 10:34
Blue	2001	10:06, 10:08, 10:10
	2002	10:16, 10:18, 10:20
	2003	10:26, 10:28, 10:30
	2004	10:36, 10:38, 10:40
Violet	3001	10:00, 10:02, 10:04, 10:06, 10:08, 10:10, 10:12
	3002	10:07, 10:09, 10:11, 10:13, 10:15, 10:17, 10:19
	3003	10:14, 10:16, 10:18, 10:20, 10:22, 10:24, 10:26
	3004	10:21, 10:23, 10:25, 10:27, 10:29, 10:31, 10:33
Orange	4001	10:08, 10:10, 10:12, 10:14
	4002	10:18, 10:20, 10:22, 10:24
	4003	10:28, 10:30, 10:32, 10:34
	4004	10:38, 10:40, 10:42, 10:44
Green	5001	9:55, 9:57, 10:04, 10:06
	5002	10:05, 10:07, 10:14, 10:16
	5003	10:15, 10:17, 10:24, 10:26
	5004	10:25, 10:27, 10:34, 10:36

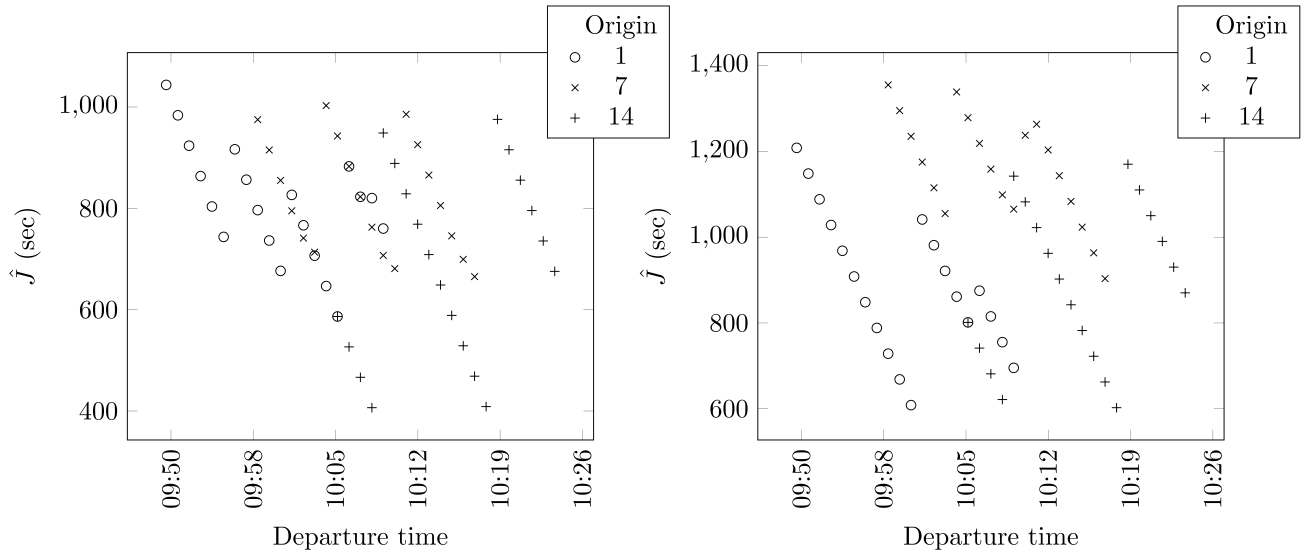
(b) Schedule

Group	Origin	Dest.	t_g^{ED}	t_g^{EA}	t_g^{LA}	Dem.
1	1	6	09:55	10:07	10:12	5
2	1	6	09:55	10:10	10:15	6
3	1	6	09:50	10:04	10:09	4
4	1	6	10:05	10:17	10:22	8
5	1	6	10:05	10:20	10:25	2
6	1	6	10:00	10:14	10:19	2
7	1	6	09:58	10:27	10:32	8
8	1	6	10:00	10:24	10:30	9
9	1	6	10:05	10:37	10:42	11
10	1	5	09:52	10:05	10:10	1
11	1	5	10:05	10:15	10:20	5
12	1	5	09:55	10:25	10:30	4
13	1	5	09:57	10:36	10:40	5
14	7	6	09:58	10:10	10:15	7
15	7	6	10:08	10:20	10:25	2
16	7	6	10:15	10:35	10:40	4
17	7	5	10:13	10:33	10:38	4
18	7	5	10:03	10:18	10:23	7
19	14	6	10:05	10:14	10:20	9
20	14	6	10:13	10:22	10:28	5
21	14	6	10:20	10:36	10:40	8
22	14	5	10:05	10:12	10:18	4
23	14	5	10:10	10:20	10:25	2
24	14	5	10:15	10:30	10:36	6

(c) Demand table

Figure 2: Network, schedule, and demand table (Tong and Richardson 1984)

time interval, we generate 1000 random passenger journeys following policy μ^{d*} starting from o and ending at d. For any (i,t) associated to every journey, θ is drawn from the distribution $\{p^\theta\}_{\theta \in \Theta_i(t)}$. Then, we evaluate the percentage of trajectories that are same as the optimal path corresponding to $(o,d) \in O \times D$ in the perfectly reliable network to calculate the reliability of that path. Table 1 shows the results of the reliability of paths evaluated in the perfectly reliable network. We observe that paths of origin-destination pairs 14-5, 14-6, and 7-6 have reliability greater than 90%. This could be possible because the origin-destination pairs are either directly connected or connected



(a) Destination 6

(b) Destination 5

Figure 3: Expected cost between various origin-desination pairs for varying departure times in case of uncapacitated assignment

with a reliable transfer in the network. Further, the path corresponding to the origin-destination pair 7-5 shows the least reliability.

Table 1: Reliability of optimal paths in the perfectly reliable network

Origin	Destination	Reliability of optimal path
1	5	0.78
7	5	0.49
14	5	1.0
1	6	0.80
7	6	0.98
14	6	0.90

Based on the above expected costs and whether or not late and early arrival penalty is being applied, we present the optimal departure time results for various passenger groups in Table 2. When the penalties are not applied, we look for a departure time that comes after the earliest departure time and provides the least expected cost to the respective destination. This outputs similar departure times for groups that have the same origin and neighboring earliest departure times. When the early and late arrival penalties are applied, we look for departure time that provides the least expected cost based on (7). The penalties sometimes cause a passenger group to depart early or late to arrive at the destination in a given time interval.

For assigning passengers we use the departure times calculated based on the penalties. The average passenger flow obtained after running Algorithm 2 is visualized in Figure 4. The flow of passengers on various links is varied according to the line width of various links in the figure. The

Table 2: Optimal departure time of passenger groups

Group	Penalties not included	Penalties included	Group	Penalties not included	Penalties included
1	10:05	10:05	13	10:00	10:09
2	10:05	10:05	14	10:17	10:03
3	10:05	09:55	15	10:17	10:10
4	10:05	10:05	16	10:17	10:17
5	10:05	10:05	17	10:17	10:17
6	10:05	10:05	18	10:17	10:03
7	10:05	10:09	19	10:08	10:08
8	10:05	10:09	20	10:18	10:18
9	10:05	10:09	21	10:24	10:24
10	10:00	10:00	22	10:18	10:08
11	10:09	10:09	23	10:18	10:18
12	10:00	10:09	24	10:18	10:18

transfer links are represented using dashed lines. If a link is not shown between two nodes, then either such link does not exist in the network, or the flow of passengers on that link is zero. We can observe that most passengers prefer taking the first and second trips of various routes in the network. This is because most passenger groups have departure times closer to the departure times of the first and second trips of various transit routes departing from their origins. We observe the highest flow on the second trip of red and blue routes. This is because together these two routes connect both destinations (5 and 6). The passenger groups going from origin 14 to destination 6 prefer taking the orange route and the passenger groups going from 1 to 5 or 6 prefer taking the transfer 3-4. However, we observe some flow on the first trip of the green route from origin 1 to destination 6 that takes a transfer to the orange route. The passengers going from origin 7 to destination 5 prefer taking the transfer 8-2 from the violet to the red route. To go to destination 5, we observe some passengers taking transfers 12-4 and 3-4 from the second trip of the orange and red routes to the third trip of the blue route. We do not observe a significant flow of passengers for the fourth trip of various transit routes. This is because most groups do not have departure time window as late as compared to the departure times of fourth trips of various routes departing from various origins.

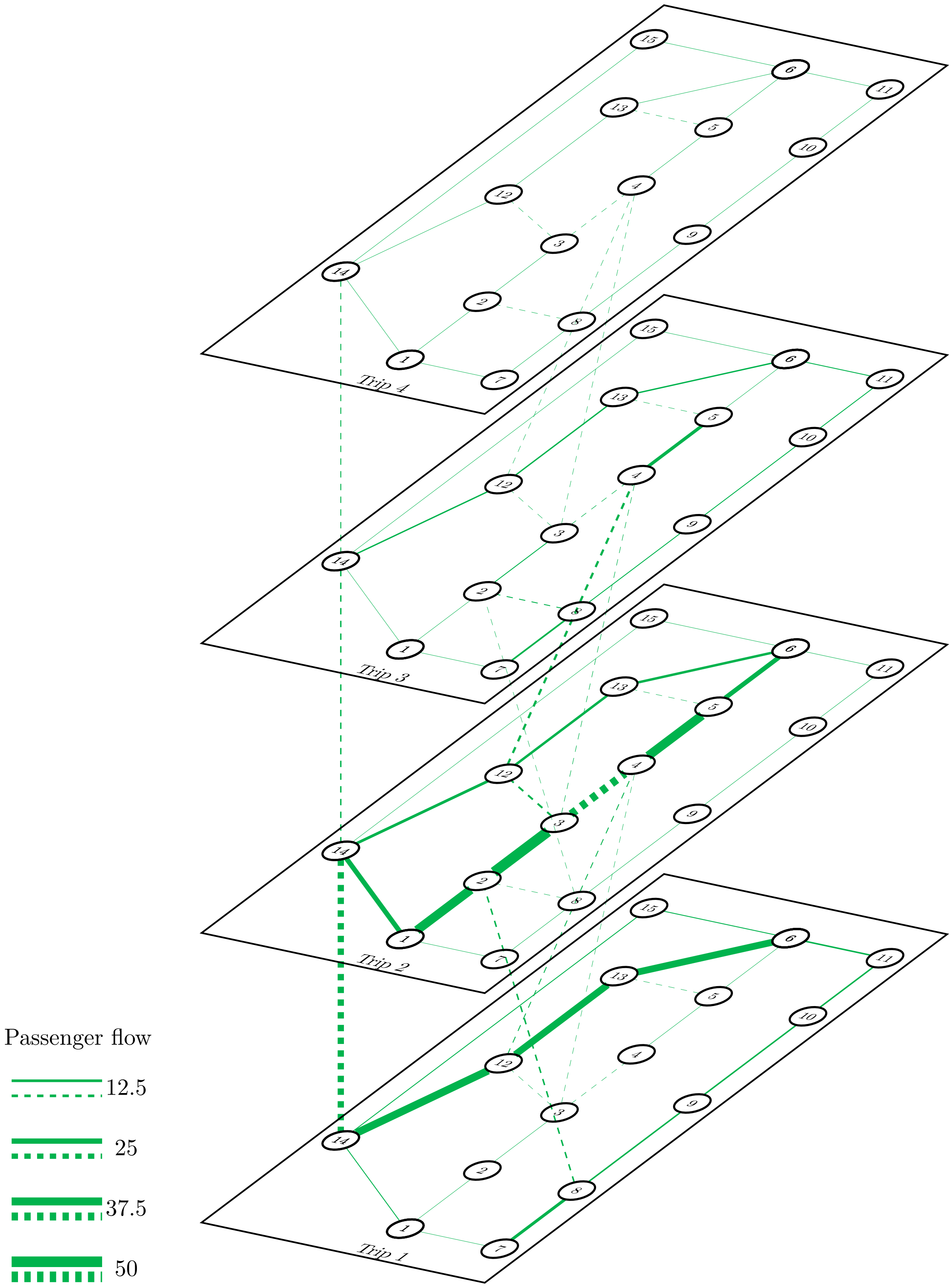


Figure 4: Passenger flow on various trips for uncapacitated transit assignment

6.2. Capacitated assignment

In this section, we present the results of the capacitated transit assignment. We start by creating the transfer links using Algorithm 6. For this case, it creates seven waiting transfer links, twenty four walking transfer links, and thirty four walking links for failed transfers. The number of generated states is 29,832, which is six times higher as compared to the uncapacitated case. We ran the assignment Algorithm 5 with the gap tolerance value $\epsilon = 0.05\%$. It took 140 iterations and 8.5 minutes to converge to the solution with required tolerance gap. We plot the convergence behavior of the algorithm in Figure 5, where we can observe a continuous decline in the gap value with every iteration. The overall convergence is achieved fairly quickly.

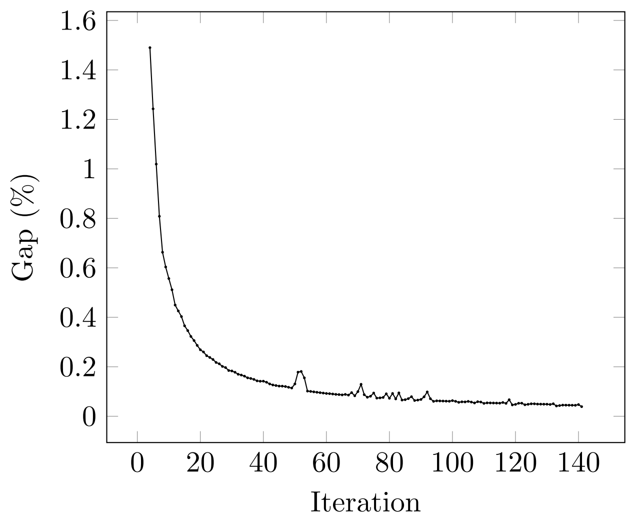
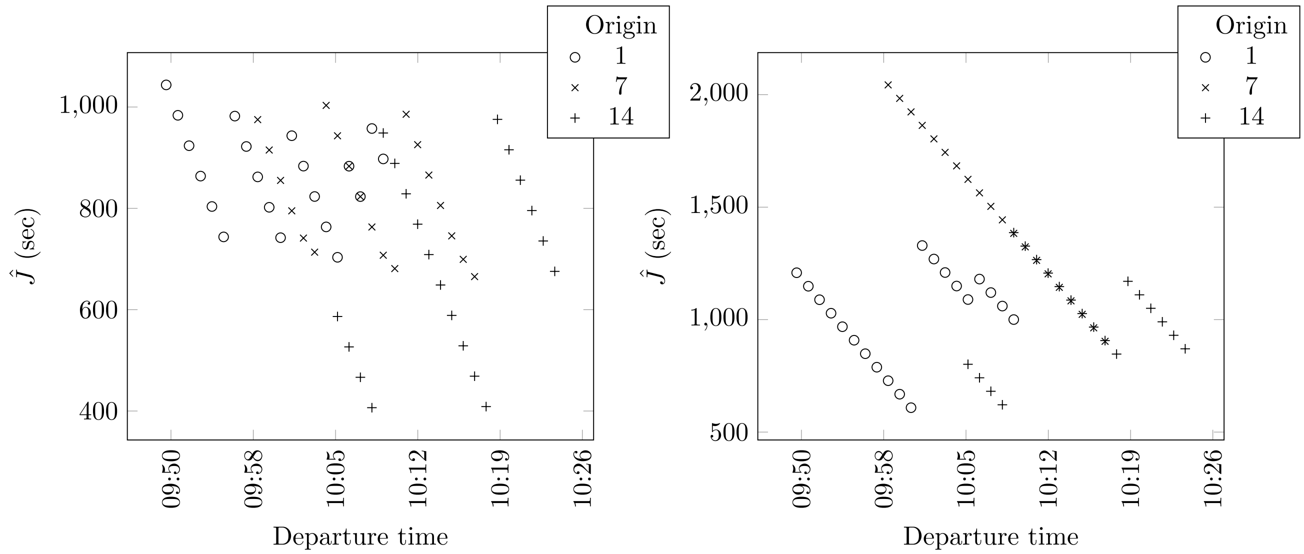


Figure 5: Converege behavior of MSA algorithm

The final values of the expected cost of travel between various origin-destination pairs are plotted in Figure 6. The average cost of travel to destination 5 from various origins is more than the destination 6. This is because of the presence of paths without transfer between various origins and destination 6. On the other hand, one has to take at least one transfer to get to destination 5. Due to limited capacity, passengers miss transfers, which leads to higher expected travel times. If we compare the expected cost from various origins to destination 6 in both uncapacitated and capacitated cases, we find that the expected cost of traveling between 1-6 is higher in the case of capacitated assignment as compared to the uncapacitated assignment. This is because passengers who do not get the preferred option of the red route due to limited capacity would either have to take the violet route or the green route resulting in higher expected cost. For destination 5, the expected cost of travel between 7-5 in the case of capacitated assignment has risen considerably as compared to the uncapacitated assignment. This is because passengers who want to take transfer 8-2 coming from 7 on the violet route do not get the priority over passengers who are continuing their journey on link 2-3 of the red route.

The converged departure time probabilities for various groups are visualized in Figure 7. Out of



(a) Destination 6

(b) Destination 5

Figure 6: Expected cost between various origin-desination pairs for varying departure times in case of capacitated assignment

twenty four groups, ten groups have only one departure time, i.e., the probability of departing at a single departure time by these groups is 1. We further observe that eleven groups have two values in their departure time support and three groups have three or more values in their departure time support.

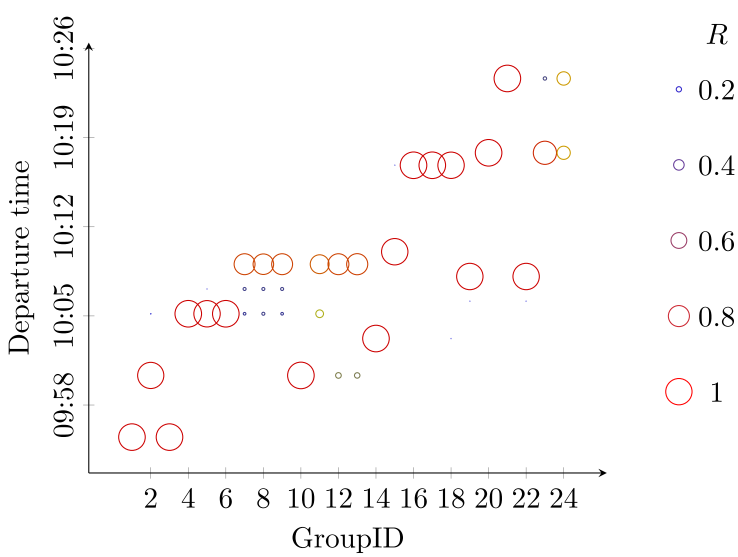


Figure 7: Converged values of departure time probabilities R

The uncapacitated assignment does not give us capacity-feasible flows. This is evident from the flow values visualized in Figure 4, where the first trip of the orange route and the second trip

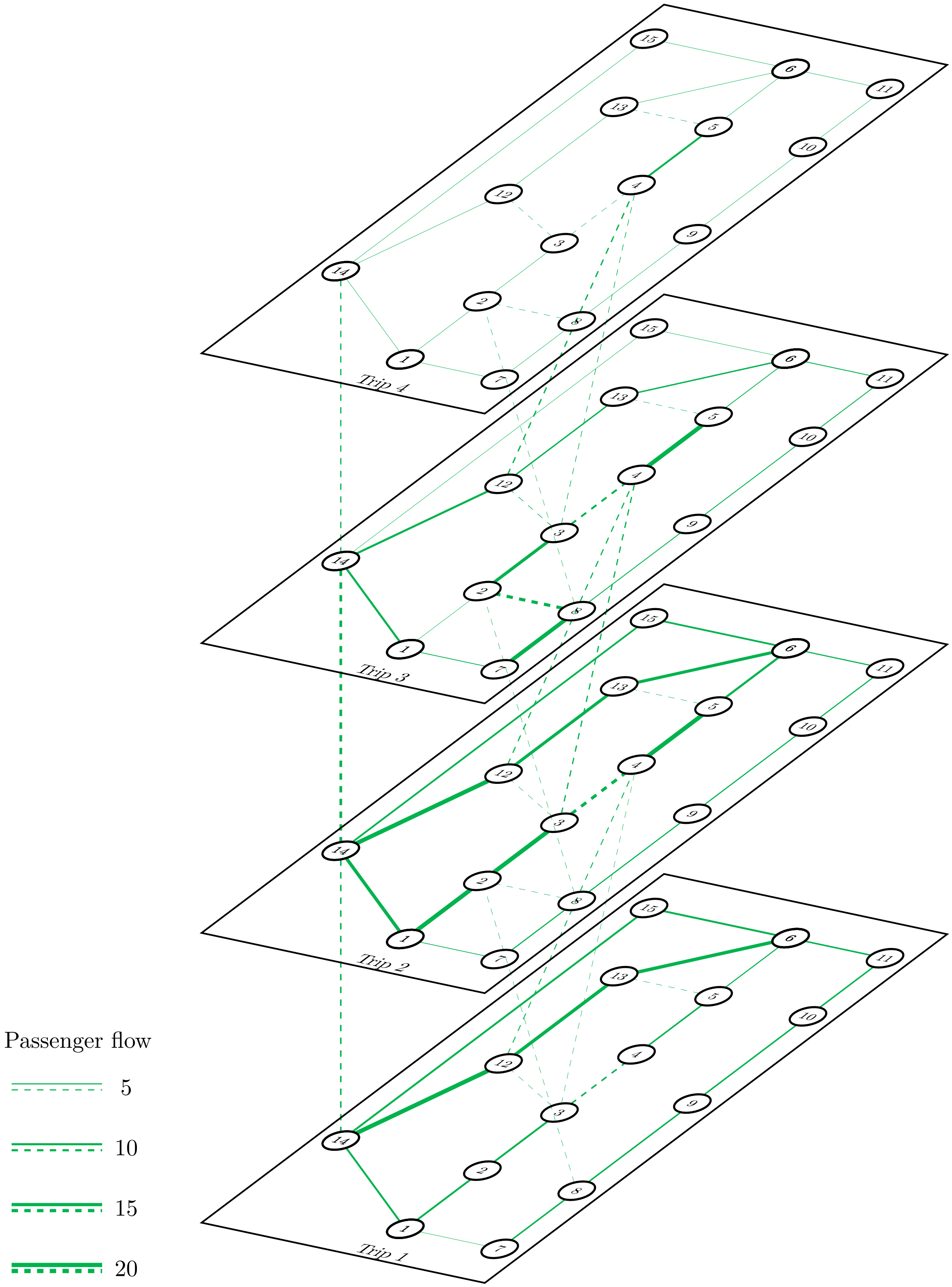


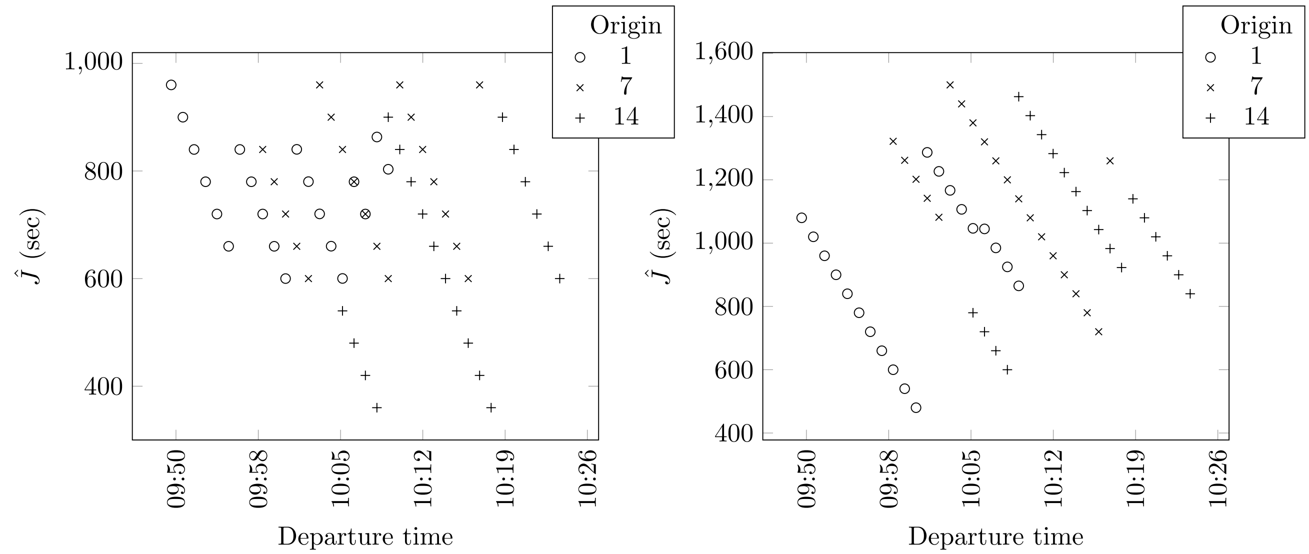
Figure 8: Passenger flow on various trips for capacitated transit assignment

of red and violet routes carry more flow than their capacity (20 passengers). The capacitated assignment results in more realistic passenger flow on various trips and routes, which is visualized in Figure 8. Due to the limited capacity of transit routes, passengers have to shift from their most preferred choice to other choices. By looking at the figure, we can see that various segments of many attractive trip options are running at near or full capacity. This includes the first trip of the orange route, the second trip of red, green, orange, and blue routes, and the third trip of violet and blue routes. A significant proportion of passengers taking the first trip of the orange route in case of the uncapacitated assignment are distributed to the second and third trip of the same route. To travel between 1-6, the orange route and combination of red and blue routes are the most popular choices. Both choices include one transfer and provide improved expected costs as compared to direct routes (green and violet). To travel between 7-5, passengers prefer taking two transfers 8-2 and 3-4 within second or second to third trips of the respective routes. Some passengers going from 14 to 5 have to face denied boarding on the blue route due to the only option to get to destination 5. This results in non-zero flow on transfer links 12-4 between various trips of orange and blue routes. Finally, we do not observe any passengers that have to walk to their destination due to failed transfer. This is because most passenger groups have departure times closer to the departure times of the first and second trips of various transit routes departing from their origins. This results in the availability of an alternative trip for passengers to take in case of missed transfers.

6.3. Comparison to reliable schedule-based assignment

We further compare the results of the capacitated schedule-based assignment computed in the previous section to the capacitated assignment results when buses are not delayed and follow the perfectly reliable schedule. For this purpose, we assume that link travel times have only one realization, i.e., scheduled travel time w.p. 1.0. We ran the assignment Algorithm 5 with the gap tolerance value $\epsilon = 0.05\%$. It took 92 iterations and 57 seconds to converge to the solution with the required tolerance gap. The expected costs to go between various origin-destination pairs for varying departure times are plotted in Figure 9. We compare these costs with the ones given in Figure 6. For destination 6, the overall pattern in the trend of values computed in both cases is the same. However, we observe that the perfectly reliable network provides lower expected costs as compared to the unreliable network. Moreover, we see that for some origins, there are more points in Figure 9(a). For example, one cannot depart after 10:19 in Figure 6(a) from origin 7 and still reach destination 6, but it is possible to do so if there is a perfectly reliable network (see Figure 9(a)). For destination 5, the expected cost to go from various origins in unreliable network (Figure 6(b)) is significantly higher than the reliable network (Figure 9(b)). We see a lot more points in Figure 9(b) because all the transfers are available. This analysis shows that the strategies evaluated with reliable schedule is overly optimistic but not very realistic.

We plot the final average flow values in the network after running the Algorithm 5 in Figure 10 and compare its results with the unreliable assignment results shown in Figure 8. The main observation is that there is more transferring flow in the case of reliable networks. For example,



(a) Destination 6 (b) Destination 5

Figure 9: Expected cost between various origin-desination pairs for varying departure times in case of capacitated assignment for perfectly reliable network

there is a non-zero flow on transfer links 13-5 and 3-12 of various trips. This is because the schedule is perfectly reliable and passengers can make various transfers which are not possible if there is a delayed service. This shows that assignment results computed based on a reliable schedule assumption give more transferring passenger flow than should happen in practice. Previous studies have used penalties to avoid such a large number of transfers. However, we let the algorithm do the penalization systematically and realistically.

6.4. Application to real case study

We use the Minneapolis transit network to demonstrate the application of the presented methodology on a large-scale network. To evaluate the impact of multiple transit route options, we have selected 13 high ridership routes, including routes 2, 3, 4, 5, 6, 11, 18, 113, 114, 115, Blue Line, Green Line, and the Red Line (see Figure 11). We use synthetic transit demand going to the University of Minnesota campus during morning peak hours (7-9 AM) obtained from the 2010 activity-based travel demand model for Twin Cities, MN, developed by Metropolitan Council (Cambridge Systematics 2015). The data contains information about passenger trip origins, destinations, and preferred arrival and departure times. Since the departure and arrival times are available on a 30-min scale, we have subtracted a uniformly distributed random time between 5 and 20 minutes from the departure time and added a random time between 5 and 20 minutes to the arrival time to obtain the earliest departure time and the latest arrival time respectively. Additionally, the earliest arrival time was calculated by subtracting a random time between 5 and 10 minutes from the latest arrival time of passengers. The schedule-based transit network was created using the GTFS data provided by Metro Transit, which is the primary agency in Twin Cities, MN, offering an integrated network of buses, light rails, and a commuter train. The travel time to traverse access

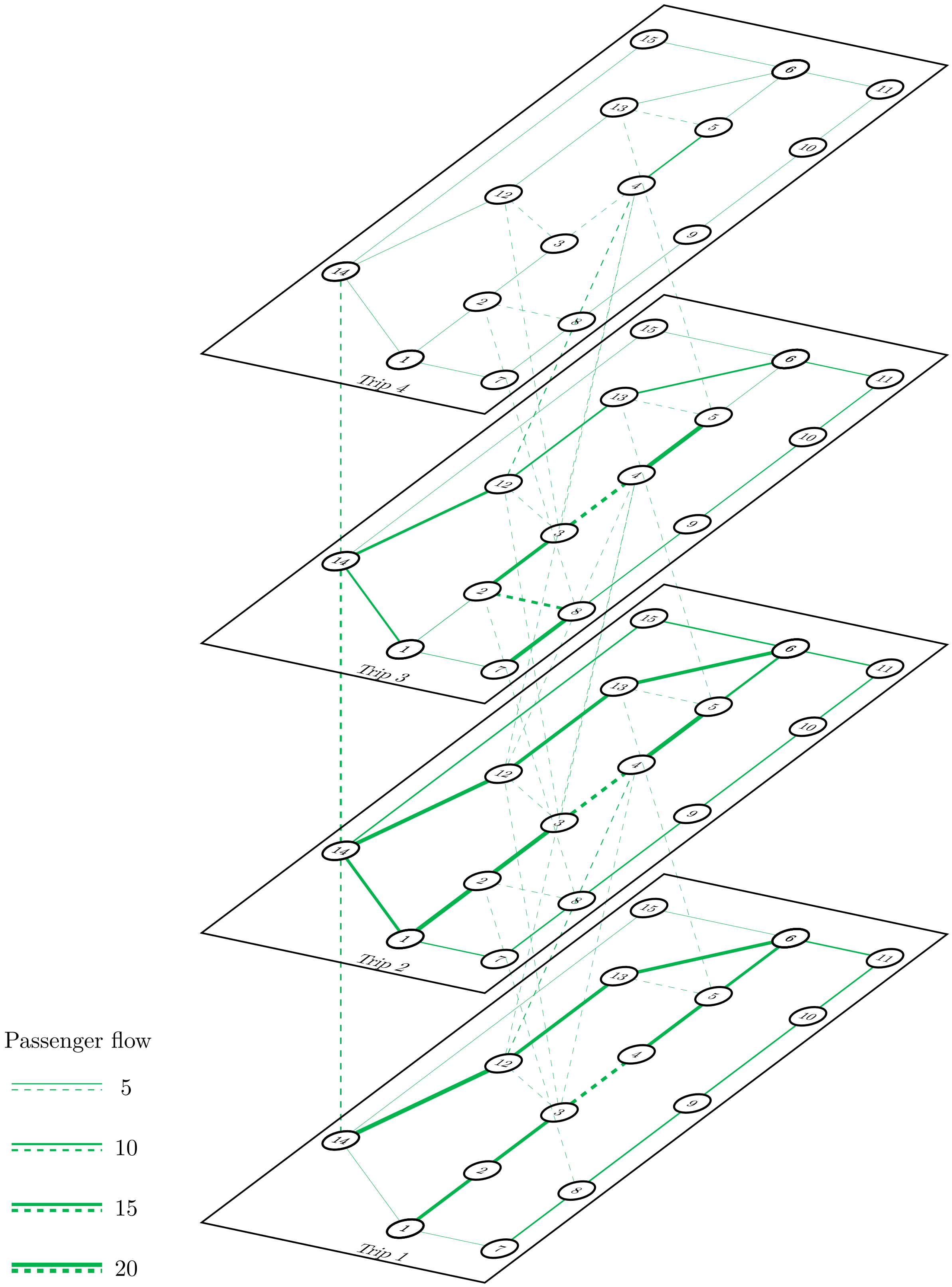


Figure 10: Passenger flow on various trips using capacitated transit assignment for perfectly reliable network

links or walking transfer links was calculated by dividing their Euclidean distance by the average walking speed (which is assumed to be 3 mi/hr). We use 0.75mi and 0.25mi as walking thresholds for creating access/egress and transfer links respectively.

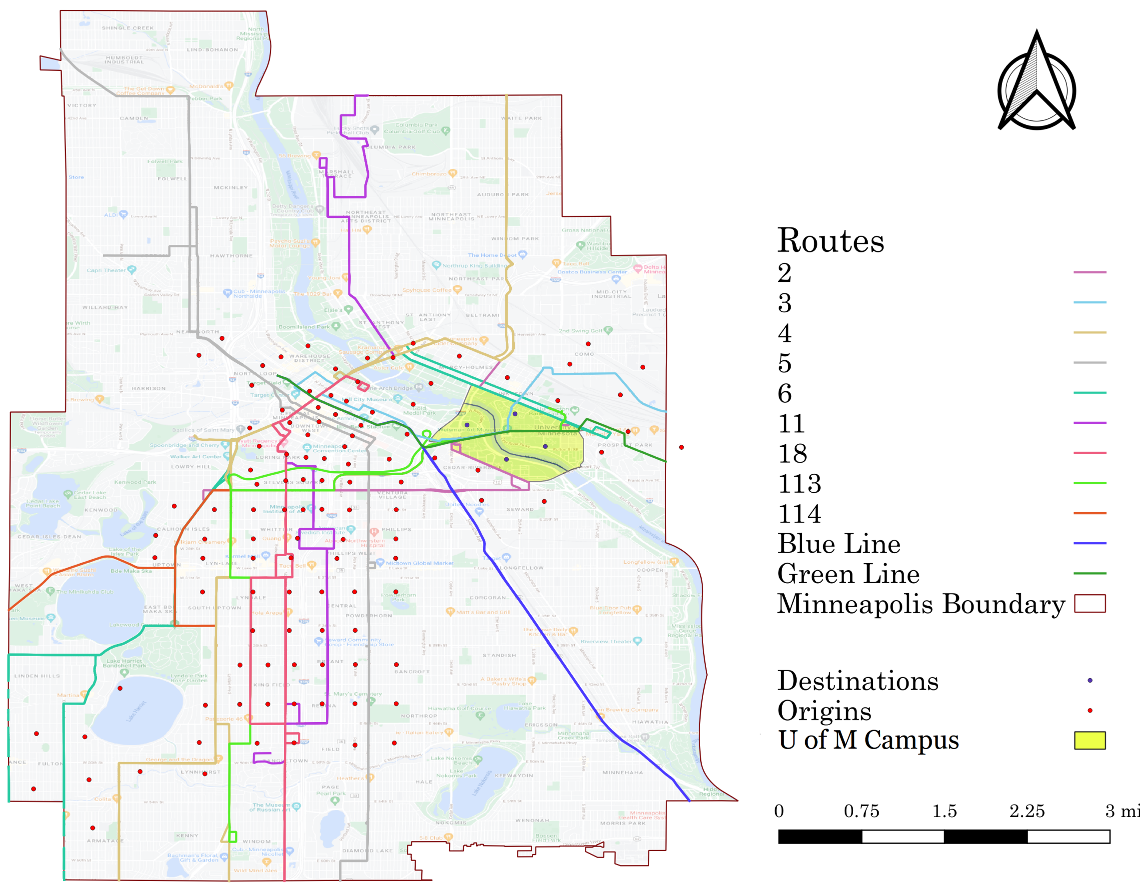


Figure 11: Minneapolis transit network (For interpretation of colors, please refer to the web version of this article)

The selected SB transit network has 510 stops, 13 routes, 101 trips, 3742 nodes, 3333 access/egress links, 3557 in-vehicle links, 3703 waiting/walking transfer links, 153 O-D pairs (82 origins, 4 destinations), and 302 passenger groups. Since we could not arrange historical AVL data for this study, we assumed the support of random travel times (in seconds) of in-vehicle links as below:

$$\begin{cases} \bar{c}_{ij}, & \bar{c}_{ij} \leq 120 \\ \bar{c}_{ij}, 1.1\bar{c}_{ij}, & 120 \leq \bar{c}_{ij} < 240 \\ \bar{c}_{ij}, 1.1\bar{c}_{ij}, 1.2\bar{c}_{ij} & 240 \leq \bar{c}_{ij} < 360 \\ \bar{c}_{ij}, 1.2\bar{c}_{ij}, 1.5\bar{c}_{ij} & \bar{c}_{ij} \geq 360 \end{cases}$$

where, $\bar{c}_{ij}$ is the scheduled travel time (in seconds) of link (i, j).

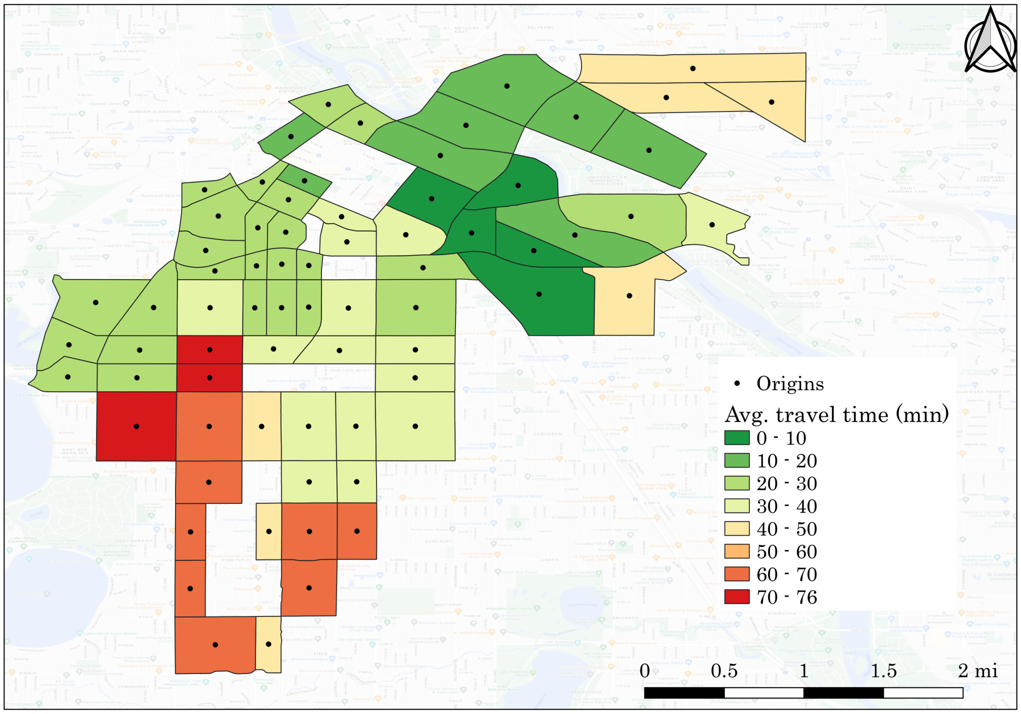


Figure 12: Average travel time (min) from various origin zones departing at 8:00 AM aggregated over destinations (For interpretation of colors, please refer to the web version of this article)

Other parameters are assumed as given in Section 6. The number of generated states is 4,02,451. We ran the Algorithm 5 with the gap tolerance value $\epsilon = 1\%$. It took 2 iterations and 8,040 seconds to converge to the solution with required tolerance gap. The average travel time from various origins departing at 8:00 A.M. aggregated over various destinations is shown in Figure 12. The area around the campus is accessible within average travel time of 10 minutes. The Como area and Downtown East are also accessible within average travel time of 20 minutes. The average travel time to the campus increases in Uptown area, where it can range from 50-70 minutes of travel time.

The average passenger flow in the network, aggregated over various transit trips, is shown in Figure 13. The flow of passengers on various links is varied according to the color intensity of various links in the figure. In Southwest Minneapolis region, passengers prefer boarding routes 4, 5, 11, 18, and 114. Since routes 3, 2, 6, 113, and the Green Line go to the University campus, these routes have the highest ridership, as shown in Figure 14. Passengers either take these routes directly or transfer to them to get to the campus. Figure 15 shows the aggregated number of passengers transferring from one route to another. Route 2 and Green Line have the highest number transfers to reach the University campus because of their higher frequency.

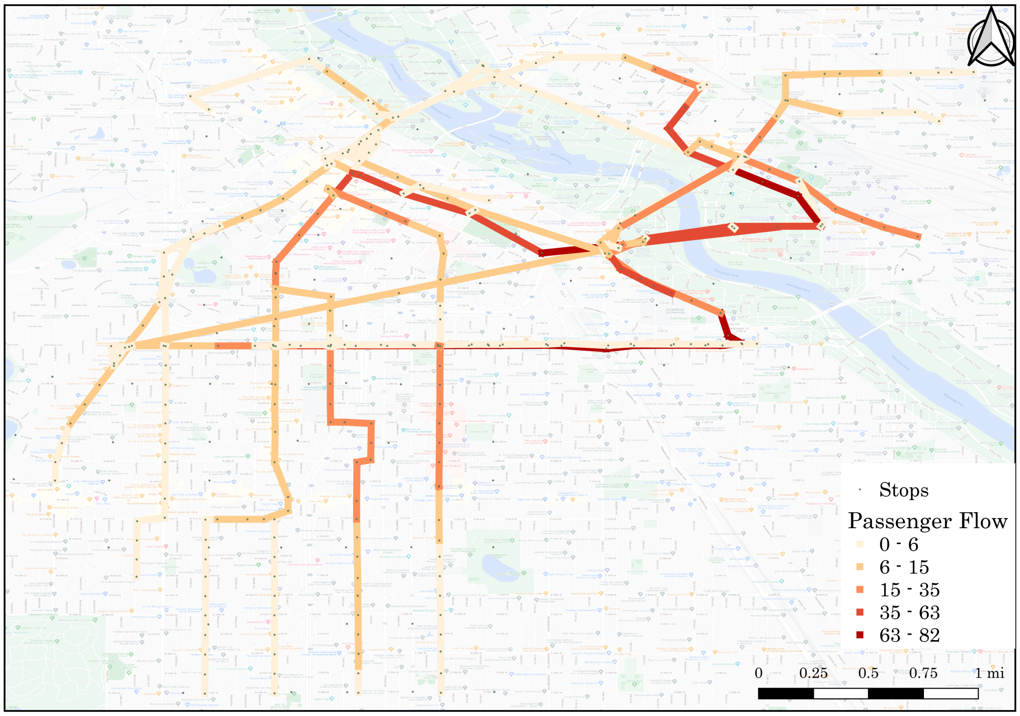


Figure 13: Aggregated passenger flow on Minneapolis transit network (For interpretation of colors, please refer to the web version of this article)

7. Conclusions and directions for future research

The current research develops a schedule-based transit assignment model that predicts passenger route choice behavior in the presence of online bus arrival information. The availability of online information induces an adaptive behavior, where a passenger who faces failed transfer due to early or late arrival of buses, can consider alternative bus routes to minimize their expected cost to destination. We present two transit assignment models based on whether or not the limited capacity of transit vehicles is considered. The uncapacitated assignment model is useful for transit systems with low ridership, whereas the capacitated assignment model is useful for transit systems with high ridership. In both cases, we propose that passengers adopt strategies to travel and use the stochastic shortest path as a modeling tool to characterize passenger hyperpaths. Under restrictive assumptions, a linear program can be solved to perform the uncapacitated assignment. On the other hand, the capacitated assignment is more complex than the uncapacitated assignment. This is because the strategic behavior of passengers is observed not only because of online information but also due to the limited capacity of transit vehicles. For this purpose, we formulate the capacitated assignment problem as a variational inequality problem, which is solved using an MSA-based heuristic algorithm. The algorithm runs the shortest path as well as a loading procedure to incorporate realistic passenger behavior. The MSA algorithm shows good convergence performance on the conducted experiments. We present case studies based on the Tong and Richardson 1984 and Twin Cities schedule-based transit networks. The results evaluated the departure times of various

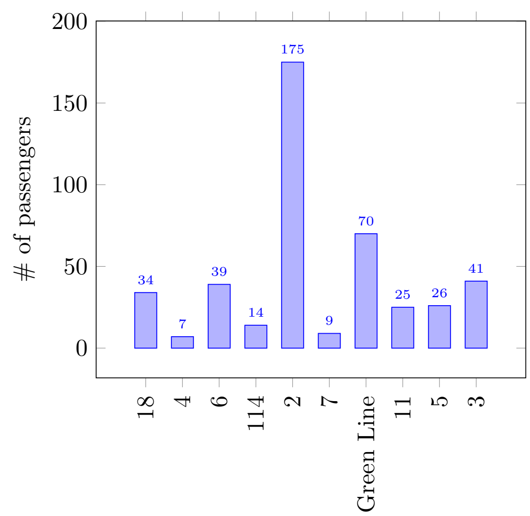


Figure 14: Highest ridership routes

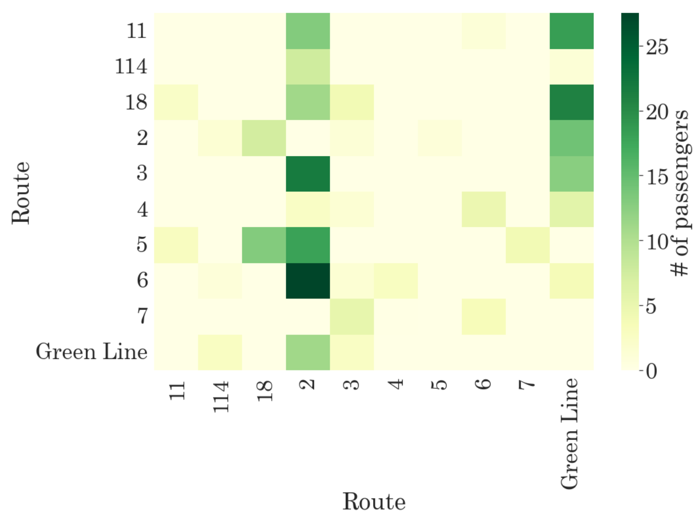


Figure 15: Transferring passenger flow

groups and passenger flows on various trips of transit routes. The computational time required to perform both uncapacitated and capacitated assignments was within 10 minutes for Tong and Richardson 1984 network. The analysis shows that the strategies evaluated with reliable schedule assumption is overly optimistic but not very realistic. We observed that such assumption leads to unreliable paths in the network and produces more transferring flow than the proposed model. The limited capacity results in high-dimensional strategies and more complex behavior. The denied boarding leads to higher expected costs to passengers. In the case study on the subnetwork of the Twin Cities transit network with artificial demand, we found that University of Minnesota students traveling from residential areas to campus may choose transfer paths in the event of highly unreliable service on direct transit routes. Route 2 and the Green Line were found to carry the highest number of transferring passengers.

One of the disadvantages of the capacitated assignment model is the explosion of state space due to incorporation of availability vector in the state space. This results in the high computational time required to solve the corresponding SSP. For the assignment, one needs to solve the SSP several times, which could make it difficult to produce assignment results. Future research should focus on proposing techniques to solve this problem faster. This could be achieved using approximate dynamic programming algorithms. Nevertheless, the exact methods developed in this study will help in evaluating the accuracy of the approximation algorithms. Furthermore, a model-free reinforcement learning approach can also be used to predict passenger behavior in the presence of online information. The calibration of such a model using travel behavior data (e.g., Automatic Fare Collection data) will require a significant effort. Finally, the current model provides a flexible framework to incorporate various choice probabilities. For example, one can use logit-based route choice probabilities to achieve a stochastic user equilibrium.

Acknowledgements

This research is conducted at the University of Minnesota Transit Lab, currently supported by the following, but not limited to, projects:

- National Science Foundation, United States of America, award CMMI-1831140
- Freight Mobility Research Institute (FMRI), Tier 1 Transportation Center, U.S. Department of Transportation: award RRK78/FAU SP#16-532 AM2, AM3, and AM4
- Minnesota Department of Transportation, United States of America, Contract No. 1036342 Work Order No. 08, Contract No. 1003325 Work Order No. 111, and Contract No. 1044277
- Metropolitan Council, United States of America Contract No. SG-2021-021
- Hennepin County, United States of America Contract No. UM0921

References

Andreatta, G. and Romeo, L. 1988, 'Stochastic shortest paths with recourse', *Networks* **18**(3), 193–204.

Baillon, J.-B. and Cominetti, R. 2008, 'Markovian traffic equilibrium', *Mathematical Programming* **111**(1), 33–56.
URL: *https://doi.org/10.1007/s10107-006-0076-2*

Bertsekas, D. P. 2012, *Dynamic Programming and Optimal Control*, fourth edn, Athena Scientific, Belmont, Massachusetts.
URL: *www.athenasc.com/dpbook.html*

Billi, C., Gentile, G. and Nguyen, S. 2004, Rethinking the wait model at transit stops, *in* 'Proceedings of TRISTAN V : The Fifth Triennial Symposium on Transportation Analysis', number November, Le Gosier, pp. 1–8.
URL: *http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.157.454&rep=rep1&type=pdf*

Binder, S., Maknoon, Y. and Bierlaire, M. 2017, 'Exogenous priority rules for the capacitated passenger assignment problem', *Transportation Research Part B: Methodological* **105**, 19–42.

Boyles, S. D. and Rambha, T. 2016, 'A note on detecting unbounded instances of the online shortest path problem', *Networks* **67**(4), 270–276.

Brakewood, C. and Watkins, K. 2019, 'A literature review of the passenger benefits of real-time transit information', *Transport Reviews* **39**(3), 327–356.

Cambridge Systematics, I. 2015, Model User Guide, 2010 Travel Behavior Inventory, Technical report, Chicago, IL.

Cats, O. and Gkioulou, Z. 2017, ‘Modeling the impacts of public transport reliability and travel information on passengers’ waiting-time uncertainty’, *EURO Journal on Transportation and Logistics* **6**(3), 247–270.

Cepeda, M., Cominetti, R. and Florian, M. 2006, ‘A frequency-based assignment model for congested transit networks with strict capacity constraints: characterization and computation of equilibria’, *Transportation research part B: Methodological* **40**(6), 437–459.

Chen, P. W. and Nie, Y. M. 2015, ‘Optimal transit routing with partial online information’, *Transportation Research Part B: Methodological* **72**, 40–58.

Cheung, R. K. 1998, ‘Iterative methods for dynamic stochastic shortest path problems’, *Naval Research Logistics (NRL)* **45**(8), 769–789.

Chriqui, C. and Robillard, P. 1975, ‘Common bus lines’, *Transportation science* **9**(2), 115–121.

Cominetti, R. and Correa, J. 2001, ‘Common-lines and passenger assignment in congested transit networks’, *Transportation science* **35**(3), 250–267.

Crisalli, U. 1970, ‘Dynamic transit assignment algorithms for urban congested networks’, *WIT Transactions on The Built Environment* **44**.

De Cea, J. and Fernández, E. 1993, ‘Transit assignment for congested public transport systems: an equilibrium model’, *Transportation science* **27**(2), 133–147.

Fonzone, A. 2015, ‘What do you do with your app?: Study of bus rider decision making with real-time passenger information’, *Transportation Research Record* **2535**, 15–24.
URL: *https://doi.org/10.3141/2535-02*

Fonzone, A. and Schmöcker, J. D. 2014, ‘Effects of transit real-time information usage strategies’, *Transportation Research Record* **2417**, 121–129.

Gao, S., Frejinger, E. and Ben-Akiva, M. 2008, ‘Adaptive route choice models in stochastic time-dependent networks’, *Transportation Research Record* **2085**(1), 136–143.

Gao, S. and Huang, H. 2012, ‘Real-time traveler information for optimal adaptive routing in stochastic time-dependent networks’, *Transportation Research Part C: Emerging Technologies* **21**(1), 196–213.
URL: *http://dx.doi.org/10.1016/j.trc.2011.09.007*

Gardner, C. B., Nielsen, S. D., Eltved, M., Rasmussen, T. K., Nielsen, O. A. and Nielsen, B. F. 2021, ‘Calculating conditional passenger travel time distributions in mixed schedule-and frequency-based public transport networks using Markov chains’, *Transportation Research Part B: Methodological* **152**, 1–17.

Gentile, G., Nguyen, S. and Pallottino, S. 2005, 'Route choice on transit networks with online information at stops', *Transportation science* **39**(3), 289–297.

Google 2005, 'General Transit Feed Specification'.
URL: *http://code.google.com/ transit/spec/transit_feed_specification.htm*

Hall, R. W. 1986, 'The Fastest Path through a Network with Random Time-Dependent Travel Times', *Transportation Science* **20**(3), 182–188.

Hamdouch, Y., Ho, H. W., Sumalee, A. and Wang, G. 2011, 'Schedule-based transit assignment model with vehicle capacity and seat availability', *Transportation Research Part B: Methodological* **45**(10), 1805–1830.

Hamdouch, Y. and Lawphongpanich, S. 2008, 'Schedule-based transit assignment model with travel strategies and capacity constraints', *Transportation Research Part B: Methodological* **42**.

Hamdouch, Y., Marcotte, P. and Nguyen, S. 2004, 'Capacitated transit assignment with loading priorities', *Mathematical programming* **101**(1), 205–230.

Hamdouch, Y., Szeto, W. Y. and Jiang, Y. 2014, 'A new schedule-based transit assignment model with travel strategies and supply uncertainties', *Transportation Research Part B: Methodological* **67**, 35–67.
URL: *http://dx.doi.org/10.1016/j.trb.2014.05.002*

Hickman, M. D. and Bernstein, D. H. 1997, 'Transit service and path choice models in stochastic and time-dependent networks', *Transportation Science* **31**(2), 129–146.

Hickman, M. D. and Wilson, N. H. M. 1995, 'Passenger travel time and path choice implications of real-time transit information', *Transportation Research Part C* **3**(4), 211–226.

Islam, M. F. and Fonzone, A. 2021, 'Bus passenger path choices after consulting ubiquitous real-time information', *Travel Behaviour and Society* **23**(August 2020), 226–239.
URL: *https://doi.org/10.1016/j.tbs.2021.01.001*

Khani, A. 2019, 'An online shortest path algorithm for reliable routing in schedule-based transit networks considering transfer failure probability', *Transportation Research Part B: Methodological* **126**, 549–564.

Khani, A., Hickman, M. and Noh, H. 2014, 'Trip-Based Path Algorithms Using the Transit Network Hierarchy', *Networks and Spatial Economics* **15**(3), 635–653.

Khani, A., Hickman, M. and Noh, H. 2015, 'Trip-based path algorithms using the transit network hierarchy', *Networks and Spatial Economics* **15**(3), 635–653.

Kumar, P. and Khani, A. 2019, 'Public Transit Transfer Reliability Analysis Tool'.
URL: *https://transitapp.shinyapps.io/shniy_app/*

Kumar, P. and Khani, A. 2021, 'Adaptive park-and-ride choice on time-dependent stochastic multimodal transportation network', *Networks and Spatial Economics* .

Kumar, P. and Khani, A. 2022, 'Planning of integrated mobility-on-demand and urban transit networks', *Transportation Research Part A: Policy and Practice* **166**(October), 499–521.

Kurauchi, F., Bell, M. G. and Schmöcker, J. D. 2003, 'Capacity Constrained Transit Assignment with Common Lines', *Journal of Mathematical Modelling and Algorithms* **2**(4), 309–327.

Marcotte, P., Nguyen, S. and Schoeb, A. 2004, 'A strategic flow model of traffic assignment in static capacitated networks', *Operations Research* **52**(2), 191–212.

Miller-Hooks, E. D. and Mahmassani, H. S. 2003, 'Least Expected Time Paths in Stochastic, Time-Varying Transportation Networks', *Transportation Science* **34**(2), 198–215.

Nguyen, S. and Pallottino, S. 1988, 'Equilibrium traffic assignment for large scale transit networks', *European Journal of Operational Research* **37**(2), 176–186.

Nguyen, S., Pallottino, S. and Malucelli, F. 2001, 'A modeling framework for passenger assignment on a transport network with timetables', *Transportation Science* **35**(3), 238–249.

Nielsen, O. A. 2004, A large scale stochastic multi-class schedule-based transit model with random coefficients, *in* 'Schedule-based dynamic transit modeling: theory and applications', Springer, pp. 53–77.

Noh, H., Hickman, M. and Khani, A. 2012, 'Hyperpaths in network based on transit schedules', *Transportation research record* **2284**(1), 29–39.

Nuzzolo, A., Crisalli, U., Comi, A. and Rosati, L. 2016, 'A mesoscopic transit assignment model including real-time predictive information on crowding', *Journal of Intelligent Transportation Systems* **20**(4), 316–333.

Nuzzolo, A., Crisalli, U. and Rosati, L. 2012, 'A schedule-based assignment model with explicit capacity constraints for congested transit networks', *Transportation Research Part C: Emerging Technologies* **20**(1), 16–33.

Nuzzolo, A., Russo, F. and Crisalli, U. 2001, 'A doubly dynamic schedule-based assignment model for transit networks', *Transportation Science* **35**(3), 268–285.

Oliker, N. and Bekhor, S. 2018, 'A frequency based transit assignment model that considers online information', *Transportation Research Part C: Emerging Technologies* **88**, 17–30.

Paulsen, M., Rasmussen, T. K. and Nielsen, O. A. 2021, 'Impacts of real-time information levels in public transport: A large-scale case study using an adaptive passenger path choice model', *Transportation Research Part A: Policy and Practice* **148**(May 2020), 155–182.
URL: *https://doi.org/10.1016/j.tra.2021.03.011*

Polychronopoulos, G. H. and Tsitsiklis, J. N. 1996, 'Stochastic shortest path problems with recourse', *Networks: An International Journal* **27**(2), 133–143.

Poon, M. H., Wong, S. C. and Tong, C. O. 2004, 'A dynamic schedule-based model for congested transit networks', *Transportation Research Part B: Methodological* **38**(4), 343–368.

Provan, J. S. 2003, 'A Polynomial-Time Algorithm to Find Shortest Paths with Recourse', *Networks* **41**(2), 115–125.

Rambha, T., Boyles, S. D. and Waller, S. T. 2016, 'Adaptive Transit Routing in Stochastic Time-Dependent Networks', *Transportation Science* **50**(3), 1043–1059.
URL: *http://pubsonline.informs.org/doi/10.1287/trsc.2015.0613*

Riter, S. and McCoy, J. 1977, 'Automatic vehicle location—An overview', *IEEE Transactions on Vehicular Technology* **26**(1), 7–11.

Spiess, H. and Florian, M. 1989, 'Optimal strategies: A new assignment model for transit networks', *Transportation Research Part B* **23**(2), 83–102.

Sumalee, A., Tan, Z. and Lam, W. H. K. 2009, 'Dynamic stochastic transit assignment with explicit seat allocation model', *Transportation Research Part B: Methodological* **43**(8-9), 895–912.

Tong, C. O. and Richardson, A. J. 1984, 'A computer model for finding the time-dependent minimum path in a transit system with fixed schedules', *Journal of Advanced Transportation* **18**(2), 145–161.

Tong, C. O. and Wong, S. C. 1999, 'A stochastic transit assignment model using a dynamic schedule-based network', *Transportation Research Part B: Methodological* **33**(2), 107–121.

Wahba, M. and Shalaby, A. 2009, MILATRAS: A new modeling framework for the transit assignment problem, *in* 'Schedule-Based Modeling of Transportation Networks', Springer, pp. 1–24.

Waller, S. T. and Ziliaskopoulos, A. K. 2002, 'On the Online Shortest Path Problem with Limited Arc Cost Dependencies', *Networks* **40**(4), 216–227.

Webb, A., Kumar, P. and Khani, A. 2020, 'Estimation of passenger waiting time using automatically collected transit data', *Public Transport* **12**(2), 299–311.

Wu, J. H., Florian, M. and Marcotte, P. 1994, 'Transit Equilibrium Assignment: A Model and Solution Algorithms', *Transportation Science* **28**(3), 193–203.

Zhang, Y., Lam, W. H. K., Sumalee, A., Lo, H. K. and Tong, C. O. 2010, 'The multi-class schedule-based transit assignment model under network uncertainties', *Public Transport* **2**(1), 69–86.

Zimmermann, M., Frejinger, E. and Marcotte, P. 2021, 'A strategic markovian traffic equilibrium model for capacitated networks', *Transportation Science* **55**(3), 574–591.

Appendix A Creation of transfer links

In this study, we propose two types of assignment models, namely, uncapacitated and capacitated assignments. In both assignment models, we reduce the transfer links based on an acceptable waiting time limit. Moreover, in the uncapacitated assignment, the transfers can be further reduced based on the probability of making a transfer. For example, if there are multiple transfer trips of the same transit route available and all of them provide transfer w.p. 1, then we should only keep the trip that provides the least waiting time. This is because a passenger would not likely wait for a different trip of the same transit route. However, in the case of capacitated assignment, for a passenger, we cannot evaluate the probability of making a transfer as it depends on the availability of space which further depends on the strategies of other passengers.

The steps for creating transfer links are summarized in Algorithm 6. It takes transfer links $A_t^{'}$ created using the criteria described in Section 3 and the type of assignment as inputs and produces the final transfer links as output. The algorithm starts by initializing the final set of transfer links A_t as an empty set and collecting all the transfer nodes in the network. Then, for each transfer node i, we find all the transit routes that can be transferred from it. For each transferring route, we find the set of nodes associated with it (*connecting_nodes*) and sort them in the increasing order of their scheduled departure time. After that, for uncapacitated assignment, we create transfer links from node i to other nodes in *connecting_nodes* starting from the one for which there exists at least one arrival time instance so that the transfer can be made successfully (i.e., with a positive probability) to the one for which all its arrival time instances can be successfully transferred from any arrival time instance of node i (i.e., the transfer is made w.p. 1). If we cannot find a node that can be transferred w.p. 1, then, we create a walking link from i to all destinations. This is done to finish the journey of travelers who find themselves in a situation where there is no outgoing link to move forward. In practice, if a passenger encounters a situation when there is no bus available at the stop, then they either walk or use another mode of transportation to get to their destination. We only assume walking in our assignment, although, one can consider other modes of transportation. In the case of capacitated assignment, we create transfer links from node i to other nodes in *connecting_nodes* for which there exists at least one arrival time instance so that transfer can be made successfully (i.e., with positive probability) and provide waiting time less than δ_3. In this case, we compulsorily create walking links to various destinations as there may not be sufficient capacity in the considered transfer options.

Algorithm 6 Creation of transfer links

procedure CreateTransfers
  Inputs: $A_t^{'}$, *assignment_type*
  Output: A_t ▷ Set of final transfer nodes
  (Initialize) $A_t \leftarrow \phi$
  $transfer\_nodes \leftarrow \{i \in N : \exists j \in FS(i) \text{ s.t. } (i,j) \in A_t^{'}\}$
  for $i \in transfer\_nodes$ **do**
    $connecting\_routes \leftarrow \{r(j) : (i,j) \in A_t^{'}\}$
    for $\hat{r} \in connecting\_routes$ **do**
      $connecting\_nodes \leftarrow \{j \in FS(i) : (i,j) \in A_t^{'}, r(j) == \hat{r}\}$
      Sort nodes in *connecting_nodes* in the increasing order of their scheduled departure time
      Find the first node m in *connecting_nodes* for which $\exists t^{'} \in \tilde{t}_{k(i)}(i), t^{''} \in \tilde{t}_{k(m)}(m)$, s.t. $t^{'} + w_{ij} \leq t^{''}, \tilde{p}_i(t^{'}) > 0, \tilde{p}_m(t^{''}) > 0$.
      if *assignment_type* == "uncapacitated" **then**
        Find the first node n in *connecting_nodes* for which $\forall t^{'} \in \tilde{t}_{k(i)}(i), t^{''} \in \tilde{t}_{k(n)}(n)$, s.t. $t^{'} + w_{ij} \leq t^{''}, \tilde{p}_i(t^{'}) > 0, \tilde{p}_n(t^{''}) > 0$, and $\sum_{t^{'} \in \tilde{t}_{k(i)}(i)} \sum_{t^{''} \in \tilde{t}_{k(n)}(n)} \tilde{p}_i(t^{'})\tilde{p}_n(t^{''}) = 1$.
        if there is no such n **then**
          n is the last node in *connecting_nodes*
          Append all the links from (i,m) to (i,n) to A_t
          Create walking links from node i to all $d \in D$ if they do not exist.
        else
          Append all the links from (i,m) to (i,n) to A_t
      else if *assignment_type* == "capacitated" **then**
        Find first node n in *connecting_nodes* for which $\forall t^{'} \in \tilde{t}_{k(i)}(i), t^{''} \in \tilde{t}_{k(n)}(n)$, s.t. $t^{'} + w_{ij} \leq t^{''}, \tilde{p}_i(t^{'}) > 0, \tilde{p}_n(t^{''}) > 0$, and $t^{''} - t^{'} - w_{ij} \leq \delta_3$
        if there is no such n **then**
          n is the last node in *connecting_nodes*
          Append all the links from (i,m) to (i,n) to A_t
        else
          Append all the links from (i,m) to (i,n) to A_t
        Create walking links from node i to all $d \in D$ if they do not exist.
      else
        Raise error

Appendix B Proofs

Proof of Lemma 1 We will show this by deriving the KKT conditions of the assignment program (8). Let us associate dual variables $\{J^d(i,t,\theta)\}_{\substack{\forall \theta \in \Theta_i(t), \forall t \in \tilde{t}_{k(i)}(i), \\ \forall i \in N, \forall d \in D}}$, $\{J^d(o,t,\theta)\}_{\substack{\forall \theta \in \Theta_o(t), \forall t \in T, \\ \forall o \in O, \forall d \in D}}$,

$\{J^d(g)\}_{\substack{\forall g\in G:d_g=d,\\ \forall d\in D}}$, $\{J^d(d)\}_{\forall d\in D}$, $\{\sigma^d(i,t,\theta,j)\}_{\substack{\forall j\in u(i,t,\theta),\\ \forall(i,t,\theta)\in S,\forall d\in D}}$, and $\{\lambda^d_{gt}\}_{\substack{\forall t\in\left[t_g^{ED},t_g^{ED}+\delta_3\right],\\ \forall g\in G,\forall d\in D}}$ to the constraints (8b)-(8g) respectively. The, the Lagrangian of (8) can be written as:

$$\mathcal{L}(\mathbf{V},\mathbf{v},\mathbf{J},\sigma,\lambda) = \sum_{d\in D}\Bigg[\sum_{(i,t,\theta)\in S}\sum_{j\in u(i,t,\theta)} v^d(i,t,\theta,j)*c^\theta_{ij} +$$

$$\sum_{\substack{\forall\theta\in\Theta_i(t),\\ \forall t\in\tilde{t}_{k(i)}(i),\forall i\in N}} J^d(i,t,\theta)*\left(p^\theta\sum_{\substack{(k,t',\theta')\in S\setminus\{d\}:i\in u(k,t',\theta')\\ \&\ t=t'+c^{\theta'}_{ki}}} v^d(k,t',\theta',i) - \sum_{j\in u(i,t,\theta)} v^d(i,t,\theta,j)\right) +$$

$$\sum_{\forall\theta\in\Theta_o(t),\forall t\in T,\forall o\in O} J^d(o,t,\theta)*\left(p^\theta\sum_{\substack{g\in G:o_g=o\ \&\\ t\in\left[t_g^{ED},t_g^{ED}+\delta_3\right]}} V^d_{gt} - \sum_{j\in u(o,t,\theta)} v^d(o,t,\theta,j)\right) +$$

$$\sum_{\forall g\in G:d_g=d} J^d(g)\left(d^{o_g d}_g - \sum_{t\in\left[t_g^{ED},t_g^{ED}+\delta_3\right]} V^d_{gt}\right) + J^d(d)*\left(\sum_{g\in G:d_g=d} d^{o_g d} - \sum_{\substack{(k,t',\theta')\in S\setminus\{d\}:\\ d\in u(k,t',\theta')}} v^d(k,t',\theta',d)\right) -$$

$$\sum_{\forall j\in u(i,t,\theta),\forall(i,t,\theta)\in S} \sigma^d(i,t,\theta,j)*v^d(i,t,\theta,j) - \sum_{\forall t\in\left[t_g^{ED},t_g^{ED}+\delta_3\right],\forall g\in G} \lambda^d_{gt}*V^d_{gt}\Bigg]$$

The KKT conditions are given below:

1. *Primal feasibility*: (8b)-(8g)
2. *Dual feasibility*:

$$\sigma^d_{i,t,\theta,j} \geq 0, \forall j\in u(i,t,\theta), \forall(i,t,\theta)\in S, \forall d\in D \tag{18}$$

$$\lambda^d_{gt} \geq 0, \forall t\in\left[t_g^{ED}, t_g^{ED}+\delta_3\right], \forall g\in G, \forall d\in D \tag{19}$$

3. *Complementary slackness*:

$$v^d(i,t,\theta,j)*\sigma^d(i,t,\theta,j) = 0, \forall j\in u(i,t,\theta), \forall(i,t,\theta)\in S$$

$$V^d_{gt}*\lambda^d_{gt} = 0, \forall t\in\left[t_g^{ED}, t_g^{ED}+\delta_3\right], \forall g\in G, \forall d\in D$$

4. *Gradient of the Lagrangian wrt primal variables vanishes*:

$$\frac{\partial\mathcal{L}(\mathbf{V},\mathbf{v},\mathbf{J},\sigma,\lambda)}{\partial v^d(i,t,\theta,j)} = c^\theta_{ij} + \sum_{\theta'\in\Theta_j(t)} p^{\theta'} J^d(j,t+c^\theta_{ij},\theta') - J^d(i,t,\theta) - \sigma^d(i,t,\theta,j) = 0, \forall j\in u(i,t,\theta), \forall(i,t,\theta)\in$$

$S, \forall d \in D$

$$\frac{\partial \mathcal{L}(\mathbf{V}, \mathbf{v}, \mathbf{J}, \sigma, \lambda)}{\partial V_{gt}^{d}} = \sum_{\theta \in \Theta_{o_g}(t)} p^{\theta} J^{d}(o, t, \theta) - J^{d}(g) - \lambda_{gt}^{d} = 0, \forall t \in \left[t_g^{ED}, t_g^{ED} + \delta_3\right], \forall g \in G, \forall d \in$$

D

Using (18) and (19), we can write above two equations as:

$$c_{ij}^{\theta} + \sum_{\theta' \in \Theta_j(t)} p^{\theta'} J^{d}(j, t + c_{ij}^{\theta}, \theta') - J^{d}(i, t, \theta) \geq 0, \forall j \in u(i, t, \theta), \forall (i, t, \theta) \in S, \forall d \in D \tag{20a}$$

$$\sum_{\theta \in \Theta_{o_g}(t)} p^{\theta} J^{d}(o, t, \theta) - J^{d}(g) \geq 0, \forall t \in \left[t_g^{ED}, t_g^{ED} + \delta_3\right], \forall g \in G, \forall d \in D \tag{20b}$$

(20a) and (20b) can further be written as:

$$J^{d}(i, t, \theta) = \min_{j \in u(i,t,\theta)} \left\{ c_{ij}^{\theta} + \sum_{\theta' \in \Theta_j(t)} p^{\theta'} J^{d}(j, t + c_{ij}^{\theta}, \theta') \right\}, \forall (i, t, \theta) \in S, \forall d \in D \tag{21a}$$

$$J^{d}(g) = \min_{t \in \left[t_g^{ED}, t_g^{ED} + \delta_3\right]} \left\{ \sum_{\theta \in \Theta_{o_g}(t)} p^{\theta} J^{d}(o, t, \theta) \right\}, \forall g \in G, \forall d \in D \tag{21b}$$

(21a) and (21b) are the Bellman equations for finding the optimal policies given in (2) and (7) respectively. This completes our proof.

Appendix C Example problem

C.1 Uncapacitated assignment

For the example given in Figure 1, let us compute the optimal cost functions and optimal policy for destination d. Clearly, $\hat{J}^*(d) = 0$. $\hat{J}^*(C_1, 17) = 1 * (1 + 0) = 1$, $\hat{J}^*(C_1, 23) = 1 * (1 + 0) = 1$,

$\hat{J}^*(C_2,16) = 1*(1+0) = 1$, $\hat{J}^*(C_2,18) = 1*(1+0) = 1$, and $\hat{J}^*(C_2,23) = 1*(1+0) = 1$.

$$
\begin{aligned}
\hat{J}^*(D_2,3) &= 1*(13+\hat{J}^*(C_2,16)) = 1*(13+1) = 14\\
\hat{J}^*(D_2,5) &= 1*(13+\hat{J}^*(C_2,18)) = 1*(13+1) = 14\\
\hat{J}^*(D_2,10) &= 1*(13+\hat{J}^*(C_2,23)) = 1*(13+1) = 14\\
\hat{J}^*(B_1,2) &= 0.2*\min\{15+\hat{J}^*(C_1,17), 1+\hat{J}^*(D_2,3)\} + 0.3*\min\{15+\hat{J}^*(C_1,17), 3+\hat{J}^*(D_2,5)\}\\
&\quad + 0.5*\min\{15+\hat{J}^*(C_1,17), 8+\hat{J}^*(D_2,10)\}\\
&= 0.2*15 + 0.3*16 + 0.5*16 = 15.8\\
\hat{J}^*(B_1,8) &= 0.5*\min\{15+\hat{J}^*(C_1,15+8), \infty\} + 0.5*\min\{15+\hat{J}^*(C_1,15+8), 2+\hat{J}^*(D_2,8+2)\}\\
&= 0.5*16 + 0.5*16 = 16\\
\hat{J}^*(A_1,0) &= 0.6*(2+15.8) + 0.4*(8+16) = 20.28\\
\hat{J}^*(E_2,0) &= 0.2*(3+14) + 0.3*(5+14) + 0.5*(10+14) = 21.1\\
\hat{J}^*(o,0) &= \min\{20.28, 21.1\} = 20.28
\end{aligned}
$$

After computing the expected cost to go from various nodes at various times, one can evaualte the optimal policy by comparing these optimal costs. These are evaluated below:

$$
\begin{array}{rll}
\mu^*(o,0,\{0,0\}) = \{A_1\} & & \\
\mu^*(A_1,0,\{2\}) = \{B_1\}, & \mu^*(A_1,0,\{8\}) = \{B_1\} & \\
\mu^*(E_2,0,\{3\}) = \{D_2\}, & \mu^*(E_2,0,\{5\}) = \{D_2\} & \mu^*(E_2,0,\{10\}) = \{D_2\}\\
\mu^*(B_1,2,\{15,1\}) = \{D_2\}, & \mu^*(B_1,2,\{15,3\}) = \{C_1\} & \mu^*(B_1,2,\{15,8\}) = \{C_1\}\\
\mu^*(B_1,8,\{15,\infty\}) = \{C_1\}, & \mu^*(B_1,8,\{15,2\}) = \{D_2,C_1\} & \\
\mu^*(D_2,3,\{13\}) = \{C_2\}, & \mu^*(D_2,5,\{13\}) = \{C_2\} & \mu^*(D_2,10,\{13\}) = \{C_2\}\\
\mu^*(C_1,17,\{1\}) = \{d\}, & \mu^*(C_1,23,\{1\}) = \{d\} & \\
\mu^*(C_2,16,\{1\}) = \{d\}, & \mu^*(C_2,18,\{1\}) = \{d\} & \mu^*(C_2,23,\{1\}) = \{d\}
\end{array}
$$

Let us assume only one group of 100 passengers moving from o to d. For the sake of simplicity, we do not consider any arrival time penalties. Obviously, $t^* = 0$. Further, we can evaluate the values of transitioning flow at various states as below:

$$
\begin{array}{rll}
v(o,0,\{0,0\},A_1)=100 & v(o,0,\{0,0\},E_2)=0 & \\
v(A_1,0,\{2\},B_1)=100*0.6=60, & v(A_1,0,\{8\},B_1)=100*0.4=40 & \\
v(E_2,0,\{3\},D_2)=0, & v(E_2,0,\{5\},D_2)=0 & v(E_2,0,\{10\},D_2)=0 \\
v(B_1,2,\{15,1\},C_1)=0, & v(B_1,2,\{15,1\},D_2)=0.2*60=12 & \\
v(B_1,2,\{15,3\},C_1)=0.3*60=18, & v(B_1,2,\{15,3\},D_2)=0 & \\
v(B_1,2,\{15,8\},C_1)=0.5*60=30 & v(B_1,2,\{15,8\},D_2)=0 & \\
v(B_1,8,\{15,\infty\},C_1)=0.5*40=20, & v(B_1,8,\{15,\infty\},D_2)=0 & \\
v(B_1,8,\{15,2\},C_1)=0.5*0.5*40=10, & v(B_1,8,\{15,2\},D_2)=0.5*0.5*40=10 & \\
v(D_2,3,\{13\},C_2)=12, & v(D_2,5,\{13\},C_2)=0 & v(D_2,10,\{13\},C_2)=10 \\
v(C_1,17,\{1\},d)=48, & v(C_1,23,\{1\},d)=30 & \\
v(C_2,16,\{1\},d)=12, & v(C_2,18,\{1\},d)=0 & v(C_2,23,\{1\},d)=10
\end{array}
$$

Computing the average link flow on various links using (9), we have, $v(o,A_1)=100, v(o,E_2)=0$, $v(A_1,B_1)=100$, $v(E_2,D_2)=0$, $v(B_1,D_2)=12+20=22$, $v(B_1,C_1)=78$, $v(D_2,C_2)=22$, $v(C_1,d)=78$, $v(C_2,d)=22$.

C.2 Network loading for capacitated assignment

Using the policy computed in the previous sub-section, we evaluate the route choice probabilities as below:

$$
\begin{array}{rll}
P_{(o,0,\{0,0\},\{1,1\}),A_1}=1 & P_{(o,0,\{0,0\},\{1,1\}),E_2}=0 & \\
P_{(o,0,\{0,0\},\{0,1\}),E_2}=1 & & \\
P_{(A_1,0,\{2\},\{1\}),B_1}=1, & P_{(A_1,0,\{8\},\{1\}),B_1}=1 & \\
P_{(E_2,0,\{3\},\{1\}),D_2}=1, & P_{(E_2,0,\{5\},\{1\}),D_2}=1 & P_{(E_2,0,\{10\},\{1\}),D_2}=1 \\
P_{(B_1,2,\{15,1\},\{1,1\}),C_1}=0, & P_{(B_1,2,\{15,1\},\{1,1\}),D_2}=1 & \\
P_{(B_1,2,\{15,3\},\{1,1\}),C_1}=1, & P_{(B_1,2,\{15,3\},\{1,1\}),D_2}=0 & \\
P_{(B_1,2,\{15,8\},\{1,1\}),C_1}=1 & P_{(B_1,2,\{15,8\},\{1,1\}),D_2}=0 & \\
P_{(B_1,8,\{15,\infty\},\{1,0\}),C_1}=1, & & \\
P_{(B_1,8,\{15,2\},\{1,1\}),C_1}=0.5, & P_{(B_1,8,\{15,2\},\{1,1\}),D_2}=0.5 & \\
P_{(D_2,3,\{13\},\{1\}),C_2}=1, & P_{(D_2,5,\{13\},\{1\}),C_2}=1 & P_{(D_2,10,\{13\},\{1\}),C_2}=1 \\
P_{(C_1,17,\{1\},\{1\}),d}=1, & P_{(C_1,23,\{1\},\{1\}),d}=1 & \\
P_{(C_2,16,\{1\},\{1\}),d}=1, & P_{(C_2,18,\{1\},\{1\}),d}=1 & P_{(C_2,23,\{1\},\{1\}),d}=1
\end{array}
$$

Further, assume that the capacity of both trips is 60. For passenger loading, let us process the nodes in topological order. For node 1, we have, $V_n(o,0) = 100$. First, initialize $\pi^{(o,0,\{0,0\},\{1,1\})} = 1$. The accessing flow and residual capacities of outgoing links are $\tilde{v}_{o,A_1} = 100, \tilde{v}_{o,E_2} = 0$ and $\tilde{u}_{o,A_1} = 60, \tilde{u}_{o,E_2} = 60$. As the flow trying to access link (o, A_1) is more than its residual capacity, we have $\beta = 0.6$. Since, we have $\beta = 0.6$, we have to run more than 1 iteration of the "while loop" to finish the loading at node A_1. As we know that $P_{(o,0,\{0,0\},\{1,1\}),A_1} = 1,\ P_{(o,0,\{0,0\},\{1,1\}),E_2} = 0$, we have $v_{(o,0,\{0,0\},\{1,1\}),A_1} = 0.6 * 100 = 60, v_{(o,0,\{0,0\},\{1,1\}),E_2} = 0$. Since all the flow that reaches A_1 want to continue on the same route, we assign $V_p(A_1, 0) = 60$. This gives us $\pi^{(o,0,\{0,0\},\{1,1\})} = 0.6$ and $\pi^{(o,0,\{0,0\},\{0,1\})} = 0.4$. After updating the state availability probabilities, we now run the second iteration of the while loop. The accessing flow and residual capacities of available outgoing links are $\tilde{v}_{o,E_2} = 40$ and $\tilde{u}_{o,E_2} = 60$. Therefore, $\beta = 1$. This means that all flow can access their first available choice, i.e., $v_{(o,0,\{0,0\},\{0,1\}),E_2} = 1.0 * 40 = 40$. Since all the flow that reaches E_2 want to continue on the same route, we assign $V_p(E_2, 0) = 40$.

The next node in the topological order is A_1. Since all the flow that needs to be assigned at this node is priority flow, we have $v_{(A_1,0,\{2\},\{1\}),B_1} = 0.6 * 60 = 36$ and $v_{(A_1,0,\{8\},\{1\}),B_1} = 0.4 * 60 = 24$. This makes $V_p(B_1, 2) = 0.8 * 36 = 28.8$ and $V_n(B_1, 2) = 0.2 * 36 = 7.2$. Similarly, $V_p(B_1, 8) = 24 * 0.5 * 1 + 24 * 0.5 * 0.5 = 18$ and $V_n(B_1, 8) = 6$. Processing the node E_2, we have $v_{(E_2,0,\{3\},\{1\}),D_2} = 0.2*40 = 8, v_{(E_2,0,\{5\},\{1\}),D_2} = 0.3*40 = 12$, and $v_{(E_2,0,\{10\},\{1\}),D_2} = 0.5*40 = 20$. Therefore, $V_p(D_2, 3) = 8, V_p(D_2, 5) = 12$, and $V_p(D_2, 10) = 20$.

The next node in the topological order is B_1. This is an important node as it has both priority as well as non-priority flow to assign. Let's start with the assignment of priority flow. We have, $v_{(B_1,2,\{15,1\},\{1,1\}),C_1} = 28.8$ and $v_{(B_1,8,\{15,1\},\{1,1\}),C_1} = 18$. Clearly, $V_p(C_1, 17) = 28.8$ and $V_p(C_1, 23) = 18$. Next, we process the non-priority flow. We have accessing flow $\tilde{v}_{(B_1,D_2)} = 7.2+6 = 13.2$, residual capacity $\tilde{u}_{(B_1,D_2)} = 60 - 40 = 20$, and $\beta = 1$. This gives $v_{(B_1,2,\{15,1\},\{1,1\}),D_2} = 7.2$ and $v_{(B_1,8,\{15,2\},\{1,1\}),D_2} = 6$. Following the same procedure, we have, $v_{(D_2,3,\{13\},\{1\}),C_2} = 15.2,\ v_{(D_2,3,\{5\},\{1\}),C_2} = 12,\ v_{(D_2,3,\{10\},\{1\}),C_2} = 26$.

Calculating the average flow, we have, $v(o, A_1) = 60, v(o, E_2) = 40$, $v(A_1, B_1) = 60$, $v(E_2, D_2) = 40$, $v(B_1, D_2) = 13.2$, $v(B_1, C_1) = 46.8$, $v(D_2, C_2) = 53.2$, $v(C_1, d) = 46.8$, $v(C_2, d) = 53.2$.

Appendix D Notations used in the article

Table 3: Sets, parameters, decision variables and functions used in the current article

		Sets
$G(N, A)$	$\triangleq$	SB transit network where, N denotes the set of nodes and A denotes the set of links
T	$\triangleq$	Set of time intervals during the study period
$\mathfrak{B}$	$\triangleq$	Set of transit stops/stations
R	$\triangleq$	Set of transit routes
K	$\triangleq$	Set of transit trips
O	$\triangleq$	Set of origins
D	$\triangleq$	Set of destinations
B	$\triangleq$	Set of transit nodes
G	$\triangleq$	Set of passenger groups
A_a, A_v, A_t	$\triangleq$	Set of access/egress, in-vehicle, and transfer links
$FS(i), BS(i)$	$\triangleq$	Set of outgoing and incoming links
$\Theta_i(t)$	$\triangleq$	Set of possible information vectors at node i and time t
$X_i^{\theta}(t)$	$\triangleq$	Set of availability vectors at node i, time t, and information θ
S	$\triangleq$	State space in uncapacitated assignment
S_C	$\triangleq$	State space in capacitated assignment
		Parameters
$\delta_0, \delta_1, \delta_2, \delta_3$	$\triangleq$	Maximum acceptable time for access, egress, transferring, and waiting.
$t_g^{ED}, t_g^{EA}, t_g^{LA}$	$\triangleq$	Earliest departure, earliest arrival, and latest arrival time of group g
d_g^{od}	$\triangleq$	Demand from o to d of group g
c_{ij}^{θ}	$\triangleq$	Travel time between i and j for information θ
p^{θ}	$\triangleq$	Probability of observing information θ
η_1, η_2	$\triangleq$	Early and late arrival penalty
u_{ij}	$=$	Capacity of link (i, j)
		Decision Variables
$v^d(s, j)$	$=$	Number of passengers going to destination d arriving at state s taking action j
V_{gt}^d	$=$	Number of passengers going to destination d of group g departing at time t
v_{ij}	$=$	Aggregated average passenger flow on link (i, j)
t_g^*	$=$	Optimal departure time for group g
π^x	$=$	Probability of observing availability vector x
ϵ	$=$	Tolerance parameter used for the convergence of MSA algorithm.

Functions

$\hat{t}_k$	=	Arrival time of transit trip at a stop
$\tilde{t}_k$	=	Possible arrival times of transit trip at a stop
$\tilde{p}_i(t)$	=	Probability of bus arriving at node i at time t
$k(i)$	=	Trip associated with transit node i
$r(i)$	=	Route associated with transit node i
$w(i,j)$	=	Walking time between node i and j
$\gamma_{k(i)}$	=	Sequence of node i for trip k
μ	=	A stationary policy that specifies action to take at every node
J	=	Expected cost function
$Q(s,j)$	=	Expected cost of taking action j at state s
$\mathbf{P},\mathbf{R}$	=	Route choice and departure time choice probability function
$C(\mathbf{P},\mathbf{R})$	=	Expected cost of choice probability vector $(\mathbf{P},\mathbf{R})$