The Online Replacement Path Problem

David Adjiashvili[1], Gianpaolo Oriolo[2], and Marco Senatore[2]

[1] Institute for Operations Research (IFOR)
Eidgenössische Technische Hochschule (ETH) Zürich
Rämistrasse 101, 8092 Zürich, Switzerland
david.adjiashvili@ifor.math.ethz.ch

[2] Dipartimento di Ingegneria Civile ed Ingegneria Informatica
Università di Roma Tor Vergata
Via del Politecnico 1, 00133 Rome, Italy
{oriolo,senatore}@disp.uniroma2.it

Abstract. We study a new robust path problem, the *Online Replacement Path problem (ORP)*. Consider the problem of routing a physical package through a faulty network $G = (V, E)$ from a source $s \in V$ to a destination $t \in V$ as quickly as possible. An adversary, whose objective is to maximize the latter routing time, can choose to remove a single edge in the network. In one setup, the identity of the edge is revealed to the routing mechanism (RM) while the package is in s. In this setup the best strategy is to route the package along the shortest path in the remaining network. The payoff maximization problem for the adversary becomes the *Most Vital Arc problem (MVA)*, which amounts to choosing the edge in the network whose removal results in a maximal increase of the s-t distance. However, the assumption that the RM is informed about the failed edge when standing at s is unrealistic in many applications, in which failures occur *online*, and, in particular, after the routing has started. We therefore consider the setup in which the adversary can reveal the identity of the failed edge just before the RM attempts to use this edge, thus forcing it to use a different route to t, starting from the current node. The problem of choosing the nominal path minimizing the worst case arrival time at t in this setup is ORP. We show that ORP can be solved in polynomial time and study other models naturally providing middle grounds between MVA and ORP. Our results show that ORP comprises a highly flexible and tractable framework for dealing with robustness issues in the design of RM-s.

1 Introduction

Modeling the effects of limited reliability of networks in modern routing schemes is important in many applications. It is often unrealistic to assume that the nominal network known at the stage of decision making will be available in its entirety at the stage of solution implementation. Several research directions have emerged as a result. The main paradigm in most works is to obtain a certain 'fault-tolerant' or 'redundant' solution, which takes into account a certain set of likely network realizations at the implementation phase.

H.L. Bodlaender and G.F. Italiano (Eds.): ESA 2013, LNCS 8125, pp. 1–12, 2013.

Shortest paths are often used in order to minimize routing time. In faulty network, however, simply taking the shortest path might lead to very large delays due to link failures. Two related problems that were extensively studied in the literature are the *Most Vital Arc problem (MVA)* and the *Replacement Path problem (RP)*. MVA asks given a graph $G = (V, E)$ and two nodes $s, t \in V$ to find the edge $e \in E$ whose removal results in the maximal increase in the s-t distance in G. The input to RP additionally includes a shortest path P, and the goal is to find for every $e \in P$ a shortest s-t path P_e avoiding e. In the context of robust network design both MVA and RP should be interpreted as problems in which the RM is informed about the failed edge *in advance*, namely when standing at s. This assumption is unrealistic in many situations, in which failures occur *online*, and in particular, after the routing has started. Examples of such situations range from accidents and traffic jams in road network to truly adversarial setups, in which the adversary is motivated to conceal the failure for as long as possible.

In this paper we study the *Online Replacement Path problem (ORP)*, which is motivated by such situations. We delay the formal definition of the problem to Section 3, and instead give an intuitive description. The basic assumption in ORP is that the materialized scenario is revealed to the RM 'in the last minute', namely only when the package reaches one of the endpoints of the failed edge and attempts to cross it. From this point on the package is routed through a *detour*, namely a path from the current node to the destination that avoids the failed edge. The *robust length* of a path is the maximum total travel time over all possible failure scenarios, and the goal is to find a path with minimum robust length.

ORP models online failure scenarios that occur in many situations, some of which we described before. In other applications it is only necessary to route a certain object within a certain time, called a *deadline*. As long as the object reaches its destination before the deadline, no penalty is incurred. On the other hand, if the deadline is not met, a large penalty is due. An example of such an application is organ transportation for transplants (see e.g. Moreno, Valls and Ribes [14]), in which it is critical to deliver a certain organ before the scheduled time for the surgery. In this application it does not matter how early the organ arrives at the destination, as long as it arrives in time. In such applications it is often too risky to take an unreliable shortest path, which admits only long detours in some scenarios, whereas a slightly longer path with reasonably short detours meets the deadline in *every* scenario. Thus, this domain of applications can also benefit from ORP.

Our first result is a polynomial algorithm for ORP. Concretely, we show that optimal u-t path can be found in time $O(m + n \log n)$ in undirected graphs and $O(nm + n^2 \log n)$ in directed graphs for all sources $u \in V$ and a single destination t. We prove various properties of ORP on the way to the aforementioned algorithms. In particular, we show the existence of a tree of optimal paths, and that the robust length is monotonic with respect to taking subpaths of optimal paths. These properties lead to a natural label-setting algorithm.

In Section 4 we study *Bi-objective ORP*, the optimization problem of finding a shortest path in the graph with robust length at most a given bound B. We show that this problem admits an algorithm with running time $O(m+n\log n)$ in undirected graphs (and $O(mn+n^2\log n)$ in directed graphs). We also show that the Pareto front of the latter bi-objective problem has linear size in the size of the graph, and provide a simple algorithm to compute it in time $O(m^2+mn\log n)$, for both directed and undirected graphs. This is of course extremely nice in practical applications, as the decision maker can efficiently plot the tradeoff between the *nominal* and the *robust* length of Pareto-efficient solutions.

In Section 5 and Section 6 we study two models that provide a middle ground between MVA and ORP. In Section 5 we study the *k-Hop ORP* problem. The RM is now informed about the failed edge e as soon as it reaches a node that is k hops away from e on the nominal path. While 0-Hop ORP is simply ORP, one easily sees that $(n-1)$-Hop ORP is equivalent to MVA. For $k \in \{1, \cdots, n-2\}$ we obtain an interesting continuum of problems between ORP and MVA. We show that some of the nice properties that hold for ORP no longer hold for k-Hop ORP. In particular, while a tree of optimal paths always exists, the robust length of a subpath in this tree can be larger than the robust length of the original path. Nevertheless, we obtain a label-setting algorithm for this problem, whose running time is identical to that of our algorithms for ORP for both the directed and the undirected case (and so is independent from k). That is very interesting because, to the contrary, this is not the case with the variant where the RM is informed about the failed edge e as soon as it reaches a node that is k hops away from e in *the graph*, and not just on *the nominal path*. While this variant is equivalent to ORP for $k = 0$, we show that, already with $k = 1$, it is NP-hard to approximate within a factor of $3 - \epsilon$ for undirected graphs (and provide a simple algorithm meeting this factor), and it is strongly NP-hard to decide if there exists a nominal path with finite robust length for directed graphs.

Finally, in Section 6 we study the *ORP Game*, a two players' game related to MVA and ORP. In this game a first player, the *path builder*, is interested in arriving from s to t as quickly as possible. The second player, the *interdictor*, tries to make the latter distance as long as possible by removing a single edge from the graph. The *strategies* for the two players are the s-t paths, and the edges $e \in E$, respectively. One can see ORP and MVA as variants of the ORP Game, in which strategies are not communicated simultaneously. We show that the instances of the game which admit a pure Nash Equilibrium (NE) are exactly those where the values of the optimal solutions to ORP and MVA are equal, and build upon this fact to give an $O(m + n\log n)$-time algorithm that finds it in undirected graphs (and in time $O(mn + n^2\log n)$-time in directed graphs), or reports that no pure NE exists.

Finally, we developed a poly-time algorithm for the generalization of ORP when some fixed number of edges can fail. However, even if more involved, the algorithm goes along the same lines of that for the single-failure case, so we just defer the details to the journal version of the paper, due to space considerations.

In the next section we review related work.

2 Related Work

The Replacement Path (RP) problem was proposed by Nisan and Ronen [20] in order to study a problem in auction theory, namely that of computing *Vickrey prices.* RP is also used as a subroutine for computing the k shortest paths in a graph. The complexity of the RP problem for undirected graphs is well understood. The first paper to study this problem is due to Malik, Mittal and Gupta [13], who give a simple $O(m+n\log n)$ algorithm. A mistake in this paper was later corrected by Bar-Noy, Khuller and Schieber [4]. As a bi-product, the latter result implies an $O(m + n\log n)$-time algorithm for the Most Vital Arc (MVA) problem. This running time is asymptotically the same as a single source shortest path computation. Nardelli, Proietti and Widmayer [18] later extended the result to account for node failures. In [15] the same authors gave an algorithm that finds a detour-critical edge on a shortest path. The complexity for MVA was later improved by Nardelli, Proietti and Widmayer [17] to $O(m\alpha(m, n))$, where $\alpha(\cdot, \cdot)$ is the Inverse Ackermann function. The only general nontrivial algorithm for RP in directed graphs is due to Gotthilf and Lewenstein [10] that gave an $O(mn + n^2 \log\log n)$-time algorithm. Faster algorithms for unweighted graphs (Roditty and Zwick [23]) and planar graphs (Emek, Peleg and Roditty [8], Klein, Mozes and Weimann [12] and Wulff-Nilsen [26]) were developed. The problem of approximating replacement paths was considered by Roditty [22] and Bernstein [6]. Results for bounded edge lengths were given by various authors. We refer to the paper of Vassilevska Williams [25] and references therein for details.

Some related work was carried out in the context of *routing policies* (Papadimitriou and Yannakakis [21]), the most prominent example being the Canadian Traveler Problem (Bar-Noy and Schieber [5], Nikolova and Karger [19]). In particular, in [5] the authors consider a problem that can be seen as a policy-based variant of the problem we study in Section 3. They first claim, without proof, that their problem reduces indeed to the latter one, and then claim some results that are close to those we present in Section 3. However, as we discuss later, we believe that these results are not adequately supported in [5] by rigorous arguments.

Another problem which bears resemblance to ORP is the *Stochastic Shortest Path with Recourse problem* (SSPR), studied by Andreatta and Romeo [3]. This problem can be seen as the stochastic analogue of ORP. Finally, we briefly review some related work on robust counterparts of the shortest path problem. The shortest path problem with cost uncertainty was studied by Yu and Yang [27], who consider several models for the scenario set. These results were later extended by Aissi, Bazgan and Vanderpooten [2]. These works also considered a two-stage min-max regret criterion. Dhamdhere, Goyal, Ravi and Singh [7] developed the demand-robust model and gave an approximation algorithm for the shortest path problem. A two-stage feasibility counterpart of the shortest path problem was addressed by Adjiashvili and Zenklusen [1].

3 An Algorithm for ORP

In this section we develop an algorithm for ORP. Let us establish some notation first. We are given an edge-weighted graph $G = (V, E, \ell)$, a source $s \in V$ and destination $t \in V$, and we always assume that the edge weights ℓ are nonnegative. *Unless otherwise specified, we assume that G is indifferently directed or undirected.* Let n and m be the number of nodes and edges of the input graph, respectively. For two nodes $u, v \in V$ let $\mathcal{P}_{u,v}$ be the set of simple u-v paths in G. Let $N(u)$ be the set of neighbors of u in G. For a set of edges $A \subset E$ let $\ell(A) = \sum_{e \in A} \ell(e)$. For an edge $e \in E$ and a set of edges $F \subseteq E$, let $G - e$ and $G - F$ be the graph obtained by removing the edge e and the edges in F, respectively. For a set of edges $A \subset E$ let $V(A)$ be the set of nodes incident to edges in A. Paths are always represented as sets of edges, while walks are represented as sequences of nodes. For two walks Q_1, Q_2 with the property that last node of Q_1 is the first node of Q_2 we let $Q_1 \oplus Q_2$ be their concatenation. For a path P containing nodes u and v let $P[u, v]$ be the subpath of P from u to v. For an edge $e \in E$ and $u \in V$ let Q_u^{-e} be some fixed shortest u-t path in $G - e$ and let $\pi_u^{-e} = \ell(Q_u^{-e})$. We use the convention that $Q_u^{-e} = \emptyset$ and $\pi_u^{-e} = \infty$ if u and t are in different connected components in $G - e$.

It is convenient to define the *detour* P^{-e} of a path $P \in \mathcal{P}_{v,t}$ and an edge $e = uu' \in E$ to be the walk $P[v, u] \oplus Q_u^{-uu'}$ if $uu' \in P$ (where u is the node closer to v on P), and P, otherwise. Note that we have $\ell(P^{-e}) = \ell(P[v, u]) + \pi_u^{-uu'}$ and $\ell(P^{-e}) = \ell(P)$ in the former and the latter case, respectively.

Definition 1. *Given a node $v \in V$, the robust length of the v-t path P is*

$$\mathrm{Val}(P) = \max_{e \in E} \ell(P^{-e}).$$

ORP is to find for every $v \in V$ an optimal nominal path, namely a path P minimizing $\mathrm{Val}(P)$ *over all paths $P \in \mathcal{P}_{v,t}$.*

Our algorithm uses a label-setting approach, analogous to reverse Dijkstra's algorithm for shortest paths. In every iteration, the algorithm updates certain tentative labels for the nodes of the graph, and fixes a final label to a single node u. This final label represents the connection cost of u by an optimal path to t. For $v \in V$ we define the *potential* $y(v)$ as the minimum of $\mathrm{Val}(P)$ over all $P \in \mathcal{P}_{v,t}$. The robust length of a v-t path P is simply the maximal possible cost incurred by following P until a certain node, and then taking the best possible detour from that node to t which avoids the next edge on the path. To avoid confusion, we stress that in ORP we assume the existence of at most one failed edge in the graph. Consider next a scenario in which an edge $uu' \in P$ fails and let $u \in V$ be the node which is closer to v. Clearly, the best detour is a shortest u-t path in the graph $G - uu'$. Note that both Val(P) and $y(v)$ can attain the value ∞ in case that the path P admits no detours in some scenario, and in case all v-t paths are of this sort, respectively. Furthermore, nonnegativity of ℓ implies $\mathrm{Val}(P) \geq \mathrm{Val}(P[v, t])$, whenever $v \in V(P)$. We can prove the following useful:

Lemma 1. *Let $P_u \in \mathcal{P}_{u,t}$ and let $v \in N(u)$ be a node, not incident to P_u. Then* $\mathrm{Val}(vu \oplus P_u) = \max\{\ell(vu) + \mathrm{Val}(P_u), \pi_v^{-vu}\}$.

Our algorithm for ORP updates the potential on the nodes of the graph, using the property established by the following lemma.

Lemma 2. *Let $U \subset V$, with $t \in U$, be the set of nodes for which the potential is known, and let vu be an edge such that:*

$$vu \in \operatorname*{arg\,min}_{zw \in E : w \in U, z \notin U} \max\{\ell(zw) + y(w), \pi_z^{-zw}\}. \tag{1}$$

Then $\mathrm{Val}(vu \oplus P_u) = y(v)$ *for any optimal nominal u-t path P_u.*

Lemma 2 provides the required equation for our label-setting algorithm, whose statement is given as Algorithm 1. The algorithm iteratively builds up a set U, consisting of all nodes, for which the correct potential value was already computed. The correctness of the algorithm is a direct consequence of Lemma 2.

Algorithm 1.

Compute π_u^{-uv} for each $uv \in E$.
$U = \emptyset$; $W = V$; $y'(t) = 0$; $y'(u) = \infty \ \forall u \in V - t$.
$successor(u) = \mathrm{NIL} \ \forall u \in V$.
while $U \neq V$ **do**
  Find $u = \arg\min_{z \in W} y'(z)$.
  $U = U + u$; $W = W - u$; $y(u) = y'(u)$.
  for all $vu \in E$ with $v \in W$ **do**
    if $y'(v) > \max\{\ell(vu) + y(u), \pi_v^{-vu}\}$ **then**
      $y'(v) = \max\{\ell(vu) + y(u), \pi_v^{-vu}\}$.
      $successor(v) = u$.

We are ready to state the main result of this section. The running time is obtained by a careful implementation that is presented in the proof. Our implementation relies on an algorithm of Nardelli, Proietti and Widmayer [16] for computing swap edges in graphs with respect to a shortest path tree. We note that Bar-Noy and Schieber [5] sketch a similar algorithm with the same time bound for a problem that can be seen as a policy-based variant of ORP. Their result are stated without proof, and we are not aware of a proof that does not build on the result of Nardelli, Proietti and Widmayer [16], which appeared afterwards.

Theorem 1. *Given an instance of ORP the potential y and the corresponding paths can be computed in time $O(m + n \log n)$ in undirected graphs, and $O(mn + n^2 \log n)$ in directed graphs.*

We end this section with a simple graph-theoretical characterization of paths with finite robust value. Let $U^2 \subset V$ be the set of nodes in G that are 2-edge connected to t, i.e., all nodes u such that there are two edge-disjoint u-t paths in G. Note that, following Theorem 2, if s and t are in different components of $G[U^2]$, there will be no path with finite robust value (and of course that happens if and only if Algorithm 1 returns $y(s) = \infty$).

Theorem 2. *A path $P \in \mathcal{P}_{s,t}$ has finite robust value if and only if $V(P) \subset U^2$.*

4 Bi-objective ORP

We turn to a natural question linking ORP and the Shortest Path problem. Consider an instance of s-t ORP for which the optimal nominal path is not unique. While all optimal paths P have the same robust length, they might differ in terms of their ordinary length $\ell(P)$. We can thus be interested in obtaining a path attaining the potential with minimum length. In general, one can consider the following bi-objective problem for any bound $B \geq y(s)$.

$$z(s, B) = \min_{P \in \mathcal{P}_{s,t},\ \mathrm{Val}(P) \leq B} \ell(P).$$

The latter problem asks to find a *Pareto-optimal* s-t path in G with objective functions robust length and ordinary length. We call this problem *Bi-objective ORP*. Bi-objective ORP bears resemblance to the Bi-objective Shortest Path problem [11]. In the latter problem one seeks to obtain a Pareto-optimal s-t path in the graph with objective functions corresponding to ordinary length with respect to two different length functions. In this section we show that the two problems differ significantly in terms of their complexity. Concretely, we will show that a solution to bi-objective ORP and the entire Pareto front can be found in polynomial time. This contrasts to the Bi-objective Shortest Path problem, which is NP-hard, and its Pareto front can be of exponential size in the size of the graph. Our first result is:

Theorem 3. *Bi-objective ORP can be solved in time $O(m + n \log n)$ in undirected graphs and in time $O(mn + n^2 \log n)$ in directed graphs.*

Theorem 3 builds upon an algorithm that is similar to Algorithm 1. Let us turn to the problem of computing the Pareto front. Recall that a *Pareto front* F of a bi-objective optimization problem with objective functions f and g is a set of Pareto-optimal solutions to the problem with the property that, for every other solution X, there exists a solution $Y \in$ F such that $f(X) \geq f(Y)$ and $g(X) \geq g(Y)$. A Pareto front F for an instance of Bi-objective ORP is a set of paths, such that, for every Pareto-optimal path P, there exists a path in F with not longer robust length and not longer ordinary length. In general, having an entire Pareto front at hand is of course advantageous in practical applications, as it gives the decision maker a complete list of efficient strategies. The following theorem asserts that every Bi-objective ORP instance has a Pareto front with a linear number of paths. The front can be found in polynomial time using Algorithm 2 (note that, for undirected graphs, the algorithm is slightly different).

Algorithm 2.

1: $H = G$; $\mathrm{F} = \emptyset$.
2: **while** s and t are connected in H **do**
3: Find a shortest s-t path P in H and add it to F.
4: Find a critical edge $e \in E(H)$ (with $\mathrm{Val}(P) = \ell(P^{-e})$) and remove it from H.
5: Remove from F all dominated paths.
6: Return F.

Theorem 4. *Every instance of Bi-objective ORP admits a Pareto front* F *with at most* $2m$ *paths (*m *paths in directed graphs). The Pareto front can be found in time* $O(m^2 + mn \log n)$.

Finally, observe that Algorithm 2 is also an algorithm for ORP, as for any Pareto front F we have $y(s) = \min_{P \in \mathrm{F}} \mathrm{Val}(P)$. This algorithm can be particularly interesting for solving ORP in sparse directed graphs, where the size of the Pareto front might compare favorably with the number of nodes in the graph.

5 *k*-Hop ORP

In this section we study the k-Hop ORP problem. We assume that we are given an integer k between 0 and $n-1$ and that now the RM is informed about the failure of edge e as soon as it reaches a node that is k (or fewer) hops away from e on the nominal path P. In particular, if $e \notin P$, the RM won't be aware of the failure of e. It is easy to see that 0-Hop ORP is simply ORP and that $(n-1)$-Hop ORP is equivalent to MVA. For $k \in \{1, \cdots, n-2\}$ we obtain an interesting continuum of problems between ORP and MVA.

Apparently this new setting changes dramatically the problem. In fact, consider a (nominal) s-t path P and an edge e which belongs to P. Denote by $v(P, e)$ the first node of P that 'sees' the failure of e (note that $v(P, e) = s$ if e is at most k-hop away from s on P). Being aware of the failure of e already at $v(P, e)$ allows the RM to take a detour before getting to e, as for ORP. This justifies the following redefenition of detours. The detour P^{-e} associated with an edge $e \in E$ is defined as $P[s, v(P, e)] \oplus Q^{-e}_{v(P,e)}$ if $e \in P$, and P if $e \notin P$.

The k-Hop ORP problem is defined as finding for every $u \in V - t$ the *k-Hop potential*

$$y^k(u) = \min_{P \in \mathcal{P}_{u,t}} \mathrm{Val}^k(P),$$

where $\mathrm{Val}^k(P) = \max_{e \in E} \ell(P^{-e})$, as well as a corresponding path.

Our main result in this section is a label-setting algorithm for this problem. Obtaining this algorithm is however a more challenging task than that of obtaining such an algorithm for ORP. In particular, we will see that while there always exists a tree of optimal nominal paths, the k-hop potential needs not be a monotonic function along the paths of this tree. This property contrasts with the structure of optimal solutions to ORP. The key property of k-Hop ORP is stated in the following lemma.

Lemma 3. *Let $P_u \in \mathcal{P}_{u,t}$ be an optimal path from u, let $v \in V(P_u)$ and let $P_v \in \mathcal{P}_{v,t}$ be an optimal path from v. Then the path $P'_u = P_u[u,v] \oplus P_v$ satisfies $\mathrm{Val}^k(P'_u) = \mathrm{Val}^k(P_u)$, namely it is also optimal from u.*

Lemma 3 and the property that we state hereafter allow us to prove the correctness of a label-setting algorithm. The property follows from the fact that for any $u \in V$ and $e \in E$ one has $y^k(u) \geq \pi_u^{-e}$.

Property 1. Let $u \in V$ and $P \in \mathcal{P}_{u,t}$ be such that $\mathrm{Val}^k(P) = \pi_u^{-e}$ for some edge e seen on P by u. Then P is an optimal path from u.

Analogously to our algorithm for ORP, we proceed by incrementally computing the optimal path for every node in the graph starting from t. We maintain a set U of nodes for which a robust path was already computed. For $u \in U$ we denote this path by P_u^*. The update rule for U works as follows. First, we check if for some edge $vu \in E$ such that $v \in V \setminus U$ and $u \in U$ it holds that the path $Q = vu \oplus P_u^*$ satisfies the condition in Property 1. In other words, we check if $\mathrm{Val}^k(Q) = \pi_v^{-e}$ for some edge $e \in Q$ seen by v. If such an edge exists we set $P_v^* := Q$ and $U := U \cup \{v\}$, an update that is valid due to Property 1. Assume next that no such edge exists. We call the set U in this situation *clean*. The following lemma states an update rule for clean sets U.

Lemma 4. *Let U be clean and let $vu \in \arg\min_{qr \in E\,:\, r \in U, q \notin U} \mathrm{Val}^k(qr \oplus P_r^*)$. Then $y^k(v) = \mathrm{Val}^k(vu \oplus P_u^*)$.*

Lemmas 3 and 4 immediately imply a polynomial algorithm for k-Hop ORP. Observe that one can adopt the implementation of our label-setting algorithm for ORP to obtain the same time bounds as in Theorem 1. The details are identical, and thus omitted.

Theorem 5. *Given an instance of k-Hop ORP the potential y^k and the corresponding paths can be computed in time $O(m + n \log n)$ in undirected graphs, and $O(mn + n^2 \log n)$ in directed graphs.*

Let us make two further remarks about extensions of ORP similar to k-Hop ORP. In k-Hop ORP we assume that the RM is informed about the failed edge when it is k hops away on the nominal path. An alternative definition takes the lengths of edges on this path into account. In this problem, which we call *Radius ORP*, the integer $k \leq n-1$ is replaced by a value $R \leq \ell(E)$ called the *radius*. In this problem the RM is informed about the failed edge e on the nominal path P at the first node that is at distance at most R from its closer endpoint. The definition of $v(P, e)$ and the robust value are adapted accordingly.

We claim without proof that our algorithm for k-Hop ORP solves Radius ORP as well. Informally, this follows from fact that Lemma 3 remains correct, since it only relies on the following *monotonicity property*. The set of edges on P that a node sees is an interval on this path, and furthermore, for every two consecutive nodes $u_1, u_2 \in V(P)$, with u_2 being the closer one to t, the set of edges seen by u_1 in $P[u_2, t]$ is a subset of the set of edges seen by u_2. We defer the proof of this fact, as well as the careful treatment of Radius ORP to the journal version of the paper, due to space considerations.

We end this section with another variant of k-Hop ORP. In this variant, whose input is identical to that of k-Hop ORP, the information about the failed edge travels through the edges of the entire graph, as opposed to only the edges of the nominal path. Formally, the first node along the chosen nominal path that is informed about the failure of some edge $e \in E$ is the one closest to s that is at most k hops away from e in G. This problem, which we denote by *Strong k-Hop ORP* turns out to be NP-hard to approximate even when $k = 1$. Note that, for $k = 0$, Strong k-Hop ORP reduces to ORP, as for every path the robust value is the same in the two different problems.

Theorem 6. *for any $\epsilon > 0$ it is NP-hard to approximate Strong 1-Hop ORP within a factor of $3 - \epsilon$ in undirected graphs. In directed graphs it is strongly NP-hard to decide if there exists a nominal path with finite robust length.*

We note that in 2 s-t connected undirected graphs, every shortest path is a 3-approximation of the optimal solution to Strong 1-Hop ORP, thus the approximability of this problem is settled. The proof of this simple fact if similar to the proof of Lemma 6, and thus omitted.

6 A Two Players' Game between MVA and ORP

Let us explore next a two players' game that is the natural middle ground between the problems MVA and ORP. A first player, the *path builder*, is interested in arriving from s to t as quickly as possible. The second player, the *interdictor*, tries to make the latter distance as long as possible by removing a single edge from the graph. The *strategies* for the two players are the s-t paths, and the edges $e \in E$, respectively.

In one setup, the interdictor communicates her strategy *first*, i.e. which edge is removed from G. The path builder chooses his strategy *after*: clearly he chooses a shortest path s-t in the graph $G - e$. Therefore, the problem that the interdictor faces in this setting is clearly the MVA problem, as she will remove the edge $e \in E$ maximizing π_s^{-e}, the length of the shortest s-t path in the graph $G - e$. In the following, we let $z^*(MVA)$ be the value of an optimal solution to MVA.

In the other extreme, the path builder communicates his strategy first, i.e. an s-t path P. Then the interdictor moves, and clearly removes the edge e maximizing $\ell(P^{-e})$. Note that we assume that, if $e \in P$, the interdictor will delay the failure of the edge to the point at which the path builder attempts to cross it. Hence, the problem that the path builder faces is exactly ORP, i.e. that of choosing an s-t path with the least robust value. In the following, we let $z^*(ORP)$ be the value of an optimal solution to ORP.

The next lemma, whose simple proof we skip, shows that $z^*(ORP) \geq z^*(MVA)$.

Lemma 5. *Let P and e be an s-t path and an edge of E, respectively. Then* $\mathrm{Val}(P) \geq z^*(ORP) \geq z^*(MVA) \geq \pi_s^{-e}$.

In our two players' game, that we call the *ORP Game*, both players communicate their strategies *at the same time*. In particular, for a given s-t path P and edge $e \in E$, the payoff for the interdictor is $\ell(P^{-e})$. Lemma 5 shows that in general $z^*(ORP) \geq z^*(MVA)$. The next theorem characterizes the instances of the ORP Game admitting a pure NE as those for which $z^*(ORP) = z^*(MVA)$.

Theorem 7. *Let P and e be optimal solutions to the ORP and MVA instances on $G = (V, E)$. Then (P, e) is a pure NE of the ORP Game if and only if* $\text{Val}(P) = \pi_s^{-e}$. *Moreover, in this case,* $\text{Val}(P) = z^*(ORP) = z^*(MVA) = \pi_s^{-e}$.

Theorem 7 has also the following algorithmic implication. Recall that we can compute $z^*(MVA)$ in time $O(m + n \log n)$ [13], the same running time we obtained for unidrected ORP (Theorem 1). This clearly implies that in time $O(m + n \log n)$ we can compute a pure NE of the ORP Game in undirected graphs, if one exists, or certify that no pure NE exists. Indeed the aforementioned algorithms allow us to check the condition $z^*(ORP) = z^*(MVA)$ and compute corresponding optimal solutions, P^* and e^*, with the latter time complexity. Theorem 7 asserts that if the latter condition is satisfied, then (P^*, e^*) is a pure NE, otherwise no pure NE exists. Theorems 1 and 7 also imply a $O(mn + n^2 \log n)$ algorithm for the same problem is directed graphs.

We close this section by shortly addressing the case where $z^*(ORP) \neq z^*(MVA)$. First, in this case, Theorem 7 shows that there are no pure NE. However, the ORP Game will still admit a NE in *mixed strategies*, as for both players the sets of pure strategies is finite (s-t paths and edges). Whether it is possible to find this mixed NE in polynomial time is an interesting open question.

We conclude by analyzing the ratio $\frac{z^*(ORP)}{z^*(MVA)}$. The next lemma shows that, for undirected graphs, it is at most 3.

Lemma 6. *Let G be undirected with s-t edge-connectivity of at least two. Then* $z^*(ORP) \leq 3z^*(MVA)$.

References

1. Adjiashvili, D., Zenklusen, R.: An s - t connection problem with adaptability. Discrete Applied Mathematics 159(8), 695–705 (2011)
2. Aissi, H., Bazgan, C., Vanderpooten, D.: Approximation complexity of min-max (Regret) versions of shortest path, spanning tree, and knapsack. In: Brodal, G.S., Leonardi, S. (eds.) ESA 2005. LNCS, vol. 3669, pp. 862–873. Springer, Heidelberg (2005)
3. Andreatta, G., Romeo, L.: Stochastic shortest paths with recourse. Networks 18(3), 193–204 (1988)
4. Bar-Noy, A., Khuller, S., Schieber, B.: The complexity of finding most vital arcs and nodes. Technical report, Univ. of Maryland Institute for Advanced Computer Studies Report No. UMIACS-TR-95-96, College Park, MD, USA (1995)
5. Bar-Noy, A., Schieber, B.: The canadian traveller problem. In: SODA 1991, pp. 261–270. SIAM, Philadelphia (1991)

6. Bernstein, A.: A nearly optimal algorithm for approximating replacement paths and k shortest simple paths in general graphs. In: SODA 2010, pp. 742–755. SIAM, Philadelphia (2010)
7. Dhamdhere, K., Goyal, V., Ravi, R., Singh, M.: How to pay, come what may: Approximation algorithms for demand-robust covering problems. In: FOCS 2005, pp. 367–378. IEEE Computer Society, Washington, DC (2005)
8. Emek, Y., Peleg, D., Roditty, L.: A near-linear-time algorithm for computing replacement paths in planar directed graphs. ACM Trans. Algorithms 6, 64:1–64:13 (2010)
9. Fredman, M.L., Tarjan, R.E.: Fibonacci heaps and their uses in improved network optimization algorithms. J. ACM 34, 596–615 (1987)
10. Gotthilf, Z., Lewenstein, M.: Improved algorithms for the k simple shortest paths and the replacement paths problems. Inf. Process. Lett. 109, 352–355 (2009)
11. Hassin, R.: Approximation schemes for the restricted shortest path problem. Mathematics of Operations Research 17(1), 36–42 (1992)
12. Klein, P.N., Mozes, S., Weimann, O.: Shortest paths in directed planar graphs with negative lengths: A linear-space $O(n \log^2 n)$-time algorithm. ACM Trans. Algorithms 6, 30:1–30:18 (2010)
13. Malik, K., Mittal, A.K., Gupta, S.K.: The k most vital arcs in the shortest path problem. Operations Research Letters 8(4), 223–227 (1989)
14. Moreno, A., Valls, A., Ribes, A.: Finding efficient organ transport routes using multi-agent systems. In: Proceedings of the IEEE 3rd International Workshop on Enterprise Networking and Computing in Health Care Industry (Healthcom), pp. 233–258 (2001)
15. Nardelli, E., Proietti, G., Widmayer, P.: Finding the detour-critical edge of a shortest path between two nodes. Information Processing Letters 67(1), 51–54 (1998)
16. Nardelli, E., Proietti, G., Widmayer, P.: Swapping a failing edge of a single source shortest paths tree is good and fast. Algorithmica 35, 2003 (1999)
17. Nardelli, E., Proietti, G., Widmayer, P.: A faster computation of the most vital edge of a shortest path. Information Processing Letters 79(2), 81–85 (2001)
18. Nardelli, E., Proietti, G., Widmayer, P.: Finding the most vital node of a shortest path. In: Wang, J. (ed.) COCOON 2001. LNCS, vol. 2108, pp. 278–287. Springer, Heidelberg (2001)
19. Nikolova, E., Karger, D.R.: Route planning under uncertainty: the canadian traveller problem. In: AAAI 2008, pp. 969–974. AAAI Press (2008)
20. Nisan, N., Ronen, A.: Algorithmic mechanism design (extended abstract). In: STOC 1999, pp. 129–140. ACM, New York (1999)
21. Papadimitriou, C.H., Yannakakis, M.: Shortest paths without a map. In: Ausiello, G., Dezani-Ciancaglini, M., Della Rocca, S.R. (eds.) ICALP 1989. LNCS, vol. 372, pp. 610–620. Springer, Heidelberg (1989)
22. Roditty, L.: On the k-simple shortest paths problem in weighted directed graphs. In: SODA 2007, pp. 920–928. SIAM, Philadelphia (2007)
23. Roditty, L., Zwick, U.: Replacement paths and k simple shortest paths in unweighted directed graphs. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005. LNCS, vol. 3580, pp. 249–260. Springer, Heidelberg (2005)
24. Tarjan, R.E.: Efficiency of a good but not linear set union algorithm. J. ACM 22(2), 215–225 (1975)
25. Vassilevska Williams, V.: Faster replacement paths. In: SODA 2011, pp. 1337–1346. SIAM (2011)
26. Wulff-Nilsen, C.: Solving the replacement paths problem for planar directed graphs in $O(n \log n)$ time. In: SODA 2010, pp. 756–765. SIAM, Philadelphia (2010)
27. Yu, G., Yang, J.: On the robust shortest path problem. Computers & Operations Research 25(6), 457–468 (1998)