LA-UR- 09-00377

Approved for public release; distribution is unlimited.

Title: Network Interdiction with Budget Constraints

Author(s): Nandakishore Santhi, CCS-3, Z# 221302
Feng Pan, D Division, Z# 180428

Intended for: Submission/presentation at the following conference: "2009 IEEE International Symposium on Information Theory,Seoul, South Korea"

Los Alamos
NATIONAL LABORATORY
EST. 1943

Network Interdiction with Budget Constraints

Nandakishore Santhi
CCS and T Divisions
Los Alamos National Laboratory
Los Alamos, NM 87545, USA
Email: nsanthi@lanl.gov

Feng Pan
D Division
Los Alamos National Laboratory
Los Alamos, NM 87545, USA
Email: fpan@lanl.gov

***Abstract*—Several scenarios exist in the modern interconnected world which call for efficient network interdiction algorithms. Applications are varied, including computer network security, prevention of spreading of Internet worms, policing international smuggling networks, controlling spread of diseases and optimizing the operation of large public energy grids. In this paper we consider some natural network optimization questions related to the budget constrained interdiction problem over general graphs. Many of these questions turn out to be computationally hard to tackle. We present a particularly interesting practical form of the interdiction question which we show to be computationally tractable. A polynomial time algorithm is then presented for this problem.**

I. Introduction

In today's inter-connected world, it is often necessary to maintain open communication and transportation networks. However in the interest of fair use, it is also important to keep these networks safe and prevent abuse, and to achieve this in the most non-intrusive manner possible. This has to be done using minimal additional infrastructure in a robust as well as distributed manner, and in addition has to meet budget constraints for the the cost of installation and operation.

Examples of scenarios which require such *interdiction*, include policing drug and nuclear smuggling networks, computer network security applications where firewalls need to be setup to control the spread of Internet worms as well as future smart energy grids where dynamic load balancing will be crucial. Applications also include quarantine planning for controlling the spread of diseases.

A formal model for this practical problem is a network interdiction model, where interdiction is performed along the edges or on the nodes of a graph which represents the communication or transportation network in sufficient detail. In this paper, without loss of generality we will be considering an edge interdiction model on a directed network graph. Throughout the paper, whenever a concrete example is called for, we will use a representative transportation network used by smugglers. It should be noted that this serves only as an example and our approach is in fact quite general.

On various edges e on the network graph, let us model the probability of a smuggler evading detection from the surveillance equipment installed on that edge by a parameter called the *edge evasion probability*, π_e. The *detection probability* δ_e is naturally related to the evasion probability as $\delta_e = 1 - \pi_e$. In most natural cases, the evasion probabilities on various edges can be modeled to be statistically independent, which means that on any path p on the network, the effective evasion probability π_p is given by the product, $\pi_p = \prod_{e \in p} \pi_e$. It should be noted that statistical independence is perhaps not always a good assumption to make, for example when various enforcement devices/agencies are in constant communication, it invariably induces some inter-dependencies in their decision making.

The interdiction problem is not new. In fact, several researchers have in the past considered interdiction in various forms [2], [3], [7], [8]. However many of these formulations are known to be computationally intractable for even modestly sized networks [6]. Some of the suggested solution methods involve some form of integer linear programming which is usually computationally costly. Cutting plane methods and sub-optimal linear programming relaxations have also been proposed in the literature.

A common objective in interdiction problems is to determine an optimal allocation of budgets for installation of interdiction apparatus on individual edges such that the effective evasion probability is minimized while simultaneously satisfying some total budget constraints. In the next section, we give a formal definition of the budget constrained network interdiction problem.

III. A POLYNOMIAL TIME ALGORITHM FOR BC-SE-INT

Algorithm 1 Budget Constrained Single Edge Interdiction Algorithm (BC-SE-INT-ALGO)

INPUT:

A network graph $\mathcal{G}(\mathcal{V},\mathcal{E})$ to be interdicted along with the local functions f_e for all $e \in \mathcal{E}$, a total budget B and a tolerance value $\epsilon > 0$.

STEPS:

1. Set $\pi^1 \leftarrow 1$ and $\pi^0 \leftarrow 0$.
   Augment the original network graph to obtain a new graph $\mathcal{G}'(\mathcal{V}',\mathcal{E}')$ in the following way:
   Create a new node s and connect it to all nodes $s_i \in \mathcal{S}$ with new directed edges $e_{(s,s_i)}$. Similarly, create a new node d and connect it to all nodes $d_i \in \mathcal{D}$ with new directed edges $e_{(d_i,d)}$. All newly created edges e are marked non-interdictable, that is $f_e^{-1}(x) \mapsto \infty$ for $x \in [0,1)$.
2. while $(\pi^1 - \pi^0 > \epsilon)$ do {
3. Set $\pi' \leftarrow \frac{\pi^1+\pi^0}{2}$.
4. for all $e \in \mathcal{E}'$, compute $b_e' = f_e^{-1}(\pi')$.
5. Solve the linear program:
   Minimize, $\sum_{e\in\mathcal{E}} b_e' \cdot x_e$ subject to,

$$x_e \geq (y_i - y_j); \qquad x_e \geq (y_j - y_i)$$
$$y_s = 1, \quad y_d = 0; \qquad 0 \leq y_i, x_e \leq 1$$

   Let B' be the minimum attained.
   Let $\mathcal{E}' \supseteq \mathcal{C} \stackrel{\text{def}}{=} \{e : x_e = 1\}$.
6. if $(B' > B)$ set $\pi^0 \leftarrow \pi'$.
7. else if $(B' < B)$ set $\pi^1 \leftarrow \pi'$.
8. }
9. Set $\pi \leftarrow \pi^1$.
   For all $e \in \mathcal{C}$, set $b_e \leftarrow b_e'$ and for all $e \in \mathcal{E}\backslash\mathcal{C}$, set $b_e \leftarrow 0$.

OUTPUT:

The solution π and an associated set of edge budgets $\{b_e : e \in \mathcal{E}\}$

In order to derive an algorithm for BC-SE-INT, we will assume that the local functions are efficiently invertible - that is, $f_e^{-1}(\cdot)$ can be computed in polynomial-time. There is no loss in generality due to this assumption, since in virtually all practical scenarios this is true - moreover in the event of there being no analytical form for the inverse function, a table look-up based approach can be easily implemented. A pseudo-code for the proposed algorithm BC-SE-INT-ALGO is listed as Algorithm 1.

IV. CORRECTNESS AND COMPLEXITY OF BC-SE-INT-ALGO

To see that the algorithm BC-SE-INT-ALGO produces the correct result to an accuracy of better that an additive factor of ϵ, we can note the following. Since $\mathcal{S} \cap \mathcal{D} = \emptyset$, any $(s,d)-$path should contain at least one interdictable edge. Moreover, since the local functions are monotonic non-decreasing, an increased local budget will not increase the edge's evasion probability.

Now the linear program in step 5 is well known to have an integral polyhedra, so that at the solution, $x_e \in \{0,1\}$. This can be easily seen considering the following probabilistic argument: If $\bar{y}_i$ is a fractional point in the solution, let us use the following randomized procedure - generate a uniform random variable $\boldsymbol{u}$, then set $y_i \leftarrow 0$ if $\bar{y}_i < \boldsymbol{u}$ and set $y_i \leftarrow 1$ otherwise. Now,

$$\begin{aligned} B' &\leq \mathbb{E}\left(\sum_{e\in\mathcal{E}} b_e' \cdot x_e\right) \\ &= \sum_{e=(i,j)} b_e' \cdot \Pr\left(\boldsymbol{u} \in [\min\{\bar{y}_i,\bar{y}_j\},\ \max\{\bar{y}_i,\bar{y}_j\}]\right) \\ &= \sum_{e=(i,j)} b_e' \cdot |\bar{y}_i - \bar{y}_j| = \sum_e b_e' \cdot \bar{x}_e = B' \end{aligned}$$

Therefore step 5 finds a minimum budget interdiction cut on the original network graph such that on any $(s,d)-$path, at least one edge has evasion probability less that π'. Moreover the interdiction cut cannot involve any of the fictitious non-interdictable edges introduced in step 1. Furthermore, the monotonous property of the local functions f_e implies that an optimal interdiction cut resulting in a higher budget B', cannot have a higher evasion probability π'. Therefore each iteration of the loop from step 2 to step 8 reduces the search region for π by half at either of the steps 6 or 7, while satisfying the budget constraint and will therefore terminate with the correct solution in $\mathcal{O}(\log 1/\epsilon)$ iterations.

To estimate the complexity of BC-SE-INT-ALGO, for a precision as required by the constant ϵ, the loop from step 2 to step 8 is executed $\mathcal{O}(\log 1/\epsilon)$ times, which is again a constant. We can further improve the algorithm by substituting for the linear program in step 5 any well known algorithm for max-flow, since max-flow and min-cut are related by linear programming duality [1]. Each iteration of this loop requires a polynomial amount of time, which depends on the $(s-d)-$min-cut algorithm employed. Using an efficient max-flow algorithm as in [5], which has a complexity of $\mathcal{O}(|\mathcal{V}| \cdot |\mathcal{E}| + |\mathcal{V}|^2 \log |\mathcal{V}|)$, each iteration takes

II. Budget Constrained Single Edge Interdiction

In this section we consider a few most commonly encountered versions of the network interdiction problem. We then derive an algorithm which solves a practically important form of the interdiction problem in time polynomial in the size of the problem description.

Definition 1 (BC-INT, BC-AV-INT, BC-SE-INT)
Instance: *A directed network graph $\mathcal{G}(\mathcal{V},\mathcal{E})$; a set of efficiently computable monotonic non-increasing local budget-evasion-probability functions $f_e : \mathbb{R}^+ \mapsto [0,1]$ associated with each directed edge $e \in \mathcal{E}$; two non-empty subsets of $\mathcal{V}$, the source nodes $\mathcal{S}$ and the destination nodes $\mathcal{D}$, such that $\mathcal{S} \cap \mathcal{D} = \emptyset$; and a total interdiction budget B.*
Question 1 (BC-INT): *Find a budget assignment to each edge, b_e which satisfies the total budget constraint $\sum_{e\in\mathcal{E}} b_e \leq B$, and minimizes,*

$$\pi_{MAX} \stackrel{\text{def}}{=} \max_{p^j_{(s,d)} \in \mathcal{P}(\mathcal{S},\mathcal{D})} \prod_{e_j \in p^j_{(s,d)}} f_{e_j}(b_{e_j})$$

Question 2 (BC-AV-INT): *Find a budget assignment to each edge, b_e which satisfies the total budget constraint $\sum_{e\in\mathcal{E}} b_e \leq B$, and minimizes,*

$$\pi_{AV} \stackrel{\text{def}}{=} \sum_{p^j_{(s,d)} \in \mathcal{P}(\mathcal{S},\mathcal{D})} w_{p^j_{(s,d)}} \cdot \prod_{e_j \in p^j_{(s,d)}} f_{e_j}(b_{e_j})$$

Question 3 (BC-SE-INT): *Find a budget assignment to each edge, b_e which satisfies the total budget constraint $\sum_{e\in\mathcal{E}} b_e \leq B$, and minimizes,*

$$\pi \stackrel{\text{def}}{=} \max_{p^j_{(s,d)} \in \mathcal{P}(\mathcal{S},\mathcal{D})} \min_{e_j \in p^j_{(s,d)}} f_{e_j}(b_{e_j})$$

where $\mathcal{P}(\mathcal{S},\mathcal{D})$ is the set of all directed paths $p^j_{(s,d)}$ from some node in $\mathcal{S}$ to some node in $\mathcal{D}$, $w_{p^j_{(s,d)}}$ are positive weights associated with these paths such that $\sum_{p^j_{(s,d)}} w_{p^j_{(s,d)}} = 1$, and e_j represents a directed edge in the directed path $p^j_{(s,d)}$.

In the above definition, the local budget-evasion-probability functions $f_e(\cdot)$ can be roughly interpreted as follows: given a local arc budget of b_e for arc e, we can achieve an evasion probability of $f_e(b_e)$ at that arc. Very often in practice, the local functions f_e could be made to subsume other more complex characteristics on the network too.

For example, if in a network with a single source and destination, there are already in place other interdiction apparatus, which ensures evasion probabilities less than 1 on certain edges. Then, we may wish to calculate the residual evasion probability before installing any new apparatus by first running a Dijkstra type shortest path algorithm. Let each edge $e = (i,j)$ have a prior evasion probability of α_e. Also let us assume for example that by installing N_e apparatus of unit cost, the post-installation edge evasion probability can be reduced to $\alpha_e \cdot \beta_e^{N_e}$. Then we may wish to set as a first order approximation, $f_e(N_e) = \alpha_e \cdot \beta_e^{N_e} \cdot \prod_{e'_s \in p(s,i)} \alpha_{e'_s} \cdot \prod_{e'_d \in p(j,d)} \alpha_{e'_d}$. Here, $p(s,i)$ is the shortest path from source node s to node i when the edges e' are labeled with non-negative edge weights of $(-\log \alpha_{e'})$. Similarly $p(j,d)$ is the shortest path from node j to the destination node d.

All the three forms of interdiction problems can be seen to be practically relevant in various contexts. However, even for the simplest local functions f_e, the problems posed in questions 1 and 2 above are known to be NP-complete even to approximate within a constant factor, by a polynomial time reduction from the relatively well known VERTEX-COVER and CLIQUE problems [4]. For a simple proof of this reduction, see [6].

In this paper therefore, we will focus solely on question 3. Since the local functions f_e can be heavily non-linear, it is not immediately clear that the problem in question 3 more often than not admits a polynomial time solution. We present one such solution in the next section.

One may justify posing question 3 in favor of the other two versions in many situations. In problems where non-zero evasion probabilities have to be avoided at all costs (for example in the case of nuclear smuggling), interdiction apparatus at edge e can be reasonably modeled as requiring a cost of b_e to ensure $\pi_e = 0$. In this case, solving question 3 is equivalent to solving question 1, whereas question 2 is perhaps not practically relevant (since it is the worst case evasion probability that matters, not the average case). In many other instances, it is usually the case that the evasion probability that can be achieved is so small that a solution for question 3 is practically very close to that of question 1. Moreover, the availability of an efficient algorithm is clearly a factor to be considered. Typical solutions to interdiction problems would otherwise rely on the solution of cumbersome integer-linear-programs, which are often computationally intractable even for medium scale networks.

$\mathcal{O}(r|\mathcal{E}| + \mathcal{O}(|\mathcal{V}| \cdot |\mathcal{E}| + |\mathcal{V}|^2 \log |\mathcal{V}|))$ time, where r denotes the time required for computing the inverse function $f_e^{-1}(\cdot)$ to the required precision.

V. CONCLUSION

We considered the important practical problem of budget constrained interdiction. We posed an optimization problem which is very relevant for several practical scenarios, with the additional property of being computationally tractable. This is unlike other common variations of interdiction related problem which are typically computationally hard. We derived an algorithm which finds an optimal solution (up to any given small constant) to the problem we posed. Simulation results using an implementation of our algorithm were very promising - large networks which were typically not amenable to brute force integer programming approaches have yielded meaningful solutions while using up only reasonable computation times.

Problems of future interest include scenarios where simultaneous optimization is required over several cost functions and under multiple budget constraints. Also of interest are networks where multiple commodities are transacted. Further improvements in running time are of definite interest, as are faster approximation algorithms for use with extremely large networks. Algorithms which adapt to dynamic changes in evasion probabilities as well as models which consider statistical dependence and other stochastic variables are also of interest.

REFERENCES

[1] P. ELIAS AND A. FEINSTEIN AND C. E. SHANNON, "A Note on maximum flow through a network," *IEEE Trans. on Information Theory*, v. IT-2, pp. 117–119, 1956.

[2] E. P. DURBIN, "An Interdiction Model of Highway Transportation, RM-4945-PR" *Rand Corporation*, Santa Monica, CA, May, 1966.

[3] A. W. MCMASTERS AND T. M. MUSTIN, "Optimal Interdiction of a Supply Network," *Naval Research Logistics Quarterly*, v. 17, n 3 (Sep.), pp. 261–268, 1970.

[4] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability*, W. H. Freeman and Co., San Francisco, 1979.

[5] R. K. AHUJA AND J. B. ORLIN, "A Fast and Simple Algorithm for the Maximum Flow Problem," *Operations Research*, v. 37, n 5 (Sep.-Oct.), pp. 748–759, 1989.

[6] R. K. WOOD, "Deterministic Network Interdiction," *Mathematical and Computer Modeling*, v. 17, n 2, pp. 1–18, 1993.

[7] F. PAN AND W. CHARLTON AND D. P. MORTON, "Interdicting smuggled nuclear material," *Network Interdiction and Stochastic Integer Programming*, D. L. Woodruff (ed.), Kluwer, Boston, MA, 2003.

[8] D. P. MORTON AND F. PAN AND K. J. SAEGER, "Models for Nuclear Smuggling Interdiction," *IIE Transactions*, Jan. 2006.