THE DYNAMIC WEAPON-TARGET ASSIGNMENT PROBLEM

Patrick Hosein
Michael Athans

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA 02139

ABSTRACT

We present a progress report on our recent results on the dynamic version of the Weapon to Target Assignment (WTA) problem. In our previous paper (Hosein et. al[1]) we presented results for the Target-Based problem. In this paper we will present results for the Asset-Based problem.

In the static Asset-Based WTA problem, the offense launches missiles at valuable assets of the defense. The defense must assign its weapons to these missiles so as to minimize the damage they incur. In the dynamic version, this allocation is done in time stages such that the outcomes of the previous engagements can be used in making future assignments.

We will provide a suboptimal algorithm for the Static Asset-Based WTA problem for the case of a single class of defense weapons. We will also propose a heuristic for the dynamic version of the problem. Numerical results will be provided to show that a dynamic strategy offers a significant cost improvement over a static one.

1. INTRODUCTION

The long range objective of our research is the quantitative study of the theory of distributed C^3 organizations. Our present work has been concentrated on certain aspects of situation assessment and resource commitment.

Situation assessment entails the use of sensors to detect and track the enemy and its weapons (i.e missiles, tanks etc.). These sensors are geographically distributed so that distributed algorithms are desirable. This problem can be formulated as a distributed hypothesis testing problem. Recent results of this research may be found in the paper by Pothiawala et. al[2] in these proceedings.

The resource commitment problem deals with the optimal assignment of the defense's resources against the offense's forces so as to minimize the damage done to the defense's assets. If the battle is such that the defense has a single opportunity to engage the enemy then the problem can be formulated as a static resource allocation problem. If multiple engagements are possible (as for example in the Strategic Defense System (SDS) scenario) then better use can be made of the defense's resources by assigning them dynamically (i.e observe the outcomes of previous engagements before making further assignments). This is called a shoot-look-shoot strategy in the literature. In this paper we will provide a heuristic for the dynamic problem and make performance comparisons of the static and dynamic strategies.

The resource allocation problem will typically be solved at a C^3 node and the results transmitted to the relevant resources. Each of these C^3 nodes will therefore be of vital importance to the defense since its destruction will in effect paralyze the resources over which it has control. These nodes must therefore be defended or replicated to increase the reliability of the system. A progress report on research on the problem of reliability can be found in the paper by Walton and Athans[3] in these proceedings.

This paper is in effect a progress report on our research on the resource commitment problem. The model we use is rich enough to capture the nature of the mission (defense of the assets), enemy strength (number and effectiveness of the enemy's missiles), defense strength (number and effectiveness of the defense's weapons) and strategy and tactics (preferential defense, shoot-look- shoot etc.). It should be noted that basic research studies on these topics are virtually non-existent.

Our work is motivated by military defense problems, two examples of which are as follows. The first example involves the Anti-Aircraft Weapon (AAW) defense of Naval battle group or battle force platforms. The assets being defended are aircraft carrier(s), escort warships and support ships each of which is of some intrinsic value to the defense. The threat to these assets are enemy missiles launched from submarines, surface ships and aircraft. These missiles may have different damage probabilities which depend on the missile type, asset type, etc. The defense's weapons are different types of AAW interceptors launched from Aegis and other AAW ships. The kill probability of these weapons may also depend on the specific missile-interceptor pair. The objective of the defense is to maximize the expected surviving value of the assets. The problem is to find which AAW interceptors should be assigned to each of the enemy missiles, when should they be launched and why. This formulation allows for a preferential defense where, in a heavy attack, it may be optimal for the defense to leave "low" valued assets undefended and concentrate its resources on saving the "high" valued assets.

The second motivating example for our research is the mid-course phase of the Strategic Defense System. In this case the assets are our (the defense's) population centers, Inter-Continental Ballistic Missile (ICBM) silos, military installations, C^3 nodes, etc. The threat to these assets are enemy re-entry vehicles (RV's), surrounded by decoys. The defense's weapons are Space-based kinetic-kill vehicles (SBKKV's) and ERIS interceptors. The objective of the defense is the maximization of the expected total

To appear in Proc. 1989 Symposium on C^2 Research, Washington, D.C.

surviving value of the assets. The problem is the determination of the optimal weapon-target assignments and the timing of the interceptor launches.

2. THE STATIC WTA PROBLEM

In this section we will present the Static Asset-Based Weapon-Target Assignment problem and a sub-optimal algorithm for solving it. We will also show how an upper bound on the optimal cost can be obtained. In this problem, missiles (the targets), are launched by the offense and are aimed at valuable assets of the defense. A value is assigned to each of these assets by the defense. The defense has a limited number of weapons with which to destroy these targets. Associated with each weapon-target pair is a kill probability which is the probability that a weapon destroys the target if it is fired at it. We assume that the action of a weapon on a target is independent of all other weapons and targets. Each of the attacking missiles are aimed at exactly one of the defended assets and, if not intercepted, destroys the asset with some probability, which will be called the damage probability of the target. We will assume that the action of a target on an asset is independent of all other targets and assets. The objective of the defense is to assign weapons to targets so as to maximize the expected total value of the surviving assets.

2.1 Problem Statement

The following notation will be used:

$K \stackrel{\text{def}}{=}$ the number of assets being defended,
$W_k \stackrel{\text{def}}{=}$ the value of asset k to the defense,
$G_k \stackrel{\text{def}}{=}$ the set of targets aimed for asset k,
$n_k \stackrel{\text{def}}{=}$ the size of set G_k, ($|G_k|$),
$\pi_i \stackrel{\text{def}}{=}$ the damage probability of target i,
$p_i \stackrel{\text{def}}{=}$ the kill probability of a weapon on target i,
$N \stackrel{\text{def}}{=}$ the number of targets,
$M \stackrel{\text{def}}{=}$ the number of weapons,
$x_i \stackrel{\text{def}}{=}$ the number of weapons assigned to target i
$\vec{x} \stackrel{\text{def}}{=}$ the N-dimensional vector $[x_1, \ldots, x_N]^T$,
$X_k \stackrel{\text{def}}{=}$ the no. of weapons assigned to defend asset k,
$\vec{X} \stackrel{\text{def}}{=}$ the K-dimensional vector $[X_1, \ldots, X_K]^T$,
$Z_+^N \stackrel{\text{def}}{=}$ the set of vectors with non-negative, integer elements.

The Asset-Based WTA problem can be stated as follows:

$$\max_{\{\vec{x} \in Z_+^N\}} J(\vec{x}) = \sum_{k=1}^{K} W_k \prod_{i \in G_k} (1 - \pi_i(1-p_i)^{x_i}), \tag{1}$$

$$\text{subject to} \qquad \sum_{i=1}^{N} x_i = M.$$

The objective function is the total expected surviving asset value and the constraint is due to the fact that the number of weapons assigned must equal the number of weapons available.

Because problem 1 is separable with respect to the assets, it can be re-formulated as follows. Let $J_k(X_k)$ denote the maximum expected surviving value of asset k if X_k weapons are used to defend it. We have:

$$J_k(X_k) = \max_{\{x_i \in Z_+\}} W_k \prod_{i \in G_k} (1 - \pi_i(1-p_i)^{x_i}), \tag{2}$$

$$\text{subject to} \qquad \sum_{i \in G_k} x_i = X_k.$$

The objective function is the expected surviving value of the asset. The original problem can be restated as:

$$\max_{\{\vec{X} \in Z_+^K\}} J(\vec{X}) = \sum_{k=1}^{K} J_k(X_k) \tag{3}$$

$$\text{subject to} \qquad \sum_{k=1}^{K} X_k = M.$$

Note that this problem has K variables as compared to the original problem which had N variables.

2.2 Solution of the Static WTA Problem

We will first provide an optimal algorithm for the subproblem 2. This will provide us with the functions $J_k(X_k)$. We will then provide a sub-optimal algorithm for the problem as stated in the form 3. We will also provide an upper bound on the optimal cost. This upper bound can be used to check the quality of the solution provided by the suboptimal algorithm.

2.2.1 Solution of the Subproblem

Since the logarithm function is monotonic then if we replace the objective function of problem 2 by its logarithm then the optimal assignment of the resulting problem will also be optimal for the original problem. If we take the logarithm then the following equivalent problem must be solved:

$$\max_{\{x_i \in Z_+\}} F(\vec{X}) = \sum_{i \in G_k} \ln(1 - \pi_i(1-p_i)^{x_i}), \tag{4}$$

$$\text{subject to} \qquad \sum_{i \in G_k} x_i = X_k.$$

Note that the function $F(\vec{X})$ is a separable concave function. Problem 4 can be solved by using a greedy algorithm. Such an algorithm works by sequentially assigning weapons to the target for which the increase in the objective cost is maximum (see the thesis by Hosein[4] for details). The assignment produced by such an algorithm will be optimal for the subproblem 2

2.2.2 Solution of the Main Problem

A typical example of the function $J_k(X_k)$ is given in figure 1. Note that the function is convex for small values of X_k and concave for large values. Let us define $\bar{J}_k$ to be the hull of the function J_k (i.e $\bar{J}_k$ is the minimum concave function which is greater than or equal to the function J_k). A good approximation to this hull is the function which is the same as J_k for large values of X_k and which is the tangent of J_k which passes through the origin for small values of X_k. Let us now define the approximate problem to 3 as

$$\max_{\{\vec{X} \in Z_+^K\}} \bar{J}(\vec{X}) = \sum_{k=1}^{K} \bar{J}_k(X_k) \tag{5}$$

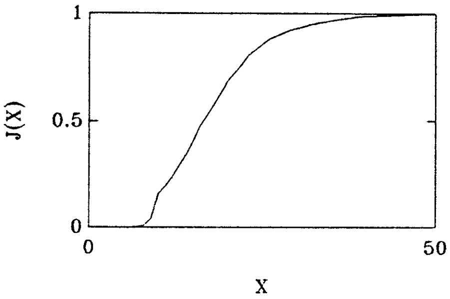

Figure 1: Typical example of the function $J_k(X_k)$

$$\text{subject to} \quad \sum_{k=1}^{K} X_k = M.$$

Since this is a separable concave optimization problem one can again show that a greedy algorithm is optimal (see Hosein[4]). Let $\vec{\bar{X}}^*$ denote the optimal solution of problem 5. By the nature of the greedy algorithm, we can show that for all but one of the assets

$$J_k(\bar{X}_k^*) = \bar{J}_k(\bar{X}_k^*).$$

Let the asset for which this equality does not hold be asset v. Also let $\vec{X}^*$ denote the optimal solution to the original problem 3. One can show that

$$J(\vec{\bar{X}}^*) + \bar{J}(\bar{X}_v^*) - J(\bar{X}_v^*) \geq J(\vec{X}^*) \geq J(\vec{\bar{X}}^*). \qquad (6)$$

Therefore the optimal solution to the approximate problem 5 can be used to obtain upper and lower bounds on the optimal cost of problem 3.

Notice that the solution to the approximate problem 5 is a near optimal solution to problem 3. The difference in cost between these two solutions is bounded by:

$$J(\vec{X}^*) - J(\vec{\bar{X}}^*) \leq \bar{J}(\bar{X}_v^*) - J(\bar{X}_v^*). \qquad (7)$$

Note that if $\bar{J}(\bar{X}_v^*) = J(\bar{X}_v^*)$ then $\vec{\bar{X}}^*$ is optimal for 3. It can also be shown that if $\bar{J}(\bar{X}_v^*) \neq J(\bar{X}_v^*)$ then by slightly increasing or slightly decreasing the number of resources one can obtain a problem for which the solution to the approximate problem 5 is also optimal for 3.

2.3 Numerical Results

Example

Consider the following problem:
$M = 200, N = 100, K = 10.$
$V_k = 1, n_k = 10, \quad k = 1, \ldots, K.$
$p_i = .7, \pi_i = .8, \quad i = 1, \ldots, N.$
Solution obtained by algorithm:
$\vec{X} = [0,0,0,20,30,30,30,30,30,30].$
$J(\vec{X}) = 5.30$
Upper bound $= 5.36$
NOTE: If $M = 180$ or $M = 210$ then the algorithm produces the optimal solution.

Optimal cost Sensitivity Analysis

Consider the following problem:
$M = 200, N = 100, K = 10.$
$V_k = 1, n_k = 10, \quad k = 1, \ldots, K.$

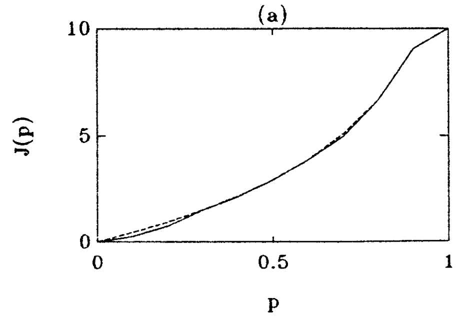

Figure 2: Plot of the cost and upper bound versus the kill probability p_i

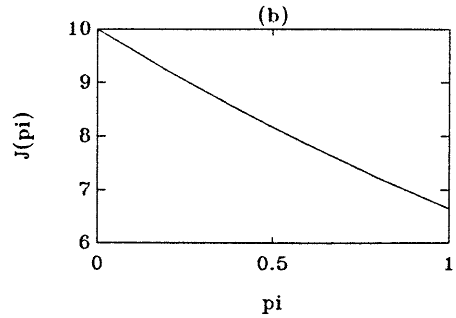

Figure 3: Plot of the cost and upper bound versus the damage probability π_i

$p_i = .8, \pi_i = 1, \quad i = 1, \ldots, N.$
In figure 2 we have plotted the cost of the solution produced by the algorithm for various values of the kill probability p. The dotted curve is the upper bound. Note that for kill probabilities greater than 0.5 the optimal cost is very sensitive to the value of the kill probability. In figure 3 we have plotted the cost for various values of the damage probability π. Note that the cost decrease almost linearly with the damage probability. In figure 4 we have plotted the cost for various numbers of weapons. Note that the cost increases almost linearly with the number of weapons. In figure 5 we have plotted the cost for various values of n_k. For this case we used $M = 2N = 20n_k$ weapons.

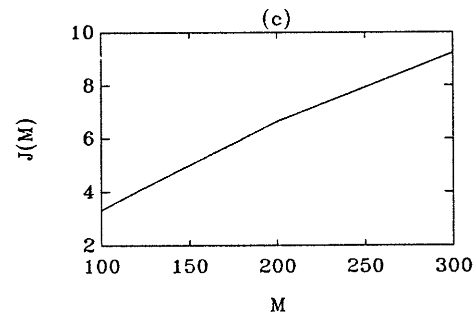

Figure 4: Plot of the cost and upper bound versus the number of weapons M

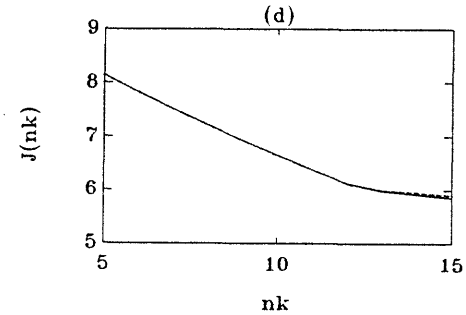

Figure 5: Plot of the cost and upper bound versus the number of targets per asset

3. THE DYNAMIC WTA PROBLEM

In this section we will consider the dynamic version of the WTA problem. We will see that a dynamic strategy increases the cost performance of the system at the expense of increased computational complexity. The complexity can be reduced by making approximations.

In this section we will consider the dynamic version of the Asset-Based Resource Allocation Problem. Because of the enormous complexity of this problem we will make the assumption that the kill probability of a weapon- target pair depends solely on the asset to which the target is aimed. We will also assume that the damage probability of a target depends solely on the asset to which the target is aimed. Under these assumptions one can show that for each stage and asset it is optimal to spread the weapons assigned to defend an asset evenly among the targets aimed for the asset. Therefore with these assumptions we can use the number of weapons assigned to the defense of an asset as the decision variable. This greatly reduces the dimensionality of the problem. We will only consider the case of two stages, however the results can easily be extended to more than two stages.

In this problem the results of the engagements of the first stage are observed before the assignments of the second stage are made. The problem is to choose the number of weapons to use in stage 1 as well as the assignment of these weapons to the targets. The objective is to maximize the expected total value of the assets which survive at the end of stage 2. Note that by the principal of optimality, the optimal static assignment will be used in stage 2.

3.1 Problem Statement

The following notation will be used.

$K \stackrel{\text{def}}{=}$ the number of assets,
$N \stackrel{\text{def}}{=}$ the number of targets at the start of stage 1,
$G_k(t) \stackrel{\text{def}}{=}$ the set of targets aimed for asset k in stage t,
$n_k(t) \stackrel{\text{def}}{=}$ the number of targets aimed for asset k in stage t,
$M \stackrel{\text{def}}{=}$ the number of weapons,
$m_t \stackrel{\text{def}}{=}$ the number of weapons to be used in stage t.
$W_k \stackrel{\text{def}}{=}$ the value of asset k,
$p_k(t) \stackrel{\text{def}}{=}$ the kill probability on a target aimed for asset k,
$X_k \stackrel{\text{def}}{=}$ the number of weapons assigned to defend asset k.

Note that for stage 1 the decision variables are m_1 and $\vec{X}$. Given $\vec{X}$ the individual target assignments can be obtained by spreading the weapons assigned to the defense of an asset as evenly as possible among the targets aimed for the asset. In the second stage one has to solve a static problem with $M - m_1$ weapons and with $n_k(2)$ targets aimed at asset k.

We will refer to the vector $\vec{n}(2)$ as the *state* of the system at the beginning of stage 2. The state of each asset evolves stochastically as follows. To simplify the expression we have left out the subscript k.

$$\Pr[n(2) = j | X = \chi] = \sum_{\ell=\underline{\ell}}^{\bar{\ell}} (1-p(1))^{\ell + j\lfloor \frac{\chi}{n(1)} \rfloor} [1-(1-p(1))^{\lceil \frac{\chi}{n(1)} \rceil}]^{\chi - n(1)\lfloor \frac{\chi}{n(1)} \rfloor - \ell} \times [1-(1-p(1))^{\lfloor \frac{\chi}{n(1)} \rfloor}]^{n(1)\lfloor \frac{\chi}{n(1)} \rfloor + n(1) + \ell - \chi - j}$$

for

$$j = 1, \ldots, n(1)$$

where

$$\underline{\ell} = \max\{j + \chi - n(1)(\lfloor \frac{\chi}{n(1)} \rfloor + 1), 0\}$$

and

$$\bar{\ell} = \min\{\chi - n(1)\lfloor \frac{\chi}{n(1)} \rfloor, j\}$$

The state evolution simply states that the number of targets which survive stage 1 is a random variable. The distribution of this random variable is obtained by convolving two binomial distributions. The success probability of one of these distributions is given by $(1-p(1))^{\lfloor \frac{\chi}{n(1)} \rfloor}$. The success probability of the other binomial distribution is given by $(1-p(1))^{\lceil \frac{\chi}{n(1)} \rceil}$. Let us denote the *optimal* cost for the static problem with state $\vec{n}(2)$ and m_2 weapons by $J_s^*(\vec{n}(2), m_2)$. The optimization problem may now be stated as

$$\max_{\{\vec{X} \in Z_+^K\}} J_d = \mathop{E}_{\{\vec{n}(2)\}} [J_s^*(\vec{n}(2), m_2)] \tag{8}$$

subject to the state evolution 8 and

$$|X_k| + m_2 = M.$$

The objective is the expectation over all possible states of the optimal second stage cost given that state. The constraint says that the number of weapons used in stage 1 ($|\vec{X}|$) plus those used in stage 2 (m_2) must be equal to the total number of weapons. One can see that even the statement of the problem is a formidable task even under the assumption that the kill probabilities are solely asset dependent.

The only decision variables over which the objective function is to be optimized are m_1 and $\vec{X}$, which is the number of weapons to be used in the first stage m_1 and the assignment of these weapons to assets $\vec{X}$. We will therefore denote the optimal cost for the case in which m_1 weapons are used in the first stage with assignment $\vec{X}$ by $J_d(m_1, \vec{X})$. The problem can therefore also be stated as:

$$\max_{\{m_1 \in Z_+\}} \left\{ \max_{\{\vec{X} \in Z_+^K\}} J_d(m_1, \vec{X}) \right\} \tag{9}$$

$$\text{subject to} \quad \sum_{k=1}^{K} X_k = m_1,$$

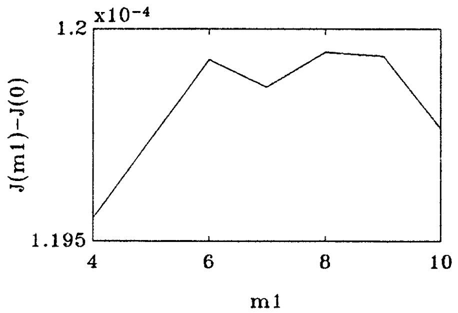

Figure 6: Plot of the expected two-stage cost versus the number of weapons used in stage 1.

$$\text{and} \qquad 0 \le m_1 \le M.$$

If we fix m_1 then the inner subproblem can be written as

$$\max_{\{\vec{X} \in Z_+^K\}} J_d(m_1, \vec{X}) \tag{10}$$

$$\text{subject to} \qquad \sum_{k=1}^{K} X_k = m_1.$$

This will be called the assignment subproblem. If we can solve the assignment subproblem then the original problem can be solved as follows. Let $\vec{X}^*$ denote the optimal assignment of the subproblem 10. Note that this optimal assignment depends on the value of m_1. However this is implicit in the solution since $\sum_{k=1}^{K} X_k = m_1$. The solution to the original problem may now be obtained by solving the following:

$$\max_{\{m_1 \in Z_+\}} J_d(m_1, \vec{X}^*) \tag{11}$$

$$\text{subject to} \qquad 0 \le m_1 \le M.$$

Each of the problems 10 and 11 will be considered separately.

3.2 Solution of the Dynamic WTA Problem

Our efforts will be concentrated on the solution of problem 10 since we will show that problem 11 has many local maxima and hence, in general, a global search will have to be done to obtain the optimal solution. We can also obtain an upper bound on the optimal cost. Details of this upper bound can be found in Hosein[4].

3.2.1 Optimization over m_1

Let us assume that we can solve the subproblem 10. In figure 6 we have plotted the cost $J_d(m_1, \vec{X}^*)$ versus m_1 for the case of $K = 2$, $n_1(2) = n_2(2) = 2$, $p_1(2) = p_2(2) = p_1(1) = p_2(1) = 0.8$, $W_1 = W_2 = 1$, and $M = 8$. For this case the optimal decision variables are $m_1^* = 4$, $X_1 = X_2 = 2$. Because the objective function has many local maxima, the global maximum can only be found by doing a global search.

3.2.2 Optimization over the Decision Variable $\vec{X}$

In this section we will consider the assignment subproblem 10. In this problem the number of weapons to be used in the first stage is fixed and the objective is to assign these weapons optimally. Recall that for the static version of this problem we were able to obtain a suboptimal algorithm but not an optimal one. In this section we will provide a suboptimal algorithm for the dynamic problem.

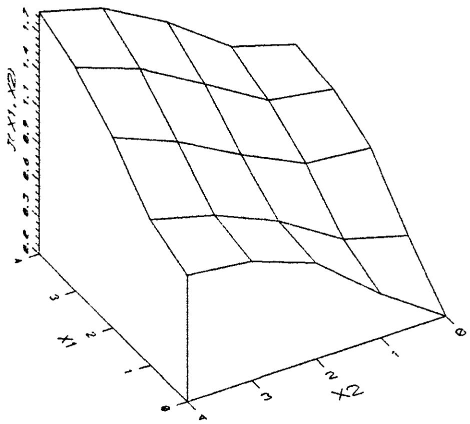

Figure 7: $J_d(X_1, X_2) vs. [X_1, X_2]$

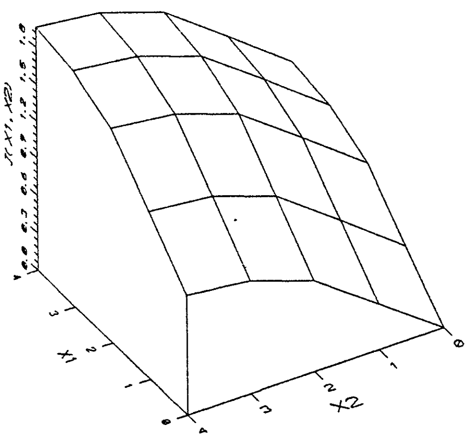

Figure 8: $\tilde{J}_d(X_1, X_2) vs. [X_1, X_2]$

The algorithm will be presented by applying it to a simple example. Consider the case in which $K = 2, \vec{n}(1) = [10, 10], \vec{W} = [1, 1], p_k(t) = 0.6$. In figure 7 we have plotted the function $J_d(X_1, X_2) vs. [X_1 X_2]$. Note that this function is non-concave. Furthermore it is non-separable with respect to the assets. We approximate J_d by a concave separable function $\tilde{J}_d$ as follows. Let $\tilde{J}_d(X_1, 0)$ be the hull of the function $J_d(X_1, 0)$. Similarly let $\tilde{J}_d(0, X_2)$ be the hull of the function $J_d(0, X_2)$. Finally let

$$\tilde{J}_d(X_1, X_2) = \tilde{J}_d(X_1, 0) + \tilde{J}_d(0, X_2).$$

The approximation $\tilde{J}_d$ is plotted in figure 8. Note that the function $\tilde{J}_d$ is concave and separable with respect to the assets. One can now use a greedy algorithm to obtain the optimal assignment of the problem with objective function $\tilde{J}_d$. This is the sub-optimal solution.

3.3 Numerical Results

Example 1

$M = 200, N = 100, K = 10, n_k = 10, p_k(t) = .6, V_k = 1, \pi_k = 1.$

STATIC STRATEGY:

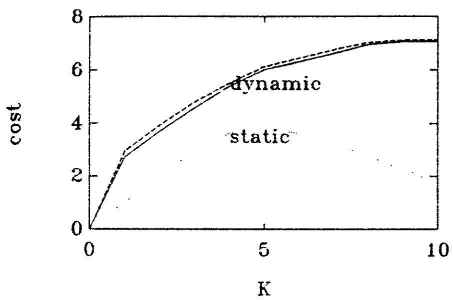

Figure 9: Expected cost vs. Number of defended Assets

Optimal static solution: $\vec{X}^* = [0,0,0,0,0,40,40,40,40,40]$.
Optimal static cost = 3.9.
DYNAMIC STRATEGY:
Solution obtained by algorithm:
$m_1 = 100$.
$\vec{X} = [10,10,10,10,10,10,10,10,10,10]$.
Lower bound on solution $\vec{X} = 7.1$
Upper bound on optimal solution = 7.9

Note that if a static strategy is used then 5 of the 10 assets are defended while if a dynamic strategy is used all 10 assets are defended. Also note that the performance of the dynamic strategy is almost twice as good as that of the static strategy. Finally note that the solution produced by our sub-optimal algorithm is close to optimal.

In figure 9 we have plotted the expected cost versus the number of assets defended for both the static and the dynamic strategies. Note that the dynamic strategy is less sensitive to the number of assets defended than the static strategy.

Example 2
$M = 100, N = 100, K = 10, n_k = 10, p_k(t) = .8, V_k = 1, \pi_k = 1$.
STATIC STRATEGY:
Optima static solution: $\vec{X}^* = [0,0,0,0,0,20,20,20,20,20]$.
Optimal static cost = 3.3
DYNAMIC STRATEGY:
Solution obtained by algorithm:
$m_1 = 70$.
$\vec{X} = [0,0,0,10,10,10,10,10,10,10]$.
Lower bound on solution $\vec{X} = 6.3$
Upper bound on optimal solution = 6.5

Note that although the number of weapons is small, in the dynamic strategy 7 of the assets are defended in the first stage. Also again note that our algorithm performs well.

Example 3
$M = 200, N = 100, K = 10, n_k = 10, V_k = 1, \pi_k = 1$.

Case 1:
$p_k(1) = .5, p_k(2) = .7$
Solution obtained by algorithm:
$m_1 = 90$.
$\vec{X} = [0,10,10,10,10,10,10,10,10,10]$.
Lower bound on solution $\vec{X} = 6.9$
Upper bound on optimal solution = 7.2
Case 2:

$p_k(1) = .7, p_k(2) = .5$
Solution obtained by algorithm:
$m_1 = 120$.
$\vec{X} = [20,20,10,10,10,10,10,10,10,10]$.
Lower bound on solution $\vec{X} = 7.7$
Upper bound on optimal solution = 8.5

Note that in case 1 less weapons are used in stage 1 because the kill probability in this stage is smaller than that in the second stage. Similarly, in case 2 more weapons are used in stage 1. Finally note that the performance obtained by using the more effective weapons in stage 1 is considerably better than that obtained by using the more effective weapons in stage 2.

4. CONCLUSIONS

We have seen that the Dynamic Weapon Allocation problem is considerably more difficult than the static one. However, the introduction of feedback can significantly improve effectiveness (by roughly a factor of two). By using simple approximations for the dynamic problem we can significantly reduce the computational complexity of the problem while maintaining the cost performance advantage over the static strategy. The algorithms we have proposed for both the static and dynamic problems are efficient and produce near optimal solutions.

This paper has been a progress report of our ongoing research. We plan to continue working on both analytical and numerical studies with the intent of providing an intuitive understanding of the problems.

ACKNOWLEDGMENT

This research was supported by the Joint Directors of Laboratories (JDL), Basic Research Group on C^3 Systems, under contract with the Office of Naval Research, ONR/N00014-85-K-0782 and ONR/N00014-84-K-0519.

REFERENCES

1. P. Hosein, J. Walton, and M. Athans, "Dynamic Weapon-Target Assignment Problems with Vulnerable C^3 Nodes," in *Proceedings of the 1988 Command and Control Symposium*, (Monterey, CA), June 1988.

2. J. Pothiawala, J. Papastavrou, and M. Athans, "Designing an Organization in a Hypothesis Testing Framework," in *Proceedings of the 1989 Command and Control Symposium*, (Washington, DC), June 1989.

3. J. Walton and M. Athans, "Strategies for Asset Defense with Precursor Attacks on the C^3 System," in *Proceedings of the 1989 Command and Control Symposium*, (Washington, DC), June 1989.

4. P. Hosein, *A Class of Dynamic Nonlinear Resource Allocation Problems.* PhD thesis, Massachusetts Institute of Technology, Cambridge, MA. (In progress).