Optimal Constrained Retreat within the Turret Defense Differential Game*

Alexander Von Moll[1,2] and Zachariah E. Fuchs[2]

***Abstract*— We extend the Turret Defense Differential Game by considering the possibility for the mobile Attacker to retreat to a particular region of the state space in lieu of engaging the Turret. In this case, the Turret cooperates with the Attacker by escorting the Attacker to the retreat surface, thereby minimizing its cost. Along the retreat trajectory, the cost associated with the Value of the Game of Engagement must be greater than the cost associated with retreating. We refer to this scenario as Optimal Constrained Retreat, wherein the aforementioned requirement is imposed as a path constraint. The optimality conditions are developed herein. We specify a boundary value problem with fixed initial state and compute the optimal trajectory numerically via backwards shooting. The corresponding trajectory contains a constrained arc wherein the path constraint associated with the Game of Engagement becomes active.**

I. Introduction

The Turret Defense Differential Game (TDDG) is played between a mobile Attacker, moving with simple motion, and an immobile Turret with bounded turn rate. In the Game, the Attacker seeks to collide with the Turret while avoiding the gaze of the Turret; the Turret seeks to impose a large cost on the Attacker's approach by aiming at it. The Attacker controls its instantaneous heading while the Turret controls its instantaneous turn rate. In this paper, we consider a scenario which layers on a higher-level decision for each player. Here, the Attacker must decide between engaging the Turret in a collision course or moving to a pre-determined retreat boundary, and the Turret must decide between activating its defenses or escorting the Attacker to the retreat boundary. In some differential games, such as pursuit-evasion, the higher-level decision involves determining task assignments resulting in smaller instances of the original game [1]. In contrast, the higher-level decision (to engage or retreat) yields either a differential game wherein the two players oppose one another, or an optimal control problem wherein the players actually cooperate [2]. We thus refer to the TDDG as the Game of Engagement (GoE).

The scenario we consider has application to defense against risk-conscious attackers (e.g. vehicles, or otherwise expensive munitions) by a stationary platform. The Turret itself may represent a weapons platform of some kind or even a surveillance asset seeking to steer its sensor to maximize observation of the incoming Attacker. Generally speaking, the methodology itself is important because real-world conflicts of warfare rarely entail a single well-defined objective. Therefore, the ability to systematically consider multiple layers of decision-making and control available to both sides is highly desirable.

In [3], the TDDG was introduced and solved (i.e. the state space was filled with equilibrium trajectories) and three singular surfaces were analyzed: the Defender (Turret) Universal Surface ($\alpha = 0$), the Defender Dispersal Surface ($\alpha = \pi$), and the Attacker Dispersal Surface wherein the Attacker chooses from a direct and indirect route to termination. There, the Game terminated when the Attacker came within some capture distance of the Turret and the players' utilities were zero-sum. In [2] a simpler game of a mobile Attacker and stationary Defender (with no turn dynamics) was used to illustrate a framework for engage or retreat scenarios in which there is a higher level decision to "play" the game, resulting in interception of the Defender, or to retreat. This meta-game is made interesting when the Defender is rewarded if the Attacker retreats and the Attacker is penalized for retreating. We solve this problem by breaking it into two subproblems: the GoE and optimal constrained retreat (OCR). Using this method, it was shown that switching intents (from retreat to engage, or vice versa) within the engage or retreat game does not yield a better utility [2]. To ensure that switching to engagement does not yield a better utility while retreating, a path constraint is imposed.

In previous work [3], the TDDG (GoE) was solved, but its solution is numerical. In this work, we consider the TDDG in the context of the engage or retreat framework and focus on OCR. Thus, in order to impose the path constraint, we must perform a numerical interpolation of the GoE solution along the OCR trajectory. The main contributions of this paper are (1) a numerical interpolation scheme of the TDDG's solution (2) derivation of the first order necessary conditions for optimality for the OCR problem, (3) specification of the BVP for when the constraint becomes active along the trajectory, and (4) a process for solving the BVP. Section II contains a formal description of the OCR problem. Section III contains the derivation of the first order necessary conditions for optimality, specification of the BVP, and special considerations for the Turret's turn control. Section IV contains the simulation results for a particular initial condition, and Section V concludes the paper with some remarks on future work.

*The views expressed in this paper are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

[1]Control Science Center of Excellence, Air Force Research Laboratory, 2210 8th St. WPAFB, OH 45433 alexander.von_moll@us.af.mil

[2]Department of Electrical Engineering and Computer Science, University of Cincinnati, Cincinnati, OH 45221

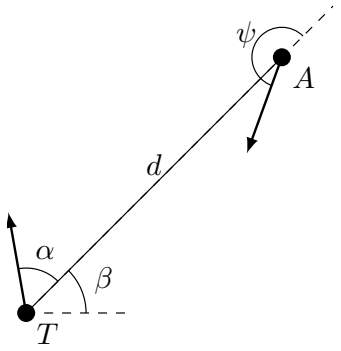

Fig. 1. Coordinate system [4]

II. Problem Description

Given a stationary Turret with bounded turn rate and a mobile Attacker, moving with simple motion, we seek a trajectory which moves the system state to the retreat surface, while minimizing cost, subject to a path constraint. The coordinate system is depicted in Fig. 1 and we define the Turret's position to be the origin of the (x, y)-plane. The dynamics of the system are a modified version of the dynamics presented in [3] wherein the natural parameters are consolidated into relative turret effectiveness, ρ, and relative time cost,

$$f(\mathbf{x}, \mathbf{u}, t) = \dot{\mathbf{x}} = \begin{bmatrix} \dot{d} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} \cos\psi \\ \omega - \frac{1}{d}\sin\psi \\ \frac{1}{d}\sin\psi \end{bmatrix}, \ \omega \in [-\rho, \rho], \tag{1}$$

where the Attacker controls ψ, its instantaneous heading angle relative to the line-of-sight from the Turret, and the Turret controls ω, its turn-rate [4].

We define the following boundary conditions,

$$\phi(\mathbf{x}(t_0), \mathbf{x}(t_f)) = \begin{bmatrix} d(t_0)\cos\beta(t_0) + 5 \\ d(t_0)\sin\beta(t_0) - 20 \\ \alpha(t_0) - \beta(t_0) + \frac{\pi}{2} \\ d(t_f)\sin\beta(t_f) - y_R \end{bmatrix} = 0 \tag{2}$$

with $t_0 = 0$ and t_f free. Note that in the natural (x, y)-plane, the first two constraints can be expressed as $x(0) = -5$ and $y(0) = 20$, respectively, while the last can be expressed $y(t_f) = y_R$. Thus, the retreat region, in this case is $y \leq y_R$. Here, we set $y_R = -20$. The initial α constraint stipulates that the Turret's initial global look angle is $\frac{\pi}{2}$ (i.e. along the positive y axis).

The cost functional is also slightly modified from that of [3],

$$J = \int_{t_0}^{t_f} \left[\theta\left(\frac{1}{2}(1 + \cos\alpha)\right) + c\right] \mathrm{d}t + \Phi(\mathbf{x}(t_f), t_f), \tag{3}$$

where $\theta \in [0, 1]$ is an additional Turret control representing some kind of rate of fire. The terminal Value function,

$$\Phi(\mathbf{x}(t_f), t_f) = P, \tag{4}$$

where $P > 0$ is a constant, penalizes the Attacker for retreating instead of engaging. The results herein are based on setting $\rho = 0.05$, $c = 0.01$, and $P = 8$. The settings for ρ and c match the settings used in [3]. For the optimal constrained retreat, the Turret and Attacker cooperate to minimize the cost in (3). Thus the Value function is defined as,

$$V_R(\mathbf{x}(t)) = \min_{\omega,\theta} \min_{\psi} J = \min_{\omega,\theta,\psi} J. \tag{5}$$

Similarly, the optimal controls are defined as,

$$u^* = (\omega^*(t), \theta^*(t), \psi^*(t)) = \underset{\omega(t),\theta(t),\psi(t)}{\arg\min} J. \tag{6}$$

The path constraint is defined as,

$$g(\mathbf{x}) = V_E(\mathbf{x}(t)) - V_R(\mathbf{x}(t)) \geq 0, \ \forall t \in [t_0, t_f], \tag{7}$$

where V_E is the Value function associated with the GoE. Let L be the integrand of the cost functional and suppose an additional state component, $l(t)$, is appended to $\mathbf{x}$ where,

$$\dot{l}(t) = -L, \ l(t_f) = 0 \tag{8}$$

which represents the remaining integral cost-to-go. Then the retreat Value function may be written,

$$\begin{aligned} V_R(\mathbf{x}(t)) &= P + \int_t^{t_f} L \, \mathrm{d}t \\ &= P - \int_t^{t_f} \dot{l}(t) \, \mathrm{d}t \\ &= P - l(t_f) + l(t) \\ &= P + l(t) \end{aligned} \tag{9}$$

In the case of backwards shooting, $l(t)$ may be easily computed, making the computation of $V_R(\mathbf{x})$ trivial.

Due to the path constraint, (7), it is necessary to solve the GoE so that $V_E(\mathbf{x})$ may be known for every point in the state space. An analytical solution of the GoE, however, is not available and a numerical solution must suffice. For a particular $\mathbf{x}$, V_E may be obtained by solving the associated boundary value problem via indirect backwards shooting. That approach, however, is infeasible for the purposes of the present work because the path constraint will need to be evaluated at *every* point along the trajectory as we numerically integrate. Instead, indirect backward shooting is used to fill the state space with equilibrium trajectories which are then sampled to generate a large set of data,

$$D = \{(d_i, \alpha_i, \sigma_{d_i}, \sigma_{\alpha_i}, V_{E_i}(d_i, \alpha_i))\}, \tag{10}$$

where each element is a tuple comprised of the state, GoE adjoints, and Value associated with starting in this state (for the GoE). The GoE adjoints are used in indirect optimal control analysis in Section III. Fig. 2 shows $V_E(d, \alpha)$ (i.e. the last column of D) for each of the sample points.

We compute $V_E(\mathbf{x})$ (and σ_d and σ_α) via a k-nearest neighbor (kNN) search with $k = 3$ and take a distance weighted average of the neighbors. The kNN search is made viable through the usage of the efficient *NearestNeighbors* package for the Julia programming language [5].

The difficulty in sampling the saddle-point equilibrium trajectories for the GoE lies in handling the singular surfaces. As mentioned previously, there are three singular

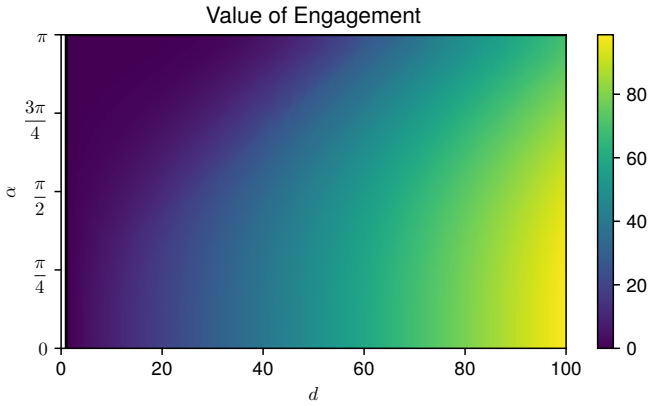

Fig. 2. Value of engagement, $V_E(\mathbf{x})$

surfaces in the GoE [3]: the Turret's Universal Surface (at $\alpha = 0$), the Turret's Dispersal Surface (at $\alpha = \pi$), and the Attacker's Dispersal Surface (ADS), which cannot be described analytically. To produce Fig. 2, the state space ($0 \leq \alpha \leq \pi$ and $1 \leq d \leq 100$) can be divided into 5 distinct regions: 1) trajectories with $\alpha_f \neq 0$ above the envelope of the ADS, 2) trajectories with $\alpha_f \neq 0$ below the envelope of the ADS, 3) "direct" trajectories emanating from the ADS, 4) "indirect" paths emanating from the ADS, and 5) trajectories in which $\alpha_f = 0$ (trajectories ending on the Turret's Universal Surface).

With all of the above definitions in place, we now formally state the problem definition:

$$\begin{aligned} &\min_{\omega(t),\theta(t),\psi(t)} J \\ &\text{s.t. } \phi(\mathbf{x}(t_0),\mathbf{x}(t_f)) = 0, \\ &\quad g(\mathbf{x}(t)) \geq 0 \quad \forall t \in [0,t_f] \end{aligned} \tag{11}$$

III. Methodology

A. Optimality Conditions

We begin by developing the first order optimality conditions for problem (11). The trajectory constraint, g, is transformed to a control constraint, h, by differentiating with respect to time,

$$\begin{aligned} h(\mathbf{x}) = \frac{\mathrm{d}g}{\mathrm{d}t} &= \frac{\mathrm{d}V_E}{\mathrm{d}t} - \frac{\mathrm{d}l}{\mathrm{d}t} \\ &= \frac{\partial V_E}{\partial \mathbf{x}}\dot{\mathbf{x}} + L \\ &= \sigma\dot{\mathbf{x}} + L \end{aligned} \tag{12}$$

where $\sigma = \begin{bmatrix}\sigma_d & \sigma_\alpha & \sigma_\beta\end{bmatrix}^\top$ are the adjoint variables associated with the GoE. When the path constraint is active, $g(\mathbf{x}) = 0$, and in order to remain on the constraint $h(\mathbf{x}) = 0$ as well. Thus when transitioning from an unconstrained arc to a constrained arc (or vice versa) $h(\mathbf{x})$ must be zero; we refer to this as the tangency condition. The Hamiltonian may be written,

$$H = \lambda\dot{\mathbf{x}} + \mu h + L, \tag{13}$$

where μ is the adjoint variable associated with the path constraint derivative, h, and

$$\mu(t) = \begin{cases} 0 & g(\mathbf{x}(t)) > 0 \\ > 0 & g(\mathbf{x}(t)) = 0. \end{cases} \tag{14}$$

Substituting (12) into (13) gives,

$$\begin{aligned} H &= \lambda\dot{\mathbf{x}} + \mu(\sigma\dot{\mathbf{x}} + L) + L \\ &= (\lambda + \mu\sigma)\,\dot{\mathbf{x}} + (1+\mu)\,L. \end{aligned} \tag{15}$$

Note that $\sigma_\beta = 0$ for all time [3]. The optimal adjoint dynamics are obtained by differentiating (15) w.r.t. each state component and substituting the state dynamics, (1), and $\sigma_\beta = 0$,

$$\dot{\lambda}_d = -\frac{\partial H}{\partial d} = (\lambda_\beta - \lambda_\alpha - \mu\sigma_\alpha)\,\frac{1}{d^2}\sin\psi \tag{16}$$

$$\dot{\lambda}_\alpha = -\frac{\partial H}{\partial \alpha} = \theta\,(1+\mu)\,\frac{1}{2}\sin\alpha \tag{17}$$

$$\dot{\lambda}_\beta = -\frac{\partial H}{\partial \beta} = 0. \tag{18}$$

The fact that $\dot{\lambda}_\beta = 0$ comes from the fact that $\frac{\partial\dot{\mathbf{x}}}{\partial\beta} = 0$ and $\frac{\partial L}{\partial\beta} = 0$ and implies that $\lambda_\beta(t)$ is constant.

The optimizing controls are obtained via Pontryagin's Minimum Principle.

$$\begin{aligned} \theta^* &= \arg\min_\theta H \\ &= \arg\min_\theta (\lambda\dot{\mathbf{x}} + \mu\sigma) + (1+\mu)\left[\theta\frac{1}{2}(1+\cos\alpha) + c\right] \end{aligned}$$

Since $\mu \geq 0$, the term $(1+\mu)$ must be positive, which implies,

$$\theta^*(t) = 0. \tag{19}$$

It is clear that the Turret should have its rate-of-fire set to zero for optimal constrained retreat. As a result,

$$\dot{\lambda}_\alpha = 0. \tag{20}$$

For the Turret's optimal turn control, we have,

$$\begin{aligned} \omega^* = \arg\min_\omega H &= \arg\min_\omega (\lambda + \mu\sigma)\,\dot{\mathbf{x}} + (1+\mu)L \\ &= \arg\min_\omega (\lambda_\alpha + \mu\sigma_\alpha)\left(\omega - \frac{1}{d}\sin\psi\right), \end{aligned}$$

which implies,

$$\omega^* = -\rho\,\mathrm{sign}\,(\lambda_\alpha + \mu\sigma_\alpha)\,. \tag{21}$$

The Attacker's optimal heading control is given by,

$$\begin{aligned} \psi^* &= \arg\min_\psi H = \arg\min_\psi (\lambda + \mu\sigma)\dot{\mathbf{x}} + (1+\mu)L \\ &= \arg\min_\psi (\lambda + \mu\sigma) \\ &= \arg\min_\psi \; (\lambda_d + \mu\sigma_d)\cos\psi \\ &\qquad + (\lambda_\alpha + \mu\sigma_\alpha)\left(\omega - \frac{1}{d}\sin\psi\right) \\ &\qquad + (\lambda_\beta + \mu\sigma_\beta)\,\frac{1}{d}\sin\psi. \end{aligned}$$

Thus ψ^* is determined by setting the vector $\begin{bmatrix}\cos\psi^* & \sin\psi^*\end{bmatrix}^\top$ to be antiparallel to the vector $\begin{bmatrix}\frac{\lambda_d+\mu\sigma_d}{\xi} & \frac{\lambda_\beta-\mu\sigma_\alpha}{d\xi}\end{bmatrix}^\top$:

$$\cos\psi^* = \frac{-\lambda_d - \mu\sigma_d}{\xi} \tag{22}$$

$$\sin\psi^* = \frac{-\lambda_\beta + \mu\sigma_\alpha}{d\xi}, \tag{23}$$

where,

$$\xi = \sqrt{(\lambda_d + \mu\sigma_d)^2 + \frac{1}{d^2}(\lambda_\beta - \mu\sigma_\alpha)^2}.$$

In order to proceed, let $\mu(t_f) = 0$, that is, the system is unconstrained at final time. This is not strictly necessary, however, it will end up being the case for the particular parameter settings used in Section III. Because components of the final state $\mathbf{x}(t_f)$ are free, the transversality condition yields [6],

$$\begin{aligned}\lambda^\top(t_f) &= \frac{\partial\Phi}{\partial\mathbf{x}(t_f)} + \nu\frac{\partial\phi}{\partial\mathbf{x}(t_f)} \\ &= 0 + \nu\begin{bmatrix}-\frac{y_R}{d(t_f)^2} & 0 & -\cos\beta\end{bmatrix}\end{aligned}$$

and thus,

$$\lambda_d(t_f) = -\nu\frac{y_R}{d(t_f)^2} \tag{24}$$

$$\lambda_\alpha(t_f) = 0 \tag{25}$$

$$\lambda_\beta(t_f) = -\nu\cos\beta. \tag{26}$$

The cosine of the optimal terminal Attacker heading is obtained by substituting (24) and (26) into (22) with $\mu(t_f) = 0$

$$\begin{aligned}\cos\psi^*(t_f) &= \frac{\nu y_R/d(t_f)^2}{\sqrt{\nu^2\frac{y_R^2}{d(t_f)^2} + \frac{1}{d(t_f)^2}\nu^2\cos\beta(t_f)^2}} \\ &\propto y_R\,\mathrm{sign}(\nu) \\ &\propto -\mathrm{sign}(\nu).\end{aligned}$$

The last expression is due to the fact that $y_R < 0$. If $\nu > 0$, we would have,

$$\dot{d}(t_f) = \cos\psi(t_f) \propto -\mathrm{sign}(\nu) = -1, \tag{27}$$

which states that distance at final time is decreasing, but for optimal constrained retreat, as long as the Turret is not placed directly on the retreat surface, the distance at final time must be increasing. Therefore, it must be the case that $\nu < 0$. Note, also, that because $\theta^* = 0$ we have $L = c$. With knowledge of the sign of ν, the cosine of the optimal Attacker's heading angle at final time can be further simplified,

$$\begin{aligned}\cos\psi^*(t_f) &= \frac{\mathrm{sign}(\nu)y_R/d(t_f)^2}{\sqrt{\frac{y_R^2}{d(t_f)^4} + \frac{1}{d(t_f)^2}\cos\beta(t_f)^2}} \\ &= \frac{-y_R}{d(t_f)\sqrt{\frac{y_R^2}{d(t_f)^2} + \cos\beta(t_f)^2}}.\end{aligned}$$

Define,

$$\chi = \sqrt{\frac{y_R^2}{d(t_f)^2} + \cos\beta(t_f)^2}.$$

Then the optimal terminal Attacker heading angle can be written,

$$\cos\psi^*(t_f) = \frac{-y_R}{d(t_f)\chi},\ \sin\psi^*(t_f) = \frac{-\cos\beta(t_f)}{\chi}. \tag{28}$$

Based on the necessary conditions for optimality [6], the Hamiltonian at final time is given by

$$H^*(t_f) = -\frac{\partial\Phi}{\partial t_f} - \nu\frac{\partial\phi}{\partial t_f} = 0 \tag{29}$$

Substituting (1) and (24)–(26) into the Hamiltonian, (13), at final time gives

$$\begin{aligned}H(t_f) = \ &-\nu\frac{y_R}{d(t_f)^2}\cos\psi(t_f) \\ &-\nu\cos\beta(t_f)\frac{1}{d(t_f)}\sin\psi(t_f) + L\end{aligned} \tag{30}$$

Substituting in (28) and (29) with $\mu = 0$ and $L = c$ into (30) yields

$$\begin{aligned}H^*(t_f) = 0 = \nu\frac{y_R}{d(t_f)^2}\cdot\frac{y_R}{\chi d(t_f)} \\ +\nu\cos\beta(t_f)\frac{1}{d(t_f)}\cdot\frac{\cos\beta(t_f)}{\chi} + c.\end{aligned} \tag{31}$$

Solving for ν:

$$\nu = \frac{-c\chi}{\frac{y_R^2}{d(t_f)^3} + \frac{\cos^2\beta(t_f)}{d(t_f)}}. \tag{32}$$

Because the retreat surface is a straight line in the (x, y)-plane, selecting either $d(t_f)$ or $\beta(t_f)$ determines the other. Therefore, the ν may be computed readily given a choice in one of these variables, and thus the terminal adjoint values, $\lambda(t_f)$, may be computed as well.

In order to compute μ when the path constraint is active (i.e. $g(\mathbf{x}) = 0$), the optimal state dynamics $\dot{\mathbf{x}}^*$ (found by substituting in the optimal adjoint variables and optimal controls) may be substituted into (12), which can be then solved for the μ in which $h(\mathbf{x}) = 0$. Alternatively, consider that in order to keep on the constraint $h(\mathbf{x}) = 0$ and similarly $\dot{h}(\mathbf{x}) = 0$:

$$\dot{h} = \frac{\partial h}{\partial\mathbf{x}}\dot{\mathbf{x}} + \frac{\partial h}{\partial\lambda}\dot{\lambda} + \frac{\partial h}{\partial\mu}\dot{\mu} = 0$$

Also, when the trajectory enters or exits a constrained arc $\mu = 0$. Thus we can append μ to $\mathbf{x}$ over the constrained portion of the trajectory with $\mu(t_1) = 0$ and,

$$\dot{\mu} = \frac{-\frac{\partial h}{\partial\mathbf{x}}\dot{\mathbf{x}} - \frac{\partial h}{\partial\lambda}\dot{\lambda}}{\frac{\partial h}{\partial\mu}},$$

where t_1 is the time at which the trajectory enters a constrained arc (in backwards time). Let t_2 be the time (in backwards time) at which the trajectory leaves the constrained arc and enters an unconstrained arc. In order for the tangency condition to be met, we have $h(\mathbf{x}(t_2)) = 0$. As a result,

the adjoint variables are subject to an additional internal boundary condition [2], [6],

$$\lambda^\top(t_2^-) = \lambda^\top(t_2^+) + \pi \frac{\partial h}{\partial \mathbf{x}}, \tag{33}$$

where π is an additional adjoint variable.

Generally, one would use (33) to solve for π by substituting into the Hamiltonian, (13), evaluated at $t = t_2^-$. However, computing $\frac{\partial h}{\partial \mathbf{x}}$ is nontrivial as it is subject to the numerical inaccuracies introduced by sampling σ (the GoE adjoints). We take a different approach which avoids the computation of $\frac{\partial h}{\partial \mathbf{x}}$. Since $\dot{\mathbf{x}}$, σ_d, and σ_α do not depend on β, $\frac{\partial h}{\partial \beta} = 0$. Thus, from (33), we have $\lambda_\beta(t_2^-) = \lambda_\beta(t_2^+)$ and are left with two unknowns: $\lambda_d(t_2^-)$ and $\lambda_\alpha(t_2^-)$. At $t = t_2^-$ we have that $h\left(\mathbf{x}(t_2^-)\right) = 0$ due to the tangency condition [6]. It is also the case that $h\left(\mathbf{x}(t_2^+)\right) = 0$ since the system is constrained after t_2. Expanding (12) yields,

$$h(\mathbf{x}) = \sigma_d \cos\psi^* + \sigma_\alpha \left(\omega^* - \frac{1}{d}\sin\psi^*\right) + c, \tag{34}$$

since $\sigma_\beta = 0$. The quantities σ_d and σ_α are continuous at t_2 and c is constant, thus, from (34) it must be true that $\psi^*(t_2^-) = \psi^*(t_2^+)$. Eqs. (22) and (23) can be manipulated to solve for $\lambda_d(t_2^-)$, noting that $\mu(t_2^-) = 0$

$$\lambda_d(t_2^-) = \frac{\lambda_\beta(t_2^-)\left(\lambda_d(t_2^+) + \mu(t_2^+)\sigma_d\right)}{\left(\lambda_\beta(t_2^+) - \mu(t_2^+)\sigma_\alpha\right)}. \tag{35}$$

Because the dynamics, (1), are autonomous and $H(t_f) = 0$ the Hamiltonian must be zero at all times including at time t_2^-. Evaluating (13) at t_2^- and substituting the value of (35) in along with $\mu = 0$ gives

$$\begin{aligned} H^*(t_2^-) = 0 &= \lambda\dot{\mathbf{x}} + \mu h + L \\ &= \lambda_d(t_2^-)\dot{d}(t_2^-) + \lambda_\alpha(t_2^-)\dot{\alpha}(t_2^-) + \lambda_\beta\dot{\beta}(t_2^-) + 0 + c \\ &= \lambda_d(t_2^-)\cos\psi^* + \lambda_\alpha(t_2^-)\left(\omega^* - \frac{1}{d}\sin\psi^*\right) \\ &\quad + \lambda_\beta(t_2^-)\frac{1}{d}\sin\psi^* + c. \end{aligned} \tag{36}$$

Substituting in (22) and (23), rearranging, and solving for $\lambda_\alpha(t_2^-)$ gives

$$\lambda_\alpha(t_2^-) = \frac{\sqrt{\lambda_d(t_2^-)^2 + \frac{1}{d^2}\lambda_\beta(t_2^-)^2} - c}{\omega^* + \frac{\lambda_\beta}{d^2\xi}} \tag{37}$$

Note that ω^* depends on the sign of λ_α (c.f. (21)), so its sign may be assumed *a priori* and then (37) must be checked for consistency.

B. Boundary Value Problem

For backwards shooting, the Boundary Value Problem (BVP) consists of choosing a $\beta(t_f)$ (which also determines λ_β, $d(t_f)$, and $\lambda_d(t_f)$), a value for t_f, and a value for $\alpha(t_f)$ and then integrating backwards in time from t_f to t_0. At t_0 the state values may be substituted into the first three components of ϕ in (2) to yield a residual (the fourth component is 0 by construction/selection of the terminal state). Formally, we have,

$$\begin{aligned} \beta_f^*, \alpha_f^*, t_f^* = &\underset{\beta_f, \alpha_f, t_f}{\arg\min} \|\phi\| \\ \text{s.t. } & g(\mathbf{x}(t)) \geq 0 \ \forall t \in [t_0, t_f], \\ & f(\mathbf{x}, \mathbf{u}, t) = \dot{\mathbf{x}}, \\ & \text{Eqs. (17)–(18), (19), (21), (22), (23).} \end{aligned} \tag{38}$$

In practice, dealing with the path constraint forces one to assume a sequence of arcs, e.g., UCU, where U denotes an unconstrained arc and C denotes a constrained arc (e.g., see [7]). Then, the times at which the system switches from an arc of one type to another must be solved for as well. Due to the similarity of this instantiation of constrained optimal retreat to that of the example in [2], let us assume that UCU is indeed the proper sequence to consider, and that the boundary conditions at $t = 0$ are such that the path constraint will indeed be activated. That is, the trajectory will begin unconstrained, transition into a constrained arc, and finally end with another unconstrained arc. In backwards time, let t_1 be the time instant of the first switch, from U to C, and let t_2 be the time instant of the second switch, from C to U. Note, also, that in order to transition from U to C, it must also be the case that $h(\mathbf{x}(t_1)) = 0$ (i.e., the tangency condition must be satisfied). Then the BVP in (38) may be reposed as

$$\begin{aligned} \beta_f^*, \alpha_f^*, t_f^*, t_1^*, \mathbf{x}_1^*, t_2^* = &\underset{\beta_f, \alpha_f, t_f, t_1, \mathbf{x}_1, t_2}{\arg\min} \|\phi\| \\ \text{s.t. } & t_2 \leq t_1, \\ & g(\mathbf{x}(t_1)) = 0, \\ & h(\mathbf{x}(t_1)) = 0, \\ & f(\mathbf{x}, \mathbf{u}, t) = \dot{\mathbf{x}}, \\ & \text{Eqs. (17)–(18), (19), (21), (22), (23).} \end{aligned} \tag{39}$$

At t_1, when the constraint first becomes active we proceed backwards in time, using the constrained version of the optimal state, adjoint, and control dynamics wherein $\mu \neq 0$. Then, at t_2, the jump condition (33) is used to update λ_d and λ_α and then the integration proceeds with $\mu = 0$.The satisfaction of $g(\mathbf{x}(t)) = 0$, $\forall t \in [0, t_f]$, $t \neq t_1$ is guaranteed by way of the optimality conditions derived in this section. In terms of the implementation, we employ a multiple shooting scheme [8] wherein the first shot comprises the first U arc (in backwards time) from t_f to t_1 and the second shot comprises the remaining C and U arcs. Thus the BVP in (39) is augmented to include the stitching constraints at t_1.

C. Turret Turn Control

If the path constraint does not become active, the trajectory is comprised of a single U arc. Consider the case where the constraint becomes active at some point along the trajectory and later becomes inactive; this corresponds to the UCU sequence of arcs.

Case 1. (Terminal unconstrained arc, $t \in [t_1, t_f]$.) This case corresponds to the final U arc in the UCU sequence, or

the singular U arc in a completely unconstrained trajectory. Here, $\mu = 0$, and $\lambda_\alpha = 0$ due to (17) and (25). According to (21) the optimal Turret control, ω^* is undefined. Once the trajectory is onthe fina unconstrained arc, the Turret's control has no bearing on the outcome. In this time interval, any $\omega_u \in [-\rho, \rho]$ is trivially optimal [2]. As a convention, let us adopt $\omega_u = -\rho \operatorname{sign}(\sigma_\alpha)$ in this case, which corresponds to the GoE control [3].

Case 2. (Constrained arc, $t \in [t_2, t_1)$.) This case corresponds to the C arc wherein the path constraint is active. Here, $\mu \neq 0$, since $\mu(t_1) = 0$ and $\mu > 0$ when the constraint is active. As in the previous case, $\lambda_\alpha = 0$. Therefore, ω^* is generally well defined according to (21) except in the following (sub)cases.

Case 2.1. (Turret locked on, $\alpha = 0$.) If $\alpha = 0$, the Turret's gaze is lined up exactly with the Attacker's position. This configuration corresponds to the Turret's Universal Surface wherein the GoE adjoint $\sigma_\alpha = 0$ [3]. Since $\lambda_\alpha = \sigma_\alpha = 0$, the Turret's optimal control is undefined. In this case, ω^* is given by (41) (see Proposition 1).

Case 2.2. (Turret looking away, $\alpha = \pi$.) If $\alpha = \pi$, the Turret is looking directly away from the Attacker's position. This configuration corresponds to the Turret's Dispersal Surface wherein the GoE adjoint σ_α is undefined [3]. Thus, the Turret's optimal control, ω^* is undefined as well. In the GoE, the Turret has the authority to choose to turn either clockwise or counterclockwise to remain on an optimal (equilibrium) trajectory. Here, however, the choices are not equivalent because the Attacker's heading control is different than in the GoE. In this case, we propose setting ω to

$$\omega_{DS} = \begin{cases} \rho & \text{if } \cos\beta \geq 0, \\ -\rho & \text{otherwise}, \end{cases} \tag{40}$$

which states that if the Attacker is to the right of the Turret, the Turret should turn counterclockwise at full rate. In this configuration, turning counterclockwise results in a larger $\dot{\alpha}$, bringing the Attacker closer to the Turret's view for the remainder of the constrained arc than if the Turret had turned clockwise (see, e.g., Fig. 1).

Case 3. (Initial unconstrained arc, $t \in [t_0, t_2]$.) This case corresponds to the initial U arc in the UCU sequence. Here $\lambda_\alpha \neq 0$ in general, due to (37). Since $\mu = 0$ when the constraint is inactive ω^* is well-defined. Moreover, because $\dot{\lambda}_\alpha = 0$, the Turret always turns in one particular direction during this part of the trajectory.

Proposition 1. *When the constraint is active ($g(\mathbf{x}) = 0$) and $\alpha = 0$, the Turret's optimal control is given by*

$$\omega_{US} = -\operatorname{sign}(\lambda_\beta) \cdot \min\left(\left| \frac{\lambda_\beta}{d^2 \sqrt{(\lambda_d + \mu\sigma_d)^2 + \frac{1}{d^2}\lambda_\beta^2}} \right|, \rho \right) \tag{41}$$

Proof. In the GoE, the Turret's singular control along the Universal Surface is $\omega = 0$, which is the control associated with keeping $\alpha = \sigma_\alpha = 0$. However, in the optimal constraint retreat scenario, a different singular control may be required to keep $\alpha = 0$. Note that $\sigma_\alpha = 0$ on the Universal Surface. Thus we have,

$$\begin{aligned} \dot{\alpha} &= \omega - \frac{1}{d}\sin\psi \\ &= \omega + \frac{\lambda_\beta - \mu\sigma_\alpha}{d^2\sqrt{(\lambda_d + \mu\sigma_d)^2 + \frac{1}{d^2}(\lambda_\beta - \mu\sigma_\alpha)^2}} \\ &= \omega + \frac{\lambda_\beta}{d^2\sqrt{(\lambda_d + \mu\sigma_d)^2 + \frac{1}{d^2}\lambda_\beta^2}}, \end{aligned}$$

and thus to keep $\alpha = 0$ the Turret should select ω such that $\dot{\alpha} = 0$. However, it is entirely possible that the Turret does not have enough control authority to keep $\dot{\alpha} = 0$ and thus control saturation must be considered – hence the $\min$ with ρ in (41).

Suppose that the Turret were to disregard (41) in the case where the first argument of the $\min$ is less than ρ. The Turret's gaze would actually turn *past* the Attacker thereby making $\alpha \neq 0$ and $\sigma_\alpha \neq 0$ and the non-singular turn control, (21), would come into play. Eq. (21) would dictate that the Turret reverse direction in order to drive $\alpha \to 0$, and the dilemma would begin again. Thus the Turret's turn control would chatter/modulate in such a way as to emulate the behavior captured in (41). □

IV. Solutions and Results

The results in this section pertain to the following initial conditions and an assumed terminal Turret look-angle

$$\mathbf{x}_0^\top = \begin{bmatrix} 20.61 & -0.2450 & 1.816 \end{bmatrix}, \qquad \alpha_f = -0.6736. \tag{42}$$

These values correspond to the Attacker beginning at $(-5, 20)$ in the xy-plane and the Turret looking along the y-axis, initially. In order to ensure feasible transition from U to C we perform a sweep of β_f for the assumed α_f. The trajectories in the sweep are integrated starting at $t = t_f$ and proceed (backwards) until the first time at which either $g(\mathbf{x}) = 0$ or $d(t)\sin\beta(t) = 20$.

The purpose of the sweep is to find a value for β_f which, when integrated until $g(\mathbf{x}) = 0$ results in the tangency condition to be satisfied: $h(\mathbf{x}) = 0$. The β_f associated with this point is then used as an initial guess for a Nonlinear Program (NLP) to fine-tune β_f to drive $h(\mathbf{x}) \to 0$ when the constraint becomes active ($g(\mathbf{x}) = 0$). The solution of the NLP also yields a guess for $t_1, d_1, \alpha_1, \beta_1$, and λ_{d_1}. A guess for t_2 (or $\Delta t = t_1 - t_2,\ \Delta t > 0$) is the last piece needed to specify a guess for the BVP, (39).

Figure 3 shows the result of the BVP solution procedure. The red portion of the Attacker trajectory indicates where the path constraint is active. The Turret's initial look angle is shown by the black arrow; the first blue arrow counterclockwise is the Turret's look angle when the constraint becomes active; the red arrow is the look angle when the trajectory leaves the constraint; the last blue arrow indicates the final Turret look angle. The open black circle indicates the initial boundary conditions at $(x, y) = (-5, 20)$. Note

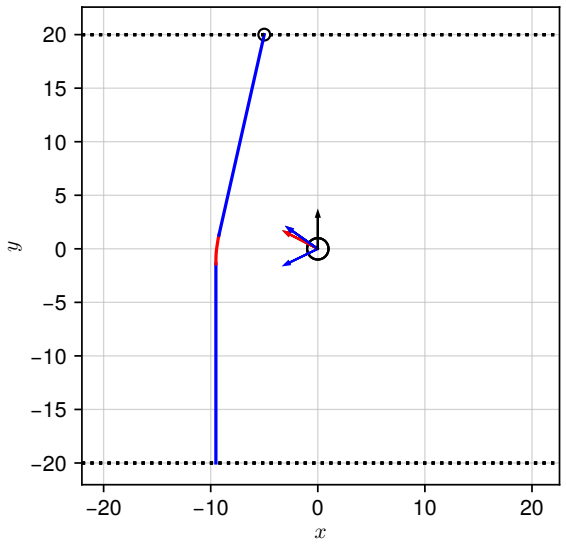

Fig. 3. Indirect backwards multiple shooting solution trajectory.

that the Turret is constantly turning counterclockwise in this case. Incidentally, the best trajectory found by the NLP corresponds to the initial guess. The reasons for this may be that the solver was not allowed to run long enough, the integrator was encumbered with too many callbacks (with too high of precision), the solver had trouble finding or moving to $\mathbf{x}(t_2)$ that satisfy $g(\mathbf{x}(t_1)) = 0$ and $h(\mathbf{x}(t_1)) = 0$, or the solver algorithm (constrained optimization by linear approximation, or COBYLA) has difficulties with this particular problem.

Figure 4 shows the state, adjoint, and control trajectories associated with the indirect solution. Figure 5 compares the Value of Engagement and Value of Retreat along the trajectory; $g(\mathbf{x})$ is the difference between the upper and lower curves, and the curves are coincident along the red constrained arc. The validity of the trajectory is guaranteed by construction: it satisfies all of the necessary conditions for optimality in (39) and, from Fig. 5, we see that $V_E \geq V_R$ along the entire trajectory. In particular, we've shown the existence of and computed optimal retreat control strategies, $\psi^*(t;\mathbf{x})$ and $\omega^*(t;\mathbf{x})$ which is sufficient to state this is a valid solution according to [2, Definition 1 & Theorem 1].

V. Conclusion

In this paper, we have considered an engage or retreat scenario as applied to a mobile Attacker against a stationary Turret with bounded turn rate. Since the solution of the Game of Engagement is known, we focused on solving the Optimal Constrained Retreat problem. The path constraint imposed by the Game of Engagement makes the Optimal Constrained Retreat problem both interesting and challenging. We developed the first-order necessary conditions for optimality which yielded a system of differential equations describing the optimal dynamics. Then, we specified a general Boundary Value Problem with known initial conditions and solved it for a particular point in the state space using backwards (multiple) shooting.

There are still some difficulties in applying the optimality conditions to find an optimal trajectory, in general. The three singular surfaces that appear in the Game of Engagement impute some of their complexities and subtleties onto the Optimal Constrained Retreat, particularly when the constraint is active. Specification of the Boundary Value Problem relied on having to assume a sequence of constrained and unconstrained arcs. Some care was taken to choose parameters and initial boundary conditions to yield a trajectory in which the path constraint would become active. Lastly, the process of computing an initial guess to satisfy the tangency condition when the constraint becomes active is nontrivial. Addressing these difficulties more thoroughly is left to future work. Additionally, we seek to compute the barrier surface which partitions the state space into a region where engagement is optimal and a region where retreat is optimal.

Acknowledgment

This paper is based on work performed at the Air Force Research Laboratory (AFRL) *Control Science Center of Excellence*. Distribution Unlimited. 22 Jan 2020. Case #88ABW-2020-0185.

References

[1] E. Garcia, D. W. Casbeer, A. Von Moll, and M. Pachter, "Multiple pursuer multiple evader differential games," *arXiv preprint arXiv:1911.03806*, 2019.

[2] Z. E. Fuchs and P. P. Khargonekar, "Generalized engage or retreat differential game with escort regions," *IEEE Transactions on Automatic Control*, vol. 62, pp. 668–681, 2017.

[3] Z. Akilan and Z. Fuchs, "Zero-sum turret defense differential game with singular surfaces," in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, 2017, pp. 2041–2048.

[4] A. Von Moll and Z. Fuchs, "Attacker dispersal surface in the turret defense differential game," in *IFAC World Congress on Automatic Control*, 2020.

[5] K. Carlsson, "Nearestneighbors.jl," 2020, accessed 2020-01-14. [Online]. Available: https://github.com/KristofferC/NearestNeighbors.jl

[6] A. E. Bryson and Y.-C. Ho, *Applied Optimal Control: Optimization, Estimation and Control*. CRC Press, 1975.

[7] O. Cots, J. Gergaud, and D. Goubinat, "Direct and indirect methods in optimal control with state constraints and the climbing trajectory of an aircraft," *Optimal Control Applications and Methods*, vol. 39, pp. 281–301, 01 2018.

[8] J. Stoer and R. Bulirsch, *Introduction to numerical analysis*. Springer Science & Business Media, 2013, vol. 12.

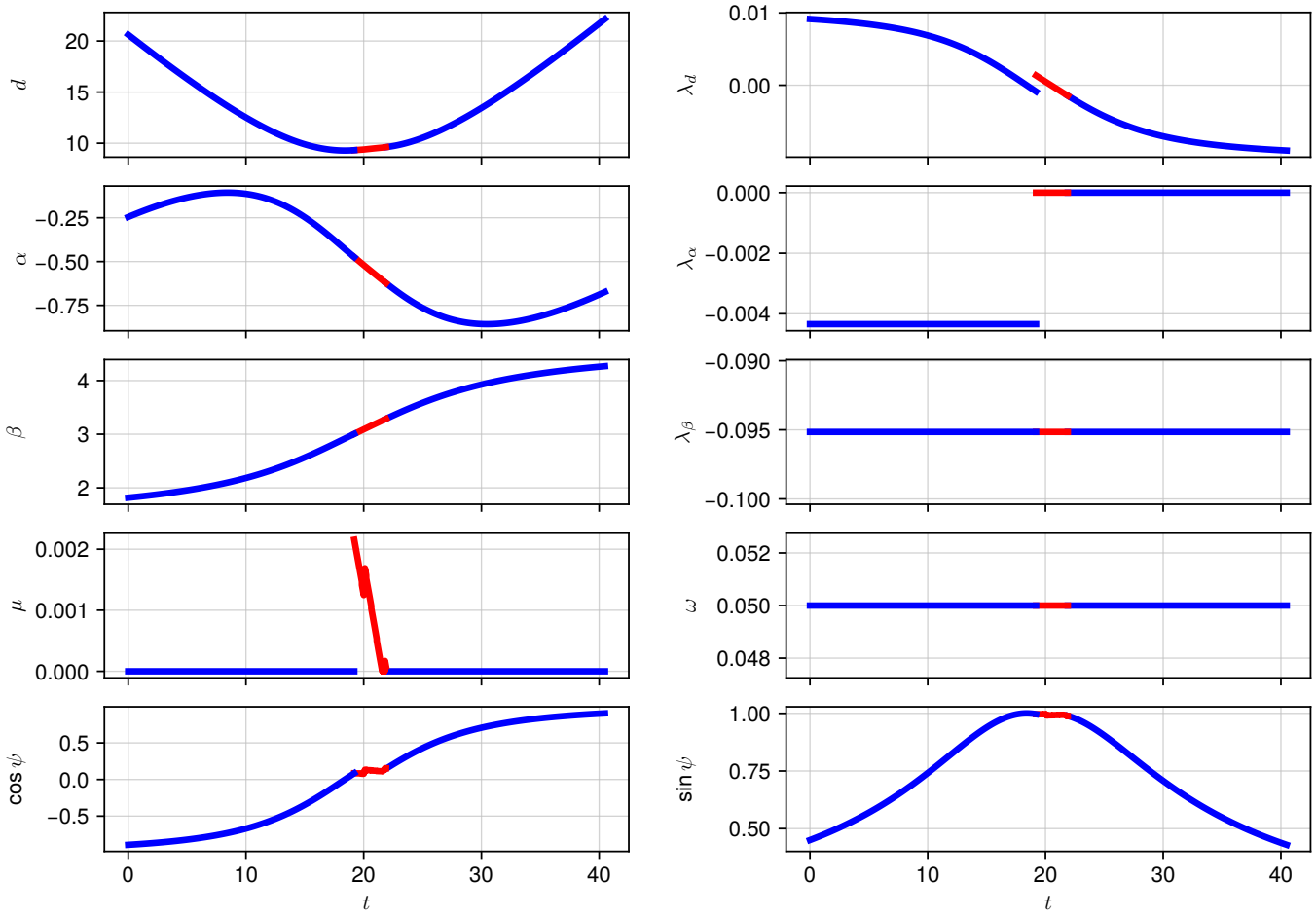

Fig. 4. Indirect backwards multiple shooting state, adjoint, and control trajectories.

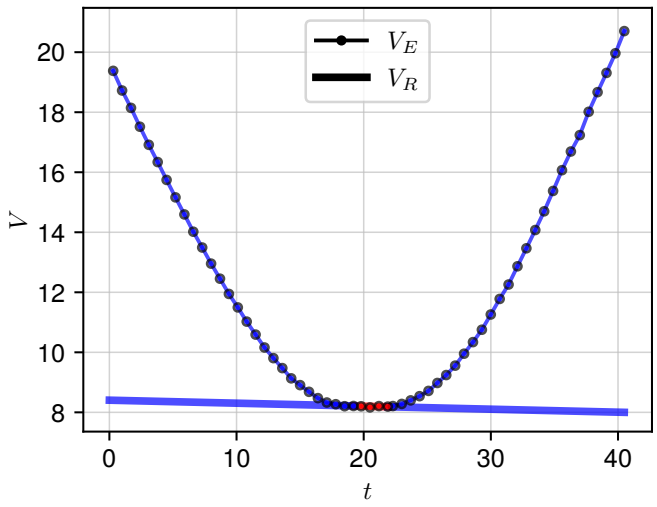

Fig. 5. Comparison of Value of Engagement and Value of Retreat along the trajectory.