

Inverse Open-Loop Noncooperative Differential Games and Inverse Optimal Control

Timothy L. Molloy, Jairo Inga, Michael Flad, Jason J. Ford, Tristan Perez, and Sören Hohmann

Abstract—We consider the problem of computing parameters of player cost functionals such that given state and control trajectories constitute an open-loop Nash equilibrium for a noncooperative differential game. We propose two methods for solving this inverse differential game problem and novel conditions under which our methods compute unique cost-functional parameters. Our conditions are analogous to persistence of excitation conditions in adaptive control and parameter estimation. The efficacy of our methods is illustrated in simulations.

Index Terms—Game theory, inverse differential games, inverse optimal control, optimal control.

I. INTRODUCTION

Numerous problems in automatic control [1]–[4], economics [5], [6], and biology [7], [8] involve modelling and analysing interactions between players (or decision makers) in processes governed by differential equations. Techniques for finding the optimal controls (or decisions) of players in noncooperative differential games given their potentially conflicting objectives have therefore been well studied (cf. [2]). The inverse problem of computing the objectives of players in noncooperative differential games from player controls has received considerably less attention [7], [9], [10]. In this paper, we propose two inverse noncooperative differential game methods and derive novel conditions under which they yield unique solutions.

The inverse problem of computing parameters of player cost functions from player decisions has been well studied in static games (i.e., games without state dynamics) and in dynamic games with state dynamics governed by processes other than differential equations [11]–[21]. Previous work on inverse static game problems has extended methods of inverse optimisation by exploiting the equivalence of one-player games and optimisation problems. For example, the idea of using Karush-Kuhn-Tucker conditions to solve inverse optimisation problems proposed in [22] has inspired several authors to solve inverse static game problems by exploiting conditions for the existence of equilibria [11], [12], [14].

Most prior treatments of inverse dynamic game problems have been in the discrete-time setting with state dynamics

described by either Markov processes with discrete (and finite) state and control spaces [17]–[21] or difference equations [23]–[25]. Many of these treatments (particularly those involving difference equations) have also focused on two-player games [23], [25]. A notable exception is [24] where the equivalence of one-player dynamic games and discrete-time optimal control problems is used to propose a method of inverse N -player noncooperative dynamic games based on the inverse optimal control method of [26].

There has been recent interest in generalising methods of discrete-time inverse dynamic games to the continuous-time setting with a focus on state dynamics governed by differential equations [7], [9], [10], [27]. These methods have primarily involved reducing the inverse differential game problem to inverse optimal control problems that can be solved with existing inverse optimal control techniques [9], [10]. The reductions have been performed under the assumption of either an open-loop information structure (cf. [9]), or knowledge of all but one of the player control laws (cf. [10]).

Due to the close relationship between the problems of inverse differential games and inverse optimal control, existing methods of inverse differential games and inverse optimal control share the same limitations. For example, the bilevel methods of inverse optimal control of [28] and [29] have computationally expensive implementations that involve the solution of optimal control problems inside numeric optimisation routines. The bilevel and nested optimisation methods of inverse differential games proposed in [7], [9] and derived from these bilevel methods of inverse optimal control thus also have computationally implementations that scale poorly with the number of players, and state and control dimensions. Similarly, the minimum principle method of inverse optimal control from [30] is unable to operate on trajectories with constrained controls, and so the minimum principle method of inverse differential games presented in [9] and [10] are also unable to handle constrained controls. Furthermore, few inverse optimal control methods have established conditions for the existence and uniqueness of solutions (with the notable exception of [31] for the special case of differentially-flat systems). Hence, most (if not all) inverse differential game methods also lack conditions for the existence and uniqueness of solutions, even in idealised settings.

The key contribution of this paper is the proposal of two methods of inverse differential games and the development of algebraic conditions under which they are guaranteed to compute unique player cost-functional parameters. Our first method builds upon the inverse differential game and inverse optimal control methods of [9], [10], [30] by seeking cost-

T. L. Molloy and J. J. Ford are with the School of Electrical Engineering and Computer Science, Queensland University of Technology (QUT), Brisbane, Australia (e-mail: {t.molloy,j2.ford}@qut.edu.au)

T. Perez was with the School of Electrical Engineering and Computer Science, Queensland University of Technology (QUT), Brisbane, Australia. He is now with Boeing Research and Technology Australia, St Lucia, Australia (e-mail: tristan.perez@boeing.com)

J. Inga, M. Flad, and S. Hohmann are with the Institute of Control Systems, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany (e-mail: {jairo.inga,flad,soeren.hohmann}@kit.edu)

The work of TM and TP was partially supported by an ARC Linkage Grant LP130100483 with the support of Boeing Research and Technology Australia.

functional parameters such that necessary conditions for the existence of open-loop Nash equilibria hold approximately with minimal violation. Our second method addresses the inability of our first method to handle trajectories with constrained controls whilst still enjoying similar guarantees on the uniqueness of solutions and not resorting to bilevel optimisation. Although open-loop differential games can also be reformulated as (static) infinite-dimensional continuous games (cf. [3]), we do not exploit this insight here and instead focus on connections between inverse differential games and inverse optimal control. Indeed, the methods and results of this paper directly provide new insight into the solution of inverse optimal control problems (particularly those involving constraints on the controls).

This paper is organised as follows. In Section II, we introduce the inverse open-loop noncooperative differential game problem. In Section III, we propose two inverse differential game methods before analysing and solving them as quadratic programs in Section IV. Finally, we present illustrative simulation results in Section V and conclusions in Section VI.

II. PROBLEM FORMULATION

Let us consider a continuous-time N -player noncooperative differential game played with an open-loop information structure. The continuous-time state process of the game is

$$\dot{x}(t) = f(t, x(t), u_1(t), \dots, u_N(t)), \quad x(0) = x_0 \quad (1)$$

on the (finite) time-interval $t \in [0, T]$ where $x(t) \in \mathbb{R}^n$ is the state vector of the game, $x_0 \in \mathbb{R}^n$ is the initial state of the game, $u_i(t) \in \mathcal{U}_i$ are control inputs belonging to the control sets $\mathcal{U}_i \subset \mathbb{R}^{m_i}$ for $i \in \mathbb{N} \triangleq \{1, \dots, N\}$, and $f : [0, T] \times \mathbb{R}^n \times \mathbb{R}^{m_1} \times \dots \times \mathbb{R}^{m_N} \mapsto \mathbb{R}^n$ is a (potentially nonlinear) vector-valued function. We assume that f is continuous in t and uniformly Lipschitz in x, u_1, \dots, u_N such that (1) admits a unique solution for every N -tuple of continuous controls (u_1, \dots, u_N) (cf. [2, Theorem 5.1]). We also assume that f is continuously differentiable in x, u_1, \dots, u_N .

The controls u_i are selected by Player i to minimise

$$\begin{aligned} J_i(u_1, \dots, u_N, \theta_i) \\ \triangleq \int_0^T \theta_i' g_i(t, x(t), u_1(t), \dots, u_N(t)) dt \end{aligned} \quad (2)$$

subject to the dynamics (1) and the control constraints $u_i(t) \in \mathcal{U}_i$. Here, $\theta_i \in \Theta_i \subset \mathbb{R}^{M_i}$ are time-invariant parameters of the cost-functional (2), $g_i : [0, T] \times \mathbb{R}^n \times \mathbb{R}^{m_1} \times \dots \times \mathbb{R}^{m_N} \mapsto \mathbb{R}^{M_i}$ are basis functions that are continuously differentiable in x, u_1, \dots, u_N . We use $'$ to denote the transpose operator. The controls $(u_1^\theta, \dots, u_N^\theta)$ and associated state x^θ constitute an open-loop Nash equilibrium solution to the game (1) and (2) with cost-functional parameters θ_i if and only if they solve the N -coupled optimal control problems

$$\begin{aligned} \inf_{u_i} J_i(u_1^\theta, \dots, u_i, \dots, u_N^\theta, \theta_i) \\ \text{s.t. } \dot{x}^\theta(t) = f(t, x^\theta(t), u_1^\theta(t), \dots, u_i(t), \dots, u_N^\theta(t)) \\ u_i(t) \in \mathcal{U}_i, \quad t \in [0, T] \\ x^\theta(0) = x_0. \end{aligned} \quad (3)$$

In the inverse open-loop noncooperative differential game problem, we assume knowledge of the dynamics f , basis functions g_i , and constraint sets \mathcal{U}_i , and we are given state x and control (u_1, \dots, u_N) trajectories on $t \in [0, T]$. Our aim is to compute cost-functional parameters θ_i for all (or some) of the players such that the states x and controls (u_1, \dots, u_N) constitute an open-loop Nash equilibrium solution to the game (1) and (2). That is, we seek parameters θ_i such that the states x and controls u_i solve the optimal control problems (3) for all $i \in \mathbb{N}$. In the case of a single player (i.e., when $N = 1$), our inverse differential game problem reduces to the problem of inverse optimal control.

For our inverse differential game problem to be well posed, the dynamics f , basis functions g_i , and parameter sets Θ_i should be specified such that the states x and controls (u_1, \dots, u_N) constitute an open-loop Nash equilibrium solution to the game (1) and (2) for some (possibly non-unique) cost-functional parameters $\theta_i^* \in \Theta_i$. Addressing the selection of suitable dynamics and basis functions is beyond the scope of this paper, so where necessary, we shall make the following assumption (see also [2], [6] and references therein for discussions on the existence of open-loop Nash equilibria).

Assumption 1: The states x and controls (u_1, \dots, u_N) constitute an open-loop Nash equilibrium solution to the game (1) and (2) with dynamics f , basis functions g_i , and unknown cost-functional parameters $\theta_i = \theta_i^* \in \Theta_i$ for $i \in \mathbb{N}$.

Under Assumption 1, the inverse differential game problem will have multiple solutions in general since if the trajectories x and (u_1, \dots, u_N) solve the optimal control problems (3) with $\theta_i = \theta_i^* \in \Theta_i$, then the trajectories will also solve the problems (3) with $\theta_i = \kappa_i \theta_i^*$ for all scaling factors $\kappa_i > 0$. The zero vectors $\theta_i = 0$ are also trivial solutions to the inverse differential game problem. Without loss of generality, we shall exclude trivial solutions and ambiguous scaling by considering parameter sets of the form $\Theta_i = \{\theta_i \in \mathbb{R}^{M_i} : \theta_{i,1} = 1\}$ where $\theta_{i,1}$ denotes the first element of θ_i . The choice of the fixed-element constraint $\theta_{i,1} = 1$ is arbitrary and results analogous to those of this paper will also hold with normalisation constraints such as $\|\theta_i\| = 1$.

III. PROPOSED METHODS

In this section, we propose two methods for solving the inverse differential game problem by exploiting necessary conditions for the existence of open-loop Nash equilibria.

A. Necessary Conditions for Open-Loop Nash Equilibria

To present the necessary conditions that we shall exploit, let us define the (real-valued) Hamiltonian functions

$$\begin{aligned} H_i(t, \lambda_i(t), x(t), u_1(t), \dots, u_N(t), \theta_i) \\ \triangleq \theta_i' g_i(t, x(t), u_1(t), \dots, u_N(t)) \\ + \lambda_i'(t) f(t, x(t), u_1(t), \dots, u_N(t)) \end{aligned} \quad (4)$$

for $i \in \mathbb{N}$ where $\lambda_i : [0, T] \mapsto \mathbb{R}^n$ are costate (or adjoint) functions. Let us also use $\nabla_x H_i(t, \lambda_i(t), \theta_i) \in \mathbb{R}^n$ and $\nabla_{u_i} H_i(t, \lambda_i(t), \theta_i) \in \mathbb{R}^{m_i}$ to denote the vectors of partial derivatives of the Hamiltonian functions with respect to $x(t)$ and $u_i(t)$, respectively, evaluated with $\theta_i, x(t)$, and $u_i(t)$. With

this notation, we have the following necessary conditions for the existence of open-loop Nash equilibria.

Theorem 1 (Theorem 6.11 of [2]): Consider the noncooperative differential game with dynamics (1) and cost functionals (2). If Assumption 1 holds such that x and (u_1, \dots, u_N) constitute an open-loop Nash equilibrium of the game with $\theta_i = \theta_i^*$, then for each $i \in \mathbb{N}$, there exists a costate trajectory $\lambda_i : [0, T] \mapsto \mathbb{R}^n$ such that

$$\dot{\lambda}_i(t) = -\nabla_x H_i(t, \lambda_i(t), \theta_i) \quad (5)$$

for $t \in [0, T]$ with terminal condition $\lambda_i(T) = 0$, and

$$u_i(t) = \arg \min_{\bar{u}(t) \in \mathcal{U}_i} H_i(t, \lambda_i(t), x(t), u_1(t), \dots, u_{i-1}^*(t), \bar{u}(t), u_{i+1}^*(t), \dots, u_N(t), \theta_i) \quad (6)$$

for all $t \in [0, T]$.

Proof: See [2, Section 6.5.1]. \square

We note that the Hamiltonian condition (6) simplifies to

$$\nabla_{u_i} H_i(t, \lambda_i(t), \theta_i) = 0 \quad (7)$$

at times $t \in [0, T]$ when the controls $u_i(t)$ are in the interior (i.e., not on the boundary) of the constraint set \mathcal{U}_i . We shall exploit this gradient condition and the costate condition (5) to propose two inverse differential game methods. Our methods primarily differ in whether (5) is treated as an objective or as a constraint. For this reason, we will refer to our first method as the soft-constrained method, and our second method as the hard-constrained method.

B. Proposed Soft-Constrained Method

In our soft-constrained method, we seek player cost-functional parameters that simultaneously minimise the violation of both the costate condition (5) and the Hamiltonian condition (6) under the following assumption.

Assumption 2: The controls $u_i(t)$ are in the interiors of the control constraint sets \mathcal{U}_i for all $t \in [0, T]$ and all $i \in \mathbb{N}$.

Under Assumption 2, the Hamiltonian condition (6) reduces to (7) for all $t \in [0, T]$. Our proposed soft-constrained method is then to compute parameters $\hat{\theta}_i$ and functions $\hat{\lambda}_i : [0, T] \mapsto \mathbb{R}^n$ for each player $i \in \mathbb{N}$ by solving

$$\begin{aligned} \inf_{\lambda_i, \theta_i} \quad & \int_0^T \|\nabla_{u_i} H_i(t, \lambda_i(t), \theta_i)\|^2 \\ & + \gamma \|\dot{\lambda}_i(t) + \nabla_x H_i(t, \lambda_i(t), \theta_i)\|^2 dt \quad (8) \\ \text{s.t.} \quad & \theta_i \in \Theta_i \end{aligned}$$

where $\gamma > 0$ is a specifiable weighting factor. Our proposed soft-constrained method intuitively seeks functions $\hat{\lambda}_i$ and parameters $\hat{\theta}_i$ such that the costate condition (5) and Hamiltonian gradient condition (7) hold (at least approximately) for all $t \in [0, T]$. If Assumption 1 holds in addition to Assumption 2, then $\hat{\theta}_i = \theta_i^*$ will be a (possibly nonunique) solution to (8). If Assumptions 1 or 2 do not hold (e.g., due to misspecified dynamics or basis functions, or imperfect trajectories), then our soft-constrained method will compute parameters and functions such that (5) and (7) hold approximately with their priority assigned via choice of γ . When $N = 1$ and $\gamma = 1$, this method reduces to the inverse optimal control approach of [30].

C. Proposed Hard-Constrained Method

In our hard-constrained method, we seek to handle control trajectories with constrained controls (i.e., control trajectories u_i with times t at which $u_i(t)$ is on the boundary of the constraint set \mathcal{U}_i). Indeed, let us consider a finite sequence of sampling times $\mathbb{K}_i \triangleq \{t_k \in [0, T] : 1 \leq k \leq K_i \text{ and } 0 \leq t_1 < t_2 < \dots < t_{K_i} \leq T\}$ for each player $i \in \mathbb{N}$. The sampling times \mathbb{K}_i may be selected from within the interval $[0, T]$, including during singular arcs, provided that the following assumption holds.

Assumption 3: The controls $u_i(t)$ are in the interiors of the control constraint sets \mathcal{U}_i for all times $t \in \mathbb{K}_i$ and all $i \in \mathbb{N}$.

Under Assumption 3, the Hamiltonian condition (6) reduces to (7) for all $t \in \mathbb{K}_i$ and our proposed hard-constrained method is to compute parameters $\tilde{\theta}_i$ for each player $i \in \mathbb{N}$ by solving

$$\begin{aligned} \inf_{\tilde{\theta}_i} \quad & \sum_{k=1}^{K_i} \|\nabla_{u_i} H_i(t_k, \lambda_i(t_k), \tilde{\theta}_i)\|^2 \\ \text{s.t.} \quad & \dot{\lambda}_i(t) = -\nabla_x H_i(t, \lambda_i(t), \tilde{\theta}_i), \quad t \in [0, T] \quad (9) \\ & \lambda_i^i(T) = 0 \\ & \tilde{\theta}_i \in \Theta_i. \end{aligned}$$

Our hard-constrained method (9) entails finding parameters $\tilde{\theta}_i$ such that the Hamiltonian gradient condition (7) holds approximately at the sampling times $t \in \mathbb{K}_i$ whilst the costate condition (5) holds exactly for all $t \in [0, T]$, regardless of if Assumptions 1 and 3 hold. However, if Assumptions 1 and 3 hold, then $\tilde{\theta}_i = \theta_i^*$ will be a (possibly nonunique) solution to (9). By considering (7) at the sampling times $t \in \mathbb{K}_i$ rather than for all $t \in [0, T]$, our hard constrained method can handle trajectories in which the controls $u_i(t)$ are on the boundaries of the constraint set \mathcal{U}_i at the infinitely many times $t \in [0, T] \setminus \mathbb{K}_i$ (i.e., times $t \in [0, T]$ not also in \mathbb{K}_i).

IV. REFORMULATION AND SOLUTION OF PROPOSED METHODS

In this section, we will reformulate our proposed methods as quadratic programs. We will exploit these reformulations to provide explicit solutions to our methods, and to establish novel conditions under which the solutions are unique and correspond to the true parameters under Assumption 1.

A. Reformulation of Soft-Constrained Method

We shall reformulate our soft-constrained method by defining the rectangular matrix $\mathcal{I} \triangleq [I \ 0] \in \mathbb{R}^{M_i \times (M_i + n)}$ where I is the square identity matrix with appropriate dimensions. Let us also define $R = I \in \mathbb{R}^{n \times n}$, $B \triangleq [0 \ I]'$, $S_i(t) \triangleq [\sqrt{\gamma} \nabla_x g_i(t) \quad \sqrt{\gamma} \nabla_x f(t)]'$ and $Q_i(t) \triangleq F_i'(t) F_i(t)$ where

$$F_i(t) \triangleq \begin{bmatrix} \sqrt{\gamma} \nabla_x g_i(t) & \sqrt{\gamma} \nabla_x f(t) \\ \nabla_{u_i} g_i(t) & \nabla_{u_i} f(t) \end{bmatrix}.$$

Here, we use the shorthand $\nabla_x f(t) \in \mathbb{R}^{n \times n}$ and $\nabla_{u_i} f(t) \in \mathbb{R}^{M_i \times n}$ to denote the matrices of partial derivatives of f with respect to $x(t)$ and $u_i(t)$, respectively, and evaluated with θ_i , $x(t)$, and $u_i(t)$ for $i \in \mathbb{N}$. Similarly, we use $\nabla_x g_i(t) \in \mathbb{R}^{n \times M_i}$, and $\nabla_{u_i} g_i(t) \in \mathbb{R}^{M_i \times M_i}$ to denote the matrices of partial

derivatives of g_i evaluated with θ_i , $x(t)$, and $u_i(t)$ for $i \in \mathbb{N}$. The following lemma reformulates (8) as a quadratic program.

Lemma 1: Consider any player $i \in \mathbb{N}$ and suppose that $\gamma > 0$ is selected such that $Q_i(t) - S_i(t)\gamma^{-1}R^{-1}S_i'(t)$ is positive semidefinite for all $t \in [0, T]$. Then the pair $(\hat{\lambda}_i, \hat{\theta}_i)$ is a solution to (8) if and only if $\hat{\lambda}_i(t) = K_i(t) [\hat{\theta}_i' \quad \hat{\lambda}_i'(t)]'$ for all $t \in [0, T]$ with $\hat{\alpha}_i = [\hat{\theta}_i' \quad \hat{\lambda}_i'(0)]'$ solving

$$\inf_{\alpha_i} \alpha_i' P_i(0) \alpha_i \quad \text{s.t.} \quad \mathcal{I} \alpha_i \in \Theta_i \quad (10)$$

where $K_i(t) \triangleq -[B'P_i(t) + S_i'(t)]$ and $P_i : [0, T] \mapsto \mathbb{R}^{(M_i+n) \times (M_i+n)}$ is the unique positive semidefinite solution to the Riccati differential equation

$$\dot{P}_i(t) = (P_i(t)B + S_i(t))(B'P_i'(t) + S_i'(t)) - Q_i(t) \quad (11)$$

for $t \in [0, T]$ with terminal condition $P_i(T) = 0$.

Proof: Consider any player $i \in \mathbb{N}$. Recalling the definition of the Hamiltonians (4), the integrand in (8) is

$$\begin{aligned} & \|\nabla_{u_i} H_i(t, \lambda_i(t), \theta_i)\|^2 + \gamma \left\| \dot{\lambda}_i(t) + \nabla_x H_i(t, \lambda_i(t), \theta_i) \right\|^2 \\ &= \left\| \begin{bmatrix} \sqrt{\gamma} \dot{\lambda}_i(t) + \sqrt{\gamma} \nabla_x g_i(t) \theta_i + \sqrt{\gamma} \nabla_x f(t) \lambda_i(t) \\ \nabla_{u_i} g_i(t) \theta_i + \nabla_{u_i} f(t) \lambda_i(t) \end{bmatrix} \right\|^2 \\ &= \left\| \begin{bmatrix} \sqrt{\gamma} \nabla_x g_i(t) & \sqrt{\gamma} \nabla_x f(t) \\ \nabla_{u_i} g_i(t) & \nabla_{u_i} f(t) \end{bmatrix} \begin{bmatrix} \theta_i \\ \lambda_i(t) \end{bmatrix} + \begin{bmatrix} \gamma^{\frac{1}{2}} I \\ 0 \end{bmatrix} \dot{\lambda}_i(t) \right\|^2 \\ &= z_i'(t) Q_i(t) z_i(t) + v_i'(t) \gamma R v_i(t) + 2z_i'(t) S_i(t) v_i(t) \end{aligned}$$

where the last line follows from the definitions of $Q_i(t)$, R , and $S_i(t)$ together with the variable substitutions

$$z_i(t) \triangleq \begin{bmatrix} \theta_i \\ \lambda_i(t) \end{bmatrix} \quad \text{and} \quad v_i(t) \triangleq \dot{\lambda}_i(t).$$

The constraint $\theta_i \in \Theta$ in (8) may be rewritten as $\mathcal{I} z_i(t) \in \Theta_i$ and the (implicit) constraint in (8) that θ_i is time-invariant is

$$\dot{z}_i(t) = \begin{bmatrix} \dot{\theta}_i \\ \dot{\lambda}_i(t) \end{bmatrix} = \begin{bmatrix} 0 \\ \dot{\lambda}_i(t) \end{bmatrix} = B v_i(t).$$

The optimisation in (8) is thus equivalent to

$$\begin{aligned} & \inf_{z_i, v_i} \int_0^T z_i'(t) Q_i(t) z_i(t) + v_i'(t) \gamma R v_i(t) \\ & \quad + 2z_i'(t) S_i(t) v_i(t) dt \quad (12) \\ & \text{s.t.} \quad \dot{z}_i(t) = B v_i(t), \quad t \in [0, T] \\ & \quad \mathcal{I} z_i(t) \in \Theta_i, \quad t \in [0, T]. \end{aligned}$$

We now note that given a function $v_i : [0, T] \mapsto \mathbb{R}^n$ together with an initial value $z_i(0) = \alpha_i \in \mathbb{R}^{M_i+n}$ with $\mathcal{I} \alpha_i \in \Theta_i$, we may solve the differential equation $\dot{z}_i(t) = B v_i(t)$ for the unique function $z_i : [0, T] \mapsto \mathbb{R}^{M_i+n}$. The optimisation in (12) is therefore essentially only over $z_i(0)$ and v_i , namely,

$$\begin{aligned} & \inf_{\alpha_i} \inf_{v_i} \int_0^T z_i'(t) Q_i(t) z_i(t) + v_i'(t) \gamma R v_i(t) \\ & \quad + 2z_i'(t) S_i(t) v_i(t) dt \quad (13) \\ & \text{s.t.} \quad \dot{z}_i(t) = B v_i(t), \quad t \in [0, T] \\ & \quad z_i(0) = \alpha_i \\ & \quad \mathcal{I} \alpha_i \in \Theta_i. \end{aligned}$$

For any $\alpha_i \in \mathbb{R}^{M_i+n}$, the inner optimisation problem over the function v_i in (13) is a linear quadratic optimal control problem. Since $\gamma R = \gamma I$ is positive definite and $Q_i(t) - S_i(t)\gamma^{-1}R^{-1}S_i'(t)$ is positive semidefinite due to the lemma conditions, the optimal control results of [32, Section 3.4] imply that for any $z_i(0) = \alpha_i \in \mathbb{R}^{M_i+n}$, the unique function solving the inner optimisation over v_i in (13) is $\hat{v}_i(t) = K_i(t) z_i(t)$ for all $t \in [0, T]$. Section 3.4 of [32] also gives that the value of the inner optimisation problem over v_i in (13) is $\alpha_i' P_i(0) \alpha_i$ for any initial state $z_i(0) = \alpha_i$. The optimisation in (13) thus simplifies to (10). It follows that the pair of functions (\hat{z}_i, \hat{v}_i) solves (12) if and only if $\hat{z}_i(0) = \hat{\alpha}_i$ solves (10) and $\hat{z}_i(t) = B \hat{v}_i(t) = B K_i(t) \hat{z}_i(t)$ for $t \in [0, T]$. The proof is complete since (12) is equivalent to (8). \square

Lemma 1 allows us to reformulate our soft-constrained method (8) as the quadratic program (10). This reformulation is new for $N > 1$ but was previously established in [30] for the inverse optimal control case $N = 1$. We shall next exploit this reformulation to derive descriptions of the parameters $\hat{\theta}_i$ that solve our soft-constrained method that are novel for all cases $N \geq 1$. We also note that the choice of $\gamma = 1$ is always sufficient to ensure that $Q_i(t) - S_i(t)\gamma^{-1}R^{-1}S_i'(t)$ is positive semidefinite for all $t \in [0, T]$ since in this case

$$\begin{aligned} & Q_i(t) - S_i(t)R^{-1}S_i'(t) \\ &= [\nabla_{u_i} g_i(t) \quad \nabla_{u_i} f(t)]' [\nabla_{u_i} g_i(t) \quad \nabla_{u_i} f(t)]. \end{aligned}$$

Other values of $\gamma > 0$, $\gamma \neq 1$, may lead to $Q_i(t) - S_i(t)\gamma^{-1}R^{-1}S_i'(t)$ not being positive semidefinite and hence (11) possessing multiple solutions.

B. Solution of Soft-Constrained Method

To describe the solutions to our soft-constrained method, let \bar{P}_i be the principal submatrix of $P_i(0)$ formed by deleting the first row and column of $P_i(0)$, let \bar{p}_i be the first column of $P_i(0)$ with deleted first element, and let \bar{P}_i^+ and r_i^P be the pseudoinverse and rank of the submatrix \bar{P}_i , respectively. Finally, let $\bar{P}_i = U_i \Sigma_i^P U_i'$ be the singular value decomposition (SVD) of \bar{P}_i where $\Sigma_i^P \in \mathbb{R}^{(M_i+n-1) \times (M_i+n-1)}$, and

$$U_i = \begin{bmatrix} U_i^{11} & U_i^{12} \\ U_i^{21} & U_i^{22} \end{bmatrix} \in \mathbb{R}^{(M_i+n-1) \times (M_i+n-1)} \quad (14)$$

is a block matrix with submatrices $U_i^{11} \in \mathbb{R}^{(M_i-1) \times r_i^P}$, $U_i^{12} \in \mathbb{R}^{(M_i-1) \times (M_i+n-1-r_i^P)}$, $U_i^{21} \in \mathbb{R}^{n \times r_i^P}$ and $U_i^{22} \in \mathbb{R}^{n \times (M_i+n-1-r_i^P)}$. Our main result for our soft-constrained method (8) follows.

Theorem 2: Consider any player $i \in \mathbb{N}$, let $\Theta_i = \{\theta_i \in \mathbb{R}^{M_i} : \theta_{i,1} = 1\}$ and suppose that Assumption 2 holds and that $\gamma > 0$ is selected such that $Q_i(t) - S_i(t)\gamma^{-1}R^{-1}S_i'(t)$ is positive semidefinite for all $t \in [0, T]$. If $(I - \bar{P}_i \bar{P}_i^+) \bar{p}_i = 0$, then all of the vectors $\hat{\theta}_i$ corresponding to all solutions $(\hat{\lambda}_i, \hat{\theta}_i)$ of (8) are of the form

$$\hat{\theta}_i = \mathcal{I} \hat{\alpha}_i \quad (15)$$

where $\hat{\alpha}_i = [1 \quad \hat{\alpha}_i']' \in \mathbb{R}^{M_i+n}$ are (potentially nonunique) solutions to (10) with $\hat{\alpha}_i \in \mathbb{R}^{M_i+n-1}$ given by

$$\hat{\alpha}_i = -\bar{P}_i^+ \bar{p}_i + U_i [0 \quad b']' \quad (16)$$

for any $b \in \mathbb{R}^{M_i+n-1-r_i^{\bar{P}}}$. Furthermore, if in addition either $U_i^{12} = 0$ or $r_i^{\bar{P}} = M_i + n - 1$, then all solutions $(\hat{\lambda}_i, \hat{\theta}_i)$ to (8) correspond to the unique parameter vector $\hat{\theta}_i$ given by

$$\hat{\theta}_i = \mathcal{I} \begin{bmatrix} 1 \\ -\bar{P}_i^+ \bar{p}_i \end{bmatrix}, \quad (17)$$

with $\hat{\theta}_i = \kappa_i \theta_i^*$ if Assumption 1 holds and if there exists a $\kappa_i > 0$ such that $\kappa_i \theta_i^* \in \Theta_i$.

Proof: We apply Lemma 1 and proceed by analysing (10).

For any $\alpha_i \in \mathbb{R}^{M_i+n}$ with $\mathcal{I}\alpha_i \in \Theta_i$ we have that $\alpha_i = [1 \quad \bar{\alpha}_i']^T$ where $\bar{\alpha}_i \in \mathbb{R}^{M_i+n-1}$ and so

$$\alpha_i' P_i(0) \alpha_i = P_{i,(1,1)}(0) + \bar{\alpha}_i' \bar{P}_i \bar{\alpha}_i + 2\bar{\alpha}_i' \bar{p}_i$$

where $P_{i,(1,1)}(0)$ is the first element of $P_i(0)$. Solutions $\hat{\alpha}_i$ of (10) are thus of the form $\hat{\alpha}_i = [1 \quad \hat{\alpha}_i']^T$ where $\hat{\alpha}_i$ solves

$$\inf_{\hat{\alpha}_i} \left\{ \frac{1}{2} \bar{\alpha}_i' \bar{P}_i \bar{\alpha}_i + \bar{\alpha}_i' \bar{p}_i \right\}.$$

Under the condition $(I - \bar{P}_i \bar{P}_i^+) \bar{p}_i = 0$ and by noting that \bar{P}_i is positive semidefinite due to $P_i(0)$ being positive semidefinite, [33, Proposition 15.2] gives that this quadratic program is solved by any $\hat{\alpha}_i$ satisfying (16) for any $b \in \mathbb{R}^{M_i+n-1-r_i^{\bar{P}}}$. The first theorem assertion (15) and (16) follows.

Now, from (16), clearly, $r_i^{\bar{P}} = M_i + n - 1$ implies that $\hat{\alpha}_i = -\bar{P}_i^+ \bar{p}_i$. We also note that $U_i^{12} = 0$ implies that

$$\hat{\alpha}_i = -\bar{P}_i^+ \bar{p}_i + \begin{bmatrix} 0 \\ (U_i^{22} b)' \end{bmatrix}'$$

for any $b \in \mathbb{R}^{M_i+n-1-r_i^{\bar{P}}}$ with $U_i^{22} b \in \mathbb{R}^n$. Thus, if either $U_i^{12} = 0$ or $r_i^{\bar{P}} = M_i + n - 1$, then the first $M_i - 1$ components of $\hat{\alpha}_i$ are invariant with respect to the free vector $b \in \mathbb{R}^{M_i+n-1-r_i^{\bar{P}}}$, and so all solutions $\hat{\alpha}_i$ to (10) satisfy

$$\mathcal{I} \hat{\alpha}_i = \mathcal{I} \begin{bmatrix} 1 \\ (-\bar{P}_i^+ \bar{p}_i)' \end{bmatrix}'.$$

The second theorem assertion follows since $\hat{\theta}_i = \mathcal{I} \hat{\alpha}_i$. Theorem 1 additionally implies that $\theta_i = \kappa_i \theta_i^*$ when Assumption 1 holds and $\kappa_i \theta_i^* \in \Theta_i$. The proof is complete. \square

Theorem 2 establishes that the conditions $U_i^{12} = 0$ and $r_i^{\bar{P}} = M_i + n - 1$ are both sufficient for ensuring that our soft-constrained method (8) yields unique parameters $\hat{\theta}_i$. These conditions will not hold when the trajectories are uninformative — for example, when the trajectories x and (u_1, \dots, u_N) are too short (i.e., on short time-horizons T) or when they correspond to a dynamic equilibrium of the dynamics in the sense that $\dot{x}(t) = 0$ for all $t \in [0, T]$. The conditions $U_i^{12} = 0$ and $r_i^{\bar{P}} = M_i + n - 1$ therefore have an intuitive interpretation as persistence of excitation conditions (analogous to similar concepts in parameter estimation and adaptive control). Furthermore, the last assertion of Theorem 2 highlights that these conditions become sufficient conditions for ensuring that our soft-constrained method yields unique parameters that only differ from the true parameters θ_i^* by an unknown factor $\kappa_i > 0$ when Assumptions 1 and 2 hold.

The condition $U_i^{12} = 0$ can hold when $r_i^{\bar{P}} < M_i + n - 1$. If $U_i^{12} = 0$ but $r_i^{\bar{P}} < M_i + n - 1$, then the second assertion

of Theorem 2 implies that all pairs $(\hat{\lambda}_i, \hat{\theta}_i)$ solving our soft-constrained method will share the unique parameter vector $\hat{\theta}_i$ given by (17) but may not share a common function $\hat{\lambda}_i$. Before we illustrate this result, we shall reformulate and solve our hard-constrained method (9) as a quadratic program.

C. Reformulation of Hard-Constrained Method

Let us reformulate our proposed hard-constrained method (9) by considering any player $i \in \mathbb{N}$ and defining $W_i \in \mathbb{R}^{M_i \times M_i}$ as the positive semidefinite matrix given by

$$W_i \triangleq \sum_{k=1}^{K_i} \|\nabla_{u_i} g_i(t_k) + \nabla_{u_i} f(t_k) L_i(t_k)\|^2 \quad (18)$$

where $L_i : [0, T] \mapsto \mathbb{R}^{n \times M_i}$ solves the differential equation

$$\dot{L}_i(t) = -\nabla_x g_i(t) - \nabla_x f(t) L_i(t) \quad (19)$$

for $t \in [0, T]$ with terminal boundary condition $L_i(T) = 0$. We then have the following lemma.

Lemma 2: Consider any player $i \in \mathbb{N}$. Our hard-constrained method (9) is equivalent to the quadratic program

$$\inf_{\theta_i} \theta_i' W_i \theta_i \quad \text{s.t.} \quad \theta_i \in \Theta_i \quad (20)$$

with W_i given by (18) and L_i solving (19) with $L_i(T) = 0$.

Proof: Consider any player $i \in \mathbb{N}$. Multiplying the differential equation defined in (19) by θ_i gives

$$\begin{aligned} \dot{L}_i(t) \theta_i &= -\nabla_x g_i(t) \theta_i - \nabla_x f(t) L_i(t) \theta_i \\ &= -\nabla_x H_i(t, L_i(t) \theta_i, \theta_i) \end{aligned}$$

for $t \in [0, T]$ where the last line follows from the definition of the Hamiltonian (4). Similarly, multiplying the boundary condition $L_i(T) = 0$ by θ_i gives $L_i(T) \theta_i = 0$. Thus, by substituting $\lambda_i(t) = L_i(t) \theta_i$, $\dot{\lambda}_i(t) = \dot{L}_i(t) \theta_i$ and $\lambda_i(T) = L_i(T) \theta_i = 0$, the optimisation problem (9) becomes

$$\begin{aligned} \inf_{\theta_i} \quad & \sum_{k=1}^{K_i} \|\nabla_{u_i} g_i(t_k) \theta_i + \nabla_{u_i} f(t_k) L_i(t_k) \theta_i\|^2 \\ \text{s.t.} \quad & \dot{L}_i(t) \theta_i = -\nabla_x g_i(t) \theta_i - \nabla_x f(t) L_i(t) \theta_i, \quad t \in [0, T] \\ & L_i(T) \theta_i = 0 \\ & \theta_i \in \Theta_i. \end{aligned}$$

This optimisation problem simplifies to (20) since

$$\sum_{k=1}^{K_i} \|\nabla_{u_i} g_i(t_k) \theta_i + \nabla_{u_i} f(t_k) L_i(t_k) \theta_i\|^2 = \theta_i' W_i \theta_i$$

due to the definition of W_i where L_i solves (19) with terminal condition $L_i(T) = 0$. The proof is complete. \square

Analogous to Lemma 1, Lemma 2 greatly simplifies the solution of our hard-constrained method (9). Indeed, for each player $i \in \mathbb{N}$, we may find parameters $\hat{\theta}_i$ solving (9) by solving the differential equation (19) for L_i , computing the matrix W_i , and then solving the quadratic program (20). Lemma 2 also enables us to now develop our main result providing an explicit form of the solutions to (20) and hence our hard-constrained method (9).

D. Solution of Hard-Constrained Method

To describe the solutions to our hard-constrained method, let \bar{W}_i be the principal submatrix of W_i formed by deleting the first row and column of W_i , and let \bar{w}_i denote the first column of W_i with its first element deleted. Furthermore, let $r_i^{\bar{W}}$ be the rank of \bar{W}_i , let \bar{W}_i^+ be the pseudoinverse of \bar{W}_i , and let $\bar{W}_i = V_i \Sigma_i^{\bar{W}} V_i'$ be the SVD of \bar{W}_i where $V_i \in \mathbb{R}^{(M_i-1) \times (M_i-1)}$ and $\Sigma_i^{\bar{W}} \in \mathbb{R}^{(M_i-1) \times (M_i-1)}$. Our main result for our hard-constrained method (9) follows.

Theorem 3: Consider any player $i \in \mathbb{N}$, let $\Theta_i = \{\theta_i \in \mathbb{R}^{M_i} : \theta_{i,1} = 1\}$, and suppose that Assumption 2 holds. If $(I - \bar{W}_i \bar{W}_i^+) \bar{w}_i = 0$, then the $\tilde{\theta}_i$ solving (9) all satisfy

$$\tilde{\theta}_i = \begin{bmatrix} 1 \\ \tilde{\theta}_i \end{bmatrix} \quad (21)$$

where the vectors $\tilde{\theta}_i \in \mathbb{R}^{M_i-1}$ are given by

$$\tilde{\theta}_i = -\bar{W}_i^+ \bar{w}_i + V_i \begin{bmatrix} 0 & b' \end{bmatrix}'$$

for any $b \in \mathbb{R}^{M_i-1-r_i^{\bar{W}}}$. Furthermore, if $r_i^{\bar{W}} = M_i - 1$ then

$$\tilde{\theta}_i = \begin{bmatrix} 1 \\ -\bar{W}_i^+ \bar{w}_i \end{bmatrix} \quad (22)$$

is the unique solution to (9), with $\tilde{\theta}_i = \kappa_i \theta_i^*$ if Assumption 1 holds and if there exists a $\kappa_i > 0$ such that $\kappa_i \theta_i^* \in \Theta_i$.

Proof: The proof is similar to that of Theorem 2 with application of Lemma 2 and use of [33, Proposition 15.2]. \square

Theorem 3 gives that the rank condition $r_i^{\bar{W}} = M_i - 1$ is sufficient for ensuring that our hard-constrained method (9) yields unique parameters $\tilde{\theta}_i$. As with the uniqueness conditions for our soft-constrained method (8) established in Theorem 2, the rank condition of Theorem 3 has an intuitive interpretation as a persistence of excitation condition and will fail to hold when the inverse differential game problem is ill-posed due to uninformative trajectories (e.g., due to short time-horizons T or degenerate system dynamics). The rank condition of Theorem 3 will also fail to hold if there are too few (informative) sampling times t_k in \mathbb{K}_i . Under Assumption 1, the last assertion of Theorem 3 highlights that $r_i^{\bar{W}} = M_i - 1$ becomes a sufficient condition for our hard-constrained method to yield unique parameters $\tilde{\theta}_i$ that differ from the true parameters θ_i^* by an unknown factor κ_i .

E. Comparison of Methods

Theorems 2 and 3 imply that our soft-constrained and hard-constrained methods can be implemented without solving differential game or optimal control problems. They can also both compute the cost-functional parameters of each player separately. In contrast, existing nested optimisation and bilevel optimisation methods of inverse differential games proposed in [9] require the repeated solution of differential game or optimal control problems, and may require the parameters of all players to be computed simultaneously. Indeed, our methods are significantly less computationally complex than existing methods since they only involve the solution of a differential equation (either (11) or (19)) followed by a submatrix pseudoinverse, rank calculation, and a potential SVD (of

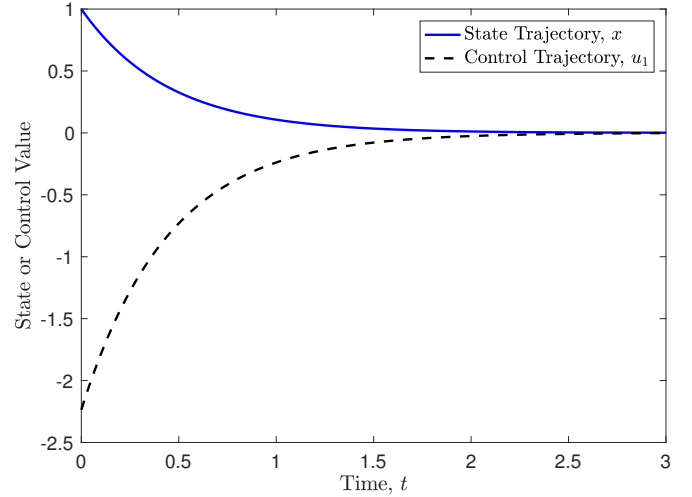


Fig. 1. State and control trajectories that solve the optimal control problem in our illustrative example with $T = 3$, $x_0 = 1$, and $\theta_1 = \theta_1^* = [1 \ 3]'$.

either \bar{P}_i or \bar{W}_i) for each player. Our hard-constrained method is slightly less computationally expensive than our soft-constrained method due to the lower dimensionality of $\bar{W}_i \in \mathbb{R}^{(M_i-1) \times (M_i-1)}$ compared to $\bar{P}_i \in \mathbb{R}^{(M_i+n-1) \times (M_i+n-1)}$.

V. SIMULATION EXAMPLES

In this section, we examine our methods in two examples. We first provide an optimal control example comparing the conditions of Theorems 2 and 3. We then consider a two-player game to examine the performance of our methods with larger state, parameter, and control spaces.

A. Illustrative Example

Consider a linear quadratic optimal control problem, or one-player differential game, with state process $\dot{x}(t) = u_1(t)$ with $x_0 = 1$, $u_1(t) \in \mathbb{R}$, and cost-functional J_1 of the form in (2) with $T = 3$, $g_1(t, x(t), u_1(t)) \triangleq [u_1^2(t) \ x^2(t) \ x(t)u_1(t)]'$ and $\theta_1 = \theta_1^* = [1 \ 5 \ 2]'$. To illustrate our methods, we numerically solved this optimal control problem for the trajectories x and u_1 shown in Fig. 1. We gave both methods knowledge of these trajectories together with the basis functions g_1 and dynamics f so that Assumption 1 holds. We also note that the trajectories are unique solutions to the optimal control problem since θ_1^* satisfies the positive definite and positive semidefinite conditions of [32, Section 3.4].

Considering our soft-constrained method with $\gamma = 1$, and applying Lemma 1 and Theorem 2, the submatrix \bar{P}_1 is

$$\bar{P}_1 = \begin{bmatrix} 0.9951 & -0.6126 & 0.9951 \\ -0.6126 & 0.4614 & -0.6126 \\ 0.9951 & -0.6126 & 0.9951 \end{bmatrix}$$

which is rank deficient. Computing the SVD of \bar{P}_1 yields

$$U_1 = \begin{bmatrix} -0.6445 & -0.2909 & -0.7071 \\ 0.4113 & -0.9115 & -0.0000 \\ -0.6445 & -0.2909 & 0.7071 \end{bmatrix}$$

and so $U_1^{12} \neq 0$ which implies that unique parameters cannot be computed in this example with our soft-constrained method.

Considering our hard-constrained method and applying Lemma 2 and Theorem 3 with $\mathbb{K}_1 = \{0, 3\}$ yields

$$\bar{W}_1 = \begin{bmatrix} 0.0000 & 0.0015 \\ 0.0015 & 0.7988 \end{bmatrix}$$

which is trivially full rank, and so our hard-constrained method is able to uniquely recover the parameters $\hat{\theta}_1 = \theta_1^* = [1 \ 5 \ 2]'$. We were unable to find any \mathbb{K}_1 with $K_i \geq 2$ such that our hard-constrained method failed to uniquely recover θ_1^* . However, our hard-constrained method could not compute unique parameters for any \mathbb{K}_i with $K_i = 1$ (which is intuitive since it is seeking to compute two components of θ_1^*).

This example illustrates that the conditions of Theorems 2 and 3 for our soft-constrained and hard-constrained methods are not equivalent in general. Indeed, from this example we see that the conditions of Theorem 3 are likely to be more easily satisfied than those of Theorem 2 because there are fewer optimisation variables in our hard-constrained method and so less potential for nonuniqueness (e.g., due to different pairs of functions and parameters resulting in the same value of the Hamiltonian gradients in our soft-constrained method).

B. Two-Player Collision Avoidance Example

We now investigate our methods in a simulated two-player collision avoidance differential game closely related to the model proposed in [7] for the behaviours of birds in mid-air collision scenarios. Consider two players moving in three-dimensions and let Player i 's position and velocity be $q_i(t) \triangleq [q_{i,1}(t) \ q_{i,2}(t) \ q_{i,3}(t)]' \in \mathbb{R}^3$ and $\dot{q}_i(t) \triangleq [\dot{q}_{i,1}(t) \ \dot{q}_{i,2}(t) \ \dot{q}_{i,3}(t)]' \in \mathbb{R}^3$, respectively. Let the state of the game be the positions and velocities of the two players, namely, $x(t) = [q_1'(t) \ \dot{q}_1'(t) \ q_2'(t) \ \dot{q}_2'(t)]' \in \mathbb{R}^{12}$. Let the state evolve according to the kinematic equations

$$\dot{x}(t) = \begin{bmatrix} \bar{A} & 0 \\ 0 & \bar{A} \end{bmatrix} x(t) + \begin{bmatrix} \bar{C} \\ 0 \end{bmatrix} u_1(t) + \begin{bmatrix} 0 \\ \bar{C} \end{bmatrix} u_2(t) \quad (23)$$

for $t \in [0, T]$ from the initial state $x(0) = [x_1'(0) \ x_2'(0)]'$ where $u_i(t) = [u_{i,1}(t) \ u_{i,2}(t) \ u_{i,3}(t)]' \in \mathbb{R}^3$ are the player acceleration control inputs, and the matrices

$$\bar{A} \triangleq \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad \bar{C} \triangleq \begin{bmatrix} 0 \\ I \end{bmatrix}$$

describe the mappings between position, velocity, and acceleration. Both players seek to use minimal accelerations to avoid colliding with the other player whilst attempting to track along a particular coordinate axis. The basis functions are thus

$$g_i(t, x(t), u_1(t), u_2(t)) = [\delta(t) \ u_{i,1}^2(t) \ u_{i,2}^2(t) \ u_{i,3}^2(t) \ q_{i,1}^2(t) \ q_{i,2}^2(t) \ q_{i,3}^2(t)]' \in \mathbb{R}^7$$

where the first basis function is the inverse of the squared distance between the players, namely, $\delta(t) \triangleq \|q_1(t) - q_2(t)\|^{-2}$; the second, third, and fourth basis functions are the squared player acceleration components; and, the last three basis

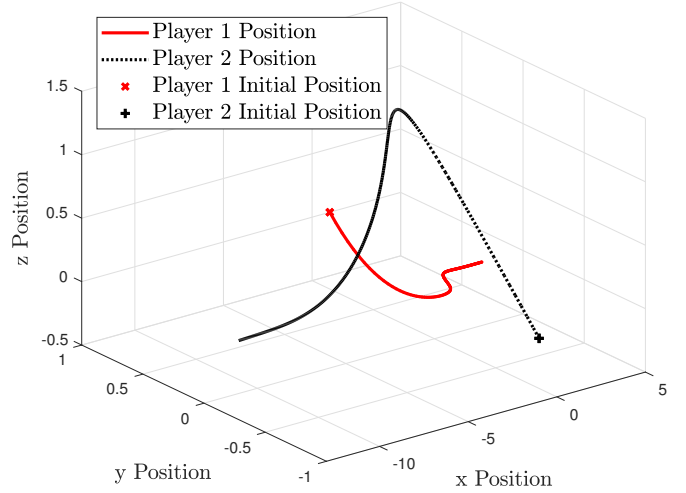


Fig. 2. Two-Player Collision Avoidance Game: Positions of both players given by simulated open-loop Nash equilibrium trajectories x and (u_1, u_2) .

functions are the squared player position components. The players weight these basis functions with the parameter vectors $\theta_1 = \theta_1^* = [1 \ 3 \ 5 \ 0.6 \ 0 \ 3 \ 5]'$ and $\theta_2 = \theta_2^* = [1 \ 2 \ 1 \ 5 \ 0 \ 0.5 \ 0.2]'$.

To compare our methods, we numerically solved this game for open-loop Nash equilibrium trajectories x and (u_1, u_2) with $T = 15$, $q_1(0) = [1 \ 1 \ 0]'$, $q_2(0) = [-1 \ -1 \ 0]'$, $\dot{q}_1(0) = [-1 \ -1 \ -1]'$ and $\dot{q}_2(0) = [1 \ 1 \ 1]'$. The position trajectories are shown in Fig. 2. We gave both methods knowledge of the open-loop Nash equilibrium trajectories x and (u_1, u_2) , dynamics f , and basis functions g_i so that Assumption 1 holds. In this example, we also note that the cost-functionals and Hamiltonians are convex in the controls and so the conditions of Theorem 1 are both necessary and sufficient for the existence of open-loop Nash equilibria.

In the case of our soft-constrained method, by applying Lemma 1 and Theorem 2 with $\gamma = 1$, we found that both $\bar{P}_1 \in \mathbb{R}^{18 \times 18}$ and $\bar{P}_2 \in \mathbb{R}^{18 \times 18}$ are rank deficient with $r_1^{\bar{P}} = 12$ and $r_2^{\bar{P}} = 12$. However by computing the SVDs of \bar{P}_1 and \bar{P}_2 , we found that U_1^{12} and U_2^{12} are zero matrices. From Theorem 2, we thus have that our soft-constrained method yields the unique parameters $\hat{\theta}_1 = \theta_1^*$ and $\hat{\theta}_2 = \theta_2^*$ with a percentage error of less than 0.05% (due to the numeric tolerance of ordinary differential equation solver used, i.e., MATLAB's `ode113`). We note however that since \bar{P}_1 and \bar{P}_2 are rank deficient, the functions $\hat{\lambda}_i$ solving (8) are nonunique.

To consider our hard-constrained method, we considered sampling times \mathbb{K}_1 and \mathbb{K}_2 with $1 \leq K_1, K_2 \leq 6$ times evenly spaced in the interval $[0, 15]$ and applied Lemma 2 and Theorem 3. For a small number of sampling times $K_1 = K_2 < 3$ at the start and end of the interval $[0, 15]$, we found that \bar{W}_1 and \bar{W}_2 were rank deficient and so our hard-constrained method fails to compute unique parameters. For a larger number of sampling times $K_1 = K_2 \geq 3$ evenly spaced in $[0, 15]$, we found that \bar{W}_1 and \bar{W}_2 were full rank and so our hard-constrained method yields the (unique) parameters $\hat{\theta}_1 = \theta_1^*$ and $\hat{\theta}_2 = \theta_2^*$ with a percentage error of less than 0.17% (due to the numeric tolerance of MATLAB's

ode113). In this example, our hard-constrained method is not sensitive to the choice of sampling times since it yields unique parameters with as few as $K_1 = K_2 = 3$ sampling times as close together as the numeric precision of the trajectories.

VI. CONCLUSION

We propose two methods of inverse open-loop noncooperative differential games. Our soft-constrained method computes parameters that simultaneously minimise the violation of conditions on both the Hamiltonian and costate functions. In contrast, our hard-constrained method handles trajectories with constrained controls by computing parameters that minimise the violation of the conditions on the Hamiltonians under a constraint that the conditions on the costate functions hold exactly. We formulate our methods as quadratic programs and establish novel conditions analogous to persistence of excitation under which they yield unique parameters.

Our methods and results are developed without explicit consideration of noisy trajectories or misspecification of the dynamics or basis functions. The development of inverse differential game methods capable of explicitly handling noisy trajectories, misspecified dynamics, and misspecified basis functions therefore remains an important open problem. Similarly, methods capable of withstanding attacks by adversaries that obstruct the computation of parameters are also yet to be investigated (despite recent results for static games in [34]). The dual problems of our quadratic formulations are potential starting points for these investigations.

ACKNOWLEDGEMENT

We gratefully acknowledge Grace Garden for the many fruitful discussions inspiring Section V-B.

REFERENCES

- [1] R. Isaacs, *Differential Games: Mathematical Theory with Application to Warfare and Pursuit and Optimisation*. New York: Dover Publications, 1965.
- [2] T. Basar and G. J. Olsder, *Dynamic noncooperative game theory*, 2nd ed. New York, NY: Academic Press, 1999, vol. 23.
- [3] L. J. Ratliff, S. A. Burden, and S. S. Sastry, "On the Characterization of Local Nash Equilibria in Continuous Games," *IEEE Transactions on Automatic Control*, vol. 61, no. 8, pp. 2301–2307, Aug 2016.
- [4] M. Flad, L. Fröhlich, and S. Hohmann, "Cooperative shared control driver assistance systems based on motion primitives and differential games," *IEEE Transactions on Human-Machine Systems*, vol. 47, no. 5, pp. 711–722, Oct 2017.
- [5] E. Dockner, *Differential games in economics and management science*. Cambridge University Press, 2000.
- [6] J. Engwerda, *LQ dynamic optimization and differential games*. West Sussex, U.K.: John Wiley & Sons, 2005.
- [7] T. L. Molloy, G. S. Garden, T. Perez, I. Schiffner, D. Karmaker, and M. Srinivasan, "An Inverse Differential Game Approach to Modelling Bird Mid-Air Collision Avoidance Behaviours," in *System Identification (SYSID 2018), 18th IFAC Symposium on*, Stockholm, Sweden, July 2018.
- [8] J. W. Weibull, *Evolutionary game theory*. MIT press, 1997.
- [9] T. L. Molloy, J. Ford, and T. Perez, "Inverse Noncooperative Differential Games," in *Decision and Control (CDC), 2017 IEEE 56th Annual Conference on*, Melbourne, Australia, December 2017.
- [10] S. Rothfuß, J. Inga, F. Köpf, M. Flad, and S. Hohmann, "Inverse Optimal Control for Identification in Non-Cooperative Differential Games," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 14 909–14 915, Jul. 2017.
- [11] D. Bertsimas, V. Gupta, and I. Paschalidis, "Data-driven estimation in equilibrium using inverse optimization," *Mathematical Programming*, vol. 153, no. 2, pp. 595–633, 2015.
- [12] I. C. Konstantakopoulos, L. J. Ratliff, M. Jin, C. J. Spanos, and S. S. Sastry, "Inverse modeling of non-cooperative agents via mixture of utilities," in *Decision and Control (CDC), 2016 IEEE 55th Conference on*. IEEE, 2016, pp. 6327–6334.
- [13] I. C. Konstantakopoulos, L. J. Ratliff, M. Jin, S. S. Sastry, and C. J. Spanos, "A Robust Utility Learning Framework via Inverse Optimization," *IEEE Transactions on Control Systems Technology*, vol. PP, no. 99, pp. 1–17, 2017.
- [14] L. J. Ratliff, M. Jin, I. C. Konstantakopoulos, C. Spanos, and S. S. Sastry, "Social game for building energy efficiency: Incentive design," in *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sept 2014, pp. 1011–1018.
- [15] V. Kuleshov and O. Schrijvers, "Inverse game theory: Learning utilities in succinct games," in *Int. Conf. on Web and Internet Economics (WINE) 2015*, December 2015, pp. 413–427.
- [16] G. F. Stocco and G. Cybenko, "Inverse game theory: learning the nature of a game through play," in *Proc. SPIE 8359, Sensors, and Command, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense XI*, E. M. Carapezza, Ed., May 2012, pp. 835 905–835 905–10.
- [17] V. J. Hotz and R. A. Miller, "Conditional choice probabilities and the estimation of dynamic models," *The Review of Economic Studies*, vol. 60, no. 3, pp. 497–529, 1993.
- [18] V. J. Hotz, R. A. Miller, S. Sanders, and J. Smith, "A simulation estimator for dynamic models of discrete choice," *The Review of Economic Studies*, vol. 61, no. 2, pp. 265–289, 1994.
- [19] V. Aguirregabiria and P. Mira, "Sequential estimation of dynamic discrete games," *Econometrica*, vol. 75, no. 1, pp. 1–53, 2007.
- [20] P. Bajari, C. L. Benkard, and J. Levin, "Estimating dynamic models of imperfect competition," *Econometrica*, vol. 75, no. 5, pp. 1331–1370, 2007.
- [21] X. Lin, P. A. Beling, and R. Cogill, "Multiagent Inverse Reinforcement Learning for Two-Person Zero-Sum Games," *IEEE Transactions on Games*, vol. 10, no. 1, pp. 56–68, March 2018.
- [22] A. Keshavarz, Y. Wang, and S. Boyd, "Imputing a convex objective function," in *Intelligent Control (ISIC), 2011 IEEE International Symposium on*. IEEE, 2011, pp. 613–619.
- [23] D. Tsai, T. L. Molloy, and T. Perez, "Inverse two-player zero-sum dynamic games," in *Australian Control Conference (AUCC), 2016*, Newcastle, Australia, Nov 2016.
- [24] T. L. Molloy, J. Ford, and T. Perez, "Inverse noncooperative dynamic games," in *IFAC 2017 World Congress*, Toulouse, France, July 2017.
- [25] F. Köpf, J. Inga, S. Rothfuß, M. Flad, and S. Hohmann, "Inverse Reinforcement Learning for Identification in Linear-Quadratic Dynamic Games," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 14 902–14 908, Jul. 2017.
- [26] T. L. Molloy, J. J. Ford, and T. Perez, "Finite-horizon inverse optimal control for discrete-time nonlinear systems," *Automatica*, vol. 87, pp. 442 – 446, 2018.
- [27] P. Arcidiacono, P. Bayer, J. R. Blevins, and P. B. Ellickson, "Estimation of dynamic discrete choice models in continuous time with an application to retail competition," *The Review of Economic Studies*, vol. 83, no. 3, pp. 889–931, 2016.
- [28] K. Mombaur, A. Truong, and J.-P. Laumond, "From human to humanoid locomotion—an inverse optimal control approach," *Autonomous robots*, vol. 28, no. 3, pp. 369–383, 2010.
- [29] H. El-Hussieny, A. A. Abouelsoud, S. F. M. Assal, and S. M. Megahed, "Adaptive learning of human motor behavior: An evolving inverse optimal control approach," *Engineering Applications of Artificial Intelligence*, vol. 50, pp. 115–124, 2016.
- [30] M. Johnson, N. Aghasadeghi, and T. Bretl, "Inverse optimal control for deterministic continuous-time nonlinear systems," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, Dec 2013, pp. 2906–2913.
- [31] N. Aghasadeghi and T. Bretl, "Inverse optimal control for differentially flat systems with application to locomotion modeling," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, May 2014, pp. 6018–6025.
- [32] B. Anderson and J. Moore, *Optimal control: linear quadratic methods*. Englewood Cliffs, NJ: Prentice Hall, 1990.
- [33] J. Gallier, *Geometric Methods and Applications: For Computer Science and Engineering*, 2nd ed. Springer, 2013.
- [34] R. Jia, I. C. Konstantakopoulos, B. Li, and C. Spanos, "Poisoning attacks on data-driven utility learning in games," in *2018 Annual American Control Conference (ACC)*, June 2018, pp. 5774–5780.