Distributionally Robust Fair Transit Resource Allocation During a Pandemic

Weijun Xie
Department of Industrial & Systems Engineering, Virginia Tech, Blacksburg, VA 24061, wxie@vt.edu.

Luying Sun
Department of Industrial & Systems Engineering, Virginia Tech, Blacksburg, VA 24061, luyingsun@vt.edu.

Tim Witten
Blacksburg Transit, Blacksburg, VA 24060, TWitten@blacksburg.gov.

This paper studies Distributionally robust Fair transit Resource Allocation model (DrFRAM) under Wasserstein ambiguity set to optimize the public transit resource allocation during a pandemic. We show that the proposed DrFRAM is highly nonconvex and nonlinear and is, in general, NP-hard. Fortunately, we show that DrFRAM can be reformulated as a mixed-integer linear programming (MILP) by leveraging the equivalent representation of distributionally robust optimization and monotonicity properties, binarizing integer variables, and linearizing nonconvex terms. To improve the proposed MILP formulation, we derive stronger ones and develop valid inequalities by exploiting the model structures. Besides, we develop scenario decomposition methods using different MILP formulations to solve the scenario subproblems and introduce a simple yet effective No-one-left based approximation algorithm with a provable approximation guarantee to solve the model to near optimality. Finally, we numerically demonstrate the effectiveness of the proposed approaches and apply them to real-world data provided by the Blacksburg Transit.

Key words: Distributionally Robust, Mixed-Integer Programming, Strong Formulations, Valid Inequalities

History:

1. Introduction

The recent outbreak of the SARS-CoV-2 (i.e., COVID-19 disease) has been profoundly influencing society. Many schools have been shut down and turning to online; many restaurants, shopping malls have been either closed or significantly reducing the capacity; many firms have required employees to work from home. On the other hand, as the process of the vaccination has been accelerated and the promise of the vaccines, the number of COVID-19 infection cases in the U.S. has been decreasing since January 2021. Therefore, many states are now re-opening the business as well as schools; for example, the Commonwealth of Virginia’s public schools have been re-opened since March 2021. Undoubtedly, these policies have tremendously influenced peoples outdoor activities, which causes a significant demand uncertainty for public transit. Besides demand uncertainty, to keep passengers safe and prevent transmission of the coronaviruses, it is of vital importance to observe the social distancing within the public transit, which in turn will reduce the capacity of
transit vehicles (e.g., buses) and thus affect the entire operations of the public transit systems. This paper is trying to solve this issue. The results in this paper are also useful for another wave of COVID-19 outbreaks or preparing for a future pandemic.

The authors are collaborating with local Blacksburg Transit, a daily outdoor transportation tool used by many students and residents at the Town of Blacksburg. Since the training of employees and installation of bus stops can cause much longer delays, the transit operators in Blacksburg Transit would like to keep the current routes and schedules as they are but to optimize the transit resources to maximize their utilization rates as well as social equity. This motivates us to study the fair transit resource allocation problem. Like many modern transit systems, Blacksburg Transit stores the passengers’ alighting and boarding data, which allows us to build a data-driven optimization model using their datasets.

1.1. Literature Reviews

Transit system design and optimization has been widely studied in the literature (see, e.g., Daganzo and Ouyang 2010, Ceder 2016 and references therein). The works in transit are mainly on optimal scheduling and routing to serve the public better. For example, Chien and Schonfeld (1997) incorporated spatial characteristics and demand patterns of urban areas into their optimization model. Other important earlier works can be found in De Cea and Fernández (1993), Quadrifoglio et al. (2006), Guan et al. (2006), Jeihani et al. (2013) or in the review papers (Guihaire and Hao 2008, Hörcher and Tirachini 2021). Recently, Nourbakhsh and Ouyang (2012) focused on the flexible transit design on low-demand areas, and Ouyang et al. (2014) proposed continuum approximation approach for bus design under heterogeneous demand. Chen and Nie (2017) attempted to integrate an e-hailing system with public transit. Iliopoulou and Kepaptsoglou (2019) jointly optimized network design and charging infrastructure locations for electric buses. Abdolmaleki et al. (2020) built a new model for synchronizing timetables in a transit network that minimizes the total transfer waiting time. Other interesting works can be found in Román-De la Sancha et al. (2018), Matisziw et al. (2006), Jin et al. (2016), Agarwal and Ergun (2008), Mahéo et al. (2019), among many others. Different from these works, we are focusing on transit resource allocation given the predetermined routes.

Recent advances in stochastic and robust optimization provide tools for the transit design under uncertainty (see, e.g., Li et al. 2009, Fernandes et al. 2018, Li et al. 2008, Fu and Lam 2014, Hamdouch et al. 2014, Hadas and Shnaiderman 2012 for the earlier interesting works). For example, Kulshrestha et al. (2014) developed a robust model for bus dispatching for evacuations to optimize the total travel time under demand uncertainty. An and Lo (2016) studied transit system design under demand uncertainty using two-stage stochastic programming. Liang et al. (2019) proposed
a two-step modeling framework for the bus transit network design considering an existing metro network and demand uncertainty. In Yoon and Chow (2020), the authors provided a sequential learning framework to address the demand uncertainty when designing lines and routes in the transit system. Different from existing ones, our modeling framework is data-driven without any knowledge of the underlying distribution, which uses real-world data from a transit system. Besides, to fulfill the requirements of transit operators, our proposed model does not alter the existing lines and routes.

The study of transit design during a pandemic, especially the COVID-19 pandemic, is rather limited. As shown in Liu et al. (2020), the COVID-19 pandemic caused a major transit demand decline for many public transit systems in the United States due to the fear of contracting the disease, practicing social distancing, and lockdown policy, which has also been observed by many social workers. Mo et al. (2021) studied the epidemic spreading model in the time-varying transit network. Chen et al. (2020) redesigned the campus bus systems by shortening the routes and enforcing social distancing. The most relevant to our research are Gkiotsalitis and Cats (2021), Yang and Nie (2020), which optimized metro service frequency during the COVID-19 pandemic. Different from Gkiotsalitis and Cats (2021), Yang and Nie (2020), we not only optimize the number of buses assigning to different routes but also try to stick to one type of buses per route to avoid drivers unnecessary confusion. More importantly, we incorporate both demand uncertainty and alighting rate uncertainty, and thus our proposed model is more flexible and data-driven. The scope of our model is also different from Gkiotsalitis and Cats (2021), Chen et al. (2020). Instead of minimizing the operation costs or maximizing the profits, we focus on optimizing social equity for each route and minimizing the passenger abandon rate due to capacity restrictions.

1.2. Summary of Main Contributions

The objective of this study, motivated by Blacksburg Transit, is to determine optimal transit resource allocation to minimize the highest utilization rate and the largest passenger abandon rate to achieve better social equity under stochastic passenger arrival and alighting rates. This gives rise to Distributionally robust Fair transit Resource Allocation model (DrFRAM) under a data-driven ambiguity set. The main contributions of DrFRAM are summarized as below:

(i) We study DrFRAM under type-$\infty$ Wasserstein ambiguity set. We prove that DrFRAM cannot be solved in polynomial time unless $P = NP$ even when the problem is deterministic and there are only two types of buses.

(ii) We derive monotonicity properties of the DrFRAM, using which we can significantly simplify the DrFRAM and linearize the nonlinear components in the objective of DrFRAM.
(iii) We propose a mixed-integer linear programming (MILP) formulation for DrFRAM by binarizing integer variables and linearizing the nonlinear functions using McCormick inequalities. To strengthen the MILP formulation, we derive stronger ones and develop valid inequalities by exploiting the model structures. These formulations allow us to design efficient scenario decomposition methods.

(iv) Besides these exact methods, we develop an approximate scenario decomposition method using objective cuts to speed up the convergence and No-one-left based approximation algorithm for solving DrFRAM to near-optimality. Both approaches come with theoretical approximation guarantees, demonstrating the strengths of these approaches.

(v) We numerically demonstrate the effectiveness of the proposed approaches and apply them to solve the real-world instances using the data provided by Blacksburg Transit.

**Notation.** The following notation is used throughout the paper. We use bold-letters (e.g., $\mathbf{x}, \mathbf{A}$) to denote vectors and matrices and use corresponding non-bold letters to denote their components. Given a vector or matrix $\mathbf{x}$, its zero norm $\|\mathbf{x}\|_0$ denotes the number of its nonzero elements. We let $\mathbf{e}$ be the vector or matrix of all ones, and let $\mathbf{e}_i$ be the $i$th standard basis vector. Given an integer $n$, we let $[n] := \{1, 2, \ldots, n\}$, and use $\mathbb{R}^n_+ := \{\mathbf{x} \in \mathbb{R}^n : x_i \geq 0, \forall i \in [n]\}$. Given a real number $t$, we let $(t)_+ := \max\{t, 0\}$. Given a finite set $I$, we let $|I|$ denote its cardinality. We let $\tilde{\xi}$ denote a random vector and denote its realizations by $\xi$. We use superscript $k \in [N]$ denote the index of scenario $k$. For a matrix $\mathbf{A}$, we let $\mathbf{A}_i$ denote $i$th row of $\mathbf{A}$ and $\mathbf{A}_j$ denote $j$th column of $\mathbf{A}$. Additional notation will be introduced as needed.

**Organization.** The remainder of the paper is organized as follows. In Section 2, we introduce the model formulation. Sections 3 and 4 show model properties and derive an equivalent mixed-integer linear programming formulation. Section 5 shows convexification of two substructures and derivation of strong valid inequalities and Section 6 develops approximation algorithms. Finally, Section 7 numerically demonstrates the effectiveness of the proposed solution approaches and Section 8 concludes this paper.

### 2. Model Formulation

This paper concerns a Distributionally robust Fair transit Resource Allocation model (DrFRAM). For the strategic planning purpose, the modeling of a transit system is based on the seminal work in Hadas and Shnaiderman (2012) using aggregate arrival rates and alighting rates and not relying on the user-specific origin-destination data, since the former is easy to obtain and the latter is difficult to get access to in the transit data available to us. We also take Blacksburg Transit, which only provides us arrival and boarding data, into consideration. In DrFRAM, there are $K$ different types of buses, where each bears a nominal capacity $c_k$ for each $k \in [K]$, and there are $\eta_k$ number of type-$k$
buses. Note that during a pandemic, to observe the social distancing strictly, the bus capacities will be decreased proportionally. Thus, let a pandemic factor \( \delta_k \in [0, 1] \) denote the percent of a type-\( k \) bus capacity that allows being operated. That is, only \( \lfloor \delta_k c_k \rfloor \) number of passengers will be allowed onto the bus during a pandemic. Suppose that there are \( I \) distinct bus routes, and for each bus route \( i \in [I] \), there are \( J_i \) bus stops. Since passengers can alight and depart a bus at any bus stop for each \( i \in [I] \) and \( j \in [J_i] \), we let the random parameter \( \tilde{a}_{ij} \in [0, 1] \) denote the proportion of passengers alighting from the bus and let the random variable \( \tilde{\lambda}_{ij} \in \mathbb{Z}^+ \) represent the passenger arrival rate during a unit time interval. Note that \( \tilde{a}, \tilde{\lambda} \) are both random since they can vary over time. For notational convenience, we let \( \tilde{\xi} = (\tilde{a}, \tilde{\lambda}) \). Since the historical data are available to us, in DrFRAM, we use the Wasserstein distance to characterize the random parameters, which can both ensure the model to data-driven and robust (see, e.g., Kuhn et al. 2019).

The transit operators would like to know that given the predetermined routes (so they should not re-train their employees), what is the fairest way to allocate buses so that the passengers will be the best serviced and the safety policies should be strictly enforced? To help them make a better decision, we let the integer variable \( n_{ik} \in \mathbb{Z}^+ \) denote the number of type-\( k \) \( \in [K] \) buses being allocated to route \( i \in [I] \). Due to consistency and for ease of management, the transit operators would like to ensure that each route is subject to the same type of buses whenever possible. Thus, we let the binary variable \( x_{ik} \in \{0, 1\} \) denote whether type-\( k \) buses will be assigned to route \( i \in [I] \) or not. Using this notation, we see that \( \sum_{k \in [K]} x_{ik} \lfloor \delta_k c_k \rfloor \) represents the bus capacity at route \( i \in [I] \).

In our model, given a realization \( \xi \) of the random parameters \( \tilde{\xi} \), we let \( L_{ij}(\xi) \) denote the number of passengers remaining on a bus right after \( j \)th stop at route \( i \in [I] \). The operation managers would like to know an optimal way to allocate the buses, which is fair to passengers from different routes. That is, they want to ensure both the highest utilization rate among all the routes and the largest abandon rate due to limited bus capacity to be low.

With the notation introduced above, we are ready to present the mathematical formulation of DrFRAM as below:

\[
\text{(DrFRAM)} \quad \begin{align*}
\v^* &= \min_{n, x} \sup_{\tilde{\xi} \in P} \mathbb{E}_{\tilde{\xi}} \left[ Q(n, x, \tilde{\xi}) \right], \\
\text{s.t.} \quad &\sum_{k \in [K]} x_{ik} = 1, \quad \forall i \in [I], \\
&n_{ik} \leq \eta_k x_{ik}, \quad \forall i \in [I], \forall k \in [K], \\
&\sum_{i \in [I]} n_{ik} \leq \eta_k, \quad \forall k \in [K], \\
&\sum_{i \in [I]} n_{ik} \geq 1, \quad \forall k \in [K], \\
x_{ik} \in \{0, 1\}, \quad n_{ik} \in \mathbb{Z}^+, \quad \forall i \in [I], \forall k \in [K].
\end{align*}
\]
In DrFRAM (1), the objective (1a) is to minimize the worst-case resource planning outcomes, where ambiguity set \( \mathcal{P} \) denotes a family of probability distributions, and \( Q(n, x, \xi) \) denotes the random recourse function, which will be specified later. Constraints (1b) ensure that each route will commit to one type of bus. Constraints (1c) and (1d) jointly show that the number of buses allocated to a particular route is no larger than the number of available buses. Constraints (1e) show that to achieve social equity, each route should have at least one bus. Constraints (1f) specify the boundaries of the decision variables.

Given a realization \( \xi \) of random parameters \( \tilde{\xi} \) and the values of first-stage decisions \((n, x)\), we can express the recourse function in the following way:

\[
Q(n, x, \xi) = \min_{L(\xi)} Q(n, x, \xi, L(\xi)) : = \left\{ \begin{array}{l}
\max_{i \in [I], j \in [J]} \frac{L_{ij}(\xi)}{\sum_{k \in [K]} n_{ik} [\delta_k c_k]} \\
+ \omega \max_{i \in [I], j \in [J]} \max_{1, \lambda_{ij}} \left[ 0, 1 + \frac{1}{\lambda_{ij}} \left( (1 - a_{ij}) L_{i,j-1}(\xi) - \sum_{k \in [K]} n_{ik} [\delta_k c_k] \right) \right]
\end{array} \right.
\]

s.t. \[
L_{ij}(\xi) = \min \left\{ \sum_{k \in [K]} n_{ik} [\delta_k c_k], \left( (1 - a_{ij}) L_{i,j-1}(\xi) + \lambda_{ij} \right) \right\}, \forall i \in [I], \forall j \in [J],
\]

\[
L_{i0}(\xi) = 0, L_{ij}(\xi) \in \mathbb{Z}^+, \forall i \in [I], \forall j \in [J],
\]

where \( \omega \geq 0 \) is the weight that balances the importance of the highest bus utilization rate and the largest passenger abandon rate. The objective (2a) is to minimize the weighted highest utilization rate and the largest passenger abandon rate. The objective (2a) is to minimize the weighted highest utilization rate and the largest passenger abandon rate. The objective (2) is to enhance the social equity. Note that we choose the min-max fairness measure in (2), widely-used in the fairness related literature (Radunovic and Le Boudec 2007, Du et al. 2017, Wu et al. 2020). Constraints (2b) postulate the number of passengers on board at each stop, which follows the convention from the existing transit literature (see, e.g., Hadas and Shnaiderman 2012). Constraints (2c) specify the boundary conditions of the recourse decisions, i.e., the initialization and integrality of variables \( L(\cdot) \).

### 2.1. Wasserstein Ambiguity Set

In this subsection, we briefly introduce the notion of Wasserstein ambiguity set and its attractive properties, which are suitable for DrFRAM. In this work, there are historical data which were collected during the operations. Thus, given an empirical distribution \( \mathbb{P}_{\tilde{\xi}} \) constructed using i.i.d. historical data \( Z = \{ \tilde{\xi}^l = (\tilde{a}^l, \tilde{\lambda}^l) \}_{l \in [N]} \) such that \( \mathbb{P}_{\tilde{\xi}}\{ \tilde{\xi} = \tilde{\xi}^l \} = 1/N \), this paper considers the data-driven Wasserstein ambiguity set (see, e.g., Gao and Kleywegt 2016, Blanchet et al. 2019a, Esfahani...

\[
P_q = \{\mathbb{P} : W_q(\mathbb{P}, \mathbb{P}_{\xi}) \leq \theta\},
\]

(3)

where \(\theta \geq 0\) denotes the Wasserstein radius and for any \(q \in [1, \infty]\), the Wasserstein distance \(W_q(\cdot, \cdot)\) is defined as

\[
W_q(\mathbb{P}_1, \mathbb{P}_2) = \inf \left\{ q \sqrt{\int_{\Xi \times \Xi} \|\xi_1 - \xi_2\|^q \text{d}Q(\xi_1, \xi_2) : Q \text{ is a joint distribution of } \xi_1 \text{ and } \xi_2 \text{ with marginals } \mathbb{P}_1 \text{ and } \mathbb{P}_2, \text{ respectively} \right\},
\]

where the support

\[
\Xi = \{(a, \lambda) : a_{ij} \in [0, 1], \lambda_{ij} \in \mathbb{Z}^+, \forall i \in [I, \forall j \in [J_i]\}
\]

When \(q = \infty\), it reduces to the \(\infty\)-Wasserstein distance

\[
W_{\infty}(\mathbb{P}_1, \mathbb{P}_2) = \inf \left\{ Q \text{-ess sup } \|\xi_1 - \xi_2\| : Q \text{ is a joint distribution of } \xi_1 \text{ and } \xi_2 \text{ with marginals } \mathbb{P}_1 \text{ and } \mathbb{P}_2, \text{ respectively} \right\}.
\]

Above, \(Q\text{-ess sup } \|\cdot\|\) is the essential supreme of \(\|\cdot\|\) with respect to the joint distribution \(Q\), which is formally defined as

\[
Q \text{-ess sup } \|\xi_1 - \xi_2\| = \inf \left\{ \Delta : Q(\|\xi_1 - \xi_2\| > \Delta) = 0 \right\}.
\]

It has been shown in Bertsimas et al. that its corresponding type \(\infty\)-Wasserstein ambiguity set \(\mathcal{P}_\infty\) has the following equivalent form

\[
\mathcal{P}_\infty = \left\{ \frac{1}{N} \sum_{\ell \in [N]} \chi(\tilde{\xi} - \xi^\ell) : \exists \xi^\ell \in \Xi, \|\xi^\ell - \tilde{\xi}\| \leq \theta, \forall \ell \in [N] \right\},
\]

(4)

where \(\chi(\cdot)\) is the Dirac delta function. This neat representation has a straightforward interpretation, i.e., the worst-case distribution is also supported by \(N\) points and each support point can only deviate at most \(\theta\) amount from one of the empirical data \(Z = \{\tilde{\xi}^k\}_{k \in [N]} \subset \Xi\).

Since \(W_{\infty}(\cdot, \cdot)\) has the similar statistical performance as \(W_q(\cdot, \cdot)\) with \(q \in [1, \infty)\) (Fournier and Guillin 2015, Trillos and Slepčev 2015, Xie 2020, Xie et al. 2021) but has more attractive computational properties, we focus on \(\infty\)-Wasserstein ambiguity set, i.e., throughout this paper, we suppose that \(\mathcal{P} = \mathcal{P}_\infty\).

3. Model Properties

This section explores the properties of DrFRAM (1) and shows its computational complexity.

According to the worst-case characterization in (4) of type-\(\infty\) Wasserstein ambiguity set \(\mathcal{P} = \mathcal{P}_\infty\). DrFRAM (1) can be equivalently represented as the following deterministic counterpart:

\[
(DrFRAM) \quad v^* = \min_{n, x} \left\{ \frac{1}{N} \sum_{\ell \in [N]} \sup_{\xi^\ell \in \Xi, \|\xi^\ell - \tilde{\xi}\| \leq \theta} Q(\mathbf{n}, \mathbf{x}, \tilde{\xi}) : (1b) - (1f) \right\}.
\]

(5)
Model (5) can be interpreted as a robustification of Sampling Average Approximation (SAA) by choosing the worst-case perturbation for each empirical sample. Our model properties are based on this reformulation, and we finally choose a particular norm $\| \cdot \|$ in Section 3.2 to further simplify DrFRAM (5).

### 3.1. Computational Complexity of DrFRAM (5)

We observe that DrFRAM (5) is a mixed-integer nonlinear program and thus expect that it is an NP-hard problem. This motivates us to derive mixed integer programming techniques and approximation algorithms to solve it.

The NP-hardness of DrFRAM (5) is established upon the well-known NP-complete problem – partition problem.

**Proposition 1** Solving DrFRAM (5) is NP-hard even when $N = 1, K = 2, \theta = 0$.

**Proof:** Let us consider the following NP-complete problem.

*(Partition Problem)* Given an integer $T \in \mathbb{Z}_+$, consider $n$ positive numbers $\{\alpha_i\}_{i \in [T]} \subseteq \mathbb{Z}_+$ having an even sum, is there a partition $S_1, S_2$ such that $\sum_{i \in S_1} \alpha_i = \sum_{i \in S_2} \alpha_i = \beta, S_1 \cap S_2 = \emptyset, S_1 \cup S_2 = [T]$?

We show that finding a feasible solution of a special case of DrFRAM (5) can be reduced to the partition problem. First of all, suppose there is only $N = 1$ empirical sample and the Wasserstein radius $\theta = 0$. We also assume that there are $K = 2$ types of buses, there are $\eta_k = \beta$ buses for each $k \in [K]$ and the reduced bus capacity is $[\delta_k c_k] = C > \max_{i \in [T]} a_i$. Suppose that there are $I = T$ number of routes and for each route $i \in [I]$, there is only $J_i = 1$ stop and the empirical alight rate $\bar{a}_{i1} = 1$ and arrival rate $\bar{\lambda}_{i1} = a_i$. Under this setting, DrFRAM (5) can be reduced to the following optimization problem:

\[
\begin{align*}
v^* &= \min_{n,x} \max_{i \in [T]} \frac{a_i}{C \sum_{k \in [2]} n_{ik}}, \\
\text{s.t.} & \quad \sum_{k \in [2]} x_{ik} = 1, \forall i \in [T], \\
& \quad n_{ik} \leq \beta x_{ik}, \forall i \in [T], \forall k \in [2], \\
& \quad \sum_{i \in [T]} n_{ik} \leq \beta, \forall k \in [2], \\
& \quad \sum_{i \in [T]} n_{ik} \geq 1, \forall k \in [2], \\
& \quad x_{ik} \in \{0, 1\}, n_{ik} \in \mathbb{Z}^+, \forall i \in [T], \forall k \in [2].
\end{align*}
\]

Thus, in model (6), the optimal value $v^* \leq \frac{1}{\beta}$ if and only if there exists a feasible solution $(n, x)$ satisfying the $\sum_{k \in [2]} n_{ik} \geq a_i$ for all $i \in [I]$. Or equivalently, the following set is nonempty:

\[
\mathcal{T} := \{(n, x) : n_{ik} \geq a_i x_{ik}, \forall i \in [T], \forall k \in [2], (6b) - (6f)\}.
\]
Note that $2\beta = \sum_{i \in \mathcal{T}} a_i$. Thus, in set $\mathcal{T}$, we must have $n_{ik} = a_i x_{ik}$. Hence, set $\mathcal{T}$ is nonempty if and only if the following integer program is nonempty:

$$
\sum_{i \in \mathcal{T}} a_i x_{ik} = \beta, \quad \sum_{k \in [2]} x_{ik} = 1, \quad \forall i \in [T], \forall k \in [2],
$$

which is exactly equivalent to the partition problem.

Hence, checking the special case (6) having a feasible solution with its objective value no larger than $1/C$ is equivalent to solving the partition problem. This proves the NP-hardness of DrFRAM (5). \hfill \Box

Proposition 1 shows that even under simple settings, DrFRAM (5) may not be polynomial solvable. This motivates us to develop exact and approximation algorithms to solve it, which serve different purposes. For example, exact algorithms can be useful to verify the correctness of approximation algorithms and can solve many instances to optimality, while approximation algorithms, usually more scalable, can provide high-quality initial solutions to speed up the exact ones.

### 3.2. Theoretical Sensitivity Analyses and Model Simplification

In this subsection, we analyze the properties of the recourse function $Q(n, x, \xi)$ and the second-stage objective function $Q(n, x, \xi, L)$. These properties allow us to simplify DrFRAM (5) by linearizing the nonlinear objective function.

We begin with the results for the second-stage objective function $Q(n, x, \xi, L)$.

**Proposition 2** For the second-stage objective function $Q(n, x, \xi, L)$, we have the following sensitivity results:

- Function $Q(n, x, \xi, L)$ is monotone non-decreasing as the recourse decision $L_{ij}$ increases for some $i \in [I]$ and $j \in [J]$;
- Function $Q(n, x, \xi, L)$ is monotone non-increasing as the first-stage decision $n_{ik}$ increases for some $i \in [I]$ and $k \in [K]$;
- Function $Q(n, x, \xi, L)$ is monotone non-decreasing as the passenger arrival rate $\lambda_{ij}$ increases for some $i \in [I]$ and $j \in [J]$; and
- Function $Q(n, x, \xi, L)$ is monotone non-increasing as the passenger alighting rate $a_{ij}$ increases for some $i \in [I]$ and $j \in [J]$.

**Proof:** We prove the results according to their orders.

- The first monotonicity result is simply because when the number of passengers on the buses increases, both bus utilization rate and passenger abandon rate grow or stay the same.
The second monotonicity result is because when the number of buses allocated to a particular route increases, both its bus utilization rate and passenger abandon rate decrease or stay the same.

The third, the fourth monotonicity results are because when more passengers arrive at a bus stop of a particular route, the route’s bus utilization rate and passenger abandon rate increase or stay the same, decrease or stay the same, respectively.

Since the minimization operator does not change the monotonicity of a function. Hence, results in Parts (ii)-(iv) still hold for the recourse function $Q(n, x, \xi)$.

**Corollary 1** For the recourse function $Q(n, x, \xi)$, we have the following sensitivity results:

- Recourse function $Q(n, x, \xi)$ is monotone non-increasing as the first-stage decision $n_{ik}$ increases for some $i \in [I]$ and $k \in [K]$;

- Recourse function $Q(n, x, \xi)$ is monotone non-decreasing as the passenger arrival rate $\lambda_{ij}$ increases for some $i \in [I]$ and $j \in [J_i]$; and

- Recourse function $Q(n, x, \xi)$ is monotone non-increasing as the passenger alighting rate $a_{ij}$ increases for some $i \in [I]$ and $j \in [J_i]$.

The results in Corollary 1 inspire us to choose a proper norm $\| \cdot \|$ in the type-\(\infty\) Wasserstein ambiguity set. Specifically, since random parameters $\tilde{a}$ and $\tilde{\lambda}$ have different magnitude, we choose the weighted $\ell_\infty$ norm as $\|\xi\| = \max\{\gamma\|a\|_\infty, \|\lambda\|_\infty\}$ (recall that we let $\xi = (a, \lambda)$), where the positive weight $\gamma > 0$ is to demonstrate the balance of the importance of both parameters. In practice, one can choose a proper $\gamma$ such that the scaled vector $\gamma a$ and vector $\lambda$ have the similar order of magnitude.

In the spirits of Corollary 1 and the weighted $\ell_\infty$ norm, DrFRAM (5) admits the following equivalent representation.

**Proposition 3** Suppose $\|\xi\| = \max\{\gamma\|a\|_\infty, \|\lambda\|_\infty\}$ for some $\gamma > 0$, then we have

\[
\text{(DrFRAM)} \quad v^* = \min_{n, x} \left\{ \frac{1}{N} \sum_{\ell \in [N]} Q(n, x, \hat{\xi}^\ell) : (1b) - (1f) \right\},
\]

where $\hat{\xi}^\ell = (\hat{a}^\ell, \hat{\lambda}^\ell)$ with $\hat{a}_{ij}^\ell = \max\{0, \bar{a}_{ij}^\ell - \theta / \gamma\}$ and $\hat{\lambda}_{ij}^\ell = \max\{0, \bar{\lambda}_{ij}^\ell + [\theta]\}$.

**Proof:** The simplification of the inner supremums in DrFRAM (5) follows from the monotonicity results in Corollary 1 and the definition of the weighted $\ell_\infty$ norm.

Proposition 3 removes an obstacle in DrFRAM (5) by finding a closed-form worst-case representation for each scenario. Thus, we will mainly focus on addressing the second obstacle, which is the nonlinearity and non-convexity in the second-stage problem.
4. Linearizing the Second-stage Problem

As mentioned in the previous section, the difficulty of DrFRAM (7) resides in the second-stage problem (2) (i.e., the representation of the recourse function). The linearization technique is mainly based on the fact that the product of a binary variable and a bounded continuous variable can be linearized using the well-known McCormick inequalities (McCormick 1976), i.e.,

\[
\{xy : x \in \{0,1\}, y \in [l,u]\} = \{z : y - u(1-x) \leq z \leq y - l(1-x), lx \leq z \leq ux, x \in \{0,1\}, y \in [l,u]\}.
\]

Hence, to begin with, we first binarize the first-stage integer variables \(n\) as

\[
n_{ik} = \sum_{r \in R_k} 2^{r-1} u_{ikr}, \forall i \in [I], \forall k \in [K],
\]

where we let \(R_k = [\log (\eta_k)]\) for each \(k \in [K]\) denote the largest possible bit when representing binary variable \(n_{ik}\).

4.1. Linearizing Constraints (2b)

In this subsection, we focus on linearizing the constraints (2b) when \(\xi = \tilde{\xi}_\ell\) for each \(\ell \in [N]\). First, to suppress the notation, we let \(L^\ell := L(\tilde{\xi}_\ell)\).

Before we linearize the constraints (2b), we observe that due to the monotonicity results in Part (i) of Proposition 2, constraints (2b) are equivalent to

\[
L^\ell_{ij} \geq \min \left\{ \sum_{k \in [K]} n_{ik} [\delta_k c_k], [(1 - \tilde{a}^\ell_{ij}) L^\ell_{i,j-1}] + \lambda_{ij} \right\}, \forall i \in [I], \forall j \in [J].
\]

Since the ceiling function is lower semi-continuous, we can replace \([(1 - a_{ij}) L^\ell_{i,j-1}]\) by its epigraph variable \(\bar{L}^\ell_{i,j-1}\) such that

\[
\bar{L}^\ell_{i,j-1} \geq (1 - \tilde{a}^\ell_{ij}) L^\ell_{i,j-1}, \bar{L}^\ell_{i,j-1} \in \mathbb{Z}_+, \forall i \in [I], \forall j \in [J].
\]

In this way, constraints (9) can be further reformulated as

\[
L^\ell_{ij} \geq \min \left\{ \sum_{k \in [K]} n_{ik} [\delta_k c_k], \bar{L}^\ell_{i,j-1} + \tilde{\lambda}^\ell_{ij} \right\}, \forall i \in [I], \forall j \in [J],
\]

Above, the piecewise minimum function can be linearized using binary variables \(y^\ell\), indicating whether each stop is fully occupied or not:

\[
L^\ell_{ij} \geq \sum_{k \in [K]} n_{ik} [\delta_k c_k] - My^\ell_{ij}, L^\ell_{ij} \geq \bar{L}^\ell_{i,j-1} + \tilde{\lambda}^\ell_{ij} - M(1 - y^\ell_{ij}), y^\ell_{ij} \in \{0,1\}, \forall i \in [I], \forall j \in [J],
\]

\[
L^\ell_{ij} \geq \sum_{k \in [K]} n_{ik} [\delta_k c_k] - (M - \tilde{\lambda}^\ell_{ij}) y^\ell_{ij}, L^\ell_{ij} \geq \bar{L}^\ell_{i,j-1} + \tilde{\lambda}^\ell_{ij} y^\ell_{ij}, y^\ell_{ij} \in \{0,1\}, \forall i \in [I], \forall j \in [J],
\]

where we choose \(M = \max_{k \in [K]} \eta_k [\delta_k c_k]\) to be the maximum number of passengers that a route can carry at the same time. The equivalence of (9) and (11) is due to the fact that \(\bar{L}^\ell_{i,j-1} \leq L^\ell_{i,j-1}\).
4.2. Linearizing the Second-stage Objective Function (2a)

For each \( \ell \in [N] \), let us use \( E_1^\ell, E_2^\ell \) to denote the first and second parts of the objective function (2a). Then according to constraints (10), equivalently, the second-stage objective function can be rewritten as

\[
Q(n, x, \xi^\ell, \hat{L}^\ell) = E_1^\ell + \omega E_2^\ell
\]

where

\[
E_1^\ell \left( \sum_{k \in [K]} n_{ik} \delta k c_k \right) \geq L_{ij}^\ell, \forall i \in [I], \forall j \in [J], E_1^\ell \in [0, 1], \quad (12a)
\]

\[
E_2^\ell \geq (\hat{\lambda}_{ij}^{\ell})^{-1} \left( \hat{L}_{i,j-1}^\ell + \hat{\xi}_{ij}^\ell - \sum_{k \in [K]} n_{ik} \delta k c_k \right), \forall i \in [I], \forall j \in [J], E_2^\ell \in [0, 1]. \quad (12b)
\]

Now it remains to linearize the bilinear terms in (12a). According to (8), we can represent variables \( n \) using binary variables \( u \). Therefore, constraints (12a) are equivalent to

\[
\sum_{k \in [K]} \sum_{r \in [R_k]} 2^{r-1} \delta k c_k E_1^\ell u_{ikr} \geq L_{ij}^\ell, \forall i \in [I], \forall j \in [J], E_1^\ell \in [0, 1], \quad (12c)
\]

Introducing McCormick inequalities to linearize the bilinear terms, we have

\[
\sum_{k \in [K]} \sum_{r \in [R_k]} 2^{r-1} \delta k c_k w_{ikr}^\ell \geq L_{ij}^\ell, \forall i \in [I], \forall j \in [J], E_1^\ell \in [0, 1], \quad (12c)
\]

\[
w_{ikr}^\ell \geq 0, w_{ikr}^\ell \geq E_1^\ell + u_{ikr} - 1, w_{ikr}^\ell \leq E_1^\ell, w_{ikr}^\ell \leq u_{ikr}, \forall k \in [K], \forall i \in [I], \forall r \in [R]. \quad (12d)
\]

4.3. An Exact Mixed-Integer Linear Programming (MILP) Reformulation for DrFRAM (7)

Let us put all the linearized pieces together and we are ready to present an exact MILP reformulation for DrFRAM (7). In particular, DrFRAM (7) is equivalent to the following MILP:

\[
(DrFRAM) \quad v^* = \min_{n, x, u} \left\{ \frac{1}{N} \sum_{\ell \in [N]} Q(n, x, u, \xi^\ell) : (1b) - (1f), (8a), (8b) \right\}, \quad (13a)
\]

where for simplicity, we slightly abuse the notation by redefining \( Q(n, x, u, \xi^\ell) \) as

\[
Q(n, x, u, \xi^\ell) = \min_{L^\ell, L^\ell, w^\ell, \xi^\ell, E^\ell} \left\{ E_1^\ell + \omega E_2^\ell : (2c), (10), (11), (12b) - (12d) \right\}. \quad (13b)
\]

Note that we can encode the entire DrFRAM (13) into the off-the-shelf solvers such as Gurobi, CPLEX, MOSEK. However, our numerical study shows that albeit effective, model (13) has difficulty solving large-scale instances, remaining large optimality gaps within an hour. Thus, we will develop valid inequalities and strong formulations based on the knapsack polytope and the disjunctive programming (Balas 1979) in the next section.
Finally, we remark that one can adopt the scenario decomposition method proposed by Ahmed (2013) to solve the DrFRAM (13) in a decomposed way. The general idea is to completely decompose MILP (13) into \(N\) subproblems, i.e., for each \(\ell \in [N]\), we solve

\[
v^\ell := \min_{n,x,u} \left\{ Q(n,x,u,\hat{\xi}) : (1b) - (1f), (8a), (8b) \right\}.
\]

Subproblem (14) can be accelerated with the valid inequalities and stronger formulations developed in the next section. Note that the average of their objective functions provides a lower bound of DrFRAM and the first-stage decision obtained in each subproblem is also feasible to the original MILP (13). Hence, we can evaluate these decisions and choose the best one as an upper bound. Next, we cut off the obtained decisions from their corresponding subproblems using the no-good cut (Ahmed 2013), i.e., for a scenario \(\ell\), given a first-stage binary solution \((\bar{x}^\ell, \bar{u}^\ell)\) (we do not need to include \(\bar{n}^\ell\) since it can be represented by \(\bar{u}^\ell\)), we add the following no-good cut

\[
\sum_{i \in [I], k \in [K]} \left( 1 - \bar{x}_{ik}^\ell \right) x_{ik} + \sum_{r \in [R_k]} \left( 1 - \bar{u}_{ikr}^\ell \right) u_{ikr} \geq 1
\]

into the \(\ell\)th subproblem; and repeat the same procedure. We terminate the solution procedure when invoking a stopping criterion. The detailed implementation can be found in Algorithm 1.

**Algorithm 1: Scenario Decomposition for Solving DrFRAM**

1. Initialization: Set \(\tau = 0, \epsilon > 0, X^\ell = \{(n,x,u) : (1b) - (1f), (8a), (8b)\}\) for all \(\ell \in [N]\), \(LB = -\infty, UB = \infty\);
2. While \(UB - LB > \epsilon\) do
   3. For \(\ell \in [N]\) do
      4. \((\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell) \in \arg\min\{ Q(n,x,u,\hat{\xi}) : (n,x,u) \in X^\ell\};
      5. \(LB = \max\{LB, \frac{1}{N} \sum_{\ell \in [N]} Q(\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell, \hat{\xi})\};\)
      6. \(UB = \min\{UB, \min_{\ell \in [N]} \frac{1}{N} Q(\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell, \hat{\xi})\};\)
      7. \(X^\ell \rightarrow X^\ell \setminus \{(\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell)\}\) for each \(\ell \in [N]\);
      8. \(\tau \leftarrow \tau + 1\)

Note that to accelerate the algorithm, rather than adding the no-good cut to each subproblem, we can add an objective cut to force the objective function to increase by at least a positive step, that is, for each \(\ell \in [N]\), we add the following inequality

\[
E_1^\ell + \omega E_2^\ell \geq Q(\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell) + \hat{\epsilon}^\ell
\]

(15)
Algorithm 2: Accelerated Scenario Decomposition for Solving DrFRAM

1 Initialization: Set $\tau = 0, \epsilon > 0, \mathcal{X}^\tau = \{(n, x, u) : (1b) - (1f), (8a), (8b)\}$ for all $\ell \in [N], \{\hat{\xi}\}_{\ell \in [N]}$, $LB = -\infty, UB = \infty$;

2 while $UB - LB > \epsilon$ do

3 for $\ell \in [N]$ do

4 $(\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell) \in \arg\min \{Q(n, x, u, \xi) : (15), (n, x, u) \in X^\ell\}$;

5 $LB = \max\{LB, 1/N \sum_{\ell \in [N]} Q(\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell, \hat{\xi}^\ell)\}$;

6 $UB = \min\{UB, \min_{\ell \in [N]} \frac{1}{N} Q(\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell, \hat{\xi}^\ell)\}$;

7 $\tau \leftarrow \tau + 1$

Algorithm 2: Accelerated Scenario Decomposition for Solving DrFRAM

1 Initialization: Set $\tau = 0, \epsilon > 0, \mathcal{X}^\tau = \{(n, x, u) : (1b) - (1f), (8a), (8b)\}$ for all $\ell \in [N], \{\hat{\xi}\}_{\ell \in [N]}$, $LB = -\infty, UB = \infty$;

2 while $UB - LB > \epsilon$ do

3 for $\ell \in [N]$ do

4 $(\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell) \in \arg\min \{Q(n, x, u, \xi) : (15), (n, x, u) \in X^\ell\}$;

5 $LB = \max\{LB, 1/N \sum_{\ell \in [N]} Q(\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell, \hat{\xi}^\ell)\}$;

6 $UB = \min\{UB, \min_{\ell \in [N]} \frac{1}{N} Q(\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell, \hat{\xi}^\ell)\}$;

7 $\tau \leftarrow \tau + 1$

into the $\ell$th subproblem, where $(\bar{n}^\ell, \bar{x}^\ell, \bar{u}^\ell)$ denotes an optimal solution of $\ell$th subproblem and $\hat{\xi}^\ell > 0$. Instead of solving the problem to optimal, the accelerated scenario decomposition can obtain a suboptimal solution. The detailed implementation can be found in Algorithm 2.

Finally, we conclude this section by showing that if the data from distributionally robust counterparts (i.e., $\{\hat{\xi}\}_{\ell \in [N]}$) have a small variability, then the naive scenario decomposition lower bound (i.e., $1/N \sum_{\ell \in [N]} v^\ell$) will not be faraway from the true optimal $v^\star$.

Proposition 4 Suppose that the weight $\omega = 0$ (i.e., the passenger abandon rate for each bus stop is negligible), and there exists a matrix $\mu = (\mu^a, \mu^\lambda) \in \Xi$ and a positive integer $\hat{\alpha} \in \mathbb{Z}^+$ such that

$$\hat{a}^\ell_{ij} = \mu^{a}_{ij}, \mu^{\lambda}_{ij} \leq |\hat{\lambda}^\ell_{ij}| \leq \hat{\alpha} \mu^{\lambda}_{ij}, \forall i \in [I], \forall j \in [J_i].$$

Then the following approximation ratio holds for the naive scenario decomposition lower bound (without any good cut or objective cut)

$$\frac{1}{\hat{\alpha}} v^\star \leq \frac{1}{N} \sum_{\ell \in [N]} v^\ell \leq v^\star.$$

Proof: We split the proof into three steps.

Step 1. The second inequality holds true since we drop the nonanticipativity constraints when computing the naive scenario decomposition lower bound.

Step 2. Next, to prove the first inequality, let us use $v^D$ to denote the optimal value of the following nominal problem as

$$v^D(\alpha) = \min_{n, x, u} \{Q(n, x, u, \alpha \mu) : (1b) - (1f), (8a), (8b)\}$$

for some positive integer $\alpha \in \mathbb{Z}^+$. In model (16), the function $Q(n, x, u, \alpha \mu)$ is defined in (13b) by letting $\hat{\xi}^\ell = \alpha \mu^\ell$. 

$$\frac{1}{\hat{\alpha}} v^\star \leq \frac{1}{N} \sum_{\ell \in [N]} v^\ell \leq v^\star.$$
According to the monotonicity results in Corollary 1, we have

$$Q(n, x, u, \hat{\alpha}^x) \leq Q(n, x, u, \hat{\alpha} \mu)$$

for all $\ell \in [N]$ and any feasible $(n, x, u)$. Thus, aggregating the above inequalities for all $\ell \in [N]$, we have

$$\frac{1}{N} \sum_{\ell \in [N]} Q(n, x, u, \hat{\alpha}^x) \leq Q(n, x, u, \hat{\alpha} \mu)$$

for any feasible $(n, x, u)$. This implies that $v^* \leq v^D(\hat{\alpha})$.

Similarly, we also have $v^\ell \geq v^D(1)$ for each $\ell \in [N]$. That is, $\frac{1}{N} \sum_{\ell \in [N]} v^\ell \geq v^D(1)$. Now it remains to show that $v^D(\hat{\alpha}) \leq \hat{\alpha} v^D(1)$.

**Step 3.** We observe that

**Claim 1** For any real number $q \in \mathbb{R}_+$ and positive integer number $\alpha \in \mathbb{Z}_{++}$, the following inequality must hold $\alpha[q] \geq [\alpha q]$.

**Proof:** This is simply because $\alpha[q] \geq \alpha q$ and the former is an integer. $\diamond$

Suppose that $(n^*, x^*, u^*)$ is an optimal first-stage solution and $(L^1, \bar{L}^1, w^1, y^1, E^1)$ is an optimal second-stage decision to model (16) with $\alpha = 1$.

Now let us define

$$L_{ij}^\hat{\alpha} = \min \left\{ \sum_{k \in [K]} n_{ik}^* |\delta_k c_k|, \hat{\alpha} L^1_{ij} \right\}, \forall i \in [I], \forall j \in [J_i],$$

$$\bar{L}_{ij}^\hat{\alpha} - 1 = [(1 - \mu^\alpha_{ij}) L_{ij}^\hat{\alpha}], \forall i \in [I], \forall j \in [J_i],$$

$$y_{ij}^\hat{\alpha} = \mathbb{I} \left( L_{ij}^\hat{\alpha} < \sum_{k \in [K]} n_{ik}^* |\delta_k c_k| \right), \forall i \in [I], \forall j \in [J_i],$$

$$E^\hat{\alpha}_1 \equiv \max_{i \in [I], j \in [J_i]} \frac{L_{ij}^\hat{\alpha}}{\sum_{k \in [K]} n_{ik}^* |\delta_k c_k|},$$

$$E^\hat{\alpha}_2 \equiv \max_{i \in [I], j \in [J_i]} \left( \hat{\alpha} \mu_{ij}^\lambda \right)^{-1} \left( \bar{L}_{ij}^\hat{\alpha} - \hat{\alpha} \mu_{ij}^\lambda - \sum_{k \in [K]} n_{ik}^* |\delta_k c_k| \right)_+, $$

$$w_{ikr}^\hat{\alpha} = E^\hat{\alpha}_1 w^*_{ikr}, \forall i \in [I], \forall k \in [K], \forall r \in [R_k].$$

According to the definition of model (16) with $\alpha = \hat{\alpha}$, we see that $(n^*, x^*, u^*)$ is a feasible first-stage solution and $(L^\hat{\alpha}, \bar{L}^\hat{\alpha}, w^\hat{\alpha}, y^\hat{\alpha}, E^\hat{\alpha})$ satisfies constraints (2c), (10), (12b)–(12d). It remains to show that constraints (11) also hold.

Before proceeding, we observe the following fact:

**Claim 2** For any $i \in [I], j \in [J_i]$, we must have $\bar{L}_{ij}^\hat{\alpha} \leq \hat{\alpha} L_{ij}^1$. 
Proof: Note that
\[ \hat{L}_{ij} = \max \left\{ \left( 1 - \mu_i^a \right) \hat{L}_{ij}^1 \right\} = \left( 1 - \mu_i^a \right) \hat{L}_{ij}^1, \]
where the first inequality is due to the definition of \( \hat{L}_{ij}^1 \) and the second one is due to Claim 1. \( \Diamond \)

There are two cases:

- When \( \hat{L}_{ij} < \sum_{k \in [K]} n^*_k \delta_{ck} \), it is sufficient to show that \( \hat{L}_{ij} \geq \hat{L}_{ij}^1 + \hat{\alpha} \mu_i^\lambda \). This must be true since we have
\[ \hat{L}_{ij} = \hat{L}_{ij}^1, \hat{L}_{ij}^1 \leq \hat{L}_{ij}^1, L_{ij} \geq L_{ij}^1 + \mu_i^\lambda \]
where the first inequality is due to Claim 2 and the second one is due to the fact that \((L^1, \hat{L}^1, w^1, y^1, E^1)\) is an optimal (and of course feasible) second-stage decision to model (16) with \( \alpha = 1 \).

- When \( \hat{L}_{ij} = \sum_{k \in [K]} n^*_k \delta_{ck} \), we have \( y_i^a = 0 \). It is sufficient to show that \( \hat{L}_{ij} \geq \hat{L}_{ij}^1 \). This is true since
\[ \hat{L}_{ij} = \left( 1 - \mu_i^a \right) \hat{L}_{ij}^1 \leq \sum_{k \in [K]} n^*_k \delta_{ck} = \hat{L}_{ij} \]
where the inequality is due to the fact that \( \hat{L}_{ij}^1 \leq \sum_{k \in [K]} n^*_k \delta_{ck} \). \( \Diamond \)

Finally, we observe that the objective function is
\[ E_1^\hat{\alpha} + \omega E_2^\hat{\alpha} = \max_{i \in [I], j \in [J]} \frac{L_{ij}^1}{\sum_{k \in [K]} n^*_k \delta_{ck}} + \omega \max_{i \in [I], j \in [J]} (\hat{\alpha} \mu_i^\lambda)^{-1} \left( \hat{L}_{ij}^1 + \hat{\alpha} \mu_i^\lambda - \sum_{k \in [K]} n^*_k \delta_{ck} \right) \]

\[ = \max_{i \in [I], j \in [J]} \min \left\{ \sum_{k \in [K]} n^*_k \delta_{ck} \hat{L}_{ij}^1 \right\} \]
\[ \leq \max_{i \in [I], j \in [J]} \hat{\alpha} \min \left\{ \sum_{k \in [K]} n^*_k \delta_{ck} L_{ij}^1 \right\} = \hat{\alpha} E_1^1 := v^D(1) \]

where the inequality is due to the fact that \( \min \{ a, \hat{\alpha} b \} \leq \hat{\alpha} \min \{ a, b \} \) for any non-negative numbers \( a, b \) and positive integer \( \hat{\alpha} \).

Since \((n^*, x^*, u^*)\) is a feasible first-stage solution and \((L^\hat{\alpha}, \hat{L}^\hat{\alpha}, w^\hat{\alpha}, y^\hat{\alpha}, E^\hat{\alpha})\) is a feasible second-stage solution to model (16) with the objective value at most \( \hat{\alpha} v^D(1) \), this proves that \( v^D(\hat{\alpha}) \leq \hat{\alpha} v^D(1) \). This completes the proof. \( \square \)

Proposition 4 shows that, not very surprisingly, if the alighting rate for each bus stop is deterministic, and the passenger demand does not vary too much from scenario to scenario, then the naive scenario decomposition lower bound tends to be very close to the true optimality. This is consistent with what we have found in the numerical study section. However, in general, Algorithm 1 suffers from slow convergence when being employed to find a global optimal solution. Therefore, in the next section, we will derive stronger formulations and valid inequalities to further strengthen MILP (13).
5. Stronger Formulations and Valid Inequalities

In this section, we strengthen the MILP model (13) by developing different families of valid inequalities. Our results and strong formulations are based on studying the mixed-integer substructures of model (13). Specifically, we consider convexifying the following two substructures to derive stronger formulations. First, for each \( i \in [I] \), we consider the substructure of variables \((x_i, u_{i,:})\) by projecting out variables \(n_i\). That is, for each \( i \in [I] \), let us define

\[
X_i^1 = \{(x_i, u_{i,:}) : \exists n_i, (1b) - (1c), (8a), (8b)\}.
\]

(17a)

The second substructure is defined for each sample \( \ell \in [N] \) and each \( i \in [I] \) and \( j \in [J_i] \) as

\[
X_{ij\ell}^2 = \{(x_i, u_{i,:}, L_{ij\ell}^I) \in X_i^1 \times \mathbb{R}^+ : (12c), (12d)\}.
\]

(17b)

5.1. Convexification of Set \( X_i^1 \)

We will first convexify \( X_i^1 \). According to (17a), set \( X_i^1 \) is equivalent to

\[
X_i^1 = \left( \bigvee_{k \in [K]} (X_{ik}^1 \land \{x_{ik} = 1\}) \right) \land \left\{ x_i \in \{0,1\}^K : \sum_{k \in [K]} x_{ik} = 1 \right\}
\]

where

\[
X_{ik}^1 := \left\{ (x_i, u_{i,:}) : \sum_{r \in [R_k]} 2^{r-1} u_{ikr} \leq \eta_k x_{ik}, u_{ikr} \in \{0,1\}, \forall r \in [R_k] \right\}.
\]

Suppose that \( \eta_k = 2^{|i_k|} + \ldots + 2^{|i_k|} + 2^{|R_k| - 1} \) and set \( I := \{i_1, \ldots, i_{sk}, R_k\} \). Since \( 2^{r-1} \) are superincreasing, according to proposition 3.4 in Gupte et al. (2013), the convex hull of \( X_{ik}^1 \land \{x_{ik} = 1\} \) can be described as

\[
\text{conv} \left( X_{ik}^1 \land \{x_{ik} = 1\} \right) = \left\{ (x_i, u_{i,:}) : x_i \in [0,1]^K, x_{ik} = 1, u_{ikr} \leq |I_r|, \forall r \in [R_k] \setminus I_r \right\}
\]

where \( I_r := \{s \in I : s > r\} \).

Our convex hull description of set \( X_i^1 \) replies on the well-known disjunctive programming (Balas 1979).

**Lemma 1** Suppose \( \bar{s} \) polyhedra \( \bar{X}^i = \{\bar{y} \in \mathbb{R}^n : A^i \bar{y} \leq b^i\} \) with \( A^i \in \mathbb{R}^{m_i \times \bar{n}} \) and \( b \in \mathbb{R}^{m_i} \) for each \( i \in [\bar{s}] \) share the same recession cone. Then the following result holds

\[
\text{conv} \left\{ \left( \bigvee_{s \in [\bar{s}]} (\bar{X}^s \land \{\lambda_i = 1\}) \right) \land \left\{ \lambda \in \{0,1\}^{\bar{s}} : \sum_{i \in [\bar{s}]} \lambda_i = 1 \right\} \right\}
\]

\[
= \left\{ (\bar{y}, \lambda) \in \mathbb{R}^n \times [0,1]^{\bar{s}} : A^i \bar{y}^i \leq b^i \lambda_i, \forall i \in [\bar{s}], \sum_{i \in [\bar{s}]} \bar{y}^i = \bar{y} \right\}.
\]
According to Lemma 1 and the fact that the convex hull of union of sets is equal to the convex hull of union of convex hulls of sets, we arrive at the complete description of convex hull of set $X_i^1$. This result is summarized below.

**Proposition 5** For each $i \in [I]$, the convex hull of the set $X_i^1$ is equal to

$$
\text{conv}(X_i^1) = \left\{ \left( x_i, u_i : u_{ikr} + \sum_{r \in R_k} u_{ikr} \leq |I_r| x_{ik}, \forall r \in [R_k] \setminus I, \forall k \in [K] \right) \right\}.
$$

(18)

5.2. Convexification of Set $X_{ij}^2$

Similarly, to convexify set $X_{ij}^2$, we first rewrite it as a disjunction

$$
X_{ij}^2 = \left( \bigvee_{k \in [K]} (X_{ijk}^2 \land \{ x_{ik} = 1 \}) \right) \land \left\{ x_i \in \{0, 1\}^K : \sum_{k \in [K]} x_{ik} = 1 \right\},
$$

(19)

where

$$
X_{ijk}^2 \land \{ x_{ik} = 1 \} := \left\{ (x_i, u_i, l_i, w_i, e_i^1) : (x_i, u_i \in X_i^1, \sum_{r \in [R_k]} 2^{-1} [\delta_k c_k] w_{ikr} \geq L_{ij}^f, w_{ikr} = u_{ikr}, l_i, e_i^1 \in \mathbb{R}_+, E_i^1 \in [0, 1], x_{ik} = 1 \} \right\}.
$$

We first observe that the convex hull of set $X_{ijk}^2 \land \{ x_{ik} = 1 \}$ is equal to the intersection of the constraint $\sum_{r \in [R_k]} 2^{-1} [\delta_k c_k] w_{ikr} \geq L_{ij}^f$ and the convex hull of the feasible region subject to the remaining constraints.

**Lemma 2** The following characterization holds for $\text{conv}(X_{ijk}^2 \land \{ x_{ik} = 1 \})$, i.e.,

$$
\text{conv}(X_{ijk}^2 \land \{ x_{ik} = 1 \}) = \text{conv}(\bar{X}_{ijk}^2) \cap \left\{ (w_i^f, l_i^f) : \sum_{r \in [R_k]} 2^{-1} [\delta_k c_k] w_{ikr} \geq L_{ij}^f \right\},
$$

where

$$
\bar{X}_{ijk}^2 = \left\{ (x_i, u_i, l_i^f, w_i^f, e_i^1) : (x_i, u_i, l_i^f, w_i^f, e_i^1) \in X_i^1, w_{ikr} = u_{ikr}, l_i^f, e_i^1 \in \mathbb{R}_+, E_i^1 \in [0, 1], x_{ik} = 1 \} \right\}.
$$

**Proof:** Since the convex hull of the intersection of two sets is contained in the intersection of the convex hulls of two sets, we must have

$$
\text{conv}(X_{ijk}^2 \land \{ x_{ik} = 1 \}) \subseteq \text{conv}(\bar{X}_{ijk}^2) \cap \left\{ (w_i^f, l_i^f) : \sum_{r \in [R_k]} 2^{-1} [\delta_k c_k] w_{ikr} \geq L_{ij}^f \right\}.
$$

It remains to show that

$$
\text{conv}(X_{ijk}^2 \land \{ x_{ik} = 1 \}) \supseteq \text{conv}(\bar{X}_{ijk}^2) \cap \left\{ (w_i^f, l_i^f) : \sum_{r \in [R_k]} 2^{-1} [\delta_k c_k] w_{ikr} \geq L_{ij}^f \right\}.
$$
Indeed, for any \((x_i, u_i, L_{ij}, w_{i\ell}, E_1^\ell) \in \text{conv}(\bar{X}^2_{ijtk}) \cap \{(w_{i\ell}^r, L_{ij}^r): \sum_{r \in [R_k]} 2^{r-1} |\delta_k c_k| w_{ikr}^r \geq L_{ij}^r\}\), there exists a finite collection \(\{(x_i^r, u_i^r, L_{ij}^r, w_{i\ell}^r, E_1^\ell)\}_{\tau \in [q]}\) and \(\{\alpha_\tau\}_{\tau \in [q]} \subset [0, 1]\) such that
\[
\sum_{\tau \in [q]} \alpha_\tau (x_i^r, u_i^r, L_{ij}^r, w_{i\ell}^r, E_1^\ell) = (x_i, u_i, L_{ij}, w_{i\ell}, E_1^\ell), \sum_{\tau \in [q]} \alpha_\tau = 1
\]
and \(\beta := \sum_{\tau \in [q]} \alpha_\tau \sum_{r \in [R_k]} 2^{r-1} |\delta_k c_k| w_{ikr}^r \geq L_{ij}^r\).

Now let us define
\[
\tilde{L}_{ij}^\tau = \frac{L_{ij}^\ell}{\beta} \sum_{\tau \in [R_k]} 2^{r-1} |\delta_k c_k| w_{ikr}^r
\]
for each \(\tau \in [q]\). Clearly, we have
\[
\sum_{\tau \in [q]} \alpha_\tau (x_i^r, u_i^r, \tilde{L}_{ij}^\tau, w_{i\ell}^r, E_1^\ell) = (x_i, u_i, L_{ij}, w_{i\ell}, E_1^\ell),
\]
\[(x_i^r, u_i^r, \tilde{L}_{ij}^\tau, w_{i\ell}^r, E_1^\ell) \in X^2_{ijtk} \cap \{x_{ik} = 1\}, \forall \tau \in [q].\]

Thus, \((x_i, u_i, L_{ij}, w_{i\ell}, E_1^\ell) \in \text{conv}(X^2_{ijtk} \cap \{x_{ik} = 1\})\).

It turns out that according to Lemma 2 and proposition 3.1. in Gupte et al. (2013), sets \(\text{conv}(\bar{X}^2_{ijtk})\) and \(\text{conv}(X^2_{ijtk} \cap \{x_{ik} = 1\})\) can be completely described using the result in Proposition 5 and the fact that variables \(x_i\) are independent of others and constraint system with respect to \(x_i\) is integral. These results are summarized below.

**Proposition 6** For each \(i \in [I], j \in [J], k \in [K], \ell \in [N]\), the sets \(\text{conv}(\bar{X}^2_{ijtk})\) and \(\text{conv}(X^2_{ijtk} \cap \{x_{ik} = 1\})\) admit the following complete descriptions:

\[
\text{conv}(\bar{X}^2_{ijtk}) = \left\{(x_i, u_i, L_{ij}, w_{i\ell}, E_1^\ell): u_{ikr} + \sum_{\tau \in [R_k]} u_{ikr} - \sum_{\tau \in [R_k]} w_{ikr} \leq |I_r| E_1^\ell, \forall r \in [R_k] \setminus I_r, \right. \]
\[
\left. \sum_{\tau \in [R_k]} w_{ikr} \leq |I_r| (1 - E_1^\ell), \forall r \in [R_k] \setminus I_r, 0 \leq w_{ikr} \leq E_1^\ell, \forall r \in [R_k], 0 \leq u_{ikr} - w_{ikr} \leq (1 - E_1^\ell), \forall r \in [R_k] \right\}.
\]

\[
\text{conv}(X^2_{ijtk} \cap \{x_{ik} = 1\}) = \left\{(x_i, u_i, L_{ij}, w_{i\ell}, E_1^\ell): u_{ikr} + \sum_{\tau \in [R_k]} u_{ikr} - \sum_{\tau \in [R_k]} w_{ikr} \leq \right. \]
\[
\left. |I_r|(1 - E_1^\ell), \forall r \in [R_k] \setminus I_r, 0 \leq w_{ikr} \leq E_1^\ell, \forall r \in [R_k], 0 \leq u_{ikr} - w_{ikr} \leq (1 - E_1^\ell), \forall r \in [R_k], \sum_{r \in [R_k]} 2^{r-1} |\delta_k c_k| w_{ikr}^r \geq L_{ij}^r \right\}.
\]
Following Lemma 1 and the fact that the convex hull of union of sets is equal to the convex hull of union of convex hulls of sets, we arrive at the following complete description of convex hull of set $X^2_{ij\ell}$.

**Proposition 7** For each $i \in [I], j \in [J]$, and $\ell \in [N]$, the set $\text{conv}(X^2_{ij\ell})$ admits the following complete description:

$$\text{conv}(X^2_{ij\ell}) = \left\{ \left( x_i, u_{i\cdot j\cdot}, L^\ell_{i\cdot j\cdot}, w^\ell_{i\cdot j\cdot}, E^\ell_{i\cdot j\cdot} \right) : \begin{align*}
    x_i &\in [0, 1]^K, \quad \sum_{k \in [K]} x_{ik} = 1, \quad \sum_{k \in [K]} \hat{E}^\ell_{1ik} = E^\ell_{1i}, \quad \hat{E}^\ell_{1ik} \leq x_{ik}, \\
    w_{ikr} + \sum_{\tau \in \mathcal{I}_r} u_{ikr} &\leq |\mathcal{I}_r| \hat{E}^\ell_{1ik}, \forall r \in [R_k] \setminus \mathcal{I}, \forall k \in [K], \\
    u_{ikr} + \sum_{\tau \in \mathcal{I}_r} u_{ikr} - w_{ikr} &\leq 1, \quad \hat{E}^\ell_{1ik}, \forall k \in [K], \\
    0 &\leq w_{ikr} \leq \hat{E}^\ell_{1ik}, \forall r \in [R_k], \forall k \in [K], \\
    0 &\leq u_{ikr} - w_{ikr} \leq (1 - \hat{E}^\ell_{1ik}), \forall r \in [R_k], \forall k \in [K], \\
    2^{r-1} |\delta_k c_k| w^\ell_{ikr} &\geq \hat{L}^\ell_{ijk}, \forall k \in [K], \\
    \sum_{r \in [R_k]} \hat{L}^\ell_{ijk} &\leq L^\ell_{ij}, \quad 0 \leq \hat{E}^\ell_{1ik} \leq E^\ell_{1i} \leq 1, \quad \hat{L}^\ell_{ijk} \geq 0, \forall k \in [K] \end{align*} \right\}$$

(22)

### 5.3. Valid Inequalities

In this subsection, we develop valid inequalities to further strengthen the MILP model (13).

**One-bus-per-route Inequalities:** First, at-least-one-bus-per-route constraints (1e) together with the binarization constraints (8a) imply that at least one of the corresponding binary variables must be nonzero, i.e., the following valid inequalities must hold:

$$\sum_{r \in [R_k]} u_{ikr} \geq x_{ik}, \forall i \in [I], \forall k \in [K].$$

(23)

**Under-capacity Inequalities:** Let us define $M = \min_{k \in [K]} |\delta_k c_k|$ as the minimum route capacity at any route. Suppose each route has no capacity restriction, i.e., any passenger is supposed to get on the bus, then we can compute the ideal number of passengers remaining on the bus using the following formula:

$$t^\ell_{ij} = \left(1 - \hat{d}^\ell_{ij} \right) t^\ell_{i,j-1} + \bar{x}^\ell_{ij}, \forall i \in [I], \forall j \in [J], \forall \ell \in [N],$$

(24)

where we let $t^\ell_{i0} = 0$ for all $i \in [I]$ and $\ell \in [N]$. Now let us define $j^\ell_{i*} = \min\{J, \min_{j \in [J]} \{t^\ell_{i0} > M\}\}$ to be the first bus stop such that the number of passengers planning to be on board is greater than the minimum capacity for each $i \in [I]$ and $\ell \in [N]$. Then, clearly, for any bus stop $j \in [j^\ell_{i*} - 1]$, it is under-capacity, i.e., we must have

$$y^\ell_{ij} = 1, \forall i \in [I], \forall j \in [j^\ell_{i*} - 1], \forall \ell \in [N].$$

(25)
Utilization Rate when Being Fully Occupied: If one bus stop is fully occupied, then the bus utilization rate must be one. Thus, we have the following valid inequalities:

\[
1 - y^f_{ij} \leq E_1^f, \forall i \in [I], \forall j \in [J], \forall \ell \in [N].
\] (26)

Tightening Big-M Coefficients: Note that in (11), we must have \( L^f_{ij} \geq \hat{L}^f_{ij-1} \). Hence, in the first part of (11), we can reduce \( M \) by \( M - \hat{\lambda}^f_{ij} \) and in the second part of (11), \( \hat{\lambda}^f_{ij} - M(1 - y^f_{ij}) \) can be tightened by \( \hat{\lambda}^f_{ij} \). Therefore, we arrive at the following stronger inequalities

\[
L^f_{ij} \geq \sum_{k \in [K]} n_{ik} [\delta_k c_k] - (M - \hat{\lambda}^f_{ij}) y^f_{ij}, L^f_{ij} \geq \hat{L}^f_{ij-1} + \hat{\lambda}^f_{ij} y^f_{ij}, \forall i \in [I], \forall j \in [J], \forall \ell \in [N].
\] (27)

Passengers Being Unserved when a Bus is Fully Occupied: We note that the passenger abandon rate becomes positive only when the bus is fully occupied at some moments. Thus, we must have

\[
E_2^f \leq E_1^f, \forall \ell \in [N].
\] (28)

Lower Bounding the Bus Utilization Rate: The disjunctive programming result in Section 5.2 implies that \( \hat{E}_{1ik}^f \geq \hat{L}_{ijk}^f/(n_{ik} [\delta_k c_k]) \). Since \( n_{ik} \leq \hat{M}_k := \min\{\eta_k, \sum_{k \in [K]} \eta_k - I + 1\} \), we must have

\[
\hat{E}_{1ik}^f \geq \frac{\hat{L}_{ijk}}{\hat{M}_k [\delta_k c_k]}, \forall i \in [I], \forall j \in [J], \forall \ell \in [N].
\] (29)

6. Approximation Algorithms Based on No-One-Left Policy

We observe that the intricacy of the MILP model (13) comes from the linearization of the first part of the objective function using binary variables \( u \) and the linearization of constraints (2b) using auxiliary variables \( y \). To avoid both, we propose an approximation scheme using the notion of the so-called No-one-left policy. Recall that in the previous section, we define \( \iota \) to be the ideal number of passengers at each bus stop when there is no capacity restriction. Our No-one-left policy follows the same concept.

Thus, we propose to solve the following simplified model as

\[
\text{(No-one-left)} \quad \min_{n, x} \left\{ \frac{1}{N} \sum_{\ell \in [N]} \mathcal{Q}(n, x, \hat{\xi}^f) : (1b) - (1f) \right\},
\] (30a)

where we define \( \mathcal{Q}(n, x, \hat{\xi}^f) = \max_{i \in [I], j \in [J]} \frac{\ell^f_{ij}}{\left( \sum_{k \in [K]} n_{ik} [\delta_k c_k] \right)} \). Note that \( \mathcal{Q}(n, x, \hat{\xi}^f) \) admits a second-order conic representation as

\[
\mathcal{Q}(n, x, \hat{\xi}^f) = \min \left\{ \mathcal{E}^f : \left\| \left( 2 \max_{j \in [J]} \sum_{k \in [K]} n_{ik} [\delta_k c_k] - \mathcal{E}^f \right) \right\|_2 \leq \sum_{k \in [K]} n_{ik} [\delta_k c_k] + \mathcal{E}^f, \forall i \in [I] \right\}.
\] (30b)
After solving No-one-left model (30), we can evaluate its objective value $v^N$ by plugging in its optimal first stage decision $(\bar{n}, \bar{x})$ into DrFRAM (7), i.e.,

$$v^N = \frac{1}{N} \sum_{\ell \in [N]} Q(\bar{n}, \bar{x}, \hat{\xi}^\ell).$$

The following result shows the approximation bound of the No-one-left policy.

**Proposition 8** Suppose that $(\bar{n}, \bar{x}, \bar{E})$ and $(n^*, x^*, E^*)$ are optimal solutions to No-one-left model (30) and DrFRAM (13), respectively. Then we have

$$v^* \leq v^N \leq v^* + \frac{\omega}{N} \| (\bar{E} - e)_+ \|_0 + \frac{M - 1}{N} \| E^*_2 \|_0,$$

where $M := \min_{k \in [K]} \lfloor \delta_k c_k \rfloor$ and $\bar{M} := \max_{i \in [I], j \in [J], \ell \in [N]} t^\ell_{ij}$.

**Proof:** First, $v^* \leq v^N$ is due to the feasibility of the first stage decision $(\bar{n}, \bar{x})$.

Next, we note that if $E_\ell > 1$, then we have $1 \leq Q(\bar{n}, \bar{x}, \hat{\xi}^\ell) \leq 1 + \omega$. Therefore, we must have

$$v^N - \frac{\omega}{N} \| (\bar{E} - e)_+ \|_0 \leq \frac{1}{N} \sum_{\ell \in [N]} E^\ell.$$

(31)

According to the optimality of $(\bar{n}, \bar{x}, \bar{E})$ to No-one-left model (30), we have

$$\frac{1}{N} \sum_{\ell \in [N]} E^\ell \leq \frac{1}{N} \sum_{\ell \in [N]} \max_{i \in [I], j \in [J]} t^\ell_{ij} \leq v^* + \frac{\bar{M} - 1}{M} \| E^*_2 \|_0.$$  

(32)

Combining these two inequalities (31) and (32) together, we arrive at the conclusion. 

Proposition 8 shows that $(\bar{n}, \bar{x})$ is optimal to DrFRAM (7) if there is no passenger being abandoned (i.e., $E_2^* = 0$) in an optimal solution to the DrFRAM model and all the utilization rates are zero in the No-one-left model.

### 6.1. Enhancing No-one-left Policy

In this subsection, we propose to enhance the No-one-left policy by incorporating the abandon rate for each scenario which reaches a full utilization rate of some routes and resolving the problems again. Namely, suppose $(\bar{n}, \bar{x})$ denotes an optimal first-stage decision of No-one-left model (30). Let us denote two subsets of scenarios, $\mathcal{N}_- = \{ \ell \in [N] : Q(\bar{n}, \bar{x}, \hat{\xi}^\ell) \leq 1 \}$ and $\mathcal{N}_+ = \{ \ell \in [N] : Q(\bar{n}, \bar{x}, \hat{\xi}^\ell) > 1 \}$. Then the enhanced model is defined as follows:

$$\text{(EnModel) } \min_{n, x} \left\{ \frac{1}{N} \left[ \sum_{\ell \in \mathcal{N}_-} Q(n, x, \hat{\xi}^\ell) + \sum_{\ell \in \mathcal{N}_+} Q^E(n, x, \hat{\xi}^\ell) \right] : (1b) - (1f) \right\},$$

(33a)
where $Q(n, x, \hat{\xi}^\ell)$ is defined in (30) and
\[
Q^E(n, x, \hat{\xi}^\ell) = \min_{L^\ell, \bar{L}^\ell, y^\ell, E^2_2} \{1 + E^2_2 : (10), (11), (12b), E^2_2 \in [0, 1]\}.
\]

Once solving the EnModel (33), we can repeat the procedure until invoking the stopping criteria. This procedure is summarized in Algorithm 3.

**Algorithm 3: Enhancing No-one-left Policy**

1. Initialization: Solve the No-one-left model (30) with an optimal first-stage solution $(\bar{n}, \bar{x})$; set $t = 0$, $N_\sim = [N]$ and initialize $t_{\text{max}}$;

2. while Set $N_\sim$ is changing and $t < t_{\text{max}}$ do

3. Define $N_\sim = \{\ell \in [N] : Q(\bar{n}, \bar{x}, \hat{\xi}^\ell) \leq 1\}$ and $N_+ = \{\ell \in [N] : Q(\bar{n}, \bar{x}, \hat{\xi}^\ell) > 1\}$;

4. Solve the EnModel (33);

5. $t \leftarrow t + 1$.

7. Numerical Study

In this section, we present a set of numerical results to compare the strengths of different model formulations and test the effectiveness of distinct methods using both small and large random instances as well as real-world data provided by Blacksburg Transit. For the random instances, passenger arrival rates were generated from uniform distributions with the minimum value ranging from 2 to 20 and maximum value ranging from 22 to 40, and the proportion of passengers alighting from the bus was randomly generated from triangular distributions with lower limit ranging from 0 to 0.2, upper limit ranging from 0.6 to 1, and mode ranging from 0.45 to 0.55. A time limit of 3600 seconds was set for solving each instance. For Blacksburg Transit instance, operation data including number of buses, capacities, routes and stops were provided by Blacksburg Transit. Passenger arrival and alighting data were collected in September 2020. All the instances were coded in Python 3.7.0 with calls to Gurobi 9.0.3 on a personal computer with a 1.9 GHz Intel Core i7 processor and 16G memory.

7.1. Results of Small Instances

In this section, we tested five small instances to compare the performances of MILP formulations as well as their continuous relaxations, where the number of bus routes $I \in \{5, 6\}$ and the number of scenarios $N = 5$. There are $K = 3$ types of buses with nominal capacities $c_1 = 60, c_2 = 80, c_3 = 120$ and each route has 10 to 40 stops. We generated 50 cases based on these instances using different
parameter combinations (i.e., different values of $I, \eta, \delta, \omega, \theta$). Particularly, cases 1-25, 26-35, 36-40, 41-45, 46-50 correspond to instances 1, 2, 3, 4, 5, respectively. For all the cases, we fixed the normalized parameter $\gamma = 100$.

The numerical results of exactly solving different MILP formulations are reported in Table 1, where $a - b$ in the “Case” column means instance #a and case #b, “MILP.B” represents MILP (13), “OPT” represents the optimal objective value obtained, “MILP.VI” represents MILP (13) with (23)-(28), “MILP.CONV” represents MILP (13) with (18), (22), (27), and (29), “MILP.VI.CONV” represents MILP (13) with (18), (22), (23)-(29). Without doubts, we see that all the formulations are able to find an optimal solution. MILP.VI improves MILP.B’s solution time on average but introduces more nodes to explore. This is probably because the valid inequalities introduced may force the solver to explore different branches before reaching an optimal solution. MILP.CONV requires fewer nodes to explore, but it usually takes more time to solve. This is possibly because of additional variables introduced to describe the convex hulls. The running time of MILP.VI.CONV is in between of MILP.VI and MILP.CONV, but it has the least number of nodes to explore. Hence, MILP.VI is the best among these methods.

We see that if the Wasserstein radius $\theta$ increases, the objective value increases, and the solution time does not change too much, since any increase in $\theta$ may result in larger passenger arrival rates and smaller passenger alighting rates. As the pandemic factor $\delta$ or the total number of buses increases and other parameters stay the same, both the objective value and the solution time decrease, since larger $\delta$ or larger total number of buses implies larger capacities for some routes and thus more passengers to get onto the bus. The weight $\omega$ does not affect the objective value and the solution time too much when the objective value is less than 1. When the objective value exceeds 1, a larger $\omega$ results in a larger objective value while the solution time stays the same. Since the problem size grows as the number of routes $I$ increases, the solution time also increases.

Results of continuous relaxations of different MILP formulations are reported in Table 2, where we use “C.MILP.B,” “C.MILP.VI,” “C.MILP.CONV,” “C.MILP.VI.CONV” to denote the continuous relaxations of MILP.B, MILP.VI, MILP.CONV, MILP.VI.CONV, respectively. We also define “Gap” as the percentage of the difference between the objective value and OPT divided by OPT. It is evident that both C.MILP.VI and C.MILP.CONV are better than C.MILP. This demonstrates the effectiveness of the proposed valid inequalities and the proposed convexification results. Clearly, the integration of valid inequalities and convexification results is the best, i.e., C.MILP.VI.CONV yields the smallest optimality gap, which is around 90% on average. Such results indicate that the MILP.VI.CONV model is the most effective in improving the root gap. We also notice that valid inequalities contribute more than convexification results in improving the optimality gap since C.MILP.VI is consistently better than C.MILP.CONV. Besides, C.MILP.CONV takes more time
Weijun Xie, Luying Sun, Tim Witten: Distributionally Robust Fair Transit Resource Allocation

than C.MILP.VI. This also explains why MILP.VI performs better than MILP.CONV in Table 1. Regarding both the solution time and the formulation strength, among the four different MILP formulations, MILP.VI tends to be the best. Since the continuous relaxation values of MILP.B are nearly 0, deriving stronger MILP formulations is of necessity.

We then compare the scenario decomposition Algorithm 1 based on four different MILP formulations (i.e., MILP.B, MILP.VI, MILP.CONV, MILP.VI.CONV) as well as two approximation algorithms (i.e., Algorithm 2, Algorithm 3) and results are reported in Table 3. Compared to directly solving MILP formulations using a solver as shown in Table 1, we see that for each MILP method, its corresponding scenario decomposition counterpart is more effective and has a much shorter solution time. More precisely, except MILP.B model, employing scenario decomposition Algorithm 1 reduces the solution time by about 60% comparing to solving its corresponding MILP formulation. This demonstrates the effectiveness of scenario decomposition Algorithm 1. As scenario decomposition based on MILP.VI method took the least time for nearly all the instances, we used it to the accelerated scenario decomposition Algorithm 2.

We see that if the Wasserstein radius $\theta$ increases, the solution time does not change too much for all the scenario decomposition methods. As the pandemic factor $\delta$ or the total number of buses
increases, the solution time decreases. As the weight $\omega$ increases, the solution time of Algorithm 1 based on MILP.B, MILP.CONV, MILP.VI.CONV decreases, while the solution time of Algorithm 1 based on MILP.VI increases. As the number of routes $I$ increases, the solution time increases for all the scenario decomposition methods.

When implementing the accelerated scenario decomposition Algorithm 2, we forced the lower bound to increase by 1% at each iteration. Therefore, compared to scenario decomposition Algorithm 1 based on MILP.VI, the accelerated scenario decomposition Algorithm 2 significantly reduces the running time by around 80%. Although the accelerated scenario decomposition Algorithm 2 may miss the optimal solution by cutting off plausible solutions, we see from Table 3 that it consistently finds optimal solutions for all the testing cases. We also applied the approximation Algorithm 3 by setting the maximum iteration to be 1000. We notice from Table 3 that for all the testing cases, the approximation Algorithm 3 can obtain an optimal solution and takes within a second to solve. This suggests that the approximation Algorithm 3 can be a strong alternative when the exact methods might not work well. We also see that no matter how the parameters change, the solution time does not change too much for both Algorithm 2 and Algorithm 3.

<table>
<thead>
<tr>
<th>Case</th>
<th>C MILP.B</th>
<th>C MILP.VI</th>
<th>C MILP.CONV</th>
<th>C MILP.VI.CONV</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 1</td>
<td>0.000</td>
<td>0.02</td>
<td>0.089</td>
<td>0.089</td>
</tr>
<tr>
<td>1 - 2</td>
<td>0.000</td>
<td>0.02</td>
<td>0.091</td>
<td>0.091</td>
</tr>
<tr>
<td>1 - 3</td>
<td>0.000</td>
<td>0.02</td>
<td>0.093</td>
<td>0.093</td>
</tr>
<tr>
<td>1 - 4</td>
<td>0.000</td>
<td>0.02</td>
<td>0.094</td>
<td>0.094</td>
</tr>
<tr>
<td>1 - 5</td>
<td>0.000</td>
<td>0.02</td>
<td>0.098</td>
<td>0.098</td>
</tr>
<tr>
<td>Average</td>
<td>0.000</td>
<td>0.02</td>
<td>0.096</td>
<td>0.096</td>
</tr>
<tr>
<td>1 - 6</td>
<td>0.000</td>
<td>0.02</td>
<td>0.089</td>
<td>0.089</td>
</tr>
<tr>
<td>1 - 7</td>
<td>0.000</td>
<td>0.02</td>
<td>0.091</td>
<td>0.091</td>
</tr>
<tr>
<td>1 - 8</td>
<td>0.000</td>
<td>0.02</td>
<td>0.093</td>
<td>0.093</td>
</tr>
<tr>
<td>1 - 9</td>
<td>0.000</td>
<td>0.02</td>
<td>0.094</td>
<td>0.094</td>
</tr>
<tr>
<td>1 - 10</td>
<td>0.000</td>
<td>0.02</td>
<td>0.098</td>
<td>0.098</td>
</tr>
<tr>
<td>Average</td>
<td>0.000</td>
<td>0.02</td>
<td>0.096</td>
<td>0.096</td>
</tr>
<tr>
<td>1 - 11</td>
<td>0.000</td>
<td>0.02</td>
<td>0.089</td>
<td>0.089</td>
</tr>
<tr>
<td>1 - 12</td>
<td>0.000</td>
<td>0.02</td>
<td>0.091</td>
<td>0.091</td>
</tr>
<tr>
<td>1 - 13</td>
<td>0.000</td>
<td>0.02</td>
<td>0.093</td>
<td>0.093</td>
</tr>
<tr>
<td>1 - 14</td>
<td>0.000</td>
<td>0.02</td>
<td>0.094</td>
<td>0.094</td>
</tr>
<tr>
<td>1 - 15</td>
<td>0.000</td>
<td>0.02</td>
<td>0.098</td>
<td>0.098</td>
</tr>
<tr>
<td>Average</td>
<td>0.000</td>
<td>0.02</td>
<td>0.096</td>
<td>0.096</td>
</tr>
<tr>
<td>1 - 16</td>
<td>0.000</td>
<td>0.02</td>
<td>0.044</td>
<td>0.044</td>
</tr>
<tr>
<td>1 - 17</td>
<td>0.000</td>
<td>0.02</td>
<td>0.045</td>
<td>0.045</td>
</tr>
<tr>
<td>1 - 18</td>
<td>0.000</td>
<td>0.02</td>
<td>0.046</td>
<td>0.046</td>
</tr>
<tr>
<td>1 - 19</td>
<td>0.000</td>
<td>0.02</td>
<td>0.047</td>
<td>0.047</td>
</tr>
<tr>
<td>1 - 20</td>
<td>0.000</td>
<td>0.02</td>
<td>0.048</td>
<td>0.048</td>
</tr>
<tr>
<td>Average</td>
<td>0.000</td>
<td>0.02</td>
<td>0.048</td>
<td>0.048</td>
</tr>
<tr>
<td>2 - 21</td>
<td>0.000</td>
<td>0.03</td>
<td>0.023</td>
<td>0.023</td>
</tr>
<tr>
<td>2 - 22</td>
<td>0.000</td>
<td>0.03</td>
<td>0.023</td>
<td>0.023</td>
</tr>
<tr>
<td>2 - 23</td>
<td>0.000</td>
<td>0.03</td>
<td>0.023</td>
<td>0.023</td>
</tr>
<tr>
<td>2 - 24</td>
<td>0.000</td>
<td>0.03</td>
<td>0.023</td>
<td>0.023</td>
</tr>
<tr>
<td>2 - 25</td>
<td>0.000</td>
<td>0.03</td>
<td>0.024</td>
<td>0.024</td>
</tr>
<tr>
<td>Average</td>
<td>0.000</td>
<td>0.03</td>
<td>0.023</td>
<td>0.023</td>
</tr>
<tr>
<td>2 - 26</td>
<td>0.000</td>
<td>0.03</td>
<td>0.031</td>
<td>0.031</td>
</tr>
<tr>
<td>2 - 27</td>
<td>0.000</td>
<td>0.03</td>
<td>0.031</td>
<td>0.031</td>
</tr>
<tr>
<td>2 - 28</td>
<td>0.000</td>
<td>0.03</td>
<td>0.031</td>
<td>0.031</td>
</tr>
<tr>
<td>2 - 29</td>
<td>0.000</td>
<td>0.03</td>
<td>0.031</td>
<td>0.031</td>
</tr>
<tr>
<td>2 - 30</td>
<td>0.000</td>
<td>0.03</td>
<td>0.034</td>
<td>0.034</td>
</tr>
<tr>
<td>Average</td>
<td>0.000</td>
<td>0.03</td>
<td>0.031</td>
<td>0.031</td>
</tr>
<tr>
<td>2 - 31</td>
<td>0.000</td>
<td>0.03</td>
<td>0.031</td>
<td>0.031</td>
</tr>
<tr>
<td>2 - 32</td>
<td>0.000</td>
<td>0.03</td>
<td>0.031</td>
<td>0.031</td>
</tr>
<tr>
<td>2 - 33</td>
<td>0.000</td>
<td>0.03</td>
<td>0.031</td>
<td>0.031</td>
</tr>
<tr>
<td>2 - 34</td>
<td>0.000</td>
<td>0.03</td>
<td>0.031</td>
<td>0.031</td>
</tr>
<tr>
<td>2 - 35</td>
<td>0.000</td>
<td>0.03</td>
<td>0.031</td>
<td>0.031</td>
</tr>
<tr>
<td>Average</td>
<td>0.000</td>
<td>0.03</td>
<td>0.031</td>
<td>0.031</td>
</tr>
</tbody>
</table>
Table 3 Results of Four Scenario Decomposition Algorithm 1 Based on Four Different MILP Formulations and Two Approximate Methods (i.e., Algorithm 2, Algorithm 3)

<table>
<thead>
<tr>
<th>Case</th>
<th>MILP.B</th>
<th>MILP.VI</th>
<th>MILP.VI CONV</th>
<th>MILP.CONV</th>
<th>Algorithm 2</th>
<th>Algorithm 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time</td>
<td>Gap (%)</td>
<td>Time</td>
<td>Gap (%)</td>
<td>Time</td>
<td>Gap (%)</td>
</tr>
<tr>
<td>1 - 1</td>
<td>22</td>
<td>8.0</td>
<td>37</td>
<td>21.0</td>
<td>8.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 2</td>
<td>14</td>
<td>6.0</td>
<td>36</td>
<td>20.0</td>
<td>12</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 3</td>
<td>27</td>
<td>9.0</td>
<td>128</td>
<td>27.0</td>
<td>10</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 4</td>
<td>40</td>
<td>8.0</td>
<td>83</td>
<td>71.0</td>
<td>11</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 5</td>
<td>72</td>
<td>8.0</td>
<td>162</td>
<td>31.0</td>
<td>10</td>
<td>0.0</td>
</tr>
<tr>
<td>Average</td>
<td>35</td>
<td>8.0</td>
<td>62</td>
<td>31.0</td>
<td>10</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 6</td>
<td>35</td>
<td>9.0</td>
<td>39</td>
<td>32.0</td>
<td>9.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 7</td>
<td>19</td>
<td>7.0</td>
<td>35</td>
<td>21.0</td>
<td>7.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 8</td>
<td>50</td>
<td>6.0</td>
<td>34</td>
<td>20.0</td>
<td>6.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 9</td>
<td>23</td>
<td>13.0</td>
<td>49</td>
<td>18.0</td>
<td>15</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 10</td>
<td>22</td>
<td>17.0</td>
<td>69</td>
<td>38.0</td>
<td>10</td>
<td>0.0</td>
</tr>
<tr>
<td>Average</td>
<td>29</td>
<td>10.0</td>
<td>45</td>
<td>26.0</td>
<td>9.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 11</td>
<td>21</td>
<td>14.0</td>
<td>45</td>
<td>28.0</td>
<td>8.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 12</td>
<td>17</td>
<td>11.0</td>
<td>38</td>
<td>18.0</td>
<td>7.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 13</td>
<td>15</td>
<td>13.0</td>
<td>38</td>
<td>17.0</td>
<td>11</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 14</td>
<td>19</td>
<td>14.0</td>
<td>55</td>
<td>25.0</td>
<td>13</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 15</td>
<td>49</td>
<td>13.0</td>
<td>69</td>
<td>49.0</td>
<td>11</td>
<td>0.0</td>
</tr>
<tr>
<td>Average</td>
<td>29</td>
<td>10.0</td>
<td>45</td>
<td>26.0</td>
<td>9.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 16</td>
<td>39</td>
<td>13.0</td>
<td>65</td>
<td>36.0</td>
<td>5.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 17</td>
<td>25</td>
<td>11.0</td>
<td>52</td>
<td>33.0</td>
<td>5.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 18</td>
<td>27</td>
<td>10.0</td>
<td>45</td>
<td>35.0</td>
<td>5.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 19</td>
<td>33</td>
<td>9.0</td>
<td>76</td>
<td>33.0</td>
<td>5.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 20</td>
<td>33</td>
<td>9.0</td>
<td>54</td>
<td>31.0</td>
<td>3.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Average</td>
<td>31</td>
<td>9.0</td>
<td>58</td>
<td>34.0</td>
<td>8.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 21</td>
<td>20</td>
<td>6.0</td>
<td>28</td>
<td>25.0</td>
<td>2.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 22</td>
<td>14</td>
<td>6.0</td>
<td>26</td>
<td>18.0</td>
<td>4.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 23</td>
<td>12</td>
<td>5.0</td>
<td>20</td>
<td>17.0</td>
<td>3.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 24</td>
<td>15</td>
<td>6.0</td>
<td>24</td>
<td>19.0</td>
<td>2.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1 - 25</td>
<td>13</td>
<td>7.0</td>
<td>26</td>
<td>17.0</td>
<td>2.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Average</td>
<td>25</td>
<td>6.0</td>
<td>45</td>
<td>24.0</td>
<td>5.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2 - 26</td>
<td>38</td>
<td>7.0</td>
<td>84</td>
<td>47.0</td>
<td>9.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2 - 27</td>
<td>44</td>
<td>7.0</td>
<td>51</td>
<td>29.0</td>
<td>9.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2 - 28</td>
<td>41</td>
<td>8.0</td>
<td>57</td>
<td>29.0</td>
<td>8.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2 - 29</td>
<td>35</td>
<td>5.0</td>
<td>66</td>
<td>20.0</td>
<td>9.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2 - 30</td>
<td>25</td>
<td>12.0</td>
<td>53</td>
<td>24.0</td>
<td>9.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Average</td>
<td>25</td>
<td>8.0</td>
<td>62</td>
<td>30.0</td>
<td>8.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2 - 31</td>
<td>35</td>
<td>10.0</td>
<td>80</td>
<td>39.0</td>
<td>10</td>
<td>0.0</td>
</tr>
<tr>
<td>2 - 32</td>
<td>34</td>
<td>7.0</td>
<td>49</td>
<td>24.0</td>
<td>6.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2 - 33</td>
<td>30</td>
<td>6.0</td>
<td>50</td>
<td>21.0</td>
<td>9.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2 - 34</td>
<td>27</td>
<td>6.0</td>
<td>43</td>
<td>19.0</td>
<td>9.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2 - 35</td>
<td>21</td>
<td>9.0</td>
<td>53</td>
<td>22.0</td>
<td>6.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Average</td>
<td>29</td>
<td>8.0</td>
<td>55</td>
<td>25.0</td>
<td>8.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

According to the results in Table 1, Table 2, and Table 3, we see that MILP.VI is the most efficient one among four MILP formulations, scenario decomposition Algorithm 1 based on MILP.VI denoted by “Algorithm 1.VI” is the most efficient one among four different scenario decomposition methods using different MILP formulations, and both approximation methods are quite good. To further compare those methods, we applied MILP.B, MILP.VI, SceDecomp.VI, Algorithm 2 and Algorithm 3 to more difficult instances with a larger objective value. For Algorithm 3, we let the maximum number of iterations equal to 5 and 100 to see the influence of the maximum number of iterations on the solution quality. These two different configurations are denoted by Algorithm 3(max iter 5) and Algorithm 3(max iter 100). Results are displayed in Table 4.

From Table 4, similar to previous results, we see that MILP.VI reduces the solution time compared to MILP.B. Although Algorithm 1.VI outperforms MILP.VI, Algorithm 2 is the best among these three. We also see that Algorithm 3 (max iter 5) has the shortest solution time, but it misses the optimal solution, while Algorithm 3(max iter 100) takes a longer average solution time. However, its solution quality is better, and its average solution time is overall shorter than that of accel-
iterated scenario decomposition Algorithm 1. Therefore, in practice, we suggest using approximation Algorithm 3 with a properly chosen maximum number of iterations to solve difficult instances.

Table 4 Results of Instances with a Larger Objective Value

<table>
<thead>
<tr>
<th>Case</th>
<th>ω</th>
<th>θ</th>
<th>OPT</th>
<th>MILP-B</th>
<th>MILP.VI</th>
<th>Algorithm 1 VI</th>
<th>Algorithm 2</th>
<th>Algorithm 3 (max itr 5)</th>
<th>Algorithm 3 (max itr 100)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 - 36</td>
<td>0.25</td>
<td>0.0</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
</tr>
<tr>
<td>3 - 37</td>
<td>0.25</td>
<td>0.0</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
</tr>
<tr>
<td>3 - 38</td>
<td>0.5</td>
<td>0.0</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
</tr>
<tr>
<td>3 - 39</td>
<td>1</td>
<td>0.0</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
</tr>
<tr>
<td>3 - 40</td>
<td>1.5</td>
<td>0.0</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 - 41</td>
<td>0.5</td>
<td>0</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
</tr>
<tr>
<td>4 - 42</td>
<td>0.25</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
</tr>
<tr>
<td>4 - 43</td>
<td>0.0</td>
<td>0</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
</tr>
<tr>
<td>4 - 44</td>
<td>1</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
</tr>
<tr>
<td>4 - 45</td>
<td>1.5</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
<td>1.105</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 - 46</td>
<td>0</td>
<td>0</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
</tr>
<tr>
<td>5 - 47</td>
<td>0.25</td>
<td>0</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
</tr>
<tr>
<td>5 - 48</td>
<td>0.5</td>
<td>0</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
</tr>
<tr>
<td>5 - 49</td>
<td>1</td>
<td>0</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
</tr>
<tr>
<td>5 - 50</td>
<td>1.5</td>
<td>0</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
<td>1.006</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7.2. Results of Larger Instances

This subsection focuses on larger instances. We generated 3 larger instances where the number of bus routes $I \in \{10, 20\}$ and the number of scenarios $N \in \{10, 20\}$. There are $K = 3$ types of buses with nominal capacities $c_1 = 60, c_2 = 80, c_3 = 120$ and each route has 10 to 40 stops. The pandemic factor $\delta$ is 0.25. We tested on 20 cases based on these instances using different parameter configurations (i.e., different values of $N, I, \eta, \omega, \theta$). Particularly, cases 51-60, 61-65, 66-70 correspond to instances 6, 7, 8, respectively.

Results of different MILP formulations are reported in Table 5 and Table 6, where “Obj.Val” denotes the best upper bound, “LB” denotes the best lower bound, “Opt.Gap” represents the optimality gap computed as the percentage of the difference between Obj.Val and LB divided by Obj.Val, and “/” denotes the cases that no solution was found within the time limit (i.e., 3600 seconds). Unfortunately, no case can be solved to optimality in time limit. For the cases 64, 65, 69, 70, MILP.VI.CONV cannot even find a feasible solution within the time limit. We notice that MILP.VI consistently obtains the best lower bound among the four MILP formulations almost for each case. However, albeit promising, MILP.VI may not be ideal for solving extremely large instances since the optimality gap is quite large within the time limit.

Results of continuous relaxations of MILP formulations are displayed in Table 7. Similarly, we see that the continuous relaxation of MILP.VI.CONV is the best compared to other formulations; however, it takes the longest time to compute. On the other hand, the continuous relaxation values of MILP.VI are comparable to those of MILP.VI.CONV but takes a much shorter time. This somehow explains why MILP.VI works the best among all the MILP formulations. Results of cases 61-70 show that for very large-scale cases, even solving a continuous relaxation takes a relatively
### Table 5 Results of MILP.B and MILP.VI for Solving Larger Instances

<table>
<thead>
<tr>
<th>Case</th>
<th>N</th>
<th>f</th>
<th>( \eta )</th>
<th>( \omega )</th>
<th>( \theta )</th>
<th>MILP.B</th>
<th>MILP.VI</th>
<th>MILP.B</th>
<th>MILP.VI</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 - 51</td>
<td>0</td>
<td>0.743</td>
<td>3600</td>
<td>39.8</td>
<td>0.447</td>
<td>16791</td>
<td>0.743</td>
<td>3600</td>
<td>34.8</td>
</tr>
<tr>
<td>6 - 52</td>
<td>0.25</td>
<td>0.703</td>
<td>3600</td>
<td>48.0</td>
<td>0.397</td>
<td>10684</td>
<td>0.763</td>
<td>3600</td>
<td>37.6</td>
</tr>
<tr>
<td>6 - 53</td>
<td>0.5</td>
<td>0.775</td>
<td>3600</td>
<td>39.8</td>
<td>0.466</td>
<td>20265</td>
<td>0.775</td>
<td>3600</td>
<td>32.1</td>
</tr>
<tr>
<td>6 - 54</td>
<td>1</td>
<td>0.778</td>
<td>3600</td>
<td>35.1</td>
<td>0.505</td>
<td>14401</td>
<td>0.778</td>
<td>3600</td>
<td>34.3</td>
</tr>
<tr>
<td>6 - 55</td>
<td>1.5</td>
<td>0.817</td>
<td>3600</td>
<td>32.9</td>
<td>0.549</td>
<td>21098</td>
<td>0.817</td>
<td>3600</td>
<td>41.1</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td></td>
<td>0.743</td>
<td>3600</td>
<td>39.8</td>
<td>0.447</td>
<td>16791</td>
<td>0.743</td>
<td>3600</td>
</tr>
<tr>
<td>7 - 61</td>
<td>0</td>
<td>0.743</td>
<td>3600</td>
<td>24.3</td>
<td>0.563</td>
<td>27659</td>
<td>0.743</td>
<td>3600</td>
<td>34.5</td>
</tr>
<tr>
<td>7 - 62</td>
<td>0.25</td>
<td>0.763</td>
<td>3600</td>
<td>30.5</td>
<td>0.530</td>
<td>25626</td>
<td>0.763</td>
<td>3600</td>
<td>40.8</td>
</tr>
<tr>
<td>7 - 63</td>
<td>0.5</td>
<td>0.775</td>
<td>3600</td>
<td>39.4</td>
<td>0.469</td>
<td>18724</td>
<td>0.775</td>
<td>3600</td>
<td>44.7</td>
</tr>
<tr>
<td>7 - 64</td>
<td>1</td>
<td>0.778</td>
<td>3600</td>
<td>35.3</td>
<td>0.503</td>
<td>12028</td>
<td>0.778</td>
<td>3600</td>
<td>35.1</td>
</tr>
<tr>
<td>7 - 65</td>
<td>1.5</td>
<td>0.817</td>
<td>3600</td>
<td>27.5</td>
<td>0.593</td>
<td>34584</td>
<td>0.817</td>
<td>3600</td>
<td>46.9</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td></td>
<td>0.743</td>
<td>3600</td>
<td>34.0</td>
<td>0.563</td>
<td>27659</td>
<td>0.743</td>
<td>3600</td>
</tr>
</tbody>
</table>

### Table 6 Results of MILP.CONV and MILP.VI.CONV for Solving Larger Instances

<table>
<thead>
<tr>
<th>Case</th>
<th>MILP.CONV</th>
<th>MILP.VI.CONV</th>
<th>MILP.CONV</th>
<th>MILP.VI.CONV</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 - 51</td>
<td>0.755</td>
<td>3600</td>
<td>49.5</td>
<td>0.381</td>
</tr>
<tr>
<td>6 - 52</td>
<td>0.773</td>
<td>3600</td>
<td>48.2</td>
<td>0.400</td>
</tr>
<tr>
<td>6 - 53</td>
<td>0.788</td>
<td>3600</td>
<td>42.9</td>
<td>0.469</td>
</tr>
<tr>
<td>6 - 54</td>
<td>0.861</td>
<td>3600</td>
<td>55.4</td>
<td>0.352</td>
</tr>
<tr>
<td>6 - 55</td>
<td>0.864</td>
<td>3600</td>
<td>55.4</td>
<td>0.352</td>
</tr>
<tr>
<td>Average</td>
<td>0.755</td>
<td>3600</td>
<td>54.8</td>
<td>0.371</td>
</tr>
<tr>
<td>7 - 62</td>
<td>1.392</td>
<td>3600</td>
<td>81.0</td>
<td>0.261</td>
</tr>
<tr>
<td>7 - 63</td>
<td>1.112</td>
<td>3600</td>
<td>80.6</td>
<td>0.215</td>
</tr>
<tr>
<td>7 - 64</td>
<td>1.134</td>
<td>3600</td>
<td>80.5</td>
<td>0.272</td>
</tr>
<tr>
<td>7 - 65</td>
<td>1.332</td>
<td>3600</td>
<td>81.1</td>
<td>0.252</td>
</tr>
<tr>
<td>8 - 66</td>
<td>1.333</td>
<td>3600</td>
<td>80.3</td>
<td>0.267</td>
</tr>
<tr>
<td>Average</td>
<td>1.392</td>
<td>3600</td>
<td>80.7</td>
<td>0.261</td>
</tr>
</tbody>
</table>

long time to solve to optimality, indicating that exact methods may also have trouble in solving these cases to optimality within an hour.

Results of scenario decomposition Algorithm 1 based on four different MILP formulations, accelerated scenario decomposition Algorithm 2 based on MILP.VI and approximation algorithm Algorithm 3 (with maximum iteration equal to 5) are shown in Table 8 and Table 9, where “Best.Obj” represents the best objective value obtained by four MILP formulations and Algorithm 1 based on four MILP formulations for each case. It is seen that Algorithm 3 is the only method that can find feasible solutions to cases 61-65. For cases 66-70, only Algorithm 1 based on MILP.VI, Algorithm 2, and Algorithm 3 can obtain feasible solutions within the time limit. For the difficult cases 61-70, Algorithm 3 find better solutions than the best ones output by the exact methods. For easy cases 51-60, both Algorithm 2 and Algorithm 3 find exactly the same solutions as the
Table 7: Results of Continuous Relaxations of Four Different MILP Formulations for Solving Larger Instances

<table>
<thead>
<tr>
<th>Case</th>
<th>C.MILP.B</th>
<th>C.MILP.VI</th>
<th>C.MILP.CONV</th>
<th>C.MILP.VI.CONV</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 - 51</td>
<td>0.000</td>
<td>0.16</td>
<td>0.044</td>
<td>0.83</td>
</tr>
<tr>
<td>6 - 52</td>
<td>0.000</td>
<td>0.15</td>
<td>0.045</td>
<td>0.96</td>
</tr>
<tr>
<td>6 - 53</td>
<td>0.000</td>
<td>0.34</td>
<td>0.046</td>
<td>1.09</td>
</tr>
<tr>
<td>6 - 55</td>
<td>0.000</td>
<td>0.19</td>
<td>0.048</td>
<td>0.67</td>
</tr>
<tr>
<td>Average</td>
<td>0.040</td>
<td>0.83</td>
<td>0.044</td>
<td>0.98</td>
</tr>
<tr>
<td>6 - 56</td>
<td>0.000</td>
<td>0.16</td>
<td>0.044</td>
<td>0.98</td>
</tr>
<tr>
<td>6 - 57</td>
<td>0.000</td>
<td>0.15</td>
<td>0.045</td>
<td>0.74</td>
</tr>
<tr>
<td>6 - 58</td>
<td>0.000</td>
<td>0.14</td>
<td>0.046</td>
<td>0.62</td>
</tr>
<tr>
<td>6 - 59</td>
<td>0.000</td>
<td>0.14</td>
<td>0.046</td>
<td>0.85</td>
</tr>
<tr>
<td>6 - 60</td>
<td>0.000</td>
<td>0.14</td>
<td>0.048</td>
<td>0.79</td>
</tr>
<tr>
<td>Average</td>
<td>0.039</td>
<td>0.80</td>
<td>0.044</td>
<td>0.98</td>
</tr>
</tbody>
</table>

best ones output by the exact methods. Overall, we conclude that among all the exact methods, when solving moderate-sized cases, Algorithm 1 based on MILP.VI outperforms other methods by providing the smallest optimality gaps; on the other hand, for the difficult cases, none of the exact methods works well, MILP.VI and Algorithm 1 based on MILP.VI are slightly better than others since they can consistently find a better solution. We also see that Algorithm 3 consistently finds either a better solution or the same quality solution. Therefore, in practice, when facing very large-scale cases or involving multiple-round cross-validations, we suggest using Algorithm 3.

Table 8: Results of Three Scenario Decomposition Algorithm 1 Based on Three Different MILP Formulations (i.e., MILP.B, MILP.CONV, MILP.CONV.VI) for Solving Larger Instances

<table>
<thead>
<tr>
<th>Case</th>
<th>Scenario Decomposition</th>
<th>MILP.B</th>
<th>MILP.VI</th>
<th>MILP.VI.CONV</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 - 51</td>
<td></td>
<td>0.743</td>
<td>0.708</td>
<td>3600 4.7</td>
</tr>
<tr>
<td>6 - 52</td>
<td></td>
<td>0.763</td>
<td>0.734</td>
<td>3600 3.8</td>
</tr>
<tr>
<td>6 - 53</td>
<td></td>
<td>0.775</td>
<td>0.745</td>
<td>3600 3.9</td>
</tr>
<tr>
<td>6 - 54</td>
<td></td>
<td>0.778</td>
<td>0.748</td>
<td>3600 3.8</td>
</tr>
<tr>
<td>6 - 55</td>
<td></td>
<td>0.817</td>
<td>0.785</td>
<td>3600 3.9</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>0.760</td>
<td>0.737</td>
<td>3600 4.0</td>
</tr>
<tr>
<td>7 - 61</td>
<td>/ / 3600 / / 4.4 / / 3600 / / 3.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7 - 62</td>
<td>/ / 3600 / / 4.4 / / 3600 / / 3.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7 - 63</td>
<td>/ / 3600 / / 4.4 / / 3600 / / 3.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7 - 64</td>
<td>/ / 3600 / / 4.4 / / 3600 / / 3.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7 - 65</td>
<td>/ / 3600 / / 4.4 / / 3600 / / 3.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>0.760</td>
<td>0.737</td>
<td>3600 4.0</td>
</tr>
</tbody>
</table>

Weijun Xie, Luying Sun, Tim Witten: Distributionally Robust Fair Transit Resource Allocation
Table 9 Results of the Scenario Decomposition Algorithm 1 Based on MILP.VI, Denoted by Algorithm 1.VI, and Two Approximate Methods (i.e., Algorithm 2, Algorithm 3) for Solving Larger Instances

<table>
<thead>
<tr>
<th>Case</th>
<th>Algorithm 1.VI</th>
<th>Best Obj</th>
<th>Algorithm 2</th>
<th>Algorithm 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 - 51</td>
<td>0.743</td>
<td>0.719</td>
<td>3.3</td>
<td>0.743</td>
</tr>
<tr>
<td>6 - 52</td>
<td>0.763</td>
<td>0.748</td>
<td>2.0</td>
<td>0.763</td>
</tr>
<tr>
<td>6 - 53</td>
<td>0.775</td>
<td>0.757</td>
<td>2.2</td>
<td>0.775</td>
</tr>
<tr>
<td>6 - 54</td>
<td>0.778</td>
<td>0.762</td>
<td>2.0</td>
<td>0.778</td>
</tr>
<tr>
<td>6 - 55</td>
<td>0.817</td>
<td>0.797</td>
<td>2.5</td>
<td>0.817</td>
</tr>
<tr>
<td>Average</td>
<td>3600.0</td>
<td>2.4</td>
<td></td>
<td>133.0</td>
</tr>
<tr>
<td>6 - 56</td>
<td>0.743</td>
<td>0.719</td>
<td>3.5</td>
<td>0.743</td>
</tr>
<tr>
<td>6 - 57</td>
<td>0.763</td>
<td>0.748</td>
<td>2.0</td>
<td>0.763</td>
</tr>
<tr>
<td>6 - 58</td>
<td>0.775</td>
<td>0.754</td>
<td>2.7</td>
<td>0.775</td>
</tr>
<tr>
<td>6 - 59</td>
<td>0.778</td>
<td>0.756</td>
<td>1.5</td>
<td>0.778</td>
</tr>
<tr>
<td>6 - 60</td>
<td>0.817</td>
<td>0.800</td>
<td>2.1</td>
<td>0.817</td>
</tr>
<tr>
<td>Average</td>
<td>3600.0</td>
<td>2.3</td>
<td></td>
<td>195.0</td>
</tr>
<tr>
<td>7 - 61</td>
<td>/</td>
<td>/</td>
<td>/</td>
<td>/</td>
</tr>
<tr>
<td>7 - 62</td>
<td>/</td>
<td>/</td>
<td>/</td>
<td>/</td>
</tr>
<tr>
<td>7 - 63</td>
<td>/</td>
<td>/</td>
<td>/</td>
<td>/</td>
</tr>
<tr>
<td>7 - 64</td>
<td>/</td>
<td>/</td>
<td>/</td>
<td>/</td>
</tr>
<tr>
<td>7 - 65</td>
<td>/</td>
<td>/</td>
<td>/</td>
<td>/</td>
</tr>
<tr>
<td>Average</td>
<td>3600.0</td>
<td>2.5</td>
<td></td>
<td>119.0</td>
</tr>
<tr>
<td>8 - 66</td>
<td>0.768</td>
<td>0.699</td>
<td>9.4</td>
<td>0.798</td>
</tr>
<tr>
<td>8 - 67</td>
<td>0.789</td>
<td>0.718</td>
<td>9.0</td>
<td>0.923</td>
</tr>
<tr>
<td>8 - 68</td>
<td>0.800</td>
<td>0.723</td>
<td>9.7</td>
<td>1.349</td>
</tr>
<tr>
<td>8 - 69</td>
<td>0.849</td>
<td>0.769</td>
<td>10.3</td>
<td>1.294</td>
</tr>
<tr>
<td>8 - 70</td>
<td>0.849</td>
<td>0.769</td>
<td>9.4</td>
<td>0.867</td>
</tr>
<tr>
<td>Average</td>
<td>3600.0</td>
<td>9.6</td>
<td></td>
<td>3600.0</td>
</tr>
</tbody>
</table>

7.3. Blacksburg Transit Case Study

In this section, we applied approximation Algorithm 3 to solve the real-world Blacksburg Transit allocation problem since the dataset provided by Blacksburg Transit is of a similar scale as the largest cases studied in the previous section. The operation data was provided by Blacksburg Transit, and passenger arrival and alighting data were collected in September 2020, amidst the COVID-19 pandemic. Blacksburg Transit has 17 routes, three types of buses with different capacities, where the number of buses with the nominal capacity equal to 60, 80, 120 are 8, 20, 8, respectively, and each route has 10 to 40 stops. As public policies and government regulations change as the pandemic evolves, we study the pandemic factor $\delta$ varying from 0.25 (social distancing required by CDC) to 1 (fully operated). We set the weight $\omega = 0.5$ in this numerical study since the priority during a pandemic is to enforce the social distancing. We adopted the following cross-validation procedure to choose the best Wasserstein radius $\theta$ for a given $\delta$: (i) randomly select 20 scenarios as the training data and the remaining 8 scenarios are for testing; (ii) for each $\theta \in \{0, 0.2, \ldots, 2\}$, solve the DrFRAM model using the training data and evaluate the solution using the test data; (iii) repeat (i) and (ii) for 15 times and derive the asymptotic 95% confidence interval for the mean of training and testing data (sample mean $\pm 2.145 \times$ sample standard deviation/$\sqrt{20}$). We selected the smallest $\theta$ such that its confidence interval of DrFRAM objective value using the training data is beyond the confidence interval of risk-neutral objective value using the testing data. A time limit of 3600 seconds was set for a single run. Note that when all the DrFRAM objective values corresponding to different Wasserstein radii $\theta$ are much less than 1, then the cross-validation takes about 1000 seconds; otherwise, if for some Wasserstein radii, their corresponding DrFRAM objective values are around or exceeding 1, then the cross-validation can take up to 15 hours.
We see that for any $\delta \in \{0.5, 0.75, 1\}$ and for any $\theta \in \{0, 0.2, \ldots, 2\}$, since objective values for all the scenarios are observed to be less than 1, according to Proposition 8 and its remark, the solution obtained by approximation Algorithm 3 can be very close to the optimality. Besides, since we obtain the same solution when $\delta \geq 0.5$, we only consider $\delta \in \{0.25, 0.5\}$. Thus, according to our cross validation results in Table 10 and Figure 1, we chose $\theta = 2.2$ as the best Wasserstein radius when $\delta = 0.25$ and $\theta = 4.4$ when $\delta = 0.5$.

<table>
<thead>
<tr>
<th>$\delta$</th>
<th>$\theta$</th>
<th>$\delta = 0.25$</th>
<th>$\delta = 0.5$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$95%$ C.I. of training</td>
<td>$95%$ C.I. of testing</td>
</tr>
<tr>
<td>0.2</td>
<td>0.555</td>
<td>(0.276, 0.279)</td>
<td>(0.284, 0.288)</td>
</tr>
<tr>
<td>0.4</td>
<td>0.555</td>
<td>(0.277, 0.279)</td>
<td>(0.287, 0.290)</td>
</tr>
<tr>
<td>0.6</td>
<td>0.555</td>
<td>(0.277, 0.279)</td>
<td>(0.287, 0.290)</td>
</tr>
<tr>
<td>0.8</td>
<td>0.555</td>
<td>(0.277, 0.279)</td>
<td>(0.287, 0.290)</td>
</tr>
<tr>
<td>1</td>
<td>0.555</td>
<td>(0.277, 0.279)</td>
<td>(0.287, 0.290)</td>
</tr>
<tr>
<td>1.2</td>
<td>0.664</td>
<td>(0.275, 0.282)</td>
<td>(0.363, 0.406)</td>
</tr>
<tr>
<td>1.4</td>
<td>0.693</td>
<td>(0.275, 0.282)</td>
<td>(0.363, 0.406)</td>
</tr>
<tr>
<td>1.6</td>
<td>0.693</td>
<td>(0.275, 0.282)</td>
<td>(0.363, 0.406)</td>
</tr>
<tr>
<td>1.8</td>
<td>0.695</td>
<td>(0.275, 0.282)</td>
<td>(0.363, 0.406)</td>
</tr>
<tr>
<td>2</td>
<td>0.695</td>
<td>(0.275, 0.282)</td>
<td>(0.363, 0.406)</td>
</tr>
<tr>
<td>2.2</td>
<td>1.166</td>
<td>(0.996, 1.029)</td>
<td>(0.601, 0.615)</td>
</tr>
<tr>
<td>2.4</td>
<td>1.165</td>
<td>(1.006, 1.046)</td>
<td>(0.506, 0.599)</td>
</tr>
<tr>
<td>2.6</td>
<td>1.165</td>
<td>(1.006, 1.046)</td>
<td>(0.506, 0.599)</td>
</tr>
</tbody>
</table>

We evaluated the current bus assignment from Blacksburg Transit and compared the results with ours for the whole $N = 28$ scenarios. Results are shown in Figure 2. It is seen that our solutions can significantly decrease the utilization rate and passenger abandon rate for different Wasserstein radii. The improvements of weighted highest bus utilization rate and largest passenger abandon rate are 50% and 55% on average when the weight $\delta = 0.25$ and $\delta = 0.5$, respectively. We also compare the results for each route with the best tuned Wasserstein radius, which is illustrated in Figure 3. We see that our solution is of much less fluctuation than the current one in both
bus utilization rate and passenger abandon rate. For $\delta = 0.25$ and the best tuned Wasserstein radius $\theta = 2.2$, our route-based bus utilization rate and passenger abandon rate has mean 0.793 and standard deviation 0.184, while the current one has mean 0.842 and standard deviation 0.410. For $\delta = 0.5$ and the best tuned Wasserstein radius $\theta = 4.4$, our route-based bus utilization rate and passenger abandon rate has mean 0.614 and standard deviation 0.123, while the current one has mean 0.718 and standard deviation 0.394. This implies that our solution can indeed be much fairer and can significantly reduce the transit resource inequalities among different routes.

Figure 2  A Comparison of the Weighted Highest Bus Utilization Rate and the Largest Passenger Abandon Rate when Wasserstein Radius Varies

Figure 3  A Comparison of the Weighted Bus Utilization Rate and Passenger Abandon Rate for Each Route. Here, x-ticks are route names from Blacksburg Transit.
8. Conclusion

In this paper, we study the transit resource allocation problem to minimize the highest utilization rate and the largest passenger abandon rate under stochastic passenger arrival and alighting rates. We propose a DrFRAM under type-$\infty$ Wasserstein ambiguity set, which is proven to be NP-hard. To simplify the DrFRAM, we derive the monotonicity properties of the DrFRAM and use McCormick inequalities to linearize the nonlinear components, which allows us to derive an MILP formulation. To further improve the MILP formulation, valid inequalities and stronger formulations are derived. We also develop scenario decomposition methods and No-one-left based approximation algorithm for solving DrFRAM. Finally, we numerically demonstrate the effectiveness of the proposed approaches in both small and large instances and apply them to solve the real-world instance using the data provided by Blacksburg Transit. Comparing to the current allocation plan, our result is demonstrated to be more robust and fairer.

References


