Transportation Research Part B 153 (2021) 70–90

Contents lists available at ScienceDirect

Transportation Research Part B

journal homepage: www.elsevier.com/locate/trb

Minimising emissions in traffic assignment with non-monotonic arc costs

J. Tidswell [a,*], A. Downward [a], C. Thielen [b], A. Raith [a]

[a] *Department of Engineering Science, University of Auckland, New Zealand*
[b] *TUM Campus Straubing for Biotechnology and Sustainability, Technical University of Munich, Germany*

ARTICLE INFO

Keywords:
Traffic assignment
Wardrop equilibrium
Emissions
Non-monotonic cost functions
Complexity analysis
Heuristics

ABSTRACT

The modelling of vehicle emissions within Traffic Assignment (TA) has been studied in the literature as emissions such as carbon monoxide and carbon dioxide are detrimental to the populace's health as well as to the environment. TA is employed as a means to identify the potential to reduce vehicle emissions by obtaining emissions-minimising traffic patterns. TA captures the flow-dependent cost to traverse an arc in a so-called arc cost function, which often captures travel time, travel cost, or emissions. Arc cost functions that model emissions are naturally non-monotonic (partly increasing and partly decreasing) with respect to arc flow. Studies that make use of emission-based arc cost functions in TA generally assume a positive, increasing function, or do not discuss the computational complexities that arise when the arc cost functions are non-monotonic. In this paper, we investigate the implications of non-monotonic arc costs within the TA methodology and address the complexity of the resulting problem. We suggest adjustments to solution algorithms to heuristically allow the computation of TA solutions with non-monotonic arc costs. We present several methods to find good solutions to the TA problem with non-monotonic arc costs in the absence of a unique emissions-minimising solution. We compare these methods by applying them to several test networks for non-monotonic arc cost functions that model different emission types.

1. Introduction

The significant contribution of greenhouse gases from vehicles has provoked interest in estimating emissions in traffic models, such as in the TA model, with the aim of identifying traffic patterns that reduce vehicle emissions. In the *static* TA model with fixed demand as used in this paper, average network parameters over a given time period are used to predict traffic patterns at a high level. While more detailed traffic models such as the *dynamic* TA model exist, they are more commonly employed for small network instances due to the increased complexity of the models.

Although emissions arc cost functions are convex functions of average speed, they are non-monotonic functions of arc flow, violating one of the standard assumptions in TA models (Raith et al., 2016; Patil, 2016). The inclusion of an emissions cost within TA is discussed in the literature to some extent. One approach uses emissions models corresponding to arc cost functions that are increasing functions of flow for the given network (Yin and Lawphongpanich, 2006), leading to standard TA problems solved using conventional algorithms. An alternative approach enforces a speed limit, which also ensures that the resulting emissions functions are increasing functions of flow (Raith et al., 2016; Patil, 2016). While the use of a speed limit is convenient and intuitive, in practice, the speed limits required to ensure an increasing arc cost function may be unenforceable and unrealistically constrain

* Correspondence to: Department of Engineering Science, The University of Auckland, Private Bag 92019, Auckland Mail Centre, Auckland 1142, New Zealand.
E-mail address: j.tidswell@auckland.ac.nz (J. Tidswell).

https://doi.org/10.1016/j.trb.2021.08.007
Received 29 September 2020; Received in revised form 13 June 2021; Accepted 20 August 2021
Available online 25 September 2021
0191-2615/

users, for instance by setting relatively low speed limits on highways. In other literature incorporating emission functions that are non-monotonic with respect to flow, the existence of multiple equilibria is noted; however, the issues that arise in the methodology when the arc costs are non-monotonic are not addressed (Benedek and Rilett, 1998; Sugawara and Niemeier, 2002).

Under the assumption of a positive, increasing arc cost function in TA, there is a unique equilibrium, with arc flows that minimise system cost. However, a non-monotonic arc cost function may result in the existence of multiple local minima. We aim to incorporate non-monotonic emission functions into TA to find solutions with minimal emission costs in traffic networks. Within the literature, the TA problem with non-monotonic arc costs is generally approached by making assumptions that render the arc cost functions non-decreasing in order to identify solutions — which cannot always be justified. In this paper, we explicitly address the challenges that arise when using non-monotonic cost functions in TA, with particular focus on identifying solutions with minimal system cost *without using speed limits*, which is not well-covered in the current literature. We formally show that identifying cost-minimising traffic patterns for the TA problem with non-monotonic arc costs is NP-hard. We propose adaptations of the *path equilibration* algorithm to find good solutions heuristically. Different approaches to applying our heuristic are explored (iterating from a solution that is optimal with respect to travel time or from a convex combination of different TA solutions, iteratively adjusting a weighting of travel time and emissions, and randomised initial solutions), and we present computational results for our algorithm on several test networks. Our results hold for TA with non-monotonic cost functions in general, but our discussion and computational experiments mainly focus on the problem with emissions cost functions that motivated the presented research.

This paper is organised as follows. The TA problem is introduced in Section 2, where the case of TA with non-monotonic arc cost functions is discussed and a brief overview of related literature is given. Section 3 shows that computing a traffic pattern with minimal system cost in the TA problem with non-monotonic arc cost functions is NP-hard. Different heuristic approaches to solve the problem, despite its challenging nature, are introduced in Section 4, based on the path equilibration algorithm for TA. The results of our computational experiments are shown in Section 5 and the paper concludes with Section 6.

2. The traffic assignment problem

The TA problem addresses the modelling of user path (or route) choice for a given road network. Users will simultaneously select routes to minimise their perceived personal travel cost, leading to a distribution of users in the road network that is referred to as an *equilibrium*, as introduced in the following.

We are given a road transport network represented by a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ and K *commodities* (or *origin–destination (OD) pairs)* $\{(s_i, t_i, d_i)\}_{i=1}^{K}$, where $s_i \in \mathcal{V}$ and $t_i \in \mathcal{V}$ denote origin and destination of commodity i, respectively, and $d_i \in \mathbb{R}^{+}$ denotes the *demand* of commodity i. We let $\mathcal{P}_i$ denote the set of all (simple) paths in $\mathcal{G}$ from s_i to t_i and write $\mathcal{P} := \cup_{i=1}^{K} \mathcal{P}_i$. To avoid trivialities, we assume that $\mathcal{P}_i \neq \emptyset$ for all i. A *path flow* h is represented by a vector of non-negative values $(h_p)_{p\in\mathcal{P}} \in \mathbb{R}^{|\mathcal{P}|}$. A path flow h is called *feasible* if $\sum_{p\in\mathcal{P}_i} h_p = d_i$ for all i. An *arc flow* is a non-negative vector $f = (f_a)_{a\in\mathcal{A}} \in \mathbb{R}^{|\mathcal{A}|}$. Every path flow h defines an arc flow f such that $f_a := \sum_{p\in\mathcal{P}:a\in p} h_p$, which we refer to as the arc flow induced by h. Here we use p as a path index but also use it to refer to the set of arcs that make up path p. An arc flow f is called *feasible* if there exists a feasible path flow h that induces f. The units of both f and h are typically the average flow rate over a given time.

Every arc a has an associated flow-dependent arc cost function $c_a(f_a)$. An example of an arc cost function is travel time $c_a(f_a) = t_a(f_a)$, which is often modelled using the so-called BPR (Bureau of Public Roads) travel time function

$$t_a(f_a) = t_a^0 \cdot \left(1 + \alpha \cdot (f_a/k_a)^{\beta}\right), \tag{1}$$

with parameters $\alpha > 0$, $\beta > 1$ (typical values are $\alpha = 0.15$ and $\beta = 4$), t_a^0 being the free-flow travel time and k_a the practical capacity (Bureau of Public Roads, 1964).

A UE solution for the TA problem, where users selfishly minimise their personal perceived cost, is a traffic pattern that satisfies Wardrop's first equilibrium principle (Wardrop, 1952):

> *The journey times on all the routes used are equal, and less than those which would be experienced by a single vehicle on any unused route.*

The principle can expressed mathematically for a feasible path flow h by the conditions

$$h_p\left(\hat{c}_p(f) - u^i\right) = 0 \quad \forall\, p \in \mathcal{P}_i,\ \forall\, i = 1, \ldots, K, \tag{2a}$$

$$\hat{c}_p(f) - u^i \geq 0 \quad \forall\, p \in \mathcal{P}_i,\ \forall\, i = 1, \ldots, K, \tag{2b}$$

where u^i represents the minimum path cost for the OD pair (s_i, t_i, d_i), and $\hat{c}_p(f)$ is the perceived path cost for path p. Letting $\hat{c}_a(f_a)$ denote the perceived cost of an arc a by users, this perceived cost for a path p is given as

$$\hat{c}_p(f) = \sum_{a\in p} \hat{c}_a(f_a). \tag{3}$$

The use of a *perceived* cost allows us to find different types of TA traffic patterns using the same solution methods by choosing an appropriate perceived arc cost function. Defining

$$\hat{c}_a(f_a) = c_a(f_a), \tag{4}$$

a traffic pattern that satisfies the conditions (2) will be a UE.

In the following, we focus on traffic patterns that adhere to Wardrop's second principle, where users behave cooperatively (Wardrop, 1952), often referred to as SO (equilibrium) solutions. We focus on SO solutions since these traffic patterns lead to a minimised total (system-wide) cost for a network and, thus, allow us to identify the lowest emissions that can be achieved. A UE solution with an emissions objective, on the other hand, would assume that travellers are aware of their emissions and choose routes with the sole focus of minimising these emissions, which may not be a realistic assumption.

The cooperative behaviour of users from an SO solution minimises the total (system-wide) cost for a network, which is defined as

$$C(f) = \sum_{a\in\mathcal{A}} f_a c_a(f_a). \tag{5}$$

The cooperative behaviour that produces an SO solution can also be captured by conditions (2) by altering the perceived cost to

$$\hat{c}_a(f_a) = c_a(f_a) + f_a c_a'(f_a), \tag{6}$$

where the term $f_a c_a'(f_a)$ ensures that users consider their effect on other users' cost on arc a (Sheffi, 1984). A traffic pattern that satisfies the conditions (2) with a perceived cost of (6) will produce an SO solution.

The optimisation formulation for the TA problem is convenient for establishing the existence and uniqueness of solutions. An SO solution for the TA problem can be obtained by solving the following optimisation problem:

$$\min \quad \sum_{a\in\mathcal{A}} f_a c_a(f_a) \tag{7a}$$

$$\text{s.t.} \quad \sum_{p\in\mathcal{P}_i} h_p = d_i, \qquad i = 1,\dots,K \tag{7b}$$

$$f_a = \sum_{p\in\mathcal{P}:a\in p} h_p, \qquad a\in\mathcal{A} \tag{7c}$$

$$h_p \geq 0, \qquad p\in\mathcal{P}. \tag{7d}$$

A UE solution can also be found as an optimal solution of (7) by adjusting the objective function (7a) to $\min \sum_{a\in\mathcal{A}} \int_0^{f_a} c_a(x)dx$. In the following, we will refer to both types of TA solutions as equilibria. The following theorem clarifies under what conditions an equilibrium solution (2) exists.

Theorem 1. *An equilibrium for the (static) TA problem with fixed demand exists given that the network $\mathcal{G}$ is strongly connected (with non-negative demands d_i for $i = 1,\dots,K$) and that, for each arc a, the function $\hat{c}_a(f_a)$ is positive and continuous for all $f_a \in [0,\infty)$.*

The proof of Theorem 1 is given in Patriksson (2015, Section 2.3.1). From this theorem, we can see that, provided $c_a(f_a)$ is positive and continuous, a UE solution exists. It is sufficient that $c_a(f_a)$ is non-decreasing to ensure that the SO solution exists, as the function in Eq. (6) is then also positive and continuous.

When an equilibrium solution for the TA problem exists, there may be many such equilibria. Multiple equilibria may be present due to the properties of the cost functions. A total cost can be calculated for each equilibrium, with each equilibrium corresponding to a stationary point in the equivalent optimisation problem (7). Uniqueness can be established according to the following theorem.

Theorem 2. *A unique equilibrium for the (static) TA problem with fixed demand exists given that the network is strongly connected, the demands d_i for $i = 1,\dots,K$ are positive, and that, for each arc a, the function $\hat{c}_a(f_a)$ is positive, continuous, and strictly increasing for $f_a \in [0,\infty)$.*

The proof of Theorem 2 is given in Patriksson (2015, Section 2.3.2). It is additionally shown that the equilibrium arc costs will be unique if the arc cost function is non-decreasing. In the optimisation formulation of the SO problem (7), if $c_a(f_a)$ is positive, continuous, and strictly increasing for $f_a \in [0,\infty)$, then the SO problem is convex as the Hessian is positive definite (Sheffi, 1984).

2.1. TA with non-monotonic arc costs

Arc cost functions in TA are typically assumed to be increasing or at least non-decreasing. A non-monotonic arc cost function $c_a(f_a)$, however, might be increasing with respect to flow in some parts of its domain and decreasing in others, which violates this common assumption. We investigate the use of non-monotonic arc cost functions in the context of modelling emissions.

There exist various means of modelling vehicle emissions, where common models are functions of average speed that are strictly convex and 'u-shaped' (Sugawara and Niemeier, 2002; Song et al., 2013; Patil, 2016). They can be given as

$$e_a(v_a) = \mathfrak{a}/v_a + \mathfrak{b} + \mathfrak{c}\, v_a + \mathfrak{d}\, v_a^2 \qquad \text{for } 0 < v_a \leq v_a(0), \tag{8}$$

with speed v_a and appropriately chosen parameters $\mathfrak{a}, \mathfrak{b}, \mathfrak{c}, \mathfrak{d}$ according to emission type and vehicle fleet composition to give emissions in g/km (Song et al., 2013). Here, $v_a(0)$ is the average free-flow speed on arc a, the highest speed that can be attained on arc a. Assuming a travel-time function such as in Eq. (1), average speed can be calculated as

$$v_a(f_a) = {}^{s_a}/_{t_a(f_a)}, \tag{9}$$

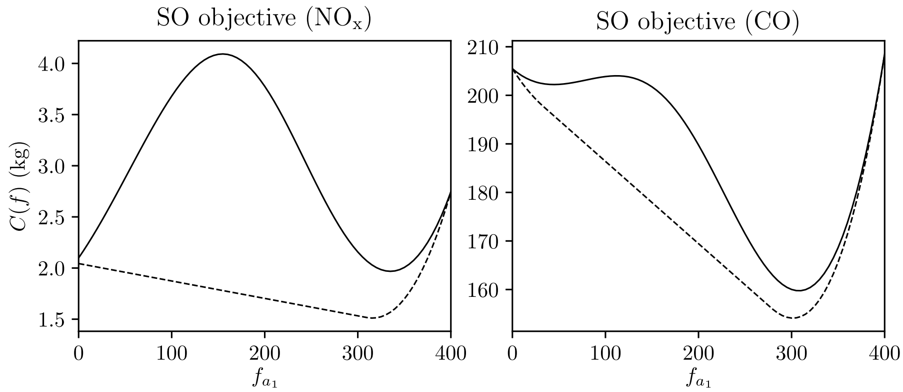

Fig. 1. Total system costs for the network with 2 parallel arcs in Example 2.1, where the dotted line represents the total-cost function (7a) with an emission-minimising speed limit, and the solid line without a speed limit. The total cost $C(f)$ is shown as a function of the flow f_{a_1} on arc a_1 (where $f_{a_2} = 400 - f_{a_1}$).

where s_a is the length of arc a in km, and $t_a(f_a)$ is travel time in hours. Emissions can then be expressed in terms of f_a via

$$e_a(f_a) = e_a(v_a(f_a)), \tag{10}$$

where $e_a(f_a)$ is non-monotonic.

Previous work (Raith et al., 2016; Patil, 2016) has ensured the existence of a unique total-cost objective value of (7) by using an increasing emissions function by incorporating a speed limit $v_a^{\max}$, such that $\bar{v}_a(f_a) = \min\{v_a(f_a), v_a^{\max}\}$, giving the emission rate with an enforced speed limit as

$$\bar{e}_a(f_a) = e_a(\bar{v}_a(f_a)). \tag{11}$$

Setting the speed limit to be at an emission-minimising speed $v_a^{\max} = \arg\min_{v_a} e_a(v_a)$, we can find a lower bound for the SO TA problem with emissions costs. When the emissions function is identical for all arcs (as is the case in the problem instances in our computational experiments), the emission-minimising speed limit $v_a^{\max}$ is the same across the network.

The total cost $C(f)$ (see Eq. (5)) for the optimisation problem (7) with $c_a(f_a) = s_a e_a(f_a)$ (without speed limits) may not be convex (or even pseudo- or quasi-convex), as Example 2.1 below illustrates. Hence, there may be locally optimal solutions of (7) with different total costs, each corresponding to a different equilibrium solution according to conditions (2) with cost functions (6) (Roughgarden, 2002, Remark B.1.4).

We wish to find solutions to the SO TA problem with non-monotonic arc costs, which we refer to as the SO-NM problem. We are specifically interested in finding solutions to the SO-NM problem in the context of emissions, which we refer to as the SO-E problem, where 'good' solutions to this problem have low total emissions. These solutions can inform transport planners about the minimum system-wide emissions that could be achieved without imposing measures such as speed limits in the transport system.

A related problem is the UE TA problem with non-monotonic arc costs denoted by UE-NM, which is not a focus of our research as it is not well motivated in the context of emissions, since it is unlikely that users' route choice would solely depend on a desire to minimise emissions. In the following, we focus on SO-NM but occasionally remark on UE-NM as, for instance, heuristics discussed here for SO-NM can also be used for UE-NM.

We show that, even in a simple network such as in Example 2.1, multiple local minima of the total-cost function may exist for the SO-NM problem. The specific local minimum found for the SO-NM problem will depend on the initial solution and methodology employed.

Example 2.1. Consider a network with two parallel arcs a_1 and a_2 connecting two nodes A and B with a total demand of 400 from node A to B. Suppose we are interested in minimising either NO_x or CO emissions, employing emissions functions from Song et al. (2013) and using suitable arc parameters. We can compare the objectives of solutions to the SO-E problem for all possible flow combinations using the total-cost function stated in Eq. (7a). Furthermore, we can additionally compare this to the case where emission-minimising speed limits are enforced with Eq. (11). These results are shown in Fig. 1, where we observe a unique minimum for the total cost with speed limits (and non-decreasing arc cost function). In contrast, the total cost without speed limits has multiple local minima due to the non-monotonic arc cost functions.

Although the uniqueness of equilibria for the SO-NM or UE-NM problems cannot be guaranteed, we can show that equilibrium solutions exist. For the UE-NM problem, if arc costs are positive, Theorem 1 guarantees the existence of an equilibrium. For the SO-NM problem, however, the perceived arc costs $\hat{c}_a(f_a)$ may be negative due to the derivative of the non-monotonic cost function (see Eq. (6)), as demonstrated in Example 2.2 below with $c_a(f_a)$ replaced by $s_a e_a(f_a)$. Because of this, we cannot rely on Theorem 1 to guarantee the existence of an equilibrium.

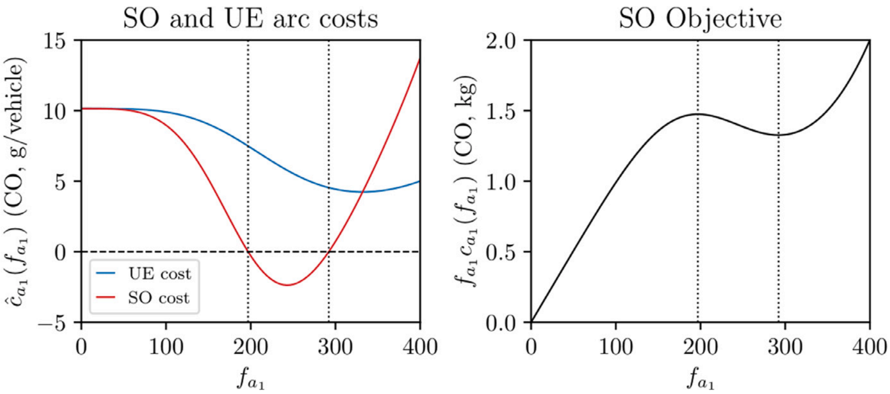

Fig. 2. UE and SO costs for a single arc (left) in Example 2.2, and corresponding total cost for CO emissions (right). Parameters are $k = 175$ in (8), $s = 1$ km in (9), and a free-flow speed of 120 km/h.

Proposition 3. *There exists an equilibrium for the SO-NM problem given that the set $\mathcal{P}$ of feasible paths contains only simple paths.*

Proof. The existence of an equilibrium can be proven based on the optimisation problem (7) by using Weierstrass' theorem, which states that a continuous cost function attains its minimum on a non-empty, compact set (Bazaraa et al., 2013).

To prove the proposition, we apply the same arguments as in the proof of Theorem 2.4 in Patriksson (2015). However, due to the non-monotonic arc costs, we cannot assume that arc costs are positive, which Patriksson (2015) uses to ensure a finite set of feasible paths. By restricting the set $\mathcal{P}$ of feasible paths to simple paths (i.e., paths that visit each node at most once), however, we can ensure that the set of feasible paths is finite. Therefore, Weierstrass' theorem ensures the existence of an equilibrium for the SO-NM problem as there exists a minimum for the equivalent optimisation problem (7). □

Due to the presence of negative arc costs in the SO-NM problem, there may be negative cycles in the graph, meaning that the set of feasible paths may not be finite if non-simple paths are permitted. Proposition 3 assumes that only simple paths form the feasible set of paths for the SO-NM problem. This restriction is reasonable in the context of TA, where a single user should not be able to increase congestion (and thereby decrease the overall emissions) by travelling along the same arc more than once.

For an SO-NM problem with $\lim_{f_a \to +\infty} \hat{c}_a(f_a) > 0$, such as for the SO-E problem, the restriction to simple paths would not be necessary to ensure a finite set of paths since the cost of any cycle eventually becomes positive when a user travels along the cycle sufficiently often. Hence, network users will travel along a combination of path segments and cycles that are traversed a finite number of times. However, since the shortest path (SP) problem with negative cycles must still be solved to identify the corresponding cycles, allowing paths with cycles greatly increases the complexity of the problem.

Example 2.2. Consider a network with two nodes A and B, one arc a_1 from A to B, and another arc a_2 from B to A with identical arc parameters. The UE cost $\hat{c}_{a_1}(f_{a_1}) = c_{a_1}(f_{a_1})$ and the SO cost $\hat{c}_{a_1}(f_{a_1}) = c_{a_1}(f_{a_1}) + f_{a_1}c'_{a_1}(f_{a_1})$ on arc a_1 are shown for increasing flow in Fig. 2. The UE arc cost is positive for all feasible flows, while the SO arc cost is negative if the flow f_{a_1} on arc a_1 is between 197 and 292 units (between the vertical dotted lines in Fig. 2). Hence, if there is a demand between 197 and 292 units from A to B and the same demand from B to A, the SO cost on each arc is negative, producing a negative cycle. Due to the existence of this negative cycle, there are no SPs for the OD pairs. The reason for the negative SO arc cost is that the increased cost of adding a user to the arc is offset by the amount they reduce the cost for all other users.

2.2. Literature overview on emissions in TA

A majority of the literature on modelling emissions within TA employs emission functions that are either effectively increasing with respect to flow (with high emission-minimising speeds), or involve a speed limit that ensures that the resulting function is increasing (Yin and Lawphongpanich, 2006; Yang et al., 2012; Patil, 2016; Raith et al., 2016). Emissions functions are typically employed within the literature either as functions that inherently account for the heterogeneity of the fleet of vehicles or as functions that are calibrated for a single vehicle type (Benedek and Rilett, 1998; Yin and Lawphongpanich, 2006; Yang et al., 2012; Song et al., 2013; Patil, 2016).

The SO-E problem is investigated in Sugawara and Niemeier (2002) in the context of CO emissions. A u-shaped emissions-speed function is derived from lab data with an emission-minimising speed of 73.2 km/h. Simulated annealing is used in the line search step of the Frank–Wolfe algorithm to find a locally optimal solution to the SO-E problem. However, the authors do not address the negative arc costs that can occur when solving the SO-NM problem, as demonstrated in Example 2.2, which would suggest that the network properties in their test network do not produce negative-cost cycles. Rilett and Benedek (1994) and Benedek and Rilett (1998) also study CO emissions for both UE-E and SO-E problems, acknowledging that the solutions they find may not be globally optimal. They employ an emissions function from the TRANSYT 7-F model (Penic and Upchurch, 1992), which has an emission-minimising speed of 75.4 km/h. This produces a strictly increasing arc cost function $c_a(f_a)$ for the majority of feasible speeds, in contrast to the models proposed by Sugawara and Niemeier (2002), Song et al. (2013) and Patil (2016) that have lower emission-minimising speeds. Rilett and Benedek (1994) suggest a so-called system-equitable assignment would be of interest, where the objective captures the cost to residents of a neighbourhood, with the possibility of including constraints on emissions for certain neighbourhoods. Yin and Lawphongpanich (2006) use the same emissions function from the TRANSYT 7-F model, focusing on determining tolling schemes that enforce SO-E solutions assuming the total-cost objective (as in Eq. (7)) is convex, which is the case when the speeds are limited to 75 km/h.

Dynamic TA (DTA) models have been studied for some time now, and have more recently been gaining popularity in the research community (Vickrey, 1969; Arnott et al., 1990; Chiu et al., 2011; Zhang and Qian, 2020). While DTA models are appealing as they are able to better capture, for instance, traffic dynamics, queuing behaviour, and departure time choice, they also come with challenges due to complexity of model calibration, interpretability of results, lack of properties such as solution uniqueness or existence, and computational difficulties that restrict the ability to consider large networks (Chiu et al., 2011). Comparatively little research has considered emissions as part of DTA models, see Wang et al. (2018) and references therein. It is noted that some of the more sophisticated emission models are better suited for inclusion in a DTA model as they require input data that is more fine-grained than that of an average-speed TA model. However, current research in this field mainly relies on simulation-based studies contrasting travel time and emissions for small to medium-sized instances, as well as analytical models of emission-based DTA for a small single-destination network instance where it is a challenge to (approximately) solve the complex formulations (Aziz and Ukkusuri, 2012; Long et al., 2018).

In the following, we focus on static TA models due to the availability of better-understood analytical models and available (exact) solution algorithms, and since these static TA models are known to still be “favoured by policymakers for strategic transportation planning” (Wang et al., 2018, p. 371).

3. NP-hardness

In this section, we show that the SO-NM problem is NP-hard to solve. In order to obtain this result, we show that even (a restricted version of) the SO-E problem, which is a special case of the more general SO-NM problem, is NP-hard.

In the SO-E problem, the cost function is given by $c_a(f_a) = s_a e_a(v_a(f_a))$ and expresses the emissions per vehicle on an arc $a \in \mathcal{A}$ as a function of the arc flow f_a. Therefore, we also refer to c_a as the emissions function of the arc a in the following. Formally, the SO-E problem is defined as follows:

Definition 1 (*Emissions-Optimal Flow Problem (SO-E)*)**.**

INSTANCE:	A directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ with an arc length $s_a \in \mathbb{Q}^+$ and an emissions function c_a given by $c_a(f_a) = s_a e_a(v_a(f_a))$ for each arc $a \in \mathcal{A}$, and K origin–destination pairs $\{(s_i, t_i, d_i)\}_{i=1}^K$.
TASK:	Find a feasible arc flow f minimising the total emissions given as $\sum_{a \in A} f_a c_a(f_a)$.

In the decision version of the problem, a bound $B \in \mathbb{Q}^+$ on the total emissions is given and the question is whether there exists a feasible flow with total emissions at most B.

In the restricted version of the SO-E problem that we use to show NP-hardness, we additionally assume that the emissions functions have the following property:

Property 1. *For each arc $a \in \mathcal{A}$, there exists a value $\hat{f}_a > 0$ such that $c_a(f_a)$ is twice continuously differentiable on $(0, \hat{f}_a]$ and*

(a) $c'_a(f_a) < 0$ *on* $(0, \hat{f}_a]$,
(b) $c''_a(f_a) \leq 0$ *on* $(0, \hat{f}_a]$.

In order to talk about NP-hardness, we have to specify how instances of the SO-E problem are encoded. Our hardness result holds even if the emissions functions are obtained by using the standard BPR travel-time functions (1) with $\alpha = 0.15$ and $\beta = 4$ to determine travel time and, thus, speed as a function of arc flow, and then using an emissions function (as a function of speed) as specified in (8) – both of which are rational functions and can be encoded by storing their rational parameters. Note that Property 1 holds (under certain choices of the coefficients) when the emissions-flow functions are obtained from BPR travel-time functions (1) together with emissions-speed functions (8) as in Song et al. (2013).

We show NP-hardness of SO-E by presenting a reduction from the well-known 3-SAT problem (see, e.g., Karp, 1972). An instance of 3-SAT is given by n variables $x_1, \dots, x_n$ and m clauses $C_1, \dots, C_m$. Each clause C_i is the disjunction of up to three literals, where a literal is either a variable x_i or its negation $\bar{x}_i$. The question is whether there exists an assignment of boolean values to the variables $x_1, \dots, x_n$ that satisfies all clauses $C_1, \dots, C_m$. Here, a clause C_i is satisfied if and only if at least one of the literals in C_i is true, so, e.g., $C_i = (x_1 \vee \bar{x}_2 \vee x_3)$ is satisfied if and only if x_1 is true, $\bar{x}_2$ is true (i.e., x_2 is false), or x_3 is true.

We now provide the reduction from 3-SAT to SO-E. Given an instance of 3-SAT with variables $x_1, \dots, x_n$ and clauses $C_1, \dots, C_m$, we construct an instance of SO-E as follows: All arcs $a \in \mathcal{A}$ have unit length $s_a = 1$ and the same function $c_a(f_a)$ expressing the emissions per vehicle as a function of the flow f_a on the arc. This function is obtained from a BPR travel-time function with $\alpha = 0.15$ and $\beta = 4$ and an emissions function (as a function of speed) as in Song et al. (2013) (Eq. (8)) and satisfies Property 1. Since the

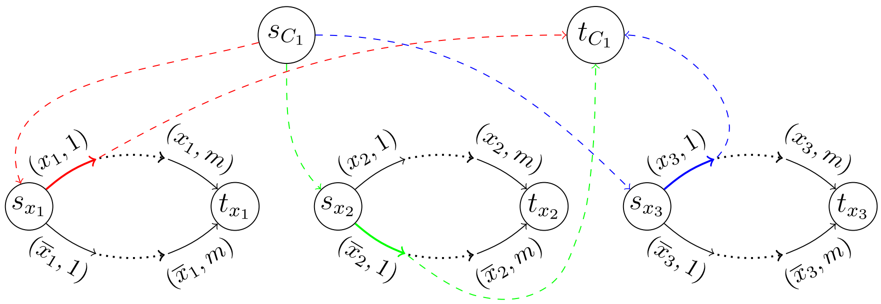

Fig. 3. Construction of the graph $\mathcal{G}$ from the 3-SAT instance in the proof of Theorem 4. For better readability, the clause nodes s_{C_j} and t_{C_j} and their adjacent arcs are only drawn for the first clause C_1. The three coloured paths correspond to the three literals x_1, $\bar{x}_2$, and x_3 contained in clause C_1. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

function is identical for all arcs, we drop the arc index for the rest of the reduction and simply denote this function by c instead of c_a, and, similarly, denote the corresponding value $\hat{f}_a$ as in Property 1 by $\hat{f}$ instead of $\hat{f}_a$.

The directed graph $\mathcal{G}$ is constructed as illustrated in Fig. 3. For every variable x_i, there are two nodes s_{x_i} and t_{x_i} that are connected by two s_{x_i}-t_{x_i}-paths $p_{x_i} = \{(x_i,1),(x_i,2),\ldots,(x_i,m)\}$ and $p_{\bar{x}_i} = \{(\bar{x}_i,1),(\bar{x}_i,2),\ldots,(\bar{x}_i,m)\}$. Here, (x_i,j) and $(\bar{x}_i,j)$ denote the jth arcs in the paths p_{x_i} and $p_{\bar{x}_i}$, respectively (no notation is introduced for the intermediate nodes on these paths). Moreover, there is a *variable commodity* $(s_{x_i},t_{x_i},\hat{f}/2)$ for each variable x_i that travels from s_{x_i} to t_{x_i} and has a demand of $\hat{f}/2$.

For every clause C_j, there are two nodes s_{C_j} and t_{C_j}. If clause C_j contains the three literals l_1,l_2,l_3, then there is an arc from s_{C_j} to the start node of each of the three arcs (l_1,j), (l_2,j), and (l_3,j) as well as an arc from the end node of each of those three arcs to t_{C_j}, shown as dashed coloured arcs in Fig. 3. Moreover, there is a *clause commodity* $(s_{C_j},t_{C_j},\hat{f}/2)$ for each clause C_j that travels from s_{C_j} to t_{C_j} and has a demand of $\hat{f}/2$. This completes the description of the constructed SO-E instance.

We now show that the given instance of 3-SAT has a solution (i.e., a truth assignment ϕ for the variables that satisfies all clauses) if and only if there exists a feasible arc flow f for the SO-E instance with total emissions at most $B := m\cdot\hat{f}\cdot c(\hat{f})+(n+1)\cdot m\cdot\hat{f}/2\cdot c(\hat{f}/2)$.

To this end, first assume that the 3-SAT instance admits a solution ϕ. Then we define a feasible flow f for the SO-E instance as follows: For each variable x_i, we send all flow of the corresponding variable commodity $(s_{x_i},t_{x_i},\hat{f}/2)$ along the path p_{x_i} if x_i = true under ϕ, and send all flow of this commodity along the path $p_{\bar{x}_i}$ if x_i = false under ϕ. Moreover, since ϕ is a solution of the 3-SAT instance, every clause C_j contains at least one literal l that has value true, and we send all flow of the corresponding clause commodity $(s_{C_j},t_{C_j},\hat{f}/2)$ along the path that uses the arc (l,j) (if clause C_j contains several literals that have value true, we choose one of them arbitrarily). The resulting flow f is clearly feasible. In this flow, every clause commodity uses two arcs alone (the first and the last arc of its path), which, in total, yields $2m$ arcs with flow $\hat{f}/2$ on which overall emissions of $2m\cdot\hat{f}/2\cdot c(\hat{f}/2)$ are generated. Furthermore, every clause commodity uses one arc (the middle arc in its path) jointly with a variable commodity, which yields m arcs with flow $\hat{f}$ on which overall emissions of $m\cdot\hat{f}\cdot c(\hat{f})$ are generated. Moreover, since the path of each of the n variable commodities contains exactly m arcs, there are $n\cdot m-m=(n-1)\cdot m$ arcs with positive flow remaining that are only used by one variable commodity (and no clause commodity) each. Thus, overall emissions of $(n-1)\cdot m\cdot\hat{f}/2\cdot c(\hat{f}/2)$ are generated on these remaining arcs. In total, this means that the total emissions of the flow f equal

$$\begin{aligned}&2m\cdot\hat{f}/2\cdot c(\hat{f}/2)+m\cdot\hat{f}\cdot c(\hat{f})+(n-1)\cdot m\cdot\hat{f}/2\cdot c(\hat{f}/2)\\&=m\cdot\hat{f}\cdot c(\hat{f})+(n+1)\cdot m\cdot\hat{f}/2\cdot c(\hat{f}/2)=B.\end{aligned}$$

Now assume that the 3-SAT instance does not have a solution. We show that, in this case, every feasible flow for the SO-E instance has total emissions strictly larger than B. To this end, we first show two lemmas concerning the structure of optimal flows for the SO-E instance.

Lemma 1. *There exists an optimal flow for the SO-E instance in which the flow of each variable commodity is sent along a single path.*

Proof. We show that, for any variable commodity $(s_{x_i},t_{x_i},\hat{f}/2)$ and any fixed choice of the flow of the other commodities, the total emissions are minimised either by sending all the $\hat{f}/2$ flow units of this variable commodity along the path p_{x_i} or by sending all the $\hat{f}/2$ flow units along the path $p_{\bar{x}_i}$.

Since there are only the two paths p_{x_i} and $p_{\bar{x}_i}$ available for commodity $(s_{x_i},t_{x_i},\hat{f}/2)$, we can represent any feasible flow of this commodity by the amount $0\le\theta\le\hat{f}/2$ of flow units of the commodity that are sent along the path p_{x_i} – the flow on the path $p_{\bar{x}_i}$ is then given as $\hat{f}/2-\theta$. If we denote the total flow of all other commodities on each arc a that is contained in one of these two paths by f_a^u, we have $0\le f_a^u\le\hat{f}/2$ since each arc of p_{x_i} and $p_{\bar{x}_i}$ can be used by at most one of the clause commodities (and none of the other variable commodities). The total emissions generated on the arcs of the paths p_{x_i} and $p_{\bar{x}_i}$ can then be written as

$$\sum_{a\in p_{x_i}:f_a^u+\theta>0}(f_a^u+\theta)\cdot c(f_a^u+\theta)+\sum_{a\in p_{\bar{x}_i}:f_a^u+\hat{f}/2-\theta>0}(f_a^u+\hat{f}/2-\theta)\cdot c(f_a^u+\hat{f}/2-\theta).$$

The first derivative of this expression with respect to θ is

$$\sum_{a\in p_{x_i}: f_a^u+\theta>0} \left[c(f_a^u+\theta)+(f_a^u+\theta)\cdot c'(f_a^u+\theta)\right]$$
$$-\sum_{a\in p_{\bar{x}_i}: f_a^u+\hat{f}/2-\theta>0} \left[c(f_a^u+\hat{f}/2-\theta)+(f_a^u+\hat{f}/2-\theta)\cdot c'(f_a^u+\hat{f}/2-\theta)\right],$$

and the second derivative is

$$\sum_{a\in p_{x_i}: f_a^u+\theta>0} \Big[2\,\underbrace{c'(f_a^u+\theta)}_{\le 0}+\underbrace{(f_a^u+\theta)}_{\ge 0}\cdot\underbrace{c''(f_a^u+\theta)}_{\le 0}\Big]$$
$$+\sum_{a\in p_{\bar{x}_i}: f_a^u+\hat{f}/2-\theta>0} \Big[2\,\underbrace{c'(f_a^u+\hat{f}/2-\theta)}_{\le 0}+\underbrace{(f_a^u+\hat{f}/2-\theta)}_{\ge 0}\cdot\underbrace{c''(f_a^u+\hat{f}/2-\theta)}_{\le 0}\Big],$$

which is non-positive. Thus, the total emissions generated on the arcs of the paths p_{x_i} and $p_{\bar{x}_i}$ is a concave function of θ on $[0,\hat{f}/2]$, which is, consequently, minimised for either $\theta=\hat{f}/2$ or $\theta=0$, i.e., either if all flow of the variable commodity is sent along p_{x_i}, or if all of this flow is sent along $p_{\bar{x}_i}$. □

Lemma 2. *There exists an optimal flow for the SO-E instance in which the flow of each clause commodity is sent along a single path.*

Proof. We show that, given any clause commodity $(s_{C_j}, t_{C_j}, \hat{f}/2)$ and any fixed choice of the flow of the other commodities such that the flow of commodity $(s_{C_j}, t_{C_j}, \hat{f}/2)$ is split on $r\ge 2$ paths, we can reroute this flow onto $r-1$ paths without increasing the total emissions.

Let $0<\theta_1,\theta_2<\hat{f}/2$ denote the amounts of flow of commodity $(s_{C_j}, t_{C_j}, \hat{f}/2)$ sent along two different paths $p_1=\{a_{11},a_{12},a_{13}\}$ and $p_2=\{a_{21},a_{22},a_{23}\}$, respectively. Then, by construction of the graph $\mathcal{G}$, only the middle arcs a_{12}, a_{22} might be used also by other commodities, namely by (at most) one variable commodity each. Denoting the flows of these variable commodities on a_{12}, a_{22} by $0\le f_{a_{12}}^u, f_{a_{22}}^u\le \hat{f}/2$, respectively, the total emissions generated on the arcs of the paths p_1 and p_2 can then be written as

$$2\theta_1\cdot c(\theta_1)+(f_{a_{12}}^u+\theta_1)\cdot c(f_{a_{12}}^u+\theta_1)+2\theta_2\cdot c(\theta_2)+(f_{a_{22}}^u+\theta_2)\cdot c(f_{a_{22}}^u+\theta_2).$$

Keeping the sum $\eta:=\theta_1+\theta_2$ of the flows on p_1 and p_2 fixed, we can write this as a function of only θ_1 by substituting $\theta_2=\eta-\theta_1$, which yields

$$2\theta_1\cdot c(\theta_1)+(f_{a_{12}}^u+\theta_1)\cdot c(f_{a_{12}}^u+\theta_1)$$
$$+2(\eta-\theta_1)\cdot c(\eta-\theta_1)+(f_{a_{22}}^u+\eta-\theta_1)\cdot c(f_{a_{22}}^u+\eta-\theta_1).$$

The first derivative of this expression with respect to θ_1 is

$$2\left[c(\theta_1)+\theta_1\cdot c'(\theta_1)\right]+c(f_{a_{12}}^u+\theta_1)+(f_{a_{12}}^u+\theta_1)\cdot c'(f_{a_{12}}^u+\theta_1)$$
$$-2\left[c(\eta-\theta_1)+(\eta-\theta_1)\cdot c'(\eta-\theta_1)\right]-c(f_{a_{22}}^u+\eta-\theta_1)$$
$$-(f_{a_{22}}^u+\eta-\theta_1)\cdot c'(f_{a_{22}}^u+\eta-\theta_1),$$

and the second derivative is

$$2\Big[2\,\underbrace{c'(\theta_1)}_{\le 0}+\underbrace{\theta_1}_{\ge 0}\cdot\underbrace{c''(\theta_1)}_{\le 0}\Big]+2\,\underbrace{c'(f_{a_{12}}^u+\theta_1)}_{\le 0}+\underbrace{(f_{a_{12}}^u+\theta_1)}_{\ge 0}\cdot\underbrace{c''(f_{a_{12}}^u+\theta_1)}_{\le 0}$$
$$+2\Big[2\,\underbrace{c'(\eta-\theta_1)}_{\le 0}+\underbrace{(\eta-\theta_1)}_{\ge 0}\cdot\underbrace{c''(\eta-\theta_1)}_{\le 0}\Big]$$
$$+2\,\underbrace{c'(f_{a_{22}}^u+\eta-\theta_1)}_{\le 0}+\underbrace{(f_{a_{22}}^u+\eta-\theta_1)}_{\ge 0}\cdot\underbrace{c''(f_{a_{22}}^u+\eta-\theta_1)}_{\le 0},$$

which is non-positive. Thus, the total emissions generated on the arcs of the paths p_1 and p_2 is a concave function of θ_1 on $[0,\hat{f}/2]$, which is, consequently, minimised for either $\theta_1=0$ or $\theta_1=\hat{f}/2$. This shows that the total emissions do not increase when re-routing all flow from path p_1 to path p_2 (if the minimum is obtained for $\theta_1=0$) or vice versa (if the minimum is obtained for $\theta_1=\hat{f}/2$), which shows the lemma. □

Theorem 4. *The SO-E problem is strongly NP-hard, even if the graph is a DAG (directed acyclic graph), the demands of all commodities are identical, and all arcs have the same length and the same emissions function obtained from a BPR travel-time function (1) with $\alpha=0.15$ and $\beta=4$ and an emissions function (8) (as a function of speed) as in Song et al. (2013).*

Proof. By Lemma 1, in order to show that any feasible flow has total emissions strictly larger than B, it suffices consider only those feasible flows in which the flow of each variable commodity is sent along a single path. Each such flow of the variable commodities corresponds to a truth assignment of the variables in the 3-SAT instance by setting variable x_i to true if and only if the corresponding variable commodity sends its flow along p_{x_i}. Moreover, by Lemma 2, we can further restrict attention to those flows in which also each clause commodity sends its flow along a single path. Given a flow of the variable commodities corresponding to a truth assignment of the variables, a clause is then satisfied by this truth assignment if and only if there exists a path for the corresponding clause commodity whose middle arc is used by one of the variable commodities.

Thus, the assumption that the 3-SAT instance does not have a solution means that, in every optimal single-path-per-commodity flow as in Lemmas 1 and 2, there is at least one clause commodity that uses the middle arc of its path alone. If $k \geq 1$ denotes the number of such commodities in a given optimal flow, the total emissions of this optimal flow can then be lower-bounded by

$$\begin{aligned}
&2m \cdot \hat{f}/2 \cdot c(\hat{f}/2) + (m-k) \cdot \hat{f} \cdot c(\hat{f}) + \left[(n-1) \cdot m + 2k\right] \cdot \hat{f}/2 \cdot c(\hat{f}/2) \\
&\quad = B - k \cdot \hat{f} \cdot c(\hat{f}) + 2k \cdot \hat{f}/2 \cdot c(\hat{f}/2) \\
&\quad = B + \underbrace{k \cdot \hat{f}}_{>0} \cdot \underbrace{\left[c(\hat{f}/2) - c(\hat{f})\right]}_{>0} > B,
\end{aligned}$$

which shows that the total emissions of every feasible flow must be strictly larger than B as claimed. □

In summary, we have shown that the SO-E problem – and, more generally, the SO-NM problem – is strongly NP-hard. To the best of our knowledge, the computational complexity of the UE-NM problem has not been studied formally and it remains an open question whether UE-NM is also NP-hard. In the next section, we show how to adapt an iterative algorithm for TA to heuristically obtain good solutions of the SO-E problem and the SO-NM problem.

4. Heuristics for TA with non-monotonic arc cost functions

There are various challenges with identifying solutions to the SO-NM problem. Most TA algorithms move from an initial feasible solution towards an equilibrium by identifying SPs for the OD pairs at the current flow and iteratively moving closer to an equilibrium. Non-monotonic arc cost functions within TA pose difficulties for path finding, for example, due to the presence of negative cycles as in Example 2.2. In this section, we outline the path equilibration (PE) algorithm for solving the TA problem (Dafermos and Sparrow, 1969; Florian and Hearn, 1995). Examples of other TA algorithms include the Frank–Wolfe algorithm and the so-called Algorithm B, both of which also rely on SP calculation and iterative shifting of flow (Frank and Wolfe, 1956; Dial, 2006). We then propose heuristic adaptations to the PE algorithm to solve the SO-NM problem. The adaptations made to the PE algorithm are significant, with the resulting heuristic algorithm denoted as the PE-NM algorithm. With the PE-NM algorithm, we can compute equilibria for the SO-NM problem. However, there may be many such equilibria, as shown in Example 2.1, each of which may have a different total cost. The presented PE-NM approach can also be used to solve UE-NM. Section 4.4 will suggest methods that aim to compute equilibria with low total costs for the SO-E problem.

4.1. Finding an initial solution

An initial solution can be constructed with the all-or-nothing (AON) assignment, outlined in Algorithm 1. For the AON assignment, the network $\mathcal{G}$ begins with zero flow on all arcs, such that all arcs have a cost of $\hat{c}_a(0)$. We define the set of active paths $\mathcal{P}^+$, which is initialised as $\mathcal{P}^+ = \emptyset$ and will contain paths with positive flow or zero-flow paths to be equilibrated. Furthermore, we define the subset $\mathcal{P}_i^+ \subseteq \mathcal{P}^+$ to be the set of active paths for the OD pair (s_i, t_i, d_i). For each OD pair, a SP is found using an SP algorithm. This path is then assigned flow equal to d_i, and it is added to $\mathcal{P}^+$. For each path for an OD pair (s_i, t_i, d_i) added to $\mathcal{P}^+$, all arcs that make up the path have their flows updated by adding the path's flow d_i. After all demand is assigned for all OD pairs, the arc costs are updated to $\hat{c}_a(f_a)$.

Algorithm 1: The all-or-nothing (AON) assignment algorithm

Input : Network $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ and cost function $\hat{c}_a(f_a)$.
Output: Initial feasible path flow h with arc flow f for the TA problem with network $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ and active path set $\mathcal{P}^+$.

Initialise arc costs as $\hat{c}_a(0)$ for all $a \in \mathcal{A}$ and set $\mathcal{P}^+ = \emptyset$;
for $i = 1, \ldots, K$ **do**
  Find a shortest path p_i^* from s_i to t_i;
  Add path p_i^* to path set $\mathcal{P}_i^+$;
  Assign $h_{p_i^*} = d_i$;
  Update arc flows $f_a = f_a + d_i$ for all $a \in p_i^*$;
Update arc costs $\hat{c}_a(f_a)$ for all $a \in \mathcal{A}$;

4.2. The path equilibration algorithm

Once an initial solution has been found, the PE algorithm is used to find an equilibrium. The PE algorithm takes advantage of the equilibrium conditions (2) in order to solve the TA problem, with each OD pair being treated as a separate equilibrium sub-problem.

We outline a variation of the PE algorithm in Algorithm 2. The PE algorithm begins with an initial solution, such as the one produced by the AON assignment, and iteratively improves this solution. An equilibrium is found for a fixed set of paths, *equilibrating* the active paths $p \in \mathcal{P}_i^+$ for a given OD pair (s_i, t_i, d_i). The equilibration of paths (line 5 in Algorithm 2) involves the shifting of path flow from a maximum-cost path (with positive flow) to a minimum-cost path. This continues until the two paths have the same cost or all flow is shifted, and is repeated until all paths $p \in \mathcal{P}_i^+$ for OD pair (s_i, t_i, d_i) with positive flow have the same path cost. For arc costs $\hat{c}_a(f_a)$ that are continuous, positive, and non-decreasing, this flow shift must either decrease the total cost or shift all flow. Paths with zero flow are then removed from the active path set. We employ a bisection line search algorithm to identify the amount of flow to shift between two paths. This equilibration is iteratively repeated for all OD pairs until there is no change in path flow, meaning the algorithm has identified an equilibrium for the given path set. Following this, the active path set is improved by adding SPs for each OD pair if they are less costly than all respective active paths. The improvement of the active path set uses arc costs evaluated at the current arc flows. If no SPs are added to the active path set, the active path set can no longer be improved, and we have converged to a final equilibrium.

A variation of the PE algorithm finds an equilibrium for a predefined path set without using SPs to improve it. The variation is often used in the context of non-additive path costs, where SP calculations are difficult (Cascetta et al., 1997; Lo and Chen, 2000; Chen et al., 2012). This allows the computation of equilibria when identifying paths may be difficult, but limits the solution space.

Algorithm 2: The path equilibration algorithm

Input : Initial feasible path flow h with arc flow f for the TA problem with network $\mathcal{G} = (\mathcal{V}, \mathcal{A})$, active path set $\mathcal{P}^+$, and arc cost function $\hat{c}(f)$.
Output: Equilibrium path flow h and arc flow f satisfying conditions (2).

1 **while** *not converged* **do**
2   **while** *not equilibrated* **do**
3   **for** $i = 1, \dots, K$ **do**
4   Update path costs $\hat{c}_p(f)$ for all $p \in \mathcal{P}_i^+$; // for current f
5   Equilibrate path set $\mathcal{P}_i^+$; // shifting flows for $p \in \mathcal{P}_i^+$
6   Update arc costs $\hat{c}_a(f_a)$ for all $a \in p : p \in \mathcal{P}_i^+$;
7   Remove paths with zero flow from $\mathcal{P}_i^+$;
8   **for** $i = 1, \dots, K$ **do**
9   Find the shortest path p_i^* from s_i to t_i;
10   **if** $\hat{c}_{p_i^*}(f) < \min_{p \in \mathcal{P}_i^+} \hat{c}_p(f)$ **then**
11   Add p_i^* to $\mathcal{P}_i^+$;
12   **if** $\mathcal{P}^+$ *is unchanged* **then**
13   converged = True;

4.3. The PE algorithm with non-monotonic arc costs

When applying the PE algorithm to TA problems with non-monotonic arc costs, we cannot guarantee that the path flows will converge to an equilibrium because the algorithm assumes positive, non-decreasing arc cost functions. In Sections 4.3.1–4.3.3, we identify challenges that need to be addressed to be able to apply the PE algorithm to the SO-NM or UE-NM problems and suggest various heuristic adaptations in order to allow the identification of equilibria. The adaptations can also be incorporated into other TA algorithms that work with path sets. Algorithm 3 summarises the resulting PE-NM algorithm and Section 4.3.4 shows that obtained solutions are indeed equilibria.

4.3.1. Negative arc costs

As shown in Example 2.2, non-monotonic arc cost functions can produce negative arc costs when solving the SO-NM problem. In the case of UE-NM, negative costs only occur if the non-monotonic arc cost function is negative for some level of flow. Negative arc costs cause increased complexity in the SP problem that arises in the initialisation with the AON assignment (Algorithm 1) and the iterations of the PE algorithm, due to the possibility of negative cycles in the network.

The PE algorithm iteratively improves an initial feasible solution. Initialisation with the AON assignment relies on costs $\hat{c}_a(0)$, ensuring that, for the SO-E problem, negative cycles will not be present. If $c_a(0) < 0$ for the general SO-NM or UE-NM problems, arcs can instead be assigned a cost $c_a(0) = 1$ to allow the AON assignment to obtain an initial feasible solution.

In the iterations of PE, the existence of negatives cycles can be checked for a given arc flow f for a network $\mathcal{G}$. If no negative cycle is present, the SP calculation can be accomplished with SP algorithms that allow negative arc costs. When a negative cycle is present, the simple (or elementary) shortest path (SSP) problem needs to be solved, where we assume the set of feasible paths $\mathcal{P}$ is restricted to simple paths, as stated in Section 2, to ensure feasibility.

The SSP problem in a network with negative cycles is NP-hard, as it is akin to the longest (simple) path problem, which is known to be NP-hard (Karger et al., 1997). The SSP problem has been considered previously in the context of the vehicle routing problem with resource constraints (Feillet et al., 2004; Righini and Salani, 2008). Solution methods for the SSP problem generally involve improving label dominance rules. Furthermore, the search space for the SSP can be reduced by providing an upper bound on the path cost, from either a path found via a heuristic or using the cost of a path in the active path set. However, the resulting problem remains computationally difficult, especially when resource constraints cannot be used to limit the solution space (Chabrier, 2006; Drexl and Irnich, 2014).

As SSP is computationally challenging, we avoid SSP calculations by using a heuristic to find a 'good' path rather than the SSP whenever a negative cycle is present. One approach to solving the SSP problem heuristically to obtain a 'good' path is to alter the network costs to remove negative cycles temporarily. This can be done by iteratively using the Bellman–Ford algorithm (Bellman, 1958) to identify negative cycles and changing negative arc costs within the cycle to zero. The negative cycles can also be identified by iteratively solving the minimum-cost flow problem with demands of zero and capacities of 1, such that the solution will have positive flow on arcs that make up negative cycles — where each iteration will find a set of disconnected negative cycles. Once no more negative cycles are found, we can use the Bellman–Ford algorithm to identify shortest paths using these temporary costs, which will be 'good' paths for the true network costs.

To adapt the PE algorithm for the SO-NM problem, whenever we require a SP calculation, we use the Bellman–Ford algorithm. In the absence of a negative cycle, a SP is obtained; otherwise, we use the minimum-cost flow heuristic described above to return a 'good' path. If the good path has a cost less than the minimum cost of an active path for the given OD pair, it is added to $\mathcal{P}_i^+$; otherwise we move on to the next OD pair despite the possible existence of a shorter unused path (see lines 9–15 of Algorithm 3). When the convergence criteria have been satisfied, the algorithm checks for a negative cycle. If there are no negative cycles, we are at an equilibrium since the lack of negative cycles means that there does not exist a shorter unused path for any OD pair. Otherwise, the current traffic flows may not be at an equilibrium due to the use of 'good' paths rather than SPs, and we must solve the SSP problem. If new SPs are found from the SSP problem, they are added to the set of active paths and the PE algorithm is restarted. If none are found, we are at an equilibrium (see lines 16–24 of Algorithm 3).

Algorithm 3: PE-NM: PE algorithm with non-monotonic arc costs

Input : Initial feasible path flow h with arc flow f for the TA problem with network $\mathcal{G} = (\mathcal{V}, \mathcal{A})$, active path set $\mathcal{P}^+$, and arc cost function $\hat{c}(f)$.
Output: Equilibrium path flow h and arc flow f satisfying conditions (2).

1 **while** *not converged* **do**
2   **while** *not equilibrated* **do**
3   **for** $i = 1, \dots, K$ **do**
4   Update path costs $\hat{c}_p(f)$ for all $p \in \mathcal{P}_i^+$; // for current f
5   Equilibrate path set $\mathcal{P}_i^+$ employing uniform line search to identify best stationary point; // shifting flows for $p \in \mathcal{P}_i^+$
6   Update arc costs $\hat{c}_a(f_a)$) for all $a \in p : p \in \mathcal{P}_i^+$;
7   Remove paths with zero flow from $\mathcal{P}_i^+$;
8   **for** $i = 1, \dots, K$ **do**
9   Run Bellman–Ford SP algorithm for OD pair s_i to t_i;
10   **if** *There exists a negative cycle* **then**
11   Apply minimum-cost flow heuristic to find a 'good' path p_i^*;
12   **else**
13   Let p_i^* be the shortest path from the Bellman–Ford algorithm;
14   **if** $\hat{c}_{p_i^*}(f) < \min_{p \in \mathcal{P}_i^+} \hat{c}_p(f)$ **then**
15   Add p_i^* to $\mathcal{P}_i^+$;
16   **if** $\mathcal{P}^+$ *is unchanged* **then**
17   converged = True;
18   **for** $i = 1, \dots, K$ **do**
19   Run Bellman–Ford SP algorithm for OD pair s_i to t_i;
20   **if** *there exists a negative cycle* **then**
21   Solve SSP and obtain p_i^*;
22   **if** $\hat{c}_{p_i^*}(f) < \min_{p \in \mathcal{P}_i^+} \hat{c}_p(f)$ **then**
23   Add p_i^* to $\mathcal{P}_i^+$;
24   converged = False;

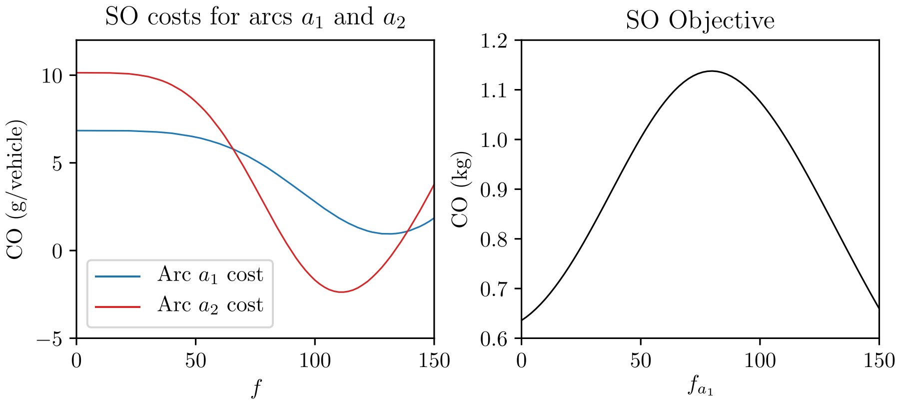

Fig. 4. The SO costs for CO emissions of two parallel arcs (left) in Example 4.1, where allocating all flow to either arc results in a local minimum as seen in the SO objective function (right).

4.3.2. Ineffective path choice

The PE algorithm relies on adding SPs to the active path set $\mathcal{P}^+$. Shifting path flow from a non-shortest path to a shortest path during PE will move the solution closer to equilibrium when arc cost functions are non-decreasing. If no new shortest paths can be added to $\mathcal{P}^+$ after the paths within $\mathcal{P}^+$ have been equilibrated, we are at an equilibrium. However, for decreasing or non-monotonic arc cost functions, adding only SPs can prevent the PE-NM algorithm from identifying certain equilibria, as shown in Example 4.1, and these equilibria may have lower total costs.

Example 4.1. Consider two parallel arcs a_1 and a_2 connecting a single OD pair with demand 150 and with parameters such that we obtain the arc cost functions in Fig. 4. Suppose there is a flow of 150 units on arc a_1 and arc a_2 represents an unused path. This may occur either through AON initialisation, as $\hat{c}_{a_1}(0) \leq \hat{c}_{a_2}(0)$, or partway through the PE algorithm. The minimal total cost could be obtained by shifting all of the flow from arc a_1 to arc a_2. However, since arc a_2 currently has no flow on it, its current cost is $\hat{c}_{a_2}(0)$, which is higher than $\hat{c}_{a_1}(150)$ – implying that arc a_2 cannot be a SP. Using the PE methodology, we would not consider arc a_2 as a promising choice, and would not find the global minimum $f_{a_1} = 0$, $f_{a_2} = 150$.

To address this issue heuristically, one could consider adding promising paths to the active path set in addition to SPs in the PE framework. Here, promising paths are generally low-cost paths, however, depending on the shape of the non-monotonic function, shifting flow to higher cost paths at the current flow level may reduce the total cost by a significant amount. Promising paths could be obtained from a k-shortest-paths algorithm (Yen, 1971; Eppstein, 1998; Hershberger et al., 2007) that identifies the k shortest simple paths in a network, but requires that there are no negative arc weights to work efficiently. During equilibration, all paths $p \in \mathcal{P}_i^+$ with positive flow can be compared with the k newly added paths – rather than only comparing maximum-cost and minimum-cost paths – to identify path flow shifts that decrease the total cost. Adding k paths expands the local search area (the extent of which depends on the value of k), prioritising arcs that have a low cost.

Another method to identify promising paths for a given network flow f and maximum-cost path $p \in \mathcal{P}^+$ with flow h_p is to update the cost of each arc a in the network to $c_a(f_a + \delta)$ except for those arcs on the given path p, which would remain at $c_a(f_a)$, and then compute a SP. The value of δ would correspond to the amount of flow we wish to shift from the maximum-cost path (for example the amount δ that gives the lowest cost on the maximum-cost path), such that the resulting SP would be optimal for shifting δ units of flow. This would ideally be repeated many times for values $\delta \in [0, h_p]$ for the maximum-cost path to identify a near-optimal amount δ of flow to shift, which is not practical.

Our implementation of the PE-NM algorithm (Algorithm 3), does not use any of the discussed methods to address potentially bad path choice as they greatly increase computation time. We instead assume that SPs are sufficient for producing low-cost equilibria.

4.3.3. Stationary points in equilibration stage

The equilibration of two paths p_1 and p_2 in the PE algorithm (line 5 in Algorithms 2 and 3) involves solving the sub-problem of shifting flow between the two paths to achieve equal path cost (or such that the maximum-cost path has zero flow). The sub-problem for minimising cost over two paths p_1 and p_2, with flow h_1 and h_2, respectively, for fixed total flow $h_1 + h_2$ can be stated as

$$\min \quad \sum_{a \in p_1 \setminus p_2} (f_a + \delta) c_a(f_a + \delta) + \sum_{a \in p_2 \setminus p_1} (f_a - \delta) c_a(f_a - \delta) \tag{12a}$$

$$\text{s.t.} \quad -h_1 \leq \delta \leq h_2, \tag{12b}$$

where δ represents the amount of flow shifted from path p_2 to path p_1.

Paths with equal path costs for an OD pair correspond to stationary points for the problem (12) (Sheffi, 1984, equation 3.26). For a strictly increasing arc cost function, there is one stationary point that corresponds to a global minimum. For a non-monotonic arc

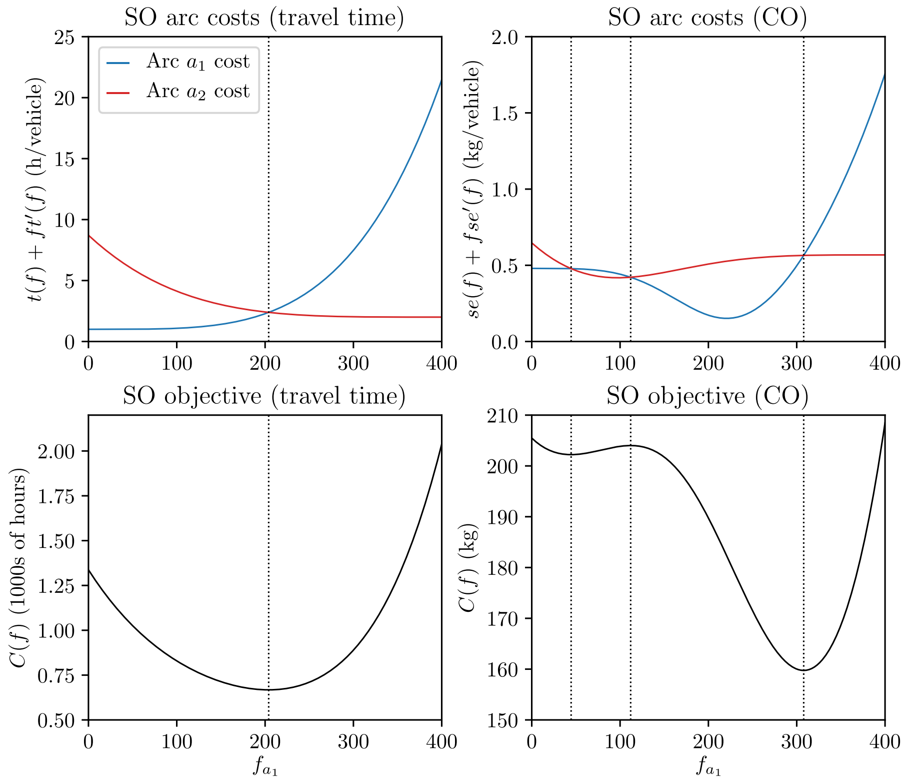

Fig. 5. Plots of SO costs and total cost for travel time and CO emissions for two parallel arcs. All functions are shown as a functions of f_{a_1}, where $f_{a_2} = 400 - f_{a_1}$.

cost function, it is possible that there are multiple such stationary points corresponding to local minima or maxima, as demonstrated in Example 4.2. Choosing such a stationary point arbitrarily may result in a poor-quality local minimum or even a local maximum.

Example 4.2. Consider two parallel arcs a_1 and a_2 connecting an OD pair with a demand of $d = 400$. We use the notation from the optimisation formulation (7), which is equivalent to (12) when the network consists of two arcs. Fig. 5 indicates the total cost for the users to obtain an SO solution for travel time on the left and for CO emissions on the right. The arc costs required to obtain the SO equilibrium (as in Eq. (6)) are at the top of the figure, while the total cost is below.

For strictly increasing arc cost functions, such as travel time, there is one unique path flow pattern that results in equal path costs for a given pair of paths, corresponding to the stationary point that is the minimum of (12). For a non-monotonic arc cost function such as emissions, there can be several (in this case three) path flow patterns that give equal path costs, corresponding to stationary points in the total CO cost in the bottom right of Fig. 5.

To adapt the path cost equilibration step for an OD pair in the PE algorithm, one must identify all values of δ corresponding to stationary points and extreme cases (shifting all or zero flow), evaluate (12) at each such value of δ, and shift the amount δ of flow that produces the lowest cost. This increases the computational effort needed to equilibrate each path combination, but will avoid locally maximum solutions for the OD pair sub-problem. Employing a uniform line search over the domain of feasible path flows for the two paths can be used to find the stationary points, but, depending on the interval size, not all stationary points may be found.

Even if we can identify all stationary points, by choosing the minimal total-cost path flow pattern for a given pair of paths we cannot guarantee that the resulting equilibrium will have the lowest total cost.

Another potential problem is that the equilibration of two paths is only employed when they do not have equal cost. However, the two paths may have the same cost, but a path flow pattern that corresponds to a local maximum for (12). Given the path flow pattern is at a local maximum, we would want to redistribute the flow such that the paths have the same cost as each other, but are instead at a local minimum (or one path has zero flow and a higher cost). Adding SPs during the algorithm may be sufficient to avoid this issue. For certain networks, however, an undetected local maximum may be unavoidable, but could be addressed by manually comparing paths, or adjusting the arc flows slightly and resolving.

In our implementation of the PE-NM algorithm (Algorithm 3), we make use of a uniform line search to identify path flow patterns in the equilibration step of the algorithm (see line 5 of Algorithm 3).

4.3.4. Convergence of the PE-NM algorithm

We know that there exist equilibrium solutions for the UE-NM and SO-NM problems as stated in Section 2, Theorem 1 (if the non-monotonic cost function of UE-NM is positive and continuous) and Section 2.1, Proposition 3 (SO-NM). However, we still need to confirm that our PE-NM algorithm (Algorithm 3) is guaranteed to converge to an equilibrium.

There exists a known lower bound of the total cost — for emissions we can use the total cost for the network with emission-minimising speed limits, which can be calculated with the original PE algorithm defined in Section 4.2. Moreover, we begin with a feasible starting solution such as the solution produced by the AON algorithm outlined in Algorithms 1, and 3 iterates to generate a sequence of flow solutions with strictly decreasing total costs after each equilibration step. Therefore, convergence is guaranteed, but is not necessarily obtained within finitely many steps.

Iterations of PE-NM involve adding unused SPs or 'good' paths, as well as equilibrating path costs for the active path set. For the equilibration of a given OD pair i, we iteratively shift flow between a maximum-cost path and a minimum-cost path in the active path set $\mathcal{P}_i^+$ until all paths for the OD pair with positive flow have the same cost. Specifically, we solve problem (12) for the minimum- and maximum-cost paths p_1 and p_2 in the active path set $\mathcal{P}_i^+$. The maximum-cost path must have positive flow, and both paths have different costs (otherwise, equilibration would not be required). Solving (12) for p_1, p_2 results in changes to only the arcs that make up p_1 and p_2, while all other arcs in the network remain unchanged. As the arc cost functions are non-monotonic, there may be multiple stationary points of (12) that produce an equal cost for p_1, p_2 as shown in Example 4.2.

The arc cost functions are assumed to be continuously differentiable and, for paths p_1 and p_2, the total cost under arc flow f is

$$C_{p_1,p_2}(f) = \sum_{a \in p_1 \cup p_2} f_a c_a(f_a). \tag{13}$$

We denote by (h^j, f^j) the path flow h^j at the current iteration of Algorithm 3 and the arc flow f^j induced by h^j. We consider a particular OD pair i and paths $p_1, p_2 \in \mathcal{P}_i$ that are minimum- and maximum-cost paths, respectively, in the active path set $\mathcal{P}_i^+$ of the current iteration.

If we are at a local minimum for (12), then the paths either have equal cost or the maximum-cost path has zero flow and can be removed from the active path set. In Algorithm 3, this means that OD pair i is equilibrated and the algorithm would either consider the next OD pair, or terminate if all OD pairs are equilibrated. If we are not already at a local minimum for (12), we must be able to find an amount δ of flow to shift between the two paths with resulting path and arc flow (h^{j+1}, f^{j+1}) (as demonstrated in (12) in Section 4.3.3) such that $C_{p_1,p_2}(f^{j+1}) < C_{p_1,p_2}(f^j)$, as $C_{p_1,p_2}(f)$ is continuous. Therefore, there exists (h^{j+1}, f^{j+1}) with $C_{p_1,p_2}(f^{j+1}) < C_{p_1,p_2}(f^j)$ with the corresponding total cost for the network $C(f^{j+1}) < C(f^j)$. We note that (h^{j+1}, f^{j+1}) does not have to be a minimum, only such that $C_{p_1,p_2}(f^{j+1}) < C_{p_1,p_2}(f^j)$. While iteratively shifting flow for an OD pair i in Algorithm 3 until all active paths in $\mathcal{P}_i^+$ with positive flow have the same cost, and repeating this process for each OD pair, we strictly decrease the total cost for the network with every flow shift. Once all OD pairs have been equilibrated, we have converged to an equilibrium for the current active path set, which is a local stationary point (equilibrium) for the set of active paths considered.

Iteratively updating the active path set with 'good' paths or SPs allows the convergence to an equilibrium of the updated path set with lower total cost. If only SPs are used (i.e., no other 'good' paths), then the algorithm is guaranteed to converge to a local stationary point (equilibrium) for the full path set of the entire network.

Within the algorithm, the OD pairs are processed sequentially. In general, the order in which OD pairs are equilibrated can affect the convergence rate of a TA algorithm. Additionally, for a SO-NM problem with multiple solutions, the order in which OD pairs are equilibrated may affect the resulting equilibrium that is found.

4.4. Initialisation methods and modifications of the PE-NM algorithm

The SO-NM problem can now be solved with the PE-NM algorithm (Algorithm 3). However, the algorithm cannot be guaranteed to find an equilibrium that has a globally optimal system cost, and may instead find locally optimal solutions in the corresponding optimisation problem (7) due to the non-convexity of the problem.

The initial solution has a strong effect on the resulting solution found. A poor initial solution may result in the PE-NM algorithm evaluating a poor path set and ultimately producing an equilibrium with high system cost. In the rest of this section, we propose solution methods that focus on identifying initial solutions. When combined with the PE-NM algorithm, these methods will heuristically produce good solutions to the SO-NM problem.

4.4.1. Initialising solutions from other TA problems

Rather than constructing an initial solution through the AON algorithm, it is possible to use the path flows from a previous TA solution as a starting point. These previous solutions will generally have well-distributed flows, and will ideally be similar to the path flows of the solution for the SO-NM problem. A promising solution to use as an initial starting point for the SO-NM problem is the equilibrium that results from the TA problem that minimises total travel time — the unique SO travel time (SOTT) solution. Another solution, specific to the context of emissions, is the SO solution that minimises emissions with emission-minimising speed limits – the SO limited emissions (SOEM-L) solution – found by using emissions costs described in Eq. (11). Once the path flows are loaded onto the network from a previous solution, the PE-NM algorithm can be carried out to identify a local optimum of the SO-NM problem.

4.4.2. Weighted combination of travel time and emission cost

An equilibrium with low emissions involves users taking routes on which they drive at or near emissions-minimising speeds. This behaviour means that, in large networks with low total demand, users seeking to minimise emissions may take somewhat indirect routes to their destinations, actively crossing paths with other OD pairs to increase congestion and attain a lower, emission-minimising speed. In reality, this type of route choice could be un-intuitive as network users are unlikely to be fully aware of the emissions they are causing, and also unlikely to be willing to minimise solely emissions. Hence, a weighted objective of emissions and travel time could provide more practical routes, which also leads to an alternative approach to solving SO-E outlined in the following.

Consider an arc cost function of

$$c_a(f_a) = \lambda t_a(f_a) + s_a e_a(v_a(f_a)), \tag{14}$$

where λ is some non-negative weighting of travel time. Moreover, recall that $t_a(f_a)$ (Eq. (1)) is an increasing function while $e_a(v_a(f_a))$ (Eq. (10)) is a non-monotonic function. Here we focus on the SO-E problem, but the same method may apply to any SO-NM problem. First observe that, for a large enough λ, the corresponding TA problem is convex with respect to the total-cost objective and, therefore, has a unique solution. As the value of λ is decreased, the solution comes closer to the desired solution of the SO-E problem (where $\lambda = 0$ gives the emissions cost function), but will result in a non-convex SO for a value of λ less than some certain threshold value λ^*. We now investigate for what range of λ the weighted sum of emissions and travel time as given by Eq. (14) is non-decreasing, giving a convex SO problem.

Lemma 3. *The weighted sum of emissions and travel time as given by Eq. (14) is non-decreasing for $\lambda \geq v_a(f_a)^2 \cdot e'_a(v_a(f_a))$ for all $f_a \geq 0$.*

Proof. We have:

$$c_a(f_a) = \lambda t_a(f_a) + s_a e_a(v_a(f_a)) \tag{15a}$$

$$= \lambda t_a(f_a) + s_a e_a\left(\frac{s_a}{t_a(f_a)}\right) \tag{15b}$$

$$c'_a(f_a) = t'_a(f_a)\left(\lambda - s_a^2 e'_a\left(\frac{s_a}{t_a(f_a)}\right)\frac{1}{t_a(f_a)^2}\right) \tag{16}$$

Since $t'_a(f_a) \geq 0$, in order to obtain $c'_a(f_a) \geq 0$ for all $f_a \geq 0$ from (16), we need

$$\lambda - s_a^2 e'_a\left(\frac{s_a}{t_a(f_a)}\right)\frac{1}{t_a(f_a)^2} \geq 0.$$

If we rearrange the inequality and substitute $v_a(f_a) = s_a/t_a(f_a)$, this leads to

$$\lambda \geq v_a(f_a)^2 \cdot e'_a(v_a(f_a)). \quad \square \tag{17}$$

We wish to find λ^*, the smallest non-negative value of λ such that $c'_a(f_a) \geq 0$ for all $f_a \geq 0$.

Theorem 5. *Given an increasing travel-time function $t_a(f_a)$ and a non-monotonic emissions function $e_a(v_a(f_a))$, with $e_a(v_a)$ being convex and 'u-shaped' as in Eq. (8), and a decreasing $v_a(f_a)$, we have $\lambda^* = v_a(0)^2 e'_a(v_a(0))$.*

Proof. The problem of finding λ^*, the smallest value of λ such that $c'_a(f_a) \geq 0$ for all $f_a \geq 0$, is equivalent to finding

$$\lambda^* = \max_{f_a \geq 0}\left\{v_a(f_a)^2 e'_a(v_a(f_a))\right\}. \tag{18}$$

As $e_a(v_a(f_a))$ is convex and 'u-shaped' with respect to average speed, there exists f_a^* that corresponds to an emissions-minimising speed $v_a(f_a^*)$, such that $e'_a\left(v_a(f_a^*)\right) = 0$.

For $f_a > f_a^*$, $e_a(v_a(f_a))$ is increasing in f_a and any non-negative λ will give a non-decreasing $c_a(f_a)$. Therefore, λ^* will be found in the interval $[0, f_a^*]$. As $v_a(f_a)^2 e'_a(v_a(f_a))$ is a decreasing function of f_a for $f_a \in [0, f_a^*]$, we obtain

$$\lambda^* = v_a(0)^2 e'_a(v_a(0)). \quad \square$$

For a given network, we can identify the maximum free-flow speed over all arcs, allowing the computation of λ^* to guarantee a monotonic arc cost for all arcs in the network, which results in a convex TA problem with a unique solution. This is demonstrated in Fig. 6, where we can additionally note that, for speeds less than or equal to the emission-minimising speed, any non-negative choice of λ will give an increasing cost function $c_a(f_a)$.

To obtain a solution to the SO-E problem from this, the path flows from the resulting convex SO problem can be set as an initial solution for a new problem with a smaller λ. This problem is now non-convex, but should have a solution that is close to the SO travel-time solution. This process can be continued, initialising the next problem with path flows from the previous, iteratively decreasing λ until reaching $\lambda = 0$ and giving a solution to the SO-E problem. Iterations are made by taking n steps between the smallest value of λ that produces a convex problem to $\lambda = 0$. We will refer to this process as *λ-Step* (λ-S) when comparing the methods.

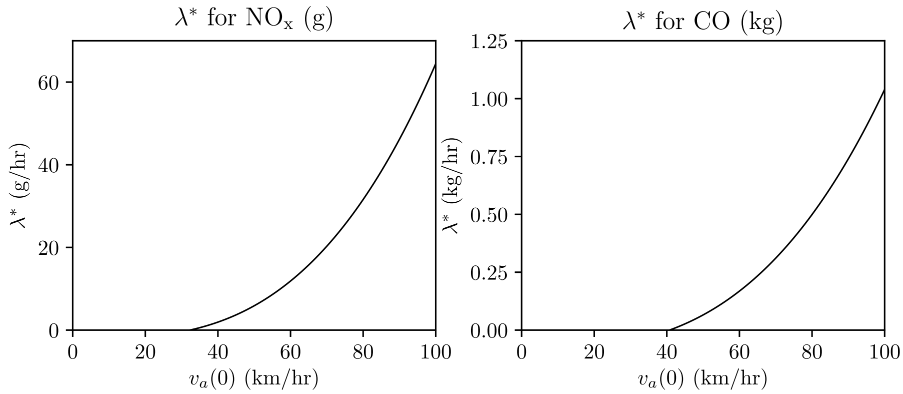

Fig. 6. The minimum value λ^* for λ required to obtain an increasing arc cost function for a given free-flow speed $v_a(0)$ for NO_x and CO emissions.

4.4.3. *Convex combination as initial solution*

Extending the idea of using previously found solutions as initial solutions as in Section 4.4.1, multiple previous solutions can be used to derive an initial solution. A convex combination of path flows from different solutions gives a wide range of initial starting solutions with a promising set of initial paths. A set of promising initial solutions in the context of emissions consists of the SOEM-L, SOTT, and the solution from the final iteration of the λ-S method (Section 4.4.2). As the λ-S solution is a part of the initial solution set, many solutions evaluated by the PE-NM may be similar or identical to the λ-S solution; however, they should explore a large portion of nearby solutions. We will refer to this as the *Convex Combination* (CC) method when comparing the methods.

4.4.4. *Random initial solutions*

One of the issues when producing solutions for the SO-NM problem is that many promising paths are not considered within the PE-NM algorithm, as outlined in Section 4.3.2. A way around this is to save all paths identified during the PE-NM algorithm for a given TA problem, and build up a *path set* from a range of TA problems with different objectives. Considering a wide range of TA problems provides a more extensive path set. We can then generate an initial solution from a random selection of paths with a random distribution of OD demand over them. Specifically, for a given OD pair, a random integer n between 1 and the total number of paths from the path set for this OD pair are chosen to be active via a geometric distribution. Then, n uniform random numbers between 0 and 1 are generated and attributed to each of the active paths. These are normalised such that they sum up to 1, and the corresponding proportion of the demand for the OD pair is allocated to the respective active paths. This is repeated for all OD pairs to give an initial solution. We will refer to this as the *Random Restart* (RR) method when comparing the methods.

5. Computational results

We investigate solving the SO-E problem with PE-NM for a range of networks from Transportation Networks for Research Core Team (2020). We employ the λ-Step (λ-S), Convex Combinations (CC), and Random Restarts (RR) methods outlined in Section 4.4 in order to identify solutions to the SO-E problem for CO and NO_x emissions for the networks of Anaheim, Barcelona, and Winnipeg. The Anaheim network consists of 416 nodes, 914 arcs, and 1406 OD pairs, the Barcelona network has 1020 nodes, 2522 arcs, and 7922 OD pairs, and the Winnipeg network has 1052 nodes, 2836 arcs, and 4345 OD pairs.

The networks of Barcelona and Winnipeg are considered large. The units for these networks are not stated, we assume units of kilometres and minutes. As there tend to be more negative cycles in networks with NO_x arc cost functions (which has a lower emissions-minimising speed), the SSP is required to be solved more often. Further, the size of the networks of Barcelona and Winnipeg means that the SSP takes an extremely long time to solve exactly. To allow the identification of a range of solutions for NO_x for these larger networks, we remove the SSP search and solely rely on the simple path heuristic to identify good paths rather than SPs. This means that the presented solutions for minimising NO_x emissions in the networks of Barcelona and Winnipeg are only equilibria with respect to the final active path set and not the network as a whole, in order to allow the identification of solutions in a reasonable time. This is similar to the use of fixed path sets, which is a common approach in the literature (Cascetta et al., 1997; Lo and Chen, 2000; Chen et al., 2012). Solutions are computed to a tolerance such that the relative gap (RGAP) value is less than 10^{-5}, where the RGAP is commonly used to determine convergence of TA algorithms (Dial, 2006; Nie, 2010; Perederieieva et al., 2015). Specifically, we solve to a tolerance of 10^{-3}, and then resume the algorithm with an updated tolerance of 10^{-5} as this consistently solved faster than using a tolerance of 10^{-5} from the beginning.

In Section 5.1, we discuss the results obtained by our proposed solution methods for the network of Anaheim in detail to demonstrate the techniques in the context of CO and NO_x emissions. We then present general results of the solution methods for all networks in Section 5.2.

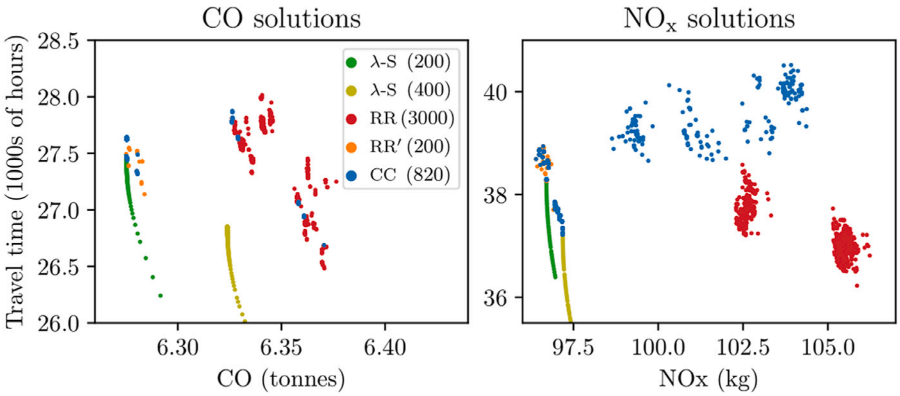

Fig. 7. Local solutions for the Anaheim network found for methods to minimise NO_x and CO emissions versus the total travel time for the corresponding solution. Here, λ-S is the λ-Step method, RR random restarts, RR′ improved random restarts, and CC convex combination.

5.1. Solutions methods applied to the Anaheim network

The network of Anaheim is of medium size, which is well suited for a detailed examination of the results generated by the different solution methods. The total emissions and total travel time of all equilibria found though the employed methods are displayed in Fig. 7. We can immediately note the variation in equilibria for both types of emissions, which are slightly more widespread for NO_x. For this analysis, the λ-S method was run twice for each emission type, once with 200 steps and once with 400 steps, with each run beginning with the same initial solution. The CC method used path sets from the solutions of SOTT, SOEM-L, and the final result from the λ-S with 200 steps, evaluating 820 combinations. The RR method was run 3000 times. A second smaller set of paths was constructed from only the paths making up equilibria in the λ-S method and used to produce the solutions of RR′ as a means of exploring the solution space around a specific area. The path set for RR′ is a subset of the RR path set, such that initial solutions of RR′ should lie closer to the solutions of the λ-S method than those of RR. We make use of RR′ to attempt to reduce the number of iterations the RR method would have to be run in order to obtain good quality solutions. The RR′ method was run 200 times for this analysis.

The λ-S method produced good quality solutions in a short time, taking 1200 s for 200 steps to produce a single solution. Using 200 steps produced a lower total cost at the final iteration compared to using 400 steps. This is most likely due to the smaller step size forcing the algorithm to pick very similar solutions – in this case identifying a worse local equilibrium – while the larger step size for the λ-S method allows it to 'jump' to a different local equilibrium. For this particular problem, the equilibrium identified with fewer steps (200) happened to have a lower total cost ultimately, and a solution of similar quality could be found using 20 steps (taking 370 s). However, this behaviour cannot be guaranteed to occur and was not consistent between networks. The CC method produced a range of different quality solutions – dependent on the initial path set used – which took the PE-NM algorithm 1200 s to find 200 solutions on average. Due to the large number of paths added for the RR method, the time to compute solutions was long, where the PE-NM took approximately 5600 s to find 200 solutions. In comparison, the RR′ method with a smaller path set resulted in the PE-NM taking 1000 s to find 200 solutions on average. An advantage of the CC and RR (and RR′) methods is that they can also be run in parallel, speeding up computation time.

The solution method proposed by Sugawara and Niemeier (2002) – a variation of the Frank–Wolfe algorithm with simulated annealing (FW-SA) in the line search – was implemented for comparison, adjusted similarly to the PE-NM algorithm to identify good paths and SSPs when required. This method produces similar quality solutions to that of the RR method and, like the RR method, it needs to be run multiple times to ensure wide coverage of solutions. However, the FW-SA algorithm required a much lower tolerance to converge to a solution and requires calibration of simulated annealing parameters. The Frank–Wolfe algorithm is known to have issues obtaining higher levels of convergence, which can be seen in the comparison between the PE-NM and FW-SA algorithms in Fig. 8. Because of this, the FW-SA algorithm requires significantly more time to find a solution than the PE-NM algorithm, especially for larger networks.

5.2. General performance of the solution methods

In this section, we consider the application of the solution methods to various networks. The results of employing solution methods to the SO-E problem for CO emissions for the networks of Anaheim, Barcelona, and Winnipeg are shown in Table 1, and NO_x emissions in Table 2. Table 3 lists run-times of the different variants of PE-NM. We additionally include the results from using the PE-NM algorithm on an initial solution found from the AON algorithm. While the solutions are equilibria that minimise emissions (listed as "EM"), the total travel-time cost for the network is also of interest (listed as "TT") and is used to further compare equilibria in Tables 1 and 2. The equilibrium denoted by "SOEM-L" minimises the corresponding emission (either CO or NO_x) with optimal speed limits (11) that are imposed to ensure that emission functions become non-decreasing (Raith et al., 2016). SOEM-L is an

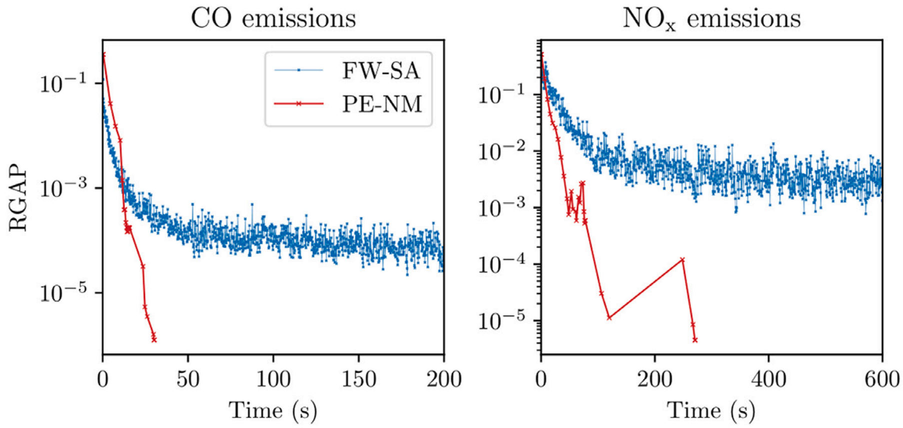

Fig. 8. The RGAP value over time, comparing algorithms PE-NM and FW-SA (implementation as proposed in Sugawara and Niemeier, 2002) on the Anaheim network for emission types CO and NO_x when initialised with the same AON assignment. The FW-SA algorithm was terminated after 200 and 600 s for the CO and NO_x problems, respectively.

Table 1
Total emission costs (EM) for the best solutions found using various methods, for CO emissions. The values for the various methods are shown as a percentages relative to the SOEM-L solution. The emission cost is presented in kg and the total travel time (TT, shown for interest) is in days.

Network		SOEM-L	AON (%)	λ-S (%)	CC (%)	RR (%)	RR′ (%)
Anaheim	EM	5411.9	116.98	115.95	115.95	115.95	115.95
	TT	1316.2	87.25	86.94	86.94	87.01	86.97
Barcelona	EM	4015.3	106.52	106.50	106.50	106.48	106.50
	TT	674.9	82.45	82.40	82.31	82.84	82.66
Winnipeg	EM	2701.6	105.66	105.45	105.45	105.45	105.45
	TT	674.9	85.16	85.67	85.81	85.88	85.82

Table 2
Total emission costs (EM) for the best solutions found using various methods, for NO_x emissions. The values for the various methods are shown as a percentages relative to the SOEM-L solution. The emission cost is presented in kg and the total travel time (TT) is in days.

Network		SOEM-L	AON (%)	λ-S (%)	CC (%)	RR (%)	RR′ (%)
Anaheim	EM	54.3	203.29	178.10	177.84	177.23	177.44
	TT	1649.0	93.43	97.54	97.56	99.05	97.68
Barcelona	EM	40.3	162.13	149.04	148.59	147.99	148.09
	TT	1228.3	102.26	91.96	92.29	93.61	93.19
Winnipeg	EM	27.0	146.17	141.46	141.46	140.48	140.86
	TT	825.0	99.72	92.77	92.79	95.25	94.48

idealised solution assuming the necessary speed limits can be enforced on all arcs. However, as noted in the introduction, these speed limits may not be easily enforceable throughout the network and also lead to unrealistically low speed limits on highways, for example.

The SOEM-L solution is used as a reference since it obtains the minimal total emissions cost for the network, but its travel time is relatively high due to the imposed speed limits. The SOEM-L solution can be found relatively quickly with the standard PE algorithm since the arc cost functions are non-decreasing once speed limits are imposed. The solution values for the various solution methods are therefore shown as percentages relative to the SOEM-L solution.

In general, SO-E solutions have a lower travel time cost and a higher emissions cost when compared to the SOEM-L solution, since the optimal speeds enforced by speed limits in the SOEM-L solution are typically unobtainable in the SO-E solutions that do not use speed limits. Fig. 9 presents the difference in arc flows between the SOEM-L solution and the λ-S solution using 200 steps for CO emissions in the network of Anaheim. In the figure, red arcs indicate higher flow in the λ-S solution while blue arcs indicate higher flow in the SOEM-L solution. The dark blue arcs have significantly higher flow in the SOEM-L solution, while the dark red arcs have significantly higher flow in the λ-S solution, where both dark blue and red arcs are primarily highways. This difference is likely attributed to the enforced speed limits in the SOEM-L solution ensuring that users drive at emission-minimising speeds despite having low flow, whereas the λ-S solution requires users to take the same route as other network users (for instance dark red arcs) in order to drive at speeds close to the emissions-minimising speed. Furthermore, the λ-S solution has higher flow for some residential roads (primarily light red arcs) when compared to the SOEM-L solution, likely attributed to users traversing arcs with free-flow speeds that are closer to the emissions-minimising speed.

Table 3
CPU time in minutes for the PE-NM algorithm to find an equilibrium solution for each solution method for both CO and NO_x emissions. The SOEM-L solutions are found using the PE algorithm. The λ-S method was run using 50 steps, the CC method for 55 combinations, and RR and RR′ both 100 times.

Network		SOEM-L	AON	λ-S (50)	CC (55)	RR (100)	RR′ (100)
Anaheim	CO	0.0	0.5	2.2	9.5	23.3	18.2
	NO_x	0.1	4.3	49.5	388.2	311.5	366.4
Barcelona	CO	2.0	2.4	36.5	50.7	416.8	331.6
	NO_x	1.9	43.1	654.1	1955.6	6336.2	6555.8
Winnipeg	CO	2.0	2.9	35.1	51.7	455.7	412.4
	NO_x	3.6	128.0	1330.5	4658.2	6812.2	7761.8

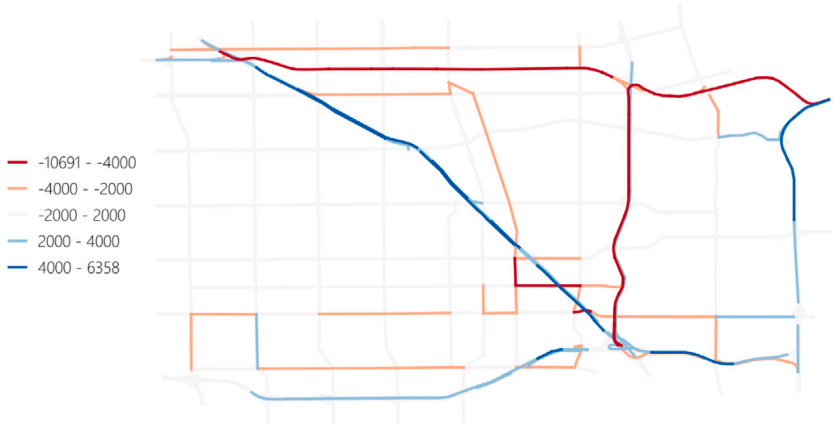

Fig. 9. Flow difference between the SOEM-L solution and the λ-S solution using 200 steps for CO emissions in the network of Anaheim. Here, the blue arcs indicate higher flow in the SOEM-L solution, while the red arcs indicate higher flow in the λ-S solution. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

For the λ-S method, we present the best solution from three runs, using 30, 50, and 80 steps. The CC method was employed using path sets from the solutions of SOTT, SOEM-L, and the final result from the best solution found from the λ-S method, with 55 combinations evaluated (using 10 equally spaced weightings between each two solutions). The RR method was run 100 times with a large path set of saved paths from every SP evaluated during the λ-S method with 50 steps. The RR′ method was also run 100 times, with a smaller path set constructed from only the paths making up equilibria in the λ-S method with 50 steps.

With respect to the results shown in Table 1, we can note that all applications of the PE-NM algorithm produce relatively similar solutions for each network. This indicates that there may be few local minima for the CO objective — possibly due to the relatively high emissions-minimising speed for CO emissions (40.8 km/h), however, we cannot guarantee that there is not a better quality solution.

The solutions found from applying the PE-NM algorithm to the AON assignment initialisation for the test networks are of good quality for the CO emissions and may be sufficient to use rather than employing the solution methods. In general, all methods produce good quality solutions. However, the CC and RR methods as described here are dependent on using the results from the λ-S. Furthermore, the CC and RR methods may need to be run a large number of times to produce a good solution, as using a single instance may produce a poor quality equilibrium. As the λ-S method generally produces good quality solutions, it may be sufficient to rely solely on the λ-S method. If higher quality solutions are desired, the CC and RR (or RR′) methods could then be employed to seek further improvement of the result of the λ-S method. However, these results are specific to the presented networks, such that similar behaviour of results may not be seen for other networks.

In contrast to the CO results in Table 1, the NO_x results in Table 2 indicate that solely using the PE-NM algorithm on the AON initial solution may produce a low-quality equilibrium. The remaining solution methods obtain equilibria of lower cost due to the ability to explore the solution space to a higher degree. Additionally, the low emissions-minimising speed of NO_x (32.3 km/h) leads to more locally optimal solutions.

The time taken to find an equilibrium using the PE-NM algorithm for the various solution methods is shown in Table 3. This includes finding a solution from an AON initial solution, using a step size of 50 for the λ-S method, 55 combinations evaluated for the CC method, and 100 runs for the RR and RR′ methods. The SOEM-L solutions are found using the PE algorithm. The CO solution for networks can be found relatively quickly – likely due to the high emissions-minimising speed – while the NO_x solution has a lower emissions-minimising speed, which produces more negative-cost cycles. The NO_x solution for the Winnipeg network takes much longer than one would expect to solve, considering the networks of Barcelona and Winnipeg are of similar size. The difference in solution times is likely due to the network of Barcelona having approximately three times the demand in the network compared to Winnipeg, 185,000 units of flow versus 65,000 (while Anaheim has 105,000). The lower demand for the Winnipeg network means there will be more arcs with negative costs, which are produced at low-flow, high-speed values. This produces more negative cycles in the network, such that difficult SSP calculations must be completed more often as well as heuristic methods to find good paths — which are not as efficient as common SP algorithms.

6. Conclusion

In this paper, we investigate issues that arise when arc costs in TA are non-monotonic. Various initialisation methods combined with an adjusted descent algorithm are tested to find good solutions when the optimal solution is difficult to find.

In general, the proposed method labelled as λ-Step performed well for the tested networks, identifying good quality solutions consistently. Additionally, the combination of travel time and emission objectives employed within the λ-Step method is of interest when a pure emission objective is not practical in terms of realistic costs to the users or their proposed routes to minimise emissions. The resulting flow distributions are not trivial to enforce, where calculating realistic tolls is complex due to the non-monotonic arc costs.

Further research includes an investigation into modifying the PE algorithm or other similar equilibrium algorithms to find good solutions to the TA problem with non-monotonic arc costs. This may include studying the influence of different ordering methods of OD pair equilibration in PE-NM on the resulting solution quality, or determining alternative paths to add to the active path set instead of solely SPs to improve the solution. Furthermore, methods of escaping local stationary points should be investigated to further improve equilibrium solutions.

Another direction of future research is related to tolling. TA literature investigates how to enforce a particular traffic pattern obtained as an SO solution by adequate tolling, ensuring that a desired SO traffic pattern can be obtained as UE flow. While this is well-understood under standard assumptions, tolling would be applied in the context of non-monotonic arc cost functions when considering SO-NM and SO-E. Here, we cannot guarantee the existence of revenue-neutral or non-negative tolling schemes, given the network is not acyclic and the arc costs are non-monotonic (Chen and Yang, 2012). However, it may still be possible to identify schemes that involve both positive and negative tolls (Chen and Yang, 2012).

Future research could also consider the emissions of different types of vehicles separately by modelling each vehicle type (or class) with individual OD demand as well as distinct arc cost functions for emissions. TA problems with type-specific arc cost functions are theoretically challenging, even in the case where arc cost functions are not non-monotonic (Sheffi, 1984). A simpler case is a TA problem where the different types of vehicles affect each others' arc cost functions symmetrically (Sheffi, 1984), however, this would not be the case when considering emission functions by vehicle type. Alternative fuel vehicles with different emission profiles could be modelled similarly, again leading to the outlined challenges.

CRediT authorship contribution statement

J. Tidswell: Methodology, Investigation, Data curation, Writing – original draft, Software, Visualization. **A. Downward:** Methodology, Formal analysis, Writing – review & editing. **C. Thielen:** Methodology, Formal analysis, Writing – original draft, Writing – review & editing. **A. Raith:** Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Acknowledgements

This work was partially supported by the New Zealand National Science Challenge "Science for Technological Innovation" [grant number UOAX1707]. The work of Clemens Thielen was partially supported by the Bavarian State Ministry of Education, Science and the Arts within the Bavarian Funding Programme for the Initiation of International Cooperations (BayIntAn) [grant number BayIntAn_KUEI_2016_130].

References

Arnott, R., de Palma, A., Lindsey, R., 1990. Departure time and route choice for the morning commute. Transp. Res. B 24, 209–228.

Aziz, H., Ukkusuri, S., 2012. Integration of environmental objectives in a system optimal dynamic traffic assignment model. Comput.-Aided Civ. Infrastruct. Eng. 27, 494–511.

Bazaraa, M.S., Sherali, H.D., Shetty, C.M., 2013. Nonlinear Programming: Theory and Algorithms. John Wiley & Sons.

Bellman, R., 1958. On a routing problem. Quart. Appl. Math. 16 (1), 87–90.

Benedek, C.M., Rilett, L.R., 1998. Equitable traffic assignment with environmental cost functions. J. Transp. Eng. 124, 16–22, URL: https://doi.org/10.1061/(ASCE)0733-947X(1998)124:1(16).

Bureau of Public Roads, 1964. Traffic Assignment Manual. U.S. Department of Commerce, Urban Planning Division, Washington D.C, URL: http://hdl.handle.net/2027/ien.35556036790061?urlappend=%3Bseq=156.

Cascetta, E., Russo, F., Vitetta, A., 1997. Stochastic user equilibrium assignment with explicit path enumeration: Comparison of models and algorithms. IFAC Proc. Vol. 30 (8), 1031–1037.

Chabrier, A., 2006. Vehicle routing problem with elementary shortest path based column generation. Comput. Oper. Res. 33 (10), 2972–2990.

Chen, L., Yang, H., 2012. Managing congestion and emissions in road networks with tolls and rebates. Transp. Res. B 46 (8), 933–948, URL: https://doi.org/10.1016/j.trb.2012.03.001.

Chen, A., Zhou, Z., Xu, X., 2012. A self-adaptive gradient projection algorithm for the nonadditive traffic equilibrium problem. Comput. Oper. Res. 39 (2), 127–138.

Chiu, Y.-C., et al., 2011. Dynamic Traffic Assignment: A Primer (Transportation Research Circular E-C153). Transportation Research Board.
Dafermos, S., Sparrow, F., 1969. The traffic assignment problem for a general network. J. Res. Natl. Bureau Stand. B 73B, 91–118, URL: https://doi.org/10.6028/jres.073b.010.
Dial, R.B., 2006. A path-based user-equilibrium traffic assigment algorithm that obviates path storage and enumeration. Transp. Res. B 40 (10), 917–936, URL: https://doi.org/10.1016/j.trb.2006.02.008.
Drexl, M., Irnich, S., 2014. Solving elementary shortest-path problems as mixed-integer programs. OR Spectrum 36 (2), 281–296.
Eppstein, D., 1998. Finding the k shortest paths. SIAM J. Comput. 28 (2), 652–673.
Feillet, D., et al., 2004. An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems. Netw.: Int. J. 44 (3), 216–229.
Florian, M., Hearn, D., 1995. Network equilibrium models and algorithms. Handb. Oper. Res. Manag. Sci. 8, 485–550.
Frank, M., Wolfe, P., 1956. An algorithm for quadratic programming. Nav. Res. Logist. 3 (1–2), 95–110.
Hershberger, J., Maxel, M., Suri, S., 2007. Finding the k shortest simple paths: A new algorithm and its implementation. ACM Trans. Algorithms 3 (4), 45:1–45:19.
Karger, D., Motwani, R., Ramkumar, G.D.S., 1997. On approximating the longest path in a graph. Algorithmica 18, 82–98.
Karp, R.M., 1972. Reducibility among combinatorial problems. In: Miller, R.E., Thatcher, J.W. (Eds.), Complexity of Computer Computations: Proceedings of a Symposium on the Complexity of Computer Computations. In: The IBM Research Symposia Series, Plenum Press, pp. 85–103.
Lo, H.K., Chen, A., 2000. Traffic equilibrium problem with route-specific costs: formulation and algorithms. Transp. Res. B 34 (6), 493–513.
Long, J., Chen, J., Szeto, W., Shi, Q., 2018. Link-based system optimum dynamic traffic assignment problems with environmental objectives. Transp. Res. D 60, 56–75.
Nie, Y.M., 2010. A class of bush-based algorithms for the traffic assignment problem. Transp. Res. B 44 (1), 73–89.
Patil, G.R., 2016. Emission-based static traffic assignment models. Environ. Model. Assess. 21, 629–642, URL: https://doi.org/10.1007/s10666-015-9498-7.
Patriksson, M., 2015. The Traffic Assignment Problem: Models and Methods. Courier Dover Publications.
Penic, M.A., Upchurch, J., 1992. TRANSYT-7F: Enhancement for fuel consumption, pollution emissions, and user costs. Transp. Res. Rec. 1360, 104–111.
Perederieieva, O., et al., 2015. A framework for and empirical study of algorithms for traffic assignment. Comput. Oper. Res. 54, 90–107.
Raith, A., Thielen, C., Tidswell, J., 2016. Modelling and optimising fuel consumption in traffic assignment problems. In: Proceedings of the 38th Australasian Transport Research Forum (ATRF), pp. 1–13.
Righini, G., Salani, M., 2008. New dynamic programming algorithms for the resource constrained elementary shortest path problem. Netw.: Int. J. 51 (3), 155–170.
Rilett, L.R., Benedek, C.M., 1994. Traffic assignment under environmental and equity objectives. Transp. Res. Rec.: J. Transp. Res. Board 1443, 92–99.
Roughgarden, T., 2002. Selfish Routing (Ph.D. thesis). Cornell University.
Sheffi, Y., 1984. Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods. Prentice-Hall.
Song, Y.Y., Yao, E.J., Zuo, T., Lang, Z.F., 2013. Emissions and fuel consumption modeling for evaluating environmental effectiveness of ITS strategies. Discrete Dyn. Nat. Soc. 581945, URL: https://doi.org/10.1155/2013/581945.
Sugawara, S., Niemeier, D., 2002. How much can vehicle emissions be reduced?: Exploratory analysis of an upper boundary using an emissions-optimized trip assignment. Transp. Res. Rec.: J. Transp. Res. Board 1815, 29–37, URL: https://doi.org/10.3141/1815-04.
Transportation Networks for Research Core Team, 2020. Transportation networks for research. https://github.com/bstabler/TransportationNetworks (accessed May 2020).
Vickrey, W., 1969. Congestion theory and transport investment. Amer. Econ. Rev. 59, 251–260.
Wang, Y., et al., 2018. Dynamic traffic assignment: A review of the methodological advances for environmentally sustainable road transportation applications. Transp. Res. B 111, 370–394.
Wardrop, J.G., 1952. Some theoretical aspects of road traffic research. Proc. Inst. Civ. Eng. Part II (1–2), 325–378, URL: https://doi.org/10.1680/ipeds.1952.11259.
Yang, H., Wang, X., Yin, Y., 2012. The impact of speed limits on traffic equilibrium and system performance in networks. Transp. Res. B 46 (10), 1295–1307, URL: https://doi.org/10.1016/j.trb.2012.08.002.
Yen, J.Y., 1971. Finding the K shortest loopless paths in a network. Manage. Sci. 17 (11), 712–716.
Yin, Y., Lawphongpanich, S., 2006. Internalizing emission externality on road networks. Transp. Res D 11 (4), 292–301, URL: https://doi.org/10.1016/j.trd.2006.05.003.
Zhang, P., Qian, S., 2020. Path-based system optimal dynamic traffic assignment: A subgradient approach. Transp. Res. B 134, 41–63.