Neural networks for differential games

Olivier Bokanowski,*† Xavier Warin‡

February 2, 2024

Abstract

We study deterministic optimal control problems for differential games with finite horizon. We propose new approximations of the strategies in feedback form, and show error estimates and a convergence result of the value in some weak sense for one of the formulations. This result applies in particular to neural networks approximations. This work follows some ideas introduced in Bokanowski, Prost and Warin (PDEA, 2023) for deterministic optimal control problems, yet with a simplified approach for the error estimates, which allows to consider a global optimization scheme instead of a time-marching scheme. We also give a new approximation result between the continuous and the semi-discrete optimal control value in the game setting, improving the classical convergence order $O(\Delta t^{1/2})$ to $O(\Delta t)$, under some assumptions on the dynamical system. Numerical examples are performed on elementary academic problems related to backward reachability, with exact analytic solutions given, as well as a two-player game in presence of state constraints. We use stochastic gradient type algorithms in order to deal with the minimax problem.

Keywords: differential games, two-player games, neural networks, deterministic optimal control, dynamic programming principle, Hamilton Jacobi Isaacs equation, front propagation, level sets, non-anticipative strategies.

Acknowledgments: This research benefited from the support of the FMJH Program PGMO and from the support to this program from EDF.

1 Introduction

There exists different formulations of the value of a differential games, with notable contributions from Isaacs [41], Fleming [33] (using piecewise constant controls), as

*Université Paris Cité, Laboratoire Jacques-Louis Lions (LJLL), F-75013 Paris, France
†Sorbonne Université, CNRS, LJLL, F-75005 Paris, France `olivier.bokanowski@u-paris.fr`
‡EDF R&D & FiME, 91120 Palaiseau, France `xavier.warin@edf.fr`

well as Friedman [34, 35]. The definition proposed by Elliot and Kalton [28], employing non-anticipative strategies, has become widely accepted (see in particular [14] for the equivalence of the notions).

Non-anticipative strategies, according to this definition, can be interpreted as the optimal response of the first player (hereafter denoted as "a"), to an adverse control from the second player (denoted as "b"), utilizing only past knowledge.

Building upon the viscosity approach introduced by Crandall and Lions in [24], the value is also a solution of an Hamilton-Jacobi-Issacs (HJI) equation for differential games [29]. Recent developments in differential games with state constraints and their characterization by Hamilton-Jacobi equations are discussed in Buckdan *et al.* [21]; see also Bettiol *et al.* [15], Cardaliaguet *et al.* [23], Bardi *et al.* [11].

Additionally, Vinter *et al.* [56, 31], relying on Cardaliaguet *et al.* [22], explore the possibility to represent the non-anticipative strategy as a function of the current state and of the adverse control [7]. This approach is also employed, for example, by Bardi and Soravia as illustrated in Eq. (2.5) and Theorem 2.3 of their work [12].

In our work, we adopt a semi-discrete time setting, which is more straightforward to consider. Our primary objective is to revisit equivalent definitions of the corresponding semi-discrete values.

On the other hand, the classical error bound for the time-marching approximation of the value of a differential game concerning its continuous limit is known to be of order $O(\tau^{1/2})$, where τ represents the time step mesh (Souganidis [55], Crandall-Lions [25] for finite difference schemes; see also the textbook [8]), and the proof relies on viscosity arguments.

In line with the approach presented in [17], our contribution is to establish an error bound of order $O(\tau)$ under a separability assumption on the dynamics, typically valid for pursuit-evasion games, and certain convexity assumptions.

When considering the numerical approximation of the value, pioneering works that employ a full-grid approach where first developed for the approximation of the Hamilton-Jacobi-Isaacs (HJ) partial differential equation. For games and practical implementations we mention the works of Bardi, Falcone and Soravia [9, 10], particularly in connection with semi-Lagrangian schemes, the finite difference approaches including Markov Chains approximations [46], finite difference schemes (including monotone schemes [25], semi-Lagrangian schemes as in [26, 30], ENO or WENO higher-order schemes [50, 52], finite element methods [42] [53], discontinuous Galerkin methods [39, 48], Additionally, max-plus approaches have been explored [1].

However, the utility of grid-based methods is constrained to low dimensions due to the curse of dimensionality. Consequently, alternative approaches have emerged, including sparse grid methods [18, 36], tree structure approximation algorithms (such as [3]), tensor decomposition methods [27], hierarchical approximations [2], radial basis function approaches [32], and more.

Recently, neural network approximations have been developed to represent, optimize, and address problems in average or large dimensions, often in conjunction with stochastic gradient algorithms.

It is worth noting that, in the realm of stochastic control, Deep Neural Network (DNN) approximations have previously found application in gas storage optimization, as in [13], where the neural network approximates the control (specifically the quantity of gas injected or withdrawn from the storage). This methodoly has been adapted and popularized recently for solving Backward Stochastic Differential Equations (BSDEs) in [37] (referred to as the deep BSDE algorithm), see also [38]. Additionally, notable works include [40] and [7], focusing on approximating stochastic control problems within finite horizons using Bellman's dynamic programming principle.

In the deterministic context, [54] employs DNNs to approximate the Hamilton-Jacobi-Bellman (HJB) equation, as represented in (2), for solving state-constrained reachability control problems in dimensions up to $d = 10$.

This work specifically investigates neural network approximations for deterministic differential games in the form of Eq. (1). The presented algorithms are demonstrated on a running cost optimal control problem, but the approach can be generalized to Bolza problems, as discussed in [4, 5]. Our particular emphasis lies in achieving a rigorous error analysis for such approximations.

We propose two schemes: one is a "global" scheme, which directly attempts to approximate the desired value; the other is a "local" scheme, or time-stepping scheme, that initiates from the terminal value and proceeds backward until it approximates the value at the initial time. This approach is similar to the "Lagrangian" scheme in [19] or in connection with the "Performance Iteration" scheme presented in [40].

We highlight two major differences compared to [19]. Firstly, we provide new representation formulas for games, presented as minimax expectation values over "feedback strategies" and feedback controls. One is associated with the approach of Elliot and Kalton, while the other appears to be novel. Each representation formula naturally lends itself to approximations using neural networks and optimization algorithms for the strategy, employing Stochastic Gradient Descent Ascent (SGDA) like algorithms.

Secondly, our convergence proof strategy is distinct and more straightforward. It enables us to establish the convergence of the "global" scheme in neural network spaces in a weak sense. In [19], the convergence was proved for a "local" time-marching scheme following the dynamic programming principle. However, within the game context, we encountered challenges in adapting the present proof strategy to establish the convergence of the "local" algorithm. We leave this aspect for future developments.

The structure of the paper is outlined as follows. Section 2 provides the general setting and definitions. A semi-discrete problem is introduced, accompanied by various formulas utilizing different concepts of non-anticipative strategies. An error bound of order $O(\tau)$ is established between the continuous problem and the semi-discrete counterpart, particularly applicable to dynamics decomposed in the form $f(x,a,b) = f_1(x,a) + f_2(x,b)$, as observed in pursuit-evasion games.

In Section 3, different expectation formulas for the semi-discrete value are presented. This includes formulations employing the classical notion of non-anticipative strategies, as well as a modified version.

Section 4 presents two Deep Neural Network (DNN) schemes. Following this, Section 5 provides error estimates and a convergence result for one of the algorithms.

Finally, Section 6 offers some elementary benchmark numerical tests. These tests include examples in dimensions $d = 2$, with analytic solutions provided for comparison purposes. Additionally, a two-player game example in dimension $d = 4$ is presented, with a comparative analysis involving a finite difference scheme.

Notations. Given any two sets X and Y we denote by $\mathcal{F}(X,Y)$, or Y^X, the set of functions from X to Y. If X and Y are Borel sets (with σ-algebra $\mathcal{B}_X$ and $\mathcal{B}_Y$ resp.), then we denote by $\mathcal{M}(X,Y)$ the set of measurable functions from $(X,\mathcal{B}_X)$ to $(Y,\mathcal{B}_Y)$. Unless otherwise precised, $|.|$ is a norm on $\mathbb{R}^q$ ($q \geq 1$). The notation $[\![p,q]\!] = \{p, p+1, \ldots, q\}$ is used, for any integers $p \leq q$. For any function $\alpha : \mathbb{R}^p \to \mathbb{R}^q$ for some $p,q \geq 1$, $[\alpha] := \sup_{y\neq x} \frac{|\alpha(y)-\alpha(x)|}{|y-x|}$ denotes the corresponding Lipschitz constant. We also denote $a \vee b := \max(a,b)$ for any $a,b \in \mathbb{R}$.

Data availability statement: we do not analyse or generate any datasets, because our work proceeds within a theoretical and mathematical approach.

2 Setting of the problem and first results

Preliminary definitions. Let the following assumptions hold on the sets A, B and functions f, g, φ:

(H0) *A and B are non-empty compact subsets of $\mathbb{R}^{n_A}$ and $\mathbb{R}^{n_B}$ with $n_A, n_B \geq 1$.*

(H1) *$f : \mathbb{R}^d \times A \times B \to \mathbb{R}^d$ is Lipschitz continuous and we denote $[f]_1, [f]_2, [f]_3 \geq 0$ constants such that*

$$|f(x,a,b) - f(x',a',b')| \leq [f]_1|x-x'| + [f]_2|a-a'| + [f]_3|b-b'|, \quad \forall (x,x') \in (\mathbb{R}^d)^2,\ \forall (a,a') \in A^2,\ \forall (b,b') \in B^2.$$

(H2) *$g : \mathbb{R}^d \to \mathbb{R}$ is Lipschitz continuous.*

(H3) *$\varphi : \mathbb{R}^d \to \mathbb{R}$ is Lipschitz continuous.*

The following standard definitions can be found in the textbook [8, Chap. VIII]. We consider $\mathcal{A}_T := \{a : (0,T) \to A,\ \textit{measurable}\}$ and similarly $\mathcal{B}_T := \{a : (0,T) \to B,\ \textit{measurable}\}$. The set of non-anticipative (continuous) strategies, denoted $\Gamma_{(0,T)}$, is defined as the set of functions $\alpha : \mathcal{B}_T \to \mathcal{A}_T$, such that for all $t \in [0,T]$, $b_{|[0,t]} \equiv \bar{b}_{|[0,t]} \Rightarrow \alpha[b]_{|[0,t]} \equiv \alpha[\bar{b}]_{|[0,t]}$. For given controls $a \in \mathcal{A}_T$ and $b \in \mathcal{B}_T$, we denote by $y^{a,b}_{0,x}$ the unique Carathéodory solution of $\dot{y}(s) = f(y(s), a(s), b(s))$, a.e. $s \in (0,T)$

and $y(0) = x$. The continuous value we consider is defined by

$$v_0(x) \quad = \quad \inf_{\alpha \in \Gamma_{(0,T)}} \sup_{b \in \mathcal{B}_T} \Big(\max_{s \in (0,T)} g(y_{0,x}^{\alpha[b],b}(s)) \Big) \bigvee \varphi(y_{0,x}^{\alpha[b],b}(T)). \tag{1}$$

The value $v_0(x)$ is also equal to the solution $v(t,x)$ at time $t = 0$ of the following Hamilton-Jacobi-Bellman-Isaacs (HJBI) partial differential equation with an obstacle term, in the viscosity sense (see for instance [16])

$$\min\left(-v_t - \max_{b \in B} \min_{a \in A} (\nabla_x v \cdot f(x,a,b)),\ v - g(x) \right) = 0, \quad t \in [0,T],\ x \in \mathbb{R}^d \tag{2a}$$

$$v(T,x) = \max(\varphi(x), g(x)), \quad x \in \mathbb{R}^d. \tag{2b}$$

Note that the "inf sup" in (1) is reverted into a "max min" in (2). This is classical for the HJI equation for games, one may also look at Lemma 3.3(ii) to understand this in a simplified context. In this game the first player with control a aims to minimize the cost, while the second player with control b aims to maximize the cost. Such a value is motivated by target problems, or backward reachability, with state constraints and under disturbances.

Remark 2.1. *By considering the functional* (1)*, assuming that* $\mathcal{T} = \{x, \varphi(x) \le 0\}$ *is the target and that* $\mathcal{K} := \{x,\ g(x) \le 0\}$ *is the set of state constraints, then* $v_0(x) \le 0$ *is equivalent to have, for any* $\varepsilon > 0$*, the existence of a non-anticipative strategy* α_ε *such that for any adverse control* $b \in \mathcal{B}_T$*,* $y_{0,x}^{\alpha_\varepsilon[b],b}(T) \in \mathcal{T}_\varepsilon$ *(we reach a neighborhood of the target set at final time) and* $y_{0,x}^{\alpha_\varepsilon[b],b}(t) \in \mathcal{K}_\varepsilon$ *for all* $t \in [0,T]$ *(we stay in a neighborhood of the set of state constraints), where* $\mathcal{T}_\varepsilon := \{x,\ \varphi(x) \le \varepsilon\}$ *and* $\mathcal{K}_\varepsilon := \{x,\ g(x) \le \varepsilon\}$*. Hence the value* $v_0(x)$ *is a level set function for the following "robust" backward reachable set under state constraints*

$$\Omega := \bigcap_{\varepsilon > 0} \{x \in \mathbb{R}^d,\ \exists \alpha \in \Gamma_{(0,T)},\ \forall b \in \mathcal{B}_T,\ y_{0,x}^{\alpha[b],b}(T) \in \mathcal{T}_\varepsilon \text{ and } y_{0,x}^{\alpha[b],b}(t) \in \mathcal{K}_\varepsilon\ \forall t \in [0,T]\}$$

in the sense that $\{x,\ v_0(x) \le 0\} \equiv \Omega$*. See also [21] for this type of problems. The above approach allows to consider state constraints problems and avoid technical difficulties concerning boundary of the set of state constraints and the definition of the value. This can be also applied to more general Bolza problem for games and with state constraints as explained in [4].*

Note that the methodology developed in this paper can be adapted to deal with other functional costs, such as the sum of a distributional cost with a terminal cost

$$\int_0^T \ell(y_{0,x}^{\alpha[b],b}(s), \alpha[b](s), b(s)) ds + \varphi(y_{0,x}^{\alpha[b],b}(T)). \tag{3}$$

The semi-discrete problem. Following [8, Chap. VIII], we introduce a semi-discrete problem, corresponding to a time discretization of the problem. For a given $N \geq 1$, the set of discrete controls are A^N and B^N. The set of *discrete non-anticipative strategies*, denoted S_N, is the set of measurable functions $\alpha : B^N \to A^N$ such that for any $0 \leq k \leq N-1$:

$$\big(\forall 0 \leq j \leq k,\, b_j = \bar{b}_j\big) \Rightarrow \big(\forall 0 \leq j \leq k,\, \alpha[b]_k = \alpha[\bar{b}]_k\big).$$

Hereafter we will allow both notations $\alpha[b]$ or $\alpha(b)$ when α is a strategy.

Remark 2.2. *Notice at this stage that the mesurability condition in the strategies is not really necessary (in particular Theorem 2.3 would hold with strategies simply defined as functions : $B^N \to A^N$). However later on all functions and strategies will be needed to be measurable because we will apply them to a random variable and consider expectations.*

This also means that $\alpha = (\alpha_0, \dots, \alpha_{N-1})$ where α_k is only a function of $(b_0, \dots, b_k)$:

$$\forall k, \quad \alpha_k = \alpha_k(b_0, \dots, b_k).$$

For any controls $a = (a_0, a_1, \dots) \in A^N$ and $b = (b_0, b_1, \dots) \in B^N$, we define $(X^{a,b}_{k,x})_{k \geq 0}$ recursively by

$$X^{a,b}_{0,x} = x \qquad \text{and} \qquad X^{a,b}_{k+1,x} = F(X^{a,b}_{k,x}, a_k, b_k),\ \forall k \geq 0.$$

Here $F : \mathbb{R}^d \times A \times B \to \mathbb{R}^d$ can be a one time-step approximation of the dynamics, and is assumed at least continuous in its variables. The simplest example is to consider the Euler scheme with time step $\tau = T/N$:

$$F(x, a, b) = x + \tau f(x, a, b). \tag{4}$$

The semi-discrete value (corresponding to the approach of Elliott and Kalton [28]) is then defined by

$$V_0(x) = \inf_{\alpha \in S_N} \sup_{b \in B^N} J_0(x, \alpha[b], b) \tag{5}$$

for some given functional $J_0 : \mathbb{R}^d \times A^N \times B^N$. In the case of (1), J_0 can be defined by

$$J_0(x, a, b) := \left(\max_{0 \leq k \leq N-1} g(X^{a,b}_{k,x}) \right) \bigvee \varphi(X^{a,b}_{N,x}),$$

where $\varphi(.)$, $g(.)$ and $F(.)$ are given functions. (In the case of (3), we could consider $\sum_{k=0}^{N-1} \tau \ell(X^{a,b}_{k,x}, a_k, b_k) + \varphi(X^{a,b}_{N,x})$.) Notice that if $g = -\infty$ (or some large negative value), then $J_0(x, a, b) = \varphi(X^{a,b}_{N,x})$.

More general Runge-Kutta schemes can be considered for the definition of F, as well as multi-step approximations, as in [19], but in the present work this will not be necessary in order to obtain the convergence of the neural network schemes.

The first objective is to recall some equivalent formulations for $V_0(x)$.

Following Fleming's approach [33], corresponding to using piecewise constant controls, let us also define the value $\overline{V}_0(x)$ by

$$\overline{V}_0(x) := \max_{b_0} \min_{a_0} \quad \cdots \quad \max_{b_{N-1}} \min_{a_{N-1}} J_0(x, a, b)$$

for controls $a = (a_0, \ldots, a_{N-1}) \in A^N$ and controls $b = (b_0, \ldots, b_{N-1}) \in B^N$.

We also introduce a third definition, which is

$$\tilde{V}_0(x) := \inf_{\alpha \in \mathcal{G}^N} \sup_{b \in B^N} \tilde{J}_0(x, \alpha[b], b) \tag{6}$$

where

$$\mathcal{G} := \mathcal{M}(\mathbb{R}^d \times B, A),$$

(hence $\alpha = (\alpha_0, \ldots, \alpha_{N-1}) \in \mathcal{G}^N$ means that $\alpha_k \in \mathcal{G}$ for all k), and the payoff function $\tilde{J}_0$ is the same as the usual payoff function J_0 but using modified trajectories $\tilde{X}^{a,b}_{k,x}$ such that $\tilde{X}^{a,b}_{0,x} := x$ and

$$\tilde{X}^{a,b}_{k+1,x} := F(\tilde{X}^{a,b}_{k,x}, \alpha_k(\tilde{X}^{a,b}_{k,x}, b_k), b_k), \quad k \geq 0.$$

More precisely, whenever $\alpha \in \mathcal{G}^N$ and $b \in B^N$, we have

$$\tilde{J}_0(x, \alpha[b], b) := \max_{0 \leq k \leq N-1} g(\tilde{x}_k) \vee \varphi(\tilde{x}_N) \tag{7}$$

with $\tilde{x}_k = \tilde{X}^{\alpha[b],b}_{k,x}$, now defined by

$$\tilde{x}_0 = x \text{ and } \tilde{x}_{k+1} = F(\tilde{x}_k, \alpha_k(\tilde{x}_k, b_k), b_k), \; k \geq 0. \tag{8}$$

Notice that $a_0 = \alpha_0(x, b_0)$, making explicit a possible dependence in x in α_0. This is not really necessary for our results at this stage in the sense that, for a given $x \in \mathbb{R}^d$ we could have also considered controls $\alpha \in \mathcal{M}(B, A) \times \mathcal{G}^{N-1}$ and, instead of (6):

$$\tilde{V}_0(x) := \inf_{\alpha \in \mathcal{M}(B,A) \times \mathcal{G}^{N-1}} \sup_{b \in B^N} \tilde{J}_0(x, \alpha[b], b). \tag{9}$$

We shall still allow the notation $\alpha(x, b) \equiv \alpha[b](x)$, as well as $\alpha_k(x, b_k) \equiv \alpha_k[b_k](x)$.

Notice, in the definition of V_0 and of $\tilde{V}_0(x)$, that the "inf" is a "min". Indeed, by using a measurable selection theorem [20, Theorem 1], the measurability of $J_0(x, ., .)$ (resp. $\tilde{J}_0(x, ., .)$) and the compactness of A and B, there exists a minimum α which is measurable (see also the direct definition of optimal strategies $\bar{\alpha}_k$ and α^*_k in Theorem 2.5 below).

Let us first remark that the previous values are identical.

Theorem 2.3. *For all $x \in \mathbb{R}^d$,*

$$V_0(x) = \overline{V}_0(x) = \tilde{V}_0(x).$$

Remark 2.4. *Note that the feedback controls used in $\tilde{V}_0$ are classical and natural, but the fact that the value $\tilde{V}_0(x)$ exactly corresponds to $V_0(x)$ seems less classical (we were not able to find a reference for this statement).*

We have also some information on the corresponding optimal strategies. We first need to introduce, in the same way, for $x \in \mathbb{R}^d$, the intermediate values

$$V_k(x) := \max_{b_0} \min_{a_0} \quad \cdots \quad \max_{b_{N-k-1}} \min_{a_{N-k-1}} \left(\max_{j=0,\ldots,N-k-1} g(X^{a,b}_{j,x}) \right) \bigvee \varphi(X^{a,b}_{N-k,x}).$$

The following dynamic programming relation is well known: for $0 \leq k \leq N-1$ and $x \in \mathbb{R}^d$:

$$V_k(x) := \max_{b \in B} \min_{a \in A} \left(g(x) \bigvee V_{k+1}(F(x,a,b)) \right). \tag{10}$$

In view of Theorem 2.3 (the identity $V_0(x) = \overline{V}_0(x)$), the function $V_k(x)$ also satisfies

$$V_k(x) = \inf_{\alpha \in \mathcal{M}(B^{N-k},A)} \sup_{b \in B^{N-k}} J_k(x, \alpha[b], b) \tag{11}$$

where $J_k(x, \alpha[b], b) := \max_{k \leq j \leq N-1} g(x_k) \vee \varphi(x_{N-k})$, with $x_{k+1} = F(x_k, \alpha_k[b_0, \ldots, b_k], b_k)$ for $k \geq 0$ and $x_0 = x$. In the same way, as for the identity $\tilde{V}_0(x) = \overline{V}_0(x)$ in Theorem 2.3, we have

$$V_k(x) = \inf_{\alpha \in \mathcal{M}(\mathbb{R}^d \times B, A)^{N-k}} \sup_{b \in B^{N-k}} \tilde{J}_k(x, \alpha[b], b) \tag{12}$$

where $\tilde{J}_k(x, \alpha[b], b) := \max_{k \leq j \leq N-1} g(\tilde{x}_k) \vee \varphi(\tilde{x}_{N-k})$, with $\tilde{x}_{k+1} = F(\tilde{x}_k, \alpha_k[\tilde{x}_k, b_k], b_k)$ for $k \geq 0$ and $\tilde{x}_0 = x$. From these definitions we can deduce the following result.

Theorem 2.5. *(i) Let $x \in \mathbb{R}^d$. Any element $\bar{\alpha} = (\bar{\alpha}_k)_{0 \leq k \leq N-1}$ of S_N such that, for all k,*

$$\bar{\alpha}_k[b_0, \ldots, b_k] \in \operatorname*{argmin}_{a_k \in A} g(\bar{x}^b_k) \vee V_{k+1}(F(\bar{x}^b_k, a_k, b_k)), \quad \text{for a.e. } (b_i)_{0 \leq i \leq k} \in B^{k+1}$$

where $\bar{x}^b_0 := x$ and for $0 \leq i \leq k-1$: $\bar{x}^b_{i+1} = F(\bar{x}^b_i, \bar{\alpha}_i[b_0, \ldots, b_i], b_i)$, is an optimal non-anticipative strategy for $V_0(x)$ in the sense that it reaches the infimum in (11).
(ii) Any element $\alpha^* = (\alpha^*_k)_{0 \leq k \leq N-1}$ of $\mathcal{G}^N$ such that, for all k,

$$\alpha^*_k(x,b) \in \operatorname*{argmin}_{a \in A} g(x) \vee V_{k+1}(F(x,a,b)), \quad \text{for a.e. } (x,b) \in \mathbb{R}^d \times B$$

is an optimal non-anticipative strategy for $\tilde{V}_0(x)$, for a.e. $x \in \mathbb{R}^d$ in the sense that it reaches the infimum in (12).

Notice that from (i)-(ii), any optimal non-anticipative strategy $\alpha^* \in \mathcal{G}^N$ leads to an optimal non-anticipative strategy $\bar{\alpha} \in \mathcal{S}_N$ defined by

$$\bar{\alpha}_k(x, b_0, \ldots, b_k) := \alpha_k^*(\bar{x}_k^b, b_k), \quad \forall k \tag{13}$$

where $\mathcal{S}_N$ is now the set of measurable functions $\alpha : \mathbb{R}^d \times B^N \to A$ such that α_k is only a function of x and $b_0, \ldots, b_k$:

$$\alpha_k(x, b_0, \ldots, b_N) = \alpha_k(x, b_0, \ldots, b_k), \quad k = 0, \ldots, N-1, \ a.e.\ (x, b) \in \mathbb{R}^d \times B^N.$$

Theorems 2.3 and 2.5 will be proved in Appendix A. The equality $V_0 = \overline{V}_0$ is well known since the works of Elliott and Kalton [28], but the other identity with $\tilde{V}_0$ seems classical but we were not able to find a reference for it. It will be useful to define our numerical schemes. As a consequence, the non-anticipative strategy $a_k = \alpha_k(b_0, \ldots, b_k)$ (for a given starting point x), can be also searched in the form of $a_k = \tilde{\alpha}_k(x_k, b_k)$ where x_k corresponds to the position of the trajectory at time t_k.

Remark 2.6. *The advantage of the formulation of $\tilde{V}_0$ is that it reduces the complexity of the representation of the non-anticipative strategies: in each strategy for V_0, α_k is a function from $\mathbb{R}^d \times B^k$ to A - if we make explicit the dependency over $x \in \mathbb{R}^d$, while for $\tilde{V}_0$ it becomes only a function from $\mathbb{R}^d \times B$ to A.*

We also give here a new error estimate between the semi-discrete value $V_0(x)$ and the value of the continuous problem $v_0(x)$. When the value is Lipschitz continuous, as it is the case here, the known error estimate is of order $O(\tau^{1/2})$ (see e.g. [8]), and the proof can be obtained by using viscosity arguments. Adding a separability assumption on the dynamics, we can improve this result, for games, as follows.

Theorem 2.7. *Assume the dynamics has a separate dependency in the controls, that is:*

$$f(x, a, b) = f_1(x, a) + f_2(x, b) \tag{14}$$

for some Lipschitz continuous functions f_1, f_2. Assume furthermore that $f_1(x, A)$ and $f_2(x, B)$ are convex for all $x \in \mathbb{R}^d$. Consider the Euler scheme approximation $F(x, a, b) = x + \tau f(x, a, b)$. Then there exists a constant $C \geq 0$ such that, for all $x \in \mathbb{R}^d$:

$$|v_0(x) - V_0(x)| \leq C(1 + |x|)\ \tau.$$

Notice that for pursuit-evasion games, with dynamics of the form

$$f((y, z), a, b) = (g_1(y, a), g_2(z, b))$$

the relation (14) holds true with $f_1(x, a) = (g_1(y, a), 0)$ and $f_2(x, a) = (0, g_2(z, b))$.

Theorem (2.7) is proved in Appendix B. The proof is based on approximation of trajectories. Note that for more complex Runge Kutta schemes F and multi-step approximations, a similar error estimate of order $O(\tau)$ would hold (see [19]).

3 Expectation formula for min-max problems

Let us recall the following Lemma that links pointwise minimization over open-loop controls $a \in A$ and minimization of an averaged value over feedback controls $a \in \mathcal{A}$. This is the same Lemma as [19, Lemma 3.2] stated continuous functions Q, but that is now stated for slightly more general measurable functions Q.

From now on we consider a random variable X on some probability space and consider the following assumptions.

(H4) *X is a random variable with values in $\mathbb{R}^d$, and $\mathbb{E}[|X|] < \infty$.*
(H5) X admits a Lebesgue measurable density ρ supported on $\bar{\Omega}$ for some $\Omega \subset \mathbb{R}^d$ and such that $\rho(x) > 0$ a.e. $x \in \Omega$, with $|\partial\Omega| = 0$ (i.e., $\partial\Omega$ is negligible).

Lemma 3.1. *Assume (H4). Let a given measurable function $Q : \mathbb{R}^d \times A \times B \to \mathbb{R}$, with linear growth ($\exists C \geq 0$, $\forall (x,a)$, $Q(x,a) \leq C(1+|x|)$), and such that $a \in A \to Q(x,a)$ is continuous for a.e. x.*
(i)

$$\mathbb{E}\Big[\inf_{a\in A} Q(X,a)\Big] = \inf_{a\in\mathcal{A}} \mathbb{E}\Big[Q(X,a(X))\Big]. \tag{15}$$

Furthermore an optimal $a^* \in \mathcal{A}$ that minimizes the quantity (15) *exists.*
(ii) Assume furthermore (H5). Then

$$\bar{a}(.) \in \operatorname*{argmin}_{a\in\mathcal{A}} \mathbb{E}\Big[Q(X,a(X))\Big] \quad \Longleftrightarrow \quad \Big(\bar{a}(x) \in \operatorname*{argmin}_{a\in A} Q(x,a),\ \ a.e.\ x \in \Omega\Big).$$

Proof. A proof is given for sake of completeness.

(i) Let I (resp. J) denote the left hand side (resp. r.h.s.) of (15). For any $a \in \mathcal{A}$, we have $Q(X,a(X)) \geq \min_{a\in A} Q(X,a)$ and therefore $\mathbb{E}[Q(X,a(X)] \geq I$, hence $J \geq I$. Conversely, there exists $a^* \in \mathcal{A}$ such that $a^*(x) \in \operatorname*{argmin}_{a\in A} Q(x,a)$ for a.e. $x \in \mathbb{R}^d$ (by a measurable selection Theorem, see for instance Theorem 1 of [20]). Hence $J \leq \mathbb{E}\big[Q(X,a^*(X))\big] = I$, which concludes to (i) and the existence of a minimizer $a^* \in \mathcal{A}$.

(ii) For any minimizer $\bar{a} \in \mathcal{A}$ of J, we have: $\mathbb{E}[Q(X,\bar{a}(X)) - \inf_{a\in A} Q(X,a)] = 0$. But the integrand is a.s. positive, hence we have $Q(X,\bar{a}(X)) = \inf_{a\in A} Q(X,a)$ a.s. Therefore $\bar{a}(X) \in \operatorname*{argmin}_{a\in A} Q(X,a)$ a.s., from which we can conclude, for a.e. x in the support of ρ, that $\bar{a}(x) \in \operatorname*{argmin}_{a\in A} Q(x,a)$ (using assumption (H5)).

□

In the same way, in order to tackle the minimization problem such as

$$\inf_{a\in A} \sup_{b\in B} Q(x,a,b)$$

for a measurable function $Q : \mathbb{R}^d \times A \times B \to \mathbb{R}$ (continuous in its variable (a, b), for a.e. x), we would like to proceed by considering inf/sup of an averaged functional using feedback controls $a \in \mathcal{A}$ and $b \in \mathcal{B}$. Notice that because Q is continuous in (a, b) and A and B are compact sets, the infimum or supremum of $Q(x, a, b)$ with respect to a and b are reached and we can write min/max without ambiguity. The previous Lemma can be extended as follows.

Lemma 3.2. *Assume (H4). Let a given measurable function $Q : \mathbb{R}^d \times A \times B \to \mathbb{R}$, with linear growth ($\exists C \geq 0$, $\forall (x, a, b)$, $Q(x, a, b) \leq C(1 + |x|)$), and such that $(a, b) \in A \times B \to Q(x, a, b)$ is continuous for a.e. x.*
(i)

$$\mathbb{E}\left[\inf_{a\in A}\sup_{b\in B} Q(X, a, b)\right] = \inf_{a\in\mathcal{A}}\sup_{b\in\mathcal{B}} \mathbb{E}\left[Q(X, a(X), b(X))\right]. \tag{16}$$

and an optimal $a^* \in \mathcal{A}$ that minimizes the quantity (16) exists.
(ii) Assume furthermore (H5). Let $\bar{a} \in \mathcal{A}$. The following statements are equivalent:

$$\bar{a}(.) \in \operatorname*{arginf}_{a\in\mathcal{A}} \sup_{b\in\mathcal{B}} \mathbb{E}\left[Q(X, a(X), b(X))\right] \equiv \operatorname*{argmin}_{a\in\mathcal{A}} \mathbb{E}\left[\max_{b\in B} Q(X, a(X), b)\right]$$

and

$$\bar{a}(x) \in \operatorname*{argmin}_{a\in A}\ \max_{b\in B} Q(x, a, b), \quad a.e.\ x \in \Omega.$$

Proof. By applying Lemma 3.1 to the function $R(x, a) = \max_{b\in B} Q(x, a, b)$, it holds

$$\mathbb{E}\big[\min_{a\in A}\max_{b\in B} Q(X, a, b)\big] = \min_{a\in\mathcal{A}} \mathbb{E}\big[\max_{b\in B} Q(X, a(X), b)\big]. \tag{17}$$

Then, for any $a \in \mathcal{A}$, and for any $b \in \mathcal{B}$ the following inequality is immediate:

$$\mathbb{E}[\max_{b\in B} Q(X, a(X), b)\big] \geq \mathbb{E}\big[Q(X, a(X), b(X))\big],$$

therefore also

$$\mathbb{E}[\max_{b\in B} Q(X, a(X), b)\big] \geq \sup_{b\in\mathcal{B}} \mathbb{E}\big[Q(X, a(X), b(X))\big]. \tag{18}$$

Indeed equality holds in (18): it suffices to choose $b^* \in \mathcal{B}$ such that

$$b^*(x) \in \operatorname*{argmax}_{b\in B} Q(x, a(x), b) \quad \text{a.e.} \quad x \in \mathbb{R}^d.$$

Hence using (17) and the equality in (18) leads to

$$\min_{a\in\mathcal{A}} \mathbb{E}\big[\max_{b\in B} Q(X, a(X), b)\big] = \inf_{a\in\mathcal{A}}\sup_{b\in\mathcal{B}} \mathbb{E}\big[Q(X, a(X), b(X))\big]$$

which concludes to the desired result.

The proof of *(ii)* is deduce from Lemma 3.1 *(ii)* applied to the function R. □

We now give a formula in order to reverse the order $\inf_a \sup_b$ and $\sup_b \inf_a$. In general the equality $\inf_{a\in A} \sup_{b\in B} Q(a,b) = \sup_{b\in B} \inf_{a\in A} Q(a,b)$ is only true when Q admits a saddle points (a^*, b^*) or more generally if for instance Q is convex-concave (for all b, $a \to Q(a,b)$ is convex and for all a, $b \to Q(a,b)$ is concave) by Von Neumann's Theorem [49].

Recalling that $A^B := \{f : B \to A\}$, notice that for all a,

$$\sup_{b\in B} Q(a,b) = \sup_{b[.]\in A^B} Q(a, b[a]). \tag{19}$$

The following result is a particular case of Theorem 1.4.1 of [34].

Lemma 3.3. *Let Q be a bounded function, and A, B any two sets.*

(i) We have

$$\inf_{a\in A} \sup_{b\in B} Q(a,b) = \sup_{b[.]\in A^B} \inf_{a\in A} Q(a, b[a]) \tag{20}$$

and, in view of (19)*, this is also equal to* $\inf_{a\in A} \sup_{b[.]\in A^B} Q(a, b[a])$.

(ii) In the same way, we have

$$\sup_{b\in B} \inf_{a\in A} Q(a,b) = \inf_{a[.]\in B^A} \sup_{b\in B} Q(a[b], b) \tag{21}$$

and this is also equal to $\sup_{b\in B} \inf_{a[.]\in B^A} Q(a[b], b)$.

(iii) The previous statements are still valid if Q is a measurable function and A^B (resp B^A) is replaced by the space of measurable functions $\mathcal{M}(B,A)$ (resp. $\mathcal{M}(A,B)$).

Proof. We give a proof for self-completeness. We assume that Q is continuous and inf and sup are reached to simplify the proofs, but otherwise the proof can be obtained by approximation arguments.

We focus on the proof of (20) (since (21) is similar). Let $\alpha := \inf_{a\in A} \sup_{b\in B} Q(a,b)$ and $\beta := \sup_{b[.]} \inf_{a\in A} Q(a, b[a])$. Obviously, $Q(a, b[a]) \le \sup_{b\in B} Q(a,b)$, for any a and $b[.]$. Hence $\inf_a Q(a, b[a]) \le \inf_{a\in A} \sup_{b\in B} Q(a,b)$, and taking the supremum over $b[.]$, we deduce $\beta \le \alpha$.

Conversely, for any $a \in A$, let $b[a] \in B$ such that $b[a] \in \text{argmax}_{b\in B} Q(a,b)$, so that $Q(a, b[a]) = \sup_{b\in B} Q(a,b)$. Then $\beta \ge \inf_{a\in A} Q(a, b[a]) = \inf_{a\in A} \sup_{b\in B} Q(a,b) = \alpha$. Therefore we conclude to $\beta = \alpha$.

The proof of (iii) is similar. □

Now we generalize the previous commutation results to the case of inf/sup over feedback controls and expectation formulas.

Lemma 3.4. *Assume (H4). Let a given measurable function $Q : \mathbb{R}^d \times A \times B \to \mathbb{R}$, with linear growth ($\exists C \geq 0$, $\forall (x,a,b)$, $Q(x,a,b) \leq C(1+|x|)$), and such that $(a,b) \in A \times Q(x,a,b)$ is continuous a.e. x. It holds*

$$\mathbb{E}\Big[\inf_{a\in A^B}\sup_{b\in B} Q(X,a[b],b)\Big] = \inf_{\alpha\in\mathcal{G}}\sup_{b\in\mathcal{B}} \mathbb{E}\Big[Q(X,\alpha(X,b(X)),b(X))\Big] \tag{22}$$

$$= \mathbb{E}\Big[\sup_{b\in B}\inf_{a\in A} Q(X,a,b)\Big] \tag{23}$$

$$= \sup_{b\in\mathcal{B}}\inf_{a\in\mathcal{A}} \mathbb{E}\Big[Q(X,a(X),b(X))\Big]. \tag{24}$$

where A^B denotes the set of functions from B to A, and $\mathcal{G} := \mathcal{M}(\mathbb{R}^d \times B, A)$.

Proof. We just have to prove the first equality (22). Indeed, the equality between the l.h.s. of (22) and (23) comes from Lemma 3.3, and the equality between (23) and (24) comes from Lemma 3.2 (reverting sup and inf). Let I (resp. J) denote the left- (resp. right-) hand side of (22).

For a.e. $x \in \mathbb{R}^d$ and for all $b \in B$, $\inf_{a\in A} Q(x,a,b)$ is reached by some $a = \alpha^*(x,b)$. Also, $\sup_{b\in B}\big(\inf_{a\in A} Q(x,a,b)\big)$ is reached by some $b = b^*(x)$ (by using the compactness assumptions on A and B). By using a measurable selection Theorem (see for instance Theorem 1 of [20]), we can find furthermore $\alpha^* \in \mathcal{G}$ and then $b^* \in \mathcal{B}$. By using Lemma 3.1 (with "sup" instead of "inf"), we obtain

$$J = \inf_{\alpha\in\mathcal{G}} \mathbb{E}\big[\sup_{b\in B} Q(X,\alpha(X,b),b)\big]. \tag{25}$$

Then, in particular,

$$J \leq \mathbb{E}\big[\sup_{b\in B} Q(X,\alpha^*(X,b),b)\big] = \mathbb{E}\big[\sup_{b\in B}\inf_{a\in A} Q(X,a,b)\big]$$

(where we have used the definition of α^* for the last identity). We also have $\sup_{b\in B}\inf_{a\in A} Q(x,a,b) = \inf_{a\in A^B}\sup_{b\in B} Q(x,a[b],b)$, for a.e. $x \in \mathbb{R}^d$ by Lemma 3.3. Hence we deduce that $J \leq I$.

Conversely, for any $\alpha \in \mathcal{G}$, $\mathbb{E}\big[\sup_{b\in B} Q(X,\alpha(X,b),b)\big] \geq \mathbb{E}\big[\sup_{b\in B}\inf_{a\in A} Q(X,a,b)\big]$. Therefore

$$J \geq \mathbb{E}\big[\sup_{b\in B}\inf_{a\in A} Q(X,a,b)\big].$$

By using again Lemma 3.3, we deduce that $J \geq I$. □

Now all the tools are in place to formulate a general expectation formula related to the definition (6) for the value $\tilde{V}_0$. Recall that $\mathcal{G} := \mathcal{M}(\mathbb{R}^d \times B, A)$.

Notation 3.5. *When $\alpha \in \mathcal{G}^N$ and $b \in \mathcal{B}^N$, we denote*

$$\tilde{J}_0(x, \alpha[b], b) := \max_{0 \le k \le N-1} g(\tilde{x}_k) \vee \varphi(\tilde{x}_N),$$

where $\tilde{x}_0 = x$ and $\tilde{x}_{k+1} = F(\tilde{x}_k, \alpha_k(\tilde{x}_k, b_k(\tilde{x}_k)), b_k(\tilde{x}_k))$ for $k \ge 0$. Of course in the case when b is constant (i.e., $b \in B^N$), we find the same definition as the previous function $\tilde{J}_0$ used in (6).

Recalling the definition of $\tilde{V}_0$ given by (6), by using similar arguments as in Lemma 3.4, we obtain the following representation formula for $\tilde{V}_0$.

Theorem 3.6. *Let assumptions (H0)-(H4) be satisfied.*

(i)

$$\mathbb{E}\big[\tilde{V}_0(X)\big] = \mathbb{E}\big[\inf_{\alpha \in \mathcal{G}^N} \sup_{b \in B^N} \tilde{J}_0(X, \alpha[b], b)\big] = \inf_{\alpha \in \mathcal{G}^N} \sup_{b \in \mathcal{B}^N} \mathbb{E}\big[\tilde{J}_0(X, \alpha[b], b)\big]. \tag{26}$$

(ii) Assume furthermore (H5), and that $\alpha^* \in \mathcal{G}^N$ in the r.h.s. of (26) *is optimal, then it is an optimal strategy for $\tilde{V}_0(x)$ in the sense that*

$$\tilde{V}_0(x) = \sup_{b \in B^N} \tilde{J}_0(x, \alpha^*[b], b), \quad a.e.\ x \in \Omega.$$

4 Algorithms

We propose two algorithms. The first one follows the min-max formulation (26) in order to characterize the optimality of α. The second algorithm follows the dynamic programming principle (in a similar way as in [19] for control problems). The algorithms are well defined assuming that X is a random variable on $\mathbb{R}^d$ with $\mathbb{E}[|X|] < \infty$.

Algorithm 1 (Global scheme) Let $\hat{\mathcal{G}}$ (resp. $\hat{\mathcal{B}}$) be approximation spaces for $\mathcal{G}$ (resp. $\mathcal{B}$). Let $\eta = (\eta_1, \eta_2)$ be in $(\mathbb{R}_+^*)^2$ (margin errors), and let X be some r.v. on $\mathbb{R}^d$.

- compute feedback strategy and control $(\hat{\alpha}, \hat{b}) \in \hat{\mathcal{G}}^N \times \hat{\mathcal{B}}^N$ according to

$$(\hat{\alpha}, \hat{b}) \in \eta - \arg \inf_{\alpha \in \hat{\mathcal{G}}^N} \sup_{b \in \hat{\mathcal{B}}^N} \mathbb{E}\Big[\tilde{J}_0(X, \alpha[b], b)\Big] \tag{27}$$

(in a sense made precise below)
- set

$$\hat{V}_0(x) := \tilde{J}_0(x, \hat{\alpha}[\hat{b}], \hat{b}) \tag{28}$$

More precisely, the notation "$\eta - \arg\inf\sup$" in (27) means, by convention, for some $\eta = (\eta_1, \eta_2)$, that

$$\sup_{b\in\hat{\mathcal{B}}^N} \mathbb{E}\big[\tilde{J}_0(X, \hat{\alpha}[b], b)\big] \le \inf_{\alpha\in\hat{\mathcal{G}}^N} \sup_{b\in\hat{\mathcal{B}}^N} \mathbb{E}\big[\tilde{J}_0(X, \alpha[b], b)\big] + \eta_1 \tag{29a}$$

and

$$\mathbb{E}\big[\tilde{J}_0(X, \hat{\alpha}[\hat{b}], \hat{b})\big] \ge \sup_{b\in\hat{\mathcal{B}}^N} \mathbb{E}\big[\tilde{J}_0(X, \hat{\alpha}[b], b)\big] - \eta_2. \tag{29b}$$

This allows for solving the min-max problem on $\hat{\mathcal{G}}^N \times \hat{\mathcal{B}}^N$ within some margin error, as this is the case in practice. From a computational point of view, the min-max problem will be approximated by an adapted version of the stochastic gradient descent-ascent algorithm (SGDA), see Section 6.

Algorithm 2 (Local scheme) Let $\hat{\mathcal{G}}$ (resp. $\hat{\mathcal{B}}$) be a given finite-dimensional space for the approximation of $\mathcal{G}$ (resp. $\mathcal{B}$). Let $\eta_n = (\eta_{n,1}, \eta_{n,2})$ be a sequence of positive numbers. Set $\hat{V}_N := g \vee \varphi$. For $n = N-1, \ldots, 0$:

- compute feedback strategy and controls $(\hat{\alpha}_n, \hat{b}_n)$ according to

$$(\hat{\alpha}_n, \hat{b}_n) \in \eta_n - \arg\inf_{\alpha\in\hat{\mathcal{G}}} \sup_{b\in\hat{\mathcal{B}}} \mathbb{E}\Big[g(X_n) \bigvee \hat{V}_{n+1}(F(X_n, \alpha(X_n, b(X_n))), b(X_n))\Big] \tag{30}$$

(in a sense made precise below)

- set

$$\hat{V}_n(x) := g(x) \bigvee \hat{V}_{n+1}\Big(F(x, \hat{\alpha}_n(x, \hat{b}_n(x)), \hat{b}_n(x))\Big) \tag{31}$$

More precisely, let us denote $F^{\alpha[b],b}(x) := F(x, a(x, b(x)), b(x))$, then "$\eta_n - \arg\inf\sup$" in (30) means, by convention, for some $\eta_n = (\eta_{n,1}, \eta_{n,2})$, that

$$\begin{aligned} &\sup_{b\in\hat{\mathcal{B}}} \mathbb{E}\big[g(X_n) \vee \hat{V}_{n+1}\left(F^{\hat{\alpha}_n[b],b}(X_n)\right)\big] \\ &\qquad \le \inf_{\alpha\in\hat{\mathcal{G}}} \sup_{b\in\hat{\mathcal{B}}} \mathbb{E}\big[g(X_n) \vee \hat{V}_{n+1}\left(F^{\alpha[b],b}(X_n)\right)\big] + \eta_{n,1} \end{aligned} \tag{32a}$$

$$\begin{aligned} &\mathbb{E}\big[g(X_n) \vee \hat{V}_{n+1}\Big(F^{\hat{\alpha}_n[\hat{b}_n],\hat{b}_n}(X_n)\Big)\big] \\ &\qquad \ge \sup_{b\in\hat{\mathcal{B}}} \mathbb{E}\big[g(X_n) \vee \hat{V}_{n+1}\left(F^{\hat{\alpha}_n[b],b}(X_n)\right)\big] - \eta_{n,2}. \end{aligned} \tag{32b}$$

This allows for solving the min-max problem on $\hat{\mathcal{G}} \times \hat{\mathcal{B}}$ within some margin error.

In this algorithm, only the feedback strategies/controls $(\hat{\alpha}_k, \hat{b}_k)$ are stored ($\hat{V}_n$ is not stored). Each evaluation of the value $\hat{V}_{n+1}(x)$ uses the previous strategies $(\hat{\alpha}_{n+1}, \ldots, \hat{\alpha}_{N-1})$ and controls $(\hat{b}_{n+1}, \ldots, \hat{b}_{N-1})$ to compute the approximated characteristics, in a full Lagrangian philosophy.

5 Convergence analysis

In this section an error estimate for algorithm 1 is given. For algorithm 2, because we do not have $\hat{V}_0(x) \geq V_0(x)$ in general, we were not able to obtain an error estimate as in the setting of [19]. Here, we will only give error estimates for the difference $\mathbb{E}[\hat{V}_0(X)] - \mathbb{E}[V_0(X)]$, which we call a "weak" error estimate.

First we state an approximation result of the exact value by Lipschitz continuous strategies and feedback controls. From now on, for given constants $L, M \geq 0$, let us denote

$$\mathcal{G}_L := \{\alpha \in \mathcal{G},\ [\alpha] \leq L\}, \quad \mathcal{B}_M := \{b \in \mathcal{B},\ [b] \leq M\}.$$

In the following Lemmata we assume (H0)-(H4).

Lemma 5.1. (i) *Assume that the boundary of B is of null Lebesgue's measure. Let $\varepsilon_1 > 0$. Then there exists $L \geq 0$ and $\alpha^* \in (\mathcal{G}_L)^N$ such that*

$$\mathbb{E}[V_0(X)] \geq \sup_{b \in \mathcal{B}^N} \mathbb{E}\big[\tilde{J}_0(X, \alpha^*[b], b)\big] - \varepsilon_1. \tag{33}$$

(ii) *Let $\hat{\alpha}$ be a given strategy of $(\mathcal{G}_L)^N$. Let $\varepsilon_2 > 0$. Then there exists $M \geq 0$ and $b^* \in (\mathcal{B}_M)^N$ such that*

$$\mathbb{E}[V_0(X)] \leq \mathbb{E}\big[\tilde{J}_0(X, \hat{\alpha}[b^*], b^*)\big] + \varepsilon_2. \tag{34}$$

Notice that (33) can be written, in the same way, in the form of

$$\mathbb{E}[V_0(X)] \geq \inf_{\alpha \in (\mathcal{G}_L)^N} \sup_{b \in \mathcal{B}^N} \mathbb{E}\big[\tilde{J}_0(X, \alpha[b], b)\big] - \varepsilon_1,$$

which means an approximation of $\mathbb{E}[V_0(X)]$ by using Lipschitz continuous strategies.

Proof. (i) It is in principle possible to follow the regularization approach of HJB-Isaacs equation with Lipschitz continuous controls as in [14]. We can also obtain the approximation by a mollifying argument as follows. Let $(\alpha, b) \in \mathcal{G} \times \mathcal{B}$. We first consider $\alpha^\varepsilon := \rho_\varepsilon * \alpha_k$, a mollifying sequence for α_k ($\alpha_k \in \mathcal{M}(\mathbb{R}^d \times B, A)$ can be extended on $\mathbb{R}^d \times \mathbb{R}^{n_B}$ by $\alpha_k(x, b) = 0$ whenever $b \notin B$.) Therefore $\lim_{\varepsilon \to 0} \alpha_k^\varepsilon(x, b) = \alpha_k(x, b)$ a.e. on $\mathbb{R}^d \times int(B)$, for all k. By using the assumption that ∂B is of null Lebesgue's measure ($\lambda(\partial B) = 0$), we get the a.e. convergence on $\mathbb{R}^d \times B$. In view of the expression of $J_0(x, a, b)$, which is continuous in the variable a_{N-1}, $\tilde{J}_0(x, \alpha[b], b)$ will be continuous in its dependence on the variable α_{N-1}. Hence by Lebesgue's dominated convergence Theorem, we first have $\lim_{\varepsilon_{N-1} \to 0} \mathbb{E}[\tilde{J}_0(X, \alpha^\varepsilon[b], b)] = \mathbb{E}[\tilde{J}_0(X, \alpha[b], b)]$, where $\alpha^\varepsilon = (\alpha_0, \ldots, \alpha_{N-2}, \alpha_{N-1}^\varepsilon)$, i.e., we only regularize the α_{N-1} part. Then we can regularize in the same way α_{N-2} by some $\alpha_{N-2}^{\varepsilon_{N-2}}$ for ε_{N-2} small enough. We proceed by recursion until $\alpha_0^{\varepsilon_0}$. The regularized functions α_k^ε are then Lipschitz continuous on $\mathbb{R}^d \times B$. This gives the desired result.

(ii) we can proceed in the same way. $\square$

For given $\varepsilon_1, \varepsilon_2 > 0$, α^* and β^* are chosen as in the previous Lemma. Let $L := [\alpha^*]$ and $M := [b^*]$, so that $\alpha^* \in (\mathcal{G}_L)^N$ and $b^* \in (\mathcal{B}_M)^N$.

Lemma 5.2. *(i) Let α and $\bar\alpha$ be two elements of $(\mathcal{G}_L)^N$, and let $b \in B^N$. Let x, y be in $\mathbb{R}^d$ and assume that for all $0 \le k \le N$, $X^{\alpha[b],b}_{k,x} \in \Omega_N$, where Ω_N is some given subset of $\mathbb{R}^d$. Then it holds*

$$\max_{0\le k\le N} |X^{\bar\alpha[b],b}_{k,y} - X^{\alpha[b],b}_{k,x}| \le e^{C_{1,L}T}\left(|y-x| + C_2 T \max_{0\le k\le N-1} \|\bar\alpha_k - \alpha_k\|_{L^\infty(\Omega_N\times B)}\right)$$

where $C_{1,L} := [f]_1 + [f]_2 L$ and $C_2 := [f]_2$.
(ii) Let α be in $(\mathcal{G}_L)^N$, and let $b, \bar b$ be in $(\mathcal{B}_M)^N$ for some $M \ge 0$. Let x, y be in $\mathbb{R}^d$ and assume that for all $0 \le k \le N$, $X^{\alpha[b],b}_{k,x} \in \Omega_N$. Then it holds

$$\max_{0\le k\le N} |X^{\alpha[\bar b],\bar b}_{k,y} - X^{\alpha[b],b}_{k,x}| \le e^{C_{3,L,M}T}\left(|y-x| + C_{4,L} T \max_{0\le k\le N-1} \|\bar b_k - b_k\|_{L^\infty(\Omega_N)}\right)$$

with $C_{3,L,M} := [f]_1 + ([f]_2 L + [f]_3)M$ and $C_{4,L} := L[f]_2 + [f]_3$.

Proof. (i) This is a discrete Gronwall estimate. Let $x_k = X^{\alpha[b],b}_{k,x}$ and $y_k = X^{\bar\alpha[b],b}_{k,y}$ for $k \ge 0$. We have $x_{k+1} = x_k + \tau f(x_k, \alpha_k(x_k, b_k), b_k)$ and $y_{k+1} = y_k + \tau f(y_k, \bar\alpha_k(y_k, b_k), b_k)$. Therefore

$$\begin{aligned}&\big|y_{k+1} - x_{k+1}\big| \\ &\qquad \le |y_k - x_k|(1 + \tau([f]_1 + [f]_2[\alpha_k])) + \tau[f]_2|\bar\alpha_k(x_k, b_k) - \alpha_k(x_k, b_k)|.\end{aligned}$$

with $[f]_1 + [f]_2[\alpha_k] \le [f]_1 + [f]_2 L = C_1$. Then by induction:

$$|X^{\bar\alpha[b],b}_{n,y} - X^{\alpha[b],b}_{n,x}| \le (1 + C_1\tau)^n \left(|y-x| + C_2\tau \sum_{k=0,\dots,n-1} \Big|\bar\alpha_k(x_k, b_k) - \alpha_k(x_k, b_k)\Big|\right)$$

where $C_1 := [f]_1 + [f]_2 L$ and $C_2 := [f]_2$. Notice that $(1 + C_1\tau)^n \le e^{C_1 T}$ for $t_n = n\tau \le T$. The desired result follows.
(ii) Let $x_k = X^{\alpha[b],b}_{k,x}$ and $y_k = X^{\alpha[\bar b],\bar b}_{k,y}$ for $k \ge 0$, we have now

$$\begin{aligned}x_{k+1} &= x_k + \tau f(x_k, \alpha_k(x_k, b_k(x_k)), b_k(x_k)) \\ y_{k+1} &= y_k + \tau f(y_k, \alpha_k(y_k, \bar b_k(y_k)), \bar b_k(y_k)).\end{aligned}$$

Therefore

$$\begin{aligned}&\big|y_{k+1} - x_{k+1}\big| \\ &\qquad \le |y_k - x_k|(1 + \tau([f]_1 + [f]_2[\alpha_k])[\bar b_k]) + \tau([f]_2[\alpha_k] + [f]_3)|\bar b_k(x_k) - b_k(x_k)|.\end{aligned}$$

with $[f]_1 + [f]_2[\alpha_k][\bar b_k] \le [f]_1 + [f]_2 LM =: C_{3,L,M}$ and $[f]_2[\alpha_k] + [f]_3 \le [f]_2 L + [f]_3 =: C_{4,L}$. We conclude then as in (i). □

Theorem 5.3 (error estimate). *Assume (H0)-(H3), and that X is a random variable with compact support denoted Ω_0 and such that $\mathbb{E}[|X|] < \infty$. Consider the strategy and controls obtained by algorithm 1, and assume that $\hat{a} \in (\hat{\mathcal{G}}_L)^N$ and $\hat{b} \in (\hat{\mathcal{B}}_M)^N$, where L, M are constants large enough in order that the estimates of Lemma 5.1 hold. Let α^* and β^* be such that* (33) *and* (34) *hold. Then there exists positive constants C_L and $C_{L,M}$ such that*

$$\mathbb{E}[\hat{V}_0(X)] - \mathbb{E}[V_0(X)] \leq C_L \max_{0\leq k\leq N-1} d_{L^\infty(\Omega_N\times B)}(\alpha_k^*, \hat{\mathcal{G}}_L) + \eta_1 + \varepsilon_1 \tag{35}$$

and

$$\mathbb{E}[\hat{V}_0(X)] - \mathbb{E}[V_0(X)] \geq -C_{L,M} \max_{0\leq k\leq N-1} d_{L^\infty(\Omega_N)}(b_k^*, \hat{\mathcal{B}}_M) - \eta_2 - \varepsilon_2 \tag{36}$$

where

$$\Omega_N := \{X_{k,x}^{a,b},\ x \in \Omega_0,\ 0 \leq k \leq N,\ (a,b) \in A^N \times B^N\}. \tag{37}$$

Remark 5.4. *Notice that by standard Gronwall estimates, if $\Omega_0 \subset B(0, r_0)$, and if we denote C, L two constants such that $\|f(x,a,b)\| \leq C + L\|x\|$ (i.e. $L = [f]_1$ and $C = \max_{A\times B} \|f(0,a,b)\|$), then $\Omega_N \subset B(0, r_N)$ where $r_N := e^{LT}(r_0 + CT)$.*

Proof of Theorem 5.3. We first consider the upper bound. By the scheme definition, using (29a) (i.e., the η_1-suboptimality of $\hat{\alpha}$), and by the definition of $V_0 = \tilde{V}_0$ and the ε_1-suboptimality of α^* with respect to $\tilde{V}_0$:

$$\begin{aligned}
\mathbb{E}[\hat{V}_0(X)] - \mathbb{E}[V_0(X)] &\leq \inf_{\alpha\in\hat{\mathcal{G}}^N} \sup_{b\in\hat{\mathcal{B}}^N} \mathbb{E}\Big[\tilde{J}_0(X, \alpha[b], b)\Big] + \eta_1 - \sup_{b\in\mathcal{B}^N} \mathbb{E}\Big[\tilde{J}_0(X, \alpha^*[b], b)\Big] + \varepsilon_1 \\
&\leq \inf_{\alpha\in\hat{\mathcal{G}}^N} \sup_{b\in\mathcal{B}^N} \mathbb{E}\Big[\tilde{J}_0(X, \alpha[b], b)\Big] - \sup_{b\in\mathcal{B}^N} \mathbb{E}\Big[\tilde{J}_0(X, \alpha^*[b], b)\Big] + \eta_1 + \varepsilon_1
\end{aligned}$$

where we have used the fact that $\hat{\mathcal{B}} \subset \mathcal{B}$ in the last inequality. Therefore we have

$$\begin{aligned}
\mathbb{E}[\hat{V}_0(X)] - \mathbb{E}[V_0(X)] &\leq \inf_{\alpha\in\hat{\mathcal{G}}^N} \sup_{b\in\mathcal{B}^N} \mathbb{E}\big[\tilde{J}_0(X, \alpha[b], b) - \tilde{J}_0(X, \alpha^*[b], b)\big] + \eta_1 + \varepsilon_1 \\
&\leq \inf_{\alpha\in\hat{\mathcal{G}}_L^N} \mathbb{E}\Big[\sup_{b\in B^N} (\tilde{J}_0(X, \alpha[b], b) - \tilde{J}_0(X, \alpha^*[b], b))\Big] + \eta_1 + \varepsilon_1
\end{aligned}$$

where we have furthermore restricted the space $\hat{\mathcal{G}}^N$ to $\hat{\mathcal{G}}_L^N$. Recall that $\tilde{J}_0(x, \alpha[b], b) = \max_{0\leq k\leq N-1} g(x_k) \vee \varphi(x_N)$ where $x_k = X_{k,x}^{\alpha[b],b}$, which also corresponds to $x_0 = x$ and $x_{k+1} = F(x_k, \alpha_k(x_k, b_k), b_k)$. Denoting $y_k = X_{k,x}^{\alpha^*[b],b}$, we have

$$\begin{aligned}
|\tilde{J}_0(x, \alpha^*[b], b) - \tilde{J}_0(x, \alpha[b], b)| &\leq \max_{0\leq k\leq N-1} |g(y_k) - g(x_k)| \vee |\varphi(y_N) - \varphi(x_N)| \\
&\leq \max([g], [\varphi]) \max_{0\leq k\leq N} |y_k - x_k|.
\end{aligned}$$

By using the estimate of Lemma 5.2(i), we obtain

$$\max_{0\le k\le N} |y_k - x_k| \le e^{C_1 T} C_2 T \max_{0\le k\le N-1} \|\alpha_k - \alpha_k^*\|_{L^\infty(\Omega_N\times B)}$$

and therefore with $C_L := \max([g],[\varphi]) C_2 T e^{C_1 T}$ (which only depends of L and of the data)

$$\begin{aligned}\mathbb{E}[\hat V_0(X)] - \mathbb{E}[V_0(X)] \\ \le C_L \mathbb{E}\big[1_{\Omega_N}\big] \inf_{\alpha\in\hat{\mathcal{G}}_L^N} \max_{0\le k\le N-1} \|\alpha_k - \bar\alpha_k\|_{L^\infty(\Omega_N\times B)} + \eta_1 + \varepsilon_1. \qquad (38)\end{aligned}$$

Hence by using the fact that $\mathbb{E}\big[1_{\Omega_N}\big] \le 1$, we conclude to

$$\mathbb{E}[\hat V_0(X)] - \mathbb{E}[V_0(X)] \quad \le C_L \bigg(\max_{0\le k\le N-1} \inf_{\alpha\in\hat{\mathcal{G}}_L} \|\alpha_k - \alpha_k^*\|_{L^\infty(\Omega_N\times B)} \bigg) + \eta_1 + \varepsilon_1.$$

Since $\inf_{\alpha\in\hat{\mathcal{G}}_L} \|\alpha - \alpha_k^*\|_{L^\infty(\Omega_N\times B)} = d(\alpha_k^*, \hat{\mathcal{G}}_L)$, this concludes the upper bound estimate.

For the lower bound, by the scheme definition, using (29b) (i.e., the η_2-suboptimality of $\hat\alpha, \hat b$), and by using Lemma 5.1 and the ε_2-suboptimality of some $b^* \in (\mathcal{B}_M)^N$ (for some $M \ge 0$) with respect to $\tilde V_0$ as in (34), we obtain

$$\mathbb{E}[\hat V_0(X)] - \mathbb{E}[V_0(X)] \quad \ge \quad \sup_{b\in\hat{\mathcal{B}}^N} \mathbb{E}\Big[\tilde J_0(X, \hat\alpha[b], b)\Big] - \eta_2 - \bigg(\mathbb{E}\Big[\tilde J_0(X, \hat\alpha[b^*], b^*)\Big] - \varepsilon_2\bigg).$$

Hence, using also the fact that $(\hat{\mathcal{B}}_M)^N \subset \hat{\mathcal{B}}^N$,

$$\mathbb{E}[\hat V_0(X)] - \mathbb{E}[V_0(X)] \quad \ge \quad - \inf_{b\in(\hat{\mathcal{B}}_M)^N} \mathbb{E}\Big[\big|\tilde J_0(X, \hat\alpha[b^*], b^*) - \tilde J_0(X, \hat\alpha[b], b)\big|\Big] - \eta_2 - \varepsilon_2.$$

By using the estimate of Lemma 5.2(ii), we obtain

$$|\tilde J_0(x, \hat\alpha[b^*], b^*) - \tilde J_0(x, \hat\alpha[b], b)| \le \max([g],[\varphi]) C_{4,L} T e^{C_{3,L,M} T} \max_{0\le k\le N-1} \|b_k^* - b_k\|_{L^\infty(\Omega_N)}.$$

We conclude to the desired estimate, with $C_{L,M} := \max([g],[\varphi]) C_{4,L} T e^{C_{3,L,M} T}$. □

From the previous estimates, we deduce the following convergence result.

Theorem 5.5 (**convergence**). *Assume (H0)-(H3), and that X is a random variable with compact support and $\mathbb{E}[|X|] < \infty$. Let $N \ge 1$ be fixed. Let Θ be the set of parameters of the approximation spaces $\hat{\mathcal{G}}_L, \hat{\mathcal{B}}_M$. Let us assume that for all compact K, and fixed L, M,*

$$\forall \alpha \in \mathcal{G}_L, \quad \lim_{\Theta\to\infty} d_{L^\infty(K\times B)}(\alpha, \hat{\mathcal{G}}_L) = 0,$$

and also, in the same way,

$$\forall b \in \mathcal{B}_M, \quad \lim_{\Theta \to \infty} d_{L^\infty(K)}(b, \hat{\mathcal{B}}_M) = 0.$$

Then for any $\varepsilon > 0$, there exists constants L, M large enough, constants $(\eta_i, \varepsilon_i)_{i=1,2}$ small enough, such that, for Θ large enough, algorithm 1 gives

$$\left|\mathbb{E}[\hat{V}_0(X) - V_0(X)]\right| \leq \varepsilon. \tag{39}$$

In conclusion, in the above weak sense, we can say that algorithm 1 converges to the value $\mathbb{E}[V_0(X)]$.

6 Numerical examples

Feedforward neural networks. In our approximations we use Feedforward neural networks for the approximation of the feedback control strategies and for the adversarial controls. We denote by

$$\mathcal{L}^{\rho}_{d_1,d_2} = \left\{\phi : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2} : \exists\ (\mathcal{W}, \beta) \in \mathbb{R}^{d_2 \times d_1} \times \mathbb{R}^{d_2},\ \phi(x)\ =\ \rho(\mathcal{W}x + \beta)\ \right\}$$

the set of layer functions with input dimension d_1, output dimension d_2, and activation function $\rho : \mathbb{R}^{d_2} \to \mathbb{R}^{d_2}$. The operator $x \in \mathbb{R}^{d_1} \mapsto \mathcal{W}x + \beta \in \mathbb{R}^{d_2}$ is an affine mapping with $\mathcal{W}$ a matrix called weight, and β a vector called bias. The activation is applied component-wise i.e., $\rho(x_1, \ldots, x_{d_2}) = \big(\hat{\rho}(x_1), \ldots, \hat{\rho}(x_{d_2})\big)$ with $\hat{\rho} : \mathbb{R} \mapsto \mathbb{R}$ non decreasing. When ρ is the identity function, we simply write $\mathcal{L}_{d_1,d_2}$ and when $\rho(x) = \max(x, 0)$ we write $\mathcal{L}^{ReLu}_{d_1,d_2}$.

Each control with values in $\mathbb{R}^{d_1}$ is approximated in the space of neural networks with L hidden layers of m neurons using the ReLu activation function for internal activation:

$$\mathcal{N}_{d_0,d_1,L,m} = \Big\{\varphi : \mathbb{R}^{d_0} \to \mathbb{R}^{d_1} : \exists \phi_0 \in \mathcal{L}^{ReLu}_{d_0,m},\ \exists \phi_i \in \mathcal{L}^{ReLu}_{m,m}, i = 1, \ldots, L-1,$$
$$\exists \phi_L \in \mathcal{L}_{m,d_1}, \varphi\ =\ \xi\ \circ\ \phi_L \circ \phi_{L-1} \circ \cdots \circ \phi_0\Big\}$$

and $\xi : \mathbb{R}^{d_1} \to \mathbb{R}^{d_1}$ is a final activation function. When for instance $a = a(x, b)$ and $b = b(x)$ then a is approximated in $\mathcal{N}_{d+n_B,n_A,L,m}$ and b in $\mathcal{N}_{d,n_B,L,m}$ assuming $A \subset \mathbb{R}^{n_A}$ and $B \subset \mathbb{R}^{n_B}$. Depending on the test case, since A and B are compact sets, a final activation function ξ_A mapping $\mathbb{R}^{n_A}$ to A or ξ_B mapping $\mathbb{R}^{n_B}$ to B is used. In particular for $A = B = [-1, 1]$ (as in the first three examples), the function $\xi_A(x) = \xi_B(x) = \tanh(x)$ for $x \in \mathbb{R}$ is used. In the last example, $A = B = B_2(0, 1)$ is the unit ball of $\mathbb{R}^2$ for the Euclidean norm, then we will used $\xi_A(x) = \xi_B(x) = \frac{x}{\|x\|_2} \tanh(\|x\|_2)$ for $x \in \mathbb{R}^2$ (following [19]). The previous set

$\mathcal{N}_{d_0,d_1,L,m}$ is parametrized by $\theta = (\mathcal{W}_0, \beta_0, \dots, \mathcal{W}_L, \beta_L)$ defining the layer functions; a function ϕ in this set is denoted $\phi^\theta(.)$ to emphasize on the θ dependency.

Min-Max optimization. Our typical problem is to deal with

$$\min_{\theta_A} \max_{\theta_B} \mathbb{E}\Big[J(Z, A^{\theta_A}(Z), B^{\theta_B}(Z))\Big]$$

where Z is a random variable in $\mathbb{R}^d$ over a set of parameters θ_A and θ_B where $A^{\theta_A} \in \mathcal{N}_{d,n_A,L,m}$, $B^{\theta_B} \in \mathcal{N}_{d,n_B,L,m}$ and for a given cost functonal J. Stochastic gradient methods are a classicaly used to deal with general (possibly non convex/non concave) problems of the form

$$\min_x \max_y \mathbb{E}\big[Q(x, y, Z)\big]$$

where Z is a random variable. At each step i of a stochastic gradient algorithm, we consider $N_{batch} \in \mathbb{N}^*$ and i.i.d. $z^i = (z^i_q)_{1 \le q \le N_{batch}}$ with same law as Z ($z^i_q \sim Z$), and

$$f(x, y, (z^i_q)) := \frac{1}{N_{batch}} \sum_{q=1}^{N_{batch}} Q(x, y, z^i_q).$$

Most algorithms such as SGDA [47], AGDA [47], γ-GDA of [44] extended to the stochastic case are designed for convex-concave problems and may behave badly on our problems. Since we deal with general functional we consider here *Potential reduction algorithms* [51], [44] extended to the stochastic case. The following iteration, with $(\eta_i)_{i\ge 0}$ a sequence of positive step size where $\eta_0 = \eta$, gives the general outline of an iterative resolution algorithm:

$$\begin{aligned} y^{i+1} =& \operatorname*{argmax}_y f(x^i, y, z^i) \\ x^{i+1} =& x^i - \eta_i \nabla_x f(x^i, y^{i+1}, z^i). \end{aligned}$$

In order to get a feasible algorithm, we introduce integers $M_{epoch}, M^{pote}_{epoch} \ge 1$ and positive sequences $(\eta_i)_{i\ge 0}$, $(\rho_{i,k})_{i,k\ge 0}$ with $\eta_0 := \eta$ and $\rho_{i,0} := \rho$. The resulting algorithm is as follows.

Potential reduction algorithm for $\min_x \max_y \mathbb{E}[Q(x, y, Z)]$:

In the case of a $\min_{a[\cdot]} \max_b Q(a[b], b)$ problem, M^{pote}_{epoch} is therefore the number of internal iterations optimizing the parameters of b, and M_{epoch} is the number of external iterations. In practice, each optimization step is achieved using the ADAM optimizer [45] using an adaptive learning rates derived from estimates of first and second moments of the gradients.

In all examples the computational domain is a parallelipedic box $\Omega \subset \mathbb{R}^d$ (depending on the example), Z is the uniform random variable on Ω and therefore the batch points are drawn uniformly in Ω.

Start with randomly chosen set of parameters x^0, y^0.
for $i = 0, \dots M_{epoch} - 1$ **do**
  $y \leftarrow y^i$; $\rho_{i,0} \leftarrow \rho$
  for $k = 0, \dots M_{epoch}^{pote} - 1$ **do**
    $y \leftarrow y + \rho_{i,k} \nabla_y f(x^i, y, (w_q^{i,k}))$ with $w_q^{i,k} \sim Z$
  $y^{i+1} \leftarrow y$
  $x^{i+1} \leftarrow x^i - \eta_i \nabla_x f(x^i, y, (z_q^i))$ with $z_q^i \sim Z$

Multi-step approximations. Finally, for the approximation of the dynamics we consider a multi-step approximation F instead of the Euler scheme (4). Let $p \geq 1$ be a given integer (the number of intermediate sub-steps), for given $x \in \mathbb{R}^d$ and controls $(a, b) \in A \times B$, we first define a Runge Kutta step by

$$F_h(x, a, b) := x + \frac{h}{2}(f(x, a, b) + f(x + hf(x, a, b), a, b)), \quad \text{with } h := \tfrac{\tau}{p}$$

(here corresponds to the "Heun" scheme). Then we define

$$y = F(x, a, b)$$

by $y = y_p$ where

$$y_0 := x \ \text{ and } \ y_{k+1} = F_h(y_k, a, b),\ k = 0, \cdots, p-1.$$

We then consider the following approximation $V_{0,p}(x)$ of the continuous value $v_0(x)$:

$$V_{0,p}(x) = \inf_{\alpha \in S_N} \sup_{b \in B^N} \tilde{J}_{0,p}(x, \alpha[b], b) \tag{40}$$

where

$$\tilde{J}_{0,p}(x, \alpha[b], b) := \left(\max_{0 \leq k \leq N-1} G(\tilde{x}_k, \alpha_k(\tilde{x}_k, b_k(\tilde{x}_k)), b_k(\tilde{x}_k)) \right) \bigvee \varphi(\tilde{x}_N), \tag{41}$$

with $\tilde{x}_0 := x$ and $\tilde{x}_{k+1} := F(\tilde{x}_k, \alpha_k(\tilde{x}_k, b_k(\tilde{x}_k)), b_k(\tilde{x}_k))$ for $k = 0, \dots, N-1$, and

$$G(x, a, b) := \max_{0 \leq j < p} g(Y_{j,x}^{a,b})$$

with $Y_{0,x}^{a,b} := x$ and $Y_{j+1,x}^{a,b} := F_h(Y_{j,x}^{a,b}, a, b)$ for $j = 0, \dots, p-1$ (for given $(a, b) \in A \times B$), which amounts to taking the maximum of $g(.)$ along the intermediate substeps of the trajectory. Hereafter we consider $p = 5$ in all the numerical examples, and the value $V_{0,p}(x)$ (resp. $\tilde{J}_{0,p}$) will be still denoted $V_0(x)$ (resp. $\tilde{J}_0$).

One can show that the general statements (representation formula, error estimates, convergence results) remain valid for this multi-step approximation, see e.g. [19] in the one-player context.

All numerical tests are performed using Python 3.10 and Tensorflow, on a Dell xps13-9320 under linux ubuntu 13th Gen. Intel® Core™ i7-1360P with 12 cores (up to 5 GHz), 32 GiB RAM.

The following numerical examples correspond to computing backward reachable sets for two player differential games. Hence we can focus on zero-level sets of the value in order to visualise the performance of the algorithms.

Example 1. We consider the control problem with control sets $A = B = [-1, 1]$ and the following dynamics, for $x \in \mathbb{R}^2$ and $(a, b) \in A \times B$:

$$f(x, a, b) = aRx + cb\frac{x}{\|x\|}$$

with constant $c = 0.3$, $R = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ (rotation matrix), $\|.\|_2$ is the Euclidean norm. We fix $T = 0.6\pi$. The terminal cost $\varphi(.)$ is made precise in appendix C. It is designed in order that the solution $v_0(x)$ can be solved analytically. The target set, defined as $\{x, \varphi(x) \leq 0\}$, is also represented in Fig. 1 (left).

We first consider for $v_0(x)$ the general definition (1) with a negative obstacle term such as $g(x) := -1$. Since $\min \varphi > -1$ in this example, this amounts to

$$v_0(x) = \inf_{\alpha \in \mathcal{G}} \sup_{b \in \mathcal{B}_T} \varphi(y_{0,x}^{\alpha[b],b}(T)).$$

The functional to be optimized is chosen as in (26) of Thereom 3.6, that is

$$\inf_{\alpha \in \mathcal{G}^N} \sup_{b \in \mathcal{B}^N} \mathbb{E}\big[\tilde{J}_0(X, \alpha[b], b)\big] \tag{42}$$

with functional cost $\tilde{J}_0$ defined as in (41) and we apply algorithm 1 to deal with the inf sup.

The neural network space for A_θ and B_θ are $\mathcal{N}_{2+1,1,3,20}$ and $\mathcal{N}_{2,1,3,20}$ (3 layers with 30 neurons each), and the function $\xi = \tanh$ is used for both outputs. The following numerical parameters are used $M_{epoch} = 500$, $M_{epoch}^{pote} = 5$, $N_{batch} = 1000$ (which means 2.5×10^6 evaluation of the functional cost), with initial learning rates $\rho = \eta = 2 \times 10^{-3}$. Typical CPU time is about 56 sec. for the obstacle case.

Results are given in Figure 1. Here we plug in the computed optimal feedback controls α (and computed optimal adverse controls b) in order to draw pictures from the estimate $V_0(x) \simeq \tilde{J}_0(x, \alpha[b], b)$. We observe in both cases a very good numerical convergence of the global scheme towards the reference solution.

Secondly, using the same data we now consider an other similar problem with an obstacle function given by

$$g(x) := \min(\bar{\varepsilon}, \max(-\bar{\varepsilon}, r_B - \|x - q\|_2), \quad \text{with } \bar{\varepsilon} := 0.2$$

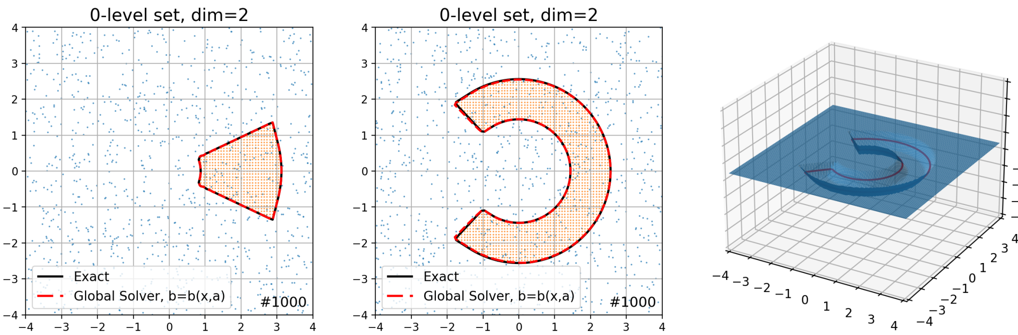

Figure 1: (Example 1, no obstacle) Results obtained with the "global scheme": 0-level set of the terminal data (left), 0-level set of the numerical solution $V_0(x)$ (center) and corresponding surface plot (right). Dimension $d = 2$, using $N = 4$ time steps.

with center $q = (0.5, 1.5)$ and radius $r_B := 0.5$, so that $\{x,\ g(x) \leq 0\}$ corresponds to the disk $B(q, r_B)$. We do not have anymore an analytical solution for this obstacle case. Instead, we use a high-order finite difference scheme in order to compute a reference solution (WENO3-TVDRK3 scheme of Jiang and Peng [43], using a Cartesian mesh of size 401^2). Results with the obstacle term are given in Figure 2, and we still observe a very good numerical convergence using the same parameters as before.

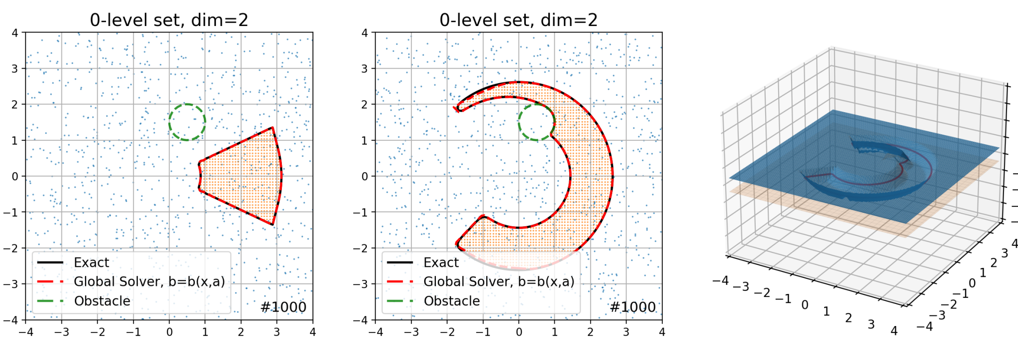

Figure 2: (Example 1, with obstacle) Results obtained with the "global scheme". Left: 0-level set of the terminal data (in red) and of the obstacle function (dotted green line); center: 0-level set of the numerical solution $V_0(x)$: right: corresponding surface plot. Dimension $d = 2$, using $N = 4$ time steps.

Finally, we also present results obtained with the time-marching algorithm 2 ("local scheme") in Figure 3, where the value V_k is estimated for different time index $k \in \{4, 3, 2, 1, 0\}$, with $k = 4$ corresponding to $t_4 = T$ (the target data), and for $k = 0$, $t_0 = 0$ corresponding to the approximation of $V_0(.)$ by the algorithm. The neural network space for both A_θ and B_θ is now $\mathcal{N}_{2,1,3,20}$, the other parameters being otherwise unchanged. We observe quite similar results, although the number of iterations needed might be greater than in algorithm 1 in order to obtain good results. Indeed, each time step requires a full optimization procedure, whereas algorithm 1 needs only one optimization problem to be solved.

For the other forthcoming Examples 2 to 4, our findings is that similar behavior holds, and algorithm 1 leads to similar results with a reduced CPU time cost than with algorithm 2. Therefore we will not report the results with algorithm 2 in detail.

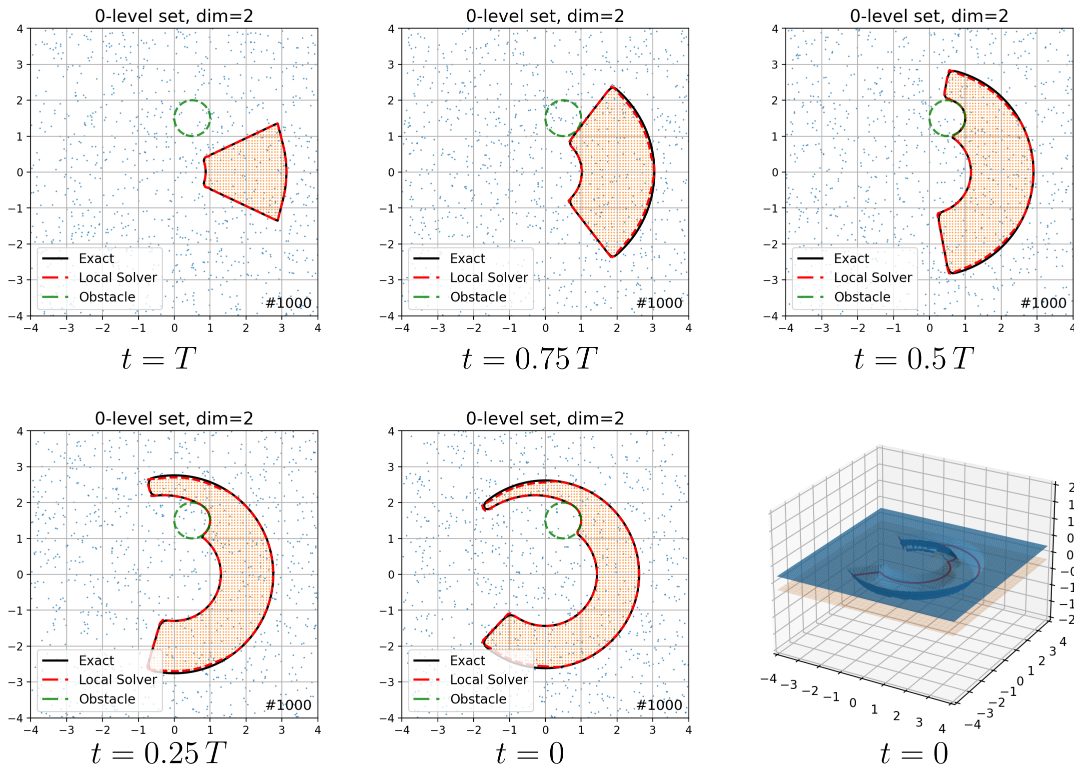

Figure 3: (Example 1, with obtacle) Results obtained with the algorithm 2 ("local scheme"). 0-level sets of $v(t, \cdot)$ at different times $t = \frac{k}{N}T$, $k = 0, \ldots, N$. Dimension $d = 2$, using $N = 4$ time steps.

Example 2. We consider the control problem with control sets $A = B = [-1, 1]$ and the following dynamics, for $x \in \mathbb{R}^2$ and $(a, b) \in A \times B$:

$$f(x, a, b) = \begin{pmatrix} 2\max(-1, \min(1, a - 2b)) \\ a + b \end{pmatrix}$$

(we use again $\xi = \tanh$), and the terminal time and value are given by $T = 0.4$ and

$$\varphi(x) = \min(0.5, \max(-0.5, \|x\|_\infty - 1)).$$

In this case we found an analytical formula for the value (see appendix C). In this exemple it happens that both $\min\max$ and $\max\min$ formulations are equivalent (i.e., the game *has a value*) although this is not immediate when looking at the dynamics.

We have tested different number of time steps $N = 2, 4, 8, 16$, with same optimization problem (42) as in Example 1. The numerical parameters are otherwise as follows: $A_\theta \in \mathcal{N}_{2+1,1,3,20}$ and $B_\theta \in \mathcal{N}_{2,1,3,20}$ (3 layers of 30 neurons each), with $\xi_A = \xi_B = \tanh$ for the output, and $M_{epoch} = 500$, $M_{epoch}^{pote} = 5$, $N_{batch} = 1000$ (hence an overall of 2.5×10^6 evaluations of the functional cost), with initial learning rates $\rho = \eta = 2 \times 10^{-3}$, computational domain $\Omega := [-3, 3]^2$.

Results are given in Figure 4 for $N = 2, 4, 8$. We also give an error table 1 for $N = 2, 4, 8, 16$. For the error we have chosen a *local relative error* around the 0-level set of the value: for a given treshold parameter $\eta > 0$, for a given cartesian mesh grid $(x_i) \in \Omega$ of 101^2 points, we set

$$e_{L^1,loc} := \frac{\sum_{i,\, x_i \in \Omega_\eta} |V_0(x_i) - v_0(x_i)|}{\sum_{i,\, x_i \in \Omega_\eta} 1}$$

where $\Omega_\eta := \{x \in \Omega^2,\ |v_0(x)| \leq \eta\}$. Here the values of v_0 lay in $[-0.5, 0.5]$ and we have set $\eta = 0.2$. We observe roughly a convergence of order between 0.5 and 1 with respect to $\tau = \frac{T}{N}$. (Results with a global relative L^1 error are similar on this example.)

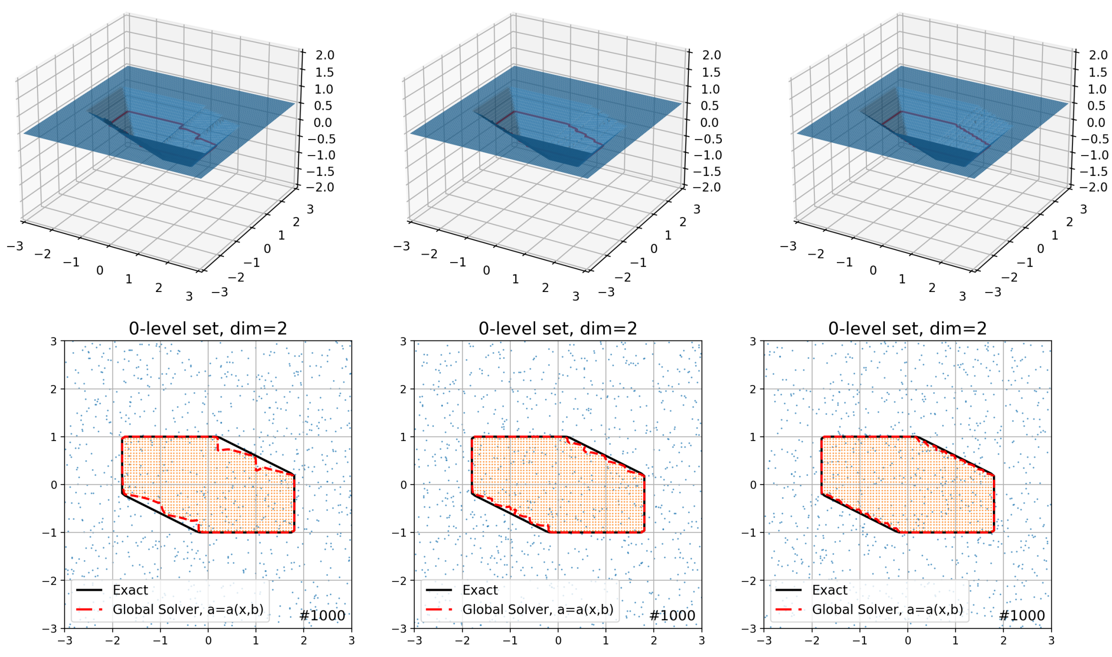

Figure 4: (Example 2) Results obtained with the global scheme Dimension $d = 2$, using $N = 2, 4$ and 8 time steps (left, middle and right).

N	CPU time (s)	$e_{L^1,loc}$	order
2	83.	2.95e-02	-
4	155.	1.79e-02	0.72
8	311.	1.27e-02	0.50
16	809.	7.49e-03	0.76

Table 1: (Example 2) Error table with respect to time discretisation parameter N

Example 3. In this example the dynamics is given by

$$f(x,a,b) = \begin{pmatrix} 2(1-|a-b|) \\ a+b \end{pmatrix}$$

with same terminal value φ as in the previous example.

In a first case we consider the usual value

$$V_0^-(x) := \inf_{\alpha\in\Gamma^N} \sup_{b\in B^N} J_0(x,\alpha[b],b).$$

In a second case we consider also the $\sup_\alpha \inf_b$ problem, corresponding to

$$V_0^+(x) := \sup_{\alpha\in\Gamma^N} \inf_{b\in B^N} J_0(x,\alpha[b],b).$$

Remark 6.1. *We have*

$$V_0^-(x) \le V_0^+(x) \tag{43}$$

and hence $\{x,\ V_0^+(x)\le 0\} \subset \{x,\ V_0^-(x)\le 0\}$*: the negative region of* V_0^+ *must be included in the one of* V_0^-*. Indeed, we already know that*

$$V_0^-(x) = \sup_{b_0}\inf_{a_0}\sup_{b_1}\inf_{a_1}\cdots J_0(x,a,b).$$

In the same way, we have

$$V_0^+(x) = \inf_{b_0}\sup_{a_0}\inf_{b_1}\sup_{a_1}\cdots J_0(x,a,b).$$

By using the symmetry of f $(f(x,a,b)=f(x,b,a))$ *and the fact that* $A=B$*, we have also* $J_0(x,a,b)=J_0(x,b,a)$ *and*

$$V_0^+(x) = \inf_{a_0}\sup_{b_0}\inf_{a_1}\sup_{b_1}\cdots J_0(x,a,b).$$

By using the inequality $\inf_a\sup_b \ge \sup_b\inf_a$ *we get the desired inequality.*

For the numerical tests, we use the parameter $T = 0.4$ and $N = 4$ time steps. Neural network spaces are as in the previous example: $A_\theta \in \mathcal{N}_{2+1,1,3,40}$ and $B_\theta \in \mathcal{N}_{2,1,3,40}$ (3 layers of 40 neurons each), with $\xi = \tanh$ for the output for both controls values. The numerical parameters for SG are otherwise as follows: $N_{batch} = 8000$ batch points, $M_{epoch} = 3000$ and $M^{pote}_{epoch} = 10$ internal iterations, and initial learning rates $\rho = \eta = 10^{-3}$, computational domain $\Omega := [-3, 3]^2$.

Results are given in Figure 5, together with the exact solutions which are given in appendix C. CPU time is about 960 sec for each example.

In this case we remark that the min and max do not commute, in the sense that in general

$$\min_a \max_b f(x, a, b) \cdot p \neq \max_b \min_a f(x, a, b) \cdot p.$$

We found this example numerically more difficult, with important oscillations in the stochastic gradient (SG) algorithm, compared to the previous examples, and the need of a finer sampling and greater number of SG iterations in order to get relevant results.

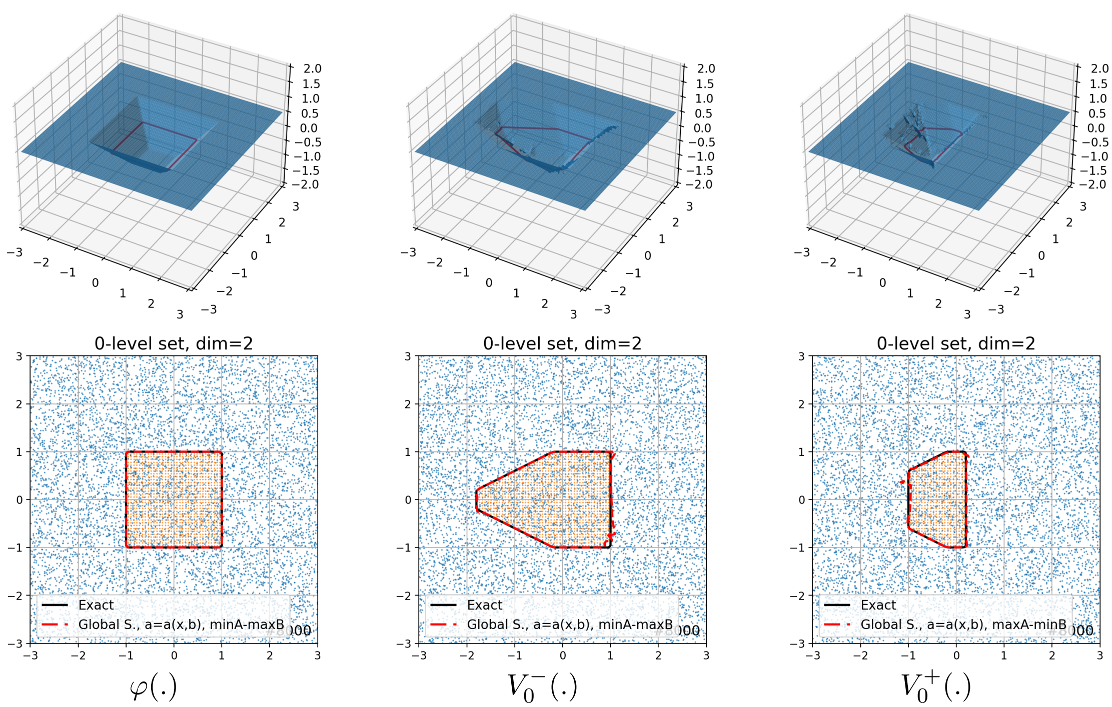

Figure 5: (Example 3, $d = 2$) Results obtained with the global scheme using $N = 4$ time iterations, left: terminal data, middle: V_0^-, right: V_0^+ (see text).

Example 4. We consider now a two-player game, where the first player is $X_1 = (x_1, x_2)$ with dynamics given by

$$\begin{cases} \dot{x}_1(t) = V_1 a_1(t) \\ \dot{x}_2(t) = V_1 a_2(t) \end{cases}$$

with a control $a(t) = (a_1(t), a_2(t)) \in A := B_2(0,1)$ which is the "direction" of $\dot{X}_1$, where $B_2(0,1)$ is the unit ball of $\mathbb{R}^2$ for the Euclidean norm, and the second player is modelized similarly by $X_2 = (x_3, x_4)$ with dynamics

$$\begin{cases} \dot{x}_3(t) = V_2 b_1(t) \\ \dot{x}_4(t) = V_2 b_2(t) \end{cases}$$

with an adverse control $b(t) = (b_1(t), b_2(t)) \in B := B_2(0,1)$. The horizon parameter and velocities are as follows

$$T = 4,\ V_1 = 1,\ V_2 = 0.7.$$

In this example, $x \in \mathbb{R}^4$ and the global dynamics is therefore given by

$$f(x,a,b) := (V_1 a_1,\ V_1 a_2,\ V_2 b_1,\ V_2 b_2).$$

This example is more complex than the previous examples in the sense that we will not have an analytic solution and it is higher dimensional. However a reference solution can still be obtained by using a classical full grid approach with a finite difference scheme (we use here a WENO3-RK3 finite difference scheme as described in Jiang and Peng [43], with a uniform grid of 51^4 points in space).

The first player, starting at a given point $(x_1, x_2) \in \mathbb{R}^2$ aims to reach a target $\mathcal{C} = B_2(x_A, r_A)$ (the Euclidian ball centered at x_A and of radius r_A), at some time τ before T, keeping away from the the second player starting at $(x_3, x_4) \in \mathbb{R}^2$ (i.e. $\|X_1(t) - X_2(t)\|_2 \geq R_0$, for a given threshold $R_0 > 0$) for all $t \in [0, \tau]$ whatever the adverse control can be, staying also in a given domain $X_1 \in \mathcal{K}_1 = \mathbb{R}^4 \backslash int(\mathcal{O})$ of $\mathbb{R}^2$, where $\mathcal{O}$ is a square obstacle which defined by

$$\mathcal{O} := B_\infty(x_B, r_B) = \{x \in \mathbb{R}^2,\ \|x - x_B\|_\infty \leq r_B\}$$

(so we require that $X_1(t) \in \mathcal{K}_1$ for all $t \in [0, \tau]$).

Following [16], the problem is a reachability problem with state constraints, which can be reformulated with level sets as follows. We consider the value $v_0(x)$ as in (1), which we recall

$$v_0(x) = \inf_{\alpha[.] \in \Gamma_{(0,T)}} \sup_{b \in \mathcal{B}_T} \max\left(\varphi\big(y_{0,x}^{\alpha[b],b}(T)\big),\ \max_{\theta \in (0,T)} g\big(y_{0,x}^{\alpha[b],b}(\theta)\big)\right)$$

where the obstacle function g and the terminal cost φ are now defined. For $x = (x_1, x_2, x_3, x_4)$, let the terminal cost be defined by

$$\varphi(x) := \|(x_1, x_2) - x_A\|_2 - r_A$$

(so that $\varphi(x) \le 0 \Leftrightarrow (x_1, x_2) \in B_2(x_A, r_A)$). For the obstacle part, let

$$g(x) := \max(g_0((x_1, x_2)), g_a(x))$$

where $g_0(.)$ is a level set function to encode the obstacle $B_\infty(x_B, r_B)$:

$$g_0(x) := r_B - \|(x_1, x_2) - x_B\|_\infty$$

so that $g_0(x) \le 0 \Leftrightarrow x \in \mathbb{R}^2 \backslash B_\infty(x_B, r_B)$. Finally $g_a(.)$ is defined by

$$g_a(x) := R_0 - \|(x_1, x_2) - (x_3, x_4)\|_2, \quad \text{with } R_0 := 1,$$

so that $g_a(x) \le 0 \Leftrightarrow \|(x_1, x_2) - (x_3, x_4)\|_2 \ge R_0$ (i.e., $g_a(x)$ is a level set function to encode the avoidance of X_1 with X_2).

For the numerical computations we use also the parameters

$$x_A = (3, 0),\ r_A = 1 \qquad \text{and} \qquad x_B = (0.5, 1.5),\ r_B = 0.75.$$

The whole computational domain is $\Omega := [-5, 5]^4$. Plots of the terminal data $\varphi(.)$ and of the obstacle function $g(.)$ are shown in Figure 6, with a cut in the plane $\mathbb{R}^2 \times \{(\overline{x}_3, \overline{x}_4)\}$ with $(\overline{x}_3, \overline{x}_4) = (0, -2)$, which will be the adverse starting position for showing all results.

The functional to be optimized is chosen as in (42), that is:

$$\inf_{\alpha \in \mathcal{G}^N} \sup_{b \in \mathcal{B}^N} \mathbb{E}\big[\tilde{J}_0(X, \alpha[b], b)\big]$$

(with functional cost $\tilde{J}_0$ as in (41)).

We consider the global solver algorithm 1. Computations are done with neural networks using 3 layers of 40 neurons each, ($A_\theta \in \mathcal{N}_{4+1,2,3,40}$ and $B_\theta \in \mathcal{N}_{4,2,3,40}$). which corresponds to $(N_a, N_b) = (3684, 3604)$ network parameters for functions (a_k, b_k) at each time step, the number of batch points is $N_{batch} = 50000$ and the number of SG iterations $M_{epoch} = 5000$ (after which small oscillations remain) $M^{pote}_{epoch} = 5$ internal iterations, with $\rho = \eta = 10^{-3}$. CPU time is around 34000 sec. for $N = 8$.

Results are given in Figure 7 (for resp. $N = 2$, 4 and 8 time steps, left side). The doted (orange) region corresponds to the points (x_1, x_2) or $\mathbb{R}^2$ such that $v(0, x) \le 0$, where $x = (x_1, x_2, \overline{x}_3, \overline{x}_4)$ and $(\overline{x}_3, \overline{x}_4)$ has a given value (e.g. $(-2, 0)$ in the graphics). This region represents the feasible region, from which it is safely possible to reach the target before capture. The left figure represents the target (corresponding to the negative region of the value at terminal time). About 10% of the batch points are also projected in the plane of visualization.

We have also tested a reversed formulation, i.e., based on

$$\sup_{b \in \mathcal{B}^N} \inf_{\alpha \in \mathcal{G}^N} \mathbb{E}\big[\tilde{J}_0(X, \alpha[b], b)\big] \tag{44}$$

and algorithm 1. Note that this value is the same as

$$\sup_{b\in\mathcal{B}^N}\inf_{a\in\mathcal{A}^N}\mathbb{E}\big[\tilde{J}_0(X,a(X),b(X))\big] \tag{45}$$

(in a same way as (19)), and it can be shown that it has the same limit value $v_0(x)$ as $N\to\infty$. We observe that this gives numerically more precise results for a lower computational cost. Results for this formulation are also presented in Figure 7 (center and right columns), for $N\in\{2,4,8\}$. Here the neural networks are composed of 3 layers of 20 neurons each, i.e., $A_\theta\in\mathcal{N}_{4+1,2,3,20}$ and $B_\theta\in\mathcal{N}_{4,2,3,20}$ (which corresponds respectively to $(N_a,N_b)=(1044,1004)$ network parameters for functions (a_k,b_k) at each time step), $N_{batch}=20000$, $M_{epoch}=1000$ (number of SG iterations), with $M_{epoch}^{pote}=10$ internal iterations, using initial learning rates $\rho=\eta=2\times10^{-3}$. We observe a numerical convergence after about $M_{epoch}=500$ iterations, after which small oscillations remain. CPU time is around 760 sec. for $N=4$ and 1600 sec. for $N=8$.

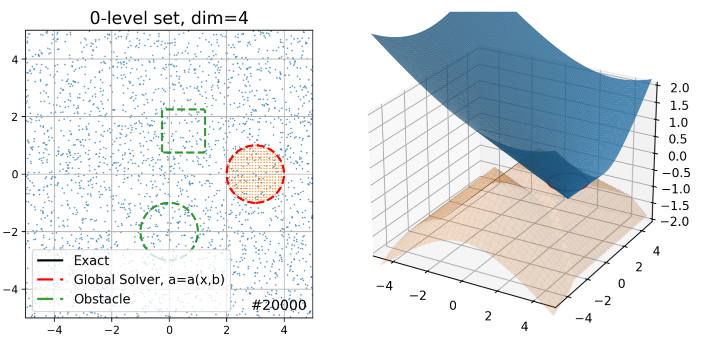

Figure 6: (Example 4, $d=4$) cut in plane $(\overline{x}_3,\overline{x}_4)=(0,-2)$ for the adverse starting position. Plots of the terminal data and obstacle functions: 0-level sets (left) and values (right).

Conclusion. We have demonstrated the relevance of our approximations through several numerical tests, complemented by a mathematical framework for convergence analysis. It is noteworthy that the quality of convergence may still depend on the method employed to address the min-max, a distinct matter not directly addressed in the present work. We aim to explore more complex problems using the methodology outlined in this study.

A Proof of Theorem 2.3

Let us begin with the first equality. To simplify, we consider the case $N=2$, the general case $N\geq1$ being similar.

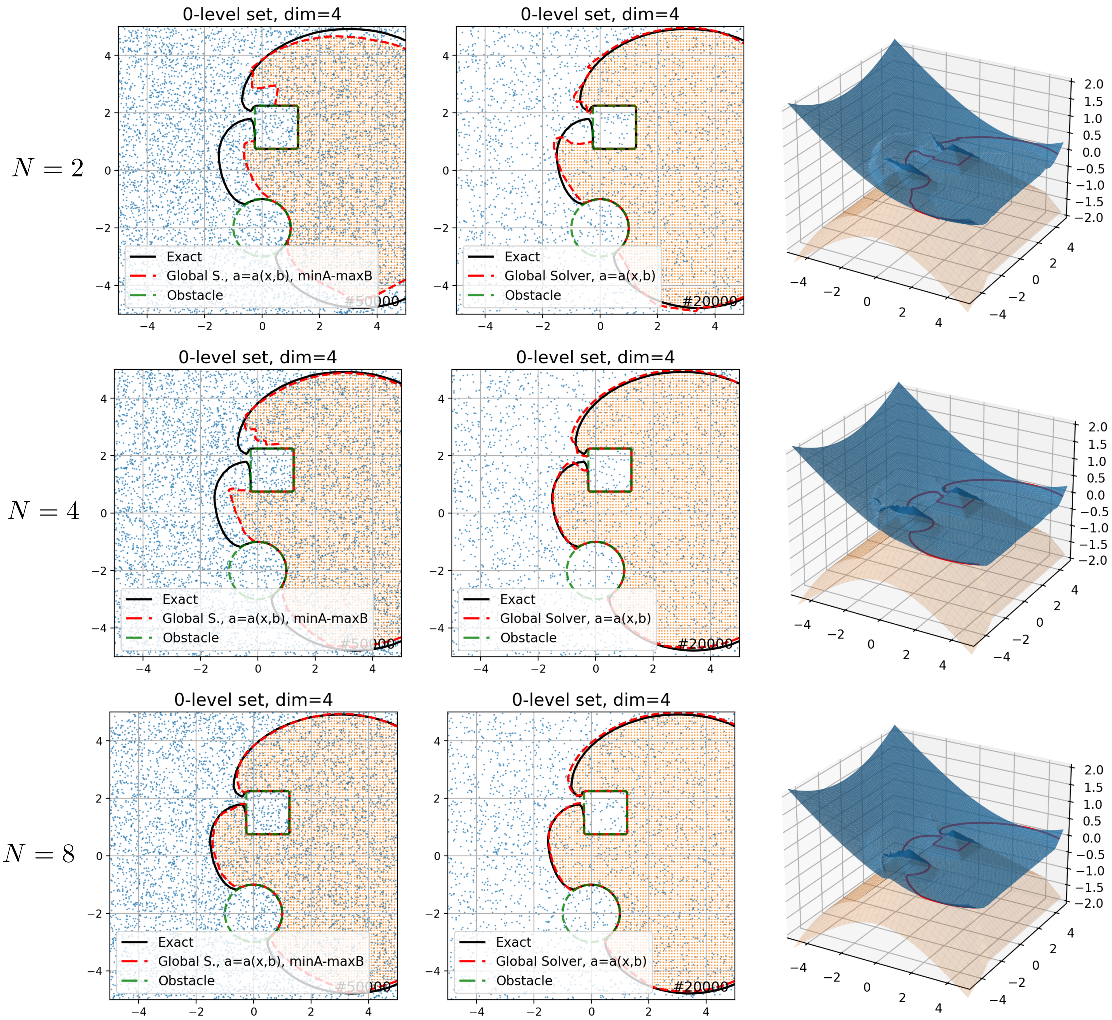

Figure 7: (Example 4, $d = 4$) cut in plane $(\overline{x}_3, \overline{x}_4) = (0, -2)$ for the adverse starting position. Left : "Global" scheme with $N = 2, 4, 8$ time iterations for approach $\inf_\alpha \sup_b$ (0-level sets). Center and right: "Global" scheme for approach $\sup_b \inf_\alpha$ and $N = 2, 4, 8$ (0-level sets and value).

In order to prove that $V_0(x) = \overline{V}_0(x)$, let us first show $V_0(x) \geq \overline{V}_0(x)$. By using the general fact that $\inf_p \sup_q Q(p,q) \geq \sup_q \inf_p Q(p,q)$, we have

$$\begin{aligned} V_0(x) &= \inf_{\alpha_0[.]} \inf_{\alpha_1[.,.]} \sup_{b_0} \sup_{b_1} \varphi(X_{2,x}^{\alpha[b],b}) \\ &\geq \inf_{\alpha_0[.]} \sup_{b_0} \sup_{b_1} \inf_{\alpha_1[.,.]} \varphi(X_{2,x}^{\alpha[b],b}) \end{aligned}$$

where $\alpha_0[.]$ denotes any function of $A^B \equiv \mathcal{F}(B, A)$, and $\alpha_1[.,.]$ any function of $A^{B\times B} \equiv \mathcal{F}(B \times B, A)$. But $X_{2,x}^{\alpha[b],b} = F(x_1, \alpha_1[b_0, b_1], b_1)$ where $x_1 = X_{1,x}^{\alpha[b],b}$ depends only of $\alpha_0[b_0]$ and b_0. Therefore it is easy to see that $\inf_{\alpha_1[.,.]} \varphi(X_{2,x}^{\alpha[b],b}) = \inf_{\alpha_1[.,.]} \varphi(F(x_1, \alpha_1[b_0, b_1], b_1) = \inf_{a_1\in A} \varphi(F(x_1, a_1, b_1))$, which leads to

$$V_0(x) \geq \inf_{\alpha_0[.]} \sup_{b_0} \sup_{b_1} \inf_{a_1} \varphi(X_{2,x}^{(\alpha_0[b_0],a_1),(b_0,b_1)}).$$

Now recall that $\inf_{\alpha_0\in B^A} \sup_{b_0\in B} Q(\alpha_0[b_0], b_0) = \sup_{b_0\in B} \inf_{a_0\in A} Q(a_0, b_0)$ (see Lemma 3.3), hence we can exchange $\inf_{\alpha_0}$ and $\sup_{b_0}$ to obtain

$$V_0(x) \geq \sup_{b_0} \inf_{a_0} \sup_{b_1} \inf_{a_1} \varphi(X_{2,x}^{(\alpha_0[b_0],a_1),(b_0,b_1)}) \equiv \overline{V}_0(x).$$

To prove the reverse inequality, let us define $\bar{\alpha}_0 \in A^B$ (where $A^B = \mathcal{F}(B, A)$) and $\bar{\alpha}_1 \in A^{B^2} \equiv \mathcal{F}(B^2, A)$ such that:

$$\bar{\alpha}_0[b_0] \in \operatorname*{arginf}_{a_0\in A} \left(\sup_{b_1} \inf_{a_1} \varphi(X_{2,x}^{(a_0,a_1),(b_0,b_1)}) \right) \tag{46}$$

and

$$\bar{\alpha}_1[b_0, b_1] \in \operatorname*{arginf}_{a_1\in A} \varphi(X_{2,x}^{(\bar{\alpha}_0[b_0],a_1),(b_0,b_1)}). \tag{47}$$

Then, by using the definition of V_0, $\bar{\alpha}_1$ and $\bar{\alpha}_0$, we obtain

$$\begin{aligned} V_0(x) &\leq \sup_{b_0} \sup_{b_1} \varphi(X_{2,x}^{\bar{\alpha}[b],b}) \\ &= \sup_{b_0} \left(\sup_{b_1} \inf_{a_1} \varphi(X_{2,x}^{(\bar{\alpha}_0[b_0],a_1),(b_0,b_1)}) \right) \\ &= \sup_{b_0} \inf_{a_0} \sup_{b_1} \inf_{a_1} \varphi(X_{2,x}^{(a_0,a_1),(b_0,b_1)}) \equiv \overline{V}_0(x). \end{aligned}$$

Hence the desired result.

Notice that for the general case $N \geq 1$, for a given $x \in \mathbb{R}^d$, an optimal non-anticipative strategy $\bar{\alpha} = (\bar{\alpha}_k) \in S_N$ can be obtained by choosing first

$$\bar{\alpha}_0[b_0] \in \operatorname*{argmin}_{a_0\in A} g(x) \vee V_1(F(x, a_0, b_0)),$$

then by choosing

$$\bar{\alpha}_1[b_0, b_1] \in \operatorname*{argmin}_{a_1 \in A} g(\bar{x}_1) \vee V_2(F(\bar{x}_1, a_1, b_1))$$

where we use the notation $\bar{x}_1 = F(x, \bar{\alpha}_0[b_0], b_0)$, and so on, choosing at any step $k \leq N-1$:

$$\bar{\alpha}_k[b_0, \dots, b_k] \in \operatorname*{argmin}_{a_k \in A} g(\bar{x}_k) \vee V_{k+1}(F(\bar{x}_k, a_k, b_k))$$

where we use the recursive notation $\bar{x}_k := F(\bar{x}_{k-1}, \bar{\alpha}_{k-1}[b_0, \dots, b_{k-1}], b_{k-1})$, $k \geq 0$ and $\bar{x}_0 := x$.

Now let us turn to the second equality between $\tilde{V}_0(x)$ and $\overline{V}_0(x)$. We first prove $\tilde{V}_0(x) \geq \overline{V}_0(x)$ in the case $N = 2$. By using the general fact that $\inf_p \sup_q Q(p,q) \geq \sup_q \inf_p Q(p,q)$, we obtain

$$\begin{aligned}
\tilde{V}_0(x) &= \inf_{\alpha_0[.,.] \in \Gamma} \inf_{\alpha_1[.,.] \in \Gamma} \sup_{b_0} \sup_{b_1} \varphi(X_{2,x}^{\alpha[b],b}) \\
&\geq \inf_{\alpha_0[.,.]} \sup_{b_0} \sup_{b_1} \inf_{\alpha_1[.,.] \in \Gamma} \varphi(X_{2,x}^{\alpha[b],b}) \\
&\geq \inf_{\alpha_0[.,.]} \sup_{b_0} \sup_{b_1} \inf_{a_1} \varphi(X_{2,x}^{(\alpha_0[x,b_0],a_1),(b_0,b_1)}).
\end{aligned}$$

For the last term we have used the fact that, for any $b = (b_0, b_1)$, $\inf_{\alpha_1[.,.] \in \Gamma} \varphi(X_{2,x}^{\alpha[b],b}) = \inf_{a_1} \varphi(X_{2,x}^{(\alpha_0[x,b_0],a_1),(b_0,b_1)})$. In the same way, as in Lemma 3.3, we have now

$$\tilde{V}_0(x) \;\geq\; \sup_{b_0} \inf_{a_0} \sup_{b_1} \inf_{a_1} \varphi(X_{2,x}^{(a_0,a_1),(b_0,b_1)}) \;=\; \overline{V}_0(x).$$

The general case when $N \geq 1$ can be proved in the same way.

In order to prove the reverse inequality, let α_k^* be a function of Γ such that

$$\alpha_k^*(x,b) \in \operatorname*{arginf}_{a \in A} g(x) \vee V_{k+1}(F(x,a,b)) \tag{48}$$

Recall the DPP (for V_k or $\overline{V}_k$) : $V_k(x) = \sup_b \inf_a g(x) \vee V_{k+1}(F(x,a,b))$. Then $V_k(x) = \sup_{b_k} g(x) \vee V_{k+1}(F(x, \alpha_k^*(x,b_k), b_k))$.

Hence we deduce, denoting $x_1^* = F(x, \alpha_0^*(x,b_0), b_0))$, $x_2^* = F(x_1^*, \alpha_1^*(x_1^*, b_1), b_1)$, and so on, in the expressions below:

$$\begin{aligned}
\bar{V}_0(x) &= \sup_{b_0} g(x) \vee \bar{V}_1(F(x, \alpha_0^*(x,b_0), b_0)) \;\equiv\; \sup_{b_0} g(x) \vee \bar{V}_1(x_1^*) \\
&= \sup_{b_0} \sup_{b_1} g(x) \vee g(x_1^*) \vee \overline{V}_2(x_2^*) \\
\dots &= \sup_{b_0} \sup_{b_1} \cdots \sup_{b_{N-1}} g(x) \vee g(x_1^*) \vee \cdots \vee g(x_{N-1}^*) \vee \varphi(x_N^*).
\end{aligned}$$

In particular, $\alpha^* := (\alpha_0^*, \alpha_1^*, \dots)$ is an optimal element of Γ^N for $\tilde{V}_0$, in the sense that it reaches the infimum in (6). Hence we conclude to the desired result.

B Proof of Theorem 2.7

In all this section the assumptions of Theorem 2.7 holds on the dynamics, in particular $f(x,a,b) = f_1(x,a) + f_2(x,b)$. The first result consists in comparing continuous trajectories (of the form $y_{0,x}^{\alpha[b],b}(t)$) with discrete Euler scheme trajectories (of the form $X_{k,x}^{\alpha^d[b^d],b^d}$), with an error of order $O(\tau)$.

From these estimates, the proof of Theorem 2.7 will follow.

Proposition B.1. *Let $T > 0$, $N \in \mathbb{N}^*$ and $\tau = \frac{T}{N}$. There exists $C_T > 0$, depending only on T, such that:*
(i) for any $\alpha \in \Gamma_{(0,T)}$ non-anticipative strategy, there exists $\alpha^d \in S_N$ (a "discrete" non-anticipative strategy) such that

$$\forall b^d \in B^N,\ \exists b \in \mathcal{B}_T, \qquad \sup_{k\in[\![0,N]\!]} \|y_{0,x}^{\alpha[b],b}(t_k) - X_{k,x}^{\alpha^d[b^d],b^d}\| \leq C_T(1+\|x\|)\tau; \tag{49}$$

(ii) for any $\alpha^d \in S_N$, there exists a non-anticipative strategy $\alpha \in \Gamma_{(0,T)}$ such that

$$\forall b \in \mathcal{B}_T,\ \exists b^d \in B^N, \qquad \sup_{k\in[\![0,N]\!]} \|y_{0,x}^{\alpha[b],b}(t_k) - X_{k,x}^{\alpha^d[b^d],b^d}\| \leq C_T(1+\|x\|)\tau. \tag{50}$$

Proof. (i) Let α be in $\Gamma_{(0,T)}$. We construct $\alpha^d = (\alpha_0,\ldots,\alpha_{N-1}) : B^N \to A^N$ as follows: for any $b^d = (b_0,\ldots,b_{N-1})$ in B^N, we consider the piecewise constant control b such that $b(t) := b_k$ on $[t_k, t_{k+1}[$, and we set, for all k, $\alpha_k^d(b^d) := a_k \in A$ such that

$$\frac{1}{\tau}\int_{t_k}^{t_{k+1}} f_1(y^{\alpha[b],b|}(t_k), \alpha[b](s))ds = f_1(y^{\alpha[b],b}(t_k), a_k).$$

This is possible since we assume that $f_1(x,A)$ is convex for all x (see e.g. [6]).

Let us first check that $\alpha^d \in S_N$.

Now let us denote $y_k := y_{0,x}^{\alpha[b],b}(t_k)$ and $x_k := X_{k,x}^{\alpha^d[b^d],b^d}$. In order to prove the desired bound, we need to establish, for some constant $C \geq 0$,

$$\sup_{0\leq k\leq N} \|y_k - x_k\| \leq C(1+\|x\|)\tau.$$

Notice that

$$\begin{aligned}\frac{1}{\tau}\int_{t_k}^{t_{k+1}} f(y_k, \alpha[b](s), b(s))ds &= f_1(y_k,a_k) + \frac{1}{\tau}\int_{t_k}^{t_{k+1}} f_2(y_k, b(s))ds \\ &= f_1(y_k,a_k) + f_2(y_k,b_k) \ = \ f(y_k,a_k,b_k)\end{aligned}$$

Hence, with $y(t) := y_{0,x}^{\alpha[b],b}(t)$, it holds $y_k = y(t_k)$ and

$$\begin{aligned} y_{k+1} &= y_k + \int_{t_k}^{t_{k+1}} f(y(s), \alpha[b](s), b(s))ds \\ &= y_k + \int_{t_k}^{t_{k+1}} f(y_k, \alpha[b](s), b(s))ds + \varepsilon_k \end{aligned} \tag{51}$$

where $|\varepsilon_k| \le \tau L \max_{[t_k,t_{k+1}]} |y(s) - y(t_k)|$, with $L = [f]_1$. We have also, by construction of a_k and of $b(.)$:

$$y_{k+1} \;=\; y_k + \tau f(y_k, a_k, b_k) + \varepsilon_k. \tag{52}$$

Furthermore by standard Gronwall estimates, with $|f(x, a, b)| \le L|x| + C$ where $C = \max_{a,b} |f(0, a, b)|$, we have $|y(t)| \le (TC + |x|)e^{LT}$, and then

$$\sup_{s\in[t_k,t_{k+1}]} |y(s) - y(t_k)| \le \tau(L(TC + |x|)e^{LT} + C) \equiv C_2\tau \tag{53}$$

(where C_2 depends linearly on $|x|$). Hence $\varepsilon_k \le C_2 L\tau^2$ for all $0 \le k \le N - 1$. On the other hand we have

$$x_{k+1} = x_k + \tau f(x_k, a_k, b_k). \tag{54}$$

Using (52) and (54) we deduce $|y_{k+1} - x_{k+1}| \le |y_k - x_k|(1 + L\tau) + |\varepsilon_k|$. By using a discrete Gronwall estimate, and since $e_0 = 0$ and $N\tau = T$, we deduce $\max_{0\le k\le N} |y_k - x_k| \le (1 + L\tau)^N \sum_{0\le k\le N-1} |\varepsilon_k| \le e^{LT} C_2 LT\tau$.

This concludes to the desired bound.

(ii) Conversely, let $x \in \mathbb{R}^d$ and let $\alpha^d \in S_N$ be a discrete non-anticipative strategy. We aim to define some $\alpha \in \Gamma_{(0,T)}$ associated to α^d. Let $b \in \mathcal{B}_T$ be a given measurable control. To this control b, we are going to associate a control $b^d = (b^d_0, \dots, b^d_{N-1}) \in B^N$, a piecewise constant control $\alpha[b]$ as well as a set of points $y_k = y^{\alpha[b],b}_{0,x}(t_k)$ as follows. For $k = 0$, let $y_0 := x$. From the values $b|_{[0,t_1[}$ and by using the convexity of $f_2(y_0, B)$, we can find $b^d_0 \in B$ such that

$$\frac{1}{\tau}\int_0^{t_1} f_2(y_0, b(s))ds = f_2(x, b^d_0).$$

Then we set

$$\alpha[b](s) := \alpha^d_0[b_0], \quad \forall s \in [0, t_1[.$$

Now we construct $\alpha[b]$ in a recursive way. Assume that $y_k = y^{\alpha[b],b}_{0,x}(t_k)$ is known, using the knowledge of $b(s)$ and of $\alpha[b](s)$ for $s \in [0, t_k[$. By using the convexity of $f_2(y_k, B)$, there exists $b^d_k \in B$ such that

$$\frac{1}{\tau}\int_{t_k}^{t_{k+1}} f_2(y_k, b(s))ds = f_2(x, b^d_k).$$

Then we set

$$\alpha[b](s) := \alpha^d_k[b^d_0, \dots, b^d_k], \quad \forall s \in [t_k, t_{k+1}[.$$

This allows to define $y_{k+1} = y_{0,x}^{\alpha[b],b}(t_{k+1})$. We can check also that the contructed α is non-anticipative. Finally, denoting $y(t) := y_{0,x}^{\alpha[b],b}(t)$, we have

$$\begin{aligned} y_{k+1} &= y_k + \int_{t_k}^{t_{k+1}} f(y(s), \alpha[b](s), b(s))ds \\ &= y_k + \int_{t_k}^{t_{k+1}} f(y_k, \alpha[b](s), b(s))ds + \varepsilon_k \\ &= y_k + \tau f(y_k, \alpha_k^d[b], b_k^d) + \varepsilon_k \end{aligned}$$

by construction and where where $\|\varepsilon_k\| \leq C_2 L\tau$ as in the proof of (i).

On the other hand, by the definitions of $x_k := X_{k,x}^{\alpha^d[b^d],b^d}$, we have

$$x_{k+1} = x_k + \tau f(x_k, \alpha_k^d[b], b_k^d)$$

Hence we obtain a similar error estimate as in (i) and this concludes the proof of (ii). □

Proof of Theorem 2.7. Notice first that (by using estimate (53)) we have

$$\begin{aligned} &|\max_{s\in(0,T)} g(y(s)) \vee \varphi(y(T)) - \max_{0\leq k\leq N-1} g(y(t_k)) \vee \varphi(y(T))| \\ &\qquad\leq \max_{0\leq k\leq N-1} \left| \max_{s\in[t_k,t_{k+1}]} g(y(s)) - g(y(t_k)) \right| \leq [g]C_2\tau. \end{aligned}$$

Let $\Phi(x_0, \ldots, x_N) := \max_{0\leq k\leq N-1} g(x_k) \vee \varphi(x_N)$. From the definition of $v_0(x)$ and the previous inequality we have

$$\left| v_0(x) - \inf_{\alpha\in\Gamma_{(0,T)}} \sup_{b\in\mathcal{B}_T} \Phi((y_{0,x}^{\alpha[b],b}(t_k))_{k\geq 0}) \right| \leq [g]C_2\tau.$$

Let $\varepsilon > 0$. There exists $\alpha \in \Gamma_{(0,T)}$ such that

$$\sup_{b\in\mathcal{B}_T} \Phi((y_{0,x}^{\alpha[b],b}(t_k))) \leq v_0(x) + [g]C_2\tau + \varepsilon. \tag{55}$$

By Proposition B.1(i) there exists $\alpha^d \in S_N$ such that (49) holds. This implies in particular that

$$\forall b^d \in B^N,\ \exists b \in \mathcal{B}_T,\ \left| \Phi((y_{0,x}^{\alpha[b],b}(t_k))) - \Phi((X_{k,x}^{\alpha^d[b^d],b^d})) \right| \leq ([g] \vee [\varphi])\, C_T(1 + \|x\|)\tau.$$

Therefore also

$$\sup_{b^d\in B^N} \Phi((X_{k,x}^{\alpha^d[b^d],b^d})) \leq \sup_{b\in\mathcal{B}_T} \Phi((y_{0,x}^{\alpha[b],b}(t_k))) + ([g] \vee [\varphi])\, C_T(1 + \|x\|)\tau,$$

from which we deduce, using (55)

$$V_0(x) \leq v_0(x) + C_2 L\tau + ([g] \vee [\varphi])\, C_T(1 + \|x\|)\tau + \varepsilon.$$

Letting $\varepsilon \downarrow 0$, this gives the upper bound for $V_0(x) - v_0(x)$ in the desired form. The lower bound is obtained in a similar way from Proposition B.1(ii). □

C Analytical formula for some examples

Data and analytical formula for Example 1 of Section 6. In this example we consider two constants $x_{A,1} := 2$ and $r_A = 1.2$ (the initial data is somehow centered at $x_A := (2, 0)$ and has a radius r_A in polar coordinates). Let $atan2(x_2, x_1)$ denote the angle $\theta \in]-\pi, \pi]$ of the point $x = (x_1, x_2)$.

We first define the following function $\bar{u}_1(t, x)$, for any $t > 0$ and $x \in \mathbb{R}^2$:

$$\bar{u}_1(t, x) := \min(\bar{\varepsilon}, \max(\underline{\varepsilon}, \bar{u}_2(t, r_1(x), \theta_p(t, x))))$$

where $r_1(x) := \cos(\theta_p)x_1 - \sin(\theta_p)x_2$, $\theta_p(t, x) := \min(\max(atan2(x_2, x_1), -t), t)$, and $\bar{u}_2(t, r, \theta) := \max(|r - x_{A,1}| + bt, 2\pi|\theta - \theta_p)|) - r_A$.

Let $t_0 := 0.25$. We define the terminal cost φ as follows:

$$\varphi(x) := \bar{u}(t_0, x).$$

Then one can see that the following analytical formula holds:

$$v(t, x) := \bar{u}(T - t + t_0, x).$$

Analytical formula for Example 2 of Section 6. It is given by

$$
\begin{aligned}
&x_{\min} = -1,\ x_{\max} = 1,\ y_{\min} = -1,\ y_{\max} = 1\\
&r_1 = \max(x_{\min} - 2t - x, x - (x_{\max} + 2t))\\
&r_2 = \max(y_{\min} - y, y - y_{\max})\\
&x_1 = 1 - 2t,\ y_1 = 1\\
&x_2 = 1 + 2t,\ y_2 = 1 - 2t\\
&x_3 = 1.5 + 2t,\ y_3 = 1.5 - 2t,\ z_3 = 0.5\\
&p = z_3/((x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1))\\
&r_3 = p\ ((x_2 - x_1)(y - y_1) - (y_2 - y_1)(x - x_1))\\
&r_4 = p\ ((x_2 - x_1)(-y - y_1) - (y_2 - y_1)(-x - x_1))\\
&r_5 = \max(r_1, r_2, r_3, r_4)
\end{aligned}
$$

and

$$v(t, x) := \min(0.5, \max(-0.5, r_5)).$$

Analytical formula for Example 3 of Section 6. We found the following formula, similar to the previous case, adapted with the following values. First define, for the "$\min_\alpha \max_b$" problem :

$$\bar{r}_1^- = \max(x_{\min} - 2t - x, x - x_{\max})$$

or, for the "$\max_{\alpha} \min_{b}$" problem :

$$\bar{r}_1^+ = \max(x_{\min} - x, x - (x_{\max} - 2t)).$$

Then define

$$\begin{aligned} \bar{r}_3 &= p\ ((x_2 - x_1)(y - y_1) - (y_2 - y_1)(-x - x_1)) \\ \bar{r}_5^{\pm} &= \max(\bar{r}_1^{\pm}, r_2, \bar{r}_3, r_4) \end{aligned}$$

(with same p value as before), and

$$v^{\pm}(t, x) := \min(0.5, \max(-0.5, \bar{r}_5^{\pm})).$$

Then $v_0^{\pm}(x) \equiv v^{\pm}(0, x)$. Hence $V_0^{\pm}$ is an approximation of $v^{\pm}(0, x)$.

References

[1] M. Akian, S. Gaubert, and A. Lakhoua. The max-plus finite element method for solving deterministic optimal control problems: basic properties and convergence analysis. *SIAM J. Control Optim.*, 47(2):817–848, 2008.

[2] M. Akian, S. Gaubert, and S. Liu. A multi-level fast-marching method for the minimum time problem, 2023.

[3] A. Alla, M. Falcone, and L. Saluzzi. A tree structure algorithm for optimal control problems with state constraints. *Rendiconti di Matematica e delle sue Applicazioni. Serie VII*, 41(3-4):193–221, 2020.

[4] A. Altarovici, O. Bokanowski, and H. Zidani. A general Hamilton-Jacobi framework for non-linear state-constrained control problems. *ESAIM: Control, Optimisation and Calculus of Variations*, 19(337–357), 2013 (2012 electronic).

[5] M. Assellaou, O. Bokanowski, A. Desilles, and H. Zidani. Value function and optimal trajectories for a maximum running cost control problem with state constraints. Application to an abort landing problem. *ESAIM Math. Model. Numer. Anal.*, 52(1):305–335, 2018.

[6] J.-P. Aubin and A. Cellina. *Differential inclusions*, volume 264 of *Comprehensive studies in mathematics.* Springer, Berlin, Heidelberg, New York, Tokyo, 1984.

[7] A. Bachouch, C. Huré, N. Langrené, and H. Pham. Deep Neural Networks Algorithms for Stochastic Control Problems on Finite Horizon: Numerical Applications. *Methodol. Comput. Appl. Probab.*, 24(1):143–178, 2022.

[8] M. Bardi and I. Capuzzo-Dolcetta. *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations.* Systems and Control: Foundations and Applications. Birkhäuser, Boston, 1997.

[9] M. Bardi, M. Falcone, and P. Soravia. Fully discrete schemes for the value function of pursuit-evasion games. In *Advances in dynamic games and applications (Geneva, 1992)*, volume 1 of *Ann. Internat. Soc. Dynam. Games*, pages 89–105. Birkhäuser Boston, Boston, MA, 1994.

[10] M. Bardi, M. Falcone, and P. Soravia. Numerical methods for pursuit-evasion games via viscosity solutions. In *Stochastic and differential games*, volume 4 of *Ann. Internat. Soc. Dynam. Games*, pages 105–175. Birkhäuser Boston, Boston, MA, 1999.

[11] M. Bardi, S. Koike, and P. Soravia. Pursuit-evasion games with state constraints: dynamic programming and discrete-time approximations. *Discrete Contin. Dynam. Systems*, 6(2):361–380, 2000.

[12] M. Bardi and P. Soravia. Approximation of differential games of pursuit-evasion by discrete-time games. In *Differential games—developments in modelling and computation (Espoo, 1990)*, volume 156 of *Lect. Notes Control Inf. Sci.*, pages 131–143. Springer, Berlin, 1991.

[13] C. Barrera-Esteve, F. Bergeret, C. Dossal, E. Gobet, A. Meziou, R. Munos, and D. Reboul-Salze. Numerical methods for the pricing of swing options: a stochastic control approach. *Methodology and computing in applied probability*, 8(4):517–540, 2006.

[14] E. N. Barron, L. C. Evans, and R. Jensen. Viscosity solutions of Isaacs' equations and differential games with Lipschitz controls. *J. Differential Equations*, 53(2):213–233, 1984.

[15] P. Bettiol, P. Cardaliaguet, and M. Quincampoix. Zero-sum state constrained differential games: existence of value for Bolza problem. *Internat. J. Game Theory*, 34(4):495–527, 2006.

[16] O. Bokanowski, N. Forcadel, and H. Zidani. Reachability and minimal times for state constrained nonlinear problems without any controllability assumption. *SIAM J. Control Optim.*, 48(7):4292–4316, 2010.

[17] O. Bokanowski, N. Gammoudi, and H. Zidani. Optimistic Planning Algorithms For State-Constrained Optimal Control Problems. *Computers & Mathematics with Applications*, 109(1):158–179, 2022.

[18] O. Bokanowski, J. Garcke, M. Griebel, and I. Klompmaker. An adaptive sparse grid semi-Lagrangian scheme for first order Hamilton-Jacobi Bellman equations. *Journal of Scientific Computing*, 55(3):575–605, 2013.

[19] O. Bokanowski, A. Prost, and X. Warin. Neural networks for first order HJB equations and application to front propagation with obstacle terms. *Partial Differ. Equ. Appl.*, 4:45, 2023.

[20] L. D. Brown and R. Purves. Measurable selections of extrema. *Ann. Statist.*, 1:902–912, 1973.

[21] R. Buckdahn, P. Cardaliaguet, and M. Quincampoix. Some recent aspects of differential game theory. *Dyn. Games Appl.*, 1(1):74–114, 2011.

[22] P. Cardaliaguet and S. a. Plaskacz. Invariant solutions of differential games and Hamilton-Jacobi-Isaacs equations for time-measurable Hamiltonians. *SIAM J. Control Optim.*, 38(5):1501–1520, 2000.

[23] P. Cardaliaguet, M. Quincampoix, and P. Saint-Pierre. Pursuit differential games with state constraints. *SIAM J. Control Optim.*, 39(5):1615–1632, 2000.

[24] M. G. Crandall and P.-L. Lions. Viscosity solutions of Hamilton-Jacobi equations. *Trans. Amer. Math. Soc.*, 277(1):1–42, 1983.

[25] M. G. Crandall and P. L. Lions. Two approximations of solutions of hamilton-jacobi equations. *Mathematics of Computation*, 43(167):1–19, 1984.

[26] K. Debrabant and E. R. Jakobsen. Semi-Lagrangian schemes for linear and fully nonlinear Hamilton-Jacobi-Bellman equations. In *Hyperbolic problems: theory, numerics, applications*, volume 8 of *AIMS Ser. Appl. Math.*, pages 483–490. Am. Inst. Math. Sci. (AIMS), Springfield, MO, 2014.

[27] S. Dolgov, D. Kalise, and K. K. Kunisch. Tensor Decomposition Methods for High-dimensional Hamilton–Jacobi–Bellman Equations. *SIAM Journal on Scientific Computing*, 43(3):A1625–A1650, Jan. 2021.

[28] R. J. Elliott and N. J. Kalton. *The existence of value in differential games.* American Mathematical Society, Providence, R.I., 1972. Memoirs of the American Mathematical Society, No. 126.

[29] L. C. Evans and P. E. Souganidis. Differential games and representation formulas for solutions of Hamilton-Jacobi-Isaacs equations. *Indiana Univ. Math. J.*, 33(5):773–797, 1984.

[30] M. Falcone and R. Ferretti. Numerical methods for Hamilton-Jacobi type equations. In *Handbook of numerical methods for hyperbolic problems*, volume 17 of *Handb. Numer. Anal.*, pages 603–626. Elsevier/North-Holland, Amsterdam, 2016.

[31] P. Falugi, P.-A. Kountouriotis, and R. B. Vinter. Differential games controllers that confine a system to a safe region in the state space, with applications to surge tank control. *IEEE Trans. Automat. Control*, 57(11):2778–2788, 2012.

[32] G. Ferretti, R. Ferretti, O. Junge, and A. Schreiber. An adaptive multilevel radial basis function scheme for the hjb equation. *IFAC-PapersOnLine*, 50(1):1643–1648, 2017. 20th IFAC World Congress.

[33] W. H. Fleming. The convergence problem for differential games. II. In *Advances in Game Theory*, Ann. of Math. Stud., No. 52, pages 195–210. Princeton Univ. Press, Princeton, NJ, 1964.

[34] A. Friedman. *Differential games.* Pure and Applied Mathematics, Vol. XXV. Wiley-Interscience [A division of John Wiley & Sons, Inc.], New York-London, 1971.

[35] A. Friedman. Existence of extended value for differential games of generalized pursuit-evasion. *J. Differential Equations*, 13:172–181, 1973.

[36] J. Garcke and A. Kröner. Suboptimal feedback control of PDEs by solving HJB equations on adaptive sparse grids. *J. Sci. Comput.*, 70(1):1–28, 2017.

[37] J. Han, A. Jentzen, and W. E. Solving high-dimensional partial differential equations using deep learning. *Proc. Natl. Acad. Sci. USA*, 115(34):8505–8510, 2018.

[38] J. Han and J. Long. Convergence of the Deep BSDE Method for Coupled FBSDEs. *Probability, Uncertainty and Quantitative Risk*, 5(1):5, Dec. 2020. arXiv:1811.01165 [cs, math].

[39] C. Hu and C.-W. Shu. A discontinuous Galerkin finite element method for Hamilton-Jacobi equations. *SIAM Journal on Scientific Computing*, 21:666–690, 1999.

[40] C. Huré, H. Pham, A. Bachouch, and N. Langrené. Deep neural networks algorithms for stochastic control problems on finite horizon: convergence analysis. *SIAM J. Numer. Anal.*, 59(1):525–557, 2021.

[41] R. Isaacs. *Differential games. A mathematical theory with applications to warfare and pursuit, control and optimization.* John Wiley & Sons, Inc., New York-London-Sydney, 1965.

[42] M. Jensen and I. Smears. On the convergence of finite element methods for Hamilton-Jacobi-Bellman equations. *SIAM J. Numer. Anal.*, 51(1):137–162, 2013.

[43] G.-S. Jiang and D. Peng. Weighted ENO schemes for Hamilton-Jacobi equations. *SIAM J. Sci. Comput.*, 21(6):2126–2143, 2000.

[44] C. Jin, P. Netrapalli, and M. Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? In *International conference on machine learning*, pages 4880–4889. PMLR, 2020.

[45] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[46] H. J. Kushner and P. G. Dupuis. *Numerical methods for stochastic control problems in continuous time*, volume 24 of *Applications of mathematics*. Springer, New York, 2001. Second edition.

[47] Y. Lei, Z. Yang, T. Yang, and Y. Ying. Stability and generalization of stochastic gradient methods for minimax problems. In *International Conference on Machine Learning*, pages 6175–6186. PMLR, 2021.

[48] F. Li and C.-W. Shu. Reinterpretation and simplified implementation of a discontinuous Galerkin method for Hamilton-Jacobi equations. *Applied Mathematics Letters*, 18:1204–1209, 2005.

[49] J. v. Neumann. Zur theorie der gesellschaftsspiele. *Math. Ann.*, 100(5):295–320, 1928.

[50] S. Osher and C.-W. Shu. High essentially nonoscillatory schemes for Hamilton-Jacobi equations. *SIAM J. Numer. Anal.*, 28(4):907–922, 1991.

[51] M. Razaviyayn, T. Huang, S. Lu, M. Nouiehed, M. Sanjabi, and M. Hong. Nonconvex min-max optimization: Applications, challenges, and recent theoretical advances. *IEEE Signal Processing Magazine*, 37(5):55–66, 2020.

[52] C.-W. Shu. High order ENO and WENO schemes for computational fluid dynamics. In *High-order methods for computational physics*, volume 9 of *Lect. Notes Comput. Sci. Eng.*, pages 439–582. Springer, Berlin, 1999.

[53] I. Smears and E. Süli. Discontinuous Galerkin finite element methods for time-dependent Hamilton-Jacobi-Bellman equations with Cordes coefficients. *Numer. Math.*, 133(1):141–176, 2016.

[54] B. Somil and J. T. Claire. DeepReach: A Deep Learning Approach to High-Dimensional Reachability. In *International Conference on Robotics and Automation (ICRA)*, 2021.

[55] P. E. Souganidis. Max-min representations and product formulas for the viscosity solutions of Hamilton-Jacobi equations with applications to differential games. *Nonlinear Anal.*, 9(3):217–257, 1985.

[56] R. Vinter, J. Clark, and M. James. The interpretation of discontinuous state feedback control laws as nonanticipative control strategies in differential games. *IEEE Transactions on Automatic Control*, 49(8):1360–1365, 2004.