A Dynamic Programming approach on a tree structure for finite horizon optimal control problems

Alessandro Alla
works in collaboration with M. Falcone, L. Saluzzi

PUC RIO

Third AFOSR Monterey Workshop
Computational Issues in Nonlinear Control

Monterey, October 8, 2019

Outline

1. Introduction
   - Dynamic programming principle and HJB equations
   - Classical Semi-Lagrangian approach
2. Dynamic programming on a Tree Structure
   - Tree Structure Algorithm
   - A priori error estimates
3. HJB-POD approach on Tree Structure
4. Numerical tests

Outline

1. Introduction
   - Dynamic programming principle and HJB equations
   - Classical Semi-Lagrangian approach
2. Dynamic programming on a Tree Structure
   - Tree Structure Algorithm
   - A priori error estimates
3. HJB-POD approach on Tree Structure
4. Numerical tests

HJB equation for the finite horizon problem

Controlled Dynamics and Cost Functional

$$\begin{cases} \dot{y}(t,u) = f(y(t),u(t),t), & t \in (t,T] \\ y(t) = x \end{cases}$$

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

$$u(t) \in \mathcal{U} = \{u : [t,T] \to U \subset \mathbb{R}^m, \text{ measurable } \}$$

Value Function

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

HJB equation for the finite horizon problem

Optimal Feedback Map

$$u^*(x,t) = \arg\min_{u \in U} \{L(x,u,t) + \nabla v(x,t) \cdot f(x,u,t)\}$$

Dynamic Programming Principle

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

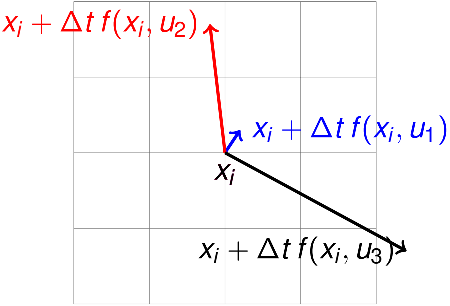
HJB equation

$$-\frac{\partial v}{\partial t}(x,t) + \lambda v(x,t) = \min_{u \in U} \{L(x,u,t) + \nabla v(x,t) \cdot f(x,u,t)\}$$

Classical approach

Semi-Lagrangian scheme ($\lambda = 0$)

$$\begin{cases} V_i^{n-1} = \min\limits_{u \in U}[\Delta t\, L(x_i, u, t_n) + V^n(x_i + \Delta t\, f(x_i, u, t_n))], \; n = N, \dots, 1 \\ V_i^N = g(x_i) \end{cases}$$



Discretization: constant Δt for time and N_u controls

Cons of the approach

- $V^n(x_i + \Delta t\, f(x_i, u, t_n))$ needs an interpolation operator
- Requires a numerical domain chosen a priori and selection of BC
- The curse of dimensionality makes the problem difficult to solve in high dimension (need e.g. model order reduction)

Outline

1. Introduction
   - Dynamic programming principle and HJB equations
   - Classical Semi-Lagrangian approach
2. Dynamic programming on a Tree Structure
   - Tree Structure Algorithm
   - A priori error estimates
3. HJB-POD approach on Tree Structure
4. Numerical tests

Main references for this talk

- A., M. Falcone, L. Saluzzi, *An efficient DP algorithm on a tree-structure for finite horizon optimal control problems*, SISC, 2019
- A., M. Falcone, L. Saluzzi, *High-order Approximation of the Finite Horizon Control Problem via a Tree Structure Algorithm*, IFAC proceeding, 2019
- A., L. Saluzzi, *A HJB-POD approach for the control of nonlinear PDEs on a tree structure*, submitted, 2019, `https://arxiv.org/abs/1905.03395`
- L. Saluzzi, A., M. Falcone, *Error estimates for a tree structure algorithm solving finite horizon control problems*, submitted, 2018, `https://arxiv.org/pdf/1812.11194.pdf`

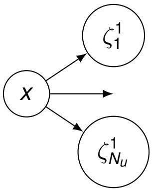
Tree Structure Algorithm (A., Falcone, Saluzzi, '19)

Inputs

Initial condition $x \in \mathbb{R}^d$, discrete set of controls $U = \{u_1, \ldots, u_{N_u}\}$

Starting with x, we follow the dynamics given by the discrete controls

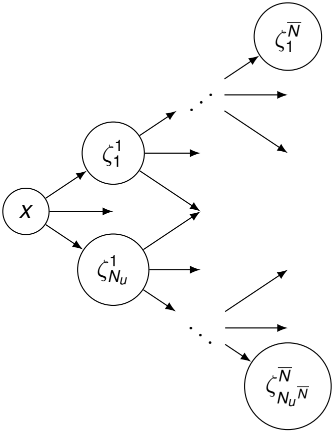
$$\mathcal{T}^1 = \{\zeta_i^1\}_i = \{x + \Delta t\, f(x, u_i, t_0)\}_i, \quad i = 1, ..., N_u$$



Tree Structure Algorithm (A., Falcone, Saluzzi, '19)

Given the nodes in the previous level, we build the following one

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



Approximation of the value function

The numerical value function $V(x,t)$ will be computed on the tree in space, while in time we consider a piecewise constant function

$$V(x,t) = V^n(x) \quad \forall x, \forall t \in [t_n, t_{n+1})$$

Computation of the value function on the tree

The tree structure defines a grid $\mathcal{T}^{\mathbf{n}} = \{\zeta_j^n\}_{j=1}^{N_u^n}$ and $n = 0, \ldots, \overline{N}$, where we can compute the numerical value function:

$$\begin{cases} V^n(\zeta_i^n) = \min_{u \in U} \{ V^{n+1}(\zeta_i^n + \Delta t\, f(\zeta_i^n, u, t_n)) + \Delta t\, L(\zeta_i^n, u, t_n) \} & \zeta_i^n \in \mathcal{T}^n \\ V^{\overline{N}}(\zeta_i^{\overline{N}}) = g(\zeta_i^{\overline{N}}) & \zeta_i^{\overline{N}} \in \mathcal{T}^{\overline{N}} \end{cases}$$

Pros vs Cons

Pros

- We do not need interpolation, the nodes $x_i + \Delta t\, f(x_i, u, t_n)$ belong to the grid by construction
- If the dynamics is autonomous, we can compute

$$V^n(\zeta),\ \forall \zeta \in \cup_{k=0}^{n} \mathcal{T}^k$$

- Mitigation of the curse of dimensionality (e.g. , $d \gg 10$)

Cons

- Still have dimensionality issues. In fact, given N_u controls and $\overline{N}$ time steps, the cardinality of the tree is $O(N_u^{\overline{N}+1})$

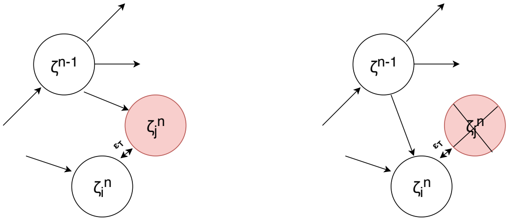
Solution: Pruning the tree

Solution: Pruning the tree

Pruning rule

Given a threshold $\varepsilon_{\mathcal{T}}$, two nodes ζ_i^n and ζ_j^n will be merged if

$$\|\zeta_i^n - \zeta_j^n\| \leq \varepsilon_{\mathcal{T}}$$



Reasonable: if $\zeta_i^n \approx \zeta_j^n \implies V^n(\zeta_i^n) \approx V^n(\zeta_j^n)$ ($V^n(x)$ is Lipschitz)

Error estimates

Theorem (Falcone, Giorgi, '99)

Let f, L and g be Lipschitz continuous and bounded, then

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

Theorem (Saluzzi, A., Falcone, '18)

Let f, L and g be Lipschitz continuous, bounded. Furthermore, let L and g be **semiconcave** *and $f \in C^1$, then*

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

Remark

One can obtain the same order of convergence in the case of the pruned tree if the pruning tolerance $\varepsilon_{\mathcal{T}}$ is chosen properly.
(e.g. $\varepsilon_{\mathcal{T}} \approx O(\Delta t^2)$ for forward Euler)

Outline

1 Introduction
- Dynamic programming principle and HJB equations
- Classical Semi-Lagrangian approach

2 Dynamic programming on a Tree Structure
- Tree Structure Algorithm
- A priori error estimates

3 HJB-POD approach on Tree Structure

4 Numerical tests

Control of Partial Differential Equations via DP

The discretization of a PDE leads to a large system of ODEs. The approximation of the correspondent HJB equations is unfeasible. **Curse of dimensionality.**

Proper Ortoghonal Decomposition (Kunisch, Hinze, Volkwein,...)

POD decomposition allows to reduce the number of variables to approximate partial differential equations.

The goal is to approximate optimal control problems in infinite dimension coupling numerical schemes for HJBs with POD techniques.
Refs: Kunisch, Volkwein and Xie (2004), A., Falcone (2013, 2014), A., Hinze (2015), A., Falcone, Kalise (2016), A. Falcone, Volkwein (2017), A. Schmidt, Haasdonk (2017)

Reduced Order Modelling Control Problem

MOR ansatz

$$y(t) \approx \Psi y^\ell(t) \quad \Psi^T \Psi = I, \qquad \Psi \in \mathbb{R}^{n\times\ell}$$

Compact Notations

$x^\ell := \Psi^T x, \quad y^\ell(t) := \Psi^T y(t)$

$f^\ell(y^\ell(t), u(t), t) := \Psi^T f(\Psi y^\ell(t), u(t), t), L^\ell(y^\ell(t), u(t)) := L(\Psi y^\ell(t), u(t)).$

$$\begin{cases} \dot{y}^\ell(t) = f^\ell(y^\ell(t), u(t)), \quad t \in [0, T], \\ y^\ell(0) = x^\ell \in \mathbb{R}^\ell. \end{cases}$$

The cost functional is:

$$J^\ell_{x^\ell}(u) = \int_0^T L^\ell(y^\ell(t), u(t), t) e^{-\lambda t}\, dt + g(y^\ell(T))$$

Proper Orthogonal Decomposition and SVD

Given **snapshots** $(y(t_0), \dots, y(t_n)) \in \mathbb{R}^m$

We look for an orthonormal basis $\{\psi_i\}_{i=1}^{\ell}$ in $\mathbb{R}^m$ with $\ell \ll \min\{n, m\}$ s.t.

$$J(\psi_1, \dots, \psi_\ell) = \sum_{j=1}^{n} \alpha_j \left\| y_j - \sum_{i=1}^{\ell} \langle y_j, \psi_i \rangle \psi_i \right\|^2 = \sum_{i=\ell+1}^{d} \sigma_i^2$$

reaches a minimum where $\{\alpha_j\}_{j=1}^{n} \in \mathbb{R}^+$.

$$\min J(\psi_1, \dots, \psi_\ell) \quad s.t. \langle \psi_i, \psi_j \rangle = \delta_{ij}$$

Singular Value Decomposition: $Y = \Psi \Sigma V^T$.

For $\ell \in \{1, \dots, d = rank(Y)\}$, $\{\psi_i\}_{i=1}^{\ell}$ are called **POD basis** of rank ℓ.

ERROR INDICATOR: $\mathcal{E}(\ell) = \dfrac{\sum_{i=1}^{\ell} \sigma_i^2}{\sum_{i=1}^{d} \sigma_i^2}$ with σ_i singular values of the SVD.

Reduced Order Modelling Control Problem

Reduced Value Function

$$v^{\ell}(x^{\ell}, t) := v(\Psi^T x, t) = \inf_{u \in \mathcal{U}_{ad}} J^{\ell}_{x^{\ell}, t}(u)$$

Reduced HJB equation

$$-\frac{\partial v^{\ell}(x^{\ell}, t)}{\partial t} + \lambda v^{\ell}(x^{\ell}, t) + \sup_{u \in U} \{-\nabla_{x^{\ell}} v^{\ell}(x^{\ell}, t) \cdot f^{\ell}(x^{\ell}, u, t) - L^{\ell}(x^{\ell}, u, t)\} = 0$$

Feedback Control

$$u^{\ell,*}(x, t) = \min_{u \in U} \{f(x, u) \cdot \nabla_{\mathbf{x}^{\ell}} \mathbf{v}^{\ell}(\mathbf{x}^{\ell}, \mathbf{t}) + L(x, u, t)\}$$

HJB-POD on a tree structure

$$
\begin{cases}
V^{n,\ell}(\zeta_i^{n,\ell}) = \min_{u\in U}\{ V^{n+1,\ell}(\zeta_i^{n,\ell} + \Delta t\, f^{\ell}(\zeta_i^{n,\ell}, u, t_n)) + \Delta t\, L^{\ell}(\zeta_i^{n,\ell}, u, t_n)\}, \\
\qquad\qquad\qquad\qquad\qquad \zeta_i^{n,\ell} \in \mathcal{T}^{n,\ell},\, n = \overline{N}-1, \dots, 0, \\
V^{\overline{N},\ell}(\zeta_i^{\overline{N},\ell}) = g^{\ell}(\zeta_i^{\overline{N},\ell}), \qquad\qquad \zeta_i^{\overline{N},\ell} \in \mathcal{T}^{\overline{N},\ell}.
\end{cases}
$$

Theorem (A., Saluzzi, 2019)

Let f, L and g be Lipschitz continuous, bounded. Moreover let L and g be semiconcave and $f \in C^1$, then there exists a constant $C(T, |x|)$ such that

$$
\sup_{s\in[0,T]} |v(x,s) - V^{\ell}(x^{\ell}, s)| \le C(T,|x|)(\|Id - \mathcal{P}^{\ell}\| + \Delta t), \quad \forall x \in \mathbb{R}^d,
$$

where $\mathcal{P}^{\ell}$ is the projection operator.

Outline

1. Introduction
   - Dynamic programming principle and HJB equations
   - Classical Semi-Lagrangian approach
2. Dynamic programming on a Tree Structure
   - Tree Structure Algorithm
   - A priori error estimates
3. HJB-POD approach on Tree Structure
4. Numerical tests

Test 1: Comparison with exact solution

We consider the following dynamics

$$f(x,u) = \begin{pmatrix} u \\ x_1^2 \end{pmatrix},\; u \in U \equiv [-1,1].$$

where $x = (x_1, x_2) \in \mathbb{R}^2$, and the following cost functional:

$$L(x,u,t) = 0\,, \qquad g(x(T)) = -x_2(T)\,, \qquad \lambda = 0\,.$$

We compare the approximations according to ℓ_2 relative error

$$\mathcal{E}_2(t_n) = \sqrt{\frac{\sum_{x_i \in \mathcal{T}^n} |v(x_i,t_n) - V^n(x_i)|^2}{\sum_{x_i \in \mathcal{T}^n} |v(x_i,t_n)|^2}} \quad Err_{2,2} = \sqrt{\Delta t \sum_{n=0}^{\overline{N}} \mathcal{E}_2^2(t_n)}$$

Test 1: Comparison with exact solution

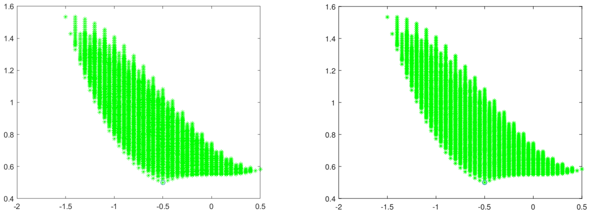


Figure: Full Tree ($|\mathcal{T}| = 2097151$) (left) and Pruned Tree with $\varepsilon_{\mathcal{T}} = \Delta t^2$ ($|\mathcal{T}| = 3151$) (right)

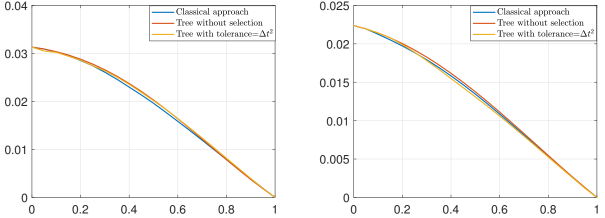


Figure: Error ℓ_2 with different initial conditions

Test 1: Comparison with exact solution

Δt	$\lvert\mathcal{T}\rvert$	CPU	$Err_{2,2}$	$Err_{\infty,2}$	$Order_{2,2}$	$Order_{\infty,2}$
0.2	63	0.05s	9.0e-02	0.122		
0.1	2047	0.35s	4.4e-02	0.062	1.04	0.98
0.05	2097151	1.1s	2.2e-02	0.031	1.02	0.99

Table: Error analysis and order of convergence of the TSA without pruning

Δt	$\lvert\mathcal{T}\rvert$	CPU	$Err_{2,2}$	$Err_{\infty,2}$	$Order_{2,2}$	$Order_{\infty,2}$
0.2	42	0.05s	9.1e-02	0.122		
0.1	324	0.08s	4.4e-02	0.062	1.05	0.98
0.05	3151	0.6s	2.1e-02	0.031	1.04	0.99
0.025	29248	2.5s	1.1e-02	0.016	1.005	0.994
0.0125	252620	150s	5.3e-03	0.008	1.004	0.997

Table: Error analysis and order of convergence of the TSA with $\varepsilon_{\mathcal{T}} = \Delta t^2$

Test 1: Comparison with exact solution

Δt	Nodes	CPU	$Err_{2,2}$	$Err_{\infty,2}$	$Order_2$	$Order_{\infty,2}$
0.2	1365	0.29s	3.5e-03	4.2e-03		
0.1	1398101	3.92s	8.6e-04	1.1e-03	2.03	1.98

Table: Table for Heun's scheme for the Full Tree

Δt	Nodes	CPU	$Err_{2,2}$	$Err_{\infty,2}$	$Order_2$	$Order_{\infty,2}$
0.2	160	0.35s	5.3e-03	7.01e-03		
0.1	2895	0.61s	8.5e-04	1.07e-03	2.65	2.71
0.05	58888	60s	2.0e-04	2.7e-04	2.11	1.99
0.025	1018012	9051s	3.9e-05	6.7e-05	2.34	2.00

Table: Table for Heun's scheme with $\varepsilon_{\mathcal{T}} = \Delta t^3$

Test 1: Comparison with exact solution

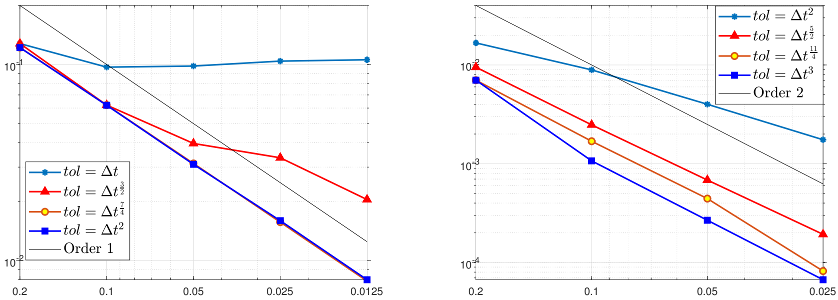


Figure: Comparison of the error $Err_{\infty,2}$ with different tolerances for Euler scheme (left) and Heun's scheme (right)

Test 2: Heat equation

We want to study the following heat equation:

$$\begin{cases} y_t = \sigma y_{xx} + y_0(x)u(t) & (x,t) \in \Omega \times [0,T], \\ y(x,t) = 0 & (x,t) \in \partial\Omega \times [0,T], \\ y(x,0) = y_0(x) & x \in \Omega\ . \end{cases}$$

and minimize the following cost functional

$$J_{y_0,t}(u) = \int_t^T \left(\delta_1 \int_\Omega |y(s,x)|^2\, dx + \gamma |u(s)|^2 \right) ds + \int_\Omega |y(T,x)|^2\, dx$$

Semi-discrete problem (System dimension $= 10^3$)

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

$\Delta x = 10^{-3}$, $\Delta t = 0.05$, $T = 1$, $\sigma = 0.1$, $\delta_1 = 1$ and $\gamma = 0.01$.

Test 2: Heat equation

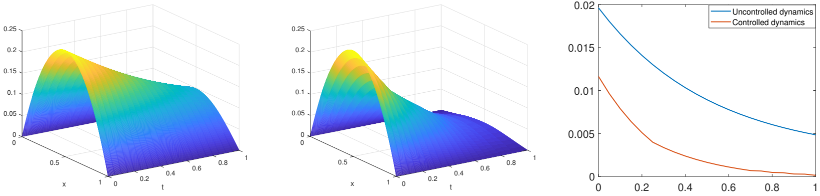


Figure: (Smooth initial condition) Uncontrolled solution (left), optimal control solution (middle), time comparison of the cost functionals (right).

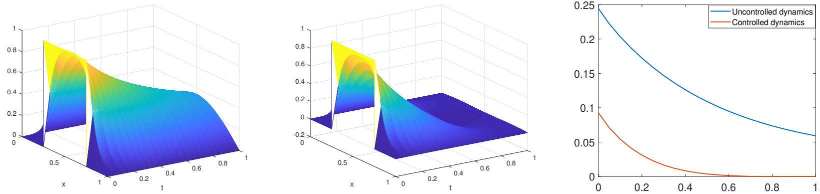


Figure: (Discontinuous initial condition) Uncontrolled solution (left), optimal control solution (middle), time comparison of the cost functionals (right).

Test 2: Heat equation

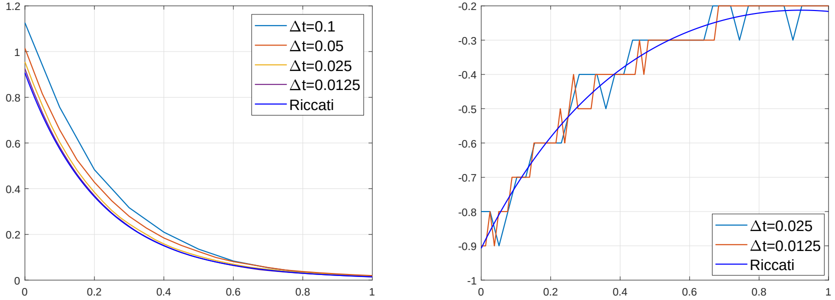


Figure: Test 2: Cost functional (left) and optimal control (right) with $N_u = 11$.

Δt	Nodes	Pruned/Full	CPU	Err_2	Err_∞	$Order_2$	$Order_\infty$
0.1	134	4.7e-09	0.14s	0.279	0.241		
0.05	863	1.2e-18	0.65s	0.144	0.118	0.95	1.03
0.025	15453	3.1e-38	12.88s	5.5e-2	5.3e-2	1.40	1.17
0.0125	849717	3.8e-78	1.1e3s	1.6e-2	1.6e-2	1.77	1.42

Test 3: 2D Reaction diffusion equation

$$\begin{cases} \partial_t y(x,t) = \sigma \Delta y(x,t) + \mu \left(y^2(x,t) - y^3(x,t) \right) + y_0(x) u(t) \\ \partial_n y(x,t) = 0 \\ y(x,0) = y_0(x) \end{cases}$$

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

POD-DEIM resolution

$T = 1$, $\sigma = 0.1$, $\mu = 5$, and $N_x = 961$.
6 POD basis to obtain a projection ratio equal to 0.9999.

Test 3: 2D Reaction diffusion equation

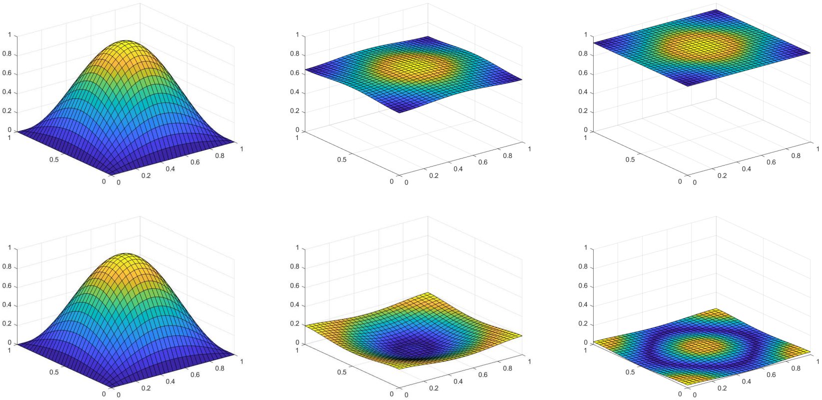


Figure: Uncontrolled solution (top) and controlled solution with full tree (bottom) for time $t = \{0, 0.5, 1\}$.

Test 3: 2D Reaction diffusion equation

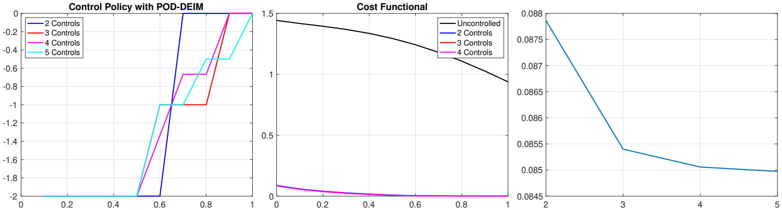


Figure: Test 1: Optimal policy (left), cost functional (middle) and $J_{y_0,0}$ (right) for U_n with $n = \{2, 3, 4, 5\}$.

	U_2	U_3	U_4	U_5
TSA-Full	5.8312s	241.5773s	3845.77s	$>$ 4 days
TSA-POD	0.5157s	19.7969s	432.0990s	1.0871$e+$04s

Table: CPU time of the TSA and the TSA-POD with a different number of controls

Test 4: Comparison MPC and TSA

$$f(x,u) = \begin{pmatrix} x_2 \\ 0.15(1 - x_1^2)x_2 - x_1 + u \end{pmatrix} \quad u \in U \equiv [-1, 0.4].$$

$$J_{x,0}(u) = \int_0^4 \left(\|y(s)\|_2^2 + \frac{1}{100}|u(s)|^2 \right) ds + \|y(4)\|_2^2.$$

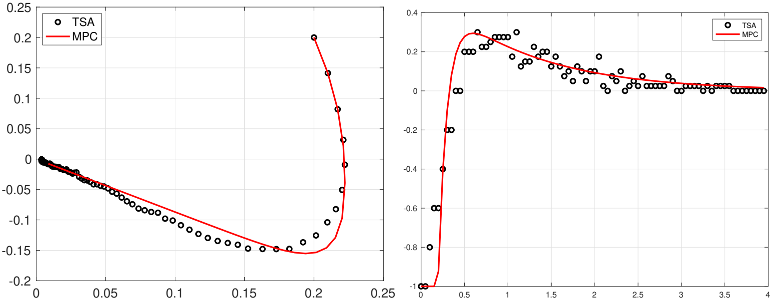


Figure: Comparison of optimal trajectory (left) and optimal control (right).

Cost MPC:0.0695 > Cost TSA: 0.0569

Conclusions and future works

Conclusions

- We presented a new algorithm to solve finite horizon optimal control problems using a **tree structure** with first order convergence.
- Introduced a pruning rule to solve dimensionality problem.
- It can be easily extended to high-order methods.
- It can be applied in general framework, with non linear dynamics and non-quadratic cost functional.
- We coupled the method with POD to obtain a more efficient algorithm.

Future works

- Stochastic extension.
- Feedback reconstruction.
- Algorithm improvements.

Thank you for your attention

1. A. Alla, M. Falcone, L. Saluzzi, *An efficient DP algorithm on a tree-structure for finite horizon optimal control problems*, SISC, 2019.
2. A. Alla, M. Falcone, L. Saluzzi, *High-order Approximation of the Finite Horizon Control Problem via a Tree Structure Algorithm*, IFAC CPDE 2019.
3. A. Alla, L. Saluzzi, *A HJB-POD approach for the control of nonlinear PDEs on a tree structure*, submitted, 2019.
4. M. Bardi, I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, Birkhäuser, Basel, 1997.
5. I. Capuzzo Dolcetta, H. Ishii, *Approximate solution of the Bellman equation of deterministic control theory*, Appl. Math. Optim., 1984.
6. M. Falcone, R. Ferretti, *Discrete time high-order schemes for viscosity solutions of Hamilton-Jacobi-Bellman equations*, Numerische Mathematik, 1994.
7. M. Falcone, 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.
8. L. Saluzzi, A. Alla, M. Falcone, *Error estimates for a tree structure algorithm on dynamic programming equations*, submitted, 2018.