

Scalable Computation of \mathcal{H}_∞ Energy Functions for Polynomial Drift Nonlinear Systems*

Nicholas A. Corbin¹ and Boris Kramer²

Abstract—This paper presents a scalable tensor-based approach to computing controllability and observability-type energy functions for nonlinear dynamical systems with polynomial drift and linear input and output maps. Using Kronecker product polynomial expansions, we convert the Hamilton-Jacobi-Bellman partial differential equations for the energy functions into a series of algebraic equations for the coefficients of the energy functions. We derive the specific tensor structure that arises from the Kronecker product representation and analyze the computational complexity to efficiently solve these equations. The convergence and scalability of the proposed energy function computation approach is demonstrated on a nonlinear reaction-diffusion model with cubic drift nonlinearity, for which we compute degree 3 energy function approximations in $n = 1023$ dimensions and degree 4 energy function approximations in $n = 127$ dimensions.

I. INTRODUCTION

In many modern engineering applications, models are both high-dimensional and nonlinear. Common approaches to controller design often require sacrificing either the high-dimensionality or the nonlinearity in order to remain computationally tractable. For applications where both the nonlinearity and the high-dimensionality are critical to the model’s accuracy, nonlinear model reduction is an attractive way to systematically reduce the system dimensionality—to keep computations manageable—while retaining important nonlinear features of the model.

Balanced truncation (BT) is one of the predominant system-theoretic model reduction approaches for linear time-invariant (LTI) systems [1]–[3]. Its success for LTI systems has stimulated much interest in extending BT to nonlinear systems. While Scherpen provided the theoretical extensions to nonlinear control-affine systems [4]–[7], developing *scalable computational methods* to implement nonlinear BT for nonlinear systems remains an active area of research.

One of the main computational challenges in nonlinear BT is solving the Hamilton-Jacobi-Bellman (HJB) partial differential equations (PDEs) for the controllability and observability-type nonlinear *energy functions*. HJB PDEs are notoriously difficult to solve for general nonlinear systems, *especially* for high-dimensional models of interest in model reduction. Solving HJB PDEs is also required in

nonlinear optimal control problems, so many approaches have been developed to approximate solutions to HJB PDEs, including state-dependent Riccati equations [8], algebraic Gramians [9]–[11], discretization techniques [12], and iterative approaches [13], [14]. We are particularly interested in Al’brekht’s method [15], [16], a power-series-based approach which has been widely used to locally approximate HJB PDE solutions for low-dimensional models [6], [17]–[22]. More recently, this power series approach has been adapted to solve optimal control problems for moderately-sized bilinear systems [23], [24], quadratic drift systems [25], [26], and polynomial drift systems [27], [28]. These works demonstrated that polynomial approximation induces tensor structure in the resulting equations for the polynomial coefficients that modern solvers (e.g., [29], [30]) can exploit to improve scalability. For computing nonlinear BT energy functions, this approach has been adapted in [31], but only for models with quadratic drift nonlinearities.

The first contribution of this article is a scalable Kronecker product-based approach to computing nonlinear BT energy function approximations for systems with *polynomial drift of arbitrary degree ℓ* , linear inputs, and linear outputs. Moreover, we provide the explicit form of the equations of the energy function coefficients in Kronecker product form. Additionally, we provide open-access software implementations for the proposed algorithms in the `cnick1/NLBalancing` repository [32] under the `v0.9.1` tag, along with the code to reproduce the results for the presented example.

The rest of the article is organized as follows. Section II reviews the definitions for the \mathcal{H}_∞ energy functions used in nonlinear balancing, along with a summary of Al’brekht’s method for approximating energy function solutions and an introduction to the Kronecker product notation. Section III presents the main contribution of the article, namely the algorithms for computing the polynomial energy function approximations for polynomial drift systems. In Section IV, we discuss crucial details to efficiently implement the proposed method in a scalable manner. We then demonstrate the method on a nonlinear heat equation finite element model in Section V, before concluding the article in Section VI.

II. BACKGROUND, DEFINITIONS, AND NOTATION

We first review the definitions for the \mathcal{H}_∞ nonlinear BT energy functions in Section II-A. In Section II-B, we review Al’brekht’s method for locally approximating solutions to HJB PDEs. Afterwards, since our method is based on the Kronecker product polynomial representation, we review

*This work was supported by the National Science Foundation under Grant CMMI-2130727.

¹N. Corbin is with the Department of Mechanical and Aerospace Engineering, University of California San Diego, La Jolla, CA 92093-0411 USA ncorbin@ucsd.edu

²B. Kramer is with the Faculty of the Department of Mechanical and Aerospace Engineering, University of California San Diego, La Jolla, CA 92093-0411 USA bmramer@ucsd.edu

basic notation and definitions relating to Kronecker product polynomial expansions in Section II-C.

A. Energy Functions for \mathcal{H}_∞ Nonlinear Balancing

Consider the nonlinear control-affine dynamical system

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)) + \mathbf{g}(\mathbf{x}(t))\mathbf{u}(t), \quad \mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t)), \quad (1)$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ is the state, $\mathbf{f}: \mathbb{R}^n \rightarrow \mathbb{R}^n$ is the nonlinear drift, $\mathbf{g}: \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$ is the nonlinear input map, $\mathbf{h}: \mathbb{R}^n \rightarrow \mathbb{R}^p$ is the nonlinear output map, $\mathbf{u}(t) \in \mathbb{R}^m$ is a vector of input signals, and $\mathbf{y}(t) \in \mathbb{R}^p$ is a vector of output signals.

The \mathcal{H}_∞ nonlinear balancing framework [7] defines a pair of energy functions that generalize the concepts of controllability and observability to (potentially unstable) systems of the form (1). These energy functions (defined next) are then balanced using a nonlinear state-space transformation, and model reduction involves truncating states determined to be less important in the balanced representation.

Definition 1: [7, Def. 5.1] Let γ be a positive constant $\gamma > 0, \gamma \neq 1$, and define $\eta := 1 - \gamma^{-2}$. The \mathcal{H}_∞ past energy of the nonlinear system (1) is defined as

$$\mathcal{E}_\gamma^-(\mathbf{x}_0) := \min_{\substack{\mathbf{u} \in L_2(-\infty, 0] \\ \mathbf{x}(-\infty) = \mathbf{0}, \mathbf{x}(0) = \mathbf{x}_0}} \frac{1}{2} \int_{-\infty}^0 \eta \|\mathbf{y}(t)\|^2 + \|\mathbf{u}(t)\|^2 dt. \quad (2)$$

If $\gamma < 1$, the \mathcal{H}_∞ future energy of the nonlinear system (1) is defined as

$$\mathcal{E}_\gamma^+(\mathbf{x}_0) := \max_{\substack{\mathbf{u} \in L_2[0, \infty) \\ \mathbf{x}(0) = \mathbf{x}_0, \mathbf{x}(\infty) = \mathbf{0}}} \frac{1}{2} \int_0^\infty \|\mathbf{y}(t)\|^2 + \frac{\|\mathbf{u}(t)\|^2}{\eta} dt, \quad (3)$$

whereas if $\gamma > 1$, the \mathcal{H}_∞ future energy is defined as

$$\mathcal{E}_\gamma^+(\mathbf{x}_0) := \min_{\substack{\mathbf{u} \in L_2[0, \infty) \\ \mathbf{x}(0) = \mathbf{x}_0, \mathbf{x}(\infty) = \mathbf{0}}} \frac{1}{2} \int_0^\infty \|\mathbf{y}(t)\|^2 + \frac{\|\mathbf{u}(t)\|^2}{\eta} dt. \quad (4)$$

The following theorem states that the energy functions, which are nominally defined by optimization problems, can be computed as the solutions to HJB PDEs.

Theorem 1: [7, Thm. 5.2] Assume that the HJB equation

$$0 = \frac{\partial \mathcal{E}_\gamma^-(\mathbf{x})}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) + \frac{1}{2} \frac{\partial \mathcal{E}_\gamma^-(\mathbf{x})}{\partial \mathbf{x}} \mathbf{g}(\mathbf{x}) \mathbf{g}(\mathbf{x})^\top \frac{\partial^\top \mathcal{E}_\gamma^-(\mathbf{x})}{\partial \mathbf{x}} - \frac{\eta}{2} \mathbf{h}(\mathbf{x})^\top \mathbf{h}(\mathbf{x}) \quad (5)$$

has a solution with $\mathcal{E}_\gamma^-(\mathbf{0}) = 0$ such that the quantity $-\mathbf{f}(\mathbf{x}) - \mathbf{g}(\mathbf{x})\mathbf{g}(\mathbf{x})^\top \partial^\top \mathcal{E}_\gamma^-(\mathbf{x}) / \partial \mathbf{x}$ is asymptotically stable. Then this solution is the past energy function $\mathcal{E}_\gamma^-(\mathbf{x})$ from (2). Furthermore, assume that the HJB equation

$$0 = \frac{\partial \mathcal{E}_\gamma^+(\mathbf{x})}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) - \frac{\eta}{2} \frac{\partial \mathcal{E}_\gamma^+(\mathbf{x})}{\partial \mathbf{x}} \mathbf{g}(\mathbf{x}) \mathbf{g}(\mathbf{x})^\top \frac{\partial^\top \mathcal{E}_\gamma^+(\mathbf{x})}{\partial \mathbf{x}} + \frac{1}{2} \mathbf{h}(\mathbf{x})^\top \mathbf{h}(\mathbf{x}) \quad (6)$$

has a solution with $\mathcal{E}_\gamma^+(\mathbf{0}) = 0$ such that the quantity $\mathbf{f}(\mathbf{x}) - \eta \mathbf{g}(\mathbf{x})\mathbf{g}(\mathbf{x})^\top \partial^\top \mathcal{E}_\gamma^+(\mathbf{x}) / \partial \mathbf{x}$ is asymptotically stable. Then this solution is the future energy function $\mathcal{E}_\gamma^+(\mathbf{x})$ from (4) for $\gamma > 1$ and from (3) for $\gamma < 1$.

B. Al'brekht's Method To Solve HJB PDEs

In general, solving the HJB PDEs (5) and (6) analytically is not feasible. In this work, in the interest of developing scalable algorithms for nonlinear balancing, we focus on the approximation technique of Al'brekht. For the special case when the dynamics (1) are analytic, then the energy function solutions to the HJB PDEs are also analytic [15], [16]. As such, Al'brekht showed that the energy function solutions to (5) and (6) can be expanded in a convergent power series

$$\begin{aligned} \mathcal{E}_\gamma^-(\mathbf{x}) &= \mathcal{E}_\gamma^{-(2)}(\mathbf{x}) + \mathcal{E}_\gamma^{-(3)}(\mathbf{x}) + \dots + \mathcal{E}_\gamma^{-(d)}(\mathbf{x}) + \dots \\ \mathcal{E}_\gamma^+(\mathbf{x}) &= \mathcal{E}_\gamma^{+(2)}(\mathbf{x}) + \mathcal{E}_\gamma^{+(3)}(\mathbf{x}) + \dots + \mathcal{E}_\gamma^{+(d)}(\mathbf{x}) + \dots \end{aligned}$$

where $^{(2)}$ denotes a quadratic term, $^{(3)}$ denotes a cubic term, and so on. According to Al'brekht, upon inserting polynomial expressions into the HJB PDEs (5) and (6), collecting terms of the same polynomial degree gives an algebraic equation for each energy function component $\mathcal{E}_\gamma^{-(i)}(\mathbf{x})$ and $\mathcal{E}_\gamma^{+(i)}(\mathbf{x})$ for $i = 2, \dots, d$. The equations for the quadratic terms $\mathcal{E}_\gamma^{-(2)}(\mathbf{x})$ and $\mathcal{E}_\gamma^{+(2)}(\mathbf{x})$ yield algebraic Riccati equations, and the remaining components $\mathcal{E}_\gamma^{-(k)}(\mathbf{x})$ and $\mathcal{E}_\gamma^{+(k)}(\mathbf{x})$ for $k = 3, \dots, d$ solve linear algebraic systems.

While Al'brekht conceptually introduced the idea of a power-series-based approach to solving HJB PDEs, it was presented in an abstract manner without explicit equations to solve and without scalable software. In this work, we adopt the Kronecker product representation to write the polynomials explicitly so that we may derive the exact form for the algebraic equations for the energy function polynomial representation.

C. Kronecker Product Definitions and Notation

The Kronecker product of two matrices $\mathbf{A} \in \mathbb{R}^{p \times q}$ and $\mathbf{B} \in \mathbb{R}^{s \times t}$ is the $ps \times qt$ block matrix

$$\mathbf{A} \otimes \mathbf{B} := \begin{bmatrix} a_{11}\mathbf{B} & \dots & a_{1q}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{p1}\mathbf{B} & \dots & a_{pq}\mathbf{B} \end{bmatrix},$$

where a_{ij} denotes the (i, j) th entry of \mathbf{A} . Repeated Kronecker products are written as

$$\mathbf{x}^{\otimes k} := \underbrace{\mathbf{x} \otimes \dots \otimes \mathbf{x}}_{k \text{ times}} \in \mathbb{R}^{n^k}.$$

For $\mathbf{A} \in \mathbb{R}^{p \times q}$, the k -way Lyapunov matrix is defined as

$$\mathcal{L}_k(\mathbf{A}) := \sum_{i=1}^k \underbrace{\mathbf{I}_p \otimes \mathbf{A} \otimes \mathbf{I}_p \otimes \dots \otimes \mathbf{I}_p}_{k \text{ factors, } \mathbf{A} \text{ in the } i\text{th position}} \in \mathbb{R}^{p^k \times p^{k-1}q}.$$

We also use the $\text{vec}[\cdot]$ operator, which stacks the columns of a matrix into a single column vector, and the *perfect shuffle matrix* $\mathbf{S}_{p \times q} \in \mathbb{R}^{pq \times pq}$ [33], [34], defined as the permutation matrix which shuffles $\text{vec}[\mathbf{A}]$ to match $\text{vec}[\mathbf{A}^\top]$:

$$\text{vec}[\mathbf{A}^\top] = \mathbf{S}_{q \times p} \text{vec}[\mathbf{A}].$$

A concept which arises when dealing with polynomials in Kronecker product form is symmetry of the coefficients (a generalization of symmetry of a matrix), as defined next.

Definition 2 (Symmetric Coefficients): Given a monomial of the form $\mathbf{w}_d^\top \mathbf{x}^{\odot}$, the coefficient $\mathbf{w}_k \in \mathbb{R}^{n^k \times 1}$ is *symmetric* if for all $\mathbf{a}_i \in \mathbb{R}^n$ it satisfies

$$\mathbf{w}_k^\top (\mathbf{a}_1 \otimes \mathbf{a}_2 \otimes \cdots \otimes \mathbf{a}_k) = \mathbf{w}_k^\top (\mathbf{a}_{i_1} \otimes \mathbf{a}_{i_2} \otimes \cdots \otimes \mathbf{a}_{i_k}),$$

where the indices $\{i_j\}_{j=1}^k$ are any permutation of $\{1, \dots, k\}$.

III. COMPUTING \mathcal{H}_∞ ENERGY FUNCTION APPROXIMATIONS FOR POLYNOMIAL DRIFT SYSTEMS

We restrict our consideration to nonlinear dynamical systems with polynomial drift and linear inputs and outputs:

$$\dot{\mathbf{x}} = \underbrace{\mathbf{A}\mathbf{x} + \sum_{p=2}^{\ell} \mathbf{F}_p \mathbf{x}^{\odot p}}_{\mathbf{f}(\mathbf{x})} + \underbrace{\mathbf{B}}_{\mathbf{g}(\mathbf{x})} \mathbf{u}, \quad \mathbf{y} = \underbrace{\mathbf{C}\mathbf{x}}_{\mathbf{h}(\mathbf{x})}, \quad (7)$$

where $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{F}_p \in \mathbb{R}^{n \times n^p}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$, and $\mathbf{C} \in \mathbb{R}^{p \times n}$. We seek to compute energy function approximations that solve the HJB PDEs (5) and (6) for systems with the particular structure in (7). According to Al'brekht's method, the energy function solutions can be represented as polynomials; we truncate the approximations to degree d and write the energy functions using the Kronecker product. Defining $\mathcal{E}_\gamma^{- (i)}(\mathbf{x}) := \frac{1}{2} \mathbf{v}_i^\top \mathbf{x}^{\odot}$ and $\mathcal{E}_\gamma^{+ (i)}(\mathbf{x}) := \frac{1}{2} \mathbf{w}_i^\top \mathbf{x}^{\odot}$, the energy functions take the explicit forms

$$\mathcal{E}_\gamma^{-}(\mathbf{x}) \approx \frac{1}{2} \sum_{i=2}^d \mathbf{v}_i^\top \mathbf{x}^{\odot}, \quad \mathcal{E}_\gamma^{+}(\mathbf{x}) \approx \frac{1}{2} \sum_{i=2}^d \mathbf{w}_i^\top \mathbf{x}^{\odot}, \quad (8)$$

with the coefficients $\mathbf{v}_i, \mathbf{w}_i \in \mathbb{R}^{n^i}$ for $i = 2, 3, \dots, d$. In the remainder of this article, the approximations in (8) are treated as equalities. From here, we derive the algebraic equations for the coefficients \mathbf{v}_i and \mathbf{w}_i for $i = 2, 3, \dots, d$. Inserting the polynomial expression (8) for the past energy function $\mathcal{E}_\gamma^{-}(\mathbf{x})$ into the HJB PDE (5), along with the polynomial dynamics in (7), the collection of degree 2 terms is

$$\begin{aligned} 0 &= \frac{1}{2} (\mathbf{v}_2^\top (\mathbf{I}_n \otimes \mathbf{x}) + \mathbf{v}_2^\top (\mathbf{x} \otimes \mathbf{I}_n)) \mathbf{A}\mathbf{x} \\ &\quad + \frac{1}{8} (\mathbf{v}_2^\top (\mathbf{I}_n \otimes \mathbf{x}) + \mathbf{v}_2^\top (\mathbf{x} \otimes \mathbf{I}_n)) \mathbf{B}\mathbf{x} \\ &\quad + \mathbf{B}^\top (\mathbf{v}_2^\top (\mathbf{I}_n \otimes \mathbf{x}) + \mathbf{v}_2^\top (\mathbf{x} \otimes \mathbf{I}_n))^\top - \frac{\eta}{2} \mathbf{x}^\top \mathbf{C}^\top \mathbf{C}\mathbf{x}. \end{aligned}$$

This is a *quadratic* algebraic equation. Assuming the equation is symmetric¹, we require equality to hold for all \mathbf{x} ; this leads to an algebraic Riccati equation, as shown in [31]. Al'brekht's method continues by collecting terms of the next degree in a recursive fashion. The collection of degree 3 and degree 4 terms lead to the *linear* algebraic equations

$$\begin{aligned} \mathcal{L}_3 (\mathbf{A} + \mathbf{B}\mathbf{B}^\top \mathbf{V}_2)^\top \tilde{\mathbf{v}}_3 &= -\mathcal{L}_2 (\mathbf{F}_2)^\top \mathbf{v}_2, \\ \mathcal{L}_4 (\mathbf{A} + \mathbf{B}\mathbf{B}^\top \mathbf{V}_2)^\top \tilde{\mathbf{v}}_4 &= -\mathcal{L}_2 (\mathbf{F}_3)^\top \mathbf{v}_2 - \mathcal{L}_3 (\mathbf{F}_2)^\top \mathbf{v}_3 \\ &\quad - \frac{9}{4} \text{vec} [\mathbf{V}_3^\top \mathbf{B}\mathbf{B}^\top \mathbf{V}_3]. \end{aligned}$$

Carrying out the process for the remaining coefficients, we arrive at the next two theorems, which give the explicit

¹ $\mathbf{x}^\top \mathbf{M}\mathbf{x}$ only implies $\mathbf{M} = \mathbf{0}$ if \mathbf{M} is symmetric.

equations to compute the polynomial coefficients \mathbf{v}_i and \mathbf{w}_i for the energy functions (8) for the polynomial drift dynamics (7).

Theorem 2 (Past energy polynomial coefficients): Let $\gamma > \gamma_0 \geq 0$ and $\eta = 1 - \gamma^{-2}$, where γ_0 denotes the smallest $\tilde{\gamma}$ such that a stabilizing controller exists for which the \mathcal{H}_∞ norm of the closed-loop system is less than $\tilde{\gamma}$. Let the past energy function $\mathcal{E}_\gamma^{-}(\mathbf{x})$, which solves the \mathcal{H}_∞ HJB PDE (5) for the polynomial drift system (7), be of the form (8) with the coefficients $\mathbf{v}_i \in \mathbb{R}^{n^i}$ for $i = 2, 3, \dots, d$. Then $\mathbf{v}_2 = \text{vec}[\mathbf{V}_2]$, where \mathbf{V}_2 is the symmetric positive semidefinite solution to the \mathcal{H}_∞ algebraic Riccati equation

$$\mathbf{0} = \mathbf{A}^\top \mathbf{V}_2 + \mathbf{V}_2 \mathbf{A} - \eta \mathbf{C}^\top \mathbf{C} + \mathbf{V}_2 \mathbf{B}\mathbf{B}^\top \mathbf{V}_2. \quad (9)$$

For $3 \leq k \leq d$, let $\tilde{\mathbf{v}}_k \in \mathbb{R}^{n^k}$ solve the linear system

$$\begin{aligned} \mathcal{L}_k (\mathbf{A} + \mathbf{B}\mathbf{B}^\top \mathbf{V}_2)^\top \tilde{\mathbf{v}}_k &= - \sum_{\substack{i,p \geq 2 \\ i+p=k+1}} \mathcal{L}_i (\mathbf{F}_p)^\top \mathbf{v}_i \\ &\quad - \frac{1}{4} \sum_{\substack{i,j > 2 \\ i+j=k+2}} ij \text{vec} [\mathbf{V}_i^\top \mathbf{B}\mathbf{B}^\top \mathbf{V}_j] \end{aligned} \quad (10)$$

Then the coefficient vector $\mathbf{v}_k = \text{vec}[\mathbf{V}_k] \in \mathbb{R}^{n^k}$ for $3 \leq k \leq d$ is obtained by symmetrization of $\tilde{\mathbf{v}}_k$.

Theorem 3 (Future energy polynomial coefficients): Let $\gamma > \gamma_0 \geq 0$ and $\eta = 1 - \gamma^{-2}$ as in Theorem 2. Let the future energy function $\mathcal{E}_\gamma^{+}(\mathbf{x})$, which solves the \mathcal{H}_∞ HJB PDE (6) for the polynomial drift system (7), be of the form (8) with the coefficients $\mathbf{w}_i \in \mathbb{R}^{n^i}$ for $i = 2, 3, \dots, d$. Then $\mathbf{w}_2 = \text{vec}[\mathbf{W}_2]$, where \mathbf{W}_2 is the symmetric positive semidefinite solution to the \mathcal{H}_∞ algebraic Riccati equation

$$\mathbf{0} = \mathbf{A}^\top \mathbf{W}_2 + \mathbf{W}_2 \mathbf{A} + \mathbf{C}^\top \mathbf{C} - \eta \mathbf{W}_2 \mathbf{B}\mathbf{B}^\top \mathbf{W}_2. \quad (11)$$

For $3 \leq k \leq d$, let $\tilde{\mathbf{w}}_k \in \mathbb{R}^{n^k}$ solve the linear system

$$\begin{aligned} \mathcal{L}_k (\mathbf{A} - \eta \mathbf{B}\mathbf{B}^\top \mathbf{W}_2)^\top \tilde{\mathbf{w}}_k &= - \sum_{\substack{i,p \geq 2 \\ i+p=k+1}} \mathcal{L}_i (\mathbf{F}_p)^\top \mathbf{w}_i \\ &\quad + \frac{\eta}{4} \sum_{\substack{i,j > 2 \\ i+j=k+2}} ij \text{vec} [\mathbf{W}_i^\top \mathbf{B}\mathbf{B}^\top \mathbf{W}_j] \end{aligned} \quad (12)$$

Then the coefficient vector $\mathbf{w}_k = \text{vec}[\mathbf{W}_k] \in \mathbb{R}^{n^k}$ for $3 \leq k \leq d$ is obtained by symmetrization of $\tilde{\mathbf{w}}_k$.

It is shown in [31] that the matrices $\mathcal{L}_k (\mathbf{A} + \mathbf{B}\mathbf{B}^\top \mathbf{V}_2)^\top$ and $\mathcal{L}_k (\mathbf{A} - \eta \mathbf{B}\mathbf{B}^\top \mathbf{W}_2)^\top$ are invertible, and hence the linear systems (10) and (12) are uniquely solvable for the energy function coefficients \mathbf{v}_k and \mathbf{w}_k .

IV. IMPLEMENTATION DETAILS

With a naive implementation, the Kronecker product-based approach scales poorly: it requires solving linear systems (10) and (12) of dimension n^k for \mathbf{v}_k , which has a cost of $O(n^{3k})$ using a direct solver. Thus efficiently solving—and even forming—the linear systems (10) and (12) is critical for scalability to systems with large state dimension n . In this section, we outline some of the key details for efficiently implementing the proposed approach, focusing on (10) since the cost for (12) is identical.

Forming the right-hand-sides of the linear systems

To begin, we must consider how to efficiently assemble the right-hand side of (10), starting with the first set of terms:

$$-\sum_{\substack{i,p \geq 2 \\ i+p=k+1}} \mathcal{L}_i(\mathbf{F}_p)^\top \mathbf{v}_i. \quad (13)$$

The matrix $\mathcal{L}_i(\mathbf{F}_p)$ has dimension $(n^i \times n^k)$, whereas the vector \mathbf{v}_i is $(n^i \times 1)$, so the cost of evaluating the Lyapunov product $\mathcal{L}_i(\mathbf{F}_p)^\top \mathbf{v}_i$ using naive matrix-vector multiplication is $O(n^{k+i})$ using level-2 BLAS operations. The dominant cost occurs for the term with $i = k - 1$, for a total cost of $O(n^{2k-1})$. Instead, we exploit the structure of the i -way Lyapunov matrix to form these terms more efficiently. Consider a single term from the sum (13), and expand it according to the definition of the i -way Lyapunov matrix

$$\mathcal{L}_i(\mathbf{F}_p)^\top \mathbf{v}_i = (\mathbf{F}_p^\top \otimes \mathbf{I}_{n^{i-1}}) \mathbf{v}_i + (\mathbf{I}_n \otimes \mathbf{F}_p^\top \otimes \mathbf{I}_{n^{i-2}}) \mathbf{v}_i + \dots$$

All of the quantities on the right-hand side are equivalent under an appropriate permutation/reshaping, so the total cost of forming this is i times the cost of computing the first term. Using the Kronecker-vec relation [33] $(\mathbf{D}^\top \otimes \mathbf{I}_p) \text{vec}[\mathbf{A}] = \text{vec}[\mathbf{AD}]$, we rewrite the first term in the sum as

$$(\mathbf{F}_p^\top \otimes \mathbf{I}_{n^{i-1}}) \mathbf{v}_i = \text{vec}[\mathbf{V}_i^\top \mathbf{F}_p],$$

which is now matrix multiplication of $(n^{i-1} \times n)$ and $(n \times n^p)$ matrices, which has a cost of $O(n^{i+p})$ using level-3 BLAS operations. Since $i+p = k+1$, this is equivalent to $O(n^{k+1})$. Performing this operation i times for the remaining terms, the total cost of evaluating $\mathcal{L}_i(\mathbf{F}_p)^\top \mathbf{v}_i$ this way is $O(in^{k+1})$. Again, the dominant cost occurs for the case $i = k - 1$, hence the total cost to form the set of terms (13) is $O(kn^{k+1})$, as opposed to $O(n^{2k-1})$ for a naive implementation.

In the remaining terms in (10)

$$-\frac{1}{4} \sum_{\substack{i,j > 2 \\ i+j=k+2}} ij \text{vec}[\mathbf{V}_i^\top \mathbf{B} \mathbf{B}^\top \mathbf{V}_j], \quad (14)$$

products like $\mathbf{V}_i^\top \mathbf{B}$ appear repeatedly, so we store them in memory to avoid repeatedly forming them; however, the dominant cost comes from multiplying these stored quantities. Treating $\mathbf{V}_i^\top \mathbf{B}$ as an $(n^{i-1} \times m)$ matrix and $\mathbf{B}^\top \mathbf{V}_j$ as an $(m \times n^{j-1})$ matrix, the multiplication $\mathbf{V}_i^\top \mathbf{B} \mathbf{B}^\top \mathbf{V}_j$ costs $O(mn^{i+j-2})$ using level-3 BLAS operations. Since $i+j = k+2$, this is equal to $O(mn^k)$. We form $k-2$ of these terms in the sum, so the overall cost is $O(kmn^k)$. Assuming $n \gg km$, this is negligible compared to the cost of forming the other terms and solving the equations.

Solving the linear systems

An efficient solver is required to avoid the $O(n^{3d})$ operations used in a naive direct solve. As has been shown in other recent works [23]–[27], [31], the k -way Lyapunov matrix structure of the linear system (10) can be exploited by modern solvers, e.g., [29], [30]. In this work, we use the k -way Bartels-Stewart algorithm from [27], though better performance may be obtained with other solvers. To summarize

the algorithm, a Schur factorization of $\mathbf{A} + \mathbf{B} \mathbf{B}^\top \mathbf{V}_2$ is used to convert the linear system into an upper-triangular form which requires solving n^{k-1} dense upper triangular linear systems of size n . Each of these n^{k-1} systems can be solved by back substitution with a cost of $O(n^2)$, for a total cost of $O(n^{k+1})$, as opposed to a cost of $O(n^{3k})$ for a naive direct solve of the full system.

Summary of key implementation details

In summary, the linear system (10) for the k th coefficient \mathbf{v}_k is formed efficiently by using the Kronecker-vec identity to reshape Kronecker products as matrix-matrix multiplications, which can be computed with $O(kn^{k+1})$ complexity. Then, using the k -way Bartels-Stewart solver from [27], (10) can be solved with $O(n^{k+1})$ complexity. Since the highest-degree coefficient \mathbf{v}_d is the most expensive to compute, the overall computational complexity is $O(dn^{d+1})$.

V. NUMERICAL RESULTS FOR A NONLINEAR HEAT EQUATION

We evaluate the scalability and convergence of the approach proposed in Sections III and IV on a finite element model of a reaction-diffusion problem. The results are obtained on a Linux workstation with an Intel Xeon W-3175X CPU and 256 GB RAM.

Model: Consider the following nonlinear heat equation modeling a reaction-diffusion system

$$\begin{aligned} z_t(x, t) &= z_{xx}(x, t) + z_x(x, t) + \frac{1}{8}z(x, t) \\ &\quad + z(x, t)^3 + \sum_{j=1}^m b_j^m(x) u_j(t), \\ y_i(t) &= \int_{\mathcal{X}_{[(i-1)\ell/p, i\ell/p]}} z(x, t) dx, \quad i = 1, \dots, p, \end{aligned}$$

where z represents temperature, $x \in [0, \ell]$ is the spatial coordinate, and t is time. The system is subject to Dirichlet boundary conditions $z(0, t) = z(\ell, t) = 0$ and an initial condition

$$z(x, 0) = 5 \cdot 10^{-5} x(x - \ell)(x - \ell/2).$$

Stability of the open-loop system was studied in [35]–[37]. Despite the origin being locally asymptotically stable, trajectories that begin near the origin can experience transient growth; this growth feeds the nonlinear terms in the model, making the model unstable despite its linearization being stable. This feature has been cited as a motivation for nonlinear model reduction methods [37].

The problem from [35]–[37] is augmented with control inputs $u_j(t)$ applied over m equally sized subdomains given by the characteristic function $b_j^m(x) = \chi_{[(j-1)\ell/m, j\ell/m]}(x)$ and outputs $y_i(t)$ that are spatial averages of the solution over p equally sized subdomains. Fig. 1 shows a simple diagram of the model.

We pick $\ell = 30$, $m = p = 4$, and we discretize the model with N linear finite elements. This leads to a finite-dimensional state-space model of dimension $n = N - 1$ with

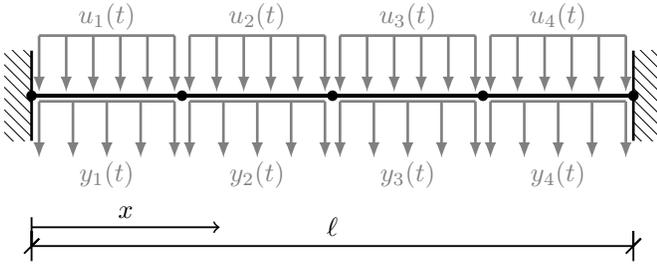


Fig. 1. Heat equation on a physical domain of length ℓ , subdivided here into $m = p = 4$ regions for the inputs $u_j(t)$ and outputs $y_i(t)$.

cubic drift nonlinearity that can be written as

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{F}_3\mathbf{x}^{(3)} + \mathbf{B}\mathbf{u}, \quad \mathbf{y} = \mathbf{C}\mathbf{x}.$$

Results: First, we investigate the convergence of the future energy function for $\eta = 0.5$ as the finite element mesh is refined². Due to spatial limitations, we omit the past energy function results and report that they are qualitatively and quantitatively comparable. In Table I, the size of model $n = N - 1$ is increased while keeping the degree of the energy function approximation fixed at $d = 3$ and $d = 4$, respectively. The energy function values are shown in the second and third columns of Table I for the initial condition \mathbf{x}_0 corresponding to $z(x, 0)$. In Fig. 2 we plot these energy function values to more clearly show the trend. As the mesh is refined, the energy function values converge, indicating that the energy function approximations are behaving as expected.

TABLE I. Future energy function convergence w.r.t. n for $d = 3, 4$.

n	$\mathcal{E}_3^+(\mathbf{x}_0)$ (CPU Sec)	$\mathcal{E}_4^+(\mathbf{x}_0)$ (CPU Sec)
3	$5.78311 \cdot 10^{-2}$ ($1.9 \cdot 10^{-3}$)	$5.87940 \cdot 10^{-2}$ ($3.8 \cdot 10^{-3}$)
7	$6.17185 \cdot 10^{-2}$ ($3.0 \cdot 10^{-3}$)	$6.28924 \cdot 10^{-2}$ ($1.3 \cdot 10^{-2}$)
15	$6.74241 \cdot 10^{-2}$ ($9.7 \cdot 10^{-3}$)	$6.87624 \cdot 10^{-2}$ ($1.3 \cdot 10^{-1}$)
31	$6.99113 \cdot 10^{-2}$ ($5.3 \cdot 10^{-2}$)	$7.13010 \cdot 10^{-2}$ ($1.6 \cdot 10^0$)
63	$7.08615 \cdot 10^{-2}$ ($4.4 \cdot 10^{-1}$)	$7.22615 \cdot 10^{-2}$ ($3.1 \cdot 10^1$)
127	$7.12533 \cdot 10^{-2}$ ($4.6 \cdot 10^0$)	$7.26545 \cdot 10^{-2}$ ($8.4 \cdot 10^2$)
255	$7.14271 \cdot 10^{-2}$ ($6.1 \cdot 10^1$)	–
511	$7.15084 \cdot 10^{-2}$ ($1.0 \cdot 10^3$)	–
1,023	$7.15476 \cdot 10^{-2}$ ($1.8 \cdot 10^4$)	–

We confirmed that the energy functions for this initial condition converge with respect to d by computing up to a $d = 10$ approximation for $n = 7$. For initial conditions close enough to the origin, the theory is well established that the Taylor approximation converges to the true energy function [15], so we omit those results since the numerical scaling with respect to n is more of interest for model reduction.

The CPU time required to compute the energy approximations is also shown in Table I to demonstrate the scalability of the proposed algorithm with respect to model dimension n . Since the variance in CPU time can be large for small models (for which the program runs in a fraction of a

²For the numerical results, the number of elements N is chosen to be a multiple of 4 so that the boundaries of the control and measurement subdomains occur at nodes in the finite element mesh.

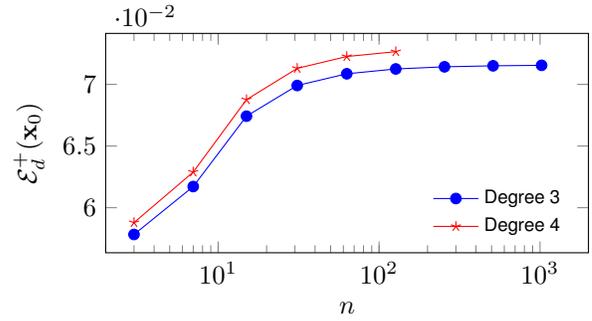


Fig. 2. Convergence w.r.t n of the future energy function evaluated at the initial condition \mathbf{x}_0 as the finite element mesh is refined.

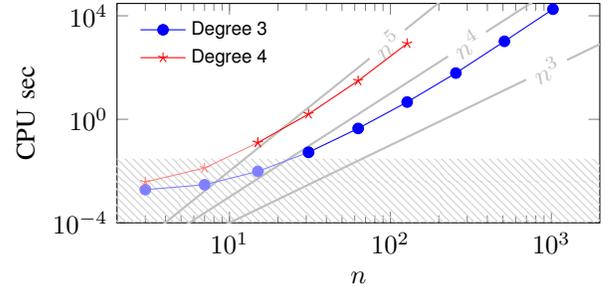


Fig. 3. Scaling of CPU time w.r.t n for $d = 3, 4$. The scaling closely approaches $O(dn^{d+1})$ as n grows, as predicted by the computational complexity analysis. The grayed out data points are very sensitive to computational variance on the order of tens of milliseconds, so they are not indicative of the true scaling of the algorithms for large models.

second), the compute time is averaged over 10 runs for the models of size $n = 63$ and smaller. The computation times are plotted in Fig. 3 to more clearly show the scaling. As discussed in Section IV, our algorithm theoretically scales as $O(dn^{d+1})$. In practice, this is in fact roughly the scaling we see for moderately sized problems, as shown in Fig. 3. We observe an apparent departure from the predicted scaling for small n , but we attribute this to the minimum overhead time to run the program, which does not scale with the model dimension and appears to be on the order of a few milliseconds. We also found that the execution time could vary by tens of milliseconds for these small model sizes, so despite averaging over several runs, these data points are highly variable. We have therefore grayed out the first few data points so that they are still visible but do not obstruct the overall scaling trend as the model dimension increases.

Since the system is odd, the energy functions are even for this example. Therefore $\mathbf{w}_3 = \mathbf{0}$, and the degree 3 energy function approximation is identical to the degree 2 LQR solution given by \mathbf{W}_2 ; a practitioner seeking to incorporate nonlinear model features in the energy function should therefore compute at least a degree 4 approximation.

VI. CONCLUSION

In this paper, we have presented a novel method for computing \mathcal{H}_∞ energy functions for nonlinear systems with polynomial drift, linear inputs, and linear outputs. The approach is based on the Kronecker product polynomial representation, which allows us to obtain explicit formulas for

the coefficients of the Taylor-series expansions of the energy functions. The main advantage over existing approaches is the ability to handle an arbitrary degree of polynomial drift nonlinearity. By exploiting the tensor structure of the Kronecker product, we have shown that the method is scalable to moderately sized problems, as demonstrated on the nonlinear heat equation example.

The proposed method opens up several directions for future research. A natural step forward involves extending the approach to nonlinear inputs and outputs so that general polynomial control-affine systems can be studied. Another direction of major interest is to use the computed energy functions for control and model reduction. The energy functions can be used directly for polynomial state-feedback controllers, which have shown some promise in the literature over LQR controllers [17], [27], [28]. The energy functions are also a key step towards obtaining reduced-order models, which can then be used to build observers and controllers for output-feedback problems. Further accelerating the computations by using more efficient solvers is also of interest.

REFERENCES

- [1] C. Mullis and R. Roberts, "Synthesis of minimum roundoff noise fixed point digital filters," *IEEE Transactions on Circuits and Systems*, vol. 23, no. 9, pp. 551–562, Sep. 1976.
- [2] B. C. Moore, "Principal component analysis in linear systems: Controllability, observability, and model reduction," *IEEE Transactions on Automatic Control*, vol. 26, pp. 17–32, Feb. 1981.
- [3] S. Gugercin and A. C. Antoulas, "A survey of model reduction by balanced truncation and some new results," *International Journal of Control*, vol. 77, no. 8, pp. 748–766, May 2004.
- [4] J. M. A. Scherpen, "Balancing for nonlinear systems," *Systems & Control Letters*, vol. 21, no. 2, pp. 143–153, Aug. 1993.
- [5] J. M. A. Scherpen and A. J. Van Der Schaft, "Normalized coprime factorizations and balancing for unstable nonlinear systems," *International Journal of Control*, vol. 60, no. 6, pp. 1193–1222, Dec. 1994.
- [6] J. M. A. Scherpen, "Balancing for nonlinear systems," Ph.D. dissertation, University of Twente, 1994.
- [7] —, " \mathcal{H}_∞ balancing for nonlinear systems," *International Journal of Robust and Nonlinear Control*, vol. 6, no. 7, pp. 645–668, Aug. 1996.
- [8] T. Cimen, "Survey of state-dependent Riccati equation in nonlinear optimal feedback control synthesis," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 4, pp. 1025–1047, Jul. 2012.
- [9] M. Condon and R. Ivanov, "Nonlinear systems – algebraic Gramians and model reduction," *COMPEL - The international journal for computation and mathematics in electrical and electronic engineering*, vol. 24, no. 1, pp. 202–219, Mar. 2005.
- [10] W. S. Gray and E. I. Verriest, "Algebraically defined Gramians for nonlinear systems," in *2006 45th IEEE Conference on Decision and Control*. IEEE, Dec. 2006.
- [11] P. Benner and P. Goyal, "Balanced truncation model order reduction for quadratic-bilinear control systems," Apr. 2017, *arXiv:1705.00160*.
- [12] M. Falcone and R. Ferretti, "Numerical methods for Hamilton-Jacobi type equations," in *Handbook of Numerical Analysis*. Elsevier, 2016, pp. 603–626.
- [13] D. Kalise and K. Kunisch, "Polynomial approximation of high-dimensional Hamilton-Jacobi-Bellman equations and applications to feedback control of semilinear parabolic PDEs," *SIAM Journal on Scientific Computing*, vol. 40, no. 2, pp. A629–A652, 2018.
- [14] S. Dolgov, D. Kalise, and K. Kunisch, "Tensor decomposition methods for high-dimensional Hamilton-Jacobi-Bellman equations," *SIAM Journal on Scientific Computing*, vol. 43, no. 3, pp. A1625–A1650, Jan. 2021.
- [15] E. G. Al'brekht, "On the optimal stabilization of nonlinear systems," *Journal of Applied Mathematics and Mechanics*, vol. 25, no. 5, pp. 1254–1266, Jan. 1961.
- [16] D. L. Lukes, "Optimal regulation of nonlinear dynamical systems," *SIAM Journal on Control*, vol. 7, no. 1, pp. 75–100, Feb. 1969.
- [17] W. L. Garrard and J. M. Jordan, "Design of nonlinear automatic flight control systems," *Automatica*, vol. 13, no. 5, pp. 497–505, Sep. 1977.
- [18] K. Fujimoto and D. Tsubakino, "Computation of nonlinear balanced realization and model reduction based on Taylor series expansion," *Systems & Control Letters*, vol. 57, no. 4, pp. 283–289, Apr. 2008.
- [19] K. Fujimoto and J. M. A. Scherpen, "Balanced realization and model order reduction for nonlinear systems based on singular value analysis," *SIAM Journal on Control and Optimization*, vol. 48, no. 7, pp. 4591–4623, Jan. 2010.
- [20] A. Krener, C. Aguilar, and T. Hunt, "Series solutions of HJB equations," *Mathematical System Theory—Festschrift in Honor of Uwe Helmke on the Occasion of his Sixtieth Birthday*, pp. 247–260, 2013.
- [21] S. Sahyoun, J. Dong, and S. M. Djouadi, "Reduced order modeling for fluid flows based on nonlinear balanced truncation," in *2013 American Control Conference (ACC)*. IEEE, Jun. 2013.
- [22] A. J. Krener, "Nonlinear Systems Toolbox," Available on request to ajkrener@nps.edu, 2019.
- [23] T. Breiten, K. Kunisch, and L. Pfeiffer, "Numerical study of polynomial feedback laws for a bilinear control problem," *Mathematical Control & Related Fields*, vol. 8, no. 3, pp. 557–582, 2018.
- [24] —, "Taylor expansions of the value function associated with a bilinear optimal control problem," *Annales de l'Institut Henri Poincaré C, Analyse non linéaire*, vol. 36, no. 5, pp. 1361–1399, Aug. 2019.
- [25] —, "Feedback stabilization of the two-dimensional Navier–Stokes equations by value function approximation," *Applied Mathematics & Optimization*, vol. 80, no. 3, pp. 599–641, Jun. 2019.
- [26] J. Borggaard and L. Zietsman, "The Quadratic-Quadratic Regulator problem: approximating feedback controls for quadratic-in-state nonlinear systems," in *2020 American Control Conference (ACC)*, Jul. 2020, pp. 818–823.
- [27] —, "On approximating Polynomial-Quadratic Regulator problems," *IFAC-PapersOnLine*, vol. 54, no. 9, pp. 329–334, 2021.
- [28] H. Almubarak, N. Sadegh, and D. G. Taylor, "Infinite horizon nonlinear quadratic cost regulator," in *2019 American Control Conference (ACC)*, Jul. 2019, pp. 5570–5575.
- [29] L. Grasedyck, "Existence and computation of low Kronecker-rank approximations for large linear systems of tensor product structure," *Computing*, vol. 72, no. 3–4, Jan. 2004.
- [30] M. Chen and D. Kressner, "Recursive blocked algorithms for linear systems with kronecker product structure," *Numerical Algorithms*, vol. 84, no. 3, pp. 1199–1216, Sep. 2019.
- [31] B. Kramer, S. Gugercin, J. Borggaard, and L. Balicki, "Nonlinear balanced truncation: Part 1—computing energy functions," Dec. 2022, *arXiv:2209.07645v2*.
- [32] N. A. Corbin and J. Borggaard, "NLbalancing repository," Available online: <https://github.com/cnick1/NLbalancing>, Mar. 2024.
- [33] H. V. Henderson and S. R. Searle, "The vec-permutation matrix, the vec operator and Kronecker products: a review," *Linear and Multilinear Algebra*, vol. 9, no. 4, pp. 271–288, Jan. 1981.
- [34] C. F. Van Loan, "The ubiquitous Kronecker product," *Journal of Computational and Applied Mathematics*, vol. 123, no. 1–2, pp. 85–100, Nov. 2000.
- [35] B. Sandstede and A. Scheel, "Basin boundaries and bifurcations near convective instabilities: a case study," *Journal of Differential Equations*, vol. 208, no. 1, pp. 176–193, Jan. 2005.
- [36] J. Galkowski, "Nonlinear instability in a semiclassical problem," *Communications in Mathematical Physics*, vol. 316, no. 3, pp. 705–722, Oct. 2012.
- [37] M. Embree, "Unstable modes in projection-based reduced-order models: how many can there be, and what do they tell you?" *Systems & Control Letters*, vol. 124, pp. 49–59, Feb. 2019.