BILEVEL NETWORK INTERDICTION MODELS: FORMULATIONS AND SOLUTIONS

R. Kevin Wood
Department of Operations Research,
Naval Postgraduate School,
Monterey, California

INTRODUCTION

A dictionary definition of *interdict*, in the military sense, is

> to destroy, cut or damage by ground or aerial firepower (enemy lines of reinforcement, supply, or communication) in order to stop or hamper enemy movement and to destroy or limit enemy effectiveness [1].

This definition is unnecessarily restrictive—for instance, it should also include ship-based firepower—but the essence is reasonable: interdiction connotes preemptive attacks that limit an enemy's subsequent ability to wage war, or carry out other nefarious activities. The mathematical study of interdiction has focused primarily on *network interdiction*, in which an enemy's activities are modeled using the constructs of network optimization (e.g., maximum flows, multicommodity flows, and shortest paths), and in which attacks target the network's components to disrupt the network's functionality. Depending on the type of network, targeted components can include bridges, road segments or interchanges, communications links or switches, and so on. This article focuses on the *bilevel network interdiction problem* (BNI), but the reader should note that much of the presentation extends to interdiction of more general systems.

Examples of what we now call network interdiction date from antiquity. Herodotus [[2] 9.49–50] describes how the Persian cavalry (in 479 BC) cut Greek supply lines and routes to water sources in a battle near the Greek city of Plataea; Livy [3, 22.8] reports that (in 218 BC) the Roman Senate ordered bridges near Rome to be destroyed to slow the advance of Hannibal and his troops; Polybius [4, 9.7] gives a different chronology for the latter incident, but the interpretation as "network interdiction" remains. Two millennia later, the American Civil War is replete with examples of both Confederate and Union forces attacking roads, bridges, rail lines, and telegraph lines to hamper the enemy's resupply, movement, and communications [5, Chapter 4]. In World War II, German submarines interdicted, that is, sank, hundreds of Allied petroleum tankers that were traveling the sea lanes of the Atlantic Ocean and elsewhere [6, Appendix 17].

Allied bombing attacks during World War II on German-controlled oil refineries and synthetic-fuel plants exemplify a more general type of *system interdiction* or "economic warfare." The German military was crippled by the lack of fuel and lubricants caused by the "Oil Plan," as the attack strategy was called. Interestingly, an acrimonious debate was waged between the proponents of the Oil Plan (system interdiction) and proponents of the "Rail Plan" (network interdiction). The Rail Plan sought to destroy rail lines and other transportation assets in Europe to restrict the movement of the German troops and equipment that would counter the Allied D-Day offensive. Ultimately, parts of both plans were implemented [7, pp. 75–78, 174–175].

Network interdiction is an important part of modern warfare where attacks on key civilian and military infrastructure can help reduce an enemy's fighting effectiveness, while incurring only limited casualties to "friendly forces" [8,9]. When planning for such attacks, the "interdictor" is typically faced with this question: Given limited attack resources and possibly other restrictions (e.g., political considerations), which network components should be attacked to reduce the enemy's war-fighting capabilities most effectively? BNI addresses this question.

Wiley Encyclopedia of Operations Research and Management Science, edited by James J. Cochran

To provide background on the modern, mathematical study of network interdiction, we first make the following definitions and assumptions [10]:

1. The interdictor, called *attacker* hereafter, acts first by using limited interdiction resources to attack components of his enemy's network.
2. The enemy, or *defender*, observes the damage caused by the attack(s) and then operates the damaged network so as to maximize his own, well-defined objective-function value.
3. The attacker understands the defender's capabilities and goals, and chooses attacks that minimize the defender's maximum achievable objective-function value.

These standard assumptions lead to the formulation of BNI and more general system-interdiction models as a type of Stackelberg game: a two-person, zero-sum, sequential-play game with two stages [11,12].

Wollmer [13] studies the problem of finding the single "most vital link" in a capacitated flow network, in his case, the arc whose deletion minimizes the maximum *s-t* flow in the network. (The reader who is unfamiliar with the terminology of network flows, e.g., "*s-t* flows," may wish to consult a standard text such as Ahuja *et al*. [14].) Here, the attacker has enough interdiction resource to attack and destroy exactly one arc, and the defender's objective is to maximize *s-t* flow. Wollmer's work may represent the earliest mathematical investigation of an instance of BNI, although as early as 1955 researchers were investigating a simpler "single-level" network interdiction problem that seeks to eliminate all *s-t* flow efficiently [15].

Danskin [16] presents some of the fundamental theory of "max–min models," which may be viewed as a generalization of BNI to system interdiction. (His "min" and "max" are reversed compared to our convention). Confusingly, "max–min" and similar terms are also used in the context of the more common two-person, zero-sum, simultaneous-play games [17, pp. 143–165]. For example, von Neumann [18] proves the famous "minimax theorem" for such games.

Mathematical studies of BNI began in earnest during the Vietnam War, with models applied to disrupt the flow of enemy troops and materiel [19,20]. Fulkerson and Harding [21] and Golden [22] investigate the problem of maximizing the length of the shortest path in a network—to slow enemy reinforcements, say—using models in which the length of each network arc can be increased linearly, within limits, based on the amount of interdiction resource applied to it. (This is a "max–min" variant of BNI.) These models are solved as parametric linear programs (LPs). The *k-most-vital-arcs problem* [23,24] is similar, but at most k arcs may be interdicted and interdiction decisions are discrete. In particular, an arc is attacked and destroyed and its length becomes infinite, or it is left untouched and keeps its nominal length. Ball *et al*. [25] show that problem to be NP-complete. Israeli and Wood [26] extend the k-most-vital-arcs problem to general resource constraints: their study of the the *shortest-path network interdiction problem* (or "maximizing the shortest path"), makes important theoretical and computational contributions to the solution of BNI, in general.

Ratliff *et al*. [27] extend Wollmer's [13] model to the problem of finding a set of n arcs in a capacitated network whose deletion minimizes the maximum *s-t* flow. While investigating problems of drug interdiction, Wood [28] generalizes that model to allow general interdiction resource constraints. (Phillips [29] considers a similar model, but allows some continuous interdiction effort; see also Steinrauf [30].) Wood shows that this *maximum-flow interdiction problem* is NP-complete, even when attacks are constrained only in cardinality. Thus, even the simpler model of Ratliff *et al*. appears to be difficult.

More general system-interdiction issues arise in the work of Grötschel *et al*. [31] and Medhi [32] who seek to evaluate the vulnerability of information networks to interdiction. Chern and Lin [33] study the interdiction of a system represented as a minimum-cost network-flow model.

Similar to the BNI studied by Wood [28], the network interdiction model of Washburn and Wood [34] aims at disrupting drug smuggling, and its efficient solution involves maximum flows. But this is a two-person, zero-sum, simultaneous-play (Cournot) game, and its purpose is quite different from BNI. Specifically, an interdictor controls one or more "inspectors" who must be placed strategically on the arcs of a transportation network to maximize the probability of detecting a drug smuggler moving surreptitiously through that network. (If the smuggler traverses arc k when an inspector is present, the smuggler is detected with known probability p_k; otherwise he goes undetected.) In a simultaneous-play model, neither player can observe the other's actions before acting himself and, consequently, solutions define probabilistic ("mixed") strategies for both players. In this case, the interdictor's strategy defines a probability distribution over the inspectors' locations, and the smuggler's strategy defines a probability distribution over paths through the network. In contrast, a solution to BNI prescribes deterministic ("pure") strategies for both players.

Deterministic strategies do not imply that BNI cannot incorporate uncertainty and probability, however. Cormican *et al*. [35] develop stochastic-programming versions of the maximum-flow interdiction model to handle uncertain interdiction successes and uncertain arc capacities. Whiteman [36], studying interdiction problems faced by the US Strategic Command, addresses uncertainty through Monte Carlo simulations of maximum-flow interdiction models. Pan *et al*. [37] maximize the expected probability of detecting a smuggler trying to transport stolen nuclear materials out of a country: nominal probabilities of detection can be improved by installing a limited number of radiation detectors at border crossings. Maximizing probability of detection is related, through a logarithmic transformation in the objective function, to shortest-path interdiction, and in that sense the model is deterministic. The model is stochastic, however, in that a probability distribution describes the smuggler's origin in the network [38].

Brown *et al*. [39,10] develop a taxonomy for bilevel system-interdiction and system-defense models, as well as for trilevel system-defense models. The bilevel and trilevel models are two-stage and three-stage Stackelberg games, respectively. BNI is an instance of a two-stage "attacker–defender model"; the trilevel defense models are "defender–attacker–defender models." In the latter case, a defender wishes to employ his limited defensive resources as efficiently as possible to "interdict the interdictor," with effectiveness being evaluated by solving a bilevel interdiction model. Bilevel system-defense models can be constructed, also: these are "defender–attacker models"; they apply when the value of attacking a system component is a fixed or easily computed value; they can be solved using the techniques described in this article; but will not be discussed further. We note that Brown *et al*. [10,40] also discuss solution techniques for all these model types and describe a number of applications to infrastructure protection. Indeed, vulnerability analysis for infrastructure is an important new application area for BNI.

Most recently, bilevel interdiction and defense models have been developed for a number of interesting applications: theater ballistic missile defense [41], planning attacks on multicommodity flow networks [42], planning attacks on communications networks [43], delaying the nuclear-weapons project of a "rogue state" [44], and attacking and defending electric-power grids [45]. The theory of "global Benders decomposition" developed in the last paper promises to be a useful computational technique for BNI (and other optimization problems), and is discussed later in this article.

The goal of the rest of this article is to describe basic theoretical models and solution techniques for BNI. A lack of space prohibits further detailed discussion of applications. More information on advanced computational techniques can be found in Magnanti and Wong [46], Israeli and Wood [26], Salmerón *et al*. [45], and Smith *et al*. [47].

A BASIC, BILEVEL, INTERDICTION MODEL

In abstract form, BNI may be stated as the following *attacker–defender model*:

$$[\text{AD0}] \quad \min_{\mathbf{x}\in X} z_0(\mathbf{x}), \quad \text{where}$$

$$z_0(\mathbf{x}) \equiv \max_{\mathbf{y}\in Y(\mathbf{x})} f(\mathbf{x},\mathbf{y}), \tag{1}$$

and where (i) $\mathbf{x} \in X$ denotes a binary vector of attack decisions that is limited by resources and perhaps logical restrictions (e.g., targets k and k' both cannot be attacked); (ii) $\mathbf{y} \in Y(\mathbf{x})$ denotes the activities that the defender will carry on after the attack, typically restricted by effects of the attack $\mathbf{x}$; and (iii) the objective function $f(\mathbf{x},\mathbf{y})$ measures the functionality of the defender's network after the attack. Thus, the attacker seeks to minimize the functionality of the network which the defender is assumed to maximize. Of course, by switching the min and the max, we can model an attacker who seeks to maximize the cost of the defender's operations.

[AD0] is a special case of a Stackelberg game in which a leader (attacker) takes some action, and a follower (defender) observes that action and its effects, and then responds optimally given that information (12). [AD0] is a two-stage game and is finished after the follower responds; more general Stackelberg games may have many stages and/or players.

For the sake of concreteness and simplicity, further development of [AD0] assumes that

1. The defender's activities take place on network arcs, which are indexed by k, and there is a one-to-one correspondence with the attacker's potential targets.
2. The defender nominally optimizes network operation by solving the following LP which is feasible for any $\mathbf{u} \geq \mathbf{0}$

$$\max_{\mathbf{y}} \quad \mathbf{c}^{\mathrm{T}}\mathbf{y} \tag{2}$$

$$\text{s.t.} \quad A\mathbf{y} \leq \mathbf{b} \tag{3}$$

$$0 \leq \mathbf{y} \leq \mathbf{u}, \tag{4}$$

3. Restrictions on the attacker assume $\mathbf{x} = \mathbf{0}$ is feasible, and are represented by

$$X = \left\{\mathbf{x} \in \{0,1\}^n \mid H\mathbf{x} \leq \mathbf{h}\right\}, \text{ and} \tag{5}$$

4. $x_k = 1$ implies that activity k is attacked, its level forced to 0, that is, $y_k = 0$.

Then, defining $U = \text{diag}(\mathbf{u})$, [AD0] takes on the following specific form

$$[\text{ADLP1}] \quad z_1^* = \min_{\mathbf{x}\in X} z_1(\mathbf{x}), \text{ where} \tag{6}$$

$$z_1(\mathbf{x}) \equiv \max_{\mathbf{y}} \quad \mathbf{c}^{\mathrm{T}}\mathbf{y} \tag{7}$$

$$\text{s.t.} \quad A\mathbf{y} \leq \mathbf{b} \tag{8}$$

$$0 \leq \mathbf{y} \leq U(\mathbf{1} - \mathbf{x}). \tag{9}$$

The inner LP in [ADLP1] might represent a simple maximum-flow problem [27,28], the optimal deployment of the defenders's armed forces [48], or the production and distribution of oil or natural gas in a belligerent country [49]. [ADLP1] extends easily to attacks on nodes, attacks on groups of arcs and/or nodes, attacks that reduce capacity only partially, and so on, but such extensions are straightforward and not considered here.

A Stackelberg game with mixed-integer variables and having just two levels of decision making is called a *bilevel mixed-integer program* (BLMIP) [50]. However, the leader's and follower's objective functions in a BLMIP are not usually diametrically opposed as they are in [AD0]; for example, see Bard and Moore [51], Wen and Yang [52], and Hansen *et al*. [53]. In fact, most algorithms developed for BLMIPs assume a strong positive correlation between the leader's and follower's objective functions. Thus, the theory and algorithms for BLMIPs do not seem well-suited for handling [ADLP1], while the special-purpose methods described in this article have had demonstrated successes.

Standard LP theory tells us that $z_1(\mathbf{x})$ is a concave function in (continuous) $\mathbf{x}$. Thus, [ADLP1] is a difficult, nonconvex minimization problem. The problem can be "convexified," however, by moving $\mathbf{x}$ into the objective of the inner maximization.

Proposition 1 [54]. *Let $\bar{r}_k$ be an upper bound on the optimal dual variable for the* *constraint $y_k \le u_k(1 - x_k)$ in [ADLP1] taken over all $\mathbf{x} \in X$. Let $\bar{\mathbf{r}} = (\bar{r}_1 \ldots \bar{r}_n)^{\mathrm{T}}$ and $\overline{R} =$ diag($\bar{\mathbf{r}}$), and define*

$$\text{[ADLP2]} \quad z_2^* = \min_{\mathbf{x} \in X} z_2(\mathbf{x}), \quad \textit{where} \tag{10}$$

$$z_2(\mathbf{x}) \equiv \max_{\mathbf{y}} \left(\mathbf{c}^{\mathrm{T}} - \mathbf{x}^{\mathrm{T}}\overline{R}\right)\mathbf{y} \tag{11}$$

$$\text{s.t.} \quad A\mathbf{y} \le \mathbf{b} \qquad [\mathbf{q}] \tag{12}$$

$$0 \le \mathbf{y} \le \mathbf{u}. \qquad [\mathbf{r}]. \tag{13}$$

*(The vectors **q** and **r**, used later, denote dual variables for their respective constraint sets when **x** is fixed.) Then, [ADLP1] and [ADLP2] are equivalent in the sense that $z_1^* = z_2^*$, and $\mathbf{x}^*$ solves [ADLP2] if and only if it also solves [ADLP1].*

Note also that $z_2(\mathbf{x})$ is a convex function in continuous **x**.

That [ADLP2] is equivalent to [ADLP1] is intuitively clear, at least when the $\bar{r}_k$ are strict upper bounds: $(\mathbf{x}^*, \mathbf{y}^*)$ solves [ADLP1] if and only if it also solves [ADLP2]. Nonstrict bounds can lead to cases where $(\mathbf{x}^*, \mathbf{y}^*)$ is optimal [ADLP2] but infeasible to [ADLP1]. In this case, however, there must exist some $\mathbf{y}^{**}$ such that $(\mathbf{x}^*, \mathbf{y}^{**})$ is optimal to [ADLP1].

To solve [ADLP2], (i) temporarily fix the variables **x** in [ADLP2] (i.e., treat **x** as data); (ii) take the dual of the resulting LP; and (iii) then release **x**. The following, equivalent mixed-integer program (MIP) results:

$$\text{[ADMIP2]} \quad \min_{\mathbf{x} \in X,\, \mathbf{q},\, \mathbf{r}} \quad \mathbf{b}^{\mathrm{T}}\mathbf{q} + \mathbf{u}^{\mathrm{T}}\mathbf{r} \tag{14}$$

$$\text{s.t.} \quad A^{\mathrm{T}}\mathbf{q} + I\mathbf{r} + \overline{R}\mathbf{x} \ge \mathbf{c} \tag{15}$$

$$\mathbf{q} \ge \mathbf{0}, \quad \mathbf{r} \ge \mathbf{0} \tag{16}$$

A standard LP-based branch-and-bound algorithm will solve [ADMIP2] if that model is not too large.

Good dual bounds $\bar{\mathbf{r}}$ are important for solving [ADMIP2] directly, but are not easy to come by except in a few instances. For instance, if the inner LP in [ADLP] corresponds to a maximum-flow model with integral capacity vector **u** , then $\bar{r}_k = 1$ is valid and tight because the value of an extra unit of arc capacity in a maximum-flow problem is 0 or 1 [35]. Theoretically, we could also use $\bar{r}_k = 100$ in this case, but the resulting LP relaxation of [ADMIP2] would be weak and solution times would suffer.

The decomposition solution method for [ADLP2] described next does not eliminate the need for good bounds on dual variables, but ancillary techniques can alleviate some of the difficulties caused by weak bounds, and decomposition has some key advantages over a branch-and-bound solution of [ADMIP2].

1. In the context of network interdiction, various studies [55,26], have demonstrated that decomposition typically solves [ADMIP2] much faster than does branch-and-bound,
2. As we shall see, decomposition can be extended to solve BNI even when the defender's optimization model is more general than an LP, and
3. Decomposition methods typically solve a sequence of "defender subproblems" in a familiar, user-friendly form, which obviates the complicated, unfamiliar dual constructs of [ADMIP2]. For instance, Salmerón *et al*. [45] evaluate the effects of an attack plan **x** on a large, regional electric-power grid using a standard electric-power model. In contrast, a MIP formulation for BNI in this case becomes unwieldy (and can only be solved for small, unrealistic test problems) [56].

BENDERS DECOMPOSITION

The decomposition algorithm described here may be viewed as solving [ADLP2] by applying Benders decomposition to [ADMIP2] [57]. The Benders methodology for solving a minimizing MIP first converts the MIP into a min–max problem by reversing the steps that we used to create [ADMIP2] from [ADLP2]: (i) temporarily fix the integer variables, (ii) take the dual of the resulting LP, and (iii) release the integer variables [58, pp. 135–143]. In the case of [ADMIP2], this conversion returns us to the more natural starting point of [ADLP2].

An optimal solution in $\mathbf{y}$ to [ADLP1] occurs at an extreme point of the (bounded) feasible region of that problem's inner LP. Because of the essential equivalence of the problems, the same holds true for solutions in $\mathbf{y}$ to [ADLP2]. Let Y denote the full, finite set of extreme points for the latter problem. Then, [ADLP2] may be expressed as this *equivalent master problem*

$$z_2^*(Y) = \min_{\mathbf{x}\in X} z_2(\mathbf{x}, Y), \quad \text{where}$$

$$z_2(\mathbf{x}, Y) \equiv \max_{\hat{\mathbf{y}}\in Y} \left(\mathbf{c}^{\mathrm{T}} - \mathbf{x}^{\mathrm{T}}\overline{R}\right)\hat{\mathbf{y}}. \tag{17}$$

That model has this formulation as a MIP

[ADMP2(Y)]

$$z_2^*(Y) = \min_{\mathbf{x},\, z} \; z \tag{18}$$

$$\text{s.t.} \quad z + \hat{\mathbf{y}}^{\mathrm{T}}\overline{R}\mathbf{x} \geq \mathbf{c}^{\mathrm{T}}\hat{\mathbf{y}} \quad \forall\, \hat{\mathbf{y}} \in Y \tag{19}$$

$$\mathbf{x} \in X. \tag{20}$$

Benders decomposition dynamically generates constraints (19) called *Benders cuts*. It solves or approximately solves the original problem by finding $\hat{Y} \subset Y$, with $|\hat{Y}| << |Y|$ it is hoped, so that [ADMP2($\hat{Y}$)] is an adequate approximation of the equivalent master problem.

Algorithm A-1. Basic Benders decomposition algorithm to solve [ADLP2]

Input: An instance of [ADLP2] and allowable optimality gap $\epsilon \geq 0$.
Output: An ϵ-optimal interdiction plan for [ADLP2], and associated objective value;
Step 0: $\hat{Y}\leftarrow\emptyset$; $\underline{z}\leftarrow -\infty$; $\overline{z}\leftarrow\infty$; $\hat{\mathbf{x}}\leftarrow\mathbf{0}$; $\hat{\mathbf{x}}^*\leftarrow\mathbf{0}$; *gap* $\leftarrow \infty$;
Step 1: Fix $\mathbf{x} = \hat{\mathbf{x}}$ in [ADLP2] and solve for $\hat{\mathbf{y}}$ and objective value $\hat{z} \equiv z_2(\mathbf{x})$; $\hat{Y}\leftarrow\hat{Y}\cup\{\hat{\mathbf{y}}\}$;
If $\hat{z} < \overline{z}$ then $\overline{z}\leftarrow\hat{z}$ and $\hat{\mathbf{x}}^*\leftarrow\hat{\mathbf{x}}$;
If $\overline{z} - \underline{z} \leq \epsilon$ then go to Step 3;
Step 2: Solve [ADMP2($\hat{Y}$)] for $z_2^*(\hat{Y})$ and $\hat{\mathbf{x}}$; $\underline{z}\leftarrow z_2^*(\hat{Y})$;
If $\overline{z} - \underline{z} > \epsilon$ then go to Step 1;
Step 3: Print "Approximate solution is," $\hat{\mathbf{x}}^*$, "with objective value,"$\overline{z}$;
Print "Provable optimality gap is", $\overline{z} - \underline{z}$;
Stop;
End of Algorithm A-1.

Algorithm A-1, or simply "A-1," is actually a special case of Benders decomposition that does not require "feasibility cuts" [59]. Such cuts are needed if the subproblems can become infeasible for certain $\mathbf{x} \in X$, which ours cannot by assumption. The correctness of the algorithm is easy to see: (i) The upper bound $\overline{z}$ is valid because it corresponds to some feasible solution of [ADLP2] for the minimizing attacker; (ii) the lower bound $\underline{z}$ is valid because it corresponds to a relaxation of the equivalent master problem [ADMP2(Y)]; (iii) if a solution $\hat{\mathbf{x}}$ ever repeats, it follows that $\overline{z} = \underline{z}$ and the algorithm must terminate; and (iv) the termination criterion is satisfied with $\underline{z} \neq \overline{z}$ or a solution repeats in a finite number of steps because X is a discrete, finite set.

The following section describes some enhancements to A-1 that may improve solution speeds, and shows how global Benders decomposition can actually solve more general problems than [ADLP2].

IMPROVING AND GENERALIZING BENDERS DECOMPOSITION

Faster Solutions with Super-Valid Inequalities

The *(relaxed) master problem* [ADMP1($\hat{Y}$)] can be strengthened in some instances by adding *super-valid inequalities* (SVIs). Intuitively, this strengthening can help alleviate some of the difficulties caused by weak dual bounds $\bar{\mathbf{r}}$. SVIs are similar to valid inequalities of integer-programming theory [60, pp. 205–295] except that they may, and typically do, eliminate feasible solutions from the master problem. We define SVIs with respect to general MIPs.

Definition. Let $\mathbf{x}$ and $\mathbf{y}$ denote the vectors of integer and continuous variables, respectively, in a MIP. The inequality $\mathbf{w}_1^{\mathrm{T}}\mathbf{x} + \mathbf{w}_2^{\mathrm{T}}\mathbf{y} \geq w_0$ is *super-valid* for this MIP if (i) adding that inequality to the MIP does not eliminate all optimal solutions, or (ii) an incumbent solution $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ is (already) optimal for the MIP.

With proper precautions, SVIs may be used within a branch-and-bound algorithm for [ADMIP2] as well as within Benders decomposition for solving [ADLP2]. Suppose, for instance, that we add a single SVI in the course of solving a MIP by either technique. If case (i) is true for that SVI, then an optimal solution will still be found via enumeration because (a) some optimal solution is still feasible, and (b) any lower bound obtained from a relaxation of the SVI-modified MIP is still a valid lower bound on z_2^*. Thus, standard fathoming tests within a branch-and-bound algorithm and the convergence tests in A-1 are valid. If case (ii) is true when we add the SVI, we simply want our algorithm to halt with a message that the incumbent $\hat{\mathbf{x}}^*$ is optimal, and this is easy to arrange. After adding the SVI

1. If the MIP is found to be infeasible, or $\underline{z} > z(\hat{\mathbf{x}}^*)$, we declare the incumbent optimal, which it is, or
2. If $\overline{z} - \underline{z} \leq \epsilon$ occurs, we declare the incumbent to be ϵ-optimal, which it is. (As often happens, our incumbent is optimal, but we only prove it to be ϵ-optimal.)

By induction, it follows that an enumeration algorithm incorporating a finite number of SVIs will also terminate correctly and finitely.

One type of SVI for our enhanced version of A-1 applied to [ADLP2] is easy to derive.

Proposition 2 [26]. *Let* $z + \hat{\mathbf{y}}^{\mathrm{T}}\overline{R}\mathbf{x} \geq \mathbf{c}^{\mathrm{T}}\hat{\mathbf{y}}$ *denote a Benders cut from Algorithm A-1 being used to solve [ADLP2].*

$$I_k(\hat{\mathbf{y}}) = \begin{cases} 1 & \text{if } \hat{y}_k > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{21}$$

Then, the following inequality is super-valid:

$$\mathbf{I}(\hat{\mathbf{y}})^{\mathrm{T}}\mathbf{x} \geq 1. \tag{22}$$

Suppose that $\overline{R}$ derives from strict dual bounds and that $\hat{\mathbf{y}}$ is the response to the feasible interdiction plan $\hat{\mathbf{x}}$. It follows that $\hat{\mathbf{y}}^{\mathrm{T}}\overline{R}\hat{\mathbf{x}} = \mathbf{I}(\hat{\mathbf{y}})^{\mathrm{T}}\hat{\mathbf{x}} = 0$, and that $\hat{\mathbf{x}}$ is made infeasible by the SVI $\mathbf{I}(\hat{\mathbf{y}})^{\mathrm{T}}\mathbf{x} \geq 1$. That is, the inequality $\mathbf{I}(\hat{\mathbf{y}})^{\mathrm{T}}\mathbf{x} \geq 1$ is not valid in the standard sense. Note also that $\hat{\mathbf{x}}$ could be an optimal solution which is made infeasible by the inequality, but if we already have an optimal solution in hand, we have free reign to restrict the solution in any way we like.

A simple extension of Proposition 2 leads to

Corollary 1. *For every Benders cut* $z + \hat{\mathbf{y}}^{\mathrm{T}}\overline{R}\mathbf{x} \geq \mathbf{c}^{\mathrm{T}}\hat{\mathbf{y}}$*, the SVI of Proposition 2 can be tightened to* $\mathbf{I}(\hat{\mathbf{y}})^{\mathrm{T}}\mathbf{x} \geq 2$ *if* $\mathbf{c}^{\mathrm{T}}\hat{\mathbf{y}} - \max_j \overline{r}_k\hat{y}_k > \overline{z}$*, can be tightened to* $\mathbf{I}(\hat{\mathbf{y}})^{\mathrm{T}}\mathbf{x} \geq 3$ *if* $\mathbf{c}^{\mathrm{T}}\hat{\mathbf{y}} - \max_{k\neq k'}\{\overline{r}_k\hat{y}_k + \overline{r}_{k'}\hat{y}_{k'}\} > \overline{z}$*, and so on.*

Of course, as $\overline{z}$ changes during the course of A-1, it may be possible to tighten previously generated SVIs.

Modifications of A-1 to incorporate SVIs are straightforward, and SVIs may improve solution times substantially. Israeli and Wood [26] demonstrate this and show how to (i) add heuristically generated SVIs to an instance of [ADMIP2] to improve branch-and-bound solution times, (ii) generalize SVIs to "ϵ-SVIs" that guarantee not to eliminate all ϵ-optimal solutions, and (iii) solve [ADLP1] and [ADLP2] in a decomposition algorithm whose master problem constraints consist solely of SVIs. The "covering algorithm" alluded to in (iii) uses no dual bounds $\overline{\mathbf{r}}$ at all, and converts Benders decomposition into a purely combinatorial procedure.

Standard Computational Enhancements for Benders Decomposition

In addition to employing SVIs, Algorithm A-1 can benefit from more standard techniques used to improve solution speeds for Benders decomposition [46]. Some of these techniques are discussed briefly below.

1. For fixed $\mathbf{x}$, [ADLP2] may have multiple extreme-point solutions $\mathbf{y}$ and a different Benders cut can be generated for each. Adding too many such cuts can slow down solutions of [ADMP2], but adding them judiciously can improve solution times greatly. Israeli and Wood [26] use this technique in the shortest-path interdiction problem, where they enumerate multiple shortest paths for a single attack plan $\mathbf{x}$ and generate cuts for each.
2. The master problem need not be solved to optimality if "sufficient progress" is made after each cut is added [61].
3. Cuts derived from interior-point subproblem solutions $\hat{\mathbf{y}}$ may prove better than those derived from extreme-point solutions. For instance, an arc k with a large flow $\hat{y}_k$ on it appears as an attractive candidate for interdiction in the solution of the maximum-flow interdiction problem. That is, it generates a cut with a large-magnitude entry in position k. But $\hat{y}_k$ may be large primarily because the solution is an extreme-point solution, not because it must be large to achieve a maximum s-t flow. So, A-1 may waste time exploring solutions with $x_k = 1$. In contrast, an interior-point solution "spreads flow around" the network, and $\hat{y}_k$ will tend to be large only if it needs to be in order to achieve a maximum s-t flow. Consequently, better guidance and better cuts may be derived from such solutions.
4. Some Benders cuts can be dominated (implied) by others, and the nondominated ones ought to be used for the sake of efficiency. Magnanti and Wong [46] provide guidance on this topic. The related work of Smith *et al*. [47] may also prove useful: that paper shows how a polynomial-sized reformulation of the master problem in an interdiction model can yield cuts that dominate an exponential number of cuts from the original formulation.

Global Benders Decomposition

Algorithm A-1 can be extended to solve instances of [AD0], in which the defender's operational model is more complicated than an LP. For example, let "[ADIP2]" denote a model identical to [ADLP2] except that $\mathbf{y}$ is required to be integral. A-1 clearly solves this problem because we can replace "the finite set of extreme points Y" used to define the equivalent master problem [ADMP2] with "the finite set of integer solutions" for [ADIP2]. Geoffrion [62] coins the phrase "generalized Benders decomposition" to describe extensions of Benders decomposition to nonlinear models analogous to [ADLP2] with convex objective functions $z_2(\mathbf{x})$; Salmerón *et al*. [45] therefore use the phrase "global Benders decomposition" to describe the solution of other models like [ADIP2], in which the issues of convexity may even be irrelevant.

In [ADIP2], $\bar{r}_k$ no longer corresponds to a bound on a dual variable. Rather, that datum must comprise an integral part of the original formulation. For instance, [ADIP2] might correspond to a max−min instance of BNI in which an attacker seeks to delay completion of a defender's project, which is modeled through the constructs of a resource-constrained PERT network. (Davis [63] discusses such PERT networks, and Brown *et al*. [40,44] discuss interdicting them.) If $x_k = 1$, task k in the project is attacked and delayed by a fixed amount: that is what $-\bar{r}_k$ would represent in [ADIP2]; otherwise $x_k = 0$ and task k requires some nominal time to complete, corresponding to c_k in that model.

We can generalize further.

Proposition 3 [45]. *Suppose that BNI has the following form:*

$$[AD3] \quad \min_{\mathbf{x}\in X} z_3(\mathbf{x}), \quad \textit{where}$$
$$z_3(\mathbf{x}) \equiv \max_{\mathbf{y}\in Y} f(\mathbf{x},\mathbf{y}), \qquad (23)$$

and where X is defined as in Equation (5), $\mathbf{y} \in Y$ can be discrete and/or continuous, and $f(\mathbf{x},\mathbf{y})$ has a general form. Furthermore, suppose that penalty vectors $\mathbf{v}(\mathbf{x})$ *can be defined so that*

$$z_3(\mathbf{x}) \geq z_3(\hat{\mathbf{x}}) + \mathbf{v}^{\mathrm{T}}(\hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}}) \; \forall \; \mathbf{x}, \; \hat{\mathbf{x}} \in X. \tag{24}$$

Then, the following master problem is equivalent to [AD3]:

[ADMP3]

$$\min_{\mathbf{x} \in X, z} z \tag{25}$$

$$\text{s.t.} \quad z - \mathbf{v}(\hat{\mathbf{x}})^{\mathrm{T}}\mathbf{x} \geq z_3(\hat{\mathbf{x}}) - \mathbf{v}^{\mathrm{T}}(\hat{\mathbf{x}})\hat{\mathbf{x}} \; \forall \; \hat{\mathbf{x}} \in X$$

Given the existence of an equivalent master problem, [AD3] may be solved via a modified version of A-1. We may assume that the attacker has an efficient method for computing $z_3(\hat{\mathbf{x}})$, that is, for evaluating the effects of an attack plan $\hat{\mathbf{x}}$ through the solution of the defender's subproblem. For instance, to evaluate the effects of attack plan $\hat{\mathbf{x}}$ on an electric-power transmission grid, the attacker can solve a nonlinear "AC optimal power-flow model" or a standard, faster, LP approximation, a "DC optimal power-flow model" [64]. Thus, the difficult part here will be defining and computing appropriate penalty vectors $\mathbf{v}(\mathbf{x})$. That task will be problem-dependent, so we expand upon the power-grid example to illustrate.

An attacker wishes to maximize the short-term, unserved demand for power in a defender's transmission grid. Thus, [ADMP3] must be converted to a maximization problem, and the inequality in Equation (26) reversed. When $\hat{x}_k = 0$, $v_k(\hat{\mathbf{x}})$ should bound the amount of unserved demand that will accrue if the status of grid component k is changed from "unattacked and functional" to "attacked and nonfunctional." The power-handling capability of the component provides a simple bound which is usually valid. If $\hat{x}_k = 1$, $v_k(\hat{\mathbf{x}})$ must reflect how much unserved demand will be eliminated if component k's status is changed in the opposite direction. Because of the existence of series components, $v_k(\hat{\mathbf{x}}) = 0$ is a reasonable, albeit crude, approximation. (Actually, unserved demand can increase after repairing a component, but this does not normally cause difficulties [45]).

The subproblem in this example is merely a LP, and an attack on component k does force its capacity to 0 as in [ADLP1]. A complication arises, however, because the destruction of a component can also improve power flow by eliminating one or more "susceptance constraints" between power lines having common end points [64]. Thus, $z_0(\mathbf{x})$ is neither concave nor convex in this application. However, this function tends to be well-behaved in practice, and the corresponding penalty vector $\mathbf{v}(\mathbf{x})$ is easily computed. The definition of that vector may seem simplistic, but Salmerón *et al*. [45] use it within global Benders decomposition to solve interdiction models on full-scale, regional transmission grids. An added benefit of the global Benders approach is that it extends to the trilevel network-defense problem for a power grid.

CONCLUSIONS

This article has described mathematical techniques for modeling and solving a BNI. BNI is a two-person, zero-sum, two-stage, sequential-play (Stackelberg) game whose solution prescribes an optimal application of limited resources to attack components of an enemy's network, and thereby limit that network's usefulness to the enemy. When a LP suffices to model optimal network operation, we show that BNI can be converted to and solved as a MIP. But, we describe special decomposition techniques that typically solve these problems more efficiently and, importantly, can solve more general problems.

REFERENCES

1. Webster. Webster's third new international dictionary. Springfield (MA): Merriam-Webster, Inc.; 1993.
2. Herodotus. Translated by Grene D, editor. The history. Chicago: University of Chicago Press; 1987.
3. Livy (Titus Livius). Translated by de Sélincourt A, editor. The war with Hannibal. Middlesex, England: Penguin Books; 1965.
4. Polybius. Translated by Scott-Kilvert I, editor. The rise of the Roman Empire. London: Penguin Books; 1979.
5. Foote S. The civil war. New York: Random House; 1974.

6. Blair C. Hitler's U-Boat war. New York: Random House; 1996.
7. MacIsaac D. Strategic bombing in world war two. New York: Garland Publishing, Inc.; 1976.
8. Joint Chiefs of Staff. Doctrine for joint operations. Joint Pub 3-0; Available at http://www.dtic.mil/doctrine/jel/new_pubs/jp3_0.pdf. Accessed 1995.
9. Joint Chiefs of Staff. Doctrine for joint interdiction operations. Joint Pub 3-03; Available at http://www.dtic.mil/doctrine/jel/new_pubs/jp3_03.pdf. Accessed 1997.
10. Brown G, Carlyle M, Salmerón J, *et al*. Defending critical infrastructure. Interfaces 2006;36:530–544.
11. von Stackelberg H. The theory of the market economy. London: German, William Hodge & Co.; 1952.
12. Simaan M, Cruz JB. On the stackelberg strategy in nonzero-sum games. J Optim Theor Appl 1973;11:533–555.
13. Wollmer R. Removing arcs from a network. Oper Res 1964;12:934–940.
14. Ahuja RK, Magnanti TL, Orlin JB. Network flows: theory, algorithms, and applications. Upper Saddle River (NJ): Prentice-Hall; 1993.
15. Harris TE, Ross FS. Fundamentals of a method for evaluating rail net capacities. Research Memorandum RM-1573, Santa Monica (CA): The RAND Corporation; 1955.
16. Danskin JM. The theory of max-min, with applications. SIAM J Appl Math 1966;14: 641–664.
17. von Neumann J, Morgenstern O. Theory of games and economic behavior. Princeton (NJ): Princeton University Press; 1953.
18. (a) von Neumann J. Zur theories der gesellschaftsspiele. Math Ann 1928;100:295–320; (b) translated by Bergmann S, Luce RD, Tucker AW, editors. Contributions to the theory of games IV. Princeton (NJ): Princeton University Press; 1959. pp. 13–42.
19. McMasters AW, Mustin TM. Optimal interdiction of a supply network. Naval Res Logist Q 1970;17:261–268.
20. Ghare PM, Montgomery DC, Turner TM. Optimal interdiction policy for a flow network. Naval Res Logist Q 1971;18:37–45.
21. Fulkerson DR, Harding GC. Maximizing the minimum source-sink path subject to a budget constraint. Math Program 1977;13:116–118.
22. Golden B. A problem in network interdiction. Naval Res Logist Q 1978;25:711–713.
23. Corley HW, Shaw DY. Most vital links and nodes in weighted networks. Oper Res Lett 1982;1:157–160.
24. Malik K, Mittal AK, Gupta SK. The k-most vital arcs in the shortest path problem. Oper Res Lett 1989;8:223–227.
25. Ball MO, Golden BL, Vohra RV. Finding the most vital arcs in a network. Oper Res Lett 1989;8:73–76. (NP-completeness of BSI).
26. Israeli E, Wood RK. Shortest-path network interdiction. Networks 2002;40:97–111.
27. Ratliff HD, Sicilia GT, Lubore SH. Finding the n most vital links in flow networks. Manage Sci 1975;21:531–539.
28. Wood RK. Deterministic network interdiction. Math Comput Model 1993;17:1–18.
29. Phillips CA. The network inhibition problem. STOC '93: Proceedings of the Twenty-fifth Annual ACM Symposium on Theory of Computing. New York: ACM Press; 1993. pp. 776–785.
30. Steinrauf R. A network interdiction model [Master's Thesis]. Monterey (CA): Operations Research Department, Naval Postgraduate School; 1991.
31. Grötschel M, Monma C, Stoer M. Computational results with a cutting plane algorithm for designing communication networks with low-connectivity constraints. Oper Res 1992;40:309–330.
32. Medhi D. A unified approach to network survivability for teletraffic networks: models, algorithms and analysis. IEEE Trans Commun 1994;42:534–548.
33. Chern MS, Lin KC. Interdicting the activities of a linear program - a parametric analysis. Eur J Oper Res 1995;86:580–591.
34. Washburn AR, Wood RK. Two-person zero-sum games for network interdiction. Oper Res 1994;43:243–251.
35. Cormican KJ, Morton DP, Wood RK. Stochastic network interdiction. Oper Res 1998;46: 184–197.
36. Whiteman PS. Improving single strike effectiveness for network interdiction. Mil Oper Res 1999;4(1):15–30.
37. Pan F, Charlton W, Morton D. A stochastic program for interdicting smuggled nuclear material. In: Woodruff DL, editor. Network interdiction and stochastic integer programming. Dordrecht, The Netherlands: Kluwer Academic Publishers; 2003. pp. 1–20.
38. Morton DP, Pan F, Saeger KJ. Models for nuclear smuggling interdiction. IIE Trans 2007;39:3–14.

39. Brown G, Carlyle M, Royset J, *et al*. On the complexity of delaying an adversary's project. In: Golden B, Raghavan S, Wasil E, editors. The next wave in computing, optimization and decision technologies. New York: Springer; 2005. pp. 3–17.
40. Brown G, Carlyle M, Salmerón J, *et al*. Analyzing the vulnerability of critical infrastructure to attack, and planning defenses. In: Greenberg H, Smith J, editors. Tutorials in operations research: emerging theory, methods, and applications. Hanover (MD): Institute for Operations Research and Management Science; 2005.
41. Brown G, Carlyle M, Diehl D, *et al*. A two-sided optimization for theater ballistic missile defense. Oper Res 2005;53:263–275.
42. Lim C, Smith JC. Algorithms for discrete and continuous multicommodity flow network interdiction problems. IIE Trans 2007;39:15–26.
43. Barkley TR. An attacker-defender model for IP-based networks [Master's Thesis]. Monterey (CA): Operations Research Department, Naval Postgraduate School; 2008.
44. Brown G, Carlyle M, Harney R, *et al*. Interdicting a nuclear weapons project. Oper Res 2009;57:866–877.
45. Salmerón J, Wood K, Baldick R. Worst-case interdiction analysis of large-scale electric power grids. IEEE Trans Power Syst 2009;24:96–104.
46. Magnanti TL, Wong RT. Accelerating Benders decomposition: algorithmic enhancement and model selection criteria. Oper Res 1981;29:464–484.
47. Smith JC, Lim C, Alptekinoglu A. Optimal mixed-integer programming and heuristic methods for a bilevel stackelberg product introduction game. Naval Res Logist 2009;56:714–729.
48. Morton D, Rosenthal RE, Lim T. Optimization modeling for airlift mobility. Mil Oper Res 1997;1(4):49–68.
49. Mehring JS, Gutterman MM. Supply and distribution planning support for Amoco (U.K.) Limited. Interfaces 1990;20(4): 95–104.
50. Ben-Ayed O. Bi-level linear programming. Comput Oper Res 1993;20:485–501.
51. Bard J, Moore J. A branch and bound algorithm for the bi-level programming problem. SIAM J Sci Stat Comput 1990;11: 281–292.
52. Wen U, Yang Y. Algorithms for solving the mixed integer two-level linear programming problem. Comput Oper Res 1990;17:133–142.
53. Hansen P, Jaumard B, Savard G. New branch-and-bound rules for linear bi-level programming. SIAM J Sci Stat Comput 1992;13:1194–1217.
54. Morton DP, Wood RK. Restricted recourse bounds for stochastic linear programming. Oper Res 1999;47:943–956.
55. Cormican KJ. Computational methods for deterministic and stochastic network interdiction problems [Masters Thesis]. Monterey (CA): Operations Research Department, Naval Postgraduate School; 1995.
56. Motto AL, Arroyo JM, Galiana FD. A mixed-integer LP procedure for the analysis of electric grid security under terrorist threat. IEEE Trans Power Syst 2005;20:1357–1365.
57. Benders JF. Partitioning procedures for solving mixed integer variables programming problems. Numer Math 1962;4:238–252.
58. Garfinkel RS, Nemhauser GL. Integer Program. New York: John Wiley & Sons; 1972.
59. Birge JR. Decomposition and partitioning methods for multistage stochastic linear programs. Oper Res 1985;33:989–1007.
60. Nemhauser GL, Wolsey LA. Integer and combinatorial optimization. New York: Wiley-Interscience; 1988.
61. Brown GG, Graves G, Honczarenko M. Design and operation of a multicommodity production/distribution system using primal goal decomposition. Manage Sci 1987;33: 1469–1480.
62. Geoffrion A. Generalized Benders decomposition. J Optim Theor Appl 1972;10:237–260.
63. Davis EW. Project scheduling under resource constraints: historical review and categorization of procedures. AIIE Trans 1973;5: 297–313.
64. Overbye TJ, Cheng X, Sun Y. A comparison of the AC and DC power flow models for LMP calculations. Proceedings, 37th Hawaii International Conference on System Sciences. Hawaii: 2004.