A Stochastic Program For Optimizing Military Sealift Subject To Attack

Dr. Javier Salmeron

Naval Postgraduate School
jsalmero@nps.edu

Dr. R. Kevin Wood

Naval Postgraduate School
kwood@nps.edu

Dr. David P. Morton

The University of Texas at Austin
morton@mail.utexas.edu

APPLICATION AREA:
Mobility and Transport of Forces

OR METHODOLOGIES:
Stochastic programming

ABSTRACT

We describe a stochastic program for planning the wartime, sealift deployment of military cargo that is subject to attack. The cargo moves on ships from U.S. or allied seaports of embarkation, to seaports of debarkation (SPODs) near the theater of war where it is unloaded and sent on to final, in-theater destinations. The question we ask is: Can a deployment-planning model, with probabilistic information on the time and location of potential enemy attacks on SPODs, successfully hedge against those attacks? That is, can this information be used to reduce the expected disruption caused by such attacks? We develop a specialized, stochastic mixed-integer program whose solutions answer that question in the affirmative for realistic deployment data. Furthermore, compared to the optimal deterministic solution, the stochastic solution incurs only a minor "disruption penalty" when no attack occurs, and outcomes for worst-case scenarios are better. Insight gained from the stochastic-programming approach also points to possible improvements in current, rule-based, scheduling methods.

INTRODUCTION

The United States Transportation Command (USTRANSCOM) plans the wartime deployment of U.S. cargo ships, and their cargo, from U.S. or allied seaports of embarkation (SPOEs) to overseas seaports of debarkation (SPODs) (USTRANSCOM 2000). This command uses little optimization to guide its planning for a deployment and, to our knowledge, no stochastic optimization to accommodate uncertainty. The purpose of this paper is (a) to develop a stochastic-optimization model that proactively plans for potential disruptions caused by enemy attacks on SPODs, and (b) to illustrate the potential benefit of using such a model with realistic deployment data.

The Problem

A military sealift deployment is driven by a flexible schedule of movement requirements contained in the Time-Phased Force-Deployment Data (TPFDD). The TPFDD, with a typical timeframe of 100 days, describes the cargo needed in the deployment and the military units to which that cargo belongs, e.g., a Marine Expeditionary Force or a Naval Mobile Construction Battalion. The TPFDD defines time windows for when cargo will be available for loading at the SPOEs, when it should pass through an SPOD, and when it should arrive at its in-theater destination. The hypothetical TPFDD used in this paper, which is based on multiple unclassified sources (see "Data" under "Computational Results" section), represents realistic estimates of cargo requirements needed to support a major regional contingency in the Persian Gulf.

Currently, TPFDD-based planning uses software tools like the Joint Flow and Analysis System for Transportation, or "JFAST" (USTRANSCOM 2000). However, the emphasis in the last few years has been on embedding such systems within a global command-and-control system so that all cargoes and lift assets are visible to planners who must deal with contingencies "on the fly." Quick responses to contingencies are important, but JFAST is a rule-based system that cannot optimize (or re-optimize) a schedule with respect to an objective such as "minimize delay." Furthermore, JFAST ignores potential disruptions to the deployment caused by enemy attacks (Koprowski 2005).

The deterministic mixed-integer programs of Aviles (1995) and Brown (1999), along with the deterministic version of the model described in this paper, address the lack of optimization in existing sealift deployment-planning systems. Within the limits of modeling approximations, these models provide an exact assignment and routing of ships to deliver the TPFDD cargoes as best possible. The models typically minimize the ton-days of late cargo, which are weighted in some fashion with respect to the amount of lateness.

While a deterministic optimization model is potentially useful, it ignores the fact that an enemy may disrupt the deployment by attacking the cargo-movement "network," probably in some forward area, i.e., near the SPODs. The potential for such disruptions is of increasing concern within the U.S. military (Joint Publication 3-11 2000, p. II-3). Attacks might be carried out by mining harbors and/or shipping channels, or by attacking SPODs with missiles carrying conventional, nuclear, chemical or biological warheads, or by terrorist attack. Therefore, our main question is: Can we plan a sealift deployment while effectively hedging against the potential disruption caused by attacks on our cargo-movement network in forward areas? Our purpose is to convince planners that current planning tools can be improved: Not only should these tools optimize, but they should also plan proactively for potential disruptions.

We build a stochastic mixed-integer programming model, called the "Stochastic Sealift Deployment Model" (SSDM), to address these issues. We focus on biological attacks on SPODs, because of special concern about such attacks (Joint Publication 3-11 2000, p. III-30). However, the concern about potential attacks on SPOEs has also increased (e.g., Army Logistician 2001, Larson and Peters 2001, pg. 119), and a "symmetric version" of our model could be used to analyze this situation. However, a model to analyze the possibility of attacks on both SPOEs and SPODs would be substantially more difficult to solve.

Biological weapons are not new, but their potential for serious military use has increased in recent years (Cohen 1997, Defense Intelligence Agency 1998), and a biological attack on an SPOD could certainly disrupt a deployment. Furthermore, biological weapons are inexpensive to produce, and over a dozen of the United States' potential adversaries may possess or may be engaged in research on such weapons (Barnaby 1999, pp. 10–11). Thus, the threat must be taken seriously. We assume that any biological attack is immediately detected, as would be the case with biological warheads delivered by ballistic missiles. This may not be a limiting assumption because new detection systems are capable of quickly detecting biological warfare agents that might be surreptitiously spread by terrorists or an enemy's special operations forces. (For example, see the papers in Leonelli and Althouse 1998.) An attacked SPOD will shut down entirely during a decontamination period, after which its cargo-handling capacity will return to normal gradually, following some recovery schedule. The severity of the attack dictates the length of the decontamination period and the recovery schedule. The state of the art in determining the potential damage caused by a biological attack is not far advanced (Alexander 1999), but SSDM can be easily adjusted to account for the latest information as it becomes available.

For simplicity, we assume at most one attack will occur during the deployment period, although that attack may strike more than one SPOD. The timing and location(s) of the attack are uncertain and follow a probability distribution developed by intelligence reports and planners. The single-attack assumption has one significant advantage: It enables us to model the deployment using a special type of stochastic program, which is easier to solve than a multiple-attack model. This assumption is reasonable at this stage of study, because no current deployment-planning models account for even a single attack, and because important insight can be gained by studying this case.

We also note that SSDM ignores airlift assets and airlift delivery requirements. In fact, airlift and sealift optimization may be considered separately, because the mode of transport for each cargo "package" is specified by the TPFDD, and the two transportation networks share few resources. Focusing on sealift makes sense because most cargo in a deployment moves by sea. For instance, in the Persian Gulf War of 1991, sealift delivered about 85% of all dry cargo (Lund et al. 1993).

Stochastic Programming and Military Deployments

Stochastic programming has seen limited application in military deployment problems, even though the study of related transportation problems under uncertainty reaches back to Ferguson and Dantzig (1956). A notable early exception is an application of two-stage stochastic programming for scheduling monthly and daily airlift with uncertain cargo demands (Midler and Wollmer 1969). Modest computational power has, presumably, impeded the application of similar techniques to modern large-scale military mobility systems.

Currently, simulation is the preferred method of dealing with uncertainty in deployments.

The Warfighting and Logistics Technology Assessment Environment (WLTAE) links warfighting and logistics simulation models into a single large simulation (Sinex et al. 1998). The logistics modules of such simulations typically use rule-based methods like JFAST and have limited, if any, capability for optimal rescheduling after an attack. However, we note that Brown (1999) does describe re-optimization techniques suitable for embedding in the WLTAE simulation model. In particular, whenever a modeled disruption in the deployment takes place, his mixed-integer program, or a faster heuristic, can re-schedule the next set of ships and cargoes to be deployed.

A series of optimization models for planning sealift deployment has been developed at the Naval Postgraduate School (Aviles 1995, Theres 1998, Alexander 1999, Brown 1999, Loh 2000, Koprowski 2005). All of these contribute to our understanding of the problem, but all have significant limitations. For instance, Theres (1998) ignores uncertainty; Aviles (1995) and Brown (1999) plan using deterministic models that assume no disruptions will occur and then re-optimize after a simulated disruption (an attack) does occur; Alexander (1999) and Loh (2000) have explicit stochastic models, but can handle only small problems and have unrealistic limitations on, or relaxations of, post-attack recourse. Kaprowski (2005) models only worst-case attacks and does not solve those models optimally.

Deterministic airlift optimization models, analogous in concept to the above sealift models, have been developed by Killingsworth and Melody (1995), Rosenthal et al. (1997) and Baker et al. (2002). These linear programs model aircraft movements by continuous variables rather than the integer variables with which we model ship movements. A continuous approximation of many, relatively small, cargo aircraft is probably appropriate, but such an approximation is inappropriate for fewer and much larger ships. Goggins (1995) developed a stochastic program with "simple recourse" (e.g., Ziemba 1974) to extend the deterministic optimization model that appeared in Rosenthal et al. (1997) to incorporate aircraft reliability. This entails modeling an airbase's capacity using an elastic constraint and paying a penalty whenever that capacity is violated in a future scenario. Granger et al. (2001) also study the effects of aircraft reliability, and apply queuing-network approximations, developed for manufacturing systems, to a simplified version of the deterministic model of Baker et al. Unlike SSDM, neither of these models allows dynamic re-routing. Mulvey et al. (1995) and Mulvey and Ruszczynski (1995) describe a two-stage stochastic program, called "STORM," that assigns aircraft to routes in the first stage and, after realizing random point-to-point cargo demands, assigns cargo to aircraft. In contrast to STORM, our scheduling paradigm does not require an *a priori* commitment to the vehicle routing schedule over the entire planning period. Powell (2005) describes techniques based on simulation and dynamic programming to handle uncertainty in airlift deployments.

Whiteman (1999) investigates a network-interdiction problem with uncertain interdiction effects using the following general approach: (a) He first solves the deterministic model, an integer program, using mean values for uncertain parameters, (b) he then investigates the solution for acceptability (sufficient reduction in expected network capacity) using Monte Carlo simulation, and (c) if it is unacceptable, he finds some near-optimal solutions to the deterministic problem and performs the same objective-function estimation procedure until an acceptable solution is found. The near-optimal solutions typically interdict more network components than does the original solution and are therefore, intuitively, more robust to some failed or partially successful interdictions. While the technique may lead to a good solution that satisfies a specified probabilistic criterion (e.g., expected capacity is reduced by at least 80%), there is no guarantee that the solution is near-optimal. When the underlying problem is convex (e.g., a linear program), convex combinations of candidate solutions are feasible and hence sometimes advocated. But again, there is no guarantee that such an approach will yield an optimal or even acceptable solution.

More generally, sensitivity analysis, parametric programming, and scenario analysis have often been suggested as ad hoc methods of handling uncertainty in mathematical programs (e.g., Ravindran et al. 1987, pp. 55–58), but these methods lack theoretical foundations and can lead to arbitrarily poor decisions (Wallace 2000). Stochastic programs, and particularly multi-stage stochastic programs, can be computationally expensive to solve. Undoubtedly, this fact has played a role in leading analysts to resort to the kind of approaches outlined above.

Two-stage, and multi-stage, stochastic programs of time-dynamic systems often model many time periods but fewer stochastic stages, again in order to maintain computational tractability. Stages are distinguished by the fact that they are separated by pre-specified, deterministic points in time at which (some of) the stochastic parameters become realized. So, in a two-stage stochastic program, the first- and second-stage decisions are made, respectively, before and after this pre-specified moment in time, and the second stage decisions are made in what is then a deterministic environment. Our SSDM departs from this paradigm in that the time at which the uncertain event occurs is random. Unlike a typical two-stage stochastic program, our first-stage represents a decision (deployment) plan for the entire planning period, but one that will be followed only until the randomly timed attack occurs, if it does occur. If no attack occurs, we implement the plan to the end of the time horizon. Otherwise, we make a second-stage re-deployment decision in what is then a deterministic environment because a second attack cannot occur.

We view SSDM as a model that falls between standard two-stage and multi-stage stochastic programs for multi-period problems. Mathematically, we can view it as a multi-stage stochastic program defined on a special type of scenario tree that has at most one uncertain event. More importantly, from the decision-making perspective, we view it as a two-stage stochastic program in which the timing of the stage is random. The size of a standard two-stage program with a fixed number of scenarios grows linearly in the number of time periods, while a multi-stage stochastic program grows exponentially with the number of time stages. As we describe in detail below, because SSDM has at most one uncertain event, it exhibits quadratic growth as a function of the number of time periods. Infanger (1994, pp. 43–47) describes a different class of multi-period problems in which capacity-expansion decisions with long lead-times result in a two-stage stochastic program, with these problems again growing only linearly in size with the number of time periods. In fact, restricting the solution space is commonly used to help create tractable, multi-stage stochastic optimization problems of various sorts. For example, optimizing over the class of time-stationary policies can help yield computationally tractable stochastic dynamic programming models (e.g., Bertsekas, 1987, Chap. 5). In a similar spirit, Mulvey et al. (2000) use nonlinear programming to search the class of "fixed-mix" investment policies in a multi-period asset-liability management model; see also Fleten et al. (2002) and Gaivoronski and de Lange (2000).

Outline of the Paper

We begin the remainder of the paper by first describing SSDM in general terms, and then in mathematical ones. We then describe our simulation of current, rule-based planning methods, which we use to compare to results from SSDM. We present computational results using data that represent a deployment similar in scope to that of the Desert Shield/Desert Storm deployment of 1991. The last section of the paper provides conclusions and points out areas for further development of SSDM.

THE STOCHASTIC SEALIFT DEPLOYMENT MODEL (SSDM)

Introduction

SSDM builds upon similar models formulated by Alexander (1999) and Loh (2000). Our model consists of four main entities, a ship-movement sub-model, a cargo-movement sub-model, linking constraints, and non-anticipativity constraints (Wets 1980). For simplicity, we begin by modeling a single type of ship, but later extend the model and computational tests to handle multiple types.

The ship-movement sub-model routes a ship from an SPOE where it is loaded, to an SPOD to be unloaded, and then back to an SPOE, not necessarily its origin. But, it also allows a ship to be re-routed from one SPOD to another in response to an attack, provided the ship has not entered a berth, but is waiting for one just outside the SPOD. Ships nominally require a fixed amount of time to unload their cargo, and they return to some SPOE immediately after unloading where they can be directly reloaded for another delivery, or wait until needed. If an attack occurs during unloading, the unloading period is extended by the decontamination period. For simplicity, ships become available for initial use according to a pre-specified schedule, once the deployment commences. (Most of these ships are civilian, converted to military use for military contingencies, according to established agreements.)

The cargo movement sub-model is similar to that for ship movement but incorporates separate constraints for each commodity called a "cargo package." For simplicity, we assume that each package is available at its assigned SPOE whenever the model decides to move it. This sub-model also adds an echelon of variables to move cargo from SPODs to final destinations. Typically, trucks or railcars, which are modeled through a single, generic transportation mode, would move this cargo. Side constraints control the movement of cargo out of the SPODs and reflect cargo-handling capacity of the port in various situations: Cargo-handling capacity reduces from its nominal value to zero immediately following an attack and during decontamination, and then ramps back up to its pre-attack level over a period of time. Because of permanent losses to personnel, post-attack cargo-handling capacity might never reach its pre-attack level, but this possibility is ignored for the sake of simplicity. (If the post-attack capacity is assumed known, the model can be trivially modified to handle this. If this capacity is uncertain, it adds scenarios to be considered.)

Linking constraints ensure that sufficient ship capacity is scheduled to carry the cargo being moved from SPOE to SPOD, being re-routed between SPODs and being moved from outside an SPOD into that port to be unloaded. Cargo is not assigned to a specific ship, so the combination of linking constraints and flow-based sub-models does imply a relaxation of real-world constraints: In effect, cargo can move between ships waiting outside an SPOD; however, this does not occur in any of our computational examples.

The model's variables and constraints are indexed by scenario, which encompasses the time and location(s) of the attack, or indicates that no attack occurs. A solution to SSDM is said to be *implementable* (Rockafellar and Wets 1991) if under any pair of scenarios a and a', with attack times $t_a \leq t_{a'}$, all decision variables are identical through time t_a-1. Non-anticipativity constraints ensure that this is the case.

The fact that we consider only a single attack substantially reduces the size of our stochastic program. Figure 1A shows a typical multi-stage scenario tree in which, after the first period, an attack may occur in any time period (at a given SPOD, say) during a four-period horizon and may occur any number of

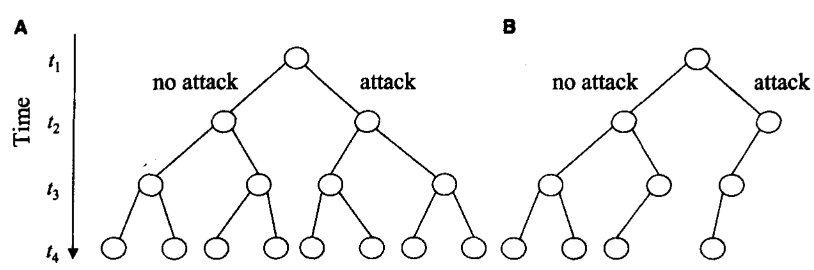

Figure 1. A standard scenario tree represents multiple attacks on a single SPOD (Figure 1A) and a specialized tree represents SSDM's assumption of at most one attack (Figure 1B).

times. For instance, the left-most leaf of the tree represents the "no-attack scenario" and the right-most leaf represents the "attack-in-every-time-period scenario." Figure 1B illustrates the scenario tree for SSDM assuming at most one attack will occur. The number of nodes and scenarios in the tree of Figure 1A grows exponentially with the number of stages while, in Figure 1B, the number of nodes grows quadratically and the number of scenarios grows linearly. To be more precise, let B denote the number of attack types and $t_{\max}$ be the number of time periods. Here, an attack type specifies which combination of SPODs are attacked and could be extended to include the severity of the attack. ($B=1$ and $t_{\max}=4$ in Figure 1B.) A scenario tree that allows repeated attacks has $(B+1)^{t_{\max}-1}$ scenarios and $((B+1)^{t_{\max}}-1)/B$ nodes while a scenario tree that allows an attack in at most one time period has only $(t_{\max}-1)B+1$ scenarios and $\frac{B}{2}t_{\max}^2-(\frac{B}{2}-1)t_{\max}$ nodes.

Of course, if the number of attack types B is too large, even the single-attack problem can become intractable. The data we analyze in this paper has two SPODs located in the Middle East. We allow a simultaneous attack on both SPODs or on either SPOD and not the other, so that $B=3$. We could also incorporate varying attack severity because of the weapons used or environmental factors, which would translate into longer or shorter decontamination periods and recovery schedules, and in doing so B would grow. For simplicity, this severity is fixed in our computations.

Mathematical Description of SSDM

This section introduces the SSDM formulation for a single ship type. We generalize to handle multiple types of ships in "Multiple Ship Types" within our "Computational Results" section.

Indices and index sets

$e \in E$ seaports of embarkation (SPOEs)

$d \in D$ seaports of debarkation (SPODs)

$f \in F$ final destinations (geographic locations where cargo is delivered)

$c \in C$ cargo packages, i.e., cargo moving from the same SPOE to the same final destination with identical available-to-load dates, and required delivery dates

$e(c)$ fixed, originating SPOE for cargo package c

$f(c)$ fixed, final destination for cargo package c

$t \in T$ time periods, $T=\{1,\ldots,t_{\max}+1\}$; $t_{\max}$ is the end of the time horizon and $t_{\max}+1$ is a fictitious time period

$T_{e(c)} \subseteq T$ allowable shipping periods for cargo c from SPOE $e(c)$ (depends on cargo availability dates, shipping delays and latest acceptable delivery date, which are defined below)

$a \in A$ attack scenarios. In addition to timing, the scenario contains the information on the SPOD or SPODs attacked, and could contain the post-attack decontamination time and recovery schedule. This set also includes the "no-attack scenario" denoted a_0

t_a attack time of scenario a ($1 < t_a \le t_{\max}$) for $a \ne a_0$; $t_{a_0} = t_{\max}+1$

$T_a \subseteq T$ time periods that run from the first period up to but not including the attack time for scenario a, i.e., $T_a=\{1,\ldots,t_a-1\}$

$T_{dta} \subseteq T$ Set of periods t' such that if a ship enters SPOD d at time t' then it will still occupy a berth there at time t under scenario a (depends on unloading time and any necessary decontamination)

$T^*_{da} \subseteq T$ time periods, if any, during which SPOD d remains contaminated under scenario a (computed using t_a defined above, and δ^U_{dta} defined below)

Data (units in parentheses)

δ^1_{ed} one-way travel time from SPOE e to just outside SPOD d (time periods)

δ^2_{de} one-way travel time from SPOD d to SPOE e (time periods)

$\delta^3_{dd'}$ one-way travel time from just outside SPOD d to just outside SPOD d' (time periods)

δ^F_{df} travel time from SPOD d to final destination f (time periods)

τ^{ALD}_c available-to-load date for cargo c. This is the earliest date (time period) the cargo is available at its SPOE for loading

τ^{CRD}_c required delivery date for cargo c at its final destination $f(c)$ (time period)

δ^{MAX} cargo delivered later than $\tau^{CRD}_c+\delta^{MAX}$ is deemed unmet demand for cargo c (time periods)

$\tau^{-}(c,d)$ earliest-possible-arrival date (time period) for cargo c at its destination $f(c)$ given that it travels through SPOD d; computed using τ_c^{ALD}, $\delta^1_{e(c),d}$ and $\delta^F_{d\,f(c)}$

$\tau^{+}(c)$ latest-possible-arrival date (time period) for cargo c at its destination $f(c)$; defined as $\tau_c^{CRD} + \delta^{MAX}$

$LPEN_{cdt}$ late-delivery penalty (penalty units/ton) for cargo c leaving SPOD d in period t. The penalty is based on the difference between actual and required delivery dates: $LPEN_{cdt} = [\max\{0, (t + \delta^F_{d,f(c)} - \tau_c^{CRD})\}]^{\alpha}$, with $\alpha > 0$; $\alpha > 1$ used in practice

$UPEN_c$ penalty for not delivering a required ton of cargo c within its required time window (penalty units/ton); $UPEN_c > \max_{d,t}\{LPEN_{cdt}\}$

ε_1 small penalty to discourage unnecessary ship voyages (penalty units/ships)

ε_2 small penalty to discourage unnecessary re-routing of ships (penalty units/ships)

$XTOT_c$ total amount of cargo c required (tons)

$VCAP$ capacity of a ship (sq. ft.)

r_c conversion rate from tons of cargo c to square feet (sq. ft./ton)

$VBTH_d$ berthing capacity at SPOD d (ships)

$VINV_{et}$ number of ships entering inventory at SPOE e at time t (ships)

δ^U_{dta} unloading time for a ship that enters SPOD d in period t under scenario a (time periods); includes decontamination time if an attack occurs during unloading. Since ships are not allowed to enter an SPOD during decontamination, this parameter is defined only for $t \in T - T^*_{da}$

$XCAP_{dta}$ capacity of SPOD d to handle cargo at time t under scenario a (tons/time period); the nominal capacity drops to zero after an attack and during decontamination, and progressively returns to its nominal or near-nominal level after decontamination

ϕ_a probability that scenario a occurs

Variables (under scenario a)

vi_{eta} number of ships in inventory at SPOE e at time t

vs_{edta} number of ships starting voyages from SPOE e to SPOD d at time t

vb_{dta} number of ships at waiting area outside SPOD d at time t

$vrr_{dd'ta}$ number of ships re-routed from SPOD d to SPOD $d' \neq d$ at time t

vh_{dta} number of ships entering berth at SPOD d at time t

vr_{deta} number of ships returning from SPOD d to SPOE e at time t

xs_{cdta} tons of cargo c shipped at time t from SPOE $e(c)$ to SPOD d

xb_{cdta} tons of cargo c at waiting area outside SPOD d at time t

$xrr_{cdd'ta}$ tons of cargo c re-routed from SPOD d to SPOD $d' \neq d$ at time t

xh_{cdta} tons of cargo c entering berth at SPOD d at time t

xi_{cdta} tons of cargo c in inventory at SPOD d at time t awaiting shipment to its final destination $f(c)$

xw_{cdta} tons of cargo c transported to its destination $f(c)$ from SPOD d at time t

xu_{ca} tons of unmet demand for cargo c

Formulation

$$\text{minimize} \sum_a \sum_c \sum_d \sum_t \phi_a LPEN_{cdt} xw_{cdta} + \sum_a \sum_c \phi_a UPEN_c xu_{ca} + \varepsilon_1 \sum_a \sum_e \sum_d \sum_t \phi_a vs_{edta} + \varepsilon_2 \sum_a \sum_d \sum_{d' \neq d} \sum_t \phi_a vrr_{dd'ta} \quad (1)$$

subject to:

$$-vi_{e,t-1,a} - \sum_d vr_{de,t-\delta^2_{de},a} + \sum_d vs_{edta} + vi_{eta} = VINV_{et} \quad \forall e,t,a \quad (2)$$

$$vh_{dta} - \sum_e vs_{ed,t-\delta^1_{ed},a} + vb_{dta} - vb_{d,t-1,a} + \sum_{d' \neq d} vrr_{dd'ta} - \sum_{d' \neq d} vrr_{d'd,t-\delta^3_{d'd},a} = 0 \quad \forall d,t,a \quad (3)$$

$$-vh_{dta} + \sum_e vr_{de,t+\delta^U_{dta},a} = 0 \quad \forall d,a,t \in T - T^*_{da} \quad (4)$$

$$\sum_{t' \in T_{dta}} vh_{dt'a} \leq VBTH_d \; \forall d,t,a \quad (5)$$

$$vh_{dta} \equiv 0 \; \forall d,a,t \in T^*_{da} \quad (6)$$

$$vr_{deta} \equiv 0 \, \forall \, d, a, t \in T^*_{da} \tag{7}$$

$$\sum_{d} \sum_{t \in T_{e(c)}} xs_{cdta} \leq XTOT_c \; \forall \, c, a \tag{8}$$

$$xh_{cdta} - xs_{cd,t-\delta^1_{e(c)d},a} + xb_{cdta} - xb_{cd,t-1,a} + \sum_{d' \neq d} xrr_{cdd'ta} - \sum_{d' \neq d} xrr_{cd'd,t-\delta^3_{d'd},a} = 0 \, \forall \; c, d, t, a \tag{9}$$

$$-xh_{cdta} + xi_{cd,t+\delta^U_{dta},a} - xi_{cd,t+\delta^U_{dta}-1,a} + xw_{cd,t+\delta^U_{dta},a} \leq 0 \; \forall c, d, t, a \tag{10}$$

$$\sum_{c} xw_{cdta} \leq XCAP_{dta} \; \forall \, d, t, a \tag{11}$$

$$-\sum_{d} \sum_{t=\tau^-(c,d)-\delta^F_{d,f(c)}}^{\tau^+(c)-\delta^F_{d,f(c)}} xw_{cdta} - xu_{ca} = -XTOT_c \; \forall c, a \tag{12}$$

$$\sum_{c|e(c)=e} r_c \, xs_{cdta} - VCAP \, vs_{edta} \leq 0 \quad \forall \, e, d, t, a \tag{13}$$

$$\sum_{c} r_c \, xrr_{cdd'ta} - VCAP \, vrr_{dd'ta} \leq 0 \quad \forall \, d, d' \neq d, t, a \tag{14}$$

$$\sum_{c} r_c \, xh_{cdta} - VCAP \, vh_{dta} \leq 0 \quad \forall d, t, a \tag{15}$$

All variables are non-anticipative, e.g.,

$$vi_{eta'} = vi_{eta} \; \forall \, e, a, a', t \in T_a \cap T_{a'} \tag{16}$$

All variables are non–negative (17)

Ship variables are integer:

$$vi_{eta}, vs_{edta}, vb_{dta}, vrr_{dd'ta}, vh_{dta}, vr_{deta} \tag{18}$$

Any variable with a time index not in T is fixed to 0, e.g.,

$$vr_{de,t-\delta^2_{de},a} \equiv 0 \; \forall d, e, t \leq \delta^2_{de}, a \tag{19}$$

Description of the Formulation

The basic premise of the model is to meet demands for cargo of various types during specified delivery time windows, although this will probably not be possible given limited system capacities, especially after attacks. This component of the objective function (1),

$$\sum_{c} \sum_{d} \sum_{t} LPEN_{cdt} xw_{cdta} + \sum_{c} UPEN_c xu_{ca}, \tag{20}$$

measures the *disruption* associated with scenario a. The first term corresponds to late deliveries with the per-ton penalty $LPEN_{cdt}$, which will increase as the function τ^α, where α is positive and τ is the number of periods the cargo is late. We use $\alpha > 1$ to express, roughly, "One ton of cargo late for t periods is worse than t tons of cargo late for one period." The second term in (20) strongly penalizes cargo not arriving during an acceptable delivery window: Such cargoes are absorbed as unmet demand with a penalty that exceeds the penalty for the latest acceptable delivery. Thus, ignoring the last two terms, the objective function measures the *total expected disruption* for a deployment plan. We note that large inventories of early-arriving cargo could be vulnerable to attack, but explicit penalties are not required to compensate. If early-arriving cargo is vulnerable, it ends up being delayed in one or more attack scenarios and is therefore "penalized" appropriately. The last two terms of the objective function are small factors to eliminate unnecessary ship movements.

The ship-movement sub-model is represented by constraints (2)–(7) and associated variables. The cargo-movement sub-model is represented by constraints (8)–(12) and associated variables. Constraints (13)–(15) link the two sub-models and constraints (16) account for non-anticipativity and ensure implementability of the decision variables with respect to the various scenarios. Note that, although these constraints are written for every pair (a, a'), it suffices to enforce them for appropriately defined pairs of "consecutive scenarios."

Of course, all variables are non-negative and the ship variables are required to be integer; see constraints (17) and (18), respectively. Additionally, variables with time indices outside of T do not represent true model entities and must be fixed to 0; see constraints (19).

Constraints (2) are ship-supply constraints for each SPOE. The supply of ships at time t includes those ships that become available via $VINV_{et}$ according to a pre-designated plan which does not depend on the scenario a; it includes those ships that have returned from earlier deliveries ($vr_{de,t-\delta^2_{de},a}$); and it includes those ships that have previously been put into "inventory" at the SPOE awaiting assignment ($vi_{e,t-1,a}$). The supply of ships is used to deliver cargo (vs_{edta}) or is held in inventory (vi_{eta}).

Constraints (3) are flow-balance constraints for the ships just outside the SPODs. A ship can arrive from an SPOE ($vs_{ed,t-\delta^1_{ed},a}$) or by being re-routed from another SPOD ($vrr_{d'd,t-\delta^3_{d'd},a}$). A ship that arrives can wait outside the SPOD for an available berth (vb_{dta}), it can enter the SPOD and berth (vh_{dta}), or it can be re-routed to an alternate SPOD ($vrr_{dd'ta}$).

Constraints (4) ensure that a ship entering an SPOD (vh_{dta}) does not leave and return to an SPOE ($vr_{de,t+\delta^u_{dta},a}$) until it has time δ^U_{dta} to unload, and decontaminate if necessary. Constraints (5) ensure that berthing capacity is not exceeded by the ships that have entered the SPOD. Constraints (6) and (7) ensure that no ships enter or leave a contaminated SPOD.

Constraints (8) are supply constraints for cargo; they are inequalities because, under certain scenarios, it may be determined that certain cargo cannot reach its destination within the allotted time window, and thus it will not be shipped at all. Constraints (9) balance flow of cargo just outside the SPODs, analogous to constraints (3) for ships.

Constraints (10) are inequality versions of flow-balance constraints for cargo inside the SPOD. Cargo that enters the SPOD at time t (xh_{cdta}) becomes available to enter inventory ($xi_{cd,t+\delta^u_{dta},a}$) or be shipped out to its final destination ($xw_{cd,t+\delta^u_{dta},a}$) after it has been unloaded, and possibly decontaminated, at time $t+\delta^U_{dta}$. Cargo in inventory from an earlier unloading is also available for shipment ($xi_{cd,t+\delta^u_{dta}-1,a}$). These constraints are inequalities because it is possible for cargo to arrive so late, or be trapped for decontamination inside the SPOD so long, that it cannot reach its final destination in time to be of any value. In effect, such cargo is unloaded at the SPOD and is subsequently ignored by the model.

Constraints (11) limit shipments of unloaded cargo out of the SPOD depending on that SPOD's cargo-handling capacity. The nominal capacity for each SPOD drops to zero immediately after an attack and during subsequent decontamination. The capacity then increases toward the nominal capacity during a post-decontamination period following some recovery schedule. Constraints (12) are simply the demand constraints for each cargo, with variables xu_{ca} absorbing unmet demand.

Constraints (13), (14) and (15) ensure that cargo is transported only if there is sufficient capacity on the ships that must move that cargo. These constraints cover cargo moving from SPOE to SPOD, from one SPOD to another, and from just outside an SPOD into its docks, respectively.

SIMULATING RULE-BASED PLANNING

Ideally, we would like to compare deployment plans developed through optimizing SSDM to plans developed through current rule-based planning methods. SSDM explicitly incorporates and evaluates the total expected disruption across all scenarios, but to evaluate rule-based planning we would have to perform the following steps for a given "test case," i.e., combination of data and probability distributions for when and where potential attacks might occur:

- Use rules to create a baseline deployment plan, "Plan0," under the no-attack scenario a_0.
- For each scenario $a \neq a_0$: Evaluate the cargo movements using Plan0 up to the time of the attack, simulate the disruption caused by the attack, and then plan the (re-) deployment of ships and cargo from the attack time onward using a rule-based procedure.
- Compute the total expected disruption using the disruption values computed above and the given probability distribution.

We cannot perform the above procedure exactly because no actual deployment-planning software is available to us. However, we can simulate that procedure by replacing rule-based plans with optimization-based plans. In particular, Plan0 is determined by solving a single-scenario variant of SSDM under the no-attack scenario—call this model DSDM(a_0). The redeployment is determined through another single-scenario model running from t_a through the end of the horizon given the simulated effects of an attack at time t_a—call this model DSDM($a|a_0$). The entire procedure, or "deterministic heuristic," is denoted DSDH.

In some of our test cases, attacks can only occur late in the time horizon and it seems that planners could probably take this information into account to make the rule-based deployment plan more robust against attack. In such cases, common sense dictates that we push the cargo through the system as quickly as possible so as to minimize the amount that is susceptible to attack in later time periods. We modify DSDM(a_0) to reflect this by adding a negative penalty (i.e., a benefit) into the objective function for early cargo arrivals. In particular, if one ton of cargo package c whose required delivery date is τ_c^{CRD} actually arrives in period $t < \tau_c^{CRD}$, it incurs a "penalty" of $-\beta(\tau_c^{CRD} - t)^{\alpha}$, for some $\beta > 0$, but if it arrives after τ_c^{CRD}, it incurs the usual penalty of $(t - \tau_c^{CRD})^{\alpha}$. Model DSDM($a_0$) with this modification is denoted DSDM$'$(a_0), and the overall deterministic heuristic that uses this initial model is denoted DSDH$'$.

We have found that a single small value of β will yield solutions to DSDM$'$(a_0) that are optimal with respect to the original objective of DSDM(a_0), but do push cargo through more quickly. Thus, there are multiple optimal solutions to DSDM(a_0) and we are taking advantage of that fact. In effect, we are solving the goal program that (a) optimizes one objective, i.e., it minimizes disruption, (b) adds a constraint that requires all solutions be optimal with respect to that objective value, and (c) then optimizes a secondary objective of pushing cargo through quickly.

By using DSDH$'$, we are trying to find an acceptable solution to our stochastic program from among multiple-optimal deterministic solutions, yet in the introduction we warned that this might be impossible. However, if this can be accomplished, we will have shown that current deterministic methods can be improved in the short term, and will have provided a stochastic-programming baseline for testing those improvements. We still argue that such heuristics should be replaced in the long term, and will substantiate this argument through computational results presented in the next section.

Complementing our rule-based planning methods, we devise a procedure, called D$^+$, to compute a lower bound z^+ on the optimal objective value of SSDM. The value z^+ is computed by solving, for each $a \in A$, DSDM(a), which is the deterministic version of SSDM with the effect of the scenario a attack incorporated. The expected disruption computed over all scenarios is z^+. This is an example of the well-known "wait-and-see bound" (e.g., Birge and Louveaux 1997, p. 138), because in each scenario the optimizing planner is assumed to know if, when and where an attack will occur.

COMPUTATIONAL RESULTS

This section describes computational results for SSDM, DSDH, DSDH$'$, and D$^+$. All computation is performed on a 2 GHz Pentium V processor with 2 Gb of RAM, running under Microsoft Windows XP. Models are generated using GAMS (Brooke et al. 1999) and solved using CPLEX Version 9.0 (ILOG 2006), with a 1% relative optimality tolerance.

Data

The data describe a hypothetical deployment to the Middle East requiring the movement of about 3,000 ktons of cargo, in 11 different packages, over the course of 100 days aggregated into 50 two-day time periods. The cargo is required between periods 7 and 45 of the deployment and the maximum-lateness parameter (δ^{MAX}) is 7 periods. There are four SPOEs and there are two SPODs, denoted d_1 and d_2, in close proximity to each other in the Middle East. For initial tests, we model only a single generic cargo ship, specifically, a Roll-On/Roll-Off (RoRo) vessel with capacity of 150,000 sq. ft. of cargo per trip. (Typically, floor space, rather than volume or weight, is the limiting factor for military sealift capacity.) This ship is typical of those used in planning exercises (Alexander 1999, Surface Deployment and Distribution Command 2002). The travel time between SPOEs and SPODs ranges from 3 to 12 periods. 158 ships become available at the four SPOEs to load cargo according to a pre-specified schedule over the course of the first 15 periods.

This hypothetical deployment is much like the one executed under Operation Desert Shield/Desert Storm in 1990 and 1991, although the time frame is compressed by about 50% to reflect more modern requirements (Alexander 1999), and the SPOEs are aggregated. The deployment represents the movement of five Army Divisions plus one Corps Support Command; one Marine Division plus two Expeditionary Brigades, one Air Wing and one Force Service Support Group; a few small units such as Navy Construction Battalions; and, pre-positioned afloat materiel, including ammunition. The data are derived from a number of unclassified sources including Army Field Manual 100-17-3 (Department of the Army 1999); the Deployment Planning Guide (Surface Deployment and Distribution Command 2001); the Army's Operational Logistics Planner (United States Combined Arms Support Command 1997); a paper on Naval Expeditionary Logistics (National Academy Press 1999); a paper on logistics requirements for Desert Shield/Desert Storm by Matthews and Holt (1992); the World Port Database (2006); and others.

In Desert Shield/Desert Storm, 95% of U.S. sealift cargo arrived at two SPODs in Saudi Arabia, Al Jubayl and Ad Dammam, which are situated about 100 km from each other on the Persian Gulf. Hence, our modeling of two proximate SPODs is realistic. Most of the Army and Marine Corps cargo originating in the continental U.S. came from Kentucky, North Carolina, Georgia, Florida, and Texas and left primarily via the ports of Newport News, VA, Wilmington, NC, Southport (Sunny Point), NC, Savannah, GA, Jacksonville, FL, Houston TX, and Beaumont, TX. Those ports handled over 80% of the cargo originating in the continental U.S.; personnel flew to their destinations. Two of our SPOEs represent aggregated versions of these ports. The third SPOE represents the U.S. military base of Diego Garcia in the Indian Ocean. "In DESERT SHIELD/STORM, the transportation time was minimized because of the foresight in prepositioning ordnance aboard ships in Diego Garcia" (Naval Historical Center 2006). The VIIth Army Corps, based in Germany, generated the bulk of the cargo originating outside of the continental U.S. and deployed primarily through these North Sea ports: Bremerhaven, Emden, and Nordenham Germany; Amsterdam and Rotterdam, The Netherlands; and Antwerp, Belgium. (These ports accounted for over 75% of all cargo originating outside of the continental U.S.) The fourth SPOE represents the aggregation of these ports, which we believe is reasonable given their geographical proximity. Matthews and Holt (1992) is the primary reference for the above discussion.

Under normal conditions, a ship is unloaded in two periods and the port has 150 ktons/period of cargo-handling capacity to forward that cargo to its final destination. Any attack on an SPOD, however, will close the port for a number of periods for decontamination, during which the cargo-handling capacity is lost entirely and the unloading process halts. Decontamination commences immediately after the attack and, upon completion, ships continue to unload at their standard rate. However, other cargo-handling capacity at the port only returns to normal gradually, according to a given recovery schedule. We consider a fixed decontamination period of seven periods with capacity recovering at a rate of 25% per period after decontamination.

The objective function of SSDM, equation (1), measures total expected disruption to the deployment (see previous section "Description of the Formulation"). Disruption resulting from late deliveries is measured in terms of "weighted kton-periods." Specifically, k ktons of cargo that are τ periods late incur a penalty of $k \times \tau^{1.5}$. Disruption resulting from an unmet delivery of k ktons of cargo from package c is $k \times \tau_c^{1.5}$, where τ_c is a strict upper bound on the number of periods late that package c is still considered worth delivering.

Test Cases and Results

In the following, we analyze the benefits of the stochastic solution using combinations of attack types and probability distributions, which we call test cases. In practice, analysts would develop these from intelligence reports. The attack types are:

$\mathcal{D}^{1,2} = \{\{d_1\}, \{d_2\}\}$: An attack occurs at SPOD d_1 or at SPOD d_2, but not both, or no attack occurs;

$\mathcal{D}^{1,12} = \{\{d_1\}, \{d_1, d_2\}\}$: Mutually exclusively, an attack occurs at d_1, both SPODs are attacked simultaneously, or no attack occurs; and

$\mathcal{D}^{1,2,12} = \{\{d_1\}, \{d_2\}, \{d_1, d_2\}\}$: Mutually exclusively, d_1 is attacked, d_2 is attacked, both d_1 and d_2 are attacked simultaneously, or no attack occurs.

The probability distributions for the test cases are defined by (a) the probability of no attack, $\phi_{a_0} = 0.5$, (b) by the assumption that in any given period, an attack of any element of, e.g., $D^{1,2,12}$, is equally likely, and (c) the following conditional distributions for the timing of an attack:

U1: Uniform distribution over periods 4 through 40,
T1: Triangular distribution over periods 4 through 40 with mode 22,
U2: Uniform distribution over periods 4 through 18,
T2: Triangular distribution over periods 4 through 18 with mode 11,
U3: Uniform distribution over periods 26 through 40, and
T3: Triangular distribution over periods 26 through 40 with mode 33.

The first distribution, U1, is the "baseline distribution" accounting for almost no information. T1 employs the same range of periods but gives more likelihood to attacks occurring in the middle of the deployment. U2 and T2 represent a situation in which planners believe the enemy will strike early in the deployment: Perhaps the enemy believes, and our intelligence suggests, that early strikes will have a strong psychological effect against us and compound our scheduling problems in a way that a later strike would not. U3 and T3 represent the anticipation of late strikes: Perhaps intelligence reports indicate that the enemy will experience some delay in deploying his biological weapons, or may only want to use them as a last resort.

Table 1 describes the set of test cases and associated model statistics. Table 2 displays the computational results for the test cases under the various models and solution procedures: SSDM, DSDH and DSDH′, as well as the lower-bounding procedure D^+.

The SSDM column of Table 2 reports the total expected disruption. (More precisely it reports the optimal objective function value of SSDM, including the terms ϵ_1 and ϵ_2, but these additional terms make up at most 0.1% of the reported value.) For the other models, we report the percent deviation with respect to the SSDM solution. Specifically, Table 2 shows 100 × (z(*model*) − z(SSDM))/*bound*, where z(*model*) represents the objective function value of *model*, and *bound* equals z(D^+) when reporting D^+, but equals z(SSDM) when reporting DSDH or DSDH′. In summary, the results show that:

1. SSDM reduces total expected disruption over the simulated rule-based planning of DSDH by an average of 22% with a range of 1% to 47%. With respect to the improved heuristic DSDH′, SSDM reduces expected disruption by an average of 8%, with a range of 1% to 14%.
2. Even though DSDH′ was intended to improve results with late attacks (U3 and T3), it also performs well relative to DSDH when the attack can occur over the widest range of periods (U1 and T1). Under early attacks (U2 and T2), DSDH outperforms DSDH′, but all such differences are small, i.e., at most 0.3%; thus, we only discuss DSDH′ in what follows.
3. The lower bound provided by D^+ is below the near-optimal solution value of SSDM by an average of 29%. This indicates that even if DSDH′ does provide a good solution, we still must solve SSDM to verify its quality.
4. Early-attack cases leave the least flexibility for SSDM to improve upon rule-based planning. The average reduction in disruption of SSDM over DSDH′ is 3%, with a range of 1–6% for the six U2 and T2 cases.
5. Conversely, the stochastic program has the greatest leverage when attacks can only occur late in the deployment. The analogous average and range for the six U3 and T3 cases are 12% and 8–14%. Finally, the range and average for the U1 and T1 cases are 8% and 5–11%.

Table 1. Problem definitions and sizes for the stochastic sealift deployment model SSDM and its deterministic counterpart DSDM(a_0), which assumes no attack occurs. Each test case is described by: (a) The subsets of SPODs where the attacks may occur (e.g., $\{\{d_1\},\{d_1, d_2\}\}$ represents a case where either d_1, or d_1 and d_2, may be attacked, but not d_2 alone), and (b) the conditional probability distribution for the time of attack, given that an attack occurs. The number of scenarios $|A|$ is also shown. Problem sizes are given in terms of numbers of structural constraints m, continuous variables n_1 and integer variables in n_2.

			Problem Sizes					
			SSDM			DSDM(a_0)		
Attack types	Distributions	$\|A\|$	m	n_1	n_2	m	n_1	n_2
$\{d_1\},\{d_2\}$	U1, T1	75	605,346	672,399	44,260	2,691	5,254	596
	U2, T2	31	165,170	229,013	18,300			
	U3, T3		321,410	320,965				
$\{d_1\},\{d_1,d_2\}$	U1, T1	75	605,346	672,399	43,890			
	U2, T2	31	165,170	229,013	18,150			
	U3, T3		321,410	320,965				
$\{d_1\},\{d_1\},\{d_1,d_2\}$	U1, T1	112	906,674	1,003,004	65,720			
	U2, T2	46	264,410	339,270	27,000			
	U3, T3		480,770	475,908				

We explain the general dominance of DSDH′ over DSDH (result 2 above) as follows: Even if attacks can occur at any time, it makes sense to push cargo through the system as quickly as possible because, if an attack occurs, delayed cargo has as much slack time as

Table 2. Results for SSDM in ktons × days$^{1.5}$ of total expected disruption, and for related models in percent deviation with respect to SSDM. All models are solved with a 1% optimality gap. Overall, stochastic planning with SSDM reduces disruption significantly over basic rule-based planning with DSDH and the improved deterministic heuristic DSDH′. DSDH′ leads to smaller disruption than DSDH except in the "early attack" cases of U2 and T2. Average run times for D$^+$, SSDM, DSDH and DSDH′ are 89, 280, 48, and 45 seconds, respectively.

			Objective-function values			
Attack types	Distribution	$\|A\|$	D$^+$	SSDM	DSDH	DSDH′
$\{d_1\}$, $\{d_2\}$	U1	75	−52%	4,648	22%	5%
	T1	75	−44%	4,896	13%	5%
	U2	31	−67%	4,426	1%	1%
	T2	31	−80%	4,755	1%	1%
	U3	31	−49%	4,491	47%	14%
	T3	31	−49%	4,386	41%	8%
$\{d_1\},\{d_1,d_2\}$	U1	75	−11%	12,118	23%	10%
	T1	75	−7%	13,022	18%	11%
	U2	31	−24%	9,877	6%	6%
	T2	31	−22%	11,189	4%	5%
	U3	31	−4%	12,768	44%	13%
	T3	31	−1%	12,507	43%	13%
$\{d_1\}$, $\{d_2\}$, $\{d_1,d_2\}$,	U1	112	−18%	9,750	23%	7%
	T1	112	−11%	10,483	18%	10%
	U2	46	−33%	8,301	4%	4%
	T2	46	−31%	8,979	2%	2%
	U3	46	−9%	10,084	42%	12%
	T3	46	−7%	9,894	40%	10%

possible to reach its destination. However, pushing cargo quickly is a "double-edged sword": The higher disruption for DSDH′ compared to SSDM is largely explained by the fact that DSDH′ moves cargo too fast to the SPODs in a few scenarios, and a large quantity becomes trapped in the attacked SPOD(s) in those scenarios and cannot reach its destination in time. SSDM better balances the speedy arrival of cargo against the flexibility to reroute cargo waiting outside an SPOD that may be attacked.

Table 3 helps investigate the behavior of the various procedures under likely and especially disruptive scenarios. Since the no-attack scenario is likely to occur, we want to know if the solutions from the stochastic model are robust in this scenario. The table shows that they are. The table also shows that the worst-case scenarios for the stochastic model are usually less disruptive than the worst-case scenarios for rule-based planning.

The results above indicate that the stochastic-programming approach may yield substantial improvements over rule-based planning. Indeed, because we have simulated rule-based planning with a "tuned," deterministic optimization model—that optimization model is likely to give better results than any rule-based heuristic—the actual improvements may be even greater than those demonstrated.

It is important to understand when the need for using a model such as SSDM is most acute. As described above, when SSDM has the most time prior to the attack to hedge (i.e., the U3 and T3 distributions), the value of the stochastic solution is the largest. We test this sensitivity with respect to the distribution governing the time of attack by considering a

Table 3. Results for special scenarios for SSDM solutions and solutions of DSDH′, all in ktons × days$^{1.5}$ of disruption. Legend is as follows: No-attack scenario: Disruption (objective function value) for the stochastic planning method when no attack occurs, and percent improvement when we plan assuming no attack will occur. Worst-case scenarios: Worst disruption, across all scenarios, for the given method, with the DSDH′ percentage again being relative to SSDM. Columns under "No-attack scenario" show that only a small penalty is paid for hedging against potential attacks when none occurs. "Worst-case scenarios" show that the worst disruption observed with the stochastic model is usually better than the worst disruption observed under rule-based planning (DSDH′).

			Objective-function values			
			No-attack scenario		Worst-case scenarios	
Attack types	Distribution	$\|A\|$	DSDH′	SSDM	DSDH′	SSDM
$\{d_1\},\{d_2\}$	U1	75	−9%	2,154	20%	14,468
	T1	75	−2%	2,004	20%	14,468
	U2	31	0%	1,974	0%	14,060
	T2	31	−2%	2,017	0%	14,102
	U3	31	−2%	2,004	11%	15,767
	T3	31	−3%	2,034	11%	15,767
$\{d_1\},\{d_1,d_2\}$	U1	75	−3%	2,034	14%	60,253
	T1	75	−6%	2,089	14%	60,253
	U2	31	−2%	2,017	3%	46,212
	T2	31	−11%	2,185	3%	46,212
	U3	31	−5%	2,077	16%	54,803
	T3	31	−2%	2,004	24%	58,701
$\{d_1\},\{d_2\},\{d_1,d_2\}$	U1	112	−4%	2,049	14%	60,253
	T1	112	−4%	2,049	14%	60,253
	U2	46	−2%	2,017	3%	46,212
	T2	46	−2%	2,017	3%	46,212
	U3	46	−5%	2,064	16%	54,803
	T3	46	−2%	2,004	24%	58,701

triangular distribution, denoted T3′, in which we change the mode of the triangular distribution from 33 to 40. The associated computational results for each of the three attack types are shown in Table 4. The expected disruption of SSDM's solution is better than that of simulated rule-based planning by 25%, 14%, and 12% for the three types of attack. (These values have increased from 8%, 13%, and 10%, respectively, for the original T3 distribution.) These results are consistent with the trend in Table 3 of SSDM solutions becoming more valuable as we move from early attack times, to the widest range of attack times, to late attack times.

Multiple Ship Types

SSDM can be extended to accommodate multiple ship classes, with different capacities and speeds. This lends more realism to results, but does tax our computational machinery. Nonetheless, we can still show substantial improvements over deterministic alternatives. The modifications to the model presented in the previous "Mathematical Description of SSDM" section follow:

Additional or modified sets and data:

$s \in S$	Set of ship types, e.g., {FSS, RoRo, LMSR}
$T_{se(c)}$	Allowable shipping periods for cargo c from SPOE $e(c)$ using ship s
T_{sdta}	Set of periods t' such that if ship s enters SPOD d at time t' then it will still occupy a berth there at time t under scenario a
δ^1_{sed}	One-way travel time for ship s from SPOE e to just outside SPOD d (time periods)
δ^2_{sde}	One-way travel time for ship s from SPOD d to SPOE e (time periods)
$\delta^3_{sdd'}$	One-way travel time for ship s from just outside SPOD d to just outside SPOD d' (time periods)
δ^U_{sdta}	Unloading time for ship s that enters SPOD d in period t under scenario a (time periods)
$VCAP_s$	Capacity of ship type s (sq. ft.)
$VINV_{set}$	Number of ships of type s entering inventory at SPOE e at time t (ships)

Variables:

We add the index s to variables to obtain: vi_{seta}, vs_{sedta}, vb_{sdta}, $vrr_{sdd'ta}$, vh_{sdta}, vr_{sdeta}, xs_{scdta}, xb_{scdta}, $xrr_{scdd'ta}$, xh_{scdta}. Each variable retains its original meaning with the added qualifier "for ship type s." The only variables that do not require an additional index are xi_{cdta}, xw_{cdta} and xu_{ca}.

According to these modifications, the objective function (1) and constraints (2)–(10), (13)–(16) and (18)–(19) should be generalized to accommodate ship types. For example, constraints (2) become

$$-vi_{se,t-1,a} - \sum_d vr_{sde,t-\delta^2_{sde},a} + \sum_d vs_{sedta} + vi_{seta} = VINV_{set} \forall s,e,t,a. \qquad (2')$$

We consider three ship types in the Department of Defense's surge sealift fleet (Congressional Budget Office 1997): Fast Sealift Ship (FSS), Large Medium-Speed Roll-on/Roll-off Ship (LMSR), and the same RoRo container ship used in the initial tests. (More ship types, such as Breakbulks or Special-purpose ships, may be

Table 4. Results for the three types of attacks under a modified triangular distribution for time of attack. The new distribution for the time of attack, T3′, results from changing the mode of the original triangular attack-time distribution T3 from period 33 to period 40. As before, the probability of no attack is 1/2. Average run times for D$^+$, SSDM, and DSDH′ are 41, 122 and 12 seconds, respectively.

Test Case	$\lvert A \rvert$	Objective Function Values		
		D$^+$	SSDM	DSDH′
$\{\{d_1\},\{d_2\}\}$-T3′	31	−41%	3,682	25%
$\{\{d_1\},\{d_1,d_2\}\}$- T3′	31	−6%	10,989	14%
$\{\{d_1\},\{d_2\},\{d_1,d_2\}\}$- T3′	46	−13%	8,780	12%

added to our model, as needed, at an additional computational cost.) While capacity and speed may vary for vessels within the same ship class (Military Sealift Command Ship Inventory 2006), we assume those to be the same for this testing; see Table 5.

To allow us to compare with the single-ship case, port capacities remain the same, as does the total square footage of ship capacity in the fleet. In particular, 158 RoRo ships × 150,000 sq. ft./ship = 23,700,000 sq. ft. converts to 7 FSSs, 116 RoRos and 14 LMSRs. Fewer vessels are needed due to the higher capacity of the FSSs and LMSRs. Remark: We can easily model the fact that FSSs and LMSRs require more berthing space than RoRos, by modifying constraint (5) as follows:

$$\sum_{s \in S} \sum_{t' \in T_{sdta}} SIZE_s \, vh_{dt'a} \leq SIZEVBTH_d \; \forall \, d, t, a, \tag{5'}$$

where $SIZE_s$ is the size of ship type s, and $SIZEVBTH_d$ is the berthing capacity at SPOD d. For simplicity, we maintain the original constraint (5) in our testing, i.e., port capacity is limited by the number of ships, rather than by their aggregate size.

As shown in Table 5, initial lift capacity as ships become available (by period, at each SPOE) also remains the same. Even with fixed ship capacity, we expect a solution with less expected unmet demand, because an intelligent plan can exploit the higher speeds of FSSs and LMSRs to deliver more cargo during the planning period.

Test case $\mathcal{D}^{1,2} = \{\{d_1\}, \{d_2\}\}$ with uniform distribution U3 corresponds to one of the single-ship cases from Table 2 where the stochastic solution most strongly outperforms that obtained using heuristics. To reduce the size of the associated multiple-ship model, we test a variant of this problem in which we modify the scenario tree by modeling attacks only on even-numbered periods from periods 26 to 40. The resulting scenario tree has 17 leaf nodes instead of 31, including one no-attack scenario. Before turning to the multiple-ship version of this model, we note that the single-ship models under these two scenario trees render stochastic solutions within 1% of each other, and similar bounds are also found by the heuristics (see Table 6), justifying the use of the smaller scenario tree for the multiple-ship problem: The stochastic model still has over 445,000 constraints and 466,000 variables, including 30,000 discrete variables.

The utility of a tractable, reduced-size, scenario tree is further corroborated by our "out-of-sample" assessments from Table 6 (last two rows). Here, the SSDM solution, which is based on 17 scenarios, is tested against two cases with 31 and 91 scenarios, respectively. The 31-scenario problem allows attacks on every

Table 5. Ship data: Number of sealift ships under the control of the U.S. Transportation Command and their readiness status. For example, four LMSR (large, medium-speed, roll-on/roll-off) sealift ships become available in period two (i.e., on day three or four) of the planning time at some of the four SPOEs (the specific breakdown by SPOE is not shown for simplicity). Characteristics: Days Port Procedure (DPP) needed to load/unload; Capacity (sq. ft.); Speed (knots). Travel times (days) from each SPOE (W, E in the U.S., F1 at Diego Garcia and F2 in Europe) to either SPOD. In comparison with the original case (only RoRo ships) there are fewer vessels in the multiple-ship case, but initial lift capacity is maintained for every period and even for every SPOE (not shown). For example, in period 1, twelve of the 58 RoRos from the original case are replaced by two FSSs and four LMSRs in the multiple-ship case.

	Total number of ships in readiness status (period ready)							Characteristics			Travel time from each SPOE			
Ship type	1	2	3	5	10	18	Total	DPP	Cap	Sp	W	E	F1	F2
SL-7 FSS	2	2	2	1	0	0	7	4	200,000	27	16	11	3	8
RoRo	46	24	16	6	11	13	116	4	150,000	18	24	17	5	12
LMSR	4	4	4	2	0	0	14	4	350,000	24	18	13	4	9
Total	52	30	22	9	11	13	137		(23,700,000)					
Original case (only RoRo)	58	36	28	12	11	13	158	4	150,000	18	24	17	5	12

Table 6. The first case is from Table 2: We use RoRo ships only, with attack type $\mathcal{D}^{1,2} = \{\{d_1\}, \{d_2\}\}$, and discrete uniform distribution "U3" over periods 26, 27,...,40 (in addition to a no-attack scenario with probability 0.5). The second case is a simplification of the first, where attacks are allowed only on even-numbered periods (26, 28,...,40). These SSDM solutions are within 1% of each other. The third case uses the same simplified scenario tree to incorporate multiple ship types, showing a significant improvement for the stochastic solution, and a substantial degradation in the relative quality of the heuristic solutions provided by DSDH and DSDH′. This multi-ship SSDM solves in approximately 800 seconds within 1% of optimality. The fourth case has all 26, 27,...,40 attack days. In the interest of tractability, the SSDM method shows the SSDM 17-scenario solution evaluated against the 31 scenarios of this case. Even though we have left out some attack days to build our SSDM solution, it still outperforms those provided by heuristics. The last case replaces the deterministic decontamination time of 10 periods by three equally-likely scenarios of 8, 10 and 12 periods. Note: This is reflected in our δ^U_{dta} (unloading time) parameter. Again, even though the SSDM solution is based on the 17-scenario case, it retains its significant improvement with respect to those from the heuristics.

	Objective function values			
Test case	D⁺	SSDM	DSDH	DSDH′
Single-ship (only RoRo). 31 scenarios	−49%	4,491	47%	14%
Single-ship (only RoRo). 17 scenarios	−49%	4,539	56%	11%
FSS, RoRo and LMSR. 17 scenarios	−148%	1,402	273%	61%
FSS, RoRo and LMSR. 31 scenarios	−163%	1,473*	256%	57%
FSS, RoRo and LMSR. 91 scenarios	−196%	1,657*	210%	52%

(*) Out-of-sample assessment using the solution from SSDM 17-scenario case. 1,375 is the optimal value for 31 scenarios, but this takes too long to compute, so we compare to the 17-scenario case, which has an optimal value of 1,473.

period from periods 26 through 40; the 91-scenario problem expands that problem by incorporating three equally-likely decontamination scenarios of 8, 10 and 12 periods. In both cases, the SSDM solution substantially outperforms the solutions obtained using the heuristics, DSDH and DSDH′. While a 1%-gap solution to the 17-scenario SSDM can be achieved in approximately 800 seconds, a solution of similar quality to the 31-scenario SSDM takes several hours. If several hours is too long, the solution to the 17-scenario problem might be acceptable. For example, the 1%-gap solution to the 31-scenario SSDM has a value of 1,375, while the value realized by the 17-scenario problem applied to the 31-scenario out-of-sample tree is 1,473. But, 1,473 still indicates a substantial improvement over the solutions provided by the heuristics.

From Table 6 (row 3), the single-ship SSDM yields a solution with objective value 4,539, whereas the DSDH and DSDH′ heuristics yield solutions approximately 56% and 11% worse, respectively. For the multiple-ship case, our hypothesis that unmet cargo would be substantially reduced is confirmed: The new SSDM solution improves from 4,539 to 1,402 ktons× days$^{1.5}$, or 224%.

Relatively speaking, the heuristics DSDH and DSDH′ perform worse when solving multiple-ship instances than for single-ship cases. Specifically, the DSDH and DSDH′ solutions are 273% and 61% worse than for SSDM, respectively. (The analogous percentages for out-of-sample testing are smaller but not dramatically different.) This behavior can be attributed to two facts: First, planning for the no-attack scenario is even more inefficient now, because ships trapped at an attacked SPOD cannot be easily "replaced" now that fewer total ships are available. And, since new FSSs and LMSRs have higher capacity, losing one of these to an attack is a more important setback. Second, the incentive for pushing cargo through early, which motivates DSDH', does not yield results that are as good as for the RoRo-only scenarios because FSSs and LMSRs are faster vessels.

In summary, our results suggest that even a small fleet of FSSs and LMSRs replacing a capacity-equivalent number of RoRos would provide substantial planning flexibility and less expected disruption. But, this benefit can be fully realized only if stochastic programming is used to guide planning.

CONCLUSIONS

This paper has devised a specialized, stochastic mixed-integer programming model for planning the delivery of sealift cargo in a wartime deployment, subject to possible enemy attacks on one or more seaports of debarkation (SPODs). The attacks are simulated by halting berthing, and cargo handling, until decontamination is complete at an attacked SPOD, and by then modeling cargo-handling capacity that gradually returns to normal. The timing and location of an attack are uncertain. We focus on the effects of a single biological attack (possibly on two SPODs simultaneously), but the model could be modified for multiple attacks or conventional, nuclear or chemical attacks.

The stochastic program SSDM and two (simulated) rule-based planning schemes have been tested with data from a realistic wartime deployment with 158 ships becoming available at different times during the deployment, 11 cargo types, four seaports of embarkation where cargo is loaded and two SPODs where cargo is unloaded before reaching its final destination. Our test cases assume there is a substantial probability of no attack, but if an attack does occur, it occurs with known probability distribution for timing and location.

Expected cargo lateness, measured in weighted ton-periods, improves by up to 25% for a single-ship-type deployment, but up to 60% with multiple ship types, depending on data and probabilistic assumptions. These improvements are relative to expected results obtained using a simulation of current, rule-based planning techniques. (In fact, we compare against a "tuned" rule-based technique that is rooted in a deterministic optimization model and may overestimate the efficacy of rule-based techniques; hence, our comparisons are conservative.) However, there is little price to pay in terms of cargo lateness for the stochastic solution if no attack occurs. In conclusion, hedging against a possible attack can provide substantial benefits if an attack occurs, and incurs only a minor disruption penalty otherwise.

Our simulations of current rule-based plans have shown that it may be possible to establish rules that are more robust against potential attacks early in the deployment horizon, without using a special stochastic-programming model. In these cases, SSDM improves over rule-based planning, designed to push cargo through the system as quickly as possible, by only an average of 3%. This contrasts, however, with larger averages of 8% when an attack may occur at almost any time (uniform distribution for attack time), and 11% and 17% for two sets distributions that model attacks that can only occur late in the deployment. Except in the case of early attack, it may be impossible to adjust a rule-based system to behave nearly as well as a stochastic one. But it is probably impossible to know how well rule-based planning is performing without an optimal, stochastic solution to serve as a baseline lower bound. So, while computationally cheaper rule-based systems can be improved, stochastic programming provides a superior approach with respect to provable solution quality.

The current emphasis in the U.S. military's deployment planning is for providing up-to-the-minute tracking of all cargo and transportation assets, with the ability to quickly respond to contingencies. This is no doubt important, but planners can expect more timely arrival of cargo into a theater of war if they proactively plan for those contingencies.

Further testing and development of SSDM are warranted. The model should be tested against wartime deployment situations in other parts of the world. Conceptually, the model is easy to expand for other sources of uncertainty such as the location of the cargo-carrying ships

Milita

when deployment planning is commenced. SSDM currently assumes that if an attack occurs during the deployment, there will only be one, although it may affect more than one SPOD. We argued that SSDM may be viewed as a two-stage stochastic program in which the stage occurs at a random time, and that the size of the model grows quadratically with the number of time periods. This idea could be extended to allow at most two attacks, i.e., a three-stage multi-period model with random stage timing. The size of that model would grow with the cube of the number of time periods. As the models grow larger in this manner, their solution will require the development of specialized algorithms.

ACKNOWLEDGEMENTS

The authors would like to thank the students at the Naval Postgraduate School whose research has provided the foundation for our work: Tammy Glaser, Steven Aviles, Michael Theres, Christopher Alexander and Loh Long Piao. David Morton's research was supported by the National Science Foundation through grant DMI-0217927 and by the State of Texas Advanced Research Program through grant #003658-0405-1999. Javier Salmerón's research was supported by a National Research Council Postdoctoral Fellowship. Kevin Wood's research was supported by the Office of Naval Research and the Air Force Office of Scientific Research. The authors also thank two anonymous referees for suggestions that have improved the paper.

REFERENCES

Alexander, C., 1999. "Effects of Biological Warfare on Throughput at Seaports of Debarkation during a Major Theater War," M.S. Thesis, Operations Research Dept., Naval Postgraduate School, Monterey, CA, September.

Army Logistician, 2001. "Defense Agency Studies Vulnerabilities of Seaports to Attack," July-August.

Aviles, S., 1995. "Scheduling Army Deployments to Two Nearly Simultaneous Major Regional Conflicts," M.S. Thesis, Operations Research Dept., Naval Postgraduate School, Monterey, CA, September.

Baker, S.F., Morton, D.P., Rosenthal, R.E. and Williams, L.M., 2002. "Optimizing Military Airlift," *Operations Research*, **50**, pp. 582–602.

Barnaby, W., 1999. *The Plague Makers: The Secret World of Biological Warfare*, Vision Paperbacks, London.

Bertsekas, D.P., 1987. *Dynamic Programming: Deterministic and Stochastic Models*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Birge, R. and Louveaux, F., 1997. *Introduction to Stochastic Programming*, Springer-Verlag, New York.

Brooke, A., Kendrick, D. and Meeraus, A., 1999. *GAMS, A User's Guide*, GAMS Development Corporation, Washington, DC.

Brown, G.G., Dell, R.F., and Wood, R.K., 1997. "Optimization and Persistence," *Interfaces*, **27**, pp. 15-37.

Brown, W., 1999. "Re-optimization of Time-Phased Force Deployment Plans in Response to Emergent Changes during Deployment," M.S. Thesis, Operations Research Dept., Naval Postgraduate School, Monterey, CA, September.

Cohen, W., 1997. Report of the Quadrennial Defense Review, May.

Congressional Budget Office, 1997. "Moving U.S. Forces: Options for Strategic Mobility". http://www.cbo.gov/showdoc.cfm?index=11&sequence=4 (accessed October 2006).

Defense Intelligence Agency, 1998. "The Worldwide Biological Warfare Threat, A Global Perspective," 17 March.

Department of Defense, 1997. "Summary Report: Assessment of the Impact of Chemical and Biological Weapons on Joint Operations in 2010," October.

Department of the Army, 1999. "Reception, Staging, Onward Movement, and Integration," Field Manual 100-17-3, Headquarters, Department of the Army, Washington, DC, 17 March.

Ferguson, A. and Dantzig, G.B., 1956. "The Allocation of Aircraft to Routes—An Example of Linear Programming under Uncertain

Demands," *Management Science*, **3**, pp. 45–73.

Fleten, S.-E., Hoyland, K., Wallace, S.W., 2002. "The Performance of Stochastic Dynamic and Fixed Mix Portfolio Models," *European Journal of Operational Research*, **140**, pp. 37–49.

Gaivoronski, A.A., and de Lange, P.E., 2000. "An Asset Liability Management Model for Casualty Insurers: Complexity Reduction vs. Parameterized Decision Rules," *Annals of Operations Research*, **99**, pp. 227–250.

Goggins, D., 1995. "Stochastic Modeling for Airlift Mobility," M.S. Thesis, Operations Research Dept., Naval Postgraduate School, Monterey, CA, September.

Granger, J., Krishnamurthy, A. and Robinson, S.M., 2001. "Stochastic Modeling of Airlift Operations," *Proceedings of the 33rd Conference on Winter Simulation*, pp. 432–440.

ILOG, 2006. *CPLEX 9.0 Reference Manual*, ILOG, S.A., France. http://www.ilog.com (accessed October 2006).

Infanger, G., 1994. *Planning Under Uncertainty: Solving Large-Scale Stochastic Linear Programs*, The Scientific Press Series, Boyd & Frasier, Danvers, Massachusetts.

Joint Publication 3-11, 2000. "Joint Doctrine for Operations in Nuclear, Biological, and Chemical (NBC) Environments," 11 July.

Koprowski, P., 2005. "Interdicting a Force Deployment: Two-sided Optimization of Asset Selection, Lift Scheduling and Multi-commodity Load Planning," M.S. Thesis, Operations Research Dept., Naval Postgraduate School, Monterey, CA, March.

Killingsworth, P. and Melody, L.J., 1995. "Should C-17s Be Used to Carry In-Theater Cargo During Major Deployments?" DB-171-AF/OSD, RAND Corporation, Santa Monica, CA.

Larson, E.V. and Peters, J.E., 2001. "Preparing the U.S. Army for Homeland Security: Concepts, Issues, and Options," Rand Corporation, Santa Monica, CA.

Leonelli, J. and Althouse, M.L., 1998. "Air Monitoring and Detection of Chemical and Biological Agents," *Proceedings of SPIE*, Volume 3533, Boston, 2–3 November.

Loh, L.P., 2000. "Deterministic and Stochastic Models of Biological Attacks on Seaports of Debarkation during A Major Theater War," M.S. Thesis, Operations Research Dept., Naval Postgraduate School, Monterey, CA, September.

Lund, J., Berg, R. and Replogle, C., 1993. "Project AIR FORCE Analysis of the Air War in the Gulf: An Assessment of Strategic Airlift Operational Efficiency," Report R-4269/4-AF, RAND Corporation, Santa Monica, CA.

Matthews, J.K. and Holt, C.J., 1992. "So Many, So Much, So Far, So Fast: United States Transportation Command and Strategic Deployment for Operation Desert Shield/Desert Storm," Office of the Chairman of the Joint Chiefs of Staff, United States Transportation Command, Washington, DC. 7

Midler, J.L. and Wollmer, R.D., 1969. "Stochastic Programming Models for Airlift Operations," *Naval Research Logistics Quarterly*, **16**, pp. 315–330.

Military Sealift Command Inventory, 2006. "Sealift Ships". http://www.msc.navy.mil/inventory/inventory.asp?var=PM5 (accessed October 2006).

Surface Deployment and Distribution Command, 2001. "Deployment Planning Guide," TEA Pam 700-5, May.

Surfeace Deployment and Distribution Command, 2002. "Logistics Handbook for Strategic Mobility Planning," TEA Pam 700-2, September.

Mulvey, J.M., Vanderbei, R.J. and Zenios, S.A., 1995. "Robust Optimization of Large-Scale Systems," *Operations Research*, **43**, pp. 264–281.

Mulvey, J.M. and Ruszczynski, A, 1995. "A New Scenario Decomposition Method for Large-Scale Stochastic Optimization," *Operations Research*, **43**, pp. 477–490.

Mulvey, J.M., Gould, G. and Morgan, C., 2000. "An Asset and Liability Management System for Towers Perrin-Tillinghast," *Interfaces*, **30**, pp. 96-114.

National Academy Press, 1999. "Naval Expeditionary Logistics: Enabling Operation

Maneuver from the Sea," National Academy Press, Washington, DC.

Naval Historical Center, 2006. "The United States Navy in 'Desert Shield'/'Desert Storm'. Part II: The Gathering Storm - The Build-Up of U.S. Forces," http://www.history.navy.mil/wars/dstorm/ds2.htm (accessed October 2006).

Powell, W., 2005. "The Optimizing-simulator: Merging Simulation and Optimization Using Approximate Dynamic Programming," in M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds., *Proceedings of the 2005 Winter Simulation Conference*, pp. 96–109.

Ravindran, A., Phillips, D.T. and Solberg, J.J., 1987. *Operations Research—Principles and Practice.* John Wiley & Sons, New York.

Rockafellar, R.T. and Wets, R.J-B, 1991. "Scenarios and Policy Aggregation in Optimization under Uncertainty," *Mathematics of Operations Research*, **16**, pp. 119–147.

Rosenthal, R.E., Baker, S.F., Weng, L.T., Fuller, D.F., Goggins, D., Toy, A.O., Turker, Y., Horton, D., Briand, D., and Morton, D.P., 1997. "Application and Extension of the Thruput II Optimization Model for Airlift Mobility," *Military Operations Research*, **3**, pp. 55–74.

Sinex, C.C., Kerchner, D., Cox, K., Basile, S., Sellers, W., Hindman, S., Loftus, S. and Hummel, J., 1998. "WLTAE: A HLA Federation for Logistics and Warfighting Models," 98F-SIW-118, Simulation Interoperability Workshop, September.

Theres, M., 1998. "Uni-Modal Versus Bi-Modal Transportation: A Comparative Study," M.S. Thesis in Operations Research, Naval Postgraduate School, Monterey, CA, September.

United States Combined Arms Support Command, OPLOGPLN Operations Logistics Planner, version 1.40, 1997.

USTRANSCOM, 2000. Understanding the Defense Transportation System, USTRANSCOM Handbook 24-2, 3rd edition, 1 September.

Wallace, S.W., 2000. "Decision Making Under Uncertainty: Is Sensitivity Analysis of Any Use?" *Operations Research*, **48**, pp. 20–25.

Wets, R.J-B, 1980. "Stochastic Multipliers, Induced Feasibility and Nonanticipativity in Stochastic Programming," in *Stochastic Programming*, University of Oxford, Oxford, England, Academic Press, pp. 137–146.

Whiteman, P.S., 1999. "Improving the Effectiveness of Single-Strike Effectiveness for Network Interdiction," *Military Operations Research*, **4**, pp. 15–30.

World Port Database, 2006. Available from https://www.tea.army.mil/DATA/default.htm (restricted upon granting of an Army Knowledge Online account).

Ziemba, W.T., 1974. "Stochastic Programs with Simple Recourse," in P.L. Hammer, G. Zoutendijk, eds. *Mathematical Programming in Theory and Practice*, North-Holland, Amsterdam, pp. 213–273.