Stochastic Games and Inverse Lyapunov Methods in Air Operations

William M. McEneaney [1] Kazufumi Ito [2]

Abstract

A mathematical representation of the air operations command and control problem is developed. The size of the problem implies a need for reduction to subproblems. Two of these subproblems are discussed here. First, the development of an approach to the generation of approximate optimal aircraft routing through a hostile region is given. Once this is established, a stochastic game is solved to determine the time-ordering of aircraft engagements with surface-to-air missile batteries, for the ultimate purpose of engaging a strategic target.

1 Introduction

The command and control (C^2) problem for air operations in the presence of antagonistic forces is considered. Specifically, the adversarial forces take the form of SAM (surface-to-air missle) batteries and geographically fixed strategic targets.

The seemingly natural problem formulation of a game was used. The problem is decomposed into geographically isolated regions. In each region, an approximate aircraft route to the strategic target is generated via a reverse Lyapunov technique. Once this is established, a zero-sum stochastic game approach is used to determine the air operations strategy. This may include multiple sorties to remove intervening SAMs. At the same time, the opponent's control includes the ability to turn the radars associated with the SAMs on and off. When on, they have a higher probability of damaging the aircraft as well as a higher probability of being damaged themselves. A hierarchical approach is taken to the problem of allocation of resources between the regions.

[1]Dept. of Mathematics, North Carolina State University, Raleigh, NC 27695–8205, USA, http://www4.ncsu.edu/~wmm/, wmm@eos.ncsu.edu, Research partially supported by AFOSR grant F08671-98-0-1098, DARPA grant F30602-99-2-0548, and NSF grant DMS-9971546.

[2]Dept. of Mathematics, North Carolina State University, Raleigh, NC 27695–8205, USA, kito@eos.ncsu.edu, Research partially supported by DARPA grant F30602-99-2-0548.

2 Model

One significant difficulty with the C^2 problem is the lack of an obvious model. We will present a model which seems "reasonable", and design objectives within the context of that model. Justification of the model is outside the scope of this paper. The objects which will be of interest here are aircraft (belonging to what will be termed the "blue" player), SAMs (belonging to the "red" player), and strategic targets (belonging to red).

We will reduce the state of the i^{th} (blue) aircraft at time t to a pair, $Y_i^A(t) = (D_i^A(t), X_i^A(t))$ where D_i^A will represent the health status of the aircraft, and X_i^A will represent its position. Note that since the scope of the C^2 problem is large, we will not model the dynamics of each aircraft in detail; we will not include velocity, attitude, mass and so forth as part of the state. When the problem is decomposed into separate subproblems below, we will abuse notation in the sense that in one subproblem, X_i^A will represent a position taking continuous values in $\mathbf{R}^2$, while for the other subproblem this will indicate position among a discrete set of alternatives; the meaning will be completely obvious by context. We will suppose that the health status take values in the discrete set $\{1, 2, 3, 4\}$ where 1 represents healthy, 2 and 3 represent various levels of damage (or need of maintenance), and 4 indicates that the aircraft has been destroyed.

We will assume similar state models for the SAMs. The i^{th} SAM state will be represented by the pair $Y_i^R(t) = (D_i^R(t), X_i^R(t))$ where D_i^R will represent the health status of the SAM, and X_i^R will represent its position. (Note that there exist both mobile and fixed-site SAMs.) Similar comments as those above can be made with regard to X_i^R. As for the health status of the SAMs, we let D_i^R take values in $\{1, 2, 3\}$ where 1 represents healthy, 2 represents damage (or need of maintenance), and 3 indicates that the SAM has been destroyed (not repairable). Lastly, we will take a similar model for the strategic targets, where the pair will be denoted as $Y_i^T(t) = (D_i^T(t), X_i^T(t))$ with $D_i^T(t) \in \{1, 2, 3\}$, where 1, 2 and 3 will have the same meaning as for the SAMs. Let the number of blue aircraft be N_A, the number of red SAMs be N_R, and the number of red strategic targets be N_T. Let $\vec{Y}^A = \{Y_i^A\}_{i=1}^{N_A}$, $\vec{Y}^R = \{Y_i^R\}_{i=1}^{N_R}$ and

$\vec{Y}^T = \{Y_i^T\}_{i=1}^{N_T}$. Throughout, we will use the convention of uppercase letters for the state processes and lowercase for values that the state process may take on, that is, $\vec{Y}^A(t) = \vec{y}^A$ indicates that the aircraft state process has the value $\vec{y}^A$ at time t.

The objective is not clearly defined in a mathematical sense. For blue, it may sometimes be to destroy a strategic target while minimizing damage to the aircraft; in other situations it may be more general attrition of both SAMs and targets. In order to simplify matters, we will assume here that both players are using the same objective function. That is, blue is trying to minimize the worst case (maximum) payoff, and red is trying to maximize their worst case (minimum) of the same payoff. The time-horizon over which these objectives should be met is not fixed. We choose to consider an exit cost, without running cost terms. Let τ be the exit time. We define the exit time to be the time when either: 1) all the blue aircraft have been destroyed or 2) the red strategic target(s) has(have) been destroyed and the surviving blue aircraft have returned to base. We let the set of states satisfying one of the exit conditions be denoted by $\mathcal{E}$. In order to capture the objective in a reasonably simple payoff function, one can consider, for instance, a linear payoff with parameters which can be varied depending on the value of the assets such as

$$\Psi(\vec{y}^A, \vec{y}^R, \vec{y}^T) \doteq \mu_A \Big[\sum_{i=1}^{N_A} d_i^A\Big] - \mu_R \Big[\sum_{i=1}^{N_R} d_i^R\Big] - \mu_T \Big[\sum_{i=1}^{N_T} d_i^T\Big] \tag{1}$$

where μ_A, μ_R, μ_T are the parameters. The presence of the expectation in the above equation is due to the fact that the dynamics of the health status of the objects will involve random outcomes of engagements and maintenance.

As mentioned in the introduction, the problem is being decomposed into subproblems. The dynamic models for the subproblems will be discussed below.

3 Optimal Routing Problem

The first step taken is to find the optimal routes to the target(s) for the aircraft while avoiding the SAMs. First we discuss the control problem formulation.

3.1 Control Problem Formulation

We formulate the optimal routing problem from the base $\bar{x}^A = (\bar{x}_1^A, \bar{x}_2^A)$ to the target $x^T \in \mathbf{R}^2$ as an optimal (exit) control problem;

$$\min_u \quad \int_0^\tau (1 + \sum_{i=1}^{N_R} \sigma_i \, \ell_i(X^A - x_i^R)) \, dt \tag{3.1}$$

subject to $X^A(0) = \bar{x}^A$, $X^A(\tau) = x^T$ and

$$\frac{d}{dt} X^A(t) = u(t), \quad |u(t)| \le 1, \tag{3.2}$$

where $X^A(t) \in \mathbf{R}^2$ is the location of an aircraft, $u(t) = (u_1(t), u_2(t))$ is the velocity control, and x_i^R is the i^{th} opponent SAM site with strength σ_i, $1 \le i \le N_R$. The loss function ℓ_i represents a loss due to flying close to the site i, and for example $\ell_i(x^A - x_i^R) = \frac{1}{|x^A - x_i^R|^2}$. That is, the optimal route is determined so that the sum of time and total loss is minimized. The optimal control of (3.1) is given by the feedback law [4]

$$u(t) = -\frac{V_{x^A}(X^A(t))}{|V_{x^A}(X^A(t))|} \tag{3.3}$$

where the value function V satisfies the Hamilton-Jacobi-Bellman equation

$$\begin{aligned} &-|V_{x^A}(x^A)| + \ell(x^A) = 0, \quad V(x^T) = 0 \\ &\text{with } \ell(x^A) = 1 + \sum_{i=1}^{N_R} \sigma_i \, \ell_i(x^A - x_i^R), \end{aligned} \tag{3.4}$$

Next, we describe the numerical method to HJ equation (3.4). Let $V_{i,j}$ denote an approximation of V at each grid point (x_i, y_j) which is uniformly distributed over a square domain Ω in $\mathbf{R}^2$. Let $h > 0$ be the stepsize and we define the backward and forward difference

$$(D_x^-)_{i,j} V = \tfrac{V_{i,j} - V_{i-1,j}}{h}, \quad (D_x^+)_{i,j} V = \tfrac{V_{i+1,j} - V_{i,j}}{h}$$

$$(D_y^-)_{i,j} V = \tfrac{V_{i,j} - V_{i,j-1}}{h}, \quad (D_y^+)_{i,j} V = \tfrac{V_{i,j+1} - V_{i,j}}{h}$$

We use the upwinding method of Godnov to discretize (3.4):

$$\begin{aligned} \Big(&[\max((D_x^-)_{i,j} V, -(D_x^+)_{i,j} V, 0)]^2 \\ &+ [\max((D_y^-)_{i,j} V, -(D_y^+)_{i,j} V, 0)]^2 \Big)^{\frac{1}{2}} = \ell_{i,j} \end{aligned} \tag{3.5}$$

where $\ell_{i,j} = \ell(x_i, y_j)$. We employ the fixed point iterate [5]: let $V_{i,j}^n$ denote the n-th iterate and we update $V_{i,j}^{n+1}$ by solving (3.5) for $V_{i,j}^{n+1}$ at each grid point, given $V_{i+1,j}^n$, $V_{i-1,j}^n$, $V_{i,j+1}^n$ $V_{i,j-1}^n$.

The exact step is given as

$$a_{i,j} = \min(V_{i-1,j}^n, V_{i+1,j}^n), \quad b_{i,j} = \min(V_{i,j-1}^n, V_{i,j+1}^n),$$

$$c_{i,j} = \ell_{i,j}^2, \quad s_{i,j} = c_{i,j} - (a_{i,j} - b_{i,j})^2$$

$$\begin{cases} V_{i,j}^{n+1} = \frac{1}{2}(a_{i,j} + b_{i,j}) + \sqrt{c_{i,j} + s_{i,j}} & \text{if } s_{i,j} \ge 0 \\ V_{i,j}^{n+1} = \min(a_{i,j} + b_{i,j}) + \sqrt{c_{i,j}} & \text{if } s_{i,j} < 0. \end{cases}$$

We have the boundary (exit) condition $V_{\bar{i},\bar{j}} = 0$ at the target grid and also we set $V_{i,j} = \infty$ at the boundary of Ω. The initial iterate can be set as $V_{i,j}^0 = |(x_i, y_j) - x^T|$.

3.2 Multi-Body Dynamic Formulation

We propose the following feedback law based on multi-body interaction dynamics, which can be implemented in real time. Let x_j^T be the target location with value w_j for $1 \leq j \leq N_T$. A route $X^A(t)$, $t \geq 0$ is determined as a solution to

$$\frac{d}{dt}X^A(t) = -\frac{W_{x^A}(X^A(t))}{|W_{x^A}(X^A(t))|}, \quad X^A(0) = \bar{x}^A \tag{3.6}$$

where the potential function W is given by

$$W(x^A) = \sum_{j=1}^{N_T} w_j\,|x^A - x_j^T| + \sum_{i=1}^{N_R} \sigma_i\, U(|x^A - x_i^R|),$$

e.g., $U(|x^A - x_i^R|) = \frac{1}{|x^A - x_i^R|}$. Thus, the force field W_{x^A} is given by

$$-W_{x^A}(x^A) = -\sum_{j=1}^{N_T} w_j\,\frac{x^A - x_j^T}{|x^A - x_j^T|} + \sum_{i=1}^{N_R} \sigma_i\,\frac{x^A - x_i^R}{|x^A - x_i^R|^3}.$$

Here the term $-\frac{x^A - x_j^R}{|x^A - x_j^R|}$ represents an attracting force to the target j and the term $\frac{x^A - x_i^R}{|x^A - x_i^R|^3}$ is for a repelling force from the SAM site i.

We can relate a closed loop system (3.6) to the optimal control problem (3.1)–(3.2) as follows. We define the performance index $\ell(x^A)$ by $\ell = |W_{x^A}|$. Note that if $\sigma_i = 0$ and $N^A = 1$, then $V = W = |x^A - x^T|$ and $u(t) = -\frac{X^A(t) - x^T}{|X^A(t) - x^T|}$ is optimal. $|W_x(x^A)|$ attains local minima and maxima at the same points as $\ell(x^A)$ defined by (3.4) does.

Similarly, we also construct a movement of SAMs as follows. We assume that they protect the targets while avoiding voids.

$$\frac{d}{dt}X_i^R(t) = \frac{\tilde{W}_{x_i^R}(X^A(t), X^R(t))}{|\tilde{W}_{x_i^R}(X^A(t), X^R(t))|} \tag{3.7}$$

where the potential function $\tilde{W}$ is given by

$$\tilde{W}(x^A, x^R) = -\sum_{j=1}^{N_T} w_j\,|x_i^R - x_j^T| + \sum_{i=1}^{N_R}\sum_{j=1}^{N_R} \tilde{\sigma}_{i,j}\,\tilde{U}(|x_i^R - x_j^R|) + \sum_{i=1}^{N_R} \sigma_i\, U(|x^A - x_i^R|)$$

e.g., $\tilde{U}(|x_i^R - x_j^R|) = \frac{1}{|x_i^R - x_j^R|}$. Thus the force field $\tilde{W}_{x^R}$ is given by

$$\tilde{W}_{x_i^R}(x^A, x^R) = -\sum_{j=1}^{N_T} w_j\,\frac{x_i^R - x_j^T}{|x^R - x_j^T|} + \sum_{j\neq i}^{N_R} \tilde{\sigma}_{i,j}\,\frac{x_i^R - x_j^R}{|x_i^R - x_j^R|^3} - \sigma_i\,\frac{x_i^R - x^A}{|x_i^R - x^A|}$$

with attracting force $-\frac{x_i^R - x_j^T}{|x_i^R - x_j^T|}$ to the target j and repelling forces $\frac{x_i^R - x_i^R}{|x_i^R - x_j^R|^3}$.

A combined closed loop dynamics (3.6)–(3.7) has the game theoretic interpretation that is similar to the one for the optimal control problem.

4 Discrete Stochastic Game

We consider the problem where a single strategic target is selected, and an approximate path from the blue base to that target has been generated. As discussed above, there may be one or more SAM sites intervening along this path. At this level, the positional dynamics will be specified only in a general way. Let the SAMs be indexed as $\{1, 2, 3, ..., N_R\}$. Let the aircraft position take values in the set $\mathcal{L} \doteq \{B, 1, 2, 3, ..., N_R, N_R + 1\}$ where B indicates the (blue) base and $N_R + 1$ indicates the (red) strategic target. We suppose a discrete time model where each time step occurs only when either an aircraft engages a SAM, an aircraft engages the target, or an aircraft returns to base. More than one such activity can occur at each step. The aircraft control for each aircraft, $U_i^A(t)$, must be specified at each time step. The set of possible values is $\mathcal{U} = \mathcal{L} \cup \{0\}$ where numbers between $U_i^A = 1$ and $U_i^A = N_R + 1$ indicate attack the corresponding red SAM or target, $U_i^A = B$ indicates return to base, and $U_i^A = 0$ indicates "do nothing". Note that the dynamics of the motion is simply $X_i^A(t+1) = U_i^A(t)$ when $U_i^A(t) \neq 0$ and $X_i^A(t+1) = X_i^A(t)$ when $U_i^A(t) = 0$. We place some restrictions on the allowable controls. The control actions will be organized into cycles of length, n_c. That is, each cycle will consist of n_c time steps. At the start of each cycle, all aircraft must be at the base. Consequently, we require $U_i^A(t) = B$ for all $i \leq N_A$ and all $t = kn_c - 1$ for all $k \geq 1$. We also require that for any $t = kn_c - 1$

if there exists i such that $X_i^A(t) = B$ and $D_i^A(t) = 1$, then there must be a $k \leq N_A$ with $D_k^A(t) \neq 4$ such that $U_k^A(t) \neq B$.

$$(CC)$$

Note that this last requirement forces at least some aircraft to engage red during each cycle for which there is a fully healthy aircraft.

It will be assumed for this subproblem that the SAMs cannot move during the duration of the game. The controls for the i^{th} SAM at (discrete) time t is $G_i^R(t)$, taking values in $\{0, 1\}$ where 0 indicates radar on and 1 indicates radar off. As mentioned in the introduction, when the radar is on, the probability of the SAM inflicting damage on the aircraft rises - as does the probabilty that the aircraft can inflict damage on the SAM.

The health status of each of the objects will transition according to a discrete–time Markov chain model. The transisiton probabilities will be state/control dependent. To simplify matters, assume that multiple aircraft can attack a single SAM, but that the aircraft need only engage one SAM at a time. At each time-step, where a SAM is under attack, we let the transition probabilty be given by the matrices P^{R01}, P^{R02}, P^{R11} and P^{R12} indicating the transition probabilities for the cases where a SAM with radar off is being attacked by a single aircraft, a SAM with radar off is being attacked by multiple aircraft simultaneously, a SAM with radar on is being attacked by a single aircraft, and a SAM with radar on is being attacked by multiple aircraft simultaneously, respectively. Of course, there are many more possibilities, but we consider only these for simplicity. If a SAM radar is on, and the SAM is not under attack during that time step, then we assume the SAM health status remains constant with probability one. Lastly, if a SAM site is off and not under attack, the health may improve through maintenance, with a tranistion probability given by P^{R00}. The state $d_i^R = 3$ will be an absorbing state for all the transistion matrices. In particular, maintenance cannot repair a SAM once it has entered state 3. The transition probabilities for the red target are the same as those for a SAM with radar off.

Let the corresponding probabilities for the aircraft during engagement be given by P^{A01}, P^{A02}, P^{A11} and P^{A12} where these stand for the same situations as those indicated for the SAMs above. We assume that the probability of transitioning to state 4 (down) is nonzero for all of the above matrices (i.e. that the last columns have no zero entries). We also allow the aircraft to undergo maintenance while at the base ($U_i^A(t) = B$), and let the transition probabilities be P^{A00}. For the aircraft, one must also consider the possibility of damage due to flying over SAMs with radars that are on while enroute from one point to the next. For instance, if $X_i^A(t) = 1$ and $X_i^A(t+1) = U_i^A(t) = 3$, and if SAMs 2 and 4 are between 1 and 3, then aircraft i could suffer damage while flying over each of the SAMs 2 and 4 – if they are on. We let the transition probability for aircraft health due to flying over a SAM that is on ($G_j^R(t) = 1$) and not destroyed ($D_j^R(t) \neq 3$) be P^{A1F} for each SAM that is flown over. In the above example, if SAM 3 is on and aircraft i is the only one attacking, then its transition probability for this time step is given by $P^{A1F}P^{A1F}P^{A11}$. Lastly, the destroyed/down state will be absorbing for all the transition matrices including P^{A00}.

Here we will consider a simplified information pattern that is chosen to mimic the real world situation in a rather loose way. Specifically, we will consider the game where at each time step blue chooses its control given the current state, and then red chooses its control given the current state *plus* the control choice for blue at the current time. In other words, we are interested here in an upper value (recall blue is minimizing and red maximizing). Let the value function for this game be denoted by $V(\vec{y}^A, \vec{y}^R, \vec{y}^T)$. Since it is quite standard, we do not include a proof of the DPE (dynamic programming equation) which is given as follows.

Theorem 4.1 *The value function satisfies*

$$V(\vec{y}^A, \vec{y}^R, \vec{y}^T) = \min_{\vec{u}^A \in \mathcal{U}^{N_A}} \max_{\vec{g}^R \in \{0,1\}^{N_R}} \Big[E\Big\{ V(\vec{Y}^A(1), \vec{Y}^R(1), \vec{Y}^T(1)) \,\Big|\, \vec{Y}^A(0) = \vec{y}^A, \vec{Y}^R(0) = \vec{y}^R, \vec{Y}^T(0) = \vec{y}^T, \Big\}\Big]$$
$$\doteq \min_{\vec{u}^A \in \mathcal{U}^{N_A}} \max_{\vec{g}^R \in \{0,1\}^{N_R}} \mathcal{G}^{\vec{u}^A, \vec{g}^R}[V](\vec{y}^A, \vec{y}^R, \vec{y}^T).$$

We will now indicate how this value function can be obtained through repeated appplication of the backward dynamic programming operator. First however, we will need the following lemma which essentially implies that there is a positive probability of reaching the absorbing states in a fixed number of steps.

Lemma 4.2 *There exists $n < \infty$ and $\delta > 0$ such that for any sequence of controls for blue and red*

$$P\Big[(\vec{Y}^A(t+n), \vec{Y}^R(t+n), \vec{Y}^T(t+n)) \in \mathcal{E} \,\Big|\, (\vec{Y}^A(t), \vec{Y}^R(t), \vec{Y}^T(t)) = (\vec{y}^A, \vec{y}^R, \vec{y}^T) \Big] \geq \delta$$

for any $(\vec{y}^A, \vec{y}^R, \vec{y}^T)$ where we recall $\mathcal{E}$ was the exit set.

Proof: Let $t_1 = \min\{s > t : s = kn_c + 1 \text{ for some nonnegative integer} k\}$. Then, by condition (CC), there exists $i_1 \leq N_A$ such that $X_{i_1}^A(t_1) \neq B$, and consequently, there exists $\delta_1 > 0$ (dependent on the choice of transition matrices) such that $P(D_{i_1}^A(t_1) = 4) \geq \delta_1$. Let $\Omega_1 \subseteq \Omega$ (the sampls space) be given by $\Omega_1 = \{\omega \in \Omega : D_{i_1}^A(t_1) = 4\}$. For points in Ω_1 such that $(\vec{Y}^A(t_1), \vec{Y}^R(t_1), \vec{Y}^T(t_1)) \notin \mathcal{E}$ (where not all the aircraft are down), let $t_2 = \min\{s > t_1 : s = kn_c + 1 \text{ for some nonnegative integer } k\}$. Then again by condition (CC), there exists $i_2 \leq N_A$ such that $X_{i_2}^A(t_2) \neq B$, and consequently, $P(\{\omega \in \Omega_1 : D_{i_2}^A(t_2) = 4\}) \geq \delta_1$. Since state 4 is absorbing, This implies that $P(D_{i_1}^A(t) = 4,\, D_{i_2}^A(t) = 4) \geq \delta_1^2$ for all $t \geq t_2$. Proceeding inductively, one finds that by, at most, time $t+n_cN_A$, the state is in E with probability no less than $\delta_1^{N_A}$. ∎

Define the backward dynamic programming (DP) algortihm as follows. Let the terminal value be

$$W(0, \vec{y}^A, \vec{y}^R, \vec{y}^T) = \begin{cases} \Psi(\vec{y}^A, \vec{y}^R, \vec{y}^T) & \text{if } (\vec{y}^A, \vec{y}^R, \vec{y}^T) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases}$$

(We remark that the choice of 0 is irrelevant.) Given $W(k,\cdot)$, one computes $W(k-1,\cdot)$ by the backward dynamic programming operator given by

$$W(k-1,\vec{y}^A,\vec{y}^R,\vec{y}^T) = \min_{\vec{u}^A\in\mathcal{U}^{N_A}} \max_{\vec{g}^R\in\{0,1\}^{N_R}} \Bigg[\mathrm{E}\Big\{ W(k,\vec{Y}^A(1),\vec{Y}^R(1),\vec{Y}^T(1)) \;\Big|\; \vec{Y}^A(0)=\vec{y}^A,\vec{Y}^R(0)=\vec{y}^R,\vec{Y}^T(0)=\vec{y}^T, \Big\}\Bigg]$$
$$\doteq \min_{\vec{u}^A\in\mathcal{U}^{N_A}} \max_{\vec{g}^R\in\{0,1\}^{N_R}} \mathcal{G}^{\vec{u}^A,\vec{g}^R}[W(k,\cdot)](\vec{y}^A,\vec{y}^R,\vec{y}^T)$$

if $(\vec{y}^A,\vec{y}^R,\vec{y}^T) \notin \mathcal{E}$ and $W(k-1,\vec{y}^A,\vec{y}^R,\vec{y}^T) = \Psi(\vec{y}^A,\vec{y}^R,\vec{y}^T)$ otherwise.

Lemma 4.3 *This backward dynamic programming propagation operator is a contraction.*

Proof: Once one has Lemma 4.2, the proof of this lemma is a minor variation of standard results, but in this case for a game with an exit criterion. (See, for instance, [2], [3] for similar results.) We will simply indicate some of the main points. Let W_1 and W_2 be given by the backward DP with possibly different conditions at $k=0$. For simplicity, use the notation $\vec{y} \doteq (\vec{y}^A,\vec{y}^R,\vec{y}^T)$. Note that (for $k<0$)

$$\begin{aligned} W_1(k,\vec{y}) - W_2(k,\vec{y}) &= \min_{\vec{u}^A\in\mathcal{U}^{N_A}} \max_{\vec{g}^R\in\{0,1\}^{N_R}} \mathcal{G}^{\vec{u}^A,\vec{g}^R}[W_1(k+1,\cdot)](\vec{y}) \\ &\quad - \min_{\vec{u}^A\in\mathcal{U}^{N_A}} \max_{\vec{g}^R\in\{0,1\}^{N_R}} \mathcal{G}^{\vec{u}^A,\vec{g}^R}[W_2(k+1,\cdot)](\vec{y}). \end{aligned}$$

Choose u_1^A to be $\frac{\varepsilon}{n}$-optimal for W_1 and then choose g_1^R to be $\frac{\varepsilon}{n}$-optimal for W_2 given the same control u_1^A as used for W_1. Then

$$\begin{aligned} W_1(k,\vec{y}) - W_2(k,\vec{y}) \\ \le \mathcal{G}^{\vec{u}_1^A,\vec{g}_1^R}[W_1(k+1,\cdot) - W_2(k+1,\cdot)](\vec{y}) + \frac{2\varepsilon}{n}. \end{aligned}$$

Repeating this process, one finds that (for $k<-n$) and proper choice of feedback controls,

$$\begin{aligned} W_1(k,\vec{y}) - W_2(k,\vec{y}) \\ \le \prod_{m=1}^{n} \{\mathcal{G}^{\vec{u}_m^A,\vec{g}_m^R}\}[W_1(k+n,\cdot) - W_2(k+n,\cdot)](\vec{y}) + 2\varepsilon \end{aligned}$$

where we are using the $\prod$ notation to indicate operator composition. Alternatively, one may write this as

$$W_1(k,\vec{y}) - W_2(k,\vec{y}) \le \sum_{\vec{z}\notin\mathcal{E}} \Big\{ [W_1(k+n,\vec{z}) - W_2(k+n,\vec{z})] \cdot P^n_{\vec{y},\vec{z}}(\{\vec{u}_m^A\}_{i=1}^n, \{\vec{g}_m^R\}_{i=1}^n) \Big\} + 2\varepsilon$$

where this last term indicates the probability of transitioning from $\vec{y}$ to $\vec{z}$ in n steps given the feedback control processes specified in the arguments. Using symmetry and the Lemma 4.2, one obtains

$$\begin{aligned} |W_1(k,\vec{y}) - W_2(k,\vec{y})| \\ \le \max_{\vec{z}}\{|W_1(k+n,\vec{z}) - W_2(k+n,\vec{z})|\} \\ \sum_{\vec{z}\notin\mathcal{E}} P^n_{\vec{y},\vec{z}}(\{\vec{u}_m^A\}_{i=1}^n, \{\vec{g}_m^R\}_{i=1}^n) + 2\varepsilon \\ \le \max\{|W_1(k+n,\vec{z}) - W_2(k+n,\vec{z})|\}(1-\delta) + 2\varepsilon. \end{aligned}$$

This then yields

$$\begin{aligned} \|W_1(k,\cdot) - W_2(k,\cdot)\|_\infty \\ \le (1-\delta)\|W_1(k+n,\cdot) - W_2(k+n,\cdot)\|_\infty. \end{aligned}$$

■

The proof of convergence is also standard, and so we state the result without proof.

Theorem 4.4 $W(k,\vec{y}^A,\vec{y}^R,\vec{y}^T)$ *converges to the value function* $V(\vec{y}^A,\vec{y}^R,\vec{y}^T)$ *as* $k \downarrow -\infty$ *for all points in the state space.*

We remark that since the controls spaces are finite, the controls actually converge in a finite number of steps.

5 Reducing the Computations

The above algorithm for the computation of the value function (and corresponding control policies) suffers from the curse of dimensionality typical for DP algorithms. Specifically, notice that computation of $\mathcal{G}^{\vec{u}^A,\vec{g}^R}[W(k,\cdot)](\vec{y})$ may require summing the product of $W(k,\vec{z})$ and $P^{\vec{u}^A,\vec{g}^R}_{\vec{y},\vec{z}}$ over all possible values of $\vec{z}$ for each point $\vec{y}$. More specifically, the computations for $W(k-1,\vec{y})$ (for each $\vec{y}$) require $O(4^{N_A}(N_R+N_T+1)^{N_A}3^{N_R}3^{N_T})$ operations, even without optimization over blue and red control policies. We will discuss one of the methods being used to reduce these computation costs. The method will involve an approximation of W at each step. The result will be that the computational costs *per $\vec{y}$ point* will be reduced from the above exponential growth in the number of dimensions to only linear growth in the number of dimensions. This is a tremendous reduction in computational costs which makes the difference between feasibility and infeasibility of computation for low-dimensional problems. The growth in the number of points at which we must evaluate W remains exponential in the number of dimensions of course.

We introduce the following operator which is essentially an approximation operator for the value function or DP iterates around any given point $\vec{y}$. In order to reduce the notation, we will consider a simplified state space where $\vec{y} = (y_1, y_2, y_3)$ with $y_1 \in \{1,2,3,4\}$ and $y_2, y_3 \in \{1,2,3\}$. This will reduce notation without losing the flavor of the method. Define the matrices A^i for $i = 1,2,3$ given by $A^i_{j,k} = 1$ if $j = k = i$ and $A^i_{j,k} = 0$ otherwise. Then, given $\vec{y}$, define the approximation operator for approximation around $\vec{y}$ by

$$\mathcal{H}_{\vec{y}}[V(\cdot)](\vec{z}) \doteq \begin{cases} \left[\frac{1}{\sum_{i=1}^{3} |z_i - y_i|}\right] \cdot \sum_{i=1}^{3} \left[|z_i - y_i| \right. \\ \qquad \left. V(A_i(\vec{z} - \vec{y}) + \vec{y})\right] & \text{if } \vec{z} \neq \vec{y} \\ V(\vec{y}) & \text{if } \vec{z} = \vec{y}. \end{cases}$$

The operator is essentially an approximation operator where convex combinations are used to approximate V for states which are not directly along a basis direction from the point around which V is being approximated. Although we will not discuss the error analysis here, we note that of course the appropriateness of an approximator of this form depends critically on the nature of the value function itself which, in turn, depends on the choice of terminal payoff, Ψ. Recall that since the problem is rather loosely defined, we have great freedom in the choice of Ψ. Now, note that the approximation operator is a nonexpansive map for any $\vec{y}$. The backward DP operator of the previous section will now be replaced by the approximate backward DP operator given by

$$W(k-1, \vec{y}) \doteq \min_{\vec{u}^A \in \mathcal{U}^{N_A}} \max_{\vec{g}^R \in \{0,1\}^{N_R}} \mathcal{G}^{\vec{u}^A, \vec{g}^R} [\mathcal{H}_{\vec{y}}[W(k, \cdot)](\cdot)](\vec{y})$$

if $\vec{y} \notin \mathcal{E}$ and $W(k-1, \vec{y}) = \Psi(\vec{y})$ otherwise. Using the nonexpansivity of this approximation operator and the contraction property of the backward DP, one can obtain the following result in a straightforward manner similar to that of the previous section.

Theorem 5.1 *The approximate backward DP operator is a contraction, and the corresponding iterates converge to a fixed point of the operator.*

Lastly, we indicate the promised reduction in computation via the approximation. Recall that each of the transitions is independent. Suppose for this simplified problem that the transition matrices for y_1, y_3, y_3 are given by P^1, P^2, P^3 respectively, where we are suppressing the dependence of each P on the states and controls. (Note that in this simplified problem, we have actually eliminated that position state for the aircraft.) Then the approximate backward DP takes the form

$$\begin{aligned} & W(k-1, \vec{y}) && (2) \\ & \doteq \min_{\vec{u}^A \in \mathcal{U}^{N_A}} \max_{\vec{g}^R \in \{0,1\}^{N_R}} W(k, \vec{y}) P^1_{y_1,y_1} P^2_{y_2,y_2} P^3_{y_3,y_3} \\ & \quad + \sum_{z_1=1}^{4} W(k, z_1, y_2, y_3) Q^1_{|z_1-y_1|} P^1_{y_1,z_1} \\ & \quad + \sum_{z_2=1}^{3} W(k, y_1, z_2, y_3) Q^2_{|z_2-y_2|} P^2_{y_2,z_2} \\ & \quad + \sum_{z_3=1}^{3} W(k, y_1, y_2, z_3) Q^3_{|z_3-y_3|} P^3_{y_3,z_3} && (3) \end{aligned}$$

where

$$Q^1_{|z_1-y_1|} \doteq \frac{\sum_{z_2=1}^{3} |z_2 - y_2| P^2_{y_2,z_2} + \sum_{z_3=1}^{3} |z_3 - y_3| P^3_{y_3,z_3}}{\sum_{i=1}^{3} |z_i - y_i|}$$

with analogous definitions for $Q^2_{|z_2-y_2|}$ and $Q^3_{|z_3-y_3|}$. Note that these Q^i may be pre-computed. Thus the approximate DP (3) has only linear growth in the compuations which must be performed at each step (per point in the state space).

References

[1] T. Basar and P. Bernhard, **$\mathbf{H}_\infty$–Optimal Control and Related Minimax Design Problems**, Birkhäuser (1991).

[2] D.P. Bertsekas, **Dynamic Programming, Deterministic and Stochastic Models**, Prentice Hall, (1976).

[3] J. Filar and K. Vrieze, **Competitive Markov Decision Proceses**, Springer (1997).

[4] W.H. Fleming and H.M. Soner, **Controlled Markov Processes and Viscosity Solutions**, Springer (1991).

[5] E.Rouy and A.Tourin, A viscosity solutions approach to shape-from-shading, SIAM Numer. Anal. 29 (1992), 867-884.