Minimising emissions in traffic assignment with non-monotonic arc costs

J. Tidswell[1], A. Downward[1], C. Thielen[2], and A. Raith[1]

[1]Department of Engineering Science, University of Auckland, New Zealand
[2]TUM Campus Straubing for Biotechnology and Sustainability,
Technical University of Munich, Germany
j.tidswell@auckland.ac.nz[1], clemens.thielen@tum.de[2]

Abstract

The modelling of vehicle emissions within traffic assignment (TA) has been studied in literature, as emissions such as carbon monoxide and carbon dioxide are detrimental to the populaces' health as well as to the environment. TA is employed as a means to identify the potential to reduce vehicle emissions by the manipulation of traffic patterns. Studies that make use of emission arc cost functions in TA generally assume a positive, increasing function, or do not discuss the computational complexities that arise when the cost functions are non-monotonic. In this paper we investigate the issues that exist within TA methodology when the arc costs are non-monotonic, and present adjustments to algorithms to allow a solution to the TA problem to be found. We suggest several methods to employ in order to find good solutions to the TA problem with non-monotonic arc costs. We compare the methods by applying them to several test networks for a range of emission types.

Key words: Traffic assignment, emissions, non-monotonic, optimisation, heuristics.

1 Introduction

The significant contribution of greenhouse gases from vehicles has provoked interest in estimating emissions in traffic models, such as in the Traffic Assignment (TA) model with the aim of identifying traffic patterns that reduce vehicle emissions. The incorporation of emissions functions in TA has been accomplished to some extent, either with the use of emissions functions that are increasing for the given network (Yin and Lawphongpanich 2006), or by enforcing a speed limit, which ensures the resulting emissions functions are increasing (Raith, Thielen, and Tidswell 2016; Patil 2016). While the use of a speed limit is convenient and intuitive, in practice the speed limits required to ensure an increasing function may be unrealistic, and constrain users in an unrealistic manner. In other literature incorporating non-monotonic emission functions, the possibility 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).

We aim to incorporate non-monotonic emission functions into TA in order to find realistic attainable minimal emission costs in traffic networks. This is not trivial, where under the assumption of a positive increasing arc cost function in TA results in a unique minimal system cost, a non-monotonic arc cost may result in multiple local minima.

2 Background

We are given a road transport network represented by a directed graph $G = (V, A)$ and K *commodities* (or *origin-destination (OD) pairs)* $\{(s_i, t_i, d_i)\}_{i=1}^K$, where $s_i \in V$ and $t_i \in V$ 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 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 vectors 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 $f = (f_a)_{a\in A} \in \mathbb{R}^{|A|}$. 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. 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 period.

For every arc there exists an increasing average travel time function $t_a(f_a)$ that could, for example be defined as (Bureau of Public Roads 1964):

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

Average travel speed is then $v_a(f_a) = {}^{s_a}\!/\!{}_{t_a(f_a)}$, where s_a is the length of arc a. Arc emissions are a strictly convex function of speed, but generally not a convex function of arc flow f_a. They could be given as

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

with appropriate choice of parameters $\mathfrak{a}, \mathfrak{b}, \mathfrak{c}, \mathfrak{d}$ according to emission and vehicle type (Song et al. 2013). Emissions can then also be expressed in terms of f_a as

$$e_a(f_a) = e_a(v_a(f_a)). \tag{3}$$

In general, an arc cost function can be defined as $c_a(f_a)$, such that we can formulate the following optimisation problem:

$$\begin{aligned}
\min \quad & \sum_{a\in A} f_a \cdot c_a(f_a) && && \text{(4a)}\\
\text{s.t.} \quad & \sum_{p\in\mathcal{P}_i} h_p = d_i, && i = 1, \ldots, K && \text{(4b)}\\
& f_a = \sum_{p\in\mathcal{P}:a\in p} h_p, && a \in A && \text{(4c)}\\
& h_p \geq 0, && p \in \mathcal{P} && \text{(4d)}
\end{aligned}$$

where the solution to this problem is known as the *system optimal* (SO) solution to the TA problem, which minimises the total general cost in the system.

For an arc cost of $c_a = e_a(f_a)s_a$ the objective may not be convex (or pseudo-convex or quasi-convex), as Example 3.1 will illustrate, hence there may be locally optimal solutions. This problem has been also investigated in Sugawara and

Niemeier (2002) by using simulated annealing to find a local solution, however they do not address how they bypass the negative arc costs that occur when solving an SO problem with non-monotonic arc costs, which is shown to occur in Example 3.2.

Previous work has ensured an increasing emissions function by incorporated a speed limit $v_a^{\max}$ such that $\hat{v}_a(f_a) := \min\{v_a(f_a), v_a^{\max}\}$, giving the emission rate with an enforced speed limit as:

$$\hat{e}_a(f_a) := e_a(\hat{v}_a(f_a)), \tag{5}$$

Setting $v_a^{\max} = \arg\min_{v_a} e_a(v_a)$, we can find a lower bound for the TA problem with emissions costs, albeit with a network-wide speed limit set at the unique speed that minimises the corresponding emission.

2.1 Methodology for solving the common TA problem

Here we outline a method for solving the more common TA problem with positive non-decreasing arc costs. In general, we find an initial feasible solution by All Or Nothing (AON) assignment and then descend to an equilibrium by means of the Path Equilibration (PE) algorithm, employing a bisection method line search to equilibrate flows, as outlined in the following.

For AON assignment, the network G begins with zero flow on all arcs, such that arcs have a cost of $c_a(0)$, $\forall\, a \in A$. We define the set of active paths $P_+ = \{\emptyset\}$ which will contain paths with positive flow or zero-flow paths to be equilibrated. For each OD-pair defined in $\{(s_i, t_i, d_i)\}_{i=1}^K$ the shortest path (SP) is found from s_i to t_i. This path is then assigned flow equal to d_i, and it is added to P_+. For each added path, all arcs that make up the path have their flows updated to be consistent with the added path flows. To find the SPs we use Dijkstra's single-source SP algorithm. After all demand is assigned for all OD-pairs the arc costs are updated to $c_a(f_a)$.

Having found an initial solution, we employ the PE algorithm, which involves iterations of adding new SPs and shifting path flows for each OD-pair individually until all are at an equilibrium, and which terminates when the solution has converged to a predetermined level of accuracy. Specifically, Dijkstra's single-source SP algorithm is used to add new paths, while a bisection line search is used to find the amount of flow to shift between two paths to give the same path cost. For the equilibration of a given OD-pair s_i, t_i, flow is shifted from the maximum cost path to the minimum cost path until all used paths have equal cost. This is repeated for all OD-pairs $\{(s_i, t_i, d_i)\}_{i=1}^K$ iteratively, until we reach an equilibrium.

To solve the SO problem defined in (4), the PE algorithm is applied with modified arc costs of the form $c_a(f_a) + f_a c_a'(f_a)$, whereas using arc costs of $c_a(f_a)$ would result in a user equilibrium (UE) (Sheffi 1985). For arc cost functions that are strictly increasing, the solution will have unique arc flows and a unique system cost. If the arc cost functions are non-decreasing the equilibrium will have a unique system cost. Other algorithms that descend to an equilibrium exist, such as the Frank-Wolfe algorithm and the so-called Algorithm B (Frank and Wolfe 1956; Dial 2006). These algorithms are employed based on the assumptions that the TA problem has positive, increasing arc costs (Patriksson 2015).

3 TA with non-monotonic arc cost functions

The use of non-monotonic arc cost functions with respect to flow violates the common assumptions made when solving TA problems. As solution methods for system optimal (SO) TA problems involve the derivative of the arc cost, it is possible for arc costs to have negative values. Further, the non-monotonicity of the arc costs results in a non-convex TA problem, such that the global minimum cost solution for the network is difficult to find due to the presence of multiple stationary points. We wish to devise a range of methods to identify 'good' solutions to this specific non-convex TA problem, where the non-monotonic arc cost is in the context of emissions and good solutions have a low emissions cost. To find these good solutions we build on the standard TA problem methodology, where various issues arise when incorporating non-monotonic arc costs. We will outline these issues and suggest ways of addressing them in order to find equilibria.

3.1 Multiple local minima

A network with non-monotonic continuous arc cost functions may have multiple equilibria with different system costs (Roughgarden 2002, Remark B.1.4). We show that even in a simple network such as in Example 3.1 multiple minima in the objective function may exist when the arc cost function is non-monotonic.

The local minima found for the TA problem with non-monotonic arc costs will depend on the initial solution and methodology employed. In Example 3.1 the minimum for the speed restricted case is close to that of the unrestricted case. In other networks this gap may be large, and is dependent on the network properties. It is possible that the global minimum for both problem instances may be of equal value, where there exists a flow f that corresponds to the minimum emissions cost attained by enforcing a speed limit in Equation 5

Example 3.1. *Consider a network with two parallel arcs a and b connecting two nodes with total demand of 400. We can compare the emissions objective cost for all possible flow combinations using the objective stated in Equation* (4a)*, and additionally compare this to the case where emission-minimising speed limits are enforced with Equation* (5)*. This is shown in Figure 1, where we can note one unique minimum for the objective with speed limits (and non-decreasing arc cost function), while the objectives without speed limits have multiple local minima due to the non-monotonic arc cost function.*

3.2 Negative arc costs

When solving the SO TA problem using equilibrium principles, arc costs are defined as $c_a(f_a) + f_a c'_a(f_a)$, where $f_a c'_a(f_a)$ is the user's influence on the cost to other users of the arc. For function $c_a(f_a)$ that is decreasing over some domain, negative SO arc costs may be present in the network, such as in the emissions cost function case as shown in Figure 2. Solving the SO TA problem with non-monotonic arc costs may produce a network with negative cycles, meaning standard SP methods cannot necessarily be employed, and it is now feasible for a user to reduce their cost by cycling. The possibility of cycling is shown in Example 3.2, where cycling would act in a similar way to increasing the demand in a network. As the emission arc cost the function is pseudo-convex, a user would only cycle until they incur a cost of 0 on the

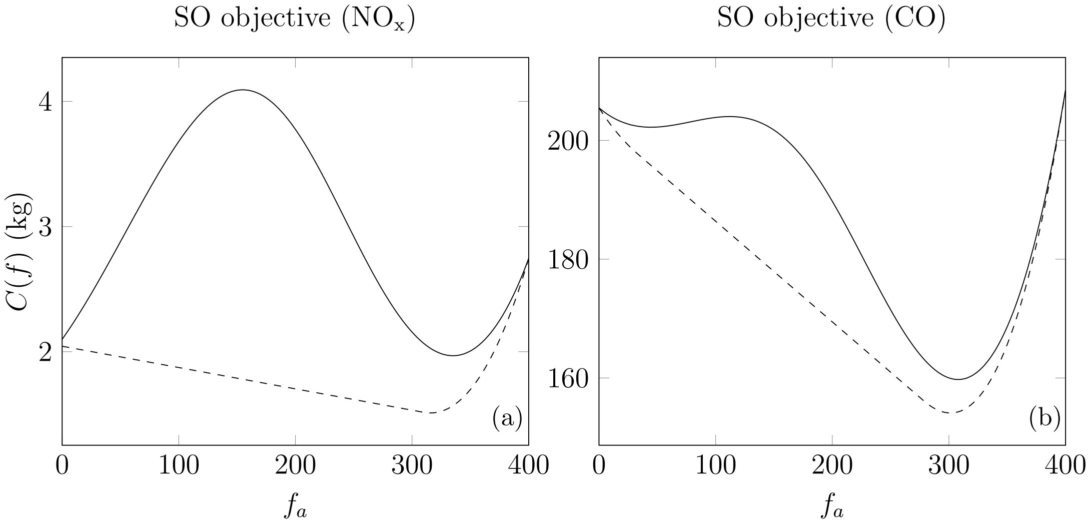

Figure 1: Total system costs for a network with 2 parallel arcs, where the dotted line represents the cost with a emission-minimising speed limit, and the solid line without a speed limit.

arc, rather than cycle indefinitely. In order to proceed and produce a solution that is practical, we assume that all paths taken by users must be simple paths, meaning we must solve the simple shortest path (SSP) problem.

Example 3.2. *Consider a network with two nodes, A and B, with one directed arc from A to B, and another directed arc from B to A with identical arc parameters. The UE cost, $c_a(f_a)$, and the SO cost, $c_a(f_a) + f_a c'_a(f_a)$, are shown for increasing flow in Figure 2. The UE cost is positive for all feasible flows while the SO cost is not. Suppose there is a demand d of 250 units from A to B and the same for B to A. In this case the cost on each arc is negative, producing a negative cycle. If we allow non-simple paths, the optimal solution would be to cycle until the travel cost on the arc is zero (approximately 300 units), with users being represented on the same arc more than once. Notably, if allowing cycles, the optimal solution would involve cycling if the demand is between approximately 150 and 300 units of flow.*

Notably the SSP problem in a network with negative cycles is NP-hard, as it is akin to the longest (simple) path problem. For this reason we attempt to avoid SSP calculations as much as possible by using a heuristic to find a 'good' path rather than the SSP whenever a negative cycle is present, in order to reduce computation time. The search space for the SSP can be reduced by providing a good path found via a heuristic or the cost of an existing path in the active path set as an upper bound.

As AON relies on costs $c_a(0)$, negative cycles will not be present in the initialisation of the TA problem. For PE, whenever we require a SP calculation we check for negative cycles using the Goldberg-Radzik single source SP algorithm (Goldberg and Radzik 1993; Cherkassky, Goldberg, and Radzik 1996) which is capable of finding the SP in a network with negative arc costs. If no negative cycle is found, we can use the resulting SPs from the Goldberg-Radzik algorithm. If there is a negative cycle, we use a heuristic to return a good path rather than the SP, where the good path will have a cost less than or equal to the minimum cost of an active path for the given OD-pair. When the PE algorithm reaches the termination condition, check

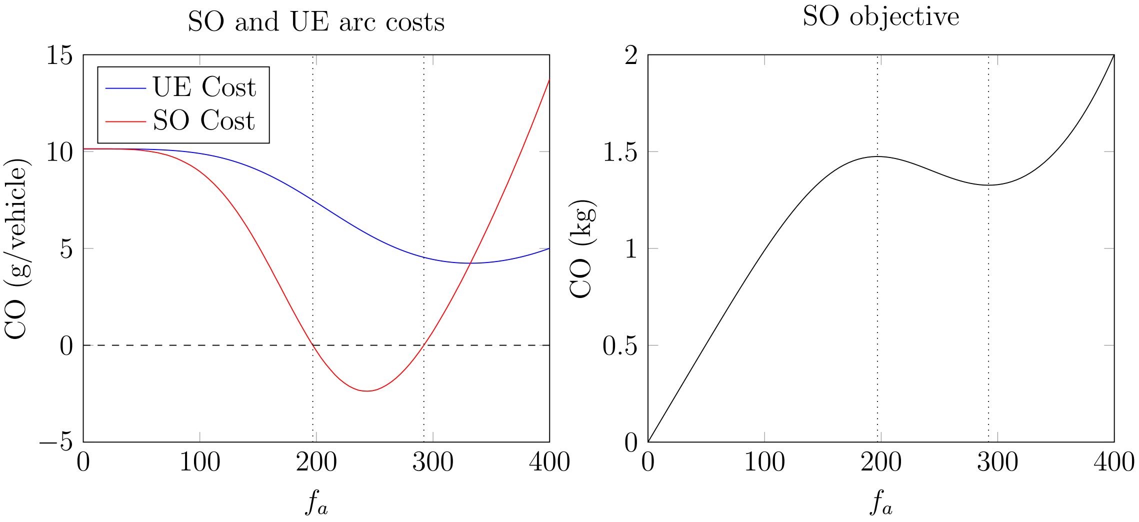

Figure 2: UE and SO costs for a single arc (left), and corresponding total objective for CO emissions (right). Parameters are k of 175, s of 1 km, and v_0 of 120 km/hr.

for a negative cycle. If there are no negative cycles, we are at an equilibrium (as the lack of negative cycles means that there does not exist a shorter unused path for any OD). If a negative cycle has been detected, 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, add them to the set of active paths and restart the PE algorithm. If none are found, we are at an equilibrium.

3.3 Ineffective shortest paths in the PE algorithm

Standard TA algorithms rely on finding SPs to shift flow to in order to reduce the objective value. For decreasing or non-monotonic arc cost functions, the SP path may not be the best path to try and shift flow to, and as shown in Example 3.3, limiting the PE algorithm to adding only SPs prevents the descent to certain equilibria.

A heuristic approach to finding promising paths is to add the k-SPs with minimum possible arc cost, where k is a predetermined number of paths to add. We can then compare all paths pairwise during equilibration to see how the flow would shift (only accepting shifts that actually decrease the objective cost over the arcs of the two compared paths). Testing this approach does not yield promising results, where adding the k-SPs with $k = 10$ rather than the standard methodology of adding the SP resulted in an improvement of less than 0.01% of the system objective in a test case (Anaheim, `https://github.com/bstabler/TransportationNetworks`).

Another method for a given maximum cost path with flow x is to update all arc costs in network to $c(f + \delta)$ except for those arcs on the given path, which would remain at $c(f)$. We can then find the SP with these new costs, where the resulting SP would be optimal for shifting δ units of flow. This would ideally be repeated for values $\delta \in [0, x]$, for every single path, which is not practical.

Example 3.3. *Consider two parallel arcs a and b connecting a single OD-pair with parameters such that we obtain the functions in Figure 3. Suppose there is a flow of*

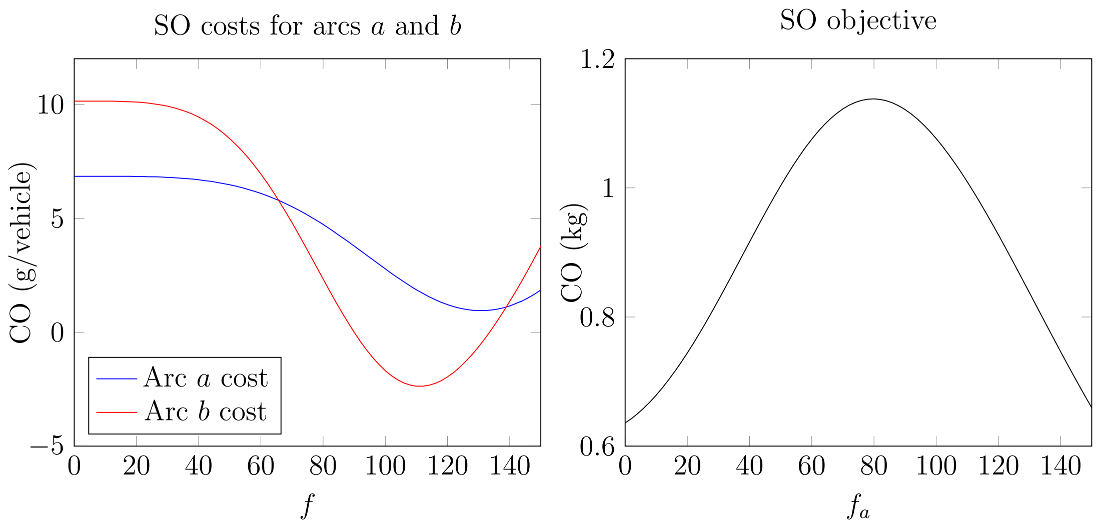

Figure 3: The SO costs for CO emissions of two parallel arcs (left), where allocating all flow to either arc results in a local minimum as seen in the SO objective function (right).

150 *units on arc a. This may occur either through AON initialisation, as* $c_a(0) \leq c_b(0)$*, or part way through the PE stage, where arc b represents an unused path.*

The optimal solution to this problem is to shift all of the flow from arc a to arc b, however as arc b has no flow on it, the current weight is $c_b(0)$*, which is higher than* $c_a(150)$*, such that a SP calculation will not return arc b. Using this methodology we would not consider a as a promising choice, and would not find the global minimum.*

3.4 Multiple solutions in the PE algorithm

Assuming that we have a range of candidate paths across which we wish to distribute flow within the PE algorithm, an issue arises when comparing two paths. Sheffi (1985) Equations 3.26 indicate that paths with equal path costs for an OD-pair correspond to stationary points in the objective cost for the paths. With a non-decreasing arc cost function there is one stationary point that is a minimum for the objective. For a non-monotonic arc cost function it is possible that there are multiple stationary points, as demonstrated in Example 3.4.

To remedy this the PE algorithm must be adapted to identify all stationary points and extreme cases (shifting all or zero flow), evaluate the objective cost at the appropriate flow, and return the flow combination that produces the lowest objective cost. This increases the computation needed to equilibrate each path combination, but will avoid local maximum solutions. However, we still cannot guarantee a globally optimal solution to the overall problem.

This feature of the non-monotonic arc cost TA problem also reveals another issue. The equilibration of two paths is employed when their costs are not equal. However, it is possible that the two paths have the same cost, but a flow distribution that corresponds to a local maximum. In this case we would ideally want to re-distribute the flow to such that the paths have the same cost but are at a local minimum. The adding of SPs during the algorithm may be sufficient to avoid this issue, however in specific networks it may be necessary to check that no paths lie at a local maximum objective cost.

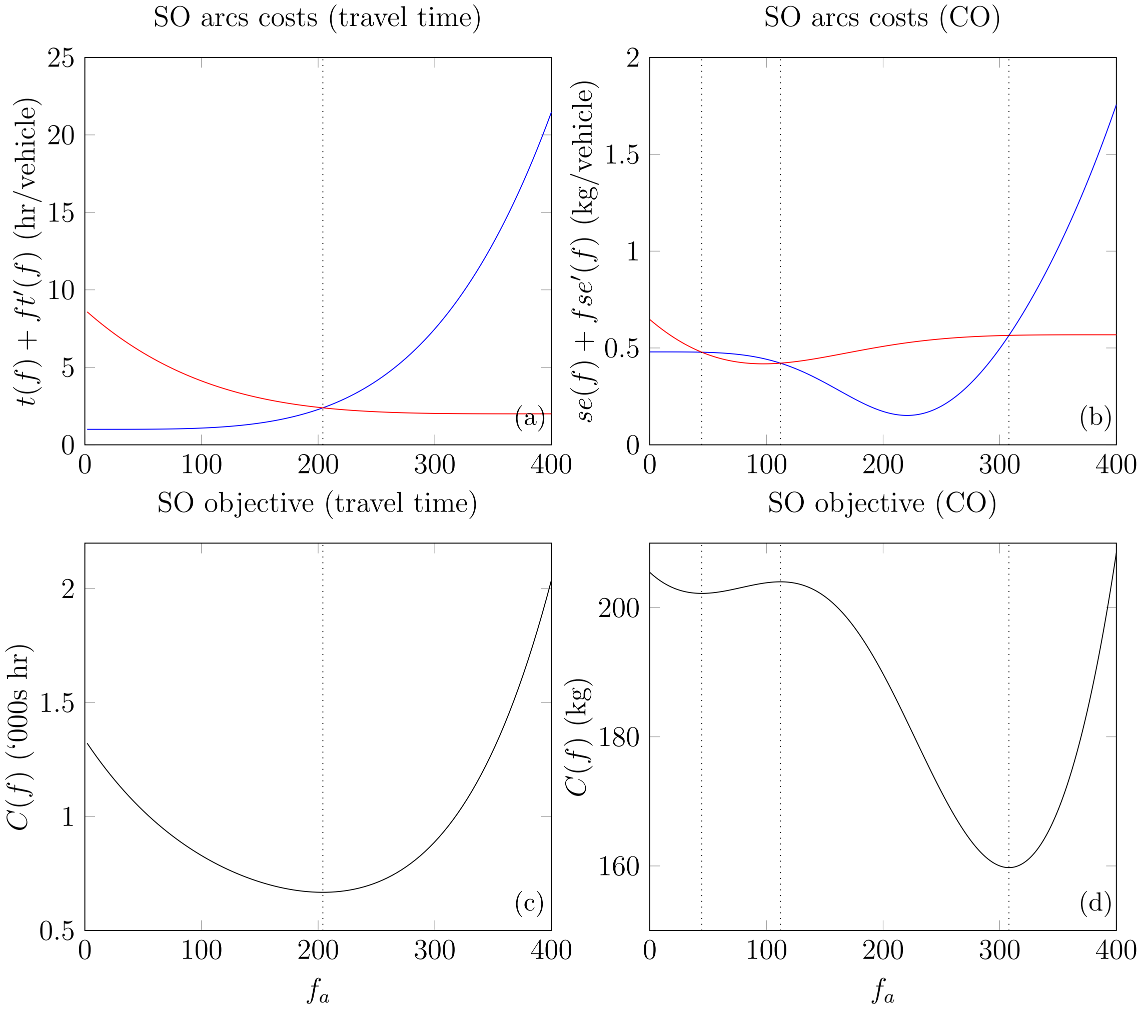

Figure 4: Plots of SO costs and objective cost for travel time and CO emissions for two parallel arcs. All functions are shown with respect to f_a, where $f_b = 400 - f_a$.

Example 3.4. *Consider two parallel arcs connecting an OD-pair with a demand $d = 400$. Figure 4 indicates the travel cost for the users to obtain a SO solution for travel time on the left and CO emissions on the right, where the arc costs are above and the system objective below.*

For strictly increasing functions such as travel time there is one unique flow distribution that results in equal path cost for a given pair of paths, corresponding to the stationary point which is the minimum of the problem, indicated in Figure 4c. For a non-monotonic arc cost function such as emissions, we can see there are three flow distributions that correspond to stationary points in the objective function in Figure 4d, such that choosing a distribution arbitrarily may result in a poor-quality local minimum, or even a local maximum.

3.5 Convergence conditions

With the incorporation of non-monotonic arc cost functions into the TA problem and the corresponding changes to methodology, we need to confirm that convergence to an equilibrium can be guaranteed. We assume that there exists a known lower bound of the total system cost – for emissions we can use the total system cost for the network with emission-minimising speed limits, which can be calculated through regular TA methods. We assume a feasible starting solution such as from the AON algorithm. If we iterate such that the total system cost is strictly decreasing for every equilibration, we will have a strictly decreasing sequence with a known lower bound, and therefore will be able to guarantee convergence.

Iterations of PE involve the adding of a SP (which represents the possibility of a user switching routes to a cheaper path if it exists) as well as equilibrating path costs over active paths for a given OD-pair. For the equilibration of a given OD-pair, we iteratively shift flow between the maximum cost path and the minimum cost path in the active path set until all paths of the OD-pair have equal cost or are unused.

By the continuous differentiability of the cost function, given two paths with initial flows $(f_a^0, f_b^0) = f^0$, there must exist a distribution of the conserved flow that corresponds to either a stationary point or boundary point (shifting either all or zero flow) in the objective cost $C(f^*)$, such that $C(f^*) < C(f^0)$. We cannot have $C(f^*) = C(f^0)$ as this would imply either that the paths already had the same cost (in which case equilibration would not be needed), or that the SP is more expensive than the active path costs (in which case it is not a SP).

We can perform line searches to find the points where two paths have equal cost and evaluate their corresponding objective costs, along with the boundary points. If we shift the amount of flow that corresponds to the minimum objective cost out of all possible points, the total system cost must decrease.

3.6 Enforcing TA solutions with non-monotonic arc costs

In general, SO traffic patterns can be enforced for users minimising their own cost function by tolling roads to encourage the desired driving pattern. The traffic pattern resulting from a TA problem with non-monotonic arc costs can also be enforced, however, depending on the network properties more realistic tolling strategies such as non-negative or revenue-neutral may be unobtainable (Chen and Yang 2012).

4 Methods of solving TA with non-monotonic arc costs

The non-convex TA problem, due to not non-decreasing arc cost functions, can be approached by adjusting the methods used to solve the convex TA problem. However, the descent algorithms will no longer guarantee a globally optimal system cost, and may instead find poor local solutions. As the problem is no longer convex, employing different descent algorithms such as the Frank-Wolfe algorithm may result in different local solutions, due to the different ways of converging to a solution.

The initial solution has a strong effect on the resulting solution found. A poor initial solution may result in the descent methods evaluating a poor path set and ultimately produce an equilibrium with high system cost. The solution methods proposed focus on identifying initial solutions that will produce good solutions to the TA problem.

4.1 Loading network flows as an initial solution

Rather than constructing an initial solution through the AON algorithm, it is possible to load the path flows of a previous TA solution as a starting point, such as the SO travel time (SOTT) solution, which is unique. These previous solutions will generally have well distributed flows, and will ideally be similar to the path flows of our desired solution. Once the path flows of a previous solution are loaded onto the network, the PE stage can be carried out to descend to a local optimum.

4.2 Initialise with a convex combination of path flows

Extending the idea of loading previous solutions in Section 4.1, multiple previous solutions can be used as a basis for 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. We will refer to this as *Convex Combination* (CC) for comparing the methods.

4.3 Weighted combination of time/emissions and monotonicity

Consider a generalised arc cost of $c_a(f_a) = \lambda t_a(f_a) + s_a e_a(v_a(f_a))$, where λ is some weighting of travel time, and recalling that $t_a(f_a)$ is an increasing function while $e_a(v_a(f_a))$ is a non-monotonic function. For a large enough λ, the corresponding TA problem is convex with respect to the system objective function and is notably unique. As λ decreases, the traffic flow solution becomes closer to the desired SO emissions (SOEM) traffic pattern, but will for a certain λ become non-convex, where $\lambda = 0$ gives the emissions cost function. We wish to investigate for what range of λ the weighted cost of emissions and time $c_a(f_a) = \lambda t_a(f_a) + s_a e_a(v_a(f_a))$ is monotonic, giving a convex TA problem.

$$c_a(f_a) = \lambda t_a(f_a) + s_a e_a(v_a(f_a)) \tag{6a}$$

$$= \lambda t_a(f_a) + s_a e_a\left(\frac{s_a}{t_a(f_a)}\right) \tag{6b}$$

$$c'_a(f_a) = t'_a(f_a)\left(\lambda - s_a^2 e'_a\left(\frac{s_a}{t_a(f_a)}\right)\frac{1}{t_a(f_a)^2}\right) \tag{6c}$$

Since $t'_a(f_a) \geq 0$, in order for $c'_a(f_a) \geq 0$, $\forall f_a$, from (6c) we require:

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

If we rearrange the inequality and substitute $v_a(f_a) = {}^{s_a}/_{t_a(f_a)}$ we find:

$$\lambda \geq v_a(f_a)^2 e'_a\left(v_a(f_a)\right).$$

We wish to find the smallest $\lambda = \lambda^*$ such that $c'_a(f_a) \geq 0$, $\forall f_a$. λ^* can be shown to correspond to the value at $f_a = 0$. This is:

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

For a given network we can identify the maximum free-flow speed over all arcs, allowing the computation of the minimum λ to guarantee a monotonic arc cost for all arcs in the network, resulting in a convex TA problem with a unique solution. This is demonstrated in Figure 5, where $\lambda = 0$ at the speed that would give an increasing emissions function.

To obtain a emission-minimising solution form this, the path flows from the resulting convex TA problem can be set as an initial solution for a new problem with

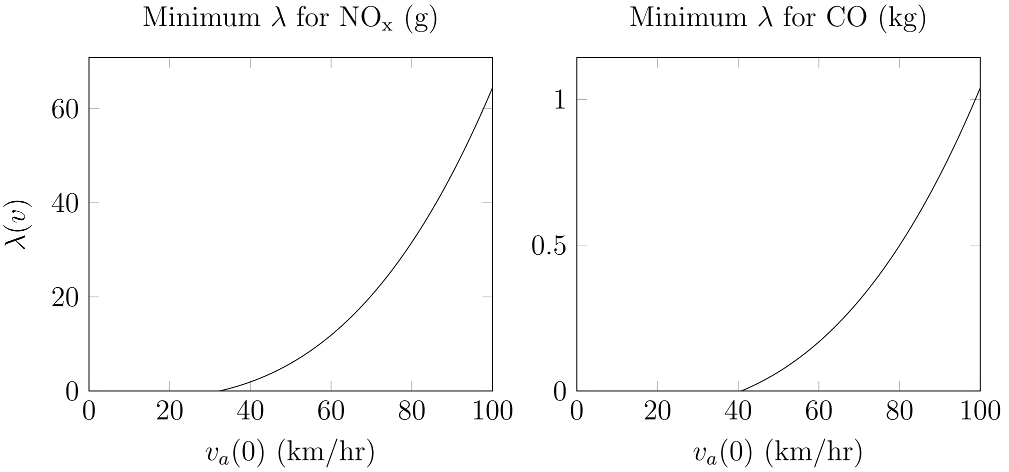

Figure 5: The minimum lambda weighting required to obtain an increasing arc cost function for a given free-flow speed $v_a(0)$.

a smaller λ. The solution to this 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 zero and giving a solution to the SOEM problem. We will refer to this process as λ-*Step* for comparing the methods, where we take n steps from the minimal λ that produces a convex problem to a λ of 0.

4.4 Random initial solutions

One of the issues when producing solutions to the non-convex TA problem is that many promising paths are not considered due to the methods of descent. A way around this is to save all paths considered in a given TA problem and build up a base of 'good' paths from a range of solutions, including intermediate paths found to provide a large number of options. We can 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 uniform random integer n between 1 and the total number of paths from the total path set for that OD-pair are chosen to be active. 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 to 1, and that proportion of the demand for the OD-pair is allocated to the respective active path. This is repeated for all OD-pairs to give an initial solution. We will refer to this as *Random Restarts* for comparing the methods.

An extension is to limit the PE algorithm to only consider paths from the path set, rather than evaluating SPs, which avoids the issue of negative arcs and cycles within the network. This does come at the cost of not being able to consider new paths, which may have produced an improved solution yet to be recorded.

4.5 Case study

We apply the initialisation methods to the road network of Anaheim, which has 38 origin and destination nodes (1406 pairs), 416 nodes, and 914 arcs. We focus on making use of λ-Step (λ-S), Convex Combinations (CC), and Random Restarts (RR). We employ these methods to find solutions to the TA problem with NO_x and

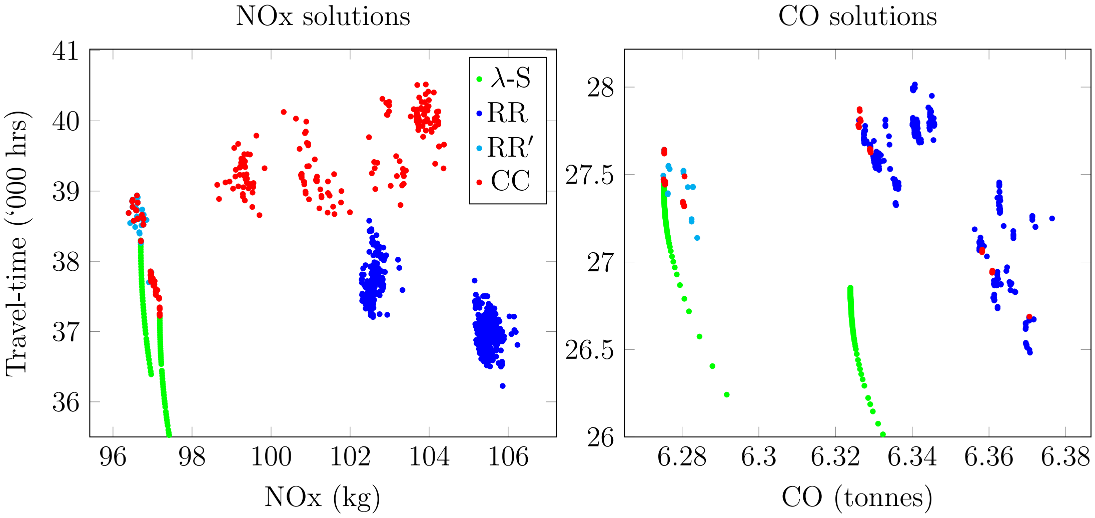

Figure 6: Local solutions found for methods to minimise emissions, NOx and CO respectively, versus the total travel time for the corresponding solution. Here λ-S is the λ-Step method, RR random restarts, and CC convex combination.

CO emission arc cost functions, and present the results in Figure 6.

The λ-S method was run twice for each emission type, once with 200 steps and once with 400 steps, where the smaller number of steps resulted in a lower objective value. The CC method was employed using path sets from the solutions of the SO TT solution, SO emissions with speed limits solution, and the final result from the λ-S with 200 steps, where 820 combinations were evaluated. The RR method was initially run 3000 times with a large path set of saved paths from all previous runs of λ-S and CC problems. A second smaller set of paths was constructed from the solutions of just the λ-S method to produce the solutions 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, the significant difference in solutions is most likely due to the low likelihood of starting in a good position with a large path set.

5 Conclusions

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 were tested in order to find good solutions, where the optimal solution is difficult to find.

In general the method labelled as λ-Step is the most promising, where good quality solutions can be found consistently. Additionally the combination of travel time and emission objectives is of interest, where 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 investigation into modifying the PE algorithm or other similar equilibrium algorithms in order to find good solutions to the TA problem with non-monotonic arc costs, and determining the extent to which driving patterns may be enforced.

References

Benedek, CM, and LR Rilett. 1998. “Equitable traffic assignment with environmental cost functions.” *Journal of transportation engineering* 124 (1): 16–22.

Bureau of Public Roads. 1964. “Traffic Assignment Manual.” *US Department of Commerce.*

Chen, L, and H Yang. 2012. “Managing congestion and emissions in road networks with tolls and rebates.” *Transportation Research Part B: Methodological* 46 (8): 933–948.

Cherkassky, BV, AV Goldberg, and T Radzik. 1996. “Shortest paths algorithms: Theory and experimental evaluation.” *Mathematical programming* 73 (2): 129–174.

Dial, RB. 2006. “A path-based user-equilibrium traffic assignment algorithm that obviates path storage and enumeration.” *Transportation Research Part B: Methodological* 40 (10): 917–936.

Frank, M, and P Wolfe. 1956. “An algorithm for quadratic programming.” *Naval Research Logistics (NRL)* 3 (1-2): 95–110.

Goldberg, AV, and T Radzik. 1993. “A heuristic improvement of the Bellman-Ford algorithm.” *Applied mathematics letters* 6 (3): 3–6.

Patil, GR. 2016. “Emission-based static traffic assignment models.” *Environmental Modeling & Assessment* 21 (5): 629–642.

Patriksson, M. 2015. *The traffic assignment problem: models and methods.* Courier Dover Publications.

Raith, A, C Thielen, and J Tidswell. 2016. “Modelling and optimising fuel consumption in traffic assignment problems.” *Australasian Transport Research Forum (ATRF).*

Roughgarden, T. 2002. “Selfish Routing.” Ph.D. diss., Cornell University.

Sheffi, Y. 1985. *Urban transportation networks.* Volume 6. Prentice-Hall, Englewood Cliffs, NJ.

Song, YY, EJ Yao, T Zuo, and ZF Lang. 2013. “Emissions and Fuel Consumption Modeling for Evaluating Environmental Effectiveness of ITS Strategies.” *Discrete Dynamics in Nature and Society.*

Sugawara, S, and D Niemeier. 2002. “How much can vehicle emissions be reduced?: exploratory analysis of an upper boundary using an emissions-optimized trip assignment.” *Transportation Research Record: Journal of the Transportation Research Board*, no. 1815:29–37.

Yin, Y, and S Lawphongpanich. 2006. “Internalizing emission externality on road networks.” *Transportation Research Part D: Transport and Environment* 11 (4): 292–301.