On the Multistage Shortest Path Problem Under Distributional Uncertainty

Sergey S. Ketkov

Received: 30 May 2022 / Accepted: 29 January 2023 / Published online: 20 February 2023
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

In this paper, we consider an ambiguity-averse multistage network game between a user and an attacker. The arc costs are assumed to be random variables that satisfy prescribed first-order moment constraints for some subsets of arcs and individual probability constraints for some particular arcs. The user aims at minimizing its cumulative expected loss by traversing between two fixed nodes in the network, while the attacker’s objective is to maximize the user’s expected loss by selecting a distribution of arc costs from the family of admissible distributions. In contrast to most of the related studies, both the user and the attacker can dynamically adjust their decisions at particular nodes of the user’s path. By observing the user’s decisions, the attacker may reveal some additional distributional information associated with the arcs emanated from the current user’s position. It is shown that the resulting multistage distributionally robust shortest path problem (DRSPP) admits a linear mixed-integer programming reformulation (MIP). In particular, we distinguish between acyclic and general graphs by introducing different forms of non-anticipativity constraints. Finally, we perform a numerical study, where the quality of adaptive decisions and computational tractability of the proposed MIP reformulation are explored with respect to several classes of synthetic network instances.

Keywords Shortest path problem · Distributionally robust optimization · Polyhedral uncertainty · Piecewise constant decision rules · Mixed-integer programming

Mathematics Subject Classification 90C17 · 90C27 · 90C47
1 Introduction

Distributionally robust optimization (DRO) is a methodology for addressing uncertainty in optimization problems, where the probability distribution of uncertain parameters is only known to reside within an ambiguity set (or a family) of admissible distributions; see, e.g., the studies in [5, 17, 18, 20, 40]. A standard one-stage DRO problem can be usually formulated as a bi-level problem, where the expected value of the loss function (or some measure of risk, if the decision-maker is risk-averse) is minimized under the worst-case possible distribution of uncertain parameters.

Typically, in one-stage DRO problems the decisions are implemented here-and-now, before the realization of uncertainty. However, the distributionally robust optimization approach may also address dynamic (or, equivalently, multistage) optimization problems, where decisions adapt to the uncertain outcomes as they unfold in stages; see, e.g., [5, 20, 23, 26]. In other words, in multistage DRO problems decision variables at the current stage can be thought as some functions of the uncertain problem parameters observed up to this stage.

In this study, we consider a multistage version of the distributionally robust shortest path problem (DRSPP), where the vector of arc costs/travel times in a given network is subject to uncertainty. From the game theoretical perspective, our problem can be viewed as a dynamic zero-sum game between two decision-makers, which are referred to as a user and an attacker. The user attempts to minimize its expected loss by traversing between two fixed nodes in a given network. On the other hand, the attacker aims at maximizing the user’s expected loss by selecting a distribution of arc costs within a given family of probability distributions. The outlined game is dynamic in the sense that both the user and the attacker are able to adjust their decisions at particular nodes of the user’s path.

In fact, our multistage formulation is motivated and built upon the related one-stage formulation of DRSPP in [28]. Specifically, we preserve the same form of the ambiguity set and the objective function but introduce some auxiliary distributional constraints that can be verified by the user dynamically while traversing through the network.

1.1 Related Literature

A rather large number of studies consider one-stage versions of DRSPP assuming different forms of ambiguity sets and objective criteria; see, e.g., [13, 19, 39, 44]. Under some assumptions on the geometry of ambiguity sets and functional properties of the objective function, one-stage problems can be reduced to single-level linear or nonlinear mixed-integer programming (MIP) problems. In particular, these single-level reformulations typically rely on strong duality results for moments problems due to Isii [25] and Shapiro [35]. We also refer to [29] for another version of DRSPP, in which the uncertain problem parameters are related not to the arc costs/travel times in a given network but to some resource constraints.

It can also be argued that in a number of practical applications the arc costs/travel times in a given network can only be observed using a finite training data set; see,
In this case, one may attempt to construct an ambiguity set, which contains the nominal distribution of the cost vector with a required confidence level. For example, some data-driven approaches to constructing ambiguity sets are based on estimating the mean and the covariance matrix of uncertain problem parameters [17] or using some distance metrics from the empirical distribution of the data [2, 18].

In contrast to one-stage problems, multistage distributionally robust formulations of the shortest path problem are considered by relatively few authors. The major limitation of this problem setting is that recourse decisions, i.e., the decisions affected by uncertainty, are discrete and, therefore, the application of standard linear or piecewise linear decision rules [5, 20] is rather limited. In the following, we discuss multistage problem formulations that may account the shortest path problem as a special case.

First, we refer to the study by Hanasusanto et al. [23], where a two-stage DRO problem with binary recourse decisions is approximated by using a restricted number of preselected second-stage decisions. The authors prove that the proposed approximation is exact, if the number of preselected decisions is sufficiently large. Despite the fact that the study in [23] may address rather general types of ambiguity sets and objective criteria, it is not quite clear how to extend these theoretical results to a multistage problem setting.

Another line of research is focused on exact dynamic programming-based algorithms assuming that the uncertain parameters are stage-wise independent. To the best of our knowledge, Yu and Shen [43] are among the first to consider multistage distributionally robust MIPs. The authors investigate the concept of decision-dependent ambiguity sets, where parameters of the first- and the second-order moment constraints at a stage $t \in \mathbb{Z}_+$ are linearly dependent on the user’s decision at the previous stage, $t - 1$. The authors reformulate the stage-wise problems as linear MIP or semidefinite programming problems. Then, different versions of stochastic dual dynamic integer programming (SDDiP) algorithm [45] are applied for solving the resulting multistage problems exactly or deriving objective bounds. The latter algorithm is shown to be finitely convergent and rather effective for moderate-sized problem instances.

In conclusion, we discuss a multistage shortest path interdiction model proposed by Sefair et al. [34] and a multistage robust MIP problem formulation of Bertsimas and Dunning [3]. Despite the fact that these models are deterministic, they provide some interesting insights for the multistage DRSPP considered in the current study.

Sefair et al. [34] consider a zero-sum multistage network game between a user and an attacker, where the attacker attempts to maximize the user’s loss by blocking a subset of arcs (subject to a budgetary constraint) any time the user reaches a node in the network. Furthermore, the user may dynamically alter its path by observing the arcs blocked by the attacker. The authors in [34] demonstrate that the attacker’s problem can be solved in polynomial time for acyclic graphs and admits an exact exponential-state dynamic programming algorithm in the general case.

On the other hand, Bertsimas and Dunning [3] consider a class of linear MIP problems, where the uncertain problem parameters are only known to reside within a polyhedral uncertainty set. The uncertainty set is partitioned into a number of disjoint polyhedrons and the recourse decisions are assumed to be piecewise constant functions on the generated subsets. This modeling approach enables to obtain approximations...
of multistage robust MIP problems, where the quality of approximation is adjusted by the choice of an appropriate partition scheme.

1.2 Our Approach and Contributions

In this study, we design a multistage version of DRSPP, which is based on the related one-stage model in [28] but allows to capture a successive revelation of distributional information to the user. In this regard, we first discuss our construction of the ambiguity set and the objective function for the one-stage problem. Next, we formulate some research questions that need to be addressed when the problem unfolds in stages.

One-stage problem. We consider a weighted connected directed graph $G := (N, A, c)$, where $N$ and $A$ denote its sets of nodes and directed arcs, respectively, and $c = \{c_a, a \in A\}$ is a nonnegative cost vector. We also denote by $s \in N$ and $f \in N$ the source and the destination nodes in $G$, respectively. Then, a standard risk-neutral one-stage DRSPP can be formulated as:

$$\min_{y \in Y} \max_{Q \in \mathcal{Q}} \mathbb{E}_Q[\ell(c, y)],$$

where $Y \subseteq \{0, 1\}^{|A|}$ is a set of all incidence vectors corresponding to feasible $s - f$ paths in $G$, $y \in Y$ is an incidence vector of a path selected by the user and $\ell(c, y)$ is a given loss function. Furthermore, for any fixed $y \in Y$ the attacker attempts to maximize the user’s expected loss by selecting a distribution $Q$ of the cost vector $c$ from a family of distributions $\mathcal{Q}$.

In the related one-stage model in [28], we assume that the user has the following partial information about the distribution $Q$. First, the costs of particular arcs $a \in A$ are subject to individual probability constraints of the form:

$$Q[c_a \in [l_a^{(j)}, u_a^{(j)}]] \in [q_a^{(j)}, \overline{q}_a^{(j)}] \quad \forall j \in \mathcal{D}_a, a \in A,$$

where $[l_a^{(j)}, u_a^{(j)}] \subseteq [l_a, u_a], \ j \in \mathcal{D}_a = \{1, \ldots, d_a\}$, is a finite set of subintervals; $q_a^{(j)}$ and $\overline{q}_a^{(j)}$ are contained in $[0, 1]$ and bound the probability that the random cost $c_a$ belongs to the $i$th subinterval. In particular, we assume that for each $a \in A$ and $j = 1$ the constraints

$$Q[c_a \in [l_a^{(j)}, u_a^{(j)}]] \in [q_a^{(j)}, \overline{q}_a^{(j)}]$$

are support constraints with $l_a^{(j)} = l_a, u_a^{(j)} = u_a$ and $q_a^{(j)} = q_a^{(j)} = 1$ for some $l_a, u_a \in \mathbb{R}_{>0} \cup \{0\}$. Secondly, for a real-valued matrix $B \in \mathbb{R}^{|A| \times k}$ and a vector $b \in \mathbb{R}^k, k \in \mathbb{Z}_+$, we introduce linear expectation constraints of the form:

$$\mathbb{E}_Q[BC] \leq b.$$

Hence, the ambiguity set $\mathcal{Q}$ is formed by all distributions that satisfy the distributional constraints (2) and (3).

\(\copyright\) Springer
We argue that in contrast to the moment-based ambiguity sets [17] or ambiguity sets based on a distance metric from the empirical distribution [2, 18], our distributional constraints allow incomplete knowledge of the training data set. Formally, the individual probability constraints (2) can be constructed by using interval-censored observations with respect to particular arcs; see, e.g., [36]. From the application perspective, the interval uncertainty can be motivated by measurement errors or sensors detection limits in environmental measurements; see, e.g., [30]. At the same time, the linear expectation constraints (3) can be constructed by leveraging linear combinations of arc costs with respect to some subsets of arcs. For example, in our problem setting the user may only have access to random observations of a total cost with respect to some routes in the network; this type of feedback is also known as “bandit” feedback in online learning problem settings [9]. We refer the reader to Sect. 2.2 in [28] and Sect. 2.3 in the current study for further details on the construction of distributional constraints from data.

The key theoretical result of [28] indicates that the one-stage DRSPP of the form (1) with a linear loss function \( \ell(c, y) = c^\top y \) can be reformulated as a robust shortest path problem with a polyhedral uncertainty set; see, e.g., [10] for a review of robust combinatorial optimization problems. Furthermore, the resulting problem admits a linear mixed-integer programming (MIP) reformulation and, thus, can be solved using off-the-shelf MIP solvers.

**Multistage problem.** In the one-stage formulation, the user selects a path here-and-now before the realization of uncertainty, i.e., an unknown distribution \( Q \in \mathcal{Q} \). In the multistage formulation of DRSPP, we, in turn, attempt to address the following research questions:

Q1. *Is there a benefit for the user to alter the chosen path, if it observes some additional distributional information while traversing through the network?*

Q2. *How much can the user gain by leveraging such adaptive decisions?*

Q3. *Can the resulting multistage formulation be solved at hand using off-the-shelf MIP solvers?*

We address a dynamic revelation of distributional information to the user by introducing some auxiliary distributional constraints (in addition to the constraints in \( Q \) that are known to the user a priori) associated with the arcs emanated from a current user’s position. We assume that the auxiliary constraints can be verified by the user while traversing through the network. Simply speaking, the user forms a list of auxiliary constraints at the beginning of the game. Then for each constraint in the list the attacker decides whether this constraint is satisfied or not and reveals its response, i.e., “yes” or “no,” to the user as soon as the user achieves the respective node in the network.

From the practical perspective, the auxiliary distributional constraints can be motivated, e.g., by some additional data observed by the user from Bluetooth sensors [1, 22] or GPS-equipped floating vehicles [38]. In other words, if sensors are placed at particular nodes of the given network, then the user may quantify the travel times between two successive detection stations; see, e.g., [22]. This additional information can be collected dynamically while traversing through the network and used to verify the
auxiliary constraints; we refer to Example 2.2 and Sect. 2.3 for a more comprehensive discussion.

Following the one-stage formulation, we suppose that the user’s objective is to select a path with the least possible expected cost. The attacker, in turn, attempts to maximize the user’s objective function by selecting a distribution of arc costs from the ambiguity set \( Q \). However, in contrast to the one-stage problem, both decision-makers are able to adjust their decisions at particular nodes of the user’s path. In other words, the user’s problem unfolds in stages, where at each stage the attacker reveals (if necessary) some new distributional information to the user and the user picks a subsequent node of its path.

To the best of our knowledge, the concept of using auxiliary distributional constraints is not discussed in the context of multistage DRO problems. The idea of our approach is to partition the initial ambiguity set into a number of disjoint subsets, where each subset corresponds to a vector of attacker’s responses (recall that the attacker provides a binary response to each auxiliary constraint). Then we show that the user’s decision can be viewed as a piecewise constant function on the generated ambiguity sets subject to some non-anticipativity constraints; see, e.g., [3, 20].

In contrast to the study by Sefair et al. [34], we consider more general types of constraints from the attacker’s perspective (instead of a unique budget constraint) and account some partial distributional information available to the user. In contrast to the study by Yu and Shen [43], our model exploits linear expectation constraints of the form (3), which make the uncertain parameters stage-wise dependent and thereby “destroy” dynamic programming-based algorithms. Finally, our solution approach is, in a sense, similar to the one developed by Bertsimas and Dunning [3]. However, from the methodological point of view, the partitions in [3] are not motivated by any practical applications and are used to provide a better approximation for the nominal multistage problem. Meanwhile, the partition of the ambiguity set in our setting is motivated by the auxiliary distributional constraints observed by the user in the considered game.

Our contributions for the multistage model can be summarized as follows:

- We formulate the multistage DRSPP and describe two classes of non-anticipativity constraints, for acyclic and general graphs, respectively (Sects. 2.1 and 2.2).
- We show that the auxiliary distributional constraints can be constructed and verified using some data from Bluetooth sensors (Sect. 2.3).
- Under some mild assumptions, the multistage problem is reformulated as a one potentially large linear MIP problem (Sect. 3). These results address our research question Q3.
- The obtained MIP reformulation is used in our numerical study, where its computational tractability and the quality of adaptive decisions are explored numerically (Sect. 4). Hence, we address the research questions Q1 and Q2.

## 2 Base Model

**Notation.** All vectors and matrices are labeled by bold letters. For a network \( G := (N, A, c) \), we denote by \( N \) and \( A \) its sets of nodes and directed arcs, respectively,
whereas $c$ is a nonnegative cost vector. Let $s$ and $f$ be the source and the destination nodes, respectively. For each node $i \in N$, we refer to $RS_i$ ($FS_i$) as the set of arcs directed into (and out of) node $i$. The space of all probability distributions on $\mathbb{R}^{|A|}$ is denoted as $Q_0(\mathbb{R}^{|A|})$. Finally, for each $a \in A$ we denote by $Q_a \in Q_0(\mathbb{R})$ the marginal distribution induced by a joint distribution $Q \in Q_0(\mathbb{R}^{|A|})$.

2.1 Modeling Assumptions

As outlined in Sect. 1, we consider a directed weighted connected graph $G := (N, A, c)$, where the distribution $Q$ of the cost vector $c$ belongs to an ambiguity set $Q$ given by:

$$Q := \left\{ Q \in Q_0(\mathbb{R}^{|A|}) : Q \text{ satisfies the constraints (2) and (3)} \right\}. \quad (4)$$

Furthermore, we note that the set $Y$ of all feasible path-incidence vectors can be expressed as:

$$Y = \left\{ y \in \{0, 1\}^{|A|} : \sum_{a \in FS_i} y_a - \sum_{a \in RS_i} y_a = \begin{cases} 1, & \text{if } i = s \\ -1, & \text{if } i = f \\ 0, & \text{otherwise} \end{cases} \forall i \in N \right\} \quad (5a)$$

$$\sum_{a \in FS_i} y_a \leq 1 \forall i \in N \right\}. \quad (5b)$$

where the constraints (5a) are standard flow conservation constraints and the constraints (5b) ensure that each node is visited at most once; see, e.g., [37]. We note that cycles are prohibited due to the support constraints, i.e., the probability constraints (2) for each $a \in A$ and $j = 1$, and the selected form of objective criterion in (1).

Throughout the paper, we make the following modeling assumptions:

A1. Each node $i \in N$ is visited by the user at most once.

A2. Both the user and the attacker have complete information about the initial family of distributions $Q$ and the network $G$.

A3. In addition to the distributional constraints in $Q$, the user forms a list $L$ of auxiliary distributional constraints given by:

$$L := \bigcup_{i \in N \setminus \{f\}} \left\{ Q_a [c_a \in [\tilde{\gamma}^{(j)}_a, \tilde{\mu}^{(j)}_a]] \leq \tilde{\gamma}^{(j)}_a \right\} \forall j \in \tilde{D}_a, \forall a \in FS_i ; \quad (6a)$$

$$\mathbb{E}_Q \left\{ \sum_{a \in FS_i} p_j a c_a \right\} \leq p_j 0 \right\} \forall j \in \tilde{K}_i \right\}, \quad (6b)$$

where $\tilde{D}_a = \{1, \ldots, \tilde{d}_a\}$, $a \in A$, and $\tilde{K}_i = \{1, \ldots, \tilde{k}_i\}$, $i \in N \setminus \{f\}$, are potentially empty sets of indexes.

Assumption A1 can be explained, e.g., by a nonstationarity in the attacker’s distribution (a similar assumption is made by Xu and Mannor [42] with regard to
distributionally robust Markov decision processes). Put differently, if the user returns to a node multiple times, then the attacker may simply modify the underlying distribution of arc costs. In this situation, the user cannot exploit the previously collected distributional information and, hence, it is not favorable for the user to visit nodes multiple times.

In the first part of Assumption \( A2 \), we envision that the user has some initial information about the distribution of arc costs. As briefly outlined in Sect. 1.2, this information can be collected beforehand by leveraging some historical data. The second part of Assumption \( A2 \) is rather standard in the robust and distributionally robust optimization literature. Nevertheless, we refer to the studies in [7, 8, 27] for multistage shortest path interdiction models, in which the attacker has incomplete knowledge about the structure of the underlying network and observes the existence and precise costs of particular arcs by observing the user’s decisions.

Assumption \( A3 \) indicates that the auxiliary distributional constraints (\( i \)) have a form similar to the initial distributional constraints in \( Q \) and (\( ii \)) are associated only with the arcs emanated from a node \( i \in N \setminus \{ f \} \). In the following, we demonstrate that the first property, (\( i \)), is necessary for deriving a linear MIP reformulation of the proposed multistage optimization problem. On the other hand, violation of the second property, (\( ii \)), is not favorable for the attacker (as it may allow the user to observe more distributional information and, therefore, effectively mitigate the impact of future attacker’s responses).

In conclusion, we note that the constraints (6a) and (6b) are necessary only if they refine the initial distributional constraints in \( Q \). A possible verification of this property is discussed in more detail within Sect. 3.2. In the following example, we provide some intuition behind the auxiliary distributional constraints (6), as well as the first and the second research questions, \( Q1 \) and \( Q2 \), from Sect. 1.2.

\textbf{Example 2.1} We consider a network \( G \) depicted in Fig. 1 with \( s = 1 \) and \( f = 8 \). The arc costs are supposed to satisfy support constraints outlined inside each arc, i.e., \( c_a \in [l_a, u_a] \) with probability 1. Furthermore, we introduce a unique linear expectation constraint given by:

\[ \mathbb{E}_Q \left\{ \sum_{a \in A'} c_a \right\} \leq 1, \tag{7} \]
where \( A' := \{(2, 4), (2, 5), (3, 6), (3, 7)\} \). Next, we consider three different cases with respect to the additional distributional information observed by the user:

- **Case 1 (no additional information).** If the user picks a path before the realization of \( Q \in Q \), then the user’s path contains exactly one arc from \( A' \) and the attacker may set the expected cost of this arc equal to 1. Then the worst-case expected loss incurred by the user is also equal to 1.

- **Case 2 (full information).** If the user picks a path after the realization of \( Q \in Q \), then the optimal attacker’s decision is to set expected costs of 0.25 for each arc in \( A' \). Then the worst-case expected loss of the user equals 0.25 (this value can be also viewed as a lower bound on the user’s expected loss).

- **Case 2 (successive revelation of information).** Assume that, if the user stays at node 2, then it may compare the expected costs of \((2, 4)\) and \((2, 5)\). In other words, the attacker indicates whether the following linear expectation constraint holds or not:

\[
\mathbb{E}_Q\{c_{(2,4)} - c_{(2,5)}\} \leq 0.
\]

In this case, it is optimal for the user to traverse through \((1, 2)\) and then to pick an arc with the smallest expected cost. In fact, the expected cost of the latter arc cannot exceed 0.5 due to the initial constraint (7), which results in the worst-case expected loss of 0.5.

Comparing the first and the third cases, we conclude that the user may benefit from using dynamic decisions. Furthermore, in view of Case 2, the user’s expected loss in our setting can be bounded from below by an optimal objective function value of the max-min problem, i.e., the one-stage problem (1) where the order of “max” and “min” operators is reversed.

\[\Box\]

### 2.2 Multistage Formulation

Following our discussion in Sect. 1.1, we encode all feasible attacker’s responses to the constraints in \( L \) with binary vectors \( r_j \in \{0, 1\}^{\left| L \right|} \) for \( j \in \{1, \ldots, 2^{|L|}\} \). In particular, for any fixed \( j \) we observe that \( r_{jm} = 1 \), if the \( m \)th constraint in \( L \) is satisfied and \( r_{jm} = 0 \), otherwise. For each vector of attacker’s responses \( r_j, j \in \{1, \ldots, 2^{|L|}\} \), we construct a new ambiguity set \( Q_j \) that combines the initial distributional constraints in \( Q \) with the auxiliary constraints induced by \( r_j \). For ease of exposition, we refer to “\( v \geq w \)” as an opposite constraint to “\( v \leq w \).” Formally, for each \( j \in \{1, \ldots, 2^{|L|}\} \) we have:

\[
Q_j := Q \cap \left\{ Q \in Q_0(\mathbb{R}^{|A|}) \mid \right. \\
\left. \begin{align*}
\text{the } m \text{th constraint in } L, & \text{ if } r_{jm} = 1, \\
\text{an opposite of the } m \text{th constraint}, & \text{ if } r_{jm} = 0,
\end{align*} \right. \forall m \in \{1, \ldots, |L|\}.
\]

Then, irrespective of the order of auxiliary distributional constraints in \( L \), the multistage DRSPP can be formulated as follows:
\[
\max_{j \in \{1, \ldots, 2^{\vert L \vert}\}} \min_{y_j \in Y} \max_{Q_j \in Q_j} \mathbb{E} Q_j \{ c^\top y_j \}
\]

s.t. non-anticipativity constraints with respect to \(y_j, j \in \{1, \ldots, 2^{\vert L \vert}\}\),

where \(y_j, j \in \{1, \ldots, 2^{\vert L \vert}\}\), denotes a decision of the user under complete knowledge of the vector of attacker’s responses \(r_j\) or, equivalently, the associated ambiguity set \(Q_j\). Formally, in \((F_{ms})\) the user attempts to minimize its worst-case expected loss taking into account all possible realizations of attacker’s responses and subject to some non-anticipativity constraints. We need to enforce non-anticipativity as long as the user learns the attacker’s responses associated with a node \(i \in N \setminus \{f\}\) only when it reaches node \(i\). In other words, for any fixed \(j, \ell \in \{1, \ldots, 2^{\vert L \vert}\}\) the user’s paths \(P_j\) and \(P_\ell\) (induced by the incidence vectors \(y_j\) and \(y_\ell\), respectively) must coincide whenever the user is not able to distinguish between the ambiguity sets \(Q_j\) and \(Q_\ell\).

First, we construct non-anticipativity constraints for acyclic graphs. In this regard, for any fixed \(j, \ell \in \{1, \ldots, 2^{\vert L \vert}\}\), \(j \neq \ell\), we denote by \(N_{j, \ell} \subseteq N\) a set of nodes at which the user may learn the actual ambiguity set, either \(Q_j\) or \(Q_\ell\), that is enforced by the attacker. More specifically, for every node in \(N_{j, \ell}\) there exists an associated distributional constraint in \(L\) such that the attacker’s responses to this constraint in \(r_j\) and \(r_\ell\) are different. As a remark, in Example 2.1 we have \(|L| = 1\) and \(N_{1,2} = \{2\}\).

The following result holds.

**Proposition 2.1** Assume that \(j, \ell \in \{1, \ldots, 2^{\vert L \vert}\}, j \neq \ell\), and let \(G\) be a directed acyclic graph. Then for each node \(i \in N \setminus N_{j, \ell}\) the following constraints ensure non-anticipativity:

\[
y_j, a = y_\ell, a \quad \forall a \in FS_i, \text{ if all nodes in } N_{j, \ell} \text{ are reachable from } i, \quad (8a)
\]

\[
|y_j, a - y_\ell, a| \leq \sum_{n \in \tilde{N}_{j, \ell}} \sum_{a' \in FS_n} y_{j, a'} \quad \forall a \in FS_i, \text{ if a subset of nodes } \tilde{N}_{j, \ell} \subseteq N_{j, \ell} \text{ is not reachable from } i. \quad (8b)
\]

**Proof** We consider three particular situations:

(i) Assume that for a node \(i \in N \setminus N_{j, \ell}\) all nodes in \(N_{j, \ell}\) are reachable from \(i\). Then any node in \(N_{j, \ell}\) can be visited by the user only after \(i\) since the graph \(G\) is acyclic. Hence, at node \(i\) we have a lack of information to distinguish between the ambiguity sets \(Q_j\) and \(Q_\ell\), which results in the constraints \((8a)\).

(ii) Assume that for a node \(i \in N \setminus N_{j, \ell}\) a subset of nodes \(\tilde{N}_{j, \ell} \subseteq N_{j, \ell}\) is not reachable from \(i\). In view of the first part of non-anticipativity constraints \((8a)\) and the definition of \(\tilde{N}_{j, \ell}\), it is rather straightforward to check that either both paths induced by \(y_j\) and \(y_\ell\) contain a node in \(\tilde{N}_{j, \ell}\), or both do not. In fact, non-anticipativity at node \(i\) is needed only if the user’s path induced by \(y_j\), say \(P_j\), does not contain any node in \(\tilde{N}_{j, \ell}\), i.e.,

\[
\sum_{n \in \tilde{N}_{j, \ell}} \sum_{a' \in FS_n} y_{j, a'} = 0,
\]

\(\Box\) Springer
Fig. 2 A pair of user’s decisions satisfying the non-anticipativity requirement for some \( j, \ell \in \{1, \ldots, 2^{|L|}\} \). The set \( N_{j, \ell} \) is assumed to contain two nodes, 2 and 4, depicted in bold. Some feasible values of \( t_j \) are depicted outside the nodes of \( P_j \).

see constraints (8b). Otherwise, there exists a node \( n \in P_j \cap \tilde{N}_{j, \ell} \) and we have either \( i \not\in P_j \) or \( i \in P_j \) and \( i \) is visited after \( n \); recall the definition of \( \tilde{N}_{j, \ell} \). In both cases, the constraints (8a) and the flow conservation constraints (5a) are sufficient to guarantee non-anticipativity and, thus, we make the constraints (8b) non-binding.

(iii) If the user stays at a node \( i \in N_{j, \ell} \), then it may distinguish between the ambiguity sets \( Q_j \) and \( Q_\ell \). Hence, we do not need to enforce non-anticipativity constraints.

These observations conclude the proof.

We note that reachability in the formulation of Proposition 2.1 can be checked efficiently for each pair of nodes, e.g., by Floyd–Warshall algorithm; see [14]. Furthermore, constraints (8a) and (8b) are, in fact, linear constraints with respect to the decision variables.

Unfortunately, the non-anticipativity constraints become substantially more complicated for general graphs containing directed cycles. For example, there may exist a cycle containing a node \( i \in N \setminus N_{j, \ell} \) and \( i' \in N_{j, \ell} \) for some fixed \( j, \ell \in \{1, \ldots, 2^{|L|}\} \). In this case, we need to enforce non-anticipativity at node \( i \) only if node \( i' \) is visited by the user after \( i \). For example, in Fig. 2 we consider a general graph, where the non-anticipativity requirement (8a) for acyclic graphs is violated. Indeed, all nodes in \( N_{j, \ell} = \{2, 4\} \) are reachable from node 5, but we do not need to enforce non-anticipativity at this node, as node 2 is visited before.

In order to resolve the outlined issue, we introduce new variables \( t_j \in \mathbb{R}^{|N|} \), \( j \in \{1, \ldots, 2^{|L|}\} \), related to a sequence at which nodes are visited by the user under scenario \( j \); see constraints (9a)–(9c) used in the formulation of Proposition 2.2. For instance, in Fig. 2 some feasible values of \( t_j \) associated with the user’s path \( P_j \) are depicted. As a remark, similar variables and constraints are used in the Miller–Tucker–Zemlin (MTZ) formulation of the traveling salesmen problem [31].

We formulate the following result for general graphs.

Proposition 2.2 Assume that \( j, \ell \in \{1, \ldots, 2^{|L|}\}, j \neq \ell, \) and the graph \( G \) is general. Then for each node \( i \in N \setminus N_{j, \ell} \), the following constraints ensure non-anticipativity:

\[
t_{j,s} = 0 \quad (9a)
\]
\begin{align}
0 \leq t_{j,i} & \leq |N| - 1 \quad \forall i \in N \tag{9b} \\
t_{j,a_1} - t_{j,a_2} & \leq -1 + |N|(1 - y_{j,a}) \quad \forall a \in A \tag{9c} \\
|y_{j,a} - y_{\ell,a'}| & \leq \sum_{n \in N_{j,\ell}} \min \left\{ \max\{t_{j,i} - t_{j,n}; 0\} + 2 - \sum_{a' \in F_{S_n}} y_{j,a'} - \sum_{a' \in F_{S_i}} y_{j,a'}; \sum_{a' \in F_{S_n}} y_{j,a'} \right\} \quad \forall a \in F_{S_i}. \tag{9d}
\end{align}

**Proof** Henceforth, we fix \( j, \ell \in \{1, \ldots, 2|L|\}, j \neq \ell \). First, we note that the decision vector \( y_j \in Y \) induces a simple \( s - f \) path

\[ P_j = \{n_1 \rightarrow n_2 \rightarrow \ldots \rightarrow n_{T+1}\}, \]

where \( n_1 = s, n_{T+1} = f \) and any intermediate node is visited at most once. In fact, constraints (9a)–(9c) assign labels \( t_{j,i} \) to each node \( i \in N \) so that the label at \( s \) is equal to 0 and the labels at each subsequent node in \( P_j \) are incremented at least by one compared to the predecessor node. Each label is also bounded from above by \( |N| - 1 \), which is the maximal possible length of a simple \( s - f \) path in \( G \).

Next, we note that the user’s paths \( P_j \) and \( P_\ell \) (induced by the incidence vectors \( y_j \) and \( y_\ell \), respectively) must coincide until any node \( n \in N_{j,\ell} \) is reached. This observation implies that non-anticipativity constraints are necessary only for the nodes of \( P_j \) and \( P_\ell \) that are visited before \( n \). We consider two particular cases:

1. Assume that the user stays at a node \( i \in N \setminus N_{j,\ell} \). Then non-anticipativity at \( i \) is needed only if all nodes in \( N_{j,\ell} \) are either not contained in \( P_j \) or there exists \( n \in N_{j,\ell} \) that is visited after node \( i \) assuming that \( i \in P_j \). Hence, for each node \( n \in N_{j,\ell} \) we verify whether this node is contained in \( P_j \) and whether it is visited before or after node \( i \).

Formally, we calculate the following indicators:

\[ w_{j,i,n} = \min \left\{ \max\{t_{j,i} - t_{j,n}; 0\} + 2 - \sum_{a' \in F_{S_n}} y_{j,a'} - \sum_{a' \in F_{S_i}} y_{j,a'}; \sum_{a' \in F_{S_n}} y_{j,a'} \right\}. \]

We observe that, if \( n \) is not contained in \( P_j \), then \( \sum_{a' \in F_{S_n}} y_{j,a'} = 0 \) and, hence, \( w_{j,i,n} = 0 \); recall that \( y_j \) must satisfy (5b). Otherwise, if \( n \) is contained in \( P_j \), then

\[ w_{j,i,n} = \min \left\{ \max\{t_{j,i} - t_{j,n}; 0\} + 1 - \sum_{a' \in F_{S_i}} y_{j,a'}; 1 \right\}. \]

In this case, \( w_{j,i,n} = 0 \) only if \( i \in P_j \), i.e., \( \sum_{a' \in F_{S_i}} y_{j,a'} = 1 \), and \( n \) is visited after \( i \), i.e., \( t_{j,i} \leq t_{j,n} - 1 \). Otherwise, if \( n \) is visited before \( i \) or \( i \not\in P_j \), then \( w_{j,i,n} = 1 \). As a result, we conclude that non-anticipativity is guaranteed by the constraints (9d).
(ii) If the user stays at a node \( i \in N_{j, \ell} \), then it may distinguish between the ambiguity sets \( Q_j \) and \( Q_\ell \). Hence, we do not need to enforce non-anticipativity constraints. These observations conclude the proof. \( \square \)

Similar to Proposition 2.1 the non-anticipativity constraints (9a)–(9d) can be reformulated as linear constraints but after applying some standard linearization techniques. We discuss this point as well as a linear MIP reformulation of \((F_{ms})\) in Sect. 3. In the remainder of this section, we explore how the auxiliary distributional constraints given by (6) can be constructed from real data observations.

### 2.3 Construction of the Auxiliary Constraints from Data

The results of this section are motivated and similar to the related results of Sect. 2.2 in [28], where the initial distributional constraints in \( Q \) are constructed from data. In the following, we assume that the attacker fixes a nominal distribution \( Q^* \in Q \) and has access to a data set given by \( \hat{n} \in \mathbb{Z}_{>0} \) independent observations obtained from this distribution, i.e.,

\[
\hat{C} := \left\{ (\hat{c}^{(k)}_1, \ldots, \hat{c}^{(k)}_{|A|})^\top, \; k \in \{1, \ldots, \hat{n}\} \right\}.
\]  

We also note that an optimal solution of \((F_{ms})\) provides a user’s decision for every possible sequence of attacker’s responses.

A natural question arising is how the user can identify the actual sequence of attacker’s responses. In this regard, after solving the multistage formulation \((F_{ms})\) we exploit the data set (10) to verify the auxiliary constraints (6) dynamically at the respective nodes of the user’s path. This procedure is referred to as a constraint verification procedure.

**Verification of probability constraints.** Suppose that the user stays at a node \( i \in N \setminus \{f\} \) and needs to verify a probability constraint of the form

\[
q^*_a := Q^*_a(c_a \in [\underline{u}_a, \overline{u}_a]) \leq \tilde{q}_a
\]  

for some \( a \in FS_i \). From Hoeffding inequality [24], we observe that for any \( \varepsilon > 0 \)

\[
\Pr \left\{ |q^*_a - \frac{1}{\hat{n}} \sum_{k=1}^{\hat{n}} \chi_{a,k}| \geq \varepsilon \right\} \leq 2 \exp(-2\hat{n}\varepsilon^2),
\]  

where \( \frac{1}{\hat{n}} \sum_{k=1}^{\hat{n}} \chi_{a,k} \) refers to an empirical probability that \( c_a \in [\underline{u}_a, \overline{u}_a] \), i.e.,

\[
\chi_{a,k} = \begin{cases}
1, & \text{if } \hat{c}_a^{(k)} \in [\underline{u}_a, \overline{u}_a] \\
0, & \text{otherwise,}
\end{cases}
\]
for each \( k \in \{1, \ldots, \hat{n}\} \). Hence, with any prescribed confidence level \( \gamma \in (0, 1) \) we may guarantee that

\[
q^*_a \in \left[ \frac{1}{\hat{n}} \sum_{k=1}^{\hat{n}} \chi_{a,k} - \varepsilon, \frac{1}{\hat{n}} \sum_{k=1}^{\hat{n}} \chi_{a,k} + \varepsilon \right],
\]

where the parameter \( \varepsilon \) is defined by setting the right-hand side of (12) equal to \( 1 - \gamma \). As a remark, using the same arguments one may construct the initial probability constraints (2).

Furthermore, if the number of samples, \( \hat{n} \), is sufficiently large (and, therefore, \( \varepsilon \) is sufficiently small), then we may distinguish between the following two alternatives:

\[
\frac{1}{\hat{n}} \sum_{k=1}^{\hat{n}} \chi_{a,k} + \varepsilon \leq \tilde{q}_a \quad \text{or} \quad \frac{1}{\hat{n}} \sum_{k=1}^{\hat{n}} \chi_{a,k} - \varepsilon > \tilde{q}_a.
\]

In the former case the constraint (11) holds with probability of at least \( \gamma \); in the latter case it is violated with probability of at least \( \gamma \). Importantly, in order to verify a constraint of the form (11), it suffices to know only the number of random samples that belong to the interval \([\tilde{l}_a, \tilde{u}_a]\). Also, if the constraint cannot be verified with \( \hat{n} \) samples, then one may generate the attacker’s response at random or assume the worst-case scenario for the user. We exploit a variation of this idea within our numerical study in Sect. 4.

**Verification of linear expectation constraints.** In a similar way, one may verify a linear expectation constraint of the form:

\[
\mathbb{E}_{Q^*} \left\{ \sum_{a \in FS_i} p_a c_a \right\} \leq p_0
\]

for some \( p_a \in \mathbb{R}, a \in FS_i \), and \( p_0 \in \mathbb{R} \). That is, by Hoeffding inequality we observe that:

\[
\Pr \left\{ \left| \mathbb{E}_{Q^*} \left\{ \sum_{a \in FS_i} p_a c_a \right\} - \frac{1}{\hat{n}} \sum_{k=1}^{\hat{n}} \sum_{a \in FS_i} p_a \hat{c}_a^{(k)} \right| \geq \varepsilon \right\} \leq 2 \exp \left( -2 \hat{n} \frac{\varepsilon^2}{\overline{S}_i - \underline{S}_i} \right),
\]

where \( \overline{S}_i = \sum_{a \in FS_i} \max_{c_a \in [l_a, u_a]} p_a c_a \), \( S_i = \sum_{a \in FS_i} \min_{c_a \in [l_a, u_a]} p_a c_a \) and \( i \in \mathbb{N}\backslash\{f\} \). Then for any confidence level \( \gamma \in (0, 1) \) we may guarantee that:

\[
\mathbb{E}_{Q^*} \left\{ \sum_{a \in FS_i} p_a c_a \right\} \in \left[ \frac{1}{\hat{n}} \sum_{k=1}^{\hat{n}} \sum_{a \in FS_i} p_a \hat{c}_a^{(k)} - \varepsilon, \frac{1}{\hat{n}} \sum_{k=1}^{\hat{n}} \sum_{a \in FS_i} p_a \hat{c}_a^{(k)} + \varepsilon \right],
\]

where \( \varepsilon \) is defined by setting the right-hand side of (14) equal to \( 1 - \gamma \). As in the previous case, the above procedure can be slightly modified to construct the initial expectation constraints (3).
We note that the constraint (13) can be verified following the same arguments as for the probability constraint (11). However, in contrast to (12), the inequality in (14) exploits linear combinations of arc costs with respect to the arcs \( a \in FS_i \). In the following example we demonstrate how the auxiliary constraints (6a) and (6b) can be constructed and verified by using information from Bluetooth sensors.

**Example 2.2** We envision a network, where the arc costs/travel times are measured using Bluetooth sensors; see, e.g., [1, 22]. The sensors are placed at particular nodes of the network and one may quantify travel times between two successive sensors; see Fig. 3. For example, in Fig. 3a the sensors are placed at nodes \( O, F \) and \( K \), which yields that the user may receive random observations of individual arc travel times \( c_{OF} \) and \( c_{OK} \). On the other hand, in Fig. 3b and c the user may only observe some linear combinations of individual arc travel times (we additionally assume that \( c_{OF} = c_{FO} \) in Fig. 3c).

Then, by leveraging the observed linear combinations of arc travel times and the constraint verification procedure, the user may verify linear expectation constraints of the form (6b). Analogously, if random observations of individual arc travel times are subject to interval uncertainty, then one may verify probability constraints of the form (6a).

We conclude that the auxiliary distributional constraints (6a) and (6b) can be motivated in a rather straightforward way by the outlined sensor-related application context. Some further intuition behind the choice of the auxiliary distributional constraints (6a) and (6b) is implied by our solution procedure for the multistage problem \( (F_{ms}) \) and is discussed within Sect. 3.

### 3 Solution Approach

This section is organized as follows. First, in Sect. 3.1 we briefly describe the key theoretical results for the one-stage model in [28]. Then in Sect. 3.2 we use the aforementioned results as well as Propositions 2.1 and 2.2 to reformulate the multistage DRSPP \( (F_{ms}) \) as a linear MIP problem.
3.1 MIP Reformulation of the One-Stage Problem

We recall that the one-stage DRSPP in [28] can be expressed as:

$$z_{\text{static}}^* := \min_{y \in Y} \max_{Q \in \mathcal{Q}} \mathbb{E}_Q\{c^\top y\}, \quad \text{(F\textsubscript{os})}$$

where the ambiguity set $\mathcal{Q}$ and the set of feasible decisions $Y$ are given by (4) and (5), respectively. The optimization problem (F\textsubscript{os}) is also referred to as a static problem and its optimal objective function value is denoted as $z_{\text{static}}^*$.

It turns out that (F\textsubscript{os}) can be recast as a robust shortest path problem with a polyhedral uncertainty set. More specifically, the following result holds.

**Theorem 3.1** (Theorem 2 in [28]) Assume that

$$\mathcal{S}_0 := \{\bar{c} \in \mathbb{R}^{|A|} : \mathbf{L} \leq \bar{c} \leq \mathbf{U}; \quad \mathbf{B} \bar{c} \leq \mathbf{b}\},$$

where for each $a \in A$

$$L_a := \min_{Q_a \in \tilde{\mathcal{Q}}_a} \mathbb{E}_{Q_a}\{c_a\}, \quad \text{(15a)}$$
$$U_a := \max_{Q_a \in \tilde{\mathcal{Q}}_a} \mathbb{E}_{Q_a}\{c_a\}, \quad \text{(15b)}$$

and

$$\tilde{\mathcal{Q}}_a := \left\{ Q_a \in \mathcal{Q}_0(\mathbb{R}) : Q_a\{c_{a} \in [l_{a}^{(i)}, u_{a}^{(i)}]\} \in [q_{a}^{(i)}, q_{a}^{(i)}] \quad \forall i \in \mathcal{D}_a \right\}.$$

Then the DRSPP of the form (1) is equivalent to the following robust shortest path problem with polyhedral uncertainty:

$$\min_{y \in Y} \max_{\bar{c} \in \mathcal{S}_0} \bar{c}^\top y. \quad \text{(16)}$$

The intuition behind Theorem 3.1 can be explained as follows. Since the objective function in (F\textsubscript{os}) is linear, it can be seen as a function of expected costs, $\bar{c} := \mathbb{E}_Q\{c\}$, i.e.,

$$\mathbb{E}_Q\{c^\top y\} = (\mathbb{E}_Q\{c\})^\top y = \bar{c}^\top y.$$

One may also show that the individual probability constraints (2) for each $a \in A$ can be expressed in terms of box constraints

$$L_a \leq \bar{c}_a \leq U_a$$

with respect to the expected costs $\bar{c}_a$, which yields the desired result.

In order to resolve the moment problems (15a) and (15b), we make the following additional assumptions:
A4. For each $a \in A$, there exists a marginal distribution $Q_a \in Q_0(\mathbb{R})$ such that 

$$Q_a[l_a^{(i)} \leq c_a \leq u_a^{(i)}] \in (q_a^{(i)}, \overline{q}_a^{(i)})$$

whenever $q_a^{(i)} < \overline{q}_a^{(i)}$, $i \in D_a$.

A5. For each $a \in A$ and any pair of subintervals in (2), namely $[l_a^{(i_1)}, u_a^{(i_1)}]$ and $[l_a^{(i_2)}, u_a^{(i_2)}]$, $i_1, i_2 \in D_a$, we have $l_a^{(i_1)} \neq u_a^{(i_2)}$ and $l_a^{(i_2)} \neq u_a^{(i_1)}$.

Simply speaking, Assumptions A4 and A5 guarantee that strong duality for the moment problems (15a) and (15b) holds. This prerequisite allows us to obtain finite linear programming reformulations of (15a) and (15b); we refer to Lemma 2 in [28] for further details.

Summarizing the discussion above, the one-stage formulation in [28] can be tackled by solving $2|A|$ linear programming problems and a single robust shortest path problem (16). Importantly, the latter problem can be recast as a linear MIP problem by dualizing the second-level linear programming problem. Formally, the following result holds.

Proposition 3.1 (Theorem 3 in [28]) Let 

$$S_0 := \left\{ \overline{c} \in \mathbb{R}^{|A|} : L \leq \overline{c} \leq U; \ B\overline{c} \leq b \right\} = \left\{ \overline{c} \in \mathbb{R}^{|A|} : B_0\overline{c} \leq b_0 \right\}. \quad (17)$$

Then the one-stage DRSPP ($F_{os}$) admits the following mixed-integer programming reformulation:

$$z^{*}_{static} = \min_{y, \lambda} \left\{ b_0^T \lambda : \lambda \geq 0, \ -y + B_0^T \lambda = 0, \ y \in Y \right\}. \quad (18)$$

3.2 MIP Reformulation of the Multistage Problem

The key observation that we use next is that each family of distributions $Q_j$, $j \in \{1, \ldots, 2^{|L|}\}$ (corresponding to a vector of attacker’s responses $r_j \in \{0, 1\}^{|L|}$), contains the same types of distributional constraints as those in $Q$; recall Assumption A3. Therefore, in view of Theorem 3.1, each ambiguity set $Q_j$, $j \in \{1, \ldots, 2^{|L|}\}$, can be seen as a polyhedral uncertainty set 

$$S_j := \left\{ \overline{c} \in \mathbb{R}^{|A|} : B_j\overline{c} \leq b_j \right\} \subseteq S_0 \quad (19)$$

in terms of expected costs. Next, we provide a linear MIP reformulation of the multi-stage problem ($F_{ms}$).

Theorem 3.2 Let $G$ be a general graph and assume that each ambiguity set $Q_j$, $j \in \{1, \ldots, 2^{|L|}\}$, is described by the polyhedral uncertainty set $S_j$ given by (19). Also, set $M_1 = |N| - 1$. Then the multistage DRSPP ($F_{ms}$) can be reformulated as the following linear MIP problem:

$$z^{*}_{dynamic} = \min_{y, t, v, w, z} \quad (20a)$$
\begin{align*}
t_{j,s} &= 0 \\
\text{s.t. } 0 &\leq t_{j,i} \leq |N| - 1 \quad \forall i \in N \\
t_{j,a_1} - t_{j,a_2} &\leq -1 + |N|(1 - y_{j,a}) \quad \forall a \in A \\
\forall j \in \{1, \ldots, 2^{|\mathcal{L}|}\} \quad (20b) \tag{20b}
\end{align*}

\begin{align*}
v_{j,i,n} &\geq 0 \\
v_{j,i,n} &\geq t_{j,i} - t_{j,n} \\
v_{j,i,n} &\leq M_1 \tilde{v}_{j,i,n} \\
v_{j,i,n} &\leq t_{j,i} - t_{j,n} + M_1 (1 - \tilde{v}_{j,i,n}) \\
w_{j,i,n} &\leq v_{j,i,n} + 2 - \sum_{a' \in FS_n} y_{j,a'} - \sum_{a' \in FS_i} y_{j,a'} \\
w_{j,i,n} &\leq \sum_{a' \in FS_n} y_{j,a'} - \sum_{a \in FS_i} w_{j,i,n} - \sum_{n \in N_{j,\ell}} w_{j,i,n} - \sum_{n \in N_{j,\ell}} \tilde{w}_{j,i,n} \quad \forall a \in FS_i \\
\tilde{v}_{j,i,n} &\in \{0, 1\} \\
\forall j, \ell \in \{1, \ldots, 2^{|\mathcal{L}|}\}, \\
\forall n \in N_{j,\ell,}, \\
\forall i \in N \setminus N_{j,\ell}, \\
j \neq \ell \quad (20c) \tag{20c}
\end{align*}

\begin{align*}
z &\geq b_j^T \lambda_j \\
- y_j + B_j^T \lambda_j &= 0 \\
\lambda_j &\geq 0 \\
y_j &\in Y \\
\forall j \in \{1, \ldots, 2^{|\mathcal{L}|}\}. \quad (20d) \tag{20d}
\end{align*}

**Proof** First, we provide an epigraph reformulation of \((\mathbf{F}_{ms})\) by using an auxiliary variable \(z \in \mathbb{R}\), that is,

\begin{align*}
\min z \\
\text{s.t. non-anticipativity constraints (9),} \quad (21b) \\
z &\geq \min \max_{y_j \in Y} \mathbb{E}_{Q_j} \{c^T y_j\} \quad \forall j \in \{1, \ldots, 2^{|\mathcal{L}|}\}. \quad (21c) \tag{21c}
\end{align*}

Next, the minimum in the right-hand side of (21c) indicates that for each \(j \in \{1, \ldots, 2^{|\mathcal{L}|}\}\) there exists some \(\tilde{y}_j \in Y\) such that

\[ z \geq \max_{Q_j \in Q_j} \mathbb{E}_{Q_j} \{c^T \tilde{y}_j\}. \]

Hence, the path flow constraints \(y_j \in Y\) can be shifted to the first-level problem constraints by omitting the minimum in the right-hand side of (21c). In other words, the optimization problem (21) admits the following equivalent reformulation:

\begin{align*}
\min z \\
\text{s.t. non-anticipativity constraints (9),} \quad (22b) \tag{22b}
\end{align*}
\[ z \geq \max_{Q_j \in Q_j} \mathbb{E}_{Q_j} \{ c^\top y_j \} \quad \forall j \in \{1, \ldots, 2^{|L|}\} \quad (22c) \]

\[ y_j \in Y \quad \forall j \in \{1, \ldots, 2^{|L|}\}. \quad (22d) \]

Then, in view of Theorem 3.1, the constraints (22c) for each \( j \in \{1, \ldots, 2^{|L|}\} \) can be reformulated as polyhedral constraints with respect to expected costs \( \overline{c}_j = \mathbb{E}_{Q_j} \{ c \} \), i.e.,

\[ z \geq \max_{\overline{c}_j \in \mathcal{S}_j} \overline{c}_j^\top y_j. \quad (23) \]

Note that the maximization problem in the right-hand side of (23) is a linear program. Therefore, by strong duality we have:

\[ \max_{\overline{c}_j \in \mathcal{S}_j} \overline{c}_j^\top y_j = \min_{y_j, \lambda_j} \left\{ b_j^\top \lambda_j : \lambda_j \geq 0, \ -y_j + B_j^\top \lambda_j = 0 \right\}. \]

This observation and eliminating the minimum in the dual problem yields the following reformulation of \((F_{ms})\):

\[
\begin{align}
\min_{y, z} & \quad z \\
\text{s.t.} & \quad \text{non-anticipativity constraints (9)}, \\
& \quad z \geq b_j^\top \lambda_j \\
& \quad -y_j + B_j^\top \lambda_j = 0 \\
& \quad \lambda_j \geq 0 \\
& \quad y_j \in Y \\
& \quad \forall j \in \{1, \ldots, 2^{|L|}\}. \quad (24c)
\end{align}
\]

Finally, we linearize the non-anticipativity constraints (9) defined in Proposition 2.2. In this regard, for any fixed \( j, \ell \in \{1, \ldots, 2^{|L|}\}, n \in N_{j, \ell} \) and \( i \in N \setminus N_{j, \ell} \) we introduce new variables \( v_{j, i, n} \) and \( w_{j, i, n} \) such that:

\[
\begin{align}
v_{j, i, n} &= \max \{0; t_{j,i} - t_{j,n}\} \quad (25a) \\
w_{j, i, n} &= \min \left\{ v_{j, i, n} + 2 - \sum_{a' \in FS_n} y_{j,a'} - \sum_{a' \in FS_i} y_{j,a'}; \ \sum_{a' \in FS_n} y_{j,a'} \right\}. \quad (25b)
\end{align}
\]

The maximum in (25a) can be linearized by introducing new binary variables \( \tilde{v}_{i, j, n} \in \{0, 1\} \) and the following linear constraints [41]:

\[
\begin{align}
v_{j, i, n} &\geq 0 \\
v_{j, i, n} &\geq t_{j,i} - t_{j,n} \\
v_{j, i, n} &\leq M_1 \tilde{v}_{i, j, n} \\
v_{j, i, n} &\leq t_{j,i} - t_{j,n} + M_1 (1 - \tilde{v}_{j, i, n}).
\end{align}
\]
Specifically, as long as $|t_{j,i} - t_{j,n}| \leq |N| - 1$, we set $M_1 = |N| - 1$.

Next, the constraint (9d) implies that the sum of $w_{j,i,n}$ over $n \in N_{j,i}$ is bounded from below. Hence, in order to linearize the minimum in (25b) it is sufficient to enforce that

$$w_{j,i,n} \leq v_{j,i,n} + 2 - \sum_{a' \in FS_n} y_{j,a'} - \sum_{a' \in FS_i} y_{j,a'}$$

$$w_{j,i,n} \leq \sum_{a' \in FS_n} y_{j,a'}.$$

As a result, the multistage DRSPP ($F_{ms}$) can be expressed as a linear MIP problem (20).

Theorem 3.2 addresses the research question Q3 formulated in Sect. 1.2. Thus, the multistage DRSPP ($F_{ms}$) can be solved at hand by using off-the-shelf MIP solvers even for graphs that contain cycles. In particular, a linear MIP reformulation for acyclic graphs can be derived from (20) by using the non-anticipativity constraints (8) instead of the constraints (20b) and (20c). We point out that, if the number of auxiliary constraints, $|\mathcal{L}|$, is fixed, then the number of variables and constraints in (20) is polynomial in the size of the network.

In addition, we observe that the auxiliary constraints (6) refine the initial distributional constraints in $Q$ only if the resulting polyhedral uncertainty sets (19) are non-empty. This property can be readily checked by solving the related linear feasibility problems for each polyhedron. Meanwhile, Theorem 3.2 remains valid even if some polyhedral uncertainty sets are empty. More precisely, if a set $S_j$ is empty for some $j \in \{1, \ldots, 2|\mathcal{L}|\}$, then

$$\min_{y_j, \lambda_j} \left\{ b_j^T \lambda_j : \lambda_j \geq 0, -y_j + B_j^T \lambda_j = 0 \right\} = -\infty$$

by strong duality and, therefore, the associated constraints (20d) are non-binding.

Finally, Theorem 3.2 is also applicable to some other user’s objective criteria, for which the constraints (22c) admit a linear programming dual reformulation. In view of our results for the one-stage model in [28], the user may potentially optimize the conditional value at risk [33] or some other optimized certainty equivalent risk measures [23, 40]. We leave these extensions as a possible direction of future research.

4 Computational Experiments

In this section, we attempt to answer the research questions Q1 and Q2, i.e., we explore whether it is favorable for the user to employ dynamic decisions and how much profit the user can gain by leveraging such decisions in place of static decisions provided by the one-stage formulation ($F_{os}$). In this regard, we analyze the role of the auxiliary distributional constraints (6) in relation to both the quality of dynamic decisions and tractability of the MIP reformulation (20).
The remainder of this section is organized as follows. In Sect. 4.1, we introduce some performance metrics to assess the quality of dynamic decisions. Section 4.2 describes our construction of test instances including the structure of the network and distributional constraints. Finally, in Sect. 4.3 we provide our numerical results and the related discussion. All experiments are performed on a PC with CPU i5-7200U and RAM 8 GB. MIP problems are solved in Java with CPLEX 20.1.

4.1 Performance Metrics

In view of our discussion in Sect. 2.3, we propose the following two-step procedure for validation of our approach. In the first step, we construct the initial ambiguity set \( Q \) and solve the MIP reformulation (20) to obtain an optimal user’s decision for every possible vector of attacker’s responses. In the second step (which is referred to as a constraint verification procedure), we attempt to identify the actual sequence of attacker’s responses by verifying the auxiliary distributional constraints (6) dynamically at the respective nodes of the user’s path.

The quality of dynamic decisions in the first step is estimated using a relative gap of the form:

\[
\rho_1 := 100 \times \frac{z^*_{\text{static}} - z^*_{\text{dynamic}}}{z^*_{\text{static}} - z_{\text{lower}}},
\]

where \( z_{\text{lower}} \) provides a lower bound on the optimal objective function value of the multistage problem \( (F_{ms}) \). Following Example 2.1, we select \( z_{\text{lower}} \) as an optimal objective function value of the related max-min problem, i.e.,

\[
z^*_{\text{dynamic}} \geq z_{\text{lower}} := \max_{Q \in \mathcal{Q}} \min_{y \in \mathcal{Y}} E_Q \{ c^\top y \}.
\]

Using the proof of Theorem 3.1, it is quite easy to show that the max-min problem in (26) admits a linear programming reformulation (the details are omitted for brevity).

The relative gap \( \rho_1 \in [0, 100] \) quantifies the user’s profit (in percentages) obtained from using dynamic decisions, in relation to the static problem formulation \( (F_{os}) \). For instance, in Example 2.1 we have \( z^*_{\text{static}} = 1 \) and \( z_{\text{lower}} = 0.25 \). At the same time, it is rather straightforward to verify that \( z^*_{\text{dynamic}} \geq 0.5 \) irrespective of the auxiliary distributional constraints observed by the user. Hence, in this example \( \rho_1 \leq 50\% \) and the value of 100\% is not always achievable.

On the other hand, the quality of the constraint verification procedure can be defined as:

\[
\rho_2 := 100 \times \frac{z^*_{\text{dynamic}} - \tilde{z}_{\text{dynamic}}}{z^*_{\text{static}} - z_{\text{lower}}},
\]

where \( \tilde{z}_{\text{dynamic}} \) is our estimate of the worst-case expected loss after the constraint verification procedure. More precisely, \( \tilde{z}_{\text{dynamic}} \) can be computed by solving the following
linear programming problem:

$$\tilde{z}_{\text{dynamic}} := \max_{c \in \tilde{S}} c^T \tilde{y}$$

where $\tilde{S}$ is a polyhedral uncertainty set corresponding to the ambiguity set $\tilde{Q} \in \{ Q_1, \ldots, Q_{2L} \}$ identified by the user while traversing through the network; recall Theorem 3.1.

We observe that the sum $\rho_1 + \rho_2$ characterizes the total value (in percentages) by which the difference $z^{*}_{\text{static}} - z_{\text{lower}}$ can be reduced by leveraging our two-step validation approach. As a potential drawback, it may be the case that $\tilde{z}_{\text{dynamic}} < z_{\text{lower}}$ and, thus, the actual upper bound for $\rho_2$ is not well defined.

4.2 Test Instances

Classes of graphs. We consider two classes of fully connected layered graphs that are either acyclic or contain directed cycles. An acyclic layered graph is assumed to contain $h \in \mathbb{Z}_+$ intermediate layers and $r \in \mathbb{Z}_+$ nodes at each layer. The first and the last layer consist of unique nodes, which are the source and the destination nodes, respectively. General graphs are, in turn, obtained from acyclic graphs assuming that the arcs not adjacent to $s$ and $f$ can be traversed in both directions; see, e.g., Fig. 4.

Nominal distribution and initial ambiguity set. With respect to the nominal marginal distributions $Q^*_a, a \in A$, we assume that the arc costs $c_a$ are governed by a standard beta distribution with parameters $\alpha_a, \beta_a \in \mathbb{R}_+$ and a support given by $[0, 1]$. The parameters $\alpha_a$ and $\beta_a$ for each $a \in A$ can be defined using the mean, $m_a$, and the standard deviation, $\sigma_a$, of $c_a$, i.e.,

$$\alpha_a = \frac{m_a^2 (1 - m_a)}{\sigma_a^2} - m_a, \quad \beta_a = \alpha_a \left( \frac{1}{m_a} - 1 \right), \quad (27)$$

see, e.g., [21]. In particular, for each $a = (i, j) \in A, i < j$, we set $\sigma_a = 0.125$, and select $m_a$ uniformly at random from the interval

$$\left( \frac{1}{2} (1 - \sqrt{1 - 4 \sigma_a^2}), \frac{1}{2} (1 + \sqrt{1 - 4 \sigma_a^2}) \right).$$
Table 1 Summary of the notation used in the construction of test instances

| Parameter | Definition |
|-----------|------------|
| $h \in \mathbb{Z}_{>0}$ | A number of intermediate layers in the graph |
| $r \in \mathbb{Z}_{>0}$ | A number of nodes at each layer |
| $\kappa \in (0, 1)$ | A probability that a sensor is placed at each node |
| $\tilde{n} \in \mathbb{Z}_{>0}$ | A number of samples in the data set $\tilde{C}$ |
| $\hat{n} \in \mathbb{Z}_{>0}$ | A number of samples in the data set $\hat{C}$ |
| $\eta_i = 1 - \frac{1-\eta}{|N|}, i \in N$ | Confidence levels for the linear expectation constraints |
| $\eta \in (0, 1)$ | A confidence level for the initial ambiguity set (28) |
| $\gamma \in (0, 1)$ | A confidence level for each auxiliary constraint |

The latter condition guarantees that a beta distribution defined by (27) exists, i.e., $\alpha_a, \beta_a > 0$. In addition, we assume that the nominal marginal distributions $Q_{(i, j)}^*$ and $Q_{(j, i)}^*$ are the same for any $(i, j) \in A$. Finally, the joint distribution $Q^*$ is defined as a product of marginal distributions $Q_a^*, a \in A$.

Next, the initial ambiguity set $Q$ is constructed as follows:

$$Q = \left\{ Q \in Q_0(\mathbb{R}^{|A|}) : \mathbb{E}_Q \left\{ \sum_{a \in FS_i \cup RS_i} c_a - \sum_{(i, j) : (i, j) \in FS_i, (j, i) \in RS_i} c_{(i, j)} \right\} \leq \Gamma_i \quad \forall i \in N, \right\}.$$

Specifically, the linear expectation constraints with respect to $a \in FS_i \cup RS_i$ can be thought as some budget constraints from the attacker’s perspective (if the graph is general, then we account each arc $a \in FS_i \cup RS_i$ only once). The second linear expectation constraints in (28) indicate that the expected costs of $(i, j) \in A$ and $(j, i) \in A$ are the same; these constraints stem from the definition of $Q^*$ and are assumed to be satisfied by construction. The last constraints in (28) are the support constraints.

For every $i \in N$, we calculate the parameters $\Gamma_i$ in (28) by leveraging a data set

$$\tilde{C} = \left\{ (\tilde{c}_{1}^{(k)}, \ldots, \tilde{c}_{|A|}^{(k)})^T, k \in \{1, \ldots, \tilde{n}\} \right\}$$

obtained from the nominal distribution $Q^*$ and Hoeffding inequality (14) with a prescribed confidence level $\eta_i \in (0, 1)$. Specifically, in view of Bonferroni’s inequality [6], we set $\eta_i = 1 - \frac{1-\eta}{|N|}, i \in N$, where $\eta \in (0, 1)$ is a required confidence level for the ambiguity set $Q$. For convenience, we report the notations used in our test instances in Table 1.

Summarizing the discussion above, our construction of $Q$ is somewhat stylized and not without limitations. However, in Sect. 4.3 it is verified numerically that $z_{static}^* \neq$...
**Auxiliary constraints.** For simplicity, we suppose that the auxiliary constraints are restricted by the linear expectation constraints of the form (6b); some additional results for the auxiliary constraints (6a) are relegated to the Appendix. We recall that by Theorem 3.1 the probability constraints (6a) can also be seen as some interval constraints with respect to expected costs. However, the difference between (6a) and (6b) lies in the form of resulting expectation constraints and the data used to verify these constraints; we refer to our discussion in Sect. 2.3 and Appendix.

We assume that each node $i \in N$ is equipped with a sensor with a fixed probability $\kappa \in (0, 1)$; see, e.g., Fig. 5. Furthermore, following Example 2.2, we suggest three types of auxiliary expectation constraints associated with arcs in the forward direction from $s$ to $f$:

- **individual constraints** corresponding to the adjacent sensors, e.g., constraints for the expected cost of $(1, 2)$, $(2, 4)$ and $(2, 5)$ in Fig. 5.
- **difference constraints**, e.g., constraints with respect to the difference of expected costs of $(3, 4)$ and $(3, 5)$ in Fig. 5.
- **sum constraints**, e.g., constraints with respect to the sum of expected costs of $(3, 4)$ and $(3, 5)$ in Fig. 5 (we use these constraints, if at least one of the arcs can be traversed in both directions).

Formally, a constraint of the form (6b) related to a node $i \in N \setminus \{f\}$ can be defined as:

$$\mathbb{E}_Q \left\{ \sum_{a \in FS_i} p_a c_a \right\} = \sum_{a \in FS_i} p_a \bar{c}_a \leq p_0,$$

where $\bar{c}_a = \mathbb{E}_{Q_a} \{c_a\}$, the coefficients $p_a \in \{-1, 0, 1\}$ depend on the type of this constraint and the threshold $p_0 \in \mathbb{R}$ is, e.g., defined as an average value of the sum $\sum_{a \in FS_i} p_a \bar{c}_a$ over the initial polyhedron $S_0$:

$$p_0 := \frac{1}{2} \left( \min_{\bar{c} \in S_0} \sum_{a \in FS_i} p_a \bar{c}_a + \max_{\bar{c} \in S_0} \sum_{a \in FS_i} p_a \bar{c}_a \right).$$

We collect a required number of the auxiliary constraints by selecting a random node and a random constraint associated with this node. If the above procedure does not allow to construct the required number of constraints, then the sensors allocation is updated.
For convenience, we generate a new data set
\[
\hat{C} := \left\{ (\hat{c}_1^{(k)}, \ldots, \hat{c}_{|A|}^{(k)})^\top, \ k \in \{1, \ldots, \hat{n}\} \right\}
\]
for the constraint verification procedure; recall that we used another data set \(\tilde{C}\) to construct the initial ambiguity set \(Q\). This splitting of data is rather technical and reflects the fact that the data for the constraint verification procedure can be collected in an online manner and, thus, it is not necessarily available at the beginning of the game.

The auxiliary constraints are verified via Hoeffding inequality (14) with a fixed confidence level \(\gamma \in (0, 1)\) (which is assumed to be the same for all auxiliary constraints). As outlined before, in order to verify a constraint of the form (29) related to a node \(i \in N \setminus \{f\}\), we use only random samples (or their linear combinations) associated with the arcs \(a \in FS_i\). If the user fails to verify a auxiliary constraint using \(\hat{n}\) samples, then this constraint is verified by force as follows:

\begin{itemize}
  \item if the constraint is an individual or a sum constraint, then it is assumed to be violated (it models the worst-case scenario for the user);
  \item if the constraint is a difference constraint, then it is assumed to be satisfied with probability 0.5.
\end{itemize}

Summarizing the discussion above, we expect that for a fixed confidence level \(\gamma\) the quality of the constraint verification procedure, \((G_2)\), increases with the increase of \(\hat{n}\).

### 4.3 Results for Different Network Types

In the numerical experiments, we explore how the quality of adaptive decisions scales in the number of auxiliary constraints, the size of the network and the sample sizes, \(\hat{n}\) and \(\tilde{n}\); recall Table 1. In particular, for different parameter settings we compute the average relative gaps \((G_1)\) and \((G_2)\) and the average running time with mean absolute deviations for 50 test instances. Also, in all experiments we set \(\eta = 0.95\), \(\gamma = 0.95\) and \(\kappa = 0.5\).

#### Dependence on the number of auxiliary constraints

Let \(h = r = 3\) and \(\hat{n} = \tilde{n} = 60\). For each test instance we increase the number of auxiliary constraints, \(|\mathcal{L}|\), from 1 to 5. The results for acyclic and general graphs are reported, respectively, in Tables 2 and 3.

The observations from our computational results can be summarized as follows:

\begin{itemize}
  \item Augmenting the set of auxiliary constraints provides more information for the user and, thus, improves the quality of both adaptive decisions and the constraint verification procedure. For example, for \(|\mathcal{L}| = 5\) and acyclic graphs the total profit \(\rho_1 + \rho_2\) obtained by the user is about 22% on average.
  \item The multistage problem \((F_{ms})\) becomes more computationally complex for general graphs and with the increase in \(|\mathcal{L}|\). This fact is rather intuitive as the number of variables and constraints in \((F_{ms})\) is exponential in \(|\mathcal{L}|\) and increases for general graphs.
\end{itemize}
Table 2 Let $h = r = 3$, $\hat{n} = \tilde{n} = 60$ and assume that the graph is acyclic. We report the average relative gaps ($G_1$) and ($G_2$) and the average running times in seconds with mean absolute deviations (in brackets) over 50 random test instances.

| Number of constraints | Acyclic graphs | | | | Average time (MAD) in sec | |
|---|---|---|---|---|---|---|
| $|\mathcal{L}| = 1$ | $\rho_1$ (MAD) in % | $\rho_2$ (MAD) in % | | | | |
| $|\mathcal{L}| = 2$ | 2.0 (3.5) | 1.5 (2.8) | 0.04 (0.02) | | | |
| $|\mathcal{L}| = 3$ | 4.8 (7.2) | 5.8 (9.1) | 0.07 (0.02) | | | |
| $|\mathcal{L}| = 4$ | 6.3 (8.9) | 8.3 (11.7) | 0.17 (0.07) | | | |
| $|\mathcal{L}| = 5$ | 8.3 (10.0) | 11.3 (14.1) | 0.67 (0.33) | | | |

Table 3 Let $h = r = 3$, $\hat{n} = \tilde{n} = 60$ and assume that the graph is general. We report the average relative gaps ($G_1$) and ($G_2$) and the average running times in seconds with mean absolute deviations (in brackets) for 50 random test instances.

| Number of constraints | General graphs | | | | Average time (MAD) in sec | |
|---|---|---|---|---|---|---|
| $|\mathcal{L}| = 1$ | $\rho_1$ (MAD) in % | $\rho_2$ (MAD) in % | | | | |
| $|\mathcal{L}| = 2$ | 1.8 (3.2) | 3.3 (5.9) | 0.08 (0.05) | | | |
| $|\mathcal{L}| = 3$ | 3.6 (6.0) | 4.4 (7.3) | 0.28 (0.10) | | | |
| $|\mathcal{L}| = 4$ | 4.9 (7.1) | 9.1 (13.1) | 10.1 (6.71) | | | |
| $|\mathcal{L}| = 5$ | – | – | > 600 | | | |

Table 4 Let $h = r = 3$, $|\mathcal{L}| = 5$ and assume that the graph is acyclic. We report the average relative gaps ($G_1$) and ($G_2$) and the average running times in seconds with mean absolute deviations (in brackets) for 50 random test instances.

| # samples in $\tilde{\mathcal{C}}$ | # samples in $\hat{\mathcal{C}}$ | Acyclic graphs | | | | |
|---|---|---|---|---|---|---|
| $\hat{n} = 20$ | | $\rho_1$ (MAD) in % | $\rho_2$ (MAD) in % | Average time (MAD) in sec | | |
| $\hat{n} = 40$ | | 2.0 (3.6) | 22.7 (29.1) | 2.49 (1.80) | | |
| $\hat{n} = 60$ | | 7.9 (9.7) | 16.3 (18.5) | 3.43 (2.46) | | |
| $\hat{n} = 60$ | $\tilde{n} = 60$ | 10.2 (11.6) | 12.2 (15.2) | 3.8 (2.85) | | |
| $\hat{n} = 80$ | | 11.6 (11.8) | 9.5 (11.9) | 5.45 (5.11) | | |
| $\hat{n} = 100$ | | 12.2 (11.6) | 12.2 (13.2) | 3.89 (2.95) | | |

| $\tilde{n} = 20$ | | $\rho_1$ (MAD) in % | $\rho_2$ (MAD) in % | Average time (MAD) in sec | | |
| $\tilde{n} = 40$ | | 2.0 (3.6) | 22.7 (29.1) | 2.49 (1.80) | | |
| $\tilde{n} = 60$ | | 7.9 (9.7) | 16.3 (18.5) | 3.43 (2.46) | | |
| $\tilde{n} = 60$ | $\hat{n} = 60$ | 10.2 (11.6) | 12.2 (15.2) | 3.8 (2.85) | | |
| $\tilde{n} = 80$ | | 11.6 (11.8) | 9.5 (11.9) | 5.45 (5.11) | | |
| $\tilde{n} = 100$ | | 12.2 (11.6) | 12.2 (13.2) | 3.89 (2.95) | | |
Dependence on the sample size. In this experiment, we consider acyclic graphs and explore the value of parameters \( \hat{n} \) and \( \tilde{n} \). For each test instance and different values of \( \hat{n} \) and \( \tilde{n} \), we preserve the same nominal distribution \( Q^* \) and fix a particular placement of sensors. The results for \( h = r = 3 \) and \( |\mathcal{L}| = 5 \) are reported in Table 4.

We make the following observations:

- \((G_1)\) and \((G_2)\) tend to increase and decrease, respectively, as functions of \( \tilde{n} \). It can be argued that for sufficiently small \( \tilde{n} \) we have \( z_{\text{static}}^* \approx z_{\text{lower}} \). Our hypothesis is that in this case dynamic decisions cannot significantly improve the value of \( z_{\text{static}}^* \), but the constraint verification procedure allows to reimburse these losses.
- \((G_2)\) increases as a function of \( \hat{n} \); see the second part of Table 4. This fact is quite intuitive, since with the increase in \( \hat{n} \) the user may verify more constraints correctly.
- The solution times for the MIP formulation (20) do not depend on \( \hat{n} \) by construction and slightly depend on \( \tilde{n} \); see the last column of Table 4.

Dependence on the size of the network. In our last numerical experiment, we examine the value of the network’s size. Let \( \hat{n} = \tilde{n} = 60 \) and assume that \( |\mathcal{L}| = 5 \) and \( |\mathcal{L}| = 3 \) for acyclic and general graphs, respectively. For each class of graphs, we test different values of \( h \in \{2, 3, 4\} \) and \( r \in \{2, 3, 4\} \) and report the results in Tables 5 and 6.

We make the following observations:

- The solution times increase in both \( h \) and \( r \), while the average relative gaps \( (G_1) \) and \( (G_2) \) tend to decrease. Indeed, the size of the MIP reformulation (20) increases as a function of \( h \) and \( r \), while the role of each particular constraint in \( \mathcal{L} \) becomes less pronounced for networks of a larger size.
- The average relative gap \( (G_1) \) tends to zero faster with the increase in \( r \) rather than \( h \). In this regard, we note that the larger \( r \), the less constraints are potentially met by the user while traversing through the network.

Summary. It turns out that the user may often gain some profit by leveraging dynamic decisions instead of static ones. However, the magnitude of this profit depends on the size of the graph, the form of initial ambiguity set as well as the parameters of the constraint verification procedure. Admittedly, the primary application of our approach is to the networks of a relatively small size, in which solutions with a sufficiently high quality can be derived within a reasonable time. It can also be argued that most of the existing solution approaches to multistage problems with binary recourse decisions can only be applied to instances of a moderate size; see, e.g., the studies in [3, 4, 32] and the references therein.

5 Conclusion

In this paper, we consider a dynamic version of the shortest path problem, where the cost vector is subject to distributional uncertainty. We formulate the problem in terms of an ambiguity-averse multistage network game between a user and an attacker. The user aims at minimizing its cumulative expected loss by traversing between two fixed
Table 5  Let $|L| = 5, \hat{n} = \bar{n} = 60$ and assume that the graph is acyclic. We report the average relative gaps ($G_1$) and ($G_2$) and the average running times in seconds with mean absolute deviations (in brackets) for 50 random test instances.

| # layers | # nodes at each layer | Acyclic graphs |     |     |     |
|----------|-----------------------|----------------|-----|-----|-----|
|          |                       | $\rho_1$ (MAD) in % | $\rho_2$ (MAD) in % | Average time (MAD) in sec |
| $h = 2$  |                       | 22.0 (19.8)     | 19.5 (24.3)     | 0.42 (0.21)     |
| $h = 3$  | $r = 3$               | 10.2 (11.6)     | 12.2 (15.2)     | 3.8 (2.85)      |
| $h = 4$  |                       | 7.6 (9.3)       | 8.6 (11.5)      | 34.46 (30.34)   |
|          | $r = 2$               | 19.8 (16.3)     | 34.1 (26.2)     | 0.19 (0.05)     |
| $h = 3$  | $r = 3$               | 10.2 (11.6)     | 12.2 (15.2)     | 3.8 (2.85)      |
|          | $r = 4$               | 3.3 (5.7)       | 11.6 (15.9)     | 44.13 (47.27)   |

Table 6  Let $|L| = 3, \hat{n} = \bar{n} = 60$ and assume that the graph is general. We report the average relative gaps ($G_1$) and ($G_2$) and the average running times in seconds with mean absolute deviations (in brackets) for 50 random test instances.

| # layers | # nodes at each layer | General graphs |     |     |     |
|----------|-----------------------|----------------|-----|-----|-----|
|          |                       | $\rho_1$ (MAD) in % | $\rho_2$ (MAD) in % | Average time (MAD) in sec |
| $h = 2$  |                       | 16.6 (18.2)     | 10.4 (15.6)     | 0.72 (0.27)     |
| $h = 3$  | $r = 3$               | 4.9 (7.1)       | 9.1 (13.1)      | 10.1 (6.71)     |
| $h = 4$  |                       | 2.5 (4.2)       | 0.9 (1.7)       | 99.62 (88.53)   |
|          | $r = 2$               | 16.0 (16.9)     | 9.9 (13.7)      | 0.47 (0.12)     |
| $h = 3$  | $r = 3$               | 4.9 (7.1)       | 9.1 (13.1)      | 10.1 (6.71)     |
|          | $r = 4$               | 1.1 (2.1)       | 1.7 (3.1)       | 64.54 (43.12)   |

nodes in the network, while the attacker maximizes the user’s objective function by selecting a distribution of arc costs from an ambiguity set of candidate distributions. In contrast to the one-stage formulation, both the user and the attacker are able to adjust their decisions at particular nodes of the user’s path.

Following the related one-stage formulation, we suppose that the family of probability distributions is formed by individual probability constraints for particular arcs and linear expectation constraints for prescribed subsets of arcs. Also, while traversing through the network, the user may verify some auxiliary distributional constraints associated with the arcs emanated from the current user’s position. Specifically, we assume that at the beginning of the game the user forms a list of auxiliary distributional constraints, whereas the attacker forms a list of responses that are consistently revealed to the user.

We design two classes of non-anticipativity constraints (for acyclic and general graphs, respectively) enforcing that the user’s decision at a particular node cannot depend on future attacker’s responses. Furthermore, by using some properties of the related one-stage formulation and linear programming duality the multistage problem is reformulated as a one potentially large linear mixed-integer programming problem.
From the application perspective, we illustrate that the auxiliary distributional constraints can be constructed and verified by using some information from Bluetooth sensors. Finally, we conduct a numerical study where the one- and multistage problem formulations are compared with respect to several classes of synthetic network instances. It turns out that using adaptive decisions is practically relevant only for networks of a relatively small size, in which high-quality adaptive decisions can be obtained within a reasonable time.

With respect to future research directions, it would be interesting to explore some other types of objective criteria and ambiguity sets; see, e.g., [23], in the context of our multistage shortest path problem. At the same time, it can be argued that another types of ambiguity sets imply the need for more advanced solution techniques as our partitions of the initial ambiguity set are based on some linearity properties of the auxiliary distributional constraints. As another possible option, one may consider randomized decisions from the user’s perspective; see, e.g., the related studies in [15, 16].

Acknowledgements The article was prepared within the framework of the Basic Research Program at the National Research University Higher School of Economics (Sects. 1–2). The research is funded by RSF grant No22-11-00073 (Sects. 3–5). The author would like to thank Dr. Oleg Prokopyev for his helpful comments and suggestions. The author is also thankful to the associate editor and two anonymous referees for their constructive comments that allowed to greatly improve the paper.

Data availability statement The author confirms that all data generated or analyzed during this study are included in this published article.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Appendix

In this appendix, we consider the ambiguity set (28) with the auxiliary probability constraints of the form:

$$Q_a\{c_a \in [0.5, 1]\} \leq \tilde{q}_a$$

with $\tilde{q}_a = \tilde{q}$ for each $a \in A$; recall Sect. 2.3. As before, we assume that each node $i \in N$ is equipped with a sensor with probability $\kappa = 0.5$ and the constraints (30) correspond to the arcs $a \in A$ with the adjacent sensors. In all experiments, we set $h = r = 3, \hat{n} = \tilde{n} = 60, \alpha = \gamma = 0.95$ and assume that the graph is acyclic; recall Table 1.

For the first experiment, we set $\tilde{q} = 0.5$ and consider the relative gaps ($G_1$) and ($G_2$), as well as the running time for solving the MIP reformulation (20) as a function of $|L|$. The results are reported in Table 7. For the second experiment, we set $|L| = 5$ and explore how the quality of adaptive decisions scales in the parameter $\tilde{q}$ of the auxiliary constraint (30). For each test instance and different values of $\tilde{q}$, we preserve the same data set (10) and the same initial ambiguity set (28). The results are reported in Table 8.

We make the following observations:
Table 7 Let $h = r = 3$, $\tilde{n} = \hat{n} = 60$, $\tilde{q} = 0.5$ and assume that the graph is acyclic. We report the average relative gaps ($G_1$) and ($G_2$) and the average running times in seconds with mean absolute deviations (in brackets) for 50 random test instances.

| Number of constraints $|L|$ | Acyclic graphs $\rho_1$ (MAD) in % | $\rho_2$ (MAD) in % | Average time (MAD) in sec |
|-----------------|-------------------------------------|---------------------|--------------------------|
| $|L| = 1$        | 1.1 (2.1)                           | 1.8 (3.1)           | 0.03 (0.01)              |
| $|L| = 2$        | 1.7 (2.9)                           | 6.3 (8.8)           | 0.06 (0.02)              |
| $|L| = 3$        | 2.0 (3.4)                           | 10.2 (10.7)         | 0.18 (0.07)              |
| $|L| = 4$        | 2.4 (3.9)                           | 12.0 (11.7)         | 0.69 (0.38)              |
| $|L| = 5$        | 2.9 (4.4)                           | 13.6 (12.9)         | 4.03 (3.05)              |

Table 8 Let $h = r = 3$, $\tilde{n} = \hat{n} = 60$, $|L| = 5$ and assume that the graph is acyclic. We report the average relative gaps ($G_1$) and ($G_2$) and the average running times in seconds with mean absolute deviations (in brackets) for 50 random test instances.

| Parameter $\tilde{q}$ | Acyclic graphs $\rho_1$ (MAD) in % | $\rho_2$ (MAD) in % | Average time (MAD) in sec |
|-------------------|-------------------------------------|---------------------|--------------------------|
| $\tilde{q} = 0.1$ | 0.5 (0.9)                           | 0.2 (0.3)           | 3.06 (1.98)              |
| $\tilde{q} = 0.3$ | 1.6 (2.6)                           | 15.8 (16.1)         | 4.62 (3.89)              |
| $\tilde{q} = 0.5$ | 2.9 (4.4)                           | 13.6 (12.9)         | 4.03 (3.05)              |
| $\tilde{q} = 0.7$ | 3.4 (4.4)                           | 8.3 (8.0)           | 4.78 (4.16)              |
| $\tilde{q} = 0.9$ | 1.5 (1.8)                           | 2.7 (3.4)           | 4.84 (3.85)              |

- Comparing the obtained results in Tables 2 and 7, we observe that the quality of adaptive solutions decreases whenever the probability constraints (6a) are used instead of the expectation constraints (6b). In fact, if the constraint (30) is satisfied, then
  \[
  \mathbb{E}_{Q_a} \{c_a\} \leq 0.5 \times (1 - \tilde{q}) + 1 \times \tilde{q} = 0.5 \tilde{q} + 0.5. \tag{31}
  \]
  Otherwise, we have
  \[
  \mathbb{E}_{Q_a} \{c_a\} \geq 0 \times (1 - \tilde{q}) + 0.5 \times \tilde{q} = 0.5 \tilde{q}. \tag{32}
  \]
  This observation implies that the user receives less information in terms of expected costs by using (30) instead of individual linear expectation constraints of the form (6b).

- Both relative gaps ($G_1$) and ($G_2$) tend to decrease whenever $\tilde{q}$ approaches to its lower or upper bounds. Indeed, if $\tilde{q}$ close to its bounds, then verification of some constraints of the form (30) may not improve the user’s objective function value.

- The constraints (30) may handle interval-censored observations, which is not the case for the auxiliary expectation constraints (29).

In conclusion, we point out that the results of this section are rather consistent with the results of Sect. 4.3. That is, the quality of adaptive decisions and the computational complexity of the MIP formulation (20) increase in the number of auxiliary distributional constraints.
References

1. Asudegi, M., Haghani, A.: Optimal number and location of node-based sensors for collection of travel time data in networks. Transp. Res. Rec. 2338(1), 35–43 (2013). https://doi.org/10.3141/2338-05
2. Bayraksan, G., Love, D.K.: Data-driven stochastic programming using phi-divergences. In: The Operations Research Revolution, INFORMS, pp. 1–19 (2015)
3. Bertsimas, D., Dunning, I.: Multistage robust mixed-integer optimization with adaptive partitions. Oper. Res. 64(4), 980–998 (2016). https://doi.org/10.1287/opre.2016.1515
4. Bertsimas, D., Georgiou, A.: Design of near optimal decision rules in multistage adaptive mixed-integer optimization. Oper. Res. 63(3), 610–627 (2015). https://doi.org/10.1287/opre.2015.1365
5. Bertsimas, D., Sim, M., Zhang, M.: Adaptive distributionally robust optimization. Manag. Sci. 65(2), 604–618 (2018). https://doi.org/10.1287/mnsc.2017.2952
6. Birge, J.R., Louveaux, F.: Introduction to Stochastic Programming. Springer, Berlin (2011)
7. Borrero, J.S., Prokopyev, O.A., Sauré, D.: Sequential shortest path interdiction with incomplete information. Decis. Anal. 13(1), 68–98 (2015). https://doi.org/10.1287/dec3.2015.0325
8. Borrero, J.S., Prokopyev, O.A., Sauré, D.: Sequential interdiction with incomplete information and learning. Oper. Res. 67(1), 72–89 (2019). https://doi.org/10.1287/opre.2018.1773
9. Bubeck, S., Cesàro-Bianchi, N., Kakade, S.M.: Towards minimax policies for online linear optimization with bandit feedback. In: Mannor, S., Srebro, N., Williamson, R.C. (eds.), Conference on Learning Theory, pp. 41.1–41.14 (2012)
10. Buchheim, C., Kurtz, J.: Robust combinatorial optimization under convex and discrete cost uncertainty. EURO J. Comput. Optim. 6(3), 211–238 (2018). https://doi.org/10.1007/s13675-018-0103-0
11. Cao, Z., Guo, H., Zhang, J., Niyato, D., Fastenrath, U.: A data-driven method for stochastic shortest path problem. In: 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), IEEE, pp. 1045–1052 (2014)
12. Chassein, A., Dokka, T., Goerigk, M.: Algorithms and uncertainty sets for data-driven robust shortest path problems. Eur. J. Oper. Res. 274(2), 671–686 (2019). https://doi.org/10.1016/j.ejor.2018.10.006
13. Cheng, J., Leung, J., Lissier, A.: New reformulations of distributionally robust shortest path problem. Comput. Oper. Res. 74, 196–204 (2016). https://doi.org/10.1016/j.cor.2016.05.002
14. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, 3rd edn. MIT Press, Cambridge (2009)
15. Delage, E., Kuhn, D., Wiesemann, W.: “Dice”-sion–making under uncertainty: when can a random decision reduce risk? Manag. Sci. 65(7), 3282–3301 (2019). https://doi.org/10.1287/mnsc.2018.3108
16. Delage, E., Saif, A.: The value of randomized solutions in mixed-integer distributionally robust optimization. INFORMS J. Comput. 34(1), 333–353 (2022). https://doi.org/10.1287/ijoc.2020.1042
17. Delage, E., Ye, Y.: Distributionally robust optimization under moment uncertainty with application to data-driven problems. Oper. Res. 58(3), 595–612 (2010). https://doi.org/10.1287/opre.1090.0741
18. Esfahani, P.M., Kuhn, D.: Data-driven distributionally robust optimization using the Wasserstein metric: performance guarantees and tractable reformulations. Math. Program. 171(1–2), 115–166 (2018). https://doi.org/10.1007/s10107-017-1172-1
19. Gavriel, C., Hanasusanto, G., Kuhn, D.: Risk-averse shortest path problems. In: 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), IEEE, pp. 2533–2538 (2012)
20. Goh, J., Sim, M.: Distributionally robust optimization and its tractable approximations. Oper. Res. 58(4–part–1), 902–917 (2010). https://doi.org/10.1287/opre.1090.0795
21. Gupta, A.K., Nadarajah, S.: Handbook of Beta Distribution and its Applications. CRC Press, Boca Raton (2004)
22. Haghani, A., Hamedi, M., Sadabadi, K.F., Young, S., Tarnoff, P.: Data collection of freeway travel time ground truth with Bluetooth sensors. Transp. Res. Rec. 2160(1), 60–68 (2010). https://doi.org/10.3141/2160-07
23. Hanasusanto, G.A., Kuhn, D., Wiesemann, W.: K-adaptability in two-stage distributionally robust binary programming. Oper. Res. Lett. 44(1), 6–11 (2016). https://doi.org/10.1016/j.orl.2015.10.006
24. Hoeffding, W.: Probability inequalities for sums of bounded random variables. J. Am. Stat. Assoc. 58(301), 13–30 (1963). https://doi.org/10.1080/01621459.1963.10500830
25. Isit, K.: On sharpness of Tchebycheff-type inequalities. Ann. Inst. Stat. Math. 14(1), 185–197 (1962). https://doi.org/10.1007/BF02868641
26. Jiang, R., Guan, Y.: Risk-averse two-stage stochastic program with distributional ambiguity. Oper. Res. 66(5), 1390–1405 (2018). https://doi.org/10.1287/opre.2018.1729
27. Ketkov, S.S., Prokopyev, O.A.: On greedy and strategic evaders in sequential interdiction settings with incomplete information. Omega 92, 102161 (2020). https://doi.org/10.1016/j.omega.2019.102161
28. Ketkov, S.S., Prokopyev, O.A., Burashnikov, E.P.: An approach to the distributionally robust shortest path problem. Comput. Oper. Res. 130, 105212 (2021). https://doi.org/10.1016/j.cor.2021.105212
29. Khanjani-Shiraz, R., Babapour-Azar, A., Hosseini-Noudeh, Z., Pardalos, P.M.: Distributionally robust maximum probability shortest path problem. J. Comb. Optim. 43(1), 140–167 (2021). https://doi.org/10.1007/s10878-021-00747-9
30. Kreinovich, V., Xiang, G., Starks, S.A., Longpré, L., Ceberio, M., Araiza, R., Beck, J., Kandathi, R., Nayak, A., Torres, R., Hajagos, J.G.: Towards combining probabilistic and interval uncertainty in engineering calculations: algorithms for computing statistics under interval uncertainty, and their computational complexity. Reliab. Comput. 12(6), 471–501 (2006). https://doi.org/10.1007/s11155-006-9015-4
31. Miller, C.E., Tucker, A.W., Zemlin, R.A.: Integer programming formulation of traveling salesman problems. J. ACM (JACM) 7(4), 326–329 (1960). https://doi.org/10.1145/321043.321046
32. Postek, K., Hertog, D.: Multistage adjustable robust mixed-integer optimization via iterative splitting of the uncertainty set. INFORMS J. Comput. 28(3), 553–574 (2016). https://doi.org/10.1287/ijoc.2016.0696
33. Rockafellar, R.T., Uryasev, S.: Optimization of conditional value-at-risk. J. Risk 2, 21–42 (2000). https://doi.org/10.21314/JOR.2000.038
34. Sefair, J.A., Smith, J.C.: Dynamic shortest-path interdiction. Networks 68(4), 315–330 (2016). https://doi.org/10.1002/net.21712
35. Shapiro, A.: On duality theory of conic linear problems. In: Goberna, M.A., Lopez, M.A. (eds.) Semi-Infinite Programming, pp. 135–165. Springer, Boston (2001)
36. Sun, J.: The Statistical Analysis of Interval-censored Failure Time Data. Springer, Berlin (2006)
37. Taccari, L.: Integer programming formulations for the elementary shortest path problem. Eur. J. Oper. Res. 252(1), 122–130 (2016). https://doi.org/10.1016/j.ejor.2016.01.003
38. Vo, T.: An investigation of bluetooth technology for measuring travel times on arterial roads: a case study on spring street. Dissertation, Georgia Institute of Technology (2011)
39. Wang, Z., You, K., Song, S., Zhang, Y.: Wasserstein distributionally robust shortest path problem. Eur. J. Oper. Res. 284(1), 31–43 (2020). https://doi.org/10.1016/j.ejor.2020.01.009
40. Wiesemann, W., Kuhn, D., Sim, M.: Distributionally robust convex optimization. Oper. Res. 62(6), 1358–1376 (2014). https://doi.org/10.1287/opre.2014.1314
41. Wolsey, L.A., Nemhauser, G.L.: Integer and Combinatorial Optimization. Wiley, New York (1999)
42. Xu, H., Mannor, S.: Distributionally robust Markov decision processes. In: Lafferty, J., Williams, C., Shawe-Taylor, J., Zemel, R., Culotta, A. (eds.) Advances in Neural Information Processing Systems 23 (NIPS 2010). Curran Associates Inc., New York (2010)
43. Xu, X., Shen, S.: Multistage distributionally robust mixed-integer programming with decision-dependent moment-based ambiguity sets. Math. Program. 190(1–2), 1025–1064 (2022). https://doi.org/10.1007/s10107-020-01580-4
44. Zhang, Y., Song, S., Shen, Z.J.M., Wu, C.: Robust shortest path problem with distributional uncertainty. IEEE Trans. Intell. Transp. Syst. 19(4), 1080–1090 (2017). https://doi.org/10.1109/TITS.2017.2709798
45. Zou, J., Ahmed, S., Sun, X.A.: Stochastic dual dynamic integer programming. Math. Program. 175(1–2), 461–502 (2019). https://doi.org/10.1007/s10107-018-1249-5

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.