Fair network design problem: an application to EV charging station capacity expansion

Nagisa Sugishita[1], Ismail Sevim[1], Margarida Carvalho[1], Amira Dems[2], and Ribal Atallah[2]

[1]CIRRELT and Département d'informatique et de recherche opérationnelle, Université de Montréal
[2]Institut de Recherche d'Hydro-Quebec

Abstract

This study addresses the bilevel network design problem (NDP) with congestion. The upper-level decision-maker (a network designer) selects a set of arcs to add to an existing transportation network, while the lower-level decision-makers (drivers) respond by choosing routes that minimize their individual travel times, resulting in user equilibrium. In this work, we propose two novel single-level reformulations: one based on strong duality and the other based on the value function of the lower-level problem. Unlike existing approaches in the literature, which are specialized for optimizing the total travel time of all drivers, our approach is flexible and can optimize other metrics related to individual travel times or fairness. We discuss the differences between the two reformulations, as well as their computational performance on academic test instances of the NDP. We then apply our methods to the EV charging station capacity expansion problem. We define a metric, the cost of sustainability, to measure the service quality experienced by individual EV drivers, and optimize the charging station locations to improve this metric. We present the results of experiments using the road network in Quebec, including public fast EV charging stations.

1 Introduction

This study addresses the bilevel network design problem (NDP) with congestion. At the upper-level, a network designer determines which arcs to add to an existing transportation network. Meanwhile, at the lower-level, drivers respond by selecting routes that minimize their individual travel times, resulting in the so-called Wardrop user equilibrium. The effect of congestion is modeled by using arc travel time functions that depend on their traffic volumes. One can formulate the computation of the equilibrium as a convex optimization problem (Beckmann et al. [1955]). Thus, the NDP can be seen as a bilevel programming problem where the leader represents the network designer and the follower represents the drivers using the transportation network.

An emerging application of the NDP is the design of sustainable transportation systems. As concerns over climate change grow, the transportation sector faces increasing pressure to enhance its environmental sustainability. Electric vehicles (EVs) are expected to play a key role to this end. To promote EV adoption, it is crucial to ensure reliable access to public fast-charging stations through effective deployment. The NDP offers a framework to model the optimal sizing and placement of these stations by considering the behaviors and interactions of EV drivers.

Exact solution methods for the NDP have been studied actively in the past decades. To the best of our knowledge, they exclusively assume that the objective of the upper-level problem is the total travel time (see, e.g., Farvaresh and Sepehri [2013b], Rey and Levin [2024] and references therein). The solution methods are tailored to this specific objective function and it is not straightforward to extend the approaches to handle different objective functions or extra constraints.

However, in some contexts, there are metrics other than the total travel time that may interest the network designer. One example of such a metric is fairness. Consider a road network with conventional cars of the same specifications. The user equilibrium can be considered fair because, at the equilibrium, two drivers with the same origin and destination experience the same travel time (Patriksson [2015]). However, if we compare different setups, drivers may perceive some sort of unfairness. For example, if there is no congestion on the road (e.g., during an off-peak period), they can travel in a shorter time. This difference is referred to as free-flow unfairness by Jahn et al. [2005].

We can also consider the discrepancy between different modes of transportation. As an example, suppose we are given a road network with EVs and conventional cars. Typically, an EV requires a non-negligible amount of time to charge the battery. Furthermore, often EV charging stations are scarce and EV drivers need to deviate from their preferred routes to visit a charging station. Therefore, EV drivers may spend extra time visiting and using charging stations along their trips, which would not be necessary if they used conventional cars. This extra time spent by EV drivers can be seen as a metric to indicate the inconvenience/inefficiency experienced by the EV drivers. The fast EV charging station operator may be interested in finding the fast charging station locations accounting for such a metric.

In this paper, we present new approaches to model the NDP. Our approaches are based on single-level reformulations of the NDP as convex mixed-integer nonlinear programs (MINLP). Such reformulations can be solved with a standard convex MINLP solver and benefit from techniques developed in the literature. Furthermore, our reformulations are flexible so that we can formulate variants of the NDP that optimize different metrics other than the total travel time, such as free-flow unfairness. We also demonstrate that the capacity expansion (sizing) problem of fast EV charging stations can be modeled as the NDP. We apply our reformulations to optimize the total travel time and other inefficiency metrics (formally defined later) related to EV usage in a case study of the province of Quebec. Our computational results showcase the sensitivity of the solution to the objective, highlighting the importance of appropriately choosing the objective for the upper-level problem.

The paper is organized as follows. Section 2 provides a literature review. In Section 3, we introduce the overall network design problem. We also outline formulations of the traffic assignment problem, the lower-level problem of the NDP. Section 4 presents our single-level reformulations of the network design problem. We provide numerical experiments using test instances of the NDP in Section 5. In Section 6, the capacity expansion problem of fast EV charging stations is formulated as the NDP, and we provide numerical experiments based on the Quebec road network. We conclude the paper in Section 7.

2 Literature Review

The NDP involves the modification or installation of transportation network components, such as arcs. The decisions can be either continuous (e.g., the capacity of arcs) or discrete (e.g., the addition of new arcs), as discussed in Magnanti and Wong [1984]. In this work, we focus on the NDP with discrete decisions, and below we review related solution methods. We also examine models related to the siting and sizing of EV charging stations that are relevant to position our case study.

Discrete Network Design Problem. One of the most popular approaches for solving the NDP is the branch-and-bound algorithm. Leblanc [1975] points out that the high-point relaxation (HPR) can be used to compute a bound for each node in the branch-and-bound algorithm. He further relaxes an upper-level budget constraint, yielding a relaxation of the HPR that allows for a quicker computation of a lower bound. To solve the HPR efficiently, Farvaresh and Sepehri [2013b] use an outer-approximation method, while Bagloee et al. [2017] apply Benders' decomposition. More recently, Rey and Levin [2024] combine an outer-approximation algorithm with a column generation technique.

Another popular approach is to approximate nonlinear functions with piecewise linear ones. Farvaresh and Sepehri [2011] replace the lower-level problem with the Karush–Kuhn–Tucker (KKT) conditions and linearize the nonlinear terms, transforming the problem into a single-level mixed-integer linear programming problem. Fontaine and Minner [2014] linearize the travel time functions and approximate the lower-level problem with linear programming, which can then be reformulated as a single-level mixed-

integer linear programming problem. While these methods are not exact due to the piecewise linear approximation, their numerical experiments show that they can produce high-quality solutions.

All the methods mentioned above assume that the objective function is the total travel time and rely on the problem-specific structure. Modifying these methods to consider objective functions other than total travel time is not straightforward.

Sizing and Placement of EV Charging Stations. In EV infrastructure planning, bilevel programming is often used to model the interaction between an EV charging station operator and EV drivers. Typically, the upper-level problem seeks to minimize costs, emissions, or travel times, while the lower-level problem captures the behavior of the drivers, taking into account factors such as route choices, waiting times at charging stations, and vehicle range constraints. For example, Kınay et al. [2023] use logic-based Benders' decomposition to minimize the installation costs of charging stations. Zhang et al. [2023] consider a bilevel programming problem where the upper-level is a multi-objective optimization problem aimed at minimizing both total costs and total service time. However, neither Kınay et al. [2023] nor Zhang et al. [2023] account for the interaction between drivers, such as congestion in the lower-level problem.

Tran et al. [2021] use the NDP to model EV infrastructure planning. In this model, the upper-level decision-maker determines the locations of charging stations, while the lower-level problem is a user equilibrium problem under congestion. Similarly, Jing et al. [2017] consider a bilevel NDP problem where the lower-level problem models drivers' behavior as stochastic user equilibrium. Both studies focus on road congestion rather than congestion at EV charging stations and use heuristics to find feasible solutions quickly. In our case study, we model congestion at charging stations, optimize various metrics–including those related to fairness–by expanding stations' capacities, and use an exact MINLP solver to optimize our reformulation.

3 Model

In this section, we present the formulation of the NDP as a bilevel programming formulation.

Let $G = (V, A)$ be a directed graph. The set A contains all the *existing arcs* as well as *candidate arcs* that can be built. For each arc $a \in A$, we associate a binary variable x_a to indicate its availability: The value of x_a is 1 if arc a is *available* and 0 otherwise. We write the set of feasible values of x as $X \subset \{0,1\}^A$. For example, X can model a budget constraint. If arc $a \in A$ already exists, we have $x_a = 1$ for any $x \in X$.

Denote by K the index set of the origin-destination (OD) pairs of nodes in the network. For each $k \in K$, we denote its *origin*, *destination* and *demand volume* by o_k, d_k and e_k, respectively. Furthermore, for each $a \in A$, let θ_a be the *travel time function*, the function that maps the link flow on a (total traffic volume on a) to its travel time.

Throughout this paper, we assume the following:

Assumption 1. *1. For each demand $k \in K$, there is a path from o_k to d_k that only uses existing arcs.*

2. For each arc $a \in A$, given the link flow $s \geq 0$, the travel time function is given by

$$\theta_a(s) = f_a + g_a s^p, \tag{1}$$

where $f_a \geq 0$ and $g_a \geq 0$ are non-negative real values and $p \geq 1$ is a positive integer.

These assumptions are not restrictive. If the road network is already well-connected, all the OD pairs are traversable even without building any new arcs. We also note that the family of travel time functions modeled by (1) includes the Bureau of Public Roads (BPR) function (Patriksson [2015]) as well as a constant travel time. All the test instances we use in our numerical experiments (Sections 5 and 6) satisfy Assumption 1.

One of the popular formulations (for example, see Farvaresh and Sepehri [2013a], Wang et al. [2013] for example) of the NDP models the flow of each demand $k \in K$ along each arc $a \in A$. Let $y \in \mathbb{R}^{K \times A}$ be the variable representing the individual flows. The NDP is written as

$$\begin{aligned} \min_{x,y} \ & \sum_{a \in A} \left(\sum_{k \in K} y_{ka} \right) \theta_a \left(\sum_{k \in K} y_{ka} \right) \\ \text{s.t. } & x \in X, y \in Y(x), \end{aligned} \tag{2}$$

where $Y(x)$ is the set of user equilibria parametrized by x, which will be discussed in Section 3.1. The objective function is the total travel time experienced by all the demands, which is convex in y. In this paper, we also consider variants of the NDP that optimize other metrics (see Sections 4.1 and 6.3).

Problem (2) contains a large number of variables, in particular those related to the flow. However, it is possible to aggregate the flow variables if the origins of the corresponding OD pairs are identical without altering the optimal objective value and the optimal values of x. Such aggregation can make the formulation size smaller but weaken the formulation (the resulting continuous relaxation will give a weaker lower bound). Thus, whether such aggregation leads to a speed-up depends on the instance. This technique is well-known in the multicommodity flow literature. See the discussion by Chouman et al. [2017] and references therein. To the best of our knowledge, this flow aggregation is not exploited in the literature on the bilevel NDP under user equilibrium. However, our earlier experiments revealed that aggregating the flow variables reduced the solution time (see Appendix A). Therefore, in this paper, we consider formulations where the flow variables are aggregated.

Let $O = \{v \in V : \exists k \in K, o_k = v\}$ be the set of origins, and for each $o \in O$, let $K_o = \{k \in K : o_k = o\}$ be the set of demands whose origins are o. We also denote the flow aggregated by origins as

$$z_{oa} = \sum_{k \in K_o} y_{ka}.$$

Then, problem (2) can be rewritten as

$$\begin{aligned} \min_{x,z} \ & \sum_{a \in A} \left(\sum_{o \in O} z_{oa} \right) \theta_a \left(\sum_{o \in O} z_{oa} \right) \\ \text{s.t. } & x \in X, z \in Z(x), \end{aligned} \tag{3}$$

where $Z(x)$ is the set of aggregated flow at the user equilibrium given x.

As we will see in Section 3.1, the equilibrium flows $Z(x)$ can be modeled as the solution set of an optimization problem, and thus formulation (3) is a bilevel program. In particular, it corresponds to the optimistic formulation. If multiple user equilibria exist, we assume that the most desirable one (i.e., the one with the smallest total travel time) will be realized. The contrary is the pessimistic formulation, where we assume that the least desirable one (i.e., the one with the largest total travel time) will be realized. However, as stated by Roughgarden and Tardos [2000], in the case of multiple user equilibria, all user equilibria have the same total travel time:

$$\sum_{a \in A} \left(\sum_{o \in O} z_{oa} \right) \theta_a \left(\sum_{o \in O} z_{oa} \right) = \sum_{a \in A} \left(\sum_{o \in O} z'_{oa} \right) \theta_a \left(\sum_{o \in O} z'_{oa} \right), \qquad \forall x \in X, z, z' \in Z(x).$$

For completeness, we provide a proof in Appendix C. Therefore, under Assumption 1, there are no distinctions between the optimistic and pessimistic formulations.

3.1 Wardrop Equilibrium Conditions

In this section, we assume $x \in X$ is given and illustrate how the set of user equilibria $Z(x)$ is modeled. Let $G(x) = (V, A(x))$ be the subgraph of G that only contains available arcs:

$$A(x) = \{a \in A : x_a = 1\}.$$

Furthermore, we define the graph $\underline{G} = (V, \underline{A})$ consisting of the existent arcs, i.e.,

$$\underline{A} = \bigcap_{x \in X} A(x).$$

To describe a set of feasible flows, define for each $k \in K$ and $v \in V$

$$e'_{kv} = \begin{cases} -e_k, & \text{if } v = o_k, \\ e_k, & \text{if } v = d_k, \\ 0, & \text{otherwise,} \end{cases}$$

and the aggregated variant for each $o \in O$ and $v \in V$ by

$$e_{ov} = \sum_{k \in K_o} e'_{kv}.$$

Then, a feasible flow must satisfy the flow conservation constraint

$$\sum_{a \in A_v^+(x)} z_{oa} - \sum_{a \in A_v^-(x)} z_{oa} = e_{ov}, \quad \forall o \in O, v \in V,$$

where $A_v^+(x)$ and $A_v^-(x)$ correspond to the set of incoming and outgoing arcs of v.

We also assume each driver is self-driven and chooses the route so that their travel time is minimized. This is referred to as Wardrop's second principle. It is shown by Beckmann et al. [1955] that the user equilibrium is given as the optimal solution to the following problem:

$$\phi(x) = \min_z \sum_{a \in A(x)} \int_0^{\sum_{o \in O} z_{oa}} \theta_a(s) ds \tag{4a}$$

$$\text{s.t.} \sum_{a \in A_v^+(x)} z_{oa} - \sum_{a \in A_v^-(x)} z_{oa} = e_{ov}, \quad \forall o \in O, v \in V, \tag{4b}$$

$$z \geq 0. \tag{4c}$$

The KKT condition of (4) is given by

$$\sum_{a \in A_v^+(x)} z_{oa} - \sum_{a \in A_v^-(x)} z_{oa} = e_{ov}, \qquad \forall o \in O, v \in V, \tag{5a}$$

$$\theta_a \left(\sum_{o \in O} z_{oa} \right) - \pi_{ot} + \pi_{oh} \geq 0, \qquad \forall o \in O, a = (h,t) \in A(x), \tag{5b}$$

$$\left(\theta_a \left(\sum_{o \in O} z_{oa} \right) - \pi_{ot} + \pi_{oh} \right) z_{oa} = 0, \quad \forall o \in O, a = (h,t) \in A(x), \tag{5c}$$

$$z \geq 0. \tag{5d}$$

When the travel time function θ_a is given by (1) for all $a \in A(x)$, it can be shown (see Appendix B) that the Lagrangian dual of (4) is given by

$$\max_{\pi, \eta} \sum_{k \in K} e_k (\pi_{o_k d_k} - \pi_{o_k o_k}) - \sum_{a \in A(x)} \frac{p}{\overline{g}_a^{1/p} (p+1)} \eta_a^{\frac{p+1}{p}} \tag{6}$$

$$\text{s.t. } \eta_a \geq \pi_{ot} - \pi_{oh} - f_a, \qquad \forall o \in O, a = (h,t) \in A(x),$$

$$\eta_a = 0, \qquad \forall a \in A(x) : g_a = 0,$$

$$\eta \geq 0,$$

where $\overline{g}_a = g_a$ if $g_a > 0$ and otherwise $\overline{g}_a = 1$. This dual formulation is exploited in the literature. For example, Fukushima [1984] develops a solution method based on this dual formulation. We denote the set of optimal values of π for formulation (6) by $\Pi(x)$.

Under Assumption 1, for any $x \in X$, the primal problem (4) and dual problem (6) have a feasible solution, strong duality holds and the primal/dual variables are optimal if and only if they satisfy the KKT condition (5). This is a standard result in the literature. See Rockafellar et al. [1981] for more details. Furthermore, as discussed by Patriksson [2015], for any $x \in X$, $k \in K$ and $\pi(x) \in \Pi(x)$, $\pi_{o_k d_k}(x) - \pi_{o_k o_k}(x)$ is the travel time experienced by each individual of OD pair k to travel from o_k to d_k under the congestion at the user equilibrium.

In particular, strong duality implies that $Z(x) = \mathrm{proj}_z(Z'(x))$, where

$$Z'(x) = \left\{ \begin{pmatrix} z \\ \pi \\ \eta \end{pmatrix} : \begin{array}{ll} \displaystyle\sum_{a \in A(x)} \int_0^{\sum_{o \in O} z_{oa}} \theta_a(s) ds & \\ \quad \le \displaystyle\sum_{k \in K} e_k(\pi_{o_k d_k} - \pi_{o_k o_k}) - \sum_{a \in A(x)} \frac{p}{\overline{g}_a^{1/p}(p+1)} \eta_a^{\frac{p+1}{p}}, & \\ \displaystyle\sum_{a \in A_v^+(x)} z_{oa} - \sum_{a \in A_v^-(x)} z_{oa} = e_{ov}, & \forall o \in O, v \in V, \\ \eta_a \ge \pi_{ot} - \pi_{oh} - f_a, & \forall o \in O, a = (h,t) \in A(x), \\ \eta_a = 0, & a \in A(x) : g_a = 0, \\ z, \eta \ge 0 & \end{array} \right\}.$$

We will use this property to derive one of our single-level reformulations of the NDP in Section 4.

3.2 Capacity Expansion as Multigraph

Before discussing our solution methods, we would like to note that the NDP can model the expansion of existing roads as well as the installation of new roads. For example consider the traffic assignment on graph G_1 in Figure 1 with a single OD pair (i.e., $K = \{1\}$) from node 1 to node 3. For any demand volume $e_1 \ge 0$, there is a unique user equilibrium given by

$$z_{1a} = e_1, \quad \forall a \in \{1, 2\},$$

and the total travel time (the upper-level objective in the NDP (3)) is

$$\sum_{a \in A} \left(\sum_{o \in O} z_{oa} \right) \theta_a \left(\sum_{o \in O} z_{oa} \right) = e_1 \cdot 1 + e_1 \cdot e_1/3 = e_1^2/3 + e_1.$$

Now consider the case where we build another arc from node 2 to node 3 with travel time $s/2$ as shown in graph G_2. For any demand volume $e_1 \ge 0$, the user equilibrium is unique and given by

$$z_{11} = e_1, \quad z_{12} = 3e_1/5, \quad z_{13} = 2e_1/5,$$

with the total travel time $e_1^2/5 + e_1$. Similarly, we can show that graph G_3 has the unique user equilibrium

$$z_{1a} = e_1, \quad \forall a \in \{1, 2\},$$

with the total travel time $e_1^2/5 + e_1$. The total travel times of G_2 and G_3 are the same for any e_1, and we can view these two graphs "equivalent". Intuitively, the reciprocal of the coefficient g_a in the travel time function θ_a can be seen as the capacity of the road. In this regard, graph G_2 has two arcs of capacities 3 and 2, while graph G_3 has an equivalent arc of capacity 5. In general, if there are multiple parallel arcs with the same free-flow travel times and non-constant travel times (i.e., $g_a \neq 0$), we can aggregate them into a single arc while maintaining the total travel time at user equilibrium. See Section 8N of Rockafellar [1998] for more formal arguments.

Now, consider the NDP instance shown in graph G_4 in Figure 1. Suppose that arcs 1 and 2 are already built (existing arcs) and $X = \{x^{(1)} = (1,1,0), x^{(2)} = (1,1,1)\}$. If we do not build arc 3, the graph is equivalent to G_1. If we build arc 3, the graph coincides with G_2, which is equivalent to G_3. Therefore, the binary decision to add a new arc is equivalent to the decision of the capacity expansion of the road. We will use the equivalences described here in Section 6.

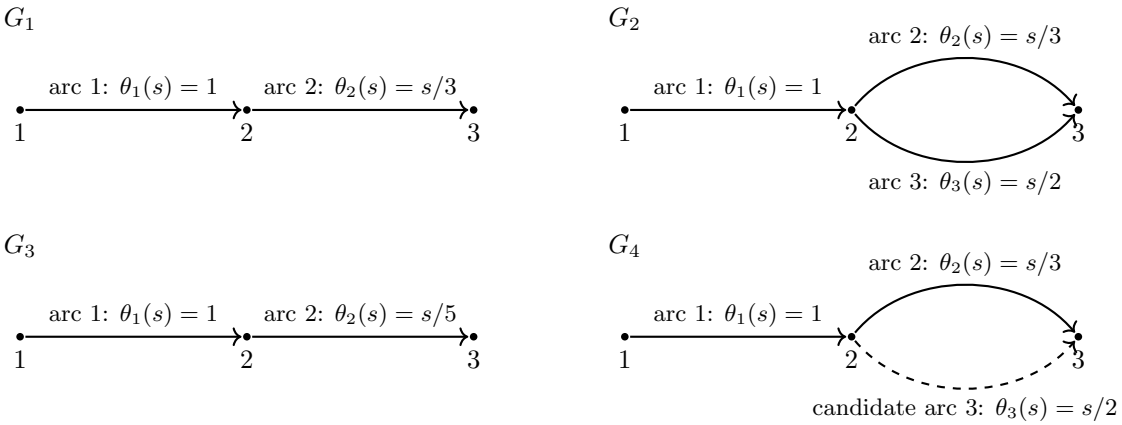

Figure 1: Example of graphs used to observe the effect of parallel arcs

4 Methodology

In this section, we propose our single-level reformulations of problem (3). The resulting formulations are convex MINLP and thus potentially solvable by existing optimization tools.

One natural approach would be replacing the optimality constraint $z \in Z(x)$ with the KKT condition of the lower level problem (4):

$$\min_{x,z,\pi} \sum_{a\in A}\left(\sum_{o\in O} z_{oa}\right)\theta_a\left(\sum_{o\in O} z_{oa}\right) \tag{7a}$$

$$\text{s.t.} \sum_{a\in A_v^+} z_{oa} - \sum_{a\in A_v^-} z_{oa} = e_{ov}, \qquad \forall o \in O, v \in V, \tag{7b}$$

$$z_{oa} \le \sum_{k\in K_o} e_k x_a, \qquad \forall o \in O, a \in A, \tag{7c}$$

$$\theta_a\left(\sum_{o\in O} z_{oa}\right) - \pi_{ot} + \pi_{oh} + N(1-x_a) \ge 0, \quad \forall o \in O, a = (h,t) \in A, \tag{7d}$$

$$\left(\theta_a\left(\sum_{o\in O} z_{oa}\right) - \pi_{ot} + \pi_{oh}\right) z_{oa} \ge 0, \qquad \forall o \in O, a = (h,t) \in A, \tag{7e}$$

$$x \in X, z \ge 0. \tag{7f}$$

where for each $v \in V$, A_v^+ and A_v^- are defined as

$$A_v^+ = \{(h,t) \in A : t = v\}, \quad A_v^- = \{(h,t) \in A : h = v\},$$

and N is a sufficiently large constant. However, this formulation is a non-convex MINLP. In particular, constraint (7d) is a level-set of a concave function. It is not straightforward to reformulate this as convex MINLP.

Instead, we seek a single-level reformulation based on strong duality. Based on the observation that $Z(x)$ is given as the projection of $Z'(x)$, we obtain the following formulation of the NDP:

$$\min_{x,y,\pi,\eta} \sum_{a\in A}\left(\sum_{o\in O} z_{oa}\right)\theta_a\left(\sum_{o\in O} z_{oa}\right) \tag{8a}$$

$$\text{s.t.} \sum_{a\in A}\int_0^{\sum_{o\in O} z_{oa}} \theta_a(s)ds$$

$$\leq \sum_{k\in K} e_k(\pi_{o_k d_k} - \pi_{o_k o_k}) - \sum_{a\in A} \frac{p}{\overline{g}_a^{1/p}(p+1)}\eta_a^{\frac{p+1}{p}}, \tag{8b}$$

$$\sum_{a\in A_v^+} z_{oa} - \sum_{a\in A_v^-} z_{oa} = e_{ov}, \qquad \forall o\in O, v\in V, \tag{8c}$$

$$z_{oa} \leq \sum_{k\in K_o} e_k x_a, \qquad \forall o\in O, a\in A, \tag{8d}$$

$$\eta_a \geq \pi_{ot} - \pi_{oh} - f_a - M_{oa}(1-x_a), \qquad \forall o\in O, a=(h,t)\in A, \tag{8e}$$

$$\eta_a = 0, \qquad \forall a\in A: g_a = 0, \tag{8f}$$

$$z\geq 0, \eta\geq 0, x\in X, \tag{8g}$$

where $M\in\mathbb{R}^{O\times A}$ is a constant such that for any $x\in X$ there exists $(\pi,\eta)\in\Pi(x)$ satisfying

$$\pi_{ot} - \pi_{oh} - f_a \leq M_{oa}, \qquad \forall o\in O, a=(h,t)\in A\setminus A(x).$$

We will examine approaches to compute valid values of M in Section 4.2.

Another formulation is based on the value function of the lower-level problem. Recall that $\phi(x)$ is the optimal objective value of (4). Therefore, we obtain a single-level reformulation of (3) as

$$\begin{aligned}
\min_{x,z}\ & \sum_{a\in A}\left(\sum_{o\in O} z_{oa}\right)\theta_a\left(\sum_{o\in O} z_{oa}\right)\\
\text{s.t.}\ & \sum_{a\in A}\int_0^{\sum_{o\in O} z_{oa}} \theta_a(s)ds \leq \phi(x),\\
& \sum_{a\in A_v^+} z_{oa} - \sum_{a\in A_v^-} z_{oa} = e_{ov}, && \forall o\in O, v\in V,\\
& z_{oa} \leq \sum_{k\in K_o} e_k x_a, && \forall o\in O, a\in A,\\
& z\geq 0, x\in X.
\end{aligned}$$

It can be shown that the value function $\phi(x)$ is non-increasing in x (see Rey and Levin [2024] for example). Therefore, the constraint involving the value function can be linearized as

$$\min_{x,z} \sum_{a\in A}\left(\sum_{o\in O} z_{oa}\right)\theta_a\left(\sum_{o\in O} z_{oa}\right) \tag{9a}$$

$$\text{s.t.} \sum_{a\in A}\int_0^{\sum_{o\in O} z_{oa}} \theta_a(s)ds \leq \phi(x') + \hat{\phi}(x')\sum_{a\in A} x'_a(1-x_a), \quad \forall x'\in X, \tag{9b}$$

$$\sum_{a\in A_v^+} z_{oa} - \sum_{a\in A_v^-} z_{oa} = e_{ov}, \qquad \forall o\in O, v\in V, \tag{9c}$$

$$z_{oa} \leq \sum_{k\in K_o} e_k x_a, \qquad \forall o\in O, a\in A, \tag{9d}$$

$$z\geq 0, x\in X. \tag{9e}$$

where $\hat{\phi}(x') = (\phi(0) - \phi(x'))$ for $x' \in X$. Rey and Levin [2024] use a closely related cut based on the value function.

Below, we make important remarks regarding formulations (8) and (9).

1. Formulations (8) and (9) are convex MINLP problems and various solution methods have been proposed. See Bonami et al. [2012] for a survey. In our implementation, we use the so-called LP/NLP-Based Branch-and-Bound algorithm as described in Appendix D.

2. Formulation (9) has a large number of constraints, whereas formulation (8) has a number of variables and constraints polynomial in $|V|$, $|A|$ and $|K|$. Nonetheless, as we will see in our computational experiments, by handling the constraints lazily, formulation (9) can be more efficient than formulation (8).

3. Formulation (9) is based on the value function $\phi(x)$. Even if the travel time function θ_a does not belong to the family of functions given in (1), it is still possible to compute the user equilibrium by solving (4) and evaluate $\phi(x)$. On the other hand, formulation (8) relies on the dual (6) of the traffic assignment problem. If the travel time function θ_a is not as in (1), the dual (6) may not be available explicitly. In such a case, formulation (8) is not applicable.

4.1 Optimization of Other Objectives

Formulation (8) contains more variables than formulation (9). These extra variables can be used to model different objective functions or constraints. For example, for each $k \in K$, as discussed in Section 3.1, $\pi_{o_k d_k} - \pi_{o_k o_k}$ gives the individual travel time of OD pair k at the user equilibrium. Let ν_k be the shortest travel time from o_k to d_k assuming there is no congestion. Then, the ratio of the extra travel time each driver of OD pair k experiences due to congestion is given by

$$(\pi_{o_k d_k} - \pi_{o_k o_k} - \nu_k)/\nu_k.$$

This metric is referred to as free-flow unfairness by Jahn et al. [2005]. We can modify the NDP (8) to optimize the total free-flow unfairness as

$$\begin{aligned} \min\ & \sum_{k \in K} (\pi_{o_k d_k} - \pi_{o_k o_k} - \nu_k)/\nu_k \\ \text{s.t. } & \text{(8b)} - \text{(8g)}. \end{aligned} \tag{10}$$

and the worst free-flow unfairness as

$$\begin{aligned} \min\ & \xi \\ \text{s.t. } & \xi \geq (\pi_{o_k d_k} - \pi_{o_k o_k} - \nu_k)/\nu_k, \quad \forall k \in K, \\ & \text{(8b)} - \text{(8g)}. \end{aligned} \tag{11}$$

Theorem 2.5 by Patriksson [2015] states that the equilibrium travel times are unique. Therefore, the total free-flow unfairness and the worst free-flow unfairness are the same across user equilibria. Similarly to the argument in Appendix C, we conclude that there is no distinction between the optimistic and pessimistic formulations of problems (10) and (11).

Note that given

$$\pi_{o_k d_k} - \pi_{o_k o_k} = \sum_{a \in A} y_{ka} \theta_a \left(\sum_{k \in K} y_{ka} \right) / e_k$$

provides an alternative formulation of the travel time of OD pair $k \in K$, problem (10) can be reformulated without the dual variables as

$$\begin{aligned} \min\ & \sum_{k \in K} \left(\sum_{a \in A} y_{ka} \theta_a \left(\sum_{k' \in K} y_{k'a} \right) / e_k - \nu_k \right) / \nu_k \\ \text{s.t. } & x \in X, y \in Y(x). \end{aligned}$$

However, the objective function of this problem is non-convex.

4.2 Computing Big-M

This subsection presents our approach to compute a valid value of big-M. This can be done by bounding the difference of optimal dual values in $\Pi(x)$. The following proposition gives a bound for a specific $x \in X$, $k \in K$ and a pair of nodes (h, t).

Proposition 1. *Let $x \in X$, $o \in O$, $h, t \in V$ and $\delta > 0$. Let $z \geq 0$ be such that for each $o' \in O$ and $v \in V$, we have*

$$\sum_{a \in A_v^+(x)} z_{o'a} - \sum_{a \in A_v^-(x)} z_{o'a} = \begin{cases} e_{o'v} - \delta, & \text{if } o' = o \text{ and } v = h, \\ e_{o'v} + \delta, & \text{if } o' = o \text{ and } v = t, \\ e_{o'v}, & \text{otherwise,} \end{cases}$$

Then, under Assumption 1, for any $\pi(x) \in \Pi(x)$, we have

$$\pi_{ot}(x) - \pi_{oh}(x) \leq \frac{1}{\delta} \left(\sum_{a \in A(x)} \int_0^{\sum_{o' \in O} z_{o'a}} \theta_a(s) ds - \phi(x) \right).$$

Proof. Let L be the Lagrangian of problem (4) (Appendix B). Using strong duality, for any optimal dual values $\pi(x) \in \Pi(x)$, we obtain

$$\begin{aligned} \phi(x) &= \min_{z' \geq 0} L(z', \pi(x)) \\ &\leq L(z, \pi(x)) \\ &= \sum_{a \in A(x)} \int_0^{\sum_{o' \in O} z_{o'a}} \theta_a(s) ds - \sum_{o' \in O} \sum_{v \in V} \pi_{o'v}(x) \left(\sum_{a \in A_v^+(x)} z_{o'a} - \sum_{a \in A_v^-(x)} z_{o'a} - e_{o'v} \right) \\ &= \sum_{a \in A(x)} \int_0^{\sum_{o' \in O} z_{o'a}} \theta_a(s) ds - \delta \left(\pi_{ot}(x) - \pi_{oh}(x) \right). \end{aligned}$$

By rearranging the terms, we obtain the desired result. □ □

Here we make an intuitive argument regarding the preceding proposition. The dual variable π is the sensitivity of the optimal objective value with respect to the RHS perturbation on constraint (4b). In particular, $\pi_{ot}(x) - \pi_{oh}(x)$ gives an underestimate of the increase in the objective value if we are forced to "consume" a unit flow at t and "reproduce" the same amount of flow at h. Let $\overline{Z}_\delta$ be the set of such flows on $G(x)$: $z \in \overline{Z}_\delta$ if and only if $z \geq 0$ and for each $o' \in O$ and $v \in V$, we have

$$\sum_{a \in A_v^+(x)} z_{o'a} - \sum_{a \in A_v^-(x)} z_{o'a} = \begin{cases} e_{o'v} - \delta, & \text{if } o' = o \text{ and } v = h, \\ e_{o'v} + \delta, & \text{if } o' = o \text{ and } v = t, \\ e_{o'v}, & \text{otherwise.} \end{cases}$$

Now, observe

$$\phi(x) + \delta(\pi_{ot}(x) - \pi_{oh}(x)) \leq \min_{z \in \overline{Z}_\delta} \sum_{a \in A(x)} \int_0^{\sum_{o' \in O} z_{o'a}} \theta_a(s) ds,$$

or

$$\pi_{ot}(x) - \pi_{oh}(x) \leq \frac{1}{\delta} \left(\min_{z \in \overline{Z}_\delta} \sum_{a \in A(x)} \int_0^{\sum_{o' \in O} z_{o'a}} \theta_a(s) ds - \phi(x) \right).$$

The above proposition replaces the minimum value in the RHS with the objective value of a feasible solution.

A flow in $\overline{Z}_\delta$ above can be obtained by combining a flow on $\underline{G}$ satisfying the demand and a δ-unit of flow on $\underline{G}$ from h to t, in the sense described in the next proposition.

Proposition 2. *Let $h, t \in V$ and $\delta > 0$. Let $z \geq 0$ be any flow on graph $\underline{G}$ satisfying the OD pairs K:*

$$\sum_{a \in \underline{A}_v^+} z_{oa} - \sum_{a \in \underline{A}_v^-} z_{oa} = e_{ov}, \quad \forall o \in O, v \in V.$$

Furthermore, let $z' \geq 0$ be any flow on graph $\underline{G}$ such that for each $v \in V$

$$\sum_{a \in \underline{A}_v^+} z'_a - \sum_{a \in \underline{A}_v^-} z'_a = \begin{cases} -\delta, & \text{if } v = h, \\ \delta, & \text{if } v = t, \\ 0, & \text{otherwise.} \end{cases}$$

Then, under Assumption 1, for any $x \in X$, $o \in O$ and $\pi(x) \in \Pi(x)$, we have

$$\pi_{ot}(x) - \pi_{oh}(x) \leq \frac{1}{\delta} \left(\sum_{a \in \underline{A}} \int_0^{z'_a + \sum_{o' \in O} z_{o'a}} \theta_a(s) ds - \underline{\phi} \right),$$

where $\underline{\phi} = \min_{x \in X} \phi(x)$.

Proof. Fix $x \in X$, $o \in O$ and define for each $o' \in O$, $a \in A$

$$\hat{z}_{o'a} = \begin{cases} z_{o'a}, & \text{if } o' \neq o, \\ z'_a + z_{o'a}, & \text{if } o' = o. \end{cases}$$

Flow $\hat{z}$ satisfies the condition in Proposition 1 and is feasible on graph $G(x)$. Thus, for any $\pi(x) \in \Pi(x)$, we obtain

$$\begin{aligned} \pi_{ot}(x) - \pi_{oh}(x) &\leq \frac{1}{\delta} \left(\sum_{a \in A(x)} \int_0^{\sum_{o' \in O} \hat{z}_{o'a}} \theta_a(s) ds - \phi(x) \right) \\ &= \frac{1}{\delta} \left(\sum_{a \in \underline{A}} \int_0^{z'_a + \sum_{o' \in O} z_{o'a}} \theta_a(s) ds - \phi(x) \right) \\ &\leq \frac{1}{\delta} \left(\sum_{a \in \underline{A}} \int_0^{z'_a + \sum_{o' \in O} z'_{o'a}} \theta_a(s) ds - \underline{\phi} \right), \end{aligned}$$

as required. □ □

This proposition is useful since one can obtain a bound valid for multiple graphs and demands.

In our implementation, we compute the values of big-M as follows. First, we solve the traffic assignment problem (4) on G (i.e., the graph with all the existing and candidate arcs) to obtain a lower bound on $\underline{\phi}$: In light of the monotonicity of ϕ, we have $\underline{\phi} \leq \phi(x)$ for any $x \in X$. Then, we solve the traffic assignment problem (4) on $\underline{G}$ to obtain a flow z as well as travel times of the arcs under the congestion. Lastly, for each pair $h, t \in V$, we obtain the shortest path on $\underline{G}$ under the congestion induced by the flow z. We consider δ unit of flow along the shortest path and use it as z'. Then we invoke Proposition 2 with these z and z'. We choose the value of δ by grid search ($\delta \in \{10^{-2}, 10^{-1}, 10^0, 10^1, 10^2\}$).

5 Computational Performance

In this section, we compare the performance of our single-level reformulations. We use instances generated by Rey and Levin [2024]. Two transportation networks, SiouxFalls and Easter Massachusetts, are used, and their statistics are shown in Table 1. For each transportation network, there are 10 test instances

created by perturbing the set of candidate arcs. Let A_1 and A_2 be the sets of existing and candidate arcs, respectively. Then, set X is defined by the knapsack constraint

$$X = \left\{ x : \sum_{a \in A_2} c_a x_a \leq b,\ x_a = 1,\ \forall a \in A_1,\ x_a \in \{0,1\},\ \forall a \in A_2 \right\}, \tag{12}$$

where c_a is the cost to construct candidate arc $a \in A_2$ and b is the budget. For more details on the instance generation, see Rey and Levin [2024]. All the methods are implemented with Python 3.9 with Gurobi 12.0. Formulation SD (8) requires two computations of the traffic assignment to compute the big-M value at the beginning, as described in Section 4.2. In formulation VF (9), constraint (9b) is generated lazily: Every time an integer solution is found, we compute the corresponding traffic assignment and check whether constraint (9b) is satisfied, adding it if necessary. To compute these user equilibria, we use the column generation technique as described by Leventhal et al. [1973].

Table 1: Statistics of the test instances

Network	Nodes	Arcs	Candidate Arcs	Budget (%)	OD pairs
SiouxFalls	24	76	10	50	528
Easter Massachusetts	74	258	10	50	1113

5.1 NDP to Minimize the Total Travel Time

First, we observe the computational performance of formulations (8) and (9) to optimize the total travel time. Table 2 shows the number of test instances solved within the two-hour time limit (Solved), the average computational time in seconds (Time), the average number of branch-and-bound nodes (# B&B Nodes), and the average number of traffic assignment computations, i.e., the average number of user equilibria computed (# TA Solution). The computational time reported in this table includes the time to compute user equilibria. For the test instances that are not solved within the time limit, the time is set to 2 hours, and the number of branch-and-bound nodes and traffic assignments computed within this 2-hour limit are used. In the column labeled "Formulation", SD (standing for strong duality) and VF (value function) refer to formulations (8) and (9), respectively. In this experiment, all the test instances are solved within the 2-hour time limit. Formulation SD (8) has a shorter average solution time on SiouxFalls, but on Easter Massachusetts, formulation VF (9) tends to be faster on average. Formulation VF (9) requires less branch-and-bound nodes than formulation SD (8), especially on Easter Massachusetts. Interestingly, both formulations require on average fewer branch-and-bound nodes and traffic assignment computations on Easter Massachusetts, a larger transportation network.

Table 2: Performance of formulations SD (8) and VF (9) to solve the NDP to minimize the total travel time

Network	Formulation	Solved	Time	# B&B Nodes	# TA Solution
SiouxFalls	SD	10	217.0	294.5	2.0
	VF	10	558.8	252.6	138.0
Easter Massachusetts	SD	10	1,383.2	166.5	2.0
	VF	10	810.0	111.3	66.8

5.2 NDP to Optimize the Fairness

We next evaluate the capability of our formulations to optimize metrics other than the total travel time, as discussed in Section 4.1. To be more specific, we solve the NDP to optimize the total free-flow

unfairness (10) and the NDP to optimize the worst free-flow unfairness (11). Table 3 shows the number of the test instances solved within the two-hour time limit (Solved), the average computational time in seconds (Time) and the average number of branch-and-bound nodes (# B&B Nodes). Compared to the optimization of the total travel time (8), the optimization of the total free-flow unfairness (10) tends to require more time on SiouxFalls, and more branch-and-bound nodes on both transportation networks. The optimization of the worst free-flow unfairness (11) requires even more computational time and branch-and-bound nodes than the optimization of the total free-flow unfairness (10) on average, and there is one test instance on which the optimal solution was not found in two hours.

Table 3: Performance of formulation SD to optimize the total travel time (8), the total free-flow unfairness (10) and the worst free-flow unfairness (11)

Network	Objective	Solved	Time	# B&B Nodes
SiouxFalls	Travel time	10	217.0	294.5
	Total free-flow unfairness	10	293.9	640.5
	Worst free-flow unfairness	10	505.9	705.3
Easter Massachusetts	Travel time	10	1,383.2	166.5
	Total free-flow unfairness	10	1,754.4	479.9
	Worst free-flow unfairness	8	2,628.4	556.3

We speculate that the reason why optimizing unfairness seems harder than optimizing travel time is as follows: Intuitively, we observe that the total travel time, the objective value of problem (8), is close to the objective value of the lower-level problem (4). In fact, when there is no congestion ($g_a = 0$ for all $a \in A$), the two objective values coincide for any feasible flow. In this setup, constraints (8b), (8e) and (8f), along with dual variables π and η, are redundant in the sense that the optimal solution of problem (8) does not change even if they are removed. Thus, these constraints are only necessary when congestion affects the system, and the "effective" formulation size might be smaller. These observations do not hold for the other objective functions used in problems (10) and (11).

6 Case Study: Fast EV Charging Station Capacity Expansion Problem

In this section, we formulate the capacity expansion problem for EV fast charging stations as the NDP. This formulation enables us to optimize charging infrastructure capacity expansion decisions, aiming to minimize either the total travel time or a specific fairness metric related to EV drivers.

In Section 6.1, we describe the setup of the problem. In Section 6.2, we present our approach for modeling the behavior (i.e., the routing decision) of EV drivers as a traffic assignment problem. This traffic assignment-based modeling allows us to formulate the capacity expansion problem of EV fast charging stations as the NDP. Section 6.3 introduces a metric quantifying the inefficiency experienced by EV drivers. Section 6.4 presents the results of the numerical experiments using the Quebec road network.

6.1 Problem Setup

Suppose we have a highway road network with fast charging stations installed at some of the nodes. We are interested in EV drivers who travel long distances through this road network, represented as OD pairs. Specifically, we focus on the drivers who need to charge their EVs along their trips in order to reach their destinations. These EV drivers consider the time required to charge their vehicles and choose their routes in a way that minimizes their total travel time. Given the current road network with fast charging stations and their associated demands, our goal is to select a subset of fast charging stations for capacity expansion to improve a specified metric, such as the total travel time of EV drivers.

We make the following assumptions:

Assumption 2.

1. *All EVs have the same vehicle range.*
2. *Each OD pair is traversable, assuming the battery is fully charged at the origin and that intermediate charging at existing stations along the route is allowed.*
3. *The travel time on a road does not depend on the number of EV drivers using that road.*
4. *The time required to charge an EV at a fast charging station (including waiting time) can be modeled as in Equation* (1)*.*

The first assumption is standard in the literature (e.g., Kınay et al. [2023]). The assumption that travel time is independent of the number of EV drivers can be justified when the ratio of EVs to conventional vehicles is small. Under Assumption 2, the traffic flow of EV drivers can be modeled as a standard traffic assignment problem after appropriate network transformation, as described in the next section.

6.2 Electric Vehicle Traffic Assignment

Let $G' = (V', A')$ be a road network. Let $O, D \subset V'$ denote the sets of nodes used as origins and destinations by the OD pairs, respectively, and let $C \subset V'$ represent the set of nodes with fast charging stations. By duplicating nodes if necessary, we assume that O, D, and C are mutually disjoint.

An EV driver must choose a path that can be completed without running out of battery. For example, consider graph G_1 in Figure 2. A filled node indicates the presence of a fast charging station, while an open node denotes the absence of one. Given that the vehicle range is 200 km, path $(1, 4, 5, 3, 8)$ is not feasible, as it is not possible to travel from node 1 to node 3 (nodes 4 and 5 do not have fast charging stations).

MirHassani and Ebrazi [2013] propose a technique to transform the road network to facilitate handling the vehicle range. For a detailed discussion, we refer to Arslan et al. [2019]. Roughly, the transformed graph $G = (V, A)$ has $V = O \cup D \cup C$, and for each $(o, d) \in O \times D$, a path from o to d in graph G corresponds to a path in road network G' that respects the vehicle range (and vice versa). The construction of A is straightforward: for each $h, t \in V$, we have $(h, t) \in A$ if and only if the distance from h to t is less than or equal to the vehicle range of an EV, and $(h, t) \in O \times C$, $C \times C$, or $C \times D$. For example, consider again road network G_1 shown in Figure 2. If there is a single OD pair (i.e., $K = \{1\}$) from node 1 to node 8, and the vehicle range of an EV is 200 km, we obtain G_2 after this transformation. Path $p_1 = (1, 2, 6, 8)$ in graph G_2 corresponds to path $p'_1 = (1, 4, 2, 4, 6, 7, 8)$ in graph G_1. The intermediate nodes 2 and 6 in path p_1 correspond to the nodes to charge an EV.

We note that the cost (travel time) of a path in the transformed network consists of both arc costs (time spent on the road) and node costs (time spent at the fast charging stations). By introducing auxiliary nodes and arcs, the node costs can be represented as costs associated with the auxiliary arcs. For example, if the time to charge an EV at a fast charging station is given by $1/2 + s^4$, graph G_2 can be further transformed into graph G_3.

6.3 Cost of Sustainability

On the transformed graph, the traffic assignment can be used to model the user equilibrium flow of EV drivers. For example, on graph G_3 in Figure 2, if the demand volume is $e_1 = 1$, the user equilibrium results in the flow shown in Figure 3. The corresponding user equilibrium flow in the original road network G_1 can be computed, as shown in Figure 3.

In equilibrium, the travel time experienced by each EV driver is $17/4$ hours. On the other hand, if we consider a conventional car, it is possible to travel from node 1 to node 8 in graph G_1 in 3 hours using path $(1, 4, 5, 8)$.[1] We define the extra travel time that an EV user experiences compared to a conventional car driver as the *cost of sustainability*. In the example above, the cost of sustainability is $(17/4 - 3)/3 = 5/12$.

[1] We assume that conventional cars are internal combustion engine (ICE) vehicles and any path is feasible for them given the abundant presence of gas stations. We further assume that the time to refuel a conventional car is negligible.

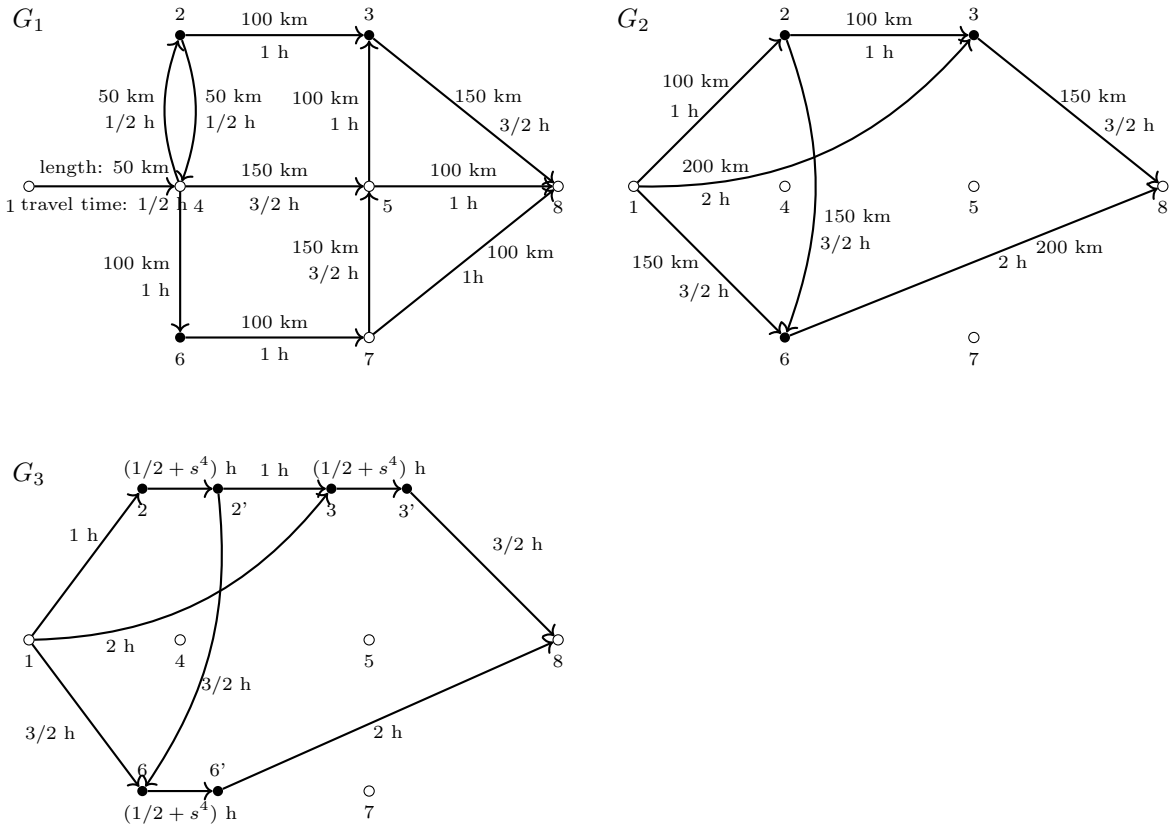

Figure 2: Graphs illustrating network transformation: Road network G_1 is transformed into the extended graph G_2, which is further converted into G_3, where the node costs in G_2 are represented as arc costs.

We note that the cost of sustainability is closely related to the free-flow unfairness objectives discussed in Section 4.1. Here, we use the travel time of a conventional car driver as the "baseline", whereas free-flow unfairness is based on the shortest travel time assuming that all travel time equals the free-flow travel time (i.e., travel time without congestion). Following the argument to derive formulations (10) and (11) to minimize free-flow unfairness, optimization of the total cost of sustainability is written as

$$\begin{aligned} \min \ & \sum_{k \in K} (\pi_{o_k d_k} - \pi_{o_k o_k} - \nu'_k)/\nu'_k \\ \text{s.t. } & \text{(8b)} - \text{(8g)} \end{aligned} \tag{13}$$

and the worst cost of sustainability as

$$\begin{aligned} \min \ & \xi \\ \text{s.t. } & \xi \geq (\pi_{o_k d_k} - \pi_{o_k o_k} - \nu'_k)/\nu'_k, \quad \forall k \in K, \\ & \text{(8b)} - \text{(8g)}. \end{aligned} \tag{14}$$

where ν'_k is the shortest time from o_k to d_k using a conventional car. In our numerical experiments, we will use the total and worst cost of sustainability as objectives to be optimized.

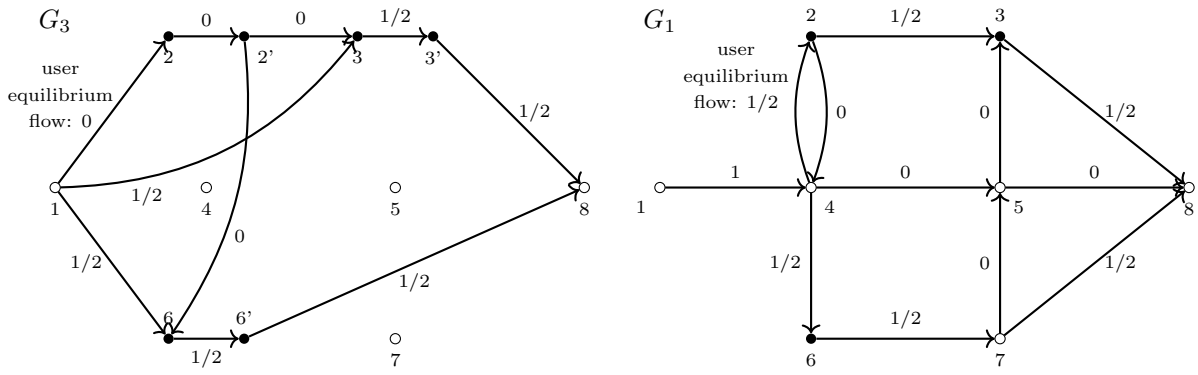

Figure 3: User equilibrium on the transformed network and on the original road network

6.4 Numerical Experiments with Quebec Road Network

Experimental Setup. To illustrate the applicability of our method, we run experiments on the road network in Quebec, including public fast EV charging stations. We use the 2021 Statistics Canada census data[2] to determine population centers. We consider the 20 population centers with the largest populations. We assume the vehicle range of an EV is 200 km and remove OD pairs whose distances are less than this vehicle range, resulting in 92 OD pairs. The demand volumes of the OD pairs are computed using the gravity model (Hodgson [1990]). We consider two demand scenarios: a low-demand scenario and a high-demand scenario where the demand volume of each OD pair is increased by 50% from that of the low-demand scenario. The histogram of the OD demand volumes is presented in Appendix E.1.

The location data of fast charging stations is obtained from *Circuit électric*[3]. We use OpenStreetMap (OpenStreetMap contributors [2024]) with OSMnx (Boeing [2024]) to compute the distances between fast charging stations and population centers using the highway network in Quebec. Based on the computed distances, we construct the transformed network as described in Section 6.2. We consider 370 fast charging stations. The resulting network has a large number of arcs. To make the instance size moderate,

[2] https://www12.statcan.gc.ca/census-recensement/2021/dp-pd/prof/

[3] https://lecircuitelectrique.com/en/

we run the traffic assignment and compute the user equilibrium using the existing fast charging stations. We then remove the arcs with zero flows at equilibrium since they are unlikely to be used after capacity expansion. The statistics of the resulting graphs are shown in Table 4.

Table 4: Statistics of the graph before and after filtering arcs

Network	Nodes	Arcs	Candidate arcs	OD pairs
Original	924	65,888	370	92
Low Demand	924	354	87	92
Hight Demand	924	421	108	92

Table 5: Frequecy of c

c	Frequency
1	89
2	228
3	18
4	24
5	1
6	7
7	2
8	1

The time to charge an EV at a fast charging station (including waiting time) is modeled as $1/2 + (s/c)^4/100$, where s is the amount of demand served at this fast charging station and c is the number of outlets of this fast charging station. The frequency of c is shown in Table 5.

We consider the capacity expansion of each charging station by 1 unit (if we have a charging station of capacity $c = 2$, we may expand it to capacity $c = 3$). For each charging station (candidate arc a), we sample the cost of the capacity expansion (the construction cost c_a of the candidate arc) from a uniform distribution between 0.5 and 1.5. We use X as defined in Equation (12).

We propose a simple greedy heuristic as a baseline. The method works as follows: We first evaluate the user equilibrium with the current set of fast charging stations in place. Next, we compute the "utility" of each fast charging station. This is defined as the ratio of the total demand served by the station to its capacity. We then sort the charging stations in decreasing order of their utility, from the most to the least used. Starting from the highest utility, we choose the stations until we reach a budget limit. After selecting the stations, we perform another traffic assignment to evaluate the objective value (e.g., total travel time). Thus, this greedy method requires two traffic assignment computations: one for evaluating the user equilibrium and another for evaluating the objective after selecting the charging stations.

Results and Discussion. First, we present the performances of our single-level formulations (8) and (9) to seek the capacity expansion decisions that minimize the total travel time of EV drivers. Table 6 shows the upper bound[4] (UB), the improvement of the upper bound relative to the greedy method (Rel.), the lower bound (LB), the gap between upper and lower bound (Gap), and the number of branch-and-bound nodes (# B&B Nodes). We set the time limit to one hour. In all the setups, formulations SD (8) and VF (9) provide tighter upper bounds (better solutions) than the greedy method. Formulation VF (9) tends to give tighter bounds than formulation SD (8), and the optimality gap is smaller in all the setups. The demand volume does not seem to have a clear impact on the optimality gaps. However, the number of branch-and-bound nodes decreases when we use the high-demand scenario. There are no clear trends in terms of methodologies' performance regarding budget variations. Thus, as long as the objective is the total travel time, formulation VF (9) seems to be more competitive than formulation SD (8) and the greedy method.

Next, we use formulations (13) and (14) to optimize the total and worst costs of sustainability, as described in Section 6.3. Recall that the cost of sustainability is defined as the extra travel time EV drivers experience compared to the drivers of conventional cars. Tables 7 and 8 show the upper and lower bounds obtained after the one-hour time limit among other statistics. The definition of each column is the same as in Table 2. Similar to the optimization of the total travel time, we observe that increasing demand makes the problem harder: the optimality gap tends to get worse and fewer branch-and-bound nodes are

[4]The best incumbent value.

Table 6: Performance of formulations SD (8) and VF (9) to solve the NDP to minimize the total travel time

Budget	Demand	Formulation	UB	(Rel.)	LB	Gap	# B&B Nodes
5	Low	Greedy	1,424.72		-	-	-
		SD	1,414.09	(-0.75 %)	1,376.14	2.76 %	1713
		VF	1,406.94	(-1.25 %)	1,376.25	2.23 %	3724
	High	Greedy	2,657.46		-	-	-
		SD	2,605.22	(-1.97 %)	2,528.32	3.04 %	987
		VF	2,581.72	(-2.85 %)	2,529.71	2.06 %	2776
10	Low	Greedy	1,399.55		-	-	-
		SD	1,385.52	(-1.00 %)	1,350.34	2.60 %	1576
		VF	1,379.23	(-1.45 %)	1,350.28	2.14 %	4300
	High	Greedy	2,563.26		-	-	-
		SD	2,515.46	(-1.86 %)	2,431.67	3.45 %	1053
		VF	2,486.97	(-2.98 %)	2,432.16	2.25 %	3050
20	Low	Greedy	1,351.52		-	-	-
		SD	1,348.91	(-0.19 %)	1,319.55	2.22 %	1771
		VF	1,344.84	(-0.49 %)	1,319.57	1.91 %	3743
	High	Greedy	2,416.45		-	-	-
		SD	2,398.83	(-0.73 %)	2,313.38	3.69 %	1198
		VF	2,371.61	(-1.86 %)	2,313.93	2.49 %	2756

explored. The budget seems to have little impact on the performance of the method. It is interesting to note that the worst cost of sustainability is harder to optimize than the total cost of sustainability, which is harder than the total travel time. On any setup, we explore the fewest branch-and-bound nodes and tend to have the worst optimality gap when optimizing the worst cost of sustainability.

Figure 4 shows the total cost of sustainability before and after capacity expansion, as suggested by the greedy method and formulation SD (13). Unsurprisingly, in both demand scenarios, the total cost of sustainability decreases after capacity expansion. Furthermore, a larger budget results in greater improvements. In each setup, formulation SD (13) produces a solution (i.e., a capacity expansion plan) that leads to a larger improvement compared to the greedy method.

A similar trend is observed for the worst cost of sustainability, as shown in Figure 5. Formulation SD, which optimizes the worst cost of sustainability (14), produces a solution with a smaller worst cost of sustainability than the greedy method. Interestingly, the marginal improvement in both total and worst costs of sustainability remains significant even with the largest budget of 20. This suggests that increasing the budget beyond 20 could likely result in further major improvements in these metrics.

Figures 6, 7 and 8 show the best solutions (i.e., the locations of fast charging stations for capacity expansion) found within a one-hour time limit by formulation SD to minimize the total travel time (8), the total cost of sustainability (13), and the worst cost of sustainability (14). These figures are generated using the low-demand scenario with a budget of 5. However, the figures illustrate that changing the objective function leads to different solutions, emphasizing the importance of selecting the appropriate objective function. It should be noted that the solutions obtained within the time limit are not proven to be optimal, and extending the time limit could lead to improvements.

We emphasize the importance of formulations SD (13) and (14). The total travel time experienced by all EV drivers is one metric to measure the quality of service provided by the fast charging station operator. However, from each user's perspective, arguably, they are more interested in the time they personally experience, rather than the total travel time experienced by all EV drivers. In this regard, our formulations (13) and (14) optimize metrics relevant to each driver's experience. Last but not least,

Table 7: Performance of formulation SD (13) to solve the NDP to minimize the total cost of sustainability

Budget	Demand	Formulation	UB	(Rel.)	LB	Gap	# B&B Nodes
5	Low	Greedy	173.37		-	-	-
		SD	169.77	(-2.08 %)	142.90	18.80 %	3074
	High	Greedy	462.55		-	-	-
		SD	453.59	(-1.94 %)	339.59	33.57 %	1755
10	Low	Greedy	163.79		-	-	-
		SD	157.51	(-3.83 %)	130.24	20.94 %	2720
	High	Greedy	426.30		-	-	-
		SD	404.35	(-5.15 %)	298.91	35.27 %	1618
20	Low	Greedy	145.36		-	-	-
		SD	144.08	(-0.88 %)	119.33	20.75 %	2897
	High	Greedy	370.35		-	-	-
		SD	360.21	(-2.74 %)	261.76	37.61 %	1881

Table 8: Performance of formulation SD (14) to solve the NDP to minimize the worst cost of sustainability

Budget	Demand	Formulation	UB	(Rel.)	LB	Gap	# B&B Nodes
5	Low	Greedy	0.56		-	-	-
		SD	0.55	(-1.85 %)	0.44	26.44 %	840
	High	Greedy	1.02		-	-	-
		SD	1.02	(-0.36 %)	0.73	38.95 %	729
10	Low	Greedy	0.56		-	-	-
		SD	0.51	(-8.95 %)	0.40	28.14 %	889
	High	Greedy	0.98		-	-	-
		SD	0.91	(-6.93 %)	0.64	43.54 %	668
20	Low	Greedy	0.50		-	-	-
		SD	0.50	(0.00 %)	0.36	38.48 %	833
	High	Greedy	0.90		-	-	-
		SD	0.86	(-4.07 %)	0.53	62.79 %	736

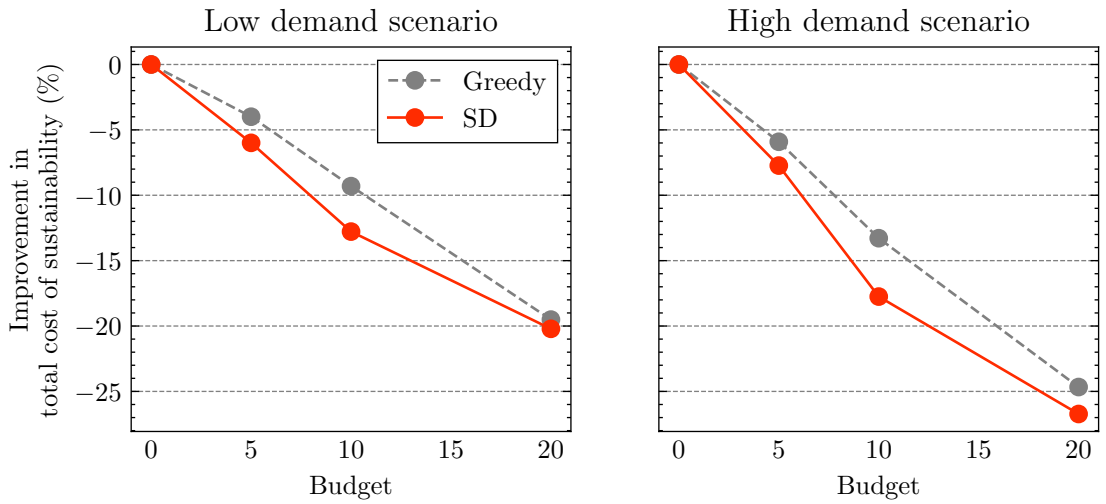

Figure 4: Improvement in the total cost of sustainability after capacity expansion suggested by the greedy method and formulation SD (13)

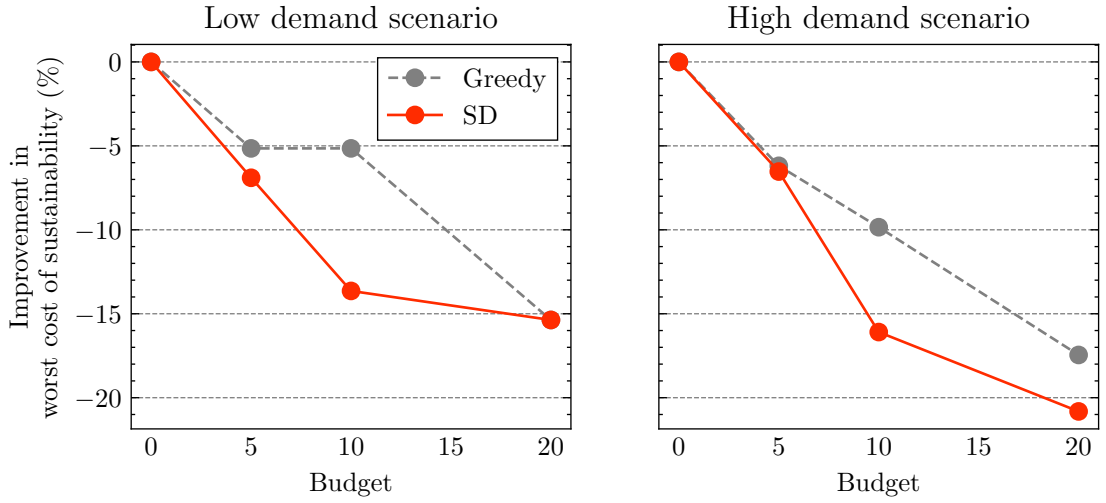

Figure 5: Improvement in the worst cost of sustainability after capacity expansion suggested by the greedy method and formulation SD (14)

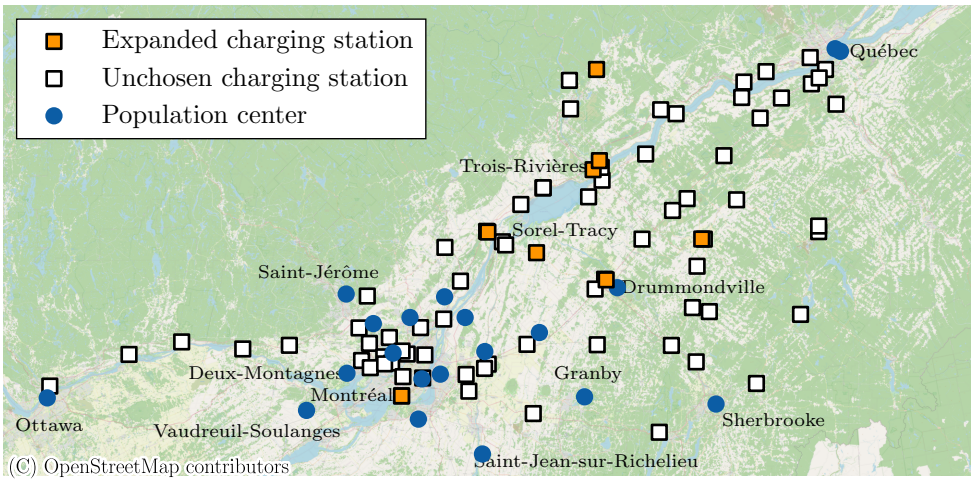

Figure 6: Selected charging stations in the low-demand scenario with a budget of 5 to optimize the total travel time

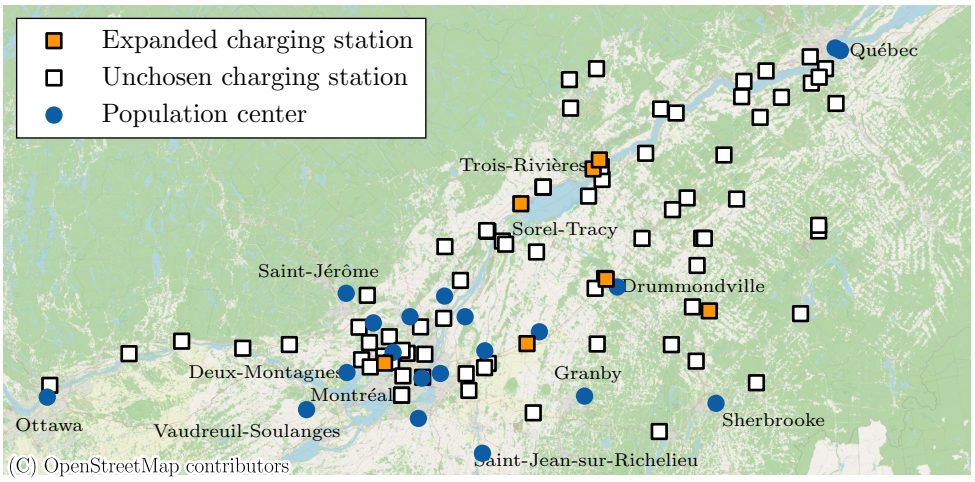

Figure 7: Selected charging stations in the low-demand scenario with a budget of 5 to optimize the total cost of sustainability

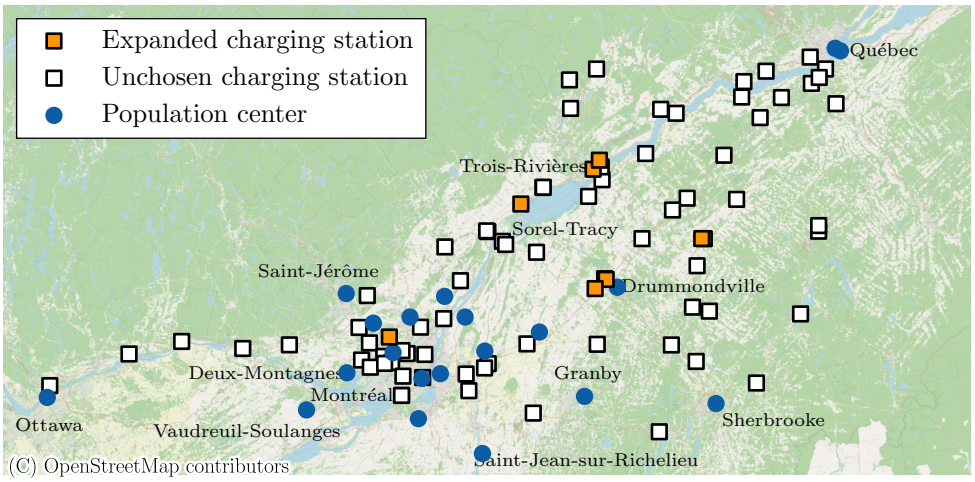

Figure 8: Selected charging stations in the low-demand scenario with a budget of 5 to optimize the worst cost of sustainability

while we focused on the cost of sustainability in this experiment (i.e., the difference between an EV driver and a conventional car driver with the same origin and destination), it is straightforward to modify the formulation to consider the absolute (rather than relative to conventional car drivers) time experienced by EV drivers.

We refer the reader to Appendix E for additional analyses, including a detailed comparison of how the best solutions found by our methodologies perform across the three considered objectives. This provides further insight into how decision-makers can use the proposed methods to develop prioritization strategies.

7 Conclusions and Future work

In this paper, we studied single-level reformulations of the network design problem. One of the reformulations was based on strong duality of the traffic assignment problem that characterizes the user equilibrium, while the other was based on the value function. Both reformulations were convex MINLPs that can be solved with a standard solution method. Furthermore, the reformulation based on strong duality introduced additional variables, providing the flexibility to formulate different objective functions or constraints. As examples, we presented the application of our reformulation to optimize the total and worst free-flow unfairness. Numerical experiments using test transportation networks were presented to analyze the performances of our reformulations. Furthermore, we demonstrated that a capacity expansion problem for electric vehicle fast-charging stations can be modeled as the network design problem. We compared the solution found by our reformulations to that of a greedy heuristic and showed improvements in the objectives, including total travel time and the cost of sustainability, i.e., the unfairness caused by the adoption of electric vehicles.

The reformulation based on strong duality assumes that the travel time function follows the BRP function. It also assumes that the OD pairs are connected, even in the absence of candidate arcs. It would be valuable to extend our method to handle more general instances with relaxed assumptions. Furthermore, in our case study on fast-charging stations, we focused primarily on capacity expansion decisions. The siting of new charging stations can be modeled either with an SOS1 constraint or by extending the logic to compute a valid value of big-M, which is left as future research. Additionally, developing a faster, decomposition-based solution method to solve our reformulation more efficiently is of interest.

A Comparison of Aggregated and Disaggregated Models

In this section, we compare the performance of the disaggregated model (formulation with disaggregated flow variable y) and the aggregated model (formulation with aggregated flow variable z) (8).

The arguments in Sections 3 and 4 can be extended to use disaggregated flow variable y. For example, we obtain a formulation of the NDP corresponding to formulation (8) but with disaggregated flow variable

y as follows:

$$\min_{x,y,\pi,\eta} \sum_{a\in A}\left(\sum_{k\in K} y_{ka}\right)\theta_a\left(\sum_{k\in K} y_{ka}\right) \tag{15}$$

$$\text{s.t.} \sum_{a\in A}\int_0^{\sum_{k\in K} y_{ka}} \theta_a(s)ds$$

$$\leq \sum_{k\in K} e_k(\pi_{kd_k} - \pi_{ko_k}) - \sum_{a\in A} \frac{p}{g_a^{1/p}(p+1)}\eta_a^{\frac{p+1}{p}},$$

$$\sum_{a\in A_v^+} y_{ka} - \sum_{a\in A_v^-} y_{ka} = e'_{kv}, \qquad \forall k\in K, v\in V,$$

$$y_{ka} \leq e_k x_a, \qquad \forall k\in K, a\in A,$$

$$\eta_a \geq \pi_{kt_a} - \pi_{kh_a} - f_a - M'_{ka}(1-x_a), \qquad \forall k\in K, a=(h,t)\in A,$$

$$\eta_a = 0, \qquad a\in A: g_a = 0,$$

$$y\geq 0, \eta\geq 0, x\in X,$$

where M' is a sufficiently large constant, which can be computed using a routine similar to Proposition 2. Analogously, we can modify formulation (9) to use disaggregated flow variable y as

$$\min_{x,y} \sum_{a\in A}\left(\sum_{k\in K} y_{ka}\right)\theta_a\left(\sum_{k\in K} y_{ka}\right) \tag{16}$$

$$\text{s.t.} \sum_{a\in A}\int_0^{\sum_{k\in K} y_{ka}} \theta_a(s)ds \leq \phi(x') + \hat{\phi}(x')\sum_{a\in A} x'_a(1-x_a), \quad \forall x'\in X,$$

$$\sum_{a\in A_v^+} y_{ka} - \sum_{a\in A_v^-} y_{ka} = e'_{kv}, \qquad \forall k\in K, v\in V,$$

$$y_{ka} \leq e_k x_a, \qquad \forall k\in K, a\in A,$$

$$y\geq 0, x\in X.$$

For each of the 10 test instances using SiouxFalls transportation network, we measure the performances of the four formulations (8), (15), (9) and (16). Table 9 shows the number of test instances solved within the two-hour time limit and the average computational time. The disaggregated formulations are always worse than the aggregated correspondents on average, in terms of the number of instances solved within the timelimit and the average computational time.

Table 9: Performances of disaggregated and aggregated models

Formulation	Aggregation	Equation	Solved	Time
SD	No	(15)	1	3,428.6
	Yes	(8)	10	217.0
VF	No	(16)	6	2,305.8
	Yes	(9)	10	558.8

B Dual of Traffic Assignment Problem

In this section, we derive the dual traffic assignment problem (6). We start with a lemma.

Lemma 1. *Let $p \geq 1$, $a \geq 0$ and $b \in \mathbb{R}^n$. For $z \in \mathbb{R}^n_+$, define*

$$f(z) = \frac{a}{p+1}\left(\sum_{i=1}^n z_i\right)^{p+1} - \sum_{i=1}^n b_i z_i.$$

1. *If $a > 0$ then*

$$\min_{z \in \mathbb{R}^n_+} f(z) = -\frac{p}{a^{1/p}(p+1)}\Big([\max\{b_i : i = 1, \ldots, n\}]_+\Big)^{(p+1)/p},$$

   where $[x]_+ = \max\{x, 0\}$.

2. *If $a = 0$, then*

$$\min_{z \in \mathbb{R}^n_+} f(z) = \begin{cases} 0, & \text{if } b_i \leq a \text{ for all } i = 1, \ldots, n, \\ -\infty, & \text{otherwise.} \end{cases}$$

In particular, Lemma 1 implies that

$$\min_{z \geq 0} f(z) = \begin{cases} \max\limits_{\eta \geq 0}\left\{-\dfrac{p}{\bar{a}^{1/p}(p+1)}\eta^{(p+1)/p} : \eta \geq b_i,\ \forall i = 1, \ldots, n\right\}, & \text{if } a > 0, \\ \max\limits_{\eta \geq 0}\left\{-\dfrac{p}{\bar{a}^{1/p}(p+1)}\eta^{(p+1)/p} : \eta = 0, \eta \geq b_i,\ \forall i = 1, \ldots, n\right\}, & \text{otherwise,} \end{cases} \tag{17}$$

where $\bar{a} = a$ if $a > 0$ and otherwise $\bar{a} = 1$.

Proof. It is straightforward to show the case when $a = 0$. In the following, we assume $a > 0$. Pick i' such that $b_{i'} \geq b_i$ for any $i = 1, \ldots, n$. For every $z \in \mathbb{R}^n_+$, we have $f(z) \geq f(z')$, where

$$z'_i = \begin{cases} \sum_{i=1}^n z_i, & i = i', \\ 0, & i \neq i'. \end{cases}$$

It follows that

$$\min_{z \in \mathbb{R}^n_+} f(z) = \min_{u \in \mathbb{R}_+} g(u),$$

where

$$g(u) = \frac{a}{p+1}u^{p+1} - b_{i'}u.$$

By computing the derivative of g, we can obtain the minimizer as

$$\min_{u \in \mathbb{R}_+} g(u) = g\left(\left(\frac{[b_{i'}]_+}{a}\right)^{1/p}\right) = -\frac{p}{a^{1/p}(p+1)}\Big([b_{i'}]_+\Big)^{(p+1)/p}.$$

□

Now, we derive the dual of problem (4). Fix x. Let π be the dual variable corresponding to con-

straint (4b). Then, the Lagrangian is

$$\begin{aligned}
L(z,\pi) &= \sum_{a\in A(x)} \int_0^{\sum_{o\in O} z_{oa}} \theta_a(s)ds - \sum_{o\in O}\sum_{v\in V} \pi_{ov}\left(\sum_{a\in A_v^+(x)} z_{oa} - \sum_{a\in A_v^-(x)} z_{oa} - e_{ov}\right) \\
&= \sum_{a\in A(x)} \left(f_a\left(\sum_{o\in O} z_{oa}\right) + \frac{g_a}{p+1}\left(\sum_{o\in O} z_{oa}\right)^{p+1}\right) \\
&\qquad - \sum_{o\in O}\sum_{a=(h,t)\in A(x)} (\pi_{ot}-\pi_{oh}) z_{oa} + \sum_{k\in K} e_k(\pi_{o_k d_k} - \pi_{o_k o_k}) \\
&= \sum_{a=(h,t)\in A(x)} \left(\frac{g_a}{p+1}\left(\sum_{o\in O} z_{oa}\right)^{p+1} - \sum_{o\in O}(\pi_{ot}-\pi_{oh}-f_a) z_{oa}\right) + \sum_{k\in K} e_k(\pi_{o_k d_k} - \pi_{o_k o_k}) \\
&= \sum_{a\in A(x)} L_a(z_a,\pi_a) + \sum_{k\in K} e_k(\pi_{o_k d_k} - \pi_{o_k o_k}),
\end{aligned}$$

where

$$L_a(z_a,\pi_a) = \frac{g_a}{p+1}\left(\sum_{o\in O} z_{oa}\right)^{p+1} - \sum_{o\in O}(\pi_{ot}-\pi_{oh}-f_a) z_{oa}.$$

The Lagrangian dual problem is

$$\max_{\pi}\min_{z\geq 0} L(z,\pi) = \max_{\pi}\left\{ \sum_{a\in A(x)} \min_{z\geq 0} L_a(z_a,\pi_a) + \sum_{k\in K} e_k(\pi_{o_k d_k} - \pi_{o_k o_k})\right\}.$$

In light of Lemma 1 and equation (17), this problem is equivalent to problem (6).

C Uniqueness of Equilibrium Travel Time

As stated in Section 3, the optimistic formulation and the pessimistic formulation of the NDP (3) coincide. In other words, if there are multiple user equilibria, all of them have the same total travel time. This is exploited by Roughgarden and Tardos [2000], but we also provide the proof for completeness.

Proposition 3. *Suppose Assumption 1 holds. For any $x \in X$ and $z(x), z(x)' \in Z(x)$, we have*

$$\sum_{a\in A}\left(\sum_{o\in O} z_{oa}(x)\right)\ \theta_a\left(\sum_{o\in O} z_{oa}(x)\right) = \sum_{a\in A}\left(\sum_{o\in O} z'_{oa}(x)\right)\ \theta_a\left(\sum_{o\in O} z'_{oa}(x)\right).$$

Proof. Pick any $x \in X$ and $\pi(x) \in \Pi(x)$. For any $o \in O$ and $z(x) \in Z(x)$, by summing (5c) for all $a \in A(x)$, we obtain:

$$\sum_{a\in A(x)} z_{oa}(x)\theta_a\left(\sum_{o'\in O} z_{o'a}(x)\right) = \sum_{v\in V}\left(\sum_{a\in A_v^+(x)} z_{oa}(x) - \sum_{a\in A_v^-(x)} z_{oa}(x)\right)\pi_{ov}(x) = \sum_{v\in V} e_{ov}\pi_{ov}(x).$$

Therefore, for any $z(x) \in Z(x)$, the total travel time is

$$\sum_{a\in A}\left(\sum_{o\in O} z_{oa}(x)\right)\ \theta_a\left(\sum_{o\in O} z_{oa}(x)\right) = \sum_{a\in A(x)}\left(\sum_{o\in O} z_{oa}(x)\right)\ \theta_a\left(\sum_{o\in O} z_{oa}(x)\right) = \sum_{o\in O}\sum_{v\in V} e_{ov}\pi_{ov}(x).$$

This implies the desired result. □

D Outer approximation methods

In this section, we present the outer-approximation method used to solve formulation (9). The method to solve formulation (8) is similar.

Using the definition of the travel cost function (1), we can rewrite formulation (9) as

$$
\begin{aligned}
\min_{x,z,\theta} \ & \sum_{a\in A} f_a\left(\sum_{o\in O} z_{oa}\right) + g_a\theta_a \\
\text{s.t.} \ & \sum_{a\in A} f_a\left(\sum_{o\in O} z_{oa}\right) + \frac{g_a}{p+1}\theta_a \le \phi(x') + \hat{\phi}(x')\sum_{a\in A} x'_a(1-x_a), && \forall x' \in X, \\
& \sum_{a\in A_v^+} z_{oa} - \sum_{a\in A_v^-} z_{oa} = e_{ov}, && \forall o\in O, v\in V, \\
& z_{oa} \le \sum_{k\in K_o} e_k x_a, && \forall o\in O, a\in A, \\
& \theta_a \ge \left(\sum_{o\in O} z_{oa}\right)^{p+1} \\
& z \ge 0, x\in X.
\end{aligned}
$$

Let us define

$$
\begin{aligned}
u(X',I') = \min_{x,z,u} \ & \sum_{a\in A} f_a\left(\sum_{o\in O} z_{oa}\right) + g_a\theta_a \\
\text{s.t.} \ & \sum_{a\in A} f_a\left(\sum_{o\in O} z_{oa}\right) + \frac{g_a}{p+1}\theta_a \le \phi(x') + \hat{\phi}(x')\sum_{a\in A} x'_a(1-x_a), && \forall x' \in X', \\
& \sum_{a\in A_v^+} z_{oa} - \sum_{a\in A_v^-} z_{oa} = e_{ov}, && \forall o\in O, v\in V, \\
& z_{oa} \le \sum_{k\in K_o} e_k x_a, && \forall o\in O, a\in A, \\
& \theta_a \ge \alpha^{p+1} + (p+1)\alpha^p\left(\sum_{o\in O} z_{oa} - \alpha\right), && \forall a\in A, \alpha\in I'_a, \\
& z \ge 0, x\in X.
\end{aligned}
$$

Problem $u(X', I')$ is equivalent to problem (9) if $X' = X$ and $I'_a = \mathbb{R}$ for all $a \in A$. However, if $X' \subsetneq X$ or $I'_a \neq \mathbb{R}$ for some $a \in A$, $u(X', I')$ is a relaxation of problem (9). In our implementation, we let $X' = I'_a = \emptyset$ for all $a \in A$ and pass $u(X', I')$ to Gurobi. Every time Gurobi solves a branch-and-bound node or finds a candidate integer solution, we search for a violated constraint (up to a tolerance of 10^{-6}), and add it as a lazy constraint.

E Additional Analyses

In this section, we provide additional analyses on the experiments in Section 6.

E.1 Histogram of Demands

Figure 9 shows the histogram of demand volumes. The histogram uses the low-demand scenario. However, the demand volumes for the high-demand scenario are obtained by scaling the low-demand volumes by 50%. Thus, the shape of the histogram for the high-demand scenario remains the same.

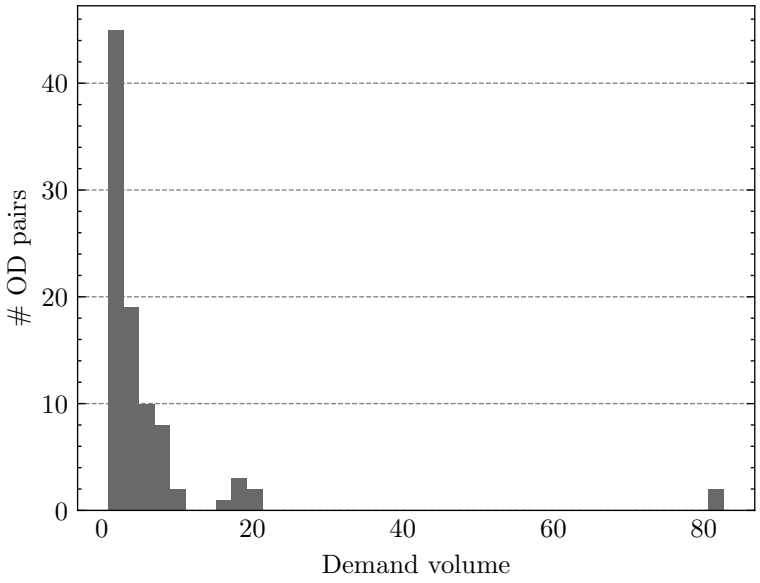

Figure 9: Histogram of the demands volumes in the low-demand scenario

E.2 Comparison of Equilibria

Figures 10 and 11 show the total travel time and the total/worst costs of sustainability of the equilibria computed by formulation SD to optimize the total travel time (8), the total cost of sustainability (13) and the worst cost of sustainability (14). The color and shading pattern of each marker indicates the metric being optimized, while the shape of the marker corresponds to the budget: a square, triangle, and circle represents a budget of 5, 10, and 20, respectively.

For larger budgets, optimizing for one objective tends to yield near-optimal values for the other objectives as a byproduct. However, for smaller budgets, differences start to emerge, highlighting the need for decision-makers in these cases to carefully choose their priorities.

E.3 Comparison of Charging Station Locations for Capacity Expansion

Figures 12 and 13 show the selected charging stations for capacity expansion in the two demand scenarios, with a budget of 5. These figures are generated using the outputs of formulation SD, which optimizes the total travel time (8). The outputs differ between demand scenarios, indicating the sensitivity of the results to the demand data.

Acknowledgments

This research was supported by Hydro-Québec, NSERC Collaborative Research and Development Grant CRDPJ 536757 - 19, and the FRQ-IVADO Research Chair in Data Science for Combinatorial Game Theory.

References

Okan Arslan, Oya Ekin Karaşan, A. Ridha Mahjoub, and Hande Yaman. A branch-and-cut algorithm for the alternative fuel refueling station location problem with routing. *Transportation Science*, 53(4): 1107–1125, 2019. ISSN 0041-1655. doi: 10.1287/trsc.2018.0869.

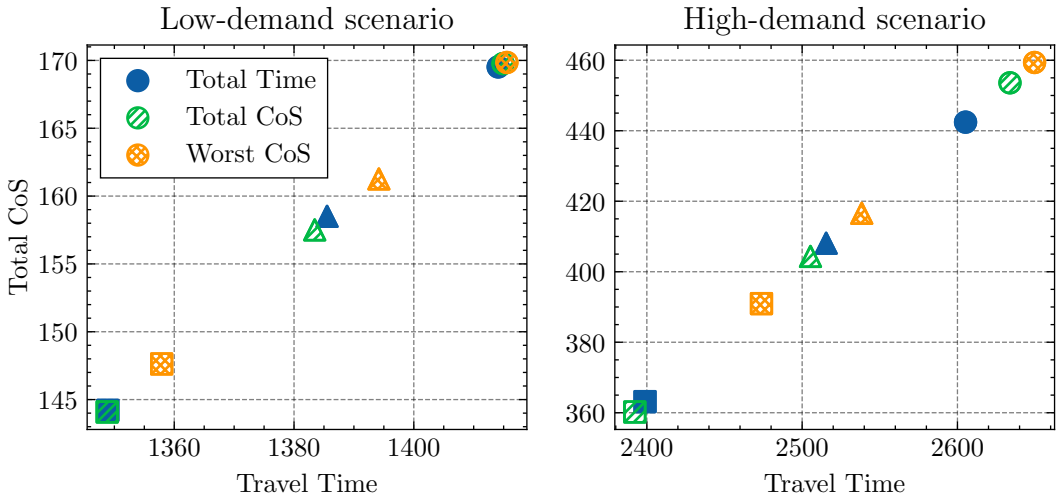

Figure 10: Total travel time and total cost of sustainability of the user equilibriums computed at various equilibria

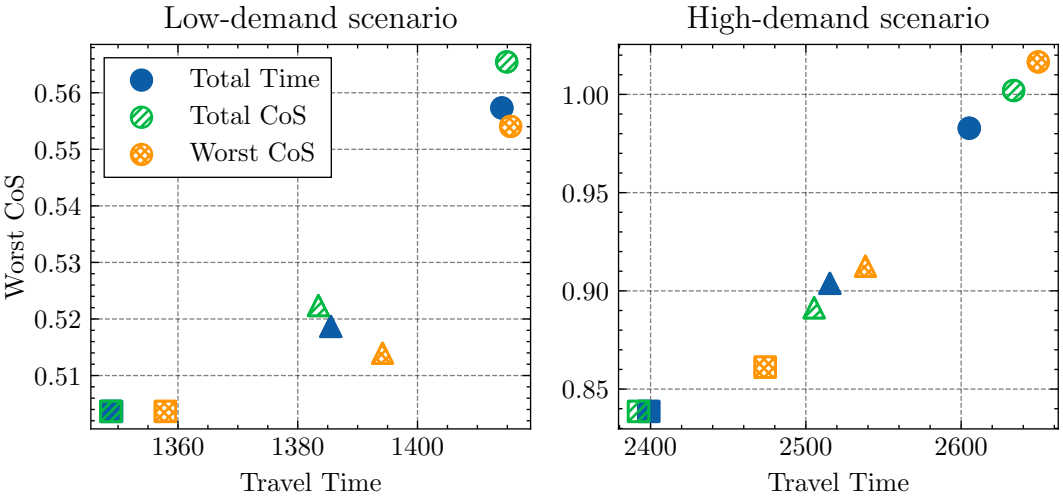

Figure 11: Total travel time and worst cost of sustainability of the user equilibriums computed at various equilibria

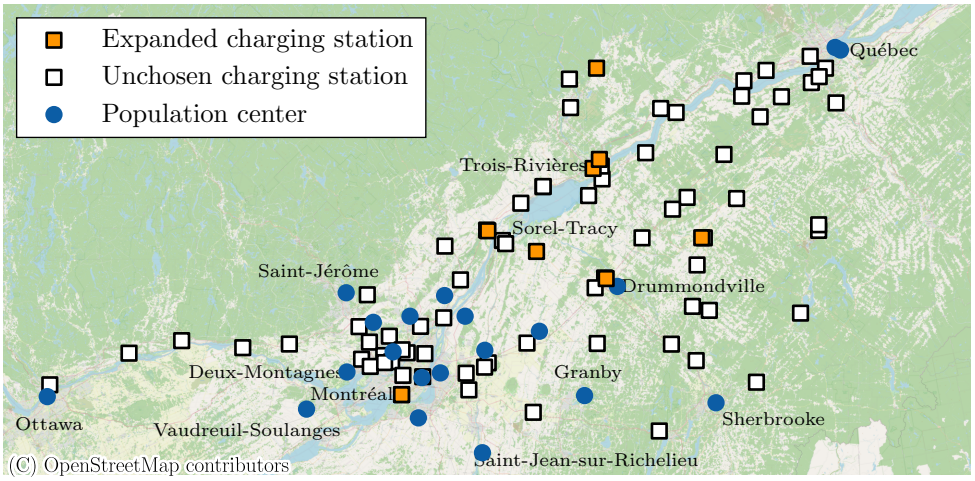

Figure 12: Selected charging stations in the low-demand scenario with a budget of 5

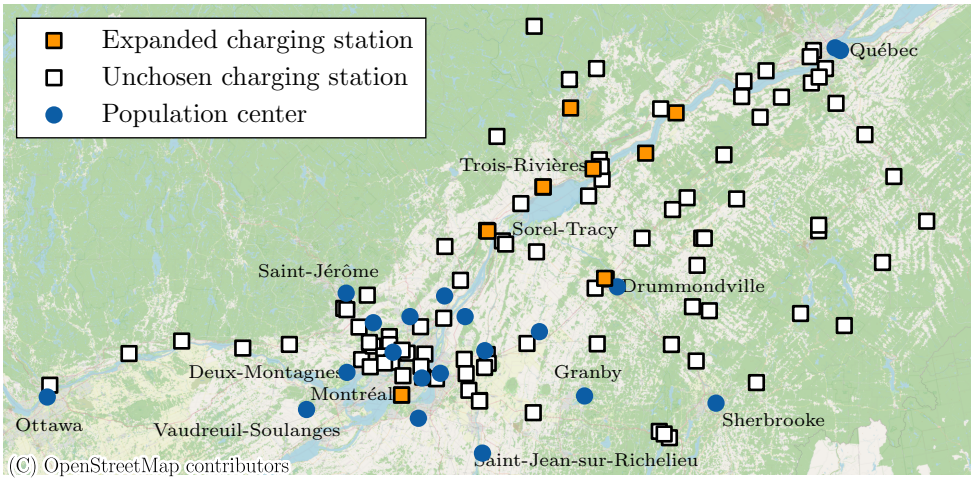

Figure 13: Selected charging stations in the high-demand scenario with a budget of 5

Saeed Asadi Bagloee, Majid Sarvi, and Michael Patriksson. A hybrid branch-and-bound and benders decomposition algorithm for the network design problem. *Computer-Aided Civil and Infrastructure Engineering*, 32(4):319–343, 2017.

Martin J. Beckmann, C. B. McGuire, and C. B. Winsten. *Studies in the Economics of Transportation*. RAND Corporation, Santa Monica, CA, 1955.

G. Boeing. Modeling and analyzing urban networks and amenities with osmnx. Working paper, University of Southern California, Munich, 2024.

Pierre Bonami, Mustafa Kilinç, Jeff Linderoth, Sven Leyffer, and Jon Lee. Algorithms and software for convex mixed integer nonlinear programs. In *Mixed Integer Nonlinear Programming*, The IMA Volumes in Mathematics and its Applications, pages 1–39. Springer New York, New York, NY, 2012. ISBN 1461419263.

Mervat Chouman, Teodor Gabriel Crainic, and Bernard Gendron. Commodity representations and cut-set-based inequalities for multicommodity capacitated fixed-charge network design. *Transportation Science*, 51(2):650–667, 2017. ISSN 0041-1655.

Hamid Farvaresh and Mohammad Mehdi Sepehri. A single-level mixed integer linear formulation for a bi-level discrete network design problem. *Transportation Research Part E: Logistics and Transportation Review*, 47(5):623–640, 2011.

Hamid Farvaresh and Mohammad Mehdi Sepehri. A branch and bound algorithm for bi-level discrete network design problem. *Networks and Spatial Economics*, 13(1):67–106, 2013a. ISSN 1566-113X.

Hamid Farvaresh and Mohammad Mehdi Sepehri. A branch and bound algorithm for bi-level discrete network design problem. *Networks and Spatial Economics*, 13:67–106, 2013b.

Pirmin Fontaine and Stefan Minner. Benders decomposition for discrete–continuous linear bilevel problems with application to traffic network design. *Transportation Research Part B: Methodological*, 70: 163–172, 2014.

Masao Fukushima. On the dual approach to the traffic assignment problem. *Transportation Research Part B: Methodological*, 18(3):235–245, 1984. ISSN 0191-2615. doi: https://doi.org/10.1016/0191-2615(84)90034-1.

M. John Hodgson. A flow-capturing location-allocation model. *Geographical Analysis*, 22(3):270–279, 1990. doi: https://doi.org/10.1111/j.1538-4632.1990.tb00210.x.

Olaf Jahn, Rolf H Mohring, Andreas S Schulz, and Nicolas E Stier-Moses. System-optimal routing of traffic flows with user constraints in networks with congestion. *Operations Research*, 53(4):600–616, 2005. ISSN 0030-364X.

Wentao Jing, Kun An, Mohsen Ramezani, and Inhi Kim. Location design of electric vehicle charging facilities: a path-distance constrained stochastic user equilibrium approach. *Journal of Advanced Transportation*, 2017(1):4252946, 2017.

Ömer Burak Kınay, Fatma Gzara, and Sibel A Alumur. Charging station location and sizing for electric vehicles under congestion. *Transportation Science*, 57(6):1433–1451, 2023.

Larry J Leblanc. An algorithm for the discrete network design problem. *Transportation Science*, 9(3): 183–199, 1975. ISSN 0041-1655.

T Leventhal, G Nemhauser, and Jr Trotter, L. A column generation algorithm for optimal traffic assignment. *Transportation Science*, 7(2):168–176, 1973. ISSN 0041-1655.

T. L. Magnanti and R. T. Wong. Network design and transportation planning: Models and algorithms. *Transportation Science*, 18(1):1–55, 1984. doi: 10.1287/trsc.18.1.1.

S. A. MirHassani and R. Ebrazi. A flexible reformulation of the refueling station location problem. *Transportation Science*, 47(4):617–628, 2013. ISSN 0041-1655. doi: 10.1287/trsc.1120.0430.

OpenStreetMap contributors. OpenStreetMap. `https://www.openstreetmap.org`, 2024. Accessed: 2024-10-31.

M. Patriksson. *The Traffic Assignment Problem: Models and Methods*. Dover Publications, Headquarters location, 2015. ISBN 978-0486787909.

David Rey and Michael Levin. A branch-and-price-and-cut algorithm for discrete network design problems under traffic equilibrium. *Optimization Online*, 2024. URL `https://optimization-online.org/?p=28420`.

R. T. Rockafellar. *Network Flows and Monotropic Optimization*. Athena Scientific, Belmont, Massachusetts, 1998. ISBN 1-886529-06-X.

R. T. Rockafellar, R. R. Meyer, O. L. Mangasarian, and S. M. Robinson. Monotropic programming: Descent algorithms and duality. In *Nonlinear Programming 4*, pages 327–366. Elsevier Inc, United States, 1981. ISBN 1483260178.

T. Roughgarden and E. Tardos. How bad is selfish routing? In *41st Annual Symposium on Foundations of Computer Science*, pages 93–102, 2000. doi: 10.1109/SFCS.2000.892069.

Cong Quoc Tran, Dong Ngoduy, Mehdi Keyvan-Ekbatani, and David Watling. A user equilibrium-based fast-charging location model considering heterogeneous vehicles in urban networks. *Transportmetrica A: Transport Science*, 17(4):439–461, 2021.

Shuaian Wang, Qiang Meng, and Hai Yang. Global optimization methods for the discrete network design problem. *Transportation Research Part B: Methodological*, 50:42–60, 2013. ISSN 0191-2615.

Bo Zhang, Meng Zhao, and Xiangpei Hu. Location planning of electric vehicle charging station with users' preferences and waiting time: multi-objective bi-level programming model and hnsga-ii algorithm. *International Journal of Production Research*, 61(5):1394–1423, 2023.