Recoverable Robust Shortest Path Problems

Christina Büsing
Technische Universität Berlin, Institut für Mathematik, Sekretariat MA 5-1, Straße des 17.Juni 136, 10623 Berlin, Germany

In this article, we investigate two different recoverable robust (RR) models to deal with cost uncertainties in a shortest path problem. RR extends the classical concept of robustness to deal with uncertainties by incorporating limited recovery actions after the full data are revealed. Our first model focuses on the case where the recovery actions are quite restricted: after a simple path is fixed in the first stage, in the second stage, after all data are revealed, any path containing at most *k* new arcs may be chosen. Thus, the parameter *k* can be interpreted as a mediator between robust optimization—no changes allowed—and optimization on the fly—an arbitrary solution can be chosen. Considering three classical scenario sets, which model uncertainties in the cost function, we show that this new problem is strongly NP-hard in all these cases and is not approximable, unless P = NP. This is in contrast to the robust shortest path problem, where, for example, an optimal solution can be computed efficiently for interval and Γ-scenarios. For series-parallel graphs and interval scenarios, we present a polynomial time algorithm for this RR setting. In our second model, the recovery set, that is, the set of paths selectable in the second stage is not limited, but deviating from the previous choice comes at extra cost. Thus, a path chosen in the first stage produces renting costs modeled as an α-fraction of the scenario cost. For an arc taken in the second stage, the remaining cost needs to be paid in addition to some extra inflation cost modeled by a β-fraction of the scenario cost, if the arc was not reserved beforehand. The complexity status of this problem is similar to the robust case. Yet, for Γ-scenarios, the problem is again strongly *NP*-hard, but can be approximated with a $\min\{2+\beta, \frac{1}{\alpha}\}$ factor. **NETWORKS, Vol. 59(1), 181–189 2012**

Keywords: uncertainties; robustness; recovery; shortest path problem; NP-hardness; approximation

1. INTRODUCTION

1.1. The k-Distance-Recoverable Robust Shortest Path Problem

One major concern in transportation is the design and extension of infrastructure in such a way that it serves the predicted future demands. For example, tracks or tunnels are built to connect important cities, canals are constructed to speed up the travel time between different seas, or highways are built to spread the traffic volume. In this planning process, costs for reforming the infrastructure are assumed to be known or are estimated. However, in the construction phase, these costs may change. For example, material costs depend on the current market price or the construction time may increase due to bad weather conditions, which induces extra costs for leasing the machinery. A goal in the planning process is to extend the infrastructure for as little cost as possible. Assuming a risk-averse policy, every extension is evaluated by its worst-case cost.

We simplify the problem described above and consider the setup of a new road connecting two fixed cities s and t. The potential roads that may be built are modeled via the set of all simple (s,t)-paths $\mathcal{P}$ in a given directed graph $G=(V,A)$. We call costs that are given at the beginning of the planning process, the first-stage costs and denote them with $c^1 : A \to \mathbb{N}$. The second kind of cost, which is uncertain at the planning stage, is modeled by a set of scenarios $\mathcal{S}$. Each scenario $S \in \mathcal{S}$ determines a cost function $c^S : A \to \mathbb{N}$, which represents a potential realization of the market prices. We assume that after the road is chosen, a scenario from the set $\mathcal{S}$ is revealed and determines the cost in the construction phase. An (s,t)-path p generates the first-stage cost $c^1(p) = \sum_{a\in p} c^1(a)$, and in each scenario the scenario cost $c^S(p) = \sum_{a\in p} c^S(a)$. Choosing a path with minimum first-stage cost and minimum maximal scenario cost corresponds to the problem of finding an optimal robust path.

However, in the construction phase, minor changes of the previously planned road are in general possible. For example, if in the case of building a tunnel a layer of granite turns up, small detours through permeable rock save cost and are realizable with little effort. Or if the ground becomes swamp land, circumventing this area seems wise. We include this

Received July 2009; accepted August 2011
Correspondence to: C. Büsing; e-mail: buesingchristina@web.de
Contract grant sponsor: Research Training Group "Methods for Discrete Structures"; Contract grant number: DFG-GRK 1408
Contract grant sponsor: Berlin Mathematical School
DOI 10.1002/net.20487
Published online 25 November 2011 in Wiley Online Library (wileyonlinelibrary.com).

possibility of taking small detours by allowing us to take an (s,t)-path p^S as soon as a scenario S is revealed, which uses up to k new arcs compared with the path chosen in the first stage. We call this new problem the k-distance recoverable robust (k-Dist-RR) shortest path problem and define it more formally in the following.

Definition 1 (k-Dist-RR Shortest Path Problem). *Let $G=(V,A)$ be a directed graph with a set of vertices V and a set of arcs A, and let s and t be two designated vertices in V. Let $c^1 : A \to \mathbb{N}$ be a first-stage cost function, $\mathcal{S}$ be a set of scenarios, where each scenario S defines a scenario cost function $c^S : A \to \mathbb{N}$, and $k \in \mathbb{N}$ be a recovery parameter. We denote by $\mathcal{P}$, the set of all simple (s,t)-paths in G. The recovery $\mathcal{P}_p^k$ of an (s,t)-path p consists of all (s,t)-paths p' with $|p' \backslash p| \leq k$, and the recovery cost $c_{RR}(p)$ is determined by*

$$c_{RR}(p) = \max_{S\in\mathcal{S}} \min_{p'\in\mathcal{P}_p^k} \sum_{a\in p'} c^S(a).$$

The first-stage cost of p is given by $c^1(p) = \sum_{a\in p} c^1(a)$. First-stage cost and recovery cost sum up to the total cost $c_T(p)$ of p, that is, $c_T(p) = c^1(p) + c_{RR}(p)$. The k-Dist-RR shortest path problem is to find an (s,t)-path $p^* \in \mathcal{P}$ with minimum total cost $c_T(p^*)$.

The parameter k represents the flexibility in the planning process for abandoning the original plan. Thus, this model can be interpreted as a mediator between "optimization on the fly," where for every scenario an optimal solution is chosen in the second stage, and robustness, where no changes are allowed after the solution is chosen in the first stage. The first case corresponds to $k = |V|$, the number of vertices V of the given graph, and the second case corresponds to $k = 0$.

In the motivating example, the first-stage cost modeled the estimated cost that occurs for building roads. Depending on the quality of this estimation, the scenario cost just consists of slight increases in these values. But also different problem settings can easily be represented. Consider, for example, a telecommunication network in which demand needs to be routed as fast as possible with respect to (w.r.t.) a given cost function c. Telecommunication networks are frequently faced with failure links, that is, a direct connection between two servers is not available. These failures can be presented by a set of scenarios, where each scenario places a fixed number of arcs at high cost and leaves the remaining cost at 0. The first-stage cost in this case models the routing cost c. In a third variant of the problem setting, no first-stage cost occurs and all costs are captured in the scenario cost. This setting emerges when the planning process induces no extra costs but during the realization process all costs may vary.

1.2. The Rent-RR Shortest Path Problem

A different problem arises after the infrastructure is built. In railway optimization, the holder of the tracks is interested in selling different time slots for crossing some tracks to other companies. Negotiations between two companies normally take place before the actual usage of the tracks. Depending on the time difference between usage and negotiation, the cost for using a track may not be fixed but remains adaptable to the current market prices. The company reserving a slot gains with this action the right to buy this time slot but is not obligated to do so. On the other hand, the company selling the slots may claim a fraction of the market price for reserving the slots. We will call this fraction the rental factor. If a company buys the slot, after it was reserved, it just pays the remaining cost. If the company buys any other slot, it needs to pay some extra handling fee, which we also assume to be a fraction of the market price. To have a guaranteed itinerary for sending their cargo, a company should reserve a path at the beginning. More formally, we obtain the Rent-RR shortest path problem.

Definition 2 (Rent-RR Shortest Path Problem). *Let $G=(V,A)$ be a directed graph and let s,t be two vertices in V. Furthermore, a rental factor $\alpha \in]0,1[$, an inflation factor $\beta \geq 0$, and a set of scenarios $\mathcal{S}$, where each scenario $S \in \mathcal{S}$ determines a scenario cost function $c^S : U \to \mathbb{N}$, are given. As before, $\mathcal{P}$ contains all simple (s,t)-paths in G. For a path $p \in \mathcal{P}$ the rent cost $c_R^S(p)$ in scenario S is defined by $c_R^S(p) = \alpha \cdot c^S(p)$ and the implementation cost $c_I(p)$ by*

$$c_I^S(p) = \min_{p'\in\mathcal{P}} (1-\alpha)c^S(p') + (\alpha+\beta) \sum_{a\in p'\backslash p} c^S(a).$$

The goal is to find a path with minimum total cost $c(p)$, defined as

$$c(p) = \max_{S\in\mathcal{S}} \left(c_R^S(p) + c_I^S(p)\right).$$

For a large inflation factor, for example, $\beta = \max_{S\in\mathcal{S}} c^S(A)$, any optimal robust solution is an optimal solution for the Rent-RR shortest path problem. In contrast to the k-Dist-RR shortest path problem, the recovery set for a first-stage solution is not restricted, that is, we can choose any path as soon as the scenario is revealed. Furthermore, there are no first-stage cost.

1.3. Related Results

There are two major trends in dealing with uncertainty given by a scenario set: stochastic programing and robust optimization. The latter method is in particular appropriate when dealing with high-risk settings or basic services like planning water and power supply networks. A solution is called robust if it remains feasible in all settings and is optimal if it minimizes its maximum scenario cost.

Robustness for discrete combinatorial optimization problems was first introduced by Kouvelis and Yu [13] in the middle of the 1990s. In [19], Yu and Yang showed that the robust shortest path problem is already weakly **NP**-hard if the set of scenarios consists of two scenarios. If the number of scenarios is not bounded by a constant, the problem is even strongly **NP**-hard. Aissi et al. [1] introduced a pseudopolynomial algorithm and a fully polynomial-time approximation scheme (FPTAS) for the first case and a lower bound on the approximation factor of $2 - \varepsilon$, $\varepsilon > 0$, for the latter case. This bound was improved by Kasperski and Zieliński [11] to $\log^{(1-\varepsilon)} |\mathcal{S}|$ for any $\varepsilon > 0$ and a discrete scenario set $\mathcal{S}$. For the special set of Γ-scenarios, Bertsimas and Sim [4] showed that an optimal robust shortest path problem can be computed by solving $|A|+1$ deterministic shortest path problems, where A denotes the set of arcs of the considered graph.

A different robust framework minimizes the maximum regret of a given solution. Such a solution is called an optimal min–max regret, relative robust or robust solution. In the case of discrete scenario sets, the same results as for the robust problem are obtained, that is, it is weakly **NP**-hard for two scenarios [19], can be solved in pseudopolynomial time for a constant number of scenarios [1] and is not approximable with a factor better than $\log^{(1-\varepsilon)} |\mathcal{S}|$ for any $\varepsilon > 0$ and a discrete scenario set $\mathcal{S}$ [11]. For interval scenarios, the relative robust shortest path problem is weakly **NP**-hard on series parallel graphs in contrast to the robust problem, which can be solved in polynomial time. This was shown by Kasperski and Zieliński [10]. Montemanni and Gambardella [15] provided an exact algorithm to solve the relative robust shortest path problem with interval scenarios via Benders decomposition.

The drawback of robustness is the unacceptable high cost of a robust solution. Furthermore, the concept ignores the fact that in most settings minor variations of a previously determined solution are possible. The idea to broaden the concept of robustness and to include some changes of the solution in the revealed setting has attracted many different researchers: Mulvey et al. [16] differentiate in their new robust model between design decisions fixed in a first stage and control decisions taken after all data are known, Ben-Tal et al. [3] introduced the concept of adjustable robustness for linear programs, Dhamdhere et al. [7] considered demand-robustness which was later called two-stage robustness (e.g., [9]), and Liebchen et al. [14] investigated RR inspired by the recovery action.

The shortest path problem has been studied in terms of demand-robustness (or later called two-stage robustness) by Dhamdhere et al. [7]. In this setting, the graph $G = (V, A)$, the source vertex s, and the cost $c : A \to \mathbb{N}$ are supposed to be known. Yet, the target vertex and the inflation factor on the cost is uncertain, that is, every scenario S defines a target vertex t^S and an inflation factor $\beta^S \geq 0$ for buying an arc in the second stage. The goal is to purchase some parts of the possible path $A_1 \subseteq A$ in advance for the cost $c(A_1)$, and to complete it for the given target vertex $t^S \in V$ to a feasible (s, t^S)-path in $A_1 \cup A_2^S$ by buying the arcs A_2^S for the cost $(1 + \beta^S)c(A_2^S)$ as soon as the scenario $S \in \mathcal{S}$ is realized. The objective is to minimize this cost over all possible scenarios, that is, to minimize $c(A_1) + \max_{S \in \mathcal{S}}(1 + \beta^S)c(A_2^S)$. Note that the cost system of Rent-RR shortest path problems is similar to this setting, but we assume the inflation factor to be fixed and the cost function to vary. Dhamdhere et al. [7] proved a 16-approximation algorithm for the demand-robust shortest path problem. A 7.1-approximation was later introduced by Golovin et al. [9].

Büsing [6] introduced the exact subgraph RR shortest path problem in which a subgraph is constructed to contain for every scenario a shortest path of the original graph. The goal is to find a subgraph of minimum cardinality. In the case of interval and Γ-scenarios, approximating the exact subset RR shortest path problem with a factor better than $|A|^{(1-\varepsilon)}$ is strongly **NP**-hard for any $\varepsilon > 0$, where A denotes the set of arcs in the considered graph. Returning the whole graph yields an $|A|$-approximation, which can be improved to an $\frac{|A|}{\ell+1}$-approximation algorithm for every constant $\ell \in \mathbb{N}$.

1.4. Our Contribution and Outline

Besides the model in [6], the k-Dist-RR shortest path problem and the Rent-RR shortest path problem are the first models that incorporate recovery action into a robust setting, which deals with uncertainties in the cost function and not in the demand. Our research focuses on the complexity status and approximation of these problems. We consider discrete scenario sets $\mathcal{S}_D$, interval scenario sets $\mathcal{S}_I$, and Γ-scenario sets $\mathcal{S}_\Gamma$ to model uncertainties in the cost functions. In a discrete scenario set, each scenario and its integer cost function are explicitly given. Interval scenario sets consist of all scenarios that determine a cost function whose values lie in a given cost interval defined by lower and upper cost bounds. For some integer Γ, Γ-scenario sets are modifications of interval scenario sets. In contrast to interval scenarios, a Γ-scenario may change at most Γ cost values from the lower bound to the corresponding upper bound.

In Section 2.1, we show that the k-Dist-RR shortest path problem is strongly **NP**-hard and not approximable for two scenarios, even if the first-stage costs are set to 0, unless **P** = **NP**. In the case of interval scenarios $\mathcal{S}_I$, considered in Section 2.2, the problem without first-stage cost is solvable in polynomial time. If the first-stage cost can be chosen arbitrarily, the k-Dist-RR shortest path problem with $\mathcal{S}_I$ turns out to be not approximable, unless **P** = **NP**. For the class of series-parallel graphs, we introduce a polynomial algorithm to solve the k-Dist-RR shortest path problem with interval scenarios.

As Γ-scenarios are a special case of interval scenarios, the k-Dist-RR shortest path problem with $\mathcal{S}_\Gamma$ is not approximable, unless **P** = **NP** (Section 2.3). Furthermore, the total cost of a given path is not computable in polynomial time for this scenario set, unless **P** = **NP**.

The results for the Rent-RR shortest path problem are closely related to those for the robust shortest path problem. In Section 3.1, we provide for discrete scenarios an L-reduction from the latter problem, and thus obtain that the Rent-RR shortest path problem is not approximable with a factor better than $\log |\mathcal{S}_D|$, unless **P** = **NP**. The interval case is solvable in polynomial time, as any shortest path w.r.t. the upper cost bounds $\overline{c}$ provides an optimal solution. However, for Γ-scenarios, the problem becomes again strongly **NP**-hard, in contrast to its robust version (Section 3.2).

In the last Section 3.3, we introduce an approximation algorithm based on a robust solution. For Γ-scenarios, we obtain a $\min(2+\beta, \frac{1}{\alpha})$-approximation for a rental factor $\alpha \in]0, 1[$ and an inflation factor $\beta \geq 0$. If $\alpha \geq 0.5$, we show that the analysis is tight.

2. THE COMPLEXITY OF THE *k*-DIST-RR SHORTEST PATH PROBLEM

2.1. Discrete Scenarios

In this section, we focus on the complexity of the k-Dist-RR shortest path problem and start by considering discrete scenario sets. For a given graph $G = (V, A)$, a discrete scenario set $\mathcal{S}_D$ consists of r scenarios $S_1, \ldots, S_r$, where each scenario determines a cost function $c^{S_i} : A \to \mathbb{N}$, $i = 1, \ldots, r$. We will show that even for two scenarios, the k-Dist-RR shortest path problem is strongly **NP**-hard and not approximable, unless $\mathbf{P} = \mathbf{NP}$. Note that the robust shortest path problem with two scenarios can be solved in pseudopolynomial time [1].

Theorem 3. *Let $G = (V, A)$ be a directed graph, s, t be two vertices in V, $c^1 : A \to \{0\}$ be the first-stage cost function, $\{S_1, S_2\}$ be two scenarios determining two cost functions $c^{S_i} : A \to \{0, 1\}$ with $|\{a \in A \mid c^{S_i}(a) = 1\}| \leq 3$, $i = 1, 2$, and $k \geq 2$ be a recovery parameter. Then solving the corresponding k-Dist-RR shortest path problem is strongly* **NP***-hard and not approximable, unless $\mathbf{P} = \mathbf{NP}$.*

Proof. We show a reduction from the two vertex disjoint path problem to the k-Dist-RR shortest path problem. Let I be an instance of the two vertex disjoint path problem given by a directed graph $G = (V, A)$ and two vertex pairs $\{v_1, u_1\}$ and $\{v_2, u_2\}$. The task in I is to decide whether two vertex disjoint paths, a (v_1, u_1)-path p_1 and a (v_2, u_2)-path p_2, exist. The **NP**-completeness of this problem follows directly from a lemma published in 1980 by Fortune et al. [8]. W.l.o.g., we assume that the graph G does not contain the arcs (v_1, u_1), (v_1, v_2), and (v_1, u_2) and that there are no outgoing arcs from u_1 and u_2 and no incoming arcs to v_2.

We show a reduction for $k = 2$ and define an instance I' of the k-Dist-RR shortest path problem with two scenarios S_1, S_2 in the following way: let $G' = (V', A')$ be an extension of G by six vertices s, t, v_1', v_2', w_1, and w_2, and 10 arcs (s, v_1'), (v_1', v_1), (v_1', u_1), (u_1, v_2'), (v_2', u_2), (v_2', v_2), (u_2, w_1), (u_2, w_2), (w_1, t), and (w_2, t) (see Fig. 1).

We define the two scenario cost functions c^{S_1} and c^{S_2} in such a way that any simple (s, t)-path with total cost equal to 0 satisfies the following conditions:

1. it contains the arcs (s, v_1'), (v_1', v_1), (u_1, v_2'), (v_2', v_2), (u_2, w_1), and (w_1, t);
2. it contains two subpaths connecting the vertex pairs $\{v_1, u_1\}$ and $\{v_2, u_2\}$;
3. it does not contain the arcs (v_1', u_1) and (v_2', u_2).

To this end, we define $c^{S_1}(a) = 1$ for $a \in \{(v_1', v_1), (v_2', v_2), (u_2, w_2)\}$ and $c^{S_1}(a) = 0$ for all other arcs

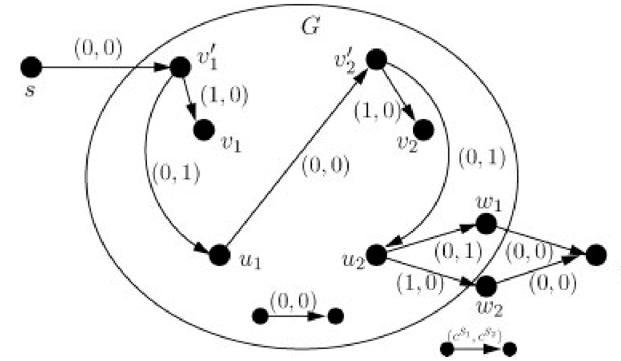

FIG. 1. The values on the arcs show the cost assignments of the two scenarios S_1 and S_2. The first-stage costs are set to 0.

$a \in A'$. Note that the only path with cost 0 w.r.t. the cost function c^{S_1} is the path $p^{S_1} = sv_1'u_1v_2'u_2w_1t$. Furthermore, we set $c^{S_2}(a) = 1$ for $a \in \{(v_1', u_1), (v_2', u_2), (u_2, w_1)\}$ and $c^{S_2}(a) = 0$ for all other arcs $a \in A'$ (see Fig. 1). If a path p satisfies the conditions from above, then $p_{[s,u_2]} \cup (u_2, w_2) \cup (w_2, t)$ is a recovery path with cost 0 w.r.t. c^{S_2}, where $p_{[a,b]}$ denotes the subpath of p connecting the vertices a and b. The scenario cost function sets exactly three arc costs to 1, all other costs remain at 0. To complete the definition of the k-Dist-RR shortest path instance I', we set the first-stage cost to 0. The size of instance I' is polynomial in the size of I.

We will now prove that there are two vertex disjoint paths p_1 and p_2 in G if and only if any optimal solution in I' has total cost 0.

($\Rightarrow$): Let p_1 be a (v_1, u_1)-path and p_2 be a (v_2, u_2)-path in G, and let both paths be vertex disjoint. Then the path

$$p = (s, v_1') \cup (v_1', v_1) \cup p_1 \cup (u_1, v_2') \cup (v_2', v_2) \cup p_2 \cup (u_2, w_1) \cup (w_1, t)$$

is a simple (s, t)-path in G' and contains the recovery paths p^{S_1} and $p^{S_2} = p_{[s,u_2]} \cup (u_2, w_2) \cup (w_2, t)$ for $k = 2$. Thus, the total cost of p is 0.

($\Leftarrow$): Let p be a simple (s, t)-path with total cost equal to 0. Let us first assume that p contains the arcs (u_2, w_2) and (w_2, t). As the only path with cost 0 in S_1 is p^{S_1} and $k = 2$, $p_{[s,u_2]} = p^{S_1}_{[s,u_2]}$. As $c^{S_2}((v_1', u_1)) = 1$, the recovery path in scenario S_2 needs to connect v_1 directly over an arc with u_1, v_2, or u_2. Yet, such an arc is not part of G' due to our assumption. Hence, p contains the arcs (u_2, w_1) and (w_1, t).

Note that now the recovery actions in S_2 are fixed to be used for the last two arcs. Thus, the subpath of p connecting s and u_2 equals the recovery path for scenario S_2. As the arcs (v_1', u_1) and (v_2', u_2) have cost 1 in scenario S_2, these two arcs are not contained in p. On the other hand, p needs to traverse the arc (u_1, v_2'), as otherwise the path p^{S_1} is not part of the recovery of p and we would obtain total cost greater than 0. Because of these properties of p, the path p connects the vertex v_1 with u_1 and v_2 with u_2. As p is a simple path, these subpaths are vertex disjoint. This observation concludes the proof.

If $k > 2$, we can just replace the two parallel paths connecting u_2 with t by two parallel paths of length k. ∎

The k-Dist-RR shortest path instance constructed in the reduction already covers two special cases mentioned in the introduction: first, the set of scenarios $\mathcal{S}_D$ models failure sets, that is, some arcs are not available in the current realization, and second, all costs are contained in the scenario cost functions, that is, no first-stage costs are paid. Note that these results depend on the fact that a simple (s,t)-path needs to be fixed in the first stage. Furthermore, the reduction proves **NP**-hardness just for the case of directed graphs, as in undirected graphs, the two vertex disjoint path problem can be solved in polynomial time [17].

Before we consider interval and Γ-scenarios, we will show that the total cost of a given (s,t)-path can be computed in polynomial time. Let p be an (s,t)-path and let $S \in \mathcal{S}$ be some scenario defining a cost function $c^S : A \to \mathbb{N}$ on the arc set A of the given graph $G = (V,A)$. We compute the best recovery path $p^S \in \mathcal{P}_p^k$ by solving a constrained shortest path (CSP-) problem. The cost functions of this CSP-problem are the scenario cost function and a distance cost function

$$d(a) = \begin{cases} 0 & \text{if } a \in p \\ 1 & \text{otherwise.} \end{cases}$$

The cost function d counts the number of arcs in a path p' that differ from the arcs in p, that is, $d(p') = |p' \backslash p|$ for any $p' \in \mathcal{P}$. To compute a path $p' \in \mathcal{P}_p^k$, we bound this value in the CSP-instance by k. In general, the CSP problem is weakly **NP**-hard and can be solved by a labeling Dijkstra algorithm in pseudopolynomial time $\mathcal{O}(n^2L^2)$, where L is the upper bound on the second cost. The labeling Dijkstra algorithm has been introduced by Aneja et al. [2]. As in our case, the bound k is smaller than n (otherwise the problem is trivial), this CSP-problem is solvable in polynomial time. Thus, by computing p^S for every scenario $S \in \mathcal{S}_D$, we obtain the total cost of a path p by

$$c_T(p) = c^1(p) + \max_{S \in \mathcal{S}_D} c^S(p^S).$$

2.2. *Interval Scenarios*

The interval scenario set $\mathcal{S}_I$ is defined indirectly for a graph $G = (V,A)$ by lower and upper cost bounds $\underline{c}(a)$ and $\overline{c}(a)$ on the scenario cost function for each arc $a \in A$, $0 \le \underline{c}(a) \le \overline{c}(a)$. For each cost function $c : A \to \mathbb{N}$ with $c(a) \in [\underline{c}(a), \overline{c}(a)]$, there exists a scenario $S \in \mathcal{S}_I$ with $c^S = c$ and every scenario cost function $c^S : A \to \mathbb{N}$ obeys these bounds. Obviously, the k-Dist-RR shortest path problem with interval scenario sets is equivalent to the k-Dist-RR shortest path problem with one discrete scenario, namely $S_{\max}$ with $c^{S_{\max}}(a) = \overline{c}(a)$ for all $a \in A$. Hence, the problem can be reduced to finding a first-stage path p and a recovery path p' with $|p' \backslash p| \le k$ minimizing $c(p) = c^1(p) + c^{S_{\max}}(p')$.

For the special case of $c^1 \equiv 0$, any shortest path w.r.t. the cost function $c^{S_{\max}}$ is an optimal solution of the corresponding k-Dist-RR instance. Yet, if the first-stage cost function can be chosen arbitrarily, we obtain, by a slight modification of the proof of Theorem 3, that the k-Dist-RR shortest path problem with interval scenarios is not approximable, unless **P** = **NP**.

Corollary 4. *The k-Dist-RR shortest path problem with interval scenarios is strongly* **NP***-hard and not approximable, unless* **P** = **NP**.

Sketch of proof. We use the same construction and notation as in the proof of Theorem 3. But instead of defining the two scenarios S_1 and S_2, we just define a first-stage cost function c^1 and one scenario $S_{\max}$. The scenario $S_{\max}$ assigns the same cost values to all arcs as the scenario S_1. Note that the scenario $S_{\max}$ represents a set of interval scenarios $\mathcal{S}_I$ by setting the upper cost bounds on every arc $a \in A$ to the value $c^{S_{\max}}(a)$ and setting the lower cost bounds to 0. Finally, c^1 assigns cost 0 to the arc (u_2, w_1) and the same cost values as the scenario S_2 to all other arcs. As in the proof of Theorem 3, there are two disjoint paths p_1 and p_2 connecting the designated vertices v_1, u_1 and v_2, u_2, respectively, if and only if there is an (s,t)-path in this instance with total cost 0. ∎

We will now consider the k-Dist-RR shortest path problem with interval scenarios on the special class of series-parallel graphs. Then, the k-Dist-RR shortest path problem with $\mathcal{S}_I$ can be solved in polynomial time. The algorithm is based on the following two properties: Let G be a series composition of G_1 and G_2, two series-parallel graphs. Any optimal solution in G using k arcs as recovery consists of an optimal solution to G_1 using i arcs as recovery and an optimal solution to G_2 using j arcs as recovery with $i+j = k$. If G is a parallel composition of G_1 and G_2, then either the optimal first-stage path p and its recovery path p' are both part of G_1 (or G_2), or p is in G_i and p' in G_j, $j \ne i$. In the second case, p is a shortest path according to c^1 and p' is a shortest path according to $c^{S_{\max}}$ with a maximal length of k arcs. A decomposition of a given series-parallel graph into parallel and series compositions starting from simple arcs can be computed in linear time [18].

Theorem 5. *An optimal solution of a k-Dist-RR shortest path problem with interval scenarios can be calculated in polynomial time on series-parallel graphs.*

2.3. *Γ-Scenarios*

The set of Γ-scenarios was introduced by Bertsimas and Sim [5] and is a modification of interval scenarios. Let $\underline{c}(a)$ and $\overline{c}(a)$ be lower and upper bounds on the scenario cost with $0 \le \underline{c}(a) \le \overline{c}(a)$ for all $a \in A$, where A is the arc set of a given graph $G = (V,A)$. A scenario $S \in \mathcal{S}_\Gamma$ is only allowed to have at most Γ cost values deviating from the lower bound, that is, $|\{a \in A \mid c^S(a) > \underline{c}(a)\}| \le \Gamma$. In contrast to the case of discrete scenarios, we show that it is already strongly **NP**-hard to compute the total cost of a given simple path.

Theorem 6. *In a k-Dist-RR shortest path problem with Γ-scenarios and $k \ge 4$ computing the total cost of a given path is a strongly* **NP***-hard problem.*

Proof. We show a reduction from the max-scenario problem. Let I be an instance of the max-scenario problem given by a directed graph $G = (V, A)$ and two vertices $s, t \in V$ such that every (s, t)-path in G contains at most four arcs. Furthermore, let $\mathcal{S}_\Gamma$ be a set of Γ-scenarios and let K be a threshold. The max-scenario problem asks whether a scenario $S \in \mathcal{S}_\Gamma$ exists such that $\min_{p\in\mathcal{P}} c^S(p) \geq K$. In [6] Büsing showed that this problem is strongly **NP**-hard. To define an instance of the k-Dist-RR shortest path problem, we add one arc a_1 to the graph G with lower and upper cost bounds $K + 1$ for the scenario cost function and assign first-stage cost 0 to all arcs in the graph. Furthermore, we set $k = 4$ and consider the path $p = a_1$. As any (s, t)-path in G is a feasible recovery path for p, the total cost of p is greater than or equal to K if and only if I is a yes-instance. ▪

Note that the decision problem whether the total cost of a given (s, t)-path p is smaller than or equal to K is in **coNP**, as in that case there always exist a scenario $S \in \mathcal{S}_\Gamma$ such that the recovery cost and the first-stage cost of p are greater than or equal to $K + 1$. Furthermore, it follows directly that also the decision version of the k-Dist-RR shortest path problem is not in **coNP** = **NP**: the decision version is defined as follows.

Given: a graph $G = (V, A)$, two vertices s and t, a first-stage cost function $c^1 : A \to \mathbb{N}$, lower and upper cost bounds $\underline{c}(a)$ and $\overline{c}(a)$ for every $a \in A$, a parameter $\Gamma \in \mathbb{N}$, a recovery parameter $k \in \mathbb{N}$ and a threshold $K \geq 0$.

Decide: whether there exists a simple (s, t)-path p with total cost smaller than or equal to K.

If this decision version is in **NP**, there is a polynomial certificate with which we can decide in polynomial time whether there exists a path p with total cost smaller than or equal to K. Thus, by adding first-stage cost of $K+1$ to all arcs in G and 0 to the arc a_1 to the constructed instance in the proof of Theorem 6, we obtain a polynomial certificate for a no-instance of the max-scenario problem, an **NP**-complete problem. This would imply **NP** = **coNP**. It remains open whether the k-Dist-RR shortest path problem with Γ-scenarios is in **coNP**.

As Γ-scenarios are a special case of interval scenarios, this problem is also strongly **NP**-hard, as a consequence from Corollary 4.

Corollary 7. *The k-Dist-RR shortest path problem with Γ-scenarios is strongly* **NP***-hard and not approximable, unless* **P** = **NP**.

We will now turn to the Rent-RR shortest path problem.

3. THE RENT-RR SHORTEST PATH PROBLEM AND ITS COMPLEXITY

3.1. Discrete Scenarios and Interval Scenarios

The complexity status of the Rent-RR shortest path problem is more similar to the robust shortest path problem, which we will show by an L-reduction. Recall that to establish an L-reduction (e.g., [12]) from an optimization problem $\mathcal{X}$ to an optimization problem $\mathcal{X}'$ we have to define a pair of functions f and g, both computable in polynomial time, and two constants $\gamma, \delta > 0$ such that for any instance I of $\mathcal{X}$:

- $f(I)$ is an instance of $\mathcal{X}'$ with $\mathrm{OPT}(f(I)) \leq \gamma\,\mathrm{OPT}(I)$ (Condition 1);
- for any feasible solution y' of $f(I)$, $g(I, y')$ is a feasible solution of I such that $|c_I(g(I, y')) - \mathrm{OPT}(I)| \leq \delta|c_{f(I)}(y') - \mathrm{OPT}(f(I))|$ (Condition 2),

where c_I is the cost function and $\mathrm{OPT}(I)$ is the value of an optimal solution of the instance I. We will now introduce an L-reduction from the robust shortest path problem to the Rent-RR shortest path problem.

Theorem 8. *The robust shortest path problem with discrete scenario sets $\mathcal{S}_D$ is L-reducible to a Rent-RR shortest path problem with $\mathcal{S}_D$ and some rent and inflation factor $\alpha \in]0, 1[$ and $\beta \geq 0$.*

Proof. We start by defining a function f, which constructs for a given robust shortest path instance I a Rent-RR shortest path instance $f(I) = I'$ such that the optimal values of both instances satisfy $\mathrm{OPT}(I) = \alpha\,\mathrm{OPT}(I')$. Let $G = (V, A)$ be a directed graph, s and t be two vertices, and $\mathcal{S}_D = \{S_1, \ldots, S_r\}$ be a set of discrete scenarios, where each scenario $S \in \mathcal{S}_D$ defines a cost function $c^S : A \to \mathbb{N}$. The goal in I is to find an (s, t)-path p minimizing the robust cost $c_{\mathrm{rob}}(p) := \max_{S\in\mathcal{S}_D} c^S(p)$. To extend this instance I of a robust shortest path problem to an instance of the Rent-RR shortest path problem I', we add r parallel (s, t)-arcs $\overline{a}_i$, $i = 1, \ldots, r$, to obtain the graph G'. Furthermore, we define a scenario set $\mathcal{S}'_D$ on this graph in the following way: for every scenario $S_i \in \mathcal{S}_D$, the set $\mathcal{S}'_D$ contains a scenario S'_i with $c^{S'_i}(a) = c^{S_i}(a)$ for all $a \in A$, $c^{S'_i}(\overline{a}_i) = 0$, and $c^{S'_i}(a) = M$ otherwise with $M = \max_{S\in\mathcal{S}_D} c^S(A)$, an upper bound on the maximum robust cost in instance I. We denote by $\mathcal{P}(G)$, $\mathcal{P}(G')$ the set of all (s, t)-paths in G, G', respectively. We will now show that the optimal values of I and I' satisfy $\mathrm{OPT}(I) = \alpha\,\mathrm{OPT}(I')$. Let $p \in \mathcal{P}(G)$. As in every scenario the implementation cost is 0,

$$\begin{aligned} c_{\mathrm{rob}}(p) &= \max_{S\in\mathcal{S}_D} c^S(p) \\ &= \max_{S'\in\mathcal{S}'_D} \min_{p'\in\mathcal{P}(G')} \alpha c^{S'}(p) \\ &\quad + \underbrace{(1-\alpha)c^{S'}(p') + (\alpha+\beta)c^{S'}(p')}_{=0} = \alpha c(p) \end{aligned}$$

for any rental factor $\alpha \in]0, 1[$ and inflation factor $\beta \geq 0$. For any other path $p \in \mathcal{P}(G')\backslash\mathcal{P}(G)$, the total cost equals $\alpha \cdot M$. Hence, $\mathrm{OPT}(I) = \alpha\,\mathrm{OPT}(I')$.

Finally, we define a function g such that Condition 2 is satisfied. To this end, we fix an arbitrary (s, t)-path p^1 in G as representation for any (s, t)-path $p \in \mathcal{P}(G')$ that is not in $\mathcal{P}(G)$, that is, for any $p \in \mathcal{P}(G')$, $g(I, p) = p^1$ if $p \notin \mathcal{P}(G)$

and $g(I,p) = p$ if $p \in \mathcal{P}(G)$. We distinguish again between a path in $\mathcal{P}(G)$ and a path in $\mathcal{P}(G')$. If $p \in \mathcal{P}(G)$, then

$$|c_{\text{rob}}(g(I,p)) - \text{OPT}(I)| = \left|\frac{1}{\alpha}c(p) - \frac{1}{\alpha}\text{OPT}(I')\right| = \frac{1}{\alpha}|c(p) - \text{OPT}(I')|.$$

If $p \in \mathcal{P}(G')\backslash\mathcal{P}(G)$, then

$$\begin{aligned}|c_{\text{rob}}(g(I,p)) - \text{OPT}(I)| &\leq |M - \text{OPT}(I)| \\ &= \left|M - \frac{1}{\alpha}\text{OPT}(I')\right| \\ &= \frac{1}{\alpha}|\alpha M - \text{OPT}(I')| \\ &= \frac{1}{\alpha}|c(p) - \text{OPT}(I')|.\end{aligned}$$

Thus, for $\delta = \frac{1}{\alpha}$, this concludes the L-reduction. ■

Corollary 9. *The Rent-RR shortest path problem with discrete scenarios $\mathcal{S}_D$ is weakly* **NP**-*hard for two discrete scenarios and not approximable with a factor* $\log^{(1-\varepsilon)}|\mathcal{S}_D|$ *for any* $\varepsilon > 0$*, unless* **P** = **NP**.

In the case of interval scenarios, every scenario is dominated by $S_{\max}$ with $c^{S_{\max}}(a) = \overline{c}(a)$ for every arc $a \in A$, where $\overline{c}(a)$ is the upper cost bound defined by the interval scenario set. Consequently, any shortest path in terms of this cost function yields an optimal solution for the Rent-RR shortest path problem.

So far, the complexity status of the robust shortest path problem and the Rent-RR shortest path problem is the same. Yet, for Γ-scenarios, we will show that the Rent-RR shortest path problem is strongly **NP**-hard, whereas the robust shortest path problem can be solved in polynomial time, as shown by Bertsimas and Sim [4].

3.2. Γ-Scenarios

One can easily prove with similar arguments as in the k-Dist-RR case that the total cost for a given path is strongly **NP**-hard to compute in a Rent-RR shortest path instance. But even without returning the total cost for a given path, the problem remains **NP**-hard.

Theorem 10. *The Rent-RR shortest path problem with $\mathcal{S}_\Gamma$ is strongly* **NP**-*hard.*

Proof. We reduce from the max-scenario problem with Γ-scenarios, which has been shown to be strongly **NP**-hard in [6]. Let I be an instance of the max-scenario problem given by a directed graph $G = (V,A)$, two designated vertices s and t, lower and upper cost bounds on the arcs $\underline{c}(a)$ and $\overline{c}(a)$ for all $a \in A$, an integer Γ, and a threshold $K \geq 0$. We denote by $\mathcal{P}(G)$ the set of all simple (s,t)-paths in G. The task in I is to decide whether a scenario $S \in \mathcal{S}_\Gamma$ exists such that $\min_{p\in\mathcal{P}(G)} c^S(p) \geq K$. The idea of the reduction is as follows: we increase the cost of all (s,t)-paths in G by a constant x and add one further path $\overline{p}$ with fixed cost y in all scenarios to the instance. The values of x, y, the rental factor α, and the inflation factor β are set in such a way that $\overline{p}$ is the optimal solution of the Rent-RR shortest path instance if and only if I is a yes-instance.

More formally, we define $\beta = \frac{1}{2K}$, $\alpha = \frac{1}{2\tilde{c}+1}\beta$ with $\tilde{c} = \max_{S\in\mathcal{S}_\Gamma} c^S(\tilde{p})$ for some $\tilde{p} \in \mathcal{P}(G)$ and add an arc (s,t) to G to define a Rent-RR shortest path instance I'. We call this extra arc also the (s,t)-path $\overline{p}$ and denote the new graph with G' and all simple (s,t)-paths with $\mathcal{P}(G')$. Finally, we increase the lower and upper cost bounds for any outgoing arc of s by $x = \frac{(1-\alpha)}{(\beta+\alpha)}(K-\varepsilon)+\delta$, for some $0 < \varepsilon < \frac{1}{2}$, and set the upper and lower cost bounds on the extra arc to $y = K + x - \varepsilon$. All other upper and lower cost bounds remain as in the instance I. We now prove that I is a yes-instance if and only if $\overline{p}$ is the optimal solution in I'.

Let I be a yes-instance, that is, there exists a scenario $S \in \mathcal{S}_\Gamma$ with $\min_{p\in\mathcal{P}(G)} c^S(p) \geq K$. For every (s,t)-path $p \in \mathcal{P}(G)$, the total cost can be bounded below by

$$\begin{aligned}c(p) = \max_{S\in\mathcal{S}_\Gamma} \min_{p'\in\mathcal{P}(G')} &\alpha c^S(p) + (1-\alpha)c^S(p') \\ &+ (\alpha+\beta)c^S(p'\backslash p) \\ &\geq \min\{(K+x), \alpha(K+x) + (1+\beta)y\}.\end{aligned}$$

For the solution $\overline{p}$ we get

$$c(\overline{p}) \leq \min\left\{y, \max_{S\in\mathcal{S}_\Gamma}(\alpha y + (1+\beta)\min_{p\in\mathcal{P}(G)} c^S(p))\right\} \leq y.$$

Because of the definition of y and x, we obtain $y < K + x$, and thus $c(\overline{p}) < c(p)$ for any path $p \in \mathcal{P}(G')\backslash\{\overline{p}\}$.

Let now I be a no-instance, that is, for every scenario $S \in \mathcal{S}_\Gamma$ there is a solution $p \in \mathcal{P}(G)$ with $c^S(p) \leq K - 1$. For the already fixed path $\tilde{p}$, we obtain an upper bound on the total cost by

$$\begin{aligned}c(\tilde{p}) = \max_{S\in\mathcal{S}_\Gamma} \min_{p'\in\mathcal{P}(G')} &\alpha c^S(\tilde{p}) + (1-\alpha)c^S(p') \\ &+ (\alpha+\beta)c^S(p'\backslash\tilde{p}) \\ &\leq \alpha(\tilde{c}+x) + (1-\alpha)x + (1+\beta)(K-1),\end{aligned}$$

if

$$(1+\beta)y > (1+\beta)(K-1) + (1-\alpha)y. \tag{3.1}$$

Furthermore, as $c^S(p) \geq x$ for all $S \in \mathcal{S}_\Gamma$ and all $p \in \mathcal{P}(G')\backslash\{\overline{p}\}$, the total cost of $\overline{p}$ sums to

$$c(\overline{p}) = \min\left\{y, \max_{S\in\mathcal{S}_\Gamma} \min_{p'\in\mathcal{P}(G')} \alpha y + (1+\beta)c^S(p')\right\} = y$$

if

$$(1-\alpha)y < (1+\beta)x. \tag{3.2}$$

Comparing these two bounds, $c(\tilde{p}) < c(\overline{p})$ if

$$\alpha(\tilde{c}+x)+(1-\alpha)x+(1+\beta)(K-1) < y. \tag{3.3}$$

As one can easily recalculate, the inequalities (3.1)–(3.3) are satisfied for the chosen values of β, α, x, and y. Thus, $\overline{p}$ is the optimal solution of the Rent-RR shortest path instance I' with $\beta = \frac{1}{2K}$ and $\alpha = \frac{1}{2\tilde{c}+1}\beta$, if and only if the max-scenario instance I is a yes-instance. ∎

We will conclude this section by introducing an approximation algorithm for the Rent-RR shortest path problem, which is based on an approximation algorithm for the robust shortest path problem.

3.3. Approximation Algorithm

As an optimal solution cannot be constructed efficiently for discrete and Γ-scenarios if $\mathbf{P} \neq \mathbf{NP}$, we are interested in approximation algorithms. An approximation algorithm constructs a first-stage solution $p \in \mathcal{P}$ and gives for every first-stage solution p and scenario $S \in \mathcal{S}$ a recovery strategy, that is, a rule how to compute the recovery solution. The following theorem states how to generate an approximation algorithm for the Rent-RR shortest path problem from a robust shortest path solution.

Theorem 11. *Let $G = (V, A)$ be a directed graph, s and t be two vertices in V, $\mathcal{S}$ be a set of scenarios, and* ALG *be an approximation algorithm for the robust shortest path problem with an approximation factor γ. For a given rental factor $\alpha \in]0,1[$ and inflation factor $\beta \geq 0$, we define algorithm* ALG$'$ *by:*

First stage: *Run* ALG *on the robust shortest path instance $(G, s, t, \mathcal{S})$ and set the first-stage solution p^r to the output of* ALG.
Recovery: *For any $S \in \mathcal{S}$ calculate an optimal solution p^S for the shortest path instance G with the cost function*

$$c'(a) = \begin{cases} (1-\alpha)\cdot c^S(a) & \forall a \in p^r \\ (1+\beta)\cdot c^S(a) & \forall a \notin p^r. \end{cases}$$

Then ALG$'$ *is an approximation algorithm with an approximation factor $\gamma' = \min\{(\gamma+1+\beta), \frac{\gamma}{\alpha}\}$ of the Rent-RR shortest path problem.*

Proof. Let p^r be a solution computed by algorithm ALG for the robust shortest path instance $(G, s, t, \mathcal{S})$ and let $\mathrm{OPT}_{\mathrm{rob}}$ be the value of an optimal robust solution, that is, $\mathrm{OPT}_{\mathrm{rob}} = \min_{p\in\mathcal{P}} \max_{S\in\mathcal{S}} c^S(p)$. Furthermore, let OPT be the value of an optimal solution of the Rent-RR shortest path instance with a rental factor $\alpha > 0$ and an inflation factor $\beta \geq 0$. We start with two lower bounds on OPT: first

$$\begin{aligned} \mathrm{OPT} &= \min_{p\in\mathcal{P}} \max_{S\in\mathcal{S}} \min_{p'\in\mathcal{P}} \alpha c^S(p) + (1-\alpha)c^S(p') \\ &\quad + (\alpha+\beta)c^S(p'\backslash p) \\ &\geq \alpha \min_{p\in\mathcal{P}} \max_{S\in\mathcal{S}} c^S(p) = \alpha\, \mathrm{OPT}_{\mathrm{rob}} \end{aligned} \tag{3.4}$$

and second

$$\mathrm{OPT} \geq \max_{S\in\mathcal{S}} \min_{p'\in\mathcal{F}} c^S(p'). \tag{3.5}$$

We use the first bound (3.4) to obtain an estimate of the maximum rental cost of p^r, more precisely,

$$c_R(p^r) = \alpha \max_{S\in\mathcal{S}} c^S(p^r) \leq \alpha\cdot\gamma\, \mathrm{OPT}_{\mathrm{rob}} \leq \gamma\, \mathrm{OPT}. \tag{3.6}$$

An upper bound on the implementation cost of p^r in any scenario $S \in \mathcal{S}_D$ is given by

$$c_I^S(p^r) \leq (1+\beta) \max_{S\in\mathcal{S}} \min_{p'\in\mathcal{P}} c^S(p') \leq (1+\beta)\,\mathrm{OPT}, \tag{3.7}$$

using inequality (3.5). Combining estimates (3.6) and (3.7), we get

$$\begin{aligned} c(p^r) &= \max_{S\in\mathcal{S}} c_R^S(p^r) + c_I^S(p^r) \leq c_R(p^r) + \max_{S\in\mathcal{S}} c_I^S(p^r) \\ &\leq \gamma\,\mathrm{OPT} + (1+\beta)\,\mathrm{OPT} \leq (\gamma+1+\beta)\,\mathrm{OPT}. \end{aligned}$$

Thus, we have a first approximation guaranty of ALG$'$.

The second guaranty is based on the recovery step. As an optimal solution w.r.t. the cost function c' is chosen in the second stage, we obtain for any $p \in \mathcal{P}$

$$\begin{aligned} c(p) &= \max_{S\in\mathcal{S}_D} \min_{p'\in\mathcal{P}} \alpha c^S(p) + (1-\alpha)c^S(p') + (1+\beta)c^S(p'\backslash p) \\ &\leq \max_{S\in\mathcal{S}_D} c^S(p). \end{aligned}$$

Because of the choice of p^r, this leads to

$$\frac{1}{\gamma} c(p^r) \leq \min_{p\in\mathcal{P}} \max_{S\in\mathcal{S}} c^S(p) \leq \frac{1}{\alpha}\,\mathrm{OPT}.$$

To sum up, algorithm ALG$'$ is a $\min\{\gamma+1+\beta, \frac{\gamma}{\alpha}\}$-approximation algorithm for the Rent-RR shortest path problem. ∎

In the case of Γ-scenarios, the robust shortest path problem can be solved in polynomial time [4], and thus we obtain a $\min\{2+\beta, \frac{1}{\alpha}\}$-approximation algorithm for the Rent-RR shortest path problem. For $\alpha \geq 0.5$, the approximation factor is tight, as we will see in the following example: Let $G = (V, A)$ be a directed graph composed of an (s,t)-arc with the cost interval $[0,1]$, also denoted as path $\tilde{p}$, and a path p from s to t with two arcs, where a cost interval of $[0, 0.5]$ is assigned to each of these arcs. For $\Gamma = 2$, both paths are optimal robust shortest paths, and thus the algorithm could choose $\tilde{p}$. This results in total cost $c(\tilde{p}) = \min\{1, \alpha + (1+\beta)\cdot 0.5\}$, whereas the path p yields the optimal total cost $c(p) = \max\{\alpha, 0.5\}$. For $\alpha \geq 0.5$, we obtain $c(\tilde{p}) = \frac{1}{\alpha}\cdot c(p)$.

4. CONCLUSIONS

We considered two different RR shortest path problems and investigated their complexity w.r.t the most common scenario sets in the literature. For all these sets, the k-Dist-RR shortest path problem is strongly **NP**-hard and not approximable, unless $\mathbf{P} = \mathbf{NP}$. For the special case of series-parallel graphs, we introduce a polynomial algorithm to solve the problem with interval scenarios.

The Rent-RR shortest path problem is closely related to the robust shortest path problem. For discrete scenarios, we provided an L-reduction from the latter problem to the Rent-RR shortest path problem, and thus show that the Rent-RR shortest path problem cannot be approximated with a factor better than $\log^{(1-\varepsilon)} |\mathcal{S}_D|$ for any $\varepsilon > 0$ and a discrete scenario set $\mathcal{S}_D$. Yet, in contrast to the robust case, the Rent-RR shortest path problem is for Γ-scenarios strongly **NP**-hard. On the other hand, any approximation algorithm for the robust shortest path problem can be adapted to an approximation algorithm for the Rent-RR shortest path problem.

Acknowledgments

The author wishes to thank Martin Groß, Christian Liebchen, Rolf H. Möhring, Martin Skutella, and Sebastian Stiller for fruitful discussions. Furthermore, the author would also like to thank the referees for their valuable comments.

REFERENCES

[1] H. Aissi, C. Bazgan, and D. Vanderpooten, "Approximation complexity of min–max (regret) versions of shortest path, spanning tree, and knapsack," Algorithms—ESA 2005, Lecture notes in computer science, Vol. 3669, Springer, Berlin, 2005, pp. 862–873.

[2] Y.P. Aneja, V. Aggarwal, and K.P.K. Nair, Shortest chain subject to side constraints, Networks 13 (1983), 295–302.

[3] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski, Adjustable robust solutions of uncertain linear programs, Math Program 99 (2004), 351–376.

[4] D. Bertsimas and M. Sim, Robust discrete optimization and network flows, Math Program 98 (2003), 49–71.

[5] D. Bertsimas and M. Sim, The price of robustness, Oper Res 52 (2004), 35–53.

[6] C. Büsing, "The exact subgraph recoverable robust shortest path problem," Robust and online large-scale optimization, Lecture notes in computer science, Vol. 5868, Springer, Berlin, 2009, pp. 231–248.

[7] K. Dhamdhere, V. Goyal, R. Ravi, and M. Singh, How to pay, come what may: Approximation algorithms for demand-robust covering problems, FOCS 2005—46th Annual IEEE Symposium on Foundations of Computer Science, IEEE Computer Society, Los Alamitos, California, 2005, pp. 367–378.

[8] S. Fortune, J. Hopcroft, and J. Wyllie, The directed subgraph homeomorphism problem, Theor Comput Sci 10 (1980), 111–121.

[9] D. Golovin, V. Goyal, and R. Ravi, "Pay today for a rainy day: Improved approximation algorithms for demand-robust min-cut and shortest path problems," STACS 2006, Lecture Notes in Computer Science, Vol. 3884, 2006, pp. 206–217.

[10] A. Kasperski and P. Zieliński, The robust shortest path problem in series-parallel multidigraphs with interval data, Oper Res Lett 34 (2006), 69–76.

[11] A. Kasperski and P. Zieliński, On the approximability of min-max (regret) network optimization problems, Inf Process Lett 109 (2009), 262–266.

[12] B. Korte and J. Vygen, Combinatorial optimization: Theory and algorithms, Springer, Berlin, 2008.

[13] P. Kouvelis and G. Yu, Robust discrete optimization and its applications, Kluwer Academic Publishers, Dordrecht, 1997.

[14] C. Liebchen, M.E. Lübbecke, R.H. Möhring, and S. Stiller, "The concept of recoverable robustness, linear programming recovery, and railway applications," Robust and online large-scale optimization, Lecture notes in computer science, Vol. 5868, Springer, Berlin, 2009, pp. 1–27.

[15] R. Montemanni and L. Gambardella, The robust shortest path problem with interval data via Benders decomposition, 4OR 3 (2005), 315–328.

[16] J.M. Mulvey, R.J. Vanderbei, and S.A. Zenios, Robust optimization of large-scale systems, Oper Res 43 (1995), 264–281.

[17] Y. Shiloach, A polynomial solution to the undirected two paths problem, J Assoc Comput Machinery 27 (1980), 445–456.

[18] J. Valdes, R. Tarjan, and E. Lawer, "The recognition of series parallel digraphs," STOC 1979—Proceedings of the Eleventh Annual ACM Symposium on Theory of Computing, ACM, New York, N.Y., 1979, pp. 1–12.

[19] G. Yu and J. Yang, On the robust shortest path problem, Comput Oper Res 25 (1998), 457–468.