

ERROR ESTIMATES FOR A TREE STRUCTURE ALGORITHM SOLVING FINITE HORIZON CONTROL PROBLEMS*

LUCA SALUZZI^{1,**} , ALESSANDRO ALLA² AND MAURIZIO FALCONE³

Abstract. In the dynamic programming approach to optimal control problems a crucial role is played by the value function that is characterized as the unique viscosity solution of a Hamilton-Jacobi-Bellman (HJB) equation. It is well known that this approach suffers from the “curse of dimensionality” and this limitation has reduced its use in real world applications. Here, we analyze a dynamic programming algorithm based on a tree structure to mitigate the “curse of dimensionality”. The tree is built by the discrete time dynamics avoiding the use of a fixed space grid which is the bottleneck for high-dimensional problems, this also drops the projection on the grid in the approximation of the value function. In this work, we present first order error estimates for the the approximation of the value function based on the tree-structure algorithm. The estimate turns out to have the same order of convergence of the numerical method used for the approximation of the dynamics. Furthermore, we analyze a pruning technique for the tree to reduce the complexity and minimize the computational effort. Finally, we present some numerical tests to show the theoretical results.

Mathematics Subject Classification. 49L20, 49J15, 49J20, 93B52.

Received June 4, 2021. Accepted October 13, 2022.

1. INTRODUCTION

The dynamic programming (DP) approach introduced by Bellman (see *e.g.* [12]) has been applied to several deterministic and stochastic optimal control problems in finite dimension. This approach has been revitalized thanks to the theory of weak solutions for Hamilton-Jacobi equations, the so-called viscosity solutions, introduced by Crandall and Lions in the middle of the 80s (see the monographs [11, 29] and list of references therein). Despite the huge amount of theoretical results and the numerical methods devoted to develop efficient and accurate algorithms for Hamilton-Jacobi equations, real applications of DP has been up to now limited to rather low dimensional problems. The solution of many optimal control problems (and in particular those governed by evolutive partial differential equations) is still accomplished *via* open-loop controls, see *e.g.* [34]. In fact DP provides an elegant characterization of the value function as the unique viscosity solution of a nonlinear partial

*Funding: This work was supported by the Italian Ministry of Instruction, University and Research (MIUR) PRIN Project 2107 (2017KKJP4X entitled “Innovative numerical methods for evolutionary partial differential equations and applications

Keywords and phrases: Dynamic programming, Hamilton-Jacobi-Bellman equation, optimal control, tree structure, error estimates.

¹ Department of Mathematics, Imperial College London, South Kensington Campus, SW7 2AZ London, UK.

² Dipartimento di Scienze Molecolari e Nanosistemi, Università Ca’ Foscari Venezia, Via Torino 155, 30172 Venezia Mestre, Italy.

³ Sapienza Università di Roma – Piazzale Aldo Moro 5, 00185 Roma, Italy.

** Corresponding author: l.saluzzi@imperial.ac.uk

differential equation (the Hamilton–Jacobi–Bellman equation) which is usually computed on a space grid, this is a major bottleneck for high-dimensional problems. However, this remains an interesting and challenging problem since by an approximate knowledge of the value function one can derive a synthesis of a feedback control law that can be plugged into the controlled dynamics. This remarkable feature of DP allows for a synthesis that can be applied to control problems with non linear dynamics and running costs. The case of a linear dynamics and quadratic costs (LQR problem) has an explicit solution based on the Riccati equation (we refer to the book [14] for a general introduction to numerical methods for the Riccati equation). It is interesting to note that this equation can be solved even in very high-dimensional spaces as in [45, 46] using Krylov subspaces. We also refer to the recent paper [13] for a comparison of various techniques.

As we said, DP suffers from the *curse of dimensionality* and even in low dimension an accurate approximation of viscosity solutions is a challenging problem due to their lack of regularity (the value function is in general just Lipschitz continuous even for regular dynamics and costs). However, the analysis of low order numerical methods is now rather complete even for a state space in \mathbb{R}^d and several methods to solve HJB equations are available (see the monographies by Sethian [44], Osher and Fedkiw [42], Falcone and Ferretti [25] for an extensive discussion of some of these methods). From the practical point of view all the classical PDE methods require a space discretization based on a space grid (or triangulation) and this implies a huge amount of memory allocations for high dimensional problems and makes the problem unfeasible for a dimension $d > 5$ on a standard computer. For some of these methods, a-priori error estimates are available, in particular the construction of a DP algorithm for time dependent problems has been addressed in [26].

Several efforts have been made to mitigate the *curse of dimensionality* for the numerical solutions of deterministic optimal control problems. Although a detailed description of these contributions goes beyond the scope of this paper, let us briefly mention for the sake of completeness other approaches that have been developed: domain decomposition [15, 27, 28, 41], max-plus algebras [1, 39, 40], Model Predictive Control [31, 32], Hopf–Lax representation formulas for Hamilton–Jacobi equations [19, 20] and characteristics based methods [48].

Recently deep learning techniques have been introduced for the approximation of the HJB equations. We mention that in [19, 20] has been proposed to apply a discrete version of Hopf–Lax representation formulas for Hamilton–Jacobi equations avoiding its global approximation on a grid. The advantage of this method is that it can be applied at every point in the space and that it can be easily parallelized. However, this method can not be used for general nonlinear control problems since the Hopf–Lax representation formula is valid only for hamiltonians of the form $H(Du)$ whereas the hamiltonians related to general optimal control problems are of the form $H(x; u; Du)$ (see *e.g.* [11]). Neural networks have also been used in *e.g.* [33] for general nonlinear PDEs with application to HJB equation too. Finally, we mention the recent papers [10, 35] where the authors present an hybrid approach based on Deep Neural Networks to solve stochastic control problems. They first approximate the optimal policy by means of neural networks and then the value function by Montecarlo regression.

In the framework of optimal control problems an efficient acceleration technique based on the coupling between value and policy iterations has been proposed and studied in [2]. Although domain decomposition coupled with acceleration techniques can help to solve problems up to dimension 10 we cannot solve problems beyond this limit with a direct approach (the recent application to a landing problem with state constraints in [9] shows the actual dimensional limitations of this approach). One way to attack high-dimensional problems is to apply first a model order reduction technique (*e.g.* Proper Orthogonal Decomposition [47]) to have a low dimensional version of the dynamics by orthogonal projections. Thus, if the reduced system of coordinates for the dynamics has a reasonably low number of dimensions (*e.g.* $d \approx 5$) the problem can be solved *via* the DP approach. We refer to the pioneering work on the coupling between model reduction and HJB approach [37] and to the recent work [6] that provides a-priori error estimates for the aforementioned coupling method. Another tentative has been made using a sparse grid approach in [30], there the authors apply HJB to the control of the wave equation and a spectral elements approximation in [36] which allows to solve the HJB equation up to dimension 12. More recently, in [21, 22] a tensor decomposition has been introduced to approximate the HJB equation.

In this paper we will analyze the method to mitigate the curse of dimensionality originally proposed in [3], based on a tree structure algorithm which does not require a spatial discretization of the problem. In this way

there is no need to store the nodes of the grid/triangulation of the computational domain. The same time discretization has been proposed by Capuzzo Dolcetta in [17] for the infinite horizon problem and it has been exploited in [23] for numerical purposes in combination with a grid projection *via* polynomial interpolation. Here we are going to abandon the space grid interpolation to exploit the tree structure based on the time discretization. Note that the tree strongly depends on the vector field, the number of steps used for the time discretization and the cardinality of the control set. Thus, the time discretization can result in a huge number of branches making the numerical approximation unfeasible. The crucial observation is that not all the tree branches must be considered to get an accurate approximation of the value, with the help of a pruning rule we can drastically reduce the complexity of the algorithm and finally solve the discrete time problem without projecting on a grid. The method applies to general nonlinear finite horizon optimal control problems without additional assumptions on the structure of the dynamical system and it has been naturally extended [4] to get high-order accurate approximations of the value function. Moreover, the TSA has been coupled in [7] with a model order reduction technique based on Proper Orthogonal Decomposition (POD) to deal with optimal control problems driven by two dimensional nonlinear PDEs whose discretization produces a dynamical system of very high dimension (order of thousands). Finally, in [5] the authors present an extension of the TSA to problems with state constraints, together with a convergence result for the value function with convex constraints.

Our main contribution here is the precise error analysis of the TSA developed in Section 3 and of the pruning technique. Note that the pruning technique is crucial to reduce the complexity of the algorithm and to attack problems in very-high dimension (a couple of examples are given here and others have been presented in [3]). We improve the order of convergence provided in [26] for the finite horizon optimal control problem and we extend the error analysis to the pruned TSA keeping the same order of convergence under the semiconcavity assumption on the vector field and the costs. Similar estimates for discrete time approximations of the infinite horizon optimal control problems can be found in [18] where a first order approximation is proved under semiconcavity assumptions, whereas high-order time approximations for the infinite horizon problem are presented in [24].

The paper is organized as follows: in Section 2 we recall some basic facts about the discrete time approximation of the finite horizon problem *via* the DP approach and we present the construction of the tree-structure related to the controlled dynamics. Section 3 contains the main results, in particular a-priori error estimates for the first order approximation. A subsection is devoted to the analysis of the error for the pruning technique used to cut off the branches of the tree in order to reduce the global complexity of the algorithm (an extension to high-order time approximations is presented in [4]). Some numerical tests are presented and analyzed in Section 4. We give our conclusions and perspectives in Section 5.

2. DYNAMIC PROGRAMMING ON A TREE STRUCTURE

This section is devoted to the essential features of the dynamic programming approach and its numerical approximation. The interested reader will find in [3] more details on the tree structure algorithm.

Let us consider the classical *finite horizon problem*. Let the system be driven by

$$\begin{cases} \dot{y}(s) = f(y(s), u(s), s), & s \in (t, T], \\ y(t) = x \in \mathbb{R}^d. \end{cases} \quad (2.1)$$

We will denote by $y : [t, T] \rightarrow \mathbb{R}^d$ the trajectory, by $f : \mathbb{R}^d \times \mathbb{R}^m \times [t, T] \rightarrow \mathbb{R}^d$ the vector field, by $u : [t, T] \rightarrow \mathbb{R}^m$ the control and by

$$\mathcal{U} = \{u : [t, T] \rightarrow U, \text{measurable}\}$$

the set of admissible controls where $U \subset \mathbb{R}^m$ is a compact set. We assume that for each $u \in \mathcal{U}$ there exists a unique solution for (2.1).

The cost functional for the finite horizon optimal control problem is given by

$$J_{x,t}(u) := \int_t^T L(y(s,u), u(s), s) e^{-\lambda(s-t)} ds + g(y(T)) e^{-\lambda(T-t)}, \quad (2.2)$$

where $L : \mathbb{R}^d \times \mathbb{R}^m \times [t, T] \rightarrow \mathbb{R}$ is the running cost and $\lambda \geq 0$ is the discount factor. In the present work we will assume that the functions f, L and g are continuous in all the variables and bounded:

$$\begin{aligned} |f(x, u, s)| &\leq M_f, & |L(x, u, s)| &\leq M_L, & |g(x)| &\leq M_g, \\ \forall x \in \mathbb{R}^d, u \in U \subset \mathbb{R}^m, s \in [t, T], \end{aligned} \quad (2.3)$$

the functions f and L are Lipschitz-continuous with respect to the first variable

$$\begin{aligned} |f(x, u, s) - f(y, u, s)| &\leq L_f |x - y|, & |L(x, u, s) - L(y, u, s)| &\leq L_L |x - y|, \\ \forall x, y \in \mathbb{R}^d, u \in U \subset \mathbb{R}^m, s \in [t, T], \end{aligned} \quad (2.4)$$

and the cost g is also Lipschitz-continuous:

$$|g(x) - g(y)| \leq L_g |x - y|, \quad \forall x, y \in \mathbb{R}^d. \quad (2.5)$$

In the sequel, we will also need to assume the semiconcavity of the functions L and g :

$$L(x + h, t + \tau, u) - 2L(x, t, u) + L(x - h, t - \tau, u) \leq C_L(|h|^2 + \tau^2), \quad \forall x \in \mathbb{R}^d, \forall h, \tau > 0, \quad (2.6)$$

$$g(x + h) - 2g(x) + g(x - h) \leq C_g |h|^2, \quad \forall x \in \mathbb{R}^d, \forall h > 0, \quad (2.7)$$

and a stronger assumption for the function f :

$$|f(x + z, u, t + \tau) - 2f(x, u, t) + f(x - z, u, t - \tau)| \leq C_f(|z|^2 + \tau^2), \quad \forall u \in U, \forall x, z \in \mathbb{R}^d, \forall t, \tau > 0. \quad (2.8)$$

Conditions (2.6)–(2.7) provide an upper bound for a discrete version of the second order derivative. Furthermore, it can be proved that (2.6) and (2.7) are equivalent to the boundedness of the second order derivative in the sense of the distribution (we refer to [16] for further details). Condition (2.8) is a stronger hypothesis since it adds a lower bound to the previous ones.

To derive optimality conditions we use the well-known Dynamic Programming Principle (DPP) due to Bellman. We first define the value function for an initial condition $(x, t) \in \mathbb{R}^d \times [t, T]$:

$$v(x, t) := \inf_{u \in \mathcal{U}} J_{x,t}(u) \quad (2.9)$$

which satisfies the DPP, *i.e.* for every $\tau \in [t, T]$:

$$v(x, t) = \inf_{u \in \mathcal{U}} \left\{ \int_t^\tau L(y(s), u(s), s) e^{-\lambda(s-t)} ds + v(y(\tau), \tau) e^{-\lambda(\tau-t)} \right\}. \quad (2.10)$$

Due to (2.10) we can derive the HJB equation for every $x \in \mathbb{R}^d$, $s \in [t, T]$:

$$\begin{cases} -\frac{\partial v}{\partial s}(x, s) + \lambda v(x, s) + \max_{u \in U} \{-L(x, u, s) - \nabla v(x, s) \cdot f(x, u, s)\} = 0, \\ v(x, T) = g(x). \end{cases} \quad (2.11)$$

Once the value function is known, by *e.g.* (2.11), then it is possible to compute the optimal feedback control as:

$$u^*(t) := \arg \max_{u \in U} \{-L(x, u, t) - \nabla v(x, t) \cdot f(x, u, t)\}, \quad (2.12)$$

where (2.12) has to be understood in an a.e. sense because viscosity solutions are Lipschitz continuous (see *e.g.* [11]).

2.1. Numerical approximation for HJB equation on a tree structure

Equation (2.11) is a nonlinear PDE of the first order which is hard to solve analytically. However, several numerical methods, such as *e.g.* finite difference or semi-Lagrangian schemes, are available to approximate the solution. In the present work we recall the semi-Lagrangian method on a tree structure based on the recent work [3]. Fixed \bar{N} as the number of temporal time steps, we introduce the semi-discrete problem with a time step $\Delta t := (T - t)/\bar{N}$:

$$\begin{cases} V^n(x) = \min_{u \in U} \{\Delta t L(x, u, t_n) + e^{-\lambda \Delta t} V^{n+1}(x + \Delta t f(x, u, t_n))\}, & n = \bar{N} - 1, \dots, 0, \\ V^{\bar{N}}(x) = g(x), & x \in \mathbb{R}^d, \end{cases} \quad (2.13)$$

where $t_n = t + n\Delta t$, $t_{\bar{N}} = T$ and $V^n(x) := V(x, t_n)$. For the sake of completeness we would like to mention that a fully discrete approach is typically based on a time discretization which is projected on a fixed state-space grid of the numerical domain, see *e.g.* [26]. The current work aims to provide error estimates for the algorithm proposed in [3].

For the reader's convenience we recall the tree structure algorithm. Let us consider a finite number of admissible controls $\{u_1, \dots, u_M\}$, obtained discretizing the control domain $U \subset \mathbb{R}^m$ with step-size Δu . A typical example is when U is an hypercube, which is discretized in all the directions with constant step-size Δu , obtaining the finite set $U^{\Delta u} = \{u_1, \dots, u_M\}$. To simplify the notations in the sequel we continue to denote by U the discrete set of controls. We will denote the tree by $\mathcal{T} := \cup_{j=0}^{\bar{N}} \mathcal{T}^j$, where each \mathcal{T}^j contains the nodes of the tree correspondent to time t_j . The first level $\mathcal{T}^0 = \{x\}$ is clearly given by the initial condition x . Starting from the initial condition x , we compute all the nodes given by the dynamics (2.1) discretized by *e.g.* an explicit Euler scheme with different discrete controls $u_j \in U$

$$\zeta_j^1 = x + \Delta t f(x, u_j, t_0), \quad j = 1, \dots, M.$$

Therefore, we have $\mathcal{T}^1 = \{\zeta_1^1, \dots, \zeta_M^1\}$. We observe that all the nodes can be characterized by their n -th *time level*, as follows

$$\mathcal{T}^n = \{\zeta_i^{n-1} + \Delta t f(\zeta_i^{n-1}, u_j, t_{n-1}), j = 1, \dots, M, i = 1, \dots, M^{n-1}\}.$$

We are considering an Euler approximation of the dynamical system to simplify the presentation, but the algorithm can be extended to high-order approximations, as illustrated in [4]. All the nodes of the tree can be briefly denoted as

$$\mathcal{T} := \{\zeta_j^n, j = 1, \dots, M^n, n = 0, \dots, \bar{N}\},$$

where the nodes ζ_i^n are the results of the discrete dynamics at time t_n with the controls $\{u_{j_k}\}_{k=0}^{n-1}$:

$$\zeta_{i_n}^n = \zeta_{i_{n-1}}^{n-1} + \Delta t f(\zeta_{i_{n-1}}^{n-1}, u_{j_{n-1}}, t_{n-1}) = x + \Delta t \sum_{k=0}^{n-1} f(\zeta_{i_k}^k, u_{j_k}, t_k),$$

with $\zeta^0 = x$, $i_k = \left\lfloor \frac{i_{k+1}}{M} \right\rfloor$ and $j_k \equiv i_{k+1} \bmod M$ and $\zeta_i^k \in \mathbb{R}^d$, $i = 1, \dots, M^k$. The notation $\lfloor \cdot \rfloor$ represents the floor function.

Although theoretically the tree structure allows to solve high dimensional problems, its construction might be expensive due to the huge amount of memory allocations, since $\mathcal{T} = O(M^{\bar{N}})$, where M is the number of controls and \bar{N} the number of time steps. For this reason we are going to introduce the following pruning criterion: two given nodes ζ_i^n and ζ_j^n will be merged if

$$\|\zeta_i^n - \zeta_j^n\| \leq \varepsilon_{\mathcal{T}}, \quad \text{with } i \neq j \text{ and } n = 0, \dots, \bar{N}, \quad (2.14)$$

for a given threshold $\varepsilon_{\mathcal{T}} > 0$. Criterion (2.14) will be useful in order to save a huge amount of memory. The selection is made on the fly and in case of two nodes within a distance $\varepsilon_{\mathcal{T}}$, we keep the first node computed, *e.g.* if ζ_i^n and ζ_j^n satisfy (2.14) with $i < j$, we will neglect the node ζ_j^n . Later, we will show a result on the threshold $\varepsilon_{\mathcal{T}} > 0$ in order to guarantee first order convergence.

Once the tree \mathcal{T} has been built, the numerical value function $V(x, t)$ will be computed on the tree nodes and we will denote by $V^n(x)$ the value function computed in x at time $t_n = t + n\Delta t$. The computation of the value function is now straightforward. The TSA defines a time dependent grid $\mathcal{T}^n = \{\zeta_j^n\}_{j=1}^{M^n}$ for $n = 0, \dots, \bar{N}$ and we can approximate (2.10) as follows:

$$\begin{cases} V^n(\zeta_i^n) = \min_{u \in U} \{e^{-\lambda \Delta t} V^{n+1}(\zeta_i^n + \Delta t f(\zeta_i^n, u, t_n)) + \Delta t L(\zeta_i^n, u, t_n)\}, & \zeta_i^n \in \mathcal{T}^n, n = \bar{N} - 1, \dots, 0, \\ V^{\bar{N}}(\zeta_i^{\bar{N}}) = g(\zeta_i^{\bar{N}}), & \zeta_i^{\bar{N}} \in \mathcal{T}^{\bar{N}}. \end{cases} \quad (2.15)$$

The minimization in (2.15) is computed by comparison on the set of discrete controls U .

Once the value function is computed on the nodes of the tree, it is possible to consider a post-processing procedure to achieve a feedback control. We consider the formula for the synthesis of the feedback control

$$u_*^n(x) = \arg \min_{u \in U} \{e^{-\lambda \Delta t} I_{\mathcal{T}^{n+1}}[V^{n+1}](x + \Delta t f(x, u, t_n)) + \Delta t L(x, u, t_n)\}$$

where $I_{\mathcal{T}^{n+1}}[V^{n+1}]$ is an interpolation operator based on the scattered data $(\mathcal{T}^{n+1}, V^{n+1}(\mathcal{T}^{n+1}))$. A detailed analysis of the computational methods to approximate the feedback control goes beyond the scopes of this work. The interested reader will find further information about this procedure in Chapter 3.1 in [43] or [8]. We note that in general it is possible to obtain a control in feedback form on the nodes of tree without the application of interpolation operators. Finally, we would like to mention that a detailed description and comparison about the classical method and tree structure algorithm can be found in [3].

Remark 2.1 (Efficient Pruning). The pruning criterion (2.14) could result in a very expensive algorithm, especially when we deal with high dimensional dynamics. This is the case of a semidiscretization of an evolutive PDE where the dimension easily reaches thousands of unknown variables. To speed up the pruning in this case, we will consider an orthogonal projection \mathcal{P} of the data onto a lower dimensional space which can capture the main features of the dynamics. This can be obtained, for instance, by a Singular Value Decomposition of a matrix containing some snapshots taken from a coarse approximation of the tree. This turns out to accelerate the algorithm, as shown in Section 4. We refer to [3, 7] for more details on this technique in different contexts.

Since the operator \mathcal{P} is a non-expansive operator, *i.e.*

$$\|\mathcal{P}x - \mathcal{P}y\| \leq \|x - y\| \quad \forall x, y \in \mathbb{R}^d. \quad (2.16)$$

Hence, if two nodes satisfy the pruning criterion (2.14), their projections satisfy the same condition due to the inequality (2.16). Thus, it will be sufficient to collect all the nodes fulfilling the pruning criterion in the projected space and to check the same criterion in the original dimension for the selected nodes.

3. ERROR ESTIMATE FOR TSA WITH EULER DISCRETIZATION

In this section we will provide an error analysis for the TSA. We denote $y(s)$ as the exact continuous solution for (2.1) and whenever we want to stress the dependence on the control u , the initial condition x and initial time t we write $y(s; u, x, t)$. We further define $y^n(u)$ as its numerical approximation by an explicit Euler scheme at time t_n . We will consider the piecewise constant extension $\tilde{y}(s; u)$ of the approximation such that

$$\tilde{y}(s, u) := y^{\lfloor s/\Delta t \rfloor}(u), \quad s \in [t, T], u \in \mathcal{U}^\Delta, \quad (3.1)$$

where $\lfloor \cdot \rfloor$ stands for the integer part and

$$\mathcal{U}^\Delta = \{u : [t, T] \rightarrow U, \text{ such that } u(s) = \sum_{k=n}^{\bar{N}-1} u^k \chi_{[t_k, t_{k+1})}(s)\}.$$

Let us now consider the discretized version of the cost functional (2.2):

$$\begin{aligned} J_{x,s}^{\Delta t}(u) &= (t_{n+1} - s)L(x, u, s) + \Delta t \sum_{k=n+1}^{\bar{N}-1} L(y^k, u^k, t_k) e^{-\lambda(t_k - s)} + g(y^{\bar{N}}) e^{-\lambda(t_N - s)} \\ &= \int_s^T L(\tilde{y}(\sigma, u; x, s), u(\sigma), \lfloor \frac{\sigma}{\Delta t} \rfloor \Delta t) e^{-\lambda(\lfloor \frac{\sigma}{\Delta t} \rfloor \Delta t - s)} d\sigma + g(\tilde{y}(T, u; x, s)) e^{-\lambda(T - s)} \end{aligned}$$

for $s \in [t_n, t_{n+1})$. We define the discrete value function as

$$V(x, t) := \inf_{u \in \mathcal{U}^\Delta} J_{x,t}^{\Delta t}(u)$$

which can be computed by the backward problem

$$\begin{aligned} V(x, s) &= \min_{u \in U} \{e^{-\lambda(t_{n+1} - s)} V(x + (t_{n+1} - s)f(x, u, s), t_{n+1}) + (t_{n+1} - s)L(x, u, s)\}, \\ V(x, T) &= g(x), \end{aligned} \quad x \in \mathbb{R}^d, s \in [t_n, t_{n+1}). \quad (3.2)$$

The aim of this section is to find a priori error estimates for the tree algorithm and show the rate of convergence of the approximation V . We will show that if the dynamics is discretized by forward Euler method the error is $O(\Delta t)$:

$$\sup_{(x,t) \in \mathbb{R}^d \times [0, T]} |v(x, t) - V(x, t)| \leq \widehat{C}(T) \Delta t \quad (3.3)$$

where Δt is the time discretization of (2.1) and v is the exact solution (2.9). We remark that the estimate guarantees the same order of convergence of the discretization scheme for the dynamical system (2.1). To

simplify the proof of the main result (3.3) we have splitted the proof into two parts (see Thms. 3.2 and 3.7). We note that this result improves the estimate in [26] under the semiconcavity assumption and it is in line with a similar result for the infinite horizon problem in [18]. To begin with, we show some estimates for the Euler scheme which will be useful to prove the error estimates for TSA. The proposition below follows directly from Grönwall's lemma and its discrete version.

Proposition 3.1. *Let us consider the exact solution trajectory $y(s; u, x, t)$ and its approximation $\tilde{y}(s; u, x, t)$ of (2.1) for a given admissible control $u \in \mathcal{U}^\Delta$. Furthermore, let us assume that assumptions (2.3) and (2.4) hold true. We then obtain the following estimates applying the Euler scheme to (2.1):*

$$|y(s; u, x, t) - \tilde{y}(s; u, x, t)| \leq M_f \Delta t e^{L_f(s-t)}, \quad (3.4)$$

$$\begin{aligned} |\tilde{y}(s; u, x+z, t+\tau) - \tilde{y}(s; u, x, t)| &\leq (|z| + M_f \tau)(1 + L_f \Delta t)^{n-k} \\ s \in [t_n, t_{n+1}) \text{ and } t+\tau &\in [t_k, t_{k+1}) \text{ with } \tau \geq 0, \quad s \geq t+\tau. \end{aligned} \quad (3.5)$$

Using Proposition 3.1, we are able to prove one side of (3.3) as shown in the following theorem.

Theorem 3.2. *Let us assume that conditions (2.3), (2.4) and (2.5) hold true. Then*

$$\sup_{(x,t) \in \mathbb{R}^d \times [0,T]} (v(t, x) - V(t, x)) \leq C(T) \Delta t, \quad \forall t \in [0, T], \quad (3.6)$$

where $C(T)$ is a constant which does not depend on the time step Δt .

Proof. First, since $\mathcal{U}^\Delta \subset \mathcal{U}$, we have

$$v(t, x) - V(t, x) \leq \inf_{u \in \mathcal{U}^\Delta} J_{x,t}(u) - \inf_{u \in \mathcal{U}^\Delta} J_{x,t}^{\Delta t}(u) \leq \sup_{u \in \mathcal{U}^\Delta} |J_{x,t}(u) - J_{x,t}^{\Delta t}(u)|.$$

For a given control $u \in \mathcal{U}^\Delta$, we use the assumptions in Proposition 3.1 to obtain the following

$$|J_{x,t}(u) - J_{x,t}^{\Delta t}(u)| \leq M_f \Delta t \left(\frac{L_L}{L_f} e^{L_f T} + L_g e^{L_f T} \right).$$

Then, we obtain the desired estimate (3.6) with $C(T) = M_f e^{L_f T} \left(\frac{L_L}{L_f} + L_g \right)$. \square

To prove the remaining side of (3.3) we need to assume the semiconcavity of the functions g, L and a stronger assumption on f . The proof of Theorem 3.7 is based on some technical lemmas that are presented below.

Proposition 3.3. *Let us consider the assumptions of Proposition 3.1 and consider the function $f(x, u, t)$ as a Lipschitz-continuous function in time and space uniformly in u satisfying (2.8), then*

$$\begin{aligned} |\tilde{y}(s; u, x+z, t+\tau) - 2\tilde{y}(s; u, x, t) + \tilde{y}(s; u, x-z, t-\tau)| &\leq \tilde{C}(T)(|z|^2 + \tau^2), \\ \forall s \geq t+\tau, \forall u \in U, \forall x, z \in \mathbb{R}^d, \forall t, \tau > 0, \end{aligned} \quad (3.7)$$

where $\tilde{C}(T)$ is a constant that depends on T but does not depend on the time step Δt .

Proof. Let us suppose that $t+\tau \in [t_k, t_{k+1})$ for some $k > 0$, $t \in [t_0, t_1)$ and $t-\tau \in [t_{-k-1}, t_{-k})$. Let us consider $s \in [t_{n+1}, t_{n+2})$, to ease the notation we will denote

$$\begin{aligned} \tilde{y}(s, u, x+z, t+\tau) &:= y_+^{n+1}, & \tilde{y}(s, u, x, t) &:= y^{n+1}, \\ \tilde{y}(s, u, x-z, t-\tau) &:= y_-^{n+1}, & f(y, u, t_n) &:= f^n(y), \end{aligned}$$

and we will drop the dependence on the control u since it is fixed for all the terms considered above. Applying only one step of the forward Euler scheme with $n \geq k$ we get

$$y_+^{n+1} - 2y^{n+1} + y_-^{n+1} = y_+^n - 2y^n + y_-^n + \Delta t (f^n(y_+^n) - 2f^n(y^n) + f^n(y_-^n)).$$

Thus, from assumption (2.8) we obtain the following

$$\begin{aligned} & |f^n(y_+^n) - 2f^n(y^n) + f^n(y_-^n)| \\ &= |f^n(y_+^n) - 2f^n(y^n) + (f^n(y^n - (y_+^n - y^n)) - f^n(y^n - (y_+^n - y^n))) + f^n(y_-^n)| \\ &\leq |f^n(y^n + (y_+^n - y^n)) - 2f^n(y^n) + f^n(y^n - (y_+^n - y^n))| \\ &\quad + |f^n(y_-^n) - f^n(y^n - (y_+^n - y^n))| \leq C_f |y_+^n - y^n|^2 + L_f |y_+^n - 2y^n + y_-^n|. \end{aligned}$$

Then, applying (3.5) we obtain

$$|y_+^{n+1} - 2y^{n+1} + y_-^{n+1}| \leq \Delta t C_1 C_2^{2(n-k)} + C_2 |y_+^n - 2y^n + y_-^n|, \quad (3.8)$$

with $C_1 = C_f(|z| + M_f \tau)^2$ and $C_2 = 1 + L_f \Delta t$. Then, iterating (3.8) we obtain

$$|y_+^{n+1} - 2y^{n+1} + y_-^{n+1}| \leq \Delta t C_1 C_2^{2(n-k)} \sum_{j=0}^{n-k} C_2^{-j} + C_2^{n-k+1} |x + z - 2y^k + y_-^k|. \quad (3.9)$$

Writing the full discrete dynamics for y^k and y_-^k , the right hand side in (3.9) becomes

$$\begin{aligned} & \Delta t C_1 C_2^{2(n-k)} \frac{1 - C_2^{-(n-k+1)}}{1 - C_2^{-1}} + C_2^{n-k+1} \Delta t \left| -2 \sum_{j=0}^{k-1} f^j(y^j) + \sum_{j=-k}^{k-1} f^j(y_-^j) \right| \\ & \leq \frac{C_1 C_2^{2(n-k)+1}}{L_f} + C_2^{n-k+1} \Delta t \left| \sum_{j=0}^{k-1} \left(f^j(y_-^j) - f^j(y^j) + f^{j-k}(y_-^{j-k}) - f^j(y^j) \right) \right|. \end{aligned} \quad (3.10)$$

Now we want to estimate last term in (3.10). Since the first term $f^j(y_-^j) - f^j(y^j)$ of the sum can be obtained as a particular case of the second one, with $k = 0$, let us now focus on the last term

$$\left| \sum_{j=0}^{k-1} \left(f^{j-k}(y_-^{j-k}) - f^j(y^j) \right) \right| \leq L_f \sum_{j=0}^{k-1} \left(|y_-^{j-k} - y^j| + \tau \right). \quad (3.11)$$

Using (3.5), we can write

$$\left| y_-^{j-k} - y^j \right| \leq \left| y_-^{j-k} - y_-^j \right| + \left| y_-^j - y^j \right| \leq \tau M_f + (|z| + M_f \tau) C_2^j.$$

Finally we get

$$|y_+^{n+1} - 2y^{n+1} + y_-^{n+1}| \leq \frac{C_1 C_2^{2(n-k)+1}}{L_f} + 2C_2^{n-k+1} L_f (\tau^2 M_f + C_2^k \tau (|z| + M_f \tau)).$$

Noting that $C_2^n = (1 + L_f \Delta t)^n \leq e^{t_n L_f}$, we obtain the desired result with the constant $\tilde{C}(T)$ equal to

$$\tilde{C}(T) = 2e^{2T} \left(\frac{C_f(\max\{1, M_f\})^2}{L_f} + L_f(2M_f + 1) \right). \quad (3.12)$$

□

Let us recall some properties for the scheme (3.2) which will be useful later since the reverse inequality in (3.6) needs the assumption of semiconcavity for the numerical approximation V . We refer to [16] for a detailed discussion on the importance of the semiconcavity in control problems.

Proposition 3.4. *Let us suppose that the functions L and g are both Lipschitz-continuous and satisfy the semiconcavity assumptions (2.6) and (2.7). Furthermore, let us consider the function $f(x, u, t)$ as a Lipschitz-continuous function in time and space uniformly in u such that it satisfies (2.8). Then the numerical solution V is semiconcave:*

$$V(x + z, t + \tau) - 2V(x, t) + V(x - z, t - \tau) \leq C_V(|z|^2 + \tau^2) \quad \forall x, z \in \mathbb{R}^n, t, \tau \geq 0. \quad (3.13)$$

Proof. Given $x, z \in \mathbb{R}^n$ and $t, \tau \in [0, T]$ such that $t + \tau \in [t_k, t_{k+1})$, $t \in [t_0, t_1)$ and $t - \tau \in [t_{-k-1}, t_{-k})$, we need to prove (3.13). By the definition of value function, we can write

$$\begin{aligned} V(x + z, t + \tau) + V(x - z, t - \tau) - 2V(x, t) &\leq \sup_{u \in U} \{ (t_{k+1} - t - \tau)L(x + z, t + \tau, u) \\ &+ (t_{-k} - t + \tau)L(x - z, t - \tau, u) - 2(t_1 - t)L(x, t, u) \} \\ &+ \sup_{u \in \mathcal{U}^\Delta} (J_T^{\Delta t}(x + z, t_{k+1}, u) + J_T^{\Delta t}(x - z, t_{-k}, u) - 2J_T^{\Delta t}(x, t_1, u)). \end{aligned} \quad (3.14)$$

We can estimate the first term on the right hand side as follows

$$\begin{aligned} &(t_{k+1} - t - \tau)L(x + z, t + \tau, u) + (t_{-k} - t + \tau)L(x - z, t - \tau, u) - 2(t_1 - t)L(x, t, u) \\ &\leq \Delta t \max\{L(x + z, t + \tau, u) + L(x - z, t - \tau, u) - 2L(x, t, u), 0\}. \end{aligned} \quad (3.15)$$

Without loss of generality, we will consider $\lambda = 0$. Given $u \in \mathcal{U}^\Delta$ and denoted by $L(y, u, t_n) = L^n(y)$, we have that the remaining right hand side is equal to

$$\begin{aligned} &\Delta t \left(\sum_{n=k+1}^{N-1} (L^n(y_+^n) + L^n(y_-^n) - 2L^n(y^n)) + \sum_{n=1}^k (L^n(y_-^n) - 2L^n(y^n)) \right) \\ &+ \Delta t \left(\sum_{n=-k}^0 L^n(y_-^n) \right) + g(y_+^N) + g(y_-^N) - 2g(y^N). \end{aligned} \quad (3.16)$$

As already done in the proof of Proposition 3.3, exploiting the properties of L , *i.e.* Lipschitz-continuity and semiconcavity with constant $C_L > 0$, for the first summation in (3.16) we have:

$$L^n(y_+^n) + L^n(y_-^n) - 2L^n(y^n) \leq C_L |y_+^n - y_-^n|^2 + L_L |y_+^n - 2y^n + y_-^n|. \quad (3.17)$$

Using (3.5), we obtain the following bound for the first term

$$\begin{aligned} \Delta t C_L \sum_{n=k+1}^{N-1} |y_+^n - y^n|^2 &\leq \Delta t C_L (|z| + M_f \tau)^2 \sum_{n=k+1}^{N-1} (1 + L_f \Delta t)^{2(N-k)} \\ &\leq \frac{C_L}{L_f} (1 + L_f \Delta t)^{2N} (|z| + M_f \tau)^2 \leq 2 \max\{M_f^2, 1\} \frac{C_L}{L_f} e^{2L_f T} (|z|^2 + \tau^2). \end{aligned} \quad (3.18)$$

Using (3.7), we obtain directly

$$\Delta t L_L \sum_{n=k+1}^{N-1} |y^n(x+z, t+\tau) - 2y^n(x, t) + y^n(x-z, t-\tau)| \leq T L_L \tilde{C}(T) (|z|^2 + \tau^2).$$

Finally we rewrite the second and third summation in (3.16) in the following way

$$\Delta t \sum_{n=1}^k [(L^n(y_-^n) - L^n(y^n)) + (L^{n-k-1}(y_-^n) - L^n(y^n))]$$

and with the same procedure used in the proof of Proposition 3.3 and applying (3.17) with g , we obtain the desired estimate. \square

Next, we introduce a further characterization of V which will turn out to be useful to prove Theorem 3.7. The proof of this proposition can be found in [26].

Proposition 3.5. *Assume that assumptions (2.3), (2.4), (2.5) hold true. Then the solution V of (2.15) is bounded (and uniformly continuous). Furthermore, the following estimate holds*

$$|V(y_0, s) - V(x_0, T)| \leq C (|y_0 - x_0| + (T - t_n) + \Delta t), \quad s \in [t_n, t_{n+1}), \forall x_0, y_0 \in \mathbb{R}^n. \quad (3.19)$$

Finally, before proving Theorem 3.7, we introduce the following lemma (proved in Lem. 4.2, p. 170 of [18]).

Lemma 3.6. *Let $\xi : \mathbb{R}^n \times [0, T] \rightarrow \mathbb{R}$ satisfy*

$$\xi(y+z, t+\tau) - 2\xi(y, t) + \xi(y-z, t-\tau) \leq C_\xi (|z|^2 + |\tau|^2),$$

$\forall y, z \in \mathbb{R}^n, \forall t, \tau \in [0, T]$ such that $t+\tau, t, t-\tau \in [0, T]$ and

$$\xi(0, 0) = 0, \quad \limsup_{(y,t) \rightarrow (0,0)} \frac{\xi(y, t)}{|y| + |t|} \leq 0.$$

Then

$$\xi(y, t) \leq \frac{C_\xi}{6} (|y|^2 + |t|^2) \quad \forall y \in \mathbb{R}^n, t \in [0, T].$$

We are now able to prove our main result.

Theorem 3.7. *Let the assumptions (2.3)–(2.4)–(2.5) hold true. Moreover, let us assume that the functions L and g are semiconcave and that the function $f(x, u, t)$ is Lipschitz continuous in space and time uniformly in u*

and it satisfies (2.8). Then

$$\sup_{(x,t) \in \mathbb{R}^d \times [0,T]} (V(t,x) - v(t,x)) \leq \overline{C}(T)\Delta t, \quad \forall t \in [0,T]. \quad (3.20)$$

Proof. The first part of the proof follows closely from [26]. We introduce the auxiliary function

$$\phi(y,t,x,s) = V(y,t) - v(x,s) + \beta_\epsilon(x-y) + \eta_\alpha(t-s),$$

where $\beta_\epsilon(x) = -\frac{|x|^2}{\epsilon^2}$ and $\eta_\alpha(s) = -\frac{s^2}{\alpha^2}$.

Since v and V are bounded, then for any $\delta > 0$, there exist $(y_1, \tau_1), (x_1, s_1)$ such that

$$\phi(y_1, \tau_1, x_1, s_1) > \sup \phi - \delta.$$

Choosing $\theta(y,x) \in C_0^\infty(\mathbb{R}^d \times \mathbb{R}^d)$, with $\theta(y_1, x_1) = 1$ and $0 \leq \theta \leq 1$, $|D\theta| \leq 1$, such that for any $\delta \in (0, 1)$,

$$\zeta(y,t,x,s) = \phi(y,t,x,s) + \delta\theta(y,x)$$

has a maximum point (y_0, τ_0, x_0, s_0) , with $y_0, x_0 \in \text{supp } \theta$ and $\tau_0, s_0 \in [0, T]$. Therefore, if we set

$$\Phi(x,s) = V(y_0, \tau_0) + \beta_\epsilon(y_0 - x) + \eta_\alpha(\tau_0 - s) + \delta\theta(y_0, x),$$

we can observe that (x_0, s_0) is a local min for $v(x,s) - \Phi(x,s)$. By definition of ζ , we have that

$$\begin{aligned} V(y_0, \tau_0) - v(x_0, s_0) + \beta_\epsilon(y_0 - x_0) + \eta_\alpha(\tau_0 - s_0) + \delta\theta(y_0, x_0) \\ \geq V(y,t) - v(x,s) + \beta_\epsilon(y-x) + \eta_\alpha(t-s) + \delta\theta(y,x). \end{aligned} \quad (3.21)$$

From (3.21) with $x = y = y_0$, $s = s_0$ and $t = \tau_0$, we get

$$|y_0 - x_0| \leq \epsilon^2(L_v + \delta), \quad (3.22)$$

and similarly, with $x = x_0$, $y = y_0$ and $s = t = \tau_0$:

$$|s_0 - \tau_0| \leq \alpha^2 L_v, \quad (3.23)$$

where L_v is the Lipschitz constant of v with respect to time and space. Using (3.21), (3.22) and (3.23), we obtain

$$V(x,s) - v(x,s) \leq V(y_0, \tau_0) - v(x_0, s_0) + (L_v + \delta)\epsilon^2 + \alpha^2 L_v + 2\delta. \quad (3.24)$$

Let us now consider three cases as suggested in [26]. We recall that in this theorem we improve their approximation by means of the semiconcavity which turns out to be essential in the third case of the proof. However, in the first two cases we can directly obtain first order convergence. Without this property we can only prove an order of convergence of $\frac{1}{2}$.

First case ($\tau_0 = T$). In this case $V(y_0, T) = g(y_0) = v(y_0, T)$. Thus, using the Lipschitz-continuity of g we obtain the desired result, setting $\alpha = \epsilon = \sqrt{\Delta t}$.

Second case ($\tau_0 \neq T, s_0 = T$). In this case $v(x_0, T) = g(x_0) = V(x_0, T)$. Supposing $\tau_0 \in [t_n, t_{n+1})$ and using the estimate (3.19) in (3.24), we obtain

$$V(x, s) - v(x, s) \leq C(|y_0 - x_0| + (T - t_n) + \Delta t) + (L_v + \delta)\epsilon^2 + \alpha^2 L_v + 2\delta.$$

Since $\tau_0 - t_n \leq \Delta t$, using (3.23) we can write that

$$T - t_n \leq L_v \alpha^2 + \Delta t,$$

and using (3.22), finally we get

$$V(x, s) - v(x, s) \leq C_3 \epsilon^2 + C_4 \alpha^2 + C_5 \Delta t + 2\delta.$$

If we set $\alpha = \epsilon = \sqrt{\Delta t}$, we get the result, since δ is arbitrary.

Third case ($\tau_0, s_0 \neq T$). We know that v is a viscosity solution, this means that there exists a control $u^* \in U$ such that

$$-\partial_s \Phi(x_0, s_0) + \lambda v(x_0, s_0) - f(x_0, s_0, u^*) \cdot \nabla_x \Phi(x_0, s_0) - L(x_0, s_0, u^*) \geq 0.$$

Thus, we obtain

$$\nabla \eta_\alpha(\tau_0 - s_0) + \lambda v(x_0, s_0) + f(x_0, s_0, u^*) \cdot (\nabla \beta_\epsilon(x_0 - y_0) - \delta \nabla_x \theta(y_0, x_0)) - L(x_0, s_0, u^*) \geq 0. \quad (3.25)$$

By the definition of V (3.2), assuming $\tau_0 \in [t_n, t_{n+1})$ we have

$$V(y_0, \tau_0) - (t_{n+1} - \tau_0) L(y_0, \tau_0, u^*) + -e^{-\lambda(t_{n+1} - \tau_0)} V(y_0 + (t_{n+1} - \tau_0) f(y_0, \tau_0, u^*), t_{n+1}) \leq 0. \quad (3.26)$$

Let us introduce

$$\xi(y, t) = V(y_0 + y, \tau_0 + t) - V(y_0, \tau_0) + (\nabla \beta_\epsilon(x_0 - y_0) + \delta \nabla_y \theta(y_0, x_0)) \cdot y + t \nabla \eta_\alpha(\tau_0 - s_0) \quad (3.27)$$

and follows that

$$\begin{aligned} & \xi(y + z, t + \tau) - 2\xi(y, t) + \xi(y - z, t - \tau) \\ & = V(y_0 + y + z, \tau_0 + t + \tau) - 2V(y_0 + y, \tau_0 + t) + V(y_0 + y - z, \tau_0 + t - \tau). \end{aligned}$$

Proposition 3.4 ensures that the function V is semiconcave. Thus, it follows that also the function ξ with $\xi(0, 0) = 0$ is semiconcave. Let us now check the last hypothesis of Lemma 3.6. Since (y_0, x_0, τ_0, s_0) is a maximum point for ζ , we obtain

$$\begin{aligned} & V(y_0 + y, \tau_0 + t) - V(y_0, \tau_0) \\ & \leq \beta_\epsilon(y_0 - x_0) - \beta_\epsilon(y_0 + y - x_0) + \eta_\alpha(\tau_0 - s_0) - \eta_\alpha(\tau_0 + t - s_0) + \delta[\theta(y_0, x_0) - \theta(y_0 + y, x_0)], \end{aligned}$$

$$\begin{aligned} \xi(y, t) & \leq \beta_\epsilon(y_0 - x_0) - \beta_\epsilon(y_0 + y - x_0) + \nabla \beta_\epsilon(x_0 - y_0) \cdot y + \eta_\alpha(\tau_0 - s_0) \\ & \quad - \eta_\alpha(\tau_0 + t - s_0) + t \nabla \eta_\alpha(\tau_0 - s_0) + \delta(\theta(y_0, x_0) - \theta(y_0 + y, x_0) + \nabla_y \theta(y_0, x_0) \cdot y). \end{aligned}$$

We note that

$$\limsup_{(t,y) \rightarrow (0,0)} \frac{\xi(y,t)}{|y| + |t|} \leq 0.$$

Applying Lemma 3.6 with $y = (t_{n+1} - \tau_0) f(y_0, \tau_0, u^*)$ and $t = t_{n+1} - \tau_0$, we obtain

$$V(y_0 + (t_{n+1} - \tau_0) f(y_0, \tau_0, u^*), t_{n+1}) \leq V(y_0, \tau_0) - (t_{n+1} - \tau_0)(\nabla \beta_\epsilon(x_0 - y_0) + \delta \nabla_y \theta(y_0, x_0)) \cdot$$

$$f(y_0, \tau_0, u^*) - (t_{n+1} - \tau_0) \nabla \eta_\alpha(\tau_0 - s_0) + C_\xi (t_{n+1} - \tau_0)^2 (1 + |f(y_0, \tau_0, u^*)|^2). \quad (3.28)$$

Inserting (3.28) in (3.26) and dividing by $t_{n+1} - \tau_0$ we obtain

$$\begin{aligned} \frac{1 - e^{-\lambda(t_{n+1} - \tau_0)}}{t_{n+1} - \tau_0} V(y_0, \tau_0) &\leq L(y_0, \tau_0, u^*) - e^{-\lambda(t_{n+1} - \tau_0)} (\nabla \beta_\epsilon(x_0 - y_0) \\ &+ \delta \nabla_y \theta(y_0, x_0)) \cdot f(y_0, \tau_0, u^*) + \nabla \eta_\alpha(\tau_0 - s_0) - C(t_{n+1} - \tau_0) (1 + |f(y_0, \tau_0, u^*)|^2). \end{aligned}$$

Finally, subtracting (3.25), we obtain

$$\begin{aligned} \frac{1 - e^{-\lambda(t_{n+1} - \tau_0)}}{t_{n+1} - \tau_0} V(y_0, \tau_0) - \lambda v(x_0, s_0) &\leq L(y_0, \tau_0, u^*) - L(x_0, s_0, u^*) \\ &+ \nabla \beta_\epsilon(y_0 - x_0) \cdot (-e^{-\lambda(t_{n+1} - \tau_0)} f(y_0, \tau_0, u^*) + f(x_0, s_0, u^*)) \\ &+ \nabla n_\alpha(\tau_0 - s_0) (1 - e^{-\lambda(t_{n+1} - \tau_0)}) + \delta \left(-e^{-\lambda(t_{n+1} - \tau_0)} \nabla_y \theta(y_0, x_0) \cdot f(y_0, \tau_0, u^*) \right) \\ &+ \delta (-\nabla_x \theta(y_0, x_0) \cdot f(x_0, s_0, u^*)) + C(t_{n+1} - \tau_0) (1 + |f(y_0, \tau_0, u^*)|^2) \\ &\leq L_L (|y_0 - x_0| + |\tau_0 - s_0|) + 2(L_v + \delta) L_f (|y_0 - x_0| + |\tau_0 - s_0|) + 2L_v \Delta t \\ &+ M\delta + C\Delta t (1 + M_f^2) \leq L(\epsilon^2 + \alpha^2) + C\Delta t + M\delta. \end{aligned}$$

Since δ is arbitrary, choosing $\alpha = \epsilon = \sqrt{\Delta t}$, we obtain the assertion. \square

Error estimate for a discrete control space

In the previous results we did not take into account the error in the minimization procedure. To make our algorithm computationally feasible, we replace the continuous set of controls U by a discrete set $U^{\Delta u}$ in order to compute the minimum by comparison over $U^{\Delta u}$. Let us define the value function computed with a discrete set of controls as

$$\bar{V}(x, t) = \inf_{u \in \bar{U}^\Delta} J_{x,t}^{\Delta t}(u), \quad (3.29)$$

where

$$\bar{U}^\Delta = \{u : [t, T) \rightarrow U^{\Delta u}, \text{ such that } u(s) = \sum_{k=n}^{\bar{N}-1} \alpha_k \chi_{[t_k, t_{k+1})}(s)\}.$$

We can obtain an error estimate for the comparison error adding the hypothesis of Lipschitz-continuity for f and L with respect to the state and the control uniformly in time, *i.e.*

$$\begin{aligned} |f(x, u, s) - f(y, \tilde{u}, s)| &\leq L_f(|x - y| + |u - \tilde{u}|), \\ |L(x, u, s) - L(y, \tilde{u}, s)| &\leq L_L(|x - y| + |u - \tilde{u}|), \\ \forall x, y \in \mathbb{R}^d, u, \tilde{u} \in U \subset \mathbb{R}^m, s \in [t, T]. \end{aligned} \quad (3.30)$$

Proposition 3.8. *Under the assumptions (3.30) and (2.5), then*

$$\sup_{(x,t) \in \mathbb{R}^d \times [0,T]} |V(x, t) - \bar{V}(x, t)| \leq C(T, m)\Delta u, \quad (3.31)$$

where m is the dimension of the control set U .

Proof. First, we can observe that $\bar{U}^\Delta \subset \mathcal{U}^\Delta$, then $V(x, t) \leq \bar{V}(x, t)$. Then, supposing $t \in [t_n, t_{n+1})$ and imposing $\lambda = 0$ for simplicity, we obtain

$$\begin{aligned} \bar{V}(x, t) - V(x, t) &\leq \bar{V}^{n+1}(x + (t_{n+1} - t)f(x, \bar{u}^n, t)) + (t_{n+1} - t)L(x, \bar{u}^n, t) \\ &\quad - V^{n+1}(x + (t_{n+1} - t)f(x, u_*^n, t)) - (t_{n+1} - t)L(x, u_*^n, t), \end{aligned}$$

where

$$u_*^n = \arg \min_{u \in U} \{V^{n+1}(x + (t_{n+1} - t)f(x, u, t)) + (t_{n+1} - t)L(x, u, t)\}$$

and \bar{u}^n is chosen such that $|\bar{u}^n - u_*^n| \leq \frac{\sqrt{m}}{2}\Delta u$. This choice is possible since $U^{\Delta u}$ is a discretization of U with step-size Δu in all directions. Then, if we carry on as Proposition 2.1 in [3], we will obtain

$$\bar{V}(x, t) - V(x, t) \leq L_L \sum_{k=n}^N \alpha_k (|\bar{x}^k - x_*^k| + |\bar{u}^k - u_*^k|) + L_g |\bar{x}^N - x_*^N|, \quad (3.32)$$

where

$$\bar{x}^k := x + \sum_{j=n}^{k-1} \alpha_j f(\bar{x}^j, \bar{u}^j, \bar{t}_j), \quad x_*^k = x + \sum_{j=n}^{k-1} \alpha_j f(x_*^j, u_*^j, \bar{t}_j),$$

$$\alpha_j = \begin{cases} t_{n+1} - t & j = n \\ \Delta t & k \geq n + 1 \end{cases}, \quad \bar{t}_j = \begin{cases} t & j = n \\ t_k & j \geq n + 1 \end{cases},$$

$$u_*^j = \arg \min_{u \in U} \{V^{j+1}(x_*^j + \alpha_j f(x_*^j, u, \bar{t}_j)) + \alpha_j L(x_*^j, u, \bar{t}_j)\}, \quad j \geq n,$$

and \bar{u}^j chosen such that

$$|\bar{u}^j - u_*^j| \leq \frac{\sqrt{m}}{2}\Delta u, \quad j \geq n.$$

By Grönwall's lemma we obtain

$$|\bar{x}^k - x_*^k| \leq e^{(t_k - t)L_f} (t_k - t)L_f \frac{\sqrt{m}}{2} \Delta u, \quad j \geq n, \quad (3.33)$$

and finally coupling (3.32) and (3.33)

$$\sup_{(x,t) \in \mathbb{R}^d \times [0,T]} |\bar{V}(x,t) - V(x,t)| \leq \Delta u \frac{\sqrt{m}}{2} (e^{TL_f} T(L_L + L_g) + TL_L). \quad (3.34)$$

□

Remark 3.9. If the function f has a sub-linear growth, *i.e.* there exist two constants K_1 and K_2 such that

$$|f(x, u, s)| \leq K_1|x| + K_2, \quad \forall x \in \mathbb{R}^d, u \in U \subset \mathbb{R}^m, s \in [t, T],$$

then the trajectory $y(s, u)$ lives in a compact set for all $s \in [t, T]$ and $u \in \mathcal{U}$. Thus, the hypothesis of boundedness (2.3) can be avoided and the conditions (2.4)–(2.8) can be considered locally, *i.e.* holding on every compact subset.

Error estimate for the TSA with pruning

In the previous section we presented an error estimate for the TSA where a first order of convergence is achieved. However, as shown numerically in [3], one can obtain the same order of convergence in the case of the pruned tree if the pruning tolerance $\varepsilon_{\mathcal{T}}$ in (2.14) is chosen properly. In this section, we extend the theoretical results of Section 3 to the pruning case. Thus, let us define the *pruned trajectory*:

$$\eta_{i_n}^{n+1} = \eta_{i_{n-1}}^n + \Delta t f(\eta_{i_{n-1}}^n, u_{j_n}, t_n) + \mathcal{E}_{\varepsilon_{\mathcal{T}}}(\eta_{i_{n-1}}^n + \Delta t f(\eta_{i_{n-1}}^n, u_{j_n}, t_n), \{\eta_i^{n+1}\}_i), \quad (3.35)$$

where the indices i_n and j_n consider the pruning strategy with

$$\mathcal{E}_{\varepsilon_{\mathcal{T}}}(x, \{x_n\}) = \begin{cases} x_k - x & \text{if } k \in \arg \min_n |x - x_n| \text{ and } |x - x_k| \leq \varepsilon_{\mathcal{T}}, \\ 0 & \text{otherwise.} \end{cases} \quad (3.36)$$

The function $\mathcal{E}_{\varepsilon_{\mathcal{T}}}(x, \{x_n\})$ can be interpreted as a perturbation of the numerical scheme and $|\mathcal{E}_{\varepsilon_{\mathcal{T}}}(x, \{x_n\})| \leq \varepsilon_{\mathcal{T}}$. As already done in (3.1), we consider the piecewise constant extension $\tilde{\eta}(s; u)$ of the approximation such that

$$\tilde{\eta}(s, u) := \eta^{\lceil s/\Delta t \rceil}(u) \quad s \in [t, T]. \quad (3.37)$$

First step is to prove that the tolerance must be chosen properly to guarantee a first order convergence of the scheme. The following result is obtained easily through Grönwall's lemma.

Proposition 3.10. *Given the approximation $\tilde{y}(s; u, x, t)$ of equation (2.1) and its perturbation $\tilde{\eta}(s; u, x, t)$ expressed in (3.35), then*

$$|\tilde{y}(s; u, x, t) - \tilde{\eta}(s; u, x, t)| \leq \varepsilon_{\mathcal{T}} \frac{s-t}{\Delta t} e^{L_f(s-t)}, \quad \forall s \in [t, T]. \quad (3.38)$$

Finally, to guarantee first order convergence, the tolerance must be chosen such that

$$\varepsilon_{\mathcal{T}} \leq C\Delta t^2. \quad (3.39)$$

Then we can define the *pruned* discrete cost functional as

$$\begin{aligned} J_{x,s}^{\Delta t,P}(u) &= (t_{n+1} - s)L(x, u, s) + \Delta t \sum_{k=n+1}^{N-1} L(\eta^k, u, t_k) e^{-\lambda(t_k - s)} \\ &\quad + g(\eta^N) e^{-\lambda(t_N - s)}, \end{aligned} \quad (3.40)$$

for $s \in [t_n, t_{n+1})$ and define the *pruned* discrete value function as

$$V^P(x, t) := \inf_{u \in \mathcal{U}^\Delta} J_{x,t}^{\Delta t,P}(u)$$

which now satisfies the following equation

$$\begin{aligned} V^P(x, s) &= \min_{u \in U} \left\{ e^{-\lambda(t_{n+1} - s)} V^P(\eta_u^{n+1}(x), t_{n+1}) + (t_{n+1} - s)L(x, u, s) \right\}, \\ V^P(x, T) &= g(x), \quad x \in \mathbb{R}^d, s \in [t_n, t_{n+1}), \end{aligned} \quad (3.41)$$

where $\eta_u^{n+1}(x) = x + (t_{n+1} - s)f(x, u, s) + \mathcal{E}_{\varepsilon_{\mathcal{T}}}(x + (t_{n+1} - s)f(x, u, s), \{\eta_i^{n+1}\}_i)$. Then, we can prove the following result, similar to Theorem 3.2.

Proposition 3.11. *Under the hypothesis of Theorem 3.2 and condition (3.39), we have*

$$|V(x, t) - V^P(x, t)| \leq C^*(T)\Delta t. \quad (3.42)$$

Finally by triangular inequality and using estimate (3.3) and (3.42), we obtain the desired result:

$$|v(x, t) - V^P(x, t)| \leq \left(C^*(T) + \widehat{C}(T) \right) \Delta t. \quad (3.43)$$

whenever condition (3.39) holds true.

3.1. Pruning in the linear case

In this section we provide an estimate on the cardinality of the pruned tree in the case of linear dynamical systems:

$$\dot{y}(t) = Ay + Bu, \quad (3.44)$$

where the control u is 1-dimensional for simplicity and $A \in \mathbb{R}^{d \times d}$, $B \in \mathbb{R}^d$. Discretizing (3.44) using a one-step scheme, we obtain the following approximation

$$y^{n+1} = S_\Delta y^n + \Delta t \widetilde{S}_\Delta B u^n, \quad (3.45)$$

where $S_\Delta = I_n + \Delta t A$ and $\widetilde{S}_\Delta = I_n$ for an explicit Euler scheme, whereas $S_\Delta = \widetilde{S}_\Delta = (I_n - \Delta t A)^{-1}$ for Implicit Euler. The matrix $I_n \in \mathbb{R}^{n \times n}$ is the identity matrix. We will write y_u^{n+1} with $u = (u^0, \dots, u^n) \in (U^{\Delta u})^{n+1}$ to stress the dependence of the state y^{n+1} on the discrete controls $\{u^i\}_{i=0}^n$. Here, we fix $U^{\Delta u} = [u_{min}, u_{max}]$. Let

us state some results related to the application of the pruning criterion to this particular case. In what follows, we are going to fix the pruning tolerance $\epsilon_{\mathcal{T}} = C\Delta t^2$ which guarantees a first order convergence for the entire procedure.

Proposition 3.12. *Let us consider two different discrete evolutions y_u^n and \tilde{y}_u^n , where $u = (u^0, \dots, u^{n-3}, u^{n-2}, u^{n-1})$ and $\tilde{u} = (u^0, \dots, u^{n-3}, \tilde{u}^{n-2}, \tilde{u}^{n-1})$ such that*

$$u^{n-2} + u^{n-1} = \tilde{u}^{n-2} + \tilde{u}^{n-1}. \quad (3.46)$$

Then, y_u^n and \tilde{y}_u^n satisfy the pruning criterion (2.14) if

$$\left\| (S_{\Delta} - I_n) \tilde{S}_{\Delta} B \right\| |u^{n-1} - \tilde{u}^{n-1}| \leq C\Delta t. \quad (3.47)$$

Moreover, if (3.47) holds for every pair $(u^{n-1}, \tilde{u}^{n-1}) \in U^{\Delta} \times U^{\Delta}$, the pruning criterion is satisfied by every pair of nodes (y_u^n, \tilde{y}_u^n) such that

$$\sum_{i=0}^{n-1} u^i = \sum_{i=0}^{n-1} \tilde{u}^i. \quad (3.48)$$

Then, the levels of the tree grow at most linearly and the cardinality of the pruned tree \mathcal{T}_P is bounded by the following estimate

$$|\mathcal{T}_P| \leq (M-1) \frac{\bar{N}(\bar{N}+1)}{2} + \bar{N} + 1,$$

or equivalently,

$$|\mathcal{T}_P| \leq (M-1) \frac{(T-t)^2 + (T-t)\Delta t}{2\Delta t^2} + \frac{T-t}{\Delta t} + 1, \quad (3.49)$$

where $\bar{N} = \frac{T-t}{\Delta t}$ is the number of time steps, M is the number of discrete controls, t is the initial time and T is the final time.

Proof. By the definition of the discrete dynamics (3.45) and equality (3.46), we obtain

$$\|y_u^n - \tilde{y}_u^n\| \leq \Delta t \left\| (S_{\Delta} - I_n) \tilde{S}_{\Delta} B \right\| |u^{n-1} - \tilde{u}^{n-1}| \leq C\Delta t^2,$$

where we have used condition (3.47) in the last inequality. This proves that the pruning criterion (2.14) is fulfilled.

Let us pass to the second part of the proposition. Using assumption (3.48), the term u^0 can be written as

$$u^0 = \sum_{i=0}^{n-1} \tilde{u}^i - \sum_{i=1}^{n-1} u^i.$$

Then, applying the first part of the proposition, u satisfies the pruning criterion with the vector

$$\left(\tilde{u}^0, \sum_{i=1}^{n-1} \tilde{u}^i - \sum_{i=2}^{n-1} u^i, u^3, \dots, u^{n-1} \right).$$

The procedure can be iterated for consecutive components, proving that y_u^n and $y_{\tilde{u}}^n$ satisfy the pruning criterion. We can pass to the final part of the proposition. First of all we note that $|\mathcal{T}_P^0| = 1$ and $|\mathcal{T}_P^1| = M$ by construction. For a general level n , we note that the cardinality of the pruned level \mathcal{T}_P^n is less or equal to the cardinality of the set of vectors $(u^0, \dots, u^{n-1}) \in [U^\Delta]^n$ with distinct sums. Indeed, if two vectors of controls $u, \tilde{u} \in [U^\Delta]^n$ have the same sum, by the previous result we know that the corresponding discrete evolution y_u^n and $y_{\tilde{u}}^n$ merge *via* the pruning criterion. The cardinalities of these two sets may be different if the pruning strategy cuts further nodes. Hence, by combinatorial computations we obtain that $|\mathcal{T}_P^n| \leq n(M-1) + 1$ for $n \geq 0$. Finally, we observe

$$|\mathcal{T}_P| = \sum_{i=0}^{\bar{N}} |\mathcal{T}_P^i| \leq (M-1) \frac{\bar{N}(\bar{N}+1)}{2} + \bar{N} + 1,$$

proving the result. □

Remark 3.13. In the case of Explicit Euler, condition (3.47) is equivalent to

$$\Delta t \|AB\| |u^{n-1} - \tilde{u}^{n-1}| \leq C \Delta t,$$

which is satisfied for every $\Delta t > 0$ and every pair $(u^{n-1}, \tilde{u}^{n-1}) \in U^\Delta \times U^\Delta$ fixing the constant

$$C = \|AB\| |u_{max} - u_{min}|.$$

This constant C may be very large, affecting the cardinality of the tree. Indeed, according to (3.49), the cardinality of the pruned tree grows as $1/\Delta t^2$. Hence, for an accuracy ε , the bound $C \Delta t < \varepsilon$ implies that the $|\mathcal{T}_P|$ grows as C^2/ε^2 , showing the dependence of the cardinality on the constant C .

4. NUMERICAL TESTS

We now provide a numerical illustration of our theoretical results by means of two test cases. We first deal with the control of the heat equation and compare our approximate value function by a very accurate simulation of the Riccati equation. We, then, compute the order of convergence of our method. The second example deals with an advection equation with a bilinear control. Both examples consider a high dimensional problem. The numerical simulations reported in this paper are performed on a laptop with one core of an Intel Core i5-3, 1 GHz and 8GB RAM. The codes are written in Matlab.

4.1. Test 1: Heat equation

In the first test we deal with the control of the linear heat equation:

$$\begin{cases} z_t = \sigma z_{xx} + z_0(x)u(t), & (x, t) \in [0, 1] \times [0, T], \\ z(0, t) = z(1, t) = 0, & t \in [0, T], \\ z(x, 0) = z_0(x), & x \in [0, 1], \end{cases} \quad (4.1)$$

where the term $z_0(x)u(t)$ allows to introduce a spatial dependence to the control input. We set $T = 1$, $\sigma = 0.15$ and $z_0(x) = x - x^2$. The discretization of (4.1) with a centered finite difference method leads to a system of ODEs:

$$\begin{cases} \dot{y}(t) &= Ay(t) + Bu(t), \\ y(0) &= y_0 \end{cases} \quad (4.2)$$

TABLE 1. Test 1: Error analysis and order of convergence for Implicit Euler scheme of the TSA with $\varepsilon_{\mathcal{T}} = 2\Delta t^2$ and 100 discrete controls.

Δt	Nodes	Nodes Ratio	CPU	Err_2	Err_∞	$Order_2$	$Order_\infty$
0.1	94		0.69s	0.170	0.184		
0.05	442	4.70	3.03s	7.8e-2	8.9e-2	1.12	1.04
0.025	3632	8.51	27s	3.9e-2	4.1e-2	1.17	1.22
0.0125	130767	34.67	1100s	9.7e-3	1.3e-2	2.00	1.65

where $A \in \mathbb{R}^{d \times d}$ is the stiffness matrix and $B \in \mathbb{R}^d$ is given by $B_i = z_0(x_i)$ for $i = 1, \dots, d$, and x_i are the points of the spatial grid. The system of ODEs (4.2) is solved *via* an implicit Euler scheme. We refer to [38] for a complete description of finite difference methods for PDEs.

The cost functional we want to minimize reads:

$$J_{y_0, t}(u) = \int_t^T \left(\|y(s)\|_2^2 + \frac{1}{100} |u(s)|^2 \right) ds + \|y(T)\|_2^2.$$

For this problem we can derive an *exact* solution solving the well-known Riccati differential equation as in [13, 14, 45], if the control is unconstrained. We will compare the numerical value function computed by the TSA and by the Riccati equation. We will compute the following relative errors

$$Err_2 = \sqrt{\frac{\sum_{n=0}^{\bar{N}} |V(y_*^n, t_n) - v(y_R^n, t_n)|^2}{\sum_{n=0}^{\bar{N}} |v(y_R^n, t_n)|^2}}, \quad Err_\infty = \frac{\max_{n=0, \dots, \bar{N}} |V(y_*^n, t_n) - v(y_R^n, t_n)|}{\max_{n=0, \dots, \bar{N}} |v(y_R^n, t_n)|},$$

where $\{y_*^n\}_n$ is the optimal trajectory computed *via* TSA, whereas $\{y_R^n\}_n$ is obtained solving the Riccati equation. In this example the control space is $U = [-1, 0]$. This control space is derived by taking control bounds of the LQR problem. We recall that the control set in the LQR problem is unbounded whereas in the TSA it is bounded. We set the dimension of (4.2), $d = 100$. We consider a time step equal to $\Delta t = 10^{-4}$ for the Riccati equation to obtain an accurate solution. To obtain an efficient pruning as explained in Remark 2.1, we construct a coarse tree using $\Delta t = 0.1$ and 2 controls. Then, we build a snapshot matrix Y , where its columns correspond to the nodes of the coarse tree. We compute the Singular Value Decomposition of the matrix Y and we consider the first left singular vector Ψ . In the construction of the tree, the nodes are first projected using the operator $\mathcal{P} := \Psi^T$ in Remark 2.1 and, then, compared to each other. Thus, the projected nodes that satisfy the pruning criterion are then compared in the original dimension. We recall that the vector Ψ is the direction of maximum variance. This choice will reduce the numbers of comparisons in the full dimension.

The pruning tolerance $\epsilon = C\Delta t^2$ limits the growth of tree to be at most quadratic with respect to the \bar{N} time steps. The constant $C = 2$ turns out to satisfy condition (3.47) for every pair of controls in U^Δ and every choice of Δt considered in the test. In the left panel of Figure 1 we show that the cardinality of the pruned tree $\mathcal{T}_{\mathcal{P}}$ for different time steps and 100 discrete controls. It is possible to note that the theoretical bound (3.49) is satisfied and it turns out that the pruning strategy is cutting more nodes than predicted by Proposition 3.12. A study of the order of convergence for the TSA with 100 discrete controls is provided in Table 1. The control space, in the LQR problem, is not discretized and that may provide some different results as the order of convergence. The order of convergence is around 1 as expected with a pruned tree with tolerance $\varepsilon_{\mathcal{T}} = 2\Delta t^2$. The table also shows the number of nodes for each Δt (second column) and how the cardinality of tree scales with respect to Δt (third column). To give an idea of the importance of the pruning we mention that the ratio of the cardinality for a full tree between $\Delta t = 0.0125$ and $\Delta t = 0.025$ is of order 10^{80} whereas the order with a pruned tree is 34.67. It is clear the huge difference and feasibility of our approach with a pruning criterion. This is due to the fact that the tree lives on a lower-dimensional manifold.

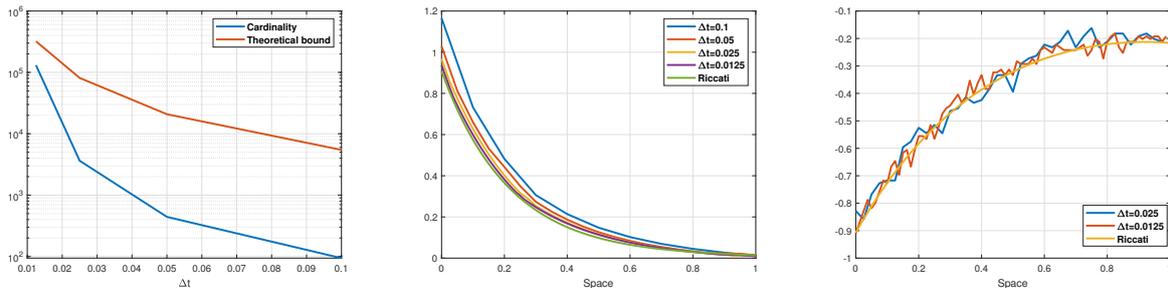


FIGURE 1. Test 1: Cardinality of the trees compared with the theoretical estimate (3.49) in logarithmic scale (*left*), cost functional (*central*) and optimal control (*right*) with 100 discrete controls.

TABLE 2. Test 1: Error analysis and order of convergence for Implicit Euler scheme of the TSA with $\varepsilon_{\mathcal{T}} = 2\Delta t^2$ and $\Delta t = 0.025$.

Discrete controls	Nodes	CPU	Err_2	Err_∞	$Order_2$	$Order_\infty$
4	2803	0.29s	0.173	6.1e-2		
7	3101	0.55s	0.138	5.2e-2	0.33	0.23
13	3351	1.02s	0.136	5.1e-2	0.02	0.03
25	3370	2.03s	0.100	4.3e-2	0.44	0.25
49	3655	4.26s	6.8e-2	3.6e-2	0.55	0.26

Finally, in the central panel of Figure 1 we show the convergence of the cost functional decreasing the temporal step size Δt for a given amount of controls (100 in the plot). As Δt gets smaller, we better approximate our reference solution. In the right panel we show the optimal policy. Clearly, since our control space is not continuous, a chattering phenomenon appears, however it does not improve significantly the quality of our results.

In Table 2 we study the behaviour of the algorithm increasing the number of discrete controls. We observe that halving the control step the CPU time doubles and both errors decrease. We do not observe a first order convergence in the control discretization because for this problem a finer control space does not influence strongly the quality of our approximation.

4.2. Test 2: Advection equation

The second test deals with the optimal control of the advection equation:

$$\begin{cases} y_t + cy_x = yu(t) & (x, t) \in [0, 3] \times [0, T], \\ y(0, t) = y(3, t) = 0 & t \in [0, T], \\ y(x, 0) = y_0(x) & x \in [0, 3], \end{cases} \quad (4.3)$$

where $c = 1.5$, $T = 1$ and $y_0(x) = \sin(\pi x)\chi_{[0,1]}(x)$. To discretize equation (4.3), we consider an upwind scheme in space, whereas we will apply an implicit Euler scheme in time (see [38]). We will minimize the following tracking cost functional:

$$J_{y_0, t}(u) = \int_t^T \int_0^3 \|y(x, s) - y_0(x - cs)\|^2 dx ds + \int_0^3 \|y(x, T) - y_0(x - cT)\|^2 dx,$$

TABLE 3. Test 2: Error analysis and order of convergence for Implicit Euler scheme of the TSA with $\varepsilon_{\mathcal{T}} = \Delta t^2$ and 2 discrete controls.

Δt	Nodes	CPU	Err_2	Err_∞	$Order_2$	$Order_\infty$
0.05	231	0.01s	0.1	0.11		
0.025	861	0.05s	0.05	0.054	0.99	1.00
0.0125	3321	0.17s	0.025	0.027	0.99	0.99
0.00625	13041	0.58s	0.014	0.0154	0.82	0.83

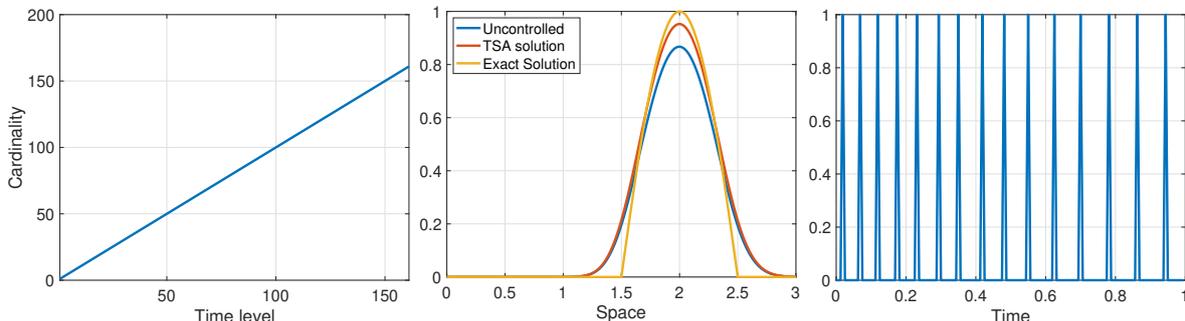


FIGURE 2. Test 2: Number of nodes for each time level (*left*), solution at final time (*middle*) and optimal control (*right*) with 2 discrete controls and $\Delta t = 0.00625$ and $\varepsilon_{\mathcal{T}} = \Delta t^2$.

where the desired state is the solution of the uncontrolled problem, *i.e.* $u \equiv 0$. The exact uncontrolled dynamics preserves over time the L^∞ and L^2 norm, while the upwind approximation introduces a term of numerical diffusion (see *e.g.* [38]). Specifically, in this case the control $u \in [0, 1]$ will reduce the numerical diffusion of the scheme. To solve the control problem we use the TSA with $U = \{0, 1\}$ and the following pruning criterion: $\varepsilon_{\mathcal{T}} = \Delta t^2$. We apply again the efficient pruning (see Rem. 2.1), building the snapshots matrix with $\Delta t = 0.1$ and 2 controls. In Table 3 we compute the error and order of convergence. In this case the exact value function is $v(t, x) = 0$ obtained with the control $u \equiv 0$ in (4.3). As expected, we achieve first of order convergence.

It is possible to note that the increasing ratio for the nodes is almost equal to 4, due to a linear growth of the cardinality. Indeed, in the left panel of Figure 2, we present the number of nodes for each time level of the tree. The linear growth in the time levels leads to a quadratic cardinality of the pruned tree *e.g.* $|\mathcal{T}^P| = \overline{N}(\overline{N} + 1)/2$. In the middle panel of Figure 2, we show the final configuration for the uncontrolled and controlled solution. In Table 4 we compare the L^∞ norm and L^2 norm for the three cases studied. It is possible to note that both norms for the controlled solution are closer to the exact one than in the uncontrolled case since the control reduces the diffusivity of the scheme. In the right panel we show the optimal control computed by our method. We can note that the numerical optimal control oscillates between the two values to counteract the numerical diffusion.

Finally, in Table 5 we report the error analysis and the order of convergence with 5 equidistant discrete controls. If we compare Tables 5 and 3, we can only note a slight improvement on the order of convergence. In the left panel of Figure 3 we report the growth of the cardinality fixing $\Delta t = 0.00625$ and 5 controls. We can observe again a linear growth, but in this case the growth rate is equal to 4. In the right panel of Figure 3, we show the optimal control which exhibits again an oscillating behaviour between two values.

5. CONCLUSION AND FUTURE WORK

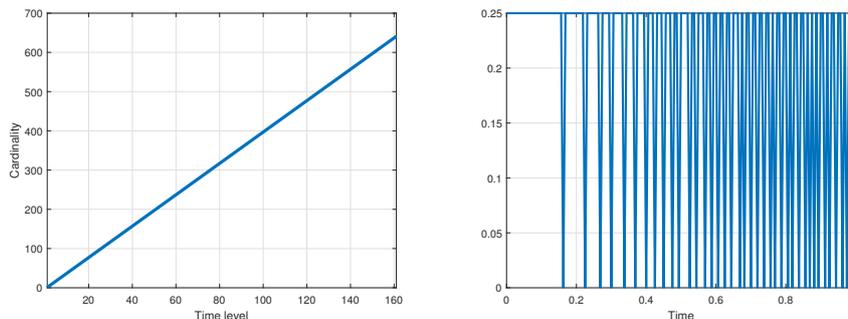
In this work we have proved error estimates for the TSA presented in [3]. The tree structure algorithm allows us to achieve the same order of convergence of the numerical method used in the time discretization of the dynamics. Our error estimate improves previous existing results on the convergence of the semi-discrete value

TABLE 4. Test 2: Comparison of the L^∞ norm and L^2 norm for the exact, the uncontrolled and the controlled solutions at final time with $\Delta t = 0.00625$.

	$\ y(T)\ _\infty$	$\ y(T)\ _2$
Exact	1	0.707
Uncontrolled	0.867	0.636
Controlled	0.953	0.699

 TABLE 5. Test 2: Error analysis and order of convergence for Implicit Euler scheme of the TSA with $\varepsilon_{\mathcal{T}} = \Delta t^2$ and 5 discrete controls.

Δt	Nodes	CPU	Err_2	Err_∞	$Order_2$	$Order_\infty$
0.05	861	0.12s	0.1	0.11		
0.025	3321	0.39s	0.05	0.054	1	1
0.0125	3041	1.46s	0.025	0.027	0.99	0.99
0.00625	51681	5.7s	0.0142	0.0154	0.83	0.83


 FIGURE 3. Test 2: Number of nodes for each time level (*left*) and optimal control (*right*) with 5 discrete controls and $\Delta t = 0.00625$ and $\varepsilon_{\mathcal{T}} = \Delta t^2$.

function adding the semiconcavity assumption. Numerical tests presented in the last section and in [3] confirm the estimate and the relevance of semiconcavity in the approximation.

The cardinality of the tree increases as the number of the control does and the time step size Δt decreases, so we need to prune the tree to reduce the complexity and to save in memory allocations and CPU time. The pruning technique is crucial to produce a more efficient algorithm. In particular, we have shown that if the pruning technique has a reasonable tolerance $\varepsilon_{\mathcal{T}}$, *e.g.* one order higher of the order of convergence of the numerical method of the ODE, we can achieve the same order of the TSA method without pruning.

As future work we aim at analyzing the numerical methods for the synthesis of feedback controls based on the tree structure algorithm.

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** (2008) 817–848.
- [2] A. Alla, M. Falcone and D. Kalise, An efficient policy iteration algorithm for dynamic programming equations. *SIAM J. Sci. Comput.* **37** (2015) 181–200.
- [3] A. Alla, M. Falcone and L. Saluzzi, An efficient DP algorithm on a tree-structure for finite horizon optimal control problems. *SIAM J. Sci. Comput.* **41** (2019) A2384–A2406.

- [4] A. Alla, M. Falcone and L. Saluzzi, High-order approximation of the finite horizon control problem *via* a tree structure algorithm. *IFAC-PapersOnLine* **52** (2019) 19–24.
- [5] A. Alla, M. Falcone and L. Saluzzi, A tree structure algorithm for optimal control problems with state constraints. *Rend. Matemat. Sue Applic.* **41** (2020) 193–221.
- [6] A. Alla, M. Falcone and S. Volkwein, Error analysis for POD approximations of infinite horizon problems *via* the dynamic programming approach. *SIAM J. Control Optim.* **55** (2017) 3091–3115.
- [7] A. Alla and L. Saluzzi, A HJB-POD approach for the control of nonlinear PDEs on a tree structure. *Appl. Numer. Math.* **155** (2020) 192–207.
- [8] A. Alla and L. Saluzzi, Feedback reconstruction techniques for optimal control problems on a tree structure, Preprint [arXiv:2210.02375](https://arxiv.org/abs/2210.02375) (2022).
- [9] M. Assellaou, O. Bokanowski and A. Desilles, 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** (2018) 305–335.
- [10] 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** (2022) 143–178.
- [11] M. Bardi and I. Capuzzo-Dolcetta, Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations. Birkhäuser, Basel (2012).
- [12] R. Bellman, Dynamic Programming. Princeton University Press, Princeton, NJ (1957).
- [13] P. Benner, Z. Bujanović, P. Kürschner and J. Saak, A numerical comparison of different solvers for large-scale, continuous-time algebraic Riccati equations and LQR problems. *SIAM J. Sci. Comput.* **42** (2020) A957–A996.
- [14] D. Bini, B. Iannazzo and B. Meini, Numerical Solution of Algebraic Riccati Equations, Fundam. Algorithms, SIAM, Philadelphia (2012).
- [15] S. Cacace, E. Cristiani, M. Falcone and A. Picarelli, A patchy dynamic programming scheme for a class of Hamilton-Jacobi-Bellman equations. *SIAM J. Sci. Comput.* **34** (2012) A2625–A2649.
- [16] P. Cannarsa and C. Sinestrari, Semiconcave functions, Hamilton-Jacobi equations and optimal control problems. Birkhäuser Boston (2004).
- [17] I. Capuzzo Dolcetta, On a discrete approximation of the Hamilton-Jacobi equation of dynamic programming. *Appl. Math. Optim.* **10** (1983) 367–377.
- [18] I. Capuzzo Dolcetta and H. Ishii, Approximate solutions of the Bellman equation of deterministic control theory. *Appl. Math. Optim.* **11** (1984) 161–181.
- [19] J. Darbon and S. Osher, Splitting enables overcoming the curse of dimensionality. Splitting methods in communication, imaging, science, and engineering, Sci. Comput., Springer, Cham (2016) 427–432.
- [20] J. Darbon and S. Osher, Algorithms for overcoming the curse of dimensionality for certain Hamilton-Jacobi equations arising in control theory and elsewhere. *Res. Math. Sci.* **3** (2016) 19–26.
- [21] S. Dolgov, D. Kalise and K. Kunisch, Tensor decompositions for high-dimensional Hamilton-Jacobi-Bellman equations. *SIAM J. Sci. Comput.* **43** (2021) A1625–A1650.
- [22] S. Dolgov, D. Kalise and L. Saluzzi, Data-driven Tensor Train Gradient Cross Approximation for Hamilton-Jacobi-Bellman Equations. Preprint [arXiv:2205.05109](https://arxiv.org/abs/2205.05109) (2022).
- [23] M. Falcone, A numerical approach to the infinite horizon problem of deterministic control theory, *Appl. Math. Optim.* **15** (1987) 1–13.
- [24] M. Falcone and R. Ferretti, Discrete time high-order schemes for viscosity solutions of Hamilton-Jacobi-Bellman equations. *Numer. Math.* **67** (1994) 315–344.
- [25] M. Falcone and R. Ferretti, Semi-Lagrangian Approximation Schemes for Linear and Hamilton-Jacobi equations. SIAM (2013).
- [26] M. Falcone and T. Giorgi, An approximation scheme for evolutive Hamilton-Jacobi equations, in W.M. McEneaney, G. Yin and Q. Zhang (eds.), “Stochastic Analysis, Control, Optimization and Applications: A Volume in Honor of W.H. Fleming”. Birkhäuser (1999) 289–303.
- [27] M. Falcone, P. Lanucara and A. Seghini, A splitting algorithm for Hamilton-Jacobi-Bellman equations. *Appl. Numer. Math.* **15** (1994) 207–218.
- [28] A. Festa, Domain decomposition based parallel Howard’s algorithm. *Math. Comput. Simul.* **147** (2018) 121–139.
- [29] W.H. Fleming and H.M. Soner, Controlled Markov processes and viscosity solutions. Springer-Verlag, New York (1993).
- [30] J. Garcke and A. Kröner, Suboptimal feedback control of PDEs by solving HJB equations on adaptive sparse grids. *J. Sci. Comput.* **70** (2017) 1–28.
- [31] L. Grüne and J. Pannek, Nonlinear Model Predictive Control. Springer (2011).
- [32] L. Grüne, Dynamic programming, optimal control and model predictive control. Handbook of Model Predictive Control. Control Eng., Birkhäuser/Springer, Cham (2019), pp. 29–52.
- [33] J. Han, A. Jentzen and E. Weinan, Solving high-dimensional partial differential equations using deep learning. *Proc. Natl. Acad. Sci. USA* **115** (2018) 8505–8510.
- [34] M. Hinze, R. Pinau, M. Ulbrich and S. Ulbrich, vol. 23 of *Optimization with PDE Constraints. Mathematical Modelling: Theory and Applications*. Springer-Verlag (2009).
- [35] C. Huré, H. Pham, A. Bachouch and N. Langrené, Deep neural networks algorithms for stochastic control problems on finite horizon: Convergence analysis. *SINUM* **59** (2021) 525–557.

- [36] D. Kalise and K. Kunisch, Polynomial approximation of high-dimensional Hamilton-Jacobi-Bellman equations and applications to feedback control of semilinear parabolic PDEs. *SIAM J. Sci. Comput.* **40** (2018) A629–A652.
- [37] K. Kunisch, S. Volkwein and L. Xie, HJB-POD based feedback design for the optimal control of evolution problems. *SIAM J. Appl. Dyn. Syst.* **4** (2004) 701–722.
- [38] R.J. Leveque, Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems. SIAM Book (2007).
- [39] W.M. McEneaney, Convergence rate for a curse-of-dimensionality-free method for Hamilton-Jacobi-Bellman PDEs represented as maxima of quadratic forms. *SIAM J. Control Optim.* **48** (2009) 2651–2685.
- [40] W.M. McEneaney, A curse-of-dimensionality-free numerical method for solution of certain HJB PDEs. *SIAM J. Control Optim.* **46** (2007) 1239–1276.
- [41] C. Navasca and A.J. Krener, Patchy solutions of Hamilton-Jacobi-Bellman partial differential equations, in A. Chiuso *et al.* (eds.), vol. 364 of *Modeling, Estimation and Control, Lecture Notes in Control and Information Sciences* (2007), pp. 251–270.
- [42] S. Osher and R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces. Springer (2003).
- [43] L. Saluzzi, A Tree-Structure Algorithm for Optimal Control Problems via Dynamic Programming, PhD thesis, Gran Sasso Science Institute (2020) <https://iris.gssi.it/handle/20.500.12571/10021>.
- [44] J.A. Sethian, Level set methods and fast marching methods. Cambridge University Press (1999).
- [45] V. Simoncini, Analysis of the rational Krylov subspace projection method for large-scale algebraic Riccati equations. *SIAM J. Matrix Anal. Appl.* **37** (2016) 1655–1674.
- [46] V. Simoncini, D.B. Szyld and M. Monsalve, On two numerical methods for the solution of large-scale algebraic Riccati equations. *IMA J. Numer. Anal.* **34** (2014) 904–920.
- [47] S. Volkwein, Model Reduction using Proper Orthogonal Decomposition, Lecture Notes, University of Konstanz (2013).
- [48] I. Yegorov, P.M. Dower and L. Grüne, A characteristics based curse-of-dimensionality-free approach for approximating control Lyapunov functions and feedback stabilization. Proceedings of the 23rd International Symposium on Mathematical Theory of Networks and Systems Hong Kong University of Science and Technology, Hong Kong, July 16–20 (2018) 342–349.

Subscribe to Open (S2O)

A fair and sustainable open access model



This journal is currently published in open access under a Subscribe-to-Open model (S2O). S2O is a transformative model that aims to move subscription journals to open access. Open access is the free, immediate, online availability of research articles combined with the rights to use these articles fully in the digital environment. We are thankful to our subscribers and sponsors for making it possible to publish this journal in open access, free of charge for authors.

Please help to maintain this journal in open access!

Check that your library subscribes to the journal, or make a personal donation to the S2O programme, by contacting subscribers@edpsciences.org

More information, including a list of sponsors and a financial transparency report, available at: <https://www.edpsciences.org/en/math-s2o-programme>