Information and Inference: A Journal of the IMA (2022) **00**, 1–30
https://doi.org/10.1093/imaiai/iaac021

Non-dissipative and structure-preserving emulators via spherical optimization

Dihan Dai[†]
Department of Mathematics, and Scientific Computing and Imaging (SCI) Institute, University of Utah, 201 Presidents Cir, 84112 Salt Lake City, USA
[†]Corresponding author. Email: dai@math.utah.edu

Yekaterina Epshteyn
Department of Mathematics, University of Utah, 201 Presidents Cir, 84112 Salt Lake City, USA

and

Akil Narayan
Department of Mathematics, and Scientific Computing and Imaging (SCI) Institute, University of Utah, 201 Presidents Cir, 84112 Salt Lake City, USA

[Received on 26 August 2021; revised on 24 April 2022; accepted on 5 July 2022]

Approximating a function with a finite series, e.g., involving polynomials or trigonometric functions, is a critical tool in computing and data analysis. The construction of such approximations via now-standard approaches like least squares or compressive sampling does not ensure that the approximation adheres to certain convex linear structural constraints, such as positivity or monotonicity. Existing approaches that ensure such structure are norm-dissipative and this can have a deleterious impact when applying these approaches, e.g., when numerical solving partial differential equations. We present a new framework that enforces via optimization such structure on approximations and is simultaneously norm-preserving. This results in a conceptually simple convex optimization problem on the sphere, but the feasible set for such problems can be very complex. We establish well-posedness of the optimization problem through results on spherical convexity and design several spherical-projection-based algorithms to numerically compute the solution. Finally, we demonstrate the effectiveness of this approach through several numerical examples.

Keywords: structure-preserving emulators; high-order accuracy; quadratic programming; geodesic convex optimization.

1. Introduction

Approximating an unknown function with a superposition of basis functions (e.g., polynomials or Fourier series) is a widely -used technique in computing and numerical analysis. For example, when solving a system of partial differential equations (PDEs), the class of spectral methods proposes such a superposition ansatz and determines the coefficients through minimization conditions on the PDE residual. Traditionally, fundamental properties of the approximation, such as stability, accuracy and computational efficiency are major considerations for the approximations. However, for certain problems, approximations are required to preserve certain implicit 'structures,' i.e., approximations should inherit certain desirable qualitative features of the original function. Such structure can include positivity, monotonicity, conservation of energy, etc. An approximation that fails to be structure-preserving may lead to numerical instability or even the failure of numerical schemes [36]. From the

broader viewpoint of building predictive emulators from data, this structure can be crucial to generate a meaningful emulator; for example, emulators built to predict population trends should not predict negative values. In this manuscript, we consider building approximations that respect general families of linear homogeneous convex inequality constraints (for which positivity and monotonicity are examples) along with a single quadratic equality constraint (an energy constraint).

Based on the existing framework for linear inequality constraints [35], we impose a new *spherical* constraint, i.e., a quadratic constraint in addition to the linear constraints. While a seemingly benign addition, this extra constraint substantially changes the optimization problem and its properties. With a coordinate vector $\widehat{\boldsymbol{p}}$ provided (e.g., a vector of Fourier coefficients), the formulation we consider in this paper gives rise to an optimization problem of the form,

$$\begin{aligned} &\min_{\widehat{\boldsymbol{v}}\in\mathbb{R}^N} \|\widehat{\boldsymbol{v}}-\widehat{\boldsymbol{p}}\|_2^2 \\ \text{s.t. } &g_k(\widehat{\boldsymbol{v}},y)\le 0, \quad \forall y\in\omega_k, k\in[K] \\ &\|\widehat{\boldsymbol{v}}\|_2=\|\widehat{\boldsymbol{p}}\|_2, \end{aligned} \tag{1.1}$$

where the linear constraints are given by the y-parameterized scalar-valued functions $g_k(\cdot,y)$ and the energy-preserving constraint is given by the equality constraint $\|\widehat{\boldsymbol{v}}\|_2=\|\widehat{\boldsymbol{p}}\|_2$. The parameters y can take values from a (possibly uncountably infinite) set ω_k, and hence the feasible set can be very complex. Generally, the feasible set in (1.1) is the intersection of a finite collection of homogeneous convex cones and a sphere. The model (1.1) corresponds to a semi-infinite programming (SIP) problem [23]. In several SIP algorithms, a discrete approximation to the domain ω_k is constructed (and perhaps refined). For linear constraints corresponding to positivity, this would correspond to requiring positivity at only a finite collection of points on the domain and hence structure is only preserved at a discrete set of points instead of on the whole domain. An additional difficulty is that the feasible set is a subset on the surface of a sphere, which is not a convex set in Euclidean space and hence the approaches from [35] do not apply. Thus, computationally solving (1.1) can be very challenging.

1.1 *Related problems and approaches*

There is existing literature on the study of optimization over ellipsoids (or spheres), which is closely related to the solution of the subproblems in the class of trust-region methods [18, 20, 30, 31]. However, in those approaches, the number of linear constraints is finite, and therefore such approaches are not directly applicable in our setting.

There are existing energy-preserving numerical methods that focus on energy-conservation of a Hamiltonian system, where a differential equation is discretized in a special way so that the energy of the discretized system is preserved, see [5, 10, 21, 29]. However, our focus is to preserve the energy of the approximation to a given function rather than the energy of a differential equation system, which is a different problem. In addition, methods built for differential equations assume very particular types of discretizations; the formulation we investigate in this paper applies to general discretizations.

A number of techniques have been proposed for preserving special kinds of structure for special choices of basis functions. To ensure positivity preservation, one can simply enforce positivity at a finite collection of points in the computational domain. The corresponding feasible set is a convex polytope and there are several algorithms available to computationally solve this problem [6]. Unfortunately, such techniques do not guarantee positivity of the approximation over the entire domain (a generally uncountable Euclidean set). An alternative to constraints over a finite set is to use special mappings.

For example, one can approximate $\sqrt{f}$ and square the resulting approximation, or approximate $\log f$ and subsequently exponentiate the approximation in order to guarantee the resulting approximation is positive. However, such mapping functions are not easy to construct for more complicated constraints, and the introduction of such maps can affect accuracy; for example, $x \mapsto \sqrt{x}$ is not smooth at $x = 0$. For univariate polynomial approximation, one can take advantage of the special representations of nonnegative polynomials (Lukács theorem) to develop optimizations with a finite number of constraints [8].

In these univariate polynomial approximation methods that leverage Lukács's theorem, Nesterov [26] represented the convex cone of nonnegative polynomials over an interval by the linear images of the cone of positive semidefinite matrices. The corresponding optimization problem can then be solved by semidefinite programming (SDP) [34]. In [27], Nie and Demmel used the Nesterov's idea to determine a rational function satisfying several shape constraints. In [1], Allen and Kirby adapt the cone representation for Bernstein polynomial basis representation. In particular, they considered the best L^2- approximation and imposed a mass conservation constraint (a linear equality). They also proposed approximation for multivariate polynomials, but the corresponding feasible set is a subset (rather than an exact characterization) of the set of all nonnegative polynomials. We also note that there exists theoretical bounds for structure-preserving univariate polynomial approximations [1, 8, 13]. In special multivariate polynomial settings with a discrete number of constraints, additional estimates are possible [22, 25]. In this work, we are interested in the problem with more generality than the previously mentioned approaches. We allow more general (non-polynomial) basis functions and do not impose any restrictions for the domain or dimension of the space.

Other approaches include using an adaptive construction scheme for certain kinds of constraints [4] or linearly scaling the high-order coefficients of the polynomial to limit the oscillations [36]. Finally, we note that there are existing theoretical investigations for structure-preserving approximation in [2, 3, 14, 28], but these investigations do not translate into algorithms.

Our approach extends the recent technique in [35], which considers building approximations with *linear* structure, in the sense that the constraints are linear with respect to the approximant. In [35], the authors formalize a model for the structure-preserving problem with linear constraints, which applies to general, nontrivial linear structure. Under a mild condition, the corresponding function approximation problem can be cast to a semi-infinite convex optimization problem in a finite-dimensional Euclidean space with a unique solution. In addition, the work in [35] develops several projection-based algorithms to preserve the desired structures. However, their method, which amounts to filtering the approximation in a nonlinear manner, does not preserve the L^2 norm, and thus is dissipative. The work of this paper preserves the quadratic (energy) norm through a modified formulation of the problem. This slight modification results in nontrivial changes to well-posedness and algorithmic development that we address.

1.2 *Contributions of this paper*

In this work, we are interested in providing theory and algorithms to address non-dissipative, structure-preserving function approximation methods of the form (1.1). Using notions of spherical convexity and spherical projections [16, 17], we show that the corresponding function approximation problem can be converted to a spherically convex feasibility problem, and establish uniqueness of the solution under mild conditions, see Theorem 3.1. Based on our theoretical results and by extending algorithms in [35], we propose three algorithms to solve the spherically convex feasibility problem; see sections 4.2, 4.3, and 4.4. Our algorithms do not rely on the discretization of the domain and therefore differ from many existing SIP algorithms [19, 32].

The setup of the general problem is as follows: We first assume that the unconstrained approximation to the unknown function is available, e.g., an unconstrained projection of the unknown function onto a finite-dimensional subspace. The unconstrained approximation is then *post-processed* via our algorithms so that the linear constraints, such as positivity, are satisfied without augmenting or reducing the quadratic energy of the approximant.

This paper is structured as follows. In Section 2, we give the theoretical framework of the structure-preserving function approximation problem as well as formalization of the constraints. In Section 3, we provide a brief overview of spherical geometry and present the uniqueness result of the function approximation problem. In Section 4, we discuss projections on the sphere and develop two algorithms for solving the function approximation problem. Finally, in Section 5, we demonstrate the efficacy of our algorithms with numerical results for polynomial and Fourier series approximations. Our energy- and structure-preserving results show similar rates of convergence as those of the unconstrained approximation as the subspace is refined.

2. Setup

Let $\Omega \subseteq \mathbb{R}^d$ be a spatial domain. Consider the Hilbert space H formed by scalar-valued functions over Ω with inner product $\langle \cdot, \cdot \rangle_H$,

$$H = H(\Omega) := \{f : \Omega \to \mathbb{R} \mid \|f\|_H < \infty\}, \qquad \|f\|^2 := \langle f, f \rangle_H,$$

A prototypical example is $H = L^2(\Omega; \mathbb{R})$. Let $V \subseteq H$ be an N-dimensional subspace spanned by orthonormal basis functions $\{v_n\}_{n\in[N]}$,

$$V = \text{span}\{v_1, \cdots, v_N\}, \qquad \langle v_j, v_k \rangle_H = \delta_{j,k}, \qquad j, k \in [N],$$

where $\delta_{j,k}$ is Kronecker delta function, and $[N] := \{1, \cdots, N\}$. Our numerical examples will be restricted to $d = 1$ or $d = 2$ on a closed interval or a closed rectangle Ω, respectively, but the theoretical framework we develop holds for general choices of d and Ω.

We assume throughout this document that V has no common zeros on Ω, i.e., that,

$$\forall x \in \Omega \;\; \exists\, v \in V \text{ such that } v(x) \neq 0.$$

This assumption is true if, for example, V contains constant functions.

2.1 *The unconstrained problem – linear measurements*

We assume availability of an unconstrained function approximation scheme from H onto V using a finite collection of data. In this section, we briefly mention canonical approaches for accomplishing this via linear measurements, but our optimization problem is independent of how this unconstrained approximation is formed.

Let $u \in H$ be a function about which a finite number of observations $\{u_m\}_{m\in[M]} := \{\phi_m(u)\}_{m\in[M]} \subset \mathbb{R}$ are available, where $\phi_1, \cdots, \phi_M$ are M linear functionals on H, and are bounded on V. The functionals can be, e.g., v_m-projections $\langle \cdot, v_m \rangle$ or pointwise evaluations $\delta_{x_m}(\cdot)$, where δ_{x_m} is the Dirac mass centered

at $x_m \in \Omega$. An approximation $p \in V$ to u is frequently built by enforcing these linear measurements:

$$\text{Find } p = \sum_{n\in[N]} \widehat{p}_n v_n \text{ satisfying } \boldsymbol{A}\widehat{\boldsymbol{p}} = \boldsymbol{b}, \tag{2.1}$$

where

$$(\boldsymbol{A})_{m,n} = \phi_m(v_n), \qquad (m,n) \in [M]\times[N], \qquad \boldsymbol{b} = [\phi_1(u), \cdots, \phi_M(u)]^\top \in \mathbb{R}^M. \tag{2.2}$$

The condition $M = N$ is necessary for the problem (2.1) to be well-posed, and so in practice one relaxes (2.1) in appropriate ways depending on whether the system is under-/over-determined. For example, with $\widehat{\boldsymbol{v}}$ the v_n-coordinates of an element $v \in V$, one could relax (2.1) in the following ways:

$$\begin{aligned}
&(M > N)\ \widehat{\boldsymbol{p}} = \underset{\widehat{\boldsymbol{v}}\in\mathbb{R}^N}{\arg\min} \left\|\boldsymbol{A}\widehat{\boldsymbol{v}} - \boldsymbol{b}\right\|_2 && \text{(Least squares)}\\
&(M = N)\ \widehat{\boldsymbol{p}} = \boldsymbol{A}^{-1}\boldsymbol{b} && \text{(Interpolation)}\\
&(M < N)\ \widehat{\boldsymbol{p}} = \underset{\widehat{\boldsymbol{v}}\in\mathbb{R}^N:\, \boldsymbol{A}\widehat{\boldsymbol{v}}=\boldsymbol{b}}{\arg\min} \|\widehat{\boldsymbol{v}}\|_1 && \text{(Compressive sampling)}
\end{aligned} \tag{2.3}$$

where $\|\cdot\|_p$ is the $\ell^p([N])$ norm on vectors. Theory for well-posedness of each of these problems is mature [9, 12, 15, 33]. The numerical results in this paper utilize the interpolation ($M = N$) formulation above for simplicity, but this choice is independent of the theory and algorithms developed in this paper. The essential idea is that we assume the ability to construct $\widehat{\boldsymbol{p}}$ that, in the absence of linear inequality or quadratic equality constraints, is considered a good approximation to the original function u based on available data.

2.2 *The constraints*

In many practical situations, we require not only a solution to (2.1), but instead a solution that also obeys certain physical constraints, such as positivity over Ω. The unconstrained approximation (2.1) need not obey any such constraints, even if the original function u does obey them, which may lead to unphysical approximations. We therefore consider the problem of imposing these additional constraints. We consider simultaneously imposing two types of constraints: a (possibly uncountable) set of linear constraints, along with a single quadratic constraint.

2.2.1 *The linear constraints.* The constraints we consider in this section are motivated by the following examples of structural desiderata:

- positivity: $p(y) \geq 0$ for all $y \in \Omega$,
- monotonicity: $p'(y) \geq 0$ for all $y \in \Omega$,
- convexity: $p''(y) \geq 0$ for all $y \in \Omega$.

As it is pointed out in [35], these constraints can be characterized by families of linear constraints and a unique solution to the linearly constrained problem is guaranteed under some mild assumptions. *Linearity* in this context refers to linearity of the constraint with respect to the function p in V. In the rest of this subsection, we briefly review the abstract formulation introduced in [35].

Assume that there are K types of linear constraints (e.g., $K = 2$ if we simultaneously impose positivity and monotonicity). For each $k \in [K]$, each type of linear constraint is a family defined by the condition,

$$L_k(v, y) \leq 0, \qquad \forall y \in \omega_k, \tag{2.4}$$

where,

- ω_k is a subset (possibly containing uncountably many elements) of the spatial domain Ω,
- $L_k(\cdot, y)$ is a y-parameterized *unit-norm* element in the dual space V^*.

The feasible set of elements $v \in V$ that satisfy (2.4) for the family-k constraint is given by

$$E_k := \{v \in V \mid L_k(v, y) \leq 0, \ \ \forall y \in \omega_k\}. \tag{2.5}$$

Positivity, monotonicity, and convexity can be describes by the abstract formulation (2.4). The *linear feasible set* E^0 is the set of all $v \in V$ that satisfy all K constraints simultaneously, and hence is the intersection of all the E_k,

$$E^0 := \bigcap_{k\in[K]} E_k = \bigcap_{k\in[K]} \{v \in V \mid L_k(v, y) \leq 0, \ \ \forall y \in \omega_k\}. \tag{2.6}$$

Note that E^0 is always non-empty since it contains 0.

Since V is N-dimensional, we can identify the feasible set E^0 in V with a feasible set in the v_n-coordinate space $\mathbb{R}^N$. By the Riesz representation theorem, for any $L \in V^*$, there exists a unique Riesz representor $\ell \in V$ such that,

$$L(v) = \langle v, \ell \rangle_H, \qquad \forall v \in V.$$

The function $\ell \in V$ can also be written explicitly using the orthonormal basis $\{v_n\}_{n\in[N]}$,

$$\ell(\cdot) = \sum_{n=1}^{N} \widehat{\ell}_n v_n(\cdot), \qquad \widehat{\ell}_n = \langle \ell, v_n \rangle = L(v_n),$$

and the following relation holds,

$$\|L\|_{V^*} = \|\ell\|_V = \|\widehat{\boldsymbol{\ell}}\|, \qquad \widehat{\boldsymbol{\ell}} = (\widehat{\ell}_1, \cdots, \widehat{\ell}_N)^{\mathrm{T}}.$$

In what follows we denote the Riesz representor for $L_k(\cdot, y)$ by $\ell_k(\cdot, y)$ and the corresponding coordinate vector by $\widehat{\boldsymbol{\ell}}_k(y) \in \mathbb{R}^N$. Since $L_k(\cdot, y)$ is unit-norm, we have

$$\|L_k(\cdot, y)\|_{V^*} = \|\ell_k(y)\|_V = \|\widehat{\boldsymbol{\ell}}_k(y)\| = 1. \tag{2.7}$$

Finally, the set $C_k \subseteq \mathbb{R}^N$ corresponding to the feasible set $E_k \subseteq V$ is given by,

$$C_k = \bigcap_{y\in\omega_k} \left\{ \widehat{\boldsymbol{v}} \in \mathbb{R}^N \middle| \left\langle \widehat{\boldsymbol{v}}, \widehat{\boldsymbol{\ell}}_k(y) \right\rangle \leq 0 \right\} =: \bigcap_{y\in\omega_k} c_k(y), \qquad k \in [K], \tag{2.8}$$

and the set $C^0 \subseteq \mathbb{R}^N$ corresponding to E^0 is

$$C^0 = \bigcap_{k\in[K]} C_k. \tag{2.9}$$

It can be verified that all $C_k, k \in [K]$ are closed, convex cones in $\mathbb{R}^N$ (and that E_k is a closed convex cone in V) [35] and thus their intersection C^0 is also a convex cone. Note that, although C^0 is simply a convex cone, the geometry of C^0 can be very complicated with infinitely many extreme points since every C_k is the intersection of infinitely many half-spaces $c_k(y)$ if the ω_k is a set with infinite cardinality, e.g., ω_k is an interval.

REMARK 2.1. In [35], an additional r_k parameter is introduced to define *affine* convex cones as feasible sets. We specialize here to the homogeneous case $r_k = 0$, so that our cones all have vertices at the origin. If $r_k \neq 0$, the problem we consider in this paper is not necessarily well-posed; see Example 2.3.

We summarize one example from [35] to demonstrate the notation and how it can be specialized to familiar types of constraints.

EXAMPLE 2.1. (Positivity) Let $\Omega = [-1, 1]$ and V be any N-dimensional subspace of $L^2(\Omega) \bigcap L^\infty(\Omega)$. We want to impose a positivity-structure for $v \in V$: $v(x) \geq 0, \forall x \in \Omega$. Thus, only $K = 1$ family is needed and $\omega_1 = \Omega$. Fixing $y \in \omega_1$, the corresponding unit-norm linear operator is given by

$$L_1(v, y) := -\lambda(y)v(y), \qquad \lambda(y) = \left(\sum_{n=1}^{N} v_n(y)^2 \right)^{-\frac{1}{2}}, \tag{2.10}$$

where $\lambda(y)$ is a y-dependent normalization factor. The corresponding y-parameterized Riesz representor $\ell_1(\cdot, y)$ and its coordinate vector $\widehat{\boldsymbol{\ell}}_1(y)$ are, respectively,

$$\ell_1(\cdot, y) = -\lambda(y) \sum_{n=1}^{N} v_n(y)v_n(\cdot), \qquad \widehat{\boldsymbol{\ell}}_1(y) = \left[-\lambda(y)v_1(y), \cdots, -\lambda(y)v_N(y) \right]^\top. \tag{2.11}$$

Once the orthonormal basis is specified, $\widehat{\boldsymbol{\ell}}_1(y)$ can thus be explicitly identified.

2.2.2 *Constraints that are "determining"* In order to establish uniqueness of the solution to our quadratic-linear constrained problem, we require an additional condition on the linear constraints (L_k, ω_k) defining C^0.

DEFINITION 2.1. The set of constraints $(L_k, \omega_k)_{k\in[K]}$ is *V-determining* if

$$v \in V \text{ and } L_k(v, y) = 0 \; \forall y \in \omega_k \forall k \in [K] \qquad \Longrightarrow \qquad v = 0$$

We will assume V-determining linear constraints, which amounts to a technical assumption about the geometry of the associated $\mathbb{R}^N$-feasible set C^0 that we later exploit. The V-determining condition precludes certain problem setups, but all the practical situations we consider in this paper are V-determining. As a simple example, to enforce positivity for every point in Ω as in Example 2.1 we have that $L_1(v, y)$ is a scaled point evaluation at y. Therefore, the V-determining condition requires that if $v \in V$ satisfies $v(y) = 0$ for every $y \in \Omega$ then $v = 0$, which is a quite natural condition.

For more intuition, the following lists some additional examples, with $\Omega = [-1, 1]$, $K = 1$, and ℓ_1 the normalized point-evaluation operator in Example 2.1,

- If $V = \text{span}\{x^j | j = 0, \ldots, N-1\}$ and $|\omega_1| \geq N$, then the linear constraint is V-determining
- If $V = \text{span}\{x^j | j = 0, \ldots, N-1\}$ and $|\omega_1| < N$, then the linear constraint is *not* V-determining
- If $V = \text{span}\{x^j | j = 1, \ldots, N\}$, and $|\omega_1| \leq N$ with $0 \in \omega_1$, then the linear constraint set is *not* V-determining.
- If $V = \text{span}\{H(x), 1 - H(x)\}$, with H the Heaviside function, and $\omega_1 = [-1, 0.5]$, then the linear constraint set is V-determining.
- If $V = \text{span}\{H(x), 1 - H(x)\}$, with H the Heaviside function, and $\omega_1 = [-1, -0.5]$, then the linear constraint set is *not* V-determining.

Note that the V-determining condition is violated only for specialized cases, e.g., either when ω_1 has finite cardinality less than N, or when V contains very special types of functions. In all the numerical examples we consider, the linear constraints are V-determining.

2.2.3 *The quadratic energy constraint* In addition to the linear constraints, we further impose a single quadratic norm constraint analogous to an L^2-energy of the function. To be precise, we impose that our constrained solution must have same norm as the unconstrained solution,

$$\|v\|_H = \|p\|_H, \tag{2.12}$$

where p is the unconstrained solution with coordinate vector $\widehat{\boldsymbol{p}}$ from solving (2.3). The corresponding discretized constraint set in $\mathbb{R}^N$ is a sphere with radius $\|\widehat{\boldsymbol{p}}\|$,

$$C^H := \{\widehat{\boldsymbol{v}} \in \mathbb{R}^N \big| \|\widehat{\boldsymbol{v}}\|_{\mathbb{R}^N} = \|\widehat{\boldsymbol{p}}\|_{\mathbb{R}^N}\}, \tag{2.13}$$

where $\widehat{\boldsymbol{v}}$ is the coordinate vector for v.

We can now state the overall procedure we consider in this paper:

1. Given data $\{u_m\}_{m\in[M]}$, solve the unconstrained problem (2.3) to obtain the unconstrained solution $\widehat{\boldsymbol{p}}$.
2. Post-process the unconstrained solution p by solving the constrained problem

$$\boldsymbol{d} = \underset{\widehat{\boldsymbol{v}}\in C}{\arg\min} \frac{1}{2}\|\widehat{\boldsymbol{v}} - \widehat{\boldsymbol{p}}\|^2, \tag{2.14}$$

where $C = C^H \bigcap C^0$. This constrained problem optimizes a quadratic function over a subset C of a sphere in $\mathbb{R}^N$.

Our focus is on the theory and algorithms for the second step, post-processing the unconstrained solution to obtain a structure-preserving solution. We show in Theorem 3.1 that the problem above has a unique solution. Furthermore, in Section 4, we are able to naturally extend the existing algorithms proposed in [35] to the new optimization problem. We end this section with two examples that illustrate why some alternative formulations to the two-step procedure above do not necessarily result in well-posed problems.

EXAMPLE 2.2. (An alternative formulation with nonunique solutions) One possible alternative to our framework proposed above is to instead consider the following constrained problem

$$\boldsymbol{d} = \arg\min_{\widehat{\boldsymbol{v}} \in C} \frac{1}{2} \|\boldsymbol{A}\widehat{\boldsymbol{v}} - \boldsymbol{b}\|^2. \tag{2.15}$$

with $\boldsymbol{A}$ and $\boldsymbol{b}$ as introduced in Section 2.1. This formulation incorporates the constraints and a least-squares problem simultaneously. However, the solution to this alternative formulation (2.15) is *not necessarily unique*. The issue lies in the fact that if the singular values of the full-rank matrix $\boldsymbol{A}$ are not all equal to 1, then the problem corresponds to optimization over an ellipsoid, which can yield non-unique solutions.

For example, consider the case when $N = M = 2$. Let

$$\boldsymbol{A} = \begin{bmatrix} 0.4 & 0 \\ 0 & 1 \end{bmatrix}, \boldsymbol{b} = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}, C = \left\{ (\cos t, \sin t) \middle| 0.01 \leq t \leq \pi - 0.01 \right\}.$$

The constrained set C is a spherically convex set (Definition 3.4). The loss/cost function associated to (2.15) is

$$cost(t) = 0.4 \cos^2 t + (\sin t - 0.5)^2 = 0.6 \sin^2 t - \sin t + 0.65,$$

which has two distinct global minima over the feasible set C at $t = \arcsin(\frac{5}{6})$ and $t = \pi - \arcsin(\frac{5}{6})$.

EXAMPLE 2.3. (Nonhomogeneous cones as in Remark 2.1) In Remark 2.1, we mention that in previous work [35] an r_k parameter is introduced to include more general linear constraints. If $r_k \neq 0$, then our optimization problem does not necessarily have unique solutions. Consider the following problem. Let the feasible set

$$C = \left\{ (x, y) \middle| x^2 + y^2 = 1 \right\} \bigcap \left\{ (x, y) \middle| y \leq 4x + 2, y \leq -4x + 2 \right\}.$$

The feasible set consists of two disjoint arcs (red arcs in Figure 1). If the unconstrained solution lies in the middle of the dark green arc (the black dot), there will be two solutions to (2.14), one from each red arc.

3. Solution to the constrained optimization problem

In this section, we will study the solution to the constrained optimization (2.14). Specifically, we will show in Theorem 3.1 that the solution is unique under reasonable conditions.

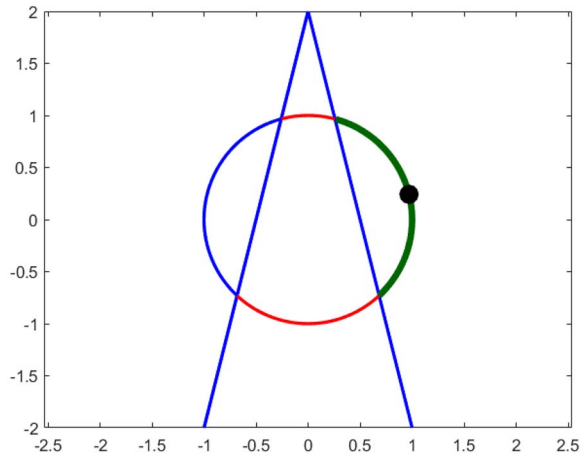

FIG. 1. An illustration to Example 2.3.

3.1 *Spherical geometry*

We introduce some definitions and relevant results for the spherical geometry in this subsection. We refer to [16, 17] for technical details. We largely focus on the unit sphere in this section, i.e., $C^H = \mathbb{S}^{N-1}$. In Section 4, we will use the more general origin-centered sphere of nonzero radius.

The *intrinsic distance* on $\mathbb{S}^{N-1}$ is defined to be the great circle distance between two points, which corresponds to the angle between the two unit vectors in the ambient space.

DEFINITION 3.1. (Intrinsic distance on the sphere) Given $\boldsymbol{u}, \boldsymbol{w} \in \mathbb{S}^{N-1}$, the intrinsic distance between them is

$$d(\boldsymbol{u}, \boldsymbol{w}) = \arccos\langle \boldsymbol{u}, \boldsymbol{w} \rangle. \tag{3.1}$$

If S is an origin-centered sphere with radius $r > 0$, then the intrinsic distance between $\boldsymbol{u}, \boldsymbol{w} \in S$ is

$$d_r(\boldsymbol{u}, \boldsymbol{w}) = r \arccos \langle \boldsymbol{u}/\|\boldsymbol{u}\|, \boldsymbol{w}/\|\boldsymbol{w}\| \rangle . \tag{3.2}$$

Note that the intrinsic distance between two points on a sphere of radius $r \neq 1$ is given by the intrinsic distance between the unit-normalized points, scaled by the radius.

DEFINITION 3.2. (Geodesics on a sphere) A geodesic on the unit sphere $\mathbb{S}^{N-1}$ is a *great circle*, i.e, the intersection curve of the sphere and a hyperplane in $\mathbb{R}^N$ through the origin. The unique arclength-parameterized geodesic segment from $\boldsymbol{u}$ to $\boldsymbol{w}$, where $\boldsymbol{u}, \boldsymbol{w} \in \mathbb{S}^{N-1}$ and $\boldsymbol{u} \neq \pm \boldsymbol{w}$, is given by

$$\gamma_{\boldsymbol{uw}}(t) = \csc d(\boldsymbol{u}, \boldsymbol{w}) \left[\boldsymbol{u} \sin(d(\boldsymbol{u}, \boldsymbol{w}) - t) + \boldsymbol{w} \sin t \right], \qquad t \in [0, d(\boldsymbol{u}, \boldsymbol{w})]. \tag{3.3}$$

The (non-unique) geodesic segments joining $\boldsymbol{u}$ and $-\boldsymbol{u}$, starting at $\boldsymbol{u}$ with velocity $\boldsymbol{v}$ satisfying $\|\boldsymbol{v}\| = 1$ at $\boldsymbol{u}$, is given by

$$\gamma_{\boldsymbol{u}\{-\boldsymbol{u}\}} := \cos(t)\boldsymbol{u} + \sin(t)\boldsymbol{v}, \qquad t \in [0, \pi]. \tag{3.4}$$

For a general sphere S centered at the origin with radius r, the geodesic segments can be defined via rescaling (3.3) or (3.4).

DEFINITION 3.3. (Exponential Mapping) The *exponential mapping* at $\boldsymbol{u}$ is defined to be

$$exp_{\boldsymbol{u}} : T_{\boldsymbol{u}}\mathbb{S}^{N-1} \to \mathbb{S}^{N-1}, \qquad \boldsymbol{v} \to \boldsymbol{u}\cos(\|\boldsymbol{v}\|) + \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}\sin(\|\boldsymbol{v}\|), \tag{3.5}$$

which maps an element on the tangent plane $T_{\boldsymbol{u}}\mathbb{S}^{N-1}$ at $\boldsymbol{u}$ to the endpoint of the geodesic segment of length $\|\boldsymbol{v}\|$ starting at $\boldsymbol{u}$ in the direction of $\boldsymbol{v}$. In (3.3), the geodesic segment can be expressed as $\gamma_{\boldsymbol{uw}}(t) = exp_{\boldsymbol{u}}(t\gamma'_{\boldsymbol{uw}}(0))$. For a general sphere S centered at the origin with radius r, the geodesic as well as the exponential mapping can be defined via rescaling (3.5).

DEFINITION 3.4. (Spherically convex set) A subset $C \subseteq \mathbb{S}^{N-1}$ is said to be *spherically convex* if for any $\boldsymbol{s}, \boldsymbol{t} \in C$, all the geodesic segments joining $\boldsymbol{s}$ and $\boldsymbol{t}$ are contained in C.

PROPOSITION 3.1. ([17], Proposition 2) Let $C \subseteq \mathbb{S}^{N-1}$. C is a spherically convex set if and only if the cone

$$K_C = \{z\boldsymbol{s} | \boldsymbol{s} \in C, z \in [0, +\infty)\} \tag{3.6}$$

is convex (in Euclidean sense) and pointed, i.e., $K_C \bigcap (-K_C) = \{\boldsymbol{0}\}$.

If we choose $C = C^H \cap C^0$ as is done in our problem articulated in Section 2.2.3, then $K_C = C^0$, the convex cone corresponding to only the linear equalities. Thus, in our framework spherical convexity of $C^H \cap C^0$ is determined by whether or not C^0 is a pointed cone.

PROPOSITION 3.2. A closed hemisphere is *not* spherically convex.

Proof. Noticing the existence of the antipodal points on a closed hemisphere, then there is a nontrivial $\boldsymbol{v}$ such that $\boldsymbol{v}, -\boldsymbol{v} \in K_C$. Therefore $K_C \cap (-K_C)$ contains at least one nontrivial point, and Proposition 3.1 yields the conclusion. □

One major utility of convex sets on the sphere is the ability to perform projections.

DEFINITION 3.5. (Spherical projection onto a closed convex set) Let $C \subset \mathbb{S}^{N-1}$ be a spherically convex, closed set. The projection of $\boldsymbol{z} \in \mathbb{S}^{N-1}$ onto C is defined to be:

$$\mathscr{P}^s_C(\boldsymbol{z}) = \left\{\boldsymbol{t} \in \mathbb{S}^{N-1} | d(\boldsymbol{t}, \boldsymbol{z}) \leq d(\boldsymbol{t}, \boldsymbol{r}), \forall \boldsymbol{r} \in C\right\}, \tag{3.7}$$

i.e., the nearest intrinsic distance projection.

The definition above does not immediately reveal uniqueness or computability for this type of projection, but the following proposition proved in [17] shows the relation between spherical projection

onto a closed spherically convex set and the Euclidean projection onto the convex cone spanned by the spherical convex set.

PROPOSITION 3.3. ([17], Proposition 8) Let $C \subseteq \mathbb{S}^{N-1}$ be a spherical convex set. Take $z \in \mathbb{S}^{N-1}$. Let $\boldsymbol{u} = \mathscr{P}_{K_C}(z)$, be the Euclidean projection of z onto K_C, the latter of which is defined in (3.6). If $\boldsymbol{u} \neq \boldsymbol{0}$, then the spherical projection of z onto C is unique, and is given by,

$$\mathscr{P}_C^s(z) = \frac{\boldsymbol{u}}{\|\boldsymbol{u}\|} = exp_z \boldsymbol{v},$$

where

$$\boldsymbol{v} = \left(-z\cot\theta + \frac{\boldsymbol{u}}{\|\boldsymbol{u}\|}\csc\theta\right)\theta, \qquad \theta = d(z, \boldsymbol{u}/\|\boldsymbol{u}\|).$$

3.2 *Uniqueness of the solution to (2.14)*

In this subsection, we present a uniqueness theorem for the solution to (2.14). Our first step is to introduce the formulation (3.8) below that is equivalent to (2.14).

LEMMA 3.1. The constrained optimization problem (2.14) is equivalent to finding the spherical projection of $\widehat{\boldsymbol{p}}$ onto the feasible set C,

$$\boldsymbol{d} = \arg\min_{\widehat{\boldsymbol{v}} \in C} d(\widehat{\boldsymbol{v}}, \widehat{\boldsymbol{p}}). \tag{3.8}$$

Proof. The proof is direct,

$$\begin{aligned}\arg\min_{\widehat{\boldsymbol{v}} \in C} \frac{1}{2}\|\widehat{\boldsymbol{v}} - \widehat{\boldsymbol{p}}\|^2 &= \arg\min_{\widehat{\boldsymbol{v}} \in C} \left(\frac{1}{2}(\|\widehat{\boldsymbol{v}}\|^2 + \|\widehat{\boldsymbol{p}}\|^2) - \langle \widehat{\boldsymbol{v}}, \widehat{\boldsymbol{p}}\rangle\right) \\ &= \arg\max_{\widehat{\boldsymbol{v}} \in C} \langle \widehat{\boldsymbol{v}}, \widehat{\boldsymbol{p}}\rangle, \\ &= \arg\min_{\widehat{\boldsymbol{v}} \in C} d(\widehat{\boldsymbol{v}}, \widehat{\boldsymbol{p}}). \end{aligned} \tag{3.9}$$

□

Using Lemma 3.1, we are able to show the following uniqueness theorem.

THEOREM 3.1. Assume the following hold for the constraint set $C = C^H \bigcap C^0$ defined by (2.8)-(2.9) and (2.13):

(a) The set of constraints $(L_k, \omega_k)_{k\in[K]}$ are V-determining in the sense of Definition 2.1.

(b) The Euclidean projection onto the linearly-constrained set satisfies $\mathscr{P}_{C^0}\widehat{\boldsymbol{p}} \neq \boldsymbol{0}$.

(c) The set C^H in (2.13) is $\mathbb{S}^{N-1}$, i.e., $\|\widehat{\boldsymbol{p}}\| = 1$.

Then the solution to (3.8) (or equivalently, (2.14)) is unique.

Proof. We first show that C is a closed spherically convex set. A subsequent application of Proposition 3.3 will prove the result.

Since $c_k(y)$ are closed half-spaces, $C^H \cap c_k(y)$ are closed hemispheres and

$$C = C^H \bigcap C^0 = \bigcap\nolimits_{y\in\omega_k, k\in[K]} (C^H \cap c_k(y))$$

is closed. On the other hand, from Proposition 3.1, the set C is spherically convex if and only if the cone K_C (see also Proposition 3.1 for the definition) is convex and pointed. Direct calculation shows that $K_C = C^0$, which has been shown to be a closed convex set in [35].

Define

$$W := C^0 \bigcap \{-C^0\} = \left\{ \boldsymbol{v} \middle| \left\langle \widehat{\boldsymbol{\ell}}_k(y), \boldsymbol{v} \right\rangle = 0, \forall y \in \omega_k, k \in [K] \right\}. \tag{3.10}$$

Take $\boldsymbol{x} \in W \subseteq C^0$. By assumption (a) and Definition 2.1, the only element v of V satisfying $L_k(v, y) = 0$ for every $y \in \omega_k$ and $k \in [K]$ is $v = 0$. Thus, $W = \{\mathbf{0}\}$ and therefore $K_C = C^0$ is pointed.

By Proposition 3.3 set C is a spherical convex set and the solution to (3.8) (or equivalently, (2.14)) is unique. □

COROLLARY 3.1. The conclusions in Theorem 3.1 hold with loosening assumption (c) to $\|\widehat{\boldsymbol{p}}\| > 0$.

The proof is direct since all arguments hold unchanged via scaling by $\|\widehat{\boldsymbol{p}}\| > 0$.

3.3 *Approximation errors from quadratic constraints*

The optimization problem we consider, (2.14) or equivalently (3.8), contains both linear and quadratic in/equality constraints in general multivariate and non-polynomial scenarios. As discussed in Section 1.1, there are many existing approaches in more specialized cases that consider only linear inequality constraints for polynomials (such as positivity). Such specializations sometimes yield approximation errors of the linearly constrained problem relative to unconstrained best approximation errors. (See, e.g., [1, 8, 13].)

Since our framework imposes an additional quadratic constraint in addition to linear constraints, a natural question is how much introduction of this extra constraint affects approximation errors. We provide a simple result below, indicating that the error committed by our linear-quadratic constraints is comparable to the error committed by only the linearly constrained problem. Therefore, any theory establishing error of the linearly constrained solution can immediately be ported to our linear-quadratically constrained case.

Precisely, we recall our notation from Section 2.1 where u is a given function in a Hilbert space H and p is some prescribed approximation to u from the finite-dimensional space V. Recalling that $E^0 \subset V$ is the closed convex set corresponding to our linear inequality constraints defined in (2.6), then the projections $\mathscr{P}_{E^0}(u)$ and $\mathscr{P}_{E^0}(p)$ onto $E^0 \subseteq V$ are unique (and the latter is equivalent to the vector projection $\mathscr{P}_{C^0}(\widehat{\boldsymbol{p}})$ in $\mathbb{R}^N$). A simple application of triangle inequality yields

$$\left\| u - \mathscr{P}_{E^0}(p) \right\|_H \leq \|u - p\|_H + \left\| p - \mathscr{P}_{E^0}(p) \right\|_H, \tag{3.11}$$

providing an upper bound on the error committed with *only* linear constraints. In particular, the first error term above is due to imposition of p as the initial unconstrained approximation, and the second term is due solely to the optimization/projection. In our linear-quadratically constrained case, if $\widehat{\boldsymbol{p}}^*$ is

any vector solution to the optimization (2.14) corresponding to the function $p^* \in V$, then we would hope that $\|u-p^*\|_H$ has error comparable to (3.11). The following lemma establishes precisely this statement.

LEMMA 3.2. Let $u \in H$ and $p \in V$ be given, where p has expansion coefficients $\widehat{\boldsymbol{p}}$. Let a linear-quadratic constraint set $C = C^H \cap C^0$ be given, where C^0 is the linear constraint set, and let $p^* \in V$ denote the function associated to a(ny) solution $\widehat{\boldsymbol{p}}^*$ of the linear-quadratic constrained problem (2.14) or equivalently (3.8). Then we have the following estimates,

$$\left\|u - p^*\right\|_H \leq \|u - p\|_H + 2\left\|p - \mathscr{P}_{E^0}(p)\right\|_H \tag{3.12}$$

$$\left\|u - p^*\right\|_H \leq 5\left\|p - p^\dagger\right\|_H + 2\sqrt{2}\left\|u - \mathscr{P}_{E^0}(u)\right\|_H, \tag{3.13}$$

where $p^\dagger$ is the H-best approximation from V, i.e.,

$$p^\dagger = \arg\min_{p \in V} \|u - p\|_H .$$

Before proving this result we emphasize the practicality of these statements: (3.12) provides an upper bound comparable to (3.11), indicating that the linear-quadratically constrained solution p^* has error similar to an approximation with only linear constraints. The estimate (3.13) is a statement in terms of best approximations, showing that the error in p^* splits into (i) a component $\|p - p^\dagger\|_H$ that is due to the quality of the initial unconstrained approximation provided, and (ii) the best approximation error from the linearly constrained space $E^0 \subseteq V$. In other words, modulo the quality of p relative to $p^\dagger$, the linear-quadratic constrained approximation p^* commits an error scaling precisely like the error in the *best possible* linearly-constrained approximation.

Proof. of Lemma 3.2 We first prove (3.12). Using the triangle inequality, we first divide the error into one part solely due to imposition of p as the starting unconstrained approximation, and a second part due to the linear-quadratic optimization p:

$$\left\|u - p^*\right\|_H \leq \|u - p\|_H + \left\|p - p^*\right\|_H \tag{3.14a}$$

We compute the last term using our notation in $\mathbb{R}^N$, i.e.,

$$\left\|p - p^*\right\|_H = \left\|\widehat{\boldsymbol{p}} - \widehat{\boldsymbol{p}}^*\right\| . \tag{3.14b}$$

To bound this norm, first assume that $\mathscr{P}_{C^0}(\widehat{\boldsymbol{p}})$, the solution to the linearly-constrained problem, is non-zero. Then define,

$$\boldsymbol{q} := \frac{\|\widehat{\boldsymbol{p}}\|}{\|\mathscr{P}_{C^0}(\widehat{\boldsymbol{p}})\|}\mathscr{P}_{C^0}(\widehat{\boldsymbol{p}}) \in C.$$

A direct computation shows,

$$\begin{aligned}\left\|\widehat{\boldsymbol{p}}-\widehat{\boldsymbol{p}}^{*}\right\| &\overset{(2.14)}{\leq} \left\|\widehat{\boldsymbol{p}}-\boldsymbol{q}\right\| \\ &\leq \left\|\widehat{\boldsymbol{p}}-\mathscr{P}_{C^0}(\widehat{\boldsymbol{p}})\right\| + \left\|\mathscr{P}_{C^0}(\widehat{\boldsymbol{p}})-\boldsymbol{q}\right\| \\ &= \left\|\widehat{\boldsymbol{p}}-\mathscr{P}_{C^0}(\widehat{\boldsymbol{p}})\right\| + \left|\left\|\mathscr{P}_{C^0}(\widehat{\boldsymbol{p}})\right\| - \left\|\widehat{\boldsymbol{p}}\right\|\right| \\ &\leq 2\left\|\widehat{\boldsymbol{p}}-\mathscr{P}_{C^0}(\widehat{\boldsymbol{p}})\right\|,\end{aligned}$$

where the last inequality uses the reverse triangle inequality. If we have $\mathscr{P}_{C^0}(\widehat{\boldsymbol{p}}) = \mathbf{0}$, then defining $\boldsymbol{q}$ instead as any feasible point in C and using the same computations along with $\|\boldsymbol{q}\| = \|\widehat{\boldsymbol{p}} - \mathscr{P}_{C^0}(\widehat{\boldsymbol{p}})\|$ proves the same inequality. Therefore,

$$\left\|\widehat{\boldsymbol{p}}-\widehat{\boldsymbol{p}}^{*}\right\| \leq 2\left\|\widehat{\boldsymbol{p}}-\mathscr{P}_{C^0}(\widehat{\boldsymbol{p}})\right\| \tag{3.15}$$

is true in general. Combining this with (3.14) proves (3.12).

Finally, since $u - p^\dagger$ is orthogonal to V then using the Pythagorean theorem can be used to begin the following inequality chain:

$$\begin{aligned}\left\|u-p^{*}\right\|_H &= \sqrt{\left\|u-p^\dagger\right\|_H^2 + \left\|p^\dagger-p^{*}\right\|_H^2} \\ &\leq \left\|u-p^\dagger\right\|_H + \left\|p^\dagger-p^{*}\right\|_H \\ &\leq \left\|p-p^\dagger\right\|_H + \left\|u-p^\dagger\right\|_H + \left\|p-p^{*}\right\|_H \\ &\overset{(3.15)}{\leq} \left\|p-p^\dagger\right\|_H + \left\|u-p^\dagger\right\|_H + 2\left\|p-\mathscr{P}_{E^0}(p)\right\|_H \\ &\leq \left\|p-p^\dagger\right\|_H + \left\|u-p^\dagger\right\|_H + 2\left(\left\|p-p^\dagger\right\|_H + \left\|p^\dagger-\mathscr{P}_{E^0}(p^\dagger)\right\|_H + \left\|\mathscr{P}_{E^0}(p^\dagger)-\mathscr{P}_{E^0}(p)\right\|_H\right) \\ &= 3\left\|p-p^\dagger\right\|_H + 2\left\|\mathscr{P}_{E^0}(p^\dagger)-\mathscr{P}_{E^0}(p)\right\|_H + \left(\left\|u-p^\dagger\right\|_H + 2\left\|p^\dagger-\mathscr{P}_{E^0}(p^\dagger)\right\|_H\right) \\ &\overset{(*)}{\leq} 5\left\|p-p^\dagger\right\|_H + \left(\left\|u-p^\dagger\right\|_H + 2\left\|p^\dagger-\mathscr{P}_{E^0}(p^\dagger)\right\|_H\right) \\ &\leq 5\left\|p-p^\dagger\right\|_H + 2\left(\left\|u-p^\dagger\right\|_H + \left\|p^\dagger-\mathscr{P}_{E^0}(p^\dagger)\right\|_H\right) \\ &\overset{(**)}{\leq} 5\left\|p-p^\dagger\right\|_H + 2\sqrt{2}\sqrt{\left\|u-p^\dagger\right\|_H^2 + \left\|p^\dagger-\mathscr{P}_{E^0}(p^\dagger)\right\|_H^2} \\ &= 5\left\|p-p^\dagger\right\|_H + 2\sqrt{2}\left\|u-\mathscr{P}_{E^0}(u)\right\|_H,\end{aligned} \tag{3.16}$$

where the inequality $(**)$ uses $|a| + |b| \leq \sqrt{2}\sqrt{a^2+b^2}$, and the inequality $(*)$ uses the non-expansive property of projections onto closed convex sets,

$$\left\|\mathscr{P}_{E^0}(q)-\mathscr{P}_{E^0}(r)\right\|_H \leq \|q-r\|_H,$$

see, e.g., [11, Theorem 3]. The final equality (3.16) is the fact that,

$$\left\|u-\mathscr{P}_{E^0}(u)\right\|_H^2=\min_{q\in E^0}\|u-q\|_H^2=\min_{q\in E^0}\left\|u-p^\dagger\right\|_H^2+\left\|p^\dagger-q\right\|_H^2=\left\|u-p^\dagger\right\|_H^2+\left\|p^\dagger-\mathscr{P}_{E^0}(p^\dagger)\right\|_H^2.$$

□

Note that Lemma 3.2 does not require uniqueness of the optimization (3.8). In particular, we do *not* require assumptions (a)-(c) from Theorem 3.1. The corollary (3.13) shows that if the unconstrained input p to our linear-quadratic optimization problem is a good approximation to u (in particular if it is the best approximation $p^\dagger$), then the optimization error committed is at most $2\sqrt{2}$ times that of the best possible linearly constrained approximation to u.

4. Algorithm: spherical projections

Having established the well-posedness of the problem (3.8), we proceed to discuss algorithms for solving the problem. In particular, we extend some procedures from [35] to the spherical optimization problem (3.8). We will shift our focus back to a general sphere centered at the origin with radius $r\neq 0$, equipped with the intrinsic distance $d_r(\cdot,\cdot)$ (Equation 3.2). Here, $r=\|\widehat{\boldsymbol{p}}\|\neq 0$ is the norm of the unconstrained solution.

4.1 *Spherical projection onto a closed hemisphere*

To start, we first compute the spherical projection of a point on the sphere onto a closed hemisphere $c_k(y)\bigcap C^H$, which later serves as an ingredient of our main algorithms. Note that Proposition 3.3 is not directly applicable since a closed hemisphere is not spherically convex The proof in this section is elementary but we provide it in order to make our work self-contained.

THEOREM 4.1. Let $\widehat{\boldsymbol{p}}$ be given, and fix (k,y). If $\widehat{\boldsymbol{\ell}}_k(y)$ is *not* parallel to $\widehat{\boldsymbol{p}}$, then the spherical projection of $\widehat{\boldsymbol{p}}$ onto the closed hemisphere $C^H\cap c_k(y)$ is unique, i.e., the solution $\boldsymbol{c}_k(\widehat{\boldsymbol{p}};y)$ to

$$\boldsymbol{c}_k(\widehat{\boldsymbol{p}};y)=\underset{\widehat{\boldsymbol{v}}\in C^H\cap c_k(y)}{\arg\min}\ d_r\left(\widehat{\boldsymbol{v}},\widehat{\boldsymbol{p}}\right), \tag{4.1}$$

is unique, and is given by

$$\boldsymbol{c}_k(\widehat{\boldsymbol{p}};y)=\begin{cases}\widehat{\boldsymbol{p}},\ \widehat{\boldsymbol{p}}\in C^H\cap c_k(y),\\ \frac{\mathscr{P}_L\widehat{\boldsymbol{p}}}{\|\mathscr{P}_L\widehat{\boldsymbol{p}}\|}\|\widehat{\boldsymbol{p}}\|,\ \widehat{\boldsymbol{p}}\notin C^H\cap c_k(y),\end{cases} \tag{4.2}$$

where $\mathscr{P}_L$ is the Euclidean projection operator onto the subspace L, with the latter defined as,

$$L:=\partial c_k(y)=\left\{\boldsymbol{s}\,|\,\left\langle\widehat{\boldsymbol{\ell}}_k(y),\boldsymbol{s}\right\rangle=0\right\}.$$

Proof. For simplicity, we will suppress $k,\widehat{\boldsymbol{p}}$, and y notationally in the proof, i.e., $\boldsymbol{c}:=\boldsymbol{c}_k(\widehat{\boldsymbol{p}};y)$, and $\widehat{\boldsymbol{\ell}}:=\widehat{\boldsymbol{\ell}}_k(y)$. Since $C^H\cap c_k(y)$, is non-empty and compact, there is at least one solution to (4.1).

Following similar computations to the proof to Lemma 3.1, it can be shown that

$$\boldsymbol{c} = \underset{\widehat{\boldsymbol{v}} \in C^H \cap c_k(y)}{\arg\max} \ \langle \widehat{\boldsymbol{v}}, \widehat{\boldsymbol{p}} \rangle, \tag{4.3}$$

is equivalent to (4.1). If $\widehat{\boldsymbol{p}} \in C^H \cap c_k(y)$, then $\boldsymbol{c} = \widehat{\boldsymbol{p}}$ is the unique solution to (4.3) by the Cauchy-Schwarz inequality, which verifies part of (4.2). Thus, the remainder of the proof assumes $\widehat{\boldsymbol{p}}$ is not in the feasible set. Let $\widehat{\boldsymbol{c}}$ be any solution to (4.3). Since $\widehat{\boldsymbol{c}}$ lies in $c_k(y)$ and since $\widehat{\boldsymbol{p}}$ lies in C^H but is not feasible, then we have

$$\left\langle \widehat{\boldsymbol{c}}, \widehat{\boldsymbol{\ell}} \right\rangle \leq 0, \qquad\qquad \left\langle \widehat{\boldsymbol{p}}, \widehat{\boldsymbol{\ell}} \right\rangle > 0.$$

By the above inequalities, any solution $\widehat{\boldsymbol{c}}$ to (4.3) satisfies,

$$\begin{aligned} \langle \widehat{\boldsymbol{c}}, \widehat{\boldsymbol{p}} \rangle &= \left\langle \mathscr{P}_L \widehat{\boldsymbol{c}}, \mathscr{P}_L \widehat{\boldsymbol{p}} \right\rangle + \left\langle (I - \mathscr{P}_L)\widehat{\boldsymbol{c}}, (I - \mathscr{P}_L)\widehat{\boldsymbol{p}} \right\rangle = \left\langle \mathscr{P}_L \widehat{\boldsymbol{c}}, \mathscr{P}_L \widehat{\boldsymbol{p}} \right\rangle + \left\langle \widehat{\boldsymbol{c}}, \widehat{\boldsymbol{\ell}} \right\rangle \left\langle \widehat{\boldsymbol{p}}, \widehat{\boldsymbol{\ell}} \right\rangle, \\ &\overset{\text{(i)}}{\leq} \left\langle \mathscr{P}_L \widehat{\boldsymbol{c}}, \mathscr{P}_L \widehat{\boldsymbol{p}} \right\rangle \overset{\text{(ii)}}{\leq} \| \mathscr{P}_L \widehat{\boldsymbol{c}} \| \, \| \mathscr{P}_L \widehat{\boldsymbol{p}} \| \overset{\text{(iii)}}{\leq} \| \widehat{\boldsymbol{c}} \| \, \| \mathscr{P}_L \widehat{\boldsymbol{p}} \| = \| \widehat{\boldsymbol{p}} \| \, \| \mathscr{P}_L \widehat{\boldsymbol{p}} \| \end{aligned}$$

The choice $\widehat{\boldsymbol{c}} = \boldsymbol{c}$ in (4.2) is the unique solution that achieves equality in (i), (ii), and (iii) above. To see this, first note that $\boldsymbol{c}$ is feasible since it lies in both C^H and $c_k(y)$, and is well-defined since $\widehat{\boldsymbol{p}}$ is not parallel to $\widehat{\boldsymbol{\ell}}_k(y)$ and hence $\mathscr{P}_L \widehat{\boldsymbol{p}} \neq 0$. Equality in (i) and (iii) can be established by noting that $\boldsymbol{c} \in L$, so that $\left\langle \boldsymbol{c}, \widehat{\boldsymbol{\ell}} \right\rangle = 0$ and $\mathscr{P}_L \boldsymbol{c} = \boldsymbol{c}$. Equality in (ii) is achieved if and only if $\mathscr{P}_L \boldsymbol{c} = \boldsymbol{c}$ has the same direction as $\mathscr{P}_L \widehat{\boldsymbol{p}}$, which the choice (4.2) satisfies. This also shows that $\boldsymbol{c}$ is the only vector that achieves this equality, and hence (4.3) (equivalently, (4.1)) has a unique solution (4.2). □

REMARK 4.1. The solution (4.2) implies that when $\widehat{\boldsymbol{p}}$ is not feasible and is not parallel to $\widehat{\boldsymbol{\ell}}_k(y)$, the solution to (4.1) can be computed by first computing a Euclidean projection onto the hyperplane L, and the simply rescaling this projection to have norm $\|\widehat{\boldsymbol{p}}\|$. We exploit this fact in algorithms.

4.2 *A greedy approach*

We first introduce a new notation for the spherical projection

$$\boldsymbol{c}_k(\widehat{\boldsymbol{p}}; y) := \mathscr{P}^s_{c_k(y)} \widehat{\boldsymbol{p}}, \tag{4.4}$$

which by Theorem 4.1 is well-defined for every $\widehat{\boldsymbol{p}}$ that is not a multiple of $\widehat{\boldsymbol{\ell}}_k(y)$.

A greedy procedure, in the spirit of the greedy algorithm of [35], iteratively updates $\widehat{\boldsymbol{p}}$ by repeatedly identifying most-violated constraints. Defining $\widehat{\boldsymbol{p}}^0 = \widehat{\boldsymbol{p}}$, and using $\widehat{\boldsymbol{p}}^j$ to denote the iterate at step j, we seek to compute,

$$\widehat{\boldsymbol{p}}^{j+1} = \mathscr{P}^s_{c_{k^*}(y^*)} \widehat{\boldsymbol{p}}^j, \qquad\qquad (k^*, y^*) := \underset{k \in [K], y \in \omega_k}{\arg\max} \ d_r(\widehat{\boldsymbol{p}}^j, c_k(y)), \tag{4.5}$$

for $j \geq 1$. Lemma 4.1 first allows us to conclude that the set of (k, y) such that $d_r(\widehat{\boldsymbol{p}}, c_k(y) \cap C^H) > 0$ is equal to the set of (k, y) such that $\widehat{\boldsymbol{p}} \notin c_k(y)$.

LEMMA 4.1. Let $\widehat{\boldsymbol{p}}$ be the solution to the current iteration, then

$$d_r(\widehat{\boldsymbol{p}}, c_k(y) \cap C^H) > 0 \Leftrightarrow \widehat{\boldsymbol{p}} \notin c_k(y), \tag{4.6}$$

where $r = \|\widehat{\boldsymbol{p}}\|$.

Proof. Let $\mathrm{dist}(\cdot,\cdot)$ be the Euclidean distance function, then

$$\mathrm{dist}(\widehat{\boldsymbol{p}}, c_k(y)) = \min_{\boldsymbol{s}\in c_k(y)} \|\widehat{\boldsymbol{p}} - \boldsymbol{s}\|_2 = \|\widehat{\boldsymbol{p}} - \mathscr{P}_L\widehat{\boldsymbol{p}}\|_2 = \|\widehat{\boldsymbol{p}}\| \sin\theta_k(y), \tag{4.7}$$

where $L := \partial c_k(y)$ is the boundary of the half-space $c_k(y)$, and $\theta_k(y) = \arccos\langle \widehat{\boldsymbol{p}}/r, \boldsymbol{c}_k(\widehat{\boldsymbol{p}}; y)/r\rangle$ is the angle between $\widehat{\boldsymbol{p}}$ and its spherical projection $\boldsymbol{c}_k(\widehat{\boldsymbol{p}}; y)$ onto the half-space $c_k(y)$. The last equality in (4.7) is true since $\mathscr{P}_L\widehat{\boldsymbol{p}}$ and $\boldsymbol{c}_k(\widehat{\boldsymbol{p}}; y)$ have the same direction (Theorem 4.1). The angle $\theta_k(y)$ is the angle between the vector $\widehat{\boldsymbol{p}}$ and the plane L, thus $\theta_k(y) \in [0, \pi/2]$.

For any (k, y), a direct calculation using the Pythagorean theorem and the definition of the intrinsic distance $d_r(\cdot,\cdot)$ yields

$$\begin{aligned}
\widehat{\boldsymbol{p}} \notin c_k(y) &\Leftrightarrow \mathrm{dist}(\widehat{\boldsymbol{p}}, c_k(y)) > 0,\\
&\Leftrightarrow \|\widehat{\boldsymbol{p}}\| \sin\theta_k(y) > 0,\\
&\Leftrightarrow \theta_k(y) > 0,\\
&\Leftrightarrow d_r(\widehat{\boldsymbol{p}}, c_k(y) \cap C^H) > 0,
\end{aligned}$$

□

The proof to Lemma 4.1 motivates the following relation

$$\begin{aligned}
(k^*, y^*) &= \underset{k\in[K], y\in\omega_k}{\arg\max}\ d_r(\widehat{\boldsymbol{p}}, c_k(y))\\
&= \underset{k\in[K], y\in\omega_k}{\arg\max}\ (\mathrm{dist}(\widehat{\boldsymbol{p}}, c_k(y)))\\
&= \underset{y\in\omega_k, k\in[K]}{\arg\min}\ \mathrm{sdist}(\widehat{\boldsymbol{p}}, c_k(y)),
\end{aligned} \tag{4.8}$$

where the Euclidean signed distance between $\widehat{\boldsymbol{p}}$ and the Euclidean half-space $c_k(y)$ can be computed by (see [35])

$$\mathrm{sdist}(\widehat{\boldsymbol{p}}, c_k(y)) = -\langle \widehat{\boldsymbol{\ell}}_k(y), \widehat{\boldsymbol{p}}\rangle. \tag{4.9}$$

Equations (4.8)-(4.9) imply that, to determine the parameters for the geodesically farthest hemisphere, we only need to determine the parameters for the Euclidean-farthest hyperplane, which is a much easier computational task.

Algorithm 1 summarizes the above iterative procedure of computing the solution to (3.8) through greedy spherical projection.

Algorithm 1: Iterative greedy spherical projection to compute the solution to (4.1).

Require: the matrix $\boldsymbol{A}$ and the observational vector $\boldsymbol{b}$, tolerance parameter $\delta \geq 0$
Require: constraints $\{(\widehat{\boldsymbol{\ell}}_k(y), \omega_k)\}_{k\in[K]}$
Ensure: greedy solution $\widehat{\boldsymbol{p}}$.
1: Compute the unconstrained solution $\widehat{\boldsymbol{p}}$, e.g. via solving (2.3).
2: **while** $\text{sdist}(\widehat{\boldsymbol{p}}, c_k(y)) \leq -\delta$ for some $k \in [K],\ y \in \omega_k$ **do**
3: compute (y^*, k^*) via (4.8)
4: update $\widehat{\boldsymbol{p}}$ via (4.5)
5: **end while**

4.3 *An averaging approach*

The greedy procedure above can lead to oscillatory behavior of the iteration trajectory. To mitigate this behavior, we introduce an averaging projection approach to suppress potential oscillatory behavior of iterates in the previous greedy approach. The notion of an average position of a collection of points on the sphere is defined by the *Karcher mean*, which is a natural extension of the Euclidean weighted average.

DEFINITION 4.1. (Karcher mean) Let $S \subset \mathbb{R}^N$ be the sphere centered at the origin with radius r. Let $\boldsymbol{q}_1, \cdots, \boldsymbol{q}_J$ be J points lying on S associated with nonnegative convex weights $w_1, \cdots, w_J \in [0, 1]$. The *Karcher mean* is given by the solution to

$$\boldsymbol{q} = \arg\min_{\boldsymbol{x}\in S} \left(\frac{1}{2} \sum_{i=1}^{n} w_i \cdot d_r^2(\boldsymbol{x}, \boldsymbol{q}_i) \right) := \arg\min_{\boldsymbol{x}\in S} f(\boldsymbol{x}). \tag{4.10}$$

The Karcher mean as defined above is unique under mild assumptions.

THEOREM 4.2. ([7], Theorem 1) With S the radius-r origin-centered sphere in $\mathbb{R}^N$, suppose that given points $\boldsymbol{q}_1, \cdots, \boldsymbol{q}_J$ all lie in a closed hemisphere $\mathscr{H} \subset S$, with at least one point $\boldsymbol{q}_j$ in the interior of $\mathscr{H}$ with $w_j > 0$. Then $f(\boldsymbol{x})$ defined in (4.10) has a single critical point $\boldsymbol{q}$ in the interior of $\mathscr{H}$, and this point $\boldsymbol{q}$ is the global minimum of f, hence the unique Karcher mean.

The definition of the Karcher mean in (4.10) can be extended to a collection of infinitely many points by integration, and our averaged projection algorithm is based on this generalized Karcher mean. Let $\widehat{\boldsymbol{p}}^0 = \widehat{\boldsymbol{p}}$ be the first iterate in the algorithm. To compute the next iterate, the averaging algorithm first identifies all the parameters (y, k) for which the associated linear constraints are violated,

$$\omega_{k^-}^j = \{y \in \omega_k | \text{sdist}(\widehat{\boldsymbol{p}}^j, c_k(y)) < 0\}. \tag{4.11}$$

Instead of projecting onto the most violated constraint (as in the previous section) we seek a point that minimizes the Karcher mean objective over all violated constraints. With $r = \|\widehat{\boldsymbol{p}}\|$, the averaged position

$\widehat{\boldsymbol{p}}^{j+1}$ at the next iteration is given by

$$\widehat{\boldsymbol{p}}^{j+1} = \underset{\boldsymbol{x}\in C^H}{\arg\min} \left(\frac{1}{2} \sum_{k\in\mathscr{D}_j^-} \int_{\omega_{k-}^j} d_r^2(\boldsymbol{x}, \boldsymbol{c}_k(\widehat{\boldsymbol{p}}^j; y)) w_k(\widehat{\boldsymbol{p}}^j, y) dy \right), \tag{4.12}$$

where $\mathscr{D}_j^- = \{k \in [K] | \omega_{k-}^j \neq \varnothing\}$ is the set of the indexes where the corresponding constraints are violated at jth iteration, and the weight,

$$w_k(\widehat{\boldsymbol{p}}^j; y) := \left(\frac{d_r^2(\widehat{\boldsymbol{p}}^j, \boldsymbol{c}_k(\widehat{\boldsymbol{p}}^j; y))}{\sum_{\ell\in\mathscr{D}_j^-} \int_{\omega_{\ell-}^j} d_r^2(\widehat{\boldsymbol{p}}^j, \boldsymbol{c}_\ell(\widehat{\boldsymbol{p}}^j; z)) dz} \right) \tag{4.13}$$

is introduced for each spherical projection $\boldsymbol{c}_k(y)$ in order to prioritize updates that mitigate the impact of the more violated constrained sets. In practice, we approximate the integral (4.12) via quadrature with positive weights, i.e.,

$$\widehat{\boldsymbol{p}}^{j+1} = \underset{\boldsymbol{x}\in C^H}{\arg\min} \left(\frac{1}{2} \sum_{k\in\mathscr{D}_j^-} \sum_{q=1}^{Q_k} w_{k,q} d_r^2(\boldsymbol{x}, \boldsymbol{c}_k(\widehat{\boldsymbol{p}}^j; y_{k,q})) \right), \tag{4.14}$$

where Q_k is the number of the quadrature points associated with the kth constraint, and $\{w_{k,q}\}_{k\in[K],q\in[Q_k]}$ are the product of the (positive) quadrature weight with an approximations to the weight (4.13) at the quadrature points $\{y_{k,q}\}_{k\in[K],q\in[Q_k]}$.

All the discussion above is provided in the context of assuming that the hemisphere condition in Theorem 4.2 holds. The following lemma shows that all the candidate spherical projection $\boldsymbol{c}_k(\widehat{\boldsymbol{p}}^j; y_{k,q})$ indeed lie on the same hemisphere.

LEMMA 4.2. The spherical projections $\boldsymbol{c}_k(\widehat{\boldsymbol{p}}; y)$ defined by (4.2) with $\widehat{\boldsymbol{\ell}} = \widehat{\boldsymbol{\ell}}_k(y), k \in [K]$ are always on the hemisphere

$$\mathscr{H} = \{\boldsymbol{x} \Big| \|\boldsymbol{x}\| = \|\widehat{\boldsymbol{p}}\|, \langle \widehat{\boldsymbol{p}}, \boldsymbol{x} \rangle \geq 0\}, \tag{4.15}$$

for any y and k.

Proof. Since $\|\boldsymbol{c}_k(\widehat{\boldsymbol{p}}; y)\| = \|\widehat{\boldsymbol{p}}\|$, we only need to verify the second condition in (4.15). Using (4.2), a direct computation yields,

$$\left\langle \widehat{\boldsymbol{p}}, \boldsymbol{c}_k(\widehat{\boldsymbol{p}}; y) \right\rangle = \|\widehat{\boldsymbol{p}}\| \|\mathscr{P}_{c_k(y)} \widehat{\boldsymbol{p}}\| \geq 0.$$

□

All the above is almost sufficient to guarantee that the algorithm described by (4.14) has a unique solution. The last obstacle we have yet to overcome is to ensure that the points $\boldsymbol{c}_k(\widehat{\boldsymbol{p}}^j; y_{k,q})$ are uniquely defined. To ensure this, we make the following assumption.

ASSUMPTION 4.1. $\widehat{\boldsymbol{\ell}}_k(y)$ is not parallel to $\widehat{\boldsymbol{p}}^j$ for all (k, y) pairs for $y \in \omega_k^-$.

Assumption 4.1 is necessary to ensure unique existence of the spherical projections $\boldsymbol{c}_k(\widehat{\boldsymbol{p}}^j; y_{k,q})$. Although we cannot yet theoretically justify of Assumption 4.1, in all of our numerical experiments, Assumption 4.1 holds. Under this assumption, we can prove uniqueness of the update (4.14).

PROPOSITION 4.1. Under Assumption 4.1, the solution $\widehat{\boldsymbol{p}}^{j+1}$ to (4.14) is unique.

Proof. Under Assumption 4.1 and Lemma 4.2, the candidate points $\boldsymbol{c}_k(\widehat{\boldsymbol{p}}^j; y_{k,q})$ are all in the interior of $\mathscr{H}$. Therefore, from Theorem 4.2, the solution $\widehat{\boldsymbol{p}}^{j+1}$ to (4.14) is unique and furthermore lies in the interior of $\mathscr{H}$. □

Algorithms that compute the average spherical projection by solving the optimization problem (4.14) can be adapted from [7, Algorithms A1 or A2) or [24, Equation 10).

4.4 *A hybrid approach*

We propose a final algorithm: the averaging algorithm of the previous section results in less oscillatory iterate trajectories, but moves relatively slowly. The algorithm in this section combines the ideas of the greedy and averaging approach. First we denote the greedy update (4.5) at the jth iteration by $\widehat{\boldsymbol{p}}_g^{j+1}$ and the average update (4.12) by $\widehat{\boldsymbol{p}}_a^{j+1}$. The hybrid update we propose moves in the direction of the averaged update $\widehat{\boldsymbol{p}}_a^{j+1}$, but with a distance defined by the greedy update $\widehat{\boldsymbol{p}}_g^{j+1}$. Specifically, at the jth iteration,

i. Compute the geodesic projection $\widehat{\boldsymbol{p}}_g^{j+1}$ and the average projection $\widehat{\boldsymbol{p}}_a^{j+1}$ via (4.5) and (4.14), respectively.

ii. If $d_r(\widehat{\boldsymbol{p}}^j, \widehat{\boldsymbol{p}}_g^{j+1})/r < 10^{-6}$, i.e., the most violated constraint is very close to the current update, we simply perform the greedy update, setting $\widehat{\boldsymbol{p}}^{j+1} = \widehat{\boldsymbol{p}}_g^{j+1}$.

iii. Otherwise, we compute $\widehat{\boldsymbol{p}}^{j+1}$ by moving $\widehat{\boldsymbol{p}}^j$ along the unique geodesic from $\widehat{\boldsymbol{p}}^j$ to $\widehat{\boldsymbol{p}}_a^{j+1}$ by a distance given by the intrinsic distance between $\widehat{\boldsymbol{p}}^j$ and $\widehat{\boldsymbol{p}}_g^{j+1}$. Let $\widetilde{\boldsymbol{p}}^j, \widetilde{\boldsymbol{p}}_a^j$, and $\widetilde{\boldsymbol{p}}_g^j$ be the unit-norm versions of $\widehat{\boldsymbol{p}}^j, \widehat{\boldsymbol{p}}_a^{j+1}$, and $\widehat{\boldsymbol{p}}_g^{j+1}$, respectively. Then the update we propose is

$$\begin{aligned} \widetilde{\boldsymbol{p}}^{j+1} &= \exp_{\widetilde{\boldsymbol{p}}^j}\left(d(\widetilde{\boldsymbol{p}}^j, \widetilde{\boldsymbol{p}}_g^{j+1})\boldsymbol{v}\right), \\ \widehat{\boldsymbol{p}}^{j+1} &= \|\widehat{\boldsymbol{p}}^j\|\widetilde{\boldsymbol{p}}^{j+1}, \end{aligned} \tag{4.16}$$

where $\boldsymbol{v}$ is the unit-speed velocity at the base point $\widetilde{\boldsymbol{p}}^j$ of the geodesic segment leading to $\widetilde{\boldsymbol{p}}_a^{j+1}$, i.e., $\boldsymbol{v} = \gamma'_{\widetilde{\boldsymbol{p}}^j \widetilde{\boldsymbol{p}}_a^{j+1}}(0)$.

We use the greedy update in step 2 above since in this case we are relatively close to the solution, and so typically the greedy procedure converges very quickly.

5. Numerical experiments

Throughout this section, we take $M = N$ observations, and the observation functionals $\{\phi_n\}_{n\in[N]}$ are chosen to be the projection functionals $\phi_n(\cdot) := \langle \cdot, v_n \rangle$ onto the given subspace V. We denote the

unknown function by u, the H-best projection onto V by v, the norm-constrained solution (the solution to (2.14)) by v_{NC}, and the linearly constrained solution by v_{LC}. I.e., v_{LC} is the solution to (2.14) but with only the linear inequality constraints,

$$v_{\mathrm{LC}} := \sum_{n\in[N]} w_n v_n, \qquad \boldsymbol{w} := \operatorname*{arg\,min}_{\widehat{\boldsymbol{v}}\in C^0} \frac{1}{2}\left\|\widehat{\boldsymbol{v}} - \widehat{\boldsymbol{p}}\right\|^2.$$

For other choices of observation functionals, e.g., pointwise observations (collocation-based approximations), our theory and algorithms can be generalized naturally.

For our univariate examples, we consider the Sobolev spaces on a general interval $[a, b]$ as our Hilbert spaces H,

$$H^q([a,b]) := \left\{u : [a,b] \to \mathbb{R} \,\middle|\, \|u\|_{H^q}^2 < \infty\right\}, \qquad \|u\|_{H^q}^2 := \sum_{j=0}^{q} \int_a^b \left[u^{(j)}(x)\right] dx, \tag{5.1}$$

and choose the subspace V according to the choice of the pair (a, b),

$$\begin{aligned} &\text{if } (a,b) = (-1,1), && \text{then } V = V^{\mathrm{poly}} := \operatorname{span}\left\{\{x^n\}_{n=0}^{N-1}\right\}, \\ &\text{if } (a,b) = (0,\pi), && \text{then } V = V^{\cos} := \operatorname{span}\left\{\{\cos nx\}_{n=0}^{N-1}\right\}. \end{aligned} \tag{5.2}$$

We will test our algorithms for $H^0(= L^2)$, H^1, and H^2 using the linear constraint sets,

- (Positivity) $U_0 := \{u \in H | u(x) \geq 0 \ \forall x \in [a,b]\}$
- (Monotonicity) $U_1 := \{u \in H | u'(x) \geq 0 \ \forall x \in [a,b]\}$
- (Convexity) $U_2 := \{u \in H | u''(x) \geq 0 \ \forall x \in [a,b]\}$

Although our theoretical result in Theorem 3.1 does not guarantee the uniqueness of the solution when a boundedness constraint imposed, we still test our algorithms with imposing the constraint,

- (Boundedness) $G_0 := \{u \in H | u(x) \leq 1 \ \forall x \in [a,b]\}$,

for some of our tests. When a boundedness constraint is imposed, the hyperplane is an affine plane. In this case, we first project the current iteration to the affine hyperplane, then rescale the point with respect to the vertex $\boldsymbol{r}_0$ of the cone (the projection of the origin onto the affine plane) to the sphere, i.e., the projection (4.4) is replaced by

$$\boldsymbol{c} = \mathscr{P}_H^s \widehat{\boldsymbol{p}} := \boldsymbol{r}_0 + \frac{\sqrt{\|\widehat{\boldsymbol{p}}\|^2 - \|\boldsymbol{r}_0\|^2}}{\sqrt{\|\mathscr{P}_H \widehat{\boldsymbol{p}}\|^2 - \|\boldsymbol{r}_0\|^2}} (\mathscr{P}_H \widehat{\boldsymbol{p}} - \boldsymbol{r}_0).$$

We will also introduce a metric to measure the change between the constrained solutions and the unconstrained solution:

$$\eta_* = \frac{\|v - v_*\|_H}{\|v - u\|_H}, \tag{5.3}$$

where asterisk "$*$" on the subscript of v can be either "LC" (the linearly constrained approximation using the dissipative formulation in [35]) or 'NC' (the non-dissipative procedure in this article). Since $v - u$ is H-orthogonal to V, the Pythagorean theorem implies,

$$\|v_* - u\|_H = \sqrt{1+\eta^2}\|v-u\|_H^2.$$

The quantity $\sqrt{1+\eta^2}$ can therefore be used to measure the error in a constrained solution relative to error in the unconstrained solution (which in this case is the H-best approximation from V). Values of η that are $O(1)$ indicate that the error committed by the constrained solution is comparable to that of the unconstrained solution. It is also interesting to measure the difference between the linearly constrained solution and the norm-constrained solution $\|v_{\mathrm{LC}} - v_{\mathrm{NC}}\|$. In all of our experiments, the norm-constrained solution v_{NC} differs only slightly from the linearly-constrained solution v_{LC}.

Algorithm 1 is the greedy algorithm, but it is also the template for the average algorithm. To apply the average algorithm, one only needs to replace the update of $\widehat{\boldsymbol{p}}$ with (4.14). In line 2 of Algorithm 1, δ is set to be 10^{-10}. In addition, we restrict the maximum number of iterations to be $10,000$ to avoid infinite loops.

Time complexity: A *single* step of the greedy algorithm requires determining (y^*, k^*) associated with the farthest hyperplane (Algorithm 1), whose complexity is basis-dependent. For univariate polynomial approximation, the complexity is $O(N^2)$, where N is the dimension of the subspace V [35]. A *single* step of the average algorithm requires computing the integrals and the Karcher mean ((4.14)). The cost of computing the integrals consists of two parts,

- determining the negative region $\mathscr{D}_j^-$: this part requires finding the zeros of the unconstrained approximation. For univariate polynomial approximation, the total complexity is $O(KN^2)$.
- applying the quadrature rule: let $Q = \max_{k\in[K]} Q_k$. Since the number of negative regions is at most $O(N)$ and computing $d_r(\boldsymbol{x}, \boldsymbol{c}_k(\widehat{\boldsymbol{p}}^j; y_{k,q}))$ is of $O(N)$, the total cost is $O(QN^2)$.

For the computation of Karcher mean, we use the algorithm of locally linear rate of convergence ([7], Algorithm A1). Let μ be its rate of convergence and ϵ be a desired tolerance level, the complexity is $O(\log_\mu \epsilon)$ and is independent of the choice of the basis functions.

The time complexity for finding a feasible solution to the original optimization problem remains unknown since we do not know how many steps the algorithms take to achieved a desired tolerance level. The algorithms converge quickly in some examples but slowly on others (e.g., Table 2). In our numerical test, the greedy algorithm performs better than the average algorithm.

5.1 *Performance comparison of algorithms*

In this section, we present the comparison of solutions from the linearly-constrained optimization in [35] and the norm-constrained optimization proposed in our work. We consider the case $(a, b) = [-1, 1]$ and degree-$(N-1)$ polynomial approximations with $V = V^{\mathrm{poly}}$. The test functions are chosen as a step function and its antiderivatives:

$$u_{j+1}(x) := c_{j+1}\int_{-1}^{x} u_j(t)dt, \qquad u_0(x) = \begin{cases} 0, & x \le 0, \\ 1, & x > 0, \end{cases} \tag{5.4}$$

TABLE 1 *Comparison of number of iterations (I) and relative errors (η) on the test function $u = u_2$ with positivity-constraint imposed for different values of N and different algorithms. The ambient Hilbert space is $H = L^2([-1, 1])$.*

	$N = 6$		$N = 31$	
	I	η	I	η
Greedy	17	1.1479	23	0.9859
Average	87	1.1483	220	0.9856
Hybrid	15	1.1494	22	0.9885

TABLE 2 *Comparison of the minimum values of v_{NC}, v'_{NC}, and v''_{NC} using average approach at 10,000 iterations. 'Converge' indicates the corresponding procedure finds a feasible solution before the maximum number of iterations is reached.*

	$N = 6$			$N = 31$		
	v_{NC}	v'_{NC}	v''_{NC}	v_{NC}	v'_{NC}	v''_{NC}
H^0	–1.05e-6	9.35e-5	–1.86e-4	–2.18e-7	–2.35e-3	–3.06e-2
H^1	converge	converge	converge	–1.33e-8	–3.84e-7	–1.51e-3
H^2	converge	converge	converge	converge	converge	converge

where c_{j+1} is a normalized constant that ensures the $u_{j+1}(1) = 1$. We provide a summary of the performance of our three proposed approaches for norm-preserving optimization Table 1. We observe that, compared to greedy approach, there is a slight decrease in the number of iterations. Both greedy approach and the hybrid approach are much faster than the average approach. The relative error of the three proposed approaches are comparable.

5.2 *Polynomial space approximation example*

In this section, we continue to consider the approximation using $(a, b) = (-1, 1)$ and $V = V^{\text{poly}}$. In our first experiment, we test the capability of our algorithms for approximating the step function $u_0(x)$ in (5.4) and the similarity between v_{LC} and v_{NC}. We compute the approximation for $N = 6$ and $N = 31$ Figure 2 and consider three choices of linear constraint sets E^0 introduced in (2.6):

(i) (positivity) $E^0 = U_0$,

(ii) (positivity and monotonicity) $E^0 = U_0 \bigcap U_1$, and

(iii) (positivity, monotonicity, boundedness) $E^0 = U_0 \bigcap U_1 \bigcap G_0$.

We note that, for (i) and (ii), the norm-constrained solutions are simply slight adjustment of the linearly constrained solution, both visually and quantitatively. The discrepancy is more obvious in case (iii). Increasing the order in the polynomial approximation further decreases the discrepancy between v_{LC} and v_{NC}. We also note that, the Gibbs'-type oscillations presented in the left column of Figure 2 can be alleviated by enforcing the monotonicity and the boundedness constraint. All computed η_{NC} values are order 1, which shows that our norm-preserving approximations are comparable to the H-best approximation.

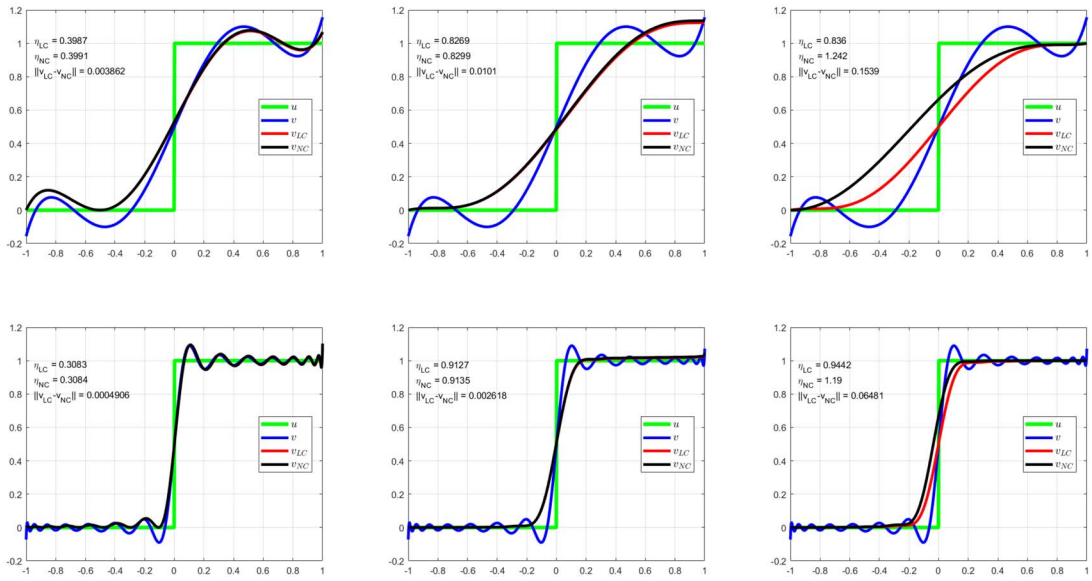

FIG. 2. Greedy algorithm results: comparison of different methods: degree 5 polynomial positivity-preserving approximation to the step function for different constraints and different polynomial spaces. Left: constraint U_0. Middle: constraint $U_0 \bigcap U_1$. Right: $U_0 \bigcap U_1 \bigcap G_0$. Top: $N = \dim V = 6$. Bottom: $N = \dim V = 31$.

In the second experiment of this section, we investigate how the choice of ambient Hilbert space H affects the accuracy of the approximation. We approximate the function $u_2(x)$ with linear constraint set $E^0 = U_0 \bigcap U_1 \bigcap U_2$ for $N = 6$ and $N = 31$ on different Hilbert spaces $H = H^0, H^1$, and H^2. We observe relatively large values of both η_{NC} and η_{LC}, but increasing the regularity of the Hilbert space and/or increasing the order of the polynomial can reduce these relative errors. Similar to the previous test, the discrepancy between v_{LC} and v_{NC} decreases as the order of polynomial order increases. It increases as the complexity of the linear constraint set increases. Nevertheless, both approximations are qualitatively good for $N = 31$. The results are shown in Figure 2.

Quantitatively, we find that the minimum values of v_{NC}, v'_{NC}, and v''_{NC} converge slowly for some examples with less regular Hilbert space $H = H^0, H^1$. Among our three proposed approach, the greedy approach and the hybrid approach performs slightly better than the average approach. For different choices of the ambient Hilbert spaces, we report in Table 2 the minimum values of v_{NC}, v'_{NC} and v''_{NC} using average approach at 10,000 iterations. The 'converge' in Table 2 indicates that the procedures achieve the desired tolerance levels. We note that, by increasing the regularity of the ambient Hilbert space, our procedures can identify a feasible solution much faster. We emphasize that the simplicity of this example belies the complexity and difficulty of the geometry of the problem, which is evidenced by algorithms requiring more iterations to complete. In the remaining examples of this paper, all the algorithms identify an element of the feasible set (to within precision tolerances).

5.3 *Convergence rate*

In this subsection, we compare the rates of convergence between the unconstrained solution and the norm-constrained solution. We consider $u = u_0(x)$ and $u = u_2(x)$ using $V = V^{\mathrm{poly}}$. The ambient Hilbert

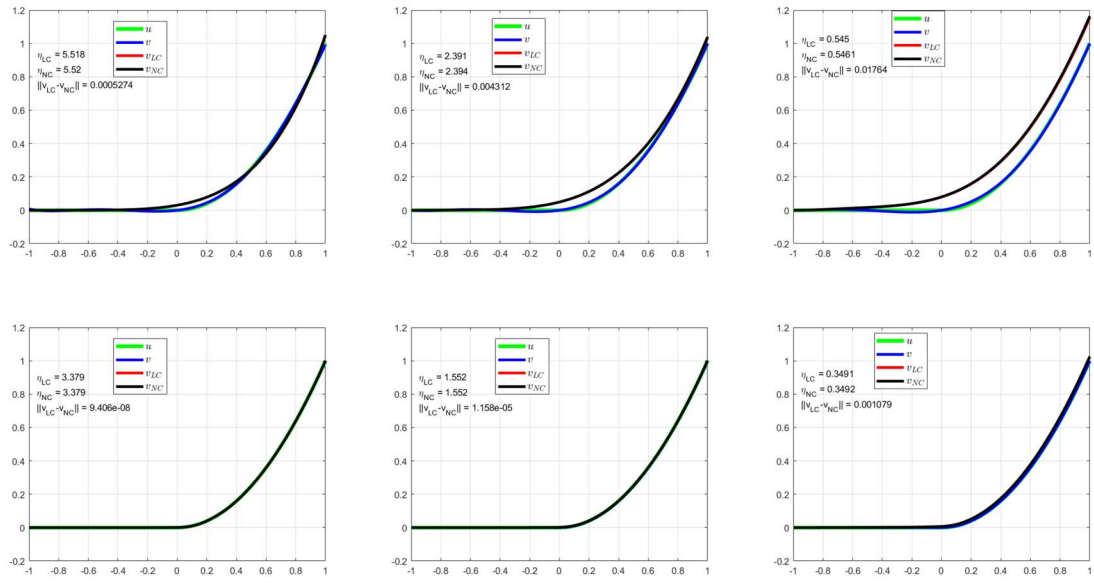

FIG. 3. Comparison of different approximations to $u_2(x)$ for different ambient Hilbert spaces and different polynomial spaces. The red curves (v_{LC}) are covered by the black curves (v_{NC}). The constraint is $U_0 \bigcap U_1 \bigcap U_2$. Left: $H = H^0$. Middle: $H = H^1$. Right: $H = H^2$. Top: $N = \dim V = 6$. Bottom: $N = \dim V = 31$.

space is $H = L^2([-1, 1])$. We compute the rate of convergence on the constrained sets U_0, $U_0 \bigcap U_1$, and $U_0 \bigcap U_1 \bigcap G_0$. We observe from Figure 4 that our norm-constrained solutions have a similar rate of convergence to the unconstrained ($H = L^2$-optimal) solution u (even when a boundedness constraint is imposed).

5.4 *M-shape function using cosine basis*

In this section, we will choose $V = V^{\cos}$ for approximating an M-shape function defined on $[0, \pi]$,

$$u(x) = \begin{cases} -\left(x - \frac{\pi}{8}\right)\left(x - \frac{\pi}{2}\right) & \frac{\pi}{8} \le x < \frac{\pi}{2}, \\ -\left(x - \frac{\pi}{2}\right)\left(x - \frac{7\pi}{8}\right) & \frac{\pi}{2} \le x < \frac{7\pi}{8}, \\ 0 & \text{otherwise}, \end{cases} \tag{5.5}$$

with positivity constraint U_0 imposed. For a cosine polynomial, the difficult part for applying our algorithm is to determine the y-parameter corresponding to the most violated constraint (or the negative y-region), which requires to find the zeros of a trigonometry polynomials. Fortunately, this difficulty can be resolved by taking advantage of the Chebyshev polynomials. The results are shown in Figure 5.

5.5 *Two-dimensional cylinder indicator function*

In our last example, we consider the approximation to a cylinder

$$u(x, y) = \begin{cases} 1 & \text{if} \sqrt{(x - 0.5)^2 + (y - 0.5)^2} < 0.5, \\ 0 & \text{otherwise}. \end{cases} \tag{5.6}$$

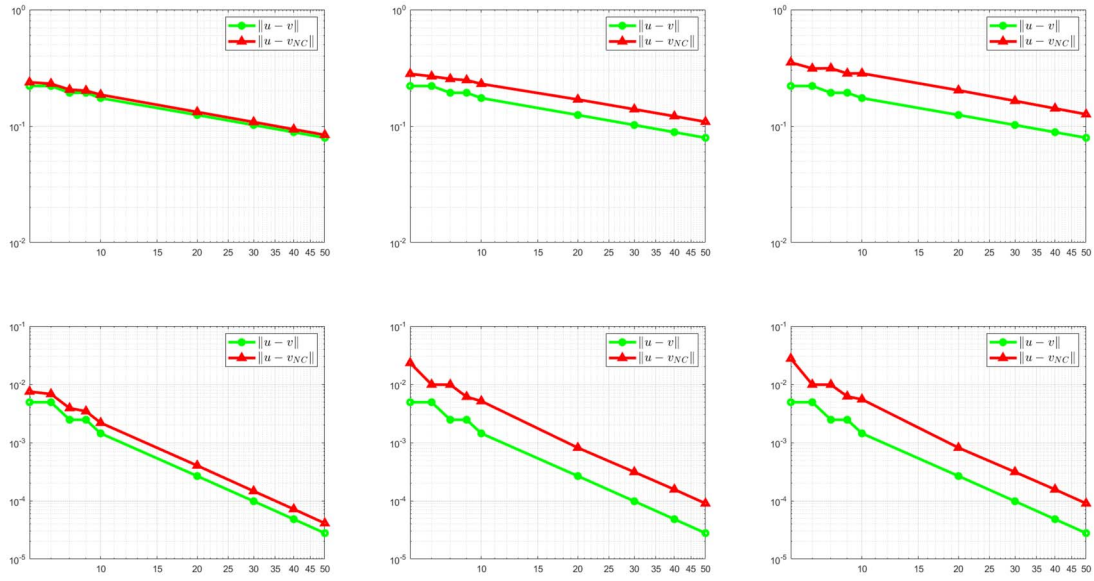

FIG. 4. Rate of convergence. Approximations to $u = u_0(x)$ (top) and $u = u_2(x)$ (bottom) with U_0 (left), $U_0 \bigcap U_1$ (middle), and $U_0 \bigcap U_1 \bigcap G_0$ (right) imposed. The x-axis indicates the dimension of the polynomial space $V = V^{\text{poly}}$. The ambient Hilbert space is $H = L^2([-1, 1])$.

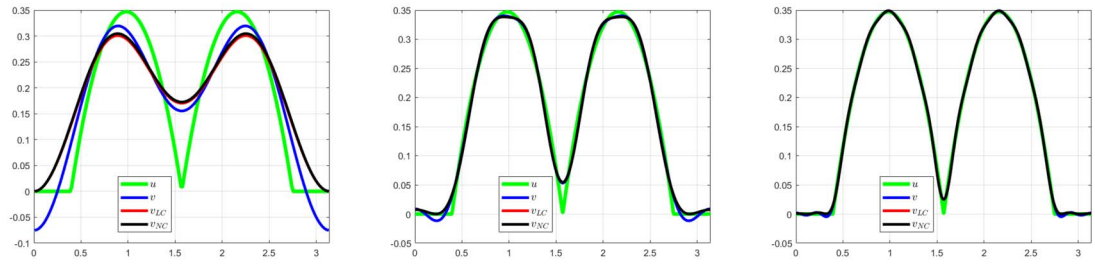

FIG. 5. Comparison of the approximations to (5.5) with different N. Constraint: U_0, positivity-preserving. From left to right: $N = 6, 16, 31$. The red curve is covered by the black curve.

The computational domain is $[-1, 1] \times [-1, 1]$, and the polynomial space is the tensor product space $V^{\text{poly}} \otimes V^{\text{poly}}$, where N is chosen to be 15. The positivity constraint U_0 is imposed. The computation requires to find the global minimum of a two-dimensional nonconvex function (4.9) (see also (2.10)–(2.11)). We use MATLAB's optimization function `fmincon`, using the sequential quadratic programming option, and approximate the global minimum by solving the optimization with several randomly initialized starting points. The constraints we set for `fmincon` are the boundaries for the computational domain.

The results are shown in Figure 6. The numerical results show that our norm-constrained approximation can preserve both the positivity and the norm by 'correcting' the linearly constrained solution. The function is entirely non-negative for both the linearly constrained solution (bottom middle, Figure 6) and the norm-constrained solution (bottom right, Figure 6).

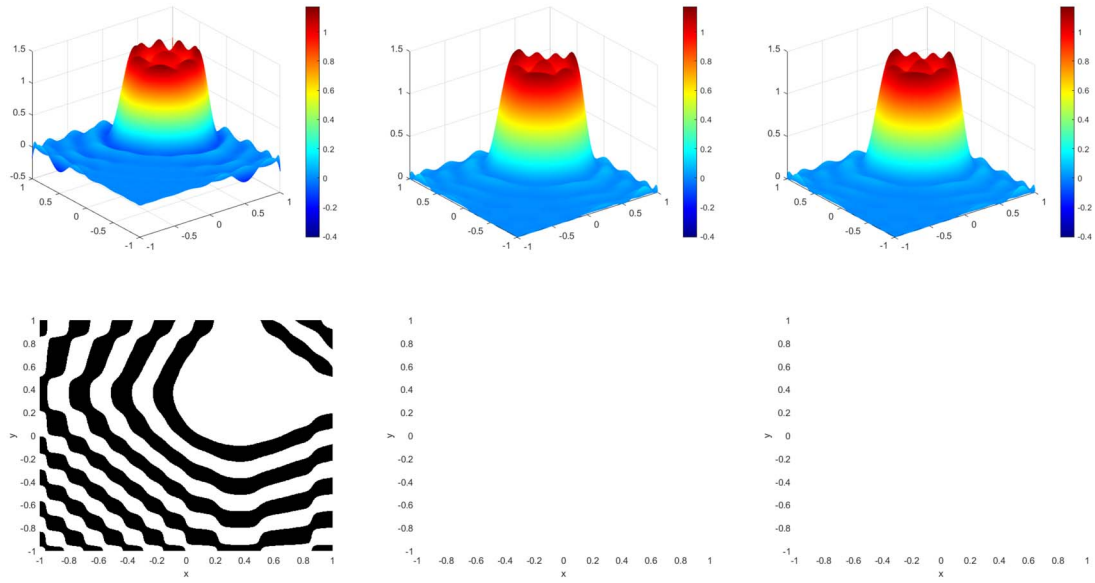

FIG. 6. Comparison of different approximations to (5.6), greedy procedure. Constraint: U_0, positivity-preserving. Top: mesh plot. Bottom: negative region indicator function, where the black region represent the region where the approximation is negative. Left: unconstrained solution u. Middle: linearly constrained solution v_{LC}. Right: Norm-constrained solution v_{NC}. $\eta_{LC} = 0.1229, \eta_{NC} = 0.1230, \|v_{LC} - v_{NC}\| = 0.0030$.

Data availability statement

No new data were generated or analyzed in support of this research.

Funding

D. Dai and A. Narayan are supported by NSF DMS-1848508 and AFOSR FA9550-20-1-0338.

Conflict of interest

There is no conflict of interest.

REFERENCES

1. ALLEN, L. & KIRBY, R. C. (2021) Bounds-constrained polynomial approximation using the Bernstein basis. arXiv preprint arXiv:2104.11819.
2. BEATSON, R. (1982) Restricted range approximation by splines and variational inequalities. *SIAM J. Numer. Anal.*, **19**(2), 372–380.
3. BEATSON, R. K. (1978) The degree of monotone approximation. *Pacific J. Math.*, **74**(1), 5–14.
4. BERZINS, M. (2007) Adaptive polynomial interpolation on evenly spaced meshes. *SIAM Rev.*, **49**(4), 604–627.
5. BESSE, C., DESCOMBES, S., DUJARDIN, G. & LACROIX-VIOLET, I. (2021) Energy-preserving methods for nonlinear Schrödinger equations. *IMA J. Numer. Anal.*, **41**(1), 618–653.
6. BOYD, S. P. & VANDENBERGHE, L. (2004) *Convex Optimization*. Cambridge University Press.
7. BUSS, S. R. & FILLMORE, J. P. (2001) Spherical averages and applications to spherical splines and interpolation. *ACM Transactions on Graphics*, **20**(2), 95–126.

8. Campos-Pinto, M., Charles, F. & Després, B. (2019) Algorithms for positive polynomial approximation. *SIAM J. Numer. Anal.*, **57**(1), 148–172.
9. Candes, E., Romberg, J. & Tao, T. (2006) Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inform. Theory*, **52**(2), 489–509.
10. Celledoni, E., McLachlan, R. I., McLaren, D. I., Owren, B., Quispel, G. R. W. & Wright, W. M. (2009) Energy-preserving Runge-Kutta methods. *ESAIM Math. Model. Numer. Anal.*, **43**(4), 645–649.
11. Cheney, W. & Goldstein, A. A. (1959) Proximity maps for convex sets. *Proc. Amer. Math. Soc.*, **10**(3), 448–450.
12. Cohen, A., Dahmen, W. & DeVore, R. (2009) Compressed sensing and best k-term approximation. *J. Amer. Math. Soc.*, **22**(1), 211–231.
13. Després, B. (2017) Polynomials with bounds and numerical approximation. *Numer. Algorithms*, **76**(3), 829–859.
14. DeVore, R. A. (1974) Degree of Monotone Approximation. in *Linear Operators and Approximation II / Lineare Operatoren und Approximation II: Proceedings of the Conference held at the Oberwolfach Mathematical Research Institute, Black Forest, March 30–April 6, 1974 / Abhandlungen zur Tagung im Mathematischen Forschungsinstitut Oberwolfach, Schwarzwald, vom 30. März bis 6. April 1974*, ed. by P. L. Butzer, & B. Szőkefalvi-Nagy, pp. 337–351. Birkhäuser Basel, Basel.
15. Donoho, D. (2006) Compressed sensing. *IEEE Trans. Inform. Theory*, **52**(4), 1289–1306.
16. Ferreira, O., Iusem, A. & Németh, S. (2014) Concepts and techniques of optimization on the sphere. *Top*, **22**(3), 1148–1170.
17. Ferreira, O. P., Iusem, A. N. & Németh, S. Z. (2013) Projections onto convex sets on the sphere. *J. Global Optim.*, **57**(3), 663–676.
18. Gander, W., Golub, G. H. & Von Matt, U. (1989) A constrained eigenvalue problem. *Linear Algebra Appl.*, **114**, 815–839.
19. Goberna, M. Á. & López, M. A. (2013) *Semi-Infinite Programming: Recent Advances*. Springer Science & Business Media.
20. Hager, W. W. (2001) Minimizing a quadratic over a sphere. *SIAM J. Optim.*, **12**(1), 188–208.
21. Hairer, E. (2010) Energy-preserving variant of collocation methods. *JNAIAM. J. Numer. Anal. Ind. Appl. Math.*, **5**, 73–84.
22. Hauck, C. & McClarren, R. (2010) Positive P_N closures. *SIAM J. Sci. Comput.*, **32**(5), 2603–2626, Publisher: Society for Industrial and Applied Mathematics.
23. Hettich, R. & Kortanek, K. O. (1993) Semi-infinite programming: Theory, methods, and applications. *SIAM Rev.*, **35**(3), 380–429.
24. Krakowski, K. A., Hüper, K. & Manton, J. H. (2007) On the computation of the karcher mean on spheres and special orthogonal groups. *RoboMat 2007, Workshop on Robotics and Mathematics*. Coimbra, Portugal: Centro Internacional de Matemática, pp. 119–124.
25. Laiu, M. P., Hauck, C. D., McClarren, R. G., O'Leary, D. P. & Tits, A. L. (2016) Positive filtered P_N moment closures for linear kinetic equations. *SIAM J. Numer. Anal.*, **54**(6), 3214–3238.
26. Nesterov, Y. (2000) Squared Functional Systems and Optimization Problems. *High Performance Optimization*. (H. Frenk, K. Roos, T. Terlaky & S. Zhang eds). US, Boston, MA: Springer, pp. 405–440.
27. Nie, J. & Demmel, J. W. (2006) Shape Optimization of Transfer Functions. *Multiscale Optimization Methods and Applications*. (W. W. Hager, S.-J. Huang, P. M. Pardalos & O. A. Prokopyev eds). US, Boston, MA: Springer, pp. 313–326.
28. Nochetto, R. & Wahlbin, L. (2002) Positivity preserving finite element approximation. *Math. Comp.*, **71**(240), 1405–1419.
29. Quispel, G. & McLaren, D. I. (2008) A new class of energy-preserving numerical integration methods. *J. Phys. A*, **41**(4), 045206.
30. Rendl, F. & Wolkowicz, H. (1997) A semidefinite framework for trust region subproblems with applications to large scale minimization. *Math. Programming*, **77**(1), 273–299.

31. Sorensen, D. C. (1997) Minimization of a large-scale quadratic function subject to a spherical constraint. *SIAM J. Optim.*, **7**(1), 141–161.
32. Stein, O. (2012) How to solve a semi-infinite optimization problem. *European J. Oper. Res.*, **223**(2), 312–320.
33. Trefethen, L. N. (2012) *Approximation Theory and Approximation Practice.* SIAM.
34. Vandenberghe, L. & Boyd, S. (1996) Semidefinite programming. *SIAM Rev.*, **38**(1), 49–95.
35. Zala, V., Kirby, M. & Narayan, A. (2020) Structure-preserving function approximation via convex optimization. *SIAM J. Sci. Comput.*, **42**(5), A3006–A3029.
36. Zhang, X. & Shu, C.-W. (2011) Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: survey and new developments. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **467**(2134), 2752–2776.