Elementary shortest-paths visiting a given set of nodes

Rafael Castro de Andrade
Universidade Federal do Ceará
Departamento de Estatística e Matemática Aplicada
Campus do Pici, Bloco 910. CEP 60.455-760 - Fortaleza, Ceará - Brasil
rca@lia.ufc.br

ABSTRACT

Consider a directed graph $G = (V, A)$ with set of nodes V and set of arcs A and let c_{uv} denote the length of an arc $uv \in A$. Given two distinguished nodes s and t of V we are interested in the problem of determining a shortest-path (in length) from s to t in G that must visit only once all nodes of a given set $P \subseteq V - \{s, t\}$, but not necessarily only these nodes. This problem is NP-hard for $P = V - \{s, t\}$. We develop two compact extended formulations for this problem. One is based on an adapted version of the cycle elimination constraints of the spanning tree polytope and the other is a new primal-dual based mixed integer formulation. Numerical experiments show that these formulations are very efficient in solving random generated instances of the problem, including Hamiltonian paths of minimum length.

KEYWORDS. shortest path visiting given nodes, compact extended formulation, linked dual-primal formulation, combinatorial optimization.

1. Introduction

The elementary $(s-P-t)-$shortest-path (for short) in a directed graph $G = (V, A)$ with set of nodes V and set of weighted arcs A consists in finding a path of minimum length between an origin node $s \in V$ and a destination node $t \in V$ that visits only once all nodes of a given set $P \subseteq V - \{s, t\}$. We know that for $P = V - \{s, t\}$ the problem is equivalent to find an Hamiltonian path of minimum length in G, which is NP-hard. In a brief literature review, we find few works on this problem (Dreyfus, 1969; Ibaraki, 1973; Saksena and Kumar, 1966) with such a solution structure. However, we can find related works (Volgenant and Jonker, 1987; Feillet et al., 2001; Letchford et al., 2013) on some NP-hard versions of this problem. Indeed, the Steiner Traveling Salesman Problem (STSP) (Letchford et al., 2013) ask for non elementary tours visiting a given set of nodes as feasible solutions (i.e. tours where nodes or arcs can be visited more than once in a feasible solution). The reader is referred to a vast literature on the TSPLIB for many solution techniques, models and algorithms for related TSP problems. Here we concentrate our effort exclusively on the elementary $(s - P - t)-$shortest-path structure, because we explore it to develop new formulations for this problem.

It seems that the first (and erroneous as showed by (Dreyfus, 1969)) algorithm for the $(s - P - t)-$shortest-path problem is due to (Saksena and Kumar, 1966). (Dreyfus, 1969) proposes to solve the problem by reducing it to an instance of the traveling salesman problem. (Ibaraki, 1973) introduces an exponential dynamic programming algorithm and a branch and bound (B&B) method. Ibaraki's model used in the B&B algorithm is defined only with continuous variables. His model is equivalent to the well known flow formulation of the classic shortest-path problem, thus relaxed node solutions in the search B&B tree are integer and present at least one cycle (Ibaraki, 1973) when the node solution is not an elementary path. The idea behind the Ibaraki's B&B algorithm is to fix at zero (one at a time) an arc of a given cycle C of a B&B node solution as branching rule to create $|C|$ new B&B subproblems. This means possibly enumerating all cycles in G in a B&B tree, because relaxed solutions of the flow based model in (Ibaraki, 1973) showed to be very weak for this problem. Because of this, we are not encouraged to extend the models presented in (Letchford et al., 2013) to our problem.

Instead, we adapt the cycle elimination constraints of the compact extended formulation for the spanning tree polytope of an undirected graph in (Martin, 1991; Yannakakis, 1991; Conforti et al., 2010) to deal with the oriented arcs of the $(s - P - t)-$shortest-path problem. In fact, we show that the Martin's formulation is not suitable for oriented graphs. Moreover, we explore a nice property of elementary paths to obtain a primal-dual based mixed integer compact extended formulation. The novelty is to characterize feasible solutions by linking primal and dual variables in a unique set of constraints exploring that property. We do this without implementing the known complementary slackness optimality condition. On the best of our knowledge, this is the first work exploring these techniques for solving the $(s - P - t)-$shortest-path problem.

2. Problem formulation

Consider $G = (V, A)$ a directed graph with set of nodes V and set of weighted arcs A. Let $c_{uv} \in \mathbb{R}_+$ represent the length of arc $uv \in A$. The problem is to determine an elementary path in G of minimum length between an origin node $s \in V$ and a destination node $t \in V$ that visits a given set $P \subseteq V - \{s, t\}$. We represent a $(s - P - t)$-path in G by a vector $x \in \{0,1\}^{|A|}$, where $x_{uv} = 1$ if uv belongs to the $(s-P-t)$-path, and $x_{uv} = 0$, otherwise. Thus, a mathematical model for this problem is

$$(Q) \quad \min_{x \in \{0,1\}^{|A|}} \sum_{uv \in A} c_{uv} x_{uv} \tag{1}$$

$$s.t. \quad \sum_{i \,|\, iv \in A} x_{iv} - \sum_{j \,|\, vj \in A} x_{vj} = \begin{cases} 1, & \text{if } v = s \\ -1, & \text{if } v = t \\ 0, & \text{otherwise} \end{cases} \quad \forall\, v \in V \tag{2}$$

$$\sum_{u \in V \,|\, uv \in A} x_{uv} = 1, \ \forall\, v \in P \tag{3}$$

$$\sum_{uv \in A(S)} x_{uv} \leq |S| - 1, \ \forall\, S \subset V \tag{4}$$

where $A(S)$ represents the set of arcs with both extremities in S. Constraints (2) define an unrestricted $(s-t)$-path in G. In (3) we impose that each node $v \in P$ must be visited by imposing that one arc enters v. Constraints (4) avoid the existence of cycles in any solution. Note that the number of these sub-tour elimination constraints is exponential. In this case, one can try to solve problem (Q) iteratively by relaxing the constraints (4) and cutting off cycles obtained at each iteration. This means solving a MIP model each iteration until its corresponding solution presents no cycle. Eliminating cycles is also the idea of the branch-and-bound algorithm in (Ibaraki, 1973), where the authors use a flow-based model that is equivalent to the one defined by (1)-(3).

Alternatively, we can adapt a compact extended formulation of (Yannakakis, 1991) for the spanning tree polytope of a non oriented and complete graph to deal with non complete digraphs. The sub-tour elimination constraints discussed in (Martin, 1991; Yannakakis, 1991; Conforti et al., 2010) are obtained based on rooted spanning trees. The reader is referred to (Martin, 1991) on how he discovered these constraints. We advert that the way decision variables are defined (interpreted) to represent these constraints in the above cited works is confusing. Be in mind that their models are correct only if determining trees in non oriented complete graphs.

To introduce our new formulation for the minimum length elementary $(s-P-t)$-path of $G=(V,A)$, we work with a complete digraph $G^+ = (V, A^+)$, with $A^+ = \{(u,v) \in V \times V \mid u \neq v\}$ being the set of arcs of G^+.

Let the decision variables x represent the characteristic vector associated with the arcs of A^+ and let $G^+(x)$ denote the subgraph of G^+ induced by the entries of x equal to one.

Let $G^+_k(x)$, for every (referential) node $k \in V$, represent an "abstract" orientation of the arcs present in x (here we do not consider the concrete [original] orientation of the arcs in x).

Proposition 1 below is an extension of a related result in (Adasme et al., 2013) to deal with the characterization of forests in oriented graphs.

Proposition 1. *Let $G^+(x)$ be a subgraph of G^+ induced by the characteristic vector x. There exist independent abstract orientations $G^+_k(x)$ of the arcs of $G^+(x)$, one for each referential node $k \in V$, verifying simultaneously the following conditions in each $G^+_k(x)$:*

1. *There is no abstract arc in $G^+_k(x)$ entering the referential node k;*

2. *There is at most one abstract arc in $G_k^+(x)$ entering a node $u \in V - \{k\}$;*

if and only if $G^+(x)$ is acyclic.

Proof. If $G^+(x)$ is acyclic, then the result is straightforward. It is not difficult to find independent abstract orientations $G_k^+(x)$ of the arcs of $G^+(x)$, one for each referential node k, verifying simultaneously the conditions above. We have to show in fact that if the digraph $G^+(x)$ induced by the non null components of x contains a cycle, then the two conditions (1) and (2) cannot be satisfied simultaneously. To see this, suppose that$G^+(x)$ contains a cycle $C = (V(C), A(C))$, with $V(C)$ and $A(C)$ being the set of nodes and arcs of C, respectively. In any abstract orientation of the arcs in $A(C)$, there are two abstract arcs adjacent to any node of $V(C)$. Consider some $k \in V(C)$ as a referential node. In this case, if condition (1) is satisfied (two abstract arcs leave the referential node k), then necessarily at least one of the remaining nodes in $V(C)$ must have two abstract arcs entering it, thus violating condition (2). In the other hand, if condition (2) is satisfied (we have one abstract arc leaving and other abstract arc entering every node in $V(C)$), then none of the nodes in $V(C)$ can be a referential. If it were the case, condition (1) should be violated because it limits the number of abstract arcs entering the referential node to zero. Thus, if $G^+(x)$ contains a cycle, both conditions above cannot be satisfied, thus concluding the proof. □

We use Proposition 1 to reformulate the problem (Q) as follows. Define now decision variables λ_{kij} to represent an abstract orientation of the arcs in $G_k^+(x)$, where k is a referential node and i and j are extremities of an arc (i, j) of $G^+(x)$. We set $\lambda_{kij} = 1$ if for the referential node k, the node j is predecessor of the node i in the abstract orientation $G_k^+(x)$; and $\lambda_{kij} = 0$, otherwise.

This interpretation we give for the λ variables differs from the one in (Martin, 1991; Yannakakis, 1991; Conforti et al., 2010). This is due to the fact that our solution is not a spanning tree. Moreover, our graph is oriented. Thus, we do not impose that the valuation of these variables depends "exactly" on the presence or absence of an arc (i, j) or (j, i) in the problem solution. If it were the case, this dependence could be defined as in (Conforti et al., 2010) by equality constraints $\lambda_{kij} + \lambda_{kji} = x_{ij}$ (for the arc (i, j)) and $\lambda_{kij} + \lambda_{kji} = x_{ji}$ (for the arc (j, i)), if both arcs are in A, for every referential node k. But these constraints cannot appear together in a same model because if we suppose that one of this arcs belongs to a feasible solution (e.g. $x_{ij} = 1$ and $x_{ji} = 0$), this results in an infeasible system. We overcome this in such a way that if an arc (i, j) belongs to the solution, then we impose that at least one of the corresponding λ variables must have the value equal to one, i.e. a referential node k observes j preceding i, say $\lambda_{kij} = 1$; or i preceding j, say $\lambda_{kji} = 1$ in $G_k^+(x)$.

Thus, a compact extended formulation for (Q) is

$$
\begin{array}{llll}
(Q2) & \min\limits_{x\in\{0,1\}^{|A^+|}} & \sum\limits_{uv\in A} c_{uv}x_{uv} & (5)\\
& s.t. & (2)-(3) & \\
& & \lambda_{kij}+\lambda_{kji} \geq x_{ij},\ \forall\, i,j,k \in V,\ i\neq j & (6)\\
& & \sum_{j\in V-\{i\}} \lambda_{kij} \leq 1,\ \forall\, i,k \in V,\ i\neq k & (7)\\
& & \lambda_{kkj} = 0,\ \forall\, j,k \in V,\ k\neq j & (8)\\
& & x_{uv} = 0,\ \forall\, u\in V, v\in V-\{u\},\ (u,v)\notin A & (9)\\
& & x_{uv}+x_{vu} \leq 1,\ \forall\, (u,v),(v,u)\in A & (10)\\
& & \lambda\in\{0,1\}^{|V\times A^+|} & (11)
\end{array}
$$

In model $(Q2)$, (2)-(3) establish that there is a path between s and t in G and that the nodes in P are visited. Constraints (6) estate that if an arc (i, j) is in the solution (i.e. $x_{ij} = 1$), then or j precedes i or i precedes j in the abstract orientation of the referential node k. Constraints (7) limit to at most one predecessor for any node i in $G_k^+(x)$, with $i \neq k$. Constraints (8) estate that none node j can precede node k when k is the referential node. Constraints (9) fix at zero all the corresponding variables related to the extra arcs we add to make G a complete digraph. Constraints (10) avoid obtaining a cycle $C = \{(i, j), (j, i)\}$ between any pair of nodes i and j when both these arcs belong to A.

Proposition 2 states that the constraints (10) are necessary for obtaining cycle-free solutions for the problem.

Proposition 2. *Relaxing constraints* (10) *in the model* $(Q2)$ *possibly leads to the occurrence of cycles in the relaxed problem solution.*

Proof. We show an example where the optimal solution of (5)-(11), without the constraints (10), contains a cycle. Consider the digraphs in Figures 1 and 2.

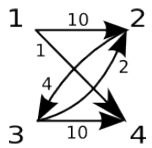

Figure 1. A digraph $G = (\{1,2,3,4\},\{(1,2),(1,4),(2,3),(3,2),(3,4)\})$**. The arc lengths are presented near each arc. We want to determine an optimal** $(1-\{2,3\}-4)$**-path of** G**.**

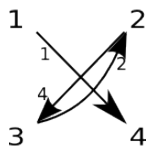

Figure 2. The optimal solution related to the digraph in Figure 1 by relaxing the constraints (10)

The optimal solution of value 7 for the resulting relaxed model (see the appendix) presents $\bar{x}_{14}$, $\bar{x}_{23}$, $\bar{x}_{32}$ and $\bar{\lambda}_{123}$, $\bar{\lambda}_{132}$, $\bar{\lambda}_{141}$, $\bar{\lambda}_{214}$, $\bar{\lambda}_{232}$, $\bar{\lambda}_{241}$, $\bar{\lambda}_{314}$, $\bar{\lambda}_{323}$, $\bar{\lambda}_{341}$, $\bar{\lambda}_{414}$, $\bar{\lambda}_{423}$, $\bar{\lambda}_{434}$, all equal to 1, with all the remaining variables being 0. Note, in this case, that the values of the λ variables do not correspond to the interpretation we give them. However, when considering (10), the optimal $(1-\{2,3\}-4)$-path solution of value 24 is $\bar{x}_{12}$, $\bar{x}_{23}$, $\bar{x}_{34}$ and $\bar{\lambda}_{121}$, $\bar{\lambda}_{132}$, $\bar{\lambda}_{143}$, $\bar{\lambda}_{212}$, $\bar{\lambda}_{232}$, $\bar{\lambda}_{243}$, $\bar{\lambda}_{312}$, $\bar{\lambda}_{323}$, $\bar{\lambda}_{343}$, $\bar{\lambda}_{412}$, $\bar{\lambda}_{423}$, $\bar{\lambda}_{434}$, all equal to 1, with all the remaining variables being 0. □

Proposition 3. *Model $(Q2)$ obtains, if it exists, a $(s-P-t)$-path of minimum cost.*

Proof. Constraints (2)-(3) establish that there is a path between s and t in G and that there are arcs visiting the nodes in P. By constraints (6), if an arc (i,j) is in the solution (i.e. $x_{ij}=1$), then at least one of the lambda variables λ_{kij} or λ_{kji} must be equal to one (i.e. indicating that there is an abstract orientation of that arc where or j precedes i or i precedes j for all referential node k). By constraints (7), any non referential node i in $G_k^+(x)$ has at most one predecessor, thus satisfying condition 2 of Proposition 1. By constraints (8), a referential node k has no predecessor, thus satisfying condition 1 of Proposition 1. By constraints (9), all non existing arcs in the original digraph G are fixed at zero. By constraints (10), we do not have a cycle of the type $C=\{(i,j),(j,i)\}$ between any pair of nodes i and j when both these arcs belong to A. Therefore, according to Proposition 1 and by the optimality condition, the solution is a $(s-P-t)$-path of minimum cost. □

Now consider the following trivial property before introducing our second compact extended formulation for the problem.

Property 1. *If $\{(s,s_1),(s_1,s_2),\cdots,(s_{p-1},s_p),(s_p,t)\}$ is a minimum length $(s-P-t)$-path of $G=(V,A)$, with $P\subseteq\{s_1,s_2,\cdots,s_p\}$ and $\pi(v)$ denotes the distance from node s to v in this path, for all $v\in\{s,s_1,s_2,\cdots,s_p,t\}$, then $\pi(s)=0$, $\pi(s_1)=c_{s,s_1}$, $\pi(s_j)=\pi(s_{j-1})+c_{s_j,s_{j-1}}$, for $j\in\{2,3,\cdots,p\}$, and $\pi(t)=\pi(s_p)+c_{s_p,t}$.*

We know that the "unrestricted" version of the minimum length $(s-P-t)$-path problem (i.e. for $P=\emptyset$) can be solved by model (1)-(2). In this case, if we associate dual variables $\pi\in\mathbb{R}^{|V|}$ with the constraints (2), then by duality theory we have that $\pi(v)-\pi(u)\le c_{uv}$, for all $(u,v)\in A$, with $\pi(s)=0$.

Proposition 4. *The dual inequalities $\pi(v)-\pi(u)\le c_{uv}$ of the classic (s,t)-shortest-path problem, for all $(u,v)\in A$, with $\pi(s)=0$, are not valid for the $(s-P-t)$-path problem.*

It is not difficult to see why these inequalities are not valid for the $(s-P-t)$-path problem (e.g. $\pi(4)-\pi(1)\le 1$ is not valid for the optimal $(1-\{2,3\}-4)$-path solution in the Figure 1). In fact, it is due to the presence of the constraints (3). However, as Property 1 must apply for the dual multipliers associated to the nodes belonging to the $(s-P-t)$-path, we need to worry only with the corresponding dual constraints related to the arcs in the solution path.

Therefore, we propose the following approach where we put together in a same model primal and some dual variables. Our idea is to characterize feasible solutions by linking primal and dual variables in a same set of constraints in order to satisfy the Property 1 and, consequently, to avoid cycles in any solution.

In the next model consider $\mathcal{M}$ a very large positive constant. The variables are the same as those defined in the above paragraphs.

$$
\begin{aligned}
(Q3) \qquad & \min_{x \in \{0,1\}^{|A|}} \sum_{uv \in A} c_{uv} x_{uv} && (12) \\
s.t. \qquad & (2) - (3) && \\
& \pi(v) - \pi(u) \leq c_{uv} + \mathcal{M}(1 - x_{uv}), \ \forall \ (u, v) \in A && (13) \\
& \pi(v) - \pi(u) \geq c_{uv} - \mathcal{M}(1 - x_{uv}), \ \forall \ (u, v) \in A && (14) \\
& \pi(s) = 0 && (15) \\
& \pi \geq \mathbf{0} && (16)
\end{aligned}
$$

Proposition 5. *If the cost vector c in (12) contains only positive entries, model $(Q3)$ gives correctly a $(s - P - t)$-path of minimum cost.*

Proof. In model $(Q3)$, constraints (2)-(3) establish that s and t are connected by a path and that the nodes in P are visited. Constraints (13) and (14) impose that if an arc (u, v) is in the solution, then Property 1 is satisfied because they became an equality constraint $\pi(v) - \pi(u) = c_{uv}$ for this arc; otherwise, both constraints became redundant. As all arc costs are positive, then no cycle can be present in a feasible solution (if it were the case, constraints (13) and (14) should be violated). The π variables correctly accumulate the distance from s to any node present in the solution. By optimality, the $(s - P - t)$-path is of minimum cost. □

The idea in model $(Q3)$ is far away implementing the well known complementary slackness conditions of linear programming. Indeed, if we take the dual problem of the model (Q), the corresponding vector of dual variables has an exponential number of elements and the corresponding constraints of the dual problem clearly does not correspond to the ones in (13) and (14).

3. Computational experiments

We use AMPL for modeling the proposed formulations. They are solved using the MIP module of IBM ILOG CPLEXAMP 12.3 (Academic Initiative). We run the instances in a PC Core 2 Duo P8600 (2.4GHz - 4G RAM) - Linux Ubuntu 10.04 LTS - Lucid Lynx. The path's origin and destination of all instances are the nodes 1 and $|V|$, respectively. These instances are random generated directed graphs with integer arc lengths randomly chosen from the interval $[1, 50]$. The set of arcs A is obtained according to a predetermined probability. The cardinality of the set P is given and its elements are chosen randomly. These last two parameters appear in each instance identifier in the next table. We adopt arbitrarily $\mathcal{M} = 10000$ in the model $(Q3)$ for all instances.

The legend in Table 3 is as follows. The first column presents the instance identifier $Inst$ composed of three parts $Prob + |V| + |P'|$: the first character indicates the probability $Prob$ used to consider or not the arcs in A (they are represented by letters **a**, **b**, **c** and **d** indicating probabilities $Prob = 0.2$, $Prob = 0.4$, $Prob = 0.7$ and $Prob = 1.0$, respectively); the second part indicates the number of nodes $|V|$ of G; and the third part indicates the number of nodes in $P \cup \{s, t\}$ (they are also indicated by letters **a**, **b**, **c** and **d** at the end of the instance identifier indicating cardinalities $|P'| = 0.25|V|$, $|P'| = 0.50|V|$, $|P'| = 0.75|V|$ and $|P'| = 1.00|V|$, respectively). The value of the continuous relaxed solution and of the optimal solution for each model are denoted by w and z, respectively. The CPU time (in seconds) to obtain the continuous relaxed solution and the optimal solution for these models are denoted by t_r and t, respectively. The total number of CPLEX MIP iterations and CPLEX branch-and-bound nodes to obtain the optimal solution for each model are denoted by $iter$ and bb, respectively.

The first element we compare in the Table 3 is the quality of the linear relaxation of the models $(Q2)$ and $(Q3)$. We reach exactly 11 optimal lower bounds (if considering integer arc costs, five other lower bounds marked with '+' can be considered optimal) in column w with the model $(Q2)$ (from a total of 48 instances), while only 5 optimal lower bounds are reached with the model $(Q3)$. In general, linear relaxed solutions obtained with the model $(Q2)$ are larger than those obtained with the model $(Q3)$. The execution times to obtain the linear relaxed solution with the model $(Q2)$ are very large when compared to the ones related to the model $(Q3)$. Observe that the lower bound obtained with the model $(Q2)$ can be considered very close to the optimal integer solution values reported in the column z. This seems to explain why CPLEX spent a high effort in solving the related integer model by the MIP approach and calling the branch-and-bound method only for few instances. The second element of our analysis concerns the quality of the optimal integer solutions. Both models reach all optimal solutions (except for the instance $a20c$ that has no feasible integer solution). The number of CPLEX MIP iterations in the column $iter$ to obtain the optimal integer solution with the model $(Q2)$ is very large when compared to the ones obtained with the model $(Q3)$ (in only one case, for the instance $b80d$, this parameter was larger for the model $(Q3)$). The CPLEX branch-and-bound method is called in 20 and 22 instances, for the models $(Q2)$ and $(Q3)$, respectively. In this occasion, the number of branch-and-bound nodes (in the column bb) in the model $(Q3)$ is larger than the related one in the model $(Q2)$ for 17 instances, being smaller only for 11 instances. The execution times to obtain the integer optimal solution with the model $(Q2)$ are much larger than those obtained with the model $(Q3)$ (except for the instance $c40b$, where the execution time is larger for the model $(Q3)$).

In our experiments there are no conclusive elements to characterize the problem difficulty in terms of the digraph density (given by the probability we use to construct the set of arcs of each instance) and the cardinality of P. If we observe the execution time t or the number of MIP iterations $iter$ of both models $(Q2)$ and $(Q3)$, there is no expressive concentration of difficult instances in any combination of these parameters.

To conclude our analysis, we observe that the good quality of the linear relaxed solution of the model $(Q2)$ is not sufficient for CPLEX saving execution time in solving these instances.

	Model $(Q2)$						Model $(Q3)$					
$Inst$	w	t_r	z	$iter$	bb	$t(s)$	w	t_r	z	$iter$	bb	$t(s)$
a20a	116.00	0.04	201	8203	62	0.98	89.06	0.00	201	2185	271	0.32
a20b	**219.00**	0.05	219	266	0	0.15	197.10	0.00	219	43	0	0.02
a20c	410.00	0.14	*	1907	0	0.54	344.19	0.00	*	7	0	0.01
a20d	384.00	0.05	409	1767	0	0.45	359.03	0.00	409	68	0	0.02
b20a	104.00	0.06	114	1246	0	1.25	99.02	0.00	114	311	19	0.45
b20b	**148.00**	0.07	148	586	0	0.26	**148.00**	0.00	148	50	0	0.01
b20c	243.33	0.16	249	3880	3	8.07	242.01	0.01	249	62	0	0.02
b20d	157.67	0.27	171	4506	0	4.81	144.01	0.02	171	1948	217	0.27
c20a	53.00	0.06	57	386	0	0.50	49.00	0.01	57	62	0	0.07
c20b	**77.00**	0.12	77	530	0	0.37	**77.00**	0.00	77	41	0	0.03
c20c	**101.00**	0.20	101	1055	0	0.67	99.00	0.00	101	69	0	0.05
c20d	**113.00**	0.30	113	1280	0	0.41	106.00	0.01	113	119	0	0.08
d20a	**25.50+**	0.08	26	334	0	0.56	25.00	0.02	26	35	0	0.06
d20b	52.42	0.22	56	2091	0	2.68	48.00	0.00	56	404	28	0.40
d20c	**66.00**	0.17	66	845	0	0.48	65.00	0.00	66	64	0	0.09
d20d	89.00	0.55	92	2992	0	7.96	86.00	0.01	92	263	13	0.37
a40a	130.00	0.91	138	75169	250	25.54	130.00	0.00	138	4633	465	3.56
a40b	**225.00**	0.82	225	2954	0	1.42	220.00	0.00	225	101	0	0.06
a40c	349.59	1.16	367	412653	437	198.89	347.00	0.01	367	22803	1689	4.35
a40d	463.00	2.98	472	38026	9	50.74	457.03	0.01	472	3135	160	0.77
b40a	95.00	0.81	98	1629	0	3.73	92.00	0.02	98	90	0	0.14
b40b	144.64	1.54	162	1131747	1482	545.56	138.01	0.01	162	13764	911	16.51
b40c	165.86	1.90	167	5738	0	7.64	158.00	0.02	167	224	0	3.04
b40d	**266.00**	3.40	266	5472	0	2.96	262.01	0.02	266	181	0	0.81
c40a	**56.17+**	1.20	57	2694	0	11.41	56.00	0.02	57	241	14	0.56
c40b	68.00	2.06	70	5544	0	15.42	68.00	0.01	70	140	0	25.48
c40c	99.25	3.03	103	18832	16	86.85	96.00	0.02	103	319	7	0.96
c40d	**128.00**	5.90	128	6358	0	6.86	**128.00**	0.02	128	129	0	0.22
d40a	43.50	1.34	48	4165	0	17.50	43.00	0.02	48	218	12	0.76
d40b	67.00	5.07	69	5031	0	45.25	67.00	0.03	69	3012	118	1.38
d40c	82.25	6.14	87	82742	81	284.93	81.00	0.04	87	13227	711	9.48
d40d	94.75	16.75	99	41530	1	335.59	93.00	0.04	99	169	0	0.34
a80a	177.00	8.57	180	18885	12	152.89	176.00	0.02	180	434	9	0.82
a80b	271.50	10.58	284	8608104	3407	20548.80	271.00	0.03	284	22516	560	54.65
a80c	376.86	26.19	379	103701	17	508.14	376.00	0.05	379	572	0	43.22
a80d	432.00	23.25	433	42330	4	329.70	421.00	0.04	433	2600	31	0.96
b80a	95.00	24.93	96	35180	18	285.33	95.00	0.04	96	192	0	41.01
b80b	161.50	31.35	163	93857	52	526.681	161.00	0.06	163	224	0	29.39
b80c	196.50	99.11	200	177107	70	976.50	192.00	0.07	200	11010	242	66.98
b80d	226.60	133.10	229	1744409	32	19180.20	226.00	0.10	229	2030586	58793	1213.70
c80a	68.00	22.39	73	2480024	2330	3494.11	65.00	0.10	73	63015	3864	92.53
c80b	**89.20+**	55.27	90	27903	0	292.86	88.00	0.14	90	252	0	0.34
c80c	118.48	66.66	120	66417	18	832.50	115.00	0.13	120	9985	0	43.44
c80d	157.00	136.60	157	18400	0	50.98	**157.00**	0.15	157	370	0	0.36
d80a	**51.00**	22.39	51	8616	0	70.09	**51.00**	0.12	51	193	0	1.14
d80b	**81.50+**	155.99	82	29669	0	376.10	80.00	0.15	82	347	0	4.13
d80c	109.14	150.91	111	4347387	715	30826.50	109.00	0.17	111	29537	657	43.84
d80d	**121.17+**	244.79	122	286899	0	5865.03	121.00	0.27	122	6109	53	20.68

$(*)$ No integer feasible solution exists for this instance.
$(+)$ The rounded lower bound $\lceil w \rceil$ is equal to the optimal solution value.

Table 1. Numerical results for the models $(Q2)$ and $(Q3)$ by using CPLEXAMP 12.3.

4. Conclusion

This work introduces two new compact extended formulations for the $(s - P - t)-$shortest-path problem in a directed graph $G = (V, A)$. The model $(Q2)$ is based on the spanning tree polytope of undirected graphs originally credited to (Martin, 1991) and presents linear relaxed solutions that are very close to optimal ones. Nevertheless, exploring this feature by the MIP module of CPLEX showed to be very time consuming. In contrast, the model $(Q3)$, although obtaining in general weaker linear relaxed solutions than the model $(Q2)$, showed to be an efficient (in terms of execution time) approach for solving this problem. We intend to perform experiments for Hamiltonian path instances from the TSPLIB to test the performance of our formulations as future work.

References

Adasme, P., Andrade, R., Letournel, M., and Lisser, A. (2013). A polynomial formulation for the stochastic maximum weight forest problem. *Electronic Notes in Discrete Mathematics*, 41(0):29 – 36.

Conforti, M., Cornuéjols, G., and Zambelli, G. (2010). Extended formulations in combinatorial optimization. *4OR - A Quarterly Journal of Operations Research*, (8):1–48.

Dreyfus, S. E. (1969). An appraisal of some shortest-path algorithms. *Operations Research*, 17(3):395–412.

Feillet, D., Dejax, P., and Gendreau, M. (2001). Traveling salesman problems with profits: An overview. *Transportation Science*, 39:188–205.

Ibaraki, T. (1973). Algorithms for obtaining shortest paths visiting specified nodes. *SIAM Review*, 15(2):pp. 309–317.

Letchford, A. N., Nasiri, S. D., and Theis, D. O. (2013). Compact formulations of the steiner traveling salesman problem and related problems. *European Journal of Operational Research*, 228(1):83 – 92.

Martin, R. K. (1991). Using separation algorithms to generate mixed integer model reformulations. *Operations Research Letters*, 10(3):119 – 128.

Saksena, J. P. and Kumar, S. (1966). The routing problem with k specified nodes. *Operations Research*, 14:538–558.

Volgenant, T. and Jonker, R. (1987). On some generalizations of the travelling-salesman problem. *The Journal of the Operational Research Society*, 38(11):1073–1079.

Yannakakis, M. (1991). Expressing combinatorial optimization problems by linear programs. *Journal of Computer and System Sciences*, 43(3):441 – 466.

Appendix: $(Q2)$ model for the digraph in Figure 1

In the following model all variables are binary. We relax the cycle elimination constraint (10) associated with the arcs $(2,3)$ and $(3,2)$. Constraints (9) to fix non existing arcs at zero are not reported.

$$\text{Minimize} \quad 10x(1,2) + x(1,4) + 4x(2,3) + 2x(3,2) + 10x(3,4) \tag{17}$$

Subject to

Flow-based conservation constraints

$$x(1,2) + x(1,4) = 1 \tag{18}$$
$$-x(1,2) + x(2,3) - x(3,2) = 0 \tag{19}$$
$$-x(2,3) + x(3,2) + x(3,4) = 0 \tag{20}$$
$$-x(1,4) - x(3,4) = -1 \tag{21}$$

Visiting constraints

$$x(1,2) + x(3,2) = 1 \tag{22}$$
$$x(2,3) = 1 \tag{23}$$

Abstract orientation of an arc

$$-x(1,2) + \lambda(1,1,2) + \lambda(1,2,1) >= 0 \tag{24}$$
$$-x(1,2) + \lambda(2,1,2) + \lambda(2,2,1) >= 0 \tag{25}$$
$$-x(1,2) + \lambda(3,1,2) + \lambda(3,2,1) >= 0 \tag{26}$$
$$-x(1,2) + \lambda(4,1,2) + \lambda(4,2,1) >= 0 \tag{27}$$
$$-x(1,4) + \lambda(1,1,4) + \lambda(1,4,1) >= 0 \tag{28}$$
$$-x(1,4) + \lambda(2,1,4) + \lambda(2,4,1) >= 0 \tag{29}$$
$$-x(1,4) + \lambda(3,1,4) + \lambda(3,4,1) >= 0 \tag{30}$$
$$-x(1,4) + \lambda(4,1,4) + \lambda(4,4,1) >= 0 \tag{31}$$
$$-x(2,3) + \lambda(1,2,3) + \lambda(1,3,2) >= 0 \tag{32}$$
$$-x(2,3) + \lambda(2,2,3) + \lambda(2,3,2) >= 0 \tag{33}$$
$$-x(2,3) + \lambda(3,2,3) + \lambda(3,3,2) >= 0 \tag{34}$$
$$-x(2,3) + \lambda(4,2,3) + \lambda(4,3,2) >= 0 \tag{35}$$
$$-x(3,2) + \lambda(1,3,2) + \lambda(1,2,3) >= 0 \tag{36}$$
$$-x(3,2) + \lambda(2,3,2) + \lambda(2,2,3) >= 0 \tag{37}$$
$$-x(3,2) + \lambda(3,3,2) + \lambda(3,2,3) >= 0 \tag{38}$$
$$-x(3,2) + \lambda(4,3,2) + \lambda(4,2,3) >= 0 \tag{39}$$
$$-x(3,4) + \lambda(1,3,4) + \lambda(1,4,3) >= 0 \tag{40}$$
$$-x(3,4) + \lambda(2,3,4) + \lambda(2,4,3) >= 0 \tag{41}$$
$$-x(3,4) + \lambda(3,3,4) + \lambda(3,4,3) >= 0 \tag{42}$$
$$-x(3,4) + \lambda(4,3,4) + \lambda(4,4,3) >= 0 \tag{43}$$

Referential nodes have no predecessor

$$\lambda(1,1,2) = 0 \tag{44}$$
$$\lambda(2,2,1) = 0 \tag{45}$$
$$\lambda(1,1,4) = 0 \tag{46}$$
$$\lambda(4,4,1) = 0 \tag{47}$$
$$\lambda(2,2,3) = 0 \tag{48}$$
$$\lambda(3,3,2) = 0 \tag{49}$$
$$\lambda(2,2,3) = 0 \tag{50}$$
$$\lambda(3,3,2) = 0 \tag{51}$$
$$\lambda(3,3,4) = 0 \tag{52}$$
$$\lambda(4,4,3) = 0 \tag{53}$$

For a referential node, any other node have at most one predecessor

$$\lambda(2,1,2) + \lambda(2,1,4) <= 1 \tag{54}$$
$$\lambda(3,1,2) + \lambda(3,1,4) <= 1 \tag{55}$$
$$\lambda(4,1,2) + \lambda(4,1,4) <= 1 \tag{56}$$
$$\lambda(1,2,1) + \lambda(1,2,3) <= 1 \tag{57}$$
$$\lambda(3,2,1) + \lambda(3,2,3) <= 1 \tag{58}$$
$$\lambda(4,2,1) + \lambda(4,2,3) <= 1 \tag{59}$$
$$\lambda(1,3,2) + \lambda(1,3,4) <= 1 \tag{60}$$
$$\lambda(2,3,2) + \lambda(2,3,4) <= 1 \tag{61}$$
$$\lambda(4,3,2) + \lambda(4,3,4) <= 1 \tag{62}$$
$$\lambda(1,4,1) + \lambda(1,4,3) <= 1 \tag{63}$$
$$\lambda(2,4,1) + \lambda(2,4,3) <= 1 \tag{64}$$
$$\lambda(3,4,1) + \lambda(3,4,3) <= 1 \tag{65}$$