On traffic domination in communication networks

Walid Ben-Ameur[1], Pablo Pavon-Marino[2], and Michał Pióro[3,4]

[1] TELECOM SudParis, 9, rue Charles Fourier, 91011 Évry, France
[2] Technical University of Cartagena, Pza. Hospital 1, 30202 Cartagena, Spain
[3] Warsaw University of Technology, Nowowiejska 15/19, 00-665 Warszawa, Poland
[4] Lund University, Box 118, 221 00 Lund, Sweden
walid.benameur@int-evry.fr,pablo.pavon@upct.es,mpp@tele.pw.edu.pl

Abstract. Input data for communication network design/optimization problems involving multi-hour or uncertain traffic can consist of a large set of traffic matrices. These matrices are explicitly considered in problem formulations for link dimensioning. However, many of these matrices are usually dominated by others so only a relatively small subset of matrices would be sufficient to obtain proper link capacity reservations, supporting all original traffic matrices. Thus, elimination of the dominated matrices leads to substantially smaller optimization problems, making them treatable by contemporary solvers. In the paper we discuss the issues behind detecting domination of one traffic matrix over another. We consider two basic cases of domination: (i) total domination when the same traffic routing must be used for both matrices, and (ii) ordinary domination when traffic dependent routing can be used. The paper is based on our original results and generalizes the domination results known for fully connected networks.

Key words: network optimization, traffic matrices domination, multi-hour optimization, uncertain traffic, graph theory

1 Introduction

Input data for optimization problems related to communication network design and planning can consist of a large set of traffic matrices. This is the case for example when multi-hour traffic or uncertain traffic is considered [1, 9, 11]. By using a set of matrices rather than a single traffic matrix, it is possible to take into account the non-coincidence of the peak load hours in different parts of the network, or to use a large set of measured matrices when the traffic is hardly predictable. In consequence, a set of traffic matrices is explicitly used in problem formulations in order to dimension the network links. However, many of the input matrices are usually dominated by others, and in effect only a relatively small subset of them can be sufficient to obtain proper link capacity reservations, supporting all original traffic matrices. Thus, elimination of the dominated matrices leads to substantially smaller optimization problems, making them treatable by the optimization solvers. Moreover, the domination relation between traffic matrices is of interest not only for identifying and then removing the dominated traffic matrices. In [7] it was shown that the domination can also be exploited to devise more efficient planning algorithms under multi-hour or uncertain traffic demand, introducing a small "upper bound" set of artificial traffic matrices dominating the entire set of the original traffic matrices and yet yielding a very good approximation of the true optimal solution.

In the paper we discuss the issues behind detecting domination of one traffic matrix over another. We consider two basic cases: total domination (when the same traffic routing must be used for both matrices – in this case we in fact consider domination of a set of traffic matrices over one additional traffic matrix), and ordinary domination (when traffic-dependent routing can be used). The paper is based on our original results presented in [2, 8].

Given a network graph $G(V, E)$, and two traffic matrices $\hat{h}$ and h, $\hat{h}$ *totally dominates* h ($\hat{h} \succeq h$ in short) if for each link capacity reservation $u : E \to \mathcal{R}_+$ and for each flow pattern $f : P \to \mathcal{R}_+$ (where P is the set of routing paths) such that (u, f) supports matrix $\hat{h}$, the solution (u, f) does also support matrix h.

The definition of ordinary traffic domination is slightly different. We say that $\hat{h}$ *ordinarily dominates* h ($\hat{h} \models h$) if for each link capacity reservation $u : E \to \mathcal{R}_+$ for which there exists a flow pattern $\hat{f} : P \to \mathcal{R}_+$ such that $(u, \hat{f})$ supports matrix $\hat{h}$, there exists a (in general different) flow pattern $f : P \to \mathcal{R}_+$ such that (u, f) supports matrix h. Obviously, total domination implies ordinary domination, i.e., $\hat{h} \succeq h$ implies $\hat{h} \models h$.

The known result (due to Gianpaolo Oriolo [6]) on total domination is as follows: in a complete graph $\hat{h} \succeq h$ if, and only if $\hat{h} \geq h$ component-wise. In [8] we have given a complete characterization of total domination which we describe in Section 2 below. In fact, total domination is discussed in a more general setting that examines when a set of traffic matrices dominates one additional traffic matrix.

For ordinary domination the known result (also due to [6]) states that in a complete graph $\hat{h} \models h$ if, and only if, matrix h can be routed in the network with link capacities equal to the elements of $\hat{h}$. This result, however, cannot be extended to arbitrary graphs simply because in an incomplete graph there may no direct links between the end nodes of some demands, making the Oriolo result not applicable. Consequently, in Section 3, we present a different kind of general necessary and sufficient condition for ordinary domination in an arbitrary graph, using findings of [2].

2 Total domination

2.1 Definition

Consider an undirected network graph $G = G(V, E)$ with the set of nodes (vertices) V and the set of undirected links (edges) E. A path p between nodes s and node t in graph G is given by a sequence of nodes $v_1 = s, v_2, v_3 \ldots, v_{n-1}, v_n =$

$t), v_i \in V, i = 1, \ldots, n$, such that $v_i v_{i+1} \in E, i = 1, 2, \ldots, n-1$. Path p is called elementary if all the nodes are different, and in effect an elementary path can be treated as a subset of links: $p \subseteq E$. Link capacity reservation $u : E \rightarrow \mathcal{R}_+$ will be identified with vector $(u_e, e \in E)$ where $u_e = u(e)$.

Let D denote the set of (undirected) traffic demands in the considered network. Each demand $d \in D$ is characterized by its end nodes $a(d)$ and $b(d)$, $(a(d) \neq b(d))$, and its traffic volume h_d. The vector $h = (h_d, d \in D)$ is referred to as traffic vector. (We prefer to use the notion of the traffic vector instead of the traffic matrix used in the introduction.) For each demand $d \in D$ we specify a set of its candidate paths P_d – a subset of all elementary paths between nodes $a(d)$ and $b(d)$ that are selected for carrying traffic of demand d. We assume that there is at most one demand between any two nodes so that the path sets $P_d, d \in D$ are mutually disjoint (in the sequel a demand d between nodes s and t will sometimes be denoted, somewhat informally, by $d = st \in D$). Finally, we put $P = \bigcup_{d \in D} P_d$ – the set of all admissible paths. Having the predefined path set P, for each link e and demand d we can define $Q_{ed} \subseteq P_d$ – the set of all candidate paths for demand d that contain link e: $Q_{ed} = \{p \in P_d : \ p \ni e\}$. The length of path $p \in P$ with respect to a given vector of link metrics (weights) $\pi = (\pi_e, \ e \in E)$ will be denoted by $|p|_\pi$.

Given graph $G(V, E)$, set of demands D, and set P of candidate paths, we define a static (fixed) flow allocation pattern as a vector $f = (f_p, \ p \in P)$, where for each $d \in D$ and $p \in P_d$, the entity f_p represents a fraction $(0 \leq f_p \leq 1)$ of traffic volume of demand $d \in D$ assigned to path p.

Consider a finite set of traffic vectors $H = \{h^t : t \in T\}$ to be supported by the network, and one additional traffic vector $h = (h_d, d \in D)$ outside set H. Set H can represent $|T|$ different traffic hours, or a set of observed realizations of a random traffic vector. For every $h^t, t \in T$, its traffic volume for demand $d \in D$ is denoted by h_d^t so that $h^t = (h_d^t, d \in D)$.

We say that a capacity reservation vector u and a flow allocation pattern f support H with respect to the given path set P if (u, f) satisfies the following linear constraints:

$$\sum_{p \in P_d} f_p = 1 \qquad d \in D \tag{1a}$$

$$\sum_{d \in D} \sum_{p \in Q_{ed}} h_d^t f_p \leq u_e \qquad e \in E, \ t \in T \tag{1b}$$

$$f_p \geq 0 \qquad p \in P. \tag{1c}$$

Constraint (1a) assures that each demand is satisfied, while constraint (1b) – that each link must support the load induced by any vector from H. Constraint (1c) ensures non-negativity of flows (and hence of capacity reservations, provided each link is in at least one path).

In the sequel we shall assume that the network graph $G(V, E)$ and the set of demands D are fixed. Then, for a given path set P and a set of traffic vectors H, the feasible set (1) will be abbreviated by $\mathcal{P}_P(H)$. Note that $\mathcal{P}_P(H)$ is a polyhedron in $\mathcal{R}_+^{|E|+|P|}$.

Now, we are ready to introduce a formal definition of total domination, extending the one from [6] to a set of traffic vectors H and a traffic vector h defined for a demand set D.

Definition 1. *Let P be a given set of candidate paths for D in a network with arbitrary graph G. We say that H totally dominates h with respect to P ($H \succeq_P h$ in short) if, and only if, each feasible solution in $\mathcal{P}_P(H)$ is also a feasible solution in $\mathcal{P}_P(h)$, i.e., $\mathcal{P}_P(H) \subseteq \mathcal{P}_P(h)$.*

(In the sequel we will skip subscript P in $\mathcal{P}_P$ and in $\succeq_P$ when the path set P is fixed.)

2.2 A sufficient condition and a necessary condition for total domination

Consider a network with an arbitrary graph G, arbitrary demand set D and the corresponding traffic vectors H and h.

Proposition 1. $\exists\, \hat{h} \in conv(H),\ \hat{h} \geq h \Rightarrow H \succeq_P h.$

Proof. Consider a set of scalar coefficients $\alpha_t \geq 0, t \in \mathcal{T}, \sum_{t\in\mathcal{T}} \alpha_t = 1$ defining the convex combination $\hat{h} = \sum_{t\in\mathcal{T}} \alpha_t h^t \in \text{conv}(H)$. Let $(u,f) \in \mathcal{P}_P(H)$. Then (u,f) is also a feasible solution of (1) for the convex combination $\hat{h}$ ($(u,f) \in \mathcal{P}_P(\hat{h})$), because the convex combination of constraints (1b) for $\mathcal{P}_P(H)$ with the above defined coefficients $\alpha_t, t \in T$ yields constraint (1b) defining polyhedron $\mathcal{P}_P(\hat{h})$.

Now suppose that $\hat{h} \geq h$. This implies that for the given $(u,f) \in \mathcal{P}_P(\hat{h})$, constraint (1a) is satisfied also for $\mathcal{P}_P(h)$ which means that (u,f) supports h. ■

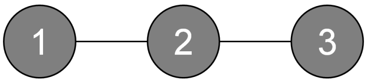

Fig. 1. A two-link network.

The sufficient condition for total domination formulated above is in general not necessary, as illustrated in Fig.1. The figure shows a 3-node, 2-link graph with three demands $13, 12, 23$, and with the set of admissible paths $P = \{\{123\}, \{12\}, \{23\}\}$. Assume $H = \{\hat{h}\}, \hat{h} = (1,0,0)$ ($\hat{h}_{13} = 1, \hat{h}_{12} = \hat{h}_{23} = 0$), and $h = (0,1,1)$ ($h_{13} = 0, h_{12} = h_{23} = 1$). Certainly, since the network is a tree, only one flow allocation pattern exists. It is obvious that every capacity vector u supporting traffic vector $\hat{h}$ (i.e., $u \geq (1,1)$), also supports h (and, in fact, vice versa). Then, $\hat{h}$ totally dominates h with respect to P (and vice versa). Still, it is not true neither that $\hat{h} \geq h$ nor that $h \geq \hat{h}$.

Below we give a (technical) property of the path set P under which the condition $\exists\, \hat{h} \in \text{conv}(H),\ \hat{h} \geq h$ holds is also necessary for the total domination.

Proposition 2. *Let G, H, h be as in Proposition 1 and let $P = \bigcup_{d \in D} P_d$. Suppose that there exists a link $\hat{e} \in E$ (the so called* enabling link*) with the following property:*

$$\forall\, d \in D\ (\exists\, p \in P_d,\ \hat{e} \in p) \wedge (\exists\, p \in P_d,\ \hat{e} \notin p), \tag{2}$$

Then, if H totally dominates h wrt P, then there exists a $\hat{h} \in conv(H)$ such that $\hat{h} \geq h$:

$$H \succeq_P h \Rightarrow \exists\, \hat{h} \in conv(H),\ \hat{h} \geq h. \tag{3}$$

Proposition 2 is proved in [8] using the means of the LP dual theory. In the next two subsections we will use property (2) to find a general characterization of total domination.

2.3 Two-connected networks

Graph $G(V, E)$ is called *2-connected* if it has at least three nodes and does not contain any *cut vertex*, i.e., any vertex $v \in V$ such that $G \setminus v$ has more connected components than G. (Below, we follow definitions and results given in [3].)

Proposition 3. *Suppose that a network graph is 2-connected (or composed of one link) and that for each demand $st \in D$ its set of candidate paths is composed of all elementary st-paths. Then*

$$H \succeq h\ \Leftrightarrow \exists\, \hat{h} \in conv(H),\ \hat{h} \geq h. \tag{4}$$

Proof. 2-connected graphs enjoy the two following properties: (i) G is 2-connected if, and only if, any two nodes $v, w \in V$ are connected by two node disjoint paths [3], and (ii) G is 2-connected if, and only if, for any two nodes $v, w \in V$ and link $e \in E$ there exists an elementary path between v and w containing link e [3].

These properties imply that in a 2-connected graph each link $\hat{e} \in E$ is enabling for any demand set D (provided P_d contains all elementary paths between its end nodes). Consider an arbitrary demand $st \in D$. There must be a path $p \in P_d$ such that $\hat{e} \notin p$. Otherwise, all elementary st-paths would contain $\hat{e}$, contradicting property (i). Also, by property (ii), there must be a path $p' \in P_d$ such that $\hat{e} \in p'$. Trivially, property (4) is valid also for graphs composed of just one link (and its end nodes). ■

2.4 General characterization of total domination

In this section we will present the main result of Section 2 – a general necessary and sufficient condition for total domination in a network with an arbitrary connected graph and path sets P_d containing all elementary paths for each demand $d \in D$. For simplifying the considerations, we assume (without loosing generality – we can assign the zero traffic volume to non-existing demands) that the network contains demands corresponding to all node pairs, i.e., $D = V^{|2|}$ ($A^{|2|}$ denotes the family of all two-element subsets of set A).

A connected graph $G(V, E)$ containing cut vertices is called *separable*. The maximal induced subgraphs of G that are not separable are called *blocks*. A block is either 2-connected or is formed by just one link. The blocks of a graph are unique. Any two blocks intersect in at most one node and this node must be a cut vertex. Two nodes s and t are in the same block if, and only if, they are connected by a path not traversing a cut vertex. Finally, any cycle must be contained in a block (see [3]).

Consider a network with a connected separable graph $G = (V, E)$ and a fixed traffic vector h. Suppose that graph G is composed of B blocks $G^b = (V^b, E^b), b \in \mathcal{B}$, where $\mathcal{B} = \{1, 2, ..., B\}$, and C is the set of cut vertices of G. Clearly, each elementary path between a pair of nodes st traverses the same sequence of blocks (the set of indices of these blocks will be denoted by $B(st)$, and the same sequence of cut vertices (denoted by $C(st)$). (Otherwise, there would be a cycle not contained in a single block.)

We treat the blocks as separate networks with their own sets of demands $D^b = (V^b)^{|2|}$ and the corresponding sets of candidate paths $P^b_d, d \in D^b$ composed of all elementary paths in G^b ($P^b = \bigcup_{d \in D^b} P^b_d$). The traffic vector h^b for each block $b \in \mathcal{B}$ is induced by the traffic vector h. The volume h^b_{st} of each demand $st \in D^b$ is either left unchanged (when both s and t are not cut vertices) or adjusted to account for the demands that have only one end node in the considered block b (i.e., when $s \in V^b, t \notin V^b$ and $b \in B(st)$) or transit the block (i.e., when $s \notin V^b, t \notin V^b$ and $b \in B(st)$). More precisely:

$$h^b_{st} = h_{st} \qquad s, t \in V^b \setminus C, s \neq t \tag{5a}$$

$$h^b_{st} = h_{st} + \sum \{h_{sw} : \ w \notin V^b, t \in C(sw)\} \qquad s \in V^b \setminus C, t \in V^b \cap C \tag{5b}$$

$$h^b_{st} = h_{st} + \sum \{h_{vw} : \ v, w \notin V^b, s, t \in C(vw)\} \quad s, t \in V^b \cap C, s \neq t. \tag{5c}$$

Proposition 4. *Consider a network with a connected graph $G = (V, E)$ split into blocks $G^b, b \in \mathcal{B}$. Let H be a set of traffic matrices, and let h be an additional traffic vector. For each $b \in \mathcal{B}$, let H^b, h^b denote the set of traffic vectors defined for H and h in the way described by* (5)*. Then,*

$$H \succeq h \Leftrightarrow \forall\, b \in \mathcal{B}\ \exists\, \hat{h}^b \in conv(H^b),\ \hat{h}^b \geq h^b. \tag{6}$$

Although Proposition 4 is quite intuitive, its detailed proof is a bit lengthy and is omitted in this survey paper. The proof is given in [8].

In the example from Fig.1, the graph is split into two blocks with $V^1 = \{1, 2\}$ and $V^2 = \{2, 3\}$, and the adjusted traffic vectors are: $\hat{h}^1 = (1), h^1 = (1), \hat{h}^2 = (1), h^2 = (1)$. Hence, $\hat{h} \succeq h$ because $\hat{h}^1 \succeq h^1$ on G^1 and $\hat{h}^2 \succeq h^2$ on G^2 (and vice versa, $h \succeq \hat{h}$ because $h^1 \succeq \hat{h}^1$ on G^1 and $h^2 \succeq \hat{h}^2$ on G^2).

3 Ordinary domination

3.1 Definition

Given graph $G(V, E)$, set of demands D, and set P of candidate elementary paths, we define a flow allocation pattern as a vector $f = (f_p,\ p \in P)$, where for each $d \in D$ and $p \in P_d$, the entity f_p represents a flow of demand $d \in D$ assigned to its path p. Note that this definition of a flow allocation pattern is a bit different than the analogous definition in Subsection 2.1 which has involved fractional flows while the current one involves absolute flows.

Consider a traffic vector h. We say that a capacity reservation vector u supports h if there exists a flow allocation pattern f such that (u, f) satisfies the following linear constraints:

$$\sum_{p \in P_d} f_p = h_d \qquad d \in D \tag{7a}$$

$$\sum_{d \in D} \sum_{p \in Q_{ed}} f_p \le u_e \qquad e \in E \tag{7b}$$

$$f_p \ge 0 \qquad p \in P. \tag{7c}$$

Constraint (7a) assures that each demand is satisfied, while constraint (7b) – that each link must support the load induced by the traffic vector h and flow f. Constraint (7c) ensures non-negativity of flows (and hence of capacity reservations, provided each link is in at least one path).

In the sequel we shall assume that the network graph $G(V, E)$, the set of demands D, and the set of candidate paths P are fixed. A capacity reservation vector $u = (u_e, e \in \mathcal{E})$ is said to support a demand vector h if, and only if, there exists a flow pattern f such that (7) is satisfied. Then, for a given traffic vector h, the set of all supporting capacity reservation vectors will be abbreviated by $\mathcal{U}_P(h)$. Note that $\mathcal{U}_P(h)$ is a polyhedron in $\mathcal{R}_+^{|E|}$.

A formal definition of ordinary domination, extending the one from [6] to networks with arbitrary graphs and demand sets D is as follows.

Definition 2. *Let P be the set of all elementary candidate paths for D in a network with arbitrary graph G. We say that $\hat{h}$ ordinarily dominates h ($\hat{h} \models_P h$ in short) if, and only if, each element u of the set $\mathcal{U}_P(\hat{h})$ (i.e., each capacity vector u supporting $\hat{h}$) is also an element of the set $\mathcal{U}_P(h)$ (i.e., u supports h), i.e., $\mathcal{U}_P(\hat{h}) \subseteq \mathcal{U}_P(h)$.*

(In the sequel we will skip subscript P in $\mathcal{U}_P(\hat{h}), \mathcal{U}_P(h)$ and in $\models_P$ when the path set P is fixed. Also, we will sometimes call ordinary domination just domination.)

As already mentioned in the introduction, in an incomplete graph there may no direct links between the end nodes of some demands, making the Oriolo result for ordinary domination not applicable. This can be seen in the example of Fig.1 where $\hat{h} \models h$, still h cannot be routed in $\hat{h}$. On the other hand, note that $\hat{h}$ can be routed in h and hence $h \models \hat{h}$.

3.2 Necessary and sufficient condition for ordinary domination

Consider a network with an arbitrary graph G, arbitrary demand set D, a given fixed set P of candidate paths (for example the set of all elementary paths for all demands in D), and two traffic vectors $\hat{h}$ and h defined for D. Let $\mathit{\Pi}$ denote the set of all vectors $\pi = (\pi_e,\ e \in E)$ such that $\pi_e \geq 0, e \in E$ and $\sum_{e \in E} \pi_e = 1$. Further, let $\lambda(\pi) = (\lambda_d(\pi),\ d \in D)$ denote the vector of the lengths of the shortest paths for the demands calculated for the link weights given by π.

Proposition 5. *$\hat{h} \models h$ if, and only if,*

$$\forall \pi \in \mathit{\Pi},\ \lambda(\pi)(\hat{h} - h) \geq 0. \tag{8}$$

Proof. Consider the following allocation problem related to (7) with a fixed capacity reservation vector u.

$$\text{minimize} \quad z \tag{9a}$$

$$[\lambda_d] \quad \sum_{p \in P_d} f_p = h_d \qquad d \in D \tag{9b}$$

$$[\pi_e] \quad \sum_{d \in D} \sum_{p \in Q_{ed}} f_p \leq u_e + z \qquad e \in E \tag{9c}$$

$$f_p \geq 0 \qquad p \in P. \tag{9d}$$

Certainly, capacity vector u supports h if, and only if, the optimal solution z^* of (9) is non-positive ($z^* \leq 0$). The problem dual to (9) reads:

$$\text{maximize} \quad W(\lambda, \pi) = \sum_{d \in D} \lambda_d h_d - \sum_{e \in E} \pi_e u_e \tag{10a}$$

$$\lambda_d \leq \sum_{e \in p} \pi_e \qquad d \in D,\ p \in P_d \tag{10b}$$

$$\sum_{e \in E} \pi_e = 1 \tag{10c}$$

$$\pi_e \geq 0 \qquad e \in E. \tag{10d}$$

Observe, that in any optimal solution (λ^*, π^*) we have that λ^*_d is equal to the length of the shortest path of demand $d \in D$ with respect to link weights π^*, i.e., $\lambda^*_d = \lambda_d(\pi^*)$. Since the optimal value of the dual function W^* ($W^* = W(\lambda^*, \pi^*)$) is equal to z^*, we deduce that capacity vector u supports h if, and only if, the optimal solution W^* of (10) is non-positive ($W^* \leq 0$), i.e., when $\sum_{d \in D} \lambda^*_d h_d \leq \sum_{e \in E} \pi^* u_e$ or, in the vector notation, $\lambda^* h \leq \pi^* u$. Because (10) is a maximization problem and because of (10b), the last condition is equivalent to: $\forall \pi \in \mathit{\Pi},\ \lambda(\pi) h \leq \pi u$.

Suppose that (8) is satisfied. By the above characterization, for each $u \in \mathcal{U}(\hat{h})$ we have that $\sum_{d \in D} \lambda_d \hat{h}_d \leq \sum_{e \in E} \pi_e u_e$ for each $\pi \in \mathit{\Pi}$. By (8), $\lambda(\pi^*) h \leq \lambda(\pi^*) \hat{h}$, and hence $\lambda(\pi^*) h \leq \pi^*_e u_e$ which means that $u \in \mathcal{U}(h)$. Thus $\hat{h} \models h$.

Now assume that $\hat{h} \models h$ and take some $\pi \in \mathit{\Pi}$. We define a specific flow $f(\pi)$ and the corresponding capacity reservation vector $u(\pi) \in \mathcal{U}(\hat{h})$ as follows. For every demand $d \in D$, put the entire demand volume $\hat{h}_d$ on one selected path in P_d that is shortest with respect to π, and define $u_e(\pi), e \in E$ as the resulting link loads. Consider the primal-dual solution pair $(f(\pi), u(\pi); \pi, \lambda(\pi))$. It follows from the saddle point conditions (see for example [4]) that the primal point $(f(\pi), u(\pi))$ is an optimal solution of (9), and the dual point $(\pi, \lambda(\pi))$ is an optimal solution of (10), so $\lambda(\pi)\hat{h} = \pi u(\pi)$. Since $u(\pi) \in \mathcal{U}(\hat{h})$, by assumption we have that $u(\pi) \in \mathcal{U}(h)$ and hence $\lambda(\pi)h \leq \pi u(\pi) = \lambda(\pi)\hat{h}$. Thus, $\lambda(\pi)(h-\hat{h}) \leq 0$ which means that (8) holds. ■

It can be shown (we omit a formal proof here) that problem

$$\text{minimize} \quad \lambda(\pi)(\hat{h} - h), \quad \pi \in \mathit{\Pi} \tag{11}$$

is $\mathcal{NP}$-hard. This fact strongly suggests that the problem of determining whether or not $\hat{h} \models h$ is $\mathcal{NP}$-hard as well.

3.3 Two special cases

In this section we will use the result of Proposition 5 to characterize domination in two important special cases. For the results discussed below we need an assumption that the path lists $P_d, d \in D$ contain all elementary paths between $a(d)$ and $b(d)$ in the network graph $G(V, E)$.

Case 1: $\hat{h}$ **directly routeable in** $G(V, E)$

Assume that $\hat{h}$ is directly routeable in $G(V, E)$. By this we mean that when $\hat{h}_d > 0$ then the network graph contains link $a(d)b(d)$ (in other words, for any demand $d \in D$ with $\hat{h}_d > 0$, its end nodes $a(d)$ and $b(d)$ are connected by a link in E). Define the capacity reservation vector $\hat{u} = (\hat{u}_e,\ e \in E)$ as: $\hat{u}_e = \hat{h}_d$ if $e = a(d)b(d)$ and $\hat{h}_d > 0$, and $\hat{u}_e = 0$, otherwise.

Proposition 6. *$\hat{h} \models h$ if, and only if, $\hat{u}$ supports h.*

Proof. Suppose that $\hat{h} \models h$. Since (trivially) $\hat{u}$ supports $\hat{h}$, by Definition 2 $\hat{u}$ supports h.

Now suppose that $\hat{u}$ supports h and consider any link metric vector $\pi \in \mathit{\Pi}$. For each pair of nodes $v, w \in V$ define α_{vw} as the length of the shortest elementary path in graph G with respect to link metrics π (we can assume, without loss of generality, that G is connected). Let $\hat{\pi} = (\hat{\pi}_e,\ e \in E)$ be the vector of link metrics given by $\hat{\pi}_e = \alpha_{vw}$ where $e = vw$. Notice that $\hat{\pi}_e \leq \pi_e, e \in E$ and that $\lambda_d(\pi) = \alpha_{a(d)b(d)}, d \in D$.

We will also show that $\lambda_d(\hat{\pi}) = \alpha_{a(d)b(d)}, d \in D$ which means that $\lambda_d(\hat{\pi}) = \lambda_d(\pi), d \in D$, i.e., the lengths of the shortest paths in G calculated for metrics $\hat{\pi}$ do not change with respect to the lengths calculated for metrics π. To see this observe first that $\lambda_d(\hat{\pi}) \leq \lambda(\pi)$ because $\hat{\pi}_e \leq \pi_e, e \in E$. Moreover, the strict inequality $\lambda_d(\hat{\pi}) < \lambda_d(\pi)$ (for a $d \in D$) would imply (contrary to the definition of $\lambda_d(\pi)$) that there exists an elementary path p between nodes $a(d)$ and $b(d)$ with $|p|_\pi < \lambda_d(\pi)$ (recall that $|p|_\pi$ denotes the length of path p calculated for link metrics π). Suppose that $\hat{p} = \{e_1, e_2, \ldots, e_n\}$ is a shortest path between $a(d)$ and $b(d)$ with respect to metrics $\hat{\pi}$ ($|\hat{p}|_{\hat{\pi}} = \lambda_d(\hat{\pi})$). Let $p^1, p^2, \ldots, p^n$ be a sequence of elementary paths in graph G between the end nodes of links $e_1, e_2, \ldots, e_n$, respectively, with $|p^i|_\pi = \hat{\pi}_{e_i}, i = 1, 2, \ldots, n$. If $\lambda_d(\hat{\pi}) < \lambda_d(\pi)$ then the concatenation of the paths $p^1, p^2, \ldots, p^n$ would contain an elementary path p between $a(d)$ and $b(d)$ with $|p|_\pi \leq \lambda_d(\hat{\pi})$. Hence, the strict inequality $\lambda_d(\hat{\pi}) < \lambda(\pi)$ would lead to a contradiction.

By assumption $\hat{u}$ supports h which means that $\lambda(\hat{\pi})h \leq \hat{\pi}\hat{u}$ (the fact that $\hat{\pi}$ is not normalized, i.e., that $\sum_{e \in E} \hat{\pi}_e$ can be less than 1, does not matter here). Hence, since $\hat{\pi}\hat{u} = \lambda(\pi)\hat{h}$ we finally get $\lambda(\pi)h \leq \lambda(\pi)\hat{h}$, and thus, by Proposition 5, $\hat{h} \models h$. ■

The result of Proposition 6 was first proven in [6], but only for fully connected network graphs.

Case 2: **Ring networks**

An important class of communication networks are ring networks whose graph $G(V, E)$ forms a cycle, i.e., a graph with n vertices (network nodes) and n edges (network links), and with $V = \{v_0, v_1, \ldots, v_{n-1}\}$ and $E = \{e_0, e_1, \ldots, e_{n-1}\}$ where $e_i = v_i v_{i+1},\ i = 0, 1, \ldots, n-1$ (in arithmetic modulo n).

We say that demand $d \in D$ crosses an *edge-cut* $\{e_i, e_j\}$ if one node of demand d is in $v_{j+1}, v_{j+2}, \ldots, v_i$ and the other node is in $v_{i+1}, v_{i+2}, \ldots, v_j$. Let $h(e_i, e_j)$ denote the load of cut $\{e_i, e_j\}$ induced by demand vector h, i.e., the sum of volumes h_d for all demands $d \in \mathcal{D}$ that cross the considered cut.

A theorem of Okamura and Seymour [5], when specialized to a cycle G, asserts that u supports h if, and only if, the *edge-cut condition*, that is,

$$\forall\ 0 \leq i, j < n,\ \ h(e_i, e_j) \leq u_{e_i} + u_{e_j} \tag{12}$$

holds (see [10]). Let $\hat{h}(e_i, e_j)$ denote the load of cut $\{e_i, e_j\}$ induced by demand vector $\hat{h}$.

Proposition 7. *$\hat{h} \models h$ if, and only if,*

$$\forall\ 0 \leq i, j < n,\ \ \hat{h}(e_i, e_j) \geq h(e_i, e_j). \tag{13}$$

Proof. Implication $\Leftarrow$ follows directly form the edge cut condition. Indeed, $u \in \mathcal{U}(\hat{h})$ means that

$$\forall\ 0 \leq i, j < n,\ \ \hat{h}(e_i, e_j) \leq u_{e_i} + u_{e_j}. \tag{14}$$

Hence, by assumption (13), property (12) holds, that is, $u \in \mathcal{U}(h)$.

The inverse implication can be proved analogously to the corresponding implication in Proposition 5. Let $\pi(e_i, e_j) \in \mathit{\Pi}$ denote the vector of multipliers corresponding to cut $\{e_i, e_j\}$ defined as follows: $\pi_e(e_i, e_j) = \frac{1}{2}$ if $e = e_i$ or $e = e_j$, and $\pi_e(e_i, e_j) = 0$, otherwise. Observe that for a cut $\{e_i, e_j\}$, inequality in (12) can equivalently be written as

$$\lambda(\pi(e_i, e_j))h \leq \pi(e_i, e_j)u. \tag{15}$$

Now assume that $\hat{h} \models h$ and consider a cut $\{e_i, e_j\}$. We define a capacity reservation vector $u(e_i, e_j) \in \mathcal{U}(\hat{h})$ as follows. For every demand $d \in D$, put the entire demand volume $\hat{h}_d$ on a shortest path with respect to $\pi(e_i, e_j)$ and for each $e \in E$ define $u_e(e_i, e_j)$ as the resulting link load. Note that the volumes $\hat{h}_d$ of the demands that do not cross cut $\{e_i, e_j\}$ do not contribute to $u_{e_i}(e_i, e_j)$ nor $u_{e_j}(e_i, e_j)$ since, by the definition of $\pi(e_i, e_j)$, their shortest paths are of length equal to 0. By the same argument the demands that cross the cut have the shortest path lengths equal to $\frac{1}{2}$, and their demand volumes contribute to $u_{e_i}(e_i, e_j) + u_{e_j}(e_i, e_j)$. This means that $\hat{h}(e_i, e_j) = u_{e_i}(e_i, e_j) + u_{e_j}(e_i, e_j)$ and hence $\lambda(\pi(e_i, e_j))\hat{h} = \pi(e_i, e_j)u(e_i, e_j)$. It follows that $\pi(e_i, e_j)$ is the optimal solution of the dual problem (10) for the capacity vector $u(e_i, e_j)$. By assumption, the capacity reservation vector $u(e_i, e_j)$ supports also h so for any $\pi \in \Pi$, $\lambda(\pi)h \leq \pi u(e_i, e_j)$. Thus, we finally see that

$$\frac{1}{2}h(e_i, e_j) = \lambda(\pi(e_i, e_j))h \leq \pi(e_i, e_j)u(e_i, e_j) = \lambda(\pi(e_i, e_j))\hat{h} = \frac{1}{2}\hat{h}(e_i, e_j) \tag{16}$$

which means that $\hat{h}(e_i, e_j) \geq h(e_i, e_j)$. ■

4 Concluding remarks

In the paper we have discussed necessary and sufficient (n-s) conditions for total and ordinary traffic domination. For the first case (total domination) we have presented a general n-s condition which can be easily checked in polynomial time. For the second case we have found an n-s condition that gives a strong evidence that checking for ordinary domination is $\mathcal{NP}$-hard, except for the two special cases we have been able to find. In fact, devising effective approximation methods for checking for ordinary domination seems to be an important and challenging research direction.

It should be noted that in Section 2 the results for total domination are given for undirected network graphs are not generally applicable to directed (even to bi-directed) graphs. On the contrary, the results for ordinary domination presented in Proposition 5 and in Proposition 6 are valid for both undirected and directed network graphs.

Acknowledgement: While working on the paper during his stay as an invited professor at Warsaw University of Technology, W. Ben-Ameur was supported by European Union in the framework of European Social Fund. P. Pavón was supported by the FP7 BONE project, by the MEC project TEC2010-21405-C02/TCM CALM, and by “Programa de Ayudas a Grupos de Excelencia de la R.

de Murcia, F. Séneca" – he had also stayed at Warsaw University of Technology while working on the results presented in the paper. M. Pióro was supported by the Polish Ministry of Science and Higher Education (grants no. 280/N-DFG/2008/0 and N517 397334), and by the Swedish Research Council (grant no. 621-2006-5509).

References

1. W. Ben-Ameur and H. Kerivin. Routing of uncertain traffic demands. *Optimization and Engineering*, 3:283–313, 2005.
2. W. Ben-Ameur and M. Pióro. On traffic domination and total traffic domination in communication networks with arbitrary graphs. Technical report, Institute of Telecommunications, Warsaw University of Technology.
3. D. Jungnickel. *Graphs, Networks and Algorithms.* Springer, 1999.
4. L. Lasdon. *Optimization Theory for Large Systems.* McMillan Publishing Co., Inc., 1970.
5. H. Okamura and P.D. Seymour. Multicommodity flows in planar graphs. *Journal of Combinatorial Theory*, 31(1):75–81, 1981.
6. G. Oriolo. Domination between traffic matrices. *Mathematics of Operations Research*, 33(1):91–96, 2008.
7. P. Pavon-Marino, B. Garcia-Manrubia, and R. Aparicio-Pardo. Multi-hour network planning based on domination between sets of traffic matrices. *to be published in Computer Networks.*
8. P. Pavon-Marino and M. Pióro. On total traffic domination in non-complete graphs. *submitted to OR Letters.*
9. M. Pióro and D. Medhi. *Routing, Flow, and Capacity Design in Communication and Computer Networks.* Morgan-Kaufmann, 2004.
10. B. Shepherd and L. Zhang. A cycle augmentation algorithm for minimum cost multicommodity flows on a ring. *Discrete Applied Mathematics*, 110(2-3):301–315, 2001.
11. S. E. Terblanche. *Contributions towards survivable Network Design with Uncertain Traffic Requirements.* PhD thesis, North-West University, South Africa, 2008.