Prioritizing Network Interdiction of Nuclear Smuggling

Dennis P. Michalopoulos, David P. Morton, and J. Wesley Barnes
Graduate Program in Operations Research and Industrial Engineering
The University of Texas at Austin
Austin, TX 78712, USA
Abstract

We develop a stochastic network interdiction model for prioritizing locations for installing radiation detectors along a nation’s border. In this one-country model, we characterize the smuggler population by a set of possible threat scenarios, where the identity of the smuggler is unknown at the time we install detectors. Detector performance depends on the threat scenario, as well as a number of additional factors such as terrestrial background radiation, geometric attenuation, and exposure time. Furthermore, the budget for installing detectors is unknown at the time the installation plan must be proposed. We model the budget as having a known probability distribution, and consequently, the solution to the problem is a rank-ordered priority list of installation locations, where one or more locations are assigned to each priority level. Upon its realization, we exhaust the budget by installing detectors at locations ranked from highest to lowest priority. The identity of the smuggler is subsequently revealed. Having full knowledge of the interdictor’s actions, the smuggler then selects an origin-destination path, which maximizes his evasion probability. Modeling the problem as a bilevel stochastic mixed-integer program, we present methods for strengthening the resulting formulation, exact and heuristic solution algorithms, and computational results. We also introduce a performance measure that quantifies the value of our prioritization model.
1 Introduction

Models for interdicting nuclear smuggling are motivated by documented incidents of illicit trafficking of nuclear material and by repeated attempts of terrorist groups to obtain a nuclear weapon. An International Atomic Energy Agency (IAEA) database on illicit trafficking reports over 1700 confirmed incidents from 1993-2009. About 20% of those involved “unauthorized possession and related criminal activities” and 15 involved highly-enriched uranium or plutonium [1].

Weapons-grade material has been seized by authorities in Russia, Germany, the Czech Republic, Lithuania, Bulgaria, Kyrgyzstan, Georgia, Greece, and France. In the majority of these cases, the material was found to have originated in Russia or other parts of the Former Soviet Union (FSU). This points to the vulnerability of our “first line of defense,” which deals with nuclear material protection, control, and accountability [31, 32]. While these efforts are critically important, they are by themselves insufficient, in part, because a complete inventory of the nuclear material that existed in Russia at the beginning of the 1990s is unrealistic.

Al Qaeda operatives have expressed a desire to carryout a nuclear attack on the US and Europe, and Al Qaeda has repeatedly attempted to obtain nuclear material, technology, and expertise. Osama bin Laden said that acquiring a nuclear weapon or other weapon of mass destruction is a “religious duty.” A radical Saudi cleric issued a fatwa authorizing use of nuclear weapons against American civilians. For further discussion of terrorist aims to acquire nuclear weapons, see, for example, Bunn [10, Chapter 2].

The threat that a terrorist could attack the US with a nuclear bomb may manifest itself in one of several ways. For instance, a terrorist may: (i) be supplied with nuclear material, or a weapon, via a rogue nation; (ii) purchase nuclear material smuggled out of the FSU; or, (iii) deploy a radiological dispersal device (RDD), comprised of more easily obtained radioactive material. The manner in which the Soviet Union dissolved, the pursuit of nuclear technologies by rogue nations, and the pursuit of nuclear energy by an increasing number of nations, have arguably increased the likelihood such threats could be realized. However, for that to occur, nuclear or other radioactive material must be smuggled across a transportation network. This provides an important opportunity to interdict the illicit material and smugglers [23].

The US Domestic Nuclear Detection Office (DNDO) is part of the Department of Homeland Security, and oversees the Global Nuclear Detection Architecture. As of FY 2007, this involved 74 federal programs and a budget of $2.8 billion, which includes $1.1 billion for combating nuclear smuggling internationally, $220 million for supporting detection at the US border, and $900 million for security and detection activities within the US [34]. One of the key international programs is called the Second Line of Defense (SLD), which since 1998, has worked to install and maintain radiation detectors at border crossings in Russia, countries of the FSU, and other key countries [33]. Domestic programs overseen by the DNDO carry out similar efforts for US border crossings.
There is a modest but growing literature in the operations research community related to detecting smuggled nuclear material. There are studies related to multi-layered, multi-detector, protocols at a single port [8, 16, 19, 25, 43, 44], analyses involving detection activities within, and around, a city [11, 26, 45], and work that involves interdicting nuclear smugglers on a larger transportation network [15, 29, 35]. Further related issues involve understanding the role of deterrence [6, 27], understanding the role of secrecy and deception [9, 39, 47], and seeking interdiction plans that are robust to a range of models of smuggler behavior [5, 20, 41].

The models of [15, 29, 35] are stochastic Stackelberg games involving a leader (an interdictor) and a follower (a nuclear smuggler). The interdictor installs radiation detectors on a transportation network knowing the smuggler’s origin-destination pair, and other characteristics we describe below, only through a probability mass function. After the detectors have been installed, a specific smuggler is revealed, who selects a path to maximize the probability of evading detection. The interdictor’s goal is to minimize the evasion probability, subject to a budget constraint, which limits the number of detectors that can be installed. In this chapter, we extend these models by assuming that the budget is uncertain at the time we must commit to a prioritization of the locations for detector installation. Like the work of Dimitrov et al. [15] and Morton et al. [29], we restrict attention to models in which the smuggler can encounter at most one detector on his origin-destination path. We call this a one-country model because this situation arises when all origins, or all destinations, are in one country, and we can only install detectors at checkpoints on the border of that country.

Under budget uncertainty, the interdictor postulates a set of budget scenarios with an associated probability mass function. The optimal solution to the prioritized model is a rank-ordered list of detector installation locations. After the budget is revealed, detectors are installed in order from highest to lowest priority until the budget is exhausted. In lieu of assuming a fixed budget, the prioritization model provides a well-hedged approach to decision making when resources that impact those decisions are not known with certainty. A second motivation for a prioritized solution is that detector installations—and more generally, other plans for hardening infrastructure—occur over time as resources become available. When a budget increment becomes available, we cannot reposition detectors, which have already been installed. In this second setting the probability mass function associated with the budget scenarios represents the relative likelihood of a smuggling attempt when the system is in the state associated with each budget increment. While the idea of a time-phased deployment of detectors applies, to keep the discussion concrete we restrict our lexicon to the former notion of having a probabilistic budget.

To illustrate our main idea, consider two fixed budget levels, $b' < b$. It is likely that the optimal detector installation plan under $b'$ is not a subset of the optimal plan under $b$. Suppose that we determine the optimal installation plan assuming budget level $b$, and then begin installing detectors at appropriate locations. Subsequently, we learn that our budget has been reduced to $b'$; i.e., our funds are insufficient to
complete the original installation plan. The subset of detectors already installed may have little overlap with the optimal plan under \( b' \). Thus, the resulting set of installations is inferior to the optimal set under this reduced budget.

Alternatively, suppose that we are given \( b \) as our installation budget for this year, and we are told that next year, there is the possibility of receiving additional funds, \( b'' - b \). Restated, in the first scenario, our total interdiction budget is \( b \), and in the second scenario, our total interdiction budget is \( b'' > b \). A myopic approach to solving this problem is to: (i) determine the optimal installation plan under \( b \); (ii) install detectors at the appropriate locations; and, (iii) if additional funds \( (b'' - b) \) are realized the following year, determine additional installation locations given that the first set is fixed. Once again, the first installation plan, assuming budget \( b \), may have little overlap with the optimal plan under \( b'' \), but given that the first set of detectors has already been deployed, we are forced to implement a potentially inferior overall interdiction plan.

A better approach to dealing with the situation just posed is to consider two budget scenarios with realizations \( b \) and \( b'' \), where each scenario is weighted according to the likelihood of its occurrence. With the optimal prioritized solution in hand, we install detectors at the highest-priority locations that are funded under \( b \), and if we obtain additional funds the following year, we install detectors at the remaining highest-priority locations until the total budget, \( b'' \), is consumed.

As a final motivating example, consider the following three interdiction plans superimposed on a map of Russia for budgets of \( b = 7, 8, \) and 9 detectors. These plans are optimal solutions to a instances of a model that we describe in Section 3, assuming a single budget scenario, known with certainty. In Figure 1a, we have labeled one of the locations as “Checkpt. 1.” When we view the interdiction plan in Figure 1b for \( b = 8 \), we see that Checkpt. 1 is no longer part of the solution. Instead, the solution now includes Checkpt. 2 and Checkpt. 3 on the eastern side of Russia. Finally, when \( b = 9 \), Checkpt. 1 rejoins the solution and is included in the interdiction plan with Checkpts. 2 and 3. Clearly certain smugglers have a similar preference for Checkpts. 2 and 3, and thus when \( b = 7 \), the optimal plan instead includes the choice of Checkpt. 1. With the additional budget provided under \( b = 8 \), the optimal plan discards Checkpt. 1 in favor of covering both Checkpts. 2 and 3, since this results in a greater decrease in smuggler evasion probability. When the budget value increases once again \( (b = 9) \), we have sufficient resources to cover all three checkpoints, as reflected in Figure 1c. These types of changes can be disconcerting to decision makers when \( b \) is uncertain, and in this setting, our prioritization model can provide a more robust approach for choosing a set of detector installation locations.

Some attention has been devoted to the idea of prioritization in the literature. Mettu and Plaxton [28] and Plaxton [36] consider a \( k \)-median problem with an uncertain budget. They seek a prioritized list of locations, which minimizes the competitive ratio, i.e., the worst-case ratio, over all values of \( k \), of the total distance customers must travel to their closest facility to the minimum distance when \( k \) is known. Lin
Figure 1: Installation plan for Russian customs checkpoints when $b = 7-9$, where detector installation locations are indicated by orange icons.
et al. [24] carry out further related work on other combinatorial optimization problems. Dean et al. [13] consider adaptive and non-adaptive versions of the stochastic knapsack problem. Both models seek to maximize expected reward, assume that the size of each item is a random variable, and that the knapsack’s residual capacity is realized only after selecting an item. See Dean et al. [12] for related work on more general stochastic packing problems. Allen et al. [2] and Seref et al. [40] consider incremental optimization problems, where a prespecified initial solution is given and then iteratively modified towards a specified goal.

The notion of solutions which are naturally nested at particular budget increments is studied by Witzgall and Saunders [46] and Hochbaum [21] in the context of the selection problem of Balinski [4] and Rhys [38]. Nehme and Morton [30] study the same issue of nestedness for models with submodularity of the benefit and cost functions. In this work, the budget increments at which solutions are nested is an output of a parametric optimization model. In contrast, in the model we describe here, these budget increments are instead specified by the decision maker; i.e., are input to the prioritization model. Koc et al. [22] propose the use of prioritization modeling for project portfolio selection under budgetary uncertainty. The prioritization scheme in Koc et al. [22] is the same that we consider here, but the underlying application differs.

The remainder of this chapter is organized as follows. Section 2 presents our approach to modeling smuggler threat scenarios, detection events, and estimating detection probabilities. In Section 3, we formulate our prioritized bipartite stochastic network interdiction problem (PrBiSNIP). In Section 4, we introduce a performance measure for quantifying the inherent value of our model. Section 5 discusses three methods for tightening the formulation. In Section 6, we present computational results related to two different bipartite smuggling networks: One for securing the Russian border, and one for securing the northern and southern borders of the contiguous United States. Section 7 concludes.

2 Computing Detection Probabilities

We briefly review how we model threat scenarios and compute associated detection probabilities. For further details see the discussion in Dimitrov et al. [14, 15] and Thoreson and Schneider [42]. We specify a threat scenario via the smuggler’s origin-destination (O-D) pair as well as: (i) the type of nuclear material being smuggled, e.g., weapons-grade plutonium, highly-enriched uranium, and spent nuclear fuel; (ii) the material’s mass of the material; (iii) shielding, e.g., lead and borated-polyethylene; and, (iv) the transporting vehicle. Two smugglers may have some overlapping characteristics (e.g., O-D pair and mass of nuclear material), but they may differ with respect to additional threat scenario parameters (e.g., shielding).

For a given threat scenario and for a given location and type of detector, the detection probability (DP) can be derived as a function of the threat characteristics, the survey strategy of the personnel operating the detector, and the detector alarm
algorithm. DPs can be estimated by simulating detector response with Monte Carlo N-Particle eXtended (MCNPX), which is a stochastic radiation transport code for modeling the interaction of radiation with its surrounding environment. Using MCNPX, alarm algorithms are applied to the results to translate count rate probability density functions into detection probabilities.

Accurate computation of DPs is complicated by a number of factors. Naturally occurring radioactive material (NORM) and other authorized radionuclide-bearing cargoes can induce false, i.e., “nuisance” alarms. NORM-bearing materials include fertilizer, cat litter, granite, and marble. Compared to background, radiation from most NORM increases detector count rates but does not dramatically alter spectra. Proper consideration of NORM can aid in mitigating nuisance alarms during the alarm algorithm design process. However, educated smugglers can also use NORM-bearing payloads to conceal nuclear material. Spectra for selected NORM can be obtained from published data sources such as [18], which provides spectra for tile and fertilizer.

Nuisance alarms occur when NORM causes the count rate to exceed the alarm algorithm’s threshold, triggering the alarm. Note that these are different from statistical false alarms, which occur when the threshold is exceeded in the absence of any local radioactive source, NORM or otherwise. A specific alarm threshold is typically selected to control the false alarm probability (FAP), which implies that alarm thresholds (and hence FAPs) are parameters that are under our control. The FAP also depends on interrogation time, which is a primary concern for portal detectors, where many short measurements generate a spatial spectrum profile.

Vehicle-specific characteristics are also important factors when considering FAP. Setting aside shielding by cargo, vehicle profiles must be considered because they shield background radiation. This baseline depression due to the vehicle profile can be approximately compensated for in real-time detection. An alarm algorithm can adjust thresholds based on baseline depression by vehicle type predicted using a particle transport model or empirical data. An extensive library of depressions by vehicle type and detector location can be found in [37]. Ely et al. [17] describe a second method which bins detector signals into energy windows, which is viable since most baseline depression does not strongly affect count ratios between energy bins.

The detection probabilities we compute via MCNPX serve as input for the prioritization models that we describe in the next section. In Section 5 we return to threat scenarios and show that if smugglers under multiple threat scenarios have identical rank-ordered preferences for the checkpoints they can traverse, then these threat scenarios can be aggregated to achieve an equivalent but smaller optimization model. This can greatly reduce the requisite scale of our prioritization model instances.

3 Prioritized Bipartite Stochastic Network Interdiction

In this section, we introduce our interdiction model (PrBiSNIP), whose goal is to minimize the probability a nuclear smuggler can evade detection on a bipartite trans-
portation network. The assumptions of our model are presented in Section 3.1, followed by its formulation in Section 3.2.

3.1 Model Assumptions

In PrBiSNIP, we assume that the characteristics of the smuggler threat scenario we will face are not known \textit{a priori}, but rather are governed by a known probability mass function (PMF). Smuggler evasion events on each arc are mutually independent, and the interdiction budget is assumed to be a random variable, independent of smuggler identity. For smuggler \( \omega \), the evasion probability for traveling through checkpoint \( k \) undetected, is given by \( q^\omega_k \) when a detector is installed, and \( p^\omega_k > q^\omega_k \) otherwise.

We assume that the interdictor and the smuggler have identical perceptions; i.e., they “agree” on all indigenous evasion probabilities at intermediate points throughout the network, as well as the evasion probability at each checkpoint, both with and without a detector installed. In the one-country interdiction model we consider, each smuggler can encounter at most one detector on an O-D path. As a result, the transportation network may be transformed to a network with bipartite structure, where one set of nodes consists of smuggler O-D pairs, and the other set of nodes consists of border checkpoints through which smugglers may choose to travel. An edge connects an O-D node to a checkpoint node, if a smuggler with this O-D pair can choose to traverse the associated checkpoint. We assume that each smuggler selects a path to maximize the probability of evading detection from his origin to his destination, knowing which subset of checkpoints have detectors. A known PMF is assumed to govern the threat scenarios.

3.2 Formulation

The timing of decisions, and observations of the stochastic parameters in PrBiSNIP, follows a three-stage process: First, the interdictor prioritizes the checkpoints, assigning them to a set of predetermined priority levels before the interdiction budget is realized. Second, after the budget is revealed, the interdictor installs detectors at locations ranging from highest to lowest priority until the budget is exhausted, where we assume unit installation costs. Third, the smuggler’s threat scenario is revealed, and having full knowledge of the interdictor’s actions, the smuggler travels from his origin, \( s^\omega \), through a checkpoint \( k \), and then to his destination, \( t^\omega \). The smuggler selects this path to maximize his evasion probability. In a preprocessing step, we can compute a path for each O-D pair via each intermediate checkpoint, and hence the smuggler’s path selection reduces to a choice of \( k \). Modeled as a stochastic mixed-integer program (MIP), the formulation of our problem is given as follows.
Sets, Indices, and Parameters

ω ∈ Ω  Sample space of smuggler threat scenarios
k ∈ K  Candidate locations for installing detectors (checkpoints)
K^ω ⊆ K  Subset of checkpoints smuggler ω may choose to traverse
β ∈ B  Sample space of budget scenarios
l ∈ L  Priority levels
p^ω_k  Smuggler ω’s evasion probability at k ∈ K if no detector is installed
q^ω_k  Smuggler ω’s evasion probability at k ∈ K if a detector is installed
(s^ω, t^ω)  Smuggler ω’s O-D pair
γ^ω_k  Product of ω’s evasion probabilities from s^ω to k and k to t^ω, k ∈ K^ω
n_l  Number of checkpoints assigned to priority level l (budget increment)
I^β_l  Equals 1 if budget scenario β funds priority level l, 0 otherwise
φ^ω  PMF on Ω
ψ^β  PMF on B

Decision Variables

x_{kl}  Equals 1 if checkpoint k ∈ K is assigned to priority level l, 0 otherwise

Formulation

Under budget scenario β, smuggler ω’s decision problem is given by

h(x, (s^ω, t^ω), I^β) = \max_{k ∈ K^ω} \left\{ γ^ω_k p^ω_k (1 - x^β_k), γ^ω_k q^ω_k x^β_k \right\}, \quad (1)

where x^β_k = \sum_{l \in L} I^β_l x_{kl}, x = (x_{kl})_{k ∈ K, l ∈ L} ∈ X, and

X = \left\{ x : \sum_{k ∈ K} x_{kl} = n_l, \sum_{l ∈ L} x_{kl} ≤ 1, \ x_{kl} ∈ \{0, 1\}, \ k ∈ K, \ l ∈ L \right\}.

The constraints in X state that: (i) exactly n_l checkpoints are assigned to each priority level l; (ii) any checkpoint can be assigned to at most one l; and, (iii) interdiction choices are modeled as binary decision variables.

Linearizing h as defined by equation (1), PrBiSNIP can be formulated as the
following stochastic linear MIP:

\[
\begin{align*}
    z^* &= \min_{x, \theta} \sum_{\omega \in \Omega} \sum_{\beta \in B} \phi^\omega \psi^\beta \theta^\omega \beta \\
    \text{s.t.} & \quad \sum_{k \in K} x_{kl} = n_l, \quad l \in L \quad (2b) \\
    \quad & \sum_{k \in K} x_{kl} \leq 1, \quad k \in K \quad (2c) \\
    \quad & x_\beta^k = \sum_{l \in L} I_\beta^l x_{kl}, \quad k \in K, \quad \beta \in B \quad (2d) \\
    \quad & \theta^\omega \beta \geq \gamma_k^\omega p_k^\omega \left(1 - x^\beta_k\right), \quad k \in K^\omega, \quad \beta \in B, \quad \omega \in \Omega \quad (2e) \\
    \quad & \theta^\omega \beta \geq \gamma_k^\omega q_k^\omega x^\beta_k, \quad k \in K^\omega, \quad \beta \in B, \quad \omega \in \Omega \quad (2f) \\
    \quad & x_{kl} \in \{0, 1\}, \quad k \in K, \quad l \in L. \quad (2g)
\end{align*}
\]

The objective function in (2a) is the evasion probability, formed by a weighted sum of evasion probabilities conditional on \(\omega\) and \(\beta\). Constraints (2b), (2c), and (2g) encode the requirements for a valid priority list. Constraint (2d) defines the value \(x^\beta_k\), which equals 1 if checkpoint \(k\) has a detector installed under budget scenario \(\beta\), and 0 otherwise. Constraints (2e) and (2f), coupled with the minimization in (2a), linearize the “max” in equation (1), ensuring \(\theta^\omega \beta = h(x, (s^\omega, t^\omega), I^\beta)\).

Model (2) can be simplified. Since \(p_k^\omega > q_k^\omega\) and \(x^\beta_k\) is binary, the right-hand side of (2f) can be replaced by \(\gamma_k^\omega q_k^\omega\), which yields \(\theta^\omega \beta \geq \gamma_k^\omega q_k^\omega\), \(k \in K^\omega, \ \beta \in B, \ \omega \in \Omega\).

Now, let \(\bar{q}^\omega = \max_{k \in K^\omega} \gamma_k^\omega q_k^\omega\), \(\omega \in \Omega\), and note that \(\bar{q}^\omega\) denotes the maximum evasion probability for \(\omega\) if every location \(k \in K^\omega\) receives a detector, i.e.,

\[
\theta^\omega \beta \geq \bar{q}^\omega, \quad \omega \in \Omega, \quad \beta \in B
\]  

(3)

can replace constraint (2f). If we let \(r^\omega_k = (\gamma_k^\omega p_k^\omega - \bar{q}^\omega)^+ \equiv \max\{\gamma_k^\omega p_k^\omega - \bar{q}^\omega, 0\}\), and \(\bar{\theta}^\omega \beta = \theta^\omega \beta - \bar{q}^\omega\), then model (2) can be transformed into the following stochastic MIP, where \(\bar{\theta}^\omega \beta\) has simple lower bounds of zero:
\[ z^* = \min_{x, \bar{\theta}} \sum_{\omega \in \Omega} \sum_{\beta \in B} \phi_{\omega \beta}^{\theta} x_{\omega \beta} \] (4a)

s.t. \[ \sum_{k \in K} x_{kl} = n_l, \quad l \in L \] (4b)

\[ \sum_{l \in L} x_{kl} \leq 1, \quad k \in K \] (4c)

\[ x_{\beta k}^\beta = \sum_{l \in L} I_l^\beta x_{kl}, \quad k \in K, \beta \in B \] (4d)

\[ \bar{\theta}_{\omega \beta} \geq r_{\omega}^\beta \left( 1 - x_{\beta k}^\beta \right), \quad k \in K^{\omega}, \beta \in B, \omega \in \Omega \] (4e)

\[ x_{kl} \in \{0, 1\}, \quad k \in K, l \in L. \] (4f)

The above transformation eliminates a total of \(|B| \sum_{\omega \in \Omega} |K^{\omega}|\) structural constraints from (2f), and replaces them with simple lower bounds, \(\bar{\theta}_{\omega \beta} \geq 0, \omega \in \Omega, \beta \in B\), which are absorbed in (4e). The linear programming (LP) relaxation of model (4) is at least as strong as that of model (2). The optimal values of models (2) and (4) are related by \(z^* = \bar{z}^* + \sum_{\omega \in \Omega} \phi_\omega \bar{q}_\omega\).

Model (4) has decision variables: \(x_{kl}, k \in K, l \in L\); \(x_{\beta k}^\beta, k \in K, \beta \in B\); and, \(\bar{\theta}_{\omega \beta}, \omega \in \Omega, \beta \in B\). When we focus on the priority list, we refer to \((x_{kl})_{k \in K, l \in L}\) as the solution of model (4). Other times, including the next section, we focus on which checkpoint receives a detector installation under a specific budget scenario, and in this case, we refer to \(x_{\beta k}^\beta = (x_k)_{k \in K}\) as the solution under budget scenario \(\beta\). Which notion we refer to as a solution will be clear from context.

### 4 The Value of Prioritization

As indicated in Section 1, optimal interdiction plans derived by assuming deterministic budgets can vary significantly under different values of the budget. As such, assuming a fixed budget in the face of uncertainty can produce a poor solution once the true budget is revealed. Prioritization deals effectively with this problem, but at a cost of additional model complexity. Thus, we seek to measure the benefit, or “value,” one gains from solving the prioritization model. Our approach is in the spirit of Birge’s [7] “value of the stochastic solution.” The typical benchmark is to compare the solution obtained by solving a stochastic program with that obtained when replacing the stochastic parameters with their expectations, but such an approach does not yield a reasonable priority list in our setting and we need another benchmark. The benchmark method we employ for obtaining a prioritized list is based on a greedy myopic procedure alluded to in one of our examples in Section 1.

Our myopic prioritized solution is constructed as follows. Let budget scenarios \(\beta_1, \beta_2, \beta_3, \ldots\) be such that \(b^{\beta_1} < b^{\beta_2} < b^{\beta_3} < \ldots\). First, for budget scenario \(\beta_1\), we solve the fixed budget problem, i.e., BiSNIP under budget value \(b^{\beta_1}\). (This is
equivalent to solving PrBiSNIP with a single budget scenario given by \( b^\beta_1 \).

Let \( z^\beta_1 \) and \( x^\beta_1 \) denote the optimal value and optimal solution for locating detectors, respectively, under this fixed budget. Next, after fixing each \( x^\beta_k = 1 \) for locations that received detectors under \( x^\beta_1 \), we proceed to budget scenario \( \beta_2 \), and then determine the optimal value, \( z^\beta_2 \), and solution under \( b^\beta_2 \), given that the detectors located under scenario \( \beta_1 \) cannot be relocated. We proceed to budget scenario \( \beta_3 \), following the same type of procedure of fixing checkpoints for receiving detectors that was used for \( \beta_2 \). The process continues for each subsequent budget scenario until the scenario with the largest budget realization has been considered.

Note that the myopic approach represents how a decision maker who does not account for budgetary uncertainty would augment an existing interdiction plan as the future unfolds and further funds become available. For each budget scenario, the procedure selects a subset of checkpoints without considering the possibility of obtaining additional funds for locating detectors.

Now, let \( z^* \) denote the optimal value of PrBiSNIP, as indicated in model (2), and let \( z^\beta \) denote the myopic value for each \( \beta \in B \). Then

\[
VoP(B) = \sum_{\beta \in B} \psi^\beta z^\beta - z^*,
\]

quantifies the value of prioritization via the difference between the value obtained from the nested myopic solutions, \( x^\beta \), \( \beta \in B \), and \( z^* \), the optimal value of model PrBiSNIP (2).

5 Improving the Formulation of PrBiSNIP

Computational experiments using a branch-and-bound algorithm (B&B) indicate that PrBiSNIP has a particularly weak LP relaxation. Even for modest-size problem instances, model (4) requires significant computational effort to solve to (near-) optimality. In this section, we begin by showing that the number of checkpoints that smuggler \( \omega \) might access for each \( \beta \in B \) is a function of the budget realization, \( b^\beta \), and we use this fact to tighten the formulation. Then, we describe conditions under which threat scenarios can be aggregated while maintaining an equivalent optimization model. Finally, we present a row generation algorithm that further tightens the formulation prior to employing B&B.

5.1 Restricting the Size of \( K^\omega \)

In constraint (4e), for each \( \beta \in B \), we consider all possible checkpoints that smuggler \( \omega \) can choose to travel through. However, we can restrict the size of \( K^\omega \), \( \omega \in \Omega \), depending on the value of \( b^\beta = \sum_{l \in L} I^\beta_l n_l \), i.e., on the value of the budget realization under scenario \( \beta \). For a fixed \( \omega \in \Omega \), let \( k_1, ..., k_{|K^\omega|} \) correspond to the ordering

\[
r^\omega_{k_1} \geq r^\omega_{k_2} \geq \cdots \geq r^\omega_{k_{|K^\omega|}},
\]
We define $K^{\omega\beta} \subseteq K^{\omega}$ as $K^{\omega\beta} = \{k_1, k_2, \ldots, k_m, k_{m+1}\}$, where $m = \min\{b^\beta, |K^{\omega}|\}$. Given (6), the definitions of $k_{m+1}$ and $r_{k_{m+1}}^{\omega\beta}$ are clear when $m < |K^{\omega}|$. If $m = |K^{\omega}|$ then $r_{k_{m+1}}^{\omega\beta} \equiv 0$, and in this case $k_{m+1}$ indexes an artificial checkpoint with an evasion probability of zero. The index ordering $k_1, \ldots, k_{|K^{\omega}|}$ depends on $\omega$ and the value of $m$ depends on both $\omega$ and $\beta$, but we suppress these dependencies to simplify notation. When the values of $r_k^\omega$ lead to ties in the ordering condition (6), those ties can be broken arbitrarily.

The intuition behind our construction is that if $m < |K^{\omega}|$ then under budget scenario $\beta$, smuggler $\omega$ will never choose to travel through any $k \notin K^{\omega\beta}$. This result comes from the fact that we can interdict at most $b^\beta$ of smuggler $\omega$’s checkpoints, and the assumption that smuggler $\omega$ always selects a checkpoint with maximum evasion probability in the residual network. Using these facts, it follows that

$$\bar{\theta}^{\omega\beta} \geq r_k^\omega - (r_k^\omega - r_{k_{m+1}}^{\omega\beta}) x_k^\beta, \ k \in K^{\omega\beta}, \ \omega \in \Omega, \ \beta \in B$$

(7)

is valid for model (4) and dominates constraint (4e).

To verify this claim, we first show that for each $(\omega, \beta)$, it suffices to consider only $k \in K^{\omega\beta}$. For any $\beta \in B$, the result is trivial for $\omega \in \Omega$ such that $m = |K^{\omega}|$, since this implies that $K^{\omega\beta} = K^{\omega} \cup \{k_{m+1}\}$ and $r_{k_{m+1}}^{\omega\beta} = 0$. Thus, we restrict our attention to the case when $m < |K^{\omega}|$. Constraint (4e) implies that $\bar{\theta}^{\omega\beta} = \max_{k \in K^{\omega}} \{r_k^\omega : x_k^\beta = 0\}$, $\beta \in B$, and since we can interdict at most $m = b^\beta$ checkpoints, it follows that $\bar{\theta}^{\omega\beta} \geq r_{k_{m+1}}^{\omega\beta}$ is valid for (4).

Next, we show that (7) is valid and dominates (4e) by considering what happens when (i) $x_k^\beta = 0$, and (ii) $x_k^\beta = 1$, $k \in K^{\omega\beta}$. If $x_k^\beta = 0$, then (7) reduces to $\bar{\theta}^{\omega\beta} \geq r_k^\omega$, i.e., (4e) and (7) are equivalent in case (i). If $x_k^\beta = 1$, then (7) reduces to $\bar{\theta}^{\omega\beta} \geq r_{k_{m+1}}^{\omega\beta}$, and (4e) reduces to $\bar{\theta}^{\omega\beta} \geq 0$. Since $\bar{\theta}^{\omega\beta} \geq r_{k_{m+1}}^{\omega\beta} \geq 0$, it follows that (7) dominates (4e), which verifies the claim.

We can once again reformulate PrBiSNIP through a transformation similar to the one described in Section 3. In particular, let $\tilde{\theta}^{\omega\beta} = \bar{\theta}^{\omega\beta} - r_{k_{m+1}}^{\omega\beta}$ and $\tilde{r}_k^{\omega\beta} = r_k^\omega - r_{k_{m+1}}^{\omega\beta}$, where we indicate the dependence of $\tilde{r}_k^{\omega\beta}$ on both $\omega$ and $\beta$ because $m$ depends on both these indices. Substituting $\tilde{\theta}^{\omega\beta}$ and constraint (7) into model (4) yields the following formulation:
\[
\begin{align*}
\min \sum_{\omega \in \Omega} \sum_{\beta \in \mathcal{B}} \phi^\omega \psi^\beta \tilde{\theta}^\omega \beta & \tag{8a} \\
\text{s.t.} \sum_{k \in K} x_{kl} = n_l, \ l \in L & \tag{8b} \\
\sum_{l \in L} x_{kl} \leq 1, \ k \in K & \tag{8c} \\
\tilde{\theta}^\omega \beta \geq \tilde{r}^\omega \beta \left( 1 - \sum_{l \in L} t^\beta_{l, kl} x_{kl} \right), \ k \in K^\omega \beta, \ \beta \in \mathcal{B}, \ \omega \in \Omega & \tag{8d} \\
x_{kl} \in \{0, 1\}, \ k \in K, \ l \in L. & \tag{8e}
\end{align*}
\]

Models (4) and (8) are equivalent. The advantage of the reformulation in model (8) is that its LP relaxation is at least as strong as that of model (4), and as our analysis suggests, it can in fact be stronger. For simplicity, we have eliminated variables \(x^\beta_k\) in presenting model (8). That said, we still employ these variables when convenient in the discussion below.

### 5.2 Threat Scenario Aggregation

We return to the notation of model (4). Consider two smugglers indexed by \(\omega_1, \omega_2 \in \Omega\), and for notational simplicity assume these two smugglers have the same set of checkpoints, denoted \(K\). Finally, suppose that for these two smugglers we have

\[
r^\omega \beta_{k_1} \geq r^\omega \beta_{k_2} \geq \cdots \geq r^\omega \beta_{|K|}, \text{ for } \omega = \omega_1 \text{ and } \omega = \omega_2 \tag{9}
\]

for \(k_1, k_2, \ldots, k_{|K|} \in K\). Restated, both smugglers, while they may have different evasion probabilities at some or all checkpoints, can rank-order the checkpoints in an identical manner.

The motivation for considering this situation in the context of the PrBiSNIP model can arise as follows. Smugglers \(\omega_1\) and \(\omega_2\) could be identical in every way, including their O-D pair and the type of material they carry, but the shielding and/or mass of the material that the two smugglers carry may differ. As a result, these two smugglers will be equally vulnerable to detection through indigenous law enforcement while traveling: from their origin to checkpoint \(k\); from checkpoint \(k\) to their destination; and, through checkpoint \(k\) if no detector is installed. In terms of the notation we introduce in model (2) we have \(\gamma^\omega_k = \gamma^\omega_{k_1} = \gamma^\omega_{k_2}\) and \(p^\omega_k = p^\omega_{k_1} = p^\omega_{k_2}\) for all \(k \in K\). The two smugglers are distinguished only by the evasion probability at a checkpoint \(k\) if a detector is installed at that checkpoint. For example, other factors being equal, a smuggler transporting material with greater mass will have a smaller evasion probability because he will provide a larger “signal” to a detector. Then, given the definitions of \(\tilde{q}^\omega\) and \(r^\omega_k = (\gamma^\omega_k p^\omega_k - \tilde{q}^\omega)^+\) from Section 3, we see that this leads to smugglers \(\omega_1\) and \(\omega_2\) rank-ordering their checkpoints in an identical manner.
In the setting above, there may be fewer positive values of, say, \( r_k^{\omega_1} \) than of \( r_k^{\omega_2} \), but they will satisfy the requisite ordering condition (9). The same result can arise under other conditions, e.g., when two smugglers have different O-D pairs in close geographic proximity to one another. Specifically, since \( r_k^\omega = (\gamma_k^\omega \bar{p}_k^\omega - \bar{q}^\omega)^+ \), two smugglers with different but identically ordered values of \( \gamma_k^\omega \bar{p}_k^\omega \) can be aggregated into a single scenario.

If we fix an interdiction plan, \( x = (x_{kl})_{k \in K, l \in L}, \in X \), then for \( \omega_1 \) and \( \omega_2 \) satisfying the ordering condition (9), we have \( k^* \in \arg\max_{k \in K} r_k^{\omega_1} (1 - x_k^\beta) \), and \( k^* \in \arg\max_{k \in K} r_k^{\omega_2} (1 - x_k^\beta) \), for the same checkpoint \( k^* \), which implies that \( \bar{\theta}^{\omega_1 \beta} = r_{k^*}^{\omega_1} \), and \( \bar{\theta}^{\omega_2 \beta} = r_{k^*}^{\omega_2} \). The contribution of \( \omega_1 \) and \( \omega_2 \) to the objective function is then given by \( \psi^\beta \left( \phi^{\omega_1} r_{k^*}^{\omega_1} + \phi^{\omega_2} r_{k^*}^{\omega_2} \right) \). Of course, \( x \) is not known ahead of time, but we can replace \( \omega_1 \) and \( \omega_2 \) by a single scenario, say \( \bar{\omega} \). In doing so, the objective function coefficient of \( \bar{\theta}^{\beta} \) is \( \psi^\beta (\phi^{\omega_1} + \phi^{\omega_2}) \), and the evasion probability of each \( k \in K \) equals

\[
\psi^\beta \left( \frac{\phi^{\omega_1} r_{k^*}^{\omega_1} + \phi^{\omega_2} r_{k^*}^{\omega_2}}{\phi^{\omega_1} + \phi^{\omega_2}} \right).
\]

Extending these ideas to \( n \geq 2 \) smuggler scenarios yields the following proposition.

**Proposition 1** Consider model (4) and let \( x \in X \). Let \( \Omega_n, n \in N \), partition \( \Omega \) such that every smuggler \( \omega \in \Omega_n \) rank-orders his checkpoints in the same fashion, i.e., for each \( n \in N \), \( \exists k_1, k_2, \ldots, k_{|K|} \) such that \( r_{k_1}^\omega \geq r_{k_2}^\omega \geq \cdots \geq r_{k_{|K|}}^\omega, \forall \omega \in \Omega_n \). Define \( \omega_n \) such that \( \phi^{\omega_n} = \sum_{\omega \in \Omega_n} \phi^{\omega} \) and \( r_k^{\omega_n} = \sum_{\omega \in \Omega_n} \phi^{\omega} r_k^{\omega}/\phi^{\omega_n} \). If \( \bar{\theta}^{\omega_n \beta} = \max_{k \in K} r_k^{\omega_n} (1 - x_k^\beta) \) then \( \phi^{\omega_n} \psi^\beta \bar{\theta}^{\omega_n \beta} = \psi^\beta \sum_{\omega \in \Omega_n} \phi^{\omega} \bar{\theta}^{\omega \beta} \).

**Proof.** Every \( \omega \in \Omega_n \) rank-orders their checkpoints the same. So, \( \exists k^* \) such that \( r_{k^*}^{\omega} = \max_{k \in K} r_k^{\omega} (1 - x_k^\beta), \forall \omega \in \Omega_n \). This same \( k^* \) maximizes \( \psi^\beta \sum_{\omega \in \Omega_n} \phi^{\omega} r_k^{\omega} (1 - x_k^\beta) \). This implies that

\[
\phi^{\omega_n} \psi^\beta \bar{\theta}^{\omega_n \beta} = \max_{k \in K} \phi^{\omega_n} \psi^\beta r_k^{\omega_n} (1 - x_k^\beta) = \max_{k \in K} \psi^\beta \sum_{\omega \in \Omega_n} \phi^{\omega} r_k^{\omega} (1 - x_k^\beta) = \psi^\beta \sum_{\omega \in \Omega_n} \phi^{\omega} \max_{k \in K} r_k^{\omega} (1 - x_k^\beta) = \psi^\beta \sum_{\omega \in \Omega_n} \phi^{\omega} r_{k^*}^{\omega} = \psi^\beta \sum_{\omega \in \Omega_n} \phi^{\omega} \bar{\theta}^{\omega \beta}.
\]
Under the conditions stated in Proposition 1, and using the notation defined in the same proposition, we can reformulate model (4) as

\[
\min_{\bar{x}_n, \bar{\theta}} \sum_{n \in N} \sum_{\beta \in B} \phi_{\omega_n} \psi_{\beta} \bar{\omega}_{\omega_n \beta} 
\]

\[
\text{s.t. } \sum_{k \in K} x_{kl} = n_l, \ l \in L 
\]

\[
\sum_{l \in L} x_{kl} \leq 1, \ k \in K 
\]

\[
\bar{\omega}_{\omega_n \beta} \geq r_{\omega_k} \left( 1 - \sum_{l \in L} \beta_{\omega_k} x_{kl} \right), \ k \in K_{\omega_n}, \ \beta \in B, \ n \in N 
\]

\[
x_{kl} \in \{0,1\}, \ k \in K, \ l \in L. 
\]

For each \( \beta \in B \) in model (10), \( r_{\omega_k} \) and \( \bar{\omega}_{\omega_n \beta} \) still represent conditional evasion probabilities, but are now conditioned on the event \( \Omega_n \). Thus, \( \phi_{\omega_n} = P(\Omega_n) \) denotes the probability of realizing a threat scenario in \( \Omega_n \). Extending the notation we introduce above, we use \( K_{\omega_n} \) to denote the checkpoints that smugglers \( \omega \in \Omega_n \) can traverse.

### 5.3 Row Generation Algorithm

We again return to the notation of model (4), and in this section, we present a class of valid inequalities for tightening the LP relaxation of PrBiSNIP. They are a straightforward generalization of the step inequalities derived to tighten the formulation of BiSNIP [29], i.e., SNIP on a bipartite network with a fixed budget level. The inequalities are also of the form of the star inequalities of Atamturk et al. [3]. These step inequalities can be identified on an as-needed basis by iteratively solving a shortest path separation problem. Let \( T^\omega = \{k_1, k_2, ..., k_n\} \subseteq K^\omega \) satisfy the ordering condition \( r_{k_1}^\omega > r_{k_2}^\omega > \cdots > r_{k_n}^\omega > r_{k_{n+1}}^\omega \equiv 0 \), where we require strict inequalities. Then the \( n \)-step inequality is given by

\[
\bar{\omega}_{\omega_n \beta} \geq r_{k_1}^\omega - (r_{k_1}^\omega - r_{k_2}^\omega)x_{k_1}^\beta - \cdots - (r_{k_{n-1}}^\omega - r_{k_n}^\omega)x_{k_{n-1}}^\beta - (r_{k_n}^\omega - r_{k_{n+1}}^\omega)x_{k_n}^\beta. 
\]

Step inequalities exploit the structure of PrBiSNIP through the rank-ordering that smugglers assign to their sets of accessible checkpoints. To see this, consider the two-step inequality \( \bar{\omega}_{\omega_n \beta} \geq r_{k_1}^\omega - (r_{k_1}^\omega - r_{k_2}^\omega)x_{k_1}^\beta - (r_{k_2}^\omega - 0)x_{k_2}^\beta \). If \( x_{k_1}^\beta = x_{k_2}^\beta = 0 \), then smuggler \( \omega \) selects \( k_1 \), and \( \bar{\omega}_{\omega_n \beta} = r_{k_1}^\omega \) by the two-step inequality. If \( x_{k_1}^\beta = 1 \), then \( \bar{\omega}_{\omega_n \beta} \) “steps down” to \( k_2 \) and reduces to \( \bar{\omega}_{\omega_n \beta} \geq r_{k_2}^\omega (1 - x_{k_2}^\beta) \), which is already an existing constraint in model (4), as well as reformulated models (8) and (10).

In general, when \( n \geq 2 \), the step inequalities are not redundant when \( x \) takes on fractional values in the linear programming relaxation of (4), leading to \( x_{k_1}^\beta \)-values that are fractional. The value of the step inequality is that it removes fractional
solutions that are otherwise feasible to the LP relaxation of PrBiSNIP. In the case of the two-step inequality, this can be seen by continuously increasing $x_{k_1}^\beta$ from 0 to 1, and then continuously increasing $x_{k_2}^\beta$ from 0 to 1.

Since there are an exponential number of step inequalities, adding all of them to the formulation of PrBiSNIP is not a practical solution approach. Our procedure for iteratively identifying a most violated inequality is based on a shortest path separation problem. The procedure takes as input $(x^{lp}, \bar{\theta}^{lp})$, the solution to the LP relaxation of PrBiSNIP, and maximizes the right-hand side of (11) by solving a shortest path problem on a directed acyclic network. The directed network has nodes $k \in K^\omega$ plus one additional node labeled $k_{|K^\omega|+1}$. We define a directed arc in the network from $k_j$ to $k_i$ with length $(r^\omega_{k_i} - r^\omega_{k_j})x_{k_i}^{\beta,lp}$ for each pair of nodes satisfying $r^\omega_{k_i} > r^\omega_{k_j}$, where $x_{k_i}^{\beta,lp} = \sum_{l \in L} I_{l}^\beta x_{kl}^{lp}$. We then find a shortest path from node $k_{|K^\omega|+1}$ to node $k_1$, and denote the set of nodes in this shortest path by $T^*$. We now present our row generation algorithm, which we describe in the context of model (4). However, we note that the same procedure applies to models (8) and (10), which we later consider in our computational results.

**PrBiSNIP Row Generation Algorithm**

**Input:** Optimality tolerance $\varepsilon > 0$

**Output:** $\varepsilon$-optimal priority list of interdiction locations, $x^* = (x_{kl}^*)_{k \in K, l \in L}$

**Step 1:** Solve the LP relaxation of (4) to obtain $(x^{lp}, \bar{\theta}^{lp})$.

**Step 2:** for each $(\omega, \beta)$ do

Step 2a: Let $T^*$ denote an optimal solution to

$$v^{\omega,\beta} = \min_{T \subseteq K^\omega} \sum_{k_i \in T} (r^\omega_{k_i} - r^\omega_{k_{|K^\omega|+1}})x_{k_i}^{\beta,lp}, \quad (12)$$

where $x_{k_i}^{\beta,lp} = \sum_{l \in L} I_{l}^\beta x_{kl}^{lp}$.

Step 2b: If $\bar{\theta}^{\omega,\beta,lp} < r^\omega_{k_1} - v^{\omega,\beta}$, add the step inequality defined by $T^*$ to (4).

end for

**Step 3:** If a step inequality was added for at least one $(\omega, \beta) \in \Omega \times B$, go to Step 1.

**Step 4:** Solve model (4), with the added step inequalities, to $\varepsilon$-optimality via B&B.

Any step inequality defined by $T \subseteq K^\omega$ which does not have $k_1 \in T$ satisfying
\( r^\omega_{k_1} = \max_{k \in K^\omega} \{ r^\omega_k \} \) is dominated by a step inequality with \( k_1 \in T \). Thus, when we solve the separation problem (12) in Step 2a, we are maximizing the right-hand side of inequality (11) over all \( T \subseteq K^\omega \). The condition \( \bar{\theta}^{\omega^\beta,lp} < r^\omega_{k_1} - v^\omega\beta \) implies that we have found a \( T^* \subseteq K^\omega \) for which inequality (11) is violated, and hence we add that inequality to model (4). After no more violated step inequalities are found, we solve the problem to \( \varepsilon \)-optimality by applying B&B to the tightened formulation.

We close this section with the following proposition, which provides conditions under which a step inequality for pair \((\omega, \beta)\) is valid for additional budget scenarios.

**Proposition 2** Assume that budget scenarios \( \beta' \) and \( \beta'' \) satisfy \( I^{\beta'} \leq I^{\beta''} \), where \( I^{\beta} \) denotes the vector \( I^{\beta} = (I^\beta_l)_{l \in L} \) which appears in constraint (4d). Furthermore, assume that \( T^* \) defines a most-violated step inequality for budget scenario \( \beta'' \), i.e., \( \bar{\theta}^{\omega^\beta,lp} < r^\omega_{k_1} - v^\omega\beta'' \). If \( \bar{\theta}^{\omega^\beta,lp} < r^\omega_{k_1} - v^\omega\beta'' \), then \( T^* \) also identifies a violated step inequality for \( \beta' \).

**Proof.** We have \( x^{\beta,lp}_k = \sum_{l \in L} I^\beta_l x^{\beta,lp}_{kl}, k \in K, \beta \in B \). By assumption we have \( I^{\beta'} \leq I^{\beta''} \), which implies \( \sum_{l \in L} I^{\beta'}_l x^{\beta,lp}_{kl} \leq \sum_{l \in L} I^{\beta''}_l x^{\beta,lp}_{kl} \), and hence \( x^{\beta',lp}_k \leq x^{\beta'' lp}_l \). This implies that \( v^{\omega^\beta'} \leq v^{\omega^\beta''} \Rightarrow r^\omega_{k_1} - v^{\omega^\beta'} \geq r^\omega_{k_1} - v^{\omega^\beta''} \), and so \( \bar{\theta}^{\omega^\beta',lp} < r^\omega_{k_1} - v^{\omega^\beta''} \leq r^\omega_{k_1} - v^{\omega^\beta'} \). It then follows that \( T^* \) defines a violated step inequality for \( \beta' \). \( \square \)

The value of Proposition 2 is that the identification of a most-violated step inequality for budget scenario \( \beta \) may imply a violated inequality for any scenario that funds fewer priority levels. The hope is that the early identification of violated inequalities will reduce the total number of separation problems, which need to be solved.

### 6 Computational Experiments and Results

In this section, we present computational results for two different one-country smuggling networks, one for bordering crossings leaving Russia and the other for border crossings entering the US. The former models are somewhat larger, and so in Section 6.1, we consider two different Russian model instances, and discuss results related to the computational methods presented in Section 5. The latter models have geographic structures that lead to interesting structural results, and so in Section 6.2, we discuss the value of prioritization (Section 4) within the context of a model designed to secure the northern and southern borders of the contiguous US. Finally, in Section 6.3, we return to the larger Russian models. For these models we illustrate and present results for a simple heuristic that generates granular priority lists \( (n_l = 1, l \in L) \) from “coarse” lists \( (n_l \geq 2, l \in L) \).

#### 6.1 Russian Model

Tables 1 and 2 show computational results for two Russian model instances: PrBiSNIP-R1 and PrBiSNIP-R2. These models differ only in that PrBiSNIP-R1 uses detectors with an alarm algorithm that accounts for the depression of background radiation due to the transporting vehicle while PrBiSNIP-R1 latter does not. (See the discussion
Both instances consist of 1320 scenarios and 265 checkpoints, resulting in $265 \times |L|$ binary decision variables. For each variant of our algorithm, we report total computation time (in seconds), and the number of generated step inequalities. Both problem instances are solved with a code implemented in a C++ programming environment on a 3.73 GHz Dell Xeon dual-processor machine with 8 GB of memory, using CPLEX version 10.1 with an absolute MIP solution tolerance of $\varepsilon = 0.0001$.

The results show that aggregating smuggler scenarios (Section 5.2) reduces computational effort more significantly than restricting their checkpoint sets (Section 5.1). We see that scenario aggregation can drastically reduce the number of step inequalities required to tighten the formulation. This outcome is not surprising, since the aggregation reduces the number of $(\omega, \beta) \in \Omega \times B$ combinations for the algorithm to consider. However, the value of restricting smuggler checkpoint sets is evident when implementing both preprocessing schemes together. The algorithm exhibits the best performance when using both methods, as indicated by the reported solution times and number of generated step inequalities in Tables 1 and 2.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>cpu no. ≥</td>
<td>cpu no. ≥</td>
<td>cpu no. ≥</td>
</tr>
<tr>
<td>5</td>
<td>27.53 1745</td>
<td>7.08 252</td>
<td>0.01 18</td>
</tr>
<tr>
<td>6</td>
<td>70.77 2365</td>
<td>11.94 316</td>
<td>0.02 26</td>
</tr>
<tr>
<td>7</td>
<td>137.87 3165</td>
<td>38.61 401</td>
<td>0.07 37</td>
</tr>
<tr>
<td>8</td>
<td>226.25 4645</td>
<td>46.99 461</td>
<td>0.07 56</td>
</tr>
<tr>
<td>9</td>
<td>358.09 5900</td>
<td>77.89 641</td>
<td>0.10 72</td>
</tr>
<tr>
<td>10</td>
<td>611.01 8771</td>
<td>154.10 923</td>
<td>0.16 115</td>
</tr>
<tr>
<td>11</td>
<td>1222.45 11497</td>
<td>536.41 1341</td>
<td>0.35 151</td>
</tr>
<tr>
<td>12</td>
<td>3017.01 15592</td>
<td>1710.92 1599</td>
<td>1.15 200</td>
</tr>
<tr>
<td>13</td>
<td>× × 5698.31 1937</td>
<td>6.23 268</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>× × × 5.69 313</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>× × × × 14.70 380</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>× × × × 55.83 452</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>× × × × 246.54 524</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>× × × × 252.61 611</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>× × × × 1701.24 718</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>× × × × × × ×</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Row generation results for PrBiSNIP-R1, $n_l = 1$, $l \in L$, where × indicates that we did not obtain a solution with absolute error within $\varepsilon = 0.0001$ of optimal within two hours. The columns under “Rest. Checkpts.”, “Scen. Agg.”, and “Both” are, in turn, the results obtained by restricting checkpoint sets based on budget scenarios (Section 5.1), threat scenario aggregation (Section 5.2), and both. For each problem instance, the term “cpu” denotes the time required to execute the algorithm of Section 5.3, and “no. ≥” denotes the number of generated step inequalities.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>cpu</td>
<td>no. $\geq$</td>
<td>cpu</td>
</tr>
<tr>
<td>5</td>
<td>36.58</td>
<td>1126</td>
<td>5.80</td>
</tr>
<tr>
<td>6</td>
<td>40.55</td>
<td>1621</td>
<td>11.12</td>
</tr>
<tr>
<td>7</td>
<td>60.19</td>
<td>2186</td>
<td>20.96</td>
</tr>
<tr>
<td>8</td>
<td>129.09</td>
<td>2767</td>
<td>28.28</td>
</tr>
<tr>
<td>9</td>
<td>166.40</td>
<td>3352</td>
<td>43.05</td>
</tr>
<tr>
<td>10</td>
<td>324.36</td>
<td>4775</td>
<td>120.18</td>
</tr>
<tr>
<td>11</td>
<td>311.09</td>
<td>5738</td>
<td>230.84</td>
</tr>
<tr>
<td>12</td>
<td>803.65</td>
<td>7411</td>
<td>593.30</td>
</tr>
<tr>
<td>13</td>
<td>4369.96</td>
<td>9363</td>
<td>3045.01</td>
</tr>
<tr>
<td>14</td>
<td>$\times$</td>
<td>$\times$</td>
<td>$\times$</td>
</tr>
<tr>
<td>15</td>
<td>$\times$</td>
<td>$\times$</td>
<td>$\times$</td>
</tr>
<tr>
<td>16</td>
<td>$\times$</td>
<td>$\times$</td>
<td>$\times$</td>
</tr>
<tr>
<td>17</td>
<td>$\times$</td>
<td>$\times$</td>
<td>$\times$</td>
</tr>
<tr>
<td>18</td>
<td>$\times$</td>
<td>$\times$</td>
<td>$\times$</td>
</tr>
<tr>
<td>19</td>
<td>$\times$</td>
<td>$\times$</td>
<td>$\times$</td>
</tr>
<tr>
<td>20</td>
<td>$\times$</td>
<td>$\times$</td>
<td>$\times$</td>
</tr>
</tbody>
</table>

Table 2: Row generation results for PrBiSNIP-R2, $n_t = 1$, $l \in L$. The table is to be read using the same conventions as in Table 1.
6.2 US Model

Figure 2 shows the 136 motor-crossing checkpoints we consider in the US model instances, and groups them into four clusters of checkpoints. Here, we consider 140 origin-destination threat scenarios, with half originating in Canada and the other half in Mexico, and we denote the model instance PrBiSNIP-US.

![Image of US border with motor-crossing checkpoints grouped into four clusters]

Figure 2: 136 motor-crossing checkpoints on the US border grouped into four clusters.

Similar to the example presented in Section 1, checkpoints in each cluster will enter, leave, and possibly reenter the solution under different budget levels, when we optimize under deterministic budget forecasts. For certain budget values, \( b \), significant changes in the optimal solution occur when there are just enough detectors to interdict an entire cluster of checkpoints. As shown in Figure 3, when the budget increases from \( b = 10 \) to \( b = 11 \), we have the ability to interdict all checkpoints on the US-Mexico border east of Big Bend. In Figures 4a and 4b, when the budget increases to \( b = 96 \) to \( b = 97 \), we can interdict all checkpoints from Mexico and all checkpoints from Canada, west of Lake Huron.

Given this type of geographic structure in the solution, we can construct budget scenarios where solutions are naturally nested. For example, consider the instance shown in Figure 5, where the prioritized and greedy myopic solutions (Section 4) coincide. Here, we see that \( \beta_1 \) funds 11 checkpoints, which is exactly enough to secure the US-Mexico border east of Big Bend. Budget scenario \( \beta_2 \) funds an additional 23 checkpoints, i.e., 34 in total, which is exactly enough to secure the remainder of the US-Mexico border to the west. Each budget scenario allows for the interdiction of
an additional cluster of checkpoints. Thus, when \( B = \{ \beta_1, \beta_2 \} \) with \( b^\beta_1 = 11 \) and \( b^\beta_2 = 34 \), and \( \psi^\beta_1 = \psi^\beta_2 = 0.5 \), the prioritized and greedy solutions coincide, which implies that \( VoP(B) = 0 \). In this case, there is no value in dealing with the additional complexity of the prioritization model, since the greedy myopic procedure discussed in Section 4 yields the same solution.

![Figure 3](image1)

(a) \( b = 10 \)  
(b) \( b = 11 \)

Figure 3: Part (a) of the figure shows the optimal solution to the US model instance with a budget to install detectors at \( b = 10 \) locations. Part (b) of the figure is identical but for \( b = 11 \). Note that the full number of checkpoints are not visible in the map due to their close proximity.

![Figure 4](image2)

(a) \( b = 96 \)  
(b) \( b = 97 \)

Figure 4: Part (a) of the figure shows the optimal solution to the US model instance with a budget to install detectors at \( b = 96 \) locations. Part (b) of the figure shows the optimal solution for \( b = 97 \).

Table 3 displays the value of prioritization under different sets of equally-likely budget scenarios for PrBiSNIP-US. In each instance, we report the number of budget scenarios, \(|B|\)”, the largest realization of the budget, \( “b^{\text{max}}” \), and \( “VoP(B)\%” \), i.e., the percent gap between the values of the greedy and the optimal prioritized solutions, relative to the optimal value \( z^* \). Here, all MIPs are solved within an absolute tolerance of \( \varepsilon = 0.001 \). In all problem instances, \( n_l = 5 \) for all \( l \in L \). For example, when \( |B| = 4 \), we have model (4), with budget realizations of \( b^\beta_1 = 5 \), \( b^\beta_2 = 10 \), \( b^\beta_3 = 15 \), and \( b^\beta_4 = 20 \); probability mass function \( \psi^\beta_1 = \psi^\beta_2 = \psi^\beta_3 = \psi^\beta_4 = \frac{1}{4} \); and, as shown
Figure 5: An example with two equally likely budget scenarios where the prioritized and greedy solutions are equal. Part (a) of the figure shows the optimal solution under $b_{\beta_1} = 11$ (i.e., $n_{l_1} = 11$) locations. Part (b) of the figure shows the solution for $b_{\beta_2} = 34$ ($n_{l_2} = 23$), under scenario $\beta_2$.

In Table 3, the prioritized solution has an evasion probability that is smaller by 1.84% over that obtained by the myopic heuristic that we describe in Section 4. As the table indicates, the value of the prioritized solution grows as the magnitude of the maximum budget realization and the number of budget scenarios, grows. When we have 10 equally-likely budget realizations of 5, 10, . . . , 50, the value of the prioritized solution is 24% better than that of the nested myopic solution. This suggests that in some settings, solving the optimal prioritization model can yield substantial improvements over what is arguably a very natural heuristic scheme.

| $|B|$ | $b_{\text{max}}$ | $VoP(B)\%$ |
|-----|-----------------|-------------|
| 1   | 5               | 0.00        |
| 2   | 10              | 0.00        |
| 3   | 15              | 1.02        |
| 4   | 20              | 1.84        |
| 5   | 25              | 2.34        |
| 6   | 30              | 2.88        |
| 7   | 35              | 9.57        |
| 8   | 40              | 15.31       |
| 9   | 45              | 20.31       |
| 10  | 50              | 24.02       |

Table 3: The value of prioritization (Section 4) for PrBiSNIP-US, expressed as a percentage of $z^*$, under different sets of equally likely budget scenarios.
6.3 Heuristic Generation of Granular Priority Lists

In this section, we consider the heuristic generation of “granular” priority lists \( (n_l = 1, \ l \in L) \) from “coarse” priority lists \( (n_l \geq 2 \text{ for some } l \in L) \). The motivation here is the following: A decision maker might be uncomfortable with placing multiple checkpoints on each priority level, but obtaining granular priority lists can be computationally expensive for large \( |B| \). For example, finding the optimal priority list for \( |B| = 20, n_l = 1, \ l \in L \), which contains 20 checkpoints, tends to be more computationally expensive than finding the optimal list for \( |B| = 5, n_l = 4, \ l \in L \), which also contains 20 checkpoints.

We present a heuristic approach to generating a granular priority list from a coarse list. To illustrate, consider an instance where \( n_l = 2, \ l \in L = \{l_1, l_2, l_3\} \), \( B = \{\beta_1, \beta_2, \beta_3\} \), and the optimal priority list is, notionally, given by

\[
x^* = (k_1, k_2, k_3, k_4, k_5, k_6).
\]

First, we map the coarse list defined by \( x^* \) to a granular list with one checkpoint at each priority level, i.e., \( n_l = 1, \ l \in L = \{l_1, l_2, l_3, l_4, l_5, l_6\} \), \( B = \{\beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6\} \). Next, a heuristic local neighborhood search—generated by permuting the components of \( x^* \) within each original priority level, one level at a time—is performed. Restated, we first evaluate both granular solutions generated by interchanging \( k_1 \) and \( k_2 \). For example, we can generate two new solutions by interchanging the priority levels of \( k_1 \) and \( k_2 \), which yields

\[
\hat{x}^1 = (k_2, k_1, k_3, k_4, k_5, k_6),
\]

\[
\hat{x}^2 = (k_1, k_2, k_3, k_4, k_5, k_6),
\]

and

\[
z^i = \sum_{\omega \in \Omega} \sum_{\beta \in B} \phi^{\omega} \psi^{\beta} \max_{k \in K^\omega} \left\{ r_k \left( 1 - \sum_{l \in L} I_l^{\beta} \hat{x}_{kl}^i \right) \right\}, \ i = 1, 2.
\]

Selecting the better of these two solutions, we proceed by fixing \( k_1 \) and \( k_2 \) at their current levels and carrying out the same procedure with \( k_3 \) and \( k_4 \). Finally, after \( k_1, k_2, k_3, \) and \( k_4 \) are fixed, we proceed by evaluating the two granular solutions generated by interchanging \( k_5 \) and \( k_6 \). If we let \( L^0 \) denote the original (coarse) set of priority levels, then the size of the solution neighborhood is equal to \( \sum_{l \in L^0} \binom{n_l}{2} \).

During the heuristic procedure, we track the best heuristic solution and objective function value, denoting them as \( \bar{x} \) and \( \bar{z} \), respectively. Thus, for any solution, \( \hat{x}^i \), if \( \hat{z}^i < \bar{z} \), we let \( \bar{x} \leftarrow \hat{x}^i \), and \( \bar{z} \leftarrow \hat{z}^i \). Clearly, \( \bar{z} \) yields an upper bound on the
optimal value of the granular list model, and we immediately obtain a lower bound via the optimal value of the coarse list model. Thus, if $z^*$ denotes the value of the optimal coarse list, then the difference between $\bar{z}$ and $z^*$ provides an estimate of the gap between $\bar{x}$ and the optimal granular list.

Table 4 shows the heuristic results for different coarse list structures. Here, we assume that $n_l = n, \forall l \in L$. In each case, we indicate $|B|$ and $n$ for the coarse list problem, and the resulting number of budget scenarios in the granular list problem. For each coarse list instance, we apply both smuggler scenario aggregation and restricted smuggler checkpoints, and solve the problem using the algorithm of Section 5.3. We report the total computation time, i.e., the total time required to solve the coarse list problem and carry out the heuristic and the estimate of the relative optimality gap.

<table>
<thead>
<tr>
<th>Coarse $(n &gt; 1)$</th>
<th>Granular $(n = 1)$</th>
<th>PrBiSNIP-R1</th>
<th>PrBiSNIP-R2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>B</td>
<td>$</td>
<td>$n$</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>10</td>
<td>0.60</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>15</td>
<td>1.26</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>20</td>
<td>4.53</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>25</td>
<td>220.94</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>12</td>
<td>1.03</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>18</td>
<td>3.90</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>24</td>
<td>69.10</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>30</td>
<td>373.21</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>14</td>
<td>1.46</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>21</td>
<td>19.06</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>28</td>
<td>4714.62</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>35</td>
<td>457.28</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>16</td>
<td>2.53</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>24</td>
<td>1065.14</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>32</td>
<td>5237.97</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>40</td>
<td>681.96</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>18</td>
<td>8.38</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>27</td>
<td>×</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>36</td>
<td>2132.86</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>45</td>
<td>1570.83</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>20</td>
<td>104.27</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>30</td>
<td>×</td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>40</td>
<td>2558.40</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>50</td>
<td>×</td>
</tr>
</tbody>
</table>

Table 4: Heuristic computational results for PrBiSNIP-R1 and PrBiSNIP-R2. Here, we consider coarse lists resulting from different combinations of $5 \leq |B| \leq 10$, and $2 \leq n_l \leq 5$, $l \in L$. For each instance, the term “cpu” denotes the total time (seconds) required to solve the coarse list problem plus the time to execute the heuristic. The term “rel. gap (%)” denotes the relative gap between the optimal coarse list and heuristic objective function values.

The best results are observed when the heuristic is applied to coarse list instances.
where \( n_l = 2 \), \( l \in L \). In many cases, the approximate granular solution is near-optimal, and requires far less computational effort than the optimal solution. Our procedure is able to quickly find approximate solutions to instances from Tables 1 and 2 that could not be solved to optimality in under two hours, e.g., \(|B| = 20\), \( n_l = 1 \), \( l \in L \).

7 Conclusion

We have presented a stochastic network interdiction model for installing radiation detectors along a nation’s border. We consider a variant of an interdiction model in which the budget is unknown at the time an installation plan is created, and hence the solution takes on the form of a rank-ordered priority list of candidate locations. The resulting stochastic MIP formulation can be tightened by aggregating smuggler threat scenarios, restricting smuggler border checkpoint choices within each budget scenario, and carrying out a separation procedure that iteratively generates members of a class of valid inequalities. Computational results show that the best results are obtained when all three procedures are implemented together. Additionally, we presented a heuristic approach for transforming “coarse” priority lists with two or more checkpoints on each level into “granular” lists with one checkpoint on each level. The heuristic is able to find near-optimal solutions to problem instances that could not be solved to optimality in under two hours by CPLEX, even with our algorithmic enhancements. The resulting upper and lower bounds suggest that the heuristic might aid in solving larger granular list problems via exact methods. Finally, we report results that measure the value of solutions from the optimal prioritization model over a natural myopic heuristic scheme for forming a prioritized list. We show that this value can be significant, and we give associated insight on the geographic structure of solutions for placing detectors for a US model instance.

Acknowledgements

This work has been supported by the National Science Foundation through grants CMMI-0653916 and CMMI-0800676, the Defense Threat Reduction Agency through grant HDTRA1-08-1-0029, and the US Department of Homeland Security under Grant Award Number 2008-DN-077-ARI021-05. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the US Department of Homeland Security.

25
Bibliography


[34] US General Accounting Office. Nuclear detection: Domestic Nuclear Detection Office should improve planning to better address gaps and vulnerabilities, 2009. GAO-09-257.


