

Meter installation for monitoring network traffic

Philippe Meurdesoif, *University Bordeaux 1, Applied Mathematics*

Pierre Pesneau, *University Bordeaux 1, Applied Mathematics*

François Vanderbeck, *University Bordeaux 1, Applied Mathematics*

Keywords: Network flow with installation costs, polyhedral approach, quadratic programming.

1. Introduction

Several interesting combinatorial problems arise from applications in telecommunication or traffic network management. Aside from the well-studied questions of network design [5] or routing [9], the problem considered here concerns the tracing of the traffic in a network with appropriate equipments while minimizing the cost of their installation. More precisely, given a network and traffic going through it, the question is what is the minimal number of arcs where traffic must be monitored to be able to reconstruct all individual routes *a posteriori*. The application that motivates this study is the tracing of skier routes in a ski resort but it could just as well be the tracing of telephone calls in a telecommunication network, or vehicles in a street network, or monitoring the run of a computer program (viewed as a path) for its validation (introduced in [8]). To the best of our knowledge no prior solution approach were proposed for such problem ([8] only treats the complexity issue).

In the rest of the paper, we shall make reference to the ski resort application. We will moreover assume that the network contains no directed cycle (a natural assumption for this application as we will see). Beyond this restrictive assumption, the results extend to other applications. The ski resort can naturally be modelled as a directed multi-graph $G(V, A)$: nodes will represent bifurcations, *i.e.*, places where a skier has to choose between several tracks; arcs correspond to track portions between two consecutive bifurcations with “down arcs” corresponding to natural down-hill tracks and “up arcs” to ski-lifts. Two extra nodes are introduced to represent respectively entrance and exit from the network: all entries (resp. exits) will be arcs coming from (resp. going to) the entry (resp. exit) node. We assume that the skiers’ ski-passes are equipped with magnetic tags. The question is what are the best places to install magnetic tag detectors that will provide individual information on the identity of any skier crossing their electromagnetic field at any time.

Clearly, it is sufficient to place at most one reader on any down-hill slope whose corresponding arc will be said to be *measured*. A path (viewed as a set of consecutive arcs) will be said *non-measured* if no arc in the path is measured. This terminology will also apply to the directed cycles of the graph and to the so-called double-paths. A *double-path* between i and j is a pair of distinct paths from i to j ; it is said *elementary* if both paths are node-disjoint. The characterization of the solutions to our problem is given by the following result.

Proposition 1 *All individual routes can be reconstructed iff there exist no non-measured directed cycles or double-paths.*

Proof: The condition is clearly necessary: if there exists a *non-measured* cycle, an individual can walk endlessly along the cycle without being measured; if there exist several *non-measured* paths between two nodes, one cannot determine which one was chosen by an individual that was measured up-hill and down-hill of these two nodes. Conversely, the condition is sufficient. Let a_1 and a_2 the arcs where two successive measures of an individual take place; let i be the head-node of a_1 and j the tail-node of a_2 . If $i = j$, the absence of *non-measured* directed cycles, guarantees that the route clearly use a_1 and a_2 in direct sequence. While, if $i \neq j$, as there is at most one *non-measured* path between i and j , then it must be the path followed by our individual route. ■

In the ski resort application, the skier is considered as entering the network at the tail-node of the arc where he

was first measured and exiting the network at the head-node of the arc where he was last measured with no possibility of returning from exit node to entry node. Moreover, one may assume that “up arcs” are already measured (since access to ski-lift is limited to tag carrying skiers). Thus “up arcs” can be removed of the graph which leaves a graph with no directed cycles (as any circuit must feature an up arc). In the sequel we therefore make the restrictive assumption that the graph is acyclic (then, we can also assume that the vertices have been numbered in a topological order). Hence, our solutions are simply characterized by the fact that the network cannot contain multiple *non-measured* paths between any pair of nodes i and j . We denote by \mathcal{DP} the set of the double-paths of G . Even with this assumption, our problem is NP-Hard in a strong sense: our problem is equivalent to the traversal marker placement problem in software validation of [8] that was proved strongly NP-Hard therein by reduction from Minimum Vertex Cover Problem (a dual of Maximum Stable Set).

The problem admits a weighted variant whereby instead of minimizing the number of monitored arcs, we consider different cost of installing monitoring equipment in each arc. We shall make use of two complementary models: either we maximize the cost (or the number) of *non-measured* arcs while ensuring that the selected arc draw no more than one path between any pair of nodes, or we minimize the cost (or the number) of *measured* arcs while ensuring that all paths but one (at most) are measured between any two nodes. We develop two formulations for the problem, one that is well suited for a cutting plane approach and the other inspired by a quadratic programming approach (although we proceed to linearize it). Our paper includes a partial characterization of the associated polyhedra, the presentation of separation routines and the theoretical comparison of the proposed formulations. Preliminary computational results complete this comparison. For the weighted case, primal bounds can be obtained by adaptation of Kruskal’s algorithm for the maximum weight spanning tree (this heuristic algorithm produces a double-path-free set of arcs of high total weight). A more combinatorial approach relies on adapting greedy approaches from set covering: the heuristic consists in iteratively selecting an arc whose cost is small compared to the the number of double-path that it “covers”, which requires a way of counting how many double-paths an arc belongs to.

2. A polyhedral approach

Our first formulation models the MCMC problem as a set covering (SC) problem: the ground set of double-paths \mathcal{DP} must be covered by *measured* arcs. We define for each arc (i, j) a variable x_{ij} stating whether arc (i, j) is *measured*. For a given set of arcs F , let us denote by x^F the incidence vector induced by F that is the vector such that $x_{ij}^F = 1$ for each arc of F and 0 otherwise; while $x(F)$ denotes the number of arcs selected in F , i.e. $x(F) = \sum_{(ij) \in F} x_{ij}$. Then, the formulation takes the form

$$\begin{aligned} \min \quad & \sum_{(i,j) \in A} c_{ij} x_{ij}, \\ \text{SC} \quad & \text{s.t.} \quad x(C) \geq 1 \quad \forall C \in \mathcal{DP}, \tag{1} \\ & 0 \leq x_{ij} \leq 1 \quad \forall (i, j) \in A, \tag{2} \\ & x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A. \tag{3} \end{aligned}$$

Inequalities (1) will be called double-path cover inequalities and state that each double-path of \mathcal{DP} have to be measured. Inequalities (2) (resp. (3)) will be called trivial (resp. integral) inequalities.

Let $\text{SC}(G)$ be the polytope associated with the MCMC problem for the graph G . In the rest of this section, we will give some additional classes of inequalities that are valid for $\text{SC}(G)$, but first we will establish the dimension of the polytope and give the conditions for the trivial inequalities to be facet defining.

Proposition 2 *The polytope $\text{SC}(G)$ is full dimensional.*

Proposition 3 *1. Inequality $x_{ij} \leq 1$ is facet defining for every arc $(i, j) \in A$.
2. Inequality $x_{ij} \geq 0$ is facet defining iff there are no multiple arcs between the nodes i and j .*

Double-path cover inequalities:

First, observe that we can restrict our attention to inequalities (1) for elementary double-paths (i.e. pairs of paths that are node disjoint except for their common origin and destination). Let \mathcal{EDP} denote the set of elementary double-paths.

Proposition 4 *A double-path cover inequality is facet defining only if the double-path inducing the inequality is elementary.*

Proof Let C be the double-path inducing the inequality, and i and j be the origin and destination of the paths inducing C . Suppose that C is not elementary. Let s be the first node where the two paths diverge and t be the first node where the two paths converge. As C is not elementary, at least one of the nodes s and t is different from i and j . Let C' be the double-path obtained by considering only the subpaths between s and t of the paths inducing the double-path C . It is clear that $C' \subset C$. Moreover, C' is also a double-path. Thus the double-path cover inequality induced by C' dominates the one induced by C and this last inequality cannot define a facet of $SC(G)$. ■

It follows that an arc (i, j) that does not belong to any elementary double-path shall not appear in any inequalities (1) on \mathcal{EDP} and hence its value can be set to zero. Equivalently, we can remove such an arc from the graph and reduce the dimension of the problem.

The separation problem associated with the class of double-path cover inequalities can be solved in polynomial time. Given a solution \hat{x} , this problem consists in checking whether all double-path cover inequalities are satisfied and if not, providing a violated one. One can solve it by computing, for every pair of nodes $(s, t) \in V^2$, a 2-shortest-path between s and t where arc costs are defined by \hat{x} . This can be done in $O(nm)$ operations by adapting a shortest-(acyclic-)path algorithm. It follows that the linear relaxation of formulation SC can be solved in polynomial time. Yet the obtained lower bound, denoted by Z_{LP}^{SC} can be quite weak (see examples in Section 4.) and the associated solution can be indeed very fractional. Cuts are necessary to sharpen the solution and improve the lower bound.

 k -Path cover inequalities:

The elementary double-path cover inequalities can be generalized to so-called k -path cover inequalities:

Proposition 5 *Let p_1, \dots, p_k be $k \geq 2$ pairwise distinct paths between two vertices s and t . Then the inequality*

$$\sum_{i=1}^k x(p_i) \geq k - 1, \quad (4)$$

is valid for $SC(G)$.

Proof The proof is by induction on k . For $k = 2$, inequality (4) is clearly valid as it is a weaker version of a double-path cover inequality (indeed note that if the paths are not pairwise arc-disjoint, some x_{ij} may appear in the sum with multiplicity more than one). Suppose the inequality is valid for $k - 1$ distinct paths and consider k pairwise distinct paths p_1, \dots, p_k . Let \mathcal{P}_i , $i = 1, \dots, k$, be the set of paths defined by $\{p_1, \dots, p_k\} \setminus \{p_i\}$. By adding together the k inequalities (4) defined by the sets \mathcal{P}_i , $i = 1, \dots, k$, we get

$$\sum_{i=1}^k (k - 1)x(p_i) \geq k.$$

The inequality (4) induced by p_1, \dots, p_k is then obtained by dividing the precedent inequality by $k - 1$ and rounding up the right hand side. ■

There is little hope to find a separation algorithm polynomial in k , but for fixed k we can compute by a dynamic program the l th shortest-path in polynomial time ([3]). However, a subset of this class of k -path

cover inequalities can be separated in polynomial time. Consider pairwise node-disjoint paths. In this case, all the coefficient of the variables in the constraint are either 0 or 1. Moreover, as the paths are pairwise node-disjoint, their number is upper bounded by the number of nodes minus 2. The separation problem for this subclass of k -path cover inequalities can then be solved in polynomial time by computing, for each couple of nodes s and t and each value of k , the minimum cost of sending k units of flow between s and t where the costs are given by the current solution. If this cost is less than $k - 1$ then the k paths that define the corresponding flow induce a violated k -path cover inequality. An efficient way of implementing this separation algorithm for a given pair of nodes (s, t) would be to use a shortest path augmentation algorithm. By this way, for a given k , we can take advantage of the solution obtained for $k - 1$ by starting the algorithm with the residual graph induced by the solution for $k - 1$.

⊖ Inequalities :

The following class of inequalities can be seen as a strengthening of 3-path cover inequalities when the 3-path has a particular structure called Θ .

Proposition 6 *Let p_1 and p_2 be two node-disjoint paths between two nodes s and t . Let a and b be two nodes of the paths p_1 and p_2 respectively, different from s and t . Let p_3 be a path between a and b that is node-disjoint from p_1 and p_2 (except for the nodes a and b). Let us pose $\Theta = \bigcup_{i=1}^3 p_i$. Then the inequality*

$$x(\Theta) \geq 2 \tag{5}$$

is valid for SC(G).

Proof Note that Θ contains exactly 3 elementary double-paths: the first is defined by p_1 and p_2 , the second one by the subpath of p_1 from s to a , p_3 and the subpath of p_2 from s to b , and the third double-path is defined by the subpath of p_1 from a to t , p_3 and the subpath of p_2 from b to t . By adding the three double-path cover inequalities induced by these three double-path, we have

$$2x(\Theta) \geq 3.$$

Dividing this inequality by 2 and rounding up the right hand side completes the proof. ■

Given a fractional solution \hat{x} , the separation problem for Θ inequalities consists in finding a set Θ that minimizes $\hat{x}(\Theta)$. If $\hat{x}(\theta) \geq 2$, then all Θ inequalities are satisfied, otherwise, the set Θ will induce a violated Θ inequality. This problem is indeed polynomial, since there exists an optimal Θ such that (with the s, a, b, t notation above) $s - a, s - b, a - b, a - t, b - t$ are shortest paths. An optimal Θ can therefore be found by enumerating on pairs (a, b) then find (separately) s and t minimizing the sum of shortest paths (all shortest paths between any pair of vertices having been computed at first once for all), which can be done in $O(n^3)$ operations.

3. A quadratic programming based approach

Using the above notation, arc (i, j) is *non-measured* if $x_{ij} = 0$. We define $\bar{x}_{ij} = 1 - x_{ij} = 1$ iff (i, j) is *non-measured* and $y_{ij} = 1$ iff there is an unmeasured path from i to j . Using these variables, our problem can be formulated in terms of unmeasured paths. The formulation involves quadratic constraints:

$$\begin{aligned} \min \quad & \sum_{(i,j) \in A} c_{ij} (1 - \bar{x}_{ij}) \\ \text{QP} \quad & \text{s.t.} \quad y_{ik} \geq y_{ij}y_{jk} \quad \forall (i, j, k) \in V^3, \tag{6} \\ & \sum_{k \in V(j)} y_{ik}\bar{x}_{kj} = \begin{cases} y_{ij} - \bar{x}_{ij} & \forall (i, j) \in A, \\ y_{ij} & \forall (i, j) \in V^2 \setminus A, \end{cases} \tag{7} \\ & \bar{x}_{ij} \in \{0, 1\} \quad \forall (i, j) \in A, \tag{8} \\ & y_{ij} \in \{0, 1\} \quad \forall (i, j) \in V^2, \tag{9} \end{aligned}$$

where $V(j) \subset V$ denotes the set of nodes incident to j . Constraints (6) enforce the transitive relation between paths. Constraints (7) play a double role: first, they define path that can be obtained either by adding an extra arc (k, j) to an existing path between i and k or by using a direct arc (i, j) when it exists; second, they ensure that no more than one path exists between i and j . With integer variables the latter inequalities can be shown to induce a transitive definition of path (by induction on $j - i$ for (i, j) -paths, with vertices numbered in a topological order); however the transitivity constraints (6) remain useful for the linear programming relaxation.

The standard linearization to formulation QP is denoted LQP:

$$\text{LQP} \quad \min \sum_{(i,j) \in A} c_{ij}(1 - \bar{x}_{ij})$$

$$\text{s.t.} \quad y_{ik} \geq y_{ij} + y_{jk} - 1 \quad \forall (i, j, k) \in V^3, \quad (10)$$

$$\sum_{k \in V(j)} w_{ikj} = \begin{cases} y_{ij} - \bar{x}_{ij} & \forall (i, j) \in A, \\ y_{ij} & \forall (i, j) \in V^2 \setminus A, \end{cases} \quad (11)$$

$$w_{ijk} \leq \bar{x}_{jk} \quad \forall (i, (jk)) \in V \times A, \quad (12)$$

$$w_{ijk} \leq y_{ij} \quad \forall (i, (j, k)) \in V \times A, \quad (13)$$

$$w_{ijk} \geq y_{ij} + \bar{x}_{jk} - 1 \quad \forall (i, (j, k)) \in V \times A, \quad (14)$$

$$\bar{x}_{ij} \in \{0, 1\} \quad \forall (ij) \in A, \quad (15)$$

$$y_{ij} \in \{0, 1\} \quad \forall (i, j) \in V^2, \quad (16)$$

$$w_{ijk} \in \{0, 1\} \quad \forall (i, (j, k)) \in V \times A, \quad (17)$$

where variables w_{ijk} have been introduced to replace $y_{ij}\bar{x}_{jk}$ ($w_{ijk} = 1$ iff there is an unmeasured path from i to k ending with arc (j, k)).

One can easily show that formulation LQP is better than SC, i.e. it yields a better LP bound (that are denoted by Z_{LP}^{LQP} and Z_{LP}^{SC} respectively).

Proposition 7 $Z_{LP}^{SC} \leq Z_{LP}^{LQP}$

Proof: We show that any solution (\bar{x}, y, w) to the LP relaxation of LQP satisfies constraints (1). First note that constraints (1) can be written in terms of variables \bar{x} in the form $\sum_{(ij) \in p} (1 - \bar{x}_{ij}) \geq 1 \Leftrightarrow \bar{x}(p) \leq \ell(p) - 1$ where $\ell(p)$ is the number of arcs in p . Second note that, for any elementary path p of length $\ell(p) = k$, i.e., $p = (i_0, i_1, i_2, \dots, i_k)$, we have

$$w_{i_0, i_{k-1}, i_k} \geq \bar{x}(p) - \ell(p) + 1.$$

Indeed, making recursive use of constraints (12-14), one can show that

$$\begin{aligned} w_{i_0, i_{k-1}, i_k} &\geq y_{i_0, i_{k-1}} + \bar{x}_{i_{k-1}, i_k} - 1 \\ &\geq w_{i_0, i_{k-2}, i_{k-1}} + \bar{x}_{i_{k-1}, i_k} - 1 \\ &\geq y_{i_0, i_{k-2}} + \bar{x}_{i_{k-2}, i_{k-1}} + \bar{x}_{i_{k-1}, i_k} - 2 \\ &\geq \dots \\ &\geq y_{i_0, i_1} + \bar{x}_{i_1, i_2} + \dots + \bar{x}_{i_{k-1}, i_k} - \ell(p) \\ &\geq \bar{x}_{i_0, i_1} + \dots + \bar{x}_{i_{k-1}, i_k} - \ell(p) + 1. \end{aligned}$$

Then, let $C = (p_1, p_2)$ be an elementary double-path from s to t and let i and j be respectively the last node before t in p_1 and p_2 . Since no arc belongs to both p_1 and p_2 we have:

$$1 \geq y_{st} \geq w_{sit} + w_{sjt} \geq \bar{x}(p_1) + \bar{x}(p_2) - \ell(p_1) - \ell(p_2) + 2,$$

and therefore

$$\bar{x}(C) \leq \ell(C) - 1.$$

■

Observe that all cuts presented above for formulation SC can be easily adapted to model LQP, via the change of variables $x_{ij} \rightarrow (1 - \bar{x}_{ij})$. LQP is a compact formulation involving a polynomial number of variables and constraints; yet its large number of variables and constraints can make it unpractical to compute.

4. Preliminary computations

We plan to develop a full blown computational comparison of the above two formulations. At the moment, we only have preliminary results using formulation SC.

On grid graphs: We consider a graph $Gr_{n,n}$ with $(n+1)$ vertices labelled $v_{1,1}, v_{1,2}, \dots, v_{1,n+1}, \dots, v_{n+1,n+1}$ and $2n(n+1)$ arcs from vertex $v_{i,j}$ to $v_{i,j+1}$ and $v_{i+1,j}$ (this graph can be drawn on a $n \times n$ grid). We assume that $c_{ij} = 1$ for any arc (ij) (the unweighted case). By constructing primal and dual solutions of same value, we can easily compute Z_{LP}^{SC} on such graphs when all $c_{ij} = 1$: $Z_{LP}^{SC}(Gr_{n,n}) = \frac{1}{2}n^2 + 1$. Adding Θ -cuts improves the solution : $Z_{LP}^{SC+\Theta}(Gr_{n,n}) \geq \frac{4}{7}n^2 - \frac{1}{7} \quad \forall n \geq 4$ (once again by constructing a dual solution), these dual bound can be compared to greedy solutions where the number of meters placed is asymptotically equivalent to $\frac{2}{3}n^2$ (although not known exactly). These results suggest that such instances appear to be particularly bad cases for the linear bounds described above.

On a real-life instance: We tested the first linear bounds on a real-life instance, modelling the French ski resort of Aussois: the graph involves 33 vertices and 57 arcs. Easy preprocessing can be applied to remove arcs belonging to no elementary double-paths (they will never be measured in an optimal solution). Then, $Z_{LP}^{SC} = 14.5$. Including Θ cuts gives a bound of $Z_{LP}^{SC+\Theta} = 15.325$. Yet, the optimal value is 17. This suggests that even on real graphs, where linear bounds are a little tighter, there is place to enhance the formulation.

References

- [1] N. Biggs. Algebraic Graph Theory. Cambridge Mathematical Library, 1974.
- [2] V. Chvátal. A greedy heuristic for the set cover problem. *Mathematics of Operation Research* 4, 233–235, 1979.
- [3] M. Gondran, M. Minoux. Graphes et algorithmes. Editions Eyrolles, Paris. 3rd edition, 1995.
- [4] D. S. Johnson. Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences* 9, 256–278, 1974.
- [5] H. Kerivin and A. R. Mahjoub. Design of Survivable Networks: A survey in *Networks*, 46, 1–21, 2005.
- [6] B. Korte, J. Vygen. Combinatorial Optimization : theory and algorithms. Springer 2002.
- [7] L. Lovász. On the ratio of optimal integral and fractional covers. *Discrete Mathematics* 13, 383–390, 1975.
- [8] S. Maheshwari. Traversal Marker Placement Problem are NP-Complete. Research report no CU-CS-092-76, Dept. of Computer Science, University of Colorado at Boulder, 1976, 6p.
- [9] M. Steenstrup (editor). Routing in Communications Networks, Prentice Hall International (UK) Ltd. 1995.