Chapter 1

TARGET-BASED WEAPON TARGET ASSIGNMENT PROBLEMS

Robert A. Murphey
Air Force Research Laboratory, Munitions Directorate,
101 W. Eglin Blvd., Ste 330
Eglin AFB, FL 32542-6810
Murphey@eglin.af.mil
Keywords: Weapon target assignment, Dynamic Programming, Stochastic programming.

1. INTRODUCTION

In this chapter we consider a class of non-linear assignment problems collectively referred to as Target-based Weapon Target Assignment (WTA). The Target-based Weapon Target Assignment problem considers optimally assigning M weapons to N targets so that the total expected damage to the targets is maximized. We use the term target-based to distinguish these problems from those that are asset-based, that is problems where weapons are assigned to targets such that the value of a group of assets is maximized supposing that the targets themselves are missiles engaging the assets. The asset-based problem is most pertinent to strategic ballistic missile *defense* problems whereas the target-based problems apply to *offensive* conventional warfare types of problems. Previous surveys of WTA problems were made by Matlin [11] and Eckler and Burr [5] the first on offensive, target based problems and the second primarily about defensive, asset based problems.

If at some time t the numbers and locations of weapons and targets are known with certainty, then a single assignment may be made at time t such that all of the weapons are committed. This formulation is denoted *static* WTA. In its most general form, static WTA is known to be $\mathcal{NP}$-complete. A more difficult problem results when assignments may be made at any of several discrete points in time. These types of

problems are known as *dynamic* WTA and are formulated in 2 ways. In the first type, the numbers and locations of targets are all known a priori. However, since the survival of a target with weapons assigned to it is probabilistic, the outcome of any time interval (i.e. which targets survive) is stochastic and so influences all future assignments. These problems are known as shoot-look-shoot since some observation of the targets is required to condition the problem at the following interval. Shoot-look-shoot problems are readily adpated to a dynamic programming formulation. In the second type of dynamic WTA, we assume that the targets are not observable, hence the outcome of each interval is deterministic. However, we cannot assume that the numbers and locations of targets are known a priori. That is, at each interval, a subset of the N targets are known with certainty and the remainder are either not known or known only stochastically. This dynamic WTA formulation may be modeled as stochastic programs.

2. STATIC ASSIGNMENT MODELS

Consider that there are M weapons and N independent targets. Define the decision variable $x_{ij},\ i = 1, 2, \ldots M,\ j = 1, 2, \ldots, N$ as

$$x_{ij} = \begin{cases} 1: & \text{weapon } i \text{ assigned to target } j \\ 0: & \text{else} \end{cases} \tag{1.1}$$

Given that weapon i engages target j, that is $x_{ij} = 1$, the outcome is random and assumes a Bernoulli distribution:

$$\begin{aligned} &P(\text{target } j \text{ is destroyed by weapon } i) = P_{ij} \\ &P(\text{target } j \text{ is not destroyed by weapon } i) = 1 - P_{ij} \end{aligned} \tag{1.2}$$

Assume that each weapon engagement is independent of every other engagement. Then the outcomes of the engagements are independent Bernouilli distributed.

Let each target be assigned a positive real number V_j to indicate preference between targets. Assume that targets may be partitioned into classes and each class has a unique value to the decision maker. Let the set of all class values be contained in $\mathbb{V}$ such that $\mathbb{V}$ is of cardinality K. Our objective is to maximize the expected damage to the targets which is equivalent to minimizing the expected target value. With $q_{ij} = (1 - P_{ij})$, the resulting integer programming formulation is:

SWTA

$$\begin{aligned}
&\text{minimize} \ \sum_{j=1}^{N} V_j \prod_{i=1}^{M} q_{ij}^{x_{ij}} \\
&\text{subject to} \\
&\qquad \sum_{j=1}^{N} x_{ij} = 1, \quad i = 1, 2, \ldots, M \\
&\qquad V_j \in \mathbb{V} \in \mathbb{R}_+^K, \ \ j = 1, 2, \ldots, N \\
&\qquad x \in \mathbb{B}^{M \times N}
\end{aligned}$$

Note that $\mathbb{R}$ denotes the real number space, and $\mathbb{Z}$ and $\mathbb{B}$ denote the integer and binary subspaces respectively. The equality constraints of SWTA ensure that each weapon is assigned to exactly one target. Notice that nothing prevents all weapons from being assigned to a single target. This problem formulation was first suggested by Manne [12].

Although the nonlinear assignment problem developed in SWTA was shown to be *NP*-complete in [10], a useful lower bound may be developed and then applied in a branch and bound or other global integer optimization scheme. A lower bound for SWTA may be obtained by relaxing integrality of x:

RSWTA

$$\begin{aligned}
&\text{minimize} \ \sum_{j=1}^{N} V_j \prod_{i=1}^{M} q_{ij}^{x_{ij}} \\
&\text{subject to} \\
&\qquad \sum_{j=1}^{N} x_{ij} = 1, \qquad\qquad i = 1, 2, \ldots, M \\
&\qquad V_j \in \mathbb{V} \in \mathbb{R}_+^K, \qquad\qquad j = 1, 2, \ldots, N \\
&\qquad 0 \le x_{ij} \le 1, \quad i = 1, 2, \ldots, M, \ j = 1, 2, \ldots, N
\end{aligned}$$

The objective of RSTWA is clearly convex while the constraints are linear resulting in a convex programming problem. A number of methods may be applied to solve RSWTA, however, for large M and N, dual methods should be much faster since the dual of RSWTA will only have M constraints. Indeed, the dual of RSWTA is simply

$$d(\lambda) = \text{minimize}_x \ L(x, \lambda)$$

where the Lagrangian is defined as:

$$L(x,\lambda) = \sum_{j=1}^{N} V_j \prod_{i=1}^{M} q_{ij}^{x_{ij}} + \sum_{i=1}^{M} \lambda_i \left(\sum_{j=1}^{N} x_{ij} - 1 \right).$$

Now it can be seen that $d(\lambda)$ has the seperable form:

$$d(\lambda) = \sum_{j=1}^{N} \min_{x} \left(V_j \prod_{i=1}^{M} q_{ij}^{x_{ij}} + \sum_{i=1}^{M} \lambda_i x_{ij} \right) - \sum_{i=1}^{M} \lambda_i$$

and by Strong Duality, there is no duality gap.

2.1 UNIFORM WEAPONS

If the weapons are all assumed to be *uniform* that is, they are all the same, then the *uniform static WTA* problem may be written as:

SWTA – U

$$\begin{aligned} &\text{minimize} && \sum_{j=1}^{N} V_j q_j^{x_j} \\ &\text{subject to} \\ &&& \sum_{j=1}^{N} x_j = M, \\ &&& V_j \in \mathbb{V} \in \mathbb{R}_+^K, \;\; j = 1, 2, \ldots, N \\ &&& x \in \mathbb{Z}^N \end{aligned}$$

where now x_j is the number of weapons assigned to target j and the index i has been dropped from q since, given that the weapons are all the same, the probability of destroying target j depends only on the target.

den Broeder *et al* [4] presented an algorithm for solving SWTA optimally in $\mathrm{O}(N + M\log(N))$ time termed the minimum marginal return (MMR). Essentially, MMR assigns weapons one at a time to the target which realizes the greatest decrease in value, where target value is defined to be $q_j^{x_j}$. den Broeder showed that this greedy strategy is optimal for the uniform weapon assumption. The algorithm is shown in Figure 1.1.

Hosein [8], later showned that a variant of iterated 1-opt local search on any feasible assignment is also optimal for SWTA-U. The iterated

Step 0. For each $j = 1, \ldots N$, let $x_j = 0$, and denote probability of survival of target j by $S_j = V_j q_j^{x_j}$. Initialize weapon index $i = 1$.

While $i \leq M$ DO

Step 1. Find target k for which munition i has greatest effect:

$$k = \arg\max_j \{S_j(1 - q_j)\}.$$

Step 2. Add munition i to target k: $x_k = x_k + 1$ and revise probability of survival of target k: $S_k = S_k q_k$.

$i = i + 1$.

Figure 1.1 Minimum Marginal Return (MMR)

1-opt variant is easily implemented for this problem by removing one weapon from target j and replacing it on target k given that the move decreases the objective value.

Note that if all targets are also uniform, q_j is replaced by q and V_j is dropped. It is easily shown that the optimal strategy in such cases is to simply spread the weapons evenly across all targets (see [10]).

2.2 RANGE RESTRICTED WEAPONS

Realistic WTA models dictate that not all weapons will be reachable by all targets. Consequently, the *range restricted uniform SWTA* problem is presented:

SWTA – URR

$$\begin{aligned}
\text{minimize} \quad & \sum_{j=1}^{N} V_j q_j^{\sum_{i \in R_j} x_{ij}} \\
\text{subject to} \quad & \\
& \sum_{j=1}^{N} x_{ij} = 1, \qquad i = 1, 2, \ldots, M \\
& V_j \in \mathbb{V} \in \mathbb{R}_+^K, \qquad j = 1, 2, \ldots, N \\
& x \in \mathbb{B}^N
\end{aligned}$$

where R_j is the set of weapon indices that are within reachable range of target j. By representing each weapon as a node and each target as a node in a bipartite graph partitioned on weapons and targets, Hosein [8] showed that SWTA-URR was in-fact solvable as a network flow problem when an edge is added between weapon i and target j only if $i \in R_j$ for x_{ij}. A supply of one is provided at each weapon node and the target nodes are all sunk to a single sink node with demand M. The capacity of all weapon to target arcs is one while the cost of these arcs is always zero. The capacity of all arcs from the targets to the sink node are each M while the costs of these arcs are

$$F_{js}(x) = V_j \left[q_j^{\lfloor x \rfloor} + (x - \lfloor x \rfloor) \left(q_j^{\lceil x \rceil} - q_j^{\lfloor x \rfloor} \right) \right]$$

where flow x on arc $\{j, s\}$ is continuous between 0 and M. Since $F_{js}(x)$ is a convex combination, the resulting network flow problem is clearly convex. Furthermore, there will always exist an integer valued solution that is optimal for the convex minimum cost network flow problem that is also optimal for SWTA-URR.

3. THE DYNAMIC ASSIGNMENT MODEL

In static WTA problems the assignment is made at a single point in time. Now suppose that assignments may be made at any of T discrete intervals in time. There are 2 fundamental types of dynamic WTA problems in the literature. The most studied is the case when the numbers and locations of targets are all known a priori. However, since the survival of a target with weapons assigned to it is probabilistic, the outcome of any time interval (i.e. which targets survive) is stochastic and so influences all future assignments. These problems are known as

shoot-look-shoot since some observation of the targets is required prior to making any future assignments. Shoot-look-shoot problems are readily adpated to a dynamic program yet prove very difficult to solve in the most general case. In the second type of dynamic WTA, we assume that the targets are not observable, hence the outcome of each interval is essentially deterministic. However, we no longer assume that the numbers and locations of targets are known a priori. That is, at each interval, a subset of the N targets are known with certainty and the remainder are either not known or known only stochastically. These dynamic WTA formulations may be modeled as stochastic programs.

3.1 SHOOT-LOOK-SHOOT PROBLEMS

In shoot-look-shoot problems, the assignment of M weapons to N targets is made over T discrete intervals in time. Therefore, the decision variable must be indexed by time as in $x_{ij}(t),\ t = 1, 2, \ldots, T$. The assumption is that after an assignment, the outcome of each engagement can be observed without error. As a result, we discuss the evolution of the *target state* as a function of time and the previous assignment. At any time t, this state is simply

$$U_j = \begin{cases} 1: & \text{target } j \text{ survives stage } t-1 \\ 0: & \text{else} \end{cases}$$

Since the survival of target j depends on the values of $x_{ij}(t-1)$ and q_{ij}, the target state is stochastic and evolves as:

$$P(U_j = u) = u \prod_{i=1}^{M} q_{ij}^{x_{ij}(t-1)} + (1-u)\left(1 - \prod_{i=1}^{M} q_{ij}^{x_{ij}(t-1)}\right)$$

Similarly, the *weapon state* is defined as

$$w_i = \begin{cases} 1: & \text{weapon } i \text{ not used in stage } t-1 \\ 0: & \text{else} \end{cases}$$

which evolves as:

$$w_i = 1 - \sum_{j=1}^{N} x_{ij}(t-1)$$

that is, weapon i is available for use in stage t only if it was not used in stage $t-1$. Implied in this model is the fact that weapon i is available

for use in stage t only if it was not used in *any previous* stage. This is based on an inductive argument that assumes that a solution of stage $t-1$ considers what weapons were not used in stage $t-2$ and so on.

Define the cost of a stage T assignment to be $F_1(u,w)$, the cost of a stage $T-1$ assignment to be $F_2(u,w)$, and so on. Now the dynamic shoot-look-shoot WTA problem (DWTA-SLS) is defined as the following recursive optimization problem:

DWTA – SLS

$$\min F_1 = \min_x \sum_{u \in \mathbb{Z}^N} P(U=u) F_2^*(u,w)$$

subject to

$$x \in \mathbb{B}^{M \times N}$$

$$\sum_{j=1}^{N} x_{ij} = 1 - w_i, \quad i = 1, \ldots, M$$

where the optimals solution to the stage $T-1$ problem, denoted by $F_2^*(u,w)$, is obtained by solving DWTA-SLS with the stage $T-1$ solution now defined by F_1. In a like manner, $F_3^*(u,w)$ is obtained and so on. Note however, that we need only use a 2-stage definition at any time. This formulation is clearly amenable to Dynamic Programming (DP) methods. Unfortunately, even for $T=2$, the number of computations required for DP is unrealistically large, being that 2^M assignments are possible in stage 1, each for which there are N^{m_2} possible assignments in stage 2, where m_2 is the number of weapons left for stage 2.

Hosein [8] studied DWTA-SLS in detail and found useful solutions for a simpler case when the probabilities of survival q_{ij} were no longer dependent on weapon or target but instead only dependent on the stage t. When this assumption is made our decision variable is now m_t, the number of weapons used in stage t. Hosein shows that these problems share an optimal strategy:

Optimal strategy of stage dependent probabilities. *For $t = 1, \ldots, T$, the m_t weapons used at stage t should be spread evenly over all the N targets.*

Of course we must still determine optimal values for $m_t, t = 1, \ldots T$. Hosein examines 3 special cases.

Case 1. *If $M \leq N$ then assign all weapons in the stage with the lowest $q(t)$.*

This is essentially the SWTA problem.

Case 2. *If* $M \geq N$ *and* $q(t) \leq q(t+1)$, $t = 1, \ldots, T-1$, *then* $m_1^* \geq N$.

Case 2 means that for $q(t)$ nondecreasing, the optimal first stage assignment is to use at least as many weapons as there are targets.

Case 3. *If* $T > 1 + \frac{M-N}{2}$ *for* $M > N$ *and* $q(t) = q$ *for all* t, *then* $m_1^* = N$.

This simply means that for large numbers of intervals and a constant q, the optimal strategy is to assign exactly N weapons, and by the fact that m_t weapons used at stage t should be spread evenly over all the N targets, we surmise that each target should get exactly one weapon in stage 1.

Finally, Hosein explored DWTA-SLS for very large numbers of targets by developing a limit on F_1 as $N \to \infty$.

3.2 STOCHASTIC DEMAND PROBLEMS

In the static WTA formulations, the decision maker knows at one instant in time the total number of targets N and their locations. Consider another WTA problem, where at some time t, only a subset of the total targets are known to the decision maker. Let the number of targets known at time t be denoted by $n(t)$ and $n(t) \leq N$ for any t. As time progresses, additional targets are discovered, hence $n(t)$ is nondecreasing with time. Since N is unknown, an assignment of weapons can only be made to the $n(t)$ weapons that are known or else reserved for targets that are expected in the future. These types of Dynamic WTA problems essentially have a stochastic demand and were first introduced in [13].

If the targets are ranked according to their value, as in the static WTA problems, then it is possible that at time t only low value targets have been found, in which case it is desirable to wait before making an assignment. If, after waiting time τ, all of the targets are discovered, that is $n(\tau) = N$, then a static WTA problem results. Assume however that there is a cost associated with waiting to make an assignment. This cost is due to the fact that each weapon has finite fuel, so once it is deployed, it has a finite time to detect and engage targets. Indeed, the target may pass out of the weapon's field of view altogether.

Define T to be the latest possible time to assign a weapon to a target. The *stochastic demand* problem (DWTA-SD) then attempts to find an assignment $x(t)$, which minimizes the value of targets over all $t \in [0, T]$. The cost function monotonically increases with time to account for the cost of waiting and, as in the Static WTA problems, is weighted by target values $V_j \in \mathbb{V}$. Assume that $\mathbb{V}$ does not change over time.

To make the problem easier to discuss, time will be discretized. A deterministic formulation for DWTA-SD is

DWTA-SD

$$\text{minimize} \quad \sum_{t=1}^{T} c(t) \sum_{j=1}^{n(i)} V_j \prod_{i=1}^{M} q_{ij}^{x_{ij}(t)}$$

subject to

$$\sum_{t=1}^{T} \sum_{j=1}^{n(t)} x_{ij}(t) = 1, \quad i = 1, 2, \ldots, M$$

$$V_j \in \mathbb{V} \in \mathbb{R}_+^K, \quad j = 1, 2, \ldots, n(T)$$

$$x(t) \in \mathbb{B}^{n(i)}, \quad t = 1, 2, \ldots, T$$

where $c(t)$ is a nondecreasing function that represents the cost of waiting. To model attrition, the number of munitions M may be a function of t. If so, $c(t)$ need only represent the cost of missed reacquisitions, which can be formulated as a linear scalar function of t. Clearly DWTA-SD is a very hard problem since at each time interval, a SWTA must be solved. Hence we consider DWTA-SD with a uniform weapon assumption:

DWTA-SDU

$$\text{minimize} \quad \sum_{t=1}^{T} c(t) \sum_{j=1}^{n(t)} V_j q_j^{x_j(t)}$$

subject to

$$\sum_{t=1}^{T} \sum_{j=1}^{n(t)} x_j(t) = M,$$

$$V_j \in \mathbb{V} \in \mathbb{R}_+^K, \quad j = 1, 2, \ldots, n(T)$$

$$x(t) \in \mathbb{Z}^{n(t)}, \quad t = 1, 2, \ldots, T$$

On-line implementation of DWTA-SD or even DWTA-SDU is unrealistic since it determines the optimal decision after the final time T has passed, yet assignments for any time interval $t < T$ must be made during that interval. A realistic model must account for the fact that at interval I, $n(t)$ are stochastic for $t > I$. There are two approaches for a stochastic model; two-stage and multistage. We will only develop the two-stage model here since it is difficult to predict exactly *when*

an unknown target might appear as would be required in a multistage model. It is fairly straightforward to model how many targets of each class remain undetected within a finite geographic region at any instant in time. As a result, the 2-stage model must be solved recursively at each time interval, each solution consisting of two subproblems: (1) a first stage problem which includes all targets detected during the current interval and all previous intervals, and (2) a second stage problem that includes all targets not yet detected. The first stage demand is deterministic while the second stage demand is stochastic and the trick is to balance assigning weapons to known targets against reserving weapons for possible, but not yet determined, future targets.

Let a random vector $\xi \in \Xi$ denotes the number of targets in each class $k,\ k = 1, \ldots, K$ that have yet to be detected (e.g. targets in stage 2). If we are given the probabilities of survival and the target values for each target class, then for any instance of random vector ξ, the second stage values of target value V_2, probability of survival q_2 and the total number of yet to be detected (stage 2) targets n_2 are easily determined. The 2-stage stochastic programming formulation is as follows:

SP

$$Z_1(x) = \min_x\ f_1(x) + E_{\xi \in \Xi}[Z_2(x, \xi^j)] \tag{1.3}$$

subject to

$$\sum_{i=1}^{n_1} x_i \leq M, \tag{1.4}$$

$$x \leq b \tag{1.5}$$

$$x \in \mathbb{Z}^{n_1} \tag{1.6}$$

where $f_1(x)$ is the stage one cost function of the first stage assignment x and is integer-convex, that is, the continuous relaxation of $f_1(x)$ is convex. $E_{\xi \in \Xi}[Z_2(x, \xi^j)]$ is the expectation operator with respect to ξ. The constraint in (1.5) is used to limit the number of weapons assigned to any single target, preventing the assignment of large numbers of weapons to high value targets. The right hand side of (1.5) is determined by setting a threshold Φ_j on the reduced value of each target as in

$$q_i^{x_i} \leq \Phi_j$$

which can always be written in the form of (1.5) by applying a logarithm. $Z_2(x, \xi^j)$ is the solution to a second stage problem that clearly shows its dependence on the stochastic parameter $\xi \in \Xi$ and the first stage decision x as captured in the following program:

S2P

$$Z_2(x,\xi^j) = \min_y \ f_2^j(y) \tag{1.7}$$

subject to

$$\sum_{i=1}^{n_1} x_i + \sum_{i=1}^{n_2(\xi^j)} y_i = M, \tag{1.8}$$

$$y \in \mathbb{Z}^{n_2} \tag{1.9}$$

where $f_2^j(y)$ is the stage two cost function of the number of weapons y assigned in the second stage. Since y depends on the outcome of ξ, $f_2^j(y)$ depends on ξ and furthermore is integer-convex. Specifically, $f_1(x)$ and $f_2^j(y)$ are the following:

$$f_1(x) = \sum_{i=1}^{n_1} V_1^i (q_1^i)^{x_i} \tag{1.10}$$

$$f_2^j(y) = \sum_{i=1}^{n_2(\xi)} V_2^i(\xi) q_2^i(\xi)^{y_i} \tag{1.11}$$

Due to the discrete support Ξ of ξ, SP may be replaced with the following so called *deterministic equivalent* program:

DESP

$$Z_1(x) = \min_x \ f_1(x) + \sum_{j=1}^{s} p^j Z_2(x,\xi^j) \tag{1.12}$$

subject to

$$\sum_{i=1}^{n_1} x_i \leq M, \tag{1.13}$$

$$x \leq b \tag{1.14}$$

$$x \in \mathbb{Z}^{n_1} \tag{1.15}$$

where s is the total number of scenarios (outcomes) for the random vector ξ.

Each stage of DESP/SP2 has an integer-convex objective and integer affine constraints. The classic approach to solving a stochastic programming problem with discrete random support is by the decomposition

method. Decomposition methods decouple the stage 1 and stage 2 problems by first solving a variant of the stage 1 problem, often called the *current problem* (CP) with a scalar variable θ taking the place of the stage 2 solution so that

$$\theta \leq \sum_{j=1}^{s} p^j Z_2(x, \xi^j).$$

The stage 2 problem is solved for each $\xi^j, j = 1, \ldots s$. Information from the stage 2 problems is used to develop a supporting hyperplane constraint which bounds the first stage resource to θ (*optimality cut*). This constraint is added to CP which is solved once again. Iterations between the CP and stage 2 problems continue until the new solution is ϵ-close to the latest optimality cut.

Unfortunately, the integrality of the objective in DESP/SP2 no longer allows supporting hyperplanes for the stage 2 problem (see for example [3], [9], [14]). Nonetheless, a lower bound on the optimal solution can be obtained by applying a decomposition method on a relaxed version of DESP/SP2. The algorithm presented here finds a lower bound by solving a similar problem to ϵ-optimality. The problem solved replaces the integrality of y in S2P with a continuous version:

CS2P

$$Z_2(x, \xi^j) = \min_y \; f_2^j(y) \tag{1.16}$$

subject to

$$\sum_{i=1}^{n_1} x_i + \sum_{i=1}^{n_2(\xi^j)} y_i \leq M, \tag{1.17}$$

$$y \geq 0, y \in \mathbb{R}^{n_2} \tag{1.18}$$

Details of the method are in [13] where it i shown that it terminates in a finite number of iterations with a solution that is ϵ-optimal for DESP/CS2P and consequently yields a lower bound solution for DESP/S2P.

`Lower Bound Algorithm For DESP/SP2.`

Step 0. Set $\theta = 0$, $L = 0$, $k = 1$.

Step 1. Solve the current problem

Current problem

$$\bar{Z}_1(x) = \min_{x,\theta} f_1(x) + \theta \tag{1.19}$$

$$\text{subject to} \sum_{i=1}^{n_1} x_i \leq M, \tag{1.20}$$

$$x \leq b \tag{1.21}$$

$$A^k x + d^k \leq \theta \quad k = 1, \ldots, L \tag{1.22}$$

$$x \geq 0, \quad x \in \mathbb{Z}^{n_1} \tag{1.23}$$

If infeasible, STOP: problem is infeasible.
Else denote solution by (x^k, θ^k).

Step 2. For $j = 1, \ldots, s$ solve second stage primal:

$$Z_2(x^k, \xi^j) = \min_y \ f_2^j(y)$$

$$\text{subject to} \sum_{i=1}^{n_1} x_i^k + \sum_{i=1}^{n_2(\xi^j)} y_i = M,$$

$$y \geq 0, \quad y \in \mathbb{R}^{n_2}$$

and its dual:

$$\max_\lambda \ \min_{y \geq 0} \ f_2^j(y) + \lambda^j \Big(\sum_{i=1}^{n_1} x_i^k + \sum_{i=1}^{n_2(\xi^j)} y_i - M \Big)$$

Let the solutions to these problems be $(y^{j\,k}, \lambda^{j\,k})$.

Step 3. Define

$$A = \sum_{j=1}^{s} p^j \lambda^{j\,k}$$

$$d = \sum_{j=1}^{s} p^j \big(f_2^j(y^k) + \lambda^{j\,k} (e^T y^{j\,k} - M) \big)$$

where e is a vector of ones of comparable size to $y^{j\,k}$. If $\theta^k < Ae^T x^k + d + \epsilon$, for $\epsilon > 0$ where here e is a vector of ones of comparable size to x, then add cut to current problem: $A^k e^T x + d^k \leq \theta$. Let $k = k + 1$. Return to step 1.

Else STOP: (x^k, θ^k) is ϵ-optimal solution for DESP/CS2P.

An algorithm for the current problem in step 1 of the lower bounding algorithm is needed since non-Boolean integrality and non-linearity prevent us from using MMR as for SWTA-U. Whereas the SWTA constrains the sum of the decision variables to equality with M, the current

problem has inequality constraints on the sum of the decisions as in equations (1.20) and (1.22). A natural approach for handling the constraints is to iteratively apply MMR to the current problem where the constraints (1.20) and (1.22) are replaced with the constraint

$$\sum_{j=1}^{N} x_j = m$$

where

$$m = \frac{\theta - d}{A}$$

and m is iterated from $0, 1, \ldots M$. Then for each value of m, the value of θ is uniquely determined and hence fixed in the objective.

References

[1] M. Athans, P. Hosein and J. Walton, Dynamic weapon target assignment problems with vulnerable $C3$ nodes, *Proceedings of the Command and Control Symposium, Monterey, CA*, (1988).

[2] J.R. Birge (1999), "Parallel decomposition of large-scale stochastic nonlinear programs," *working paper*.

[3] C.C. Carøe, and J. Tind (1998), "L-shaped decomposition of two-stage stochastic programs with integer recourse," *Mathematical Programming*, vol. 83, pages 451-464.

[4] G.G. denBroeder, R.E. Ellison, and L. Emerling (1959), "On optimum target assignments," *Operations Research*, vol. 7, pages 322-326.

[5] A.R. Eckler and S.A. Burr (1972), "Mathematical Models of Target Coverage and Missile Allocation," Military Operations Research Society Press, Alexandria, VA.

[6] M. Flood (1957), Verbal communication at The Princeton University Conference on Linear Programming.

[7] J.B. Hofman, W.A. Metler and F.L. Preston (1990), "A suite of weapon assignment algorithms for an SDI midcourse battle manager," *Naval Research Laboratory Report* **6713**.

[8] P.A. Hosein (1989), "A Class of Dynamic NonLinear Resource Allocation Problems," *Ph.D. Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, MA*.

[9] G. Laporte and F.V. Louveaux (1993), "The integer L-shaped method for stochastic integer programs with complete recourse," *Operations Research Letters*, vol. 13, pages 133-142.

[10] S.P. Lloyd and H.S. Witsenhausen (1986), "Weapons allocation is $\mathcal{NP}$-complete," *IEEE Summer Simulation Conference*.

[11] S. Matlin (1970), "A review of the literature on the missile allocation problem," *Operations Research* 18, pages 334-373.

[12] A.S. Manne (1958), "A target assignment problem," *Operations Research*, vol. 6, pages 346-351.

[13] R.A. Murphey (1999) An approximate algorithm for a weapon target assignment stochastic program, *Approximation and Complexity in Numerical Optimization: Continuous and Discrete Problems*, P.M Pardalos, (Ed.), Kluwer Academic Publishers.

[14] R. Shultz, L. Stougie, and M.H. van der Vlerk (1998), "Solving stochastic programs with integer recourse by enumeration: a framework using Gröbner basis reductions," *Mathematical Programming*, vol. 83, pages 229-252.