Game Theoretic Continuous Time Differential Dynamic Programming

Wei Sun[1], Evangelos A. Theodorou[2] and Panagiotis Tsiotras[3]

***Abstract*— In this work, we derive a Game Theoretic Differential Dynamic Programming (GT-DDP) algorithm in continuous time. We provide a set of backward differential equations for the value function expansion without assuming closeness of the initial nominal control to the optimal control solution, and derive the update law for the controls. We introduce the GT-DDP algorithm and analyze the effect of the game theoretic formulation in the feed-forward and feedback parts of the control policies. Furthermore, we investigate the performance of GT-DDP through simulations on the inverted pendulum with conflicting controls and we apply the control gains on a stochastic system to demonstrate the effect of the design of the cost function to the feed-forward and feedback parts of the control policies. Finally, we conclude with some possible future directions.**

I. Introduction

Differential game-theoretic or min-max formulations are important extensions of optimal control having direct connections to robust and H^∞ nonlinear control theory. Despite the plethora of work in this area, min-max algorithms for trajectory optimization have only very recently been derived, and have been applied to humanoid robotic control problems [1], [2]. Solving differential games (or min-max) problems is a challenging tasks. In practice, numerical algorithms inspired from the solution of optimal control problems are often used. Among them, differential dynamic programming (DDP) has recently emerged as a suitable formulation to compute feedback strategies iteratively with reasonable computational costs. Several variations of DDP algorithms have been derived and have been extensively applied to deterministic and stochastic systems in robotics, autonomous systems and computational neuroscience. In particular, in [3] a discrete time DDP algorithm is derived for nonlinear stochastic systems with state and control multiplicative noise, and applied to biomechanical models. The resulting algorithm, known as iterative Linear Quadratic Gaussian (iLQG) control, relies on first order expansion of the dynamics. In [4], second-order expansions of stochastic dynamical systems with state and control multiplicative noise are considered. The resulting algorithm, known as Stochastic Differential Dynamic Programming (SDDP), is a generalization of iLQG. In [5] random sampling techniques are proposed to improve the scalability of DDP. In [6] an infinite horizon version of discrete time DDP is derived and in [7] discrete time receding horizon DDP is applied for helicopter acrobatic maneuvers. Finally in [8], DDP is derived for deterministic nonlinear systems with controls limits and applied to control of a humanoid robot in simulation. Interestingly, although the initial derivation of DDP [9] was in continuous time, most of work on the application of DDP for solving trajectory optimization problems, including min-max DDP formulations, such as [1], [2], rely on a discrete time formulation. Compared to previous work, our contribution in this paper is the derivation of the min-max DDP conditions in continuous time. Specifically, we provide a set of backward differential equations that are easy to implement and derive the optimal policies for the two players/controllers.

With respect to the initial treatment of DDP in the book by D. H. Jacobson and D. Q. Mayne [9] our analysis and derivation of the Game-Theoretic DDP (GT-DDP) avoids a restrictive assumption of the initial derivation in [9]. This assumption was also discussed in a review paper of [9] published in 1971 by Michael K. Sain [10]. In particular, the fundamental assumption in the derivation of continuous-time DDP in [9] is that the nominal control $\bar{\mathbf{u}}$ is close to the optimal control $\mathbf{u}^*$. This assumption allows the expansion of the terms in the Hamilton-Jacobi-Bellman (HJB) Partial Differential Equation (PDE) around $\mathbf{u}^*$ instead $\bar{\mathbf{u}}$ and results in the cancelation of terms that depend on $\mathcal{H}_{\mathbf{u}^*} = 0$, where $\mathcal{H}_{\mathbf{u}^*}$ stands for the partial derivative of the Hamiltonian with respect to the control input.

GT-DDP does not rely on the assumption regarding the closeness of the nominal controls $\bar{\mathbf{u}}$ and $\bar{\mathbf{v}}$ to $\mathbf{u}^*$ and $\mathbf{v}^*$, respectively, and therefore the quadratic expansions of the terms in the HJB PDE are computed around the nominal controls $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$ and not the optimal control $\mathbf{u}^*$, $\mathbf{v}^*$. In this case, the term $\mathcal{H}_{\mathbf{u}}$ is not necessarily equal to zero.

The paper is organized as follows. In Section II the game theoretic problem is formulated and the backward Riccati equations are derived. In Section III the terminal conditions are specified and the main algorithm is presented and Section IV includes simulations results. Finally in Section V we conclude and discuss future directions.

II. Problem Formulation

We consider the following min-max problem:

$$V(\mathbf{x}(t_0), t_0) = \min_{\mathbf{u}} \max_{\mathbf{v}} \left\{ \phi(\mathbf{x}(t_f), t_f) + \int_{t_0}^{t_f} \mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v}, t) \mathrm{d}t \right\}, \quad (1)$$

[1]W. Sun is a Ph.D. candidate at the School of Aerospace Engineering, Georgia Institute of Technology, Atlanta. GA 30332-0150. USA. Email: wsun42@gatech.edu

[2]E. Theodorou is an Assistant Professor at the School of Aerospace Engineering, Georgia Institute of Technology, Atlanta. GA 30332-0150. USA. Email: evangelos.theodorou@ae.gatech.edu

[3]P. Tsiotras is a Professor at the School of Aerospace Engineering, Georgia Institute of Technology, Atlanta. GA 30332-0150. USA. Email: tsiotras@gatech.edu

subject to the dynamics

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{F}(\mathbf{x}, \mathbf{u}, \mathbf{v}, t), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \tag{2}$$

where V stands for the optimal performance index starting from $\mathbf{x}_0$ at time t_0, $\mathbf{x}(t)$ is an n-dimensional vector function of time describing the state of the dynamic system at $t \in [0, t_f]$. $\mathcal{L}$ and ϕ are scalar functions of their arguments, where $\mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v}, t)$ is the *running cost* and $\phi(\mathbf{x}(t_f), t_f)$ is the *terminal cost*. Finally, $\mathbf{u}$ is an m-dimensional vector function that represents the stabilizing control of the system, whose objective is to minimize the performance index, whereas $\mathbf{v}$ is a q-dimensional vector function representing the destabilizing control of the system that tries to maximize the performance index.

In continuous time, the analysis starts with the Hamilton-Jacobi-Bellman Isaacs (HJBI) partial differential equation. More precisely, we have:

$$-\frac{\partial V(\mathbf{x}, t)}{\partial t} = \min_{\mathbf{u}} \max_{\mathbf{v}} \Big\{ \mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v}, t) + V_{\mathbf{x}}(\mathbf{x}, t)^{\intercal} F(\mathbf{x}, \mathbf{u}, \mathbf{v}, t) \Big\}, \tag{3}$$

under the boundary condition

$$V(\mathbf{x}, t_f) = \phi(\mathbf{x}(t_f), t_f). \tag{4}$$

Given an initial/nominal trajectory of the state and control $(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})$, and letting $\delta\mathbf{x} = \mathbf{x} - \bar{\mathbf{x}}$, $\delta\mathbf{u} = \mathbf{u} - \bar{\mathbf{u}}$, $\delta\mathbf{v} = \mathbf{v} - \bar{\mathbf{v}}$, the linearized dynamics can be represented as

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = F(\bar{\mathbf{x}} + \delta\mathbf{x}, \bar{\mathbf{u}} + \delta\mathbf{u}, \bar{\mathbf{v}} + \delta\mathbf{v}, t), \tag{5}$$

$$\frac{\mathrm{d}\delta\mathbf{x}}{\mathrm{d}t} = F_{\mathbf{x}}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}}, t)\delta\mathbf{x} + F_{\mathbf{u}}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}}, t)\delta\mathbf{u} + F_{\mathbf{v}}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}}, t)\delta\mathbf{v}. \tag{6}$$

The main idea here is to take expansions of the terms in both sides of equation (3) around the nominal state and control trajectories $(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})$ to derive the update law for the stabilizing control, destabilizing control, along with the backward differential equations for the zeroth, first and second order approximation terms of the value function. Starting with the left-hand side of (3) we have:

$$\frac{\partial V(\mathbf{x}, t)}{\partial t} = \frac{\partial V(\bar{\mathbf{x}}, t)}{\partial t} + \frac{\partial V_{\mathbf{x}}^{\intercal}(\bar{\mathbf{x}}, t)}{\partial t}\delta\mathbf{x} + \frac{1}{2}\delta\mathbf{x}^{\intercal}\frac{\partial V_{\mathbf{xx}}(\bar{\mathbf{x}}, t)}{\partial t}\delta\mathbf{x}. \tag{7}$$

We also have

$$\begin{aligned}\frac{\mathrm{d}V(\bar{\mathbf{x}}, t)}{\mathrm{d}t} &= \frac{\partial V(\bar{\mathbf{x}}, t)}{\partial t} + V_{\mathbf{x}}(\bar{\mathbf{x}}, t)^{\intercal}\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t}\bigg|_{\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}}} \\ &= \frac{\partial V(\bar{\mathbf{x}}, t)}{\partial t} + V_{\mathbf{x}}^{\intercal}(\bar{\mathbf{x}}, t)F(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}}, t).\end{aligned} \tag{8}$$

Thus, we get

$$\frac{\partial V(\bar{\mathbf{x}}, t)}{\partial t} = \frac{\mathrm{d}V(\bar{\mathbf{x}}, t)}{\mathrm{d}t} - V_{\mathbf{x}}^{\intercal}(\bar{\mathbf{x}}, t)F(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}}, t). \tag{9}$$

Similarly,

$$\frac{\partial V_{\mathbf{x}}(\bar{\mathbf{x}}, t)}{\partial t} = \frac{\mathrm{d}V_{\mathbf{x}}(\bar{\mathbf{x}}, t)}{\mathrm{d}t} - V_{\mathbf{xx}}(\bar{\mathbf{x}}, t)F(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}}, t). \tag{10}$$

Finally, the partial time derivative of the Hessian of the value function takes the form:

$$\frac{\partial V_{\mathbf{xx}}(\bar{\mathbf{x}}, t)}{\partial t} = \frac{\mathrm{d}V_{\mathbf{xx}}(\bar{\mathbf{x}}, t)}{\mathrm{d}t} - \sum_{i=1}^{n} V_{\mathbf{xxx}}^{(i)(\bar{\mathbf{x}}, t)}F^{(i)}, \tag{11}$$

where $V_{\mathbf{xxx}}^{(i)}$ denotes the Hessian matrix of the i-th element of $V_{\mathbf{x}}$ and $F^{(i)}$ denotes the i-th element of $F(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}}, t)$. Henceforth, the arguments for the functions V, F, etc, are omitted for brevity, and they are evaluated at the nominal trajectory $(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})$ unless specified otherwise.

The left-hand side of (3) then becomes

$$-\frac{\partial V(\mathbf{x}, t)}{\partial t} = -\frac{\mathrm{d}V}{\mathrm{d}t} - \frac{\mathrm{d}V_{\mathbf{x}}^{\intercal}}{\mathrm{d}t}\delta\mathbf{x} - \frac{1}{2}\delta\mathbf{x}^{\intercal}\frac{\mathrm{d}V_{\mathbf{xx}}}{\mathrm{d}t}\delta\mathbf{x} + V_{\mathbf{x}}^{\intercal}F + \delta\mathbf{x}^{\intercal}V_{\mathbf{xx}}F + \frac{1}{2}\delta\mathbf{x}^{\intercal}\Big(\sum_{i=1}^{n} V_{\mathbf{xxx}}^{(i)}F^{(i)}\Big)\delta\mathbf{x}. \tag{12}$$

We now turn to the expansion of the right-hand side of (3).

$$\begin{aligned}&\mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v}, t) \\ &= \mathcal{L}(\bar{\mathbf{x}} + \delta\mathbf{x}, \bar{\mathbf{u}} + \delta\mathbf{u}, \bar{\mathbf{v}} + \delta\mathbf{v}, t) \\ &\approx \mathcal{L} + \mathcal{L}_{\mathbf{x}}^{\intercal}\delta\mathbf{x} + \mathcal{L}_{\mathbf{u}}^{\intercal}\delta\mathbf{u} + \mathcal{L}_{\mathbf{v}}^{\intercal}\delta\mathbf{v} \\ &+ \frac{1}{2}\begin{bmatrix} \delta\mathbf{x}^{\intercal} & \delta\mathbf{u}^{\intercal} & \delta\mathbf{v}^{\intercal} \end{bmatrix}\begin{bmatrix} \mathcal{L}_{\mathbf{xx}} & \mathcal{L}_{\mathbf{xu}} & \mathcal{L}_{\mathbf{xv}} \\ \mathcal{L}_{\mathbf{ux}} & \mathcal{L}_{\mathbf{uu}} & \mathcal{L}_{\mathbf{uv}} \\ \mathcal{L}_{\mathbf{vx}} & \mathcal{L}_{\mathbf{vu}} & \mathcal{L}_{\mathbf{vv}} \end{bmatrix}\begin{bmatrix} \delta\mathbf{x} \\ \delta\mathbf{u} \\ \delta\mathbf{v} \end{bmatrix}.\end{aligned} \tag{13}$$

By expanding $V_{\mathbf{x}}(\mathbf{x}, t)$, we have

$$V_{\mathbf{x}}(\mathbf{x}, t) = V_{\mathbf{x}}(\bar{\mathbf{x}} + \delta\mathbf{x}, t) = V_{\mathbf{x}} + V_{\mathbf{xx}}\delta\mathbf{x} + \frac{1}{2}\mathcal{V}, \tag{14}$$

where $\mathcal{V} \in \mathbb{R}^n$ and each element of $\mathcal{V}$ is defined as

$$\mathcal{V}^{(i)} = \delta\mathbf{x}_{\intercal}V_{\mathbf{xxx}}^{(i)}\delta\mathbf{x}.$$

The dynamic equation is expanded up to the first order, that is,

$$\begin{aligned}F(\mathbf{x}, \mathbf{u}, \mathbf{v}, t) &= F(\bar{\mathbf{x}} + \delta\mathbf{x}, \bar{\mathbf{u}} + \delta\mathbf{u}, \bar{\mathbf{v}} + \delta\mathbf{v}, t) \\ &= F + F_{\mathbf{x}}\delta\mathbf{x} + F_{\mathbf{u}}\delta\mathbf{u} + F_{\mathbf{v}}\delta\mathbf{v}.\end{aligned} \tag{15}$$

Therefore, the right-hand side of (3) can be expressed as

$$
\begin{aligned}
&\min_{\mathbf{u}}\max_{\mathbf{v}}\Big[\mathcal{L}(\mathbf{x},\mathbf{u},\mathbf{v},t)+V_{\mathbf{x}}^{\intercal}(\mathbf{x},t)F(\mathbf{x},\mathbf{u},\mathbf{v},t)\Big]\\
&\approx\min_{\delta\mathbf{u}}\max_{\delta\mathbf{v}}\Big[\mathcal{L}+\mathcal{L}_{\mathbf{x}}^{\intercal}\delta\mathbf{x}+\mathcal{L}_{\mathbf{u}}^{\intercal}\delta\mathbf{u}+\mathcal{L}_{\mathbf{v}}^{\intercal}\delta\mathbf{v}\\
&+\frac{1}{2}\begin{bmatrix}\delta\mathbf{x}^{\intercal} & \delta\mathbf{u}^{\intercal} & \delta\mathbf{v}^{\intercal}\end{bmatrix}\begin{bmatrix}\mathcal{L}_{\mathbf{xx}} & \mathcal{L}_{\mathbf{xu}} & \mathcal{L}_{\mathbf{xv}}\\ \mathcal{L}_{\mathbf{ux}} & \mathcal{L}_{\mathbf{uu}} & \mathcal{L}_{\mathbf{uv}}\\ \mathcal{L}_{\mathbf{vx}} & \mathcal{L}_{\mathbf{vu}} & \mathcal{L}_{\mathbf{vv}}\end{bmatrix}\begin{bmatrix}\delta\mathbf{x}\\ \delta\mathbf{u}\\ \delta\mathbf{v}\end{bmatrix}\\
&+V_{\mathbf{x}}^{\intercal}F+V_{\mathbf{x}}^{\intercal}F_{\mathbf{x}}\delta\mathbf{x}+V_{\mathbf{x}}^{\intercal}F_{\mathbf{u}}\delta\mathbf{u}+V_{\mathbf{x}}^{T}F_{\mathbf{v}}\delta\mathbf{v}\\
&+\delta\mathbf{x}^{\intercal}V_{\mathbf{xx}}F+\delta\mathbf{x}^{\intercal}V_{\mathbf{xx}}F_{\mathbf{x}}\delta\mathbf{x}+\delta\mathbf{x}^{\intercal}V_{\mathbf{xx}}F_{\mathbf{u}}\delta\mathbf{u}\\
&+\delta\mathbf{x}^{\intercal}V_{\mathbf{xx}}F_{\mathbf{v}}\delta\mathbf{v}+\frac{1}{2}\mathcal{V}^{\intercal}F\Big].
\end{aligned}\tag{16}
$$

Note that the term $\frac{1}{2}\mathcal{V}^{\intercal}F$ can be written as follows

$$
\begin{aligned}
\frac{1}{2}\mathcal{V}^{\intercal}F&=\frac{1}{2}\sum_{i=1}^{n}\Big(\delta\mathbf{x}_{\intercal}V_{\mathbf{xxx}}^{(i)}\delta\mathbf{x}F^{(i)}\Big)\\
&=\frac{1}{2}\delta\mathbf{x}^{\intercal}\Big(\sum_{i=1}^{n}V_{\mathbf{xxx}}^{(i)}F^{(i)}\Big)\delta\mathbf{x}.
\end{aligned}
$$

After equating (12) with (16), and canceling repeated terms, we obtain

$$
\begin{aligned}
&-\frac{\mathrm{d}V}{\mathrm{d}t}-\delta\mathbf{x}^{\intercal}\frac{\mathrm{d}V_{\mathbf{x}}}{\mathrm{d}t}-\frac{1}{2}\delta\mathbf{x}^{\intercal}\frac{\mathrm{d}V_{\mathbf{xx}}}{\mathrm{d}t}\delta\mathbf{x}\\
&=\min_{\delta\mathbf{u}}\max_{\delta\mathbf{v}}\Big\{\mathcal{L}+\mathcal{L}_{\mathbf{x}}^{\intercal}\delta\mathbf{x}+\mathcal{L}_{\mathbf{u}}^{\intercal}\delta\mathbf{u}+\mathcal{L}_{\mathbf{v}}^{\intercal}\delta\mathbf{v}\\
&+\frac{1}{2}\begin{bmatrix}\delta\mathbf{x}^{\intercal} & \delta\mathbf{u}^{\intercal} & \delta\mathbf{v}^{\intercal}\end{bmatrix}\begin{bmatrix}\mathcal{L}_{\mathbf{xx}} & \mathcal{L}_{\mathbf{xu}} & \mathcal{L}_{\mathbf{xv}}\\ \mathcal{L}_{\mathbf{ux}} & \mathcal{L}_{\mathbf{uu}} & \mathcal{L}_{\mathbf{uv}}\\ \mathcal{L}_{\mathbf{vx}} & \mathcal{L}_{\mathbf{vu}} & \mathcal{L}_{\mathbf{vv}}\end{bmatrix}\begin{bmatrix}\delta\mathbf{x}\\ \delta\mathbf{u}\\ \delta\mathbf{v}\end{bmatrix}\\
&+V_{\mathbf{x}}^{\intercal}F_{\mathbf{x}}\delta\mathbf{x}+V_{\mathbf{x}}^{\intercal}F_{\mathbf{u}}\delta\mathbf{u}+V_{\mathbf{x}}^{\intercal}F_{\mathbf{v}}\delta\mathbf{v}\\
&+\delta\mathbf{x}^{\intercal}V_{\mathbf{xx}}F_{\mathbf{x}}\delta\mathbf{x}+\delta\mathbf{x}^{\intercal}V_{\mathbf{xx}}F_{\mathbf{u}}\delta\mathbf{u}+\delta\mathbf{x}^{\intercal}V_{\mathbf{xx}}F_{\mathbf{v}}\delta\mathbf{v}\Big\}\\
&=\min_{\delta\mathbf{u}}\max_{\delta\mathbf{v}}\Big\{\mathcal{L}+\delta\mathbf{x}^{\intercal}Q_{\mathbf{x}}+\delta\mathbf{u}^{\intercal}Q_{\mathbf{u}}+\delta\mathbf{v}^{\intercal}Q_{\mathbf{v}}\\
&+\frac{1}{2}\delta\mathbf{x}^{\intercal}Q_{\mathbf{xx}}\delta\mathbf{x}+\frac{1}{2}\delta\mathbf{u}^{\intercal}Q_{\mathbf{uu}}\delta\mathbf{u}+\frac{1}{2}\delta\mathbf{v}^{\intercal}Q_{\mathbf{vv}}\delta\mathbf{v}+\delta\mathbf{u}^{\intercal}Q_{\mathbf{ux}}\delta\mathbf{x}\\
&+\delta\mathbf{v}^{\intercal}Q_{\mathbf{vx}}\delta\mathbf{x}+\delta\mathbf{u}^{\intercal}Q_{\mathbf{uv}}\delta\mathbf{v}\Big\},
\end{aligned}\tag{17}
$$

where

$$
\begin{aligned}
Q_{\mathbf{x}}&=F_{\mathbf{x}}^{\intercal}V_{\mathbf{x}}+\mathcal{L}_{\mathbf{x}},\\
Q_{\mathbf{u}}&=F_{\mathbf{u}}^{\intercal}V_{\mathbf{x}}+\mathcal{L}_{\mathbf{u}},\\
Q_{\mathbf{v}}&=F_{\mathbf{v}}^{\intercal}V_{\mathbf{x}}+\mathcal{L}_{\mathbf{v}},\\
Q_{\mathbf{xx}}&=\mathcal{L}_{\mathbf{xx}}+2V_{\mathbf{xx}}F_{\mathbf{x}},\\
Q_{\mathbf{uu}}&=\mathcal{L}_{\mathbf{uu}},\\
Q_{\mathbf{vv}}&=\mathcal{L}_{\mathbf{vv}},\\
Q_{\mathbf{ux}}&=F_{\mathbf{u}}^{\intercal}V_{\mathbf{xx}}+\mathcal{L}_{\mathbf{ux}},\\
Q_{\mathbf{vx}}&=F_{\mathbf{v}}^{\intercal}V_{\mathbf{xx}}+\mathcal{L}_{\mathbf{vx}},\\
Q_{\mathbf{uv}}&=\mathcal{L}_{\mathbf{uv}}.
\end{aligned}\tag{18}
$$

To find the optimal control $\delta\mathbf{u}^{*}$ and $\delta\mathbf{v}^{*}$, we compute the gradients of the expression in (17) with respect to $\delta\mathbf{u}$ and $\delta\mathbf{v}$, respectively, and make them equal to zero to obtain:

$$
\delta\mathbf{u}^{*}=-Q_{\mathbf{uu}}^{-1}\Big(Q_{\mathbf{ux}}\delta\mathbf{x}+Q_{\mathbf{uv}}\delta\mathbf{v}^{*}+Q_{\mathbf{u}}\Big),\tag{19}
$$

$$
\delta\mathbf{v}^{*}=-Q_{\mathbf{vv}}^{-1}\Big(Q_{\mathbf{vx}}\delta\mathbf{x}+Q_{\mathbf{vu}}\delta\mathbf{u}^{*}+Q_{\mathbf{v}}\Big),\tag{20}
$$

where $Q_{\mathbf{vu}}=Q_{\mathbf{uv}}^{\intercal}$. Notice that $\delta\mathbf{v}^{*}$ is still in the previous expression of $\delta\mathbf{u}^{*}$. We need to replace the $\delta\mathbf{v}^{*}$ term in (19) with (20) and solve for $\delta\mathbf{u}^{*}$. Similarly, we can solve for $\delta\mathbf{v}^{*}$. The final expressions for $\delta\mathbf{u}^{*}$ and $\delta\mathbf{v}^{*}$ are specified as follows:

$$
\delta\mathbf{u}^{*}=\mathbf{l}_{\mathbf{u}}+\mathbf{L}_{\mathbf{u}}\delta\mathbf{x}\quad\text{and}\quad\delta\mathbf{v}^{*}=\mathbf{l}_{\mathbf{v}}+\mathbf{L}_{\mathbf{v}}\delta\mathbf{x},\tag{21}
$$

with the feed-forward gains $\mathbf{l}_{\mathbf{v}},\mathbf{l}_{\mathbf{u}}$ and feedback gains $\mathbf{L}_{\mathbf{v}},\mathbf{L}_{\mathbf{u}}$ defined as:

$$
\mathbf{l}_{\mathbf{u}}=-\left(Q_{\mathbf{uu}}-Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vu}}\right)^{-1}\left(Q_{\mathbf{u}}-Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{v}}\right),\tag{22}
$$

$$
\mathbf{l}_{\mathbf{v}}=-\left(Q_{\mathbf{vv}}-Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{uv}}\right)^{-1}\left(Q_{\mathbf{v}}-Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{u}}\right),\tag{23}
$$

$$
\mathbf{L}_{\mathbf{u}}=-\left(Q_{\mathbf{uu}}-Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vu}}\right)^{-1}\left(Q_{\mathbf{ux}}-Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vx}}\right),\tag{24}
$$

$$
\mathbf{L}_{\mathbf{v}}=-\left(Q_{\mathbf{vv}}-Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{uv}}\right)^{-1}\left(Q_{\mathbf{vx}}-Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{ux}}\right).\tag{25}
$$

In many applications in engineering, we can design the cost function. In order to see the effect that the design of the cost function has on the feed-forward and feedback gains, we recall that $Q_{\mathbf{uu}}=\mathcal{L}_{\mathbf{uu}}$ and $Q_{\mathbf{vv}}=\mathcal{L}_{\mathbf{vv}}$. Moreover, since $\mathcal{L}_{\mathbf{uu}},\mathcal{L}_{\mathbf{vv}}$ are design parameters, we can choose them such that $\mathcal{L}_{\mathbf{uu}}>0$ and $\mathcal{L}_{\mathbf{vv}}<0$. Note also that the positive definiteness of $\mathcal{L}_{\mathbf{uu}}$ and negative definiteness of $\mathcal{L}_{\mathbf{vv}}$ are required since the role of the first controller/player is to minimize the cost while the role of the second controller/player is to maximize it. Given new $Q_{\mathbf{uu}}>0$ and $Q_{\mathbf{vv}}<0$ we have the following expressions

$$
\begin{aligned}
&Q_{\mathbf{uu}}-Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vu}}>0\Rightarrow\left(Q_{\mathbf{uu}}-Q_{\mathbf{uv}}Q_{\mathbf{vv}}^{-1}Q_{\mathbf{vu}}\right)^{-1}>0,\\
&Q_{\mathbf{vv}}-Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{uv}}<0\Rightarrow\left(Q_{\mathbf{vv}}-Q_{\mathbf{vu}}Q_{\mathbf{uu}}^{-1}Q_{\mathbf{uv}}\right)^{-1}<0.
\end{aligned}
$$

The previous matrix inequalities show that the feed-forward and feedback part of the control policies of the two players will operate such that the first player aims at reducing the cost while the second player aims at increasing it. An interesting characteristic of trajectory optimization methods such as DDP is that they provide the locally optimal state trajectory, optimal feed-forward control and locally optimal feedback gains. Here we show how the feed-forward and feedback parts of the correction terms $\delta\mathbf{u}$ and $\delta\mathbf{v}$ depend on the design of the cost function. In the simulation section we demonstrate the effect of the cost function on the feed-forward and feedback parts of the minimizing control policy

for different values of $\mathcal{L}_{\mathbf{vv}}$.

A. Backward Propagation of the Value Function

The next step is to substitute the optimal control (19) and disturbance (destabilizing control) (20) to the HJBI equation (3) in order to find the update law of the value function and its first and second order partial derivatives. Specifically, we have:

$$
\begin{aligned}
&-\frac{\mathrm{d}V}{\mathrm{d}t} - \delta\mathbf{x}^{\intercal}\frac{\mathrm{d}V_{\mathbf{x}}}{\mathrm{d}t} - \frac{1}{2}\delta\mathbf{x}^{\intercal}\frac{\mathrm{d}V_{\mathbf{xx}}}{\mathrm{d}t}\delta\mathbf{x} \\
&= \mathcal{L} + \delta\mathbf{x}^{\intercal}Q_{\mathbf{x}} + \delta\mathbf{u}^{*\intercal}Q_{\mathbf{u}} + \delta\mathbf{v}^{*\intercal}Q_{\mathbf{v}} + \delta\mathbf{u}^{*\intercal}Q_{\mathbf{ux}}\delta\mathbf{x} \\
&+ \frac{1}{2}\delta\mathbf{u}^{*\intercal}Q_{\mathbf{uu}}\delta\mathbf{u}^{*} + \delta\mathbf{u}^{*\intercal}Q_{\mathbf{uv}}\delta\mathbf{v}^{*} + \frac{1}{2}\delta\mathbf{v}^{*\intercal}Q_{\mathbf{vv}}\delta\mathbf{v}^{*} \\
&+ \delta\mathbf{v}^{*\intercal}Q_{\mathbf{vx}}\delta\mathbf{x} + \frac{1}{2}\delta\mathbf{x}^{\intercal}Q_{\mathbf{xx}}\delta\mathbf{x} \\
&= \mathcal{L} + \delta\mathbf{x}^{\intercal}Q_{\mathbf{x}} + \big(\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x}\big)^{\intercal}Q_{\mathbf{u}} + \big(\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}\big)^{\intercal}Q_{\mathbf{v}} \\
&+ \big(\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x}\big)^{\intercal}Q_{\mathbf{ux}}\delta\mathbf{x} + \big(\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}\big)^{\intercal}Q_{\mathbf{vx}}\delta\mathbf{x} \\
&+ \frac{1}{2}\delta\mathbf{x}^{\intercal}Q_{\mathbf{xx}}\delta\mathbf{x} + \big(\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x}\big)^{\intercal}Q_{\mathbf{uv}}\big(\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}\big) \\
&+ \frac{1}{2}\big(\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x}\big)^{\intercal}Q_{\mathbf{uu}}\big(\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}\delta\mathbf{x}\big) \\
&+ \frac{1}{2}\big(\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}\big)^{\intercal}Q_{\mathbf{vv}}\big(\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}\delta\mathbf{x}\big). \qquad (26)
\end{aligned}
$$

After collecting terms on the right-hand side of (26) as zeroth order, first order and second order expressions of $\delta\mathbf{x}$, we can equate the coefficients of $\delta\mathbf{x}$ on the left-hand side and right-hand side of (26) and readily obtain the backward propagation equations with respect to the value function and its first and second order partial derivatives. These backward differential equations are expressed as follows

$$
\begin{aligned}
-\frac{\mathrm{d}V}{\mathrm{d}t} &= \mathcal{L} + \mathbf{l}_{\mathbf{u}}^{\intercal}Q_{\mathbf{u}} + \mathbf{l}_{\mathbf{v}}^{\intercal}Q_{\mathbf{v}} + \frac{1}{2}\mathbf{l}_{\mathbf{u}}Q_{\mathbf{uu}}\mathbf{l}_{\mathbf{u}} + \mathbf{l}_{\mathbf{u}}^{\intercal}Q_{\mathbf{uv}}\mathbf{l}_{\mathbf{v}} \\
&\quad + \frac{1}{2}\mathbf{l}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vv}}\mathbf{l}_{\mathbf{v}}, \\
-\frac{\mathrm{d}V_{\mathbf{x}}}{\mathrm{d}t} &= Q_{\mathbf{x}} + \mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{v}} + Q_{\mathbf{ux}}^{\intercal}\mathbf{l}_{\mathbf{u}} + Q_{\mathbf{vx}}^{\intercal}\mathbf{l}_{\mathbf{v}} \\
&\quad + \mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{uu}}\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{uv}}\mathbf{l}_{\mathbf{v}} + \mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vu}}\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vv}}\mathbf{l}_{\mathbf{v}}, \\
-\frac{\mathrm{d}V_{\mathbf{xx}}}{\mathrm{d}t} &= 2\mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{ux}} + 2\mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vx}} + 2\mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vu}}\mathbf{L}_{\mathbf{u}} \\
&\quad + \mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{uu}}\mathbf{L}_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vv}}\mathbf{L}_{\mathbf{v}} + Q_{\mathbf{xx}}.
\end{aligned} \qquad (27)
$$

In many applications in engineering the cost function is designed such that the terms $\mathcal{L}_{\mathbf{vu}} = \mathcal{L}_{\mathbf{uv}}^{\intercal} = 0$. In this case the differential equations for the backward propagation of the value function are simplified as follows

$$
-\frac{\mathrm{d}V}{\mathrm{d}t} = \mathcal{L} + \mathbf{l}_{\mathbf{u}}^{\intercal}Q_{\mathbf{u}} + \mathbf{l}_{\mathbf{v}}^{\intercal}Q_{\mathbf{v}} + \frac{1}{2}\mathbf{l}_{\mathbf{u}}^{\intercal}Q_{\mathbf{uu}}\mathbf{l}_{\mathbf{u}} + \frac{1}{2}\mathbf{l}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vv}}\mathbf{l}_{\mathbf{v}}, \qquad (28)
$$

$$
\begin{aligned}
-\frac{\mathrm{d}V_{\mathbf{x}}}{\mathrm{d}t} &= Q_{\mathbf{x}} + \mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{v}} + Q_{\mathbf{ux}}^{\intercal}\mathbf{l}_{\mathbf{u}} + Q_{\mathbf{vx}}^{\intercal}\mathbf{l}_{\mathbf{v}} \\
&\quad + \mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{uu}}\mathbf{l}_{\mathbf{u}} + \mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vv}}\mathbf{l}_{\mathbf{v}}, \qquad (29) \\
-\frac{\mathrm{d}V_{\mathbf{xx}}}{\mathrm{d}t} &= 2\mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{ux}} + 2\mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vx}} + \mathbf{L}_{\mathbf{u}}^{\intercal}Q_{\mathbf{uu}}\mathbf{L}_{\mathbf{u}}, \\
&\quad + \mathbf{L}_{\mathbf{v}}^{\intercal}Q_{\mathbf{vv}}\mathbf{L}_{\mathbf{v}} + Q_{\mathbf{xx}}. \qquad (30)
\end{aligned}
$$

The backward differential equations in (27) and (28) are different with respect to the corresponding backward equations in the discrete time formulation of min-max DDP in [1] and [2]. Besides the form of the backward differential equations, one of the major differences between the discrete and continuous time formulations is on the specification of the terms $Q_{\mathbf{uu}}$ and $Q_{\mathbf{vv}}$. In the continuous case these terms are specified by $\mathcal{L}_{\mathbf{uu}}$ and $\mathcal{L}_{\mathbf{vv}}$ and therefore they are completely specified by the user. This is not the case with the discrete time formulation of min-max DDP (see equations (10) and (11) in [2]) in which the terms $Q_{\mathbf{uu}}$ and $Q_{\mathbf{vv}}$ are also functions of $V_{\mathbf{xx}}$, besides $\mathcal{L}_{\mathbf{uu}}$ and $\mathcal{L}_{\mathbf{vv}}$. The result of this observation is that for the discrete time case the positive definiteness of $Q_{\mathbf{uu}}$ and the negative definiteness of $Q_{\mathbf{vv}}$ along the nominal trajectories are not guaranteed. As we show in our derivation, this is not the case with the continuous time formulation of GT-DDP and therefore the continuous version is numerically more stable that the discrete time.

III. Terminal Conditions and the Minimax DDP Algorithm

In this section, we first specify the terminal condition for the backward differential equations with respect to the value function and its first and second order partial derivatives.

At the final time, we have (4). By taking the Taylor series expansions around $\bar{\mathbf{x}}(t_f)$ we get

$$
\begin{aligned}
\phi(\mathbf{x}(t_f), t_f) &= \phi(\bar{\mathbf{x}}(t_f) + \delta\mathbf{x}(t_f), t_f) \\
&\approx \phi(\bar{\mathbf{x}}(t_f), t_f) + \delta\mathbf{x}(t_f)^{\intercal}\phi_{\mathbf{x}}(\bar{\mathbf{x}}(t_f), t_f) \\
&\quad + \delta\mathbf{x}(t_f)^{\intercal}\phi_{\mathbf{xx}}(\bar{\mathbf{x}}(t_f), t_f)\delta\mathbf{x}(t_f) \qquad (31)
\end{aligned}
$$

Therefore, the boundary conditions at $t = t_f$ for the backward differential equations, up to second order, are given by

$$
\begin{aligned}
V(t_f) &= \phi(\bar{\mathbf{x}}(t_f), t_f), \qquad (32) \\
V_{\mathbf{x}}(t_f) &= \phi_{\mathbf{x}}(\bar{\mathbf{x}}(t_f), t_f), \qquad (33) \\
V_{\mathbf{xx}}(t_f) &= \phi_{\mathbf{xx}}(\bar{\mathbf{x}}(t_f), t_f). \qquad (34)
\end{aligned}
$$

The GT-DDP algorithm is provided in Algorithm 1.

IV. Simulation Results

In this section, we apply our algorithm to two systems, namely, the inverted pendulum and the two-player pursuit evasion game under an external flow field. The dynamics of the first problem is affine in control and the cost is quadratic in control, whereas in the second problem, the dynamics is nonlinear in control and the cost function is non-quadratic.

A. Inverted Pendulum Problem

We first apply our algorithm on the inverted pendulum with conflicting controls. In particular, the dynamics is given by $I\ddot{\theta} + b\dot{\theta} - mg\ell\sin\theta = \mathbf{u} - \mathbf{v}$, where the parameters are chosen in the simulations as $m = 1$ Kg, $\ell = 0.5$ m, $b = 0.1$, $I = m\ell^2$, $g = 9.81$ Kgm/sec^2. Our goal is to bring the pendulum from the initial state $[\theta, \dot{\theta}] = [\pi, 0]$ to $[\theta, \dot{\theta}] = [0, 0]$. The cost function is given by $J = \mathbf{x}(t_f)_{\intercal}Q_f\mathbf{x}(t_f) +$

$\int_0^{t_f} (\mathbf{u}_T R_\mathbf{u} \mathbf{u} - \mathbf{v}_T R_\mathbf{v} \mathbf{v})\, \mathrm{d}t$, where $\mathbf{x} = [\theta, \dot{\theta}]_T$, the terminal cost weight matrix $Q_f = \begin{bmatrix} 100 & 0 \\ 0 & 5 \end{bmatrix}$ and $R_\mathbf{u} = 0.1$, $R_\mathbf{v} = 0.2$.

We set the initial control to be $\mathbf{u} = \mathbf{v} \equiv 0$, the terminal time to be $t_f = 0.5$ and the multiplier $\gamma = 0.8$. As can be seen in Figure 1, the cost converges in 4 iterations. We include 10 iterations to ensure convergence. Figure 2 presents the optimal controls of $\mathbf{u}$ and $\mathbf{v}$ at the 10th iteration, as well as the corresponding optimal trajectories of the states $\theta, \dot{\theta}$.

Algorithm 1 GT-DDP Algorithm

Input: Initial condition of the dynamics $\mathbf{x}_0$, initial stabilizing control $\bar{\mathbf{u}}$ and destabilizing control $\bar{\mathbf{v}}$, final time t_f, multiplier γ and a positive constant ϵ.

Output: Optimal stabilizing control $\mathbf{u}^*$, optimal destabilizing control $\mathbf{v}^*$ and the corresponding optimal gains $\mathbf{l_u}, \mathbf{L_u}, \mathbf{l_v}, \mathbf{L_v}$.

procedure UPDATE_CONTROL($\mathbf{x}_0$, $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$, t_f)
  while $\phi(\bar{\mathbf{x}}(t_f), t_f) > \epsilon$ **do**
    Get the initial trajectory $\bar{\mathbf{x}}$ by integrating controlled dynamics forward with $\mathbf{x}_0$, $\bar{\mathbf{u}}$ and $\bar{\mathbf{v}}$;
    Compute the value of $V, V_\mathbf{x}, V_\mathbf{xx}$ at t_f according to (32)-(34);
    Integrate backward the Riccati equations (27);
    Compute $\mathbf{l_u}, \mathbf{L_u}, \mathbf{l_v}, \mathbf{L_v}$ from (22) through (25);
    Integrate (6) forward by replacing $\delta\mathbf{u}$ and $\delta\mathbf{v}$ with $(\mathbf{l_u} + \mathbf{L_u}\delta\mathbf{x})$ and $(\mathbf{l_v} + \mathbf{L_v}\delta\mathbf{x})$, respectively, to get $\delta\mathbf{x}(t)$;
    Compute $\delta\mathbf{u} = \mathbf{l_u} + \mathbf{L_u}\delta\mathbf{x}$ and $\delta\mathbf{v} = \mathbf{l_v} + \mathbf{L_v}\delta\mathbf{x}$;
    Update control $\mathbf{u}^* = \mathbf{u}^* + \gamma\delta\mathbf{u}$, where $\gamma \in [0, 1]$;
    Set $\bar{\mathbf{u}} = \mathbf{u}^*$ and $\bar{\mathbf{v}} = \bar{\mathbf{v}}^*$;
  end while
  return $\mathbf{u}^*$, $\mathbf{v}^*$, $\mathbf{l_u}, \mathbf{L_u}, \mathbf{l_v}, \mathbf{L_v}$
end procedure

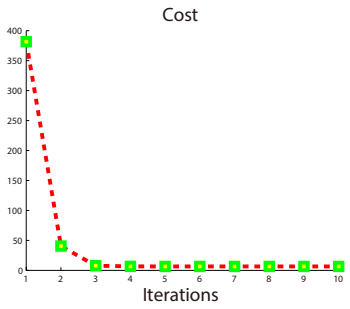

Fig. 1: Cost per iteration of the inverted pendulum with conflicting controls.

B. Inverted Pendulum Problem with Stochastic Disturbances

In this subsection, we utilize GT-DDP to guide the inverted pendulum to the desired state under the presence of stochastic disturbance that acts in the same channel as the control. Our goal is to analyze in simulation how the min-max formulation of GT-DDP affects the resulting feedfoward and feedback parts of the minimizing control policy. To this end, we consider the dynamics of the form $I\ddot{\theta} + b\dot{\theta} - mg\ell \sin\theta = \mathbf{u} + \omega$, where ω is a Gaussian noise with mean 0 and variance σ^2. The task for GT-DDP is to drive the inverted pendulum from the initial state $[\theta, \dot{\theta}] = [\pi, 0]$ to the final state $[\theta, \dot{\theta}] = [0, 0]$.

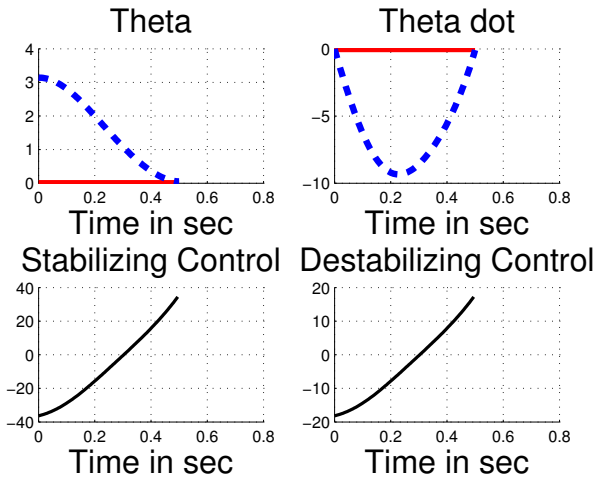

Fig. 2: Optimal controls $\mathbf{u}$ and $\mathbf{v}$ in black at the bottom and the corresponding initial trajectories of the states $\theta, \dot{\theta}$ in dashed blue at the top. Red lines represent the desired terminal states.

For our simulations, we set $\sigma = 4$ and pick $R_\mathbf{v} = 10, 0.2, 0.13$ for comparison. For every value of $R_\mathbf{v}$, we run the system with our modified control for 100 times. In Figure 3, we have three colored plots, where magenta, blue and cyan plots correspond to the case of $R_\mathbf{v} = 10, 0.2$ and 0.13, respectively. The plot of each color depicts the mean of 100 trajectories of θ with respect to time and we draw an error bar at every time step. Each error bar has a distance of the standard variance at that time step above and below the curve. Similarly, in Figure 4, we illustrate the mean and standard deviation of 100 trajectories of $\dot{\theta}$ with respect to time for the different values of $R_\mathbf{v}$.

Our simulations reveal the role of the min-max formulation of GT-DDP. In particular, Figures 3 and 4 illustrate that as $R_\mathbf{v}$ decreases, both the feed-forward and feedback parts of the control policy change. The feed-forward control steers the mean trajectory towards the desired state early for smaller values of $R_\mathbf{v}$. In addition, the locally optimal feedback gains reduce the variability of the trajectories as $R_\mathbf{v}$ decreases. The aforementioned observations indicate the interplay between the feed-forward and feedback part of the minimizing control policy under GT-DDP formulation and show how this formulation results in robust policies that shape both the mean and variance of optimal trajectories. We believe that these findings are important not only for the areas of engineering and robotics but also for modeling risk sensitive behaviors of bio-mechanical and neuromuscular systems.

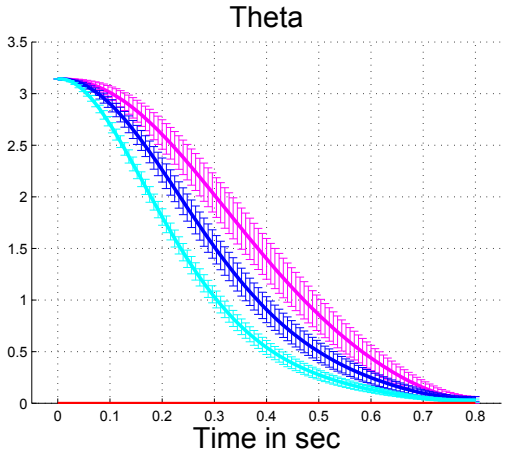

Fig. 3: Magenta, blue and cyan plots correspond to the case of $R_{\mathbf{v}} = 10, 0.2$ and 0.13, respectively. Each plot represents mean and standard variance of 100 trajectories of θ with respect to time. The red line at the bottom depicts the desired state $\theta = 0$.

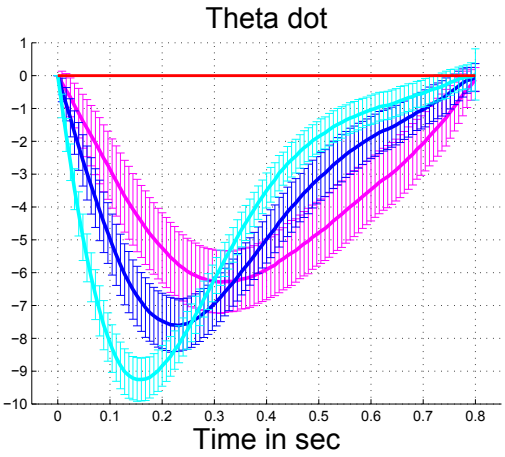

Fig. 4: Magenta, blue and cyan plots correspond to the case of $R_{\mathbf{v}} = 10, 0.2$ and 0.13, respectively. Each plot represents mean and standard variance of 100 trajectories of $\dot{\theta}$ with respect to time. The red line depicts the desired state $\dot{\theta} = 0$

V. Conclusion

In this paper, we consider a differential game problem involving two conflicting controls. By taking a Taylor series expansion of the HJBI equation around a nominal trajectory, we find the update law of both controls/players, as well as the backward propagation equations of the zeroth, first and second order approximation terms of the value function. The resulting GT-DDP algorithm, is derived using first order expansion of the dynamics in continuous time. We test GT-DDP on the inverted pendulum with conflicting controls. Finally, we demonstrate the effect of the design of the cost function to the feed-forward and feedback parts of the control policies. In particular our simulations suggests that the min-max formulation results in more robust performance. As shown in Figures 3 and 4, the profiles of the state trajectories suggest that the effect of stochastic disturbances is reduced in the GT-DDP formulation as $R_{\mathbf{v}}$ decreases.

Future research includes applications of this method to more realistic systems and dynamics with many degrees of freedom. It will also be attempted to extend this work to a stochastic version of GT-DDP for solving stochastic differential game problems. The stochastic version of GT-DDP will have direct connections to risk sensitivity and plethora of applications starting from neuromuscular and bio-mechanical systems to stochastic pursuit-evasion problems. Application to neuromuscular systems will require the extension of GT-DDP to systems with control limits and state constraints.

Acknowledgement: Funding for this research was provided by the NSF (awards CMMI-1160780 and NRI-1426945) and AFOSR (award FA9550-13-1-0029).

References

[1] J. Morimoto, G. Zeglin, and C. Atkeson, "Minimax differential dynamic programming: application to a biped walking robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp. 1927–1932 vol.2, Oct. 2003.

[2] J. Morimoto and C. Atkeson, "Minimax differential dynamic programming: An application to robust biped walking," in *In Advances in Neural Information Processing Systems 15*, Cambridge, MA: MIT Press, 2002.

[3] E. Todorov and W. Li, "A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems," in *American Control Conference*, (Portland, OR), pp. 300–306, June 8–10 2005.

[4] E. Theodorou, Y. Tassa, and E. Todorov, "Stochastic differential dynamic programming," in *American Control Conference*, (Baltimore, MD), pp. 1125–1132, June 2010.

[5] C. Atkeson and B. Stephens, "Random sampling of states in dynamic programming," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 38, pp. 924–929, Aug 2008.

[6] T. Erez, Y. Tassa, and E. Todorov, "Infinite-horizon model predictive control for periodic tasks with contacts," in *Proceedings of Robotics: Science and Systems*, (Los Angeles, CA, USA), June 2011.

[7] P. Abbeel, A. Coates, M. Quigley, and A. Y. Ng, "An application of reinforcement learning to aerobatic helicopter flight," in *In Advances in Neural Information Processing Systems 19*, p. 2007, MIT Press, 2007.

[8] Y. Tassa, N. Mansard, and E. Todorov, "Control-limited differential dynamic programming," in *Proceedings International Conference on Robotics and Automation*, (Seattle, WA), 2014.

[9] D. H. Jacobson and D. Q. Mayne, *Differential Dynamic Programming*. New York,: American Elsevier Pub. Co., 1970.

[10] W. Denham, "Differential dynamic programming," *IEEE Transactions on Automatic Control*, vol. 16, pp. 389–390, Aug 1971.