A Second-Order Reachable Sets Computational Scheme.

Olalekan Ogunmolu⋆, Ian Abraham‡, and Sylvia Herbert†.

***Abstract*— Motivated by the applications of variational Hamilton-Jacobi-Isaacs (HJI) formulations in providing a least restrictive controller in reachability problems that involve state or input constraints, we present a second-order approximation scheme to Mitchell et. al's variational HJI value functional under continuity assumptions throughout the state space. By consistently maintaining locally linear trajectories around non-linear trajectories to be synthesized, we show that with state feedback we can compute the set of states that are reachable within a prescribed verification time bound.**

I. Introduction

In this letter, we shall provide a successive approximation scheme for constructing the *discriminating kernel* for a controller's "constraints-satisfaction" design problem; at issue is returning state sets of a dynamical system that satisfy these constraints in a least restrictive sense. Our study is motivated by the scalability issues associated with Eulerian methods [1], [2]. Eulerian methods scale exponentially. Whence, we adopt a practical approach, drawing inspiration from second-order variation methods in differential optimal control. In this sentiment, we shall construct a second-order approximation method, under a smoothness assumption of the value (cost) functional. This will allow us to compute locally optimal reachable sets in an iterative manner; by extension, this returns second-order convergence certificates to the *optimal reachable set*.

A basic characteristic of a discrete, continuous, or an hybrid control system is to determine the point set within its state space that are *reachable* by the choice of a control input over a time interval. *Reachable sets* can be analyzed in a (i) *forward* sense, where system trajectories are examined to determine if they enter certain states from an *initial set*; (ii) *backward* sense, where system trajectories are examined to determine if they enter certain *target sets*; (iii) *reach set* sense, in which they are examined to see if states reach a set at a *particular time*; or (iv) *reach tube* sense, in which they are examined to determine that they reach a set *during a time interval*.

In this work, we focus on the backward construction. Backward reachability consists in avoiding an unsafe state set under the worst-possible disturbance within a given time-bound. The states sets of a backward reachable problem constitute the *discriminating kernel* that is "safety-preserving" for a finite-time horizon control problem. For an infinite-time horizon control problem, they constitute the *largest robust controlled-invariant set*. In the absence of a disturbance, the resulting kernel is said to satisfy *viability constraints*. The set associated with the kernel computed in a backward reachability framework is termed the *backward reachable set* or *BRS*.

Eulerian methods [2], [3], [4], [5] resolve the BRS as the zero-level set of an implicitly-defined value function. Constructed as an initial value problem for a *Cauchy-type* Hamilton-Jacobi-Isaacs (HJI) partial differential equation (P.D.E.) [6], [1], [7], [8], the BRS in one dimension is equivalent to the conservation of momentum equation [5]. For a multidimensional problem, and by resolving the P.D.E. on a dimension-by-dimension basis in a consistent and monotone fashion, a numerically precise and accurate solution to the HJ equations is determined [9], [2]. However, as state dimensions increase, spacetime discretization methods become impractical owing to their exponential complexity.

Here, we resort to classical successive sweep optimal control algorithms [10], [11], [12], [13] in an iterative dynamic game fashion [14]. These allow us to prescribe a second-order variation scheme for computing locally optimal reachable sets. Similar to the monotone characteristics of [15], [16], these methods (under appropriate positivity of the Hamiltonian's jacobian) assure a monotone solution on the state space.

Applications of the algorithm presented in this work can be found in: (a) biological systems (such as morphological computational problems), where an agent must regulate *dynamic equilibrium* by optimally responding to disturbing external inputs; (b) economic systems, as in a price-setting environment where given agents' preferences and budgets, the price *set* that satisfy budgetary constraints and the maximum number of agents must be found; (c) large process control plants, where a control law or policy must carefully regulate setpoints or trajectories against *all* competing constraints; (d) robotics and other control systems value chains, where the dynamic evolution of systems that possess uncertainty in inputs, states, and environmental constraints need to be systematically managed such that the *differential inclusion*, or the overall closed-loop system's states' change rate preserve *viability or discriminating constraints* [17].

The body of this letter is structured as follows: § II describes the notations used in this letter, and § III introduces methods that we will build upon in describing our proposal in § IV. We conclude the paper in § V.

⋆Microsoft Research, 300 Lafayette Street, New York, NY10012, USA lekanmolu@microsoft.com

‡Department of Mechanical and Aerospace Engineering, Yale University, New Haven, Conn, USA ian.abraham@yale.edu

†Department of Mechanical and Aerospace Engineering, UC San Diego, La Jolla, CA, USA. sherbert@eng.ucsd.edu

II. Notations and Terminology.

We employ standard vector-matrix notations. Conventions: small Latin letters are scalars; when they appear in bold-font they are vectors. Vectors are *column-stacked*. Upper-case Greek and Calligraphic letters are sets; the scalar product of two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$, written as $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$, shall denote their inner product. Arbitrary real variables e.g., t, t_0, t_f, τ, T shall denote *time*; fixed, ordered values of t shall be denoted as $t_0 \le t \le t_f$. The transpose of a matrix, $\boldsymbol{X}$, is signified by the superscript operator, T, on that matrix variable e.g., $\boldsymbol{X}^T$. The contents of a matrix or vector are real throughout. For a scalar function $\boldsymbol{V}$, $\boldsymbol{V_x}$ is the gradient vector, and $\boldsymbol{V_{xx}}$ is the jacobian matrix.

The state $\boldsymbol{x}$ belongs in the open set $\Omega \subset \mathbb{R}^n$. The pair $(\boldsymbol{x}, t)$ will be termed the *phase* of the system. The Cartesian product of $\Omega \subset \mathbb{R}^N$ and the space $T = \mathbb{R}^1$ of all time values is termed the *phase space*, $\Omega \times T$. The closure of Ω is denoted $\bar{\Omega}$ and we let $\delta\Omega$ be the boundary of Ω.

Unless otherwise stated, vectors $\boldsymbol{u}(t)$ and $\boldsymbol{v}(t)$ are reserved for admissible control (resp. disturbance) at time t. We say $\boldsymbol{u}(t)$ (resp. $\boldsymbol{v}(t)$) is piecewise continuous in t, if for each t, $\boldsymbol{u} \in \mathscr{U}$ (resp. $\boldsymbol{v} \in \mathscr{V}$), $\mathscr{U}$(resp. $\mathscr{V}$) is a Lebesgue measurable and compact set. At all times, any of $\boldsymbol{u}(t)$ or $\boldsymbol{v}(t)$ will be under the influence of a *player* such that the motion of $\boldsymbol{x}(t)$ will be influenced by the will of that player.

We are concerned with the following dynamical system

$$\dot{\boldsymbol{x}}(t) = f(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{v}(t)), \ \boldsymbol{x}(0) = \boldsymbol{x}_0, \ -T \le t \le 0 \tag{1}$$

where $\boldsymbol{x}(t)$ evolves from some initial negative time $-T$ to a final time 0; $\boldsymbol{u}(t) \in \mathbb{R}^p$ and $\boldsymbol{v}(t) \in \mathbb{R}^q$ are respectively the control and disturbance signals for some $p, q > 0$. Assumptions: $f(t, \cdot, \cdot, \cdot)$ and $\boldsymbol{x}(t)$ are bounded and Lipschitz continuous for fixed $\boldsymbol{u}$ and $\boldsymbol{v}$[1]. For the phase space $(\Omega \times T)$, the set of all controls for players $\boldsymbol{P}$ and $\boldsymbol{E}$ are respectively drawn from

$$\bar{\mathscr{U}} \equiv \{\boldsymbol{u} : [-T, 0] \to \mathscr{U} | \boldsymbol{u} \text{ measurable}, \mathscr{U} \in \mathbb{R}^m\}, \tag{2}$$

$$\bar{\mathscr{V}} \equiv \{\boldsymbol{v} : [-T, 0] \to \mathscr{V} | \boldsymbol{v} \text{ measurable}, \mathscr{V} \subset \mathbb{R}^p\}, \tag{3}$$

with $\mathscr{U} \in \mathbb{R}^{n_u}$, $\mathscr{V} \in \mathbb{R}^{n_v}$ being compact.

At issue are two players interacting in an environment over a finite horizon, $[-T, 0]$. The states evolve according to the continuous-time dynamics of Equation (1). For any admissible control-disturbance pair $(\boldsymbol{u}(t), \boldsymbol{v}(t))$ and initial phase $(\boldsymbol{x}, -T)$, there exists a unique trajectory $\boldsymbol{\xi}(t)$ such that the motion of (1) passing through phase $(\boldsymbol{x}, -T)$ under the action of control $\boldsymbol{u}(t)$, and disturbance $\boldsymbol{v}(t)$, and observed at a time t afterwards, given by,

$$\boldsymbol{\xi}(t) = \boldsymbol{\xi}(t; -T, \boldsymbol{x}, \boldsymbol{u}(\cdot), \boldsymbol{v}(\cdot)) \tag{4}$$

satisfies (1) almost everywhere (a.e.).

By [19], [20], equation (4) has the property that

$$\boldsymbol{\xi}(-T) = \boldsymbol{\xi}(-T; -T, \boldsymbol{x}, \boldsymbol{u}(t), \boldsymbol{v}(t)). \tag{5}$$

Our operational theater is that of conflicting objectives between various agents e.g. with a flight level setpoint tracking goal under an external disturbance' influence such as wind; whence, we have a pursuit-evasion game. Each player in a game shall constitute either a pursuer ($\boldsymbol{P}$) or an evader ($\boldsymbol{E}$)[2]. The set of all *strategies* executed by $\boldsymbol{P}$ (resp. $\boldsymbol{E}$) during a game (beginning at a time t) is denoted as $\mathscr{B}(t)$ (resp. $\mathscr{A}(t)$).

III. Background

Reachable sets in the context of dynamic programming and two person games is here introduced. We restrict attention to verifying a dynamical system behavior (or trajectories) within the backward reachable sets construction of Mitchell et al [1]. We then establish the viscosity solution P.D.E. to the terminal HJI P.D.E, describe the formulation of the BRS and backward reachable tube (BRT). Let now us enquire.

A. Reachability for Systems Verification

Reachability analysis is concerned with finding the set of forward (resp. backward) states that are reachable from a set of initial (resp. target) state sets, *up to a final time*. As opposed to reachability analysis for safety verification in software model checking[3] where single program runs are computed per time from an initial state, here every possible trajectory or execution of a system are computed from all possible initial (resp. terminal states) states. This verification problem usually consists in finding a *reachable states set* that lie along the trajectory of the solution to a first order nonlinear P.D.E. Typically, this solution originates from some initial state $\boldsymbol{x}_0$ at say t_0 up to a state, $\boldsymbol{x}(t_f)$, at a specified final time t_f.

In Eulerian methods, the P.D.E. must be applied at every grid point in the state space and at every instant of time within the time horizon. As the system scales in dimension, this computation scales exponentially. Once computed, the optimal value function provides a safety certificate and a corresponding safety controller defined by the spatial gradients of the value function.

B. The Backward Reachable (Target) Set

For any optimal control problem, a value function is constructed based on a user-defined optimal cost bounded and uniformly continuous for any input phase $(\boldsymbol{x}, -T)$ e.g. reach goal at the end of a time horizon

$$|g(\boldsymbol{x})| \le k, \ |g(\boldsymbol{x}) - g(\hat{\boldsymbol{x}})| \le k|\boldsymbol{x} - \hat{\boldsymbol{x}}| \tag{6}$$

for constant k and all $-T \le t \le 0$, $\hat{\boldsymbol{x}}, \boldsymbol{x} \in \mathbb{R}^n$. The set

$$\mathscr{L}_0 = \{\boldsymbol{x} \in \bar{\Omega} \,|\, g(\boldsymbol{x}) \le 0\}, \tag{7}$$

is the *target set* in the phase space $\Omega \times \mathbb{R}$ (proof in [1]). This target set can represent the failure set (to avoid) or a goal set (to reach) in the state space.

[1] This bounded Lipschitz continuity property assures uniqueness of the system response $\boldsymbol{x}(t)$ to controls $\boldsymbol{u}(t)$ and $\boldsymbol{v}(t)$ [18].

[2] Let the cursory reader not interpret $\boldsymbol{P}$ or $\boldsymbol{E}$ as controlling a single agent. In complex settings, we may have several pursuers (enemies) or evaders (peaceful citizens). However, when $\boldsymbol{P}$ or $\boldsymbol{E}$ determines the behavior of but one agent, these shall signify the agent itself.

[3] Here, a program P with a specific property, Π, is said to be safe if every computation of P is in Π and returns "unsafe" in the contrary case [21].

C. The Backward Reachable Tube

Backward reachability analysis seeks to capture all conditions under which trajectories of the system may enter a user-defined target set cf. (7). This could be desirable (in the case where the target set is a goal) or undesirable (where the target set represents the failure set).

Optimal trajectories emanating from initial phases $(\boldsymbol{x}, -T)$ where the value function is non-negative will maintain non-negative cost over the entire time horizon; therefore they will not enter the target set. Optimal trajectories from initial phases where the value function is negative will enter the target set at some point within the time horizon. For the safety problem setup in (11), we can define the corresponding *robustly controlled backward reachable tube* for $\tau \in [-T, 0]$ as the closure of the open set

$$\mathscr{L}([\tau,0],\mathscr{L}_0) = \{\boldsymbol{x} \in \Omega \,|\, \exists \beta \in \bar{\mathscr{V}}(t) \forall \boldsymbol{u} \in \mathscr{U}(t), \exists \bar{t} \in [-T,0], \boldsymbol{\xi}(\bar{t}) \in \mathscr{L}_0\}, \bar{t} \in [-T,0]. \tag{8}$$

Read: The set of states from which the strategies of $\boldsymbol{P}$, and for all controls of $\boldsymbol{E}$ imply that we reach the target set within the interval $[-T, 0]$. More specifically, following Lemma 2 of [1], the states in the reachable set admit the following properties w.r.t the value function $\boldsymbol{V}$

$$\boldsymbol{x}(t) \in \mathscr{L}(\cdot) \implies \boldsymbol{V}(\boldsymbol{x},t) \le 0 \tag{9a}$$

$$\boldsymbol{V}(\boldsymbol{x},t) \le 0 \implies \boldsymbol{x}(t) \in \mathscr{L}(\cdot). \tag{9b}$$

The goal of $\boldsymbol{P}$ is to drive the system's trajectories into the unsafe set i.e., $\boldsymbol{P}$ has $\boldsymbol{u}$ at will and aims to minimize the termination time of the game (c.f. (7)); and $\boldsymbol{E}$ seeks to avoid the unsafe set i.e., $\boldsymbol{E}$ has controls $\boldsymbol{v}$ at will and seeks to maximize the termination time of the game (c.f. (7)). For goal-satisfaction (or *liveness*) problem setups, the strategies are flipped and the backward reachable tube instead marks the states from which the evader $\boldsymbol{E}$ can successfully reach the target set despite worst-case efforts of the pursuer $\boldsymbol{P}$.

D. The Terminal HJI Value Function for Reachability.

Rather than computing the minimum cost for every possible trajectory of the system, in safety analysis it is sufficient to consider the minimum cost under optimal behavior from both players. The optimal behavior of each player depends on whether the target set represents a goal or a failure set. For a safety (avoiding a failure set) problem setup, the evader $\boldsymbol{E}$ is seeking to maximize the minimum cost (keeping the system out of the target set) and the pursuer $\boldsymbol{P}$ seeks to minimize it.

In the style of [20], suppose that the pursuer's strategy, starting at a time $-T$, is $\beta : \bar{\mathscr{U}}(-T) \to \bar{\mathscr{V}}(-T)$. Suppose further that β is provided for each $-T \le \tau \le 0$ and $\boldsymbol{u}, \hat{\boldsymbol{u}} \in \bar{\mathscr{U}}(-T)$. Then,

$$\begin{cases} \boldsymbol{u}(\bar{t}) = \hat{\boldsymbol{u}}(\bar{t}) \;\text{ for a.e. } -T \le \bar{t} \le \tau \\ \text{implies } \beta[\boldsymbol{u}](\bar{t}) = \beta[\hat{\boldsymbol{u}}](\bar{t}) \;\text{ for a.e. } -T \le \bar{t} \le \tau. \end{cases} \tag{10}$$

For a target set construction problem, we construct the differential game's (lower) value for a solution $\boldsymbol{x}(t)$ that solves (1) for $\boldsymbol{u}(t)$ and $\boldsymbol{v}(t) = \beta[\boldsymbol{u}](t)$ is

$$\boldsymbol{V}(\boldsymbol{x},t) = \inf_{\beta \in \mathscr{B}(t)} \sup_{\boldsymbol{u} \in \mathscr{U}(t)} \min_{t \in [-T,0]} g\left(0; t, \boldsymbol{x}, \boldsymbol{u}(\cdot), \boldsymbol{v}(\cdot)\right). \tag{11}$$

Computing the value function is in general challenging and non-convex. Additionally, the value function is hardly smooth throughout the state space, so it lacks classical solutions even for smooth Hamiltonian and boundary conditions. However, the value function (11) is a "viscosity" (generalized) solution [22], [19] of the associated HJ-Isaacs (HJI) PDE, i.e. solutions which are *locally Lipschitz* in $\Omega \times [-T, 0]$, and with at most first-order partial derivatives in the Hamiltonian. Equation (11) admits the viscosity solution

$$\frac{\partial \boldsymbol{V}}{\partial t}(\boldsymbol{x},t) + \min\{0, \boldsymbol{H}(t;\boldsymbol{x},p)\} = 0 \tag{12a}$$

$$V(\boldsymbol{x},0) = g(\boldsymbol{x}) \tag{12b}$$

where the vector field $p \equiv \boldsymbol{V}_{\boldsymbol{x}}$ is known in terms of the game's terminal conditions so that the overall game is akin to a two-point boundary-value problem; and the Hamiltonian $\boldsymbol{H}(t;\boldsymbol{x},\boldsymbol{u},\boldsymbol{v},\boldsymbol{V}_{\boldsymbol{x}})$ is defined as

$$\boldsymbol{H}(t;\boldsymbol{x},\boldsymbol{u},\boldsymbol{v},\boldsymbol{V}_{\boldsymbol{x}}) = \max_{u \in \mathscr{U}} \min_{v \in \mathscr{V}} \langle f(t;\boldsymbol{x},\boldsymbol{u},\boldsymbol{v}), \boldsymbol{V}_{\boldsymbol{x}} \rangle. \tag{13}$$

For more details on the construction of this P.D.E., see [1].

IV. Successive Approximation Scheme

We now introduce the quadratic approximation scheme of the value function. We seek a pair of *saddle point equilibrium* policies, $(\boldsymbol{u}^*, \boldsymbol{v}^*)$ that satisfy the following inequalities for a cost $\boldsymbol{V}$ at an initial time $-T$:

$$\boldsymbol{V}(-T;\boldsymbol{x},\boldsymbol{u}^*,\boldsymbol{v}) \le \boldsymbol{V}(-T;\boldsymbol{x},\boldsymbol{u}^*,\boldsymbol{v}^*) \le \boldsymbol{V}(-T;\boldsymbol{x},\boldsymbol{u},\boldsymbol{v}^*), \quad \forall \boldsymbol{u} \in \mathscr{U}, \boldsymbol{v} \in \mathscr{V} \text{ and } \boldsymbol{x}(-T). \tag{14}$$

A successive approximation to $\boldsymbol{V}(-T;\cdot,\cdot,\cdot)$ consists in maintaining local approximations to the global system dynamics at every iteration. The max-min over the entire control sequences reduces to a stepwise optimization over single controls, starting from a negative time $-T$ and going forward in time with the *terminal cost* of (12). First, we apply local controls $\boldsymbol{u}_r(t)$ and $\boldsymbol{v}_r(t)$ on (1) so that the nominal value is $\boldsymbol{V}(t;\cdot,\cdot,\cdot)$ for a resulting nominal state $\boldsymbol{x}_r(\tau)$; $\tau \in [-T, 0]$.

A. Local Approximations to Nonlinear Trajectories

The local system dynamics becomes

$$\dot{\boldsymbol{x}}_r(\tau) = f(t;\boldsymbol{x}_r(\tau),\boldsymbol{u}_r(\tau),\boldsymbol{v}_r(\tau)); \quad \boldsymbol{x}_r(0) = \boldsymbol{x}_{r_0}. \tag{15}$$

Our dynamics now describe variations from the nonlinear system cf. (1) with state and control pairs $\delta\boldsymbol{x}(t)$, $\delta\boldsymbol{u}(t)$, $\delta\boldsymbol{v}(t)$ respectively[4]. Therefore, we write

$$\boldsymbol{x}(t) = \boldsymbol{x}_r(t) + \delta\boldsymbol{x}(t),\; \boldsymbol{u}(t) = \boldsymbol{u}_r(t) + \delta\boldsymbol{u}(t), \tag{16a}$$

$$\boldsymbol{v}(t) = \boldsymbol{v}_r(t) + \delta\boldsymbol{v}(t),\; t \in [-T,0]. \tag{16b}$$

[4]Note that $\delta\boldsymbol{x}(t)$, $\delta\boldsymbol{u}(t)$, and $\delta\boldsymbol{v}(t)$ are respectively measured with respect to $\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{v}(t)$ and are not necessarily small.

Abusing notation, we drop the templated time arguments in (16) so that the canonical problem is now

$$\frac{d}{dt}(\boldsymbol{x}_r+\delta\boldsymbol{x}) = f(t;\boldsymbol{x}_r+\delta\boldsymbol{x},\boldsymbol{u}_r+\delta\boldsymbol{u},\boldsymbol{v}_r+\delta\boldsymbol{v}), \tag{17}$$

$$\boldsymbol{x}_r(0)+\delta\boldsymbol{x}(0) = \boldsymbol{x}(0), \tag{18}$$

with the associated terminal value

$$\begin{aligned}-\frac{\partial \boldsymbol{V}}{\partial t}(\boldsymbol{x}_r+\delta\boldsymbol{x},t) &= \min\left\{\boldsymbol{0}, \max_{\delta\boldsymbol{u}\in\mathscr{U}}\min_{\delta\boldsymbol{v}\in\mathscr{V}}\left\langle f(t;\boldsymbol{x}_r+\delta\boldsymbol{x},\right.\right.\\ &\left.\left.\boldsymbol{u}_r+\delta\boldsymbol{u},\boldsymbol{v}_r+\delta\boldsymbol{v}),\frac{\partial \boldsymbol{V}}{\partial \boldsymbol{x}}(\boldsymbol{x}_r+\delta\boldsymbol{x},t)\right\rangle\right\},\\ \boldsymbol{V}(\boldsymbol{x}_r,0) &= g(0;\boldsymbol{x}_r(0)+\delta\boldsymbol{x}(0));\end{aligned} \tag{19}$$

and state trajectory

$$\boldsymbol{\xi}(t) = \boldsymbol{\xi}(t;t_0,\boldsymbol{x}_r+\delta\boldsymbol{x},\boldsymbol{u}+\delta\boldsymbol{u},\boldsymbol{v}+\delta\boldsymbol{v}). \tag{20}$$

For $-T\le t\le 0$ and a $\tau\in[t,0]$, let the optimal cost for using the optimal control $\boldsymbol{u}^\star(\tau)=\boldsymbol{u}_r(\tau)+\delta\boldsymbol{u}^\star(\tau)$ be $\boldsymbol{V}^\star(\boldsymbol{x}_r,\tau)$; and the optimal cost for using $\boldsymbol{u}_r(\tau)$ be $\boldsymbol{V}_r(\boldsymbol{x}_r,t)$. Suppose further that the difference between these two costs on the phase $(\boldsymbol{x}_r,t)$ is $\tilde{\mathbf{V}}^\star$, then

$$\tilde{\mathbf{V}}^\star = \boldsymbol{V}^\star(\boldsymbol{x}_r,t)-\boldsymbol{V}_r(\boldsymbol{x}_r,t). \tag{21}$$

Rational for our successive approximation scheme: Given the optimality principle [23, Ch. III], we can focus on the optimal return function $\boldsymbol{V}^\star(\boldsymbol{x},t)$. The end conditions at the trajectories' terminal surface on $\boldsymbol{V}^\star$, in a backward reachability setting (using cf. (12b)), can be retrogressively computed $\boldsymbol{V}^\star(\boldsymbol{x},t)$ for all feasible phase, $(\boldsymbol{x},t)$, so that the value function extremals' field $\boldsymbol{V}^\star(\boldsymbol{x},t)$ is produced for the complete phase space (Ω,T). It is well-established that second-order successive schemes generate a "quadratic" terminal convergence [24].

B. Main Results

We now state the main results of this paper.

Theorem 1: Suppose that $\boldsymbol{V}$ is smooth enough, the HJI variational inequality cf. (12b) admits the following approximated expansion in the state variation $\delta\boldsymbol{x}$ about the nominal trajectory $\boldsymbol{x}_r$:

$$\begin{aligned}&-\frac{\partial \boldsymbol{V}_r}{\partial t}-\frac{\partial \tilde{\mathbf{V}}}{\partial t}-\left\langle\frac{\partial \boldsymbol{V}_{\boldsymbol{x}}}{\partial t},\delta\boldsymbol{x}\right\rangle-\frac{1}{2}\left\langle\delta\boldsymbol{x},\frac{\partial \boldsymbol{V}_{\boldsymbol{xx}}}{\partial t}\delta\boldsymbol{x}\right\rangle=\\ &\min\left\{\boldsymbol{0},\max_{\delta\boldsymbol{u}\in\mathscr{U}}\min_{\delta\boldsymbol{v}\in\mathscr{V}}\left\langle f^T(t;\boldsymbol{x}_r+\delta\boldsymbol{x},\boldsymbol{u}_r+\delta\boldsymbol{u},\boldsymbol{v}_r+\delta\boldsymbol{v}),\right.\right.\\ &\left.\left.\boldsymbol{V}_{\boldsymbol{x}}+\boldsymbol{V}_{\boldsymbol{xx}}\delta\boldsymbol{x}\right\rangle\right\}.\end{aligned} \tag{22}$$

Furthermore, this expansion is bounded by $O(\delta\boldsymbol{x}^3)$.

Proof: For the moment, let us concentrate on the l.h.s. of (19). Our derivations closely follow that of a single-player controller by Jacobson [12]. For every possible nonlinear trajectory $\boldsymbol{x}$, we choose a nominal local state $\boldsymbol{x}_r$ that is sufficiently close to $\boldsymbol{x}$. Suppose the optimal terminal cost, $\boldsymbol{V}^\star$, is sufficiently smooth to allow a power series expansion in the state variation $\delta\boldsymbol{x}$ about nominal state, $\boldsymbol{x}_r$, we find that

$$\begin{aligned}\boldsymbol{V}^\star(\boldsymbol{x}_r+\delta\boldsymbol{x},t) = \boldsymbol{V}^\star(\boldsymbol{x}_r,t)+\langle\boldsymbol{V}_{\boldsymbol{x}},\delta\boldsymbol{x}\rangle+\frac{1}{2}\langle\delta\boldsymbol{x},\boldsymbol{V}^\star_{\boldsymbol{xx}}\delta\boldsymbol{x}\rangle\\ +\text{h.o.t.}\end{aligned} \tag{23}$$

Here, h.o.t. signifies higher order terms. This expansion scheme is consistent with Volterra-series model order reduction methods [25] or differential dynamic programming schemes that decompose nonlinear systems as a summation of Taylor series expansions [13]. Using (21), (23) becomes

$$\begin{aligned}\boldsymbol{V}^\star(\boldsymbol{x}_r+\delta\boldsymbol{x},t) = \boldsymbol{V}_r(\boldsymbol{x}_r,t)+\tilde{\mathbf{V}}^\star+\langle\boldsymbol{V}_{\boldsymbol{x}},\delta\boldsymbol{x}\rangle+\\ \frac{1}{2}\langle\delta\boldsymbol{x},\boldsymbol{V}^\star_{\boldsymbol{xx}}\delta\boldsymbol{x}\rangle+\text{h.o.t.}\end{aligned} \tag{24}$$

The expansion in (24) may be more costly than solving for the original value function owing to the large dimensionality of the states as higher order terms are expanded. However, consider:

- If $\boldsymbol{x}_r$ is sufficiently close to $\boldsymbol{x}$, $\boldsymbol{V}_r(\boldsymbol{x}_r,t)$, will be sufficiently close to those that originate in (1);
- If the above is true, the state variation $\delta\boldsymbol{x}$ will be sufficiently small owing to the fact that $\boldsymbol{x}\approx\boldsymbol{x}_r$ cf. (16).

Therefore, we can avoid the infinite data storage requirement by truncating the expansion in (24) at, say, the quadratic (second-order) terms in $\delta\boldsymbol{x}$. Seeing that $\delta\boldsymbol{x}$ is sufficiently small, the second-order cost terms will dominate higher order terms, and this new cost will result in an $O(\delta\boldsymbol{x}^3)$ approximation error, affording us realizable control laws that can be executed on the system (1). From (23), we have

$$\boldsymbol{V}^\star(\boldsymbol{x}_r+\delta\boldsymbol{x},t) = \boldsymbol{V}_r+\tilde{\mathbf{V}}^\star+\langle\boldsymbol{V}_{\boldsymbol{x}},\delta\boldsymbol{x}\rangle+\frac{1}{2}\langle\delta\boldsymbol{x},\boldsymbol{V}^\star_{\boldsymbol{xx}}\delta\boldsymbol{x}\rangle. \tag{25}$$

Denoting by $\boldsymbol{V}^\star_{\boldsymbol{x}}$ the co-state on the r.h.s of (19), we can similarly expand it up to second order terms as follows

$$\boldsymbol{V}^\star_{\boldsymbol{x}}(\boldsymbol{x}_r+\delta\boldsymbol{x},t) = \frac{\partial \boldsymbol{V}^\star_r}{\partial \boldsymbol{x}}(\boldsymbol{x}_r,t)+\langle\boldsymbol{V}^\star_{\boldsymbol{xx}}(\boldsymbol{x}_r,t),\delta\boldsymbol{x}\rangle. \tag{26}$$

Note that *the co-state in* (26) *and parameters on the r.h.s. of* (25) *are evaluated on the nominal model, specifically at the phase* $(\boldsymbol{x}_r,t)$. Substituting (25) and (26) into (19), abusing notation by dropping the superscripts and the templated phase arguments, we find that

$$\begin{aligned}&-\frac{\partial \boldsymbol{V}_r}{\partial t}-\frac{\partial \tilde{\mathbf{V}}}{\partial t}-\left\langle\frac{\partial \boldsymbol{V}_{\boldsymbol{x}}}{\partial t},\delta\boldsymbol{x}\right\rangle-\frac{1}{2}\left\langle\delta\boldsymbol{x},\frac{\partial \boldsymbol{V}_{\boldsymbol{xx}}}{\partial t}\delta\boldsymbol{x}\right\rangle=\\ &\min\left\{0,\max_{\delta\boldsymbol{u}}\min_{\delta\boldsymbol{v}}\left\langle f^T(t;\boldsymbol{x}_r+\delta\boldsymbol{x},\boldsymbol{u}_r+\delta\boldsymbol{u},\boldsymbol{v}_r+\delta\boldsymbol{v}),\right.\right.\\ &\left.\left.\boldsymbol{V}_{\boldsymbol{x}}+\boldsymbol{V}_{\boldsymbol{xx}}\delta\boldsymbol{x}\right\rangle\right\}.\end{aligned} \tag{27}$$

Observe that $\boldsymbol{V}_r+\tilde{\mathbf{V}}$, $\boldsymbol{V}_{\boldsymbol{x}}$, and $\boldsymbol{V}_{\boldsymbol{xx}}$ are all functions of the phase $(\boldsymbol{x},r)$ so that

$$\frac{d}{dt}(\boldsymbol{V}_r+\tilde{\mathbf{V}}) = \frac{\partial}{\partial t}(\boldsymbol{V}_r+\tilde{\mathbf{V}})+\left\langle f^T(t;\boldsymbol{x}_r,\boldsymbol{u}_r,\boldsymbol{v}_r),\boldsymbol{V}_{\boldsymbol{x}}\right\rangle \tag{28a}$$

$$\dot{\boldsymbol{V}}_{\boldsymbol{x}} = \frac{\partial \boldsymbol{V}_{\boldsymbol{xx}}}{\partial t}+\langle f^T(t;\boldsymbol{x}_r,\boldsymbol{u}_r,\boldsymbol{v}_r),\boldsymbol{V}_{\boldsymbol{xx}}\rangle \tag{28b}$$

$$\dot{\boldsymbol{V}}_{\boldsymbol{xx}} = \frac{\partial \boldsymbol{V}_{\boldsymbol{xx}}}{\partial t}. \tag{28c}$$

Algorithm 1 Variational P.D.E. Successive Approximation.

1: **procedure**
2: Approximate the nonlinear system dynamics c.f. (1), starting with the pursuer and evader's local control schedules, $\{\boldsymbol{u}_r(t)\}_{t=T}^{0}$ and $\{\boldsymbol{v}_r(t)\}_{t=T}^{0}$, assumed to be available;
3: Run the system's passive dynamics with $\{\boldsymbol{u}_r(t)\}_{t=T}^{0}, \{\boldsymbol{v}_r(t)\}_{t=0}^{T}$ to generate nominal state trajectories $\{\boldsymbol{x}_r(t)\}_{t=T}^{0}$, with neighboring trajectories $\{\boldsymbol{x}(t)\}_{t=0}^{T}$;
4: Choose a small neighborhood, $\{\delta\boldsymbol{x}(t)\}_{t=T}^{0}$ of $\{\boldsymbol{x}(t)\}_{t=T}^{0}$, which provides an optimal reduction in cost as the dynamics no longer represent those of $\{\boldsymbol{x}(t)\}_{t=T}^{0}$;
5: New state and control sequence pairs become $\delta\boldsymbol{x}(t) = \boldsymbol{x}(t) - \boldsymbol{x}_r(t)$, $\delta\boldsymbol{u}(t) = \boldsymbol{u}(t) - \boldsymbol{u}_r(t)$, $\delta\boldsymbol{v}(t) = \boldsymbol{v}(t) - \boldsymbol{v}_r(t)$.
6: **end procedure**

■

Remark 1: If the solution to (22) converges to a local optimum, then the backward reachable set(tube) will converge to a local extrema. In addition, if we overapproximate the resulting numerical solution, the reachable set or tube will locally converge to an optimal region in the state space [26]. The approximation scheme proceeds as follows: The left hand side of (27) admits a quadratic form, so that we can regress a quadratic form to fit the functionals and derivatives of the optimal structure of the nominal system. The r.h.s. can be similarly expanded as above. Define

$$\boldsymbol{H}(t;\boldsymbol{x},\boldsymbol{u},\boldsymbol{v},\boldsymbol{V}_{\boldsymbol{x}}) = \langle \boldsymbol{V}_{\boldsymbol{x}}, f(t;\boldsymbol{x},\boldsymbol{u},\boldsymbol{v})\rangle \tag{29}$$

so that (27) becomes

$$\begin{aligned} -\frac{\partial \boldsymbol{V}_r}{\partial t} - \frac{\partial \tilde{\boldsymbol{V}}}{\partial t} - \left\langle \frac{\partial \boldsymbol{V}_{\boldsymbol{x}}}{\partial t}, \delta\boldsymbol{x}\right\rangle - \frac{1}{2}\left\langle \delta\boldsymbol{x}, \frac{\partial \boldsymbol{V}_{\boldsymbol{xx}}}{\partial t}\delta\boldsymbol{x}\right\rangle = \\ \min\Big\{\boldsymbol{0}, \max_{\delta\boldsymbol{u}}\min_{\delta\boldsymbol{v}}[\boldsymbol{H}(t;\boldsymbol{x}_r+\delta\boldsymbol{x},\boldsymbol{u}_r+\delta\boldsymbol{u},\boldsymbol{v}+\delta\boldsymbol{v},\boldsymbol{V}_{\boldsymbol{x}}) + \\ \langle \boldsymbol{V}_{\boldsymbol{xx}}\delta\boldsymbol{x}, f(t;\boldsymbol{x}_r+\delta\boldsymbol{x},\boldsymbol{u}_r+\delta\boldsymbol{u},\boldsymbol{v}_r+\delta\boldsymbol{v})\rangle]\Big\}. \end{aligned} \tag{30}$$

Expanding the r.h.s. about $\boldsymbol{x}_r, \boldsymbol{u}_r, \boldsymbol{v}_r$ up to second-order only[5], we find that

$$\begin{aligned} \min\Big\{\boldsymbol{0}, \max_{\delta\boldsymbol{u}}\min_{\delta\boldsymbol{v}}[\boldsymbol{H} + \langle \boldsymbol{H}_{\boldsymbol{x}} + \boldsymbol{V}_{\boldsymbol{xx}}f, \delta\boldsymbol{x}\rangle + \\ \langle \boldsymbol{H}_{\boldsymbol{u}}, \delta\boldsymbol{u}\rangle + \langle \boldsymbol{H}_{\boldsymbol{v}}, \delta\boldsymbol{v}\rangle + \langle \delta\boldsymbol{u}, (\boldsymbol{H}_{\boldsymbol{ux}} + f_{\boldsymbol{u}}^T\boldsymbol{V}_{\boldsymbol{xx}})\delta\boldsymbol{x}\rangle \\ + \langle \delta\boldsymbol{v}, (\boldsymbol{H}_{\boldsymbol{vx}} + f_{\boldsymbol{v}}^T\boldsymbol{V}_{\boldsymbol{xx}})\delta\boldsymbol{x}\rangle + \frac{1}{2}\langle \delta\boldsymbol{u}, \boldsymbol{H}_{\boldsymbol{uv}}\delta\boldsymbol{v}\rangle \cdot \\ + \frac{1}{2}\langle \delta\boldsymbol{v}, \boldsymbol{H}_{\boldsymbol{vu}}\delta\boldsymbol{u}\rangle + \frac{1}{2}\langle \delta\boldsymbol{u}, \boldsymbol{H}_{\boldsymbol{uu}}\delta\boldsymbol{u}\rangle + \frac{1}{2}\langle \delta\boldsymbol{v}, \boldsymbol{H}_{\boldsymbol{vv}}\delta\boldsymbol{v}\rangle \\ + \frac{1}{2}\langle \delta\boldsymbol{x}, (\boldsymbol{H}_{\boldsymbol{xx}} + f_{\boldsymbol{x}}^T\boldsymbol{V}_{\boldsymbol{xx}} + \boldsymbol{V}_{\boldsymbol{xx}}f_{\boldsymbol{x}})\,\delta\boldsymbol{x}\rangle\Big]\Big\}. \end{aligned} \tag{31}$$

When capture occurs[6], we must have the Hamiltonian of the value function be zero as a necessary condition for the players' saddle-point controls [27], [28] i.e.

$$\boldsymbol{H}_{\boldsymbol{u}}(t;\boldsymbol{x}_r,\boldsymbol{u}_r^{\star},\boldsymbol{v}_r,\boldsymbol{V}_{\boldsymbol{x}}) = 0; \boldsymbol{H}_{\boldsymbol{v}}(t;\boldsymbol{x}_r,\boldsymbol{u}_r,\boldsymbol{v}_r^{\star},\boldsymbol{V}_{\boldsymbol{x}}) = 0. \tag{32}$$

where $\boldsymbol{u}_r^{\star}$ and $\boldsymbol{v}_r^{\star}$ respectively represent the optimal control laws for both players at time t.

A state-control relationship of the following form is sought:

$$\delta\boldsymbol{u} = \boldsymbol{k}_{\boldsymbol{u}}\delta\boldsymbol{x}, \quad \delta\boldsymbol{v} = \boldsymbol{k}_{\boldsymbol{v}}\delta\boldsymbol{x} \tag{33}$$

so that (31) in the context of (32) yields

$$\boldsymbol{H}_{\boldsymbol{u}} + \boldsymbol{H}_{\boldsymbol{uu}}\delta\boldsymbol{u} + \left(\boldsymbol{H}_{\boldsymbol{ux}} + f_{\boldsymbol{u}}^T\boldsymbol{V}_{\boldsymbol{xx}}\right)\delta\boldsymbol{x} + \frac{1}{2}\boldsymbol{H}_{\boldsymbol{uv}}\delta\boldsymbol{v} = 0 \tag{34a}$$

$$\boldsymbol{H}_{\boldsymbol{v}} + \boldsymbol{H}_{\boldsymbol{vv}}\delta\boldsymbol{v} + \left(\boldsymbol{H}_{\boldsymbol{vx}} + f_{\boldsymbol{v}}^T\boldsymbol{V}_{\boldsymbol{xx}}\right)\delta\boldsymbol{x} + \frac{1}{2}\boldsymbol{H}_{\boldsymbol{vu}}\delta\boldsymbol{u} = 0. \tag{34b}$$

Using (32) and equating like terms in the resulting equation to those in (33), we have the following for the state gains:

$$\begin{aligned} \boldsymbol{k}_{\boldsymbol{u}} &= -\frac{1}{2}\boldsymbol{H}_{\boldsymbol{uu}}^{-1}\left[\boldsymbol{H}_{\boldsymbol{uv}}\boldsymbol{k}_{\boldsymbol{v}} + 2\left(\boldsymbol{H}_{\boldsymbol{ux}} + f_{\boldsymbol{u}}^T\boldsymbol{V}_{\boldsymbol{xx}}\right)\right], \text{ and} \\ \boldsymbol{k}_{\boldsymbol{v}} &= -\frac{1}{2}\boldsymbol{H}_{\boldsymbol{vv}}^{-1}\left[\boldsymbol{H}_{\boldsymbol{vu}}\boldsymbol{k}_{\boldsymbol{u}} + 2\left(\boldsymbol{H}_{\boldsymbol{vx}} + f_{\boldsymbol{v}}^T\boldsymbol{V}_{\boldsymbol{xx}}\right)\right]. \end{aligned} \tag{35}$$

Putting the maximizing $\delta\boldsymbol{u}$ and the minimizing $\delta\boldsymbol{v}$ into (31), whilst neglecting terms in $\delta\boldsymbol{x}$ beyond second-order, we have

$$\begin{aligned} \min\Big\{\boldsymbol{0}, \Big[\boldsymbol{H} + \Big\langle\Big(\boldsymbol{H}_{\boldsymbol{x}} + \boldsymbol{V}_{\boldsymbol{xx}}f + \boldsymbol{k}_{\boldsymbol{u}}^T\boldsymbol{H}_{\boldsymbol{u}} + \boldsymbol{k}_{\boldsymbol{v}}^T\boldsymbol{H}_{\boldsymbol{v}}\Big), \delta\boldsymbol{x}\Big\rangle \\ + \frac{1}{2}\Big\langle \delta\boldsymbol{x}, \Big(\boldsymbol{H}_{\boldsymbol{xx}} + f_{\boldsymbol{x}}^T\boldsymbol{V}_{\boldsymbol{xx}} + \boldsymbol{V}_{\boldsymbol{xx}}f_{\boldsymbol{x}} + \boldsymbol{k}_{\boldsymbol{u}}^T\boldsymbol{H}_{\boldsymbol{uu}}\boldsymbol{k}_{\boldsymbol{u}} \\ + \boldsymbol{k}_{\boldsymbol{v}}^T\boldsymbol{H}_{\boldsymbol{vv}}\boldsymbol{k}_{\boldsymbol{v}}\Big)\,\delta\boldsymbol{x}\Big\rangle\Big]\Big\}. \end{aligned} \tag{36}$$

Now, we can compare coefficients with the l.h.s. of (30) and find the quadratic expansion of the nominal value function admits the following analytical solution on its right hand side:

$$-\frac{\partial \boldsymbol{V}_r}{\partial t} - \frac{\partial \tilde{\boldsymbol{V}}}{\partial t} = \min\{\boldsymbol{0}, \boldsymbol{H}\} \tag{37a}$$

$$-\frac{\partial \boldsymbol{V}_{\boldsymbol{x}}}{\partial t} = \min\Big\{\boldsymbol{0}, \boldsymbol{H}_{\boldsymbol{x}} + \boldsymbol{V}_{\boldsymbol{xx}}f + \boldsymbol{k}_{\boldsymbol{u}}^T\boldsymbol{H}_{\boldsymbol{u}} + \boldsymbol{k}_{\boldsymbol{v}}^T\boldsymbol{H}_{\boldsymbol{v}}\Big\} \tag{37b}$$

$$\begin{aligned} -\frac{\partial \boldsymbol{V}_{\boldsymbol{xx}}}{\partial t} = \min\Big\{\boldsymbol{0}, \boldsymbol{H}_{\boldsymbol{xx}} + f_{\boldsymbol{x}}^T\boldsymbol{V}_{\boldsymbol{xx}} + \boldsymbol{V}_{\boldsymbol{xx}}f_{\boldsymbol{x}} \\ + \boldsymbol{k}_{\boldsymbol{u}}^T\boldsymbol{H}_{\boldsymbol{uu}}\boldsymbol{k}_{\boldsymbol{u}} + \boldsymbol{k}_{\boldsymbol{v}}^T\boldsymbol{H}_{\boldsymbol{vv}}\boldsymbol{k}_{\boldsymbol{v}}\Big\}. \end{aligned} \tag{37c}$$

Furthermore, comparing the above with (28) and noting that $-\dot{\boldsymbol{V}}_r = 0$[7], we find that

$$-\dot{\tilde{\boldsymbol{V}}} = -\frac{\partial \tilde{\boldsymbol{V}}}{\partial t} \triangleq \min\{\boldsymbol{0}, \boldsymbol{H} - \boldsymbol{H}(t;\boldsymbol{x}_r,\boldsymbol{u}_r,\boldsymbol{v}_r,\boldsymbol{V}_{\boldsymbol{x}})\} \tag{38a}$$

$$-\dot{\boldsymbol{V}}_{\boldsymbol{x}} = \min\{\boldsymbol{0}, \boldsymbol{H}_{\boldsymbol{x}} + \boldsymbol{V}_{\boldsymbol{xx}}\left(f - f(t;\boldsymbol{x}_r,\boldsymbol{u}_r,\boldsymbol{v}_r)\right) \tag{38b}$$

$$+ \boldsymbol{k}_{\boldsymbol{u}}^T\boldsymbol{H}_{\boldsymbol{u}} + \boldsymbol{k}_{\boldsymbol{v}}^T\boldsymbol{H}_{\boldsymbol{v}}\Big\} \tag{38c}$$

$$\begin{aligned} -\frac{\partial \boldsymbol{V}_{\boldsymbol{xx}}}{\partial t} = \min\Big\{\boldsymbol{0}, \boldsymbol{H}_{\boldsymbol{xx}} + f_{\boldsymbol{x}}^T\boldsymbol{V}_{\boldsymbol{xx}} + \boldsymbol{V}_{\boldsymbol{xx}}f_{\boldsymbol{x}} \\ + \boldsymbol{k}_{\boldsymbol{u}}^T\boldsymbol{H}_{\boldsymbol{uu}}\boldsymbol{k}_{\boldsymbol{u}} + \boldsymbol{k}_{\boldsymbol{v}}^T\boldsymbol{H}_{\boldsymbol{vv}}\boldsymbol{k}_{\boldsymbol{v}}\Big\} \end{aligned} \tag{38d}$$

[5]This is because the l.h.s. was truncated at second order expansion previously. Ultimately, the $\delta\boldsymbol{u}, \delta\boldsymbol{v}$ terms will be quadratic in $\delta\boldsymbol{x}$ if we neglect h.o.t.

[6]A capture occurs when $\boldsymbol{E}$'s separation from $\boldsymbol{P}$ becomes less than a specified e.g. capture radius.

[7]The stage cost is zero for a reachable sets computational problem.

where $\boldsymbol{k_u}$ and $\boldsymbol{k_v}$ are as defined in (33). Note that at a saddle point, the first-order necessary condition for optimality cf. (32) implies

$$-\dot{\tilde{\boldsymbol{V}}} = \min\{\boldsymbol{0}, \boldsymbol{H} - \boldsymbol{H}(t;\boldsymbol{x}_r,\boldsymbol{u}_r,\boldsymbol{v}_r,\boldsymbol{V_x})\} \tag{39a}$$

$$-\dot{\boldsymbol{V}}_{\boldsymbol{x}} = \min\{0, \boldsymbol{H_x} + \boldsymbol{V_{xx}}(f - f(\boldsymbol{x}_r,\boldsymbol{u}_r,\boldsymbol{v}_r))\} \tag{39b}$$

$$-\frac{\partial \boldsymbol{V_{xx}}}{\partial t} = \min\Big\{\boldsymbol{0}, \boldsymbol{H_{xx}} + f_{\boldsymbol{x}}^T\boldsymbol{V_{xx}} + \boldsymbol{V_{xx}}f_{\boldsymbol{x}} + \boldsymbol{k_u}^T\boldsymbol{H_{uu}}\boldsymbol{k_u} + \boldsymbol{k_v}^T\boldsymbol{H_{vv}}\boldsymbol{k_v}\Big\} \tag{39c}$$

whereupon every quantity in (39) is evaluated at $\boldsymbol{x}_r, \boldsymbol{u}^\star$.

The boundary conditions for (39) at $t = 0$ is

$$\boldsymbol{V}(\boldsymbol{x}_r, 0) = \boldsymbol{g}(0;\boldsymbol{x}_r(0)); \tag{40}$$

so that

$$\tilde{\boldsymbol{V}}(0) = 0 \tag{41a}$$

$$\boldsymbol{V_x}(0) = \boldsymbol{g_x}(0;\boldsymbol{x}_r(0)) \tag{41b}$$

$$\boldsymbol{V_{xx}}(0) = \boldsymbol{g_{xx}}(0;\boldsymbol{x}_r(0)). \tag{41c}$$

The following control laws are then applied

$$\boldsymbol{u} = \boldsymbol{u}_r + \boldsymbol{k_u}\delta\boldsymbol{x}, \tag{42}$$

$$\boldsymbol{v} = \boldsymbol{v}_r + \boldsymbol{k_v}\delta\boldsymbol{x}. \tag{43}$$

Therefore, at any time on a nominal trajectory, a local approximation of $\boldsymbol{V}$ consists in solving the following system of equations

$$\begin{aligned} -\left[\boldsymbol{E} + \boldsymbol{F}\delta\boldsymbol{x} + \frac{1}{2}\delta\boldsymbol{x}\boldsymbol{G}\delta\boldsymbol{x}\right] &= \min\{\boldsymbol{0}, \boldsymbol{H} \\ -\boldsymbol{H}(t;\boldsymbol{x}_r,\boldsymbol{u}_r,\boldsymbol{v}_r,\boldsymbol{V_x}) + \boldsymbol{H_x} &+ \boldsymbol{V_{xx}}(f - f(t;\boldsymbol{x}_r,\boldsymbol{u}_r,\boldsymbol{v}_r)) \\ +\boldsymbol{H_{xx}} + f_{\boldsymbol{x}}^T\boldsymbol{V_{xx}} + \boldsymbol{V_{xx}}f_{\boldsymbol{x}} &+ \boldsymbol{k_u}^T\boldsymbol{H_{uu}}\boldsymbol{k_u} + \boldsymbol{k_v}^T\boldsymbol{H_{vv}}\boldsymbol{k_v}\Big\} \end{aligned} \tag{44}$$

where $\boldsymbol{E}, \boldsymbol{F},$ and $\boldsymbol{G}$ are appropriately defined.

V. Conclusion.

In this letter, we have presented a second order state-feedback local approximation scheme for computing the minimizing disturbance and maximizing controller that constitutes the optimal value function in a minimax dynamic game setting for HJ-Cauchy type equations. Leveraging second-order truncation to Taylor's series expansion of all nonlinear trajectories and system dynamics requisite for computing the reachable sets, our scheme provides an approximation scheme for determining reachable sets as stipulated in (9).

References

[1] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 947–957, 2005.

[2] J. A. Sethian, "Level Set Methods And Fast Marching Methods: Evolving Interfaces In Computational Geometry, Fluid Mechanics, Computer Vision, And Materials Science," *Robotica*, vol. 18, no. 1, pp. 89–92, 2000.

[3] ——, "A Fast Marching Level Set Method For Monotonically Advancing Fronts," *Proceedings of the National Academy of Sciences*, vol. 93, no. 4, pp. 1591–1595, 1996.

[4] S. Osher and C.-W. Shu, "Efficient Implementation of Essentially Non-Oscillatory Shock Capturing Schemes," NASA Langley Research Center, Hampton, Virginia, Tech. Rep., 1987.

[5] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations," *Journal of computational physics*, vol. 79, no. 1, pp. 12–49, 1988.

[6] I. Mitchell, "Games of two identical vehicles," *Dept. Aeronautics and Astronautics, Stanford Univ.*, no. July, pp. 1–29, 2001.

[7] ——, "A Robust Controlled Backward Reach Tube with (Almost) Analytic Solution for Two Dubins Cars," *EPiC Series in Computing*, vol. 74, pp. 242–258, 2020.

[8] I. M. Mitchell, "A Toolbox of Level Set Methods." *UBC Department of Computer Science Technical Report TR-2007-11*, 2007.

[9] S. Osher and C.-W. Shu, "High-Order Essentially Nonoscillatory Schemes for Hamilton-Jacobi Equations," *SIAM Journal of Numerical Analysis*, vol. 28, no. 4, pp. 907–922, 1991.

[10] S. K. Mitter, "Successive approximation methods for the solution of optimal control problems," *Automatica*, vol. 3, no. 3-4, pp. 135–149, 1966.

[11] S. R. McReynolds, "The successive sweep method and dynamic programming," *Journal of Mathematical Analysis and Applications*, vol. 19, no. 3, pp. 565–598, 1967.

[12] D. H. Jacobson, "New second-order and first-order algorithms for determining optimal control: A differential dynamic programming approach," *Journal of Optimization Theory and Applications*, vol. 2, no. 6, pp. 411–440, 1968.

[13] D. H. Jacobson and D. Q. Mayne, *Differential Dynamic Programming*. American Elsevier Publishing Company, Inc., New York, NY, 1970.

[14] O. Ogunmolu, N. Gans, and T. Summers, "Minimax iterative dynamic game: Application to nonlinear robot control tasks," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 6919–6925.

[15] M. G. Crandall and A. Majda, "Monotone Difference Approximations For Scalar Conservation Laws," *Mathematics of Computation*, vol. 34, no. 149, pp. 1–21, 1980.

[16] M. Crandall and A. Majda, "The method of fractional steps for conservation laws," *Numerische Mathematik*, vol. 34, no. 3, pp. 285–314, 1980.

[17] J.-P. Aubin, A. M. Bayen, and P. Saint-Pierre, *Viability Theory*, 2011, no. August 2015.

[18] L. Evans and P. E. Souganidis, "Differential games and representation formulas for solutions of Hamilton-Jacobi-Isaacs equations," *Indiana Univ. Math. J*, vol. 33, no. 5, pp. 773–797, 1984.

[19] M. G. Crandall and P.-L. Lions, "Viscosity solutions of hamilton-jacobi equations," *Transactions of the American mathematical society*, vol. 277, no. 1, pp. 1–42, 1983.

[20] L. Evans and P. E. Souganidis, "Differential Games And Representation Formulas For Solutions Of Hamilton-Jacobi-Isaacs Equations," *Indiana Univ. Math. J*, vol. 33, no. 5, pp. 773–797, 1984.

[21] R. Jhala and R. Majumdar, "Software model checking," *ACM Computing Surveys (CSUR)*, vol. 41, no. 4, pp. 1–54, 2009.

[22] P.-L. Lions, *Generalized solutions of Hamilton-Jacobi equations*. London Pitman, 1982, vol. 69.

[23] R. Bellman, *Dynamic programming*. Princeton University Press, 1957.

[24] L.-Z. Liao and C. A. Shoemaker, "Convergence in unconstrained discrete-time differential dynamic programming," *IEEE Transactions on Automatic Control*, vol. 36, no. 6, pp. 692–706, 1991.

[25] C. Gu, "Qlmor: A projection-based nonlinear model order reduction approach using quadratic-linear representation of nonlinear systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 30, no. 9, pp. 1307–1320, 2011.

[26] L.-Z. Liao and C. A. Shoemaker, "Convergence in unconstrained discrete-time differential dynamic programming," *IEEE Transactions on Automatic Control*, vol. 36, no. 6, pp. 692–706, 1991.

[27] A. Merz, "The game of two identical cars," *Journal of Optimization Theory and Applications*, vol. 9, no. 5, pp. 324–343, 1972.

[28] R. Isaacs, "Differential games: A mathematical theory with applications to warfare and pursuit, control and optimization." *Kreiger, Huntigton, NY*, 1965.