POLYNOMIAL APPROXIMATION OF HIGH-DIMENSIONAL HAMILTON-JACOBI-BELLMAN EQUATIONS AND APPLICATIONS TO FEEDBACK CONTROL OF SEMILINEAR PARABOLIC PDES

DANTE KALISE* AND KARL KUNISCH†

Abstract. A procedure for the numerical approximation of high-dimensional Hamilton-Jacobi-Bellman (HJB) equations associated to optimal feedback control problems for semilinear parabolic equations is proposed. Its main ingredients are a pseudospectral collocation approximation of the PDE dynamics, and an iterative method for the nonlinear HJB equation associated to the feedback synthesis. The latter is known as the Successive Galerkin Approximation. It can also be interpreted as Newton iteration for the HJB equation. At every step, the associated linear Generalized HJB equation is approximated via a separable polynomial approximation ansatz. Stabilizing feedback controls are obtained from solutions to the HJB equations for systems of dimension up to fourteen.

Key words. Optimal Feedback Control, Hamilton-Jacobi-Bellman Equations, Nonlinear Dynamics, Polynomial Approximation, High-dimensional Approximation

AMS subject classifications. 49J20, 49LXX, 49MXX

1. Introduction.

Optimal feedback controls for evolutionary control systems are of significant practical importance. Differently from open-loop optimal controls, they do not rely on knowledge of the initial condition and they can achieve design objectives, as for instance stabilisation, also in the presence of perturbations. Furthermore, the online synthesis of feedback control can be implemented in a real-time setting. It is well-known that their construction relies on special Hamilton-Jacobi-Bellman (HJB) equations, see for instance [4, 21]. The solution of the HJB equation is the value function associated to the optimal control problem, and its gradient is used to construct the optimal feedback control. In the very special, but important case of a linear control system with quadratic cost without constraints on the control or the state variables, the HJB equation reduces to a Riccati equation which has received a tremendous amount of attention, both for the cases when the control system is related to ordinary or to partial differential equations. Otherwise one has to deal with the HJB equation which is a partial differential equation whose spatial dimension is that of the control system. Thus optimal feedback control for partial differential equations leads to HJB equations in infinite dimensions [20]. After semi-discretization in space of the controlled partial differential equation (PDE), the HJB equation is posed in a space of dimension corresponding to the spatial discretization of the PDE [22]. For standard finite element or finite difference discretizations this leads to high-dimensional HJB equations. This is one of the instances which is referred to as *the curse of dimensionality* [9].

Many attempts to tackle the difficulties posed for numerically solving the HJB equations arising in optimal control have been made in the past or are currently being investigated. We refer, for instance, to [21], which mainly focuses on semi-Lagrangian schemes, and further references given there. A related approach to numerical optimal feedback control of PDEs is to semi-discretize the dynamics and to add a model order

*Imperial College London, Department of Mathematics, South Kensington Campus, London SW7 2AZ, United Kingdom (dkaliseb@ic.ac.uk).

†University of Graz, Institute of Mathematics and Scientific Computing, Heinrichstr. 36, A-8010 Graz, Austria and Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austrian Academy of Sciences, Altenberger Straße 69, 4040 Linz, Austria (karl.kunisch@uni-graz.at).

reduction step, either with Balanced Truncation or Proper Orthogonal Decomposition, in order to reduce the dimension of the dynamics to a number that is tractable for grid-based, semi-Lagrangian schemes. This approach has been successfully explored, for instance, in [1, 32, 35] and references therein. It strongly relies on a trustworthy representation of the dynamics via low-dimensional manifolds. Such a low-dimensional representation may deteriorate when nonlinear and/or advection effects are relevant. Thus, it is important to strive for techniques, or combinations of techniques, which allow to solve higher dimensional problems.

Another direction of research evolves around generalizing the Riccati-based approach to allow for nonlinearities in the state equation. One such technique is termed state-dependent Riccati equation [19]. Here the coefficients in the 'ordinary' Riccati equation are functions of the state rather than constants as in the case of linear state equations. Another approach realizes the fact that the Riccati equation can be interpreted as the equation satisfied by the first term arising in the power series expansion of the value function, and attempts to improve by realizing also higher order terms in the expansion. These methods are succinctly explained in [7].

Yet another technique which has received a considerable amount of attention is termed Successive Galerkin Approximation. Roughly speaking, the nonlinear HJB equation associated to the continuous-time optimal control problem is solved by means of a Newton method. At each iteration, the control law is fixed. This leads to a Generalized Hamilton-Jacobi equation (GHJB) which is linear. The iteration is closed by an update of the control law based on the gradient of the value function. This method was intensively investigated in [5, 6], see also [7], and the references given in these citations. It is worth mentioning that the discrete-time counterpart of this method corresponds to the well-known policy iteration or Howards' algorithm [31, 12, 2].

The numerical examples in [5, 6, 7] do not go beyond dimension five, and most, if not all, of the published numerical results for nonlinear HJB equations do not exceed dimension eight [11, 24, 26]. An alternative sparse grid approach for high-dimensional approximation of HJB equations based on open-loop optimal control has been presented in [33], with tests up to dimension six. Numerical methods relying on tensor calculus have been shown to perform well in high-dimensional settings where the associated HJB equation is a linear PDE [41]. A key feature of these works is the use of sparse tensor products either for the construction of the basis or for the representation of the solution. This idea constitutes a cornerstone of high-dimensional approximation [10, 18], and their applicability ranges from sparse grid approximations [15], to polynomial chaos expansion [40, 29] and uncertainty quantification [28].

In the present paper, to solve optimal control problems for certain classes of semi-linear parabolic equations we shall proceed as follows. To accommodate the curse of dimensionality, the discretization of the PDE is based on a pseudospectral collocation method, allowing a higher degree of accuracy with relatively few collocation points. To solve the resulting HJB we utilize a Newton method based on the GHJB equation as described above. Next, the discretization of the GHJB equation is addressed through a Galerkin approximation with polynomial, globally supported, ansatz functions. While this mitigates the curse of dimensionality in terms of removing the mesh structure, it leads to high-dimensional integrals. We therefore resort to separable representations for the system dynamics and for the basis set of the polynomial approximation. The separability assumption reduces the computation of the Galerkin residual equation to products of one-dimensional integrals. The combination of these procedures allowed us to solve HJB equations related to nonlinear control systems up to dimension

fourteen by means of basic parallelization tools. The successful use of the Newton procedure requires to provide a feasibly initialization, i.e. a sub-optimal, stabilizing control. Since we do not consider constraints, this is not restrictive for finite horizon problem, but can be challenging for infinite horizon problems, and specifically for the stabilization problems which are considered in the present paper. In this respect we developed a continuation procedure based on the use of a discount factor. Specifically, we consider a nested iterative procedure: within the outer loop the value of a positive discount factor is driven to zero, within the inner loop the HJB equation is solved approximately for a fixed discount factor. With this approach, which, is summarized in Algorithms 1 and 2 below, we managed to solve optimal feedback stabilization problems for semilinear parabolic equations with different stability behavior of the desired steady state.

Let us give a brief outline of the paper. Section 2 sets the stage and provides the discussion of a special case to facilitate the understanding of the following material. In Section 3 the solution process of the HJB equation is detailed. In Section 4 we provide the formulas which are needed to numerically realize the discretized HJB equation after a separable basis has been chosen. Numerical experiments are documented in Section 5. There we can also find comparisons to suboptimal feedback strategies based on Riccati and asymptotic expansion techniques.

2. Infinite horizon optimal feedback control. We consider the following undiscounted infinite horizon optimal control problem:

$$\min_{u(\cdot)\in\mathcal{U}} \mathcal{J}(u(\cdot), x_0) := \int_0^\infty \ell(x(t)) + \gamma|u(t)|^2\, dt$$

subject to the nonlinear dynamical constraint

$$\dot{x}(t) = f(x(t)) + g(x)u(t)\,, \quad x(0) = x_0,$$

where we denote the state $x(t) = (x_1(t), \dots, x_d(t))^t \in \mathbb{R}^d$, the control $u(\cdot) \in \mathcal{U}$, with $\mathcal{U} = \{u(t) : \mathbb{R}_+ \to U \subset \mathbb{R}^m\}$, the state running cost $\ell(x) > 0$, and the control penalization $\gamma > 0$. Furthermore, we assume the running cost and the system dynamics $f(x) : \mathbb{R}^d \to \mathbb{R}^d$ and $g(x) : \mathbb{R}^d \to \mathbb{R}^{d\times m}$ to be $\mathcal{C}^1(\mathbb{R}^d)$. Throughout it is assumed that $f(0) = 0$ and $\ell(0) = 0$. Our focus is therefore asymptotic stabilization to the origin.

It is well-known that the optimal value function

$$V(x_0) = \inf_{u(\cdot)\in\mathcal{U}} J(u(\cdot), x_0)$$

characterizing the solution of this infinite horizon control problem is the unique viscosity solution of the Hamilton-Jacobi-Bellman equation

$$\min_{u\in U}\{DV(x)^t(f(x) + g(x)u) + \ell(x) + \gamma|u|^2\} = 0\,, \quad V(0) = 0\,, \tag{1}$$

with $DV(x) = (\partial_{x_1}V, \dots, \partial_{x_d}V)^t$. Here we follow the convention of dropping the subscript of x_0. We study this equation in the unconstrained case, i.e., $U \equiv \mathbb{R}^m$, where the explicit minimizer u^* of (1) is given by

$$u^*(x) = \operatorname*{argmin}_{u\in U}\{DV(x)^t(f(x) + gu) + \ell(x) + \gamma|u|^2\} = -\frac{1}{2\gamma}g(x)^t DV(x)\,. \tag{2}$$

note that by inserting this expression for the optimal control in (1), we obtain the equivalent HJB equation

$$DV(x)^t f(x) - \frac{1}{4\gamma} DV(x)^t g(x) g(x)^t DV(x) + \ell(x) = 0\,, \tag{3}$$

which under further assumptions can be simplified to the Riccati equation associated to linear-quadratic infinite horizon optimal feedback control.

The methodology we present in this work is applicable to systems fitting the aforedescribed setting, although for the sake of simplicity we restrict the presentation by the following choices:

(i) the control $u(t)$ is a scalar variable, i.e. $m = 1$.
(ii) the running cost $\ell(x)$ is quadratic, i.e. $x^T Q x$, with Q positive-definite,
(iii) the control term $g(x) \equiv g$ is a constant vector in $\mathbb{R}^d$.

At this point, our setting differs from the linear-quadratic case as it allows nonlinear dynamics. For the numerical scheme that we develop the following assumption is crucial:

ASSUMPTION 1. *The free dynamics $f(x) : \mathbb{R}^d \to \mathbb{R}^d, f(x) := (f_1(x), \dots, f_d(x))^t$ is a sum of separable functions in every coordinate $f_i(x)$*

$$f_i(x) = \sum_{j=1}^{n_f} \prod_{k=1}^{d} \mathcal{F}_{(i,j,k)}(x_k)\,,$$

where $\mathcal{F}(x) : \mathbb{R}^d \to \mathbb{R}^{d \times n_f \times d}$ is a tensor-valued function. In the case $g = g(x)$, then we shall also assume a similar separable structure for $g(x)$.

Separated representations are a fundamental tool for mitigating the curse of dimensionality, often leading to algorithms that scale linearly in d. Its computational efficiency depends on the number of summands or *separation rank* (n_f above). In this work the we shall assume the separated representation is exact. This can be readily checked as we consider dynamical systems arising from the application of the method of lines to semilinear parabolic PDEs. This translates into working over an ODE control system of the form $\dot{x}(t) = Ax(t) + N(x(t)) + u(t)$, where A is a linear operator, and $N(x(t))$ is a polynomial source term. Even though such a setting is quite general and covers a wide class of meaningful problems, it excludes an important set of agent-based control models where the governing dynamics depend on a metric interaction between states, i.e. $f(x) = f(\|x_i - x_j\|)$ (see [13] and references therein for control-related examples). Nonetheless, it is possible to address the problem of finding an approximate best separated representation of fixed rank n_f, but this procedure and its error analysis is beyond the scope of the present work. The interested reader can find in [10] a thorough presentation of this topic, with a concrete application to linear HJB equations in [41].

Under the framework provided by Assumption 1, the methodology can be directly applied to multidimensional control signals, non-quadratic state costs, and state dependent $g(x)$ which corresponds to bilinear control systems. The additional computational cost of addressing multidimensional control signals scales linearly with the dimension of u, whereas the computational burden associated to a bilinear control system will depend on the separability degree of $g(x)$. Non-quadratic, separable state costs can incorporated at a negligible computational cost with straightforward modifications of our setting.

2.1. Towards optimal feedback control of semilinear parabolic equations. In the following, we illustrate how the presented framework sets the grounds for a computational approach for approximate optimal feedback controllers for nonlinear PDEs. We consider the following optimal stabilization problem:

$$\min_{u(\cdot)\in L^2([0;+\infty))} \mathcal{J}(u(\cdot, X_0) := \int_0^\infty \|X(\cdot,t)\|^2_{L^2(\mathcal{I})} + |u(t)|^2 \, dt \tag{4}$$

subject to the semilinear parabolic equation

$$\begin{aligned} &\partial_t X(\xi,t) = \partial_{\xi\xi} X(\xi,t) - X(\xi,t)^3 + \chi_\omega(\xi)u(t)\,, \quad \xi \in \mathcal{I} = [-1,1]\,, t \in \mathbb{R}^+, \\ &\partial_\xi X(-1,t) = \partial_\xi X(1,t) = 0\,, \quad X(\xi,0) = X_0\,. \end{aligned} \tag{5}$$

In this case, the scalar control acts through the indicator function $\chi_\omega(\xi)$, with $\omega \subset \mathcal{I}$. At the abstract level, this corresponds to an infinite-dimensional optimal control problem. A first step towards the application of the proposed framework is the space discretization of the system dynamics, leading to finite-dimensional state space representation. The use of the pseudospectral collocation methods for parabolic equations has been studied in [37, 39], and leads to a state space representation of the form

$$\dot{X}(t) = AX(t) - X(t)^3 + Bu(t)\,,$$

where the discrete state $X(t) = (X_1(t), \ldots, X_d(t))^t \in \mathbb{R}^d$ corresponds to the approximation of $X(\xi,t)$ at d collocation points $\xi_i = -cos(\pi i/d)$, $i = 1, \ldots, d$, and X^3 is the coordinatewise power. The matrices $A \in \mathcal{M}^{d\times d}$ and $B \in \mathbb{R}^d$ are finite-dimensional approximations of the Laplacian and control operators, respectively. Such a discretization of the dynamics directly fulfills the separability required in Assumption 1, as the i-th equation of the dynamics reads

$$\dot{X}_i(t) = A_{i,1}X_1(t) + \ldots + A_{i,d}X_d(t) - X_i(t)^3 + B_i u(t)\,,$$

with a separability degree $n_f = d+1$. It is very important to note that semidiscretization in space of a wide class of time-dependent PDEs will lead to finite-dimensional state space representations of this type, thus the applicability of the presented framework is only limited by the dimensionality of the associated HJB equation. This motivates the choice of a pseudospectral collocation method for the discretization, as it is possible to obtain a meaningful representation of the dynamics with considerably fewer degrees of freedom than classical low-order schemes. However, if pseudospectral collocation is not a suitable discretization method for the dynamics, model reduction procedures such as balanced truncation, proper orthogonal decomposition, or reduced basis techniques shall also lead to separable state-space representations. Once the finite-dimensional state state space representation is obtained, we proceed to approximate the solution of the associated HJB equation (1), leading to the optimal feedback controller (2).

We now present a preview of the numerical results of the proposed approach. Further details of the numerical scheme will be developed in the forthcoming sections. The system dynamics in (5), are approximated in 12 collocation points (14 with b.c.s'), and therefore our approximation scheme seeks for a solution of a 12-dimensional HJB equation, which allows the computation of online optimal feedback

controllers. We compare our HJB-based controller (HJB) to the linear-quadratic controller (LQR) obtained by linearization of the system dynamics, and to an approximation method for the HJB equation based on power series expansion (PSE) [25, 42]. In Figure 1 we observe the basic features of the dynamics and the control schemes. The uncontrolled system dynamics (diffusion+dissipative source term) are stable, but stabilization is extremely slow. The control algorithms considerably reduce the transient phase. However, the control signals are different, and the HJB-based controller generates a feedback control with reduced overall cost (4). Observe that at the beginning of the time horizon even the signs of the LQR-, PSE-, and HJB-based controls differ.

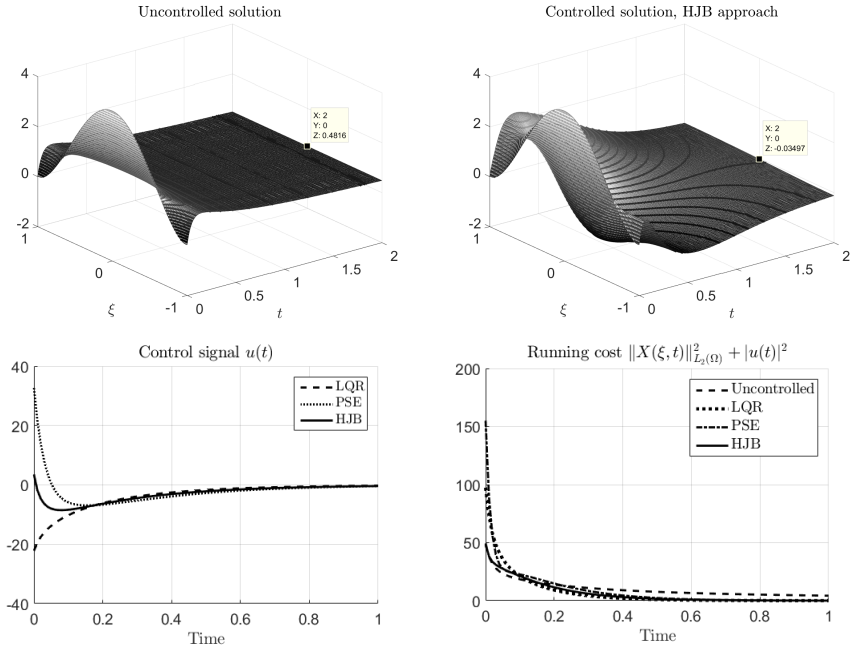


FIG. 1. *A first preview of the stabilization of the semilinear parabolic equation* (5). *Initial condition:* $X_0(\xi) = 4(\xi-1)^2(\xi+1)^2$. *Dynamics are stable but slow. Total closed-loop costs* $\mathcal{J}(u, X_0)$*:* **i)** *Uncontrolled: 13.45,* **ii)** *LQR: 7.39,* **iii)** *PSE: 9.43,* **iv)** *HJB: 6.56 .*

3. Approximate iterative solution of HJB equations. In this section, we construct a numerical scheme for the approximation of the HJB equation

$$\min_{u\in U}\{DV(x)^t(f(x)+gu)+\ell(x)+\gamma|u|^2\}=0\,,\quad V(0)=0\,, \tag{6}$$

where $U \equiv \mathbb{R}$. We recall two additional features in this equation which render the application of classical approximation techniques difficult: the absence of a variational formulation, and the minimization with respect to the control variable u, which makes the HJB equation fully nonlinear. The simplest numerical approach to these problems is the use of monotone, grid-based discretizations (finite differences, semi-Lagrangian), in conjunction with a fixed point iteration for the value function V, which typically depends on the use of a discount factor. The so-called "value iteration" procedure was first presented by Bellman in [8], and although it has become

a standard solution method for low-dimensional HJB equations, it suffers from three major drawbacks. First, the grid-based character of the scheme makes it inapplicable for high-dimensional dynamics, as the total number of degrees of freedom scales exponentially with respect to the dimension of the dynamical system. This corresponds to the most classical statement of the so-called *curse of dimensionality*. Second, the contractive mapping includes a minimization procedure which needs to be solved for every grid point at every iteration. Third, the Lipschitz constant of the contractive mapping goes to 1 when the discretization parameter goes to 0, becoming extremely slow for fine-mesh solutions. In order to circumvent these limitations, we develop a numerical scheme combining an iteration on the control variable rather than the value function, together with a polynomial expansion for the value function to mitigate the computational burden associated to mesh-based schemes.

3.1. Successive approximation of HJB equations. In the following, we revisit the method presented in [5, 6], which is referred as Successive Approximation Algorithm. We begin by defining the set of admissible controls.

DEFINITION 1 (Admissible control). *We say that a feedback mapping $u := u(x)$ is admissible on $\Omega \subset \mathbb{R}^d$, denoted as $u \in \mathcal{A}(\Omega)$, if $u(x) \in \mathcal{C}(\Omega)$, $u(0) = 0$, and $\mathcal{J}(u(x(\cdot)), x_0) < \infty$ for all $x_0 \in \Omega$.*

Starting from an admissible initial guess $u^0(x)$, the Successive Approximation Algorithm (Algorithm 1 below) generates the pair (V^*, u^*) which solves equation (6). Algorithm 1 corresponds to a Newton method for solving equation (6), and in the

Algorithm 1 Successive Approximation Algorithm

Given $u^0(x) \in \mathcal{A}(\Omega)$ and $tol > 0$
while $error > tol$ **do**
Solve

$$DV^i(x)^t(f(x) + gu^i) + \ell(x) + \gamma|u^i|^2 = 0\,, \quad V^i(0) = 0\,. \tag{7}$$

Update

$$u^{i+1}(x) = -\frac{1}{2\gamma} g^t DV^i(x)\,,$$

$error = \|V^i - V^{i-1}\|$
end while
return (V^*, u^*)

linear-quadratic setting it is equivalent to the Newton-Kleinmann iteration for solving the Riccati equation. It can be also directly identified with the policy iteration algorithm for HJB equations (see [2] and references therein), although in this context the usual setting includes a discount factor which relaxes the admissibility assumption, as well as discrete-time dynamics. Consequently, it is applied to a Bellman equation with no continuous gradient. In both cases, the core ingredient of the algorithm is to generate a decreasing sequence of values V^i by solving an associated sequence of linear problems. In our case this translates into solving, for a given $u(x)$ at each iteration,

the Generalized Hamilton-Jacobi-Bellman (GHJB) equation

$$\begin{aligned} \mathcal{G}(DV;u) =& 0\,, \quad V(0)=0\,, \\ \mathcal{G}(p,u) :=& p^t(f(x)+gu)+\ell(x)+\gamma|u|^2\,. \end{aligned} \tag{8}$$

The following result from [5] summarizes relevant properties of the GHJB equation.

PROPOSITION 2. *If Ω is a compact subset of $\mathbb{R}^d$, $f(x)$ is Lipschitz continuous on Ω and $f(0)=0$, $l(x)\geq 0$ is strictly increasing in Ω, $\gamma>0$, and $u\in\mathcal{A}(\Omega)$, then:*

1. *There exists a unique $V(x)\in C^1(\Omega)$ satisfying (8).*
2. *$V(x)$ is a Lyapunov function of the controlled system.*
3. *$V(x)=\mathcal{J}(u,x)$, for all $x\in\Omega$.*
4. *The update $u^+(x):=-\frac{1}{2\gamma}g^tDV(x)$ satisfies $u^+\in\mathcal{A}(\Omega)$.*
5. *If V^+ satisfies $\mathcal{G}(DV^+;u^+)=0$, then $V^+\leq V$ for all $x\in\Omega$.*

3.2. A continuation procedure. A critical aspect of the Successive Approximation Algorithm 1 is its initialization, which requires the existence of an admissible control $u^0(x)$ which in view of (4) means that it asymptotically stabilizes all the initial conditions in Ω. For asymptotically stable dynamics, this is trivially satisfied by $u^0(x)=0$. For more general cases, the computation of stabilizing feedback controllers is a challenging task. A partial answer is to consider the stabilizing feedback associated to the linearized system dynamics. However, this feedback is only locally stabilizing, and therefore the identification of a suitable domain Ω where this control law is admissible becomes relevant. For low dimensional dynamics, this has been studied in the context of Zubov's method in [17]. An alternative solution that we propose is to consider a discounted infinite horizon control problem

$$\min_{u(\cdot)\in\mathcal{U}} \mathcal{J}(u(\cdot),x_0) := \int_0^\infty e^{-\lambda t}\left(\ell(x(t))+\gamma|u(t)|^2\right)dt\,, \qquad \lambda>0\,,$$

where the inclusion of the discount factor λ relaxes the admissibility condition. Recently, in [23, 38], the link between discounted optimal control and asymptotic stabilization has been discussed, and under certain conditions, the discounted control problem can generate optimal controls that are also admissible for the undiscounted problem. We recall that the associated HJB equation for the infinite horizon optimal control problem is given by

$$\lambda V(x)+\min_{u\in U}\{DV(x)^t(f(x)+gu)+\ell(x)+\gamma|u|^2\}=0\,, \quad V(0)=0\,, \tag{9}$$

and the associated GHJB reads

$$\begin{aligned} \mathcal{G}_\lambda(V,DV;u) =& 0\,, \quad V(0)=0\,, \\ \mathcal{G}_\lambda(q,p,u) :=& \lambda q + p^t(f(x)+gu)+\ell(x)+\gamma|u|^2\,. \end{aligned} \tag{10}$$

We consequently modify the Successive Approximation Algorithm in order to embed it within a path-following iteration with respect to the discount factor:

Algorithm 2 A Discounted Path-Following Approximation Algorithm

Given $\lambda > 0$, $\epsilon > 0$, and $\beta \in (0,1)$,
while $\lambda > \epsilon$ **do**
Solve for (V, u)

$$\lambda V(x) + \min_{u \in U}\{DV(x)^t(f(x) + gu) + \ell(x) + \gamma|u|^2\} = 0\,, \tag{11}$$

with Algorithm 1 and initial guess u^0.
Update

$$u^0 = u\,,$$
$$\lambda = \beta\lambda\,.$$

end while
return (V^*, u^*)

For a sufficiently large λ, this algorithm can be initialized with $u^0_\lambda = 0$. Continued reduction of the discount factor using hotstart every time when (11) is called with a reduced λ-value, leads to an approximate solution of equation (6).

3.3. Spectral element approximation of the GHJB equation. So far we have discussed the iterative aspects of a computational method for solving HJB equations. We now address the numerical approximation of the GHJB equation.

$$\mathcal{G}_\lambda(V, DV; u) = 0\,, \quad V(0) = 0\,. \tag{12}$$

For this purpose, we consider an expansion $V_n(x)$ of the form

$$V_n(x) = \sum_{j=1}^{n} c_j \phi_j(x) \equiv \Phi_n \mathbf{c}\,,$$

where $\Phi_n := (\phi_1(x), \ldots, \phi_n(x))$, with $\phi_j \in C^\infty(\Omega, \mathbb{R})$ belonging to a complete set of basis functions in $L^2(\Omega, \mathbb{R})$, and $\mathbf{c} = (c_1, \ldots, c_n)^t$. In particular, we shall often generate Φ_n from a multidimensional monomial basis as illustrated in Figure 2, which directly satisfies the boundary condition $V_n(0) = 0$. The coefficients c_j are obtained by imposing the Galerkin residual equation

$$\langle \mathcal{G}_\lambda(V_n, DV_n; u), \phi_i \rangle_{L^2(\Omega)} = 0\,, \quad \forall \phi_i \in \Phi_n\,. \tag{13}$$

REMARK 3. *The convergence of V_n has been studied thoroughly in [5]. It follows a power series argument, and requires conditions for uniform convergence of pointwise convergent series, in order to guarantee that $u_n := -\frac{1}{2}\gamma^{-1} g^t DV_n(x) \in \mathcal{A}(\Omega)$ for n sufficiently large. In our particular case, we further assume that the dynamics (f, g) are polynomial (as illustrated in Section 2.1). Therefore, under the assumptions of Theorem 26 in [5], by choosing a multidimensional monomial basis (of degree ≥ 2) and an admissible control $u^0 \in \mathcal{A}(\Omega)$, it can be established that, $\forall \epsilon > 0$, $\exists K$ such that for $n > K$, $\|V - V_n\|_{L^2(\Omega)} < \epsilon$, and $u_n(x) \in \mathcal{A}(\Omega)$.*

We now focus on the different terms involved in the approximation of the GHJB equation. Since this equation is meant to be solved within the iterative loop described in the previous section, we assume that $u(x)$ can be expressed in the form

$$u(x) = -\frac{1}{2}\gamma^{-1} g^t DV_n^0(x)\,, \tag{14}$$

where $V^0(x)$ corresponds to the value function of the previous iteration, approximated with the expansion

$$V_n^0(x) = \sum_{j=1}^{n} c_j^0 \phi_j(x).$$

Below we shall write $\mathbf{c}^0$ for $(c_1^0, \ldots, c_n^0)^t$. We proceed by expanding case by case the different terms of the Galerkin residual equation

$$\langle \lambda V_n + DV_n^t(f(x) + gu) + \ell(x) + \gamma|u|^2, \phi_i \rangle_{L^2(\Omega)} = 0\,, \quad \forall \phi_i \in \Phi_n\,. \tag{15}$$

1) $\boldsymbol{\langle \lambda V_n, \phi_i \rangle_{L^2(\Omega)}}$: it is directly verifiable that

$$\langle \lambda V_n, \phi_i \rangle_{L^2(\Omega)} = \mathbf{M}_{(i,\bullet)}\mathbf{c}\,, \quad \mathbf{M} \in \mathbb{R}^{n\times n}\,, \quad \mathbf{M}_{(i,j)} = \lambda \langle \phi_i, \phi_j \rangle_{L^2(\Omega)}\,.$$

2) $\boldsymbol{\langle DV_n^t f, \phi_i \rangle_{L^2(\Omega)}}$: by inserting the expansion we obtain

$$DV_n^t f = \sum_{j=1}^{n} c_j D\phi_j^t f\,,$$

and therefore

$$\langle DV_n^t f, \phi_i \rangle_{L^2(\Omega)} = \mathbf{F}_{(i,\bullet)}\mathbf{c}\,, \quad \mathbf{F} \in \mathbb{R}^{n\times n}\,, \quad \mathbf{F}_{(i,j)} := \langle D\phi_j^t f, \phi_i \rangle_{L^2(\Omega)}\,.$$

3) $\boldsymbol{\langle DV_n^t gu, \phi_i \rangle_{L^2(\Omega)}}$: the relation (14) leads to

$$DV_n^t gu = DV_n^t \left(-\frac{1}{2}\gamma^{-1} g g^t DV_n^0 \right) = -\frac{1}{2}\gamma^{-1} \sum_{j=1}^{n} c_j D\phi_j^t \left(g g^t \sum_{k=1}^{n} c_k^0 D\phi_k \right)^t\,,$$

such that

$$\langle DV_n^t gu, \phi_i \rangle_{L^2(\Omega)} = \mathbf{G}_{(i,\bullet)}\mathbf{c}\,, \quad \mathbf{G} \in \mathbb{R}^{n\times n}\,,$$
$$\mathbf{G}_{(i,j)} = -\frac{1}{2}\gamma^{-1} \sum_{k=1}^{n} c_k^0 \langle g^t D\phi_k D\phi_j^t g, \phi_i \rangle_{L^2(\Omega)}\,.$$

4) $\boldsymbol{\langle l(x), \phi_i \rangle_{L^2(\Omega)}}$: we further assume that

$$\langle l(x), \phi_i \rangle_{L^2(\Omega)} = \langle x^t Q x, \phi_i \rangle_{L^2(\Omega)}\,, \qquad Q > 0 \in \mathbb{R}^{d\times d}\,.$$

5) $\boldsymbol{\langle \gamma|u|^2, \phi_i \rangle_{L^2(\Omega)}}$: note that

$$\gamma|u|^2 = \frac{1}{4}\gamma^{-1}(g^t DV_n^0)^2 = \frac{1}{4}\gamma^{-1} \left(\sum_{j=1}^{n} c_j^0 g^t D\phi_j \right)^2\,,$$

leading to

$$\langle \gamma|u|^2, \phi_i \rangle_{L^2(\Omega)} = (\mathbf{c}^0)^t \mathbf{U}_{(i,\bullet)}\mathbf{c}^0\,,$$

$\mathbf{U} \in \mathbb{R}^{n\times n\times n}$ is given by

$$\mathbf{U}_{(i,j,k)} = \langle (g^t D\phi_j)(g^t D\phi_k), \phi_i \rangle_{L^2(\Omega)}\,.$$

After discretization, the GHJB (13) reduces to a parameter-dependent linear system for $\mathbf{c}$

$$\left(\mathbf{M} + \mathbf{F} + \mathbf{G}(\mathbf{c}^0)\right)\mathbf{c} = \mathbf{b}(\mathbf{U}, \mathbf{c}^0)\,,$$

where $\mathbf{b}$ is given by the expansion of $l(x) + \gamma|u|^2$ (terms **4)** and **5)** in the list above).

4. Computation of integrals via separable expansions. Under Assumption 1 concerning the separability of the free dynamics f, and with the construction of a separable set of basis functions by taking the tensor product of one-dimensional basis functions as shown in Figure 2, the calculation of the d-dimensional inner products of the Galerkin residual equation of the previous section is reduced to the product of one-dimensional integrals. In the following, we provide further details of this procedure.

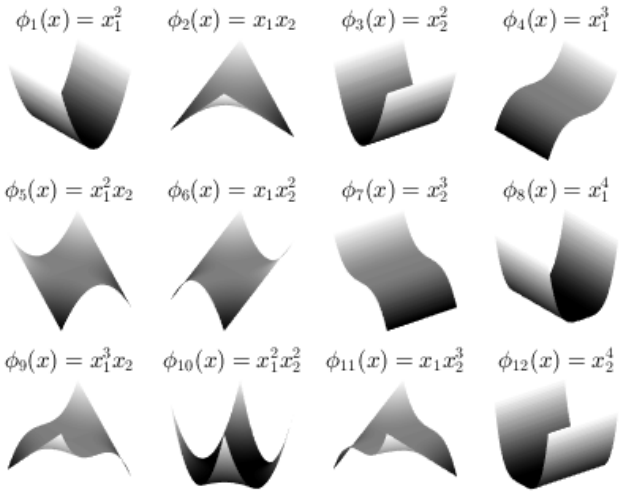


FIG. 2. *Two dimensional monomial basis. The first three basis functions correspond to the terms of the Riccati ansatz for the linear-quadratic control problems, where the value function is known to be a quadratic form $x^t\Pi x$. Adding terms of higher order allows a more accurate solution for nonlinear control problems. We construct the high-order terms by limiting the degree of the monomials.*

4.1. Generation of a multi-dimensional basis. The multi-dimensional basis functions
$\Phi_n := (\phi_1(x), \dots, \phi_n(x))$ for the expansion of V_n are generated as follows. We start by choosing a polynomial degree $M \in \mathbb{N}$, and a one-dimensional polynomial basis $\varphi_M : \mathbb{R} \to \mathbb{R}^M$. For the sake of simplicity, we consider the monomial basis $\varphi_M = (1, x, \dots, x^M)^T$, but the same ideas apply for other basis, such as orthogonal polynomials. The multidimensional basis is generated as a subset of the d-dimensional tensor product of one-dimensional basis, such that

$$\Phi_n \equiv \left\{ \phi \in \bigotimes_{i=1}^{d} \varphi_M(x_i)\,, \text{ and } deg(\phi) \leq M \right\}$$

i.e., we construct a full multidimensional tensorial basis and then we remove elements according to the approximation degree M. The elimination step is fundamental and is twofold. If no elimination is performed, the cardinality of Φ_n would be M^d, and again one would face the *curse of dimensionality* that also affects grid-based schemes. By reducing the set to multdimensional monomials of degree at most M, the cardinality

n of the set Φ_n is given by

$$n = \sum_{m=1}^{M} \left(\begin{array}{c} d+m-1 \\ m \end{array} \right) , \tag{16}$$

which replaces the exponential dependence on d by a combinatorial one. This formula is evaluated in Table 1 for different values of interest for M and d. By considering globally defined polynomial basis functions, the dependence on the dimension is replaced by the combinatorial expression (16). The dimensional reduction of the basis is particularly significant for low order polynomial approximation (up to degree 6). A second justification for the way in which we generate the basis set has a control-theoretical inspiration. A well-known result in optimal feedback control is that if the dynamics are linear, and the running cost is quadratic, the value function associated to the infinite horizon control problem (in the unconstrained case and other technical assumptions) is a quadratic form, i.e. is of the form $V(x) = x^t \Pi x$, which fits precisely the elements generated for Φ_n with a monomial basis when $M = 2$ and linear elements are eliminated. Therefore, our basis can be interpreted as a controlled increment, accounting for the nonlinear dynamics, of the basis required to recover the solution of the control problem associated to the linearized dynamics around the equilibrium point.

	Full monomial basis				Even-degree monomials			
$d \backslash M$	2	4	6	8	2	4	6	8
6	27	209	923	3002	21	147	609	1896
8	44	494	3002	12869	36	366	2082	8517
10	65	1000	8007	43757	55	770	5775	30085
12	90	1819	18563	125969	78	1443	13819	89401
14	119	3059	38759	319769	105	2485	29617	233107

TABLE 1

Number of elements n in the basis, as a function of the dimension d and the total polynomial degree M. The global polynomial approximation partially circumvents the curse of dimensionality, as the dimension of the basis no longer depends exponentially on the dimension, but rather combinatorially.

REMARK 4. *Theorem 7.1 in [6] states parity conditions to reduce the polynomial basis Φ_n. Under the assumptions $l(x) = x^t Q x$, and $g \in \mathbb{R}^d$, if*

i) Ω is a symmetric rectangle around the origin, i.e., $\Omega = [-l_1, l_1] \times \ldots \times [-l_d, l_d]$,

ii) the free dynamics are odd-symmetric on Ω, i.e. $f(-x) = -f(x)$, for all $x \in \Omega$,

then $V_n(x)$ is an even-symmetric function, i.e., $V_n(-x) = V_n(x)$, and therefore odd-degree monomials are excluded from the basis. A direct corollary is that in the linear quadratic case, where the linear dynamics are trivially odd-symmetric, $V(x)$ is a quadratic form.

Finally, for the calculation presented in the following, it is important to note that due to the construction procedure, the basis elements directly admit a separable representation

$$\phi_i(x) = \prod_{j=1}^{d} \phi_i^j(x_j) = \prod_{j=1}^{d} x_j^{\nu_j} , \quad \text{with} \ \sum_j \nu_j \leq M , \tag{17}$$

where each component $\phi_i^j(x) \in \varphi_M$.

4.2. High-dimensional integration. We begin by recalling that

$$f_i(x) = \sum_{j=1}^{n_f} \prod_{k=1}^{d} \mathcal{F}_{(i,j,k)}(x_k)\,, \tag{18}$$

where $\mathcal{F}(x) : \mathbb{R}^d \to \mathbb{R}^{d\times n_f \times d}$ is a tensor-valued function, and that $g \in \mathbb{R}^d$.

As in the previous section, we proceed term by term, to obtain the summands in (15). The integration is carried over the hyperrectangle $\Omega = \Omega_1 \times \ldots \times \Omega_d$.

1) $\boldsymbol{\langle \lambda V_n, \phi_i \rangle_{L^2(\Omega)}}$: this term is directly assembled from the calculation of

$$\langle \phi_i, \phi_j \rangle_{L^2(\Omega)} = \prod_{k=1}^{d} \int_{\Omega_k} \phi_i^k(x_k)\phi_j^k(x_k)\,dx_k$$

2) $\boldsymbol{\langle DV_n^t f, \phi_i \rangle_{L^2(\Omega)}}$: This term involves the calculation of

$$\langle D\phi_j^t f, \phi_i \rangle_{L^2(\Omega)} = \sum_{p=1}^{d} \langle f_p \partial_{x_p}\phi_j, \phi_i \rangle_{L^2(\Omega)}\,.$$

which is expanded by using the separable structure of the free dynamics

$$\langle f_p \partial_{x_p}\phi_j, \phi_i \rangle_{L^2(\Omega)} = \sum_{l=1}^{n_f} \langle \left(\prod_{m=1}^{d} \mathcal{F}(p,l,m) \right) \partial_{x_p}\phi_j, \phi_i \rangle_{L^2(\Omega)}\,,$$

where

$$\begin{aligned}
&\langle \left(\prod_{m=1}^{d} \mathcal{F}(p,l,m) \right) \partial_{x_p}\phi_j, \phi_i \rangle_{L^2(\Omega)} \\
&= \left(\prod_{\substack{m=1 \\ m\neq p}}^{d} \int_{\Omega_m} \mathcal{F}(p,l,m)\phi_i^m \phi_j^m(x_m)\,dx_m \right) \left(\int_{\Omega_p} \mathcal{F}(p,l,p)\phi_i^p \partial_{x_p}\phi_j^p(x_p)\,dx_p \right)
\end{aligned}$$

3) $\boldsymbol{\langle DV_n^t g u, \phi_i \rangle_{L^2(\Omega)}}$: In this case, we need to work on the expression

$$\langle g^t D\phi_k D\phi_j^t g, \phi_i \rangle_{L^2(\Omega)} = \sum_{l,m=1}^{d} g_l g_m \langle \partial_{x_l}\phi_k \partial_{x_m}\phi_j, \phi_i \rangle_{L^2(\Omega)},$$

which is obtained directly from the computations for $\boldsymbol{\langle \gamma |u|^2, \phi_i \rangle_{L^2(\Omega)}}$ in **5)** below.

4) $\boldsymbol{\langle l(x), \phi_i \rangle_{L^2(\Omega)}}$:

$$\langle l(x), \phi_i \rangle_{L^2(\Omega)} = \langle x^t Q x, \phi_i \rangle_{L^2(\Omega)} = \sum_{j,k=1}^{d} Q_{(j,k)} \langle x_j x_k, \phi_i \rangle_{L^2(\Omega)}\,,$$

where with a similar argument as in the previous term we expand

$$\langle x_j x_k, \phi_i \rangle_{L^2(\Omega)} = \left(\prod_{\substack{p=1 \\ p\neq j \\ p\neq k}}^{d} \int_{\Omega_p} \phi_i^p(x_p)\,dx_p \right) \left(\int_{\Omega_j} \phi_i^j(x_j)x_j\,dx_j \right) \left(\int_{\Omega_k} \phi_i^k(x_k)x_k\,dx_k \right)\,.$$

5) $\langle \gamma|u|^2, \phi_i\rangle_{L^2(\Omega)}$: This term requires the computation of the inner product

$$\langle (g^t D\phi_j)(g^t D\phi_k), \phi_i\rangle_{L^2(\Omega)} = g^t \tilde{\mathbf{U}}_{(\mathcal{I},\bullet)} g\,, \quad \mathcal{I} = (i,j,k)\,,$$

with $\tilde{\mathbf{U}} \in \mathbb{R}^{n\times n\times n\times d\times d}$ given by

$$\tilde{\mathbf{U}}_{(\mathcal{I},l,m)} := \langle \partial_{x_l}\phi_j \partial_{x_m}\phi_k, \phi_i\rangle_{L^2(\Omega)}\,.$$

By using the separable representation of the basis functions

$$\partial_{x_l}\phi_j = \left(\prod_{\substack{p=1\\ p\neq l}}^{d} \phi_j^p(x_p)\partial_{x_l}\right)\phi_j^l(x_l)$$

we expand the inner product

$$\tilde{\mathbf{U}}_{(\mathcal{I},l,m)} = \left(\prod_{\substack{p=1\\ p\neq l\\ p\neq m}}^{d} \int_{\Omega_p} \phi_i^p\phi_j^p\phi_k^p(x_p)\,dx_p\right)\left(\int_{\Omega_l} \phi_i^l\phi_k^l\partial_{x_l}\phi_j^l(x_l)\,dx_l\right)\dots \tag{19}$$
$$\dots\left(\int_{\Omega_m} \phi_i^m\phi_j^m\partial_{x_m}\phi_k^m(x_m)\,dx_m\right)\,.$$

Initialization. The first iteration, with a stabilizing initial guess u^0, requires special attention. If it is obtained via a Riccati-type argument, then initialization follows directly from (14). Otherwise we shall relax this requirement, and only assume that the initial stabilizing controller is given in separable form

$$u^0(x) = \sum_{j=1}^{n_u}\prod_{k=1}^{d} \mathcal{U}^0_{(j,k)}(x_k)\,,$$

In this case, we must recompute the term:

- $\langle \gamma|u^0|^2, \phi_i\rangle_{L^2(\Omega)}$

$$\begin{aligned}\langle \gamma|u^0|^2, \phi_i\rangle_{L^2(\Omega)} &= \gamma\langle (\sum_{j=1}^{n_u}\prod_{k=1}^{d}\mathcal{U}^0_{(j,k)}(x_k))^2, \phi_i\rangle\\ &= \gamma\sum_{j,l=1}^{n_u}\langle\left(\prod_{k=1}^{d}\mathcal{U}^0(j,k)\right)\left(\prod_{k=1}^{d}\mathcal{U}^0(l,k)\right), \phi_i\rangle\,,\\ &= \gamma\sum_{j,l=1}^{n_u}\prod_{k=1}^{d}\int_{\Omega_k}\mathcal{U}^0(j,k)\mathcal{U}^0(l,k)\phi_i^k(x_k)\,dx_k\,.\end{aligned}$$

As for the term $\langle DV_n^t g u^0, \phi_i\rangle_{L^2(\Omega)}$, which needs to be computed differently in the first iteration, we can proceed in the same way as for $\langle DV_n^t f, \phi_i\rangle_{L^2(\Omega)}$, since both gu^0 and f have the same separable structure, it just takes to assign $f_i = g_i u^0$.

4.3. Computational complexity and implementation. Among the expressions developed in the previous subsection, the overall computational burden is governed by the approximation of

$$\langle \gamma |u|^2, \phi_i \rangle_{L^2(\Omega)} \,,$$

which requires the assembly of the 5-dimensional tensor $\tilde{\mathbf{U}} \in \mathbb{R}^{n\times n\times n\times d\times d}$. Each entry of this tensor is a d-dimensional inner product, which under the aforementioned separability assumptions is computed as the product of d, one-dimensional integrals. Thus, the total amount of one-dimensional integrals required for the proposed implementation is $O(n^3 d^3)$. A positive aspect of our approach is that the assembly of tensors like $\tilde{\mathbf{U}}$ falls within the category of *embarrassingly parallelizable* computations, so the CPU time scales down almost directly with respect to the number of available cores. Furthermore, $\tilde{\mathbf{U}}$ can be entirely computed in an offline phase, before entering the iterative loops in Algorithms 1 and 2. However, for values of interest of n and d, such as $d > 10$ and $n = 4$, Table 1 indicates that $n^3 d^3$ is indeed a very large number. A rough estimate of the CPU time required for the assembly of $\tilde{\mathbf{U}}$ is given by

$$\mathrm{CPU}(\tilde{\mathbf{U}}) = \frac{t_{1d} \times n^3 \times d^3}{\#cores} \,,$$

where t_{1d} corresponds to the time required for the computation of a one-dimensional integral. Therefore, it is fundamental for an efficient implementation to reduce t_{1d} to a bare minimum. From closer inspection of the expression (19), we observe that all the terms can be identified as elements of the tensors $\mathcal{M}, \mathcal{K} \in \mathbb{R}^{M\times M\times M}$

$$\mathcal{M}_{(i,j,k)} := \int_{x_l}^{x_u} \varphi_i(x)\varphi_j(x)\varphi_k(x)\, dx \,, \quad \mathcal{K}_{(i,j,k)} := \int_{x_l}^{x_u} \varphi_i(x)\varphi_j(x)\partial_x\varphi_k(x)\, dx \,.$$

Both $\mathcal{M}$ and $\mathcal{K}$ can be computed exactly with a Computer Algebra System, or approximated under suitable quadrature rules. We follow this latter approach, implementing an 8-point Gauss-Legendre quadrature rule. After having computed $\mathcal{M}$ and $\mathcal{K}$, the assembly of (19) reduces to d calls to properly indexed elements of these tensors. This approach requires a careful bookkeeping of the separable components of each multi-dimensional basis function ϕ_i. In this way, an entry of $\tilde{\mathbf{U}}$ takes of the order of 10^{-7} seconds and the overall CPU time is kept within hours for problems of dimension up to 12.

5. Computational implementation and numerical tests.

5.1. Convergence of the polynomial approximation. We assess the convergence of the polynomial approximation in a 1D test, with

$$f(x) = 0 \,, \quad g = 1 \,, \quad l(x) = \frac{1}{4R}\left(x^2\, \mathrm{e}^x + 2\, x\, \mathrm{e}^x + 4\, x^3\right)^2 \,, \quad \Omega = (-1, 1) \,,$$

such that the exact solution of equation (3) is given by

$$V(x) = x^4 + x^2 e^x \,.$$

We implement the path-following version (Algorithm 2), starting with $u^0 = 0$, $\lambda = 1$ and a threshold value $\epsilon = 1 \times 10^{-6}$, a parameter $\beta = 0.5$, and an internal tolerance $tol = 10^{-8}$. The relative error for Table 2 is defined as

$$\text{error} := \frac{\|V_n(x) - V(x)\|_{L^2(\Omega)}}{\|V(x)\|_{L^2(\Omega)}}$$

and number of iterations for different polynomial degree approximations are shown in Table 2 and Figure 3.

	Monomial basis		Legendre basis	
n(degree)	error	iterations	error	iterations
2	1.1539	53	1.4127	52
4	0.2541	49	0.3643	58
6	0.015	52	0.0206	52
8	5.01×10^{-4}	55	6.41×10^{-4}	53
10	8.33×10^{-6}	55	1.072×10^{-5}	55

TABLE 2

1D polynomial approximation of the infinite horizon control problem with nonquadratic running cost. The number n denotes the total number of basis functions.

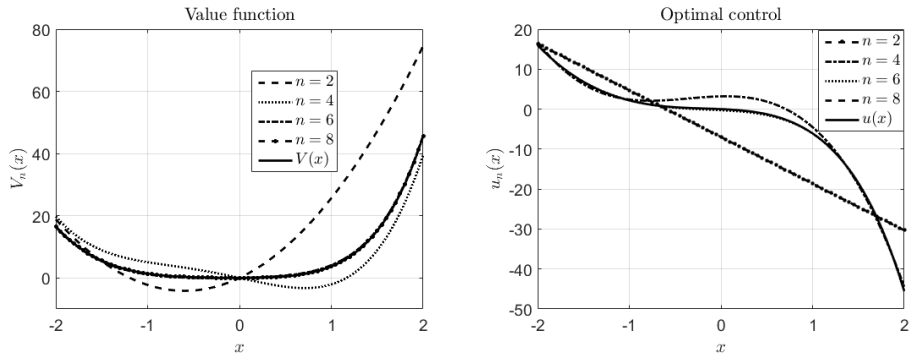


FIG. 3. *1D polynomial approximation of the infinite horizon control problem with nonquadratic running cost. Approximation with monomial basis. The number n denotes the total number of basis functions.*

5.2. Optimal feedback control of semilinear parabolic equations. Similarly as in Section 2.1, we consider the following optimal control problem

$$\min_{u(\cdot)\in L^2([0;+\infty))} \mathcal{J}(u(\cdot),X_0) := \int_0^\infty \|X(\xi,t)\|^2_{L^2(\mathcal{I})} + \gamma u(t)^2\,dt\,, \tag{20}$$

subject to the semilinear dynamics

$$\begin{aligned} \partial_t X(\xi,t) &= \mathcal{L}(X,X_{\xi\xi}) + \mathcal{N}(X,\partial_\xi X) + \chi_\omega(\xi)u(t)\,, \qquad \text{in } \mathcal{I}\times\mathbb{R}^+\,,\\ X(\xi,0) &= X_0(\xi)\,, \quad \xi\in\mathcal{I}\,, \end{aligned}$$

where the linear operator $\mathcal{L}$ is of the form $\mathcal{L} := \sigma\partial_{\xi\xi}X(\xi,t) + rX(\xi,t)$ with $r\in\mathbb{R}$, and $\mathcal{N}$ is a nonlinear operator such that $\mathcal{N}(0,0)=0$. The scalar control acts through the indicator function $\chi_\omega(\xi)$, with $\omega\subset\mathcal{I}$ The system is closed under suitable boundary conditions. We choose $\mathcal{I}=(-1,1)$, $\omega=(-0.5,-0.2)$, $\sigma=0.2$, and $\gamma=0.1$. The

nonlinearity covers both advective, Burgers'-type, and polynomial source terms. In order to generate a low-dimensional state space representation of the dynamics, we resort to a pseudospectral collocation method with Chebyshev polynomials as in [37] (for further details we also refer to [39, p. 107]. By considering d collocation points $\xi_i = -cos(\pi i/d)\,, i = 1, \ldots, d$, the continuous state $X(\xi, t)$ is discretized into $X(t) = (X_1(t), \ldots, X_d(t))^t \in \mathbb{R}^d$, where $X_i(t) = X(\xi_i, t)$. The semilinar PDE dynamics are thus approximated by the $d-$dimensional nonlinear ODE system

$$\dot{X}(t) = AX(t) + N(X(t)) + Bu(t)\,, \tag{21}$$

where the operators (A, N, B) correspond to the finite-dimensional realization of $(\mathcal{L}, \mathcal{N}, \chi_\omega(\xi))$ through the Chebyshev pseudospectral method. Therefore, the number of collocation points governs the dimension of the resulting nonlinear ODE system (21), and consequently determines the dimension of the domain Ω where the associated HJB equation is solved. In the following, Tests 1-3 are computed in 14 collocation points, which after including boundary conditions lead to a 12 dimensional domain Ω for the HJB equation. Test 4 is solved in 14 dimensions. The high-dimensional solver was implemented in MATLAB, parallelizing the tensors assembly, and tests were run on a muti-core architecture 8x Intel Xeon E7-4870 with 2,4Ghz, 1 TB of RAM. The MATLAB pseudoparallelization distributes the tasks among 20 workers. Representative performance details are shown in Table 3. The assembly of high-dimensional tensor that enter the iterative algorithm accounts for over 80% of the total CPU time. This percentage increases when Algorithm 1 is implemented for asymptotically stable dynamics, as it requires a much lower number of iterations. Note that much of the work done during the assembly phase is independent of the dynamics (see for instance (19)), and therefore can be re-used in latter problems, mitigating the overall computational burden.

Test	Dimension	CPU-assembly	CPU-iterative (#)
1	10	2.061×10^3[s]	4.221×10^2[s](32)
1	12	1.945×10^4[s]	3.377×10^3[s](32)
4	14	1.557×10^5[s]	3.102×10^4[s](37)

TABLE 3

CPU times for different tests and dimensions. CPU-assembly corresponds to the amount of time spent in offline assembly of the different terms of the Galerkin residual equation (13). *CPU-iterative refers to the amount of time spent inside Algorithm 2.*

We now turn to the specification of parameters for the solution of the HJB equation. We set $\Omega = (-2, 2)^d$, and consider a monomial basis up to order 4 as described in Section 4. Depending on the dynamics of every example, we will neglect odd-degree basis functions as in Remark 4. All the integrals are approximated with an 8 point Gauss-Legendre quadrature rule. Whenever system dynamics are stable at the origin, the value function is obtained from the undiscounted Algorithm 1, initialized with $u^0 = 0$. When the dynamics are unstable over Ω, we implement Algorithm 2, with $\lambda = 1$, $\epsilon = 10^{-6}$, and $\beta = 0.9$. The initializing controller is given by the solution of the associated linear-quadratic optimal feedback, as described below. For both implementations, the tolerance of the algorithm is set to $tol = 10^{-8}$. In the following

tests, we compare the HJB-based feedback control with respect to the uncontrolled dynamics ($u = 0$), the linear-quadratic optimal feedback (LQR), and the power series expansion type of controller (PSE). We briefly describe these controllers. The well-known LQR feedback controller corresponds to the HJB synthesis applied over the linearized system around the origin

$$\dot{X}(t) = AX(t) + Bu(t)\,, \tag{22}$$

and results in the optimal feedback control law given by

$$u^* = -\gamma^{-1} B^t \Pi X\,,$$

where $\Pi \in R^{d\times d}$ is the unique self-adjoint, positive-definite solution of the algebraic Riccati equation

$$A^t\Pi + \Pi A - \Pi B\gamma^{-1}B^t\Pi + Q = 0\,,$$

and X^tQX corresponds to the finite-dimensional approximation of $\|X(\xi,t)\|_{L^2(\mathcal{I})}$. Once this controller has been computed, the high-order PSE feedback is obtained as

$$u^* = -\gamma^{-1}B^t(\Pi X - (A^t - \Pi B\gamma^{-1}B^t)^{-1}\Pi N_l(X))\,,$$

where $N_l(X)$ corresponds to the lowest order term of the nonlinearity $N(X)$.Variations of such feedback laws have been discussed in previous publications, see eg. [16] and references therein. For the Burgers' equation it was observed numerically in [42] that this suboptimal nonlinear controller leads to an increased closed-loop stability region with respect to the LQR feedback applied for the linearized dynamics.

Test 1: Viscous Burgers'-like equation. In this first test we address nonlinear optimal stabilization of advective-reactive phenomena, by considering a 1D Burgers'-like model with $(\xi,t) \in \mathcal{I}\times\mathbb{R}^+$ given by

$$\begin{aligned}
\partial_t X(\xi,t) &= \sigma\partial_{\xi\xi}X(x,t) + X(\xi,t)\partial_\xi X(\xi,t) + 1.5X(\xi,t)e^{-0.1X(\xi,t)} + \chi_\omega(\xi)a(t)\,,\\
X(\xi_l,t) &= X(\xi_r,t) = 0\,, \quad t\in\mathbb{R}^+,\\
X(\xi,0) &= -\text{sign}(\xi)\,, \quad \xi\in\mathcal{I}\,.
\end{aligned}$$

The feedback stabilization of Burgers' equation (without the exponential source term) has been thoroughly studied in different contexts, including the work of [16], and the recent work [34]. Since our interest is the study of optimal stabilization, we consider an additional source term $1.5X(\xi,t)e^{-0.1X(\xi,t)}$ such that the origin is not asymptotically stable. This can be appreciated in the numerical results shown in Figure 4. For this model, we consider a reduced-order state space representation of 12 states, solving a HJB equation over $\Omega = (-2,2)^{12}$. The value function is approximated with a monomial basis including both even and odd-degree polynomials up to degree 4. In Figure 4 we can compare the uncontrolled solution to the LQR- and HJB-controlled solutions, where the LQR decay is significantly slower that the one of the HJB synthesis. The HJB controller stabilizes at a higher speed, which is reflected both in the plots and in the total costs. The HJB controller obtains a reduction of approximately 18% with respect to the LQR cost. More importantly, the control signals differ in sign, magnitude, and speed. Such a behavior illustrates the nonlinear character of both the control problem and the feedback law.

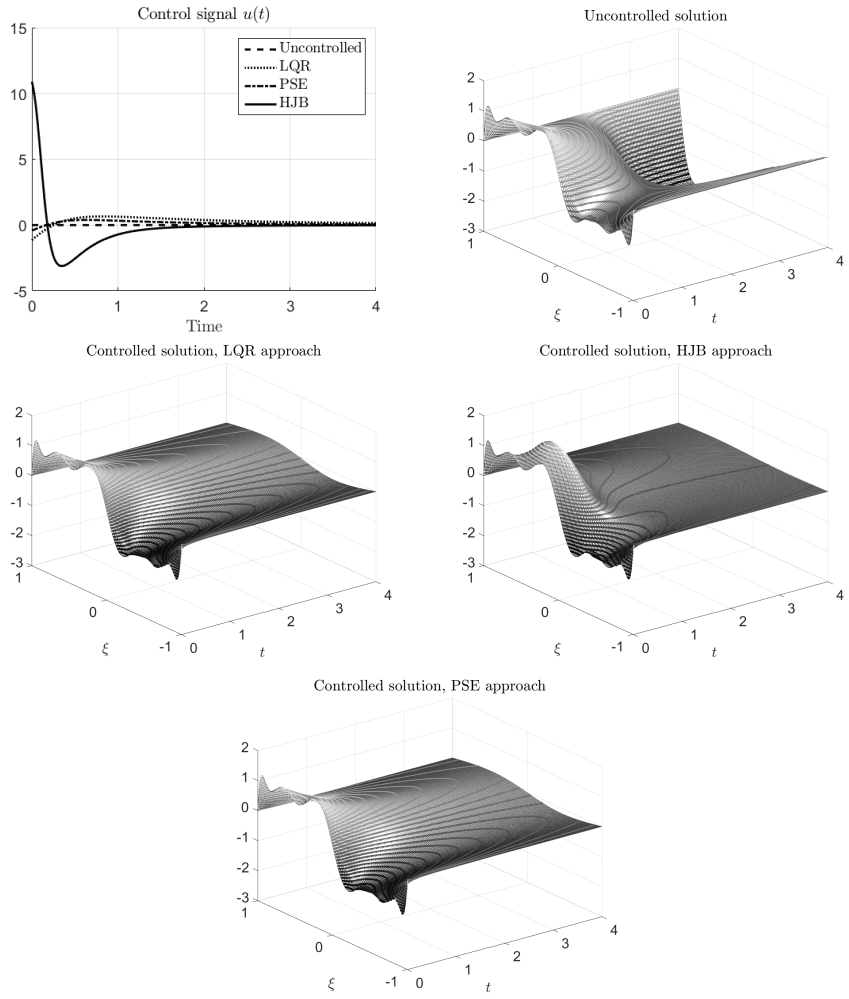


FIG. 4. *Test 1: Viscous Burgers'-like equation.* $X(\xi,0) = -sign(\xi)$. *Total costs* $\mathcal{J}(u,X)$*:* **i)** *Uncontrolled:* $+\infty$*,* **ii)** *LQR: 7.55,* **iii)** *PSE: 6.87,* **iv)** *HJB: 6.25*

Test 2: Diffusion with unstable reaction term. We now turn our attention to a diffusion equation with nonlinearity $\mathcal{N}(X) = X^3$ (the case with the reversed inequality sign in front of the cubic term was already treated in Subsection 2.1),

$$\begin{aligned}
\partial_t X(\xi,t) &= \sigma\partial_{\xi\xi}X(\xi,t) + X(\xi,t)^3 + \chi_\omega(\xi)a(t)\,, \qquad \text{in } \mathcal{I}\times\mathbb{R}^+\,,\\
\partial_\xi X(\xi_l,t) &= \partial_\xi X_(\xi_r,t) = 0\,, \quad t\in\mathbb{R}^+\,,\\
X(\xi,0) &= \delta(\xi-1)^2(\xi+1)^2\,, \quad \delta\in\mathbb{R}^+\,, \xi\in\mathcal{I}\,.
\end{aligned}$$

We close the system with Neumann boundary conditions. The origin $X(\xi,t)\equiv 0$ is an unstable equilibrium of the uncontrolled dynamics. Any other initial condition is unstable with finite time blow-up. In this case, feedback controls can only provide local stabilization, and the purpose of this numerical test is to show that HJB-based synthesis leads to an increased closed-loop asymptotic stability region when compared to LQR, and PSE controllers. For this purpose, we compute feedback controls with the

LQR, PSE and HJB approaches, for initial conditions of the form $X_0(\xi) = \delta(\xi-1)^2(\xi+1)^2$, with $\delta \in \mathbb{R}^+$. The HJB feedback is computed with Algorithm 2 initialized with a nonlinear feedback control law provided by the PSE approach. The test is carried out over $\Omega = (-2, 2)^{12}$, and the value function is approximated with monomial basis elements of degree 2 and 4. Numerical results are presented in Figure 5, for $\delta = 2$ and for a series of increased values of δ in Table 4. As the magnitude of the initial condition grows, the locally stabilizing LQR and PSE controllers are not able to prevent the finite blow-up of the dynamics. This eventually also happens for the HJB feedback, but at a much larger value of δ (we report the last value $\delta = 4$ until which the HJB control stabilizes the dynamics).

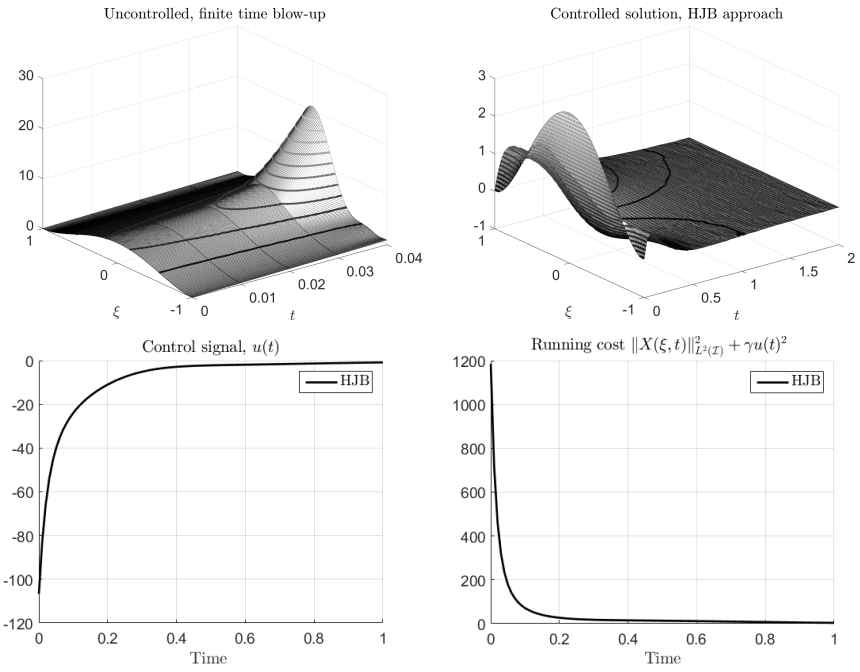


FIG. 5. *Test 2: Diffusion with unstable reaction term. Uncontrolled dynamics leads to a finite-time blow up.*

$$\mathcal{N}(X) = X^3,\ X(\xi, 0) = \delta(\xi - 1)^2(\xi + 1)^2$$

Controller	$\delta = 2$	$\delta = 3$	$\delta = 4$
Uncontrolled	$+\infty$	$+\infty$	$+\infty$
LQR	4.14	$+\infty$	$+\infty$
PSE	4.09	14.09	$+\infty$
HJB	4.06	13.98	50.36

TABLE 4

Cubic source term $\mathcal{N}(X) = X^3$ and increasing initial conditions. The HJB feedback law is the one which exhibits the largest closed-loop stability region.

Test 3: Newell-Whitehead equation. . The diffusion-reaction equation

$$\begin{aligned}
\partial_t X(\xi,t) &= \sigma\partial_{\xi\xi}X(\xi,t) + X(\xi,t)(1-X(\xi,t)^2) + \chi_\omega(\xi)a(t)\,, \qquad \text{in } \mathcal{I}\times\mathbb{R}^+\,,\\
\partial_\xi X(\xi_l,t) &= \partial_\xi X(\xi_r,t) = 0\,, \quad t\in\mathbb{R}^+\,,\\
X(\xi,0) &= X(\xi,0) = cos(2\pi\xi)cos(\pi\xi)+\delta)\,, \quad \delta\in\mathbb{R}^+, \xi\in\mathcal{I}\,,
\end{aligned}$$

corresponds to a particular case of the so-called Schlögl model, whose feedback stabilization has been studied in [14, 27]. This is a special case of a bistable system with ± 1 as stable and 0 as unstable equilibria. Here we use in an essential manner that we consider Neumann boundary conditions. For Dirichlet conditions the only equilibrium is the origin. Such systems arise for instance in Rayleigh-Benard convection and describe excitable systems such as neurons or axons. As in the previous example, the reduced state-space is chosen as $\Omega = (-2,2)^{12}$, and the basis elements for the HJB approach are even degree monomials of degree 2 and 4. Numerical results for the different controllers are shown in Figure 6. While all the feedback laws effectively stabilize the initial condition $X_0(\xi) = cos(2\pi\xi)cos(\pi\xi)+2$ to the origin, the HJB feedback has the smallest overall cost $\mathcal{J}(u,X)$. As in Test 1, it can be observed that the three feedback strategies have a considerably different transient behavior. Note that the LQR controller, which neglects the effect of the nonlinearity $\mathcal{N}(X) = -X^3$, has an increased control magnitude with respect to the nonlinear controllers which are able to account the dissipative effect of the nonlinearity. For the sake of com-

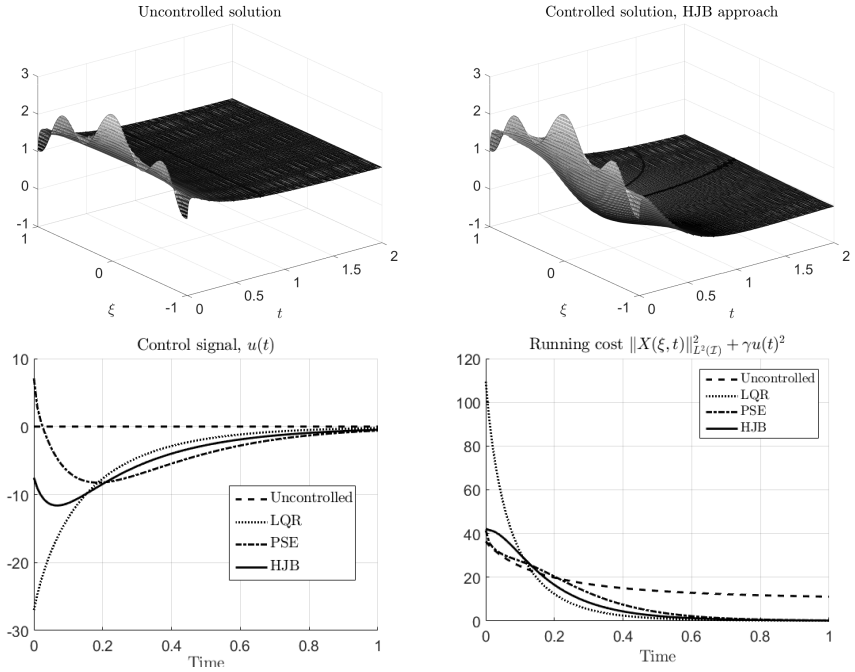


FIG. 6. *Test 3: Newell-Whitehead equation. Initial condition* $X_0(\xi) = cos(2\pi\xi)cos(\pi\xi)+2$. *Uncontrolled dynamics are attracted by the stable equilibrium* $X = 1$. *Total costs* $\mathcal{J}(u,X)$ **i)** *Uncontrolled:* ∞, **ii)** *LQR: 10.17,* **iii)** *PSE:9.69,* **iv)** *HJB: 8.85*

pleteness, we also consider this test case with a switch of the sign of nonlinearity, i.e., $\mathcal{N}(X) = -X^3$. This case is more demanding than Test 2, as now the linear part is $\sigma\partial_{\xi\xi}X + X$. However, the performance of the controllers is similar as in Test 2, and

the results are summarized in Table 5. Again, the HJB feedback law has an increased closed-loop stability region compared to the LQR and PSE controllers.

$$\mathcal{N}(X) = X^3,\ X(\xi,0) = cos(2\pi\xi)cos(\pi\xi) + \delta$$

Controller	$\delta = 1$	$\delta = 1.5$	$\delta = 2$
Uncontrolled	$+\infty$	$+\infty$	$+\infty$
LQR	5.09	$+\infty$	$+\infty$
PSE	4.92	20.02	$+\infty$
HJB	4.89	17.35	31.02

TABLE 5

Test 3 with $\mathcal{N}(X) = X^3$, for increasing initial conditions. Different local control strategies are not able to stabilize the dynamics for large initial conditions. The HJB control law has an increased region of the state space where it can stabilize.

Test 4: Degenerate Zeldovich equation. In this last test case, we consider the model given by

$$\begin{aligned}
\partial_t X(\xi,t) &= \sigma\partial_{\xi\xi}X(\xi,t) + X(\xi,t)^2 - X(\xi,t)^3 + \chi_\omega(\xi)a(t)\,, \qquad \text{in } \mathcal{I}\times\mathbb{R}^+\,,\\
\partial_\xi X(\xi_l,t) &= \partial_\xi X(\xi_r,t) = 0\,, \quad t\in\mathbb{R}^+\,,\\
X(\xi,0) &= 4(\xi-1)^2(\xi+1)^2\,, \quad \xi\in\mathcal{I}\,.
\end{aligned}$$

This equation, which arises for instance in combustion theory, has $X \equiv 1$ as stable and $X \equiv 0$ as unstable equilibria. For this case, we increase the dimension of the HJB domain to 14, i.e., $\Omega = (-2,2)^{14}$, and the basis functions are monomials of odd and even degree up to 4. Numerical results are shown in Figure 7, where it can be seen that the HJB controller yields the smaller overall cost $\mathcal{J}(u,X)$. Note that the PSE controller for this case has a diminished performance as compared even to the LQR controller. This can be explained by the fact that the PSE controller only takes into account the lowest order nonlinearity, in this case $\mathcal{N}_l(X) = X^2$, neglecting the cubic term. This is a well-known drawback of this controller, and therefore justifies the need of more complex synthesis methods for nonlinear feedback design, such as the proposed HJB approach.

Concluding remarks. A systematic technique for the computational approximation of HJB equations in optimal control problems related to semilinear parabolic equations was presented. To partially circumvent the curse of dimensionality, the dynamics of the parabolic equation are approximated by a pseudospectral collocation method, and the generalized HJB equation is approximated by separable multidimensional basis functions of a given order. The numerical results show that the feedback controls obtained by the proposed methodology differ and improve upon applying Riccati approaches to the linearized equations. The generalized HJB approach has been addressed in earlier publications, reporting on numerical results with lower dimensions than here and in part restrained enthusiasm about the numerical performance, possibly due to the lack of a systematic initialization procedure. For the class of problems considered in this paper the results were consistently better than Riccati approaches. The use of the discount factor path-following technique as proposed in Algorithm 2 is essential for stabilizing to unstable equilibria.

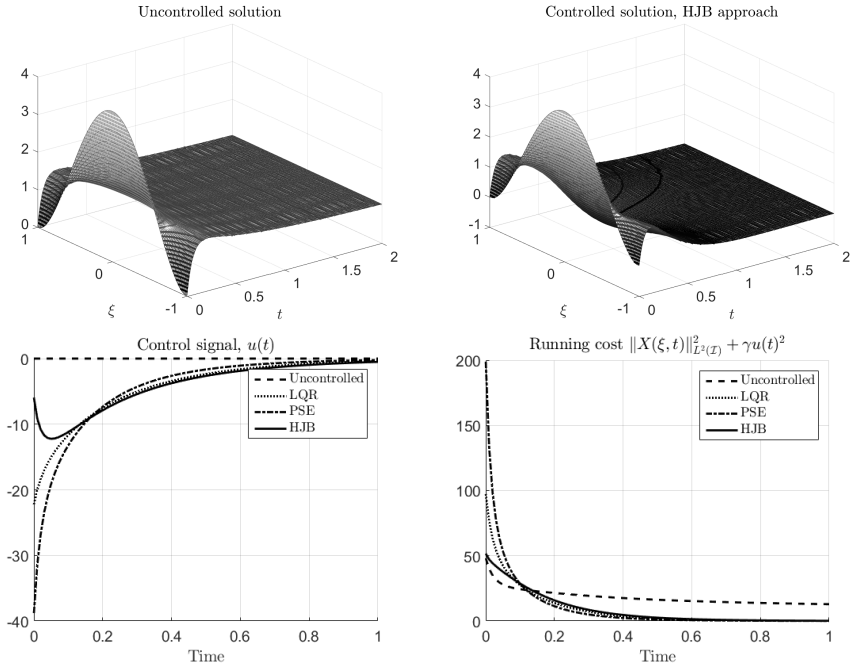


FIG. 7. *Test 4: Degenerate Zeldovich equation. Initial condition* $X_0(\xi) = 4(\xi-1)^2(\xi+1)^2$. *Total costs* $\mathcal{J}(u,X)$ **i)** *Uncontrolled:* ∞, **ii)** *LQR: 9.45,* **iii)** *PSE: 11.25,* **iv)** *HJB: 8.91*

Acknowledgments. The authors gratefully acknowledge support by the ERC advanced grant 668998 (OCLOC) under the EU's H2020 research program. Finally, D. Kalise wishes to dedicate this paper to the memory of Alexander Vasiliev.

REFERENCES

[1] A. Alla and M. Falcone. *An Adaptive POD Approximation Method for the Control of Advection-Diffusion Equations*, Control and Optimization with PDE Constraints, Internat. Ser. Numer. Math. **164**(2013), 1–17.

[2] A. Alla, M. Falcone and D. Kalise. *An efficient policy iteration algorithm for dynamic programming equations*, SIAM J. Sci. Comput., 37(1), A181-A200, 2015.

[3] H.T. Banks and K. Kunisch. *The linear regulator problem for parabolic systems,* SIAM J. Control Optim. **22**(5) (1984), 684-698.

[4] M. Bardi and I. Capuzzo-Dolcetta. *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations*, Birkhäuser Boston, 1997.

[5] R. W. Beard, G. N. Saridis, and J. T. Wen. *Galerkin approximation of the Generalized Hamilton-Jacobi-Bellman equation* Automatica **33**(12)(1997) 2159–2177.

[6] R. W. Beard, G. n. Saridis, and J. T. Wen. *Approximate solutions to the Time-Invariant Hamilton-Jacobi-Bellman equation,* J. Optim. Theory Appl. **96**(3)(1998) 589–626.

[7] S. C. Beeler, H. T. Tran, and H. T. Banks. *Feedback control methodologies for nonlinear systems* J. Optim. Theory Appl. **107**(1)(2000), 1–33.

[8] R. Bellman. *A Markovian decision process*, Indiana Univ. Math. J. **6**(4)(1957), 679–684.

[9] R. Bellman. *Adaptive control processes: a guided tour*, Princeton University Press, 1961.

[10] G. Beylkin and M. Mohlenkamp. *Algorithms for numerical analysis in high dimensions*, SIAM J. Sci. Comput. **26**(6)(2005), 2133–2159.

[11] O. Bokanowski, J. Garcke, M. Griebel, and I. Klompmaker. *An Adaptive Sparse Grid Semi-Lagrangian Scheme for First Order Hamilton-Jacobi Bellman Equations,* J. Sci. Comput. **55**(3)(2013), 575–605.

[12] O. Bokanowski, S. Maroso and H. Zidani. *Some Properties of Howards' Algorithm*, SIAM J.

Numer. Anal. **47**(4)(2009), 3001–3026.
[13] M. Bongini, M. Fornasier and D. Kalise. *(Un)conditional consensus emergence under perturbed and decentralized feedback controls*, Discrete Contin. Dyn. Syst. **35**(9)(2015), 4071–4094.
[14] T. Breiten and K. Kunisch. *Feedback stabilization of the Schlögl model by LQG-balanced truncation*, Proc. European Control Conference 2015, doi: 10.1109/ECC.2015.7330698.
[15] H. Bungartz and M. Griebel. *Sparse grids*, Acta Numerica **13**(2004), 147–269.
[16] J.A. Burns and S. Kang. *A control problem for Burgers' equation with bounded input/output*, Nonlinear Dynamics **2**(4)(1991), 235–262.
[17] F. Camilli, L. Grüne and F. Wirth. *A Generalization of Zubov's Method to Perturbed Systems*, SIAM J. Control Optim., **40**(2)(2001), 496—515.
[18] A. Chkifa, A. Cohen and C. Schwab. *High-Dimensional Adaptive Sparse Polynomial Interpolation and Applications to Parametric PDEs*, Found. Comput. Math. **14**(2014), 601-633. https://doi.org/10.1007/s10208-013-9154-z
[19] J.R. Cloutier. *State-dependent Riccati equation techniques: an overview*, Proc. Am. Control Conf. 1997, doi: 10.1109/ACC.1997.609663.
[20] M.G. Crandall and P.L. Lions. *Viscosity solutions of Hamilton-Jacobi-Bellman equations in infinite dimensions: part I*, J. Func. Anal. **62**(1985), 379–396.
[21] M. Falcone, and R. Ferretti. *Semi-Lagrangian approximation schemes for linear and Hamilton-Jacobi equations*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 2014.
[22] R. Ferretti. *Internal approximation schemes for optimal control problems in Hilbert spaces*, J. Math. Systems Estim. Control **7**(1)(1997), 1–25.
[23] V. Gaitsgory, L. Grüne, C. M. Kellett and S.R. Weller. *Stabilization with discounted optimal control : the discrete time case*, preprint, 11pp., 2016.
[24] J. Garcke and A. Kröner. *Suboptimal feedback control of PDEs by solving HJB equations on adaptive sparse grids*, J. Sci. Comput., doi:10.1007/s10915-016-0240-7 (2016).
[25] W. L. Garrard. *Suboptimal Feedback Control of Nonlinear Systems.* Automatica **8**(1972), 219–221.
[26] A.A. Gorodetsky, S. Karaman and Y.M. Marzouk. *High-dimensional stochastic optimal control using continuous tensor decompositions*, arXiv:1611.04706v1 (2016).
[27] M. Gugat and F. Tröltzsch. *Boundary feedback stabilization of the Schlögl system*, Automatica **51**(2015), 192–199.
[28] M. Gunzburger, C. G. Webster and G. Zhang. *Sparse Collocation Methods for Stochastic Interpolation and Quadrature*, in Handbook of Uncertainty Quantification (2017), Springer, 717–762.
[29] V. H. Hoang and C. Schwab. *Sparse tensor Galerkin discretization of parametric and random parabolic PDEs- Analytic regularity and generalized polynomial chaos approximation*, SIAM J. Math. Anal. **45**(5)(2013), 3050–3083.
[30] M. B. Horowitz, A. Damle, and J. W. Burdick. *Linear Hamilton Jacobi Bellman equations in high dimensions*, Proc. IEEE Conf. Decis. Control 2014, 5880–5887.
[31] R. Howard. *Dynamic Programming and Markov Processes*, The M.I.T. Press, 1960.
[32] D. Kalise and A. Kröner. *Reduced-order minimum time control of advection-reaction-diffusion systems via dynamic programming,* Proc. 21st International Symposium on Mathematical Theory of networks and Systems, 1196-1202 (2014).
[33] W. Kang and L. Wilcox. *Mitigating the Curse of Dimensionality: Sparse Grid Characteristics Method for Optimal Feedback Control and HJB Equations*, arXiv:1507.04769 (2016).
[34] A. Kröner and S. S. Rodrigues. *Remarks on the Internal Exponential Stabilization to a Nonstationary Solution for 1D Burgers Equations*, SIAM J. Control Optim. **53**(2)(2015) 1020–1055.
[35] K. Kunisch, S. Volkwein, L. Xie. *HJB-POD Based Feedback Design for the Optimal Control of Evolution Problems,* SIAM J. Appl. Dyn. Syst. **4** (2004), 701-722.
[36] I. Lasiecka and R. Triggiani. *Control theory for partial differential equations: continuous and approximations theories,* Encyclopedia of mathematics and its applications 74, Cambridge University Press, 2000.
[37] D. Olmos and B. D. Shizgal. *A pseudospectral method of solution of Fisher's equation,* J. Comput. Appl. Math. **193**(1)(2006), 219–242.
[38] R. Postoyan, L. Busoniu, D. Nesic, and J. Daafouz. *Stability of infinite-horizon optimal control with discounted cost.* Proc. IEEE Conf. Decis. Control 2014, 3903–3908.
[39] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations,* Springer Ser. Comput. Math **23**, 2008.
[40] C. Schwab and C. Gittelson. *Sparse tensor discretizations of high-dimensional parametric and stochastic PDEs*, Acta Numerica **20**(2011), 291–467.

[41] E. Stefansson and Y.P. Leong. *Sequential Alternating Least Squares for Solving High Dimensional Linear Hamilton-Jacobi-Bellman Equations,* Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), doi: 10.1109/IROS.2016.7759553, 2016.

[42] L. Thevenet, J. M. Bouchot, and J. P. Raymond. *Nonlinear feedback stabilization of a two-dimensional Burgers equation*, ESAIM Control Optim. Calc. Var. **16**(4)(2010), 929–955.