ICS 2011
12th INFORMS Computing Society Conference

© 2011 INFORMS | ISBN 978-0-9843378-1-1
DOI 10.1287/ics.2011.0020

Interdiction of a Markovian Evader

Alexander Gutfraind, Aric Hagberg
Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545
{agutfraind.research@gmail.com, hagberg@lanl.gov}

David Izraelevitz, Feng Pan
Risk Analysis and Decision Support Systems, Los Alamos National Laboratory,
Los Alamos, New Mexico 87545 {izraelevitz@lanl.gov, fpan@lanl.gov}

Abstract Shortest path network interdiction is a combinatorial optimization problem on an activity network arising in a number of important security-related applications. It is classically formulated as a bilevel maximin problem representing an "interdictor" and an "evader." The evader tries to move from a source node to the target node along a least-cost path while the interdictor attempts to frustrate this motion by cutting edges or nodes. The interdiction objective is to find an optimal set of edges to cut given that there is a finite interdiction budget and the interdictor must move first. We reformulate the interdiction problem for stochastic evaders by introducing a model in which the evader follows a Markovian random walk guided by the least-cost path to the target. This model can represent incomplete knowledge about the evader, and the resulting model is a nonlinear 0-1 optimization problem. We then introduce an optimization heuristic based on betweenness centrality that can rapidly find high-quality interdiction solutions by providing a global view of the network.

Keywords network interdiction; stochastic optimization; guided random walk; betweenness centrality

1. Introduction

Mathematical modeling of network interdiction originated in the study of military supply chains and interdiction of transportation networks (Ghare et al. [11], McMasters and Mustin [16]). The problem is currently studied in different classes of networks and in a variety of contexts, and finds applications in countering of nuclear proliferation programs (Morton et al. [18]), control of infectious diseases (Pourbohloul et al. [22]), and disruption of terrorist networks (Memon and Larsen [17]). The underlying networks may represent transportation networks, as well as social or activity networks. Recent interest in the problem has been in part due to the threat of smuggling of nuclear materials and devices (Pan et al. [21]). Interdiction corresponds to the installation of special radiation-sensitive detectors across transportation links.

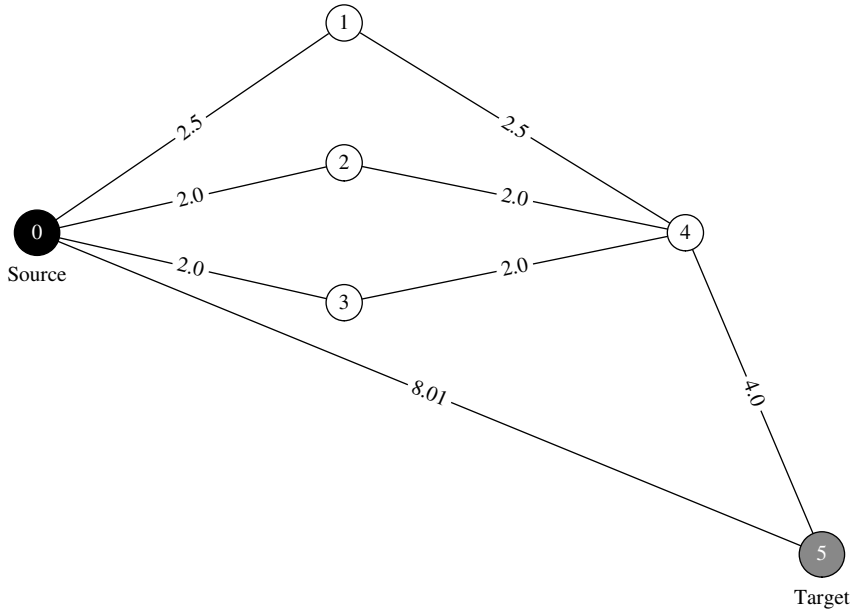
The problem is often posed in terms of two agents called "interdictor" and "evader" where the evader attempts to minimize some objective function in the network, e.g., distance, cost, or risk when traveling from network location s to location t, while the interdictor attempts to limit success by removing network nodes or edges. The interdictor has limited resources and can thus only remove a finite set of nodes or edges. In the simplest formulation, the interdictor seeks to identify a set of edges (or nodes) on the network whose removal maximizes the least cost from a source to a destination node, while the evader seeks to find and traverse a best unimpeded path. This interdiction problem is known as the "most vital edges" (or "most vital nodes") problem (Corley and Sha [8]) and it has been shown to be

NP-hard (Bar-Noy et al. [3]) and NP-hard to approximate to better than a factor of 2 (Boros et al. [6]). Methods for solving network interdiction problems have included exact algorithms for solving integer programs, such as branch-and-bound, as well as decomposition methods to rebuild the network by iteratively adding relevant paths to reduce the size of both the underlying network and the number of binary decision variables. A more recent approach, based on structure-dependent cutting planes, exploits the relationship between the ordered set of evasion paths and binary interdiction variables (Pan and Morton [20]).

A common assumption in many studies is that there is perfect knowledge of hard-to-compute network parameters, such as the cost to the evader to traverse an edge in terms of resource consumption or probability of detection. However, it is clear that the evader, and, to a lesser extent, the interdictor, have unreliable and incomplete information about the network. These uncertainties place the interdiction problem within stochastic optimization, where one seeks to find those edges that are vital *on average*. Indeed, under uncertainty the evader must be described in probabilistic terms. By constructing such probabilistic evader models one can expect to find good interdiction solutions even when the evader's motion cannot be fully known. The promise of such a solution has inspired several recent studies in stochastic interdiction (Morton et al. [18], Atkinson et al. [1], Bayrak and Bailey [5], Janjarassuk and Linderoth [15], Reich and Lopes [23], Gutfraind et al. [13], Dimitrov and Morton [9]).

Failure to account for evader uncertainty can lead to suboptimal decisions, namely, solutions that do not maximize (and even decrease) the evader's expected cost to reach the target. Consider for instance the network in Figure 1. There are four paths from the source to the target: one each through nodes 1, 2, 3 and the one direct path $(0, 5)$ with costs 9, 8, 8, and 8.01, respectively. If only one edge can be removed, the solution in the least-path-cost formulation is to remove edge $(4, 5)$ which increases the path cost from 8.0 to 8.01. However if the evader is unable to determine which path has the least cost and takes any path with equal (or nearly equal) probability, then this solution is not optimal. Interdiction at $(4, 5)$ actually *decreases* the expected cost from ≈8.25 to 8.01, because it removes the costly path

FIGURE 1. Example network where the shortest path interdiction formulation produces a suboptimal solution when interdicting a single edge.



Notes. Interdicting that edge $(4, 5)$ decreases the expected path cost. Interdicting any one of $(0, 2)$, $(2, 4)$, $(0, 3)$, or $(3, 4)$ increases the expected path cost.

through node 1. The optimal choice is interdiction of any one of the edges $(0,2)$, $(2,4)$, $(0,3)$, or $(3,4)$, which increases the expected cost from $\approx$8.25 to $\approx$8.33.

This example illustrates the effects of evader randomness on interdiction choices. If the evader chooses suboptimal (more random) paths but the interdictor assumes the evader acts optimally then the interdiction solution can actually help the evader. In the opposite case, when the evader always chooses optimal paths but the interdictor assumes suboptimal evader choices, interdiction cannot possibly help the optimal evader: the least-cost path cannot become shorter if some edges were cut.

In this paper we introduce a Markovian network interdiction model which can describe a wide range of network evaders (§2). We then demonstrate the general framework with a simple model based on evader decision-making mechanisms (§4). Finally we develop efficient heuristic algorithms for the interdiction problem based on the structure of the graph and then present performance results comparing various heuristic methods (§5).

2. The Interdiction Model

Our interdiction formulation is a stochastic generalization of the max-min shortest path interdiction problem (termed the "least-cost path" interdiction problem, to be exact) (Ghare et al. [11], McMasters and Mustin [16], Israeli and Wood [14]). In the least-cost path formulation an evader attempts to traverse a network on a path from an origin s to a destination t. Let p be some path between s and t in a graph $G(N,A)$ with the set of nodes N and the set of weighted edges A. In this paper, a path is a sequence of edges of the form, $(s,i_1),(i_1,i_2),\ldots,(i_k,t)$, and any node, except t, can be visited more than once on a path. Let $c(p)$ be the path cost computed by summing the cost C_{ij} over the edges (i,j) of p, and any self-looped edge has zero cost, $C_{ii}=0$. The edge costs are assumed to be given in the problem and may depend on direction (in the case that $G(N,A)$ is a directed graph). Here "edge cost" is used interchangeably with "edge weight."

The network interdiction strategy is represented by an interdiction set X which is a subset of the edge set A of cardinality b (budget). The decision variable x_{ij} is set to 1 if edge $(i,j)\in X$, i.e., (i,j) is interdicted, and $x_{ij}=0$ otherwise. Interdiction increases the cost of traversing (i,j) by a constant $D_{ij}\geq 0$. One may write $C'_{ij}=C_{ij}+x_{ij}D_{ij}$ but it is more convenient to use C_{ij} at all times to denote cost that includes possible interdiction. This makes the matrix $\mathbf{C}$ a function of X. When the value of D_{ij} is very large all paths avoid the interdicted edge (i,j) (assuming that there is an alternative path) which effectively removes the edge (i,j) from the graph. Such an edge-cost model of interdiction is more general than the model based on outright cutting of edges, and avoids damaging the graph topology (cases where nodes become cut off from the target).

In the shortest path model, the evader only travels on paths of lowest cost, and is fully aware of increases in edge costs caused by interdiction decisions. This gives the optimization problem

$$\max_{X\subset A,\,|X|=b}\ \min_{p\in PT} c(p), \tag{1}$$

where $c(p)$ is implicitly a function of X, and PT is the set of paths from s to t. The above formulation is for interdiction of edges but of course, a similar problem could be considered for node interdiction (by introducing for all $i\in N$ node costs D_i and decision variables on nodes, x_i).

A stochastic version of the interdiction problem can be constructed by supposing that an evader may take any path from s to t, according to some probability distribution, rather than always choosing a least-cost path. Randomness in the evader path decision could be due to our lack of knowledge of how the evader travels through the network. It could also be caused by the evader's uncertainty about interdiction decisions X or network costs, mistaken cost

computations, or possibly even by intent to increase unpredictability. Suppose the evader selects path p with probability $P(p)$. The expected cost of traveling from s to t is then

$$E[c] = \sum_{p \in PT} P(p)c(p). \tag{2}$$

The interdiction problem becomes

$$\max_{X \subset A, |X|=b} \sum_{p \in PT} P(p)c(p), \tag{3}$$

where $P(p)$ is now the probability of traversing a path. Both $P(p)$ and the path cost $c(p)$ depend on the interdiction set X, but the notation here will not represent this dependence since it is pervasive: from the evader's point of view, the interdiction set is part of the graph structure. The interdiction set X will be discussed again only in §5 dealing with interdiction algorithms.

The probability $P(p)$ implicitly contains the evader's strategy. The shortest-path optimization problem (1) is clearly just a special instance of (3) when the expectation is conditioned on traversal of only least-cost paths.

To compute the expected cost $E[c]$, we rewrite it in terms of the edge costs and the number of visits to each edge. If F_{ij} is the expected number of visits of edge (i, j) by an evader, then

Lemma 1.

$$E[c] = \sum_{p \in PT} P(p)c(p) = \sum_{(i,j) \in A} C_{ij}F_{ij}. \tag{4}$$

Proof. By definition $F_{ij} = \sum_{p \in PT:\,(i,j) \in p} P(p)$, and F_{ij} can in general be larger than 1 because paths may revisit (i, j). The equivalency follows as

$$\begin{aligned} E[c] &= \sum_{p \in PT} P(p)c(p), \\ &= \sum_{p \in PT} P(p) \sum_{(i,j) \in p} C_{ij}, \\ &= \sum_{(i,j) \in A} C_{ij} \sum_{p \in PT:\,(i,j) \in p} P(p), \\ &= \sum_{(i,j) \in A} C_{ij}F_{ij}. \qquad \square \end{aligned}$$

The expected cost $E[c]$ is now expressed through the expected number of visits to all edges (the F_{ij} values). The latter quantity may be hard to compute in general because every evader path could in principle visit edge (i, j), while the number of possible paths can be very large and even unbounded. Fortunately, one particular class of stochastic models—Markov chains—gives a closed-form expression for F_{ij}.

3. Markovian Evaders

We model the stochastic evader as a Markov chain that has its states at the nodes of the network. In the most general case, the chain is completely described by (1) a distribution of starting nodes, $\mathbf{a}$, and (2) a Markovian transition probability matrix, $\mathbf{M}$. In the next section, we will provide derivations of $\mathbf{M}$ for some realistic applications by examining the decision-making mechanisms of a rational evader frustrated by uncertain information. Such an evader tends to make transitions to move closer to the target.

Consider for now the most general case. The motion of the evader is just a Markov chain with an absorbing state at the target node t. An element M_{ij} of the transition probability

matrix is the probability of motion from node i to node j along edge (i,j). The matrix $\mathbf{M}$ must satisfy two conditions: (1) Absorption at t: $M_{tt}=1$ and $M_{ti}=0$ for all $i\neq t$, and (2) Access to t: from any starting state $i\neq t$ there is a positive probability of reaching state t in a finite number of transitions. Because of condition (1) the transition matrix of an absorbing Markov chain can be arranged into the following canonical form:

$$\mathbf{M}=\begin{pmatrix}\widehat{\mathbf{M}} & \mathbf{R}\\ \mathbf{0} & 1\end{pmatrix}.$$

Here the matrix $\widehat{\mathbf{M}}$ ($n-1$ by $n-1$) contains the transition probabilities among transient states. The matrix $\mathbf{R}$ ($n-1$ by 1) specifies the probabilities of transition from the transient states to the absorbing state.

Similarly, the edge cost matrix for an absorbing Markovian evader takes a specific form:

$$\mathbf{C}=\begin{pmatrix}\widehat{\mathbf{C}} & \mathbf{S}\\ \mathbf{Z} & 0\end{pmatrix}.$$

Here the matrix $\widehat{\mathbf{C}}$ ($n-1$ by $n-1$) contains the costs for transition among transient states. The matrix $\mathbf{S}$ ($n-1$ by 1) specifies the costs for moving to the absorbing state, while $\mathbf{Z}$ (1 by $n-1$) are cost for edges out of the absorbing states—those edges are never traversed. The element $C_{tt}=0$ implies that there is no cost to remain at the target node t.

Given the matrix $\widehat{\mathbf{M}}$, the *Fundamental Matrix* $\mathbf{N}$ of the chain is

$$\mathbf{N}=(\mathbf{I}-\widehat{\mathbf{M}})^{-1}.$$

Theorem 1. *Element N_{ij} of the fundamental matrix gives the expected number of visits to state j if starting at state i* (*Grinstead and Snell* [12, *Theorem* 11.4]).

In general the starting state of the evader is given by a distribution $\mathbf{a}$ over the nodes. For convenience, the absorbing node t is excluded from $\mathbf{a}$, which is $n-1$-dimensional. The expected number of visits to edge (i,j) before absorption at t is as follows.

Corollary 1.

$$F_{ij}=[\mathbf{aN}]_i M_{ij}. \tag{5}$$

The expected cost $E[c]$ for a Markovian evader can be found by substituting (5) into (1) (Saerens et al. [24]).

Theorem 2.

$$E[c]=\mathbf{aN}\,\mathrm{diag}[\widehat{\mathbf{M}}\widehat{\mathbf{C}}^T+\mathbf{R}\mathbf{S}^T], \tag{6}$$

where $\mathrm{diag}[\widehat{\mathbf{M}}\widehat{\mathbf{C}}^T+\mathbf{R}\mathbf{S}^T]$ *denotes the column vector of the diagonal elements of matrix* $\widehat{\mathbf{M}}\widehat{\mathbf{C}}^T+\mathbf{R}\mathbf{S}^T$.

In a special case where the edge cost is always 1, i.e., $C_{ij}=1$, $\forall\,(i,j)\in A$, $E[c]$ in (6) reduces to the well-known expression for expected time-to-absorption: $\mathbf{aNe}$.

The objective in the Markovian network interdiction problem is to maximize $E[c]$. In the interdiction model, edge cost depends on the interdiction variable X. In turn, the transition matrix and the fundamental matrix depend on X too. Therefore, this results in the nonlinear optimization problem

$$\max_{X\subset A,\,|X|=b}\mathbf{aN}\,\mathrm{diag}[\widehat{\mathbf{M}}\widehat{\mathbf{C}}^T+\mathbf{R}\mathbf{S}^T]. \tag{7}$$

This optimization problem could be termed the *Single Markovian Evader Network Interdiction* problem. The distribution of starting nodes is assumed to be given and independent of the interdiction strategy X, while the $\mathbf{M}$ matrix is assumed to be determined as soon as the

graph and X are known. In numerical computations the most computationally demanding part resides in finding $\mathbf{aN} = \mathbf{a}(\mathbf{I} - \widehat{\mathbf{M}})^{-1}$, which requires Gaussian elimination in general.

The problem in (7) can be generalized for the case of multiple evaders where each evader represents a threat scenario or an adversarial group. Each evader k then has certain probability $w^{(k)}$ of occurring ($\sum_k w^{(k)} = 1$), as well as a distinctive source distribution $\mathbf{a}^{(k)}$, target node $t^{(k)}$ and transition matrix $\mathbf{M}^{(k)}$. The generalized objective is a weighted sum of Equation (6) over all evaders.

A recent paper considered a similar problem modeled as a Markov Decision Process (MDP) (Dimitrov and Morton [9]). In that MDP model interdicting a directed edge only changes the transition probabilities of the edge's source-node. Hence the MDP model can be transformed to an integer linear program. In our approach, interdicting an edge affects transition probabilities of *all* nodes which have directed paths to the source-node (see details in §4). As a result, our interdiction problem (7) is a nonlinear program.

4. Evader Models

As was noted in the introduction, the evader may often be unable to determine correctly the least-cost paths to the target because of incomplete and inaccurate information about the network topology, interdiction decisions, or costs along alternative paths.

We now develop concrete Markovian models that incorporate uncertainty in the path of the evader. These types of models have analogues in other contexts. For example, a similar model was developed for routing in ad-hoc wireless networks. In that application the objective is to transmit messages through the network while reducing latencies and traffic loads (Barrett et al. [4]).

4.1. The Least-Cost-Guided Evader

We suppose that at each node i the evader will consider several paths from i to t and select the one that *appears* to have the lowest cost. Putting this in the content of a Markovian model, we define p_i be any least-cost path from i to t, with cost denoted by $c(p_i)$. Suppose the evader has a destination t and node j is any node in the neighborhood of i ($j \in G_i$). The transition probability from i to j is

$$M_{ij} = \frac{e^{-\lambda(c(p_i) - C_{ij} - c(p_j))}}{\sum_{j \in G_i} e^{-\lambda(c(p_i) - C_{ij} - c(p_j))}}, \tag{8}$$

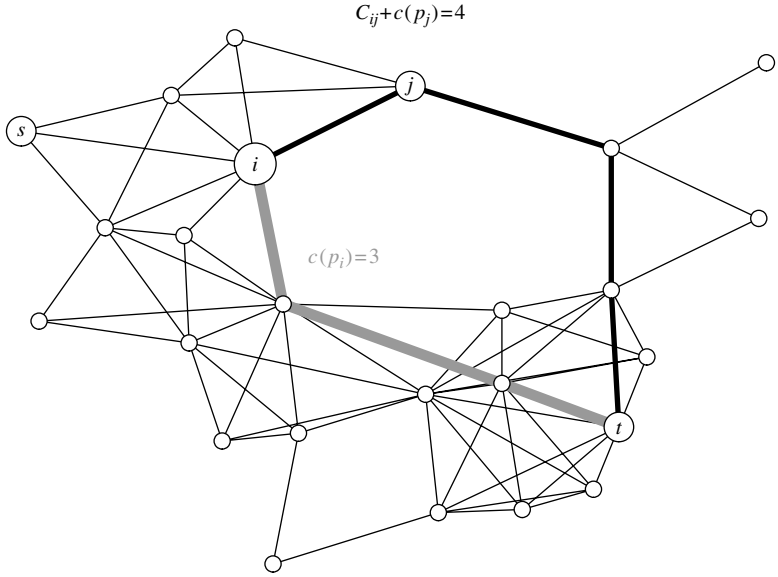
where $\lambda \geq 0$ is a parameter (see Figure 2).

The adherence to least-cost paths is determined by the parameter λ. When $\lambda \to \infty$ the evader moves deterministically along the least-cost path (or paths) and when $\lambda \to 0$ the motion is perfectly random. The least-cost path has the highest probability, but the difference with other paths vanishes as $\lambda \to 0$. Hence, the model can be called the "least-cost-guided evader."

Notice that although M_{ij} values in Equation (8) depend on the cost of least-cost path, when $\lambda < \infty$ this dependence is a smooth function of path costs. Thus the new formulation provides a more desirable description of evader motion because it avoids the sensitivity to path costs seen in the shortest-path evader model. The process of computing the probabilities involves running Dijkstra's algorithm to find the distance to the target node from each node i, which gives $c(p_i)$.

The expression for M_{ij} in Equation (8) could be used to describe a variety of evaders, even evaders who do not reevaluate their paths at every node. For example, an evader could decide in advance which path to follow, but occasionally make mistakes due to poor information. In those cases the Markovian framework and this particular model provide an approximate description of the stochasticity, assigning substantial nonzero probability to paths with cost near the least cost.

FIGURE 2. Computation of the transition probabilities M_{ij}.



Notes. The least-cost path from node i to the target t is the path p_i (thick grey) with cost $c(p_i) = 3$. Through node j the shortest path to t is (thick black) path p_i with cost $C_{ij} + c(p_j) = 4$.

4.2. The Least-Risk-Guided Evader

In some applications the evader may base decisions on the risk of crossing an edge rather than the cost. In those cases, each edge in the network is assigned a value Y_{ij} for the probability of successful evasion, instead of a cost C_{ij}. The evader attempts to find the path to the target t that offers the greatest probability of evasion which is is just the product of those Y_{ij} values along the path.

Let q_{ij} be the probability of successful evasion on a path consisting of the edge (i, j) and then of least-risk paths from j to the target. One choice is to assume that an evader would traverse edge (i, j) with probability *proportional* to q_{ij}, or more generally, proportional to a positive power of q_{ij}:

$$M_{ij} \propto \left(\frac{q_{ij}}{q_{i*}} \right)^{\lambda}, \tag{9}$$

where $\lambda > 0$ is a parameter, $q_{i*} = \max_j q_{ij}$ is the probability of evasion if any least-risk path from i to the target is followed (the constant of proportionality is found from $\sum_j M_{ij} = 1$).

4.3. The Nonretreating Evader

A simple variant of the least-cost-guided model is the nonretreating evader. In this model it is assumed that an evader always moves to nodes that are closer to the target node t than the current node. To represent this model, we add a 0-1 coefficient z_{ij} to Equation (8). Set $z_{ij} = 0$ if node i is at least as close to the target as node j, and $z_{ij} = 1$ otherwise. Namely, $z_{ij} = 0$ if $c(p_i) \leq c(p_j)$, where $c(p_i)$ and $c(p_j)$ are the smallest costs of paths to the target from nodes i and j, respectively, computed by summing the edge weights. The nonretreating evader model requires accurate information about the linear ordering of true node distance values, $c(p_i)$, for all nodes i.

An interesting effect of such nonretreating motion is that the evader will never cross a node or an edge twice. Consequently the set of nodes becomes a partially ordered set and as a result, there exists a relabeling σ of the nodes such that if $c(p_i) > c(p_j)$ then $\sigma(i) > \sigma(j)$. A simple (nonunique) procedure is to label the target node t as 0 ($\sigma(t) = 0$) and then rank the nodes in the order of their distance (cost) along least-cost path to t, breaking ties arbitrarily.

Computationally, this is the same as the order the nodes are reached by a shortest path (Dijkstra's) algorithm starting at t. The transition probability becomes

$$\widehat{M}_{ij} = \begin{cases} M_{ij}, & c(p_i) > c(p_j), \\ 0, & c(p_i) \leq c(p_j). \end{cases}$$

In this case all paths must reach the target after at most $|N|-1$ steps, where $|N|$ is the number nodes in G, and hence $\widehat{\mathbf{M}}$ becomes nilpotent of power $|N|-1$. Moreover, by labeling the nodes up in order of increasing cost, $\widehat{\mathbf{M}}$ can be written as a lower-triangular matrix with zero diagonal. For example, if the evader traverses a 2×3 grid with the target at one corner then one possible σ gives the matrix

$$\widehat{\mathbf{M}} = \begin{pmatrix} 0 & & & & & \\ 1 & 0 & & & & \\ 1 & 0 & 0 & & & \\ 0 & 1 & 0 & 0 & & \\ 0 & 0.5 & 0.5 & 0 & 0 & \\ 0 & 0 & 0 & 0.5 & 0.5 & 0 \end{pmatrix}.$$

The special matrix structure facilitates an order-of magnitude speedup in the computation of Equation (6). For a general $\mathbf{M}$, computing $\mathbf{a}(\mathbf{I}-\mathbf{M})^{-1}$ involves Gaussian elimination at a cost of $2|N|^3/3$ operations. For a nilpotent lower-triangular $\widehat{\mathbf{M}}$ the cost falls to $O(|N|^2)$ since we can use backward-forward substitutions instead of Gaussian elimination. The cost of computing the objective function Equation (6) is also expected to drop to $O(|N|^2)$ despite the need to reorder the matrix $\mathbf{C}$ when the nodes are relabeled.

5. Solving the Markovian Interdiction Problem

The challenge of network interdiction consists of developing both realistic models and tractable algorithms. The Markovian evader model adds realism but does not reduce the computational complexity of finding good interdiction solutions. Indeed it is clear that the Markovian model is computationally hard because in the limit of $\lambda \to \infty$, the model becomes the least-cost interdiction problem which is NP-Hard (Ball et al. [2], Bar-Noy et al. [3]) and also hard to approximate (Boros et al. [6]). Therefore, this section discusses solution heuristics based on network structure.

A common approach to solving many combinatorial optimization problems is based on local, or neighborhood, search algorithms such as simulated annealing (Osman and Kelly [19]). But those general-purpose local search algorithms do not scale well to larger problems or find poor solutions. The solution space may be exponential in the budget so any iterative improvement process of local search can only explore a very small fraction of solutions in a polynomial number of steps. It follows that high-quality solutions can only come from more specialized solvers that exploit the structure of the interdiction problem. We explore algorithms based on ranking functions that rank edges according to global information about graph structure.

5.1. Betweenness Centrality Heuristic

The most successful ranking function we found is derived from the shortest-path betweenness centrality. The shortest-path betweenness centrality of an edge is the fraction of least-cost paths between all pairs of nodes in a network that cross the edge (Freeman [10]). This metric identifies those edges that are critical to connectivity within a network, such as bridge edges that joins two graph components, because they participate in a large number of least-cost paths linking nodes on a network.

We constructed an heuristic based on shortest-path betweenness centrality by considering only paths between the sources **a** and the target t of the evader. Recall that a_s is the probability that the evader would start at node s. Let $\sigma_{st,X}$ be the number of least-cost paths between nodes s and the target node t in the graph with interdiction set X. Similarly, let $\sigma_{st,X}(e)$ be the number of those paths that pass through edge e. Therefore, we define the source-weighted centrality of edge e with respect to t as the sum

$$H_X(e) = \sum_{s:\, t \neq s \in V} a_s \frac{\sigma_{st,X}(e)}{\sigma_{st,X}}. \tag{10}$$

Notice that this quantity needs to be recomputed during execution of an interdiction problem: as the interdiction set X is increased, the costs of the edges change and so are the least-cost paths. An algorithm for calculating a metric of this kind for all $e \in A$ in $O(|A| + |N| \log |N|)$ time is found in Brandes [7]. In the case of multiple evaders, the heuristic is computed for each evader and weighted based on $w^{(k)}$.

5.2. Algorithms

We use the betweenness heuristic $H_X(e)$ to rank the edges e in the network given the interdiction set X. This heuristic leads to a simple algorithm, termed Betweenness (Algorithm 1), that performs a sequential selection of edges.

Algorithm 1 (Betweenness algorithm using global heuristic H for budget b)

$X \leftarrow \varnothing$
while $b > 0$ **do**
  $X \leftarrow X \cup \{\arg\max_{e \in A \setminus X} H_X(e)\}$, resolving ties arbitrarily
  $b \leftarrow b - 1$
Output(X)

The betweenness algorithm is fast since it does not evaluate the objective function (the expected cost to the evader). Rather, the algorithm computes the ranking heuristic and then re-evaluates it after the interdicted edge is chosen. The heuristic is called b times: once for each of the budgeted edges.

For comparison we also use a more computationally-expensive greedy algorithm (Algorithm 2) that constructs the interdiction set X incrementally. At each of the b steps, the greedy algorithm examines every edge $e \in A \setminus X$ and computes the updated objective value from adding e, $E[c \mid X \cup \{e\}]$. This value is found by updating the cost of traversing edge e and using Equation (6). The algorithm then selects an edge with the highest increase ($\Delta_X(e)$).

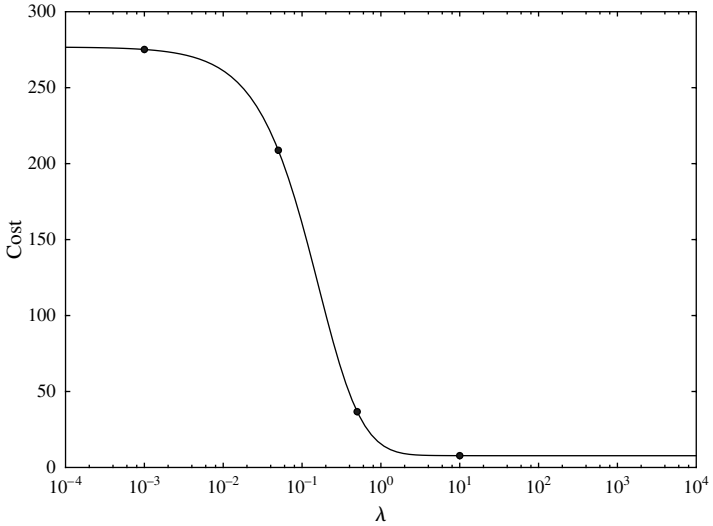
Algorithm 2 (Greedy algorithm for the construction of the interdiction set X with budget b)

$X \leftarrow \varnothing$
while $b > 0$ **do**
  for all $e \in A$ **do**
    $\Delta_X(e) := E[c \mid X \cup \{e\}] - E[c \mid X]$
  $X \leftarrow X \cup \{\arg\max_{e \in A \setminus X} \Delta_X(e)\}$, resolving ties arbitrarily
  $b \leftarrow b - 1$
Output(X)

5.3. Performance Results

We now demonstrate the performance of the Greedy and Betweenness algorithms on a sample network interdiction problem and show the effect of varying the randomness parameter λ.

FIGURE 3. The expected cost of reaching the target from the source as a function of the parameter λ for an example network.



Notes. For large values of λ the model chooses only the shortest path and the expected cost is lowest. As λ decreases the cost increases as the paths become more random. For $\lambda = 0$ the paths are completely random and the cost is at the maximum. The expected cost is calculated by Equation (6) with the evader model M given by Equation (8). The network is a 10×10 directed grid with 10 randomly added shortcut edges and the target and source are chosen randomly. Each of the edges have weights chosen uniformly from $[0.5, 1.5]$. The marked points will be used in performance evaluations, presented in Figure 4.

We used a network which consists of a 10×10 grid of directed edges with 10 added shortcuts between random pairs of nodes for a total of 420 edges. Weights were assigned to each edge by choosing uniformly at random from the interval $[0.5, 1.5]$. We selected 2 distinct targets at random (i.e., 2 evaders) each with 5 source locations.

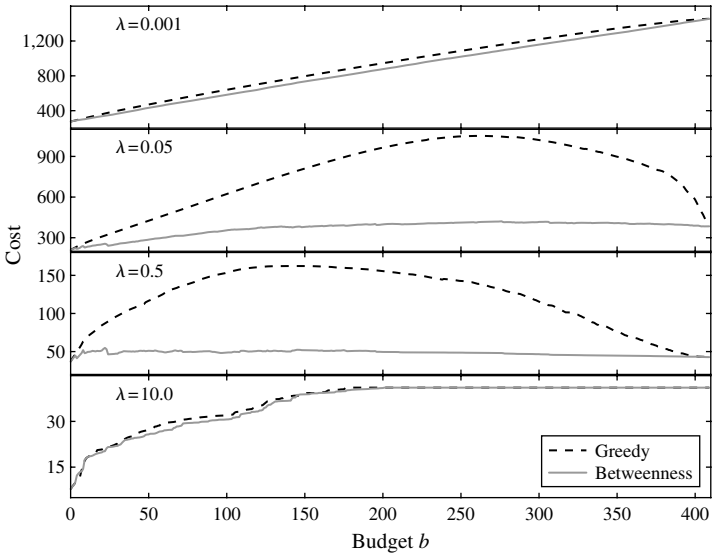
The motion of the evader followed the least-cost-guided model. In this model, the effect of the parameter λ on the expected cost for the evader (before interdiction) is not linear, as shown in Figure 3. At low values of λ the motion is random and the cost is the highest. As λ is increased the evader follows paths that are closer to any optimal path and the cost decreases continuously toward the minimum achievable at large λ. The transition between the cost of random motion and the optimal cost occurs rapidly over a small range of λ where the most diverse behavior is found. This transition in behavior was observed in other random and structured graphs and real-world networks that we examined and is a feature of the nonlinear dependence of the path probabilities from Equation (8).

Figure 4 shows characteristic performance results for both the Greedy and Betweenness algorithms for various λ. The performance is measured in terms of the expected cost given by Equation (6). Interdiction of an edge causes the weight of the edge to increase by a fixed value D_{ij}. We set the added increase to be half the diameter of the network which in this case is $D_{ij} = 4.5$.

For small budgets the Betweenness algorithm and Greedy algorithm produce comparable results as measured by the increase in cost for all λ values. The Betweenness algorithm is considerably cheaper in computational cost. As the budget is increased the Betweenness heuristic performs very well for larger λ. But for smaller λ, as the evader randomness increases, the algorithm performance difference diverges indicating that the Betweenness heuristic is no longer effective. At very low values of λ the evader motion is random and no algorithm is expected to be effective.

A particularly interesting phenomenon is the nonmonotonicity of the expected cost. Namely, for some low λ values the expected cost $E[c]$ sometimes actually decreases after the

FIGURE 4. Comparison of the Greedy (2) and Betweenness (1) algorithms for given budgets on the sample grid network described in Figure 3.



Notes. Four different values of λ are shown corresponding to different levels of randomness in the evader path selection. When the randomness of the evader is low (high λ) the Betweenness algorithm performs very well compared to the higher computational cost Greedy algorithm. As the randomness increases the algorithms' performance diverges after very small budgets—demonstrating that the Betweenness heuristic is no longer effective. At low values of λ the evader motion is random and no algorithm will be effective. The convergence of the algorithms at large budgets occurs because we do not allow an edge to be interdicted more than once and at that budget every edge in the graph is interdicted and the costs are the same.

interdiction set is enlarged. This effect was anticipated by the example in Figure 1 and it occurs because the behavior of the randomizing evader is fundamentally different from the behavior of the max-min evader. If we relax the budget constraint $|X| = b$ to $|X| \leq b$, the objective will be nondecreasing in the Greedy algorithm.

Other realizations of 10×10 grid networks produce similar results and are not shown here. In addition to this example we have explored the performance of the algorithms on other networks including real-world of transportation networks, such as the Washington, DC transportation transit time network and the Rome city road network. The computation cost of the Greedy algorithm becomes prohibitive in these and other urban, national and international transportation systems. Those networks have $10^3 - 10^7$ edges, depending on the spatial resolution. The Greedy algorithm running time scales as $O(|A||N|^3)$ for the least-cost-guided evader model, while the Betweenness algorithm remains feasible even on very large instances because its running time scales as $O(|A| + |N| \log |N|)$.

6. Conclusions and Outlook

Practical instances of network interdiction must invariably address the uncertainty in the network structure and evader behavior. Such behavior can be modeled using the proposed Markov chain approach, which achieves increased realism while remaining analytically penetrable. To summarize, the main contribution of this work are:

- a demonstration of the fundamental advantages of stochastic models over least-cost models,
- a stochastic model of the evader motion based on a Markovian guided random walk, and
- a scalable interdiction algorithm based on a specialized betweenness centrality function.

Future research must address both computational and modeling challenges in stochastic network interdiction. Current algorithms are effective in the case where the evader motion is partially predictable. It is not known whether more specialized heuristics can be more successful in the case of highly-stochastic adversaries. In the current model the randomness comes only from information constraints. In some problems computational constraints on the evader also play a role in determining the motion. Models that account for both kinds of constraints promise further gains in realism and would expand the range of applications where network interdiction could be used.

Acknowledgments

AG would like to thank David Shmoys and Vadas Gintautas for fruitful discussions. Part of this work was funded by DTRA Basic Research under contract IACRO 09-4693I. Released as LA-UR-08-06551.

References

[1] M. P. Atkinson, Z. Cao, and L. M. Wein. Optimal stopping analysis of a radiation detection system to protect cities from a nuclear terrorist. *Risk Analysis* 28(2):353–371, 2008.

[2] M. O. Ball, B. L. Golden, and R. V. Vohra. Finding the most vital arcs in a network. *Operations Research Letters* 8(2):73–76, 1989.

[3] A. Bar-Noy, S. Khuller, and B. Schieber. The complexity of finding most vital arcs and nodes. Technical report, University of Maryland, College Park, MD, 1995.

[4] C. L. Barrett, S. J. Eidenbenz, L. Kroc, M. Marathe, and J. P. Smith. Parametric probabilistic routing in sensor networks. *Mobile Networks and Applications* 10:529–544, 2005.

[5] H. Bayrak and M. D. Bailey. Shortest path network interdiction with asymmetric information. *Networks* 52:133–140, 2008.

[6] E. Boros, K. Borys, V. Gurevich, and G. Rudolf. Inapproximability bounds for shortest-path network interdiction problems. Technical report, Rutgers University, Piscataway, NJ, 2006.

[7] U. Brandes. A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology* 25(2):163–177, 2001.

[8] H. W. Corley and D. Y. Sha. Most vital links and nodes in weighted networks. *Operations Research Letters* 1(4):157–160, 1982.

[9] N. B. Dimitrov and D. P. Morton. Combinatorial design of a stochastic Markov decision process. *Operations Research and Cyber-Infrastructure*, Vol. 47. Operations Research/Computer Science Interfaces, Springer, New York, 167–193, 2009.

[10] L. C. Freeman. A set of measures of centrality based on betweenness. *Sociometry* 40(1):35–41, 1977.

[11] P. M. Ghare, D. C. Montgomery, and W. C. Turner. Optimal interdiction policy for a flow network. *Naval Research Logistics Quarterly* 18(1):37–45, 1971.

[12] C. M. Grinstead and J. L. Snell. *Introduction to Probability*, 2nd revised ed. American Mathematical Society, Providence, RI, 1997.

[13] A. Gutfraind, A. Hagberg, and F. Pan. Optimal interdiction of unreactive Markovian evaders. J. Hooker and W.-J. van Hoeve, eds. *Proceeding of the 6th International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, Vol. 5547. Lecture Notes in Computer Science, Springer, Berlin/Heidelberg, Germany, 102–116, May 2009.

[14] E. Israeli and R. K. Wood. Shortest-path network interdiction. *Networks* 40(2):97–111, 2002.

[15] U. Janjarassuk and J. Linderoth. Reformulation and sampling to solve a stochastic network interdiction problem. *Networks* 52(3):120–132, 2008.

[16] A. W. McMasters and T. M. Mustin. Optimal interdiction of a supply network. *Naval Research Logistics Quarterly* 17(3):261–268, 1970.

[17] N. Memon and H. Larsen. Practical algorithms for destabilizing terrorist networks. *Intelligence and Security Informatics*, Vol. 3975. Lecture Notes in Computer Science, Springer-Verlag, Berlin/Heidelberg, 389–400, 2006.

[18] D. P. Morton, F. Pan, and K. J. Saeger. Models for nuclear smuggling interdiction. *IIE Transactions* 39(1):3–14, 2007.

[19] I. H. Osman and J. P. Kelly. *Meta-Heuristics: Theory and Applications*. Kluwer Academic Publishers, Norwell, MA, 1996.

[20] F. Pan and D. P. Morton. Minimizing a stochastic maximum-reliability path. *Networks* 52(3):111–119, 2008.

[21] F. Pan, W. S. Charlton, and D. P. Morton. Interdicting smuggled nuclear material. D. L. Woodruff, ed. *Network Interdiction and Stochastic Integer Programming*. Kluwer Academic Publishers, Boston, 1–19, 2003.

[22] B. Pourbohloul, L. A. Meyers, D. M. Skowronski, M. Krajden, D. M. Patrick, and R. C. Brunham. Modeling control strategies of respiratory pathogens. *Emerging Infectious Diseases* 11(8):1246–56, 2005.

[23] D. Reich and L. Lopes. The most likely path problem on series-parallel networks. *Networks* 2011. To appear.

[24] M. Saerens, Y. Achbany, F. Fouss, and L. Yen. Randomized shortest-path problems: Two related models. *Neural Computation* 21(8):2363–2404, 2009.