Turning restriction design for congested urban traffic networks

Les R. Foulds
Daniel C. S. Duarte
Hugo A. D. do Nascimento*
Humberto Longo
Bryon R. Hall †

Instituto de Informática – INF
Instituto de Matemática e Estatística – IME †
Universidade Federal de Goiás – UFG
Goiânia-GO, Brasil

{lesfoulds,danielduarte,hadn,longo}@inf.ufg.br
bryon@mat.ufg.br

Resumo

O problema do projeto de restrição de conversões (*TRDP*) envolve descobrir um conjunto de restrições de conversões que melhore o fluxo em uma rede congestionada de tráfego urbano. Apresentamos um método de aproximação linear sequencial para encontrar uma solução heurística para um modelo não-linear do *TRDP*. O método destina-se a ajustar o conjunto corrente de restrições de conversões de uma dada rede, a fim de minimizar o custo total de viagem do usuário quando a rota escolhida é definida por princípios de equilíbrio do usuário. O método foi comparado com um método *TRDP* não linear existente, utilizando um exemplo de rede padrão da literatura. A experiência computacional preliminar com o método proposto mostra que o mesmo compara-se favoravelmente com o método não-linear existente.

Palavras-Chave: Aproximação sequencial linear, Problema de projeto de redes de tráfego, Projeto de conversões.
Área principal: L&T – Logística e Transportes.

Abstract

The Turning Restriction Design Problem (*TRDP*) involves finding a set of turning restrictions to promote flow in a congested urban traffic network. We present a sequential linear approximation method for identifying a heuristic solution to a nonlinear model of the *TRDP*. The method aims to adjust the current turning restriction regime in a given network in order to minimise total user travel cost when route choice is driven by user equilibrium principles. The method has been compared with an existing nonlinear *TRDP* method using a standard network example from the literature. Preliminary computational experience with the proposed method compares favourably with that of the nonlinear method.

Keywords: Turning restriction design problem, Network design problem, Sequential linear approximation.
Main area: L&T – Logistics and Transportation.

*Hugo do Nascimento is partially sponsored by a Scholarship of Research Productivity from CNPq (309463/2009-2).

1 Introduction

Recently, there has been increasing interest in the system control approach to improving the performance of congested traffic networks through operations research. In general, a common objective is to ease congestion by reducing the total travel cost of all users while accounting for individual route choice behaviour. The related *Network Design Problem* (*NDP*) involves infrastructural intervention improvement strategies comprising physical alterations, such as the construction of new links or the widening or resurfacing of existing ones. The objective is to make an optimal investment decision in order to minimise the total user equilibrium-based travel cost in the network. There are continuous and discrete versions of the *NDP* but as the continuous version can be considered a relaxation of the discrete one, we shall concentrate on the latter.

In contrast to the *NDP*, the *Urban Network Design Problem* (*UNDP*) involves non-intervention improvement strategies comprising non-physical adjustments to characteristics of the network, such as flow direction, turning restrictions and signal settings. These strategies are usually lower in cost and easier to implement than the *NDP* strategies. However, along with the *NDP*, the *UNDP* is a combinatorial optimisation problem that is very expensive to solve exactly and many of the later articles in our literature review concentrate on either approximate solutions to exact UNDP models or exact solutions to approximate models. The approach in the present article falls into the latter case.

One particular low-cost, effective, non-intervention strategy for the *UNDP* is that of deciding which restrictions of maneuvers at intersections should be imposed on the users of the network. That is, certain travel directions (left turn, right turn, drive straight ahead or any other exit option for a non-standard intersection) may be prohibited at each particular intersection. The question is how to select which maneuvers (if any) to restrict in order to enhance a given system performance measure. The present article is concerned with this specialisation of the *UNDP*. We shall use the term turn for maneuver, even though it may involve driving straight ahead or performing another exit option at a non-standard intersection, such as a “Y” intersection, that is not strictly a physical turn. We have constructed a *Sequential Linear Approximation* (*SLA*) method for the problem that starts with a given set of turning restrictions corresponding to the present situation in a given network, and aims to identify which additions or subtractions (allowing all practical possibilities) should be made in order to create the restriction regime that minimises user equilibrium-based total travel cost. The well-known *SLA* methodology (Palacios-Gomez et al. (1982), Bazaraa et al. (1993)) has been used with success in the petroleum industry. More recently, Sherali et al. (2003) and Foulds et al. (2011) have used *SLA* to estimate origin-destination (O-D) travel demand matrices in traffic networks.

2 Literature Review

We shall review some of the literature on the discrete *NDP* and on the *UNDP*. The earlier articles are mainly based on exact optimisation strategies and the later ones are mainly based on meta-heuristics.

Steenbrink (1974) helped to popularise many traffic engineering concepts, including the Braess paradox and some of the techniques that were known for traffic assignment at the time of the publication of his book. He also discussed the *NDP* and appraised some of the then existing branch-and-bound solution techniques for it. Steenbrink himself suggested an iterative decomposition algorithm for the *NDP* in which user equilibrium flows were approximated by system optimal flows. Unfortunately, the algorithm is not guaranteed to converge to the optimal solution. LeBlanc (1975) was one of the first to consider the version of the *NDP* that is confined to deciding which links should be improved. He presented a nonlinear mixed integer programming model of this issue and strategies for a branch and bound algorithm to solve it. However, the algorithm has limited present-day utility as it requires that any link improvement must reduce the total user cost. Magnanti and Wong (1984) presented a unified view of modelling the *NDP* and proposed a unifying framework for describing *NDP* algorithms, of which Lagrangian relaxation and dual ascent have shown to be useful in providing bounds for special cases of the problem.

Drezner and Wesolowsky (2003) introduced some variations of the *NDP* by including a facility location aspect. It is assumed that a network of potential links is given. Each link can be created at a given construction cost. The objective is to minimise the total construction and transportation costs. Two different transportation costs are considered: (i) traffic is generated between any pair of nodes and the transportation cost is the total cost for the users and (ii) demand for service is generated at each node and a facility is to be located at a node to be identified in order to satisfy the demand. The transportation cost in this case is the total cost for a round trip from the facility to each node and back. The authors considered two options with regard to the links between nodes. They can either be two-way only, or mixed, with both two-way and one-way (in either direction) allowed. When these options are combined with the two objective functions, four basic problems are created. These problems were solved by four distinct approaches: descent algorithms, simulated annealing, tabu search and genetic algorithms.

Poorzahedy and Rouhani (2007) attempted to tackle the *NDP* by using various meta-heuristics such as: ant colony systems, genetic algorithms, simulated annealing, and tabu search. Seven hybrid meta-heuristics were devised and compared with an ant colony system approach that was devised earlier by the authors. The meta-heuristics were tested on the Sioux Falls network, a classic test example that is used later in the present article. The hybrid meta-heuristics were more effective in solving the *NDP* compared to using an ant colony system alone, in the sense of finding better solutions sooner. This was also evidenced through analysing an instance of the *NDP* for an actual large city traffic network. Zhang and Gao (2007) studied the situation where the flows in opposite directions on a two-way street are seriously asymmetric; one traffic link of a two-way street is heavily congested but the other is hardly used. In order to reduce transportation congestion and make full use of the existing road resources, the authors proposed a lane reallocating approach and constructed a discrete bi-level programming model for the decision-making. Then, based on a particle swarm optimisation technique, a heuristic for the bi-level model was designed.

Foulds (1981) was one of the first to develop a branch and bound approach for the special case of the discrete *UNDP* where the objective is to identify which links should be declared one-way. Cipriani and Fusco (2004) provided a comprehensive discussion about the *UNDP* specialised to just global traffic signal settings. The authors presented an algorithm that applies the Armijo rule for step size estimation as a part of a projected gradient algorithm and compared it to other solution procedures. Moreover, numerical experiments were performed on a test network in order to investigate the shape of the objective function in order to obtain further information about the mathematical properties of the problem. Issues concerning multiplicity of solutions, algorithm convergence, and sensitivity to demand patterns were also discussed.

Ying et al. (2007) proposed an algorithm for sensitivity analysis due to link interference within the *UNDP*. Based on this sensitivity analysis algorithm, a general algorithm is provided for solving the UNDP. In particular, this algorithm is applied to identify optimal traffic signal settings. Building on the approach of Cantarella et al. (2006) by using stochastic, rather than deterministic assignment, Gallo et al. (2010) proposed a *UNDP* optimisation model and a meta-heuristic technique with the aim of optimising the layout of an urban road network by designing both the directions of existing roads and the traffic signal settings at intersections. A non-linear constrained optimisation model for solving this problem was formulated that adopts a bi-level approach in order to reduce the complexity of solution methods and the computation time. A scatter search algorithm (see Laguna (2002) and Martí et al. (2006)) based on a random descent method was proposed and tested on a practical network. Initial results showed that the proposed approach allows local optimal solutions to be obtained in reasonable computation time. And finally, Russo and Vitetta (2006) have presented a strategy that attempts to lower the number of feasible solutions to any numerical instance of the *UNDP*.

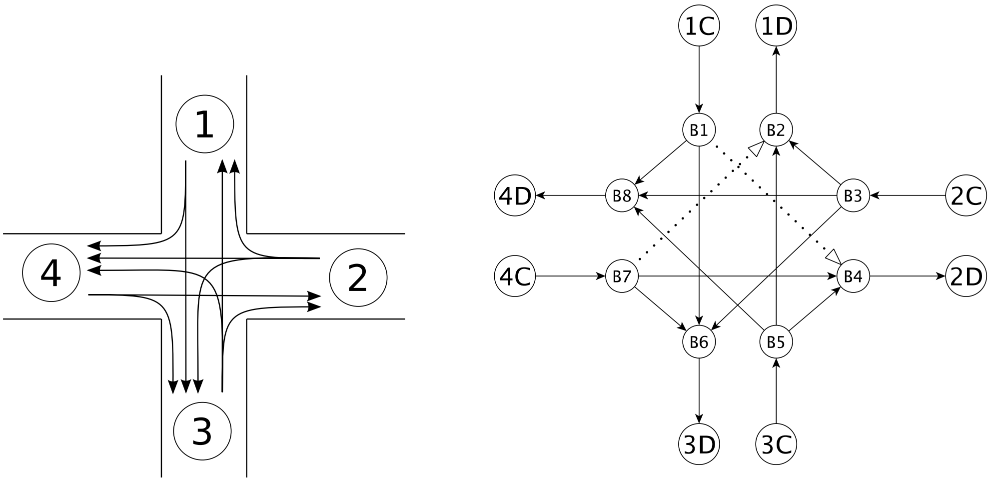
Long et al. (2010) have introduced and defined the *Turning Restriction Design Problem* (*TRDP*) as a special case of the *UNDP*. The *TRDP* involves determining the optimal set of turning restrictions to be imposed in order to minimise the total user equilibrium-based travel cost. Only left turns can be restricted. The authors developed a bi-level model of the *TRDP* in which a shortest path algorithm is used to establish user-equilibrium flows at the lower level and the resulting flows are used within a sensitivity algorithm to solve a relaxed version of the *TRDP* model. Finally, at the upper level, branch and bound strategies are used to solve the *TRDP* model to identify a promising set of turning restrictions at each iteration. The method begins with a network without any turning restrictions and progressively identifies a set of restrictions that are selected from a relatively limited subset of all possibilities, until some termination criterion is met. Only some so-called "crucial" intersections can be considered for restrictions and only left-turn restrictions are allowed. The method is based on a highly nonlinear mixed integer programming model. The advantages of making restrictions on maneuvers at intersections have been discussed by Chen and Luo (2006).

3 Developing a generalised turning restriction design model

We now explain a method for the generalised *TRDP* that is based on a model to be solved using *SLA*. The reader is referred to Foulds (1991) for the necessary graph theory notation and terminology. We shall construct a digraph $G = (N, A)$, due to Potts and Oliver (1972) that models a given traffic network with node set N and arc set A. Each node in N represents either the beginning or the end of a street, that is, an entrance or departure point of an intersection, and and each arc in A represents either a street travel direction or a feasible turning possibility at an intersection. Multiple arcs connecting the same pair of nodes are not permitted. In order to visualise the digraph construction process with a particular example, the reader is referred to Figure 1, where a particular intersection of two two-way streets is shown in Figure 1(a) and the digraph G that results from the application of the process is given in Figure 1(b). Allusions will be made (in italics) to Figure 1 during the following explanation.

We create a distinct arc (that is, with a distinct head node and a distinct tail node) for each travel direction in each street (two distinct arcs for two-way streets and just one arc for a one-way street). Let A' denote the set of arcs representing street travel directions. *The arc set {(1C, B1), (B2, 1D), (2C, B3), (B4, 2D), (3C, B5), (B6, 3D), (4C, B7), (B8, 4D)} is a subset of* A'. Next, let N denote the set of head and tail nodes of the street travel direction arcs. *The node set {1C, B1, B2, 1D, 2C, B3, B4, 2D, 3C, B5, B6, 3D, 4C, B7, B8, 4D} is a subset of* N. Note that each intersection is not represented by a unique node. Then, we create arcs that represent turning possibilities at the intersections. To begin with, it is likely in practical situations that some of the possible turns can never be allowed for reasons of safety or other physical considerations, including the presence of one-way streets.

Arcs representing such turns are never created and play no further part in the discussion. *Note that the turns (1, 2) and (4, 1) are never allowed and, consequently, the corresponding (dotted) arcs (B1, B4) and (B7, B2) are not introduced in the digraph.* Next, we establish which turns are permanently allowed and flow is always possible in them. Let A'' denote the set of such corresponding arcs, that is, arcs for which a turning restriction is never possible. *Suppose that the right turns ((1, 4), (4, 3), (3, 2) and (2, 1)) can never be restricted. In this case, the corresponding arcs in Figure 1(b): (B1, B8), (B7, B6), (B5, B4) and (B3, B2) belong to* A''. Next, we establish which turns are presently allowed but flow in them could be prohibited. Let A_I denote the set of such corresponding arcs, that is, arcs for which turning could be restricted. *Suppose that flow in the drive straight ahead "turns" ((1, 3), (3, 1), (4, 2) and (2, 4)) is presently allowed but could possibly be restricted. In this case, the corresponding arcs in Figure 1(b): (B1, B6), (B5, B2), (B7, B4) and (B3, B8) belong to* A_I. Finally, we establish which turns are presently restricted, that is, flow in them is prohibited, but that flow could possibly be allowed in them. Let A_R denote the set of such corresponding arcs, that is, arcs for which the turning restriction could be lifted. *Suppose that flow in the left turns (2, 3) and (3, 4) is presently restricted but could possibly be allowed. In this case, the corresponding arcs in Figure 1(b): (B3, B6) and (B5, B8) belong to* A_R. The procedure is repeated analogously for all the intersections (irrespective of how many incident streets there are). $G = (N, A)$ is the resulting digraph, where $A = A' \cup A'' \cup A_I \cup A_R$. Note that $A' \cup A''$ comprises arcs whose status cannot be changed, whereas $A_I \cup A_R$ comprises arcs representing the turning possibilities whose status can be changed.



(a) (b)

Figure 1: (a) A street intersection. (b) The underlying digraph G for the street intersection.

Let $\mathcal{OD}$ denote the set of origin-destination node pairs that define the rows and columns of the trip table matrix $T = (T_{ij})$, where T_{ij} is the given travel demand from origin node i to destination node j. For all $(i, j) \in \mathcal{OD}$, let n_{ij} be the number of routes from node i to node j and let p_{ij}^k denote the k^{th} (i, j) route, for $k = 1, \ldots, n_{ij}$. The route p_{ij}^k is represented by a binary vector of elements $(p_{ij}^k)_\alpha$, corresponding to the arcs $\alpha \in A$, where $(p_{ij}^k)_\alpha$ is unity if α belongs to p_{ij}^k and is zero otherwise. Also, let x_{ij}^k be the number of users of p_{ij}^k and let the travel cost of arc α be denoted by $t_\alpha(\cdot)$, $\forall\, \alpha \in A$. The following popular arc travel cost function for any arc $\alpha \in A$ representing a street has been provided by the USA Bureau of

Public Roads (BPR) (1964):

$$t_\alpha(f_\alpha) = t_\alpha^F \cdot \left[1 + \theta \cdot \left(\frac{f_\alpha}{u_\alpha}\right)^\gamma\right], \tag{1}$$

where f_α is the flow, t_α^F is the congestion-free travel cost, u_α is the effective capacity and θ and γ are parameters that must be calibrated according to the actual network being studied. For congested networks the values $\theta = 0.15$ and $\gamma = 4.00$ have been used in the United States, and $\theta = 2.62$ and $\gamma = 5.00$ in Holland and Japan (Steenbrink (1974)). Note that the travel costs of the arcs in $A_I \cup A_R$ are calculated separately with, in this case, $T_\alpha^F = 1$ and θ varying according to whether the maneuver is a left turn, a right turn or involves driving straight ahead.

The decision variables of the proposed model are denoted by y_α, $\forall\, \alpha \in A_I \cup A_R$, where y_α is set to unity if flow is allowed in α, and is set to zero otherwise. From (1) it can be seen that u_α has an inverse effect on t_α. This fact has lead us to adopt an arc capacity-based approach to the *TRDP*, that is based on the premise that changes in traffic signal settings at an intersection, altering the turning regime there, usually affect the capacity of nearby arcs to handle traffic. Towards this end, for all arcs $\alpha \in A$ and $\beta \in A_I \cup A_R$, let s_α^β be a real number that denotes the change in u_α when the turning status of β is changed. That is, u_α becomes $u_\alpha + s_\alpha^\beta$ whenever either (i) $\beta \in A_I$ and flow in β is prohibited (with y_β set to zero) or (ii) $\beta \in A_R$ and flow is allowed in β (with y_β set to unity). Note that s_α^β can be positive or negative, depending upon the relationship between α and β. Furthermore, let $E_\alpha^I = \{\beta \mid \beta \in A_I,\ s_\alpha^\beta \neq 0\}$, let $E_\alpha^R = \{\beta \mid \beta \in A_R,\ s_\alpha^\beta \neq 0\}$, let c_α denote the cost of changing the status of arc α, $\forall\, \alpha \in A_I \cup A_R$, and let B denote the budget for the total cost of all changes in arc status.

It is assumed that the flow pattern is in user equilibrium. That is, according to Wardrop's First Principle (Wardrop (1952)), all routes actually used between any origin-destination pair of nodes should have close to equal travel costs and this cost must not exceed the cost of any unused route between this pair. To represent this, let c_{ij}^k denote the unit cost of p_{ij}^k, $c_{ij}^* = \min\{c_{ij}^k \mid k = 1, \ldots, n_{ij}\}$, $K_{ij} = \{k \mid k \in \{1, \ldots, n_{ij}\}, c_{ij}^k = c_{ij}^*\}$ and $K'_{ij} = \{1, \ldots, n_{ij}\} \backslash K_{ij}$.

An initial linear mixed binary model for identifying an adjusted turning restriction regime that is within budget and tends towards user equilibrium flow principles is given next. This model is new and is one of the main contributions of the present paper.

Model M_μ:

$$\text{Minimise } z = \sum_{(i,j)\in\mathcal{OD}} \sum_{k=1}^{n_{ij}} C_{ij}^k . x_{ij}^k \tag{2}$$

subject to

$$\sum_{k=1}^{n_{ij}} x_{ij}^k = T_{ij}, \quad \forall\,(i,j) \in \mathcal{OD}, \tag{3}$$

$$\sum_{(i,j)\in\mathcal{OD}} \sum_{k=1}^{n_{ij}} (p_{ij}^k)_\alpha . x_{ij}^k \le u_\alpha + \sum_{\beta\in E_\alpha^I} s_\alpha^\beta .(1 - y_\beta) + \sum_{\beta\in E_\alpha^R} s_\alpha^\beta . y_\beta, \quad \forall\, \alpha \in A, \tag{4}$$

$$\sum_{(i,j)\in\mathcal{OD}} \sum_{k=1}^{n_{ij}} (p_{ij}^k)_\alpha . x_{ij}^k \le M . y_\alpha, \quad \forall\, \alpha \in A_I \cup A_R, \tag{5}$$

$$\sum_{\alpha\in A_I} c_\alpha .(1 - y_\alpha) + \sum_{\alpha\in A_R} c_\alpha . y_\alpha \le B, \tag{6}$$

$$x_{ij}^k \geq 0, \quad \forall\,(i,j) \in \mathcal{OD},\; k = 1,\ldots,n_{ij}, \tag{7}$$

$$y_\alpha \in \{0,1\}, \quad \forall\,\alpha \in A_I \cup A_R, \tag{8}$$

where

$$C_{ij}^k = \sum_{\alpha\in A} (p_{ij}^k)_\alpha . t_\alpha(f_\alpha), \quad \forall\, k \in K_{ij}, \tag{9}$$

$$= M_1 . \sum_{\alpha\in A} (p_{ij}^k)_\alpha . t_\alpha(f_\alpha), \quad \forall\, k \in K_{ij}', \tag{10}$$

and where M and M_1 are suitably chosen positive real numbers.

The elements of the model are now explained. The function (2) is based on (9) and (10) and represents the objective of identifying a user equilibrium assignment. Relationship (3) is a conservation of flow constraint for all travel demand. Constraint (4) enables arc capacity to be adjusted as a result of changes to the turning restriction regime. Next, (5) prevents travel in any arc with a turning restriction. And (6) introduces a budgetary constraint that enables control over the total cost of alterations that can be made to the original turning restriction regime. In the example discussed later, the c_α's are set to unity and the budget B is set to various levels, controlling the total number of changes that can be made to the original turning restriction regime. B can be set to a relatively high number, allowing any possible combination of changes to be made. Finally (7) and (8) are the usual non-negativity and binary conditions.

It should be observed that the above model is not designed to solve the nonlinear model of the *TRDP* obtained by substituting the nonlinear functions (1) into (9) and (10) and thus creating a nonlinear objective function in (2). Instead, the model has been constructed to identify at each iteration a collection of O-D routes that can be substituted into (9) and (10) to compute a set of temporarily constant objective function coefficients. These coefficients are substituted into (2) to create a linear mixed binary program. The optimal solution to this model is not necessarily optimal for the nonlinear model and thus, like many *NDP* approaches, the proposed solution technique is an approximating, iterative, heuristic procedure.

4 A sequential linear approximation approach

We propose a bi-level *SLA* scheme for the *TRDP* which is described in Algorithm 1. A traffic assignment process is applied to establish a user-equilibrium flow pattern in the given network with the original turning restrictions in place. The resulting arc flows and travel costs are then used to calculate route costs that are temporarily fixed. This creates a specific numerical *TRDP* problem that is solved to identify a promising set of turning restrictions and the consequent arc capacities. Restrictions on all travel directions: left-turn, right turn and drive straight ahead, can be imposed on each incoming stream at each intersection wherever it is feasible to make a decision on whether ongoing travel in each direction is to be permitted or not.

We now outline the *SLA* scheme where the necessary route costs are modified and held constant at each lower level iteration to provide the objective function coefficients of a mixed binary *ILP* that is used to identify a promising regime of turning restrictions. Updated arc capacities are inputted to establish a user equilibrium flow assignment at each upper level iteration. Consider the following problem, termed Problem $SLA(TR)^r$, for the r^{th} *SLA* lower level iteration that has as input $\forall\,\alpha \in A$, the current flows f_α^r together with arc travel costs $t_\alpha(f_\alpha^r)$ found by substituting f_α^r in (1):

Problem $SLA(TR)^r$:

$$\text{Minimise } z = \sum_{(i,j)\in\mathcal{OD}} \sum_{k=1}^{n_{ij}} (C_{ij}^k)^r . x_{ij}^k + \sum_{\alpha\in A_I\cup A_R} M_\alpha . \omega_\alpha \tag{11}$$

subject to (3), (5), (6), (7), (8) and (12)

$$\sum_{(i,j)\in\mathcal{OD}} \sum_{k=1}^{n_{ij}} (p_{ij}^k)_\alpha . x_{ij}^k \le u_\alpha + \sum_{\beta\in E_\alpha^I} s_\alpha^\beta .(1-y_\beta) + \sum_{\beta\in E_\alpha^R} s_\alpha^\beta . y_\beta + \omega_\alpha,\ \forall\, \alpha \in A, \tag{13}$$

$$\omega_\alpha \ge 0, \quad \forall\, \alpha \in A_I \cup A_R, \tag{14}$$

where

$$(C_{ij}^k)^r = \sum_{\alpha\in A} (p_{ij}^k)_\alpha . t_\alpha(f_\alpha^r), \quad \forall\, k \in (K_{ij})^r, \tag{15}$$

$$= M_1 . \sum_{\alpha\in A} (p_{ij}^k)_\alpha . t_\alpha(f_\alpha^r), \quad \forall\, k \in (K'_{ij})^r, \tag{16}$$

and where the M_α's are a suitably chosen positive real numbers and $(K_{ij})^r$ and $(K'_{ij})^r$ are the versions of K and K' at the r^{th} lower level iteration. Note that a new variable ω_α, $\forall\, \alpha \in A$ has been introduced in order to extend the feasible value of f_α as far above u_α as is desirable to obtain user-equilibrium flows. This extension of u_α is limited by the second term in (11).

Consider the special case where a user equilibrium assignment can be found at the upper level that is a feasible solution to (12)–(14). Next, consider the lower level problem $SLA(TR)^r$ whose objective function coefficients have been calculated by substituting the arc flows of this assignment into (15) and (16). Then an optimal solution to $SLA(TR)^r$ can be found that is a user equilibrium solution.

In general, the procedure solves the linear problem $SLA(TR)^r$ at the r^{th} lower level iteration with revised arc flows f_α, the arc travel costs $t_\alpha(f_\alpha)$ found by using function (1) and the revised route costs $(C_{ij}^k)^r$ calculated using (15) and (16). The column generation technique (Ford and Fulkerson (1958), Dantzig and Wolfe (1960), Gilmore and Gomory (1961, 1963)) can be used to generate O-D routes as they are needed as part of the optimisation process. If a stopping criterion is satisfied, the procedure is terminated and the latest computed solution is outputted. Several types of stopping criteria can be defined for the method. For instance, it can be established whether or not the arc flows of the current solution did not change significantly after a number of iterations, or whether a given number of iterations have been executed, this number being based on the network size. Whichever stopping criteria are collectively adopted, the important idea is to provide sufficient iterations to enable the arc flows, and their associated arc travel costs to converge towards user-equilibrium assignment. If no termination criterion is met, the updated arc capacities that are part of the solution to $SLA(TR)^r$ are inputted into the traffic assignment problem to be solved at the next upper level iteration. The outputted arc flows, costs and route costs $(C_{ij}^k)^{r+1}$ are used to define problem $SLA(TR)^{r+1}$ which is then formulated and solved.

Instead of simply replacing the arc flow values with the new values from the upper level, we have found it more effective at the lower level to update them with a moving average of the previous values. This approach, also applied by Sherali et al. (2003) and by Foulds et al. (2011), promotes a better transition of the arc flows from one iteration to another, avoiding drastic jumps back in forth in flow values.

Algorithm 1: $SLA(TR)$

input : A network $G = (N, A)$.
output: A set of turning restrictions.

```
// Pre-Processing.
1  Set all arc capacities to their original values.
2  r ← 1.
// Upper level.
3  Use a traffic assignment algorithm to identify a user equilibrium flow pattern for G. This
   calculate the arc flow f_α, ∀ α ∈ A.
4  Use (1) to calculate the travel cost t_α(f_α), α ∈ A.
5  Use (15) and (16) to calculate the cost of each relevant O-D route.
// Lower level.
6  Solve SLA(TR)^r.
7  Establish if any stopping criterion has been met.
8  If so, terminate the process and output the current set of rescindable turning restrictions as
   the solution.
9  If not, update the sets A_I and A_R and the arc capacities u_α, ∀ α ∈ A, according to (5) and
   (13).
10 r ← r + 1.
11 Go to step 3.
```

5 Computational experience with algorithm $SLA(TR)$

Here we discuss some preliminary computational experience with algorithm $SLA(TR)$ on a standard network from the literature in comparison with the nonlinear method of Long et al. (2010). The $SLA(TR)$ was implemented using JAVA and CPLEX v. 12.2 API for the linear model. All numerical tests of the algorithm were executed on a personal computer with a Pentium M730 processor with 1.73 GHz clock speed, $2GB$ of RAM and 150 GB of HD, running Linux. The execution time for $SLA(TR)$, for all examples to be described, ranged between 32.3 and 129.7 seconds. Two termination criteria were adopted simultaneously, the number of iterations and the degree of change from one iteration to the next. For some examples, the number of iterations caused termination and in the others it was the degree of change. The numerical example is known as the Sioux Falls network and was first introduced by LeBlanc (1975). It has 24 nodes, 76 arcs, 528 O-D pairs and 178 possible turns at intersections (any of which can be restricted). The numerical results about to be reported are summarized in Table 1, with the tests denoted by labelled rows that are referred to in parentheses in the following discussion.

We first establish a user equilibrium assignment for the original network with no turning restrictions present, meaning that users can make any of the 178 turns. The total user equilibrium cost of this assignment is z = 15,585,566.05 according to our calibrations and we use this cost as the basis for comparison for the next three results. Long et al. (2010) selected 22 "crucial" possible turning restrictions in the network, any combination of which could be imposed. The best result produced by the methods of Long et al. (2010) resulted in a turning restriction regime with 15 of the possible 22 restrictions imposed. We configured our method to reproduce this regime using our calibrations and this resulted in a 3.195% reduction in total user cost (T_1). Algorithm $SLA(TR)$ was run on the original network with the same set of 22 possible restrictions as in T_1. This resulted in a regime with 12 restrictions imposed, with a 5.077% reduction in total user cost (T_2). There are nine restrictions in common between the regimes of T_1 and of T_2. Algorithm $SLA(TR)$ was also run on the original network with a budget of $B = 22$ with each restriction introduction costing unity. That is, any combination of up to 22 restrictions can be imposed out of the possible 178. This resulted in a regime with the full limit of 22 restrictions imposed, with a 5.810% reduction in total user cost (T_3). Interestingly, there are only two restrictions in common with the regimes of T_1 and T_2. When B is increased to 178, allowing any combination of restrictions to be imposed, the best regime identified by algorithm $SLA(TR)$ has 108 restrictions, with an 8.002% reduction in total user cost (T_4).

As has been mentioned, algorithm $SLA(TR)$ is designed to deal with the practical situation in which a regime of restrictions is already imposed and the method is to be used to adjust the regime by imposing new restrictions and removing existing ones in order to attempt to minimise total user cost, taking into account user equilibrium-based route choice. In order to test this aspect of algorithm $SLA(TR)$ we applied it to the modified Sioux Falls network (T_1) created by the method of Long et al. (2010) with the 15 restrictions imposed, but with the original capacities for all arcs. This network has a user equilibrium cost of $15,507,321.89$ according to our calibrations and we use this cost as the basis for comparison for the remaining tests. When alterations can be made only to the original 22 turns, the best regime identified by algorithm $SLA(TR)$ has 17 restrictions imposed, 12 of the original ones and 5 new ones, with a 1.659% reduction in total user cost (T_5). When algorithm $SLA(TR)$ was run on this network with a budget $B = 22$ and the status of any turn was allowed to change, the best regime identified has 31 restrictions imposed, 12 of the 15 original ones and 19 new ones, with a 3.126% reduction in total user cost (T_6). When algorithm $SLA(TR)$ was run on the network with $B = 178$, this resulted in a regime with 99 restrictions with the same 12 of the 15 original ones, and 87 new ones imposed, with a 4.598% reduction in total user cost (T_7). It is evident from the just mentioned tests that $SLA(TR)$ has the capability to solve quickly and easily a variety of interesting numerical instances derived from the Sioux Falls network.

Test	Method	z	% Reduction	No. of Restrictions
T_1	Long et al.	15,087,557.74	3.195	15
T_2	$SLA(TR)$	14,794,305.21	5.077	12
T_3	$SLA(TR)$	14,679,976.38	5.810	22
T_4	$SLA(TR)$	14,338,358.57	8.002	108
T_5	$SLA(TR)$	15,250,111.58	1.659	17
T_6	$SLA(TR)$	15,022,539.58	3.126	31
T_7	$SLA(TR)$	14,794,283.44	4.598	99

Table 1: The results of applying algorithm $SLA(TR)$ to the Sioux Falls network.

6 Conclusion and summary

We have presented a successive linear approximation method, termed $SLA(TR)$, for identifying a heuristic solution to a nonlinear model of the *TRDP*. The method aims to adjust the current turning restriction regime in a given network in order to minimise the total cost when user route choice is driven by user equilibrium principles. The method has been compared with an existing nonlinear *TRDP* method using a standard network example from the literature. Preliminary computational experience with $SLA(TR)$ is favourable. The authors are in the process of refining the $SLA(TR)$ model, conducting further numerical experiments on large-scale Brazilian city networks and investigating the model's convergence properties.

References

Bazaraa, M. S., Sherali, H. D., and Shetty, C. M. (1993). *Nonlinear Programming, Theory and Applications*. John Wiley & Sons, New York, 2$^{\text{nd}}$ edition.

Cantarella, G. E., Pavone, C., and Vitetta, A. (2006). Heuristics for urban road network design: lane layout and signal settings. *European Journal of Operational Research*, 175(3):1682–1695.

Chen, K. and Luo, Z. Z. (2006). Analysis of feature of left turn traffic at level crossing and countermeasures. *Technology of Highway and Transport*, 22(2):114–118.

Cipriani, E. and Fusco, G. (2004). Combined signal setting design and traffic assignment problem. *European Journal of Operational Research*, 155(3):569–583.

Dantzig, G. B. and Wolfe, P. (1960). Decomposition principle for linear programs. *Operations Research*, 8(1):101–111.

Drezner, Z. and Wesolowsky, G. O. (2003). Network design: selection and design of links and facility location. *Transportation Research*, 37A(3):241–256.

Ford, L. R. and Fulkerson, D. R. (1958). A Suggested Computation for Multi-commodity Network Flows. *Management Science*, 5(1):97–101.

Foulds, L. R. (1981). A multicommodity flow network design problem. *Transportation Research*, 15B(4):273–284.

Foulds, L. R. (1991). *Graph theory applications*. Springer-Verlag, New York.

Foulds, L. R., do Nascimento, H. A. D., Calixto, I. C. A., Hall, B., and Longo, H. (2011). A fuzzy set approach to estimating OD matrices in congested brazilian traffic networks. In *Proceedings of the Annual Conference of SOBRAPO, The Brazilian Society of Operations Research, XLIII SBPO*, pages 1386–1397, Ubatuba-SP, Brazil.

Gallo, M., D'Acierno, L., and Montella, B. (2010). A meta-heuristic approach for solving the urban network design problem. *European Journal of Operational Research*, 201(1):144–157.

Gilmore, P. C. and Gomory, R. E. (1961). A Linear Programming Approach to the Cutting Stock Problem. *Operations Research*, 9(6):849–859.

Gilmore, P. C. and Gomory, R. E. (1963). A Linear Programming Approach to the Cutting Stock Problem, Part II. *Operations Research*, 11(6):863–888.

Laguna, M. (2002). Scatter Search. In Pardalos, P. M. and Resende, M. G. C., editors, *Handbook of Applied Optimization*, pages 183–193. Oxford University Press.

LeBlanc, L. J. (1975). An algorithm for the discrete network design problem. *Transportation Science*, 9(3):183–199.

Long, J., Gao, Z., Zhang, H., and Szeto, W. Y. (2010). A turning restriction design problem in urban road networks. *European Journal of Operational Research*, 206(3):569–578.

Magnanti, T. L. and Wong, R. T. (1984). Network design and transportation planning: models and algorithm. *Transportation Science*, 18(1):1–55.

Martí, R., Laguna, M., and Glover, F. (2006). Principles of scatter search. *European Journal of Operational Research*, 169(2):359–372.

Palacios-Gomez, F., Lasdon, L., and Engquist, M. (1982). Nonlinear optimization by successive linear programming. *Management Science*, 28:1106–1120.

Poorzahedy, H. and Rouhani, O. M. (2007). Hybrid meta-heuristic algorithms for solving network design problem. *European Journal of Operational Research*, 182(2):578–596.

Potts, R. B. and Oliver, R. M. (1972). *Flows in Transportation Networks*. Academic Press, New York.

Russo, F. and Vitetta, A. (2006). A topological method to choose optimal solutions after solving the multicriteria urban network design problem. *Transportation*, 33(4):347–370.

Sherali, H. D., Narayanan, A., and R., S. (2003). Estimation of origin-destination trip-tables based on a partial set of traffic link volumes. *Transportation Research*, 37B(9):815–836.

Steenbrink, P. A. (1974). *Optimization of Transport Networks.* Wiley and Sons, New York.

USA Bureau of Public Roads (BPR) (1964). Traffic assignment manual. Technical report, US Department of Commerce, Urban Planning Division, Washington DC, USA.

Wardrop, J. (1952). Some theoretical aspects of road traffic research. *Proceedings of the Institution of Civil Engineers, Part II*, 1(36):352–362.

Ying, J., Lu, H., and Shi, J. (2007). An algorithm for local continuous optimization of traffic signals. *European Journal of Operational Research*, 181(3):1189–1197.

Zhang, H. and Gao, Z. (2007). Two-way road network design problem with variable lanes. *Journal of Systems Science and Systems Engineering*, 16(1):50–61.