

MIT Open Access Articles

Data-Driven Transit Network Design at Scale

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

Citation: Bertsimas, Dimitris, Ng, Yee Sian and Yan, Julia. 2021. "Data-Driven Transit Network Design at Scale." *Operations Research*, 69 (4).

As Published: 10.1287/OPRE.2020.2057

Publisher: Institute for Operations Research and the Management Sciences (INFORMS)

Persistent URL: <https://hdl.handle.net/1721.1/144080>

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Terms of use: Creative Commons Attribution-Noncommercial-Share Alike



Authors are encouraged to submit new papers to INFORMS journals by means of a style file template, which includes the journal title. However, use of a template does not certify that the paper has been accepted for publication in the named journal. INFORMS journal templates are for the exclusive purpose of submitting to an INFORMS journal and should not be used to distribute the papers in print or online or to submit the papers to another publication.

Data-Driven Transit Network Design at Scale

Dimitris Bertsimas

Sloan School of Management and Operations Research Center, Massachusetts Institute of Technology, dbertsim@mit.edu

Yee Sian Ng

Operations Research Center, Massachusetts Institute of Technology, yeesian@mit.edu

Julia Yan

Operations Research Center, Massachusetts Institute of Technology, jyyan@mit.edu

Mass transit remains the most efficient way to service a densely-packed commuter population. However, reliability issues and increasing competition in the transportation space have led to declining ridership across the United States, and transit agencies must also operate under tight budget constraints. Recent attempts at using bus network re-design to improve ridership have attracted attention from various transit authorities. However, the analysis seems to rely on ad hoc methods, for example, considering each line in isolation and using manual incremental adjustments with backtracking. We provide a holistic approach to designing a transit network using column generation. Our approach [scales](#) to hundreds of stops, and we demonstrate its usefulness on a case study with real data from Boston.

Key words:

History:

1. Introduction

The United Nations (2014) projects that urban populations will increase from 54 percent of the global population in 2014 to 66 percent in 2050. Together with overall population growth, this represents an increase of 2.5 billion people. With more of the world's population living in cities, it is increasingly important to provide transportation that can efficiently serve a densely-settled population while also achieving societal goals of sustainability and equity.

Mass transit is critical in achieving these goals; however, in recent years, it has faced challenges such as restrictive public budgets and significant outside competition from ride-sharing companies and private bus or shuttle services (Bouton et al. 2015). Many American cities such as Philadelphia (Laughlin 2018), Los Angeles (Nelson 2018), Washington D.C. (Siddiqui 2018) are seeing declining bus ridership, prompting transit authorities to consider what can be done to halt this decline.

A recent bus network re-design in Houston led to a 6.8% increase in ridership across the bus and light rail networks (Binkovitz 2016), inspiring other cities to also consider re-designing their bus networks. Examples include Philadelphia (Laughlin 2017), Boston (Vaccaro 2018), St. Louis (Schlinkmann 2018), and Edmonton (Stolte 2018).

The problem of designing a set of sequences of stops, called *lines* or *routes*, in order to service a commuting population is called the Transit Network Design Problem (TNDP). The TNDP is a challenging combinatorial optimization problem and has been well-studied in the literature, which we survey in Section 2. Despite this body of work, advanced techniques are not used to a significant extent in the planning process for actual network design. For instance, the Service Delivery Policy outlined by the Massachusetts Bay Transportation Authority (2017) analyzed the network by evaluating bus lines individually without considering the network as a whole, and improvements to bus networks were performed as incremental adjustments to individual bus lines. Such a strategy clearly limits the scope of a potential bus network re-design.

One of the main barriers to leveraging advanced techniques is scalability; many algorithms have not been proven on the scale that real transit networks require, which can be up to hundreds or thousands of stops. Due to the combinatorial explosion in the number of possible service routes, much of the work in this area relies on the use of heuristics and metaheuristics, or limits the scope to selection of a set of pre-existing transit lines. Our work makes the following contributions:

1. We present an optimization model that generates new transit networks while addressing the issues of interest to transit authorities, which are principally *ridership*, *connectivity*, and *budget*. Ridership represents serviced demand, and is also the main driver of revenue to a transit agency. Connectivity ensures that commuters can go from origin to destination in a relatively direct manner. Budget recognizes the labor and financial constraints that agencies operate under.
2. We address two crucial interests from the commuter perspective: number of transfers, and travel time. If either of them are not adequately serviced, then transit ridership will decrease. Therefore, properly accounting for them is key to generate practical transit network designs that addresses commuters' needs.
3. We develop a scalable algorithm based on column generation, and demonstrate its efficacy and tractability on a real network in Boston with hundreds of stops and thousands of edges.

The rest of the paper is outlined as follows. In Section 2, we provide an overview of the literature on the TNDP. In Section 3, we describe our model and algorithmic approach. In Section 4, we show computational results on a variety of case studies involving both synthetic and real data. Finally, we offer concluding remarks in Section 5.

2. Literature Review

The TNDP is well-studied; for a review of material until the 1980s, see Magnanti and Wong (1984), and for more recent reviews, see Guihaire and Hao (2008) and Farahani et al. (2013).

The goal of the TNDP is to design a set of lines for buses or trains to serve transit demand in a cost-effective way. Auxiliary objectives important to transit operators such as service area coverage may also be incorporated. Most methods either begin with a pre-specified set of potential lines or iteratively generate new lines before selecting a final set of lines to operate.

Much of the early work on the TNDP focused on heuristic solution methods. Typically, the origin-destination demand matrix was sorted from highest to lowest demand, and bus routes were generated using fast shortest-path computations between high-demand nodes. Mandl (1980) generate an initial line set by computing the shortest paths between terminal nodes, and then uses local search to iteratively improve the total travel time on the network. Ceder and Wilson (1986) and Baaj and Mahmassani (1995) extended this work by including additional lines that are no longer than a factor of the length of the shortest paths. In addition, Baaj and Mahmassani (1995) considered criteria for local node insertions to further expand the generated lines. More recently, Gattermann et al. (2017) generated lines using the minimum spanning tree on the transit network. Along a similar vein, metaheuristics such as genetic algorithms (Cipriani et al. 2012, Walteros et al. 2013), simulated annealing (Zhao and Zeng 2006), and tabu search (Lownes and Machemehl 2010) have also been used to iteratively improve upon initial heuristically-generated line sets. Zhao and Ubaka (2004) introduced the notion of route-directness and network-directness constraints to capture geometric characteristics of desirable lines, and used a hill-climb search algorithm to make local improvements to an initial network. Yu et al. (2012) used ant-colony optimization and considered direct demands and single transfers in route design. Pinelli et al. (2016) combined and made local improvements to common trajectories gleaned from mobile phone data to propose modified bus lines.

Another body of work has employed mathematical optimization to solve network design problems. Benefits of mathematical optimization include modeling flexibility, problem insight, and certificates of optimality or bounds on solution quality; however, many models have had scalability issues at practical network sizes. Most approaches split the decision process into multiple phases, including the optimal selection of a subset of transit stops (Murray 2003), the selection of a subset of given transit lines Guan et al. (2006), or frequency-setting and passenger routing on a set of given transit lines (Bussieck et al. 1997, Goossens et al. 2004, Schöbel and Scholl 2006, Cancela et al. 2015). Even with these decompositions, Guan et al. (2006) scaled to a network of only 49 stops, which was preprocessed to reduce the size to nine stops. Cancela et al. (2015) additionally

considered the passenger perspective through a lower-level assignment model, and scaled to a larger network of 84 nodes and 143 edges. Goossens et al. (2004) performed computational experiments on a network of 141 nodes and 177 edges.

Relatively fewer papers in mathematical optimization have addressed the critical but more complicated step of generating new transit lines. Wan and Lo (2003) used mixed-integer optimization to create a fixed number of routes between given bus stops that would minimize operating costs subject to capacity constraints. However, the formulation is not practical, scaling only to a network of ten stops. Barra et al. (2007) used constraint programming to define service level goals and budget limits, but it was computationally difficult to find a solution even for a small fifteen-stop case. Rather than relying purely on branch-and-bound, Marín and Jaramillo (2009) used a variant of Benders decomposition to solve a network design problem on 24 stops and 264 edges.

In contrast to these smaller-scale examples, Borndörfer et al. (2007) employed column generation to scale up their model to a network of 410 stops and 891 edges, a truly large-scale application. Their model sought to generate a set of lines, choose a subset of lines to operate, and set service frequencies in order to minimize a combination of total operating costs and travel time. However, they remained closely tethered to the original network design by only considering already-extant edges in their computational study, so that the new lines were rearrangements of existing lines. Furthermore, in their model of commuter behavior, they ignored transfers between lines of the same mode and allowed for passenger flows of unlimited travel times and transfers in their model. Borndörfer and Karbstein (2012) expanded on Borndörfer et al. (2007) by modeling passenger behavior in greater detail; in particular, passenger flows were penalized if they involve a transfer. However, although the model improved the accounting of passenger flows, it did not consider the problem of generating new lines, instead taking the initial set of lines as given. In more recent work, Jin et al. (2016) also used column generation to generate bus lines, although they focused on the smaller-scale application of responding to disruptions on targeted subsets of the transit network.

Our work is most similar to Borndörfer et al. (2007) and Borndörfer and Karbstein (2012). As compared to Borndörfer et al. (2007), we explicitly model both direct and indirect passenger routes while generating new transit lines and also setting frequencies on large-scale networks. In addition, rather than allowing continuous frequencies, we select from a subset of frequencies, thus avoiding the issue of generating many lines of unrealistically low frequency. In contrast to Borndörfer and Karbstein (2012), we do not model distinct commuter paths along the network in as fine detail; we still account for travel times and transferring, but opt for a less granular approach in favor of gaining the ability to generate new transit lines. Finally, in contrast to both works, our transit networks only serve commuters who experience reasonable travel times, rather than allowing arbitrarily high travel times in passenger flows.

3. Methods

We consider the problem of designing a transit network, and provide a description of the problem parameters in Table 1.

Table 1 Summary of the Notation

Symbol	Description
\mathcal{N}	set of all transit stops
\mathcal{D}	set of all origin-destination pairs (commutes)
$\mathcal{T}_{u,v}$	the set of all viable transfer stops for commute (u, v) .
\mathcal{L}	set of all transit lines
\mathcal{F}	set of all service frequencies
$\text{stops}(\ell)$	set of all stops on line ℓ
$d_{u,v}$	demand for commute (u, v)
c_ℓ	total travel time along line ℓ
$\delta_{u,v}$	direct travel time between stops u and v
ρ_f	relative cost at frequency f
γ_f	relative level of direct ridership at frequency f
λ_f	relative level of transferring ridership at frequency f
N	number of transit stops, $ \mathcal{N} $
B	budget available for network design

The transit network is built on a set of N transit stops, which is denoted by \mathcal{N} and indexed from $1, \dots, N$. The transit network's purpose is to service commuting demand; the notation $d_{u,v}$ will be used to refer to the demand for the *commute* from origin u to destination v . The set of all commutes with **positive** demand will be referred to as \mathcal{D} . Each commute may be associated with a number of different route options, each of which is a different sequence of stops that can take a commuter from origin u to destination v .

Vehicles on the network are set to travel along a fixed sequence of stops, which we call a *line*. In practice, transit lines are typically bidirectional, so we take each line to actually represent two lines traveling in opposite directions between the terminal stops. We assume that the operating cost of a transit line ℓ is related to the sum of the travel times $\delta_{u,v}$ between pairs of consecutive stops u and v on the line; this summation will be denoted by c_ℓ . Each line is also associated with a frequency chosen from a set of frequencies \mathcal{F} . For example, this set could comprise the frequencies of dispatching vehicles every 15, 30, and 60 minutes (high, medium, and low frequencies, respectively). We further assume that the operating cost of a transit line should be related to its service frequency in addition to its total length, and that at a given service frequency $f \in \mathcal{F}$, the cost of the transit line is given by $c_\ell \rho_f$, where the constant ρ_f scales the cost appropriately with the frequency. This simplified cost structure omits fixed costs, but our model can easily be extended to account for such cost structures. The total budget available is given by B .

We also introduce the coefficients γ_f and λ_f for frequency level $f \in \mathcal{F}$ to denote relative levels of direct ridership and transferring ridership, respectively. In our model, these coefficients are set relative to the highest frequency, which should accrue the highest levels of ridership while also costing the most. Accordingly, for the highest service frequency, we set the relevant cost coefficient ρ_f and direct ridership coefficient γ_f both to equal 1.0, indicating that the full cost is incurred and all of the ridership is captured. The transferring coefficient λ_f might be less than 1.0 even at the highest level of frequency if commuters are not willing to make transfers. At lower service frequencies, all three coefficients should decay as the transit service becomes cheaper and less appealing, especially as commuters are unwilling to make transfers at low service frequencies. This intuition can be used to set reasonable coefficients, where the cost coefficients are roughly proportional to frequency level, the direct ridership coefficients might decay more slowly due to inelasticity in commuting demand, and the transferring ridership coefficients might decay more quickly due to commuter aversion to transfers. Some illustrative examples are given in the computational experiments; however, in a practical application, the ridership coefficients should be determined using survey data.

We use boldface letters to denote vectors, and generalized inequalities on them are taken to be element-wise. The boldface letter \mathbf{e} refers to the vector of all ones, and $|\mathcal{S}|$ refers to the cardinality of set \mathcal{S} .

3.1. Serving direct passengers

For simplicity, we first consider a network where commuters only take transit if they can get from origin directly to destination without transfers. We also first assume that there is a set \mathcal{L} of all feasible transit lines; later in this section, we will describe how to efficiently construct this set. This initial model uses the following variables:

- $x_{\ell,f}$: 1 if line $\ell \in \mathcal{L}$ is operated at frequency level f , 0 otherwise, and
- $\theta_{u,v}$: the fraction of demand for commute $(u, v) \in \mathcal{D}$ that is served by the network.

To design a network that has high ridership, we solve the following integer optimization problem:

$$\text{TNDP}(\mathcal{L}) = \max_{\mathbf{x}, \boldsymbol{\theta}} \sum_{(u,v) \in \mathcal{D}} d_{u,v} \theta_{u,v} \quad (1a)$$

$$\text{s.t. } \theta_{u,v} \leq \sum_{\substack{\ell \in \mathcal{L}: \\ u \in \text{stops}(\ell), \\ v \in \text{stops}(\ell)}} \sum_{f \in \mathcal{F}} \gamma_f x_{\ell,f} \quad \forall (u,v) \in \mathcal{D}, \quad (1b)$$

$$\sum_{\ell \in \mathcal{L}} \sum_{f \in \mathcal{F}} c_{\ell} \rho_f x_{\ell,f} \leq B \quad (1c)$$

$$\sum_{f \in \mathcal{F}} x_{\ell,f} \leq 1 \quad \forall \ell \in \mathcal{L}, \quad (1d)$$

$$\boldsymbol{\theta} \leq \mathbf{e} \quad (1e)$$

$$\mathbf{x} \in \{0, 1\}^{\mathcal{L} \times \mathcal{F}}. \quad (1f)$$

The objective (1a) maximizes the total demand that is served by the transit network. Constraint (1b) requires that some line that connects both u and v should be operated in order for those commuters to be serviced, and also scales the ridership according to the frequency coefficients γ_f . Constraint (1c) is the budget constraint on the lines to be operated. Constraint (1d) requires that each line can only be operated at one frequency. Constraint (1e) enforces that the fraction of demand served should be at most one, which is important in case a particular commute can be served by multiple appealing lines in constraint (1b). Note that since \mathcal{D} only includes commutes (u, v) with positive demand $d_{u,v}$ and since the objective maximizes the total demand served, nonnegativity constraints on θ are not necessary. Capacity constraints are omitted here for ease of notation, but can easily be incorporated.

Our model focuses on the operator perspective and seeks to maximize ridership, which a common metric used by transit authorities to measure operating performance (Binkovitz 2016). In maximizing ridership, we allow for the case that a transit network is unable to service all demand. This is important because when budgets are tight, or when a transit agency is considering expansion into new areas, the requirement that all demand is served may be onerous.

Our model considers a discrete set of frequencies to choose from, which follows the convention of Goossens et al. (2004) and Borndörfer and Karbstein (2012). A possible simplification might be instead to use continuous \mathbf{x} variables to model frequencies, as in Borndörfer et al. (2007). However, in practice, the frequencies are actually determined from a discrete set; this set is often small, sometimes even just at the granularity of “high” versus “low” frequency. In addition to more closely reflecting true operator decision-making, using a discrete set of frequencies also allows for the modeling of effects such as diminishing marginal returns through appropriate setting of the ρ_f and γ_f variables. Finally, choosing frequencies from a discrete set makes for more interpretable networks, rather than encouraging many lines of very low frequency.

Problem (1) is in general difficult to solve, as the set \mathcal{L} of all feasible transit lines will be extremely large. However, it can be solved efficiently using column generation. For a comprehensive overview of column generation, see Barnhart et al. (1998). We begin with the linear relaxation, following examples such as Borndörfer et al. (2007) and Jin et al. (2016):

$$\text{MP}(\mathcal{L}) = \max_{\mathbf{x}, \theta} \sum_{(u,v) \in \mathcal{D}} d_{u,v} \theta_{u,v} \quad (2a)$$

$$\text{s.t. } \theta_{u,v} \leq \sum_{\substack{\ell \in \mathcal{L}: \\ u \in \text{stops}(\ell), \\ v \in \text{stops}(\ell)}} \sum_{f \in \mathcal{F}} \gamma_f x_{\ell,f} \quad \forall (u,v) \in \mathcal{D}, \quad (2b)$$

$$\sum_{\ell \in \mathcal{L}} \sum_{f \in \mathcal{F}} c_{\ell} \rho_f x_{\ell,f} \leq B \quad (2c)$$

$$\sum_{f \in \mathcal{F}} x_{\ell, f} \leq 1 \quad \forall \ell \in \mathcal{L}, \quad (2d)$$

$$\boldsymbol{\theta} \leq \mathbf{e} \quad (2e)$$

$$\mathbf{x} \geq 0. \quad (2f)$$

which we call $\text{MP}(\mathcal{L})$ to stand for the *master problem*. We have not written out the constraint $\mathbf{x} \leq \mathbf{e}$, since constraint (2d) combined with the nonnegativity constraint on \mathbf{x} will naturally enforce the elementwise upper bound on \mathbf{x} .

The dual of $\text{MP}(\mathcal{L})$ is as follows:

$$D(\mathcal{L}) = \min_{\mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{s}} Bq + \sum_{\ell \in \mathcal{L}} r_{\ell} + \sum_{(u, v) \in \mathcal{D}} s_{u, v} \quad (3a)$$

$$\text{s.t.} \quad - \sum_{\substack{(u, v) \in \mathcal{D}: \\ u \in \text{stops}(\ell), \\ v \in \text{stops}(\ell)}} \gamma_f p_{u, v} + c_{\ell} \rho_f q + r_{\ell} \geq 0 \quad \forall (u, v) \in \mathcal{D}, \forall f \in \mathcal{F}; \quad (3b)$$

$$p_{u, v} + s_{u, v} = d_{u, v} \quad \forall (u, v) \in \mathcal{D}, \quad (3c)$$

$$\mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{s} \geq 0. \quad (3d)$$

Evidently, by solving the dual (3), we also solve the primal (2).

Our column generation algorithm proceeds iteratively as follows. First, we begin with a restricted set of transit lines $\bar{\mathcal{L}} \subset \mathcal{L}$ and the corresponding subset of variables $(x_{\ell, f})_{\ell \in \bar{\mathcal{L}}, f \in \mathcal{F}}$. We solve the *restricted master problem*, which is the primal (2) on this restricted set to get a primal solution $\bar{\mathbf{x}}$ and corresponding dual solution $(\bar{\mathbf{p}}, \bar{q}, \bar{\mathbf{r}}, \bar{\mathbf{s}})$. To see if the solution $\bar{\mathbf{x}}$ is optimal for the full problem, we check whether there exists some line $\ell \in \mathcal{L} \setminus \bar{\mathcal{L}}$ at frequency $f \in \mathcal{F}$ that violates the constraint (3b), i.e.,

$$c_{\ell} \rho_f \bar{q} < \sum_{\substack{(u, v) \in \mathcal{D}: \\ u \in \text{stops}(\ell), \\ v \in \text{stops}(\ell)}} \gamma_f \bar{p}_{u, v}. \quad (4)$$

The right-hand-side of the violated constraint (4) is interpreted as the increase in the primal ridership objective due to servicing commutes (u, v) on line ℓ . For example, a dual variable $p_{u, v}$ might represent increased ridership from commutes (u, v) that are not already served by the network, either because no line exists connecting stops u and v , or because any connecting lines were not included in the network due to the budget constraint. The left-hand-side of the violated constraint (4) contains dual variable q , which represents the increase in ridership associated with a unit increase in budget, and is in the appropriate units converting budget to ridership. Condition (4) therefore requires that, for a new line to be profitable, the ridership increase should outweigh the

associated costs of the line. For example, if the budget constraint is not tight, then $\bar{q} = 0$ by complementary slackness, and the algorithm searches for any lines with positive ridership. **Note that for the newly generated line ℓ , complementary slackness implies that $r_\ell = 0$, and so is omitted from (4).**

New profitable lines ℓ can be generated by solving the *subproblem*, which is formulated as an integer optimization problem. When only considering the cost of connecting adjacent stops, the subproblem is a shortest path problem. However, the problem is more complicated with the additional ridership computation. To solve the subproblem, we first define a directed graph $G(\mathcal{V}, \mathcal{E})$ on nodes \mathcal{V} and directed edges \mathcal{E} . The node set \mathcal{V} consists of the transit stops \mathcal{N} (labeled $1, \dots, N$), the source (labeled 0), and the sink (labeled $N + 1$). The edge set \mathcal{E} consists of the following types of directed edges:

- edges from the source (0) to all stop nodes $u \in \mathcal{N}$,
- edges between the stop nodes, and
- edges from the stop nodes $u \in \mathcal{N}$ to the sink ($N + 1$).

The subproblem uses the following decision variables:

- $h_{i,j}$: 1 if edge $(i, j) \in \mathcal{E}$ is used, 0 otherwise, and
- $g_{u,v}$: 1 if commute $(u, v) \in \mathcal{D}$ can be served, 0 otherwise.

A transit line corresponds to a simple path from source to sink in $G(\mathcal{V}, \mathcal{E})$. In practice, transit lines are typically bidirectional, so we take each transit line to actually represent two transit lines traveling in opposite directions between the terminal stops. Therefore, a transit line can serve any commuters whose origin and destination are both on the path from source to sink regardless of ordering, and the relationship between the \mathbf{h} and \mathbf{g} variables is enforced by the following constraints:

$$g_{u,v} \leq \sum_{v' \in \text{In}(u)} h_{v',u} \quad \forall (u, v) \in \mathcal{D}, \quad (5a)$$

$$g_{u,v} \leq \sum_{u' \in \text{In}(v)} h_{u',v} \quad \forall (u, v) \in \mathcal{D}, \quad (5b)$$

where the right-hand sides of constraints (5) indicate that nodes u and v respectively are present in the generated line. We use $\text{In}(u)$ ($\text{Out}(u)$) to refer to the set of nodes, including the source (sink), that have edges incoming to (outgoing from) node u . Constraints (5) are needed because the \mathbf{g} variables will be maximized in the objective function.

Recalling that the general cost c_ℓ of a transit line can be computed by summing the travel times between consecutive stops on the line, we search for a profitable line at frequency $f \in \mathcal{F}$ by solving the following integer optimization problem:

$$\text{SP}_f(\bar{\mathbf{p}}, \bar{q}; G(\mathcal{V}, \mathcal{E})) = \max_{\mathbf{h}, \mathbf{g}} \sum_{(u,v) \in \mathcal{D}} \gamma_f \bar{p}_{u,v} g_{u,v} - \sum_{(u,v) \in \mathcal{E}} \rho_f \bar{q} \delta_{u,v} h_{u,v} \quad (6a)$$

$$\text{s.t. } \sum_{u \in \mathcal{N}} h_{0,u} = 1 \quad (6b)$$

$$\sum_{u \in \mathcal{N}} h_{N+1,u} = 1 \quad (6c)$$

$$\sum_{v \in \text{Out}(u)} h_{u,v} - \sum_{v \in \text{In}(u)} h_{v,u} = 0 \quad \forall u \in \mathcal{N}, \quad (6d)$$

$$g_{u,v} \leq \sum_{v' \in \text{In}(u)} h_{v',u} \quad \forall (u,v) \in \mathcal{D}, \quad (6e)$$

$$g_{u,v} \leq \sum_{u' \in \text{In}(v)} h_{u',v} \quad \forall (u,v) \in \mathcal{D}, \quad (6f)$$

$$\sum_{\substack{(u,v) \in \mathcal{E}: \\ u \in \mathcal{S}, v \in \mathcal{S}}} h_{u,v} \leq |\mathcal{S}| - 1 \quad \forall \mathcal{S} \subset \mathcal{V}, \quad (6g)$$

$$\mathbf{h} \in \{0, 1\}^{\mathcal{E}}, \mathbf{g} \in \{0, 1\}^{\mathcal{D}}. \quad (6h)$$

In Problem (6), the objective (6a) checks for the profitability condition (4). Constraints (6b) through (6d) are network flow constraints, and constraints (6e) and (6f) correspond to (5). Constraints (6g) are *subtour elimination constraints* that are required to prevent the formation of edge-disjoint cycles; otherwise the objective could gain credit for connecting stops belonging to separate cycles. Since the number of potential cycles in the graph is large, these constraints are added lazily using branch-and-cut: violated constraints are identified and added to the problem at incumbent integral solutions.

Although the subproblem has been formulated in a basic way, a variety of conditions of interest to transit planners can be modeled using additional constraints. We provide some illustrative examples here.

- **Depots:** In some cities, transit lines must begin and end at certain predesignated depot stops. This can be modeled in the subproblem by eliminating all edges between the source and non-depot nodes, and similarly for the sink.
- **Edge Lengths:** It may not be desirable for buses to make stops that are too close or far apart. This can be modeled in the subproblem by eliminating all edges between stops that are either too short or too long, which has the additional benefit of significantly sparsifying the graph $G(\mathcal{V}, \mathcal{E})$.
- **Line Length:** It may not be desirable for a transit line to contain too many or too few stops. This can be modeled in the subproblem by restricting the number of edges that are used.
- **Obstacles:** If it is impossible to travel from stop u to stop v , then that edge can be eliminated. Algorithm 1 describes the full column generation scheme.

Note that the subproblem $\text{SP}_f(\bar{\mathbf{p}}, \bar{q}; G(\mathcal{V}, \mathcal{E}))$ has been defined for a particular frequency $f \in \mathcal{F}$. In order for the master problem to be solved to optimality, there should not be any frequency for

Algorithm 1 Column generation algorithm for TNDP (1)

Require: Initial set of transit lines $\bar{\mathcal{L}}$, tolerance $\epsilon > 0$

- 1: Solve $\text{MP}(\bar{\mathcal{L}})$, get primal solution $(\bar{\mathbf{x}}, \bar{\boldsymbol{\theta}})$ and dual solution $(\bar{\mathbf{p}}, \bar{q}, \bar{\mathbf{s}})$
 - 2: $J_f \leftarrow$ objective value of $\text{SP}_f(\bar{\mathbf{p}}, \bar{q}; G(\mathcal{V}, \mathcal{E}))$ for all $f \in \mathcal{F}$
 - 3: $\ell_f \leftarrow$ transit line solution to $\text{SP}_f(\bar{\mathbf{p}}, \bar{q}; G(\mathcal{V}, \mathcal{E}))$ for all $f \in \mathcal{F}$
 - 4: **while** $\max_{f \in \mathcal{F}} J_f > \epsilon$ **do**
 - 5: $f^* \leftarrow \arg \max_{f \in \mathcal{F}} J_f$
 - 6: $\bar{\mathcal{L}} \leftarrow \bar{\mathcal{L}} \cup \ell_{f^*}$
 - 7: Solve $\text{MP}(\bar{\mathcal{L}})$, get primal solution $(\bar{\mathbf{x}}, \bar{\boldsymbol{\theta}})$ and dual solution $(\bar{\mathbf{p}}, \bar{q}, \bar{\mathbf{s}})$
 - 8: $J_f \leftarrow$ objective value of $\text{SP}_f(\bar{\mathbf{p}}, \bar{q}; G(\mathcal{V}, \mathcal{E}))$ for all $f \in \mathcal{F}$
 - 9: $\ell_f \leftarrow$ transit line solution to $\text{SP}_f(\bar{\mathbf{p}}, \bar{q}; G(\mathcal{V}, \mathcal{E}))$ for all $f \in \mathcal{F}$
 - 10: Solve TNDP($\bar{\mathcal{L}}$) (a mixed-integer optimization problem), get solution $(\mathbf{x}^*, \boldsymbol{\theta}^*)$
 - 11: **return** \mathbf{x}^*
-

which a transit line of positive reduced cost can be found, which means that the subproblem would need to be solved at all frequencies. To save computation time, we can modify Algorithm 1 so that early iterations solve the subproblem at just one frequency, and only check the other frequencies if no transit line of positive reduced cost can be found. Other column management strategies can be found in Barnhart et al. (1998).

3.2. Serving passengers with transfers

The assumption of Problem (1) that commuters will only take transit if it can get them directly from origin to destination is a restrictive one. In ignoring transferring commuters, transit networks are forced to be excessively connected to gain ridership. However, this could lead to significant redundancies and therefore cost inefficiencies in the transit network. We therefore generalize to the case with two lines, where commuters are willing to make a single transfer. We call the model in Section 3.1 the *direct-route* model, and we call the model in this section the *single-transfer* model. It is possible to generalize our approach to an arbitrary number of transfers, but we restrict our attention to a single transfer for two reasons. The main reason is that more than one transfer tends to deter most of the population from taking transit, with the exception of lower-income, transit-dependent riders. The time lost due to an inefficient sequence of transfers tends to fall disproportionately on these passengers who cannot afford alternatives, so we focus on allowing only one transfer in the interest of equity. In focusing on only a single transfer, we follow the example of papers such as Baaq and Mahmassani (1995) and Borndörfer and Karbstein (2012).

Problem (1) can be extended to incorporate single transfers by adding variables $z_{(u,v),w} \in [0, 1]$ for $(u, v) \in \mathcal{D}$ and $w \in \mathcal{T}_{u,v}$, where $\mathcal{T}_{u,v}$ is defined as a set of stops that could serve as viable transfer stops between stops u and v ; these variables denote the fraction of commuters for (u, v) who make a transfer through intermediate station $w \in \mathcal{T}_{u,v}$. We further discuss $\mathcal{T}_{u,v}$ at the end of this section; for now, it suffices to say that $\mathcal{T}_{u,v}$ can be precomputed outside of the master problem. With the addition of these new variables, constraint (1b) is replaced with the following:

$$\theta_{u,v} \leq \sum_{\substack{\ell \in \mathcal{L}: \\ u \in \text{stops}(\ell), \\ v \in \text{stops}(\ell)}} \sum_{f \in \mathcal{F}} \gamma_f x_{\ell,f} + \sum_{w \in \mathcal{T}_{u,v}} z_{(u,v),w} \quad \forall (u, v) \in \mathcal{D}, \quad (7a)$$

$$z_{(u,v),w} \leq \sum_{f \in \mathcal{F}} \sum_{\substack{\ell \in \mathcal{L}: \\ u \in \text{stops}(\ell), \\ v \notin \text{stops}(\ell), \\ w \in \text{stops}(\ell)}} \lambda_f x_{\ell,f} \quad \forall (u, v) \in \mathcal{D}, \forall w \in \mathcal{T}_{u,v}; \quad (7b)$$

$$z_{(u,v),w} \leq \sum_{f \in \mathcal{F}} \sum_{\substack{\ell \in \mathcal{L}: \\ u \notin \text{stops}(\ell), \\ v \in \text{stops}(\ell), \\ w \in \text{stops}(\ell)}} \lambda_f x_{\ell,f} \quad \forall (u, v) \in \mathcal{D}, \forall w \in \mathcal{T}_{u,v}. \quad (7c)$$

The first summation in constraint (7a) is as before and represents the direct-route commute options; the second summation represents the single-transfer commute options. Although the \mathbf{z} variables might appear quite numerous at first glance, the set $\mathcal{T}_{u,v}$ may be substantially smaller than the total number of stops for a variety of reasons. For example, there may be only a limited number of stops that are large enough to sustain multiple higher-frequency transit lines, and of those stops, some may be too out-of-the-way to expect commuters to willingly use them as a mid-commute transfer. Constraints (7b) and (7c) together indicate that the ridership of a particular (u, v) commute via a transfer through w would depend on the minimum service level between the two stages of the commute. The λ_f coefficients are intended to model the fact that commuters will be less likely to make transfers at lower frequency levels, and the original formulation (2) is recovered when $\lambda_f = 0$ for all $f \in \mathcal{F}$. We will refer to the single-transfer master problem with constraints (7) as MP2(\mathcal{L}).

The dual for the new master problem, with dual variables $\boldsymbol{\pi}^{(1)}$ and $\boldsymbol{\pi}^{(2)}$ corresponding to constraints (7b) and (7c), is as follows:

$$\begin{aligned} \text{D2}(\mathcal{L}) = \min_{\substack{\mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{s}, \\ \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(2)}}} & Bq + \sum_{\ell \in \mathcal{L}} r_\ell + \sum_{(u,v) \in \mathcal{D}} s_{u,v} \quad (8a) \\ \text{s.t.} & \sum_{\substack{(u,v) \in \mathcal{D}: \\ u \in \text{stops}(\ell), \\ v \in \text{stops}(\ell)}} \gamma_f p_{u,v} + \sum_{\substack{(u,v) \in \mathcal{D}: \\ u \in \text{stops}(\ell), \\ v \notin \text{stops}(\ell)}} \sum_{w \in \mathcal{T}_{u,v} \cap \text{stops}(\ell)} \lambda_f \pi_{(u,v),w}^{(1)} \\ & + \sum_{\substack{(u,v) \in \mathcal{D}: \\ u \notin \text{stops}(\ell), \\ v \in \text{stops}(\ell)}} \sum_{w \in \mathcal{T}_{u,v} \cap \text{stops}(\ell)} \lambda_f \pi_{(u,v),w}^{(2)} \geq c_\ell \rho_f q + r_\ell \end{aligned}$$

$$\forall \ell \in \mathcal{L}, \forall f \in \mathcal{F}; \quad (8b)$$

$$p_{u,v} + s_{u,v} = d_{u,v} \quad \forall (u,v) \in \mathcal{D}, \quad (8c)$$

$$-p_{u,v} + \pi_{(u,v),w}^{(1)} + \pi_{(u,v),w}^{(2)} = 0 \quad \forall (u,v) \in \mathcal{D}, \forall w \in \mathcal{T}_{u,v}; \quad (8d)$$

$$\mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{s}, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(2)} \geq 0 \quad (8e)$$

As the only source of infeasibility in the dual comes from constraint (8b), we have a natural extension of the condition for the direct-route model in equation (4) that now accounts for single-transfer commuters. A new transit line $\ell \in \mathcal{L} \setminus \bar{\mathcal{L}}$ may be added to $\bar{\mathcal{L}}$ if the following condition is met:

$$c_\ell \rho_f \hat{q} < \sum_{\substack{(u,v) \in \mathcal{D}: \\ u \in \text{stops}(\ell), \\ v \in \text{stops}(\ell)}} \gamma_f p_{u,v} + \sum_{\substack{(u,v) \in \mathcal{D}: \\ u \in \text{stops}(\ell), \\ v \notin \text{stops}(\ell)}} \sum_{w \in \mathcal{T}_{u,v} \cap \text{stops}(\ell)} \lambda_f \pi_{(u,v),w}^{(1)} + \sum_{\substack{(u,v) \in \mathcal{D}: \\ u \notin \text{stops}(\ell), \\ v \in \text{stops}(\ell)}} \sum_{w \in \mathcal{T}_{u,v} \cap \text{stops}(\ell)} \lambda_f \pi_{(u,v),w}^{(2)}. \quad (9)$$

The dual variables $p_{u,v}$, $\pi_{(u,v),w}^{(1)}$, and $\pi_{(u,v),w}^{(2)}$ all correspond to the ridership that stands to be gained by servicing commute (u,v) : either by servicing the commuters directly, or indirectly via a transfer through w .

Modifying the subproblem to accommodate the new condition (9) is straightforward, but requires more variables to model the various cases of servicing demand. For the first case where the commutes are connected directly, we can use the same \mathbf{g} variables as before. For the second case, where the commute is serviced indirectly by transferring to or from another line, we introduce the variables $\mathbf{g}^{(1)} \in \{0, 1\}^{\mathcal{D}}$ and $\mathbf{g}^{(2)} \in \{0, 1\}^{\mathcal{D}}$, and add the following constraints for all commutes (u,v) :

$$g_{u,v}^{(1)} \leq \sum_{v' \in \text{In}(u)} h_{v',u}, \quad (10a)$$

$$g_{u,v}^{(1)} \leq \sum_{w \in \mathcal{T}_{u,v}} \sum_{v' \in \text{In}(w)} h_{v',w}, \quad (10b)$$

$$g_{u,v}^{(1)} \leq 1 - \sum_{u' \in \text{In}(v)} h_{u',v}, \quad (10c)$$

$$g_{u,v}^{(2)} \leq \sum_{u' \in \text{In}(v)} h_{u',v}, \quad (10d)$$

$$g_{u,v}^{(2)} \leq \sum_{w \in \mathcal{T}_{u,v}} \sum_{v' \in \text{In}(w)} h_{v',w}, \quad (10e)$$

$$g_{u,v}^{(2)} \leq 1 - \sum_{v' \in \text{In}(u)} h_{v',u}. \quad (10f)$$

Finally, we modify the subproblem objective as follows:

$$\max_{\mathbf{h}, \mathbf{g}} \sum_{(u,v) \in \mathcal{D}} \left(\gamma_f \bar{p}_{u,v} g_{u,v} + \sum_{w \in \mathcal{T}_{u,v}} \lambda_f \left(\pi_{(u,v),w}^{(1)} g_{u,v}^{(1)} + \pi_{(u,v),w}^{(2)} g_{u,v}^{(2)} \right) \right) - \sum_{(u,v) \in \mathcal{E}} \rho_f \bar{q} \delta_{u,v} h_{u,v}. \quad (11)$$

The coefficients γ_f and λ_f present a model of commuter behavior that is admittedly a simplification, as it separates dependence on service frequency from other factors such as in-vehicle travel time (which will be discussed in Section 3.3) and any idiosyncratic features of individual lines. However, it is a useful simplification that allows for consideration of service frequency while generating new transit lines. Further dependence of these coefficients on individual lines can be incorporated at the expense of substantial additional complexity in the subproblem. An alternative heuristic approach might be to use the γ_f and λ_f coefficients to generate new lines, but use a more detailed model of commuter behavior when selecting the set of lines to operate in the master problem.

3.3. Considering travel times

The task of computing travel times and the resulting route choices directly within the line-generation subproblem is a complex one. Prior works such as Baaaj and Mahmassani (1995) and Borndörfer et al. (2007) have avoided this complexity by alternating between route generation and assignment models. Although Borndörfer et al. (2007) present an exact formulation, they allow for arbitrarily long travel times and ignore transfers. This is not entirely realistic, since if the commuters' only options are routes of long travel time and many transfers, they will likely leave the transit system for other alternatives. By contrast, the assignment model of Baaaj and Mahmassani (1995) is detailed, but the route generation is heuristic.

We now turn to the problem of travel time in our own framework, and first address commutes that are confined to a single line. Recalling that the direct-route model assumes that commuters are willing to take transit as long as there is a route connecting its origin and destination, this implies that a Hamiltonian path through all of the stops would adequately service all demand, so long as it is feasible for the budget constraint. However, such a path will likely be inefficient for many commuters, particularly for commuters whose origins and destinations lie close to opposing terminals. Such commuters may likely seek alternative transit options. In our model, we will assume that commuters for commute (u, v) where u and v are on the same line, will only take transit if they are able to follow some sequence of edges $\omega = \{(u, w_2), (w_2, w_3), \dots, (w_k, v)\}$ that satisfies the following condition:

$$\sum_{(w_i, w_j) \in \omega} \delta_{w_i, w_j} \leq \Gamma \delta_{u, v}, \quad (12)$$

for some $\Gamma > 1$. Namely, commuters will only take transit if the travel time experienced on the transit network is not much greater than the direct travel time between origin and destination.

Incorporating travel time restrictions in the master problem is straightforward. In the direct-route model, we could modify constraint (2b) so that the right-hand-side omits any $x_{\ell, f}$ terms

that correspond to lines where the travel time from origin u to destination v on line ℓ is excessive, and the necessary modifications to the single-transfer model are similar. However, the subproblem poses a greater challenge. The most natural approach is to ensure that ridership is only accrued along paths that are sufficiently short using cutting planes. At every intermediate solution in the branch-and-bound tree, we check whether the transit line contains some path $\omega = \{(w_1, w_2), (w_2, w_3), \dots, (w_{k-1}, w_k)\}$ with length exceeding $\Gamma\delta_{w_1, w_k}$. If so, we could introduce the following constraint:

$$g_{w_1, w_k} \leq \sum_{i=1}^{k-1} (1 - h_{w_i, w_{i+1}}). \quad (13)$$

Although the approach (13) achieves the goal of counting only those commuters whose travel times are not excessive, this quickly becomes intractable even on small networks. The key difficulty is that many paths may exist that connect each pair of stops, requiring the addition of many weak cuts.

We propose a solution that achieves tractability by enforcing stricter constraints on the transit lines produced by the subproblem. Instead of using condition (12) only to calculate the reduced cost of the line, we enforce the condition for *every* pair of stops on the line. Although this may be a little excessive, it seems to hold in practice; on the MBTA's key network, which are defined by the agency as the high-ridership and high-frequency lines, we found that 97.8% of the stop pairs within the lines satisfied condition (12) for $\Gamma = 1.5$, and 99.8% of the stop pairs satisfied it for $\Gamma = 2.0$.

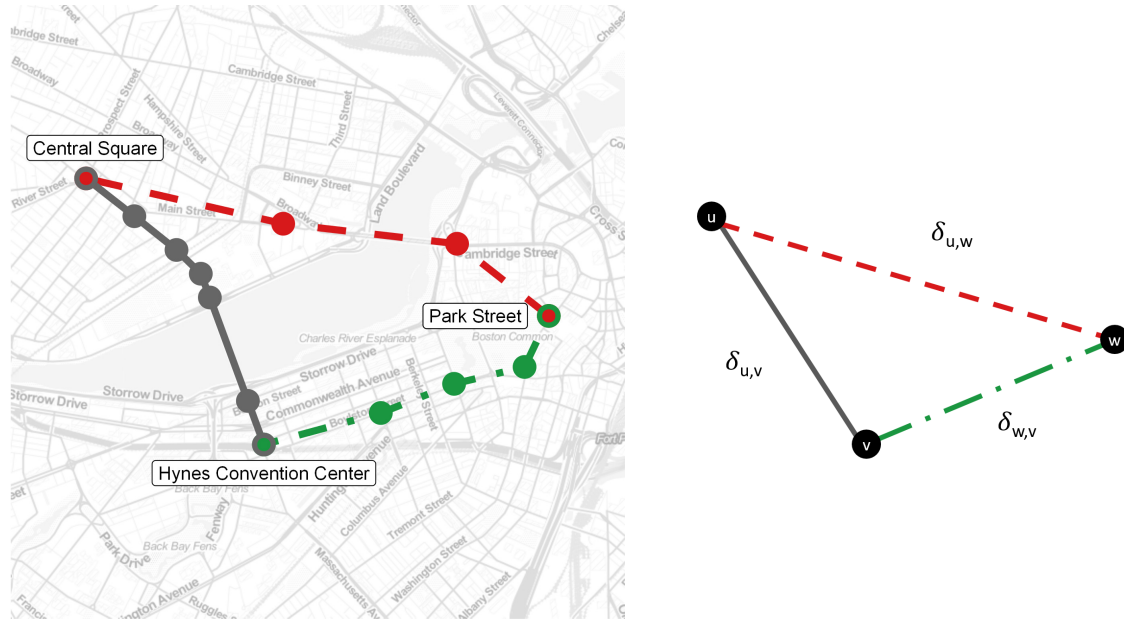
Our approach is implemented as follows. At intermediate solutions in the branch-and-bound tree, we check whether the transit line contains some path $\omega = \{(w_1, w_2), \dots, (w_{k-1}, w_k)\}$ with end-to-end travel time exceeding $\Gamma\delta_{w_1, w_k}$. If such a ω is found, we introduce the following constraint:

$$\sum_{i=1}^{k-1} h_{w_i, w_{i+1}} \leq k - 2. \quad (14)$$

Constraint (14) suffers from the same issue as (13) where cutting a single path at a time will be intractable on large networks, although it can be strengthened by using a tournament constraint. However, if ω is also the shortest path on all of the stops $\sigma = \{w_1, \dots, w_k\}$, we instead use the following, stronger constraint:

$$\sum_{w \in \sigma} \sum_{w' \in \text{Out}(w)} h_{w, w'} \leq k - 2, \quad (15)$$

which cuts not just the path ω corresponding to the current solution, but also all other paths connecting all of the stops in σ . Constraint (15) is valid because if the shortest path connecting

Figure 1 An illustration of excessive travel times due to transfers.

(a) A Boston example. The transfer at Park Street makes the travel time from Central Square to Hynes Convention Center significantly longer than the direct path, which is unappealing for commuters.

(b) A cartoon of the Central Square (u) to Hynes Convention Center (v) via Park Street (w) example.

all stops σ does not satisfy condition (12), then no other paths connecting all stops σ will satisfy that condition either. Checking whether ω is the shortest path connecting all of the stops σ could potentially be costly; in the worst case, such an operation would need to be performed on all subpaths of all visited solutions in the branch-and-bound tree. Rather than verifying this exactly, we use a fast nearest-insertion heuristic to compute the shortest path on σ and compare the cost of that path to that of ω . The heuristic begins with the first node in the path, considers each of the remaining nodes, and inserts whichever node is closest at the end of the path; this is repeated until no nodes are remaining, and can also be repeated for different starting nodes in the path.

We now turn to travel time for single-transfer commuters who are forced to make a transfer through less direct routes. An illustrative example from the Boston-area subway and bus network is shown in Figure 1. To get from Central Square to Hynes Convention Center via the subway, a commuter must take the Red Line (dashed red) from Central Square to Park Street before transferring to the Green Line (dot-dashed green). This is a significant increase in travel time relative to the direct bus route (solid gray), making the subway unappealing to commuters.

To limit the travel time experienced by single-transfer commuters, we precompute transfer stops for each commute that would not take commuters too far out of their way. In particular, we define

a set of transfer stops for commute (u, v) as follows:

$$\mathcal{T}_{u,v} := \{w \in \mathcal{N} \mid \delta_{u,w} + \delta_{w,v} \leq \Lambda \delta_{u,v}\}, \quad (16)$$

where Λ is a constant greater than one. In Figure 1b, the displayed transfer stop w would be disqualified from being in $\mathcal{T}_{u,v}$ based on definition (16). The restriction of the transfer stops, combined with the travel time restriction of (12), has the effect of preventing a commuter's total travel time for commute (u, v) from being more than $\Gamma\Lambda$ times the direct travel time $\delta_{u,v}$.

Restricting the set of transfers for a commute (u, v) to only occur through a stop $w \in \mathcal{T}_{u,v}$ keeps travel times on the network reasonable and prevents generation of excessively long commuter routes. Furthermore, this modeling approach makes the underlying network of edges significantly sparser, [reducing the size of the subproblem](#).

3.4. Speeding up the subproblem

Much of the computational expense in the subproblem comes from the addition of the subtour elimination constraints (6g) and the travel time constraints (15), particularly in dense graphs. We propose a fast preprocessing step that significantly reduces the running time of the subproblem. The intent is to preprocess the edge set \mathcal{E} so that cycles will not appear in the subproblem graph, thus removing the need for subtour elimination constraints. As an auxiliary effect, the preprocessing tends to limit travel times on the network, so fewer travel time constraints will be needed as well. For the remainder of the paper, we will call the subproblem on the full edge set with subtour elimination constraints the *full* subproblem, and the subproblem on the preprocessed edge set the *preprocessed* subproblem.

The preprocessing is done by choosing a particular geographic direction and removing all edges that are not aligned with that geographic direction, within a certain range of tolerance. For a direction \mathbf{z} and a parameter $\Delta \in [0, 1]$, the preprocessed edge set $\mathcal{E}(\mathbf{z}, \Delta) \subseteq \mathcal{E}$ is defined as follows:

$$\mathcal{E}(\mathbf{z}, \Delta) = \left\{ (u, v) \in \mathcal{E} \mid \frac{\text{vec}(u, v) \cdot \mathbf{z}}{\|\text{vec}(u, v)\| \|\mathbf{z}\|} \geq \Delta \right\}, \quad (17)$$

where we use the notation $\text{vec}(u, v)$ to indicate the vector pointing from stop u to stop v . For $\Delta \geq 0$, cycles will not appear in the graph, and larger values of Δ will tend to make the transit lines more oriented along the direction \mathbf{z} . Orienting the transit line along the direction \mathbf{z} will also keep travel times short, which will reduce the number of travel time constraints that need to be added. In the computations, we must choose a set of directions \mathbf{z} that is exhaustive enough to provide a variety of high-quality solutions; such a set could for example comprise the cardinal and intercardinal directions. We must also set Δ to a value that eliminates cycles while also not overly restricting the edge set so as to eliminate the optimal solution.

Algorithm 1 is modified to accommodate the edge-preprocessing as follows. The subproblem is solved over an exhaustive set of directions, producing a set of different transit lines. If the subproblem terminates with positive objective value for any direction \mathbf{z} , then the most profitable of these lines are added to $\bar{\mathcal{L}}$. If all versions of the subproblem terminate with objective value zero, then there are no profitable lines that can be added, and the algorithm terminates with the optimal network design for the linear relaxation (2). This process is described in full detail in Algorithm 2. The loop in line 2 is easily parallelizable over all directions, allowing for further computational speedup.

Algorithm 2 Column generation algorithm for solving TNDP (1), with edge preprocessing

Require: Initial set of transit lines $\bar{\mathcal{L}}$, tolerance $\epsilon > 0$, line directedness parameter $\Delta \in [0, 1]$

- 1: Solve $\text{MP}(\bar{\mathcal{L}})$, get primal solution $(\bar{\mathbf{x}}, \bar{\boldsymbol{\theta}})$ and dual solution $(\bar{\mathbf{p}}, \bar{q}, \bar{\mathbf{s}})$
 - 2: **for all** directions \mathbf{z} **do**
 - 3: $J_{\mathbf{z}} \leftarrow \text{SP}(\bar{\mathbf{p}}, \bar{q}; G(\mathcal{V}, \mathcal{E}(\mathbf{z}, \Delta)))$
 - 4: $\ell_{\mathbf{z}} \leftarrow$ transit line solution to $\text{SP}(\bar{\mathbf{p}}, \bar{q}; G(\mathcal{V}, \mathcal{E}(\mathbf{z}, \Delta)))$
 - 5: **while** $\max_{\mathbf{z}} J_{\mathbf{z}} > \epsilon$ **do**
 - 6: $\bar{\mathcal{L}} \leftarrow \bar{\mathcal{L}} \cup \ell_{\arg \max_{\mathbf{z}} \pi_{\mathbf{z}}}$
 - 7: Solve $\text{MP}(\bar{\mathcal{L}})$, get primal solution $(\bar{\mathbf{x}}, \bar{\boldsymbol{\theta}})$ and dual solution $(\bar{\mathbf{p}}, \bar{q}, \bar{\mathbf{s}})$
 - 8: **for all** directions \mathbf{z} **do**
 - 9: $J_{\mathbf{z}} \leftarrow \text{SP}(\bar{\mathbf{p}}, \bar{q}; G(\mathcal{V}, \mathcal{E}(\mathbf{z}, \Delta)))$
 - 10: $\ell_{\mathbf{z}} \leftarrow$ transit line solution to $\text{SP}(\bar{\mathbf{p}}, \bar{q}; G(\mathcal{V}, \mathcal{E}(\mathbf{z}, \Delta)))$
 - 11: Solve TNDP($\bar{\mathcal{L}}$), get integral solution $(\mathbf{x}^*, \boldsymbol{\theta}^*)$
 - 12: **return** \mathbf{x}^*
-

In the next section, we demonstrate the benefit and tractability of Algorithms 1 and 2.

4. Computational Results

In this section, we evaluate the performance of our column generation algorithm on synthetic and real-world data. All methods were implemented using the `Julia` language (Bezanson et al. 2014) and the optimization package `JuMP` (Lubin and Dunning 2015) using the Gurobi solver v8.1 (Gurobi Optimization, Inc. 2016). Computational experiments were run on a laptop with an Intel i7-6500U processor and 16GB of RAM.

In Section 4.1, we show that Algorithm 1 produces intuitive results for a small synthetic grid network. In Section 4.2, we move to a large-scale example in Boston with hundreds of stops. We compare the performance of the preprocessed subproblem to the full subproblem, and show that

preprocessing improves the solution time while also preserving competitive objective values. We also demonstrate that Algorithm 2 [can be solved efficiently](#) for our Boston example and examine the solution bus networks.

4.1. A small synthetic network

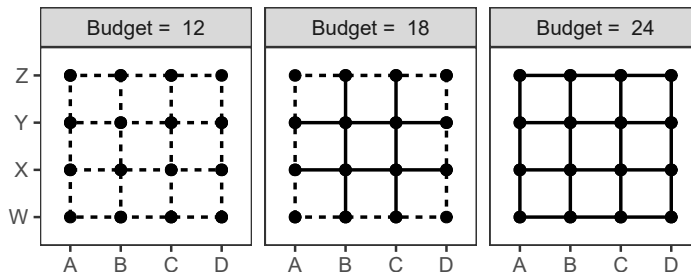
Our first computational experiment was inspired by the Houston network re-design (Binkovitz 2016), which dramatically simplified its bus routing from a hub-and-spoke framework to a pure grid network. The appeal of a grid network is that due to its relative simplicity, a commuter can get from any origin to any destination on the grid with at most one transfer. Our computational experiments on the synthetic dataset aim to illustrate the intuitive appeal of the grid and to illuminate the differences between the various models of Section 3.

We created a toy sixteen-stop network with coordinates on the integer lattice $\{1, 2, 3, 4\}^2$, representing the intersections of cross streets that we label A-D (directed vertically) and W-Z (directed horizontally). To avoid degeneracy and thereby improve computation time, a small amount of noise uniformly distributed between 0 and 0.01 was added to the coordinates of each stop. Edges were allowed between every stop and its horizontal, vertical, and diagonal immediate neighbors. Demand between every pair of stops (u, v) was set to $d_{u,v} = 100$, for a total of $\sum_{u,v} d_{u,v} = 24,000$. The budget was set to $B = 12, 18, \text{ and } 24$, the last of which is exactly the budget needed to sustain a high-frequency grid network. We allowed for two frequency levels, high and low, [where the low level operates at half the frequency of the high level \(for example, every 10 minutes versus every 20 minutes\)](#). The cost coefficients ρ_f and frequency coefficients γ_f were set that so that the low-frequency lines were half the cost of the high-frequency lines but could get three quarters of the direct ridership relative to the high-frequency lines; this was intended to model diminishing marginal returns, as well as the fact that some of the population might be inelastic in their use of transit, accommodating the lower frequency rather than switching to alternatives. The λ_f coefficients were set so that [only one quarter](#) of riders would be willing to make transfers between low-frequency lines, but that all riders are willing to make transfers between high-frequency lines. Concretely, the vectors of parameters were set at $\boldsymbol{\rho} = [1.0, 0.5]$, $\boldsymbol{\gamma} = [1.0, 0.75]$, and $\boldsymbol{\lambda} = [1.0, 0.25]$. [Note that these are toy values that are only meant to be illustrative; in practical application, surveys would need to be conducted to estimate actual values.](#) We set $\Gamma = \Lambda = 1.5$, meaning that commuters along a single line experience travel times no longer than 50% more than the direct travel time between their origin and destination, and commuters who transfer experience travel times no more than 125% more than the direct travel time between their origin and destination.

On this small network, we were able to solve the full subproblem with subtour elimination constraints (6g), using Algorithm 1. In each iteration of the column generation algorithm, we also

added all possible subsequences of consecutive stops from the lines that were generated, which produced greater variety among the columns while also reducing the number of iterations required to converge.

Figure 2 Synthetic bus networks generated on a grid network at varying budgets.



Note. Solid lines represent high-frequency bus lines; dashed lines represent low-frequency bus lines.

Table 2 Running time and objective values on a toy network.

Budget	Iterations	Run. Time (min)	Ridership
12	12	1.8	16,100
18	21	2.9	19,900
24	9	0.7	24,000

The networks produced at each budget level are shown in Figure 2, and the running times are shown in Table 2. Each of the problem instances converged quickly, with the longest running time still under three minutes. Figure 2 illustrates the intuitive appeal of a grid network; at the highest budget of $B = 24$, the high-frequency grid can serve all of the demand with reasonable travel times and at most one transfer. The core of the high-frequency grid is maintained even at budget $B = 18$, with the perimeter of the grid split into two low-frequency bus lines that traverse streets A to Z and W to D, which service many direct passengers while also allowing for transfers to the high-frequency grid. Finally, at budget $B = 12$, coverage on the grid is maintained in order to service the spread-out demand, albeit all at low frequency. This reflects the relative inelasticity of the travel demand according to the λ_f coefficients we set; a choice of λ_f where more commuters abandon transit at low frequencies would lead to transit networks of lower coverage but higher service frequencies.

These computational results are not surprising; from a grid-type network of stops, one would expect an efficient solution to have a grid structure. However, [they provide](#) a useful validation that our algorithm produces intuitive results for a simple test case, which is a key step towards implementability.

4.2. A large-scale case study from Boston

Our last set of computational experiments focus on the bus network in the greater Boston area, operated by the Massachusetts Bay Transportation Authority (MBTA). We show that our method is practical for a real-world network of hundreds of stops, and that real benefits can be realized from taking an optimization-based approach to transit network design.

We obtained bus stop and line information from the General Transit Feed Specification (GTFS) provided by the Massachusetts Bay Transit Authority (2014). Although our approach might be applied to a generic multimodal transit network, we use buses for our case study because of the flexibility of re-designing bus networks relative to rail or other more resource-intensive modes of transportation. We focused on stops in the central business district as well as densely-populated surrounding areas such as Cambridge and Somerville. The data contained many clusters of duplicate bus stops; in order to reduce the problem size while retaining detailed information about the network, we rounded all coordinates to the nearest 0.005 in latitude and longitude, leading to a rough grid of 233 stops spaced about 0.4 miles apart. We obtained driving distances between stops from the Google Distance Matrix API. In considering new edges to add to the network, we considered edges of length 1.0 miles or less, which gave us 2,042 edges to consider. We restricted ourselves to consider bus lines containing at most 20 stops. The restricted master problem was seeded with the 1, 15, 22, 23, 28, 39, 57, 66, and 77 bus lines from the MBTA key bus network, which generally covered our geographic region of interest. The lines in this initial set were altered slightly due to the rounding of the stop coordinates, but still followed the general trajectory of the original bus lines.

We also obtained automated passenger count data for 2017 from the MBTA, which detailed the average number of passengers boarding and alighting buses at each stop for each hour. Using techniques from Bertsimas and Yan (2018), we estimated the hourly origin-destination demand matrices, and aggregated them to produce a daily demand matrix. For computational tractability, we ignored origin-destination pairs that saw fewer than 50 passengers over the course of the entire day. We also removed a few origin-destination pairs where the origin and destination were disconnected in the network. The resulting demand matrix had a total demand of 93,826 trips, with 597 origin-destination pairs seeing activity.

Since the counts provided by the MBTA are taken entirely from passengers who are already using MBTA services, we expect the existing network to be adequate, and were interested in seeing what the performance gains might be using a demand matrix that did not come from the MBTA. Furthermore, with significant concerns about how the rise of alternatives such as ride-sharing services might impact ridership for public transit (Bouton et al. 2015), it is useful to examine

the demand from alternative modes of transport to see how the bus network might better service those who have chosen those who did not elect to take the bus. Blue Bikes (formerly Hubway), a bike-sharing company in the Boston area, made its 2011-13 trip data publicly available through a data challenge (Metropolitan Area Planning Council 2013); by mapping each Blue Bikes dock to the closest MBTA stop, we obtained an alternative origin-destination matrix. For maximal impact, we focused on the trips from August 2013, which had the highest monthly trip volume. After eliminating disconnected origin-destination pairs as well as those that saw fewer than 50 passengers over the course of the month, we were left with a total demand of 50,621 trips, with 534 origin-destination pairs seeing activity.

Our decision variables included two settings of high versus low frequencies, which were motivated by the distinction that the MBTA draws between its key (high frequency) and non-key (low frequency) network. *Service is offered once every 30 minutes during peak hours on the non-key network, versus once every 10 minutes during peak hours on the key network; accordingly, the cost parameter ρ_f is set so that low frequencies cost about one-third relative to high frequencies.* We set the direct ridership parameter γ_f so that the low-frequency lines would see 45% of the direct ridership of high-frequency lines; like in Section 4.1, this was intended to illustrate some inelasticity in commuter transit choices. We set the transferring ridership parameter λ_f so that commuters would be fully willing to transfer between high-frequency lines, but unwilling to transfer between low-frequency lines. Concretely, the vectors of parameters were set at $\boldsymbol{\rho} = [1.0, 0.3]$, $\boldsymbol{\gamma} = [1.0, 0.45]$, and $\boldsymbol{\lambda} = [1.0, 0.0]$. As in the previous section, we set $\Gamma = \Lambda = 1.5$, unless otherwise specified. We do not claim that these are the right values for the Boston network; the intention is merely to set these parameters to reasonable values on which to test our approach. In a practical setting, surveys should be conducted to determine true parameter values.

4.2.1. Solving the subproblem at scale We first compare the performance of the preprocessed subproblem proposed in Section 3.4 to the full subproblem with subtour elimination constraints (6g) and illustrate how the various parameters may be set. We solved a single iteration of the direct-route subproblem on the MBTA data, both with and without preprocessing. The preprocessing was run on a range of parameter settings between $\Delta = 0.2$ (most restricted, fewest edges) to $\Delta = 0.0$ (least restrictive, most edges) over a range of four directions \mathbf{z} evenly spaced 45 degrees apart. Although a more granular set of directions does produce different subsets of the edge set, we did not see material difference in the final results with larger edge sets.

To show the effect of preprocessing on the subproblem, we also include results for $\Gamma = 1.4$ as well as $\Gamma = 1.5$, the lower Γ value meaning that travel times on the network are more constrained.

Since the preprocessed subproblem must be solved once for each \mathbf{z} , we warm-started each instance using a partial solution from the previous instance. *As the previous instance was solved using*

Table 3 Performance of preprocessed and full subproblems.

Γ	Model	Run. Time (s)	Objective
1.4	$\Delta = 0.2$; 4 instances	7	8,591
1.4	$\Delta = 0.1$; 4 instances	57	8,591
1.4	$\Delta = 0.0$; 4 instances	51	8,591
1.4	Full	411	8,591
1.5	$\Delta = 0.2$; 4 instances	5	8,591
1.5	$\Delta = 0.1$; 4 instances	7	9,150
1.5	$\Delta = 0.0$; 4 instances	12	10,028
1.5	Full	141	10,256

Higher values of Δ mean that more edges were removed from the graph.

a different value of \mathbf{z} , it was not necessarily feasible for the current instance; in order to create a feasible warm start, we chose whatever sub-path of the previously generated line was feasible, breaking ties by objective value if there were multiple such sub-paths. The full (no preprocessing) subproblem was warm-started with the solution from the preprocessed subproblem with $\Delta = 0.0$.

The running times and objective values for each model are shown in Table 3. The full subproblem is substantially harder to solve than the preprocessed subproblems, all of which terminate in an order of magnitude less time, with running times generally decreasing further as more edges are removed. In addition to these substantially lower running times, the preprocessed subproblems also achieve roughly comparable solution quality as the full subproblem, especially at the lowest $\Delta = 0.0$ setting; for example, at $\Gamma = 1.5$, the $\Delta = 0.0$ instances find a solution that is only 2.2% lower in objective value than the full subproblem. At $\Gamma = 1.4$, all models achieve identical objective values, reflecting the fact that removing edges may still preserve the optimal solution if travel times are more restricted.

These results illustrate the usefulness of preprocessing the edge set in the subproblem on large networks. A flexible enough Δ parameter keeps enough edges to find high-quality or even optimal solutions. The most flexible $\Delta = 0.0$ seems to be the best choice to give high-quality solutions; however, if shorter travel times are desired, a more restrictive Δ may still provide good results. In the experiments that follow, we solve only the pre-processed subproblems and set $\Delta = 0.0$; in this way, we accept a small decrease in solution quality over each iteration as a tradeoff for substantially increasing the speed at which columns are generated.

4.2.2. Designing the Boston network We now turn to the problem of designing a network for Boston based on our MBTA and Blue Bikes demand matrices.

For ease of plotting and interpreting the networks, we added constraints to the master problem that ensured that each edge chosen in the final network would be covered by at most one line; the subproblem was also modified accordingly. An exception was made for the lines belonged to

the original MBTA network, so that it would remain feasible. These constraints are likely more restrictive than what happens in practice, as real-world transit networks generally show some redundancy in order to provide more direct-route options and also share resources. However, these constraints might be easily modified to accommodate capacity restrictions on various edges in the network. A further constraint was motivated by the prevalence of relatively short-distance commutes in the Blue Bikes dataset, which led to the generation of many short bus lines that could involve as few as two edges. As a result, for the Blue Bikes dataset we imposed the additional constraint that bus lines should comprise at least seven edges.

The algorithm was capped at a maximum of 20 iterations; each iteration involved adding both a high-frequency and a low-frequency line to the restricted master problem, so that at the end 40 lines were generated in addition to the original lines from the key bus network.

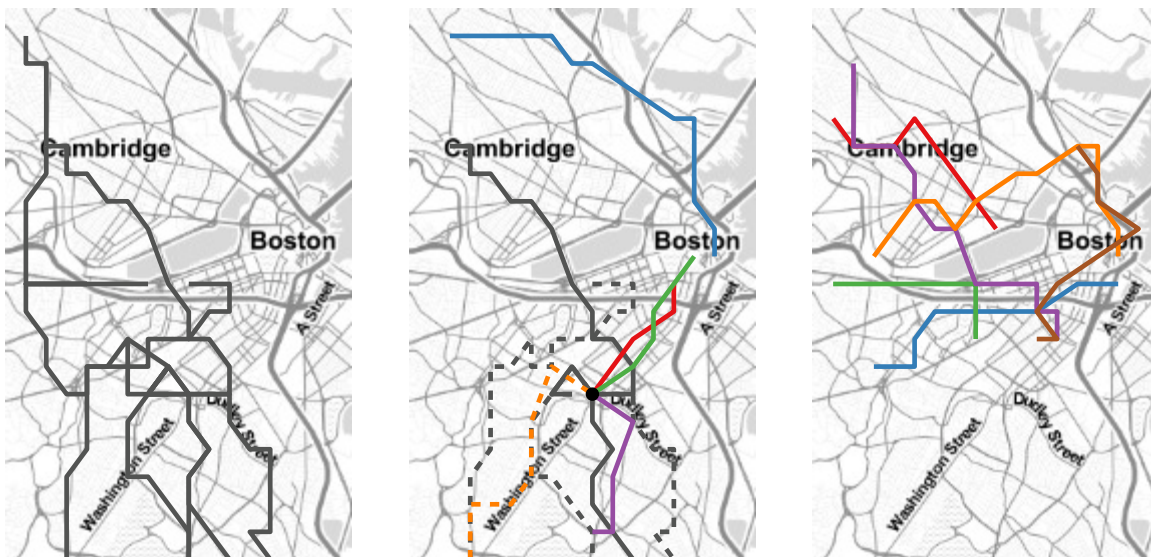
The results of Algorithm 2 over a range of budgets is shown in Table 4 for both the MBTA and Blue Bikes datasets. We began with a subset of the MBTA key network (“Original”) as the initial set of lines $\bar{\mathcal{L}}$ for the restricted master problem, before generating additional lines using Algorithm 2 (“New”). [The “ridership” column is computed using the objective values in the restricted master problem for both the “Original” and “New” networks.](#) On the demand matrix estimated from the MBTA data, our column generation procedure found networks that serviced 27 to 35% more ridership than the original network. The improvement was even more pronounced on the Blue Bikes demand. As expected, given that Blue Bikes users are opting for an alternate mode of transportation, the original network does not effectively service the demand, while the new network is able to achieve triple the ridership of the old.

In addition to enabling substantial ridership gains, Algorithm 2 [can also be solved efficiently](#). The running times for each of the budgets and demand scenarios are shown in Table 4, and range from a quarter to half of an hour. Since network design is an offline problem and cities rarely redesign their transit networks, this an appropriate level of computational effort to be practically useful, and could even scale to larger problem sizes.

Maps on the Boston network under various demand scenarios are shown in Figure 3. Figure 3a shows the original set of lines that begin the restricted master problem, which were taken from the MBTA key network. Figures 3b and 3b then show the lines that were generated and then chosen in the final networks for the MBTA and Blue Bikes demand data at, respectively, at a budget of $B = 40$. Lines from the original set are shown in gray, while generated lines are shown in color. Low frequency is indicated by dashed lines, and high frequency is indicated by solid lines.

From Figure 3b, we see that the new network stays somewhat close to the original network; for example, it preserves the 1 bus crossing the river from Cambridge into Boston at high frequency, and maintains other bus lines from the original network that cover the service region at a lower

Figure 3 (Color online) A subset of the key MBTA bus network, as well as bus networks produced by column generation on the MBTA and Blue Bikes demand matrices for $B = 40$.



(a) Original network used to start the restricted master problem
 (b) New network generated on MBTA demand
 (c) New network generated on Blue Bikes demand

Note. Lines from the existing MBTA network are shown in gray; generated lines are shown in color. Dashes indicate low-frequency lines. The light gray region between Cambridge and Boston is the Charles River.

frequency. The new lines generated comprise (i) a high-frequency hub-and-spoke geometry emanating from Dudley Square, which is marked with a black circle, and (ii) high-frequency coverage in the northeast corner of the map. In both cases, the generated lines unsurprisingly mirror existing MBTA service that were included in the demand data but not our original set of bus lines.

None of the exact lines from the original network appear in the new network in Figure 3c, although there are some similarities; for example, the purple and green lines in Figure 3c are variants of lines that do appear in the original network. Sections of the orange and brown lines in Figure 3c also cover similar areas as the red, blue, and green lines in Figure 3b. In contrast to the other maps, Figure 3c shows more support around the northern half of the map. Moreover, while the original network is largely oriented from north to south, Figure 3c shows that much of the network generated on the Blue Bikes demand is oriented roughly from east to west, suggesting that more resources could be devoted to supporting these cross-town trajectories.

5. Conclusion

In this work, we address the problem of designing lines for urban transit networks. In particular, we seek to maximize ridership on a transit network, accounting for the fact that passengers will

Table 4 Running time and objective values on the Boston network

Demand Matrix	Budget	Run. Time (min)	Ridership	
			Original	New
MBTA	30	22.1	33,683	42,648
MBTA	40	30.1	38,035	50,154
MBTA	50	33.4	40,154	54,102
Blue Bikes	30	14.4	8,921	26,509
Blue Bikes	40	17.6	9,332	28,726
Blue Bikes	50	19.6	9,396	32,719

choose to take transit if one of the possible routes was appealing in travel time and number of transfers. We discuss how to incorporate a variety of physical and operational constraints on lines, and present computational experiments to show our method’s tractability and usefulness. In the first experiment, we show that our method produces a simple grid network from a synthetic dataset, which mirrors real-life observations in cities such as Houston on the appeal of grid networks. In the second experiment, we propose a preprocessing step that greatly improves the tractability of our algorithm, and then demonstrate substantial ridership gains for a real bus network in Boston. These ideas present opportunities for transit authorities to perform holistic re-design of their transit networks in order to offer a service that is both cost-efficient and appealing to commuters.

Acknowledgments

The authors are thankful to collaborators from Mitsubishi Electric Corporation for productive discussions and to the reviewers for helpful comments that improved the manuscript. Research was partially supported by a grant from Mitsubishi Electric Corporation.

References

- Baaj, M Hadi, Hani S Mahmassani. 1995. Hybrid route generation heuristic algorithm for the design of transit networks. *Transportation Research Part C: Emerging Technologies* **3**(1) 31–50.
- Barnhart, Cynthia, Ellis L Johnson, George L Nemhauser, Martin WP Savelsbergh, Pamela H Vance. 1998. Branch-and-price: Column generation for solving huge integer programs. *Operations research* **46**(3) 316–329.
- Barra, Alexandre, Luis Carvalho, Nicolas Teypaz, Van-Dat Cung, Ronaldo Balassiano. 2007. Solving the transit network design problem with constraint programming. *11th World Conference in Transport Research-WCTR 2007*.
- Bertsimas, Dimitris, Julia Yan. 2018. From physical properties of transportation flows to demand estimation: An optimization approach. *Transportation Science* **52**(4) 1002–1011.
- Bezanson, Jeff, Alan Edelman, Stefan Karpinski, Viral B. Shah. 2014. Julia: A fresh approach to numerical computing. *arXiv preprint arXiv:1411.1607*.

- Binkovitz, Leah. 2016. A year after bus redesign, METRO Houston ridership is up. URL <https://kinder.rice.edu/2016/08/16/a-year-after-redesign-metro-ridership-is-up/>. Accessed: 2018-08-02.
- Borndörfer, Ralf, Martin Grötschel, Marc E Pfetsch. 2007. A column-generation approach to line planning in public transport. *Transportation Science* **41**(1) 123–132.
- Borndörfer, Ralf, Marika Karbstein. 2012. A direct connection approach to integrated line planning and passenger routing. *12th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 47—57.
- Bouton, Shannon, Stefan M Knupfer, Ivan Mihov, Steven Swartz. 2015. Urban mobility at a tipping point. *McKinsey and Company* .
- Bussieck, Michael R, Peter Kreuzer, Uwe T Zimmermann. 1997. Optimal lines for railway systems. *European Journal of Operational Research* **96**(1) 54–63.
- Cancela, Héctor, Antonio Mauttone, María E Urquhart. 2015. Mathematical programming formulations for transit network design. *Transportation Research Part B: Methodological* **77** 17–37.
- Ceder, Avishai, Nigel HM Wilson. 1986. Bus network design. *Transportation Research Part B: Methodological* **20**(4) 331–344.
- Cipriani, Ernesto, Stefano Gori, Marco Petrelli. 2012. Transit network design: A procedure and an application to a large urban area. *Transportation Research Part C: Emerging Technologies* **20**(1) 3–14.
- Farahani, Reza Zanjirani, Elnaz Miandoabchi, Wai Yuen Szeto, Hannaneh Rashidi. 2013. A review of urban transportation network design problems. *European Journal of Operational Research* **229**(2) 281–302.
- Gattermann, Philine, Jonas Harbering, Anita Schöbel. 2017. Line pool generation. *Public Transport* **9**(1-2) 7–32.
- Goossens, Jan-Willem, Stan Van Hoesel, Leo Kroon. 2004. A branch-and-cut approach for solving railway line-planning problems. *Transportation Science* **38**(3) 379–393.
- Guan, JF, Hai Yang, Sumedha Chandana Wirasinghe. 2006. Simultaneous optimization of transit line configuration and passenger line assignment. *Transportation Research Part B: Methodological* **40**(10) 885–902.
- Guihaire, Valérie, Jin-Kao Hao. 2008. Transit network design and scheduling: A global review. *Transportation Research Part A: Policy and Practice* **42**(10) 1251–1273.
- Gurobi Optimization, Inc. 2016. Gurobi optimizer reference manual. URL <http://www.gurobi.com>.
- Jin, Jian Gang, Kwong Meng Teo, Amedeo R Odoni. 2016. Optimizing bus bridging services in response to disruptions of urban transit rail networks. *Transportation Science* **50**(3) 790–804.
- Laughlin, Jason. 2017. SEPTA looks to Texas for ideas for bus route redesign. *The Inquirer* URL <http://www.philly.com/philly/business/transportation/septa-overhaul-bus-service-houston-model-20170911.html>. Accessed: 2018-08-02.

- Laughlin, Jason. 2018. SEPTA lost millions of bus trips last year, feds say. *The Inquirer* URL <http://www.philly.com/philly/business/transportation/septa-lost-riders-bus-train-trolley-transit-fta-20180323.html>. Accessed: 2018-08-02.
- Lownes, Nicholas E, Randy B Machemehl. 2010. Exact and heuristic methods for public transit circulator design. *Transportation Research Part B: Methodological* **44**(2) 309–318.
- Lubin, Miles, Iain Dunning. 2015. Computing in operations research using julia. *INFORMS Journal on Computing* **27**(2) 238–248.
- Magnanti, Thomas L, Richard T Wong. 1984. Network design and transportation planning: Models and algorithms. *Transportation science* **18**(1) 1–55.
- Mandl, Christoph E. 1980. Evaluation and optimization of urban public transportation networks. *European Journal of Operational Research* **5**(6) 396–404.
- Marín, Ángel G, Patricia Jaramillo. 2009. Urban rapid transit network design: accelerated benders decomposition. *Annals of Operations Research* **169**(1) 35–53.
- Massachusetts Bay Transit Authority. 2014. GTFS developers. URL <https://www.mbta.com/developers/gtfs>.
- Massachusetts Bay Transportation Authority. 2017. MBTA service delivery policy. URL <https://cdn.mbta.com/sites/default/files/fmcb-meeting-docs/reports-policies/2017-mbta-service-delivery-policy.pdf>. Accessed: 2018-08-02.
- Metropolitan Area Planning Council. 2013. Hubway Data Visualization Challenge. URL <http://hubwaydatachallenge.org/>. Accessed: 2018-08-02.
- Murray, Alan T. 2003. A coverage model for improving public transit system accessibility and expanding access. *Annals of Operations Research* **123**(1-4) 143–156.
- Nelson, Laura J. 2018. Ridership on Metro fell to the lowest level in more than a decade last year. *Los Angeles Times* URL <http://www.latimes.com/local/lanow/la-me-metro-ridership-20180124-story.html>. Accessed: 2018-08-02.
- Pinelli, Fabio, Rahul Nair, Francesco Calabrese, Michele Berlingiero, Giusy Di Lorenzo, Marco Luca Sbodio. 2016. Data-driven transit network design from mobile phone trajectories. *IEEE Transactions on Intelligent Transportation Systems* **17**(6) 1724–1733.
- Schlinkmann, Mark. 2018. Bus routes around St. Louis could see overhaul as Metro seeks public feedback. *St. Louis Post-Dispatch* URL http://www.stltoday.com/news/traffic/along-for-the-ride/bus-routes-around-st-louis-could-see-overhaul-as-metro/article_fb1131c6-5484-592f-879c-f8ccd329322a.html. Accessed: 2018-08-02.
- Schöbel, Anita, Susanne Scholl. 2006. Line planning with minimal traveling time. *5th Workshop on Algorithmic Methods and Models for Optimization of Railways (ATMOS'05)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

- Siddiqui, Faiz. 2018. Ridership on Metro fell to the lowest level in more than a decade last year. *Washington Post* URL https://www.washingtonpost.com/local/trafficandcommuting/falling-transit-ridership-poses-an-emergency-for-cities-experts-fear/2018/03/20/ffb67c28-2865-11e8-874b-d517e912f125_story.html. Accessed: 2018-08-02.
- Stolte, Elise. 2018. “Almost every route is new”: Officials call for help on redesigned bus network. *Edmonton Journal* URL <http://edmontonjournal.com/news/local-news/almost-every-route-is-new-officials-call-for-help-on-redesigned-bus-network>. Accessed: 2018-08-02.
- United Nations. 2014. World urbanization prospects. URL <https://esa.un.org/unpd/wup/Publications/Files/WUP2014-Report.pdf>. Accessed: 2018-08-02.
- Vaccaro, Adam. 2018. With new services and big plans, MBTA buses are getting some love. *Boston Globe* URL <https://www.bostonglobe.com/metro/2018/04/01/with-new-services-and-big-plans-mbta-buses-are-getting-some-love/BqbPyAcerHcriBJ011PByM/story.html>. Accessed: 2018-08-02.
- Walteros, Jose L, Andrés L Medaglia, Germán Riaño. 2013. Hybrid algorithm for route design on bus rapid transit systems. *Transportation Science* **49**(1) 66–84.
- Wan, Quentin K, Hong K Lo. 2003. A mixed integer formulation for multiple-route transit network design. *Journal of Mathematical Modelling and Algorithms* **2**(4) 299–308.
- Yu, Bin, Zhong-Zhen Yang, Peng-Huan Jin, Shan-Hua Wu, Bao-Zhen Yao. 2012. Transit route network design-maximizing direct and transfer demand density. *Transportation Research Part C: Emerging Technologies* **22** 58–75.
- Zhao, Fang, Ike Ubaka. 2004. Transit network optimization-minimizing transfers and optimizing route directness. *Journal of Public Transportation* **7**(1) 4.
- Zhao, Fang, Xiaogang Zeng. 2006. Simulated annealing-genetic algorithm for transit network optimization. *Journal of Computing in Civil Engineering* **20**(1) 57–68.