A Double Decomposition Algorithm for Network Planning and Operations in Deviated Fixed-route Microtransit

Bernardo Martin-Iradi
Institute for Transport Planning and Systems, ETH Zurich, Zurich, Switzerland

Alexandria Schmid, Kayla Cummings, Alexandre Jacquillat
Sloan School of Management and Operations Research Center, MIT, Cambridge, MA

Microtransit offers opportunities to enhance urban mobility by combining the reliability of public transit and the flexibility of ride-sharing. This paper optimizes the design and operations of a deviated fixed-route microtransit system that relies on reference lines but can deviate on demand in response to passenger requests. We formulate a *Microtransit Network Design (MiND)* model via two-stage stochastic integer optimization, with a first-stage network design and service scheduling structure and a second-stage vehicle routing structure. We derive a tight second-stage relaxation using a subpath-based representation of microtransit operations in a load-expanded network. We develop a double-decomposition algorithm combining Benders decomposition and subpath-based column generation. We prove that the algorithm maintains a valid optimality gap and converges to an optimal solution in a finite number of iterations. Results obtained with real-world data from Manhattan show that the methodology scales to large and otherwise-intractable instances, with up to 10-100 candidate lines and hundreds of stops. Comparisons with transit and ride-sharing suggest that microtransit can provide win-win outcomes toward efficient mobility (high demand coverage, low costs, high level of service), equitable mobility (broad geographic reach) and sustainable mobility (limited environmental footprint). We provide an open-source implementation to enable replication.

Key words: Microtransit, stochastic optimization, Benders decomposition, column generation

1. Introduction

Major cities face critical challenges to meet mobility needs in the midst of rising congestion, greenhouse gas emissions and socioeconomic inequalities. Static transit infrastructure offers limited flexibility to respond to ever-changing mobility needs, resulting in a ridership decline (The Economist 2018) and transit deserts (Allen 2017). Simultaneously, ride-sharing provides flexible, on-demand mobility services, but low-occupancy vehicles still lead to high fares, congestion, and emissions. This context identifies opportunities to leverage emerging *microtransit* services toward efficient, equitable, and sustainable mobility. Broadly defined by the US DoT (2016) as "privately owned and operated shared transportation system(s) that can offer fixed routes and schedules, as well as flexible routes and on-demand scheduling," microtransit shepherds the digital capabilities and operating flexibility of ride-sharing into the realm of public transit. Yet, microtransit raises critical questions about *how* to combine transit and ride-sharing components into low-cost, high-quality services and how to develop dedicated analytics capabilities (McKinsey & Co. 2018).

This paper develops a two-stage stochastic integer optimization methodology to support the design and operations of *deviated fixed-route microtransit*—a hybrid microtransit system based on transit lines to consolidate passenger demand into high-capacity vehicles and on-demand deviations in response to passenger requests. Our model optimizes network design and service scheduling under demand uncertainty in the first stage, and on-demand operations in the second stage. We propose a scalable methodology, relying on (i) a network-based second-stage formulation with a tight linear relaxation but an exponential number of subpath-based variables; and (ii) a double-decomposition algorithm combining Benders decomposition and subpath-based column generation. The objectives of the paper are to establish its scalability in large and practical problem instances, and to assess the performance of deviated fixed-route microtransit in the urban mobility ecosystem.

Our first contribution is to formulate a *Microtransit Network Design (MiND)* model via two-stage stochastic optimization (Section 3). The first-stage problem optimizes network design and service scheduling by selecting line-based *reference trips*. The second-stage problem reflects on-demand routing operations in response to passenger requests, under vehicle capacity and time window constraints; in particular, microtransit operations must visit checkpoints on the reference lines at designated times. The objective combines minimizing planning costs, maximizing ridership and maximizing passenger level of service. For simplicity, we focus primarily on a MiND-VRP problem, corresponding to a vehicle routing setting in which all passengers have the same origin or the same destination—motivated by use cases such as airport or university shuttles. In EC.4, we extend the methodology and main results to a MiND-DAR problem, corresponding to a dial-a-ride setting in which passengers request transportation from origin to destination; and in EC.5, we extend them to a MiND-Tr problem that allows transfers between microtransit lines.

One of the main complexities of the problem lies in its discrete second-stage structure—a capacitated vehicle routing problem with time windows. To retain a tight recourse formulation, we propose a subpath-based representation of second-stage microtransit operations in a load-expanded network. Each node encodes a checkpoint on the reference line and a vehicle load, and each arc characterizes on-demand operations between checkpoints. Subpaths encapsulate time window constraints, and load expansion accommodates vehicle capacities without big-M constraints, enabling a continuous recourse function approximation. We show that our subpath-based variables enable a more effective formulation than a compact formulation (tighter second-stage formulation), than a segment-based benchmark with variables connecting consecutive stops in a granular time-load-expanded network (sparser network) and than a path-based benchmark with variables connecting the start to the end of each line (much slower rate of exponential growth in the number of variables).

Our second contribution is a scalable double-decomposition (DD) algorithm combining Benders decomposition and subpath-based column generation (Section 4). The Benders decomposition scheme iterates between a first-stage network design problem and second-stage routing problems, exploiting the nested block-diagonal structure to decompose on-demand operations for each reference trip in each scenario. The column generation scheme adds subpath-based variables iteratively in the Benders subproblem. We develop exact and heuristic label-setting algorithms to generate subpaths of negative reduced cost while keeping track of vehicle load and level of service. Our methodology induces a double-decomposition structure: the column generation pricing problem adds subpath-based arcs into load-expanded networks (i.e., local on-demand deviations between checkpoints); the Benders subproblem combines them into full second-stage paths (i.e., full microtransit trips for each reference trip in each scenario), and the Benders master problem optimizes first-stage planning decisions accordingly (i.e., network design and service scheduling). This algorithm converges to the partial relaxation of the (tight) subpath-based formulation with mixed-integer first-stage variables and continuous second-stage variables. To guarantee convergence and second-stage integrality, we augment the DD scheme with integer L-shaped cuts from Laporte and Louveaux (1993) (DD&ILS) and the unified branch-and-Benders-cut (UB&BC) algorithm from Mahéo et al. (2024) (UB&DD). Ultimately, the algorithm yields a finite and exact double decomposition methodology for the two-stage stochastic integer optimization problem.

Our third contribution is to demonstrate the scalability of our methodology to large and practical MiND instances (Section 5). We develop a real-world setup in Manhattan using data from the NYC Taxi & Limousine Commission (2021). Results show the benefits of our subpath-based formulation and our double decomposition methodology. Specifically, the DD methodology yields certifiably optimal, or near-optimal solutions to the problem, and the broader DD&ILS and UB&DD algorithms converge to an exact solution. Altogether, our methodology scales to large and otherwise intractable instances of the MiND-VRP, of the size of the full Manhattan network with up to 100 candidate lines, hundreds of stations, thousands of passenger requests, and 5-20 demand scenarios; and it scales to instances with 10–40 candidate lines, over a hundred stations, and thousands of passenger requests for the MiND-DAR and MiND-Tr extensions. Our results also show the practical benefits of our stochastic optimization methodology, with a value of the stochastic solution of 5–7% against a deterministic benchmark. Ultimately, this paper contributes the first integrated methodology to optimize microtransit design and operations under uncertainty, and a methodology that scales to much larger instances than previous approaches in microtransit operations.

Our final contribution is to derive evidence that deviated fixed-route microtransit can provide win-win mobility outcomes (Section 6). As compared to ride-sharing, microtransit consolidates demand into high-capacity vehicles along reference lines. As compared to fixed-route transit, it increases demand coverage and improve passenger level of service by leveraging on-demand flexibility. In turn, the optimized microtransit network has a broader catchment area than its fixed-route counterpart, especially in otherwise unserved regions. Finally, demand consolidation and high coverage result in a significant decrease in distance traveled per passenger, with cost benefits and environmental benefits. Since Manhattan represents a high-density region, these results can be seen as conservative estimates of the impact of microtransit in lower-density areas with fewer transit alternatives. Altogether, deviated fixed-route microtransit can contribute to efficient mobility (high demand coverage, low costs per passenger, high service levels), equitable mobility (broad geographic reach), and sustainable mobility (limited environmental footprint). These results have inspired ongoing collaborations toward the pilot deployment of new microtransit solutions.

2. Background, motivation and literature review

Practical. Microtransit seeks to combine the efficiencies of public transit with the flexibility of ride-sharing. Our first-stage problem relates to transit planning (Desaulniers and Hickman 2007, Ortega et al. 2018, Wei et al. 2022, Sun et al. 2023). These problems have been solved with heuristics (Ceder and Wilson 1986, Walteros et al. 2015) and exact methods in small instances with 10-25 stops (Marín and Jaramillo 2009). Bertsimas et al. (2021) developed a column generation methodology for transit network design that scales to large instances with hundreds of stops.

In microtransit, one possible operating model is to design a joint system combining fixed-route transit and ride-sharing, which Chopra et al. (2023) framed via dual sourcing. Another model is to provide on-demand transportation with high-capacity vehicles, which Alonso-Mora et al. (2017) optimized in a request-trip-vehicle network. However, on-demand high-capacity operations may induce detours and delays. Blanchard et al. (2023) showed that the optimal latency in the traveling repairman problem grows with the size and dispersion of the geographic area, and grows at a supra-linear rate of $\Theta(n\sqrt{n})$ where n is the number of customers. This convex function reflects negative spatial externalities across customers induced by on-demand operations with high-capacity vehicles—that is, on-demand deviations become more costly with more passengers onboard.

This theoretical result outlines two approaches to alleviate spatiotemporal externalities in on-demand mobility: restricting vehicle occupancy—as in ride-sharing—or operating in small or concentrated areas. In practice, high-capacity microtransit has been successful in small municipalities[1] and university campuses.[2] In larger regions, zone-based microtransit operates in limited geographic locales; for instance, MetroConnect operates in 12 areas of Miami, and Metro Micro operates in eight areas of Los Angeles. It also acts as a first- and last-mile feeder into fixed-route transit (Steiner and Irnich 2020, Banerjee et al. 2021, Silva et al. 2022, Guan et al. 2023, Cummings et al.

[1] See, e.g., city.ridewithvia.com/salem-skipper, city.ridewithvia.com/newmo-newton

[2] See, e.g., ridewithvia.com/news/northeastern-university-taps-via-to-power-new-on-demand-safety-shuttle

2023).[3] In practice, multimodal microtransit introduces complexity to establish first- and last-mile zones; for instance, DART's GoLink service in Dallas partitions the service region into 34 zones. Such partitioning raises similar trade-offs as in door-to-door microtransit, between high costs with low-occupancy vehicles in small zones vs. detours and delays with larger vehicles in larger zones.

Deviated fixed-route microtransit, in contrast, consolidates demand into high-occupancy vehicles along transit routes while allowing on-demand deviations in response to passenger requests.[4] This model leverages *virtual bus stops* to consolidate pickups and dropoffs in central locations. On-demand operations with virtual bus stops induce challenging routing problems (Zhang et al. 2023). Viewed through this lens, deviated fixed-route microtransit leverages transit lines as a natural regularization, while allowing on-demand deviations with virtual bus stops.

Deviated fixed-route microtransit has been subject to limited research. Quadrifoglio et al. (2007, 2008) optimized on-demand deviations with a single vehicle. Quadrifoglio et al. (2006) and Zhao and Dessouky (2008) quantified trade-offs between frequencies, deviations, and service levels. Galarza Montenegro et al. (2022) optimized operations in a related system in which transit vehicles can skip stops. Liu et al. (2021) formulated a mixed-integer linear optimization model to optimize on-demand deviations with autonomous vehicles. All these methods focus on the operations alone (our second-stage problem), and scale to small instances with 1-5 vehicles and 10-50 stops.

Stochastic programming. Our problem combines a network design structure and a capacitated vehicle routing structure with time windows, under uncertainty. It is cast as a two-stage stochastic program with discrete recourse, a challenging class of problems (Carøe and Schultz 1999, Sen and Sherali 2006, Gade et al. 2014, Zhang and Kucukyavuz 2014, Kim and Mehrotra 2015, Bodur et al. 2017, Wang and Jacquillat 2020). Maheo et al. (2019) developed a unified branch-and-Benders-cut (UB&BC) algorithm as a general-purpose single-tree Benders decomposition algorithm with tailored branching rules, for stochastic mixed-integer programming with any mixed-integer structure in the first stage and in the second stage, and uncertainty in any parameters. In this paper, we leverage a network-based reformulation to retain a tight second-stage representation of routing operations. This approach applies extended formulations principles (Conforti et al. 2010, 2014), which have been used in lot sizing (Eppen and Martin 1987, Ahuja and Hochbaum 2008), machine scheduling (Sousa and Wolsey 1992, Pessoa et al. 2010), bin packing (Valério de Carvalho 1999, Delorme and Iori 2020), network design (Frangioni and Gendron 2009), unit commitment (Queyranne and Wolsey 2017), etc. In vehicle routing, a common approach involves modeling routing problems with temporal coordination requirements in time-expanded networks (Crainic et al. 2016, Lee et al.

[3] city.ridewithvia.com/go-connect-miami, micro.metro.net, www.dart.org/guide/transit-and-use

[4] www.nationalrtap.org/Toolkits/ADA-Toolkit/Service-Type-Requirements/Route-Deviation-Requirements; https://www.rideuta.com/Services/Flex-Routes; https://www.tricountytransit.org/flex-routes.html

2020, Agarwal and Ergun 2008, Liebchen 2008). In our setting, this representation leads to a very large formulation in a granular time-space-load network (Section 3.4).

Instead, we develop a subpath-based formulation of microtransit operations in a load-expanded network. This approach avoids time expansion by capturing time window requirements within subpaths, and enables a coarse spatial discretization between checkpoints. Macedo et al. (2011) proposed a pseudo-polynomial network flow model based on timed routes; Vazifeh et al. (2018), Bertsimas et al. (2019) relied on vehicle-sharing networks for fleet sizing and ride-sharing; in pickup-and-delivery and ride-pooling, Alyasiry et al. (2019), Zhang et al. (2023) defined subpaths from a point where the vehicle is empty to another; Rist and Forbes (2021), Hasan and Van Hentenryck (2021) considered subpaths consisting of a sequence of consecutive pickups and dropoffs. Schulz and Pfeiffer (2024) developed a fixed-path procedure to accelerate branch-and-cut algorithms for routing problems with precedence constraints, using a compact formulation. All these models focus on single-stage routing optimization. Our paper contributes a new load-expanded subpath-based network representation of microtransit operations; and it embeds it into a two-stage stochastic optimization framework to jointly optimize microtransit design and operations.

The problem is formulated as a two-stage stochastic integer program with an exponential number of second-stage variables, which we solve via Benders decomposition and column generation. Column generation has been applied to network-based extended formulations (Sadykov and Vanderbeck 2013, Delorme and Iori 2020, Hasan and Van Hentenryck 2021, Jacquillat et al. 2022); our methodology embeds it into a Benders decomposition scheme to solve the two-stage stochastic optimization problem. Combinations of column generation and Benders decomposition encompass simultaneous column-and-row generation (Muter et al. 2013), as well as path-based column generation in the Benders master problem or subproblem (Karsten et al. 2018, Zeighami and Soumis 2019, Wu et al. 2022). In contrast, our algorithm adds subpath-based variables to the Benders subproblem, which gives rise to a novel double-decomposition structure (Section 4).

In summary, our methodology contributes: (i) a subpath-based extended formulation of microtransit operations; (ii) an integrated two-stage stochastic optimization formulation with a tight subpath-based representation of second-stage vehicle routing; and (iii) a double-decomposition solution algorithm combining Benders decomposition and subpath-based column generation.

3. Microtransit Network Design (MiND) Model

The MiND optimizes the design and operations of a deviated fixed-route microtransit system, under demand uncertainty. The first-stage problem defines reference lines and service schedules (Section 3.1). The second-stage problem defines on-demand deviations in response to passenger requests, using a subpath-based representation in a load-expanded network (Section 3.2). We formulate the MiND-VRP in Section 3.3 and discuss its two-stage stochastic discrete optimization structure in Section 3.4. The extensions to MiND-DAR and Mind-Tr are in EC.4.1 and EC.5.1.

3.1. First-stage Problem: Network Design and Frequency Planning

The first-stage problem defines *reference trips*, each characterized by a reference line and a departure time. Each reference line is defined as an ordered set of checkpoints, and each reference trip determines the scheduled time at each checkpoint. Vehicles are required to visit some checkpoints at the scheduled times, but will also be allowed to visit other locations in-between (Section 3.2).

Operations occur over a roadway network. Let $\mathcal{N}^S$ denote the set of stations, including all candidate checkpoints and possible stopping locations. We represent demand as a set of passenger requests $\mathcal{P}$. In the MiND-VRP, each request $p \in \mathcal{P}$ is characterized by an origin $o(p) \in \mathcal{N}^S$ and a requested drop-off time t_p^{req}. Demand uncertainty is modeled via a set of scenarios $\mathcal{S}$; each scenario $s \in \mathcal{S}$ has probability π_s and comprises D_{ps} passengers from request $p \in \mathcal{P}$.

Network design. We pre-process candidate reference lines in a set $\mathcal{L}$. Let h_ℓ denote the cost to operate one trip of line $\ell \in \mathcal{L}$. Let $\mathcal{T}_\ell$ store time periods when a vehicle can depart from the first checkpoint in line $\ell \in \mathcal{L}$. We introduce the following decision variables to define reference trips:

$$x_{\ell t} = \begin{cases} 1 & \text{reference trip } (\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell \text{ is selected,} \\ 0 & \text{otherwise.} \end{cases}$$

Let $\mathcal{I}_\ell \subseteq \mathcal{N}^S$ index the checkpoints in reference line ℓ, of cardinality $I_\ell = |\mathcal{I}_\ell|$. Let $\mathcal{I}_\ell^{(i)}$ refer to the i^{th} checkpoint in the line, for $i \in \{1, \cdots, I_l\}$. All reference lines share the same final checkpoint $\mathcal{I}^{\text{end}} = \mathcal{I}_\ell^{(I_\ell)}$. Reference trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$ is scheduled in checkpoint $\mathcal{I}_\ell^{(i)}$ at time $T_{\ell t}(\mathcal{I}_\ell^{(i)})$.

We impose a fleet budget constraint by limiting the number of active trips at any time t:

$$\sum_{\ell \in \mathcal{L}} \sum_{t' \in \mathcal{T}_\ell : t' \leq t \leq t' + T_{\ell t}\left(\mathcal{I}^{\text{end}}\right) - T_{\ell t}\left(\mathcal{I}_\ell^{(1)}\right)} x_{\ell t} \leq F, \qquad \forall t \in \bigcup_{\ell \in \mathcal{L}} \mathcal{T}_\ell \tag{1}$$

Internal passenger assignments. Let $\mathcal{M}_p \subseteq \mathcal{L} \times \mathcal{T}_\ell$ denote the subset of reference trips that can serve request $p \in \mathcal{P}$ within a tolerance α of their requested drop-off time:

$$\mathcal{M}_p = \left\{ (\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell : \left| T_{\ell t}\left(\mathcal{I}^{\text{end}}\right) - t_p^{\text{req}} \right| \leq \alpha \right\}, \ \forall p \in \mathcal{P}$$

We define assignment variables to identify a candidate reference trip for each passenger:

$$z_{\ell pst} = \begin{cases} 1 & \text{if passenger request } p \in \mathcal{P} \text{ is internally assigned to trip } (\ell, t) \in \mathcal{M}_p \text{ in scenario } s \in \mathcal{S}, \\ 0 & \text{otherwise.} \end{cases}$$

We impose packing constraints so that each passenger is assigned to at most one reference trip.

$$\sum_{(\ell, t) \in \mathcal{M}_p} z_{\ell pst} \leq 1, \qquad \forall p \in \mathcal{P}, \forall s \in \mathcal{S} \tag{2}$$

The assignments $z_{\ell pst}$ link first-stage and second-stage decisions but are not executed in practice, since passenger service is optimized at the operational level. These variables are scenario-dependent but will be treated as first-stage variables in the algorithm; this choice enables separability across reference trips and leads to a slightly tighter second-stage relaxation. A similar methodology and similar computational results could be obtained by treating $z_{\ell pst}$ as second-stage variables.

Vehicle load. We assume that vehicles are homogeneous within each reference line $\ell \in \mathcal{L}$, with capacity C_ℓ. We impose a *target load factor* $\kappa \in (0,1)$ to induce high vehicle utilization. We also allow first-stage assignments to exceed vehicle capacities by a factor κ to create operating flexibility, but the second-stage passenger service decisions will strictly comply with vehicle capacities.

$$\sum_{p \in \mathcal{P}\,:\,(\ell,t)\in\mathcal{M}_p} D_{ps} z_{\ell pst} \geq (1-\kappa) C_\ell x_{\ell t} \qquad \forall (\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell, \forall s \in \mathcal{S} \tag{3}$$

$$\sum_{p \in \mathcal{P}\,:\,(\ell,t)\in\mathcal{M}_p} D_{ps} z_{\ell pst} \leq (1+\kappa) C_\ell x_{\ell t} \qquad \forall (\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell, \forall s \in \mathcal{S} \tag{4}$$

3.2. Second-stage Problem: On-demand Deviations

Second-stage deviations must stay within a distance Δ of the reference line and must respect scheduled times at the checkpoints. The reference schedule includes buffers between checkpoints to allow for deviations. Moreover, vehicles may skip up to K checkpoints in a row: $K=0$ induces closer adherence to the reference trip, whereas $K \geq 1$ provides more flexibility. We denote by Γ_ℓ the checkpoint pairs separated by up to K checkpoints on line $\ell \in \mathcal{L}$.

The second-stage problem involves capacitated vehicle routing with time windows for each reference trip and in each scenario. We formulate it in a load-expanded network with subpath variables characterizing on-demand deviations between checkpoints, leveraging the structure of microtransit.

Subpaths. For reference trip $(\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell$ and demand scenario $s \in \mathcal{S}$, a subpath $r \in \mathcal{R}_{\ell st}$ is identified by its starting checkpoint $u_r \in \mathcal{I}_\ell$, its ending checkpoint $v_r \in \mathcal{I}_\ell$, and the passenger requests $\mathcal{P}_r \subseteq \mathcal{P}$ served in between. The set $\mathcal{R}_{\ell st}$ includes all subpaths such that the distance to the reference line never exceeds Δ; the load satisfies $\sum_{p\in\mathcal{P}_r} D_{ps} \leq C_\ell$; the travel time does not exceed $T_{\ell t}(v_r) - T_{\ell t}(u_r)$; and up to K checkpoints are skipped. The second-stage problem selects a sequence of subpaths that (i) starts at the origin of the reference line and ends at its destination while maintaining flow balance; and (ii) serves up to C_ℓ passengers overall.

Load-expanded subpath network. We represent routing operations in a load-expanded network (Figure 1). Each node tracks the checkpoint and the vehicle load, and each arc encapsulates a subpath. Flow balance constraints capture physical flows and vehicle capacity constraints.

Let $\mathcal{C}_\ell = \{0, 1, \cdots, C_\ell\}$ store vehicle loads. For reference trip $(\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell$ and scenario $s \in \mathcal{S}$, we denote the load-expanded network by $(\mathcal{V}_{\ell st}, \mathcal{A}_{\ell st})$. Node $v_{\ell st}$ represents the end of a trip. Each other

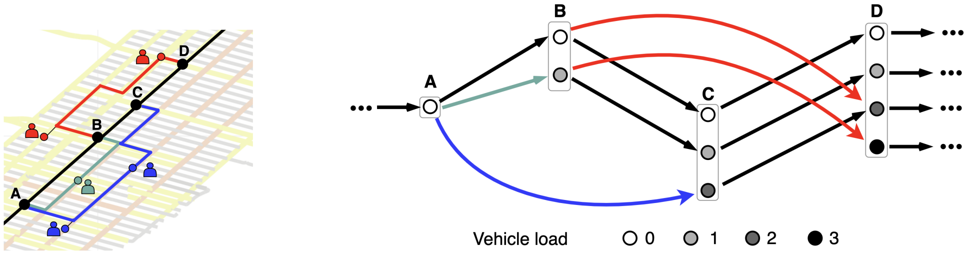


Figure 1 *Left:* **Physical network with three candidate deviations.** *Right:* **Load-expanded subpath network.**

node $n \in \mathcal{V}_{\ell st}$ corresponds to a tuple (k_n, c_n) consisting of checkpoint $k_n \in \mathcal{I}_\ell$ and load $c_n \in \mathcal{C}_\ell$; node $u_{\ell st}$ encodes the start at stop $\mathcal{I}_\ell^{(1)}$ and time $T_{\ell t}(\mathcal{I}_\ell^{(1)})$. Each arc $a \in \mathcal{A}_{\ell st}$ connects nodes $start(a) \in \mathcal{V}_{\ell st}$ and $end(a) \in \mathcal{V}_{\ell st}$; vice versa, $r(a) \in \mathcal{R}_{\ell st}$ is the subpath corresponding to arc $a \in \bigcup_{r \in \mathcal{R}_{\ell st}} \mathcal{A}_r$. We partition $\mathcal{A}_{\ell st} = \bigcup_{r \in \mathcal{R}_{\ell st}} \mathcal{A}_r \cup \mathcal{A}^v_{\ell st}$ into traveling arcs $\mathcal{A}_r$ and terminating arcs $\mathcal{A}^v_{\ell st}$:

$$\mathcal{A}_r = \left\{ (n,m) \in \mathcal{V}_{\ell st} \times \mathcal{V}_{\ell st} \, : \, k_n = u_r, k_m = v_r, c_m - c_n = \sum_{p \in \mathcal{P}_r} D_{ps} \right\}, \quad \forall r \in \mathcal{R}_{\ell st}, \tag{5}$$

$$\mathcal{A}^v_{\ell st} = \{(n,m) \in \mathcal{V}_{\ell st} \times \mathcal{V}_{\ell st} \, : \, k_n = \mathcal{I}^{\text{end}}, m = v_{\ell st}\}. \tag{6}$$

Our second-stage decisions select subpaths in the load-expanded networks via the following variables. These define on-demand deviations and pickups for each reference trip and each scenario.

$$y_a = \begin{cases} 1 & \text{if arc } a \in \mathcal{A}_{\ell st} \text{ is selected, for } (\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell, \;\; s \in \mathcal{S}, \\ 0 & \text{otherwise.} \end{cases}$$

Passenger service. The second-stage formulation aims to maximize demand coverage and passenger level of service. Coverage is formalized via a large reward M incurred for each successful pickup. Level of service is formalized via the following objectives:

1. τ^{walk}_{rp}: walking time from passenger p's origin to the pickup location via subpath $r \in \mathcal{R}_{\ell st}$;
2. τ^{wait}_{rp}: waiting time of passenger p prior to pickup via subpath $r \in \mathcal{R}_{\ell st}$;
3. $\frac{\tau^{travel}_{rp}}{\tau^{dir}_p}$: relative detour, defined as the in-vehicle travel time of passenger p via subpath $r \in \mathcal{R}_{\ell st}$ normalized with respect to the direct trip time (e.g., a taxi trip); and
4. $\frac{\tau^{late}_{\ell tp}}{\tau^{dir}_p}$, $\frac{\tau^{early}_{\ell tp}}{\tau^{dir}_p}$: relative delay and earliness of passenger p at the destination via trip $(\ell,t) \in \mathcal{M}_p$.

The model can be extended to incorporate additional practical considerations in arc cost parameters. For example, g_a could penalize subpaths that skip checkpoints to promote closer adherence to the reference line, even when $K > 0$ (although shorter subpaths arise naturally in our solutions).

In addition, we impose three restrictions to guarantee convenient service. First, passengers must not leave before their planned departure times $t^0_{\ell pt}$, corresponding to the time at which they would start walking to the nearest checkpoint without deviations. Second, they must not walk more than a limit Ω. Third, they must not wait more than a limit Ψ. Thus, pickup $p \in \mathcal{P}$ is only acceptable in location $i \in \mathcal{N}^S$ at time $\bar{t}$ via subpath $r \in \mathcal{R}_{\ell st}$ if $t^0_{\ell pt} + \tau^{walk}_{rp} \leq \bar{t} \leq t^0_{lpt} + \tau^{walk}_{rp} + \Psi$ and $\tau^{walk}_{rp} \leq \Omega$.

We define non-negative hyperparameters λ, μ, σ, and δ to weigh the level of service cost components. The arc costs in the load-expanded network are defined as follows for all $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell,\ s\in\mathcal{S}$.

$$g_a=\begin{cases}\sum_{p\in\mathcal{P}_{r(a)}} D_{ps}\left(\lambda\tau^{walk}_{r(a)p}+\mu\tau^{wait}_{r(a)p}+\sigma\frac{\tau^{travel}_{r(a)p}}{\tau^{dir}_p}+\delta\frac{\tau^{late}_{\ell tp}}{\tau^{dir}_p}+\frac{\delta}{2}\frac{\tau^{early}_{\ell tp}}{\tau^{dir}_p}-M\right) & \forall a\in\bigcup_{r\in\mathcal{R}_{\ell st}}\mathcal{A}_r,\\ 0 & \forall a\in\mathcal{A}^v_{\ell st}.\end{cases} \tag{7}$$

3.3. Two-stage Stochastic Optimization Formulation (MiND-VRP)

The MiND-VRP minimizes planning costs, maximizes demand coverage, and maximizes level of service (Equation (8)). The constraints apply fleet size, target load factors, and packing constraints (Equations (1)–(4)); enforce flow balance over load-expanded networks (Constraint (9)); and link first-stage assignments to second-stage operations (Constraint (10)). Notation is summarized in EC.1.1. The MiND-DAR and MiND-Tr are formulated similarly, albeit with extra constraints to ensure consistency between pickups, transfers and dropoffs.

$$\min \quad \sum_{\ell\in\mathcal{L}}\sum_{t\in\mathcal{T}_\ell}\left(h_\ell x_{\ell t}+\sum_{s\in\mathcal{S}}\pi_s\sum_{a\in\mathcal{A}_{\ell st}} g_a y_a\right) \tag{8}$$

s.t. First-stage constraints: Equations (1)–(4)

$$\sum_{m:(n,m)\in\mathcal{A}_{\ell st}} y_{(n,m)}-\sum_{m:(m,n)\in\mathcal{A}_{\ell st}} y_{(m,n)}=\begin{cases}x_{\ell t} & \text{if } n=u_{\ell st}\\ -x_{\ell t} & \text{if } n=v_{\ell st}\\ 0 & \text{otherwise}\end{cases} \quad \forall\ell\in\mathcal{L},t\in\mathcal{T}_\ell,s\in\mathcal{S},n\in\mathcal{V}_{\ell st} \tag{9}$$

$$\sum_{a\in\mathcal{A}_{\ell st}:p\in\mathcal{P}_{r(a)}} y_a\le z_{\ell pst} \quad \forall s\in\mathcal{S},p\in\mathcal{P},(\ell,t)\in\mathcal{M}_p \tag{10}$$

$$\boldsymbol{x},\boldsymbol{y},\boldsymbol{z} \text{ binary} \tag{11}$$

3.4. General structure, and comparison to benchmarks

We provide a general-purpose description of our two-stage stochastic discrete optimization setting to compare our subpath-based network model to alternative approaches based on compact formulations or other network-extended formulations. This description will also be used in Section 4 to develop our double decomposition algorithm in a general-purpose environment.

Compact formulation. Let $\boldsymbol{x}\in\mathcal{X}^{\text{MIO}}$ and $\boldsymbol{y}^C_s\in\mathcal{Y}^{\text{MIO-C}}_s$ denote first-stage and second-stage variables, respectively. We use $\mathcal{X}^{\text{MIO}}$ and $\mathcal{Y}^{\text{MIO-C}}_s$ to represent mixed-integer optimization regions; for instance, in a setting with n_Z integer variables and n_R continuous variables, they encode the mixed-integer set $\mathbb{Z}^{n_Z}\times\mathbb{R}^{n_R}$. The optimization problem is formulated as follows, where $\mathcal{K}$ stores indices of the first-stage decisions (e.g., line-time pairs in the MiND), $\boldsymbol{A}\boldsymbol{x}\ge\boldsymbol{b}$ encode all first-stage constraints (e.g., fleet budget), and $\boldsymbol{D}_s\boldsymbol{x}+\boldsymbol{E}_s\boldsymbol{y}^C_s\ge\widetilde{\boldsymbol{h}}_s$ encode all second-stage and linking constraints (e.g., checkpoint-to-checkpoint operations for selected microtransit lines).

$$\min \quad \sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}_j} c_k x_k+\sum_{s\in\mathcal{S}}\pi_s\widetilde{\boldsymbol{g}}^\top_s\boldsymbol{y}^C_s \tag{C}$$

$$\begin{aligned}\text{s.t.}\quad & \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b} \\ & \boldsymbol{D}_s\boldsymbol{x} + \boldsymbol{E}_s\boldsymbol{y}_s^C \geq \widetilde{\boldsymbol{h}}_s, \forall s \in \mathcal{S} \\ & \boldsymbol{x} \in \mathcal{X}^{\text{MIO}}; \quad \boldsymbol{y}_s^C \in \mathcal{Y}_s^{\text{MIO-C}},\ \forall s \in \mathcal{S}\end{aligned}$$

Extended reformulations. Our subpath-based model lifts the second-stage problem in a higher-dimensional space. Let $\mathcal{K}_j$, for $j \in \mathcal{J}$, denote a partition of the first-stage indices such that the second-stage problem is independent across subsets, with block-diagonal matrices $\boldsymbol{D}_s$ across $\mathcal{K}_j$. In the MiND, this corresponds to the partition across line-time pairs (ℓ, t). We then derive a network representation induced by scenario s and subset $\mathcal{K}_j$, for $j \in \mathcal{J}$, with node set $\mathcal{N}_{sj}$ and arc set $\mathcal{A}_{sj}$. We denote the new second-stage variables by $y_a \in \mathcal{Y}_a^{\text{MIO}}$ for each arc $a \in \mathcal{A}_{sj}$ in each scenario $s \in \mathcal{S}$ and for each partition element $j \in \mathcal{J}$; again, the sets $\mathcal{Y}_a^{\text{MIO}}$ encode the discrete requirements of variables y_a. The formulation encompass flow balance constraints in the network $(\mathcal{N}_{sj}, \mathcal{A}_{sj})$ (e.g., combining subpaths into paths, in Equation (9)) as well as general-purpose linear constraints $\sum_{a\in\mathcal{A}_{sj}} \boldsymbol{f}_{asj} y_a \geq \boldsymbol{h}_{sj}$ (e.g., demand constraints in Equation (10)). It is given in Problem ($\star$):

$$\begin{aligned}\min\quad & \sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}_j} c_k x_k + \sum_{s\in\mathcal{S}}\sum_{j\in\mathcal{J}} \pi_s \left(\sum_{a\in\mathcal{A}_{sj}} g_a y_a\right) \qquad (\star) \\ \text{s.t.}\quad & \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b} \\ & \sum_{m:(n,m)\in\mathcal{A}_{sj}} y_{(n,m)} - \sum_{m:(m,n)\in\mathcal{A}_{sj}} y_{(m,n)} = \sum_{k\in\mathcal{K}_j} b_{nsk} x_k,\ \forall s \in \mathcal{S},\ \forall j \in \mathcal{J},\ \forall n \in \mathcal{N}_{sj} \\ & \sum_{a\in\mathcal{A}_{sj}} \boldsymbol{f}_{asj} y_a \geq \boldsymbol{h}_{sj},\ \forall s \in \mathcal{S},\ \forall j \in \mathcal{J} \\ & \boldsymbol{x} \in \mathcal{X}^{\text{MIO}}; \quad y_a \in \mathcal{Y}_a^{\text{MIO}},\ \forall a \in \mathcal{A}_{sj},\ \forall s \in \mathcal{S}\ \forall j \in \mathcal{J}\end{aligned}$$

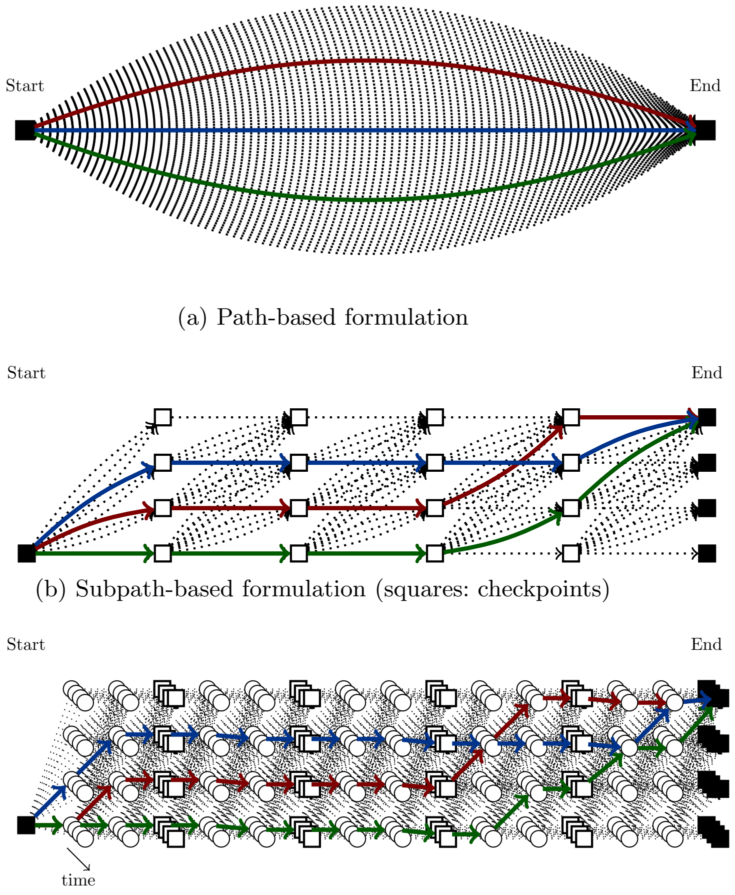
Discussion. Proposition 1 shows that the extended network-based formulation defines the same optimization problem as the compact formulation in the MiND. Moreover, it defines a tighter second-stage linear relaxation by embedding time windows into the network representation itself and capturing vehicle capacities via flow balance constraints, as compared to relying on explicit big-M constraints. This reformulation, however, involves more subpath-based variables.

Proposition 1. *The subpath-based second-stage formulation defines an equivalent mixed-integer model as the compact formulation in the MiND, with a tighter linear relaxation.*

This discussion invites comparisons to other possible network-based representations of routing operations in our second-stage problem, detailed in EC.1 and illustrated in Figure 2:

- A *segment-based* model with nodes encoding time-station-load tuples and arcs encoding segments between consecutive stops. This network is much more granular along the time dimension (each station in Figure 2c, versus each checkpoint in Figure 2b) and is also expanded in a granular temporal discretization (to capture passengers' and vehicles' time windows). Moreover, the model is further complicated by two multi-commodity flow structures with linking constraints from checkpoint to checkpoint (so the vehicle does not skip more than K checkpoints in a row) and from station to station (to maintain continuity in time and space).

- A *path-based* model in a simple origin-destination network. with arcs encoding full paths that start at the line's origin, end at its destination, and serve at most C_ℓ passengers.



(a) Path-based formulation

(b) Subpath-based formulation (squares: checkpoints)

(c) Segment-based formulation (squares: checkpoints; circles: other stations)

Figure 2 Visualization of the path-based, subpath-based and segment-based formulations.

Proposition 2 shows that the three formulations are equivalent, as long as time discretization is sufficiently granular in the segment-based benchmark (we formalize this condition in EC.1.6). The segment-based benchmark induces a weaker relaxation due to the double flow structure with linking constraints. In constrast, the subpath-based formulation achieves an equally strong relaxation as the path-based benchmark thanks to the flow balance structure on the load-expanded network. Most importantly, Proposition 3 shows the size benefits of the subpath-based formulation. The segment-based formulation features a polynomial number of variables in a dense time-station-load network

(Figure 2c). In the path-based formulation, the number of variables scales exponentially with the total number of stations along the reference line (Figure 2a). In the subpath-based formulation, the number of variables scales exponentially with the number of stations between checkpoints, and relies on a coarser checkpoint-load network representation (Figure 2b). The proposition also shows that the model becomes increasingly complex as more checkpoints can be skipped (higher K).

Proposition 2. *The path-based and subpath-based formulations are equivalent and define identical linear relaxations. If all subpath travel times are strictly less than the elapsed time between the scheduled arrival times at the checkpoints, there exists a time discretization such that the segment-based formulation is also equivalent but its linear relaxation is at most as strong.*

Proposition 3. *Consider the second-stage problem for reference trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$ in scenario $s \in \mathcal{S}$. Let Ξ be the maximum number of stations between any pair of checkpoints in Γ_ℓ. The segment-based formulation has $\mathcal{O}(T_S \cdot C_\ell^2 \cdot I_\ell \cdot \Xi^2)$ variables and $\mathcal{O}(|\mathcal{P}| + T_S \cdot C_\ell \cdot |\mathcal{N}^S| + T_S \cdot C_\ell^2 \cdot I_\ell \cdot \Xi^2)$ constraints. The subpath-based formulation has $\mathcal{O}(I_\ell \cdot C_\ell \cdot 2^\Xi)$ variables and $\mathcal{O}(|\mathcal{P}| + C_\ell \cdot I_\ell)$ constraints. The path-based formulation has $\mathcal{O}(2^{\Xi \cdot I_\ell})$ variables and $\mathcal{O}(|\mathcal{P}|)$ constraints.*

4. Double-Decomposition (DD) Algorithm

The MiND-VRP exhibits a two-stage stochastic optimization structure with a tight recourse function and exponentially many second-stage variables. We propose a double decomposition algorithm to solve its partial relaxation with discrete first-stage variables and continuous second-stage variables. The methodology relies on Benders decomposition to exploit the nested block-angular structure (Section 4.1), and on subpath-based column generation in the Benders subproblem (Section 4.2). We formalize the algorithm and establish its exactness in Section 4.3, and augment it with integer L-shaped cuts and UB&BC to retrieve an exact algorithm for Problem ($\star$) in Section 4.4. For generalizability, we describe the algorithm using the general-purpose notation from Problem ($\star$); we develop it for the MiND-VRP in EC.2.2, for the MiND-DAR in EC.4.2, and for the MiND-Tr in EC.5.2. Section 4.5 describes the label-setting algorithm for the pricing problem.

4.1. Multi-cut Benders Decomposition

Let `OPT` denote the optimal value of ($\star$). We denote the second-stage problem for a first-stage solution $\boldsymbol{x}$ by $\texttt{SP}(\boldsymbol{x})$, and its optimal solution by $\varPhi(\boldsymbol{x})$. Due to its nested block-angular structure, $\texttt{SP}(\boldsymbol{x})$ is decomposable across $s \in \mathcal{S}$ and $j \in \mathcal{J}$. Namely, $\varPhi(\boldsymbol{x}) = \sum_{s\in\mathcal{S}} \sum_{j\in\mathcal{J}} \pi_s \varphi_{sj}(\boldsymbol{x})$ where:

$$\varphi_{sj}(\boldsymbol{x}) = \min \quad \sum_{a\in\mathcal{A}_{sj}} g_a y_a \tag{12}$$

$$\text{s.t.} \quad \sum_{m:(n,m)\in\mathcal{A}_{sj}} y_{(n,m)} - \sum_{m:(m,n)\in\mathcal{A}_{sj}} y_{(m,n)} = \sum_{k\in\mathcal{K}_j} b_{nsk} x_k, \ \forall n \in \mathcal{N}_{sj} \tag{13}$$

$$\sum_{a\in\mathcal{A}_{sj}} \boldsymbol{f}_{asj} y_a \geq \boldsymbol{h}_{sj} \tag{14}$$

$$y_a \in \mathcal{Y}_a^{\mathrm{MIO}},\ \forall a \in \mathcal{A}_{sj} \tag{15}$$

We define and evaluate the total cost associated with each first-stage solution as follows:

$$\mathtt{OPT}(\boldsymbol{x}) = \boldsymbol{c}^\top \boldsymbol{x} + \Phi(\boldsymbol{x})$$

We refer to the partial relaxation as $(\mathtt{MIO}-\mathtt{LO})$. We define the Benders subproblem (BSP) as its second-stage relaxation $\overline{\mathtt{SP}}(\boldsymbol{x})$, with optimal value $\overline{\Phi}(\boldsymbol{x}) = \sum_{s\in\mathcal{S}}\sum_{j\in\mathcal{J}} \pi_s \overline{\varphi}_{sj}(\boldsymbol{x})$ where:

$$\overline{\varphi}_{sj}(\boldsymbol{x}) = \min \quad \left\{ \sum_{a\in\mathcal{A}_{sj}} g_a y_a \ :\ \text{Equations (13)–(14)},\ \boldsymbol{y} \geq \boldsymbol{0} \right\}$$

Its dual formulation in scenario $s\in\mathcal{S}$ and for partition element $j\in\mathcal{J}$ is given as follows, using $\boldsymbol{\psi}$ and $\boldsymbol{\gamma}$ to denote the dual variables of the flow balance constraints and the side constraints.

$$\max \quad \left\{ \sum_{n\in\mathcal{N}_{sj}} \sum_{k\in\mathcal{K}_j} b_{nsk} x_k \psi_{sjn} + \boldsymbol{h}_{sj}^\top \boldsymbol{\gamma}_{sj} : (\boldsymbol{\psi}_{sj}, \boldsymbol{\gamma}_{sj}) \in \mathcal{P}_{sj} \right\}, \quad \text{where:} \tag{16}$$

$$\mathcal{P}_{sj} = \left\{ (\boldsymbol{\psi}_{sj}, \boldsymbol{\gamma}_{sj})\ :\ \boldsymbol{\gamma}_{sj} \geq \boldsymbol{0};\ \psi_{sjm} - \psi_{sjn} + \boldsymbol{\gamma}_{sj}^\top \boldsymbol{f}_{(m,n),s,j} \leq g_{(m,n)},\ \forall (m,n) \in \mathcal{A}_{sj} \right\}$$

We index the extreme points of the dual polyhedron $\mathcal{P}_{sj}$ by $\{(\boldsymbol{\psi}_{sj}^u, \boldsymbol{\gamma}_{sj}^u) : u \in \mathcal{U}_{sj}\}$ and its extreme rays by $\{(\boldsymbol{\psi}_{sj}^v, \boldsymbol{\gamma}_{sj}^v) : v \in \mathcal{V}_{sj}\}$. The Benders master problem, denoted by $\mathtt{MP}(\mathcal{U}^0, \mathcal{V}^0)$, is given by

$$\min \quad \sum_{j\in\mathcal{J}} \sum_{k\in\mathcal{K}_j} c_k x_k + \sum_{s\in\mathcal{S}} \sum_{j\in\mathcal{J}} \pi_s \theta_{sj} \tag{$\mathtt{MP}(\mathcal{U}^0,\mathcal{V}^0)$}$$

$$\text{s.t.} \quad \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}$$

$$\theta_{sj} \geq \sum_{n\in\mathcal{N}_{sj}} \sum_{k\in\mathcal{K}_j} b_{nsk} x_k \psi_{sjn}^u + \boldsymbol{h}_{sj}^\top \boldsymbol{\gamma}_{sj}^u,\ \forall s\in\mathcal{S},\ \forall j\in\mathcal{J},\ \forall u \in \mathcal{U}_{sj}^0 \tag{17}$$

$$0 \geq \sum_{n\in\mathcal{N}_{sj}} \sum_{k\in\mathcal{K}_j} b_{nsk} x_k \psi_{sjn}^v + \boldsymbol{h}_{sj}^\top \boldsymbol{\gamma}_{sj}^v,\ \forall s\in\mathcal{S},\ \forall j\in\mathcal{J},\ \forall v \in \mathcal{V}_{sj}^0 \tag{18}$$

$$\boldsymbol{x} \in \mathcal{X}^{\mathrm{MIO}}$$

The Benders master problem solves a relaxation $\mathtt{MP}(\mathcal{U}^0, \mathcal{V}^0)$ containing a subset of constraints indexed by $\mathcal{U}_{sj}^0 \subseteq \mathcal{U}_{sj}$ and $\mathcal{V}_{sj}^0 \subseteq \mathcal{V}_{sj}$. By design, it yields a lower bound of $(\mathtt{MIO}-\mathtt{LO})$ and its combination with the $\overline{\mathtt{SP}}(\boldsymbol{x})$ yield an upper bound of $(\mathtt{MIO}-\mathtt{LO})$. If the gap lies within a given tolerance, Benders decomposition stops. Otherwise, we add optimality cuts by augmenting $\mathcal{U}_{sj}^0$ with an optimal extreme point $(\boldsymbol{\psi}_{sj}^u, \boldsymbol{\gamma}_{sj}^u)$ for all scenario-partition combinations $s\in\mathcal{S},\ j\in\mathcal{J}$ such that the Benders subproblem admits an optimal solution, and by augmenting $\mathcal{V}_{sj}^0$ with an extreme ray defining a direction of unboundeness $(\boldsymbol{\psi}_{sj}^v, \boldsymbol{\gamma}_{sj}^v)$ for all others.

4.2. Subpath-based Column Generation for Benders Subproblem

The main challenge with the Benders decomposition scheme lies in the large number of second-stage variables y_a in the expanded network representation. In the MiND, the arc set $\mathcal{A}_{\ell st}$ grows exponentially with the number of stations between checkpoints. To prevent the subproblem $\texttt{SP}(\boldsymbol{x})$ from becoming too computationally intensive at each iteration, we propose a column generation procedure iterating between a restricted Benders subproblem and a pricing problem.

The restricted Benders subproblem (RBSP) is formulated as $\overline{\texttt{SP}}(\boldsymbol{x})$, except that the arc sets $\mathcal{A}_{sj}$ are replaced by restricted arc sets $\mathcal{A}'_{sj} \subseteq \mathcal{A}_{sj}$. We refer to it as $\texttt{RSP}(\boldsymbol{x}, \mathcal{A}'_{sj})$ and to its optimal value as $\Phi'(\boldsymbol{x}, \mathcal{A}'_{sj})$. Namely, $\Phi'(\boldsymbol{x}, \mathcal{A}'_{sj}) = \sum_{s\in\mathcal{S}} \sum_{j\in\mathcal{J}} \pi_s \varphi'_{s,j}(\boldsymbol{x}, \mathcal{A}'_{sj})$ where:

$$\begin{aligned}
\varphi'_{s,j}(\boldsymbol{x}, \mathcal{A}'_{sj}) = \min_{\boldsymbol{y}\geq \mathbf{0}} \quad & \sum_{a\in\mathcal{A}'_{sj}} g_a y_a \\
\text{s.t.} \quad & \sum_{m:(n,m)\in\mathcal{A}'_{sj}} y_{(n,m)} - \sum_{m:(m,n)\in\mathcal{A}'_{sj}} y_{(m,n)} = \sum_{k\in\mathcal{K}_j} b_{nsk} x_k, \ \forall n \in \mathcal{N}_{sj} \\
& \sum_{a\in\mathcal{A}'_{sj}} \boldsymbol{f}_{asj} y_a \geq \boldsymbol{h}_{sj}
\end{aligned}$$

The pricing problem, denoted $\texttt{PP}(\boldsymbol{\psi}, \boldsymbol{\gamma})$, serves as dual separation by seeking a variable of negative reduced cost or proving that none exists. It is given as follows for $s\in\mathcal{S}$ and $j\in\mathcal{J}$:

$$RC = \min \ \left\{ g_{(m,n)} - (\psi_{sjm} - \psi_{sjn}) - \boldsymbol{\gamma}_{sj}^\top \boldsymbol{f}_{(m,n),s,j} : (m,n) \in \mathcal{A}_{sj} \right\} \tag{19}$$

4.3. Combining Benders Decomposition and Subpath-based Column Generation

Our solution algorithm, summarized in Figure 3, involves two interconnected decomposition structures. In an outer Benders decomposition loop, the master problem generates a feasible first-stage solution and a lower bound; the Benders subproblem generates a second-stage fractional solution and an upper bound to the partial relaxation ($\texttt{MIO}-\texttt{LO}$). At each outer iteration, the algorithm adds optimality and feasibility cuts, or certifies the optimality of the ($\texttt{MIO}-\texttt{LO}$) solution. The inner loop solves the Benders subproblem via subpath-based column generation: the restricted Benders subproblem generates a feasible solution; the pricing problem identifies variables with negative reduced cost for each scenario $s\in\mathcal{S}$ and partition $j\in\mathcal{J}$, or yields a certificate of optimality in the restricted Benders subproblem. The algorithm continues until convergence in both loops.[5]

Theorem 1 shows the exactness of the DD algorithm toward solving ($\texttt{MIO}-\texttt{LO}$). Upon termination, we solve the second-stage problem with integrality constraints to derive a feasible solution to Problem ($\star$). Thus, the DD algorithm returns a valid optimality gap for Problem ($\star$).

[5] Figure 3 shows a multi-tree implementation of the DD algorithm, based on cutting-plane version of Benders decomposition; for completeness, we describe the single-tree implementation of the algorithm in Figure EC.2, based on lazy constraints within the branch-and-cut algorithm (Fortz and Poss 2009, Bodur and Luedtke 2017).

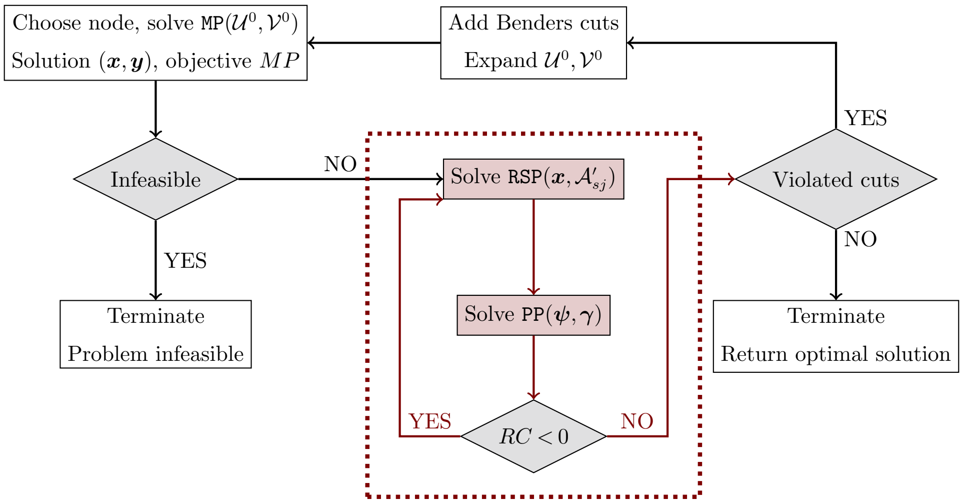

Choose node, solve $\texttt{MP}(\mathcal{U}^0, \mathcal{V}^0)$
Solution $(\boldsymbol{x}, \boldsymbol{y})$, objective MP
Add Benders cuts
Expand $\mathcal{U}^0, \mathcal{V}^0$
Infeasible
NO
YES
Terminate
Problem infeasible
Solve $\texttt{RSP}(\boldsymbol{x}, \mathcal{A}'_{sj})$
Solve $\texttt{PP}(\boldsymbol{\psi}, \boldsymbol{\gamma})$
$RC < 0$
YES
NO
Violated cuts
YES
NO
Terminate
Return optimal solution

Figure 3 **Multi-tree DD algorithm to solve Problem** $(\texttt{MIO} - \texttt{LO})$**.**

Theorem 1. *The double decomposition algorithm returns an optimal solution to Problem* ($\texttt{MIO}-\texttt{LO}$) *in a finite number of iterations along with an optimality gap for Problem* $(\star)$.

This integrated Benders decomposition and column generation algorithm gives rise to our double-decomposition structure, illustrated in Figure 4 for the MiND. The master problem solves a network design and service scheduling problem with all reference lines in all scenarios (Figure 4a). The Benders subproblem is then separable across scenarios $s \in \mathcal{S}$ and partition elements $j \in \mathcal{J}$ (i.e., across reference trips, in Figure 4b). Column generation exploits the network-based second-stage formulation to further decompose the Benders subproblem across arcs (i.e., across subpaths between checkpoints, in Figure 4c). The other way around, the pricing problem adds subpaths locally between checkpoints; the restricted Benders subproblem combines them into full paths to optimize operations along each reference trip in each scenario; and the Benders master problem brings all reference trips together to optimize first-stage network design and service scheduling decisions.

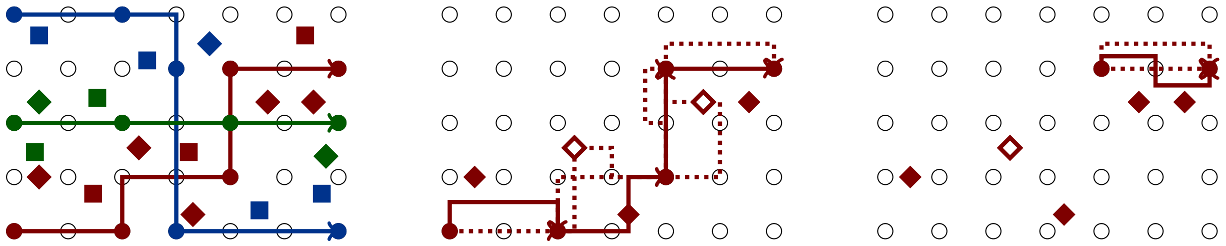

(a) Benders master problem (b) Benders subproblem (c) Pricing problem

Figure 4 **Double-decomposition algorithm.** *Left:*: **BMP with three reference lines (blue, red, green); passenger requests in two scenarios (squares, diamonds) with their first-stage assignments (colors).** *Middle:* **BSP for one reference trip and one scenario; full diamonds encode served passengers; solid lines characterize selected subpaths in RBSP.** *Right:* **PP to generate new subpath between checkpoints (solid line).**

This algorithm raises two final questions. First, the DD structure solves the partial relaxation ($\texttt{MIO}-\texttt{LO}$) to optimality, so it may still leave an optimality gap in the full problem ($\star$). As we shall see experimentally, the gap is very small in the MiND due to the tight second-stage formulation. Still, we augment the DD methodology toward an exact and finitely convergent algorithm for Problem ($\star$) in Section 4.4. Second, the scalability of the DD algorithm hinges on the efficiency of the pricing algorithm. In the MiND, we propose a label-setting algorithm that exploits an additional decomposition of the pricing problem into routing and load components, in Section 4.5.

4.4. Exact double-decomposition algorithms for Problem ($\star$).

Double decomposition with integer L-shaped cuts (DD&ILS). This approach assumes that the first-stage variables are binary, i.e., $\mathcal{X}^{\text{MIO}} = \{0,1\}^{nz}$, and that the problem has relatively complete recourse; both conditions are satisfied in the MiND. We adopt a multi-cut variant of the integer L-shaped method from Laporte and Louveaux (1993) by adding the following optimality cut to the Benders master problem, where $\underline{\boldsymbol{\Phi}}_{sj}$ denotes a global lower bound of the second-stage cost $\varphi_{sj}(\boldsymbol{x})$ and $\{\widehat{\boldsymbol{x}}^{w} : w \in \mathcal{W}^0\}$ indexes the (binary) first-stage variables visited through the algorithm.

$$\theta_{sj} \geq \underline{\boldsymbol{\Phi}}_{sj} + (\varphi_{sj}(\widehat{\boldsymbol{x}}^{w}) - \underline{\boldsymbol{\Phi}}_{sj}) \left(1 - \sum_{k \in \mathcal{K}_j : \widehat{x}_k^w = 1} (1 - x_k) - \sum_{k \in \mathcal{K}_j : \widehat{x}_k^w = 0} x_k \right), \ \forall s \in \mathcal{S}, j \in \mathcal{J}, w \in \mathcal{W}^0 \quad (20)$$

The DD&ILS algorithm (Figure EC.3) involves three interconnected loops. The two upper loops solve the partial relaxation ($\texttt{MIO}-\texttt{LO}$) with integer L-shaped cuts, via DD (Figure 3). The lower loop adds new integer L-shaped cuts if the second-stage solution violates integrality requirements.

THEOREM 2. *The DD&ILS algorithm converges in a finite number of iterations to an optimal solution of Problem ($\star$), if $\mathcal{X}^{MIO} = \{0,1\}^{nz}$ and if the problem has relatively complete recourse.*

Unified branch-and-double-decomposition algorithm (UB&DD). The UB&BC algorithm from Mahéo et al. (2024) develops a single-tree algorithm in two-stage stochastic mixed-integer programming. It relies on tailored branching rules whenever a solution satisfies first-stage integrality requirements and Benders cuts but violates second-stage integrality constraints. Throughout, it maintains lower bounds from the linear and Benders relaxations, and upper bounds from second-stage heuristics. A post-processing procedure solves second-stage mixed-integer problems to find the optimum out of all candidates first-stage solutions. In contrast, the DD methodology derives a tight network-based reformulation of the second-stage problem and solves the partial relaxation (with mixed-integer first-stage solutions and continuous second-stage solutions) via Benders decomposition and column generation. We propose a UB&DD algorithm that combines these two elements—namely, the DD methodology to circumvent the exponential number of second-stage variables, and the UB&BC methodology to restore second-stage integrality (Figure EC.4).

THEOREM 3. *The UB&DD algorithm converges finitely to an optimal solution of Problem ($\star$).*

4.5. Solving the pricing problem in the MiND-VRP.

Consider two nodes in the load-expanded network $(u,c_1),(v,c_2)\in\mathcal{V}_{\ell st}$. The pricing problem seeks a subpath that starts in checkpoint $u\in\mathcal{N}^S$ at time $T_{\ell t}(u)$ with vehicle load c_1, and ends in checkpoint $v\in\mathcal{N}^S$ at time $T_{\ell t}(v)$ with load $c_2\geq c_1$, while satisfying the maximum deviation.

We characterize subpaths in a time-expanded network $(\mathcal{U}^{uv}_{\ell st},\mathcal{H}^{uv}_{\ell st})$. Let $\mathcal{T}^{uv}_{\ell t}$ be a set of discretized time intervals between $T_{\ell t}(u)$ and $T_{\ell t}(v)$. As in the segment-based benchmark (Section 3.4), the sets $\mathcal{T}^{uv}_{\ell t}$ need to be much more granular than the first-stage sets $\mathcal{T}_\ell$ (30 seconds vs. 15 minutes, in our experiments). Of course, this discretization is only applied locally in each pricing problem, thus retaining a much more manageable structure than in the segment-based benchmark. Each node $m\in\mathcal{U}^{uv}_{\ell st}$ is represented by a tuple $(k_m,t_m)\in\mathcal{N}^S_{uv}\times\mathcal{T}^{uv}_{\ell t}$; $(u,T_{\ell t}(u))\in\mathcal{U}^{uv}_{\ell st}$ is the source node and $(v,T_{\ell t}(v))\in\mathcal{U}^{uv}_{\ell st}$ is the sink node. The arc set $\mathcal{H}^{uv}_{\ell st}$ comprises traveling arcs connecting any node pair $(i,t)\rightarrow(j,t+tt_{ij})$ with travel time tt_{ij}, and idling arcs connecting $(i,t)\rightarrow(i,t+1)$. In particular, the graph $(\mathcal{U}^{uv}_{\ell st},\mathcal{H}^{uv}_{\ell st})$ is acyclic due to time moving forward in the time-space network. Each node $m\in\mathcal{U}^{uv}_{\ell st}$ also defines passengers' waiting, walking and travel times, as well as arrival delays and earliness, which we store in parameters τ^{walk}_{mp}, τ^{wait}_{mp}, $\tau^{\text{travel}}_{mp}$, τ^{late}_{mp}, and τ^{early}_{mp}. We denote by $\mathcal{P}_m\subset\mathcal{P}$ the set of passengers that can be picked up at node $m\in\mathcal{U}^{uv}_{\ell st}$ given the walking and waiting restrictions (Section 3.2). Table EC.1 summarizes notation. We define the following variables:

$$f_{mq}=\begin{cases}1 & \text{if arc } (m,q)\in\mathcal{H}^{uv}_{\ell st} \text{ is traversed in the time-expanded road segment network,}\\ 0 & \text{otherwise.}\end{cases}$$

$$w_{mp}=\begin{cases}1 & \text{if passenger } p\in\mathcal{P}_m \text{ is picked up in node } m\in\mathcal{U}^{uv}_{\ell st},\\ 0 & \text{otherwise.}\end{cases}$$

$$\xi_m = \text{ vehicle load in node } m\in\mathcal{U}^{uv}_{\ell st}$$

Let $\widehat{g}_a$ denote the reduced cost of arc-based variable $a=((u,c_1),(v,c_2))\in\mathcal{A}_{\ell st}$. In the MiND-VRP, each partition element $j\in\mathcal{J}$ corresponds to a reference trip $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$ and each node $n\in\mathcal{N}_{sj}$ to a checkpoint-load pair $(u,c)\in\mathcal{V}_{\ell st}$; the dual variables can be written as $\psi_{\ell,s,t,(u,c)}$ and $\gamma_{\ell stp}$ (because the second-stage problem admits one side constraint per passenger, in Equation (10)). From Equation (19) (or Equation (EC.48)), the reduced cost can be separated into a routing component and a load component. The routing component comprises (i) the level-of-service penalty for served passengers, and (ii) the value of serving a passenger, captured by the reward M and the dual price $\gamma_{\ell stp}$. The load component reflects the dual cost differential $\psi_{\ell,s,t,(v,c_2)}-\psi_{\ell,s,t,(u,c_1)}$

$$\widehat{g}_a=\underbrace{\sum_{m\in\mathcal{U}^{uv}_{\ell st}}\sum_{p\in\mathcal{P}_m}d_{mp}w_{mp}}_{\text{routing component}}+\underbrace{\psi_{\ell,s,t,(v,c_2)}-\psi_{\ell,s,t,(u,c_1)}}_{\text{load component}} \tag{21}$$

$$\text{with}\quad d_{mp}=D_{ps}\left(\frac{\delta\tau^{\text{late}}_{mp}+\frac{\delta}{2}\tau^{\text{early}}_{mp}+\sigma\tau^{\text{travel}}_{mp}}{\tau^{dir}_p}+\lambda\tau^{\text{walk}}_{mp}+\mu\tau^{\text{wait}}_{mp}-M\right)+\gamma_{\ell stp}.$$

The pricing problem seeks a subpath with minimum reduced cost (Equation (22)). Constraints (23)–(25) define the load at each node, starting from load c_1 and ending with load c_2. Constraints (26) and (27) ensure that passenger pickups occur only in visited nodes, and at most once. Constraints (28) apply flow balance in the time-expanded network.

$$\mathrm{PP}_{\ell st}^{u,v,c_1,c_2} \quad \min \quad \sum_{m\in\mathcal{U}_{\ell st}^{uv}}\sum_{p\in\mathcal{P}_m} d_{mp}w_{mp} + \psi_{\ell,s,t,(v,c_2)} - \psi_{\ell,s,t,(u,c_1)} \tag{22}$$

$$\text{s.t.} \quad \xi_{(u,T_{\ell t}(u))} = c_1, \; \xi_{(v,T_{\ell t}(v))} = c_2 \tag{23}$$

$$\xi_q - \xi_m \leq \sum_{p\in\mathcal{P}_m} D_{mp}w_{mp} + C_\ell(1-f_{mq}), \quad \forall (m,q)\in\mathcal{H}_{\ell st}^{uv} \tag{24}$$

$$\xi_q - \xi_m \geq \sum_{p\in\mathcal{P}_m} D_{mp}w_{mp} - C_\ell(1-f_{mq}), \quad \forall (m,q)\in\mathcal{H}_{\ell st}^{uv} \tag{25}$$

$$w_{mp} \leq \sum_{q:(m,q)\in\mathcal{H}_{\ell st}^{uv}} f_{mq} \quad \forall m\in\mathcal{U}_{\ell st}^{uv}, \; \forall p\in\mathcal{P}_m \tag{26}$$

$$\sum_{m\in\mathcal{U}_{\ell st}^{uv}:p\in\mathcal{P}_m} w_{mp} \leq 1 \quad \forall p\in\mathcal{P}:(\ell,t)\in\mathcal{M}_p \tag{27}$$

$$\sum_{q:(m,q)\in\mathcal{H}_{\ell st}^{uv}} f_{mq} - \sum_{q:(q,m)\in\mathcal{H}_{\ell st}^{uv}} f_{qm} = \begin{cases} 1 & \text{if } m=(u,T_{\ell t}(u)), \\ -1 & \text{if } m=(v,T_{\ell t}(v)), \\ 0 & \text{otherwise.} \end{cases} \quad \forall m\in\mathcal{U}_{\ell st}^{uv} \tag{28}$$

$$\boldsymbol{f},\boldsymbol{w} \text{ binary}, \; \boldsymbol{\xi} \text{ non-negative integer} \tag{29}$$

Whenever the solution of the pricing problem is negative, we add the corresponding subpath-based arc $a\in\mathcal{A}_{\ell st}$ to the load-expanded network, by defining its cost parameter g_a as:

$$g_a = \sum_{m\in\mathcal{U}_{\ell st}^{uv}}\sum_{p\in\mathcal{P}_m} D_{ps}\left(\frac{\delta\tau_{mp}^{late}+\frac{\delta}{2}\tau_{mp}^{early}+\sigma\tau_{mp}^{travel}}{\tau_p^{dir}} + \lambda\tau_{mp}^{\text{walk}} + \mu\tau_{mp}^{\text{wait}} - M\right) w_{mp} \tag{30}$$

Remark 1 notes that the the pricing problem searches over all subpaths, including those from non-selected reference lines (Equation (28)) and non-assigned passengers (Equation (27)). Such subpaths will be primal infeasible in the restricted Benders subproblem. However, the corresponding constraints take the form "$0\leq 0$" and cannot be assumed to have zero duals. This is essential to certify the validity of Benders decomposition. In general terms, the dual polyhedron $\mathcal{P}_{sj}$ is independent on the incumbent first-stage variables $\boldsymbol{x}$, and so is the pricing problem.

Remark 1. The right-hand side of Equation (28) (resp, Equation (27)) must be 1 rather than $x_{\ell t}$ (resp., $z_{\ell pst}$) to certify optimality of the RBSP solution and guarantee the algorithm's exactness.

For each reference trip $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$ and scenario $s\in\mathcal{S}$, the pricing problem is defined for each node pair $((u,c_1),(v,c_2))\in\mathcal{V}_{\ell st}\times\mathcal{V}_{\ell st}$ in the load-expanded network. In fact, we can reduce the number of pricing problems by exploiting the decomposition of the reduced cost into routing and load component. We first maximize the load component for each differential $\nu\in\mathcal{C}_\ell$:

$$\Delta\psi_{\ell st}^{u,v,\nu} = \max\left\{\psi_{\ell stn} - \psi_{\ell stm} : (m,n)\in\mathcal{A}_{\ell st}, k_m = u, k_n = v, c_n - c_m = \nu\right\}$$

We then seek a subpath that serves ν passengers and minimizes the routing component:

$$Z^{u,v,\nu}_{\ell st} = \min \sum_{m\in\mathcal{U}^{uv}_{\ell st}} \sum_{p\in\mathcal{P}_m} d_{mp} w_{mp};\ \text{s.t.} \sum_{m\in\mathcal{U}^{uv}_{\ell st}} \sum_{p\in\mathcal{P}_m} D_{ps} w_{mp} = \nu;\ \text{Equations (26)–(29)}$$

Proposition 4 shows that we can solve one pricing problem for each *load differential* and every pair of checkpoints. This result reduces the number of pricing problem by a factor $\mathcal{O}(\max_{\ell\in\mathcal{L}} C_\ell)$, while retaining the finite convergence and exactness of the column generation scheme.

Proposition 4. *$Z^{u,v,\nu}_{\ell st} - \Delta\psi^{u,v,\nu}_{\ell st}$ is the minimum reduced cost across all arc-based variables between checkpoints u and v with load differential ν, for all $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$, $s\in\mathcal{S}$.*

Label setting. The pricing problem is a resource-constrained shortest path problem. We design a label-setting algorithm exploiting the directed and acyclic structure of $(\mathcal{U}^{uv}_{\ell st}, \mathcal{H}^{uv}_{\ell st})$.

State definition. Let $(m^\sigma, \mathbb{P}^\sigma)$ denote a state, where m^σ tracks the "current" node, and $\mathbb{P}^\sigma$ tracks the set of served passengers $p\in\mathcal{P}$ each with pickup node ρ_p. We track the reduced cost $G(m^\sigma, \mathbb{P}^\sigma)$.

Initial state: $(m^0 = m, \mathbb{P}^0 = \emptyset)$, where m is such that $k_m = u$ and $t_m = T_{\ell t}(u)$; $G(m^0, P^0) = 0$.

State transitions. For each arc $(m,q)\in\mathcal{H}^{uv}_{\ell st}$ and each passenger combination $\mathbb{P}_m \subseteq \mathcal{P}_m$, the state is updated to $(q, \mathbb{P}^\sigma \cup \mathbb{P}_m)$. For each new passenger $p\in\mathbb{P}_m \setminus \{\mathbb{P}^\sigma\}$, the pickup point is set to $\rho_p = m$. For existing passengers $p\in\mathbb{P}_m \cap \mathbb{P}^\sigma$, we update the pickup node to be $\rho_p = m$ if $d_{mp} < d_{\rho_p,p}$. This transition is admissible if the vehicle has enough capacity, i.e., if $\sum_{p\in\mathbb{P}^\sigma\cup\mathbb{P}_m} D_{ps} \le C_\ell$.

Reward function. $G(m^\sigma, \mathbb{P}^\sigma) = \sum_{p\in\mathbb{P}^\sigma} d_{\rho_p,p}$ tracks the reduced cost of a subpath up to state σ.

Dominance rule. σ^1 dominates σ^2 if $m^{\sigma^1} = m^{\sigma^2}$, $\mathbb{P}^{\sigma^1} = \mathbb{P}^{\sigma^2}$, and $G(m^{\sigma^1}, \mathbb{P}^{\sigma^1}) \le G(m^{\sigma^2}, \mathbb{P}^{\sigma^2})$.

Upon termination, we extract all non-dominated states such that $m^\sigma = m : k_m = v$ and $t_m = T_{\ell t}(v)$. We then add to the RBSP all arcs $a\in\mathcal{A}_{\ell st}\setminus\{\mathcal{A}'_{\ell st}\}$ such that $k_{start(a)} = u$, $k_{end(a)} = v$, $c_{end(a)} - c_{start(a)} = \sum_{p\in\mathbb{P}^\sigma} D_{ps}$, with negative reduced cost $\widehat{g}_a = G(m^\sigma, \mathbb{P}^\sigma) - \psi_{\ell,s,t,start(a)} + \psi_{\ell,s,t,end(a)} < 0$.

By design, the dominance rule yields the subpath of minimum reduced cost for each passenger combination—hence, for each load differential. Thus, we apply the label-setting algorithm for each pair of checkpoints $u, v\in\mathcal{I}_\ell$, but do not duplicate it for each load differential. The number of checkpoint pairs grows linearly with $|\mathcal{I}_\ell|$ because subpaths can skip up to $K\in\{0,1\}$ checkpoint. Combined with Proposition 4, we obtain the following reduction on the number of pricing problems:

Proposition 5. *The label-setting algorithm generates $\mathcal{O}(2^\Xi|\mathcal{V}_{\ell st}|)$ variables at a time by only solving $\mathcal{O}(I_\ell)$ pricing problems, for each reference trip $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$ and scenario $s\in\mathcal{S}$.*

Heuristic label-setting algorithm. The algorithm can lead to a weak dominance rule with almost-identical subpaths serving slightly different passenger combinations. In fact, subpaths are relatively short, so the pricing problem rarely rejects a passenger with a negative reduced cost ($d_{mp} < 0$) to free up capacity for a subsequent passenger. Moreover, it can be undesirable in practice to reject a passenger at a station visited by the vehicle. We therefore propose a heuristic acceleration such that, in each node $m \in \mathcal{U}^{uv}_{\ell st}$, all candidate passengers $p \in \mathcal{P}_m$ with negative reduced cost contribution ($d_{mp} < 0$) are served, as long as the vehicle does not operate at capacity. This heuristic yields an upper-bounding approximation of the pricing problem, i.e., it generates solutions with a negative reduced cost but can potentially miss other subpaths with negative reduced cost. Then, we can switch back to the full label-setting algorithm to derive a certificate of optimality. In our experiments, the heuristic results in significant speedups with high-quality solutions.

5. Computational Assessment of the Methodology

We develop a real-world experimental setup in Manhattan. We use demand data from the NYC Taxi & Limousine Commission (2021) during the morning rush (6–9 am). We define a road network and travel times using data from Google Maps, OpenStreetMap, and Uber (2020). Parameter values are reported in EC.3.1. We design candidate reference lines using breadth-first search (EC.3.2).

We consider a MiND-VRP setting corresponding to a shuttle service from Manhattan to LaGuardia Airport (LGA) with vehicles of capacity 10 to 20 passengers. We vary the number of candidate reference lines (5 to 100), the planning horizon (1 to 3 hours), the number of checkpoints that can be skipped ($K = 0, 1, 2, 3$), and the number of scenarios (5 to 20). We use a 15-minute discretization to schedule transit vehicles in the first stage (sets $\mathcal{T}_\ell$), and a 30-second discretization in the second stage (sets $\mathcal{T}^{uv}_{\ell t}$). Our problem includes up to 1,900 passenger requests, 640 candidate stops, and 100 candidate reference lines (Figure EC.5), resulting in over 1 million first-stage variables, 25,000 Benders subproblems, and 200,000 pricing problems. We also develop real-world experimental setups for the MiND-DAR in EC.4.3 and for the MiND-Tr in EC.5.3.

All models are solved with Gurobi v12.0 using the JuMP package in Julia (Dunning et al. 2017). We impose a three-hour time limit for optimization. All instances and code are available online.[6]

5.1. Benefits of Subpath Modeling and Double-decomposition Algorithm

Table 1 compares the four formulations defined in Section 3 in terms of solution quality (normalized to the best-found solution), computational times, and number of second-stage variables. All models are solved with off-the-shelf methods, using exhaustive enumeration of segments, subpaths or paths in the network-based reformulations (with up to 1 million paths per subproblem). The compact formulation, despite its much much smaller size, does not scale to even the smallest instances. This underscores the highly challenging structure of the two-stage stochastic optimization structure with discrete recourse. Among network-based approaches, the segment-based formulation features the most limited scalability, requiring 30 million variables in the smallest instance due to the granular

[6] https://github.com/martiniradi/DeviatedFixedRouteMicrotransit

time-station-load discretization. The path-based formulation scales to medium instances but its performance quickly deteriorates due to the exponential growth in the number of variables. In comparison, the subpath-based formulation requires orders of magnitude fewer variables, terminates much faster, and returns a superior solution with 10 candidate lines.

Table 1 Comparison of path-based, subpath-based, segment-based, and compact MiND-VRP formulations.

			Path-based			Subpath-based			Segment-based			Compact		
$\lvert\mathcal{L}\rvert$	Hor.	K	Sol.	CPU (s)	Arcs	Sol.	CPU (s)	Arcs	Sol.	CPU (s)	Arcs	Sol.	CPU (s)	S2 Vars
5	60	0	100	137s	3.1M	100	17s	39K	100	6,431s	30.0M	—	10,800s+	58K
5	60	1	—	—	—	—	—	—	—	10,800s+	30.0M	—	10,800s+	68K
5	120	0	100	652s	8.6M	100	206s	86K	—	—	—	—	10,800s+	121K
5	180	0	100	696s	9.6M	100	238s	110K	—	—	—	—	10,800s+	182K
10	60	0	100.3	1,490s	29.1M	100	46s	142K	—	—	—	—	10,800s+	123K

"10,800s+": optimization timeout; "—": the algorithm does not terminate due to memory limitations.

These results demonstrate the benefits of the subpath-based representation of the second-stage problem. We provide additional results in EC.6.1 by comparing the four formulations on the capacitated vehicle routing problem with time windows in the second stage alone. These results uncover the different bottlenecks of the algorithms: the branch-and-cut structure in the compact formulation due to its weak linear relaxation, the size of the segment-based formulation, and the enumeration of arc variables in the path-based and subpath-based formulations. Whereas the subpath-based model is the most scalable one, subpath enumeration remains intractable in medium-scale instances, thus motivating our double-decomposition algorithm.

Next, Table 2 compares Benders decomposition with subpath enumeration and our DD methodology with exact and heuristic label setting. Benders decomposition alone remains limited with subpath enumeration. In comparison, our double-decomposition algorithm achieves much stronger scalability by leveraging column generation in the Benders subproblem.

Specifically, when all checkpoints must be visited ($K = 0$), the DD algorithm can solve instances with 50 candidate lines and a three-hour horizon, or instances with 100 candidate lines and a two-hour horizon. When vehicles can skip checkpoints ($K = 1$), the longer subpaths result in exponentially larger second-stage problems; subpath enumeration fails to find feasible solutions, whereas DD can solve instances with up to 10 candidate reference lines and a three-hour horizon (we report more results on the role of K in EC.6.2). These improvements are driven by the small number of variables needed to guarantee convergence in column generation—namely, DD converges with up to 93% fewer variables. Moreover, the DD algorithm yields a 0.0–0.1% optimality gap whenever it terminates, confirming the tightness of our subpath-based formulation. Ultimately, the DD methodology provides certifiably optimal, or near-optimal solutions in large and otherwise-intractable instances of the problem where all benchmarks fail to even return a feasible solution.

Table 2 Assessment of exact algorithms for solving ($\texttt{MIO} - \texttt{LO}$).

			$K = 0$								$K = 1$				
			Benders			DD (exact)			DD (acceleration)		DD (exact)			DD (acceleration)	
$\|\mathcal{L}\|$	$\|\mathcal{S}\|$	Horizon	Sol.	Gap	CPU(s)	Sol.	Gap	CPU(s)	Sol.	CPU(s)	Sol.	Gap	CPU(s)	Sol.	CPU(s)
5	5	60	100	**0.0%**	39	100	**0.0%**	11	102.2	10	100	**0.0%**	165	102.1	47
		120	100	**0.0%**	513	100	**0.0%**	38	102	30	100	**0.0%**	4,834	101.2	169
		180	100	**0.0%**	599	100	**0.0%**	49	101.1	53	100	**0.0%**	4,212	101.6	217
	20	60	—	—	—	100	**0.0%**	71	101.5	47	100	**0.0%**	10,489	102.4	382
		120	—	—	—	100	**0.0%**	349	101.3	161	100	4.9%	10,800	101.1	1,224
		180	—	—	—	100	**0.0%**	535	101.1	219	100.1	3.6%	10,800	100	3,368
10	5	60	100	**0.0%**	93	100	**0.0%**	77	101.4	61	100	**0.0%**	10,380	102.1	617
		120	100	**0.0%**	836	100	**0.0%**	239	101.3	158	107.3	21.3%	10,800	100	2,732
		180	100	**0.0%**	1,047	100.1	**0.1%**	427	100.9	210	108	25.3%	10,800	100	10,800
	20	60	—	—	—	100	**0.1%**	803	101.1	558	—	—	—	—	—
		120	—	—	—	100	**0.0%**	2,354	101.2	1,622	—	—	—	—	—
		180	—	—	—	100	**0.0%**	5,473	100.8	4,412	—	—	—	—	—
50	5	60	—	—	—	100	**0.0%**	2,265	100.2	613	—	—	—	—	—
		120	—	—	—	100	1.3%	10,800	100.7	10,800	—	—	—	—	—
		180	—	—	—	100	1.8%	10,800	100.7	10,800	—	—	—	—	—
	20	60	—	—	—	100	0.9%	10,800	100.6	10,800	—	—	—	—	—
		120	—	—	—	100	12.5%	10,800	109.2	10,800	—	—	—	—	—
		180	—	—	—	100	23%	10,800	112.2	10,800	—	—	—	—	—
100	5	60	—	—	—	100	**0.0%**	5,665	100.2	1,887	—	—	—	—	—
		120	—	—	—	100	0.8%	10,800	100.4	10,800	—	—	—	—	—
		180	—	—	—	100.8	3.0%	10,800	100	10,800	—	—	—	—	—
	20	60	—	—	—	102.3	3.9%	10,800	100	10,800	—	—	—	—	—
		120	—	—	—	100	28.7%	10,800	108.6	10,800	—	—	—	—	—

"—" indicates that the algorithm does not terminate due to memory limitations.
Optimality gaps measure the difference between a feasible solution to ($\star$) and the partial relaxation ($\mathcal{P} - \texttt{MIO} - \texttt{LO}$).
Bolded values indicate instances that terminate within the time limit.

Then, our label-setting acceleration further enhances the scalability of the algorithm. These benefits are stronger with $K = 1$ because the stronger dominance criterion becomes more impactful with longer subpaths. In small instances, DD terminates up to 3 times faster with the pricing heuristic, while returning solutions within 3% of the optimum. In medium instances, the pricing heuristic actually enables higher-quality solutions in faster computational times. This is because the DD algorithm with exact label-setting algorithm does not terminate within the time limit, and the heuristic label-setting scheme enables more effective convergence due to a much smaller number of subpaths (up to 52% and 78% fewer subpaths when $K = 0$ and $K = 1$, respectively). In other words, the benefits of acceleration can outweigh the slight loss of flexibility when choosing which passengers to pick up at each station. Ultimately, by combining Benders decomposition, column generation and label-setting acceleration, our algorithm can solve realistic instances with up to 100 candidate reference lines, hundreds of stations, 5 demand scenarios and a three-hour horizon (or 20 scenarios and a one-hour horizon). These correspond to large-scale network design and routing instances, with hundreds of candidate stops and thousands of passenger requests.

Similarly, despite the higher complexity of the problem, the methodology can handle realistic MiND-DAR and MiND-Tr instances with up to 10–40 candidate lines (EC.4.3 and EC.5.3).

Finally, Table 3 reports results of DD&ILS and UB&DD. Recall that Benders decomposition and DD solve the partial relaxation $(\mathcal{P}-\texttt{MIO}-\texttt{LO})$, while providing a valid (and tight) optimality gap for Problem ($\star$), whereas the other two methods certifiably solve Problem ($\star$) (Theorems 2–3). These results identify instances where DD leaves a small optimality gap that DD&ILS can close (e.g., 8 lines, 5 scenarios, two-hour horizon). Yet, by solving the second-stage mixed-integer problem repeatedly, both the DD&ILS and UB&DD algorithms are much more computationally intensive. In medium instances, DD&ILS return the optimal solution in longer computational times than DD, whereas UB&DD fails to return a feasible solution. In larger instances, both DD&ILS and UB&DD time out whereas DD still returns a certifiably optimal solution.

Table 3 Assessment of exact algorithms to solve Problem ($\star$).

			Benders			DD			DD&ILS			UB&DD		
$\lvert\mathcal{L}\rvert$	$\lvert\mathcal{S}\rvert$	Horizon	Sol.	Gap	CPU(s)	Sol.	Gap	CPU(s)	Sol.	Gap	CPU(s)	Sol.	Gap	CPU(s)
5	5	60	100.0	**0.0%**	39	100.0	**0.0%**	11	100.0	**0.0%**	43	100.0	**0.0%**	41
		120	100.0	**0.0%**	513	100.0	**0.0%**	38	100.0	**0.0%**	348	100.0	**0.0%**	63
8	5	60	100.0	**0.0%**	93	100.0	**0.0%**	100	100.0	**0.0%**	340	100.0	**0.0%**	1,464
		120	100.2	**0.2%**	908	100.2	**0.2%**	214	100.0	**0.0%**	916	—	—	10,800
10	20	60	—	—	—	100.0	**0.0%**	803	—	—	10,800	—	—	10,800
		120	—	—	—	100.0	**0.0%**	2,354	—	—	10,800	—	—	10,800

These results underscore that the structural complexity of Problem ($\star$) stems from the exponential size of the second-stage problem rather than its discreteness. Integer L-shaped cuts and the UB&BC scheme can be critical to solve stochastic integer problems with a polynomial second-stage formulation and a weaker relaxation; in contrast, the network-based representation in Problem ($\star$) yields a very tight second-stage reformulation, thus shifting the complexity of the problem from a discrete recourse function to an exponential number of second-stage variables—and motivating our DD algorithm. As our results show, the integer L-shaped cuts and UB&BC provide limited marginal computational benefits in this setting whereas the DD methodology is instrumental in enabling convergence and deriving high-quality solutions to Problem ($\star$).

5.2. Benefits of Stochastic Optimization Methodology

Table 4 compares the stochastic optimization solution against a deterministic baseline and a clairvoyant benchmark. The MiND-VRP reduces unmet demand by 6-7% on average, while reducing passengers' walking time by 25-35% from the deterministic baseline, resulting in a high VSS—5-7% on average and up to 10%. In fact, our solution bridges 40-50% of the gap on average between the deterministic and perfect-information benchmarks. These results highlight the benefits of our two-stage stochastic optimization model (and our DD methodology to solve it) to increase demand coverage while maintaining or even improving level of service, as compared to a deterministic model (which can be solved via off-the-shelf methods).

Table 4 Value of Stochastic Solution (VSS) and Expected Value of Perfect Information (EVPI)

					Performance assessment			VSS breakdown					
$\mathcal{L}$	$\lvert\mathcal{S}\rvert$	Horizon	K	Heur.	$\lvert\frac{\text{VSS}}{Sol.}\rvert$	$\lvert\frac{\text{EVPI}}{Sol.}\rvert$	$\frac{\text{VSS}}{(\text{VSS+EVPI})}$	Unmet demand (%)	Walking time (%)	Waiting time (%)	Earliness (%)	Delay (%)	Detour (%)
10	5	60	0	✗	5.8	8.2	41.2	-6	-59.3	0.7	-2.1	13.2	1
			1	✗	2.7	9.7	22	-2.7	36.3	-1.6	-9.9	3.9	0.4
		120	0	✗	4	7.4	35.2	-4.3	-77	1.2	1.7	13	0.5
			1	✓	3	7.4	28.6	-2.8	-1.6	-6.8	3.3	-4.8	-0.8
		180	0	✗	9.9	7.4	57.4	-11	-70.6	7.4	3.4	-1.5	-0.4
			1	✓	7	5.7	55.2	-7.5	-17.4	0.4	-5.8	4.1	0.5
	20	60	0	✗	7.6	5.9	56.3	-8.1	-91.6	1.1	-5.6	15.7	1
			1	✓	9.8	6.1	61.5	-10.4	-8.1	-10.2	-11.4	23.9	-1.2
		120	0	✓	9.7	9.2	51.1	-10.8	-81.2	4.3	10.5	4.2	0.2
			1	✓	4.8	8	37.8	-5	49.7	7.4	-6.9	10.3	0
		180	0	✓	6.1	7.2	45.7	-6.6	-82.9	6.4	6.9	14.5	3.2
			1	✓	2.7	7.3	27.2	-2.8	-2.8	5.1	-0.6	14	1.4
50	5	60	0	✗	5.8	2.5	70.3	-6.5	-6.1	0	-13	6.2	0.3
		120	0	✓	5.3	7	43.1	-5.7	-2	1.3	2.7	1.8	1
		180	0	✗	4.3	10.7	30.5	-4.5	-16.7	-2.1	8.3	4.2	0.8
100	5	60	0	✓	8.8	2.3	79.4	-11.1	-56.2	-19.5	1.3	-1.5	-0.7
		120	0	✓	5	7.4	40.2	-5.6	-25.2	-7.6	0.4	0.1	0.6
		180	0	✓	2.4	15.5	13.4	-2.7	-3.7	-5.6	0.3	-3.2	1
Average 5 scenarios					**5.3**	**7.6**	**43.0**	**-5.9**	**-25.0**	**-2.7**	**-0.8**	**3.0**	**0.4**
Average 20 scenarios					**6.8**	**7.3**	**46.6**	**-7.3**	**-36.2**	**2.4**	**-1.2**	**13.8**	**0.8**

"Heur.": solution with heuristic (✓) vs exact (✗) pricing algorithm; "Sol.": stochastic optimization solution. Unmet demand is measured in number of passengers; all other components are measured per served passenger.

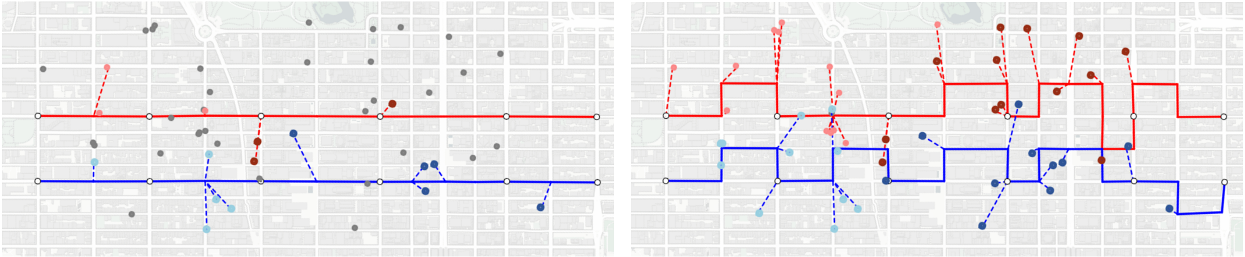
6. Practical Assessment of Deviated Fixed-route Microtransit

Finally, we conduct a comprehensive assessment of microtransit against fixed-route transit (a single-stage problem without second-stage deviations) and ride-sharing (on-demand system with vehicle capacities of 1, 2, and 4, described in EC.3.3). We use the same experimental setup as in Section 5. Recall that, since Manhattan represents a high-density region, the results can be seen as conservative estimates of the impact of microtransit in lower-density areas with fewer transit options. Again, all our insights hold in the MiND-DAR and MiND-Tr, as shown in EC.4.3 and EC.5.3.

6.1. Value of Microtransit Flexibility

Second-stage microtransit operations. Figure 5 illustrates MiND-DAR operations along two lines in Midtown Manhattan. By design, transit follows the reference line whereas microtransit deviates from the reference line in all but one checkpoint pair. Thus, the microtransit system serves more passengers (24 versus 8), at the cost of a longer distance (8.5 vs. 5.5 km). Still, the higher vehicle loads leads to a smaller distance per passenger (356 vs. 699 meters), which can translate into lower costs for the operator, lower fares for passengers, and a smaller environmental footprint.

Table 5 reports average performance and level of service in the second stage of the MiND-VRP, with different vehicle capacities and extents of flexibility ($\Delta = 600$ vs. $\Delta = 1,200$ meters; $K = 0$ vs. $K = 1$). In these experiments, we use the same reference trips for transit and microtransit. On average, microtransit can add 1–4 passengers per vehicle, while reducing walking times by 50% and wait times by 2 minutes. These benefits come with a small increase in detours (+2%)



(a) Fixed-route transit

(b) Deviated fixed-route microtransit

Figure 5 **Illustration of transit and microtransit operations in the MiND-DAR for two reference lines [light (resp. dark) blue/red circles: origins (resp. destinations) of passengers served by the blue/red line; grey circles: origins and destinations of unserved passengers; white circles: checkpoints].**

and an increase in distance traveled (+15–25%). Still, due to the large increase in utilization, distance per passenger is reduced by up to 500 meters, or 23%. These benefits become stronger with larger vehicles. Interestingly, even when microtransit vehicles are constrained to stay close to the reference lines (low deviation) and to visit all checkpoints ($K = 0$), the microtransit system can significantly improve coverage (0.5 to 3 extra passengers per vehicle, on average) and level of service (reduction in walking times by 40 seconds and in waiting times by 1 minute). In other words, even limited extents of demand-responsiveness can achieve significant performance improvements through stronger demand consolidation, higher level of service, and a smaller cost per passenger.

Table 5 **Average operating performance and level of service for fixed-route transit and microtransit.**

	Operating model			Average operating performance				Average level of service			
Cap.	Mode	Dev.	Skip?	#pass./vehicle	Util.	Dist.	Dist./pass.	Walk	Wait	Detour	Delay
10	Transit	—	—	8.05	80.50%	14.78	2.66	2.19	6.77	152.06%	-0.87
	Microtransit	Low	$K = 0$	8.61	86.12%	16.48	2.53	1.48	5.69	154.36%	-0.47
	Microtransit	High	$K = 0$	8.63	86.32%	16.57	2.51	1.44	5.58	153.62%	-0.44
	Microtransit	Low	$K = 1$	8.78	87.81%	17.08	2.54	1.17	4.51	156.50%	-0.28
	Microtransit	High	$K = 1$	8.99	89.87%	17.39	2.38	1.03	4.33	153.72%	-0.28
15	Transit	—	—	10.72	71.47%	15.06	2.40	2.27	6.88	150.73%	-1.29
	Microtransit	Low	$K = 0$	12.20	81.32%	17.17	2.17	1.57	5.90	151.82%	-0.49
	Microtransit	High	$K = 0$	12.29	81.96%	17.34	2.14	1.50	5.74	151.20%	-0.46
	Microtransit	Low	$K = 1$	12.56	83.70%	17.78	2.15	1.31	4.83	154.83%	-0.26
	Microtransit	High	$K = 1$	12.89	85.95%	18.15	1.97	1.15	4.63	151.74%	-0.31
20	Transit	—	—	12.24	61.21%	15.16	2.34	2.30	6.94	150.38%	-1.84
	Microtransit	Low	$K = 0$	15.28	76.42%	17.52	2.02	1.69	6.21	150.77%	-0.52
	Microtransit	High	$K = 0$	15.46	77.32%	17.72	1.98	1.62	6.04	150.08%	-0.50
	Microtransit	Low	$K = 1$	15.90	79.49%	18.13	1.96	1.48	5.25	153.51%	-0.28
	Microtransit	High	$K = 1$	16.16	80.78%	18.57	1.81	1.43	5.39	151.17%	-0.33

"Cap." – Capacity; "Pass." – Passenger; "Util." – Utilization; "Dist." – Distance; "Dev." – deviation.
Units: distance, distance per passenger – kilometers; walk, wait, delay/earliness – minutes.
Parameters: two-hour horizon; 10 weekday scenarios, maximum walk: 7 minutes, maximum wait: 10 minutes.

Figure 6 plots the average vehicle load in fixed-route transit vs. deviated fixed-route microtransit, for each reference line broken down into low, medium and high density (colored lines) and for each vehicle capacity (dot shapes). All observations lie above the 45-degree line as microtransit makes use of the deviations to increase vehicle load. In low-density regions, microtransit vehicles do not operate at capacity, whereas low-occupancy ride-sharing can provide high levels of service at limited detours and delays. In high-density regions, fixed-route transit already provides high demand coverage due to high synergies across passengers, so the marginal improvements from microtransit are more limited. In-between, microtransit can increase average vehicle loads by 1–5 and the number of pickups by 5–10. These results identify a medium-density regime where deviated fixed-route microtransit can be most impactful, as population density is high enough to consolidate demand into high-occupancy vehicles but too low for fixed-route transit to be as effective.

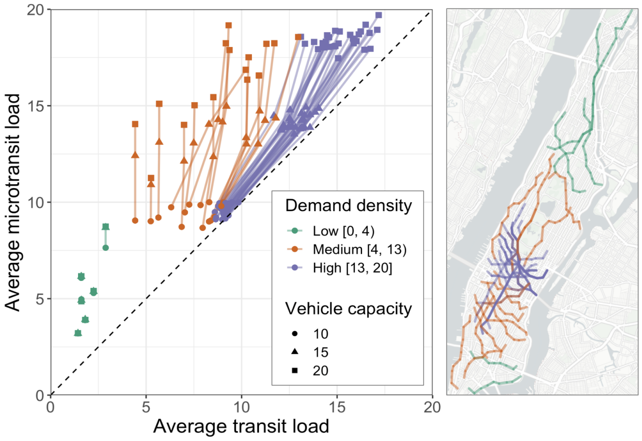
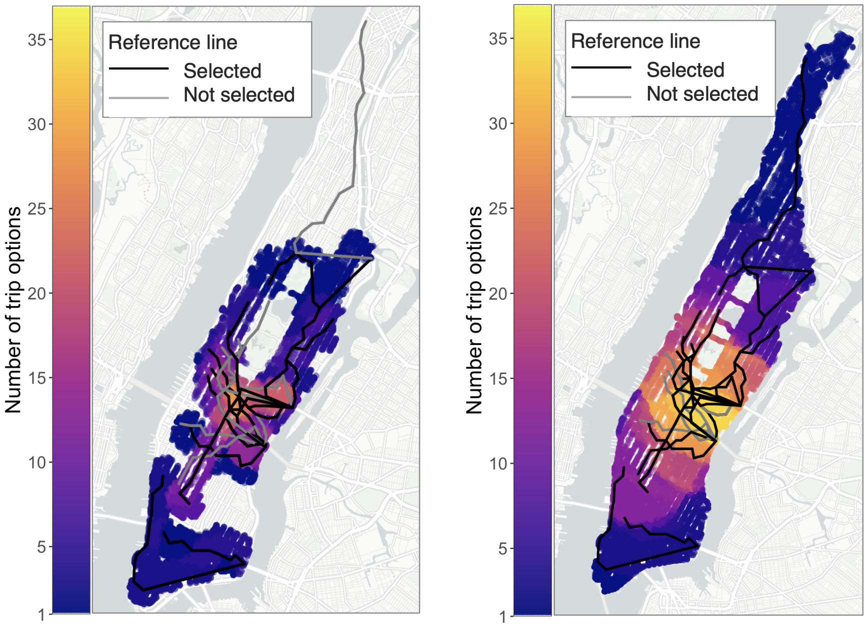


Figure 6 **Value of operating flexibility ($\Delta = 1,200$ m., $K = 0$). Low (resp. medium, high) density: lines with maximum load less than 4 (resp. 4 to 13, more than 13) passengers on average under transit.**

Network design. Figure 7 depicts the optimized first-stage networks, with reference lines labeled as “selected” if at least one reference trip is selected over the planning horizon. The figure also depicts the number of trip options from each of Manhattan’s 21,000 roadway intersections, defined as the number of reference trips with a candidate pickup location within a 5-minute walking radius. Note that the microtransit network expands the catchment area from fixed-route transit. Consistently with Figure 6, the fixed-route transit system mostly selects lines in high-demand areas. Thanks to its operating flexibility, microtransit provides more trip options: in Midtown Manhattan for instance, the number of trip options increases with microtransit from 20–25 to 30–35; overall, the average number of trip options per intersection increases by a factor of 3 (8.31 vs. 2.61). As a result, the microtransit system reaches low-demand regions, such as Uptown Manhattan. Specifically, microtransit covers 60% more intersections with at least one trip option (53.8% vs. 85.4%). By enhancing service options in high-density regions, microtransit can allocate some budget to expand its geographic reach to under-served regions, thus enhancing accessibility across the population.



(a) Fixed-route transit (b) Deviated fixed-route microtransit

Figure 7 **Reference lines and catchment areas. Parameters: 25 candidate reference lines, 2-hour horizon, 20 vehicles with a 20-passenger capacity each. Microtransit parameters: $\Delta = 1,200$ m., $K = 0$.**

6.2. Performance Assessment

We now compare the performance of microtransit against fixed-route transit and ride-sharing. To establish an apples-to-apples comparison, we fix total seating capacity across all systems (e.g., 10 transit/microtransit vehicles of capacity 10, ride-sharing with 100/50/25 vehicles of capacity of 1/2/4), and perform an out-of-sample assessment corresponding to five new weekdays. Unlike in Table 5 and Figure 6, we consider here the optimized network of reference lines in transit and microtransit. Table 6 reports average coverage, level of service, and distance traveled.

Table 6 **Average level of service of fixed-route transit, microtransit ($\Delta = 1,200$ m., $K = 0$), and ride-sharing.**

Mode	Design	Coverage	Walk	Wait	Detour	Delay	Distance
Transit	5 candidate lines	13.9%	2.06	7.06	158.56%	-1.17	356
	10 candidate lines	20.4%	2.21	6.91	146.22%	-0.79	384
	25 candidate lines	29.8%	2.03	6.8	136.98%	0.13	435
	50 candidate lines	33.6%	2.03	6.65	137.34%	-0.06	472
Microtransit	5 candidate lines	22.3%	1.68	6.22	159.99%	-0.01	419
	10 candidate lines	30.0%	1.68	6.22	146.11%	-0.15	462
	25 candidate lines	35.6%	1.53	5.82	138.52%	-0.16	471
	50 candidate lines	36.6%	1.36	5.55	141.00%	0.03	468
Rideshare	Cap. 4	36.3%	0	4.2	150.68%	13.4	1,883
	Cap. 2	44.7%	0	3.74	124.60%	8.17	3,359
	Cap. 1	50.5%	0	1.79	100.00%	1.79	5,671

Coverage: percentage of served requests; distance in kilometers; walk, wait, delay/earliness in minutes.

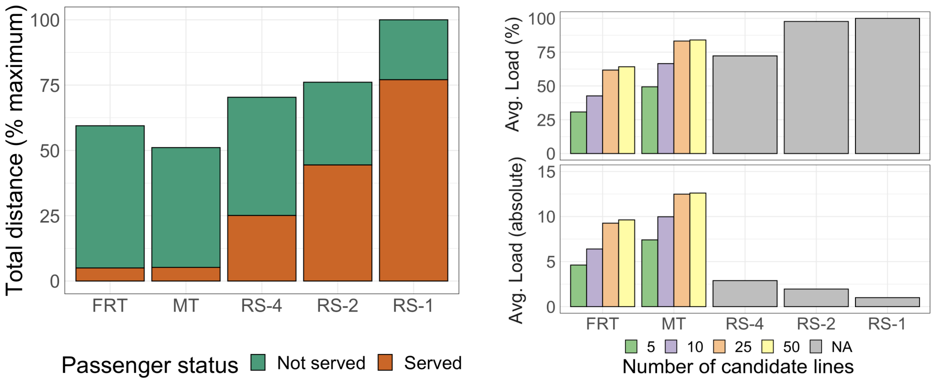
These results confirm that microtransit increases demand coverage and reduces walk and wait times from fixed-route transit, at virtually no cost in terms of detours and delays. The differences in coverage go down with more candidate lines. With 5–10 lines, the operating flexibility in microtransit is most valuable in sparse networks. With 25–50 lines, the transit network adds more reference trips in high-density regions. Since the microtransit network already achieves higher coverage with fewer lines, it allocates more lines to lower-demand regions (Section 6.1), resulting in a smaller marginal increase in coverage. At the other extreme, ride-sharing achieves high coverage with no walking (by design) and short waits, but much longer distances traveled with low-occupancy vehicles. Thus, microtransit defines a middle ground with less walk and less wait for passengers than transit, less delays than ride-sharing, and intermediate ridership and costs.

Another interesting observation stems from the comparison of microtransit to ride-pooling, both of which leverage on-demand operations to consolidate demand into multi-occupancy vehicles. On-demand door-to-door ride-pooling results in no walk and low wait times but increases detours and delays—underscoring the impact of spatiotemporal externalities, even with small-occupancy vehicles. By consolidating demand into high-capacity vehicles along reference lines, deviated fixed-route microtransit reduces distance traveled by a factor of 4 but reaches similar demand coverage and a comparable level of service—no delay, smaller detours, moderate walking times, and slightly longer wait times. These results identify deviated fixed-route microtransit as a possible pathway to provide efficient and convenient urban mobility options with high-capacity vehicles.

Figure 8 provides an out-of-sample system-wide assessment. Figure 8a plots total distance traveled, used as a proxy of operating costs and environmental footprint; it includes both the “internal” distance within the system plus the “external” distance from single-occupancy trips (e.g., taxi trips) for all unserved passengers. Microtransit reduces total distance by 10-15% versus fixed-route transit, by 20-30% versus ride-pooling, and by 50% versus single-occupancy ride-sharing. From Table 6, these benefits are driven by a much smaller internal distance than ride-sharing that outweigh the impact of smaller demand coverage, and by higher demand coverage than fixed-route transit that outweigh the slightly longer internal distances. Altogether, these results identify potential operating, economic and environmental benefits from deviated fixed-route microtransit from stronger demand consolidation than ride-sharing and from higher demand coverage than fixed-route transit.

We establish the robustness of these results in EC.6.2, and extend them in EC.4.3 and EC.5.3 to the MiND-DAR and MiND-Tr. For instance, in the MiND-DAR, deviated fixed-route microtransit increases demand coverage versus fixed-route transit, improves demand consolidation versus ride-sharing and ride-pooling, and reduces total distance versus all benchmarks (by 5-15% vs. fixed-route transit, by 40-50% vs. ride-pooling, and by over 100% vs. single-occupancy ride-sharing).

In conclusion, deviated fixed-route microtransit can contribute to efficient, equitable, and sustainable mobility. Efficiency stems from high levels of service, low operating costs and high demand coverage enabled by reference lines and on-demand flexibility. Equity stems from a microtransit



(a) Average total distance traveled.

(b) Average vehicle load.

Figure 8 **System-wide assessment of fixed-route transit (FRT), microtransit ($\Delta = 1,200$ m., $K = 0$) (MT), and ride-sharing systems with capacities 4, 2 and 1 (RS-4, RS-2 and RS-1, respectively).**

network with broader geographic reach. Sustainability stems from a smaller distance traveled per passenger enabled by high coverage and consolidation into high-capacity vehicles.

7. Conclusion

This paper optimizes the design and operations of a deviated fixed-route microtransit system endowed with advance planning capabilities along reference lines (as in public transit) and on-demand adjustments in response to passenger demand (as in ride-sharing). We formulated a *Microtransit Network Design (MiND)* model via two-stage stochastic integer optimization. The model features a first-stage network design and service scheduling structure and a second-stage capacitated vehicle routing structure with time windows. We proposed a subpath-based representation of microtransit operations in a load-expanded network, which results in a tight second-stage relaxation but an exponential number of variables. To solve it, we have developed a double-decomposition algorithm, leveraging Benders decomposition to decompose the problem per scenario and reference trip, as well as subpath-based column generation to further decompose operations between checkpoints.

Using New York City data, results showed that the methodology scales to real-world and otherwise-intractable problems, with up to 100 candidate reference lines, hundreds of stations, thousands of requests, and 5-20 demand scenarios. Practical results suggested that even limited on-demand flexibility can provide significant operating benefits by consolidating demand into high-capacity vehicles while leveraging on-demand deviations to enhance passenger level of service and increase demand coverage. At a time where hybrid solutions are emerging to design new mobility services, this paper suggests that deviated fixed-route microtransit can contribute to efficient mobility (high demand coverage, low operating costs, high levels of service), equitable mobility (high accessibility with broad geographic reach), and sustainable mobility (low environmental footprint). Based on these results, we have been collaborating with transit operators toward the deployment of deviated fixed-route microtransit, with a pilot implementation targeted in 2025.

Acknowledgments

The authors thank Joseph Kajon for helpful assistance with data analyses and experimentation.

References

Agarwal R, Ergun Ö (2008) Ship scheduling and network design for cargo routing in liner shipping. *Transportation Science* 42(2):175–196.

Ahuja RK, Hochbaum DS (2008) Solving linear cost dynamic lot-sizing problems in o (n log n) time. *Operations research* 56(1):255–261.

Allen DJ (2017) *Lost in the transit desert: Race, transit access, and suburban form* (Routledge).

Alonso-Mora J, Samaranayake S, Wallar A, Frazzoli E, Rus D (2017) On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences* 114(3):462–467.

Alyasiry AM, Forbes M, Bulmer M (2019) An exact algorithm for the pickup and delivery problem with time windows and last-in-first-out loading. *Transportation Science* 53(6):1695–1705.

Banerjee S, Hssaine C, Périvier N, Samaranayake S (2021) Real-time approximate routing for smart transit systems. *arXiv preprint arXiv:2103.06212* .

Bertsimas D, Jaillet P, Martin S (2019) Online vehicle routing: The edge of optimization in large-scale applications. *Operations Research* 67(1):143–162.

Bertsimas D, Ng YS, Yan J (2021) Data-driven transit network design at scale. *Operations Research* 69(4):1118–1133.

Bertsimas D, Yan J (2021) The edge of optimization in large-scale vehicle routing for paratransit. *Preprint* .

Blanchard M, Jacquillat A, Jaillet P (2023) Probabilistic bounds on the $k-$traveling salesman problem and the traveling repairman problem. *Mathematics of Operations Research* .

Bodur M, Dash S, Günlük O, Luedtke J (2017) Strengthened benders cuts for stochastic integer programs with continuous recourse. *INFORMS Journal on Computing* 29(1):77–91.

Bodur M, Luedtke JR (2017) Mixed-integer rounding enhanced benders decomposition for multiclass service-system staffing and scheduling with arrival rate uncertainty. *Management Science* 63(7):2073–2091.

Carøe CC, Schultz R (1999) Dual decomposition in stochastic integer programming. *Operations Research Letters* 24(1-2):37–45.

Ceder A, Wilson NH (1986) Bus network design. *Transportation Research Part B: Methodological* 20(4):331–344.

Chopra S, Martin S, Mishra PS, Smilowitz K (2023) Mobility-on-demand meets shuttles on the same mile. *Available at SSRN 4322824* .

Conforti M, Cornuéjols G, Zambelli G (2010) Extended formulations in combinatorial optimization. *4OR* 8(1):1–48.

Conforti M, Cornuéjols G, Zambelli G (2014) *Integer programming* (Springer).

Crainic TG, Hewitt M, Toulouse M, Vu DM (2016) Service network design with resource constraints. *Transportation Science* 50(4):1380–1393.

Cummings K, Vaze V, Ergun Ö, Barnhart C (2023) Multimodal transportation alliance design with endogenous demand: Large-scale optimization for rapid gains. *arXiv preprint arXiv:2301.03414* .

Delorme M, Iori M (2020) Enhanced pseudo-polynomial formulations for bin packing and cutting stock problems. *INFORMS Journal on Computing* 32(1):101–119.

Desaulniers G, Hickman MD (2007) Public transit. *Handbooks in operations research and management science* 14:69–127.

Dunning I, Huchette J, Lubin M (2017) JuMP: A modeling language for mathematical optimization. *SIAM review* 59(2):295–320.

Eppen GD, Martin RK (1987) Solving multi-item capacitated lot-sizing problems using variable redefinition. *Operations Research* 35(6):832–848.

Fortz B, Poss M (2009) An improved benders decomposition applied to a multi-layer network design problem. *Operations research letters* 37(5):359–364.

Frangioni A, Gendron B (2009) 0–1 reformulations of the multicommodity capacitated network design problem. *Discrete Applied Mathematics* 157(6):1229–1241.

Gade D, Küçükyavuz S, Sen S (2014) Decomposition algorithms with parametric gomory cuts for two-stage stochastic integer programs. *Mathematical Programming* 144(1):39–64.

Galarza Montenegro BD, Sörensen K, Vansteenwegen P (2022) A column generation algorithm for the demand-responsive feeder service with mandatory and optional, clustered bus-stops. *Networks* 80(3):274–296.

Guan H, Basciftci B, Van Hentenryck P (2023) Path-based formulations for the design of on-demand multimodal transit systems with adoption awareness. *arXiv preprint arXiv:2301.07292* .

Hasan MH, Van Hentenryck P (2021) The benefits of autonomous vehicles for community-based trip sharing. *Transportation Research Part C: Emerging Technologies* 124:102929.

Jacquillat A, Schmid A, Wang K (2022) Relay logistics: A multi-variable generation approach. *Available at SSRN 4241031* .

Karsten CV, Ropke S, Pisinger D (2018) Simultaneous optimization of container ship sailing speed and container routing with transit time restrictions. *Transportation Science* 52(4):769–787.

Kim K, Mehrotra S (2015) A two-stage stochastic integer programming approach to integrated staffing and scheduling with application to nurse management. *Operations Research* 63(6):1431–1451.

Laporte G, Louveaux FV (1993) The integer l-shaped method for stochastic integer programs with complete recourse. *Operations research letters* 13(3):133–142.

Lee J, Marla L, Jacquillat A (2020) Dynamic disruption management in airline networks under airport operating uncertainty. *Transportation Science* 54(4):973–997.

Liebchen C (2008) The first optimized railway timetable in practice. *Transportation Science* 42(4):420–435.

Liu X, Qu X, Ma X (2021) Improving flex-route transit services with modular autonomous vehicles. *Transportation Research Part E: Logistics and Transportation Review* 149:102331.

Macedo R, Alves C, de Carvalho JV, Clautiaux F, Hanafi S (2011) Solving the vehicle routing problem with time windows and multiple routes exactly using a pseudo-polynomial model. *European Journal of Operational Research* 214(3):536–545.

Mahéo A, Belieres S, Adulyasak Y, Cordeau JF (2024) Unified branch-and-benders-cut for two-stage stochastic mixed-integer programs. *Computers & Operations Research* 164:106526.

Maheo A, Kilby P, Van Hentenryck P (2019) Benders decomposition for the design of a hub and shuttle public transit system. *Transportation Science* 53(1):77–88.

Marín ÁG, Jaramillo P (2009) Urban rapid transit network design: accelerated benders decomposition. *Annals of Operations Research* 169(1):35–53.

McKinsey & Co (2018) Travel and logistics: data drives the race for customers. Technical report.

Muter I, Birbil Şİ, Bülbül K (2013) Simultaneous column-and-row generation for large-scale linear programs with column-dependent-rows. *Mathematical Programming* 142(1-2):47–82.

NYC Taxi & Limousine Commission (2021) TLC Trip Record Data. Available at: https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page.

Ortega FA, Pozo MA, Puerto J (2018) On-line timetable rescheduling in a transit line. *Transportation Science* 52(5):1106–1121.

Pessoa A, Uchoa E, De Aragão MP, Rodrigues R (2010) Exact algorithm over an arc-time-indexed formulation for parallel machine scheduling problems. *Mathematical Programming Computation* 2:259–290.

Quadrifoglio L, Dessouky MM, Ordóñez F (2008) Mobility allowance shuttle transit (MAST) services: MIP formulation and strengthening with logic constraints. *EJOR* 185(2):481–494.

Quadrifoglio L, Dessouky MM, Palmer K (2007) An insertion heuristic for scheduling mobility allowance shuttle transit (MAST) services. *Journal of Scheduling* 10(1):25–40.

Quadrifoglio L, Hall RW, Dessouky MM (2006) Performance and design of mobility allowance shuttle transit services: bounds on the maximum longitudinal velocity. *Transportation science* 40(3):351–363.

Queyranne M, Wolsey LA (2017) Tight mip formulations for bounded up/down times and interval-dependent start-ups. *Mathematical Programming* 164(1):129–155.

Rist Y, Forbes MA (2021) A new formulation for the dial-a-ride problem. *Transportation Science* 55(5):1113–1135.

Sadykov R, Vanderbeck F (2013) Column generation for extended formulations. *EURO Journal on Computational Optimization* 1(1-2):81–115.

Santi P, Resta G, Szell M, Sobolevsky S, Strogatz SH, Ratti C (2014) Quantifying the benefits of vehicle pooling with shareability networks. *PNAS* 111(37):13290–13294.

Schulz A, Pfeiffer C (2024) Using fixed paths to improve branch-and-cut algorithms for precedence-constrained routing problems. *European Journal of Operational Research* 312(2):456–472.

Sen S, Sherali HD (2006) Decomposition with branch-and-cut approaches for two-stage stochastic mixed-integer programming. *Mathematical Programming* 106:203–223.

Silva DF, Vinel A, Kirkici B (2022) On-demand public transit: A markovian continuous approximation model. *Transportation Science* 56(3):704–724.

Sousa JP, Wolsey LA (1992) A time indexed formulation of non-preemptive single machine scheduling problems. *Mathematical programming* 54:353–367.

Steiner K, Irnich S (2020) Strategic planning for integrated mobility-on-demand and urban public bus networks. *Transportation Science* 54(6):1616–1639.

Sun L, Xie W, Witten T (2023) Distributionally robust fair transit resource allocation during a pandemic. *Transportation science* 57(4):954–978.

Szufel, Przemysłlaw *et al* (2023) OpenStreetMapX.jl. https://github.com/pszufe/OpenStreetMapX.jl.

The Economist (2018) Public transport is in decline in many wealthy cities. www.economist.com/international/2018/06/21/public-transport-is-in-decline-in-many-wealthy-cities.

Uber (2020) New york city: Quarterly speed statistics by hour of day (q1 2020). Acc. Nov 2022 at https://movement.uber.com/cities/new_york/downloads/speeds?lang=en-US&tp[y]=2020&tp[q]=1.

US DoT (2016) Shared mobility current practices and guiding principles. Technical report.

Valério de Carvalho J (1999) Exact solution of bin-packing problems using column generation and branch-and-bound. *Annals of Operations Research* 86:629–659.

Vazifeh MM, Santi P, Resta G, Strogatz SH, Ratti C (2018) Addressing the minimum fleet problem in on-demand urban mobility. *Nature* 557(7706):534–538.

Walteros JL, Medaglia AL, Riaño G (2015) Hybrid algorithm for route design on bus rapid transit systems. *Transportation Science* 49(1):66–84.

Wang K, Jacquillat A (2020) A stochastic integer programming approach to air traffic scheduling and operations. *Operations Research* 68(5):1375–1402.

Wei K, Vaze V, Jacquillat A (2022) Transit planning optimization under ride-hailing competition and traffic congestion. *Transportation Science* 56(3):725–749.

Wu L, Adulyasak Y, Cordeau JF, Wang S (2022) Vessel service planning in seaports. *Operations research* 70(4):2032–2053.

Zeighami V, Soumis F (2019) Combining benders' decomposition and column generation for integrated crew pairing and personalized crew assignment problems. *Transportation Science* 53(5):1479–1499.

Zhang M, Kucukyavuz S (2014) Finitely convergent decomposition algorithms for two-stage stochastic pure integer programs. *SIAM Journal on Optimization* 24(4):1933–1951.

Zhang W, Jacquillat A, Wang K, Wang S (2023) Routing optimization with vehicle–customer coordination. *Management Science* .

Zhao J, Dessouky M (2008) Service capacity design problems for mobility allowance shuttle transit systems. *Transportation Research Part B: Methodological* 42(2):135–146.

A Double Decomposition Algorithm for Network Planning and Operations in Deviated Fixed-route Microtransit Electronic Companion

EC.1. Details on Model Formulations

EC.1.1. Notation Tables

Table EC.1 summarizes all notation for the MiND-VRP formulation.

EC.1.2. Compact Formulation for the Second-stage Problem

We define the following additional parameters:

$$\mathcal{N}_\ell^S = \text{set of stations for line } \ell\text{, within a distance } \Delta \text{ to the reference line}$$

$$\mathcal{N}_{\ell i}^- = \text{set of locations that can be visited immediately after location } i \text{ on line } \ell$$

$$\mathcal{N}_{\ell i}^+ = \text{set of locations that can be visited immediately before location } i \text{ on line } \ell$$

$$\mathcal{N}_{\ell tp}^{\text{pickup}} = \text{set of possible pickup locations for passenger } p \text{ on trip } (\ell, t)\text{, within } \Omega \text{ of their origin}$$

$$\tau_p^{\text{req}} = \text{requested time for passenger } p$$

$$\tau_{ip}^{\text{walk}} = \text{walking time from origin of passenger } p \text{ to pickup location } i$$

$$\tau_{\ell t}^{\text{dropoff}} = \text{dropoff time for reference trip } (\ell, t) \text{ at the destination}$$

We define the following decision variables:

$$v_{\ell tsi} = \begin{cases} 1 & \text{if reference trip } (\ell,t) \text{ visits checkpoint } i \in \mathcal{I}_\ell \text{ in scenario } s \in \mathcal{S}, \\ 0 & \text{otherwise.} \end{cases}$$

$$y_{\ell tsij} = \begin{cases} 1 & \text{reference trip } (\ell,t) \text{ travels from location } i \in \mathcal{N}_\ell^S \text{ to location } j \in \mathcal{N}_\ell^S \text{ in scenario } s \in \mathcal{S}, \\ 0 & \text{otherwise.} \end{cases}$$

$$w_{\ell tspi} = \begin{cases} 1 & \text{reference trip } (\ell,t) \text{ picks up passenger } p \in \mathcal{P} \text{ at location } i \in \mathcal{N}_\ell^S \text{ in scenario } s \in \mathcal{S}, \\ 0 & \text{otherwise.} \end{cases}$$

$$t_{\ell tsi}^{\text{stop}} = \text{time at which reference trip } (\ell,t) \text{ stops at location } i \in \mathcal{N}_\ell^S \text{ in scenario } s \in \mathcal{S}$$

$$t_{\ell tsp}^{\text{pickup}} = \text{time at which reference trip } (\ell,t) \text{ picks up passenger } p \in \mathcal{P} \text{ in scenario } s \in \mathcal{S}$$

The compact formulation of the second-stage problem in scenario $s \in \mathcal{S}$ and for reference trip (ℓ, t) is then given as follows:

$$\min \quad \sum_{p \in \mathcal{P}} D_{ps} \Bigg(\lambda \sum_{i \in \mathcal{N}_{\ell tp}^{\text{pickup}}} \tau_{ip}^{\text{walk}} w_{\ell tspi} + \mu \left(t_{\ell tsp}^{\text{pickup}} - \sum_{i \in \mathcal{N}_{\ell tp}^{\text{pickup}}} (\tau_p^{\text{req}} + \tau_{ip}^{\text{walk}}) w_{\ell tspi} \right)$$

$$+ \frac{\sigma}{\tau_p^{dir}} \left(\tau_{\ell t}^{\text{dropoff}} \sum_{i \in \mathcal{N}_{\ell tp}^{\text{pickup}}} w_{\ell tspi} - t_{\ell tsp}^{\text{pickup}} \right) + \left(\delta \frac{\tau_{\ell tp}^{\text{late}}}{\tau_p^{\text{dir}}} + \frac{\delta}{2} \frac{\tau_{\ell tp}^{\text{early}}}{\tau_p^{\text{dir}}} \right) \sum_{i \in \mathcal{N}_{\ell tp}^{\text{pickup}}} w_{\ell tspi}$$

Component	**Type**	**Description**
$\mathcal{N}^S$	Set	Stations: checkpoints and pickup locations
$\mathcal{E}$	Set	Directed arcs in $\mathcal{N} \times \mathcal{N}$ corresponding to roadways
$\mathcal{L}$	Set	Candidate reference lines
$\mathcal{P}$	Set	Passenger types
$\mathcal{S}$	Set	Demand scenarios
$\mathcal{C}_\ell$	Set	Vehicle loads on reference line $\ell \in \mathcal{L}$
$\mathcal{I}_\ell$	Set	Checkpoints for line $\ell \in \mathcal{L}$, of cardinality I_ℓ
$\mathcal{I}_\ell^{(i)}$	Set	i^{th} checkpoint in reference line $\ell \in \mathcal{L}$ for $i = 1, \cdots, I_\ell$
Γ_ℓ	Set	Subset of checkpoint pairs in $\mathcal{I}_\ell \times \mathcal{I}_\ell$ for line $\ell \in \mathcal{L}$ that skip up to K checkpoints in between
$\mathcal{N}_{uv}$	Set	Subset of nodes in $\mathcal{N}^S$ representing possible stations between checkpoints $u, v \in \mathcal{I}_\ell$ for each line $\ell \in \mathcal{L}$
$\mathcal{T}_\ell$	Set	Allowable departure times of a vehicle from the beginning of line $\ell \in \mathcal{L}$
$\mathcal{T}_{\ell t}^{uv}$	Set	Time intervals between the scheduled times $T_{\ell t}(u)$ and $T_{\ell t}(v)$ for checkpoint pair $(u, v) \in \Gamma_\ell$
$\mathcal{M}_p$	Set	Compatible trips in $\mathcal{L} \times \mathcal{T}_\ell$ for passenger type $p \in \mathcal{P}$
$\mathcal{R}_{\ell st}$	Set	Subpaths corresponding to reference trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$ in scenario $s \in \mathcal{S}$. Each subpath $r \in \mathcal{R}_{\ell st}$ starts in $u_r \in \mathcal{I}_\ell$ and ends in $v_r \in \mathcal{I}_\ell$.
$(\mathcal{V}_{\ell st}, \mathcal{A}_{\ell st})$	Graph	Load-expanded network of trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$ in scenario $s \in \mathcal{S}$. Every trip starts at $u_{\ell st} \in \mathcal{V}_{\ell st}$ and ends at $v_{\ell st} \in \mathcal{V}_{\ell st}$
$\mathcal{A}_r$	Set	Arcs in $\mathcal{A}_{\ell st}$ corresponding to subpath $r \in \mathcal{R}_{\ell st}$ for $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$, $s \in \mathcal{S}$
$\mathcal{A}_{\ell st}^v$	Set	Arcs in $\mathcal{A}_{\ell st}$ connecting line destination to sink node for $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$, $s \in \mathcal{S}$
$(\mathcal{U}_{\ell st}^{uv}, \mathcal{H}_{\ell st}^{uv})$	Graph	Time-expanded network from $(u, T_{\ell t}(u))$ to $(v, T_{\ell t}(v))$. Node $m \in \mathcal{U}_{\ell st}^{uv}$ is characterized by a location-time tuple (k_m, t_m)
$\mathcal{P}_m$	Set	Passengers in $\mathcal{P}$ that can be picked up in node $m \in \mathcal{U}_{\ell st}^{uv}$
$\mathcal{P}_r$	Set	Passenger types in $\mathcal{P}$ picked up by subpath $r \in \mathcal{R}_{\ell st}$ for $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$, $s \in \mathcal{S}$
K	Parameter	Number of consecutive checkpoints that can be skipped (0 or 1)
C_ℓ	Parameter	Vehicle capacity on reference line $\ell \in \mathcal{L}$
F	Parameter	Fleet size
h_l	Parameter	Cost to operate one trip via line $\ell \in \mathcal{L}$
D_{ps}	Parameter	Number of passengers of type $p \in \mathcal{P}$ in scenario $s \in \mathcal{S}$
$T_{\ell t}(n)$	Parameter	Time at which trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$ must visit checkpoint $n \in \mathcal{I}_\ell$
π_s	Parameter	Probability of scenario $s \in \mathcal{S}$
g_a	Parameter	Cost of arc $a \in \mathcal{A}_{\ell st}$ for trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$, scenario $s \in \mathcal{S}$ (Equation (EC.99))
Δ	Parameter	Maximum vehicle deviation from reference line
Ω	Parameter	Maximum walking distance for passengers
Ψ	Parameter	Maximum waiting time for passengers
α	Parameter	Time window radius around passengers' requested drop-off times to build $\mathcal{M}_p$
$\omega_{o,d}$	Parameter	Walking distance between locations o and d
$\psi_{o,d}$	Parameter	Walking time between locations o and d
τ_{rp}^{walk}	Parameter	Walk time of passenger $p \in \mathcal{P}_r$ via subpath $r \in \mathcal{R}_{\ell st}$, for $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$, $s \in \mathcal{S}$
τ_{rp}^{wait}	Parameter	Wait time of passenger $p \in \mathcal{P}_r$ via subpath $r \in \mathcal{R}_{\ell st}$, for $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$, $s \in \mathcal{S}$
$\tau_{rp}^{\text{travel}}$	Parameter	In-vehicle time of passenger $p \in \mathcal{P}_r$ via subpath $r \in \mathcal{R}_{\ell st}$, for $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$, $s \in \mathcal{S}$
$\tau_{\ell tp}^{\text{late}}$	Parameter	Delay of passenger type $p \in \mathcal{P}$ when taking trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$
$\tau_{\ell tp}^{\text{early}}$	Parameter	Earliness of passenger type $p \in \mathcal{P}$ when taking trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$
τ_p^{dir}	Parameter	Direct travel time for passenger type $p \in \mathcal{P}$
$tt(e)$	Parameter	Travel time corresponding to road segment $e \in \mathcal{E}$
τ_{mp}^{walk}	Parameter	Walk time of passenger $p \in \mathcal{P}_m$ when picked up at node $m \in \mathcal{U}_{\ell st}^{uv}$
τ_{mp}^{wait}	Parameter	Wait time of passenger $p \in \mathcal{P}_m$ when picked up at node $m \in \mathcal{U}_{\ell st}^{uv}$
$\tau_{mp}^{\text{travel}}$	Parameter	In-vehicle travel time of passenger $p \in \mathcal{P}_m$ when picked up at node $m \in \mathcal{U}_{\ell st}^{uv}$
M	Parameter	Reward for each passenger pickup
$\lambda, \mu, \sigma, \delta$	Parameters	Penalties on passenger walk time, wait time, detour, and displacement
κ	Parameter	Target vehicle load in the first-stage network design problem

Table EC.1 Notation for the MiND-VRP model and its decomposition.

$$-M \sum_{i \in \mathcal{N}_{\ell tp}^{\text{pickup}}} w_{\ell tspi} \Bigg) \tag{EC.1}$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{N}_{\ell i}^-} y_{\ell\ell tsij} - \sum_{j \in \mathcal{N}_{\ell i}^+} y_{\ell\ell tsji} = \begin{cases} x_{\ell t} & \text{if } i = o_\ell \\ -x_{\ell t} & \text{if } i = d_\ell \\ 0 & \text{otherwise.} \end{cases} \quad \forall i \in \mathcal{N}_\ell^S \tag{EC.2}$$

$$t_{\ell tsj}^{\text{stop}} \geq t_{\ell tsi}^{\text{stop}} + tt_{ij} - M^{\text{stop}}(1 - y_{\ell tsij}) \quad \forall i \in \mathcal{N}_\ell^S, j \in \mathcal{N}_{\ell i}^- \tag{EC.3}$$

$$\sum_{i \in \mathcal{N}_{\ell tp}^{\text{pickup}}} w_{\ell tspi} \leq z_{p\ell st} \quad \forall p \in \mathcal{P} \tag{EC.4}$$

$$\sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{N}_{\ell tp}^{\text{pickup}}} D_{ps} w_{\ell tspi} \leq C_\ell x_{\ell t} \tag{EC.5}$$

$$t_{\ell tsp}^{\text{pickup}} \geq t_{\ell tsi}^{\text{stop}} - M^{\text{stop}}(1 - w_{\ell tspi}) \quad \forall p \in \mathcal{P}, i \in \mathcal{N}_\ell^S \tag{EC.6}$$

$$w_{\ell tspi} \leq \sum_{j \in \mathcal{N}_{\ell i}^-} y_{\ell tsij} + \sum_{j \in \mathcal{N}_{\ell i}^+} y_{\ell tsji} \quad \forall p \in \mathcal{P}, i \in \mathcal{N}_{\ell tp}^{\text{pickup}} \tag{EC.7}$$

$$T_{\ell t}(i) - M^T(1 - v_{\ell tsi}) \leq t_{\ell tsi}^{\text{stop}} \leq T_{\ell t}(i) + M^T(1 - v_{\ell tsi}) \quad \forall i \in \mathcal{I}_\ell \tag{EC.8}$$

$$t_{\ell tsi}^{\text{stop}} \leq M^y \cdot \left(\sum_{j \in \mathcal{N}_{\ell i}^-} y_{\ell tsij} + \sum_{j \in \mathcal{N}_{\ell i}^+} y_{\ell tsji} \right) \quad \forall i \in \mathcal{N}_\ell^S \tag{EC.9}$$

$$\sum_{j \in \mathcal{N}_{\ell i}^-} y_{\ell tsij} + \sum_{j \in \mathcal{N}_{\ell i}^+} y_{\ell tsji} \geq v_{\ell tsi} \quad \forall i \in \mathcal{I}_\ell \tag{EC.10}$$

$$\sum_{j=i}^{i+K} v_{\ell ts\mathcal{I}_\ell^{(j)}} \geq x_{\ell t} \quad \forall i = 1, \ldots, I_\ell - K \tag{EC.11}$$

$$\boldsymbol{v}, \boldsymbol{y}, \boldsymbol{w} \text{ binary} \tag{EC.12}$$

$$\boldsymbol{t}^{\text{stop}}, \boldsymbol{t}^{\text{pickup}} \geq \boldsymbol{0} \tag{EC.13}$$

The objective (EC.1) minimizes the weighted sum of walking time, waiting time, relative detour, and relative delay across passengers, minus the reward for passenger pickups. Constraints (EC.2) enforce flow balance at each station. Constraints (EC.3) coordinate travel time between locations on each trip, with a big-M parameter M^{stop} activating them for pairs of consecutive locations. Constraints (EC.4) and (EC.5) ensure consistency with the first-stage solution, and Constraints (EC.5) enforce vehicle capacity. Constraints (EC.6) determine the pick-up time for each passenger as the vehicle stop time, again with big-M parameter M^{pickup} activating them for appropriate passenger-location pairs. Constraints (EC.7) ensure that passengers are only assigned to pick-up locations where the vehicle stops. Constraints (EC.8) ensure that the checkpoints on selected reference trips are visited at the pre-specified time, with big-M parameter M^T activating them only for visited checkpoints. Constraints (EC.9) define the stopping time only for those locations visited by the vehicles, again with a corresponding big-M parameter M^y. Constraints (EC.10) ensure consistency of the checkpoint stops with the routing variables, and Constraints (EC.11) ensure that the vehicle skips at most K consecutive checkpoints.

Note that the definition of the $t^{\text{stop}}_{\ell tsi}$ variables, along with Constraints (EC.3), impose the implicit condition that each reference trip visits each stop at most once in each scenario. In principle, we could relax that assumption by defining variables of the form $t^{\text{stop}}_{\ell tsi\kappa}$, where κ counts the number of times reference trip (ℓ,t) stops at location $i \in \mathcal{N}^S$ in scenario $s \in \mathcal{S}$. The formulation would be augmented with precedence constraints accordingly. We omit these details for simplicity.

This formulation features a polynomial number of decision variables and constraints, but suffers from a weak linear relaxation due to the several big-M constraints (Equations (EC.3), (EC.6), (EC.8), and (EC.9)). Results in Section 5.1 and in EC.6.1 show that, in turn, it features much more limited scalability than our network-based reformulation.

EC.1.3. Proof of Proposition 1

Consider scenario s and reference trip (ℓ,t). Recall that arc $a=(m,n) \in \mathcal{A}_{\ell st}$ corresponds to a subpath $r(a)$ defined between two checkpoints $u_r, v_r \in \mathcal{I}_\ell$. Each node $m \in \mathcal{V}_{\ell st}$ corresponds to a checkpoint-load pair (k_m, c_m) on the load-expanded network.

We introduce additional notation to describe the series of stops and passenger pickups for each subpath. We index the checkpoints by $\text{ind}(i)$, so that $\text{ind}(i) < \text{ind}(j)$ if checkpoint $i \in \mathcal{I}_\ell$ is visited earlier than checkpoint $j \in \mathcal{I}_\ell$ (if they are visited at all). The route of subpath r is defined by a sequence of H stops, indexed by $h = 1, \cdots, H$. Each stop h encodes a location-time pair $\text{Stops}(r) = \{(i^h, t^h) \text{ for } h = 1, \ldots, H\}$. For ease of notation, we also define the set of *pairs* of consecutive location-time pairs, $\text{Path}(r) = \{((i^h, t^h), (i^{h+1}, t^{h+1})), \forall h = 1, \ldots, H-1\}$. Finally, let $\text{Pax}(r,i)$ be the set of passengers picked up at location i on subpath r. Each subpath satisfies four conditions:

(i) The distance to the reference line never exceeds Δ

(ii) The load satisfies $\sum_{p \in \mathcal{P}_r} D_{ps} \leq C_\ell$

(iii) The travel time does not exceed $T_{\ell t}(v_r) - T_{\ell t}(u_r)$

(iv) Up to K checkpoints are skipped

Let $(\boldsymbol{v}^*, \boldsymbol{w}^*, \boldsymbol{y}^*, \boldsymbol{t}^{\text{stop}*}, \boldsymbol{t}^{\text{pickup}*})$ be an optimal solution to the compact formulation. We construct an equivalent feasible solution $\boldsymbol{y}$ for the subpath formulation with the same objective value. Let $\mathcal{B}_{ij}$ be the set of intermediate stops between visited checkpoints i and j. Specifically, set $\mathcal{B}_{ij} = \emptyset$ for $i, j \in \mathcal{I}_\ell$ such that $v^*_{\ell tsi} = v^*_{\ell tsj} = 1$ and $v^*_{\ell tsk} = 0$ for all $k \in \mathcal{I}_\ell$ such that $\text{ind}(i) < \text{ind}(k) < \text{ind}(j)$, i.e., if i and j are consecutive checkpoints in the solution. Then, construct $\mathcal{B}$ iteratively by starting from $i = k_m$, then identifying j such that $y^*_{\ell tsij} = 1$ and setting $\mathcal{B} \leftarrow \mathcal{B} \cup \{j\}$. Continue until $j = k_n$.

We construct $\boldsymbol{y}^*$ by setting $y^*_a = 1$ for $a = (m,n) \in \mathcal{A}_{\ell st}$ if: (1) $v^*_{\ell tsi} = 1$ for $i = k_m$ and $i = k_n$; (2) $v^*_{\ell tsi} = 0$ for all $i \in \mathcal{I}_\ell$ such that $\text{ind}(k_m) < \text{ind}(i) < \text{ind}(k_n)$; (3) $\text{Pax}(r(a), i) = \{p \in \mathcal{P} : w^*_{\ell tspi} = 1\}$ for $i \in \mathcal{B}_{k_m,k_n}$; and (4) the load c_m is equal to the total number of passengers picked up along subpaths prior to node m: $c_m = \sum_{a'=(m',n')\in\mathcal{A}_{\ell st}\setminus\{a\}:c_{n'}\leq c_m} \sum_{p\in\mathcal{P}_{r(a')}} D_{ps}y_a$ In particular, condition (3) implies that $\mathcal{P}_{r(a)} = \{p\in\mathcal{P}: \sum_{i\in\mathcal{B}_{k_m,k_n}} w^*_{\ell tspi} = 1\}$. Otherwise, $y_a = 0$.

We first show that for each a with $y_a = 1$, $r(a)$ is a valid subpath.

Condition (i). The distance to the reference line never exceeds Δ by construction, since we only consider $i\in\mathcal{N}^S_\ell$.

Condition (ii). By construction of $\mathcal{P}_{r(a)}$ and (EC.5),

$$\sum_{p\in\mathcal{P}_{r(a)}} D_{ps} = \sum_{p\in\mathcal{P}_{r(a)}} \sum_{i\in\mathcal{N}^{\text{pickup}}_{\ell tp}} D_{ps}w^*_{\ell tspi} \leq \sum_{p\in\mathcal{P}} \sum_{i\in\mathcal{N}^{\text{pickup}}_{\ell tp}} D_{ps}w^*_{\ell tspi} \leq C_\ell$$

Condition (iii). Note that $t^{\text{stop}*}_{\ell tsk_m} = T_{\ell t}(k_m)$ and $t^{\text{stop}*}_{\ell tsk_n} = T_{\ell t}(k_n)$ by (EC.8). By summing over (EC.3) for all $i,j\in\mathcal{B}_{k_mk_n}$ such that $y^*_{\ell stij} = 1$, the travel time does not exceed $T_{\ell t}(k_n) - T_{\ell t}(k_m)$:

$$\begin{aligned}
\sum_{i\in\mathcal{B}_{k_mk_n}} \sum_{j\in\mathcal{B}_{k_mk_n}:y^*_{\ell stij}=1} t^{\text{stop}*}_{\ell stj} \geq \sum_{i\in\mathcal{B}_{k_mk_n}} \sum_{j\in\mathcal{B}_{k_mk_n}:y^*_{\ell stij}=1} &(t^{\text{stop}*}_{\ell sti} + tt_{ij}) \\
t^{\text{stop}*}_{\ell stk_n} \geq t^{\text{stop}*}_{\ell stk_m} + \sum_{i\in\mathcal{B}_{k_mk_n}} \sum_{j\in\mathcal{B}_{k_mk_n}:y^*_{\ell stij}=1} &tt_{ij} \\
T_{\ell t}(k_n) \geq T_{\ell t}(k_m) + \sum_{i\in\mathcal{B}_{k_mk_n}} \sum_{j\in\mathcal{B}_{k_mk_n}:y^*_{\ell stij}=1} &tt_{ij}
\end{aligned}$$

Condition (iv). By construction, $v^*_{\ell tsk_m} = 1$, $v^*_{\ell tsk_n} = 1$, and $v^*_{\ell tsi} = 0$ for all $i\in\mathcal{I}_\ell$ such that $\text{ind}(k_m) < \text{ind}(i) < \text{ind}(k_n)$. By Constraints (EC.11), $\sum_{i=\text{ind}(k_m)+1}^{\text{ind}(k_m)+K+1} v^*_{\ell t\mathcal{I}^{(i)}_\ell} \geq 1$ and since $\text{ind}(k_n)$ is the next checkpoint index i for which $v^*_{\ell tsi} = 1$, we must have $\text{ind}(k_n) \leq \text{ind}(k_m) + K + 1$. Therefore, at most K checkpoints are skipped between k_m and k_n.

Finally, we prove that $\boldsymbol{y}$ is feasible for the subpath formulation and achieves the same objective as $(\boldsymbol{v}^*, \boldsymbol{w}^*, \boldsymbol{y}^*, \boldsymbol{t}^{\text{stop}*}, \boldsymbol{t}^{\text{pickup}*})$. Denote the objective of the compact formulation as $\texttt{OPT}$:

$$\begin{aligned}
\texttt{OPT} = &\sum_{p\in\mathcal{P}} \sum_{i\in\mathcal{N}^{\text{pickup}}_{\ell tp}} D_{ps} \left(\frac{\delta\tau^{\text{late}}_{\ell tp} + \frac{\delta}{2}\tau^{\text{early}}_{\ell tp} + \sigma\tau^{\text{dropoff}}_{\ell t}}{\tau^{\text{dir}}_p} + \lambda\tau^{\text{walk}}_{ip} + \mu(-\tau^{\text{req}}_p - \tau^{\text{walk}}_{ip}) - M \right) \cdot w^*_{\ell tspi} \\
&+ \sum_{p\in\mathcal{P}} D_{ps} \left(\frac{-\sigma}{\tau^{\text{dir}}_p} + \mu \right) \cdot t^{\text{pickup}*}_{\ell tsp}
\end{aligned}$$

By Constraints (EC.4) and by condition (3) in the construction of subpath $a = (m,n)$:

$$w^*_{\ell tspi} = \sum_{a\in\mathcal{A}_{\ell st}:\text{Pax}(r(a),i)} y_a \quad \forall p\in\mathcal{P}, i\in\mathcal{N}^{\text{pickup}}_{\ell tp} \tag{EC.14}$$

Furthermore, $t^{\text{pickup}*}_{\ell tsp}$ can be written as $\sum_{i\in\mathcal{N}^{\text{pickup}}_{\ell tp}}\sum_{a\in\mathcal{A}_{\ell st}:p\in\text{Pax}(r(a),i)}\tau^{\text{stop}}_i(a)\cdot y_a$ where $\tau^{\text{stop}}_i(a)$ is the time arc a stops at location i. Then,

$$\begin{aligned}
\texttt{OPT} = & \sum_{p\in\mathcal{P}}\sum_{i\in\mathcal{N}^{\text{pickup}}_{\ell tp}}\sum_{a\in\mathcal{A}_{\ell st}:\text{Pax}(r(a),i)} D_{ps}\left(\frac{\delta\tau^{\text{late}}_{\ell tp}+\frac{\delta}{2}\tau^{\text{early}}_{\ell tp}+\sigma\tau^{\text{dropoff}}_{\ell t}}{\tau^{\text{dir}}_p}+\lambda\tau^{\text{walk}}_{ip}+\mu(-\tau^{\text{req}}_p-\tau^{\text{walk}}_{ip})-M\right)\cdot y_a\\
&+\sum_{p\in\mathcal{P}}\sum_{i\in\mathcal{N}^{\text{pickup}}_{\ell tp}}\sum_{a\in\mathcal{A}_{\ell st}:\text{Pax}(r(a),i)} D_{ps}\left(\frac{-\sigma\tau^{\text{stop}}_i(a)}{\tau^{\text{dir}}_p}+\mu\tau^{\text{stop}}_i(a)\right)\cdot y_a\\
= & \sum_{a\in\mathcal{A}_{\ell st}}\sum_{p\in\mathcal{P}_{r(a)}} D_{ps}\left(\frac{\delta\tau^{\text{late}}_{\ell tp}+\frac{\delta}{2}\tau^{\text{early}}_{\ell tp}+\sigma(\tau^{\text{dropoff}}_{\ell t}-\tau^{\text{stop}}_{m_{ap}}(a))}{\tau^{\text{dir}}_p}+\lambda\tau^{\text{walk}}_{m_{ap},p}+\mu(\tau^{\text{stop}}_{m_{ap}}(a)-\tau^{\text{req}}_p-\tau^{\text{walk}}_{m_{ap},p})-M\right)\cdot y_a,
\end{aligned}$$

where $m_{ap}\in\mathcal{N}^{\text{pickup}}_{\ell tp}$ denotes the stop on subpath $r(a)$ where passenger p is picked up.

The objective of the subpath formulation relies on two preprocessed parameters, τ^{wait}_{mp} and $\tau^{\text{travel}}_{mp}$ (each dependent on arc a), which represent the wait time and in-vehicle time if passenger p is picked up at location m, respectively. We can re-write them as $\tau^{\text{wait}}_{mp}(a)=\tau^{\text{stop}}_m(a)-\tau^{\text{req}}_p-\tau^{\text{walk}}_{mp}$ and $\tau^{\text{travel}}_{mp}(a)=\tau^{\text{dropoff}}_{\ell t}-\tau^{\text{stop}}_m(a)$. We get:

$$\begin{aligned}
\texttt{OPT} &= \sum_{a\in\mathcal{A}_{\ell st}}\sum_{p\in\mathcal{P}_{r(a)}} D_{ps}\left(\frac{\delta\tau^{\text{late}}_{\ell tp}+\frac{\delta}{2}\tau^{\text{early}}_{\ell tp}+\sigma\tau^{\text{travel}}_{m_{ap},p}}{\tau^{\text{dir}}_p}+\lambda\tau^{\text{walk}}_{m_{ap},p}+\mu\tau^{\text{wait}}_{m_{ap},p}-M\right)\cdot y_a\\
&= \sum_{a\in\mathcal{A}_{\ell st}} g_a y_a
\end{aligned}$$

Conversely, let $\boldsymbol{y}^*$ be an optimal solution to the subpath formulation. We assume that, under that solution, each reference trip visits each stop at most once in each scenario. As discussed above, this assumption is embedded in the compact formulation itself. All arguments could be extended otherwise, but we omit these details for simplicity. In practice, it is highly unlikely that a reference trip would find it beneficial to stop multiple times in the same location.

We construct a feasible solution $(\boldsymbol{v},\boldsymbol{w},\boldsymbol{y},\boldsymbol{t}^{\text{stop}},\boldsymbol{t}^{\text{pickup}})$ for the compact formulation that corresponds to subpath solution $\boldsymbol{y}^*$ and has the same objective value. Specifically:

$$v_{\ell tsi} = \sum_{a\in\mathcal{A}_{\ell st}|\exists t':(i,t')\in\text{Stops}(r(a))} y^*_a \tag{EC.15}$$

$$y_{\ell tsij} = \sum_{a\in\mathcal{A}_{\ell st}|\exists t_1,t_2:((i,t_1),(j,t_2))\in\text{Paths}(r(a))} y^*_a \tag{EC.16}$$

$$w_{\ell tspi} = \sum_{a\in\mathcal{A}_{\ell st}:p\in\text{Pax}(r(a),i)} y^*_a \tag{EC.17}$$

$$t^{\text{stop}}_{\ell tsi} = \sum_{a\in\mathcal{A}_{\ell st}|\exists t':(i,t')\in\text{Stops}(r(a))} \tau^{\text{stop}}_i(a)y^*_a \tag{EC.18}$$

$$t^{\text{pickup}}_{\ell tsp} = \sum_{a\in\mathcal{A}_{\ell st}:p\in\mathcal{P}_{r(a)}} \tau^{\text{stop}}_{m_{ap}}(a)y^*_a \tag{EC.19}$$

From (EC.17), we have:

$$\sum_{i \in \mathcal{N}_{\ell tp}^{\text{pickup}}} w_{\ell tspi} = \sum_{a \in \mathcal{A}_{\ell st}: p \in \mathcal{P}_{r(a)}} y_a^* \quad \text{for } p \in \mathcal{P} \tag{EC.20}$$

$$\sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{N}_{\ell tp}^{\text{pickup}}} w_{\ell tspi} = \sum_{a \in \mathcal{A}_{\ell st}} \sum_{p \in \mathcal{P}_{r(a)}} y_a^* \tag{EC.21}$$

We show that $(\boldsymbol{v}, \boldsymbol{w}, \boldsymbol{y}, \boldsymbol{t}^{\text{stop}}, \boldsymbol{t}^{\text{pickup}})$ satisfies all constraints of the compact formulation.

Constraints (EC.2). If $x_{\ell t} = 0$, then $y_a^* = 0$ for all $a \in \mathcal{A}_{\ell st}$, and $y_{\ell tsij} = 0$ for all $i, j \in \mathcal{N}_\ell^S$, satisfying the constraint. If $x_{\ell t} = 1$, then we have four cases.

(1) $i = \mathcal{I}_\ell^{(1)}$: By flow balance constraints (9), $\sum_{m:(n,m) \in \mathcal{A}_{\ell st}} y^*_{(u_{\ell st}, m)} = 1$, so there exists exactly one $m = (j, c)$ such that $y^*_{(u_{\ell st}, m)} = 1$. Note that $u_{\ell st} = (\mathcal{I}_\ell^{(1)}, T_{\ell t}(\mathcal{I}_\ell^{(1)}))$. Thus, $y_{\ell tsij} = 1$ and $y_{\ell tsik} = 0$ for all $k \in \mathcal{N}_{\ell i}^-, k \neq j$.

(2) $i = \mathcal{I}_\ell^{(I_\ell)}$: By flow balance constraints (9), $\sum_{m:(m,n) \in \mathcal{A}_{\ell st}} y^*_{(m, v_{\ell st})} = 1$, so there exists exactly one node $m^* = (j, c)$ such that $y^*_{(m^*, v_{\ell st})} = 1$. Recall that $v_{\ell st}$ is a dummy sink node and thus $j = \mathcal{I}_\ell^{(I_\ell)}$. Again by (9) for $n = (j, c)$, we have:

$$\sum_{m:(m^*,m) \in \mathcal{A}_{\ell st}} y^*_{(m^*,m)} - \sum_{m:(m,m^*) \in \mathcal{A}_{\ell st}} y^*_{(m,m^*)} = 0$$
$$\implies \sum_{m:(m,m^*) \in \mathcal{A}_{\ell st}} y^*_{(m,m^*)} = 1$$

We can then conclude that for exactly one node $m = (k, c')$, we have $y^*_{(m,m^*)} = 1$. Therefore, $y_{\ell, t, s, k, \mathcal{I}_\ell^{(I_\ell)}} = 1$ for exactly one $k \in \mathcal{N}^+_{\mathcal{I}_\ell^{(I_\ell)}}$ and the constraint is satisfied.

(3) $i = \mathcal{I}_\ell^{(2)}, \ldots, \mathcal{I}_\ell^{(I_\ell - 1)}$: Let $a = (m, n) = ((i, c_m), (k_n, c_n))$ and suppose $y_a^* = 1$. By flow balance constraints (9), $\exists a' = (o, m) = ((k_o, c_o), (i, c_m))$ such that $y_{a'}^* = 1$.

$$y_a^* = 1 \implies ((i, t_1), (j, t_2)) \in \text{Paths}(r(a)) \implies y_{\ell tsij} = 1$$
$$y_{a'}^* = 1 \implies ((j', t_3), (i, t_4)) \in \text{Paths}(r(a')) \implies y_{\ell tsj'i} = 1$$

Thus, Constraints (EC.2) are satisfied

(4) $i \notin \mathcal{I}_\ell$: Assume $(i, t_1) \in \text{Stops}(r(a))$ for $a \in \mathcal{A}_{\ell st}$ with $y_a^* = 1$. Then, (i, t_1) will appear exactly twice as the end point of an arc in $\text{Paths}(r(a))$: $((i, t_1), (j, t_2))$ for some j and $((j', t_3), (i, t_1))$ for some j'. By construction, we then have $y_{\ell tsij} = y_{\ell tsj'i} = 1$ and flow balance is satisfied. If $(i, t_1) \notin \text{Stops}(r(a))$ for any t_1, then $y_{\ell tsij} = 0$ for all $j \in \mathcal{N}_\ell^S$, again satisfying flow balance.

Constraints (EC.3). For $i \in \mathcal{N}_\ell^S, j \in \mathcal{N}_{\ell i}^i$ such that $y_{\ell tsij} = 0$, Constraints (EC.3) is trivially satisfied. For i, j with $y_{\ell tsij} = 1$, then $\exists t_1, t_2$ such that $((i, t_1), (j, t_2)) \in \text{Path}(r)$, and by construction of the subpaths, $t_2 = t_1 + tt_{ij}$. Thus, Constraints (EC.3) is satisfied for i, j.

Constraints (EC.4). By Equation (EC.20) and (10):

$$\sum_{i\in\mathcal{N}^{\text{pickup}}_{\ell tp}} w_{\ell tspi} = \sum_{a\in\mathcal{A}_{\ell st}:p\in\mathcal{P}_{r(a)}} y^*_a \le z_{p\ell st} \quad \forall s\in\mathcal{S}, p\in\mathcal{P}, (\ell,t)\in\mathcal{M}_p$$

Constraints (EC.5). If $x_{\ell t}=0$, then $y^*_a=0$ for all $a\in\mathcal{A}_{\ell st}$, and $w_{\ell tspi}=0$ for all $p\in\mathcal{P}, i\in\mathcal{N}^S_\ell$, satisfying the constraint. If $x_{\ell t}=1$, by Equation (EC.21) we have:

$$\sum_{p\in\mathcal{P}}\sum_{i\in\mathcal{N}^{\text{pickup}}_{\ell tp}} D_{ps}w_{\ell tspi} = \sum_{a\in\mathcal{A}_{\ell st}}\sum_{p\in\mathcal{P}_{r(a)}} D_{ps}y^*_a \le C_\ell$$

The last inequality comes from the maximum load of a trip, which is given by c_m (and $c_m\le C_\ell$) for m such that $a=(m,v_{\ell st})\in\mathcal{A}^v_{\ell st}$ and $y^*_a=1$. By construction of the load-expanded network, c_m is equal to the sum of the load differentials of all subpaths before it, namely $\sum_{a\in\mathcal{A}_{\ell st}}\sum_{p\in\mathcal{P}_{r(a)}} D_{ps}y^*_a$.

Constraints (EC.6). If $w_{\ell tspi}=0$ for $p\in\mathcal{P}, i\in\mathcal{N}^S_\ell$, then Constraints (EC.6) is trivially satisfied. If $w_{\ell tspi}=1$, then $\exists a\in\mathcal{A}_{\ell st}$ such that $p\in\text{Pax}(r(a),i)$ and $y^*_a=1$. If $t^{\text{stop}}_{\ell tsi}=0$, then Constraints (EC.6) is clearly satisfied. If $t^{\text{stop}}_{\ell tsi}=t'>0$, then by construction $(i,t')\in\text{Stops}(r(a'))$ for some $a'\in\mathcal{A}_{\ell st}$. Since each station is visited at most once, $a=a'$ and $\exists i\in\mathcal{N}^+_\ell, a\in\mathcal{A}_{\ell st}$ such that $(i,t')\in\text{Stops}(r(a))$, $p\in\text{Pax}(r(a),i)$, and $y^*_a=1$. Thus, $t^{\text{pickup}}_{\ell tsi}=t'$, satisfying Constraints (EC.6).

Constraints (EC.7). If $w_{\ell tspi}=0$, the constraints are trivially satisfied. If $w_{\ell tspi}=1$, by construction, $\exists a\in\mathcal{A}_{\ell st}$ such that $p\in\text{Pax}(r(a),i)$ and $y^*_a=1$. Therefore, $i\in\text{Stops}(r(a))$ and either $((i,t_1),(j,t_2))\in\text{Paths}(r(a))$ or $((j,t_1),(i,t_2))\in\text{Paths}(r(a))$ for some $j\in\mathcal{N}^S_\ell$ and $t_1,t_2\ge 0$. We then conclude that Constraints (EC.7) are satisfied:

$$\sum_{j\in\mathcal{N}^-_{\ell i}} y_{\ell tsij} + \sum_{j\in\mathcal{N}^+_{\ell i}} y_{\ell tsji} \ge 1 = w_{\ell tspi}$$

Constraints (EC.8). For all $i\in\mathcal{I}_\ell$ such that $v_{\ell tsi}=0$, (EC.8) is trivially satisfied. For $i\in\mathcal{I}_\ell$ such that $v_{\ell tsi}=1$, $\exists a\in\mathcal{A}_{\ell st}, t'\in\mathbb{R}$ such that $(i,t')\in\text{Stops}(r(a))$ and $y^*_a=1$ by (EC.15). Note that $\tau^{\text{stop}}_i(a)=T_{\ell t}(i)$ for $i\in\mathcal{I}_\ell$. Thus, we have $t^{\text{stop}}_{\ell tsi}=T_{\ell t}(i)$, satisfying (EC.8).

Constraints (EC.9). If $t^{\text{stop}}_{\ell tsi}=0$, the constraint is trivially satisfied. If $t^{\text{stop}}_{\ell tsi}>0$ for $i\ne v_{r(a)}$, then $\exists j,t_1,t_2$ such that $((i,t_1),(j,t_2))\in\text{Paths}(r(a))$ and $y^*_a=1$, thus $y_{\ell tsij}=1$. If $t^{\text{stop}}_{\ell tsi}>0$ for $i=v_{r(a)}$, then $\exists j,t_1,t_2$ such that $((j,t_1),(i,t_2))\in\text{Paths}(r(a))$ and $y^*_a=1$, thus $y_{\ell tsji}=1$. Thus, the constraint is satisfied.

Constraints (EC.10). If $v_{\ell tsi}=0$, then the constraint is trivially satisfied. If $v_{\ell tsi}=1$ for $i\in\mathcal{I}_\ell$, then $\exists t'$ such that $(i,t')\in\text{Stops}(r(a))$ and $y^*_a=1$. Because $i\in\mathcal{I}_\ell$, checkpoint i must be either the first or last stop for subpath $r(a)$ for $a=(m,n)$ (i.e. $k_m=i$ or $k_n=i$). If $k_m=i$, then $\exists j,t_1,t_2$ such that $((i,t_1),(j,t_2)\in\text{Path}(r(a))$ and by construction $y_{\ell tsij}=1$ for $j\in\mathcal{N}^-_{\ell i}$. If $k_n=i$, then $\exists j,t_1,t_2$

such that $((j,t_1),(i,t_2) \in \mathrm{Path}(r(a))$ and by construction $y_{\ell tsji} = 1$ for $j \in \mathcal{N}^+_{\ell i}$. Thus, the constraint is satisfied.

Constraints (EC.11). With a slight abuse of notation, we use $i \in \{1, \cdots, I_\ell - K\}$ to denote the index of a given checkpoint as well as the checkpoint itself. Due to flow balance constraints and the construction of the subpaths, all checkpoints in $\mathcal{I}_\ell$ are either endpoints of a subpath $r(a)$ with $y^*_a = 1$ or are skipped by some subpath $r(a')$ with $y^*_{a'} = 1$. In the first case, let $a = (m,n) \in \mathcal{A}_{\ell st}$ with either $k_m = i$ or $k_n = i$, and $y^*_a = 1$. Then, $v_{\ell sti} = 1$ by construction and Constraints (EC.11) are satisfied. In the second case, there exist separate checkpoints $i^-, i^+ \in \mathcal{I}_\ell$, such that $i^- < i < i^+$ and $y^*_a = 1$ with $a = (m,n)$ and $k_m = i^-$ and $k_n = i^+$. By property (iv), subpath $r(a)$ skips at most K checkpoints between i^- and i^+. Thus, $i^+ \leq i^- + K + 1$, hence $i^+ \leq i + K$. This implies the constraint:

$$\sum_{i'=i}^{i+K} v_{\ell ts\mathcal{I}_\ell^{(i')}} \geq 1$$

Objective (EC.1). Denote the objective of the compact formulation as $\mathtt{OPT}$. Using Equation (EC.21) and the definition of $t^{\text{pickup}}_{\ell tsp}$ (EC.19), we have:

$$\begin{aligned}
\mathtt{OPT} = &\sum_{a \in \mathcal{A}_{\ell st}} \sum_{p \in \mathcal{P}_{r(a)}} D_{ps} \left(\frac{\delta \tau^{\text{late}}_{\ell tp} + \frac{\delta}{2} \tau^{\text{early}}_{\ell tp} + \sigma \tau^{\text{travel}}_{m_{ap},p}}{\tau^{\text{dir}}_p} + \lambda \tau^{\text{walk}}_{m_{ap},p} + \mu \tau^{\text{wait}}_{m_{ap},p} - M \right) \cdot y^*_a \\
= &\sum_{a \in \mathcal{A}_{\ell st}} \sum_{p \in \mathcal{P}_{r(a)}} D_{ps} \frac{\delta \tau^{\text{late}}_{\ell tp} + \frac{\delta}{2} \tau^{\text{early}}_{\ell tp} + \sigma (\tau^{\text{dropoff}}_{\ell t} - \tau^{\text{stop}}_{m_{ap}}(a))}{\tau^{\text{dir}}_p} \cdot y^*_a \\
+ &\sum_{a \in \mathcal{A}_{\ell st}} \sum_{p \in \mathcal{P}_{r(a)}} \left(\lambda \tau^{\text{walk}}_{m_{ap},p} + \mu (\tau^{\text{stop}}_{m_{ap}}(a) - \tau^{\text{req}}_p - \tau^{\text{walk}}_{m_{ap},p}) - M \right) \cdot y^*_a \\
= &\sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{N}^{\text{pickup}}_{\ell tp}} D_{ps} \left(\frac{\delta \tau^{\text{late}}_{\ell tp} + \frac{\delta}{2} \tau^{\text{early}}_{\ell tp} + \sigma \tau^{\text{dropoff}}_{\ell t}}{\tau^{\text{dir}}_p} + \lambda \tau^{\text{walk}}_{ip} + \mu (-\tau^{\text{req}}_p - \tau^{\text{walk}}_{ip}) - M \right) \cdot w_{\ell tspi} \\
&+ \sum_{p \in \mathcal{P}} D_{ps} \left(\frac{-\sigma t^{\text{pickup}}_{\ell tsp}}{\tau^{\text{dir}}_p} + \mu t^{\text{pickup}}_{\ell tsp} \right) \\
= &\sum_{p \in \mathcal{P}} D_{ps} \Bigg(\lambda \sum_{i \in \mathcal{N}^{\text{pickup}}_{\ell tp}} \tau^{\text{walk}}_{ip} w_{\ell tspi} + \mu \left(t^{\text{pickup}}_{\ell tsp} - \sum_{i \in \mathcal{N}^{\text{pickup}}_{\ell tp}} (\tau^{\text{req}}_p + \tau^{\text{walk}}_{ip}) w_{\ell tspi} \right) \\
&+ \frac{\sigma}{\tau^{\text{dir}}_p} \left(\tau^{\text{dropoff}}_{\ell t} \sum_{i \in \mathcal{N}^{\text{pickup}}_{\ell tp}} w_{\ell tspi} - t^{\text{pickup}}_{\ell tsp} \right) + \left(\delta \frac{\tau^{\text{late}}_{\ell tp}}{\tau^{dir}_p} + \frac{\delta}{2} \frac{\tau^{early}_{\ell tp}}{\tau^{\text{dir}}_p} \right) \sum_{i \in \mathcal{N}^{\text{pickup}}_{\ell tp}} w_{\ell tspi} \\
&- M \sum_{i \in \mathcal{N}^{\text{pickup}}_{\ell tp}} w_{\ell tspi} \Bigg)
\end{aligned}$$

Thus $(\boldsymbol{v}, \boldsymbol{w}, \boldsymbol{y}, \boldsymbol{t}^{\text{stop}}, \boldsymbol{t}^{\text{pickup}})$ is a feasible solution to the compact formulation and achieves the same objective value as $\boldsymbol{y}^*$ in the subpath formulation. This completes the proof. □

EC.1.4. Segment-based benchmark for second-stage problem

Component	Type	Description
$\overline{\mathcal{E}}_{\ell st}$	Set	Load-augmented road segments e associated with $road(e) \in \mathcal{E}$
$\mathcal{T}^S$	Set	Set of time periods during the planning horizon
$\mathcal{P}_e$	Set	Passengers picked up on segment $e \in \overline{\mathcal{E}}_{\ell st}$
$(\overline{\mathcal{V}}_{\ell st}, \overline{\mathcal{A}}_{\ell st})$	Graph	Time-load-expanded road network of trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$ in scenario $s \in \mathcal{S}$
$\overline{\mathcal{A}}_e$	Set	Arcs in $\overline{\mathcal{A}}_{\ell st}$ corresponding to segment $e \in \overline{\mathcal{E}}_{\ell st}$ for $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$, $s \in \mathcal{S}$
$\overline{\mathcal{A}}_{\ell st}^{idle}$	Set	Arcs in $\overline{\mathcal{A}}_{\ell st}$ representing an idling vehicle
$\overline{\mathcal{A}}_{\ell st}^{v}$	Set	Arcs in $\overline{\mathcal{A}}_{\ell st}$ connecting the line's destination to the dummy sink node
τ_{ep}^{walk}	Parameter	Walk time of passenger $p \in \mathcal{P}_e$ via segment $e \in \overline{\mathcal{E}}_{\ell st}$, $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$, $s \in \mathcal{S}$
τ_{ep}^{wait}	Parameter	Wait time of passenger $p \in \mathcal{P}_e$ via segment $e \in \overline{\mathcal{E}}_{\ell st}$, $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$, $s \in \mathcal{S}$
$\tau_{ep}^{\text{travel}}$	Parameter	In-vehicle time of passenger $p \in \mathcal{P}_r$ via segment $e \in \overline{\mathcal{E}}_{\ell st}$, $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$, $s \in \mathcal{S}$
$\overline{g}_a$	Parameter	Cost of arc $a \in \overline{\mathcal{A}}_{\ell st}$ on trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$ in scenario $s \in \mathcal{S}$

Table EC.2 Additional inputs of the segment-based formulation.

Throughout the section, we fix first-stage decisions $\boldsymbol{x}$ and $\boldsymbol{z}$, as well as scenario $s \in \mathcal{S}$. The time horizon is discretized into $T_S + 1$ intervals in the set $\mathcal{T}^S = \{0, 1, \cdots, T_S\}$, from the departure of the first trip $(t = 0)$ to the arrival of the last trip $(t = T_S)$.

To capture time and capacity constraints without relying on big-M constraints—therefore retaining a tight second-stage formulation—we build a time-load-expanded network $(\overline{\mathcal{V}}_{\ell st}, \overline{\mathcal{A}}_{\ell st})$. A dummy sink node $v_{\ell st}$ represents the end of a trip. Each other node $n \in \overline{\mathcal{V}}_{\ell st}$ is associated with a tuple (k_n, c_n, t_n), so that node n represents a vehicle's arrival to station $k_n \in \mathcal{N}^S$ at time $t_n \in \mathcal{T}^S$ with $c_n \in \mathcal{C}$ passengers. The source node is denoted by $u_{\ell st} := (\mathcal{I}_\ell^{(1)}, 0, T_{\ell t}(\mathcal{I}_\ell^{(1)}))$. We decompose the arc set $\overline{\mathcal{A}}_{\ell st} \subset \overline{\mathcal{V}}_{\ell st} \times \overline{\mathcal{V}}_{\ell st}$ into traveling arcs, idling arcs, and terminating arcs, by writing $\overline{\mathcal{A}}_{\ell st} = \bigcup_{e \in \overline{\mathcal{E}}_{\ell st}} \overline{\mathcal{A}}_e \cup \overline{\mathcal{A}}_{\ell st}^{idle} \cup \overline{\mathcal{A}}_{\ell st}^{v}$.

To characterize traveling arcs, we denote by $\overline{\mathcal{E}}_{\ell st}$ the set of possible roadways and passenger pickups. Specifically, each segment $e \in \overline{\mathcal{E}}_{\ell st}$ is associated with a raodway $road(e) \in \mathcal{E}$ and a set of passengers $\mathcal{P}_e$ who are picked up. We define traveling arcs by duplicating $e \in \overline{\mathcal{E}}_{\ell st}$ for all load pairs that correspond to the passenger pickups, and all time pairs that correspond to the travel time:

$$\overline{\mathcal{A}}_e = \left\{ \begin{aligned} (n, m) \in \overline{\mathcal{V}}_{lst} \times \overline{\mathcal{V}}_{lst} : (k_n, k_m) = road(e), \\ c_m - c_n = \sum_{p \in \mathcal{P}_e} D_{ps}, \\ t_m - t_n = tt(road(e)) \end{aligned} \right\} \qquad \forall e \in \overline{\mathcal{E}}_{\ell st} \tag{EC.22}$$

Next, each idling arc in $\overline{\mathcal{A}}_{\ell st}^{idle}$ connects nodes corresponding to two consecutive time intervals at the same station:

$$\overline{\mathcal{A}}_{\ell st}^{idle} = \{(n, m) \in \overline{\mathcal{V}}_{\ell st} \times \overline{\mathcal{V}}_{\ell st} : k_n = k_m,\ c_n = c_m,\ t_m - t_n = 1\}. \tag{EC.23}$$

Finally, each terminating arc in $\overline{\mathcal{A}}^v_{\ell st}$ connects the line's destination to the dummy sink node:

$$\overline{\mathcal{A}}^v_{\ell st} = \{(n,m) \in \overline{\mathcal{V}}_{\ell st} \times \overline{\mathcal{V}}_{\ell st} : k_n = \mathcal{I}^{\text{end}}, m = v^S_{\ell st}\}. \tag{EC.24}$$

Again, we can prune the time-load-expanded network by excluding disconnected nodes and all incident arcs. We define a segment-based cost $\overline{g}_a$ for each $a \in \overline{\mathcal{A}}_{\ell st}$ analogously to Equation (EC.99) to capture passenger walking times, waiting times, and relative arrival delays:

$$\overline{g}_a = \begin{cases} \sum_{p\in\mathcal{P}_e} D_{ps}\left(\lambda\tau^{\text{walk}}_{ep} + \mu\tau^{\text{wait}}_{ep} + \sigma\frac{\tau^{\text{travel}}_{ep}}{\tau^{\text{dir}}_p} + \delta\frac{\tau^{\text{late}}_{\ell tp}}{\tau^{\text{dir}}_p} + \frac{\delta}{2}\frac{\tau^{\text{early}}_{\ell tp}}{\tau^{\text{dir}}_p} - M\right) & \text{if } e \in \overline{\mathcal{E}}_{\ell st}, a \in \overline{\mathcal{A}}_e \\ 0 & \text{if } a \in \overline{\mathcal{A}}^{idle}_{\ell st} \cup \overline{\mathcal{A}}^v_{\ell st}. \end{cases} \tag{EC.25}$$

We define decision variables to select arcs in the time-load-expanded segment network:

$$\xi_a = \begin{cases} 1 & \text{if arc } a \text{ is selected, for } (\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell, s \in \mathcal{S}, a \in \overline{\mathcal{A}}_{\ell st}, \\ 0 & \text{otherwise.} \end{cases} \tag{EC.26}$$

Recall that $\Gamma_\ell \subset \mathcal{I}_\ell \times \mathcal{I}_\ell$ denotes the set of checkpoint pairs with up to K skipped checkpoints:

$$\Gamma_\ell = \left\{ (\mathcal{I}^{(i)}_\ell, \mathcal{I}^{(j)}_\ell) \in \mathcal{I}_\ell \times \mathcal{I}_\ell : 1 \le i < j \le I_\ell, j - i \le K+1 \right\}, \qquad \forall \ell \in \mathcal{L}$$

We define additional decision variables to select the set of checkpoint pairs that are visited:

$$\beta_{uv} = \begin{cases} 1 & \text{if checkpoints } (u,v) \in \Gamma_\ell \text{ are visited in sequence, and intermediate checkpoints are not visited,} \\ 0 & \text{otherwise.} \end{cases}$$

Recall that $\mathcal{N}^S_{uv}$ denotes the set of stations that can be visited between checkpoints u and v, and $\mathcal{T}^{uv}_{\ell t}$ denotes the valid arrival times. We link the β_{uv} decisions with the ξ_a decisions, so that the vehicle route abides by the deviation limits imposed by the reference schedule. Altogether, the segment-based formulation exhibits a double flow structure—flow from checkpoint to checkpoint along the reference line, and flow from station to station between checkpoints—with linking constraints to ensure the consistency of these two sets of decisions.

The second-stage segment-based formulation is given as follows for scenario $s \in \mathcal{S}$.

$$\min \quad \sum_{(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell} \sum_{a\in\overline{\mathcal{A}}_{\ell st}} \overline{g}_a \xi_a \tag{EC.27}$$

$$\text{s.t.} \quad \sum_{j:(i,j)\in\overline{\mathcal{A}}_{\ell st}} \xi_{(i,j)} - \sum_{j:(j,i)\in\overline{\mathcal{A}}_{\ell st}} \xi_{(j,i)} = \begin{cases} x_{lt} & \text{if } i = u_{\ell st}, \\ -x_{lt} & \text{if } i = v_{\ell st}, \\ 0 & \text{otherwise,} \end{cases} \quad \forall (\ell,t) \in \mathcal{L}\times\mathcal{T}_\ell, \forall i \in \overline{\mathcal{V}}_{\ell st} \tag{EC.28}$$

$$\sum_{e\in\overline{\mathcal{E}}_{\ell st}} \sum_{a\in\overline{\mathcal{A}}_e : p\in\mathcal{P}_e} \xi_a \le z_{plt} \qquad \forall p \in \mathcal{P},\ \forall (\ell,t) \in \mathcal{M}_p \tag{EC.29}$$

$$\sum_{v:(u,v)\in\Gamma_\ell} \beta_{uv} - \sum_{v:(v,u)\in\Gamma_\ell} \beta_{vu} = \begin{cases} x_{\ell t} & \text{if } u = \mathcal{I}^{(1)}_\ell \\ -x_{\ell t} & \text{if } u = \mathcal{I}^{\text{end}} \\ 0 & \text{otherwise} \end{cases}, \quad \forall (\ell,t) \in \mathcal{L}\times\mathcal{T}_\ell, \forall u \in \mathcal{I}_\ell \tag{EC.30}$$

$$\sum_{\substack{(i,j)\in\overline{\mathcal{A}}_{\ell st}:\\ k_j=v,\, t_j=T_{\ell t}(v)}} \xi_{(i,j)} \ge \sum_{w\in\mathcal{I}_\ell\,:\,(w,v)\in\Gamma_\ell} \beta_{wv} \qquad \forall v\in\mathcal{I}_\ell\setminus\mathcal{I}_\ell^{(1)} \tag{EC.31}$$

$$\xi_{(n,m)} \le \sum_{\substack{(u,v)\in\Gamma_\ell:\\ k_n,k_m\in\mathcal{N}_{uv}^S,\\ t_n,t_m\in\mathcal{T}_{\ell t}^{uv}}} \beta_{uv} \qquad \forall(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell, \forall(n,m)\in\overline{\mathcal{A}}_{\ell st} \tag{EC.32}$$

$$\xi_a\in\{0,1\} \qquad \forall(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell, a\in\overline{\mathcal{A}}_{\ell st} \tag{EC.33}$$

$$\beta_{uv}\in\{0,1\} \qquad \forall(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell, \forall(u,v)\in\Gamma_\ell \tag{EC.34}$$

Equations (EC.27)–(EC.29) are analogous to Equations (8)–(10). Constraint (EC.30) ensures that the vehicle does not skip more than K checkpoints in a row by selecting checkpoint pairs that form a valid path along the reference line. Equations (EC.31) and (EC.32) serve as the linking constraints, ensuring that selected checkpoints are visited at the time specified by the reference schedule, and that the vehicle visits any intermediate locations with the correct chronology. In other words, we can only select a segment if (i) its endpoints correspond to stations in $\mathcal{N}_{uv}^S$ between selected checkpoints, and (ii) its visit times fall within the reference schedule window defined by $T_{\ell t}(u)$ and $T_{\ell t}(v)$. Constraints (EC.33)–(EC.34) apply the binary requirements to the decision variables.

EC.1.5. Path-based formulation for second-stage problem

Component	Type	Description
$\mathcal{Q}_{\ell st}$	Set	Valid paths for reference trip $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$ and scenario $s\in\mathcal{S}$
$\mathcal{P}_q$	Set	Passenger pickup set corresponding to each path $q\in\mathcal{Q}_{\ell st}$
τ_{qp}^{walk}	Parameter	Walk time of passenger $p\in\mathcal{P}_r$ via path $q\in\mathcal{Q}_{\ell st}$, for $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$, $s\in\mathcal{S}$
τ_{qp}^{wait}	Parameter	Wait time of passenger $p\in\mathcal{P}_r$ via path $q\in\mathcal{Q}_{\ell st}$, for $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$, $s\in\mathcal{S}$
$\tau_{qp}^{\text{travel}}$	Parameter	In-vehicle time of passenger $p\in\mathcal{P}_r$ via path $q\in\mathcal{Q}_{\ell st}$, for $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$, $s\in\mathcal{S}$
g_q^Q	Parameter	Cost of path $q\in\mathcal{Q}_{\ell st}$ on trip $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$ in scenario $s\in\mathcal{S}$

Table EC.3 Additional inputs of the path-based formulation.

Throughout the section, we fix first-stage decisions $\boldsymbol{x}$ and $\boldsymbol{z}$, as well as scenario $s\in\mathcal{S}$.

Let $\mathcal{Q}_{\ell st}$ denote the set of all valid paths to reference trip $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$ and scenario $s\in\mathcal{S}$. Each path $q\in\mathcal{Q}_{\ell st}$ corresponds to a sequence of road segments that starts at the beginning of the line, end at its destination, satisfies flow balance in between, skips at most K checkpoints in a row, does not pick up more than C_ℓ passengers, and satisfies the reference schedule at the checkpoints. For each $q\in\mathcal{Q}_{\ell st}$, we store the passenger pickups in $\mathcal{P}_q\subset\mathcal{P}$. By definition, $\sum_{p\in\mathcal{P}_q} D_{ps}\le C_\ell$. The cost g_q^Q of each path is defined analogously to Equation (EC.99) to capture passenger level of service:

$$g_q^Q=\sum_{p\in\mathcal{P}_q} D_{ps}\left(\lambda\tau_{qp}^{\text{walk}}+\mu\tau_{qp}^{\text{wait}}+\sigma\frac{\tau_{qp}^{\text{travel}}}{\tau_p^{\text{dir}}}+\delta\frac{\tau_{\ell tp}^{\text{late}}}{\tau_p^{\text{dir}}}+\frac{\delta}{2}\frac{\tau_{\ell tp}^{\text{early}}}{\tau_p^{\text{dir}}}-M\right), \quad \forall(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell, q\in\mathcal{Q}_{\ell st}. \tag{EC.35}$$

We define the following decision variables:

$$\zeta_q = \begin{cases} 1 & \text{if path } q \text{ is selected, for } (\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell, s \in \mathcal{S}, q \in \mathcal{Q}_{\ell st}, \\ 0 & \text{otherwise.} \end{cases} \tag{EC.36}$$

The path-based formulation is given as follows for scenario $s \in \mathcal{S}$.

$$\min \sum_{(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell} \sum_{q\in\mathcal{Q}_{\ell st}} g_q^Q \zeta_q \tag{EC.37}$$

$$\text{s.t.} \sum_{q\in\mathcal{Q}_{\ell st}} \zeta_q = x_{lt} \qquad \forall (\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell \tag{EC.38}$$

$$\sum_{q\in\mathcal{Q}_{\ell st}\,:\,p\in\mathcal{P}_q} \zeta_q \leq z_{plt} \qquad \forall p \in \mathcal{P},\ \forall (\ell,t) \in \mathcal{M}_p \tag{EC.39}$$

$$\zeta_q \in \{0,1\} \qquad \forall (\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell, q \in \mathcal{Q}_{\ell st} \tag{EC.40}$$

Equations (EC.37) is analogous to Equation (8). Constraints (EC.38) ensure that exactly one path is selected for each selected reference trip. Constraints (EC.39) ensure that selected paths only serve passengers that have been assigned to that trip, analogously to Equation (10).

EC.1.6. Proof of Proposition 2

Throughout this proof, we fix the first-stage decisions $\mathbf{x}$, $\mathbf{z}$. We consider a fixed reference trip $(\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell$ as well as a fixed scenario $s \in \mathcal{S}$.

Equivalence of the path-based and subpath-based formulations.

Constructing a load-expanded subpath solution from a path solution. Let us consider a feasible solution $\widehat{\zeta}$ to the path-based formulation (Equations (EC.37)–(EC.40)) and build a feasible solution to the subpath-based formulation with the same objective value.

Assume that $x_{\ell t} = 1$, and let $q \in \mathcal{Q}_{\ell st}$ be the selected path with $\widehat{\zeta}_q = 1$ (which exists by Equation (EC.38)). By definition, the path corresponds to a sequence of road segments that starts at the beginning of line ℓ, ends at its destination, picks up at most C_ℓ passengers, visits checkpoints without skipping more than K in a row, and arrives at each checkpoint at the scheduled times. With a slight abuse of notation, let $\mathcal{I}_\ell^q := \{\nu_1, \cdots, \nu_Q\} \subseteq \mathcal{I}_\ell$ identify the ordered set of Q checkpoints visited by path q. Similarly, we decompose path q into an ordered sequence of $Q-1$ subpaths $\mathcal{R}_q := \{r_1, \cdots, r_{Q-1}\}$. The subpaths in $\mathcal{R}_q$ partition the served passengers $\mathcal{P}_q$ on path q, so that $\mathcal{P}_q = \bigcup_{r\in\mathcal{R}_q} \mathcal{P}_r$. Each subpath $r_i \in \mathcal{R}_q$ induces a unique arc $a_i := (n,m) \in \mathcal{A}_{\ell st}$ in the load-expanded network, such that (i) the arc corresponds to the subpath: $r(a_i) = r_i$; (ii) the loads are consistent with pickups: $c_n = 0$ if $i = 1$, and $c_n = \sum_{j=1}^{i-1} |\mathcal{P}_{r_j}|$ otherwise, and $c_m = \sum_{j=1}^{i} |\mathcal{P}_{r_j}|$. Let us collect these load-expanded subpath arcs into the set $\mathcal{A}_q := \{a_1, \cdots, a_{Q-1}\} \subset \mathcal{A}_{\ell st}$.

We can construct a feasible solution to the load-expanded subpath formulation.

$$\widehat{y}_a = \begin{cases} 1 & \text{if } a \in \bigcup_{q\in\mathcal{Q}_{\ell st}\,:\,\widehat{\zeta}_q=1} \mathcal{A}_q, \\ 0 & \text{otherwise,} \end{cases} \qquad \forall (\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell, \forall a \in \mathcal{A}_{\ell st}.$$

This solution satisfies the flow balance constraints in Equation (9):

- If $x_{\ell t} = 0$, no path in $\mathcal{Q}_{\ell st}$ is selected, so no arc in $\mathcal{A}_{\ell st}$ is selected either. Therefore,

$$\sum_{m:(n,m)\in\mathcal{A}_{\ell st}} \widehat{y}_{(n,m)} - \sum_{m:(m,n)\in\mathcal{A}_{\ell st}} \widehat{y}_{(m,n)} = 0 = \begin{cases} x_{\ell t} & \text{if } n = u_{\ell st}, \\ -x_{\ell t} & \text{if } n = v_{\ell st}, \\ 0 & \text{otherwise.} \end{cases}$$

- If $x_{\ell t} = 1$, we have, for each node $n \in \mathcal{V}_{\ell st}$:

$$\sum_{m:(n,m)\in\mathcal{A}_{\ell st}} \widehat{y}_{(n,m)} - \sum_{m:(m,n)\in\mathcal{A}_{\ell st}} \widehat{y}_{(m,n)} = \begin{cases} \widehat{y}_{a_1} - 0 = 1 & \text{if } n = u_{\ell st}, \\ 0 - \widehat{y}_{a_{Q-1}} = -1 & \text{if } n = v_{\ell st}, \\ \widehat{y}_{a_i} - \widehat{y}_{a_{i-1}} = 1 - 1 = 0 & \text{if } k_n = \nu_i \in \mathcal{I}^q_\ell \setminus \{\nu_1, \nu_Q\}, \\ & c_n = \sum_{j=1}^{i-1} |\mathcal{P}_{r_j}|, \\ 0 - 0 = 0 & \text{otherwise.} \end{cases}$$

The solution also satisfies the passenger linking constraints in Equations (10).

- Consider a passenger request $p \in \mathcal{P} \setminus \bigcup_{q\in\mathcal{Q}_{\ell st}:\widehat{\zeta}_q=1} \mathcal{P}_q$. The set of pickups on the selected paths $\bigcup_{q\in\mathcal{Q}_{\ell st}:\widehat{\zeta}_q=1} \mathcal{P}_q$ induces the set of pickups on the selected subpaths, so that $p \in \mathcal{P} \setminus \bigcup_{q\in\mathcal{Q}_{\ell st}:\widehat{\zeta}_q=1} \bigcup_{r\in\mathcal{R}_q} \mathcal{P}_r$. Thus, $\widehat{y}_a = 0$ for each arc $a \in \mathcal{A}_{\ell st}$ with $p \in \mathcal{P}_{r(a)}$, and

$$\sum_{a\in\mathcal{A}_{\ell st}:p\in\mathcal{P}_{r(a)}} \widehat{y}_a = 0 \le z_{\ell pst}.$$

- Consider passenger request $p \in \mathcal{P}$ served by some path $q' \in \mathcal{Q}_{\ell st}$, so that $\widehat{\zeta}_{q'} = 1$ and $p \in \mathcal{P}_{q'}$. Each pickup set $\mathcal{P}_{q'}$ has been partitioned into pickup subsets at the subpath level, so there exists $r' \in \mathcal{R}_{q'}$ with $p \in \mathcal{P}_{r'}$. This subpath has been mapped to a unique arc $a_{r'} \in \mathcal{A}_{q'}$, so that

$$\sum_{a\in\mathcal{A}_{\ell st}:p\in\mathcal{P}_{r(a)}} \widehat{y}_a = \widehat{y}_{a_{r'}} = 1 = \widehat{\zeta}_{q'} = \sum_{q\in\mathcal{Q}_{\ell st}:p\in\mathcal{P}_{q'}} \widehat{\zeta}_q \le z_{\ell pst}$$

  where the first three equalities come from the construction of paths and subpaths, the fourth equality stems from the fact that passenger p can be picked up by one subpath, and the final one stems from Equation (EC.39).

Therefore, the solution $\widehat{\boldsymbol{y}}$ is feasible in the subpath-based formulation. We now show that it achieves the same objective value as $\widehat{\zeta}$:

$$\begin{aligned}
\sum_{a\in\mathcal{A}_{\ell st}} g_a \widehat{y}_a &= \sum_{a\in\mathcal{A}_{\ell st}:\widehat{y}_a=1} g_a \\
&= \sum_{q\in\mathcal{Q}_{\ell st}:\widehat{\zeta}_q=1} \sum_{a\in\mathcal{A}_q} g_a \\
&= \sum_{q\in\mathcal{Q}_{\ell st}:\widehat{\zeta}_q=1} \sum_{r_i\in\mathcal{R}_q} g_{a_i} \\
&= \sum_{q\in\mathcal{Q}_{\ell st}:\widehat{\zeta}_q=1} \sum_{r\in\mathcal{R}_q} \sum_{p\in\mathcal{P}_r} D_{ps} \left(\lambda \tau^{\text{walk}}_{rp} + \mu \tau^{\text{wait}}_{rp} + \sigma \frac{\tau^{\text{travel}}_{rp}}{\tau^{\text{dir}}_p} + \delta \frac{\tau^{\text{late}}_{\ell tp}}{\tau^{\text{dir}}_p} + \frac{\delta}{2} \frac{\tau^{\text{early}}_{\ell tp}}{\tau^{\text{dir}}_p} - M \right)
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{q\in\mathcal{Q}_{\ell st}:\widehat{\zeta}_q=1} \sum_{p\in\mathcal{P}_q} D_{ps}\left(\lambda\tau^{\text{walk}}_{qp} + \mu\tau^{\text{wait}}_{qp} + \sigma\frac{\tau^{\text{travel}}_{qp}}{\tau^{\text{dir}}_{p}} + \delta\frac{\tau^{\text{late}}_{\ell tp}}{\tau^{\text{dir}}_{p}} + \frac{\delta}{2}\frac{\tau^{\text{early}}_{\ell tp}}{\tau^{\text{dir}}_{p}} - M\right)\\
&= \sum_{q\in\mathcal{Q}_{\ell st}:\widehat{\zeta}_q=1} g^Q_q\\
&= \sum_{q\in\mathcal{Q}_{\ell st}} g^Q_q\widehat{\zeta}_q \qquad \text{(EC.41)}
\end{aligned}
$$

The first two equalities come from the construction of paths and subpaths; the third equality leverages the uniqueness of the load-expanded subpath arc induced by the subpath sequence; the fourth equality is due to the definition of a load-expanded subpath arc cost; the fifth is due to the partition of $\mathcal{P}_q = \bigcup_{r\in\mathcal{R}_q}\mathcal{P}_r$; and the last two equalities stem from the definition of path costs g^Q_q.

In conclusion, any path solution can be mapped into a feasible subpath solution with the same objective value. Therefore, the subpath-based formulation achieves an objective that is at most equal to an optimum of the path-based formulation.

Constructing a path solution from a load-expanded subpath solution. Let us consider a feasible solution $\widehat{\boldsymbol{y}}$ to the subpath-based formulation (Equations (9)–(11)), and build a feasible solution $\widehat{\zeta}$ to the path-based formulation (Equations (EC.37)–(EC.40)) with the same objective value.

Assume that $x_{\ell t} = 1$. We leverage Equation (9) to construct a path from $u_{\ell st}$ to $v_{\ell st}$ in the load-expanded subpath network $(\mathcal{V}_{\ell st}, \mathcal{A}_{\ell st})$. Beginning from the source, we select the unique arc $a_1 \in \mathcal{A}_{\ell st}$ incident with $u_{\ell st}$ for which $\widehat{y}_{a_1} = 1$, proceeding sequentially along the directed network until $v_{\ell st}$ is reached and $Q-1$ arcs are retrieved. A unique outgoing arc is guaranteed at every intermediate node by Equation (9).

Each arc $a_i \in \mathcal{A}_q := \{a_1, \cdots, a_{Q-1}\}$ corresponds to a subpath $r_i := r(a_i) \in \mathcal{R}_{\ell st}$ and a passenger pickup set $\mathcal{P}_{r_i}$. The sequence of subpaths $\mathcal{R}_q := \{r_1, \cdots, r_{Q-1}\}$ defines a path q from $u_{\ell st}$ to $v_{\ell st}$ (by Equation (9)), skipping at most K checkpoints in a row (by definition of the subpaths $r_i \in \mathcal{R}_{\ell st}$), cohering with the scheduled arrival times associated with trip (ℓ, t) (again by definition of r_i), obeying the vehicle's capacity (by definition of the node set in the load-expanded network $\mathcal{V}_{\ell st}$), and picking up the passengers in $\mathcal{P}_q := \bigcup_{i=1}^{Q-1}\mathcal{P}_{r_i}$ (which is unique due to Equations (10)). Thus, $\widehat{\boldsymbol{y}}$ defines a unique and valid path in $\mathcal{Q}_{\ell st}$ for reference trip (ℓ, t) if $x_{\ell t} = 1$.

Let us collect all such paths in the set $\mathcal{Q}(\widehat{\boldsymbol{y}})$. We construct solution $\widehat{\zeta}$ from $\widehat{\boldsymbol{y}}$:

$$
\widehat{\zeta}_q = \begin{cases} 1 & \text{if } q \in \mathcal{Q}(\widehat{\boldsymbol{y}}),\\ 0 & \text{otherwise,}\end{cases} \qquad \forall(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell, \forall q\in\mathcal{Q}_{\ell st}.
$$

By construction, the solution satisfies Equations (EC.38). If $x_{\ell t} = 1$, we constructed a single path based on the subpath solution. If $x_{\ell t} = 0$, there was no path to construct, as no arcs were selected from $u_{\ell st}$ to $v_{\ell st}$ by Equation (9). Therefore:

$$
\sum_{q\in\mathcal{Q}_{\ell st}} \widehat{\zeta}_q = \sum_{q\in\mathcal{Q}_{\ell st}} \mathbb{1}(q\in\mathcal{Q}(\widehat{\boldsymbol{y}})) = x_{\ell t}, \quad \forall(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell.
$$

The solution also satisfies Equation (EC.39). For each reference trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$ and some passenger $p \in \mathcal{P}$, we obtain, from the construction of the path solution and Equation (10):

$$\begin{aligned}
\sum_{q \in \mathcal{Q}_{\ell st} : p \in \mathcal{P}_q} \widehat{\zeta}_q &= \sum_{q \in \mathcal{Q}(\widehat{\boldsymbol{y}})} \mathbb{1}(p \in \mathcal{P}_q) \\
&= \sum_{q \in \mathcal{Q}(\widehat{\boldsymbol{y}})} \sum_{r \in \mathcal{R}_q} \mathbb{1}\left(p \in \mathcal{P}_r\right) \\
&= \sum_{q \in \mathcal{Q}(\widehat{\boldsymbol{y}})} \sum_{a \in \mathcal{A}_q} \mathbb{1}\left(p \in \mathcal{P}_{r(a)}\right) \\
&= \sum_{a \in \mathcal{A}_{\ell st} \,:\, \widehat{y}_a = 1} \mathbb{1}(p \in \mathcal{P}_{r(a)}) \\
&= \sum_{a \in \mathcal{A}_{\ell st} \,:\, p \in \mathcal{P}_{r(a)}} \widehat{y}_a \\
&\leq z_{\ell pst}
\end{aligned}$$

Finally, the solutions $\widehat{\zeta}$ and $\widehat{\boldsymbol{y}}$ achieve the same objective values, which can be shown similarly to Equations (EC.41). Therefore, any subpath solution can be mapped into a feasible path solution with the same objective value, and the path-based formulation achieves an objective that is at most equal to the optimum of the subpath-based formulation. This concludes the proof of equivalence of the path-based and subpath-based formulations.

Equivalence of path-based and subpath-based relaxations. The arguments employed in this proof do not require the integrality of the path solution $\widehat{\boldsymbol{\zeta}}$ and of the subpath solution $\widehat{\boldsymbol{y}}$. By following the same steps as above, we can map any non-integral path solution $\widehat{\boldsymbol{\zeta}}$ into a feasible subpath solution with the same objective value, as follows:

$$\widehat{y}_a = \sum_{q \in \mathcal{Q}_{\ell st} : a \in \mathcal{A}_q} \widehat{\zeta}_q \qquad \forall (\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell, \forall a \in \mathcal{A}_{\ell st}.$$

Similarly, we can map any non-integral subpath solution $\widehat{\boldsymbol{y}}$ into a feasible path solution with the same objective value. Alternatively, we can observe that the path-based formulation is a Dantzig-Wolfe reformulation of the subpath-based formulation where Equations (9) are convexified into Equations (EC.38). Since Equations (9) already form an integral polyhedron, both formulations contain the same convex hull. This proves that the path-based and subpath-based formulations define the same linear relaxations.

Equivalence of the segment-based and subpath-based formulations.

Constructing a time-load-expanded segment solution from a load-expanded subpath solution. Let us consider a feasible solution $\widehat{\mathbf{y}}$ to the subpath-based formulation (Equations (8)–(11)) and build a feasible solution to the segment-based formulation with the same objective value.

Assume that $x_{\ell t} = 1$, and let $a \in \mathcal{A}_{\ell st}$ be a selected subpath-based arc with $\widehat{y}_a = 1$ (which exists by Equation (9)). By definition, the subpath-based arc corresponds to the load expansion of a subpath that traverses a sequence of road segments starting at checkpoint $u \in \mathcal{I}_\ell$ at time $T_{\ell t}(u)$, ending at checkpoint $v \in \mathcal{I}_\ell$ at time $T_{\ell t}(v)$, skipping up to K checkpoints in-between (i.e., $(u,v) \in \Gamma_\ell$), carrying $c_{start(a)}$ passengers in u, and carrying $c_{end(a)}$ passengers in v. Let us store the stations visited by subpath r in an ordered set $\mathcal{N}_r^S := \{\nu_1, \cdots, \nu_N\} \subseteq \mathcal{N}^S$, where $\nu_1 = u$, $\nu_N = v$, and $\nu_2, \cdots, \nu_{N-1}$ denote intermediate stations. Similarly, we decompose subpath r into a sequence of $N-1$ segments $\mathcal{E}_r := \{e_1, \cdots, e_{N-1}\}$, where segment e_i connects stations ν_i and ν_{i+1} with travel time tt_{e_i} (potentially with idling time). The segments in $\mathcal{E}_r$ partition the passengers in $\mathcal{P}_r$: $\mathcal{P}_r = \bigcup_{e \in \mathcal{E}_r} \mathcal{P}_e$.

To obtain the corresponding segment solution, we need to specify an appropriate time discretization. Due to the adherence to the reference schedule, the discretization in the segment-based formulation does not introduce errors as long as all viable subpaths are feasible in that formulation. We show that there exists a discrete time unit for which this is the case, in the following lemma.

Lemma EC.1. *Assume that the elapsed time between the scheduled arrival times at the checkpoints along the reference line are strictly larger than the travel times of the corresponding subpaths. Then, there exists a discrete time unit such that, in the corresponding time-expanded network, all feasible subpaths have an estimated travel time that is less than the elapsed time between the corresponding checkpoints' scheduled arrival times.*

Let $\Delta_{uv} := T_{\ell t}(v) - T_{\ell t}(u)$ denote the travel time between checkpoints u and v, determined by the scheduled arrival times at both checkpoints, and let us denote the travel time of subpath r by $\Delta_r := \sum_{e \in \mathcal{E}_r} tt_e$. Due to the maximum deviation from the reference line, the number of passenger pickups, and the upper bound on passengers' walking distance, the set of potential subpaths $\mathcal{R}_{\ell st}^{uv}$ between checkpoints u and v is finite. For convenience, let us denote this subset by

$$\mathcal{R}_{\ell st}^{uv} := \{r \in \mathcal{R}_{\ell st} : u_r = u, v_r = v\}.$$

By assumption, all subpaths $r \in \mathcal{R}_{\ell st}$ satisfy $\Delta_r \leq T_{\ell t}(v_r) - T_{\ell t}(u_r)$, so that $\Delta_r < \Delta_{uv}$ for each $r \in \mathcal{R}_{\ell st}^{uv}$. We define the discrete time unit between checkpoints u and v as:

$$\rho_{uv} = \min_{r \in \mathcal{R}_{\ell st}^{uv}} \frac{\Delta_{uv} - \Delta_r}{|\mathcal{E}_r|} > 0. \tag{EC.42}$$

Without loss of generality, we assume that ρ_{uv} is rational; otherwise, we can define it as the largest rational number bounded from above by the minimum given in Equation (EC.42). We define the universal discrete time unit as

$$\rho = \mathrm{GCD}\left(\{\rho_{uv} : (u,v) \in \Gamma_\ell\}\right), \tag{EC.43}$$

where GCD denotes the greatest common divisor. By construction, for each $(u,v) \in \Gamma_\ell$, there exists $R_{uv} \in \mathbb{Z}_+$ such that $\rho_{uv} = R_{uv}\rho \geq \rho$.

In the segment-based formulation, travel times are rounded up to the nearest discrete time step on each segment. The estimated travel time on each segment $e \in \mathcal{E}_r$, denoted by $\overline{\Delta}_e$, is therefore

$$\overline{\Delta}_e = \left\lceil \frac{tt_e}{\rho} \right\rceil \cdot \rho.$$

The travel time estimate of subpath $r \in \mathcal{R}^{uv}_{\ell st}$ in the segment-based formulation, denoted by $\overline{\Delta}_r$, is then given by:

$$\overline{\Delta}_r = \sum_{e \in \mathcal{E}_r} \overline{\Delta}_e = \sum_{e \in \mathcal{E}_r} \left\lceil \frac{tt_e}{\rho} \right\rceil \rho.$$

We make use of the following property:

$$\left\lceil \frac{tt_e}{\rho} \right\rceil \cdot \rho \leq \left\lceil \frac{tt_e}{\rho_{uv}} \right\rceil \cdot \rho_{uv} \leq tt_e + \rho_{uv}.$$

The first inequality stems from the fact that $\lceil a/R_{uv} \rceil \leq \lceil a \rceil / R_{uv}$ for any $a > 0$. The second inequality follows from the definition of the ceiling function. Thus, we obtain:

$$\overline{\Delta}_r \leq \sum_{e \in \mathcal{E}_r} (tt_e + \rho_{uv}) = \Delta_r + \sum_{e \in \mathcal{E}_r} \rho_{uv} \leq \Delta_r + \sum_{e \in \mathcal{E}_r} \left(\frac{\Delta_{uv} - \Delta_r}{|\mathcal{E}_r|} \right) = \Delta_{uv}, \ \forall r \in \mathcal{R}^{uv}_{\ell st}. \tag{EC.44}$$

This completes the proof of the lemma. □

Lemma EC.1 shows that there exists a discrete time unit for which all feasible subpaths in the subpath-based formulation are also feasible in the segment-based formulation. With this discretization, each segment $e_i \in \mathcal{E}_r$ induces a unique arc $\overline{a}_i := (n,m) \in \overline{\mathcal{A}}_{\ell st}$ in the time-load-expanded network, such that: (i) the arc corresponds to the segment: $e(\overline{a}_i) = e_i$; (ii) the capacities are consistent with pickups: $c_n = c_{start(a)}$ if $i = 1$, and $c_n = c_{start(a)} + \sum_{j=1}^{i-1} |\mathcal{P}_{e_j}|$ otherwise, and $c_m = c_{start(a)} + \sum_{j=1}^{i} |\mathcal{P}_{e_j}|$; and (iii) the time is consistent with travel times: $t_n = T_{\ell t}(u)$ if $i = 1$, and $t_n = T_{\ell t}(u) + \rho_{uv} \cdot \sum_{j=1}^{i-1} \lceil \frac{tt_{e_j}}{\rho_{uv}} \rceil$ otherwise, and $t_m = T_{\ell t}(u) + \rho_{uv} \cdot \sum_{j=1}^{i} \lceil \frac{tt_{e_j}}{\rho_{uv}} \rceil$. Let us collect these time-load-expanded segment arcs into the set $\overline{\mathcal{A}}_a := \{\overline{a}_1, \cdots, \overline{a}_{N-1}\} \subset \overline{\mathcal{A}}_{\ell st}$.

With these arcs, we construct a feasible solution to the time-load-expanded segment formulation.

$$\widehat{\xi}_{\overline{a}} = \begin{cases} 1 & \text{if } \overline{a} \in \bigcup_{a \in \mathcal{A}_{\ell st} : \widehat{y}_a = 1} \overline{\mathcal{A}}_a, \\ 0 & \text{otherwise,} \end{cases} \qquad \forall (\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell, \forall \overline{a} \in \overline{\mathcal{A}}_{\ell st}$$

$$\widehat{\beta}_{uv} = \begin{cases} 1 & \text{if there exists } a \in \mathcal{A}_{\ell st} : \widehat{y}_a = 1, u_{r(a)} = u, v_{r(a)} = v, \\ 0 & \text{otherwise,} \end{cases} \qquad \forall (\ell,t) \in \mathcal{L} \times \mathcal{T}_\ell, \forall (u,v) \in \Gamma_\ell.$$

This solution satisfies the flow balance constraints in Equation (EC.28).

– If $x_{\ell t} = 0$, then no subpath arc in $\mathcal{A}_{\ell st}$ is selected, so no arc in $\overline{\mathcal{A}}_{\ell st}$ is selected either. Therefore

$$\sum_{j:(i,j) \in \overline{\mathcal{A}}_{\ell st}} \widehat{\xi}_{(i,j)} - \sum_{j:(j,i) \in \overline{\mathcal{A}}_{\ell st}} \widehat{\xi}_{(j,i)} = 0 = \begin{cases} x_{lt} & \text{if } i = u_{\ell st}, \\ -x_{lt} & \text{if } i = v_{\ell st}, \\ 0 & \text{otherwise.} \end{cases}$$

– Suppose that $x_{\ell t} = 1$. Recall the sequence of subpath arcs $a \in \mathcal{A}_{\ell st}$ such that $\widehat{y}_a = 1$, which exist by Equation (9). We collect the corresponding segment arcs from $\overline{\mathcal{A}}_a$ in order into set $\overline{\mathcal{A}}_{\text{all}} = \bigcup_{a\in\mathcal{A}_{\ell st}:\widehat{y}_a=1} \overline{\mathcal{A}}_a := \{\overline{a}_1, \cdots, \overline{a}_{M-1}\}$ and corresponding time-load-expanded nodes $\overline{\mathcal{V}}^{\text{all}} = \{n_1, \cdots, n_M\}$, where $n_1 = u_{\ell st}$ and $n_M = u_{\ell st}$ and $n_2, \cdots, n_{M-1}$ refer to intermediate nodes. We have, for each node $n \in \overline{\mathcal{V}}_{\ell st}$:

$$\sum_{m:(n,m)\in\overline{\mathcal{A}}_{\ell st}} \widehat{\xi}_{(n,m)} - \sum_{m:(m,n)\in\overline{\mathcal{A}}_{\ell st}} \widehat{\xi}_{(m,n)} = \begin{cases} \widehat{\xi}_{\overline{a}_1} - 0 = 1 = x_{\ell t} & \text{if } n = u_{\ell st}, \\ 0 - \widehat{\xi}_{\overline{a}_M} = -1 = -x_{lt} & \text{if } n = v_{\ell st}, \\ \widehat{\xi}_{\overline{a}_i} - \widehat{\xi}_{\overline{a}_{i-1}} = 0 & \text{if } n = n_i \in \overline{\mathcal{V}}^{\text{all}} \setminus \{n_1, n_M\}, \\ 0 - 0 = 0 & \text{otherwise.} \end{cases}$$

The solution also satisfies the passenger linking constraints in Equations (EC.29).

– Consider a passenger request $p \in \mathcal{P} \setminus \bigcup_{a\in\mathcal{A}_{\ell st}\,:\,\widehat{y}_a=1} \mathcal{P}_{r(a)}$. The set of pickups on the selected subpaths $\bigcup_{a\in\mathcal{A}_{\ell st}\,:\,\widehat{y}_a=1} \mathcal{P}_a$ induces the set of pickups on the selected segments, so that $p \in \mathcal{P} \setminus \bigcup_{a\in\mathcal{A}_{\ell st}\,:\,\widehat{y}_a=1} \bigcup_{e\in\mathcal{E}_{r(a)}} \mathcal{P}_e$. Thus, $\widehat{\xi}_{\overline{a}} = 0$ for each arc $\overline{a} \in \overline{\mathcal{A}}_{\ell st}$ with $p \in \mathcal{P}_{e(\overline{a})}$, and

$$\sum_{\overline{a}\in\overline{\mathcal{A}}_{\ell st}\,:\,p\in\mathcal{P}_{e(\overline{a})}} \widehat{\xi}_{\overline{a}} = 0 \leq z_{\ell pst}.$$

– Consider passenger request $p \in \mathcal{P}$ served by some subpath arc $a' \in \mathcal{A}_{\ell st}$, so that $\widehat{y}_{a'} = 1$ and $p \in \mathcal{P}_{r(a')}$. Each pickup set $\mathcal{P}_{r(a')}$ has been partitioned into pickup subsets at the segment level, so that there exists some $e \in \mathcal{E}_{r(a')}$ with $p \in \mathcal{P}_e$. This segment has been mapped to a unique arc $\overline{a}' \in \mathcal{A}_{a'}$. Using Equation (10), we obtain:

$$\sum_{\overline{a}\in\overline{\mathcal{A}}_{\ell st}:p\in\mathcal{P}_{e(\overline{a})}} \widehat{\xi}_{\overline{a}} = \widehat{\xi}_{\overline{a}'} = 1 = \widehat{y}_{a'} = \sum_{a\in\mathcal{A}_{\ell st}:p\in\mathcal{P}_{r(a)}} \widehat{y}_a \leq z_{\ell pst}.$$

Next, the solution satisfies the flow balance between checkpoints in Equations (EC.30).

– If $x_{\ell t} = 0$, then no subpath arcs in $\mathcal{A}_{\ell st}$ are selected, so $\widehat{\beta}_{uv} = 0$ for all $(u,v) \in \Gamma_\ell$. Therefore, for each checkpoint $u \in \mathcal{I}_\ell$, we have:

$$\sum_{v\,:(u,v)\in\Gamma_\ell} \widehat{\beta}_{uv} - \sum_{v\,:(v,u)\in\Gamma_\ell} \widehat{\beta}_{vu} = 0 = \begin{cases} x_{\ell t} & \text{if } u = \mathcal{I}_\ell^{(1)} \\ -x_{\ell t} & \text{if } u = \mathcal{I}^{\text{end}} \\ 0 & \text{otherwise} \end{cases}$$

– If $x_{\ell t} = 1$, then we identify the sequence of subpath arcs $a \in \mathcal{A}_{\ell st}$ such that $\widehat{y}_a = 1$, which exists and defines the unique sequence of checkpoints per Equation (9). With a slight abuse of notation, this sequence is denoted by $\mathcal{I}_{\ell t} := \{\omega_1 := \mathcal{I}_\ell^{(1)}, \cdots, \omega_O := \mathcal{I}_\ell^{(I_\ell)}\}$. We obtain the flow balance constraints for each checkpoint $u \in \mathcal{I}_\ell$:

$$\sum_{v\,:(u,v)\in\Gamma_\ell} \widehat{\beta}_{uv} - \sum_{v\,:(v,u)\in\Gamma_\ell} \widehat{\beta}_{vu} = \begin{cases} \widehat{\beta}_{\omega_1,\omega_2} - 0 = 1 = x_{\ell t} & \text{if } u = \omega_1 \\ 0 - \widehat{\beta}_{\omega_{O-1},\omega_O} = -1 = -x_{\ell t} & \text{if } u = \omega_O \\ \widehat{\beta}_{\omega_i,\omega_{i+1}} - \widehat{\beta}_{\omega_{i-1},\omega_i} = 1 - 1 = 0 & \text{if } u \in \mathcal{I}_{\ell t} \backslash \{\omega_1, \omega_O\} \\ 0 & \text{otherwise} \end{cases}$$

Next, the solution satisfies the checkpoint visit constraints given in Equation (EC.31):

- If $x_{\ell t} = 0$, then $\widehat{\beta}_{uv} = 0$ for all $(u, v) \in \Gamma_\ell$, so the equation is trivially satisfied.
- If $x_{\ell t} = 1$, then we enumerate the set of visited checkpoints by the subpaths in $\mathcal{I}_{\ell t}$ using Equations (9). We consider a checkpoint $v \in \mathcal{I}_\ell \setminus \mathcal{I}_\ell^{(1)}$. If $\sum_{w \in \mathcal{I}_\ell : (w,v) \in \Gamma_\ell} \beta_{wv} = 1$, then there exists a subpath-based arc $a \in \mathcal{A}_{\ell st}$ such that $\widehat{y}_a = 1, u_{r(a)} = w$, and $v_{r(a)} = v$, which terminates in v. Per construction of the segment-based arcs, there exist segments $\overline{a}_1, \cdots, \overline{a}_{N-1} \in \overline{\mathcal{A}}_a$ such that $\widehat{\xi}_{\overline{a}_1} = \cdots = \widehat{\xi}_{\overline{a}_{N-1}} = 1$, corresponding to segments $e_1, \cdots, e_{N-1}$. Then,

$$\sum_{\substack{(i,j) \in \overline{\mathcal{A}}_{\ell st}: \\ k_j = v,\, t_j = T_{\ell t}(v)}} \widehat{\xi}_{(i,j)} = 1,$$

and the constraint is satisfied. The constraint is trivially satisfied if $\sum_{w \in \mathcal{I}_\ell : (w,v) \in \Gamma_\ell} \beta_{wv} = 0$.

Finally, the solution satisfies the checkpoint sequencing constraints given in Equation (EC.32), by construction of the $\widehat{\boldsymbol{\xi}}$ and $\widehat{\boldsymbol{\beta}}$ variables. Indeed, $\widehat{\beta}_{uv} = 1$ whenever there exists an arc $\overline{a} \in \bigcup_{a \in \mathcal{A}_{\ell st} : \widehat{y}_a = 1} \overline{\mathcal{A}}_a$ between checkpoints u and v and between times $T_{\ell t}(u)$ and $T_{\ell t}(v)$ such that $\widehat{\xi}_{\overline{a}} = 1$.

Next, the solution $\widehat{\boldsymbol{\xi}}$ achieves the same objective value as $\widehat{\boldsymbol{y}}$:

$$\begin{aligned}
\sum_{\overline{a} \in \overline{\mathcal{A}}_{\ell st}} \overline{g}_{\overline{a}} \widehat{\xi}_{\overline{a}} &= \sum_{\overline{a} \in \overline{\mathcal{A}}_{\ell st} : \widehat{\xi}_{\overline{a}} = 1} \overline{g}_{\overline{a}} \\
&= \sum_{a \in \mathcal{A}_{\ell st} : \widehat{y}_a = 1} \sum_{\overline{a} \in \mathcal{A}_{\overline{a}}} \overline{g}_{\overline{a}} \\
&= \sum_{a \in \mathcal{A}_{\ell st} : \widehat{y}_a = 1} \sum_{e \in \mathcal{E}_r} \sum_{p \in \mathcal{P}_e} D_{ps} \left(\lambda \tau_{ep}^{\text{walk}} + \mu \tau_{ep}^{\text{wait}} + \sigma \frac{\tau_{ep}^{\text{travel}}}{\tau_p^{\text{dir}}} + \delta \frac{\tau_{\ell tp}^{\text{late}}}{\tau_p^{\text{dir}}} + \frac{\delta}{2} \frac{\tau_{\ell tp}^{\text{early}}}{\tau_p^{\text{dir}}} - M \right) \\
&= \sum_{a \in \mathcal{A}_{\ell st} : \widehat{y}_a = 1} \sum_{p \in \mathcal{P}_r} D_{ps} \left(\lambda \tau_{rp}^{\text{walk}} + \mu \tau_{rp}^{\text{wait}} + \sigma \frac{\tau_{rp}^{\text{travel}}}{\tau_p^{\text{dir}}} + \delta \frac{\tau_{\ell tp}^{\text{late}}}{\tau_p^{\text{dir}}} + \frac{\delta}{2} \frac{\tau_{\ell tp}^{\text{early}}}{\tau_p^{\text{dir}}} - M \right) \\
&= \sum_{a \in \mathcal{A}_{\ell st} : \widehat{y}_a = 1} g_a \\
&= \sum_{a \in \mathcal{A}_{\ell st}} g_a \widehat{y}_a
\end{aligned} \tag{EC.45}$$

In conclusion, any subpath solution can be mapped into a feasible segment solution with the same objective value. Therefore, the segment-based formulation achieves an objective that is at most equal to the optimum of the subpath-based formulation.

Constructing a subpath solution from a time-load-expanded segment solution.

Suppose that $\widehat{\boldsymbol{\xi}}$ is a feasible solution to the segment-based formulation (Equations (EC.27)–(EC.34)). Assume that $x_{\ell t} = 1$. We leverage Equations (EC.28) to construct a subpath between checkpoints u and v and between times $T_{\ell t}(u), T_{\ell t}(v)$. Starting from the source checkpoint u, we select the arc $\overline{a}_1 \in \overline{\mathcal{A}}_{\ell st}$ incident with $u_{\ell st}$ for which $\widehat{\xi}_a = 1$, proceeding sequentially along the directed network until reaching the node corresponding to checkpoint v at time $T_{\ell t}(v)$. An outgoing arc is guaranteed at every intermediate node by Equation (EC.28), and boundary conditions at the checkpoints are guaranteed by Equation (EC.31).

Each arc $\overline{a}_i \in \overline{\mathcal{A}}_a := \{\overline{a}_1, \cdots, \overline{a}_{N-1}\}$ corresponds to a segment $e_i := e(\overline{a}_i) \in \mathcal{E}_{\ell st}$ and a passenger pickup set $\mathcal{P}_{e_i}$. The sequence of segments $\mathcal{E}_r := \{e_1, \cdots, e_{N-1}\}$ defines a subpath r from u_r to v_r, skipping at most K checkpoints in a row (by Equations (EC.30) and the definition of checkpoint pairs Γ_ℓ), adhering to the scheduled arrival times at the checkpoints (defined by Equations (EC.31)), obeying the vehicle's capacity (by definition of the node set $\overline{\mathcal{V}}_{\ell st}$ in the time-load-expanded network), and picking up the passengers in $\mathcal{P}_r := \bigcup_{i=1}^{N-1} \mathcal{P}_{e_i}$ (who are unique due to Equations (EC.29)). Thus, we obtain a unique and valid subpath-based arc in $\mathcal{A}_{\ell st}$, induced by $\widehat{\boldsymbol{\xi}}$.

Let us collect all such subpath arcs in the set $\mathcal{A}(\widehat{\boldsymbol{\xi}})$. We use $\mathcal{A}(\widehat{\boldsymbol{\xi}})$ to construct solution $\widehat{\boldsymbol{y}}$ from $\widehat{\boldsymbol{\xi}}$:

$$\widehat{y}_a = \begin{cases} 1 & \text{if } a \in \mathcal{A}(\widehat{\boldsymbol{\xi}}), \\ 0 & \text{otherwise,} \end{cases} \qquad \forall (\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell, \forall a \in \mathcal{A}_{\ell st}.$$

By construction, the solution satisfies Equations (9). If $x_{\ell t} = 1$, we constructed a unique subpath-based solution for each pair $(u, v) \in \Gamma_\ell$. If $x_{\ell t} = 0$, there was no subpath to construct. Therefore:

$$\sum_{a \in \mathcal{A}_{\ell st}} \widehat{y}_a = \sum_{a \in \mathcal{A}_{\ell st}} \mathbb{1}(a \in \mathcal{A}(\widehat{\boldsymbol{\xi}})) = x_{\ell t}, \quad \forall (\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell.$$

The solution also satisfies Equation (10). For passenger $p \in \mathcal{P}$, we obtain, from the construction of the subpath solution and Equation (EC.28):

$$\begin{aligned} \sum_{a \in \mathcal{A}_{\ell st} : p \in \mathcal{P}_{r(a)}} \widehat{y}_a &= \sum_{a \in \mathcal{A}(\widehat{\boldsymbol{\xi}})} \mathbb{1}(p \in \mathcal{P}_{r(a)}) \\ &= \sum_{a \in \mathcal{A}(\widehat{\boldsymbol{\xi}})} \sum_{e \in \mathcal{E}_{r(a)}} \mathbb{1}\left(p \in \mathcal{P}_e\right) \\ &= \sum_{a \in \mathcal{A}(\widehat{\boldsymbol{\xi}})} \sum_{\overline{a} \in \overline{\mathcal{A}}_a} \mathbb{1}\left(p \in \mathcal{P}_{e(\overline{a})}\right) \\ &= \sum_{\overline{a} \in \overline{\mathcal{A}}_{\ell st} : \widehat{\xi}_{\overline{a}} = 1} \mathbb{1}(p \in \mathcal{P}_{e(\overline{a})}) \\ &= \sum_{\overline{a} \in \overline{\mathcal{A}}_{\ell st} : p \in \mathcal{P}_{e(\overline{a})}} \widehat{\xi}_{\overline{a}} \\ &\leq z_{\ell pst} \end{aligned}$$

Finally, the solutions $\widehat{\boldsymbol{\xi}}$ and $\widehat{\boldsymbol{y}}$ achieve the same objective values, which can be shown similarly to Equation (EC.45). Therefore, any segment solution can be mapped into a feasible subpath solution with the same objective value, and the subpath-based formulation achieves an objective that is at most equal to the optimum of the segment-based formulation. This concludes the proof of equivalence of the subpath-based and segment-based formulations.

Proof that the subpath-based relaxation is at least as strong as the segment-based relaxation.

The subpath-based formulation is a Dantzig-Wolfe reformulation of the segment-based formulation. Alternatively, we can observe that the arguments to map a segment-based solution into a subpath-based solution do not require the integrality of the subpath solution $\widehat{\boldsymbol{y}}$. By following the same steps as above, we can map any non-integral subpath solution $\widehat{\boldsymbol{y}}$ into a feasible segment solution with the same objective value, as follows:

$$\widehat{\xi}_{\overline{a}} = \sum_{a \in \mathcal{A}_{\ell st} : \overline{a} \in \overline{\mathcal{A}}_a} \widehat{y}_a \qquad \forall (\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell, \forall \overline{a} \in \overline{\mathcal{A}}_{\ell st}$$

$$\widehat{\beta}_{uv} = \sum_{a \in \mathcal{A}_{\ell st} : u_{r(a)} = u, v_{r(a)} = u} \widehat{y}_a \qquad \forall (\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell, \forall u, v \in \Gamma_\ell.$$

However, a non-integral segment solution cannot be mapped directly to a subpath solution. We demonstrate this claim with an example with three checkpoints (A, B and C). Figure EC.1 shows a non-integral segment-based solution—all shown segments have a flow of 0.5. The solution satisfies the flow balance constraints from station to station given in Equation (EC.28), the flow balance constraints from checkpoint to checkpoint given in Equation (EC.30), as well as the consistency constraints between checkpoint-checkpoint flows and station-station flows given in Equations (EC.31)–(EC.32). In this solution $\widehat{\boldsymbol{\beta}}$ values are $\widehat{\beta}_{(A,B)} = \widehat{\beta}_{(B,C)} = \widehat{\beta}_{(A,C)} = 0.5$, so that the flows are split between a subpath from Checkpoint A to Checkpoint B, a subpath from Checkpoint B to Checkpoint C, and a subpath from Checkpoint A to Checkpoint C, each with a flow of 0.5. The critical observation is that the solution leverages the segments (shown in solid lines) that fall outside the spatial scope of the deviations between Checkpoints A and B as part of the subpath connecting Checkpoints A and B. Specifically, there exists a segment $(n, m) \in \mathcal{A}_{\ell st}$ with $k_a \in \mathcal{N}^S_{13} \setminus \mathcal{N}^S_{12}$ or $k_b \in \mathcal{N}^S_{13} \setminus \mathcal{N}^S_{12}$. This solution belongs to the polyhedron defined by the segment-based formulation, because $\xi_{(n,m)} = 0.5 \leq \beta_{(A,C)} = 0.5$. However, the resulting subpath is infeasible because it connects Checkpoints A and B without adhering to the maximum deviation Δ. This proves that the subpath-based relaxation is at least as strong as the segment-based relaxation. □

EC.1.7. Proof of Proposition 3

Let $D_{\min}$ be the minimum distance between pickup locations (a constant dictated by the station set $\mathcal{N}^S$) and let Π denote the maximum distance between any checkpoint pair (a constant dictated by the candidate reference lines and the value of $K = 0$ vs. $K = 1$). Recall that Δ denotes the maximum deviation from the reference line for microtransit vehicles, and Ω denotes the maximum walking distance. Therefore, the rectangular service area associated with a checkpoint pair has side lengths $\Pi + 2(\Delta + \Omega)$ and $2(\Delta + \Omega)$. The maximum number of stations between checkpoints is $\Xi = \lfloor \frac{(\Pi + 2(\Delta + \Omega))}{D_{\min}} \rfloor \cdot \lfloor \frac{2 \cdot (\Delta + \Omega)}{D_{\min}} \rfloor$. In our case study, Ξ is significantly less than $|\mathcal{N}^S| = 640$.

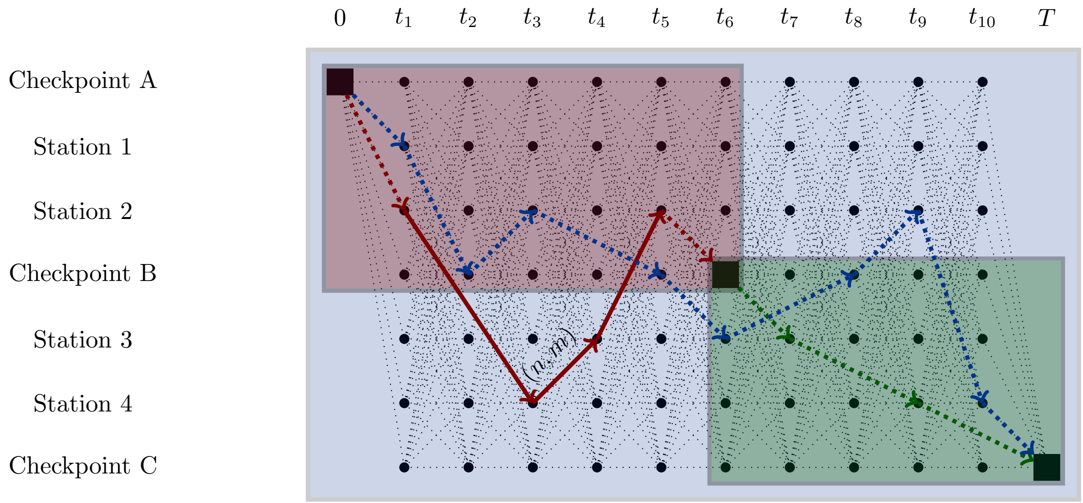


Figure EC.1 **Example of a non-integral segment solution that cannot be mapped to a subpath solution. For simplicity, the load dimension of the time-load-expanded network is omitted. The black squares encode the reference schedule at the checkpoints. The red (resp. green, blue) area represents the stations that can be reached between Checkpoints A and B (resp. between Checkpoints B and C, between Checkpoints A and C). All thick segments are associated with a flow of 0.5. The solid segments are outside the allowable region in the subpath-based formulation.**

Segment-based model.

The number of variables scales in $\mathcal{O}(T_S \cdot C_\ell^2 \cdot I_\ell \cdot \Xi^2)$.

- The $\boldsymbol{\beta}$ variables are indexed over the set of valid directed checkpoint pairs Γ_ℓ. Since K is a small constant, the number of valid checkpoint pairs scales linearly with the number of checkpoints, so that $\mathcal{O}(|\Gamma_\ell|) = \mathcal{O}\left(\sum_{i=1}^{I_\ell-(K+1)}(K+1)\right) = \mathcal{O}(I_\ell)$.
- The $\boldsymbol{\xi}$ variables are indexed over the set $\overline{\mathcal{A}}_{\ell st}$, i.e., the arcs in the time-load-expanded network. Arcs define connections between consecutive stations between a checkpoint pair (which scale in $\mathcal{O}(I_\ell \cdot \Xi^2)$) for each time period, and they can correspond to any vehicle load pair, so the number of $\boldsymbol{\xi}$ variables scales in $\mathcal{O}(\overline{\mathcal{A}}_{\ell st}) = \mathcal{O}(T_S \cdot C_\ell^2 \cdot I_\ell \cdot \Xi^2)$.

The number of constraints scales in $\mathcal{O}(|\mathcal{P}| + T_S \cdot C_\ell \cdot |\mathcal{N}^S| + T_S \cdot C_\ell^2 \cdot I_\ell \cdot \Xi^2)$.

- Equations (EC.28): There is one constraint per node in the time-load-expanded network. There is one node per combination of time periods in $\mathcal{T}^S$, vehicle loads in $\mathcal{C}_\ell$, and stations in $\mathcal{N}^S$, so that there are $\mathcal{O}(T_S \cdot C_\ell \cdot |\mathcal{N}^S|)$ flow balance constraints.
- Equations (EC.29): The passenger linking constraints scale with $\mathcal{O}(\sum_{p\in\mathcal{P}} |\mathcal{M}_p|)$. The cardinality of each set $\mathcal{M}_p$ is bounded by a small constant, so there are $\mathcal{O}(|\mathcal{P}|)$ linking constraints.
- Equations (EC.30)–(EC.31): There are $\mathcal{O}(I_\ell)$ flow balance constraints for checkpoint-to-checkpoint flows and $\mathcal{O}(I_\ell)$ schedule adherence constraints.

- Equations (EC.32): There is one constraint per arc in the time-load-expanded network to ensure consistency between the station-to-station and checkpoint-to-checkpoint flows, which grows in $\mathcal{O}(\overline{\mathcal{A}}_{\ell st}) = \mathcal{O}(C_\ell^2 \cdot T_S \cdot I_\ell \cdot \Xi^2)$, as previously established.

The complexity of Equations (EC.30)–(EC.31) is dominated by that of Equations (EC.28) and (EC.32). The result follows.

Subpath-based model. The number of variables scales in $\mathcal{O}(I_\ell \cdot C_\ell \cdot 2^\Xi)$. In particular, $\boldsymbol{y}$ scales with $\mathcal{O}(|\mathcal{A}_{\ell st}|)$, the number of arcs in the load-expanded subpath network. By definition:

$$|\mathcal{A}_{\ell st}| = |\mathcal{A}^v_{\ell st}| + \sum_{r \in \mathcal{R}_{\ell st}} |\mathcal{A}_r|.$$

As for $|\mathcal{A}^v_{\ell st}|$, there are $C_\ell + 1$ arcs connecting the last stop to the sink node (one per vehicle load). Turning to $\mathcal{A}_r$, a subpath $r \in \mathcal{R}_{\ell st}$ is the shortest path to serve the corresponding passenger set $\mathcal{P}_r$. Thus, the number of subpath variables is proportional to the number of possible sets $\mathcal{P}_r$. The number of different passengers that can be picked up at each station is bounded by a small constant, so we use the number of stations as a proportional proxy for the number of passengers that can be picked up. There are up to $\binom{\Xi}{c}$ station combinations that pick up c passengers, each of which can be replicated $C_\ell - c + 1$ times in the arc set (corresponding to initial loads $0, 1, \cdots, C_\ell - c$). Therefore, the number of subpaths is

$$\sum_{c=0}^{C_\ell} \binom{\Xi}{c} \cdot (C_\ell - c + 1) \leq (C_\ell + 1) \cdot \sum_{c=0}^{C_\ell} \binom{\Xi}{c}.$$

When $\Xi \leq C_\ell$, the binomial sum above is equal to 2^Ξ. When $\Xi > C_\ell$, it is equal to

$$\sum_{c=0}^{C_\ell} \binom{\Xi}{c} = 2^\Xi - \sum_{c=C_\ell+1}^{\Xi} \binom{\Xi}{c} \leq 2^\Xi.$$

Therefore, $\mathcal{O}(|\mathcal{A}_{\ell st}|) = \mathcal{O}(2^\Xi \cdot C_\ell \cdot I_\ell)$.

The number of constraints scales in $\mathcal{O}(|\mathcal{P}| + C_\ell \cdot I_\ell)$.

- Equations (9): There are $\mathcal{O}(\mathcal{V}_{\ell st}) = \mathcal{O}(C_\ell \cdot I_\ell)$ flow balance constraints, one per node in the load-expanded subpath network.
- Equations (10): There are $\mathcal{O}(|\mathcal{P}|)$ linking constraints, one per passenger that can be picked up by reference trip (ℓ, t).

Path-based model. The number of variables scales in $\mathcal{O}(2^{\Xi \cdot I_\ell})$. The ζ variables are indexed over the path set $\mathcal{Q}_{\ell st}$. Each path $q \in \mathcal{Q}_{\ell st}$ can be decomposed into a sequence of subpaths in $\mathcal{R}_{\ell st}$ by partitioning the path-based passenger set $\mathcal{P}_q$ into subpath-based passenger sets $\mathcal{P}_r$, that is $\mathcal{P}_q = \bigcup_{r \in \mathcal{R}_q} \mathcal{P}_r$. Recall that there are $O(2^\Xi)$ possible subpaths between each checkpoint pair, and there are $\mathcal{O}(I_\ell)$ checkpoint pairs, so we obtain $\mathcal{O}(2^{\Xi \cdot I_\ell})$ overall paths.

The number of constraints scales in $\mathcal{O}(|\mathcal{P}|)$, i.e., with the number of linking constraints (Equations (EC.39)). The model also comprises a single partitioning constraint (Equation (EC.38)), which does not affect the constraint complexity. □

EC.2. Details on Solution Algorithm

EC.2.1. Proof of Theorem 1

First, we show that the inner column generation procedure converges to an optimal solution of $\overline{\texttt{SP}}(\boldsymbol{x})$ for a given first stage solution $\boldsymbol{x}$. The restricted subproblem $\texttt{RSP}(\boldsymbol{x}, \mathcal{A}'_{sj})$ has optimal value $\varphi'(\boldsymbol{x}, \mathcal{A}'_{sj}) \geq \varphi(\boldsymbol{x})$ because $\mathcal{A}'_{sj} \subset \mathcal{A}_{sj}$ for all $s \in \mathcal{S}, j \in \mathcal{J}$. Let $\boldsymbol{y}'$ be the primal solution to $\texttt{RSP}(\boldsymbol{x}, \mathcal{A}'_{sj})$ corresponding to dual solution $(\boldsymbol{\psi}, \boldsymbol{\gamma})$. At each column generation iteration and for each $s \in \mathcal{S}, j \in \mathcal{J}$, if $\texttt{RC} < 0$, we add the corresponding arc to $\mathcal{A}'_{sj}$. If $\texttt{RC} \geq 0$ for all $s \in \mathcal{S}, j \in \mathcal{J}$, then:

$$\bar{c}_{(m,n)} = g_{(m,n)} - (\psi_{sjm} - \psi_{sjn}) - \boldsymbol{\gamma}_{sj}^{\top} \boldsymbol{f}_{(m,n),s,j} \geq 0 \quad \forall (m,n) \in \mathcal{A}_{sj}$$

We construct a feasible solution $\boldsymbol{y}$ to $\overline{\texttt{SP}}(\boldsymbol{x})$ by letting $y_a = y'_a$ for all $a \in \mathcal{A}'_{sj}$ and $y_a = 0$ for all $a \in \mathcal{A}_{sj} \setminus \mathcal{A}'_{sj}$. Then, $\boldsymbol{y}$ is optimal for $\overline{\texttt{SP}}(\boldsymbol{x})$ by the non-negativity of the reduced costs. Thus, $\varphi'_{sj}(\boldsymbol{x}, \mathcal{A}'_{sj}) = \overline{\varphi}_{sj}(\boldsymbol{x})$. There are finitely many arcs in $\mathcal{A}_{sj}$ and at each iteration with $\texttt{RC} < 0$ for some $s \in \mathcal{S}, j \in \mathcal{J}$, at least one new arc is added to $\mathcal{A}_{sj}$. Thus, the algorithm converges to an optimal solution of $\overline{\texttt{SP}}(\boldsymbol{x})$ in finitely many iterations.

Then, we show that the outer Benders decomposition scheme converges to an optimal solution of the partial relaxation $(\texttt{MIO} - \texttt{LO})$. Suppose that, at iteration t in the outer Benders decomposition loop, we solve $\texttt{MP}(\mathcal{U}^t, \mathcal{V}^t)$ and obtain solution $(\boldsymbol{x}^t, \boldsymbol{\theta}^t)$. The value $\boldsymbol{c}^{\top}\boldsymbol{x}^t + \sum_{s \in \mathcal{S}} \sum_{j \in \mathcal{J}} \pi_s \theta^t_{sj}$ satisfies a subset of Benders cuts, and thus provides a valid lower bound for $(\texttt{MIO} - \texttt{LO})$. We then solve the relaxed subproblem $\overline{\texttt{SP}}(\boldsymbol{x}^t)$ via column generation (proved to converge above).

- If $\overline{\texttt{SP}}(\boldsymbol{x}^t)$ is unbounded, then $\exists s \in \mathcal{S}, j \in \mathcal{J}$ and a corresponding direction of unboundedness $(\boldsymbol{\psi}^v_{sj}, \boldsymbol{\gamma}^v_{sj})$ which is dual feasible (i.e. $(\boldsymbol{\psi}^v_{sj}, \boldsymbol{\gamma}^v_{sj}) \in \mathcal{P}_{sj}$) and $\sum_{n \in \mathcal{N}_{sj}} \sum_{k \in \mathcal{K}_j} b_{nsk} x^t_k \psi^v_{sjn} + \boldsymbol{h}_{sj}^{\top} \boldsymbol{\gamma}^v_{sj} > 0$. This generates a valid feasibility cut (18), which is clearly violated by $(\boldsymbol{x}^t, \boldsymbol{\theta}^t)$. We add the corresponding cut to the master problem, eliminating the incumbent solution.
- If $\overline{\texttt{SP}}(\boldsymbol{x}^t)$ admits a feasible solution, let $(\boldsymbol{\psi}^u, \boldsymbol{\gamma}^u)$ be the optimal dual solution. This generates a valid optimality cut (17). If $(\boldsymbol{x}^t, \boldsymbol{\theta}^t)$ violates the cut (i.e. there exists $s \in \mathcal{S}, j \in \mathcal{J}$ such that $\varphi_{sj}(\boldsymbol{x}^t) > \theta^t_{sj}$), then we add the corresponding cut to the master problem, eliminating the incumbent solution. We also obtain a valid upper bound for $(\texttt{MIO} - \texttt{LO})$ from $\boldsymbol{c}^{\top}\boldsymbol{x}^t + \sum_{s \in \mathcal{S}} \sum_{j \in \mathcal{J}} \pi_s \varphi_{sj}(\boldsymbol{x}^t)$. If no cuts are violated, then the solution is optimal for $(\texttt{MIO} - \texttt{LO})$.

Each Benders iteration either generates a new cut to exclude the incumbent $(\boldsymbol{x}^t, \boldsymbol{\theta}^t)$ or identifies that the incumbent satisfies all cuts. Because the set of cuts is finite, the algorithm terminates in finitely many iterations.

Upon convergence, we restore integrality requirements in the subproblem by finding a feasible integer solution to $\mathtt{SP}(\boldsymbol{x}^*)$, with optimal value given by $\mathit{\Phi}'(\boldsymbol{x}^*)$. The result is a feasible integer solution to Problem ($\star$) and thus $\boldsymbol{c}^\top \boldsymbol{x}^* + \mathit{\Phi}'(\boldsymbol{x}^*)$ provides an upper bound. The optimal solution to the partial relaxation ($\mathtt{MIO} - \mathtt{LO}$) provides a valid lower bound for Problem ($\star$). These together constitute a valid optimality gap for Problem ($\star$) upon convergence of the DD algorithm. □

EC.2.2. Application of the double-decomposition algorithm to the MiND-VRP

We provide details on the double decomposition algorithm applied to the MiND-VRP partial relaxation. Note that the problem has relatively complete resource, since it is always feasible to merely follow the reference line without deviations. Therefore, the methodology involves optimality cuts but no feasibility cuts.

In the MiND-VRP, the Benders subproblem is decomposable across reference trips $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$ and scenarios $s \in \mathcal{S}$. It is given by the following Benders subproblem, for each first-stage decision $(\boldsymbol{x}, \boldsymbol{z})$:

$$\overline{\varphi}_{sj}(\boldsymbol{x}) = \min_{\boldsymbol{y} \geq \boldsymbol{0}} \quad \sum_{a \in \mathcal{A}_{\ell st}} g_a y_a \qquad \text{s.t.} \quad \text{Equations (9)-(10)} \tag{EC.46}$$

Let $\boldsymbol{\psi}$ and $\boldsymbol{\gamma}$ respectively denote the dual variables corresponding to Equations (9) and (10), respectively. The dual Benders subproblem is then formulated as follows:

$$\begin{aligned}
\max \quad & x_{\ell t} \cdot (\psi_{\ell st u_{\ell st}} - \psi_{\ell st v_{\ell st}}) - \sum_{p \in \mathcal{P} : (\ell, t) \in \mathcal{M}_p} z_{\ell pst} \cdot \gamma_{\ell stp} && && \text{(EC.47)} \\
\text{s.t.} \quad & \psi_{\ell stn} - \psi_{\ell stm} - \sum_{p \in \mathcal{P}_a} \gamma_{\ell stp} \leq g_a && \forall a = (n, m) \in \mathcal{A}_{\ell st} && \text{(EC.48)} \\
& \psi_{\ell sti} \in \mathbb{R} && \forall i \in \mathcal{V}_{\ell st} && \text{(EC.49)} \\
& \gamma_{\ell stp} \geq 0 && \forall p \in \mathcal{P} : (\ell, t) \in \mathcal{M}_p && \text{(EC.50)}
\end{aligned}$$

Let $\Lambda_{\ell st}$ store the extreme points of the dual second-stage polyhedron, each corresponding to a second-stage solution $(\boldsymbol{\psi}, \boldsymbol{\gamma})$ for reference trip $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell$ and scenario $s \in \mathcal{S}$. Let $\boldsymbol{\Lambda} = (\Lambda_{\ell st})_{(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell, s \in \mathcal{S}}$ store all extreme points. The MiND-VRP partial relaxation optimizes network design and passenger assignments subject to a piece-wise linear recourse approximation:

$$\begin{aligned}
\text{BMP}(\boldsymbol{\Lambda}) \quad \min \quad & \sum_{\ell \in \mathcal{L}} \sum_{t \in \mathcal{T}_\ell} \left(h_\ell x_{\ell t} + \sum_{s \in \mathcal{S}} \pi_s \theta_{\ell st} \right) && \text{(EC.51)} \\
\text{s.t.} \quad & \text{Equations (1)–(4)} && \text{(EC.52)} \\
& \theta_{\ell st} \geq x_{\ell t} \cdot (\psi_{\ell st u_{\ell st}} - \psi_{\ell st v_{\ell st}}) - \sum_{p \in \mathcal{P} : (\ell, t) \in \mathcal{M}_p} z_{\ell pst} \cdot \gamma_{\ell stp}, \quad \forall (\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell, \ s \in \mathcal{S}, \ (\boldsymbol{\psi}, \boldsymbol{\gamma}) \in \Lambda_{\ell st} && \text{(EC.53)} \\
& \boldsymbol{x}, \boldsymbol{z} \text{ binary} && \text{(EC.54)}
\end{aligned}$$

The restricted Benders subproblem simply solves the Benders subproblem with a subset of subpath-based arcs by $\mathcal{A}'_{\ell st} \subseteq \mathcal{A}_{\ell st}$. It is formulated as follows:

$$\varphi'_{s,j}(\boldsymbol{x}, \mathcal{A}'_{sj}) = \min_{\boldsymbol{y}\geq \boldsymbol{0}} \quad \sum_{a\in\mathcal{A}'_{\ell st}} g_a y_a \tag{EC.55}$$

$$\text{s.t.} \quad \sum_{m:(n,m)\in\mathcal{A}'_{\ell st}} y_{(n,m)} - \sum_{m:(m,n)\in\mathcal{A}'_{\ell st}} y_{(m,n)} = \begin{cases} x_{\ell t} & \text{if } n = u_{\ell st} \\ -x_{\ell t} & \text{if } n = v_{\ell st} \\ 0 & \text{otherwise} \end{cases} \quad \forall n \in \mathcal{V}_{\ell st} \tag{EC.56}$$

$$\sum_{a\in\mathcal{A}'_{\ell st}: p\in\mathcal{P}_{r(a)}} y_a \leq z_{\ell pst} \quad \forall p \in \mathcal{P}: (\ell, t) \in \mathcal{M}_p \tag{EC.57}$$

EC.2.3. Single-tree implementation of the double-decomposition algorithm

Figure EC.2 visualizes the DD methodology in a single-tree Benders decomposition implementation. This implementation solves Problem ($\texttt{MIO} - \texttt{LO}$) using a single branch-and-cut tree, by solving the master problem relaxation (continuous first- and second-stage variables) at each node, and solving the Benders subproblem whenever an integral solution is found. In other words, multi-tree implementation, shown in Figure 3, relies on a cutting-plane version of Benders decomposition, whereas single-tree implementation adds Benders cuts via lazy constraints in the branch-and-cut tree.

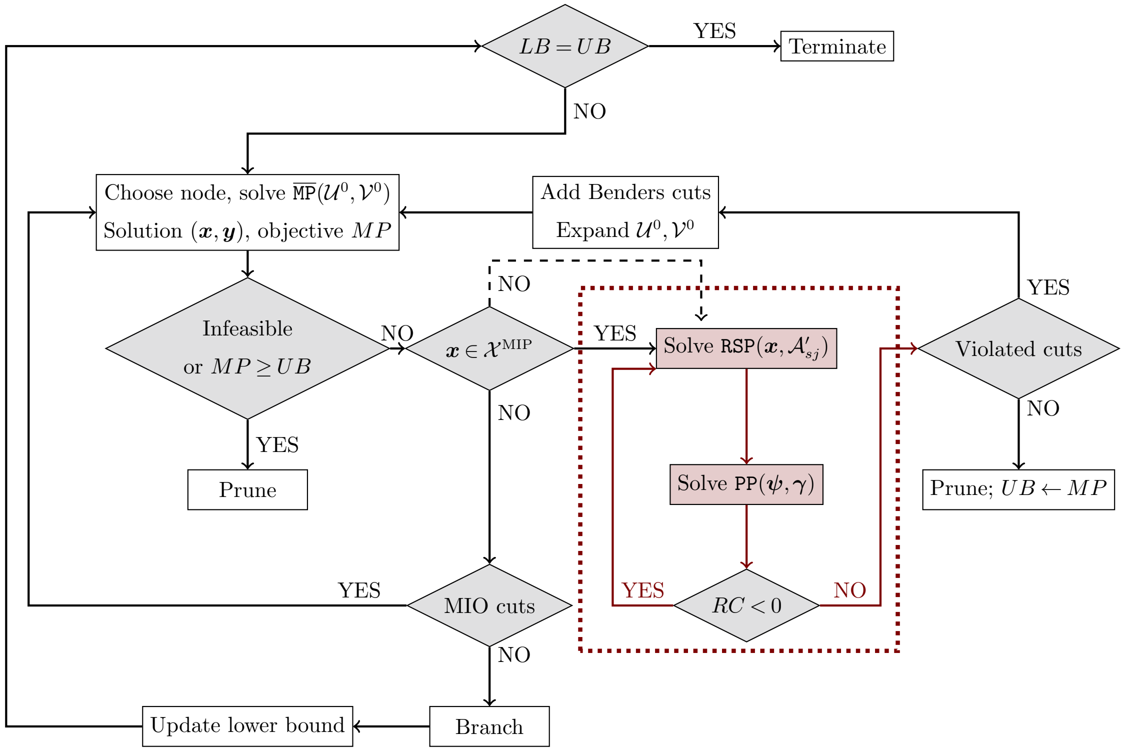

$LB = UB$
YES
Terminate
NO
Choose node, solve $\overline{\texttt{MP}}(\mathcal{U}^0, \mathcal{V}^0)$
Solution $(\boldsymbol{x}, \boldsymbol{y})$, objective MP
Add Benders cuts
Expand $\mathcal{U}^0, \mathcal{V}^0$
NO
YES
Infeasible
or $MP \geq UB$
NO
$\boldsymbol{x} \in \mathcal{X}^{\text{MIP}}$
YES
Solve $\texttt{RSP}(\boldsymbol{x}, \mathcal{A}'_{sj})$
Violated cuts
NO
YES
Prune
NO
Solve $\texttt{PP}(\boldsymbol{\psi}, \boldsymbol{\gamma})$
Prune; $UB \leftarrow MP$
YES
MIO cuts
YES
$RC < 0$
NO
NO
Update lower bound
Branch

Figure EC.2 **Single-tree DD algorithm to solve Problem** ($\texttt{MIO} - \texttt{LO}$).

EC.2.4. Details on the DD&ILS algorithm

Recall that $\underline{\varPhi}_{sj}$ denotes a global lower bound of the second-stage cost $\varphi_{sj}(\boldsymbol{x})$. We index the first-stage solutions visited throughout the algorithm by $\{\widehat{\boldsymbol{x}}^w : w \in \mathcal{W}^0\}$. The master problem, now referred to as $\texttt{MP-ILS}(\mathcal{U}^0, \mathcal{W}^0)$, comprises Benders optimality cut (it omits feasibility cuts due to the relatively complete recourse assumption) and integer L-shaped cuts, which express that $\theta_{sj} \geq \varphi_{sj}(\widehat{\boldsymbol{x}}^w)$ whenever $\boldsymbol{x} = \widehat{\boldsymbol{x}}^w$ and that $\theta_{sj} \geq \underline{\varPhi}_{sj}$ otherwise:

$$\begin{aligned}
\min \quad & \sum_{j\in\mathcal{J}}\sum_{k\in\mathcal{K}_j} c_k x_k + \sum_{s\in\mathcal{S}}\sum_{j\in\mathcal{J}} \pi_s \theta_{sj} && (\texttt{MP-ILS}(\mathcal{U}^0, \mathcal{W}^0))\\
\text{s.t.} \quad & \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}\\
& \theta_{sj} \geq \sum_{n\in\mathcal{N}_{sj}}\sum_{k\in\mathcal{K}_j} b_{nsk} x_k \psi^u_{sjn} + \boldsymbol{h}_{sj}^\top \boldsymbol{\gamma}^u_{sj}, \ \forall s\in\mathcal{S}, \ \forall j\in\mathcal{J}, \ \forall u\in\mathcal{U}^0_{sj}\\
& \theta_{sj} \geq \underline{\varPhi}_{sj} + (\varphi_{sj}(\widehat{\boldsymbol{x}}^w) - \underline{\varPhi}_{sj}) \left(1 - \sum_{k\in\mathcal{K}_j:\widehat{x}^w_k=1} (1-x_k) - \sum_{k\in\mathcal{K}_j:\widehat{x}^w_k=0} x_k\right), \ \forall s\in\mathcal{S}, j\in\mathcal{J}, w\in\mathcal{W}^0\\
& \boldsymbol{x} \in \mathcal{X}^{\text{MIO}}
\end{aligned}$$

The DD&ILS algorithm is shown in Figure EC.3, with our DD methodology highlighted in red and the integer L-shaped cuts in purple. The algorithm relies on the multi-tree implementation of the DD algorithm (Figure 3). Upon convergence, the DD algorithm returns an optimal solution to Problem ($\texttt{MIO} - \texttt{LO}$). We then restore integrality requirements in the second-stage problem and solve $\texttt{SP}(\boldsymbol{x})$. If the optimality gap lies within some tolerance, the algorithm terminates with an optimality guarantee for Problem ($\star$). Otherwise, the algorithm generates new integer L-shaped cuts. These cuts make the incumbent solutions $\{\theta_{sj} : s\in\mathcal{S}, j\in\mathcal{J}\}$ infeasible and the algorithm proceeds to the master problem. The overall DD&ILS algorithm therefore converges to an exact solution to Problem ($\star$) in a finite number of iterations, due to the binary first-stage structure.

Proof of Theorem 2 Consider master problem solution $\left(\widehat{\boldsymbol{x}}^t, \widehat{\boldsymbol{\theta}}^t\right)$ at Benders iteration t. Suppose that the double decomposition algorithm has converged, meaning that $\widehat{\theta}^t_{sj} \geq \overline{\varphi}_{sj}(\widehat{\boldsymbol{x}}^t)$ for all $s\in\mathcal{S}$ and $j\in\mathcal{J}$; in fact, $\theta^t_{sj} = \overline{\varphi}_{sj}(\widehat{\boldsymbol{x}}^t)$ by optimality (Theorem 1). We solve $\texttt{SP}(\widehat{\boldsymbol{x}}^t)$ and obtain integer optimal value $\varphi_{sj}(\widehat{\boldsymbol{x}}^t) \geq \overline{\varphi}_{sj}(\widehat{\boldsymbol{x}}^t)$ for all $s\in\mathcal{S}$ and $j\in\mathcal{J}$. Recall that $\texttt{SP}(\widehat{\boldsymbol{x}}^t)$ is feasible due to the assumption of relatively complete recourse. The integrality gap is defined as the difference in optimal value between the feasible solution to Problem ($\star$) and the optimal solution to the partial relaxation, that is:

$$\begin{aligned}
\texttt{GAP} &= \left(\boldsymbol{c}^\top\widehat{\boldsymbol{x}}^t + \sum_{s\in\mathcal{S}}\pi_s\sum_{j\in\mathcal{J}}\varphi_{sj}(\widehat{\boldsymbol{x}}^t)\right) - \left(\boldsymbol{c}^\top\widehat{\boldsymbol{x}}^t + \sum_{s\in\mathcal{S}}\pi_s\sum_{j\in\mathcal{J}}\widehat{\theta}^t_{sj}\right)\\
&= \sum_{s\in\mathcal{S}}\pi_s\sum_{j\in\mathcal{J}}(\varphi_{sj}(\widehat{\boldsymbol{x}}^t) - \widehat{\theta}^t_{sj})
\end{aligned}$$

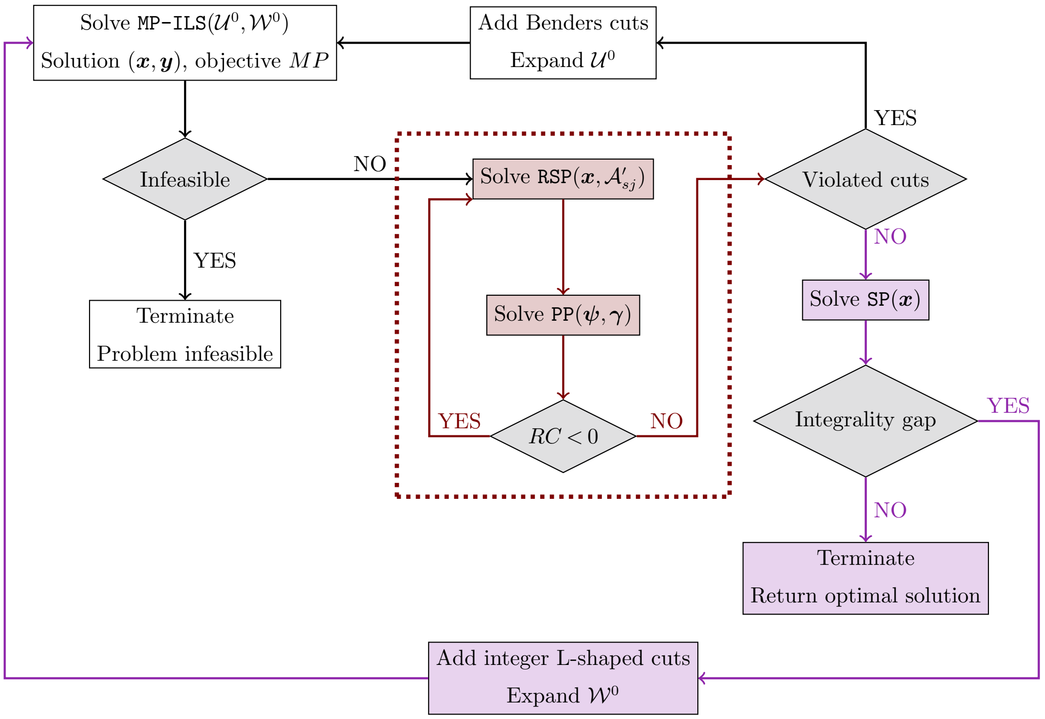

Solve MP-ILS($\mathcal{U}^0, \mathcal{W}^0$)
Solution ($\boldsymbol{x}, \boldsymbol{y}$), objective MP
Add Benders cuts
Expand $\mathcal{U}^0$
YES
Infeasible
NO
Solve RSP($\boldsymbol{x}, \mathcal{A}'_{sj}$)
Violated cuts
YES
NO
Terminate
Problem infeasible
Solve PP($\boldsymbol{\psi}, \boldsymbol{\gamma}$)
Solve SP($\boldsymbol{x}$)
YES
$RC < 0$
NO
Integrality gap
YES
NO
Terminate
Return optimal solution
Add integer L-shaped cuts
Expand $\mathcal{W}^0$

Figure EC.3 DD&ILS methodology to solve Problem ($\star$)**.**

If $\mathtt{GAP} \geq \varepsilon$ for a small tolerance $\varepsilon > 0$, then $\widehat{\theta}^t_{sj} < \varphi_{sj}(\widehat{\boldsymbol{x}}^t)$ for some $s \in \mathcal{S}, j \in \mathcal{J}$. We add the integer L-shaped cuts given in Equation (20) to the master problem, which evaluate to the valid inequality, $\theta_{sj} \geq \underline{\mathit{\Phi}}_{sj}$ for all $\boldsymbol{x} \neq \widehat{\boldsymbol{x}}^t$. Meanwhile, it eliminates the incumbent solution $\left(\widehat{\boldsymbol{x}}^t, \widehat{\boldsymbol{\theta}}^t\right)$ because it imposes that $\theta_{sj} \geq \varphi_{sj}(\widehat{\boldsymbol{x}}^t)$ when $\boldsymbol{x} = \widehat{\boldsymbol{x}}^t$:

$$\theta_{sj} \geq \underline{\mathit{\Phi}}_{sj} + (\varphi_{sj}(\widehat{\boldsymbol{x}}^t) - \underline{\mathit{\Phi}}_{sj})\left(1 - \underbrace{\sum_{k \in \mathcal{K}_j : \widehat{x}^t_k = 1} (1 - x_k) - \sum_{k \in \mathcal{K}_j : \widehat{x}^t_k = 0} x_k}_{=0 \text{ for } \boldsymbol{x} = \widehat{\boldsymbol{x}}^t}\right) \quad \forall s \in \mathcal{S}, j \in \mathcal{J}$$

If $\mathtt{GAP} < \varepsilon$, then the algorithm has converged and $(\widehat{\boldsymbol{x}}^t, \boldsymbol{\theta}^t)$ is an optimal solution of Problem ($\star$). Since $\boldsymbol{x} \in \{0,1\}^{nZ}$, there are finitely many first stage solutions $\boldsymbol{x}$ and thus finitely many cuts of the form (20). Therefore, the algorithm converges in a finite number of iterations to an optimal solution of Problem ($\star$). □

EC.2.5. Details on the UB&DD algorithm

At each node, the UB&BC algorithm solves the master problem relaxation $\overline{\mathtt{MP}}(\mathcal{U}^0, \mathcal{V}^0)$ and stores the first-stage solution $\widehat{\boldsymbol{x}}$. By design, $\overline{\mathtt{MP}}(\mathcal{U}^0, \mathcal{V}^0)$ incorporates a subset of Benders cuts and relaxes the integrality constraints. Therefore, if $\mathtt{MP}(\mathcal{U}^0, \mathcal{V}^0)$ is infeasible or returns a solution that is larger than an upper bound of Problem ($\star$), the branch can be pruned. If the solution does not satisfy first-stage integrality requirements, the algorithm proceeds to its branch-and-cut elements. Otherwise, it solves the second-stage relaxation to add Benders cuts or prove that none exist. These elements mirror the single-tree implementation of Benders decomposition (Figure EC.2); by themselves, they would solve the partial relaxation with mixed-integer first-stage variables and continuous second-stage variables (that is, Problem ($\texttt{MIO}-\texttt{LO}$)).

The difference occurs when the solution $\widehat{\boldsymbol{x}}$ of $\overline{\texttt{MP}}(\mathcal{U}^0, \mathcal{V}^0)$ satisfies first-stage integrality requirements and all Benders cuts at a given node. By design, $\widehat{\boldsymbol{x}}$ solves the full relaxation with continuous first- and second-stage variables at that node (because no Benders cut is violated), hence the partial relaxation with mixed-integer first-stage variables and continuous second-stage variables (because it satisfies the integrality constraints). For instance, at the root node, $\widehat{\boldsymbol{x}}$ solves ($\texttt{MIO}-\texttt{LO}$). However, $\widehat{\boldsymbol{x}}$ may not solve the full problem at the node. Unlike traditional single-tree implementations of Benders decomposition, the UB&BC algorithm proceeds with branching rather than pruning. It first updates a lower bound corresponding to the incumbent first-stage solution $\widehat{\boldsymbol{x}}$:

$$LB(\widehat{\boldsymbol{x}}) = \boldsymbol{c}^\top \widehat{\boldsymbol{x}} + \sum_{s\in\mathcal{S}} \sum_{j\in\mathcal{J}} \pi_s \overline{\varphi}_{sj}(\widehat{\boldsymbol{x}}) \leq \texttt{OPT}(\widehat{\boldsymbol{x}}) \tag{EC.58}$$

If the lower bound is higher than the upper bound, the branch can be pruned; otherwise, the algorithm stores the first-stage solution solves the mixed-integer second-stage problem—with an exact or heuristic algorithm—and updates the upper bound as needed:

$$UB \leftarrow \min\left\{UB, \texttt{OPT}(\widehat{\boldsymbol{x}})\right\} \geq \texttt{OPT} \tag{EC.59}$$

Upon termination, the algorithm returns the solution $\widehat{\boldsymbol{x}}$ with the smallest value of $\texttt{OPT}(\widehat{\boldsymbol{x}})$.

The UB&DD algorithm, shown in Figure EC.4, embeds our DD methodology (highlighted in red) into the UB&BC methodology from Mahéo et al. (2024) (in blue).

The final question involves solving the second-stage subproblem $\texttt{SP}(\boldsymbol{x})$ at each node where the UB&BC component (blue part in Figure EC.4) is visited. At that point, the algorithm has already solved the second-stage relaxation $\overline{\texttt{SP}}(\boldsymbol{x})$ via column generation. We can then derive a feasible second-stage solution by solving $\texttt{RSP}(\boldsymbol{x}, \mathcal{A}'_{sj})$ upon restoring integrality requirements. This is a common heuristic in column generation, which consists of solving a mixed-integer restricted master problem (i.e., the restricted Benders subproblem in our DD methodology) upon termination. The alternative would be to embed the column generation algorithm into a branch-and-price algorithm in each relevant node, which would come with huge computational costs. In fact, the column-generation heuristic is consistent with the approach from Mahéo et al. (2024), who also solve $\texttt{SP}(\widehat{\boldsymbol{x}})$ via a heuristic in the branch-and-bound tree. Following Mahéo et al. (2024), we then add a post-processing procedure to determine the optimal solution among all visited solutions $\widehat{\boldsymbol{x}}$.

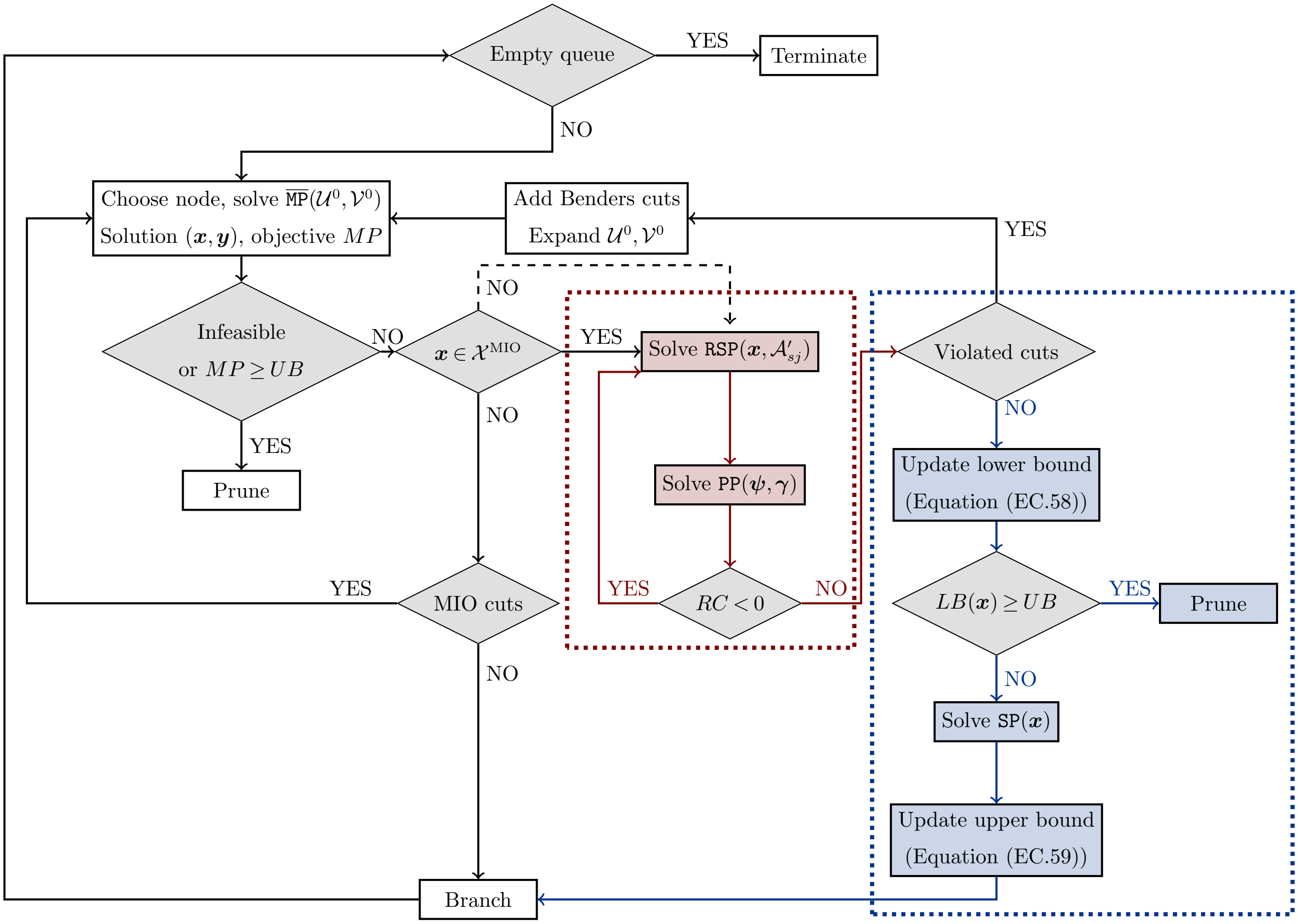

Empty queue
YES
Terminate
NO
Choose node, solve $\overline{\texttt{MP}}(\mathcal{U}^0,\mathcal{V}^0)$
Solution $(\boldsymbol{x},\boldsymbol{y})$, objective MP
Add Benders cuts
Expand $\mathcal{U}^0,\mathcal{V}^0$
YES
NO
Infeasible or $MP \geq UB$
NO
$\boldsymbol{x} \in \mathcal{X}^{\text{MIO}}$
YES
Solve $\texttt{RSP}(\boldsymbol{x},\mathcal{A}'_{sj})$
Violated cuts
NO
YES
Prune
Solve $\texttt{PP}(\boldsymbol{\psi},\boldsymbol{\gamma})$
Update lower bound (Equation (EC.58))
YES
MIO cuts
YES
$RC < 0$
NO
$LB(\boldsymbol{x}) \geq UB$
YES
Prune
NO
NO
Solve $\texttt{SP}(\boldsymbol{x})$
Update upper bound (Equation (EC.59))
Branch

Figure EC.4 UB&DD methodology to solve Problem $(\star)$.

Proof of Theorem 3 Mahéo et al. (2024) showed that UB&BC converges in a finite number of iterations to an optimal solution of the two-stage stochastic mixed-integer optimization problem. In the UB&BC scheme, nodes cannot be pruned by integrality, and so branching continues until all first-stage solution nodes have been explored or pruned (by bound or infeasibility). Thus, the algorithm stores a set of mixed-integer first-stage solutions, and evaluates them in a post-processing phase where integrality requirements are restored in the second-stage problem. As shown by Mahéo et al. (2024), an optimal first-stage solution $\boldsymbol{x}^*$ for Problem $(\star)$ will be stored in the pool because:

(i) The node of an optimal first-stage solution $\boldsymbol{x}^*$ will be visited. Assume that no optimal solution is in the pool, hence the upper bound satisfies $UB > \texttt{OPT}$. Any parent node of $\boldsymbol{x}^*$ solves a relaxation, so it will not be pruned by infeasibility; moreover, it yield a solution at most equal to $\boldsymbol{c}^\top\boldsymbol{x}^* + \overline{\Phi}(\boldsymbol{x}^*) \leq \boldsymbol{c}^\top\boldsymbol{x}^* + \Phi(\boldsymbol{x}^*) = \texttt{OPT} < UB$, so it will not be pruned by bound.

(ii) In the node itself, $\boldsymbol{x}^*$ cannot be eliminated by Benders cuts (17)-(18) because the cuts are valid for the relaxation $(\texttt{MIO}-\texttt{LO})$, and thus for Problem $(\star)$.

Combining that result with the exactness of our DD methodology for the partial relaxation ($\texttt{MIO}-\texttt{LO}$) from Theorem 1, we obtain that UB&DD converges to an exact solution to Problem ($\star$).

EC.2.6. Proof of Proposition 4

Fix a reference trip $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$ and a scenario $s\in\mathcal{S}$.

Let us consider a pair of checkpoints $(u,v)\in\Gamma_\ell$ and two load values $c_1\le c_2\in\mathcal{C}_\ell$. Let us define the load differential as $\nu=c_2-c_1$. By construction, the load component of the reduced cost satisfies:

$$\Delta\psi_{\ell st}^{u,v,\nu}\ge\psi_{\ell,s,t,(u,c_1)}-\psi_{\ell,s,t,(v,c_2)} \tag{EC.60}$$

Consider a solution $\boldsymbol{f}^*$, $\boldsymbol{w}^*$, $\boldsymbol{\xi}^*$ of the pricing problem $\mathrm{PP}_{\ell st}^{u,v,c_1,c_2}$. With a slight abuse of notation, we also refer to its optimal value as $\mathrm{PP}_{\ell st}^{u,v,c_1,c_2}$. By construction, the solution $\boldsymbol{f}^*$, $\boldsymbol{w}^*$ defines a feasible solution to the problem defining $Z_{\ell st}^{u,v,\nu}$. Indeed, the load differential satisfies

$$\sum_{m\in\mathcal{U}_{\ell st}^{uv}}\sum_{p\in\mathcal{P}_m}D_{ps}w_{mp}^*=\sum_{(m,q)\in\mathcal{H}_{\ell st}^{uv}\,:\,f_{mq}=1}(\xi_q^*-\xi_m^*)=\xi_{(v,T_{\ell t}(v))}^*-\xi_{(u,T_{\ell t}(u))}^*=c_{(v,c_2)}-c_{(u,c_1)}=\nu,$$

where the first equality is induced by Equations (24)–(25), the second equality is induced by telescoping the sum from Equation (28), the third equality is induced by Equation (23), and the last inequality is by assumption.

Therefore, the routing component of the reduced cost expression satisfies:

$$Z_{\ell st}^{u,v,\nu}\le\sum_{m\in\mathcal{U}_{\ell st}^{uv}}\sum_{p\in\mathcal{P}_m}d_{mp}w_{mp}^* \tag{EC.61}$$

From Equations (EC.60) and (EC.61), we obtain:

$$Z_{\ell st}^{u,v,\nu}-\Delta\psi_{\ell st}^{u,v,\nu}\le\sum_{m\in\mathcal{U}_{\ell st}^{uv}}\sum_{p\in\mathcal{P}_m}d_{mp}w_{mp}^*+\psi_{\ell,s,t,(v,c_2)}-\psi_{\ell,s,t,(u,c_1)}=\mathrm{PP}_{\ell st}^{u,v,c_1,c_2}$$

By taking the minimum over all arcs with a load differential ν, we obtain:

$$Z_{\ell st}^{u,v,\nu}-\Delta\psi_{\ell st}^{u,v,\nu}\le\min_{c_1,c_2\in\mathcal{C}_\ell:c_2-c_1=\nu}\mathrm{PP}_{\ell st}^{u,v,c_1,c_2},\ \forall(u,v)\in\Gamma_\ell \tag{EC.62}$$

Vice versa, let us consider two checkpoints $(u,v)\in\Gamma_\ell$ and a load differential $\nu\in\mathcal{C}_\ell$. Consider an arc $a^*\in\mathcal{A}_{\ell st}$ that maximizes the load component of the reduced cost and a solution $\boldsymbol{f}^*$, $\boldsymbol{w}^*$ that minimizes the routing component for that load differential. Specifically, the arc $a^*\in\mathcal{A}_{\ell st}$ defines a subpath that starts in checkpoint $u=k_{start(a^*)}\in\mathcal{I}_\ell$ at time $T_{\ell t}(u)$ with vehicle load $c_{start(a^*)}$, that ends in checkpoint $v=k_{end(a^*)}\in\mathcal{I}_\ell$ at time $T_{\ell t}(v)$ with load $c_{end(a^*)}=c_{start(a^*)}+\nu$, and that satisfies

$$\psi_{\ell,s,t,start(a^*)}-\psi_{\ell,s,t,end(a^*)}=\Delta\psi_{\ell st}^{u,v,\nu}$$

The solution $\boldsymbol{f}^*$, $\boldsymbol{w}^*$ satisfies Equations (26)–(29) by construction. We then define a load variable ξ_m, keeping track of the load at node $m \in \mathcal{U}^{u,v}_{\ell st}$. We initialize it with:

$$\xi_{(u,T_{\ell t}(u))} = c_{start(a^*)}.$$

Following solution $\boldsymbol{f}^*$, $\boldsymbol{w}^*$, we increase ξ_m by $\sum_{p\in\mathcal{P}_m} D_{ps}w^*_{mp}$ if we traverse $(m,q)\in\mathcal{H}^{u,v}_{\ell st}$:

$$\xi_q - \xi_m = \sum_{p\in\mathcal{P}_m} D_{ps}w^*_{mp}, \qquad \forall (m,q)\in\mathcal{H}^{u,v}_{\ell st} : f^*_{mq} = 1.$$

The variables ξ_m satisfy Equations (24)–(25) and (29) by construction. By combining it with Equations (28), and telescoping the sum, we obtain:

$$\begin{aligned}
\xi_{(v,T_{\ell t}(v))} &= \sum_{m\in\mathcal{U}^{uv}_{\ell st}:f^*_{m,(v,T_{\ell t}(v))}=1} \left(\xi_m + \sum_{p\in\mathcal{P}_m} D_{ps}w^*_{mp}\right)\\
&= \xi_{(u,T_{\ell t}(u))} + \sum_{(m,q)\in\mathcal{H}^{uv}_{\ell st}:f^*_{mq}=1}\sum_{p\in\mathcal{P}_m} D_{ps}w^*_{mp}\\
&= c_{start(a^*)} + \nu\\
&= c_{end(a^*)},
\end{aligned}$$

where the third equality comes from the initialization $\xi_{(u,T_{\ell t}(u))} = c_{start(a^*)}$ and the constraint $\sum_{m\in\mathcal{U}^{uv}_{\ell st}}\sum_{p\in\mathcal{P}_m} D_{ps}w_{mp} = \nu$, and the last equality follows from the construction of $a^*\in\mathcal{A}_{\ell st}$. Therefore, the variables ξ_m also satisfy Equations (23).

Therefore, solution $\boldsymbol{f}^*$, $\boldsymbol{w}^*$, ξ defines a feasible solution for the pricing problem $\mathrm{PP}^{u,v,c_{start(a^*)},c_{end(a^*)}}_{\ell st}$, and we have:

$$Z^{u,v,\nu}_{\ell st} - \Delta\psi^{u,v,\nu}_{\ell st} = \sum_{p\in\mathcal{P}_m} D_{ps}w^*_{mp} + \psi_{\ell,s,t,end(a^*)} - \psi_{\ell,s,t,start(a^*)}$$

Since, by construction, $c_{end(a^*)} - c_{start(a^*)} = \nu$, we obtain:

$$Z^{u,v,\nu}_{\ell st} - \Delta\psi^{u,v,\nu}_{\ell st} \geq \min_{c_1,c_2\in\mathcal{C}_\ell:c_2-c_1=\nu} \mathrm{PP}^{u,v,c_1,c_2}_{\ell st}.$$

This completes the proof that $Z^{u,v,\nu}_{\ell st} - \Delta\psi^{u,v,\nu}_{\ell st}$ is equal to the minimum reduced cost across all variables with load differential ν:

$$Z^{u,v,\nu}_{\ell st} - \Delta\psi^{u,v,\nu}_{\ell st} = \min_{c_1,c_2\in\mathcal{C}_\ell:c_2-c_1=\nu} \mathrm{PP}^{u,v,c_1,c_2}_{\ell st}$$

This completes the proof. □

EC.2.7. Proof of Remark 1

Suppose that Equation (28) is replaced with the following constraints in the pricing problem.

$$\sum_{q\,:\,(m,q)\in\mathcal{H}^{uv}_{\ell st}} f_{mq} - \sum_{q\,:\,(q,m)\in\mathcal{H}^{uv}_{\ell st}} f_{qm} = \begin{cases} x_{\ell t} & \text{if } m=(u,T_{\ell t}(u)),\\ x_{\ell t} & \text{if } m=(v,T_{\ell t}(v)),\\ 0 & \text{otherwise.}\end{cases} \qquad \forall m\in\mathcal{U}^{uv}_{\ell st} \tag{EC.63}$$

With Equation (EC.63), the resulting optimal solution to the pricing problem could not be used to construct a subpath, which by definition is a sequence of arcs connecting checkpoints u and v.

Suppose toward a contradiction that Equation (27) is replaced with the following constraints in the pricing problem.

$$\sum_{m\in\mathcal{U}^{uv}_{\ell st}\,:\,p\in\mathcal{P}_m} w_{mp} \le z_{\ell pst} \qquad \forall p\in\mathcal{P}\,:\,(\ell,t)\in\mathcal{M}_p \tag{EC.64}$$

Consider the subset of passengers $p\in\mathcal{P}_{\ell st}$ for which $z_{\ell pst}=0$. Then the pricing problem only constructs arcs over the following set:

$$\mathcal{A}_{\ell st}(\boldsymbol{z}) := \{a\in\mathcal{A}_{\ell st}\,:\, z_{p\ell st}=1, \forall p\in\mathcal{P}_{r(a)}\}.$$

As a result, the optimal dual solution $(\boldsymbol{\psi},\boldsymbol{\gamma})$ to the corresponding RMP would have unknown feasibility to the following constraints:

$$\psi_{\ell stn} - \psi_{\ell stm} - \sum_{p\in\mathcal{P}_a}\gamma_{\ell stp} \le g_{(n,m)} \qquad \forall (n,m)\in\mathcal{A}_{\ell st}\setminus\mathcal{A}_{\ell st}(\boldsymbol{z}). \tag{EC.65}$$

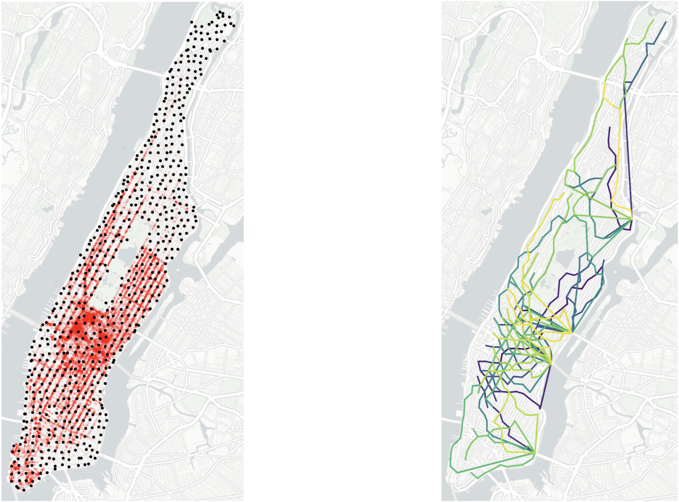
Thus, the solution $(\boldsymbol{\psi},\boldsymbol{\gamma})$ is not necessarily in $\Lambda_{\ell st}$, and the corresponding optimality cut would be violated in the Benders decomposition algorithm. □

EC.3. Experimental Setup

In this appendix, we provide details on the generation of the model inputs (EC.3.1); in particular, we present a breadth-first search tree approach to define candidate reference lines (EC.3.2). Figure EC.5 illustrates these inputs. We also detail our ride-sharing benchmarks (EC.3.3).

EC.3.1. Model Inputs

We developed a real-world experimental setup in Manhattan, using data from the NYC Taxi & Limousine Commission (2021). We filtered trips to the airports during the morning rush (6–9 am), leading to up to 1,900 passenger request per instance (shown in Figure EC.5a of the paper). We defined a road network and travel times using data from Google Maps, OpenStreetMap, and Uber (2020). We considered pickup stations 300 meters apart, leading to 640 stations (also shown in Figure EC.5a of the paper). We assumed passengers could originate from any of the approximately 20,000 roadway intersections in Manhattan, and that they would walk from their origin to the



(a) Demand and stations. (b) Reference lines.

Figure EC.5 Visualization of MiND-VRP inputs in Manhattan.

closest station. We obtained the mapping and routing inputs from the `fastest_route` functionality in the OpenStreetMapX package in Julia (Szufel, Przemysłlaw *et al.* 2023). We calibrated travel time estimates to heavy Manhattan traffic using speed data from Uber (2020). We computed average speeds during the morning rush for each roadway type present in our Manhattan map (primary, secondary, tertiary, unclassified) and used these average speeds as input to the travel time estimation function, overriding default speeds provided by OpenStreetMapX.

Recall that our MiND-VRP experiments model a shuttle service from Manhattan to LaGuardia Airport with vehicles of capacity 10 to 20 passengers. Every trip leaves Manhattan and heads directly toward LaGuardia Airport via four possible exits: the Queensboro Bridge, the Williamsburg Bridge, the Kennedy Bridge, and the Midtown Tunnel. Travel times from each exit to LaGuardia were obtained via Google Maps estimates during the morning rush.

Table EC.4 reports the parameter values used in our computational experiments (Section 5), and practical experiments for the MiND-VRP (Section 6) and MiND-DAR (Appendix EC.4.3).

EC.3.2. Reference Line Generation

We describe the process of generating the set $\mathcal{L}$ of candidate reference lines (shown in Figure EC.5b of the paper). The procedure proceeds in three steps: (i) generating a comprehensive routing graph over Manhattan; (ii) using breadth-first search (BFS) trees to generate a very large set of candidate reference lines; and (iii) clustering and filtering to obtain a small but representative final set of candidate reference lines. We describe each step in detail below.

Note that our procedure to construct and optimize reference lines relies on a training set of demand data. This process avoids any bias moving from design to evaluation.

Table EC.4 Details on input calibration for computational and practical analyses.

Model component	Section 5 value(s)	Section 6 value(s)	EC.4.3 value(s)	EC.5.3 value(s)
Ω	210 meters	420 meters	250 meters	210 meters
Ψ	10 minutes	10 minutes	10 minutes	10 minutes
Δ	600 meters	600 or 1,200 meters	300 meters	600 meters
α	5 minutes	10 minutes	10 minutes	10 minutes
C_ℓ	10 people	10, 15, or 20 people	5, 10 or 20 people	10 people
$\mathcal{T}_S$	30 seconds	30 seconds	30 seconds	30 seconds
μ	1	1	1	1
λ	1	1	1	1
σ	1	1	1	1
δ	1	1	1	1
κ	1	1	1	1
M	10,000	10,000	10,000	10,000
h_ℓ	$T_{\ell t}(\mathcal{I}_\ell^{(I_\ell)}) - T_{\ell t}(\mathcal{I}_\ell^{(1)})$	$T_{\ell t}(\mathcal{I}_\ell^{(I_\ell)}) - T_{\ell t}(\mathcal{I}_\ell^{(1)})$	$T_{\ell t}(\mathcal{I}_\ell^{(I_\ell)}) - T_{\ell t}(\mathcal{I}_\ell^{(1)})$	$T_{\ell t}(\mathcal{I}_\ell^{(I_\ell)}) - T_{\ell t}(\mathcal{I}_\ell^{(1)})$
F	$\lvert\mathcal{L}\rvert$ vehicles	10 or 20 vehicles	5 or 10 vehicles	$\lvert\mathcal{L}\rvert$ vehicles
$\mathcal{T}_\ell$	15 minute intervals	15 minute intervals	15 minute intervals	15 minute intervals
$T_{\ell t}(\mathcal{I}_\ell^{(i+1)}) - T_{\ell t}(\mathcal{I}_\ell^{(i)})$	120% of direct	120% of direct	110% of direct	120% of direct

$T_{\ell t}(\mathcal{I}_\ell^{(i+1)}) - T_{\ell t}(\mathcal{I}_\ell^{(i)})$: buffer time between arrival times at consecutive checkpoints $\mathcal{I}_\ell^{(i)}$ and $\mathcal{I}_\ell^{(i+1)}$.
$\mathcal{T}_\ell$: the frequency set is populated with departure times at evenly spaced intervals across the demand horizon.
$\mathcal{T}_S$: Time elapsed between consecutive discrete time units (between t and $t+1$) in the discretized set $\mathcal{T}^S$.

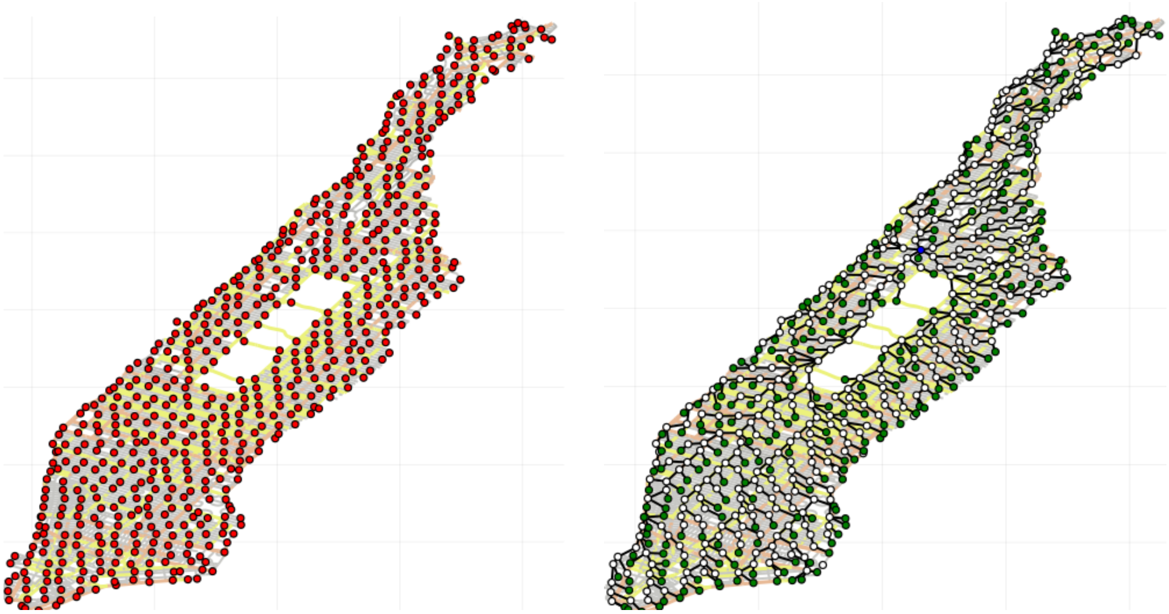
Manhattan routing graph. We build a node set using discrete locations in Manhattan by generating a grid of GPS coordinates spanning Manhattan that were each 300 meters apart, and snapping each node to the closest road intersection. The outcome of this process is a list of candidate checkpoints $\mathcal{N}$, shown in Figure EC.6a. We then build an edge set over this routing graph by connecting each node to its six closest neighbors according to their Euclidean distance. We used OpenStreetMapX to remove any edges that were impossible for a vehicle to traverse.

BFS trees. To generate a large set of reference line candidates, we build BFS trees over the routing graph. Specifically, we let each node be the root of a BFS tree over the routing network (see Figure EC.6b). We then build reference line candidates over each BFS tree, by constructing node sequences from the root node to each leaf. Ultimately, we obtain tens of thousands of distinct candidate reference lines, across all BFS trees.

Clustering and filtering. We first filter out many candidate lines that are illogical (e.g., indirect lines, very short or very long lines). We developed several metrics of line quality to systematically filter out low-quality options:

– *Minimum number of checkpoints.* Each line must visit a minimum of 10 stations.

– *Low average and maximum detour.* For each checkpoint, we compute the relative detour as the ratio of the travel time from the checkpoint to the destination (LaGuardia) with the reference line and the corresponding direct travel time. The average detour across all checkpoints should not exceed 200%, and the maximum detour should not exceed 250%.



(a) Candidate checkpoints. (b) Breadth-first search tree.

Figure EC.6 Candidate checkpoints and BFS tree (blue: root node; green: leaves; white: intermediate nodes)

- *Limited wrong-way travel.* To measure travel in the "wrong direction," we measure the percentage of a reference line's checkpoints that are farther away from LaGuardia than their immediate predecessors.
- *Demand coverage.* We assigned a popularity score to each checkpoint based on the frequency of trip requests with pickup locations close to that stop—in a training dataset. We filter out lines with a low average popularity score across its checkpoints.

Then, we remove redundancy over overlapping candidate lines, which is especially present among lines constructed from the same BFS tree. We measure the dissimilarity of two candidate lines as:

$$\text{dissim}_{k\ell} = 1 - \frac{|\mathcal{I}_k \cap \mathcal{I}_\ell|}{\min\{I_k, I_\ell\}}.$$

When $\text{dissim}_{k\ell} = 0$, lines k and ℓ share as many stops as possible and are therefore substitutable. We collect these substitutable pairs into an undirected graph, and define an updated set of candidate lines $\mathcal{L}'$ by computing a minimum vertex cover over that graph.

At this point, we are left with approximately 3,000 candidate lines in $\mathcal{L}'$. In order to retain a tractable set of candidate lines in the optimization model, we cluster them into 100 representative and high-quality options. Specifically, we formulate a bi-objective clustering model to maximize medoid quality and diversity. Let $y_\ell \in \{0,1\}$ indicate whether line $l \in \mathcal{L}'$ is selected in the final set $\mathcal{L}$, and $x_{k\ell} \in \{0,1\}$ indicate whether line $k \in \mathcal{L}'$ is assigned to the cluster with medoid line $l \in \mathcal{L}'$. We define a line-dependent parameter q_ℓ penalizing undesirable line characteristics based on the aforementioned metrics.

The clustering model maximizes line quality and minimizes the total dissimilarity among the line mapping (Equation (EC.66)), subject to partitioning constraints (Equation (EC.67)), consistency constraints (Equation (EC.68)) and budget constraints (Equation (EC.69)). We define the final reference line set as $\mathcal{L} := \{l \in \mathcal{L}' : y_\ell = 1\}$.

$$\begin{aligned}
\min \quad & \sum_{\ell \in \mathcal{L}'} q_\ell y_\ell + \lambda \sum_{k \in \mathcal{L}'} \sum_{\ell \in \mathcal{L}'} \mathrm{dissim}_{k\ell} x_{k\ell} && && \text{(EC.66)} \\
\text{s.t.} \quad & \sum_{k \in \mathcal{L}'} x_{k\ell} = 1 && \forall l \in \mathcal{L}' && \text{(EC.67)} \\
& x_{k\ell} \leq |\mathcal{L}'| y_\ell && \forall k, l \in \mathcal{L}' && \text{(EC.68)} \\
& \sum_{\ell \in \mathcal{L}'} y_\ell = 100 && && \text{(EC.69)} \\
& \boldsymbol{x} \in \{0,1\}^{\mathcal{L}' \times \mathcal{L}'} && && \text{(EC.70)} \\
& \boldsymbol{y} \in \{0,1\}^{\mathcal{L}'} && && \text{(EC.71)}
\end{aligned}$$

We constructed three candidate line sets with 100 lines each by scaling the aforementioned quality measures with the following parameter settings:

$$\begin{aligned}
q_\ell^{\text{cluster}} &= 0 && \forall \ell \in \mathcal{L}' \\
q_\ell^{\text{direct}} &= \frac{1}{3} \cdot (\mathrm{maxDetour}_\ell + \mathrm{meanDetour}_\ell + \mathrm{wrongWay}_\ell) && \forall \ell \in \mathcal{L}; \\
q_\ell^{\text{popular}} &= \mathrm{popularity}_\ell && \forall \ell \in \mathcal{L}'
\end{aligned}$$

Throughout the manuscript, we use $\boldsymbol{q}^{\text{popular}}$ as the default to focus on the demand coverage objective, except for Section 6.1 on microtransit network design, in which we consider the line sets corresponding to all three quality measures.

EC.3.3. Ride-sharing Benchmark

We build our ride-sharing benchmark using the cluster-then-route heuristic from Bertsimas and Yan (2021), originally built to generate paratransit itineraries with up to 4 passengers per vehicle. Their approach was itself based on the maximum weighted matching over a shareability network from Santi et al. (2014). To extend the approach from two- to four-passenger trips, Bertsimas and Yan (2021) first created a set of passenger pairs and then approximated the shareability network over two-passenger trips. We adopt a similar approach except that, instead of requiring all requests to be served, we maximize the number of served requests and then minimize travel times.

Single-occupancy ride-sharing. With single-occupancy vehicles, the clustering step is unnecessary. We simply apply the routing step from Bertsimas and Yan (2021) over the request set.

Two-occupancy ride-sharing. We build a pair-wise shareability network that encodes the pairs of trips that can share a vehicle. Let t_i denote the requested pickup time of request i, T_i the direct travel time of request i, and $tt(x,y)$ the travel time from location x to location y.

– If $t_j \le t_i + T_i + \Psi$, then trip j can be picked up before trip i is dropped off;

– if $t_i \le t_j + T_j + \Psi$, then trip i can be picked up before trip j is dropped off; and

– otherwise, trips i and j cannot be shared.

Then we determine whether there exists pickup times for trips i and j (in that order) such that no request is picked up early and each pickup is within Ψ of their requested times. The following conditions must hold, where x denotes the pickup time of trip i:

$$\begin{aligned} & t_i \le x \le t_i + \Psi && \text{Request } i \text{ has tolerable wait time} \\ & t_j \le x + tt(o_i,o_j) \le t_j + \Psi && \text{Request } j \text{ has tolerable wait time} \end{aligned}$$

which reduces to finding some x such that:

$$x \in [\max\{t_i, t_j - tt(o_i,o_j)\}, \min\{t_i + \Psi, t_j + \Psi - tt(o_i,o_j)\}].$$

The two requests can also share a vehicle if the symmetric problem holds, corresponding to the instance where trip j is picked up first:

$$x \in [\max\{t_j, t_i - tt(o_j,o_i)\}, \min\{t_j + \Psi, t_i + \Psi - tt(o_j,o_i)\}]$$

Bertsimas and Yan (2021) impose a maximum delay limit, but we remove this restriction to enable more ride-pooling. Finally, we determine the travel time associated with each version of the trip.

$$\begin{aligned} c_{i\to j} &= tt(o_i,o_j) + T_j \\ c_{j\to i} &= tt(o_j,o_i) + T_i \end{aligned}$$

If $c_{i\to j} \le T_i + T_j$ or $c_{j\to i} \le T_j + T_i$, then the shared trip is more efficient than serving the two requests separately. If both are efficient, then we select the best option.

The shared trips satisfying the above conditions are added to the VSN with cost $T_i + T_j - \min\{c_{i\to j}, c_{j\to i}\}$ to reflect the cost savings of pooling the requests. We solve a maximum weighted matching problem to pair requests into capacity-2 trips, with some requests potentially still served in isolation if they are not matched to any other request. We first maximize the number of served requests, and then we minimize the total travel time, subject to the fleet size limit.

Four-occupancy ride-sharing. We build a new shareability network that combines trip pairs from the pair-wise shareability network. For the MiND-VRP, we solve a simple vehicle routing problem for each candidate set of four trips to find the best sequence of stops within that set, while ensuring that no one is picked up earlier than their requested times and that none of passengers' wait times exceeds limit Ψ. For the MiND-DAR, we solve a simple dial-a-ride problem for each candidate set of four trips, which also includes precedence constraints so that each pickup occurs before the corresponding dropoff. We note that the optimal pooling configuration of two request pairs could potentially be to serve all four requests together, or to pool only a subset of these requests and serve the remaining requests separately. We proceed as in the two-occupancy case, solving a maximum weighted matching problem over the VSN to determine final trips, and then performing an identical itinerary generation procedure to the one described previously.

EC.4. Extension to the dial-a-ride setting (MiND-DAR)

EC.4.1. Modeling extension

In the dial-a-ride setting, each passenger request $p \in \mathcal{P}$ is associated with an origin $o(p)$ and a destination $d(p)$. The first-stage formulation remains unchanged, except that the set $\mathcal{M}_p$ is re-defined as the set of reference lines that cover both the origin and the destination of request $p \in \mathcal{P}$. In the second stage, we define the sets $\mathcal{P}_r^+$ and $\mathcal{P}_r^-$ ($\mathcal{P}_r = \mathcal{P}_r^+ \cup \mathcal{P}_r^-$) as the passenger requests that are picked up and dropped off, respectively, by subpath $r \in \mathcal{R}_{\ell st}$ for $(\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell, s \in \mathcal{S}$.

Level of service involves similar measures of passenger disutility. A subpath $r \in \mathcal{R}_{\ell st}$ is associated with a walking cost both for pickups (from the origin of passenger $p \in \mathcal{P}_r^+$ to the pickup location) and for dropoffs (from the dropoff location to the destination of passenger $p \in \mathcal{P}_r^-$); with a waiting cost for pickups; and with a delay cost for dropoffs. To capture detour costs, we denote by $T^+_{r(a)p}$ (resp. $T^-_{r(a)p}$) the pickup (resp. dropoff) time of passenger p on arc $a \in \mathcal{A}_{\ell st}$ such that $p \in \mathcal{P}^+_{r(a)}$ (resp. $p \in \mathcal{P}^-_{r(a)}$). The arc costs g_a are re-derived as follows.

$$g_a^{DAR} = \begin{cases} \sum_{p \in \mathcal{P}^+_{r(a)}} D_{ps} \left(\lambda \tau^{walk}_{r(a)p} + \mu \tau^{wait}_{r(a)p} - \sigma \frac{T^+_{r(a)p}}{\tau^{dir}_p} - M \right) + & \\ \qquad \sum_{p \in \mathcal{P}^-_{r(a)}} D_{ps} \left(\lambda \tau^{walk}_{r(a)p} + \sigma \frac{T^-_{r(a)p}}{\tau^{dir}_p} + \delta \frac{\tau^{late}_{\ell r(a)p}}{\tau^{dir}_p} + \frac{\delta}{2} \frac{\tau^{early}_{\ell r(a)p}}{\tau^{dir}_p} \right) & \forall a \in \bigcup_{r \in \mathcal{R}_{\ell st}} \mathcal{A}_r, \\ 0 & \forall a \in \mathcal{A}^v_{\ell st}. \end{cases} \tag{EC.72}$$

The MiND-DAR is then formulated as follows. The only difference with the MiND-VRP is the additional constraint ensuring that a passenger who is picked up needs to be dropped off (Equation (EC.76)). Note that the precedence constraint is captured by the set $\mathcal{M}_p$ and therefore does not need to be enforced explicitly in the MiND-DAR formulation.

$$\min \quad \sum_{\ell \in \mathcal{L}} \sum_{t \in \mathcal{T}_\ell} h_\ell x_{\ell t} + \sum_{s \in \mathcal{S}} \pi_s \left(\sum_{\ell \in \mathcal{L}} \sum_{t \in \mathcal{T}_\ell} \sum_{a \in \mathcal{A}_{\ell st}} g_a^{DAR} y_a \right) \tag{EC.73}$$

$$\begin{aligned}
\text{s.t.} \quad & \text{First-stage constraints: Equations (1)–(4)} && \text{(EC.74)}\\
& \text{Second-stage constraints: Equations (9)–(10)} && \text{(EC.75)}\\
& \sum_{a\in\mathcal{A}_{\ell st}\,:\,p\in\mathcal{P}^{+}_{r(a)}} y_a - \sum_{a\in\mathcal{A}_{\ell st}\,:\,p\in\mathcal{P}^{-}_{r(a)}} y_a = 0 \quad \forall s\in\mathcal{S}, p\in\mathcal{P}, (\ell,t)\in\mathcal{M}_p && \text{(EC.76)}\\
& \boldsymbol{x},\boldsymbol{y},\boldsymbol{z} \text{ binary} && \text{(EC.77)}
\end{aligned}$$

EC.4.2. Algorithmic extension

Benders decomposition. For a reference trip (ℓ,t) and a scenario s, let $\zeta_{\ell stp}$ denote the dual variable associated to the new consistency constraint between pickup and drop-off decisions (Equation (EC.76)). The Benders dual subproblem becomes:

$$\begin{aligned}
\max \quad & x_{\ell t}\cdot(\psi_{\ell,s,t,u_{\ell st}} - \psi_{\ell,s,t,v_{\ell st}}) - \sum_{p\in\mathcal{P}\,:\,(\ell,t)\in\mathcal{M}_p} z_{p\ell st}\cdot\gamma_{\ell stp} && && \text{(EC.78)}\\
\text{s.t.} \quad & \psi_{\ell,s,t,n} - \psi_{\ell,s,t,m} - \sum_{p\in\mathcal{P}^{+}_{r(a)}} (\gamma_{\ell stp} - \zeta_{\ell stp}) - \sum_{p\in\mathcal{P}^{-}_{r(a)}} \zeta_{\ell stp} \le g_a^{DAR} && \forall a=(n,m)\in\mathcal{A}_{\ell st} && \text{(EC.79)}\\
& \psi_{\ell,s,t,i}\in\mathbb{R} && \forall i\in\mathcal{V}_{\ell st} && \text{(EC.80)}\\
& \gamma_{\ell stp}\ge 0 && \forall p\in\mathcal{P}\,:\,(\ell,t)\in\mathcal{M}_p && \text{(EC.81)}\\
& \zeta_{\ell stp}\in\mathbb{R} && \forall p\in\mathcal{P}\,:\,(\ell,t)\in\mathcal{M}_p && \text{(EC.82)}
\end{aligned}$$

Note that the new dual variables do not appear in the dual objective function, so the Benders optimality cut remains unchanged.

Column generation. The restricted Benders subproblem is still obtained by restricting the decisions to a subset of arc-based variables in $\mathcal{A}'_{\ell st}$:

$$\begin{aligned}
\text{RBSP}(\mathcal{A}'_{\ell st},\boldsymbol{x},\boldsymbol{z}) \quad \min_{\boldsymbol{y}\ge\boldsymbol{0}} \quad & \sum_{a\in\mathcal{A}'_{\ell st}} g_a^{DAR} y_a && \text{(EC.83)}\\
\text{s.t.} \quad & \sum_{m:(n,m)\in\mathcal{A}'_{\ell st}} y_{(n,m)} - \sum_{m:(m,n)\in\mathcal{A}'_{\ell st}} y_{(m,n)} = \begin{cases} x_{\ell t} & \text{if } n=u_{\ell st},\\ -x_{\ell t} & \text{if } n=v_{\ell st},\\ 0 & \text{otherwise,}\end{cases} \quad \forall n\in\mathcal{V}_{\ell st} && \text{(EC.84)}\\
& \sum_{a\in\mathcal{A}'_{\ell st}\,:\,p\in\mathcal{P}^{+}_{r(a)}} y_a - \sum_{a\in\mathcal{A}'_{\ell st}\,:\,p\in\mathcal{P}^{-}_{r(a)}} y_a = 0 \quad \forall p\in\mathcal{P}, (\ell,t)\in\mathcal{M}_p && \text{(EC.85)}\\
& \sum_{a\in\mathcal{A}'_{\ell st}\,:\,p\in\mathcal{P}_{r(a)}} y_a \le z_{p\ell st} \quad \forall p\in\mathcal{P}:(\ell,t)\in\mathcal{M}_p && \text{(EC.86)}
\end{aligned}$$

In the pricing problem, we split the level-of-service parameter d_{mp} into d^{+}_{mp} and d^{-}_{mp}, corresponding to the level-of-service components associated with pickups and dropoffs, respectively. Following Section EC.4.1, we denote by $\mathcal{P}^{+}_{m}$ (resp. $\mathcal{P}^{-}_{m}$) the set of passengers that can be picked up (resp.

dropped off) and by T^+_{mp} (resp. T^-_{mp}) the pickup time (resp. dropoff time) of passenger $p \in \mathcal{P}^+_m$ (resp. $p \in \mathcal{P}^-_m$). We then define:

$$d^+_{mp} = D_{ps}\left(\lambda\tau^{\text{walk}}_{mp} + \mu\tau^{\text{wait}}_{mp} - \sigma\frac{T^+_{mp}}{\tau^{\text{dir}}_p} - M\right) + \gamma_{\ell stp} - \zeta_{\ell stp}, \qquad \forall m \in \mathcal{U}^{uv}_{\ell st}, p \in \mathcal{P}^+_m$$

$$d^-_{mp} = D_{ps}\left(\frac{\delta\tau^{\text{late}}_{mp} + \frac{\delta}{2}\tau^{\text{early}}_{mp} + \sigma T^-_{mp}}{\tau^{\text{dir}}_p} + \lambda\tau^{\text{walk}}_{mp}\right) + \zeta_{\ell stp}, \qquad \forall m \in \mathcal{U}^{uv}_{\ell st}, p \in \mathcal{P}^-_m$$

Similarly, we define the following decision variables to split pickups and dropoffs:

$$f_{mq} = \begin{cases} 1 & \text{if arc } (m,q) \in \mathcal{H}^{uv}_{\ell st} \text{ is traversed in the time-expanded road segment network,} \\ 0 & \text{otherwise.} \end{cases}$$

$$w^+_{mp} = \begin{cases} 1 & \text{if passenger } p \in \mathcal{P}^+_m \text{ is picked up at node } m \in \mathcal{U}^{uv}_{\ell st}, \\ 0 & \text{otherwise.} \end{cases}$$

$$w^-_{mp} = \begin{cases} 1 & \text{if passenger } p \in \mathcal{P}^-_m \text{ is dropped off at node } m \in \mathcal{U}^{uv}_{\ell st}, \\ 0 & \text{otherwise.} \end{cases}$$

$$\xi_m = \text{ vehicle load in node } m \in \mathcal{U}^{uv}_{\ell st}$$

The pricing problem is then formulated as follows. Equation (EC.87) minimizes the reduced cost. Constraints (EC.88)–(EC.90) define the load at each node based on the pickups and dropoffs. Constraints (EC.91) and (EC.92) ensure that a passenger can only be picked up or dropped off in a node that is visited. Constraints (EC.93) and (EC.94) guarantee that a passenger is picked up and dropped off at most once, respectively. Constraints (28) apply flow balance in the time-expanded road segment network. The remaining constraints enforce binary requirements.

$$\min \sum_{m \in \mathcal{U}^{uv}_{\ell st}} \left(\sum_{p \in \mathcal{P}^+_m} d^+_{mp} w^+_{mp} + \sum_{p \in \mathcal{P}^-_m} d^-_{mp} w^-_{mp}\right) + \psi_{\ell,s,t,end(a)} - \psi_{\ell,s,t,start(a)} \tag{EC.87}$$

$$\text{s.t.} \quad \xi_{(u,T_{\ell t}(u))} = c_{(u,c_1)}, \ \xi_{(v,T_{\ell t}(v))} = c_{(v,c_2)} \tag{EC.88}$$

$$\xi_q - \xi_m \le \left(\sum_{p \in \mathcal{P}^+_m} D_{ps} w^+_{mp} - \sum_{p \in \mathcal{P}^-_m} D_{ps} w^-_{mp}\right) + C_\ell(1 - f_{mq}), \quad \forall (m,q) \in \mathcal{H}^{uv}_{\ell st} \tag{EC.89}$$

$$\xi_q - \xi_m \ge \left(\sum_{p \in \mathcal{P}^+_m} D_{ps} w^+_{mp} - \sum_{p \in \mathcal{P}^-_m} D_{ps} w^-_{mp}\right) - C_\ell(1 - f_{mq}), \quad \forall (m,q) \in \mathcal{H}^{uv}_{\ell st} \tag{EC.90}$$

$$w^+_{mp} \le \sum_{q:(m,q) \in \mathcal{H}^{uv}_{\ell st}} f_{mq} \quad \forall m \in \mathcal{U}^{uv}_{\ell st}, \ \forall p \in \mathcal{P}^+_m \tag{EC.91}$$

$$w^-_{mp} \le \sum_{q:(m,q) \in \mathcal{H}^{uv}_{\ell st}} f_{mq} \quad \forall m \in \mathcal{U}^{uv}_{\ell st}, \ \forall p \in \mathcal{P}^-_m \tag{EC.92}$$

$$\sum_{m \in \mathcal{U}^{uv}_{\ell st} : p \in \mathcal{P}^+_m} w^+_{mp} \le 1 \quad \forall p \in \mathcal{P} \tag{EC.93}$$

$$\sum_{m \in \mathcal{U}^{uv}_{\ell st} : p \in \mathcal{P}^-_m} w^-_{mp} \le 1 \quad \forall p \in \mathcal{P} \tag{EC.94}$$

$$\sum_{q:(m,q)\in\mathcal{H}^{uv}_{\ell st}} f_{mq} - \sum_{q:(q,m)\in\mathcal{H}^{uv}_{\ell st}} f_{qm} = \begin{cases} 1 & \text{if } m=(u,T_{\ell t}(u)),\\ -1 & \text{if } m=(v,T_{\ell t}(v)),\\ 0 & \text{otherwise.}\end{cases} \quad \forall m\in\mathcal{U}^{uv}_{\ell st} \tag{EC.95}$$

$$f_{mq}\in\{0,1\} \quad \forall (m,q)\in\mathcal{H}^{uv}_{\ell st} \tag{EC.96}$$

$$w^+_{mp}\in\{0,1\} \quad \forall m\in\mathcal{U}^{uv}_{\ell st}, p\in\mathcal{P}^+_m \tag{EC.97}$$

$$w^-_{mp}\in\{0,1\} \quad \forall m\in\mathcal{U}^{uv}_{\ell st}, p\in\mathcal{P}^-_m \tag{EC.98}$$

Label setting algorithm. To distinguish pickups and dropoffs, we extend the label-setting algorithm from a two-dimensional to a three-dimensional state space. Dropoffs are treated the same way as pickups; for instance, the state transition includes checking all passenger combinations for pickups and all passenger combinations for dropoffs. This extension has two major implications that increase the computational requirements in the pricing problem. First, the dominance rule requires the dominating state to have the same set of pickups *and* the same set of dropoffs as the dominated state. Second, the set of load differential needs to be extended from $\{0,1,\cdots,C_\ell\}$ to $\{-C_\ell,\cdots,-1,0,1,\cdots,C_\ell\}$. Nonetheless, our results show that our methodology scales to meaningful practical instances of the MiND-DAR model in Manhattan, with up to 10 candidate lines, hundreds of stations, thousands of passenger requests and 5 demand scenarios—resulting in over 60,000 first-stage variables and 700 second-stage problems.

EC.4.3. Experimental results

We construct a case study setting in Midtown Manhattan, with 10 candidate lines traveling West to East from the 11th to the 1st avenue along every other street between 36th and 54th. Each line contains a checkpoint at every other avenue, and each street-avenue intersection defines a station—leading to a total of 168 stations. We calibrate demand inputs by collecting all West-to-East requests in Midtown Manhattan during the morning rush from 6 to 9 am, amounting to over 3,000 passenger requests. We set up one-hour, two-hour and three-hour instances (from 6 to 7 am, 6 to 8 am, and 6 to 9 am, respectively). For each one, we run the deviated fixed-route microtransit as well as the fixed-line transit benchmark and ride-sharing benchmarks with single-occupancy, two-occupancy and four-occupancy vehicles (see EC.3.3). We consider five demand scenarios. Again, for apples-to-apples comparison, we group results by total seating capacity (e.g., 10 transit/microtransit vehicles of capacity 10, ride-sharing with 100/50/25 vehicles of capacity of 1/2/4), and perform an out-of-sample assessment corresponding to five new weekdays.

We evaluate the system-wide performance of all optimized transportation modes in Table EC.5, broken down into level of service (demand coverage and average walking time, waiting time, delay and detour), vehicle utilization (passengers served divided by vehicle capacity), and distance

Table EC.5 Average performance of fixed-route transit, microtransit, and ride-sharing in a dial-a-ride setting.

Setting			Average level of service					Vehicle utilization		Distance traveled (km)		
Horizon	Capacity	Mode	Coverage	Walk	Wait	Delay	Detour	Absolute	Relative	Internal	External	Total
1 hour	50	Transit	6.7%	1.68	3.57	1.45	200%	3.04	40.5%	60	881	941
		Microtransit	16.5%	1.92	3.71	1.70	144%	6.30	81.4%	94	798	893
		RS Cap. 4	23.8%	0.00	3.63	6.74	183%	3.81	95.3%	293	1,082	1,375
		RS Cap. 2	36.6%	0.00	3.56	4.30	120%	1.95	97.3%	567	942	1,509
		RS Cap. 1	53.3%	0.00	2.38	2.38	100%	1.00	100.0%	1,040	754	1,793
	100	Transit	7.6%	1.68	3.51	1.64	201%	3.18	22.7%	63	881	944
		Microtransit	22.8%	2.04	3.73	1.68	154%	8.29	57.4%	96	798	894
		RS Cap. 4	42.4%	0.00	3.48	6.92	190%	3.81	95.3%	541	833	1,374
		RS Cap. 2	63.3%	0.00	3.71	4.57	122%	1.94	97.2%	1,057	569	1,626
		RS Cap. 1	85.2%	0.00	2.47	2.47	100%	1.00	100.0%	1,876	268	2,144
	200	Transit	8.7%	1.64	3.54	2.07	199%	2.74	13.7%	84	906	990
		Microtransit	26.8%	2.05	3.80	1.65	156%	7.21	36.1%	127	755	882
		RS Cap. 4	70.7%	0.00	3.55	7.24	193%	3.86	96.6%	965	421	1,386
		RS Cap. 2	94.9%	0.00	4.01	5.05	125%	1.95	97.3%	1,708	76	1,784
		RS Cap. 1	100.0%	0.00	2.66	2.66	100%	1.00	100.0%	2,131	0	2,131
2 hours	50	Transit	6.9%	1.87	3.76	1.05	195%	3.13	52.1%	153	2,403	2,557
		Microtransit	15.2%	1.83	3.72	1.69	162%	6.62	101.5%	220	2,196	2,416
		RS Cap. 4	19.2%	0.00	3.88	6.77	182%	3.82	95.4%	592	3,039	3,631
		RS Cap. 2	29.7%	0.00	3.69	4.38	120%	1.96	98.1%	1,154	2,770	3,924
		RS Cap. 1	43.9%	0.00	2.46	2.46	100%	1.00	100.0%	2,184	2,337	4,521
	100	Transit	5.9%	0.94	3.77	1.32	196%	3.63	22.2%	120	2,428	2,548
		Microtransit	19.9%	2.12	3.95	1.65	155%	11.05	66.8%	175	2,071	2,246
		RS Cap. 4	33.9%	0.00	3.84	7.11	189%	3.84	96.0%	1,113	2,535	3,649
		RS Cap. 2	53.0%	0.00	3.78	4.61	123%	1.96	98.2%	2,252	1,948	4,200
		RS Cap. 1	74.3%	0.00	2.53	2.53	100%	1.00	100.0%	4,207	1,214	5,421
	200	Transit	8.5%	0.91	3.70	1.67	198%	3.23	16.2%	187	2,331	2,517
		Microtransit	27.3%	2.09	3.88	1.65	152%	9.42	47.1%	258	1,848	2,106
		RS Cap. 4	59.1%	0.00	3.80	7.41	194%	3.85	96.3%	2,114	1,611	3,724
		RS Cap. 2	86.1%	0.00	4.03	5.07	126%	1.96	97.9%	4,117	635	4,752
		RS Cap. 1	99.5%	0.00	2.76	2.76	100%	1.00	100.0%	6,253	30	6,283
3 hours	50	Transit	6.7%	1.88	3.77	1.13	198%	4.53	57.0%	192	3,895	4,087
		Microtransit	13.5%	1.74	3.69	1.74	154%	8.25	104.0%	268	3,600	3,868
		RS Cap. 4	16.8%	0.00	3.91	6.78	182%	3.80	95.0%	887	5,444	6,331
		RS Cap. 2	26.9%	0.00	3.80	4.47	120%	1.97	98.3%	1,750	4,996	6,746
		RS Cap. 1	39.7%	0.00	2.50	2.50	100%	1.00	100.0%	3,348	4,314	7,662
	100	Transit	7.6%	1.96	3.69	0.85	201%	5.09	30.5%	174	3,866	4,040
		Microtransit	20.7%	1.99	3.83	1.74	132%	14.01	80.0%	229	3,298	3,527
		RS Cap. 4	30.0%	0.00	3.87	7.09	189%	3.81	95.2%	1,691	4,670	6,361
		RS Cap. 2	48.2%	0.00	3.89	4.69	122%	1.96	98.0%	3,451	3,753	7,204
		RS Cap. 1	68.4%	0.00	2.56	2.56	100%	1.00	100.0%	6,558	2,520	9,077
	200	Transit	8.4%	1.85	3.83	1.93	195%	3.66	18.3%	262	3,678	3,940
		Microtransit	26.1%	2.13	3.99	1.69	127%	10.37	51.8%	365	2,991	3,356
		RS Cap. 4	53.0%	0.00	3.82	7.41	195%	3.84	95.9%	3,264	3,235	6,499
		RS Cap. 2	80.3%	0.00	4.03	5.05	126%	1.96	97.9%	6,549	1,561	8,110
		RS Cap. 1	97.8%	0.00	2.78	2.78	100%	1.00	100.0%	10,751	226	10,977

Walk, wait, delay and detour are averaged across all passengers. Walk, wait, and delay are in minutes.

traveled (internal distance for served passengers plus external distance for unserved passengers). These results confirm and extend all takeaways from the MiND-VRP (Table 6 and Figure 8).

Note, first, the benefits of on-demand flexibility versus fixed-line transit: by leveraging on-demand deviations, microtransit enables significant increases in demand coverage. Specifically, microtransit serves 2 to 3 times more passengers; in the three-hour case for example, this increase translates into an improvement in vehicle utilization from 30% to 80% on average with medium system capacity and from 18% to 52% with high system capacity. Unlike in the MiND-VRP, higher demand coverage comes with a slight increase in passenger walking and waiting, primarily due to an adverse selection effect—by serving passengers with pickup or drop-off locations further away from the reference lines, for example. Nonetheless, level of service remains comparable to fixed-line transit, with walking and waiting times around 2–3 minutes on average.

Next, results underscore the impact of demand consolidation: by relying on higher-capacity vehicles along reference lines, microtransit serves fewer passengers but travels much shorter distances than ride-sharing systems. As expected, ride-sharing results in higher demand coverage with no walking and short wait times. On the other hand, ride-sharing induces longer delays because of on-demand dispatches. Four-occupancy ride-pooling can also result in higher detours than microtransit, due to the negative externalities of door-to-door transportation—even with small-occupancy vehicles—and the comparative benefits of line regularization in microtransit. Moreover, the microtransit system travels much smaller (internal) distances by using higher-capacity vehicles.

At the aggregate level, microtransit induces strong system-wide improvements against all benchmarks. As compared to fixed-line transit, on-demand deviations increase distance traveled but this effect is more than compensated by the increase in demand coverage—leading to a decrease in distance per passenger by a factor of 1.4 to 2.3. As compared to ride-sharing, microtransit decreases distance traveled by a much higher factor than the corresponding loss in demand coverage, leading to a smaller distance per passenger by a factor of 4–11 (resp. 3–6) as compared to single-occupancy ride-sharing (resp. four-occupancy ride-pooling). When accounting for the "external" distance from single-occupancy trips for all unserved passengers (assuming for instance that all unserved passengers take a taxi to their destination), microtransit reduces total distance from fixed-line transit by 5%, 13% and 15% in the three-hour case with small, medium and high system capacity, respectively; it reduces total distance from four-occupancy ride-pooling by 39%, 45% and 48%; and it reduces total distance from single-occupancy ride-sharing by 98%, 157% and 227%.

These results confirm the potential of deviated fixed-route microtransit to improve demand coverage as compared to fixed-line transit—thanks to demand-responsive operations—and to improve demand consolidation as compared to ride-sharing—thanks to high-occupancy vehicles. These combined effects can induce strong reductions in distance traveled per passenger, which can ultimately contribute to creating more effective and more affordable mobility options and to mitigating the environmental footprint of urban mobility.

EC.5. Extension to incorporate transfers (MiND-Tr)

EC.5.1. Modeling extension

In the MiND-VRP, all passengers share the same destination at the end of each reference line (or the same origin at the start). Transfers are of little use in this setting, since all lines can drop off all passengers at their destination. We define the model with transfers, referred to as MiND-Tr, in a routing setting with a set $\mathcal{D}$ of destinations. We assume that each line passes through one transfer point; that all passenger are picked up ahead of the transfer stations; and that their destinations are after the transfer stations. In other words, all passengers request transportation from an "origin zone" to a "destination zone"; first-leg trips can pick up passengers near their origin and drop them off at a transfer station; and second-leg trips will pick them up at the transfer station and drop them off at their destination. In our experiments, we consider a similar use case as in the MiND-VRP, except that passengers' destinations are split between two airports (e.g., JFK and LGA). This setting preserves the key structural components of the second-stage model.

We define a set $\mathcal{F}$ of transfer stations. Each reference trip (ℓ, t) transits through transfer station $f_\ell \in \mathcal{F}$ at time $T^{\text{tr}}_{\ell t}$, and ends in destination $d_\ell \in \mathcal{D}$. Each passenger $p \in \mathcal{P}$ is bound for destination $d(p) \in \mathcal{D}$. Passengers can be picked up on a line ℓ bound for destination $d(p)$ $(d_\ell = d(p))$; alternatively, they can be picked up by line ℓ and then transfer at transfer point $f_\ell \in \mathcal{F}$ to another line ℓ' bound for destination $d(p)$ $(d_{\ell'} = d(p))$. We impose a maximum transfer time W^{tr}. We ensure that $f_\ell \in \mathcal{I}_\ell$, i.e. the transfer station is included in the list of checkpoints; even when we allow to skip checkpoints $(K > 0)$, the vehicle must visit the transfer station to allow transfers.

Recall that, in the core MiND-VRP, $\mathcal{M}_p \subseteq \mathcal{L} \times \mathcal{T}_\ell$ denotes the subset of reference trips that can serve request $p \in \mathcal{P}$ within a tolerance α of their requested drop-off time. We extend this definition into a set of first-leg trips $\mathcal{M}^1_p$ (from the passenger's pickup location to a transfer point) and second-leg trips $\mathcal{M}^2_p$ (from a transfer point to the destination). Let $\mathcal{M}^2_p(\ell, t)$ denote the set of second-leg trips (ℓ', t') that can serve passenger $p \in \mathcal{P}$ within a tolerance α of their requested drop-off time, i.e., $\left| T_{\ell t}(\mathcal{I}^{(I_\ell)}_\ell) - t^{\text{req}}_p \right| \leq \alpha$; and that are compatible with first-leg trip (ℓ, t), i.e., (i) reference lines ℓ and ℓ' transit through the same transfer station, i.e., $f_\ell = f_{\ell'}$; and (ii) reference trip (ℓ, t) arrives to the transfer point before reference trip (ℓ', t') departs and within the maximum transfer time, i.e., $T^{\text{tr}}_{\ell t} \leq T^{\text{tr}}_{\ell' t'} \leq T^{\text{tr}}_{\ell t} + W^{\text{tr}}$. If a reference trip (ℓ, t) can pick up and drop off passenger p without a transfer (i.e., if $d_\ell = d(p)$), then the reference trip is thus included in both $\mathcal{M}^1_p$ and $\mathcal{M}^2_p(\ell, t)$.

The first-stage problem still selects reference trips (via binary variables $x_{\ell t}$) and assigns passengers to reference trips. The latter decisions are disaggregated into first-leg (pickup) and second-leg (dropoff) assignments, with binary variables $z^1_{\ell pst}$ and $z^2_{\ell pst}$ for passenger $p \in \mathcal{P}$ and scenario $s \in \mathcal{S}$. Each passenger will be assigned to first-leg and second-leg components (pickup and dropoff).

In the second stage, we define subpaths between checkpoints for the first leg of each line. The second leg travels directly between the line's transfer station and its destination, since all passengers on the line are then bound for the same destination; in particular, each line does not need to stop or deviate between the transfer station and the destination. Thus, we only define a set of pick-up subpaths $\mathcal{R}^1_{\ell st}$ for reference trip (ℓ, t) in scenario $s \in \mathcal{S}$, between checkpoints. We denote by $\mathcal{P}^1_{r(a)} \subseteq \mathcal{P}$ the set of passengers picked up by subpath $r(a) \in \mathcal{R}^1_{\ell st}$. The parameters g^1_a and $g^2_{\ell pst}$ capture coverage, walk, wait, and in-vehicle time in the first leg, and lateness/earliness in the second leg.

$$g^1_a = \begin{cases} \displaystyle\sum_{p \in \mathcal{P}^1_{r(a)}} D_{ps} \left(\lambda \tau^{walk}_{r(a)p} + \mu \tau^{wait}_{r(a)p} + \sigma \frac{\tau^{travel}_{r(a)p}}{\tau^{dir}_p} - M \right) & \forall a \in \displaystyle\bigcup_{r \in \mathcal{R}_{\ell st}} \mathcal{A}_r, \\ 0 & \forall a \in \mathcal{A}^v_{\ell st}. \end{cases} \tag{EC.99}$$

$$g^2_{\ell pst} = D_{ps} \left(\delta \frac{\tau^{late}_{\ell tp}}{\tau^{dir}_p} + \frac{\delta}{2} \frac{\tau^{early}_{\ell tp}}{\tau^{dir}_p} \right) \qquad \forall s \in \mathcal{S}, p \in \mathcal{P}, (\ell, t) \in \mathcal{M}^2_p \tag{EC.100}$$

The MiND-Tr is formulated as follows. We redefine binary variable y^1_a to indicate whether a microtransit vehicle traverses subpath-based arc $a \in \mathcal{A}_{\ell st}$ in the first-leg trip; and we introduce a new binary variable $y^2_{\ell pst}$ to assign passenger p in scenario s to reference trip (ℓ, t) from the transfer station to the destination (this is a simpler representation than subpath-based arcs, exploiting the fact that the vehicle does not deviate after the transfer point). Equation (EC.101) minimizes expected costs. Constraints (EC.102) allow each passenger to be picked up at most once, and constraints (EC.103) ensure consistency between first- and second-leg trips. Constraints (EC.104)-(EC.105) enforce the target load factor on the first-and second-leg trips. Constraints (EC.106) and (EC.107) ensure consistent assignments between the first stage and the second stage. Constraints (EC.108) state that picked up passengers are dropped off, either by the same trip or by a compatible trip via a transfer. Finally, Constraints (EC.109) enforce post-transfer capacity constraints for the trips.

$$\min \quad \sum_{\ell \in \mathcal{L}} \sum_{t \in \mathcal{T}_\ell} \left(h_\ell x_{\ell t} + \sum_{s \in \mathcal{S}} \pi_s \sum_{a \in \mathcal{A}_{\ell st}} g^1_a y^1_a + \sum_{s \in \mathcal{S}} \pi_s \sum_{p \in \mathcal{P} : (\ell, t) \in \mathcal{M}^2_p} g^2_{\ell pst} y^2_{\ell pst} \right) \tag{EC.101}$$

s.t. Fleet budget constraints (1)

$$\sum_{(\ell, t) \in \mathcal{M}^1_p} z^1_{\ell pst} \leq 1, \qquad \forall p \in \mathcal{P}, \forall s \in \mathcal{S} \tag{EC.102}$$

$$z^1_{\ell_1 pst_1} = \sum_{(\ell_2, t_2) \in \mathcal{M}^2_p(\ell_1 t_1)} z^2_{\ell_2 pst_2}, \qquad \forall p \in \mathcal{P}, (\ell_1, t_1) \in \mathcal{M}^1_p, \forall s \in \mathcal{S} \tag{EC.103}$$

$$(1 - \kappa) C_\ell x_{\ell t} \leq \sum_{p \in \mathcal{P} \,:\, (\ell, t) \in \mathcal{M}^1_p} D_{ps} z^1_{\ell pst} \leq (1 + \kappa) C_\ell x_{\ell t} \quad \forall (\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell, \forall s \in \mathcal{S} \tag{EC.104}$$

$$(1 - \kappa) C_\ell x_{\ell t} \leq \sum_{p \in \mathcal{P} \,:\, (\ell, t) \in \mathcal{M}^2_p} D_{ps} z^2_{\ell pst} \leq (1 + \kappa) C_\ell x_{\ell t} \quad \forall (\ell, t) \in \mathcal{L} \times \mathcal{T}_\ell, \forall s \in \mathcal{S} \tag{EC.105}$$

Subpath flow balance constraints (9)

$$\sum_{a \in \mathcal{A}_{\ell st} \,:\, p \in \mathcal{P}^1_{r(a)}} y^1_a \leq z^1_{\ell pst} \quad \forall s \in \mathcal{S}, p \in \mathcal{P}, (\ell, t) \in \mathcal{M}^1_p \tag{EC.106}$$

$$y^2_{\ell pst} \leq z^2_{\ell pst} \quad \forall s \in \mathcal{S}, p \in \mathcal{P}, (\ell, t) \in \mathcal{M}^2_p \tag{EC.107}$$

$$\sum_{a\in\mathcal{A}_{\ell_1 st_1}:p\in\mathcal{P}^1_{r(a)}} y^1_a = \sum_{(\ell_2,t_2)\in\mathcal{M}^2_p(\ell_1,t_1)} y^2_{\ell_2 pst_2} \quad \forall s\in\mathcal{S}, p\in\mathcal{P}, (\ell_1,t_1)\in\mathcal{M}^1_p, \tag{EC.108}$$

$$\sum_{p\in\mathcal{P}:(\ell,t)\in\mathcal{M}^2_p} y^2_{\ell pst} \le C_\ell x_{\ell t} \quad \forall s\in\mathcal{S}, \ell\in\mathcal{L}, t\in\mathcal{T}_\ell \tag{EC.109}$$

$$\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}^1, \boldsymbol{z}^2, \text{ binary} \tag{EC.110}$$

EC.5.2. Algorithmic extension

Benders decomposition. Due to Equation (EC.108), the Benders subproblem is no longer separable by reference trip $(\ell,t)\in\mathcal{L}\times\mathcal{T}_\ell$; indeed, operations on several lines need to be coordinated so that, after transfers, the load of the vehicles will not exceed their capacity.

Let $\boldsymbol{\psi}$ denote the dual variable associated to the flow balance constraints, $\boldsymbol{\gamma}^1$ the one associated to the pickup trip linking constraints (Equation (EC.106)), $\boldsymbol{\gamma}^2$ the one associated to the dropoff trip linking constraints (Equation (EC.107)), $\boldsymbol{\eta}$ the one associated to the transfer consistency constraints (Equation (EC.108)), and $\boldsymbol{\nu}$ the dual variable associated to the final leg capacity constraints (Equation (EC.109)). The Benders dual subproblem for scenario s becomes:

$$\max \quad \sum_{\ell\in\mathcal{L}}\sum_{t\in\mathcal{T}_\ell}\left(x_{\ell t}\cdot(\psi_{\ell stu_{\ell st}} - \psi_{\ell stv_{\ell st}}) - C_\ell \nu_{\ell st}\right) \tag{EC.111}$$

$$-\sum_{p\in\mathcal{P}}\left(\sum_{(\ell,t)\in\mathcal{M}^1_p} z^1_{\ell pst}\cdot\gamma^1_{\ell stp} + \sum_{(\ell,t)\in\mathcal{M}^2_p} z^2_{\ell pst}\cdot\gamma^2_{\ell stp}\right) \tag{EC.112}$$

$$\text{s.t.} \quad \psi_{\ell stn} - \psi_{\ell stm} - \sum_{p\in\mathcal{P}^1_{r(a)}} (\gamma^1_{\ell stp} - \eta_{\ell stp}) \le g^1_a \quad \forall \ell\in\mathcal{L}, t\in\mathcal{T}_\ell, a=(n,m)\in\mathcal{A}_{\ell st} \tag{EC.113}$$

$$-\gamma^2_{\ell stp} - \sum_{(\ell_2,t_2)\in\mathcal{M}^2_p(\ell,t)} \eta_{\ell_2 st_2 p} - \nu_{\ell st} \le g^2_{\ell pst} \quad \forall p\in\mathcal{P}, (\ell,t)\in\mathcal{M}^2_p \tag{EC.114}$$

$$\boldsymbol{\gamma}^1, \boldsymbol{\gamma}^2, \boldsymbol{\eta}, \boldsymbol{\nu} \ge \mathbf{0} \tag{EC.115}$$

The corresponding Benders optimality cut becomes:

$$\theta_s \ge \sum_{\ell\in\mathcal{L}}\sum_{t\in\mathcal{T}_\ell}\left(x_{\ell t}\cdot(\psi_{\ell stu_{\ell st}} - \psi_{\ell stv_{\ell st}}) - C_\ell \nu_{\ell st}\right) - \sum_{p\in\mathcal{P}}\left(\sum_{(\ell,t)\in\mathcal{M}^1_p} z^1_{\ell pst}\cdot\gamma^1_{\ell stp} + \sum_{(\ell,t)\in\mathcal{M}^2_p} z^2_{\ell pst}\cdot\gamma^2_{\ell stp}\right) \tag{EC.116}$$

Column generation. The restricted Benders subproblem for each scenario s is still obtained by restricting the decisions to a subset of arc-based variables in $\mathcal{A}'_{\ell st}$ for each reference trip (ℓ,t):

$$\min_{\boldsymbol{y}^1,\boldsymbol{y}^2\ge\mathbf{0}} \quad \sum_{a\in\mathcal{A}'_{\ell st}} g^1_a y^1_a + \sum_{p\in\mathcal{P}:(\ell,t)\in\mathcal{M}^2_p} g^2_{\ell pst} y^2_{\ell pst} \tag{EC.117}$$

$$\text{s.t.} \quad \sum_{m:(n,m)\in\mathcal{A}'_{\ell st}} y^1_{(n,m)} - \sum_{m:(m,n)\in\mathcal{A}'_{\ell st}} y^1_{(m,n)} = \begin{cases} x_{\ell t} & \text{if } n = u_{\ell st}, \\ -x_{\ell t} & \text{if } n = v_{\ell st}, \\ 0 & \text{otherwise}, \end{cases} \quad \forall \ell \in \mathcal{L}, t \in \mathcal{T}_\ell, n \in \mathcal{V}_{\ell st} \tag{EC.118}$$

$$\sum_{a\in\mathcal{A}'_{\ell st}:p\in\mathcal{P}^1_{r(a)}} y^1_a \le z^1_{\ell pst} \quad \forall p \in \mathcal{P}, (\ell,t) \in \mathcal{M}^1_p \tag{EC.119}$$

$$y^2_{\ell pst} \le z^2_{\ell pst} \quad \forall p \in \mathcal{P}, (\ell,t) \in \mathcal{M}^2_p \tag{EC.120}$$

$$\sum_{a\in\mathcal{A}'_{\ell_1 st_1}:p\in\mathcal{P}^1_{r(a)}} y^1_a = \sum_{(\ell_2,t_2)\in\mathcal{M}^2_p(\ell_1,t_1)} y^2_{\ell_2 pst_2} \quad \forall p \in \mathcal{P}, (\ell_1,t_1) \in \mathcal{M}^1_p, \tag{EC.121}$$

$$\sum_{p\in\mathcal{P}:(\ell,t)\in\mathcal{M}^2_p} y^2_{\ell pst} \le C_\ell x_{\ell t} \quad \forall \ell \in \mathcal{L}, t \in \mathcal{T}_\ell \tag{EC.122}$$

The pricing problem seeks a subpath-based arc of minimum reduced cost:

$$\min_{\ell\in\mathcal{L},t\in\mathcal{T}_\ell} \min_{a=(n,m)\in\mathcal{A}_{\ell st}} \left(g^1_a + \psi_{\ell stm} - \psi_{\ell stn} + \sum_{p\in\mathcal{P}^1_{r(a)}} (\gamma^1_{\ell stp} - \eta_{\ell stp}) \right)$$

In the pricing problem, we define the level-of-service parameter d^1_{mp} for passenger p in location m only over the level-of-service components associated with the first-leg pickups, as the second-leg dropoff components are contained in $g^2_{\ell pst}$. We denote by $\mathcal{P}^1_m$ the set of passengers that can be picked up in node $m \in \mathcal{U}^{uv}_{\ell st}$, and by T^1_{mp} the pickup time of passenger $p \in \mathcal{P}^1_m$. We then define:

$$d^1_{mp} = D_{ps}\left(\lambda\tau^{\text{walk}}_{mp} + \mu\tau^{\text{wait}}_{mp} - \sigma\frac{T^1_{mp}}{\tau^{\text{dir}}_p} - M\right) + \gamma^1_{\ell stp} - \eta_{\ell stp}, \qquad \forall \ell \in \mathcal{L}, t \in \mathcal{T}_\ell, m \in \mathcal{U}^{uv}_{\ell st}, p \in \mathcal{P}^1_m$$

Because the transfers do not impact subpaths before the transfer point, the pricing problem is identical to the one in the MiND-VRP (Equations (22)–(29)) apart from dual information in the objective function. Recall the MiND-VRP pricing problem has a binary variable w_{mp} to indicate whether passenger p is picked up at location m. The objective for the MiND-Tr pricing problem for trip (ℓ,t) in scenario $s \in \mathcal{S}$ between checkpoints u and v is given as:

$$\min \sum_{m\in\mathcal{U}^{uv}_{\ell st}} \sum_{p\in\mathcal{P}_m} d^1_{mp} w_{mp} + \psi_{\ell,s,t,end(a)} - \psi_{\ell,s,t,start(a)} \tag{EC.123}$$

Label setting algorithm. The structure of the pricing problem remains unchanged, so the label-setting algorithm is identical to the one in the MiND-VRP.

EC.5.3. Experimental results

We adapt the MiND-VRP case study to include passenger requests to LaGuardia Airport (LGA) and John F. Kennedy International Airport (JFK) between 6 and 9 am. Passenger origins are located in Manhattan and lines travel to the airport via one of four bridges. We set up instances with 10, 20, and 40 candidate lines, half of which are bound for LGA and half of which are bound for JFK. We locate the transfer stations at the last stop before each bridge.

Figure EC.7 shows a sample solution for two candidate lines. The line bound for LGA (in blue) picks up four passengers, with three headed to LGA (blue circles) and one to JFK (orange circle). The line bound for JFK picks up eight passengers, five headed to LGA and three headed to JFK. The two lines meet at a transfer point before the Queens-Midtown Tunnel where passengers can transfer to the vehicle bound for their destination.

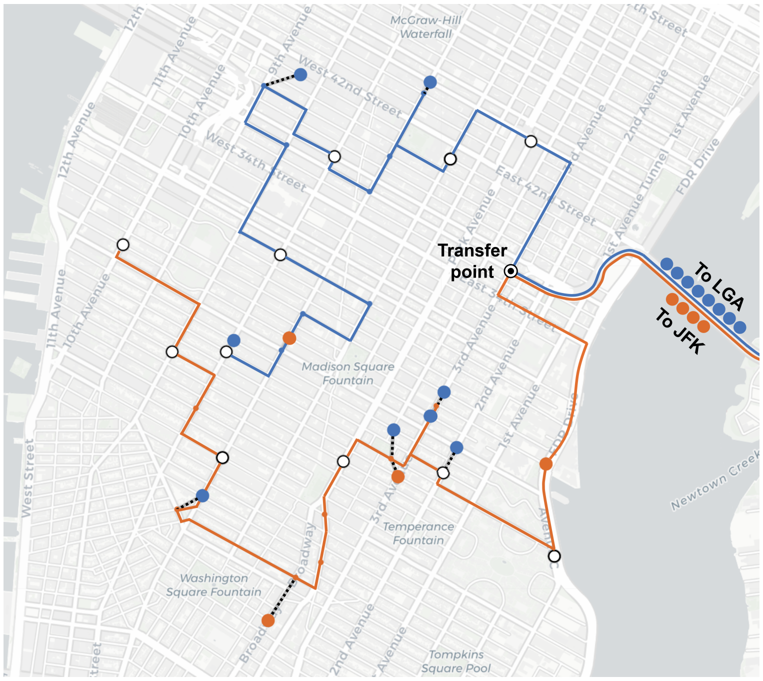


Figure EC.7 **A line bound for LaGuardia (blue) and a line bound for JFK (orange) with a transfer location just before the Queensboro bridge. Passengers are shown as circles, shaded to match the color of their destination airport, with their walking path shown as black dotted lines.**

Table EC.6 compares the deviated fixed-route microtransit solution and the fixed-line transit benchmark, each with and without transfers. As expected, transfers increase problem complexity. The MiND-Tr scales up to 10 candidate lines and 5 scenarios but leaves a larger optimality gap in larger instances. This mainly comes from the fact that the Benders subproblem is no longer separable by reference trip, resulting in longer computational requirements and denser Benders cuts. Nonetheless, transfers can improve solution performance even in larger instances that the MiND-Tr does not solve to optimality, by aggregating demand across destinations. Specifically, 31-45% of microtransit passengers transfer between lines. Transfers increase coverage by 0-3% for transit and 1-3% for microtransit. Accommodating these additional passengers comes at a minor cost in terms of wait times, detours, and delays, as well as a more significant 1-4 minute increase in walking times. Nonetheless, the net benefit of transfers is positive, resulting in 3–7% cost reductions.

Table EC.6 Comparison of fixed-route transit vs. microtransit with and without transfers

					Average level of service				Vehicle utilization		Distance traveled (km)		
$\lvert\mathcal{L}\rvert$	Mode	Sol.	Gap	Tr.	Coverage	Walk	Wait	Detour	Delay	Absolute	Relative	Internal	External	Total
10	Transit	163.5	**0.0**	—	10.8%	0	3.13	132%	0.04	1.48	38.9%	59	4,180	4,239
	Transit, transfers	163.5	**0.0**	16%	10.8%	0	3.08	137%	0.09	1.48	38.9%	56	4,180	4,236
	MT	104.4	**0.0**	—	26.5%	3.08	2.99	134%	0.03	3.64	95.8%	75	4,031	4,106
	MT, transfers	100	**0.0**	41%	27.7%	7.37	3.27	136%	0.1	3.8	100%	76	4,023	4,099
20	Transit	157.3	**0.0**	—	15.7%	0	2.18	127%	0	2.2	44.4%	122	4,985	5,107
	Transit, transfers	152.3	**0.0**	34%	17.4%	0	2.18	127%	0.01	2.44	49.2%	113	4,970	5,083
	MT	107.9	**0.0**	—	32.7%	1.13	2.25	125%	0.01	4.58	92.3%	153	4,832	4,985
	MT, transfers	100	14.5	31%	35.4%	3.77	2.32	127%	0.02	4.96	100%	154	4,805	4,959
40	Transit	147.1	**0.0**	—	24.4%	1.15	2.2	120%	0	2.43	54.4%	236	3,626	3,862
	Transit, transfers	140.2	3.1	44%	27.5%	1.54	2.2	124%	0.05	2.73	61.1%	242	3,552	3,794
	MT	103.1	1.7	—	43.7%	1.29	2.08	117%	-0.01	4.34	97.1%	296	3,120	3,416
	MT, transfers	100	27.9	45%	45%	2.5	2.16	120%	0.04	4.47	100%	301	3,093	3,394

"Sol.": total expected cost, normalized to the microtransit solution with transfers.

Walk, wait, delay and detour are averaged across all passengers. Walk, wait, and delay are in minutes.

Most importantly, these new results strengthen our main takeaways. Indeed, the microtransit system still provides improvements over the fixed-route transit benchmark in the presence of transfers, consistent with the MiND-VRP and MiND-DAR results. In fact, the performance of the microtransit system *without transfers* also outperforms the one of the transit benchmark *with transfers*. In other words, the benefits of transfers to the transit system do not cover the significant service gap between transit and microtransit. This can be explained, in part, by the spatiotemporal coordination required across reference trips to take advantage of transfers, which restricts the model's flexibility when selecting reference trips. In comparison, these results reinforce the benefits of the routing flexibility from the microtransit system: microtransit provides a 15-17% increase in coverage and a 44-59% decrease in total costs without transfers; and then, transfers enable further cost reductions of 3–7%. These benefits translate into a 3–10% reduction in the "total distance" metric in the microtransit solution as compared to transit, with or without transfer, leading to reductions in operating costs per passenger and the environmental footprint of mobility systems.

EC.6. Additional results

EC.6.1. Additional results on the benefits of the subpath-based model

Recall that our results in Section 5.1 showed the benefits of our subpath-based formulation in view of the overall two-stage stochastic integer optimization problem, as compared to the compact, segment-based and path-based benchmarks. In this appendix, we compare the four formulations for the second-stage problem alone—that is, on the capacitated vehicle routing problem with time windows. We consider instances with 5, 8, and 10 candidate lines, 5 scenarios, and 1-, 2-, and 3-hours horizons. We fix the optimal first-stage decisions, and then solve the corresponding subproblems using the direct implementation of each formulation. By design, the subpath-based and path-based models are solved via exhaustive enumeration. Table EC.7 reports pre-processing times for arc enumeration, solution times, and optimality gaps.

Table EC.7 Comparison of the four models for the second-stage problem.

		Compact			Segment			Path			Subpath		
$\lvert\mathcal{L}\rvert$	Hor.	Enum. CPU(s)	Solve CPU(s)	Gap	Enum. CPU (s)	Solve CPU(s)	Gap	Enum. CPU (s)	Solve CPU(s)	Gap	Enum. CPU (s)	Solve CPU(s)	Gap
5	60	—	>600	3.8	68	1.05	0.0	76	0.05	0.0	19	<0.01	0.0
	120	—	>600	5.7	283	1.03	0.0	539	0.05	0.0	252	<0.01	0.0
	180	—	>600	—	565	1.01	0.0	602	0.05	0.0	296	<0.01	0.0
8	60	—	>600	6.5	160	1.10	0.0	738	0.07	0.0	37	<0.01	0.0
	120	—	>600	—	616	1.04	0.0	2,735	0.11	0.0	381	<0.01	0.0
	180	—	>600	—	1,323	1.08	0.0	3,027	0.08	0.0	495	<0.01	0.0
10	60	—	>600	6.1	228	0.97	0.0	919	0.07	0.0	47	<0.01	0.0
	120	—	>600	—	1,004	1.10	0.0	4,072	0.12	0.0	481	<0.01	0.0
	180	—	>600	—	2,232	1.09	0.0	10,800+	—	—	555	<0.01	0.0

Time limit: 3 hours for enumeration, 10 minutes per subproblem.
Enum CPU: time to enumerate the decision variables (segments, subpaths, paths), Solve CPU: average time to solve single subproblem, Gap: average optimality gap at the time limit and "—" if no feasible solution was found.

Even in the smallest instances, the compact formulation takes over than ten minutes to solve for each subproblem. This stems from the weak relaxation leading to limited scalability of off-the-shelf branch-and-cut algorithms. This shows that, in our MiND problem, even the second-stage is challenging to solve by itself. All network-based formulations are comparatively more efficient, terminating in fractions of a second. Still, the subpath-based formulation solves orders of magnitude faster than the segment-based one. Although the segment-based model retains a polynomial size, its slightly weaker relaxation and, most importantly, its large number of variables in the dense time-station-load network create significant computational complexities. In the context of the full MiND problem, these differences in solution times have a significant impact because hundreds of subproblems are solved at each Benders iteration. Finally, the subpath-based formulation involves much fewer arcs than the path-based one, thereby resulting in significant speedups in pre-processing.

The path-based model could also be solved via column generation—like our subpath-based model. However, we have found that subpath-based column generation also converges orders of magnitude faster than path-based column generation (0.03 second vs. 4.1 seconds on average per pricing problem). In other words, the reductions in the number of variables and in pre-processing times identified in Table EC.7 lead to similar reductions in computational times for the subpath-based vs. path-based column generation. This stems from the fact that the label-setting algorithm is applied between checkpoints in subpath-based column generation, as opposed to from the beginning to the end of a reference trip in path-based column generation—leading to an exponential decrease in the number of paths and passenger combinations.

These results underscore that the different formulations face different bottlenecks: the branch-and-cut structure in the compact formulation due to its weak linear relaxation, the linear relaxation in the segment-based formulation due to its large size, and the generation of arc variables in the path-based and subpath-based formulations either through exhaustive enumeration at the pre-processing stage or through column generation. In larger instances, subpath or path enumeration becomes intractable, which motivated our double-decomposition algorithm.

EC.6.2. Sensitivity analyses

Sensitivity to the number of skipped checkpoints. We perform additional sensitivity analyses to study the impact of K (that is, the number of consecutive checkpoints that can be skipped) on computational times and solution quality. Results are reported in Table EC.8. We consider here a small instance with 5 candidate lines, 5 scenarios, and a one-hour horizon, as larger instances become intractable with $K = 3$. As expected, the computational times increase significantly as K becomes larger because of the exponential growth in the number of subpaths. In this setting, the algorithm can solve all instances with $K = 0$ within a minute or so, but can take up to 10-25 minutes with $K = 1$, over an hour with $K = 2$, and reaches the 3-hour time limit with $K = 3$. Even with larger values of K, the majority of the subpaths still visit consecutive checkpoints (even without the non-linear penalties associated with longer subpaths discussed in the first point above). This is primarily driven by the penalties on in-vehicle time and arrival delays, which discourage longer subpaths. In turn, a value of $K = 1$ achieves most of the benefits obtained with the larger values of K (notably, by increasing coverage by 30–40%). In comparison, the marginal benefits with $K = 2$ become smaller—although some higher-coverage solutions are obtained with $K = 3$. Altogether, these results strengthen one of our practical takeaways that a limited extent of flexibility can provide strong benefits (Table 5).

Sensitivity to the specification of the objective function. Recall that the model formulation trades off several underlying goals: operating costs, demand coverage, and passenger level of service (itself comprising walking times, waiting times, in-vehicle travel times, and delay at destination). Table EC.9 reports sensitivity analysis results to explore the trade-off between these components, both for microtransit and fixed-route transit. The first eight rows vary the four level-of-service weights together, ranging from a setting where demand coverage is prioritized (small λ, μ, σ, and δ) to a setting where passenger level of service is prioritized (large λ, μ, σ, and δ). The next rows vary one weight parameter at a time, reflecting different balances between level-of-service

Table EC.8 Results for MiND-VRP with varying K and Δ.

				Passenger metrics					Subpaths skipping k checkpoints			
K	Δ	Solution	CPU (s)	Pickups	Walk	Wait	Detour	Delay	$k=0$	$k=1$	$k=2$	$k=3$
0	600	138.5	14	135	3.11	2.68	140%	0.01	100%	—	—	—
	1200	136.6	48	139	3.02	2.76	141%	0.02	100%	—	—	—
	1800	136.6	51	139	3.02	2.77	142%	0.02	100%	—	—	—
	2400	136.6	86	139	3.02	2.76	142%	0.02	100%	—	—	—
1	600	117.9	67	178	1.57	1.77	137%	0.02	62%	38%	—	—
	1200	111.0	376	193	4.26	2.04	137%	0.02	56%	44%	—	—
	1800	110.9	819	193	3.48	1.92	137%	0.02	58%	42%	—	—
	2400	111.0	1,544	193	3.48	2.10	137%	0.02	57%	43%	—	—
2	600	116.0	157	182	1.54	1.68	138%	0.02	62%	22%	14%	—
	1200	107.5	1,223	200	1.21	1.70	137%	0.02	67%	16%	17%	—
	1800	106.1	3,970	203	1.19	1.67	137%	0.02	63%	22%	14%	—
	2400	105.7	7,724	204	1.19	1.84	137%	0.02	65%	20%	15%	—
3	600	113.1	260	188	2.98	1.53	139%	0.01	67%	8%	6%	19%
	1200	101.9	2,843	212	1.89	1.71	139%	0.02	67%	7%	7%	19%
	1800	100.0	9,370	217	2.49	2.10	149%	0.02	70%	3%	3%	23%
	2400	116.0	10,800+	184	1.52	2.78	156%	0.02	69%	12%	3%	15%

Table EC.9 Sensitivity to weight parameters (50 candidate lines, 5 scenarios, 2 hours).

First stage	Second Stage							Avg. level of service					Distance (km)		
Line cost	λ	μ	σ	δ	Mode	Lines	Trips	Coverage	Walk	Wait	Detour	Delay	Internal	External	Total
1	**0.1**	**0.1**	**0.1**	**0.1**	MT	21	42	36.6%	1.56	5.75	127%	0.01	673	12,061	12,734
					Transit	18	44	35.9%	2.16	6.88	121%	0.00	622	12,178	12,800
	1	1	1	1	MT	18	42	36.6%	1.54	5.74	128%	0.01	681	12,039	12,720
					Transit	19	44	35.9%	2.11	6.82	121%	0.01	623	12,170	12,793
	10	**10**	**10**	**10**	MT	18	42	36.5%	1.30	5.23	128%	0.01	687	12,049	12,736
					Transit	19	44	35.7%	2.14	6.73	122%	0.01	626	12,188	12,814
	100	**100**	**100**	**100**	MT	11	37	10.2%	0.00	0.86	135%	0.00	512	15,518	16,030
					Transit	6	31	2.8%	0.00	0.74	142%	-0.01	196	16,476	16,672
1	**10**	1	1	1	MT	20	43	36.8%	0.67	5.28	126%	0.01	688	12,041	12,729
					Transit	20	44	35.7%	1.78	6.64	123%	0.01	622	12,196	12,818
	100	1	1	1	MT	21	43	30.3%	0.00	4.09	127%	0.01	664	12,917	13,581
					Transit	16	42	22.4%	0.00	4.77	117%	0.00	572	13,909	14,481
1	1	**10**	1	1	MT	16	42	37.1%	2.38	6.77	130%	0.01	681	11,973	12,654
					Transit	19	44	35.7%	2.41	7.19	123%	0.00	625	12,201	12,826
	1	**100**	1	1	MT	16	42	29.9%	3.17	7.45	120%	0.01	641	12,972	13,613
					Transit	16	42	25.0%	3.20	7.75	121%	0.01	577	13,588	14,165
1	1	1	**10**	1	MT	22	42	36.0%	1.47	5.59	127%	0.01	678	12,149	12,827
					Transit	18	44	35.7%	2.16	6.87	121%	0.00	618	12,191	12,809
	1	1	**100**	1	MT	17	41	36.3%	1.57	5.72	128%	0.01	671	12,101	12,772
					Transit	20	44	35.9%	2.15	6.85	120%	0.01	619	12,172	12,791
1	1	1	1	**10**	MT	19	42	36.2%	1.50	5.61	128%	0.01	687	12,109	12,796
					Transit	20	44	35.8%	2.16	6.89	121%	0.00	624	12,181	12,805
	1	1	1	**100**	MT	20	42	36.4%	1.53	5.70	128%	0.01	679	12,084	12,763
					Transit	20	44	35.8%	2.11	6.81	122%	0.00	624	12,172	12,796

components. We consider very large variations in these weight parameters to analyze outcomes where one objective component is strongly prioritized over the others.

The results are highly robust to the choice of the underlying weight parameters. The general structure of the optimal solution remains stable across virtually all instances. Exceptions arise when the weight of the level-of-service costs (especially, of waiting costs) is multiplied by a factor of 100, in which case the model prioritizes passengers that can be served at the requested time at the expense of not covering other passengers. Otherwise, the solution remains mostly unchanged; in particular, the number of pickups varies by less than 1%, with similar detours and delays, and variations in walking and waiting times on the order of 0–2 minutes on average.

Our main takeaways are therefore robust to the objective specification. In particular, the benefits of microtransit over fixed-route transit hold across all parameter values. In most settings, the microtransit solution increases demand coverage and reduces walk and wait times as compared to fixed-route transit, at limited costs in terms of detours and distance traveled. In turn, the microtransit solution retains efficiency benefits (higher coverage and higher level of service) as well as sustainability benefits (lower total distance, after accounting for the external distance of non-covered passengers). At the extreme where passenger level is heavily prioritized and where, in particular, walking is heavily penalized, both solutions involve virtually no walking for passengers; but then, the flexibility of microtransit enables large increases in demand coverage as compared to the transit solution. Altogether, the finding that microtransit can result in win-win outcomes does not depend on idiosyncratic choices of the weight parameters but holds across a range of trade-off choices between demand coverage and passenger level of service.