Shortest Path Network Interdiction with Asymmetric Uncertainty

She'ifa Z. Punla-Green, John E. Mitchell*

Department of Mathematical Sciences, Rensselaer Polytechnic Institute, United States

Jared L. Gearhart, William E. Hart, Cynthia A. Phillips

Sandia National Laboratories, United States

Abstract

This paper considers an extension of the shortest path network interdiction problem that incorporates robustness to account for parameter uncertainty. The shortest path interdiction problem is a game of two players with conflicting agendas and capabilities: an evader, who traverses the arcs of a network from a source node to a sink node using the path of shortest length, and an interdictor, who maximizes the length of the evader's shortest path by interdicting arcs on the network. It is usually assumed that the parameters defining the network are known exactly by both players. We consider the situation where the evader assumes the nominal parameter values while the interdictor uses robust optimization techniques to account for parameter uncertainty or sensor degradation. We formulate this problem as a nonlinear mixed integer trilevel program and show that it can be converted into a mixed integer linear program with a second order cone constraint. We use random geometric networks and transportation networks to perform computational studies and demonstrate the unique decision strategies that our variant produces. Solving the shortest path interdiction problem with asymmetric uncertainty protects the interdictor from investing in the obvious strategy if that strategy hinges on key interdictions performing as promised. It also provides an alternate strategy that mitigates the risk of these worst-case possibilities.

Keywords: Networks. Bilevel programming. Interdiction. Uncertainty modelling. Robustness and sensitivity analysis

---

⋆Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA000352.
This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government. SAND2022-2235 O

*Corresponding author

Email addresses: punlas@rpi.edu (She'ifa Z. Punla-Green), mitchj@rpi.edu (John E. Mitchell), jlgearh@sandia.gov (Jared L. Gearhart), wehart@sandia.gov (William E. Hart), caphill@sandia.gov (Cynthia A. Phillips)

1. Introduction

In this paper, we propose a novel variant of the shortest path interdiction problem (SPI) that incorporates robustness to account for parameter uncertainty. The shortest path interdiction problem is a game of two players with conflicting agendas and capabilities: an evader, who traverses the arcs of a network from a source node to a sink node using the path of shortest length, and an interdictor, who maximizes the length of the evader's shortest path by interdicting arcs on the network. Each arc has a base length and an additional length added when the arc is interdicted. When solving the shortest path interdiction problem, it is assumed that these values are known exactly. This assumption could be problematic in real-world applications where these values are estimates or where you are concerned about components of the network under-performing and failing, thus deviating from the assumed parameter values. To account for this parameter uncertainty, we incorporate robust optimization methods into the interdiction problem. An attractive feature of robust optimization is that it does not require any knowledge of the probability distribution of the parameters. Instead, it assumes that the values of the parameters are confined to an uncertainty set. Due to the adversarial nature of the shortest path interdiction problem, we must be careful that any assumptions of uncertainty only apply to the player making that assumption. Therefore, our proposed variant applies robust optimization techniques to the shortest path interdiction problem using methodologies from asymmetric information. This variant will be referred to as the shortest path interdiction problem with asymmetric uncertainty.

For example, the shortest path interdiction problem can be used to make decisions for where to place sensors in a network (traffic, water, etc.). While solving the nominal SPI can be helpful in deciding where to place a limited number of sensors in a network, the interdictor might be wary of trusting the efficacy promised by the sensor manufacturer. The promised efficacy of the sensor is likely only one possible outcome within a range of potential realizations that could occur when the user actually needs to rely on the sensor. To address this concern, our variant, the SPI with asymmetric uncertainty, accounts for some level of reduction in efficacy of the sensors. Depending on the interdictor's risk tolerance this could range from mild degradation to a complete failure of one or more sensors.

1.1. Shortest Path Interdiction

Our model is a variant of the shortest path interdiction problem (Israeli and Wood (2002)). The SPI considers a directed network $G(N, A)$, where N is the set of nodes and A is the set of arcs. The evader traverses the network from a source node S to a destination node T. Each arc $(i, j) \in A$ has a base length ℓ_{ij} and a length added when that arc has been interdicted, r_{ij}. The number of arcs that the interdictor can extend is B.

The shortest path interdiction problem is modelled as a bilevel program (Formulation (1)). The evader's subproblem (1c) encapsulates the needs and wants of the evader. The evader's variable y_{ij} (y_k) takes a positive value if the arc from i to j (arc k) is traversed by the evader. Their constraint compares the flow into a node i along the arcs in the reverse star $RS(i)$ (the set of arcs that end at node i) and the flow out along the arcs in the forward star $FS(i)$ (the set of arcs that begin at node i). The right side is 1 when $i = S$ and the evader is forced to start from the source S. Similarly, the right hand side is -1 when $i = T$ and the evader must end at the destination. The remaining constraints require flow conservation on intermediate nodes. The evader seeks to the minimize the path length; their objective calculates the length of the path. The interdictor's role is represented by the outer problem. The interdictor's objective (1a) is the same as the evader's objective, but they want to maximize the length. Their interdiction decision variable x_{ij} takes the value 1 if the arc from i to j has been interdicted and 0 if not. They are constrained to only interdict B arcs (1b).

$$\max_{x \in X} \quad \sum_{k \in A} (\ell_k + x_k r_k) y_k \tag{1a}$$

$$\text{s.t.} \quad X = \{x \in \{0,1\}^{|A|} | \sum_{k \in A} x_k \leq B\}, \tag{1b}$$

$$y \in \operatorname{argmin}_{\bar{y} \geq 0} \left\{ \sum_{k \in A} (\ell_k + r_k x_k) \bar{y}_k : \sum_{k \in FS(i)} \bar{y}_k - \sum_{k \in RS(i)} \bar{y}_k = \begin{cases} 1 & i = S \\ 0 & \forall i \in N - \{S,T\} \\ -1 & i = T \end{cases} \right\} \tag{1c}$$

1.2. Robust Optimization

A key assumption of the shortest path interdiction problem is that the network parameters, ℓ and r, are known exactly to both players. However, there are many real world applications where these parameters are estimates or where the parameter represents a component of the network that could have degraded performance. It is a reasonable assumption that a player would want to model robustness into their solution to avoid deploying a solution that is far from optimal under a slightly different manifestation of the parameters. Using a library of linear programs, (Ben-Tal et al. (2009)) showed that if the "true" parameters differed from the assumed parameters by even a few digits, the nominal optimal solution can often be infeasible. Robust optimization requires that constraints hold regardless of the manifestation of the parameters within an uncertainty set. When using the robust optimal solution, we pay a price in the objective value in exchange for reliability.

For example, Figure 1 visualizes the feasible region for a linear program as the blue shaded region between the two black lines visualizing the constraints. The objective for this linear program is to try to minimize the variable x, pushing the solution as far left in the visualization as possible. The nominal optimal solution is the cyan dot at the tip of the cone. However, consider a scenario where the upper constraint relies on uncertain parameters. We know that the parameters only differ from the nominal values by some set amount so we can find the worst-case manifestation of the parameters within the uncertainty set. The constraint using those worst-case parameters is visualized with the dotted blue line. If we require that any solution to the LP satisfy that constraint for the worst-case manifestation of the parameters, the robust optimal solution is the red dot at the intersection of the new constraint's dotted blue line and the lower constraint's solid black line. The robust optimal solution is not as good as the nominal optimal solution because it is not as far to the left, but we know that it is feasible for any manifestation of the uncertain parameters and is therefore more reliable.

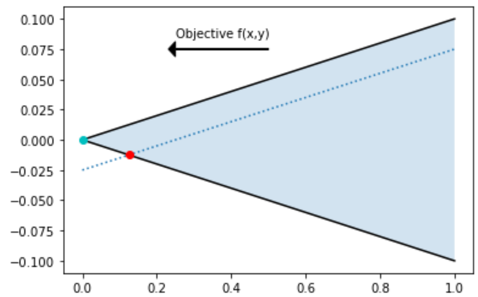

Figure 1: Robust Optimization Example

Robust optimization is an approach to handle parameter uncertainty that is suitable when we wish to mitigate worst-case effects or the probability distribution of the parameter is not available, ruling out many stochastic optimization approaches. Instead, robust optimization simply requires bounds on the parameters. Additionally, with a few reasonable assumptions, robust optimization problems are tractable despite requiring a constraint to hold for any manifestation of the parameter within the uncertainty set. While this requirement turns a linear program into a semi-infinite program due to the infinite number of constraints, robust optimization techniques can turn that semi-infinite program into a problem that can be solved by most commercial math programming solvers.

1.3. Outline and Key Contributions

The key contributions of this paper are:

- A new variant of the SPI is proposed that allows the interdictor to protect themself against parameter uncertainty by using techniques from robust optimization and asymmetric information.
- A reformulation procedure is presented that turns this new variant from a nonlinear trilevel mixed integer program into a single level mixed integer linear program with a second order cone constraint, a form that commercial math programming solvers can handle.
- A case study using a Sioux Falls transportation network is performed to demonstrate the use of our variant on applications with multiple possible sources and destinations. Two methods, super source/sink and worst-case (S,T), are compared and contrasted.

The remaining sections of the paper include a literature review (Section 2), the definition of our SPI variant (Section 3), and results of various examples demonstrating our variant's use and computational costs (Section 4).

2. Literature Review

The shortest path interdiction problem has been used to model a variety of applications. For example, (Pan et al. (2003)) address smuggling prevention with a stochastic programming variant of the SPI to decide where to place sensors to detect nuclear material. This variant addresses the issue that the evader's origin and destination are unknown at the time the sensors are placed. An extension of the stochastic nuclear smuggling model is studied by (Sullivan et al. (2014)). This extension introduces asymmetric information where the interdictor has different (and likely more accurate) information about the network parameters as compared to the evader. (Kosmas et al. (2020)) study a max flow network interdiction problem where the arcs can be added to the network after the interdiction decisions have been made. They apply this new model to disrupt illicit drug and human trafficking networks. For additional examples, (Smith and Song (2020)) provide a thorough summary of the numerous applications of network interdiction.

Additionally, there are many other variants of the shortest path interdiction problem under uncertainty. (Song and Shen (2016)) propose a method for modelling and solving the SPI with uncertain arc lengths, a risk-averse interdictor who uses a chance constraint to set a minimum threshold on the evader's path length, and an evader who waits to see the actualization of the arc lengths. (Bayrak and Bailey (2008)) study a version of the SPI where the evader and the interdictor disagree on the parameters of the network, allowing the interdictor to use the evader's incorrect assumptions to their advantage. Similarly, (Nguyen and Smith (2021)) propose a SPI variant where the interdictor does not know the base arc length when they make their decision (only that the lengths are uniformly distributed in an interval), while that evader observes the exact arc lengths. The interdictor seeks to maximize the **expected** path length of the evader.

Finally, in the more general area of bilevel problems under uncertainty, (Besançon et al. (2020)) propose computational approaches to solve a bilevel problem robustly. Their approach protects the upper level decision-maker from infeasibility if the lower level decision-maker chooses a near-optimal solution. (Buchheim and Henke (2019)) study the bilevel continuous knapsack problem where the lower level decision-maker's objective is not known exactly to the upper level decision-maker. They use robust optimization techniques to define the uncertainty set of the lower level decision-maker's profit parameter.

3. Model Formulation

The goal of our new variant is to allow the interdictor to make their decisions robust to parameter uncertainty. In a single level setting, this would be achieved by replacing the uncertain parameters with an expression that relies on some uncertainty variable and adding constraints that define the uncertainty set for that new variable. Robust optimization techniques describe how to choose the expression and uncertainty set so that the problem remains tractable. However, simply making this substitution for the robust equivalent does not work in the adversarial setting of the shortest path interdiction problem. Because the two players are optimizing in the opposite direction, the robustness is going to benefit one or the other if applied to their common objective. Therefore, we will draw upon techniques from (Bayrak and Bailey (2008)) which assume that the players have their own understanding of the network parameters and define their objectives separately. Specifically, we assume that the evader solves their shortest path problem with the nominal parameters while the interdictor solves the overall problem with robustness in the parameters. This decision was motivated by the sensor placement setting where the sensors might not perform exactly as expected or advertised but the parameters describing the base arc lengths are known more or less exactly. The robustness in r would represent any degradation in the the sensor efficacy. The approach described in this paper could be similarly applied to applications where the uncertainty is in the base length ℓ as that is the simpler case.

3.1. Model Definition and Reformulation

Our variant formulation seeks to model two players: an evader who trusts the nominal values for the parameters and an interdictor who assumes that the nominal values are potentially flawed. Thus, the interdictor chooses arcs to interdict, the evader finds the shortest path using nominal values, the robustness reduces the efficacy of interdictions, and the finally the interdictor evaluates the length of the evader's path using the adjusted parameter values.

This model differs from many similar approaches in the literature because we do not assume that the difference between the evader's and interdictor's parameter knowledge is because the interdictor must make decisions ahead of time while the evader knows the true actualization through a "wait and see" approach. Instead, we assume that evader is using the assumed or advertised parameter values while the interdictor wants to be prepared if those values are less than advertised. This assumption is less appropriate for physical networks such as transportation where the evader reasonably would have better knowledge of traffic conditions, weather, and road closures the day of as opposed to the interdictor who made their interdictions well in advance. Instead, our assumptions are reasonable for networks where observing the network the day of an attack will not help you get a better estimate of probabilities of evasion and therefore the evader must trust the advertised values they found through previous reconnaissance.

Consider Formulation (2), a modification of the shortest path interdiction problem where the interdictor assumes robustness in r. The evader's subproblem (2e) is only a minor modification from the subproblem we observed in SPI (1c). Note that the objective calculates the length of the evader's path using the arc length ℓ and incurring the **nominal** extra length $\bar{r}$ on arcs that have been interdicted. This differs from the original SPI's use of parameter r, but in this model we are assuming that the "true" value r is not known. The interdictor's objective (2a) uses the term $\bar{r} + Rv$, in place of the nominal added length $\bar{r}$, to account for the uncertainty. This represents the uncertain parameter with an affine expression that is dependent on v, an uncertainty vector, and R, an uncertainty matrix that maps and weights the uncertainty factors in v to the individual arcs in the network. Additionally, the interdictor's objective introduces a second adversary: the robustness! This phantom third player finds the worst case realization of the parameters within the geometry of the uncertainty set. To remain tractable, this geometry is assumed to be an ellipsoid (2b). The ellipsoid keeps all the parameters from going to their worst case value. Instead, a few parameters can be very off from their nominal values or many parameters can be slightly of from the nominal. A further discussion of how to define R and v can be found in Section 3.3.

$$
\begin{aligned}
\max_{x} \quad & \min_{v} \sum_{k \in A} (\ell_k + (\bar{r} + Rv)_k x_k) y_k && \text{(2a)} \\
\text{s.t.} \quad & ||v||_2 \leq 1, && \text{(2b)} \\
& \sum_{k \in A} x_k \leq B, && \text{(2c)} \\
& x \in \mathbb{B}^{|A|}, && \text{(2d)} \\
& y \in \text{argmin}_{\bar{y} \geq 0} \left\{ \sum_{k \in A} (\ell_k + \bar{r}_k x_k) \bar{y}_k : \sum_{k \in FS(i)} \bar{y}_k - \sum_{k \in RS(i)} \bar{y}_k = \begin{cases} 1 & i = S \\ 0 & \forall i \in N - \{S, T\} \\ -1 & i = T \end{cases} \right\} && \text{(2e)}
\end{aligned}
$$

The problem shown above contains two key features that make it difficult to solve. First, it is a trilevel model. This is due to the fact that it contains an inner minimization problem to handle the robustness, as well as an inner minimization problem for the evader. Techniques from bilevel programming provide guidance on how to remove the optimality condition on the evader's subproblem using duality theory. Second, it contains nonlinear terms in both the interdictor and evader objectives. Through knowledge of the problem specifics and techniques in robust optimization, we can remove those nonlinearities. The remainder of this section describes of a series of reformulations to turn this nonlinear trilevel formulation into an equivalent single level MILP with a second order cone constraint that is easily handled by most commercial solvers.

First, we replace the evader's subproblem with a collection of necessary conditions using a modification of its dual. For a given set of interdictions, the subproblem is a linear program. Consider Formulation (3), an alternative subproblem that finds the shortest path given **fixed** interdictions x^* that is a modification of the dual of the evader's subproblem (2e). This distance-based subproblem is a modification of the dual of the evader's subproblem that makes the distance variables non-negative (see Wolsey (2020) for the unmodified dual).

$$
\begin{aligned}
\max_{d} \quad & d_T && \text{(3a)} \\
\text{s.t.} \quad & d_S = 0, && \text{(3b)} \\
& d_j \leq d_i + \ell_{ij} + \bar{r}_{ij} x^*_{ij} \ \forall i \text{ adjacent to } j, \forall j \in N && \text{(3c)}
\end{aligned}
$$

The variable d_i represents the distance from the source node S to the node i. The objective d_T represents the distance from S to T. The constraint (3c) sets an upperbound on the distance to a node j by first finding the distance to its neighboring node i and then adding the length incurred by travelling from i to j. By setting this upperbound for all the nodes, when the objective maximizes d_T the maximization sets d_T equal to its tightest upper bound. At optimality, d_T is equal to the length of the shortest path.

Consider Formulation (4) which we argue is equivalent to Formulation (2). The third level of optimization has been replaced with equivalent constraints. This formulation has both the primal flow constraints (4e) and the dual distance upper-bounding constraints (3c). The objective of the primal and dual subproblems are required to be equal (4h), essentially setting the duality gap to zero. Because we have removed the evader's minimization, if we were to only use the primal constraints, the interdictor would set the evader's path to be as long as possible. However, because we also have the upper-bounding constraints, the interdictor must set the evader's path to at most have the length of the shortest path (d_T). Therefore, we see that Formulation (4) is equivalent to the original trilevel though we have removed the optimality condition on the evader's subproblem.

$$\max_{x,y,d} \quad \min_{v} \sum_{k\in A} (\ell_k + (\bar{r} + Rv)_k x_k) y_k \tag{4a}$$
$$\text{s.t.} \quad ||v||_2 \leq 1, \tag{4b}$$
$$\sum_{k\in A} x_k \leq B, \tag{4c}$$
$$x \in \mathbb{B}^{|A|}, \tag{4d}$$
$$\sum_{k\in FS(i)} y_k - \sum_{k\in RS(i)} y_k = \begin{cases} 1 & i = S \\ 0 & \forall i \in N - \{S,T\} \\ -1 & i = T \end{cases}, \tag{4e}$$
$$d_j \leq d_i + \ell_{ij} + \bar{r}_{ij} x_{ij} \;\; \forall i,j \in A, \tag{4f}$$
$$d_S = 0, \tag{4g}$$
$$d_T = \sum_{k\in A} (\ell_k + \bar{r}_k x_k) y_k, \tag{4h}$$
$$y \geq 0 \tag{4i}$$

Next, we consider the nonlinearities between x, y and v in (4a, 4h). First, the product of the interdictor's binary variable x and the evader's flow variable y can be avoided by splitting y_{ij} into w_{ij} and z_{ij}, which represent flow on an arc that has been interdicted and has not been interdicted, respectively. Like y_{ij}, w_{ij} takes a positive value if the evader traverses the arc from i to j and that arc is an interdicted arc, and similarly for z_{ij}. Either w_{ij} or z_{ij} can take a positive value, but not both. Formulation (5) shows the model without the nonlinear terms. Constraints (5i, 5h) require that z_{ij} can only be non-zero if $x_{ij} = 0$ and that w_{ij} can only be non-zero if $x_{ij} = 1$. The nominal path length in constraint (4h) would then be $d_T = \sum_{k\in A} (\ell_k z_k + (\ell_k + \bar{r}_k) w_k)$ and the robust path length in the overall objective function would be $\max_{x,w,z,d} (\min_v \sum_{k\in A} (\ell_k z_k + (\ell_k + (\bar{r} + Rv)_k) w_k))$. Second, the product between v and w in the modified objective function can be avoided by solving the following subproblem to find the worst-case v for a **fixed** $\bar{w}$ (Ben-Tal et al. (2009)):

$$\begin{aligned} \min_{v} \quad & (Rv)^T \bar{w} \\ \text{s.t.} \quad & ||v||_2 \leq 1 \end{aligned}$$

This subproblem has the closed form solution $v^* = \frac{-1}{||R^T\bar{w}||_2} R^T \bar{w}$ that will be used instead of v in the model. This means that the interdictor will calculate their objective using the worst-case manifestations of the uncertainty within the uncertainty set.

$$\max_{x,z,w,d,t,s} \quad \ell^T z + \ell^T w + \bar{r}^T w - s \tag{5a}$$

$$\text{s.t.} \quad \sum_{k \in A} x_k \le B, \tag{5b}$$

$$x \in \mathbb{B}^{|A|}, \tag{5c}$$

$$\sum_{k \in FS(i)} (w_k + z_k) - \sum_{k \in RS(i)} (w_k + z_k) = \begin{cases} 1 & i = S \\ 0 & \forall i \in N - \{S, T\} \\ -1 & i = T \end{cases}, \tag{5d}$$

$$d_j \le d_i + \ell_{ij} + \bar{r}_{ij} x_{ij} \forall ij \in A, \tag{5e}$$

$$d_S = 0, \tag{5f}$$

$$d_T = \sum_k (\ell_k z_k + (\ell_k + \bar{r}_k) w_k), \tag{5g}$$

$$z_k + x_k \le 1, \forall k \in A, \tag{5h}$$

$$w_k - x_k \le 0, \forall k \in A, \tag{5i}$$

$$t = R^T w, \tag{5j}$$

$$t^T t \le s^2, \tag{5k}$$

$$s, z, w \ge 0 \tag{5l}$$

Formulation (5) is a single level second order cone problem that can be input to most commercial solvers without any further reformulation. This model will be referred to as the shortest path interdiction with asymmetric uncertainty (SPIAU).

3.2. Multiple (S,T) Formulations

Another layer of uncertainty that appears in many applications of shortest path interdiction problems is where exactly the evader starts and ends their travels (Towle and Luedtke (2018)). There are applications where the defender wants to catch an evader from escaping a country or a city so any node outside of that country is a possible destination (Pan et al. (2003)).

Introducing a super source and/or a super sink to the network structure is a simple method of handling the uncertainty in the location of the evader's destination and source. This method does not change the model but modifies the network geometry by adding a super source (sink) that is connected to all possible sources (sinks) with arcs that have base length $\ell = 0$ and that cannot be interdicted $r = 0$. We will call this super source/sink method the SS-SPIAU to differentiate it from the single (S,T) application, though the mathematical model is the same.

In other instances, this uncertainty comes from multiple evaders on the network with different sources and sinks. To accommodate this application need, we will extend our model to multiple

(S,T) pairs. When multiple (S,T) are explicitly in the model, the interdictor must decide which objective they wish to maximize. One strategy could be to maximize the expected length over all (S,T) pairs. However, continuing our assumption that the interdictor is trying to find a solution that is reasonably conservative to worst-case outcomes, we propose a worst-case (S,T) pair version of our shortest path interdiction with asymmetric uncertainty (Formulation (6)) which aims to maximize the shortest path available to the evader.

$$\max_{x} \quad \min_{v^{(S,T)}} \min_{(S,T)\in I} \sum_{k} (\ell_k + (\bar{r} + Rv^{(S,T)})_k x_k) y_k^{(S,T)} \tag{6a}$$

$$\text{s.t.} \quad ||v^{(S,T)}||_2 \leq 1, \forall (S,T) \in I, \tag{6b}$$

$$\sum_{k\in A} x_k \leq B, \tag{6c}$$

$$x \in \mathbb{B}^{|A|}, \tag{6d}$$

$$y^{(S,T)} \in \text{argmin}_{\bar{y}\geq 0} \left\{ \sum_{k} (\ell_k + \bar{r}_k x_k)\bar{y}_k : \right. ,$$

$$\left. \sum_{k\in FS(i)} \bar{y}_k - \sum_{k\in RS(i)} \bar{y}_k = \left\{ \begin{matrix} 1 & i = S \\ 0 & \forall i \in N - \{S,T\} \\ -1 & i = T \end{matrix} \right\} \right\} \forall (S,T) \in I \tag{6e}$$

This model differs from the single (S,T) pair SPI with asymmetric uncertainty primarily in the duplication of the evader's subproblem for every (S,T) pair in the set of instances I. The uncertainty vector v is also duplicated to find the worst-case parameters for each (S,T) pair. Additionally, this model focuses the optimization of the objective on the worst (S,T) pair for the interdictor's objective, i.e. the (S,T) pair with the shortest path (evaluated in the interdictor's adjusted parameters).

Despite the extra level of optimization, this problem can be treated in the same way as the single (S,T) SPI with asymmetric uncertainty to reformulate it into a single level SOCP (Formulation (7)). Each subproblem can be replaced with the distance-based subproblem constraints. Nonlinearities in the objective function can be avoided by splitting the evader's flow variable y into flow on interdicted (w) and non-interdicted (z) arcs and by solving the robust subproblem to find the worst-case v in terms of a given w. The key difference can be found in (7b). The worst-case (S,T) can be identified by creating a new variable m that lower bounds the interdictor's objective value for each (S,T) pair. The interdictor then seeks to maximize m thus focusing on the (S,T) pair with the lowest objective value. This model (Formulation (7)) will be referred to as the worst-case (S,T) shortest path interdiction with asymmetric uncertainty (WC-SPIAU).

$$\max_{x,z,w,d,m,t,s} \quad m \tag{7a}$$

$$\text{s.t.} \quad \ell^T z^{(S,T)} + \ell^T w^{(S,T)} + \bar{r}^T w^{(S,T)} - s^{(S,T)} \geq m, \forall (S,T) \in I, \tag{7b}$$

$$\sum_{k\in A} x_k \leq B, \tag{7c}$$

$$x \in \mathbb{B}^{|A|}, \tag{7d}$$

$$\sum_{k \in FS(i)} (w_k^{(S,T)} + z_k^{(S,T)}) - \sum_{k \in RS(i)} (w_k^{(S,T)} + z_k^{(S,T)}),$$

$$= \begin{Bmatrix} 1 & i = S \\ 0 & \forall i \in N - \{S,T\} \\ -1 & i = T \end{Bmatrix}, \forall (S,T) \in I, \tag{7e}$$

$$d_j^{(S,T)} \leq d_i^{(S,T)} + \ell_{ij} + \bar{r}_{ij} x_{ij} \forall ij \in A, \forall (S,T) \in I, \tag{7f}$$

$$d_S^{(S,T)} = 0, \forall (S,T) \in I, \tag{7g}$$

$$d_T^{(S,T)} = \sum_k (\ell_k z_k^{(S,T)} + (\ell_k + \bar{r_k}) w_k^{(S,T)}), \forall (S,T) \in I, \tag{7h}$$

$$z_k^{(S,T)} + x_k \leq 1, \forall k \in A, \forall (S,T) \in I, \tag{7i}$$

$$w_k^{(S,T)} - x_k \leq 0, \forall k \in A, \forall (S,T) \in I, \tag{7j}$$

$$t^{(S,T)} = R^T w^{(S,T)}, \forall (S,T) \in I, \tag{7k}$$

$$(t^{(S,T)})^T t^{(S,T)} \leq (s^{(S,T)})^2, \forall (S,T) \in I, \tag{7l}$$

$$s^{(S,T)}, z^{(S,T)}, w^{(S,T)} \geq 0, \forall (S,T) \in I \tag{7m}$$

This worst-case method differs greatly from the super source/sink method. First, the worst-case model is significantly larger due to the duplication of the subproblem for each (S,T) pair in I. While this could lead to increased solve times for larger networks, the subproblem structure of the worst-case model could be leveraged to develop a solution method that reduces the overall computation time. This is a future extension of this model that is not discussed further in this work. Second, v is tailored to each (S,T) in the worst-case model, while it must be chosen for all (S,T) pairs simultaneously in the SS-SPIAU. Third, the SS-SPIAU lets the evader choose their path based on the nominal parameters while the worst-case (S,T) pair identified with the WC-SPIAU might not be the (S,T) pair chosen by the evader given the nominal parameters. Therefore, the SS-SPIAU works better when there is a single evader who is also evaluating which (S,T) pair to choose based on path length. The WC-SPIAU makes more sense when the interdictor is simply unsure of which (S,T) pair the evader will be using (or multiple evaders using multiple (S,T) pairs) and wants to make sure that the worst-case (S,T) is as protected as possible. These two approaches will be discussed further in Section 4.

3.3. Additional Modelling Techniques

The parameters that define the SPI (and our variants) are ℓ and r, the length of the arc and the length added to the arc if that arc has been interdicted. While this is a very useful formulation, many problems in network interdiction deal with probabilities where the defender and attacker seek to minimize and maximize the attacker's probability of evasion from S to T, respectively (Pan et al. (2003), Towle and Luedtke (2018)). An arc has a base probability of evasion p_{ij} and an interdicted probability of q_{ij}. With a few assumptions, probability network interdiction models can be converted into a shortest path model, solved, and converted back. Therefore, shortest path models discussed in this paper are also relevant to problems that use probabilities.

First, we assume that the probabilities are independent. Thus, the probability along a path is the product of the probabilities on each arc in that path. The conversion from probability to length is $\ell_{ij} = -\ln p_{ij}$. In the path SP, the length of the path is

$$L = \sum_{ij \in SP} -\ln p_{ij} = -\ln(\prod p_{ij})$$

To convert back, along a path of length L, the probability of evasion is e^{-L}.

This process still holds if arcs are interdicted, simply using q instead of p for the appropriate arcs in the product. Therefore, we must calculate r from p and q. Because r is a difference between the original length and the total length once interdicted, r must satisfy $\ell_{ij} + r_{ij} = -\ln q_{ij}$ because q is the probability of the interdicted arc. Therefore, $r_{ij} = -\ln q_{ij} + \ln p_{ij}$. This transformation will be used in the following examples.

While the network structure and parameters are necessary for the shortest path interdiction problem, the asymmetric uncertainty requires us to also define the uncertainty matrix R that links uncertainty factors to the additional arc length when interdicted. Specifically, not only do the values making up R need to be chosen but also the dimension of R. R always has a number of rows equal to the number of edges in the network but the number of columns needs to be equal to the dimension of the uncertainty vector v which is not defined by the network. We will describe two strategies of defining R and v: one that relies on data and problem-specific knowledge to define uncertainty factors that make up v and one that would work in all settings (including those without data and domain knowledge) by relying on simple bounds.

In the first strategy, ideally the uncertainty factors that make up v would be factors identified from the real-world setting of the problem. For a transportation example, factors could include weather in a particular region, traffic in a particular region, the type of road, etc. Those uncertainty factors are the components of the uncertainty vector v and R relates those factors to the specific arcs. For example, weather in a particular region would only have a non-zero value in R for arcs in that region. In this way, it is also possible to weight certain uncertainty factors to be more important or impactful than others. Note that though arcs are linked together by these uncertainty factors, the probabilities of evasion are still independent. The robustness is an external process setting the probabilities of the arcs but does not change that the probability of getting through one arc does not improve or lessen the traveller's chance of getting through the next arc.

However, if the problem does not have the data to identify uncertainty factors, the second strategy of defining of R and v is still applicable. We assume that R is dimensioned by the number of edges in the network both in rows and columns. In other words, R is a square matrix. If it is assumed that the length of the arc from i to j is bounded such that $r_{ij} \in [\bar{r_{ij}} + \sigma_{ij}, \bar{r_{ij}} - \sigma_{ij}]$, then let $R_{(ij)(ij)} = \sigma_{ij}$. The off-diagonal elements of R are 0. The uncertainty vector is also dimensioned by the number of edges and since the norm of the uncertainty vector is bounded, not all the lengths can be at their outer limits. This approach is simple because it requires minimal domain knowledge to define σ. Also, it reasonably represents that when estimating lengths it could be unlikely that all the lengths are significantly incorrectly estimated. It could be far more likely that either many lengths are slightly off or a small amount of lengths are significantly off. This method is utilized in (Ng et al. (2010)) in a robust linear programming application. This method will be used in the following examples.

We will briefly mention that special care should be taken if the original problem is a probability-based problem and the natural logarithm transformation is performed to convert it to a shortest path length-based problem. The ellipsoid uncertainty set and R are defined **within the transformation** and the shape of the possible probability parameters outside of that transformation are not necessarily an ellipsoid. Figure 2 visualizes the sets of possible parameters for two arcs where R is defined as described above. The top row shows the possible arc lengths when $p = 0.9$ and q is $0.1p$, $0.5p$, $0.9p$ from left to right. The bottom row shows the possible arc lengths when $p = 0.25$ and the same pattern of q. When q is a small fraction p, we observe parameter sets that are significantly less ellipsoidal and do not necessarily represent the parameters as intended. However, for problems where q is a large fraction p the parameter uncertainty sets closely resemble ellipsoids and behave as desired. This behavior occurs at both large and small values for p so we can see that it is a large ratio rather than a large difference between p and q that leads to the parameter set to distort away from the desired ellipsoid.

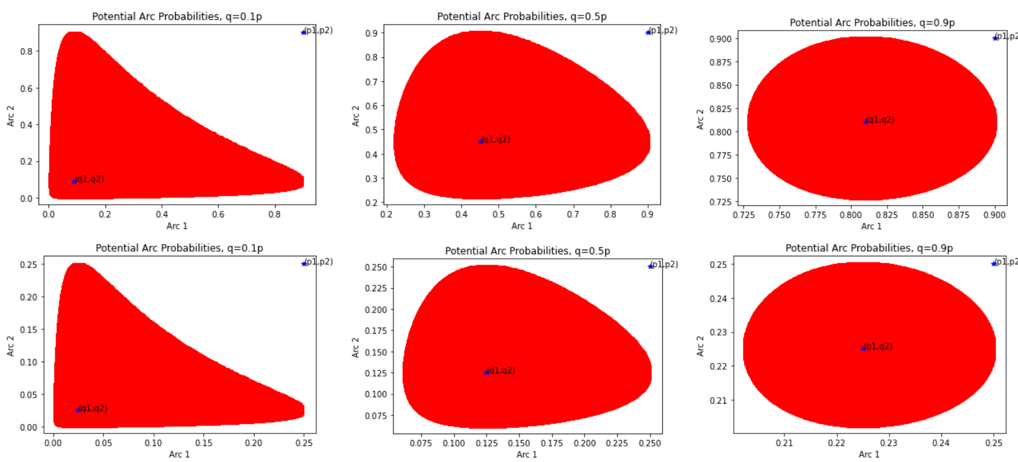

Figure 2: Parameter Set Geometries for Varying Ratios of p and q, Top Row: $p = 0.9$, Bottom Row: $p = 0.25$

4. Results

In this section, we investigate the benefits and computational difficulties of solving the shortest path interdiction problem with asymmetric uncertainty. Both single (S,T) and multiple (S,T) examples will be examined. All computational studies were performed on a consumer-end laptop with 31.8GB of RAM and an Intel i7 2.20GHz GPU. Pyomo 6.0 (Bynum et al. (2021), Hart et al. (2011)) was used to model the optimization problems and Gurobi 9.1 (Gurobi Optimization, LLC (2021)) was used to solve them.

4.1. Toy Example

Consider the toy example visualized in Figure 3. The probability of evasion p and probability of evasion when interdicted q are given in brackets for each arc: $[p,q]$. The source S and destination T are indicated and we will assume that the interdictor can interdict 2 arcs.

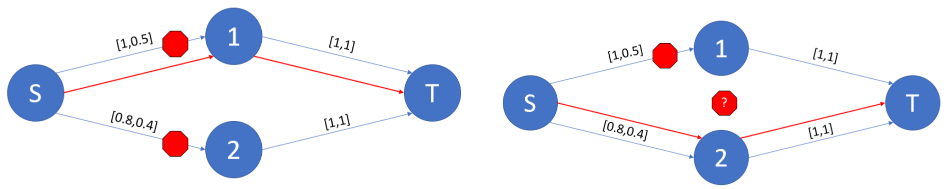

Figure 3: Toy Example Network: (L) SPI, (R) SPIAU

The left network visualizes the results of solving the shortest path interdiction problem. The interdictor interdicts the arcs (S,1) and (S,2) and the evader takes the upper path S-1-T with a probability of evasion 0.5 (as opposed to the lower path S-2-T which would have a probability of 0.4). The right network visualizes the results of solving the shortest path interdiction problem with asymmetric uncertainty. In this case, we have assumed that R is defined so the r can vary from 0 to $2\bar{r}$. In other words, it is possible that interdicting an arc will have the intended impact or no impact at all or anywhere in between. Since the objective for robustness works against the interdictor, it will never improve the impact of interdicting an arc. The results of this model show that the interdictor should only interdict the arc (S,1) and not use their second interdiction! This allows the evader to take the lower path with a probability of evasion of 0.8. While this is nonintuitive, consider the results without asymmetric uncertainty. Given our definition of R and the robustness, the worst-case manifestation of parameter uncertainty would be that the interdiction of the arc (S,1) does not impact the probability at all. The evader's probability of evasion when calculated with the adjusted probabilities would be 1! However, with the results using asymmetric uncertainty, the worst-case manifestation of the parameters does not change the probability of evasion because the evader is taking a path without interdictions. While the nominal probability of evasion is higher than SPI (0.8 versus 0.5), this strategy is reliable under worst case manifestations of the parameters (0.8 versus 1).

The strategy of luring the evader onto a slightly shorter path while avoiding paths with interdictions to provide a reliable solution is one commonly suggested when solving SPIAU. While "luring" the evader by making nonintuitive decisions might seem overly clever and not realistic in practice, the more concrete interpretation of this strategy is that it is not always the best choice to invest in the most obvious interdiction if your entire strategy hinges on that interdiction performing as expected. The SPIAU model highlights this reliance and offers an alternative strategy where you either don't rely on interdictions to perform to get to an acceptable result or have so many interdictions that the failure or underperformance of one does not catastrophically change the result.

We will quantify this reliability by defining the "regret avoided" which is calculated as follows. First, SPI is solved using the nominal parameter values. Then SPIAU is solved with the constraint that the interdictions are fixed to be the interdictions from the SPI solution. We denote the objective value from this fixed SPIAU case as z_1. Finally, SPIAU without the interdictions fixed is solved and the objective value is denoted z_2. The "regret avoided" is defined as RA $= \frac{z_1 - z_2}{z_1} * 100\%$. For our toy example, we observed that the objective value of SPIAU is $z_2 = 0.8$ and the objective value of SPIAU with the interdictions fixed to be the nominal choices is $z_1 = 1$. Therefore, the regret avoided is RA $= \frac{1-0.8}{1} * 100\% = 20\%$. A higher percentage of regret avoided is better because it indicates that solving the nominal SPI would have left the interdictor open to a very suboptimal solution if the worst-case manifestation of parameters were to occur.

4.2. Multiple (S,T) Variants Example

This section will utilize a real-world network to compare the worst-case SPIAU and the super source/sink SPIAU. The network visualized in Figure (4) with 24 nodes and 76 edges is a simplified representation of Sioux Falls, South Dakota available from (Transportation Networks for Research Core Team). This data does not contain probabilities of evasion p. We created evasion probabilities by taking the inverse of the length of each arc and then rescaling the values to a range between 0.1 to 1. The probability when interdicted q was set to be $q = 0.5p$. To test the multiple (S,T) models, we have defined I so that S can be any of the 10 nodes outside of the circle and T is node number 10 near the center of the circle. R was again defined to be a diagonal square matrix where $R_{(ij),(ij)} = r_{ij}$ so that the interdiction of an arc can have full impact, no impact, or impact anywhere in between. The number of arcs that can be interdicted was set to $B = 5$.

This model was solved using the super source (SS) and worst-case (S,T) modelling techniques to compare and contrast their approaches and results. Table 1 summarizes these experiments. The upper section of the table describes the results for the super source experiments, while the lower section of the table describes the results for the worst-case (S,T) experiments. For the SPIAU experiments (the right sections), nominal probability of evasion (PoE) describes the PoE calculated with the nominal parameter values (the evader's perception of the PoE), while the robust PoE is calculated with the worst-case manifestation of the parameters (the interdictor's perception of the PoE). The regret avoided is calculated as described in the previous section (Section 4.1).

We begin by analyzing the differences between the nominal and asymmetric uncertainty versions of the super source model (SS-SPI and SS-SPIAU). These values are found in the top section of Table 1. Four of the five interdictions are the same in both the SS-SPI and SS-SPIAU, but the SS-SPIAU chooses not to interdict the arc (5,9), instead interdicting the arc (3,4). Because the arc (16,10) is the only interdicted arc traversed by the evader in the SS-SPI model, if the worst-case parameters were to manifest, that interdiction would be impacted and the PoE would be increased. To avoid this, the SS-SPIAU recognizes that by opening up the arc (5,9), the evader will take a slightly higher PoE path traversing no interdicted arcs. This avoids over-reliance on the individual interdiction (16,10) and the corresponding significant regret.

Similarly, we will analyze the worst-case (S,T) variant results (WC-SPI and WC-SPIAU). In the WC-SPI results, we observe that the paths take the arcs to node 10 from nodes 9, 11, 15 and 16, all of which are interdicted. However, in the WC-SPIAU results, of the arcs entering node 10, only the arc from node 16 to node 10 is interdicted. We observe that the WC-SPIAU paths now primarily take arcs to node 10 from nodes 11 and 15, with the exception of the (S,T) pair (2,10) which uses the arc from 16 to 10. Again, we are now avoiding over-reliance on those interdictions

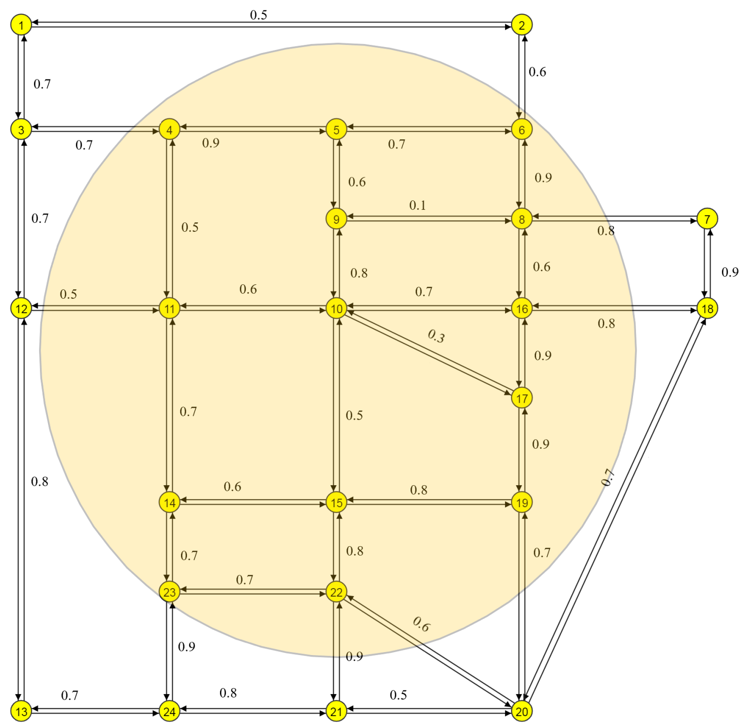

Figure 4: Sioux Falls Network

entering node 10 which would be impacted by the worst-case manifestation of the parameters. However, note that there are two (S,T) pairs that do not avoid any regret: (2,10) and (12,10). For both of these (S,T) pairs, the interdictor is unable to avoid relying on a single arc interdiction that fails under the worst-case manifestation of the parameters. Because the interdictor was unable to redirect the evader to a more reliable path, they should consider alternatives that might make the sensor placed on those arcs inherently more reliable. Finally, because this is the worst-case (S,T) variant, the solution to WC-SPIAU is not unique. The interdictor cannot reduce the robust PoE for the worst-case (S,T) pair (12,10) so any solution that keeps the robust PoE of all the other (S,T) pairs less than 0.3 is an optimal solution. There are potentially solutions that can reduce the robust PoE of these other (S,T) pairs further but as this does not change the objective of the problem, the WC-SPIAU model does not look for those solutions.

Finally, we will analyze the differences between the worst-case (S,T) and super source approaches. We observe that the evader's paths are the same for the SS-SPI and associated (S,T) pair in the WC-SPI . However, the SS-SPIAU and WC-SPIAU have different results, in both interdictions and paths. The SS-SPIAU evader chooses the path from 7 to 10 as the best for their PoE. It would be expected that the worst case (S,T) pair in the WC-SPIAU pair would also be 7 to 10. However,"worst case" in the WC-SPIAU is calculated using the **defender's robust parameter values** and is not necessarily the same as the (S,T) pair that the **evader** would choose using their

Table 1: Comparison of the Super Source and Worst-case (S,T) Shortest Path Interdiction with and without Asymmetric Uncertainty for the Sioux Falls network. Interdiction Budget B=5

	Super Source SS-SPI		**Super Source SS-SPIAU**			
	Interdict: (18,16), (5,9), (11,10), (15,10), (16,10)		Interdict: (18,16),(11,10), (15,10), (16,10) (3,4)			
(S,T)	Path	PoE	Path	Nominal PoE	Robust PoE	Regret Avoided
(20,10)	20-19-17-**16-10**	0.1984	–	–	–	–
(7,10)	–	–	7-8-6-5-9-10	0.2419	0.2419	39.04%

	Worst-case (S,T) WC-SPI		**Worst-case (S,T) WC-SPIAU**			
	Interdict: (18,16),(9,10), (11,10), (15,10), (16,10)		Interdict: (18,16), (5,9), (16,10), (21,22), (12,11)			
(S,T)	Path	PoE	Path	Nominal PoE	Robust PoE	Regret Avoided
(3,10)	3-4-5-**9-10**	0.1512	3-4-11-10	0.2099	0.2099	30.555%
(2,10)	2-6-8-**16-10**	0.1134	2-6-8-**16-10**	0.1134	0.2268	0%
(20,10)	20-19-17-**16-10**	0.1984	20-19-15-10	0.2799	0.2799	29.45%
(21,10)	21-22-**15-10**	0.1800	21-24-23-14-11-10	0.2117	0.2117	41.19%
(7,10)	7-8-**16-10**	0.1679	7-18-20-19-15-10	0.1763	0.1763	47.45%
(1,10)	1-3-4-5-**9-10**	0.1058	1-3-4-11-10	0.1469	0.1469	30.55%
(12,10)	12-**11-10**	0.1500	**12-11**-10	0.1500	0.3000	0%
(13,10)	13-12-**11-10**	0.1200	13-24-23-14-11-10	0.1852	0.1852	22.82%
(24, 10)	24-21-22-**15-10**	0.1439	24-23-14-11-10	0.2646	0.2646	8.12%
(18, 10)	18-7-8-**16-10**	0.1512	18-20-19-15-10	0.1959	0.1959	35.18%

nominal parameter values. In this network, the worst case (S,T) pair is (12,10) from the interdictor's perspective. However, the evader would choose to take the path from 20 to 10. Because of these differences in solutions, the super source model and the worst-case (S,T) model have different applications that they are suited for. The SS-SPIAU models the evader having the choice of all (S,T) pairs and picking the one with their perceived best chance of evasion. The WC-SPIAU models the defender being unsure of which (S,T) pair the evader will be using and want to get the worst case pair probability as low as possible, using their robust parameters to calculate that probability. It does not necessarily mean that the evader will be using that (S,T) pair. The evader might not be choosing the (S,T) pair based on their probability of evasion. For example, they might have a specific (S,T) already chosen. However, the defender might not know which (S,T) pair the evader will be using so they wish to focus their minimization on the worst-case probability of evasion of all the (S,T) pairs.

4.3. Computational Studies

While we have discussed the benefits of using the SPIAU model and reformulated it to be accepted by most commercial solvers, we must still discuss the computational cost of solving our model. This computational cost will be impacted by the size of the mathematical model which is driven by the size of the network. We created a set of networks generated by randomly placing N points in a unit square and connecting those nodes if they are less than radius p apart (Hagberg et al. (2008)). To keep the ratio of edges to nodes approximately fixed as we increased N, we let $p = \frac{1.5}{\sqrt{N}}$. If the graph was not connected, it was discarded and a new graph was generated. If the graph was connected, it was converted into a directional graph where every arc in the original graph became two arcs in opposite directions. We ran experiments at a variety of nodes (50, 100, 150 and 200) and a variety of interdiction budgets (5, 10, 15 and 20).

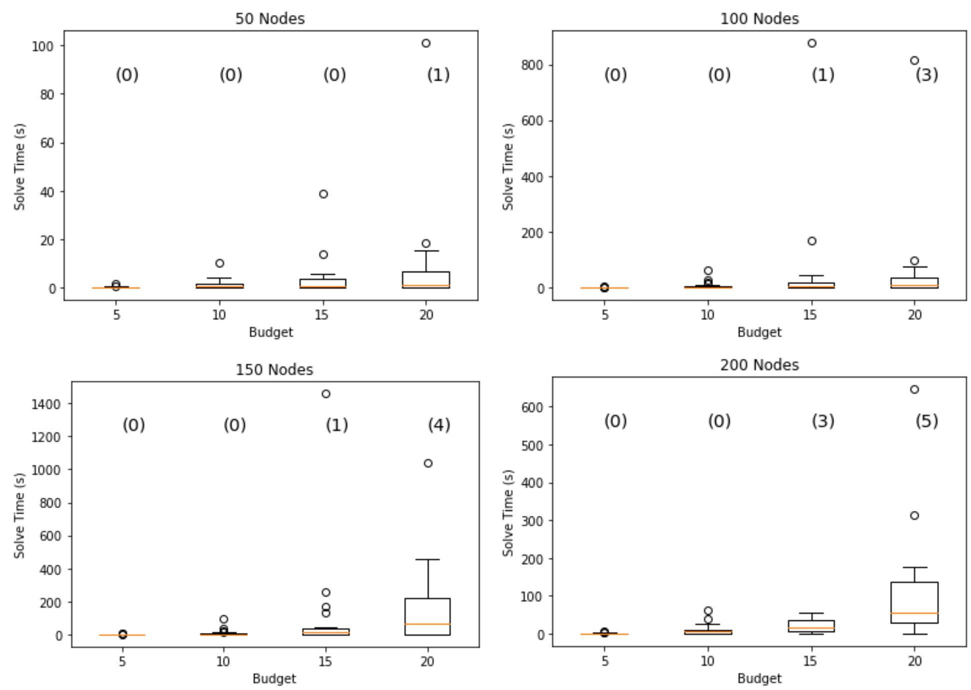

Figure 5: Solve Times by Nodes and Budget (SPIAU)

Table 2: Computational Results Nodes and Budget (times measured in seconds)

	B=5			**B=10**			**B=15**			**B=20**		
	Mean	SD	TOs	Mean	SD	TOs	Mean	SD	TOs	Mean	SD	TOs
N=50	0.278	0.297	0	1.322	2.321	0	4.000	8.797	0	9.181	22.897	1
N=100	0.484	0.583	0	6.987	14.664	0	62.513	200.945	1	68.083	194.664	3
N=150	0.950	1.246	0	10.521	22.396	0	114.773	332.887	1	165.284	267.799	4
N=200	1.646	1.687	0	10.904	15.200	0	22.777	17.395	3	120.039	167.325	5

Figure 5 and Table 2 summarize the solve times of twenty runs of each budget and number of nodes. For each combination of possible budget and nodes, the number of runs that hit the time limit are shown in parenthesis above the boxplot and in the timeout column labeled “TOs”. As expected, for a fixed budget, a larger number of nodes generally leads to longer solve times and a higher standard deviation as the MILP solved has more constraints and variables. This trend is not observed at budgets of 15 and 20 with 200 nodes. The mean and standard deviation actually decrease from the values seen at 150 nodes. However, this is likely due to the fact that the number of timeouts is very high at those budgets and that number of nodes. Because timeouts are not included when calculating mean and standard deviation, the mean appears to be have fallen because the long solve times are excluded. Similarly, we observe that for a fixed number of nodes, as the budget increases, so does the mean solve time and standard deviation. Again, the standard deviation sometimes falls at high budgets because of the higher number of timeouts reducing the data used in the calculation of the mean and reducing the standard deviation. Higher budgets mean more branching to decide the binary interdiction decisions. As more branching is required, the solve time increases as we need to explore more nodes. Additionally, the standard deviation increases as the number of nodes explored in the branching process varies more at higher budgets.

To study the computational cost of solving the worst-case SPIAU at various sizes of I, the set of possible (S, T) pairs, we use the Eastern Massachusetts transportation network, again available from (Transportation Networks for Research Core Team). This network has 74 nodes and 258 edges. Sources S and destinations T were randomly selected to make up I for each run. Smaller I are sampled from the larger I. This was run at budget $B = 5$.

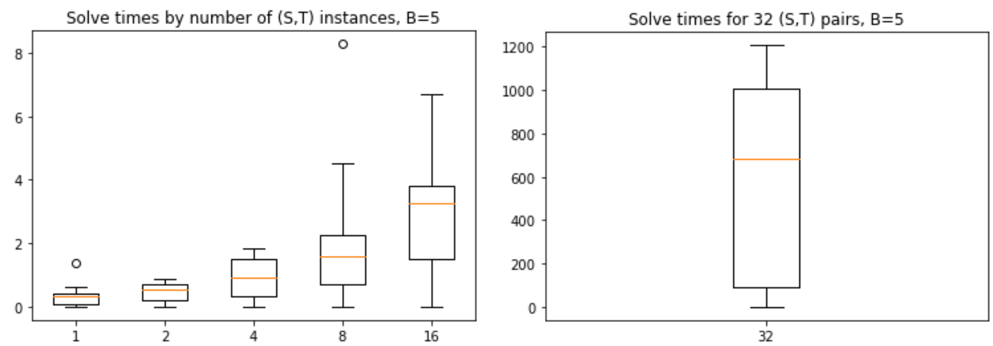

Figure 6: Multiple (S,T) Solve Times (WC-SPIAU) B=5

Intuitively, the solve time is larger for problems with more (S,T) pairs as each (S,T) pair in I adds another subproblem, increasing the number of variables and constraints. However, there is a significant increase in solve time at 32 (S,T) pairs. We observed in our testing that some (S,T) pairs lead to a significantly solve time. We have yet to succeed at quantifying why or predicting which pairs will exhibit this behavior. With regards to this computational experiment, it is hypothesized that this significant increase in solve time and variance is due to these "bad" (S,T) pairs having a higher chance to be in the set I when I is larger.

5. Conclusion

In this paper, we presented a novel variant of the shortest path interdiction problem that incorporates asymmetric uncertainty. We extended the SPI by assuming that the interdictor wants to be conservative to parameter uncertainty, specifically the additional length added when interdicted, r, without imposing their assumptions on the evader. It is assumed that the evader solves their problem with the nominal parameter values. We initially formulated this problem as a mixed-integer nonlinear trilevel problem. Using techniques from asymmetric information, linear bilevel programming and robust optimization, we were able to reformulate the model into a mixed-integer single level linear program with a second-order cone constraint. This form is easily handled by most commercial solvers. Additionally, a worst-case (S,T) variant was presented for applications with multiple sources and sinks. Finally, a brief computational study indicated how the size of the problem impacts the solve time.

We demonstrated the unique strategies highlighted by solving our models. Solving the shortest path interdiction problem with asymmetric uncertainty protects the interdictor from investing in the obvious strategy if that strategy hinges on key interdictions performing as promised. It also provides an alternate strategy that mitigates the risk of these worst-case possibilities.

References

Bayrak, H., Bailey, M.D., 2008. Shortest path network interdiction with asymmetric information. Networks 52, 133–140. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/net.20236, doi:https://doi.org/10.1002/net.20236, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/net.20236.

Ben-Tal, A., El Ghaoui, L., Nemirovski, A., 2009. Robust Optimization. Princeton Series in Applied Mathematics, Princeton University Press.

Besançon, M., Anjos, M.F., Brotcorne, L., 2020. Near-optimal robust bilevel optimization. arXiv:1908.04040.

Buchheim, C., Henke, D., 2019. The robust bilevel continuous knapsack problem with uncertain follower's objective. arXiv:1903.02810.

Bynum, M.L., Hackebeil, G.A., Hart, W.E., Laird, C.D., Nicholson, B.L., Siirola, J.D., Watson, J.P., Woodruff, D.L., 2021. Pyomo–optimization modeling in python. volume 67. Third ed., Springer Science & Business Media.

Gurobi Optimization, LLC, 2021. Gurobi Optimizer Reference Manual. URL: https://www.gurobi.com.

Hagberg, A.A., Schult, D.A., Swart, P.J., 2008. Exploring network structure, dynamics, and function using networkx, in: Varoquaux, G., Vaught, T., Millman, J. (Eds.), Proceedings of the 7th Python in Science Conference, Pasadena, CA USA. pp. 11 – 15.

Hart, W.E., Watson, J.P., Woodruff, D.L., 2011. Pyomo: modeling and solving mathematical programs in python. Mathematical Programming Computation 3, 219–260.

Israeli, E., Wood, R., 2002. Shortest-path network interdiction. Networks 40, 97–111. doi:10.1002/net.10039.

Kosmas, D., Sharkey, T.C., Mitchell, J.E., Maass, K.L., Martin, L., 2020. Interdicting Restructuring Networks with Applications in Illicit Trafficking. Technical Report. Rensselaer Polytechnic Institute. URL: https://arxiv.org/abs/2011.07093.

Ng, T.S., Sun, Y., Fowler, J., 2010. Semiconductor lot allocation using robust optimization. European Journal of Operational Research 205, 557–570. URL: https://www.sciencedirect.com/science/article/pii/S0377221710000299, doi:https://doi.org/10.1016/j.ejor.2010.01.021.

Nguyen, D.H., Smith, J.C., 2021. Network interdiction with asymmetric cost uncertainty. European Journal of Operational Research URL: https://www.sciencedirect.com/science/article/pii/S0377221721003908, doi:https://doi.org/10.1016/j.ejor.2021.04.055.

Pan, F., Charlton, W.S., Morton, D.P., 2003. A Stochastic Program for Interdicting Smuggled Nuclear Material. Springer US, Boston, MA. pp. 1–19. URL: https://doi.org/10.1007/0-306-48109-X_1, doi:10.1007/0-306-48109-X_1.

Smith, J.C., Song, Y., 2020. A survey of network interdiction models and algorithms. European Journal of Operational Research 283, 797–811. URL: https://www.sciencedirect.com/science/article/pii/S0377221719305156, doi:https://doi.org/10.1016/j.ejor.2019.06.024.

Song, Y., Shen, S., 2016. Risk-averse shortest path interdiction. INFORMS Journal on Computing 28, 527–539. URL: https://doi.org/10.1287/ijoc.2016.0699, doi:10.1287/ijoc.2016.0699, arXiv:https://doi.org/10.1287/ijoc.2016.0699.

Sullivan, K.M., Morton, D.P., Pan, F., Cole Smith, J., 2014. Securing a border under asymmetric information. Naval Research Logistics (NRL) 61, 91–100. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.21567, doi:https://doi.org/10.1002/nav.21567, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/nav.21567.

Towle, E., Luedtke, J., 2018. New solution approaches for the maximum-reliability stochastic network interdiction problem. Computational Management Science 15, 455–477. URL: http://dx.doi.org/10.1007/s10287-018-0321-1, doi:10.1007/s10287-018-0321-1.

Transportation Networks for Research Core Team, . Transportation networks for research. https://github.com/bstabler/TransportationNetworks.

Wolsey, L.A., 2020. Well-Solved Problems. John Wiley & Sons, Ltd. chapter 3. pp. 43–62. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119606475.ch3, doi:https://doi.org/10.1002/9781119606475.ch3, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781119606475.ch3.