Sum-of-squares approximations to energy functions

Hamza Adjerid

with Jeff Borggaard

Virginia Tech, May 2023

Outline

Problem formulation

Nonlinear systems

- The general form of nonlinear dynamical system:

$$E\dot{x}(t) = f(x(t)) + Bu(t), \quad y(t) = Cx(t),$$

  with $x \in \mathbb{R}^n$, $u \in \mathbb{R}^n$, $y \in \mathbb{R}^p$, $f : \mathbb{R}^n \to \mathbb{R}^n$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, and $E \in \mathbb{R}^{n\times n}$.

- One particular case of non-linearity is the quadratic one given by:

$$f(x) = Ax(t) + N_2 x^{②}(t),$$

  where $N_2 \in \mathbb{R}^{n\times n^2}$ and $x^{②}(t)$ is a 2-times Kronecker product of $x(t)$.

Problem formulation

Energy functions (J. M. Scherpen, 1996) [2]

The $\mathcal{H}_\infty$ energy functions related to nonlinear dynamical systems are:

- The $\mathcal{H}_\infty$ past energy function:

$$\mathcal{E}_\gamma^-(x_0) := \min_{u \in L_2(-\infty,0]} \frac{1}{2}\int_{-\infty}^{0} (1-\gamma^{-2})||y(t)||^2 + ||u(t)||^2 dt$$

  where $x(-\infty) = 0$, $x(0) = x_0$, and $0 < \gamma \neq 1$.

- The $\mathcal{H}_\infty$ future energy function:

$$\mathcal{E}_\gamma^+(x_0) := \min_{u \in L_2[0,\infty)} \frac{1}{2}\int_{0}^{\infty} ||y(t)||^2 + \frac{||u(t)||^2}{1-\gamma^{-2}} dt,$$

  where $x(-\infty) = 0$, $x(0) = x_0$, and $\gamma > 1$ and min is replaced by max if $0 < \gamma < 1$.

Problem formulation

Hamilton-Jacobi-Bellman equations (J. M. Scherpen and A. Van der Schaft, 1994) [3]

The $\mathcal{H}_\infty$ energy functions are solutions to the following PDEs:

$$0 = \frac{\partial \mathcal{E}_\gamma^-}{\partial x}(x) f(x) + \frac{1}{2}\frac{\partial \mathcal{E}_\gamma^-}{\partial x}(x) BB^T \frac{\partial^T \mathcal{E}_\gamma^-}{\partial x} - \frac{1}{2}(1-\gamma^{-2}) x^T CC^T x$$

$$0 = \frac{\partial \mathcal{E}_\gamma^+}{\partial x}(x) f(x) - \frac{1}{2}(1-\gamma^{-2})\frac{\partial \mathcal{E}_\gamma^+}{\partial x}(x) BB^T \frac{\partial^T \mathcal{E}_\gamma^+}{\partial x} + \frac{1}{2} x^T CC^T x$$

where $\mathcal{E}_\gamma^-(0) = \mathcal{E}_\gamma^+(0) = 0$.

Problem formulation

Energy function as an optimal control [2]

$\mathcal{E}_\gamma^-(x)$ being a solution to the HJB gives rise an to optimal control $u^*(x)$ given by:

$$u^*(x) = -R^{-1}B^T \frac{\partial \mathcal{E}_\gamma^-(x)}{\partial x},$$

with the following quadratic cost function

$$\hat{\mathcal{E}}(x_0, u) = \frac{1}{2}\int_0^\infty x^T(t)Qx(t) + u^T(t)Ru(t)dt$$

that drives the following system to stability:

$$\dot{x}(t) = -f(x(t)) + Bu(x(t)).$$

Motivation

Polynomial Approximation (B. Kramer et al., 2022) [4]

The $\mathcal{H}_\infty$ Energy functions can be approximated by :

$$\mathcal{E}_\gamma^-(x) = \frac{1}{2}\left(w_2^T x^{②} + w_3^T x^{③} + \cdots + w_d^T x^{ⓓ}\right)$$

$$\mathcal{E}_\gamma^+(x) = \frac{1}{2}\left(v_2^T x^{②} + v_3^T x^{③} + \cdots + v_d^T x^{ⓓ}\right)$$

where $\mathcal{E}_\gamma^-(0) = \mathcal{E}_\gamma^+(0) = 0$.

Motivation

- Polynomial approximations are very good around the origin. The main issue with these approximations is NEGATIVITY away from the origin. However, energy functions must be positive by definition.
- One way to overcome this issue is to propose function approximations that impose positivity.

Motivation

- Polynomial approximations are very good around the origin. The main issue with these approximations is NEGATIVITY away from the origin. However, energy functions must be positive by definition.
- One way to overcome this issue is to propose function approximations that impose positivity.

Motivation

- One alternative can be a Sum-of-Squares, formulated as follows:

$$f_{sos}(x) = \sum_{i=1}^{N} f_i^2(x),$$

where $f_i(x)$ can be any generic function and N any finite number.

- In this work, we are more interested in the sum-of-squares using polynomials

$$f_{sos}(x) = \sum_{i=1}^{N} p_i^2(x),$$

where each $p_i(x)$ is a polynomial.

Motivation

- One alternative can be a Sum-of-Squares, formulated as follows:

$$f_{sos}(x) = \sum_{i=1}^{N} f_i^2(x),$$

where $f_i(x)$ can be any generic function and N any finite number.

- In this work, we are more interested in the sum-of-squares using polynomials

$$f_{sos}(x) = \sum_{i=1}^{N} p_i^2(x),$$

where each $p_i(x)$ is a polynomial.

History

- The case of sum-of-squares of polynomials is a well analyzed problem, first studied by David Hilbert in 1900.
- Sum-of-squares, as a tractable way to perform positive semi-definite programming for dynamical systems, was first proposed by Pablo A.Parrilo in his thesis in 2000. [5]
- It allowed the control community to tackle different problems such as, optimization problems, Lyapunov stability analysis, or computation of tight upper bounds.

History

- The case of sum-of-squares of polynomials is a well analyzed problem, first studied by David Hilbert in 1900.
- Sum-of-squares, as a tractable way to perform positive semi-definite programming for dynamical systems, was first proposed by Pablo A.Parrilo in his thesis in 2000. [5]
- It allowed the control community to tackle different problems such as, optimization problems, Lyapunov stability analysis, or computation of tight upper bounds.

History

- The case of sum-of-squares of polynomials is a well analyzed problem, first studied by David Hilbert in 1900.
- Sum-of-squares, as a tractable way to perform positive semi-definite programming for dynamical systems, was first proposed by Pablo A.Parrilo in his thesis in 2000. [5]
- It allowed the control community to tackle different problems such as, optimization problems, Lyapunov stability analysis, or computation of tight upper bounds.

Formulation of SOS polynomials

Proposition: [5]

A polynomial $p(x)$ of degree $2d$ is a SOS if and only if there exists a positive semidefinite matrix Q and vector of monomials $Z(x)$ containing monomials in x of degree $\leq$ d such that:

$$p(x) = Z(x)^T QZ(x)$$

Formulation of SOS polynomials

Example:[5]

Suppose we want to know in the following polynomial

$$p(x_1, x_2) = 2x_1^4 + 2x_1^3x_2 - x_1^2x_2^2 + 5x_2^4$$

is a sum-of-squares. We let $Z(x) = [x_1^2 \quad x_1x_2 \quad x_2^2]^T$ and we put

$$p(x_1, x_2) = Z(x)^T QZ(x)$$

where $Q \in \mathbb{R}^{n\times n}$, and we look for a *PSD* matrix Q.

Formulation of SOS polynomials

Example:[5]

In fact by expanding $Z(x)^T QZ(x)$ and matching the coefficients, a *PSD* matrix Q can be formed as

$$Q = \begin{pmatrix} 2 & 1 & -3 \\ 1 & 5 & 0 \\ -3 & 0 & 5 \end{pmatrix} = LL^T \quad \text{with} \quad L = \frac{1}{\sqrt{2}} \begin{pmatrix} 2 & 0 \\ 1 & 3 \\ -3 & 1 \end{pmatrix},$$

hence

$$p(x_1, x_2) = \frac{1}{2}(2x_1^2 + x_1x_2 - 3x_2^2)^2 + \frac{1}{2}(x_2^2 + 3x_1x_2)^2$$

is, in fact, a SOS.

Formulation of SOS polynomials

The use of SOS programming in HJB problems have been addressed in prior works, e.g. [6]. In our work, we approach the problem in three different SOS formulations:

- **Completing the square:** Adding higher degree terms to a given polynomial approximation to make it SOS.
- **Completing the square and collocation method:** Lower degree terms match a given polynomial approximation, collocation used to determine highest degree terms.
- **Collocation method:** Use Parrilo's formulation of SOS within a least-squares collocation method.

Here, the approaches are applied to the the past energy function $\mathcal{E}_\gamma^-(x)$. The same procedure can be applied to the future energy function $\mathcal{E}_\gamma^+(x)$.

Formulation of SOS polynomials

The use of SOS programming in HJB problems have been addressed in prior works, e.g. [6]. In our work, we approach the problem in three different SOS formulations:

- **Completing the square:** Adding higher degree terms to a given polynomial approximation to make it SOS.
- **Completing the square and collocation method:** Lower degree terms match a given polynomial approximation, collocation used to determine highest degree terms.
- **Collocation method:** Use Parrilo's formulation of SOS within a least-squares collocation method.

Here, the approaches are applied to the the past energy function $\mathcal{E}_\gamma^-(x)$. The same procedure can be applied to the future energy function $\mathcal{E}_\gamma^+(x)$.

Formulation of SOS polynomials

The use of SOS programming in HJB problems have been addressed in prior works, e.g. [6]. In our work, we approach the problem in three different SOS formulations:

- **Completing the square:** Adding higher degree terms to a given polynomial approximation to make it SOS.
- **Completing the square and collocation method:** Lower degree terms match a given polynomial approximation, collocation used to determine highest degree terms.
- **Collocation method:** Use Parrilo's formulation of SOS within a least-squares collocation method.

Here, the approaches are applied to the the past energy function $\mathcal{E}_\gamma^-(x)$. The same procedure can be applied to the future energy function $\mathcal{E}_\gamma^+(x)$.

Formulation of SOS polynomials

The use of SOS programming in HJB problems have been addressed in prior works, e.g. [6]. In our work, we approach the problem in three different SOS formulations:

- **Completing the square:** Adding higher degree terms to a given polynomial approximation to make it SOS.
- **Completing the square and collocation method:** Lower degree terms match a given polynomial approximation, collocation used to determine highest degree terms.
- **Collocation method:** Use Parrilo's formulation of SOS within a least-squares collocation method.

Here, the approaches are applied to the the past energy function $\mathcal{E}_\gamma^-(x)$. The same procedure can be applied to the future energy function $\mathcal{E}_\gamma^+(x)$.

Formulation of SOS polynomials

The use of SOS programming in HJB problems have been addressed in prior works, e.g. [6]. In our work, we approach the problem in three different SOS formulations:

- **Completing the square:** Adding higher degree terms to a given polynomial approximation to make it SOS.
- **Completing the square and collocation method:** Lower degree terms match a given polynomial approximation, collocation used to determine highest degree terms.
- **Collocation method:** Use Parrilo's formulation of SOS within a least-squares collocation method.

Here, the approaches are applied to the the past energy function $\mathcal{E}_\gamma^-(x)$. The same procedure can be applied to the future energy function $\mathcal{E}_\gamma^+(x)$.

Completing the square

- Given a degree d approximation to $\mathcal{E}_\gamma^-(x)$

$$v(x) = \frac{1}{2}\left(v_2^\top x^{②} + \cdots v_d^\top x^{ⓓ}\right).$$

- We propose a sum-of-squares approximation $\mathcal{E}_{sos}^-(x)$ as:

$$\mathcal{E}_{sos}^-(x) = \left(\tilde{v}_1^T x + \tilde{v}_2^T x^{②} + \cdots + \tilde{v}_{d-1}^T x^{\text{(d-1)}}\right)^2$$

- The $d-1$ coefficients of $\mathcal{E}_{sos}^-$ are found by matching the lowest degree terms in $v(x)$.
- This is done without the involvement of the HJB equation. The HJB information is implicitly embedded in the approximation $v(x)$.

Completing the square

- Given a degree d approximation to $\mathcal{E}_\gamma^-(x)$

$$v(x) = \frac{1}{2}\left(v_2^\top x^{②} + \cdots v_d^\top x^{ⓓ}\right).$$

- We propose a sum-of-squares approximation $\mathcal{E}_{sos}^-(x)$ as:

$$\mathcal{E}_{sos}^-(x) = \left(\tilde{v}_1^T x + \tilde{v}_2^T x^{②} + \cdots + \tilde{v}_{d-1}^T x^{\text{(d-1)}}\right)^2$$

- The $d-1$ coefficients of $\mathcal{E}_{sos}^-$ are found by matching the lowest degree terms in $v(x)$.
- This is done without the involvement of the HJB equation. The HJB information is implicitly embedded in the approximation $v(x)$.

Completing the square

- Given a degree d approximation to $\mathcal{E}_\gamma^-(x)$

$$v(x) = \frac{1}{2}\left(v_2^\top x^{②} + \cdots v_d^\top x^{ⓓ}\right).$$

- We propose a sum-of-squares approximation $\mathcal{E}_{sos}^-(x)$ as:

$$\mathcal{E}_{sos}^-(x) = \left(\tilde{v}_1^T x + \tilde{v}_2^T x^{②} + \cdots + \tilde{v}_{d-1}^T x^{\text{(d-1)}}\right)^2$$

- The $d-1$ coefficients of $\mathcal{E}_{sos}^-$ are found by matching the lowest degree terms in $v(x)$.
- This is done without the involvement of the HJB equation. The HJB information is implicitly embedded in the approximation $v(x)$.

Completing the square

- Given a degree d approximation to $\mathcal{E}_\gamma^-(x)$

$$v(x) = \frac{1}{2}\left(v_2^\top x^{\textcircled{2}} + \cdots v_d^\top x^{\textcircled{d}}\right).$$

- We propose a sum-of-squares approximation $\mathcal{E}_{sos}^-(x)$ as:

$$\mathcal{E}_{sos}^-(x) = \left(\tilde{v}_1^T x + \tilde{v}_2^T x^{\textcircled{2}} + \cdots + \tilde{v}_{d-1}^T x^{\textcircled{d-1}}\right)^2$$

- The $d-1$ coefficients of $\mathcal{E}_{sos}^-$ are found by matching the lowest degree terms in $v(x)$.
- This is done without the involvement of the HJB equation. The HJB information is implicitly embedded in the approximation $v(x)$.

Completing the square

Example with $n = 2$:

Given a degree $d = 3$ approximation $v(x)$. Let the degree of the SOS approximation be $2(d-1) = 4$. Then

$$\mathcal{E}_{sos}^{-}(x) = \left(\tilde{v}_1^T x + \tilde{v}_2^T x^{②}\right)^2$$

where $\tilde{v}_1^T \in \mathbb{R}^{2\times 2}$ and $\tilde{v}_2^T \in \mathbb{R}^{2\times 4}$. Expanding this,

$$\mathcal{E}_{sos}^{-}(x) = x^T \tilde{v}_1 \tilde{v}_1^T x + 2x^T \tilde{v}_1 \tilde{v}_2^T x^{②} + (x^{②})^T \tilde{v}_2 \tilde{v}_2^T x^{②}$$

$$\mathcal{E}_{sos}^{-}(x) = \mathrm{vec}(\tilde{v}_1 \tilde{v}_1^T)^T x^{②} + 2\mathrm{vec}(\tilde{v}_1 \tilde{v}_2^T)^T x^{③} + \mathrm{vec}(\tilde{v}_2 \tilde{v}_2^T)^T x^{④}$$

Completing the square

Example with $n = 2$:

Given a degree $d = 3$ approximation $v(x)$. Let the degree of the SOS approximation be $2(d-1) = 4$. Then

$$\mathcal{E}^-_{sos}(x) = \left(\tilde{v}_1^T x + \tilde{v}_2^T x^{②}\right)^2$$

where $\tilde{v}_1^T \in \mathbb{R}^{2\times 2}$ and $\tilde{v}_2^T \in \mathbb{R}^{2\times 4}$. Expanding this,

$$\mathcal{E}^-_{sos}(x) = x^T \tilde{v}_1 \tilde{v}_1^T x + 2x^T \tilde{v}_1 \tilde{v}_2^T x^{②} + (x^{②})^T \tilde{v}_2 \tilde{v}_2^T x^{②}$$

$$\mathcal{E}^-_{sos}(x) = \mathrm{vec}(\tilde{v}_1 \tilde{v}_1^T)^T x^{②} + 2\mathrm{vec}(\tilde{v}_1 \tilde{v}_2^T)^T x^{③} + \mathrm{vec}(\tilde{v}_2 \tilde{v}_2^T)^T x^{④}$$

Completing the square

Example (continued):

Matching the $O(x^{②})$ and $O(x^{③})$ terms leads to the following systems of equations:

$$\tilde{v}_1 \tilde{v}_1^T = \frac{1}{2} V_2 \qquad \text{(solve by Cholesky)}$$

$$2\tilde{v}_1 \tilde{v}_2^T = \frac{1}{2} V_3 \qquad \text{(solve by backsubstitution)}$$

where $V_2 = \text{reshape}(v_2^T, 2, 2)$ and $V_3 = \text{reshape}(v_3^T, 2, 4)$.

Adding collocation

- In this approach, we write $\mathcal{E}^-_{sos,c}(x)$ in the same polynomial squared form:

$$\mathcal{E}^-_{sos,c}(x) \equiv \left(\tilde{v}_1^T x + \tilde{v}_2^T x^{②} + \cdots + \tilde{v}_{d-1}^T x^{(d-1)} + \tilde{v}_d^T x^{ⓓ}\right)^2$$

- and we solve for the first $d-1$ coefficients by matching a degree $2d-1$ polynomial approximation.
- Finally, optimizing the residual function given by:

$$J(\tilde{v}_d^T) = \sum_{k=1}^{N} (HJB(x_k))^2,$$

where N is the number of sample points in a domain $\Omega \subset \mathbb{R}^n$ and $HJB(x)$ is the residual of the HJB equation, we get:

$$\tilde{v}_d^T = \underset{\tilde{v}_d^T \in \mathbb{R}^{n^d}}{\operatorname{argmin}} J(\tilde{v}_d^T).$$

Adding collocation

- In this approach, we write $\mathcal{E}^-_{sos,c}(x)$ in the same polynomial squared form:

$$\mathcal{E}^-_{sos,c}(x) \equiv \left(\tilde{v}_1^T x + \tilde{v}_2^T x^{②} + \cdots + \tilde{v}_{d-1}^T x^{\textcircled{d-1}} + \tilde{v}_d^T x^{\textcircled{d}}\right)^2$$

- and we solve for the first $d-1$ coefficients by matching a degree $2d-1$ polynomial approximation.
- Finally, optimizing the residual function given by:

$$J(\tilde{v}_d^T) = \sum_{k=1}^{N} (HJB(x_k))^2,$$

where N is the number of sample points in a domain $\Omega \subset \mathbb{R}^n$ and $HJB(x)$ is the residual of the HJB equation, we get:

$$\tilde{v}_d^T = \underset{\tilde{v}_d^T \in \mathbb{R}^{n^d}}{\text{argmin}}\ J(\tilde{v}_d^T).$$

Adding collocation

- In this approach, we write $\mathcal{E}^-_{sos,c}(x)$ in the same polynomial squared form:

$$\mathcal{E}^-_{sos,c}(x) \equiv \left(\tilde{v}_1^T x + \tilde{v}_2^T x^{\textcircled{2}} + \cdots + \tilde{v}_{d-1}^T x^{\textcircled{d-1}} + \tilde{v}_d^T x^{\textcircled{d}}\right)^2$$

- and we solve for the first $d-1$ coefficients by matching a degree $2d-1$ polynomial approximation.
- Finally, optimizing the residual function given by:

$$J(\tilde{v}_d^T) = \sum_{k=1}^{N} (HJB(x_k))^2,$$

where N is the number of sample points in a domain $\Omega \subset \mathbb{R}^n$ and $HJB(x)$ is the residual of the HJB equation, we get:

$$\tilde{v}_d^T = \underset{\tilde{v}_d^T \in \mathbb{R}^{n^d}}{\text{argmin}}\, J(\tilde{v}_d^T).$$

Collocation method

- These two complete the square approaches resolve the negativity issue, but relies on having a good polynomial approximation. These are typically accurate locally, but have no guarantee away from the origin.
- Using Parrilo's formulation, we write $\mathcal{E}_p^-(x)$ as:

$$\mathcal{E}_p^-(x) = Z(x)^T QZ(x)$$

where $Z(x) \in \mathbb{R}^\nu$ is a vector of all monomials of degree $\leq d$ and $Q \in \mathbb{R}^{\nu\times\nu}$ is a positive definite matrix. [1]

Collocation method

- These two complete the square approaches resolve the negativity issue, but relies on having a good polynomial approximation. These are typically accurate locally, but have no guarantee away from the origin.
- Using Parrilo's formulation, we write $\mathcal{E}_p^-(x)$ as:

$$\mathcal{E}_p^-(x) = Z(x)^T Q Z(x)$$

where $Z(x) \in \mathbb{R}^\nu$ is a vector of all monomials of degree $\leq d$ and $Q \in \mathbb{R}^{\nu \times \nu}$ is a positive definite matrix. [1]

Collocation method

- The number of monomial terms, ν, in $Z(x)$ is defined as follows:

$$\nu = \sum_{i=1}^{d} deg_i(n),$$

where $deg_i(n)$ is the number of monomial terms of degree i in dimension n defined as follows:

$$deg_i(n) = \sum_{j=1}^{n} deg_{i-1}(j) \quad \text{for } i \geq 2$$

with $deg_1(n) = n$

Collocation method

- Q has a Cholesky factorization:

$$\mathcal{E}_p^-(x) = p_{sos}(x) = Z(x)^T L L^T Z(x)$$

where $L \in \mathbb{R}^{\nu \times \nu}$ is the Cholesky factor of Q

- The optimization problem arguments are the entries of L, and the residual function in this case is:

$$J(L) = \sum_{k=1}^{N} (HJB(x_k))^2$$

where N is the number of sample points in a region $\Omega \in \mathbb{R}^n$, $HJB(x)$ is the residual of the HJB equation and

$$L^* = \underset{L \in \mathbb{R}^{\nu \times \nu}}{\text{argmin}}\ J(L)$$

is the optimal solution.

Collocation method

- Q has a Cholesky factorization:

$$\mathcal{E}_p^-(x) = p_{sos}(x) = Z(x)^T L L^T Z(x)$$

where $L \in \mathbb{R}^{\nu \times \nu}$ is the Cholesky factor of Q

- The optimization problem arguments are the entries of L, and the residual function in this case is:

$$J(L) = \sum_{k=1}^{N} (HJB(x_k))^2$$

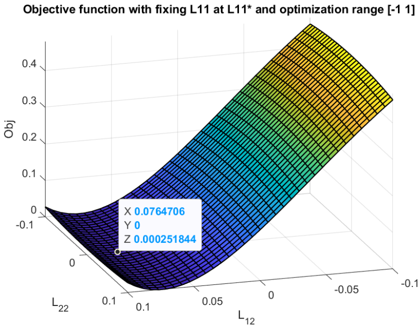
where N is the number of sample points in a region $\Omega \in \mathbb{R}^n$, $HJB(x)$ is the residual of the HJB equation and

$$L^* = \operatorname*{argmin}_{L \in \mathbb{R}^{\nu \times \nu}} J(L)$$

is the optimal solution.

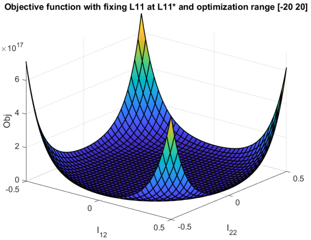
Issues for larger domains in $\mathbb{R}^n$

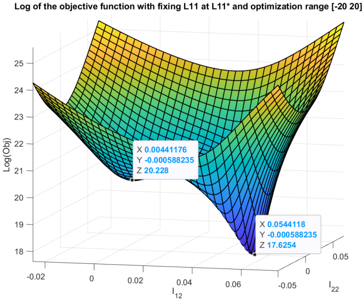
- Optimization for small regions around the origin is a convex optimization problem. For example, a degree 4 approximation in $\Omega = [-1; 1] \subset \mathbb{R}$ has a graph (fixing L_{11} at L_{11}^*) [4]



Issues for large regions Ω in $\mathbb{R}^n$

- Considering larger domains, e.g. $\Omega = [-20; 20] \subset \mathbb{R}$, the problem becomes non-convex even for low dimensions:





Windowing procedure

- Windowing can overcome this issue. Windowing procedure can be seen as an iterative way of solving the optimization problem.
- We start with an optimization in small region around the origin, where the SOS approximation will mimic the behaviour of the lower order terms of the polynomial approximation.
- Then, the optimization results are set as starting values for problems with larger domains and increasing numbers of sample points.

$$L^{*(N_1)}_{[-1;1]} \rightarrow L^{*(N_2)}_{[-10;10]} \rightarrow L^{*(N_3)}_{[-20;20]} \rightarrow \cdots$$

Windowing procedure

- Windowing can overcome this issue. Windowing procedure can be seen as an iterative way of solving the optimization problem.
- We start with an optimization in small region around the origin, where the SOS approximation will mimic the behaviour of the lower order terms of the polynomial approximation.
- Then, the optimization results are set as starting values for problems with larger domains and increasing numbers of sample points.

$$L^{*(N_1)}_{[-1;1]} \to L^{*(N_2)}_{[-10;10]} \to L^{*(N_3)}_{[-20;20]} \to \cdots$$

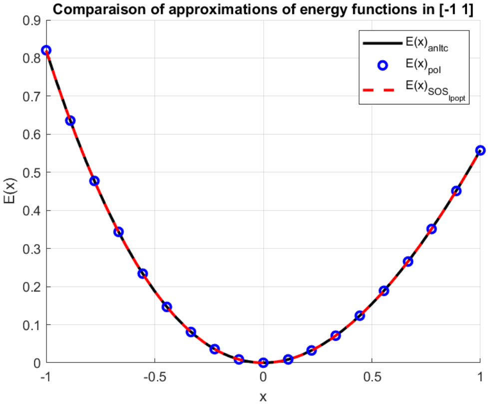
Windowing procedure

- Windowing can overcome this issue. Windowing procedure can be seen as an iterative way of solving the optimization problem.
- We start with an optimization in small region around the origin, where the SOS approximation will mimic the behaviour of the lower order terms of the polynomial approximation.
- Then, the optimization results are set as starting values for problems with larger domains and increasing numbers of sample points.

$$L^{*(N_1)}_{[-1;1]} \rightarrow L^{*(N_2)}_{[-10;10]} \rightarrow L^{*(N_3)}_{[-20;20]} \rightarrow \cdots$$

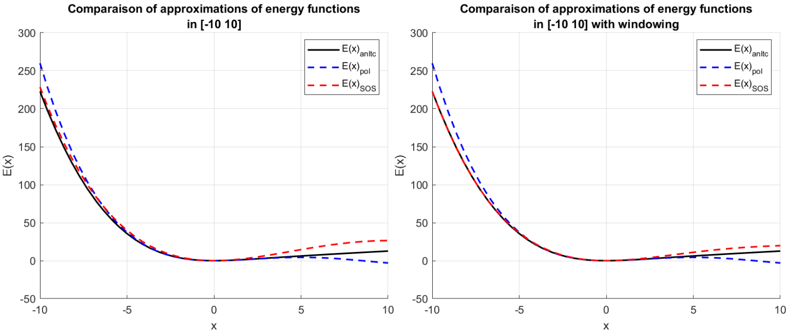
Numerical results in 1D

We simulate the collocation method on the example from [4], in $\Omega = [-1; 1]$, with a degree 4 approximation:



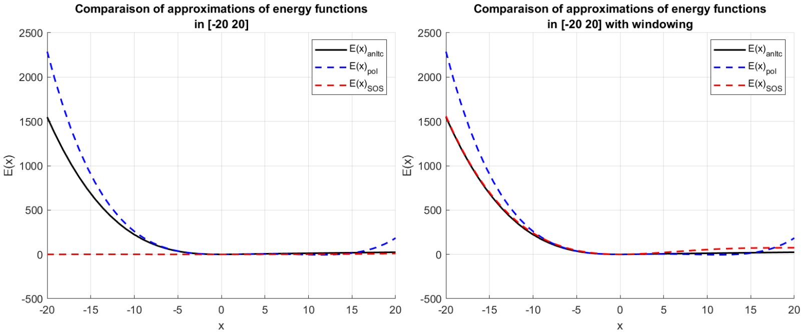
Numerical results in 1D

Optimization for $\Omega = [-10; 10]$ with and without windowing:



Numerical results in 1D

Optimization for $\Omega = [-20; 20]$ with and without windowing:



Numerical results in 1D

- We compare the quality of the approximations based on the relative error with respect to the analytical solution:

$$\text{Error} = \frac{||\mathcal{E}^-_{approx}(x) - \mathcal{E}^-_{exact}(x)||_2}{||\mathcal{E}^-_{exact}(x)||_2}$$

Ω	Poly Approximation	SOS Approximation
$[-1; 1]$	8.15×10^{-4}	7.16×10^{-4}
$[-10; 10]$	1.52×10^{-1}	5.67×10^{-2}
$[-20; 20]$	4.02×10^{-1}	6.65×10^{-2}

Numerical results in 2D

- In the 2D and higher dimensions settings, we don't generally have the exact solution, so we will measure the quality of the approximation by using it as control for the system:

$$\dot{x}(t) = -f(x(t)) + Bu(x(t))$$

where

$$u^*(x) = -B^T \frac{\partial \mathcal{E}_\gamma^-(x)}{\partial x}$$

and compare its value at $x = x_0$ with:

$$\mathcal{E}_{integral}^-(x_0) = \frac{1}{2}\int_0^\infty (1-\gamma^{-2})||y(t)||^2 + ||u(t)^*||^2 dt$$

such that $x(-\infty) = 0 \quad \text{and} \quad x(0) = x_0$

Numerical results in 2D

- We apply the collocation method with windowing to the following system, which is analogous to the 1D case system: [4]

$$A = -2I \quad B = 2I \quad C = 3I \quad N = [I \quad I] \quad \text{and} \quad \gamma = \sqrt{2}$$

where I is the identity matrix in $\mathbb{R}^{2\times 2}$

- We start from the square domain $\Omega_1 = [-0.1; 0.1] \times [-0.1; 0.1]$, using the optimal coefficients as initial points for $\Omega_2 = [-1; 1] \times [-1; 1]$, and continuing to $\Omega_3 = [-5; 5] \times [-5; 5]$ and finally to $\Omega_4 = [-7; 7] \times [-7; 7]$.

Numerical results in 2D

- We apply the collocation method with windowing to the following system, which is analogous to the 1D case system: [4]

$$A = -2I \quad B = 2I \quad C = 3I \quad N = [I \quad I] \quad \text{and} \quad \gamma = \sqrt{2}$$

  where I is the identity matrix in $\mathbb{R}^{2\times 2}$
- We start from the square domain $\Omega_1 = [-0.1; 0.1] \times [-0.1; 0.1]$, using the optimal coefficients as initial points for $\Omega_2 = [-1; 1] \times [-1; 1]$, and continuing to $\Omega_3 = [-5; 5] \times [-5; 5]$ and finally to $\Omega_4 = [-7; 7] \times [-7; 7]$.

Numerical results in 2D

- The relative error, defined as:

$$\text{Error} = \frac{||\mathcal{E}^-_{approx}(x_0) - \mathcal{E}^-_{integral}(x_0)||_2}{||\mathcal{E}^-_{integral}(x_0)||_2}$$

at 4 different initial conditions x_0 is summarized as follows:

$x_0 = (x_{01}, x_{02})$	Poly Approximation	SOS Approximation
$(-7, -7)$	1.4875×10^{-1}	9.4801×10^{-3}
$(-7, 7)$	8.875×10^{-1}	7.3976×10^{-2}
$(7, -7)$	8.875×10^{-1}	5.5123×10^{-2}
$(7, 7)$	2.4754×10^{-1}	3.0498×10^{-1}

Numerical results in 2D

- We also observed that the SOS approximation is also behaving as a good control outside the optimization region. The following results illustrate the observation:

$x_0 = (x_{01}, x_{02})$	Poly Approximation	SOS Approximation
$(-10, -10)$	2.1308×10^{-1}	3.145×10^{-2}
$(-10, 10)$	Divergence	5.8141×10^{-2}
$(10, -10)$	Divergence	1.0187×10^{-1}
$(10, 10)$	6.757×10^{-1}	3.0512×10^{-1}

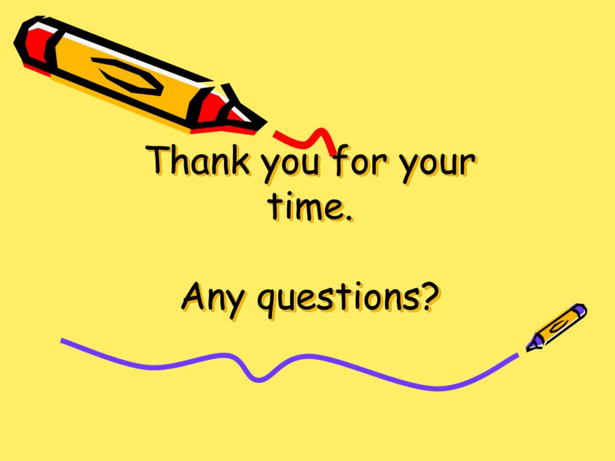
Summary

- We developed a collocation approach on how to build polynomial SOS approximations of the past energy function, with a possibility of enlarging the domain with the windowing procedure.
- We have shown the effectiveness of the proposed approach on both the 1D and 2D settings.

Future work

- Applying the proposed method on higher dimensional settings.
- Exploring more sampling strategies.
- Using more general SOS (bases different than monomials or polynomials in general).
- Explore the convergence theory of the proposed method.

Thank you



References

[1] H. Adjerid and J. Borggaard. *Sum-of-squares approximations to energy functions* (in preparation).

[2] J. M. Scherpen. *H_∞ balancing for nonlinear systems*. International Journal of Robust and Nonlinear Control, 6(7):645–668, 1996.

[3] J. M. Scherpen and A. van der Schaft. *Normalized coprime factorizations and balancing for unstable nonlinear systems*. International Journal of Control, 60(6):1193–1222, 1994.

[4] B. Kramer et al. *Nonlinear balanced truncation: Part 1–computing energy functions*. arXiv:2209.07645 [math.OC]

[5] P. A. Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. PhD thesis, California Institute of Technology, Pasadena, CA, 2000.

[6] D. Lasagna et al. *Sum-of-squares approach to feedback control of laminar wake flows*. J. Fluid Mech. (2016), vol. 809, pp. 628–663.