Multi-dimensional summation-by-parts operators for general function spaces: Theory and construction

Jan Glaubitz a, Simon-Christian Klein b, Jan Nordström c,d,*, Philipp Öffner e

a Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, USA
b Department of Mathematics, TU Braunschweig, Germany
c Department of Mathematics and Applied Mathematics, University of Johannesburg, South Africa
d Institute of Mathematics, Johannes Gutenberg University Mainz, Germany

A R T I C L E   I N F O

Article history:
Received 25 January 2023
Received in revised form 31 May 2023
Accepted 10 July 2023
Available online 22 July 2023

Keywords:
Summation-by-parts operators
Multi-dimensional
Mimetic discretization
General function spaces
Initial boundary value problems
Stability

A B S T R A C T

Summation-by-parts (SBP) operators allow us to systematically develop energy-stable and high-order accurate numerical methods for time-dependent differential equations. Until recently, the main idea behind existing SBP operators was that polynomials can accurately approximate the solution, and SBP operators should thus be exact for them. However, polynomials do not provide the best approximation for some problems, with other approximation spaces being more appropriate. We recently addressed this issue and developed a theory for one-dimensional SBP operators based on general function spaces, coined function-space SBP (FSBP) operators. In this paper, we extend the theory of FSBP operators to multiple dimensions. We focus on their existence, connection to quadratures, construction, and mimetic properties. A more exhaustive numerical demonstration of multi-dimensional FSBP (MFSBP) operators and their application will be provided in future works. Similar to the one-dimensional case, we demonstrate that most of the established results for polynomial-based multi-dimensional SBP (MSBP) operators carry over to the more general class of MFSBP operators. Our findings imply that the concept of SBP operators can be applied to a significantly larger class of methods than is currently done. This can increase the accuracy of the numerical solutions and/or provide stability to the methods.

© 2023 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

Summation-by-parts (SBP) operators mimic integration-by-parts on a discrete level. In combination with weakly enforced boundary conditions (BCs), they allow for a systematic development of energy-stable numerical methods for energy-bounded initial boundary value problems (IBVPs) [75,17,13]. Examples include finite difference (FD) [49,50,69,71], essentially non-oscillatory (ENO) and weighted ENO (WENO) [77,24,7], finite/spectral element (FE/SE) [8,1,2], discontinuous Galerkin (DG) [28,12], finite volume (FV) [59,60], flux reconstruction (FR) [45,67,63], and implicit time integration [61,53,66] methods. At their core, existing SBP operators are constructed to be exact for polynomials up to a certain degree. The underlying...
assumption—although not always stated explicitly—is that polynomials accurately approximate the solution of the problem at hand. However, for some IBVPs, polynomials are not the best choice, and other approximation spaces should be used. Many previous works have pointed out the advantages of non-polynomial approximation spaces. These include exponentially fitted schemes to solve singular perturbation problems [47,48], DG methods [78] and (W)ENO reconstructions [14,46,42] based on non-polynomial approximation spaces, radial basis function (RBF) schemes [22,26,25], and methods based on rational function approximations [57,38]. In [36], we recently developed one-dimensional SBP operators for general (non-polynomial) function spaces, referred to as function-space SBP (FSBP) operators. Although these can be applied to multi-dimensional problems using a tensor-product strategy, with the advantage of being simple and often efficient, this is not always the best choice since it limits their geometric flexibility.

Here, we develop a theory for multi-dimensional FSBP (MFSBP) operators. We focus on establishing their existence, connection to quadratures, construction, and mimetic properties of diagonal-norm MFSBP operators for general function spaces and geometries. It is demonstrated that most of the established results for previously developed multi-dimensional polynomial SBP (MSBP) operators [60,43,18] carry over to MFSBP operators. We specifically connect the existence of MFSBP operators and positive quadratures that are exact for certain, in general, non-polynomial function spaces. Building upon the theoretical existence investigation, we derive a general construction procedure for MFSBP operators. An essential part of this procedure is that positive quadratures that are exact for a certain function space must be found if an $F$-exact MFSBP operator approximating the partial derivative $\partial_\xi$ is desired. Note that the MFSBP operators presented here are not optimized w.r.t. their grid. While we desire to investigate the optimal accuracy and efficiency of our MFSBP operators, such efforts need to be tailored to specific function spaces. They will therefore be carried out in future work.

Our findings imply that the concept of MSBP operators can be applied to a significantly larger class of methods than is currently known. Another aspect of introducing the SBP framework in an existing “old” method is to gain stability, as done for the FV method in [59,60] and for RBF methods in [35], when combined with weakly enforced boundary data [33,34]. Here, we demonstrated the advantage of MFSBP operators for different linear problems. A more exhaustive numerical study will be provided in future work.

The rest of this work is organized as follows. In section 2, we establish the notation that will be used subsequently. In section 3, we introduce the concept of MFSBP operators for general function spaces and geometries. Section 4 addresses the relationship between MFSBP operators and certain surface and volume quadratures, characterizing the theoretical existence of diagonal-norm MFSBP operators. In section 5, we describe how diagonal-norm MFSBP operators can be constructed. Section 6 demonstrates that the established mimetic properties of MSBP operators carry over to the larger class of MFSBP operators. In section 7, we provide illustrative examples of MFSBP operators on triangles and circles. Section 8 offers simplistic numerical tests demonstrating the potential advantage of MFSBP operators. Finally, we offer concluding thoughts in section 9.

2. Notation

We use the following notation in the remainder of this work. Let $d \in \mathbb{N}$ be a positive integer, $\Omega \subset \mathbb{R}^d$ be an open and bounded domain with piecewise-smooth boundary $\partial \Omega$, and $n = [n_1, \ldots, n_d]^T$ be the corresponding outward pointing unit normal. Moreover, let $S = \{x_i\}_{i=1}^N$ be a set of $N$ nodes on $\Omega$ and $\xi \in \mathbb{R}^d$ be an arbitrary directional vector with length one. If $f \in C^1(\Omega)$ is a continuously differentiable function on $\Omega$, then

$$f = [f(x_1), \ldots, f(x_N)]^T,$$
$$\xi f = [(\partial_\xi f)(x_1), \ldots, (\partial_\xi f)(x_N)]^T,$$  \hspace{1cm} (2.1)

denote the nodal values of $f$ and its directional derivative $\partial_\xi f$ in $\xi$-direction on the node set $S$, respectively. Here, $\partial_\xi f$ is the directional derivative defined as $\partial_\xi f = \nabla f \cdot \xi$. Note that, for $i \in \{1, \ldots, d\}$, the usual partial derivative $\partial_\xi f$ corresponds to the directional derivative $\partial_{\xi_i} f$ in the $i$th component.

Let $P_p(\mathbb{R}^d)$ be the space of the $d$-dimensional polynomials of total degree up to $p$, which has dimension $n_p^d = \binom{p+d}{d}$. Given an exponent vector $\alpha = [\alpha_1, \ldots, \alpha_d] \in \mathbb{N}^d$, recall that the total degree of a monomial $x_1^{\alpha_1} \cdots x_d^{\alpha_d}$ is the $\ell^1$-norm of $\alpha$, $\|\alpha\|_1 = \alpha_1 + \cdots + \alpha_d$, and that the total degree of a polynomial is the largest total degree of all (non-zero coefficient) monomials spanning the polynomial. $P_p(\mathbb{R}^d)$ is therefore spanned by the monomial basis functions,

$$x^{\alpha} := x_1^{\alpha_1} \cdots x_d^{\alpha_d}, \quad \|\alpha\|_1 \leq p,$$  \hspace{1cm} (2.2)

where $\alpha = [\alpha_1, \ldots, \alpha_d]$ and $x = [x_1, \ldots, x_d]$. The nodal values of the monomial basis functions $x^{\alpha}$ and their derivative in $\xi$-direction, $\partial_\xi x^{\alpha}$, on the node set $S$ are denoted by $x^{\alpha}_{\xi} = (x^{\alpha})_{\xi}$, respectively.

3. Multi-dimensional SBP operators for general function spaces

We start by describing how MSBP operators can be extended to general function spaces.
3.1. Multi-dimensional polynomial SBP operators

MSBP operators were first introduced for FV methods in [60,74,73] and later for multi-block FD and SE methods in [43,18]. The first derivative operator is defined as follows.

**Definition 3.1 (SBP operators).** $D_\xi = P^{-1} Q_\xi$ is an SBP operator of (total) degree $p$, approximating the first derivative operator $\partial_\xi$ on the node set $\mathcal{S}$, if

(i) $D_\xi \mathbf{x}^\alpha = (\mathbf{x}^\alpha)^T_\xi$ for $\|\alpha\|_1 \leq p$,

(ii) the norm matrix $P$ is symmetric and positive definite (SPD),

(iii) $Q_\xi + Q_\xi^T = B_\xi$, and

(iv) the boundary matrix $B_\xi$ satisfies

\[
(\mathbf{x}^\alpha)^T B_\xi \mathbf{x}^\beta = \oint_{\partial \Omega} \mathbf{x}^\alpha \mathbf{x}^\beta (\xi \cdot \mathbf{n}) \, ds, \quad \|\alpha\|_1, \|\beta\|_1 \leq q, \tag{3.1}
\]

where $q \geq p$ and $(\xi \cdot \mathbf{n})$ is the inner product of the directional vector $\xi$ and the outward pointing unit normal $\mathbf{n}$.

Relation (i) ensures that $D_\xi$ is an accurate approximation of the continuous operator $\partial_\xi$ by requiring the operator to be exact for all $d$-dimensional polynomials of total degree up to $p$. Condition (ii) guarantees that $P$ induces a proper discrete inner product and norm, which are respectively given by $(\mathbf{u}, \mathbf{v})_P = \mathbf{u}^T P \mathbf{v}$ and $\|\mathbf{u}\|_P = \mathbf{u}^T P \mathbf{u}$ for $\mathbf{u}, \mathbf{v} \in \mathbb{R}^N$. Relation (iii) encodes the SBP property, which allows us to mimic integration-by-parts (IBP) on a discrete level. Recall that IBP for the $\xi$-derivative reads

\[
\int_{\Omega} u (\partial_\xi v ) \, dx + \int_{\Omega} (\partial_\xi u) v \, dx = \oint_{\partial \Omega} u v (\xi \cdot \mathbf{n}) \, ds, \quad \forall u, v \in C^1(\Omega). \tag{3.2}
\]

The discrete version of (3.2), which follows from (iii), is

\[
\mathbf{u}^T P (D_\xi \mathbf{v}) + (D_\xi \mathbf{u})^T P \mathbf{v} = \mathbf{u}^T B_\xi \mathbf{v}, \quad \forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^N. \tag{3.3}
\]

Note that the two terms on the left-hand side of (3.3) approximate the related terms on the left-hand side of (3.2). Finally, (iv) in Definition 3.1 ensures that also the right-hand side of (3.3) accurately approximates the right-hand side of (3.2). To this end, the boundary operator $B_\xi$ must be exact for $d$-dimensional polynomials of total degree up to $p$.

3.2. Multi-dimensional SBP operators for general function spaces

The main idea behind (i) in Definition 3.1 is that polynomials of total degree up to $p$ approximate the PDE solution well for $p$ high enough, and the differentiation operator $D_\xi$ should thus be exact for them. That is, we can reformulate (i) in Definition 3.1 as

\[
D_\xi f = f_\xi \quad \forall f \in \mathcal{P}_p(\mathbb{R}^d), \tag{3.4}
\]

where $\mathcal{P}_p(\mathbb{R}^d)$ denotes the space of the $d$-dimensional polynomials of total degree up to $p$. It is now clearly possible to replace $\mathcal{P}_p(\mathbb{R}^d)$ with any other subspace of $C^1(\Omega)$. Suppose it is reasonable to approximate the solution $u$ using the function space $\mathcal{F} \subset C^1(\Omega)$. We then modify (3.4) to

\[
D_\xi f = f_\xi \quad \forall f \in \mathcal{F}. \tag{3.5}
\]

Similarly, we can modify (3.1) to

\[
f_\xi^T B_\xi g = \oint_{\partial \Omega} f g (\xi \cdot \mathbf{n}) \, ds, \quad \forall f, g \in \mathcal{G}, \tag{3.6}
\]

with $\mathcal{F} \subset \mathcal{G}$, where for simplicity, we choose $\mathcal{G} = \mathcal{F}$. As for one-dimensional SBP operators [36], it is now natural to formulate the following generalization of multi-dimensional SBP operators, which we refer to as multi-dimensional function-space SBP (MFSBP) operators.

**Definition 3.2 (MFSBP operators).** $D_\xi = P^{-1} Q_\xi$ is an $\mathcal{F}$-based MFSBP operator, approximating the first derivative operator $\partial_\xi$ on the node set $\mathcal{S}$, if

(i) $D_\xi f = f_\xi$ for all $f \in \mathcal{F}$,
(ii) the norm matrix $P$ is SPD,
(iii) $Q_{\xi} + Q_{\xi}^T = B_{\xi}$, and
(iv) the boundary matrix $B_{\xi}$ satisfies
\[
\mathbf{f}^T B_{\xi} \mathbf{g} = \int_{\partial \Omega} f g (\xi \cdot \mathbf{n}) \, ds, \quad \forall f, g \in \mathcal{F}.
\]
(3.7)

Note that only (i) and (iv) in Definition 3.2 differ from Definition 3.1. Consequently, most of the results for polynomial MSBP operators carry over to our MFSBP operators, as we will demonstrate in the remainder of this paper. A similar observation was made in the one-dimensional case [36].

For simplicity, we focus on diagonal-norm MFSBP operators, for which the norm matrix $P$ is diagonal. This allows us to connect them to certain quadratures (see section 4), which simplifies their construction (see section 5) and analysis (see section 6). Furthermore, diagonal-norm SBP operators enable certain splitting techniques [58,29,65] and the extension to variable coefficients including curvilinear coordinates [72,62,68,11]. For the same reasons, we also restrict the discussion to diagonal boundary matrices [62,18,13].

4. MFSBP operators and quadratures

We now investigate the connection between MFSBP operators and certain quadratures. Similar to the one-dimensional case [36], we show that an MFSBP operator exists if and only if a specific quadrature exists.

4.1. Volume quadratures

We start by briefly commenting on multi-dimensional volume and surface quadratures [20,15,16]. Consider the volume quadrature $I_{X,W}$ on $\Omega \subset \mathbb{R}^d$ with nodes $X = \{x_n/m=1 \subset \Omega$ and weights $W = \{w_n/m=1$,
\[
I_{X,W}[f] := \sum_{n=1}^N w_n f(x_n) \approx \int_\Omega f(x) \, dx =: I[f],
\]
(4.1)
where $f : \Omega \rightarrow \mathbb{R}$ is a continuous function. We say that the quadrature $I_{X,W}$ is positive if its weights are positive, i.e., if $w_n > 0$ for all $w_n \in W$. Furthermore, given a function space $\mathcal{G}$, we say that $I_{X,W}$ is $\mathcal{G}$-exact if the exactness condition
\[
I_{X,W}[g] = I[g] \quad \forall g \in \mathcal{G}
\]
(4.2)
holds.

4.2. Characterizing the existence of MFSBP operators

As stated above, the restriction to diagonal norm matrices $P$ allows us to characterize the existence of diagonal-norm MFSBP operators in terms of positive quadratures. To this end, we introduce the concept of Vandermonde matrices. Let $\{f_1, \ldots, f_K\}$ be a basis of $\mathcal{F} \subset C^1(\Omega)$. Evaluating the basis functions at the nodes $S = \{x_n/m=1$, and writing the corresponding function values as the columns of a matrix, we get the Vandermonde matrix
\[
V = [f_1, \ldots, f_K] = \begin{bmatrix} f_1(x_1) & \cdots & f_K(x_1) \\
\vdots & \ddots & \vdots \\
f_1(x_N) & \cdots & f_K(x_N) \end{bmatrix}.
\]
(4.3)
Furthermore, for $\xi \in \mathbb{R}^d$, we denote by
\[
\partial_\xi (\mathcal{F}^2) = \{ \partial_\xi (fg) \mid f, g \in \mathcal{F} \} = \{ (\partial_\xi f)g + f(\partial_\xi g) \mid f, g \in \mathcal{F} \}
\]
(4.4)
the space of functions corresponding to the derivative in $\xi$-direction of the product of two functions from $\mathcal{F} \subset C^1(\Omega)$. We are now positioned to formulate our main result on the connection between diagonal-norm MFSBP operators and positive and $\partial_\xi (\mathcal{F}^2)$-exact quadratures.

**Theorem 4.1.** Let $\mathcal{F} \subset C^1(\Omega)$ and assume that the Vandermonde matrix $V$ in (4.3) has linearly independent columns. Further, let $B_{\xi}$ be a boundary matrix satisfying (iv) in Definition 3.2. Then there exists a diagonal-norm $\mathcal{F}$-based MFSBP operator $D_{\xi} = P^{-1}Q_{\xi}$ (with $Q_{\xi} + Q_{\xi}^T = B_{\xi}$) on the nodes $S$ if and only if there exists a positive and $\partial_\xi (\mathcal{F}^2)$-exact quadrature on $\Omega$ with nodes $S$.\[4]
Theorem 4.1 is well-known for polynomial MSBP operators. The assertion is proved using the same arguments as in the proofs of [43, Theorems 3.2 and 3.3] and [18, Theorem 2].

Remark 4.2. The Vandermonde matrix \( V \) in (4.3) is ensured to have linearly independent columns if the node set \( S \) is \( \mathcal{F} \)-unisolvent.\(^1\) This requirement is not restrictive since we never encountered a Vandermonde matrix \( V \) with linearly dependent columns for \( N \geq K \) in our numerical tests. This might not be surprising since \( V \) having linearly independent columns can be ensured if sufficiently many dense nodes are used [32, Section 2.1].

### 4.3. Surface quadratures

We can also connect the existence of the boundary operator \( B_\xi \) in Definition 3.2 to certain surface quadratures on the boundary \( \Gamma = \partial \Omega \). Consider the surface quadrature \( I_{Y,V}^{(\Gamma)} \) on the closed boundary \( \Gamma = \partial \Omega \) with surface nodes \( Y = \{x_m\}_{m=1}^M \subset \Gamma \) and weights \( V = \{v_m\}_{m=1}^M \) satisfying

\[
I_{Y,V}^{(\Gamma)}[f] := \sum_{m=1}^M v_m(\xi \cdot n(x_m)) f(x_m) \approx \oint_{\Gamma} f(\xi \cdot n) \, ds \equiv: I^{(\Gamma)}[f],
\]

where \( f : \Omega \to \mathbb{R} \) again is a continuous function. We say that the surface quadrature \( I_{Y,V}^{(\Gamma)} \) in (4.5) is positive if its weights \( v_1, \ldots, v_M \) are positive. Furthermore, given a function space \( \mathcal{G} \subset C(\Omega) \), we say that \( I_{Y,V}^{(\Gamma)} \) is \( \mathcal{G} \)-exact if the exactness condition

\[
I_{Y,V}^{(\Gamma)}[g] = I^{(\Gamma)}[g] \quad \forall g \in \mathcal{G}
\]

holds. We pre-empt that the function space \( \mathcal{G} \) for which we need exact surface quadratures will be \( \mathcal{F}^2 \) for reasons explained in the following.

Remark 4.3. The relation (4.5) assumes that we have access to the exact normals. This does not introduce a constraint for domains with a piecewise linear boundary, where the normals remain constant on each linear boundary segment. However, for complicated geometries, the normals include higher-order terms that must be integrated alongside \( \mathcal{F}^2 \) by the surface quadrature. This integration may necessitate a larger number of surface points. Importantly, the added complexity is not exclusive to the proposed MFSBP operators but is generally valid.

### 4.4. Characterizing the existence of boundary operators

We now connect the boundary operators \( B_\xi \), which are crucial to Definition 3.2 of MFSBP operators \( D_\xi = P^{-1} Q_\xi \) with \( Q_\xi + Q_\xi^T = B_\xi \), to surface quadratures on the boundary \( \Gamma = \partial \Omega \). We restrict the discussion to diagonal boundary operators supported on the surface nodes.

**Definition 4.4 (Mimetic boundary operators).** Let \( B_\xi \in \mathbb{R}^{N \times N} \) be a boundary operator on \( \Gamma = \partial \Omega \) with

\[
f^\top B_\xi g = \oint_{\Gamma} f g(\xi \cdot n) \, ds, \quad \forall f, g \in \mathcal{F},
\]

where \( (\xi \cdot n) \) is the inner product of the directional vector \( \xi \) and the outward pointing unit normal \( n \). Assume that \( B_\xi \) is defined on the node set \( S = \{x_n\}_{n=1}^N \subset \Omega \). We call \( B_\xi \) mimetic if

- (i) \( B_\xi \) is diagonal, i.e., \( B_\xi = \text{diag}(b_1, \ldots, b_N) \);
- (ii) \( B_\xi \) only acts on the boundary nodes, i.e., \( b_n = 0 \) if \( x_n \notin \Gamma \) for \( n = 1, \ldots, N \);
- (iii) The nonzero entries of \( B_\xi \) are of the form \( b_n = v_n(\xi \cdot n(x_n)) \), where \( v_n \) is a positive weight.

Remark 4.5. The mimetic boundary operators in Definition 4.4 incorporate all geometric and direction information necessary for performing the IBP procedure, which makes them closely connected to the encapsulated boundary operators used in [55,4,54].

We will see in a moment that Definition 4.4 allows us to identify the boundary matrix \( B_\xi \) with a positive and \( \mathcal{F}^2 \)-exact surface quadrature.

\(^1\) We say that the node set \( S \subset \Omega \) is \( \mathcal{F} \)-unisolvent, where \( \mathcal{F} \subset C(\Omega) \) is a linear function space if \( f \in \mathcal{F} \) and \( f(x) = 0 \) for all \( x \in S \) implies \( f = 0 \).
Theorem 4.6. Let $\mathcal{F} \subset C(\Omega)$, $\Omega$ be an open and bounded domain with closed boundary $\Gamma = \partial \Omega$, and $S = \{x_n\}_{n=1}^N$ be a node set on $\Omega$. There exists a mimetic boundary operator $B_\xi$ satisfying (4.7) if and only if there exists a positive and $F^2$-exact surface quadrature $I^{(F)}_{\Gamma \setminus \Omega}$ with points $Y \subset S \cap \Gamma$. Moreover, the non-zero entries of $B_\xi$ correspond to the products of the surface quadrature weights with $\xi \cdot n$ at the corresponding surface point.

Proof. Denote the surface quadrature points by $Y = \{y_m\}_{m=1}^M$ and the weights by $V = \{v_m\}_{m=1}^M$. The proof consists of two parts. We first show that the existence of a positive and $F^2$-exact surface quadrature implies the existence of a mimetic boundary operator. To this end, we construct as a diagonal matrix $B_\xi = \text{diag}(b_1, \ldots, b_N)$ as

$$b_n = \begin{cases} v_m(\xi \cdot n(x_m)) & \text{if } x_n = y_m, \\ 0 & \text{otherwise.} \end{cases}$$

(4.8)

By construction, $B_\xi$ satisfies (i), (ii), and (iii) in Definition 4.4. It remains to show that (4.7) holds. To this end, we can rewrite the left-hand side of (4.7) as

$$f^T B_\xi g = \sum_{m=1}^M v_m(\xi \cdot n(y_m)) f(y_m) g(y_m) = I^{(F)}_{\Gamma \setminus \Omega} [fg].$$

(4.9)

using our above construction of $B_\xi$. Now let $f, g \in \mathcal{F}$. Then $fg \in F^2$, and since $I^{(F)}_{\Gamma \setminus \Omega}$ is $F^2$-exact, we have $I^{(F)}_{\Gamma \setminus \Omega} [fg] = I^{(F)} [fg]$. Substituting this into (4.9) yields

$$f^T B_\xi g = I^{(F)} [fg] = \oint_{\Gamma} fg(\xi \cdot n) \, ds,$$

(4.10)

which shows that $B_\xi$ satisfies (4.7) and therefore is a mimetic boundary operator. In the second part of the proof, similar arguments as in the first part can be used to show that the existence of a mimetic boundary operator $B_\xi = \text{diag}(b_1, \ldots, b_N)$ implies the existence of a positive and $F^2$-exact surface quadrature $I^{(F)}_{\Gamma \setminus \Omega}$. □

5. Construction of MFSBP operators

Let $\Omega \subset \mathbb{R}^d$ be an open and bounded reference domain, let $\xi \in \mathbb{R}^d$, and let $\mathcal{F} \subset C^1(\Omega)$ be the function space for which we want a diagonal-norm MFSBP operator $D_\xi$ approximating $\partial_\xi$. The construction then proceeds as follows:

(S1) Select $N_s$ and $N_i$ nodes on the surface and interior of $\Omega$, respectively. The node set $S$ is the union of the surface and interior nodes and contains a total number of $N = N_s + N_i$ nodes.

(S2) Find a positive and $F^2$-exact surface quadrature supported on the $N_s$ surface points. Use the surface quadrature to construct a mimetic boundary matrix $B_\xi$.

(S3) Find a positive and $\partial_\xi(F^2)$-exact volume quadrature supported on the $N_i$ surface and interior points. Use the volume quadrature to construct a diagonal-norm matrix $P$.

(S4) Find a matrix $Q_\xi$ that satisfies Definition 3.2. Then construct an MFSBP operator as $D_\xi = P^{-1} Q_\xi$.

We address details on (S2), (S3), and (S4) in subsections 5.1 to 5.3. Furthermore, we comment on (S1), particularly how the nodes can be distributed, in subsection 5.4.

5.1. The boundary matrix $B_\xi$

We first describe how one can construct a mimetic boundary matrix $B_\xi$, satisfying Definition 4.4. Given is a reference element $\Omega$ with piecewise smooth boundary $\Gamma$, where we denote the smooth parts by $\Gamma_1, \ldots, \Gamma_J$. If we find mimetic boundary matrices $B_j$ on $\Gamma_j$, $j = 1, \ldots, J$, with

$$f^T B_j g = \int_{\Gamma_j} fg(\xi \cdot n) \, ds, \quad \forall f, g \in \mathcal{F},$$

(5.1)

then we get the desired mimetic boundary matrix $B_\xi$ as $B_\xi = \sum_{j=1}^J B_j$. Following Theorem 4.6, we can find such matrices $B_j$ in two steps: (i) Construct a positive and $F^2$-exact surface quadrature,

$$I^{(F)}_{\Gamma \setminus \Omega(j)} [fg] = \sum_{m=1}^{M_j} v_m^{(j)}(\xi \cdot n(x_m^{(j)})) f(x_m^{(j)}) g(x_m^{(j)}) \approx \oint_{\Gamma_j} fg(\xi \cdot n) \, ds = I^{(F)}_{\Gamma_j} [fg].$$

(5.2)
Fig. 1. Illustration of the POC algorithm. The two convex sets $C$ and $D$ correspond to (affine) linear subspaces. Starting from an initial point, the POC algorithm produces a sequence of points (illustrated by the blue dots) that converges to a point in the intersection of $C$ and $D$. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

(ii) Use the products $v_m^{(j)}(\xi \cdot n(x_m^{(j)}))$ as the non-zero diagonal entries of $B_j$ that correspond to the surface nodes on $\Gamma_j$. That is, we get

$$
(b_j)_n = \begin{cases} 
  v_m^{(j)}(\xi \cdot n(x_m^{(j)})) & \text{if } x_n = x_m^{(j)} \in \Gamma_j, \\
  0 & \text{otherwise}.
\end{cases}
$$

We were able to find positive and $\mathcal{F}^2$-exact surface quadratures using equidistant points on $\Gamma_j$ using the least-squares approach \cite{37,30,32} and the Projection Onto Convex Sets (POCS) algorithm \cite{76,40,21} if sufficiently many grid points were used. Given the two overlapping closed convex sets $C$ and $D$, the POC algorithm finds a point $x \in C \cap D$ by alternatingly projecting onto the sets $C$ and $D$. To find a positive and $\mathcal{F}^2$-exact surface quadrature, the sets $C$ and $D$ encode the positivity of the weights and exactness conditions of an $\mathcal{F}^2$-exact surface quadrature, respectively. Fig. 1 illustrates the POC algorithm for the case of $C$ and $D$ being (affine) linear spaces. We chose POCs because it is remarkably versatile. POCs can be employed not only for solving the surface quadrature but also for the volume quadrature (necessitating the enforcement of a lower bound constraint) and the anti-symmetric matrix $Q_A$ (requiring an anti-symmetric constraint). Switching between these diverse constraints demands minimal alterations to the POC algorithm and its implementation. Concurrently, we noted that the computational costs of the POC algorithm were competitive when compared to alternative methods. See Appendix A.1 for more details on the POC algorithm.

**Remark 5.1.** It is a non-trivial task in hyperbolic problems to properly treat element corners, where the normal vector becomes undefined. To avoid that difficulty (which exists for all numerical methods), we do not place any points on the corners of the domain $\Omega$.

**Remark 5.2.** Inter-element conservation in a multi-element/block method can most easily be ensured if the quadratures at the surface that connect two elements (i) have their points at the same locations and (ii) use the same weights $v_m^{(j)}$ at these points. Then, the inter-element contributions of both elements cancel out, and mass is neither created nor destroyed between elements. Otherwise, special projection and coupling methods have to be utilized \cite{68,9,10}.

**Remark 5.3 (Computational costs).** Most of the computational expense in finding the boundary matrix is dedicated to the numerical computation of the basis functions’ moments (their integrals) up to machine precision, which is needed for the exactness conditions. Relative to this, executing the POC algorithm is an inexpensive process. When executed on a single core, the total time to construct any of the two-dimensional MFSBP operators considered in the later numerical tests (see section 7 for more details) on the reference element was on the order of a second.

### 5.2. The diagonal norm matrix $P$

Assume that we have a positive and $\partial_k(\mathcal{F}^2)$-exact volume quadrature $I_{x,W}$ on $\Omega$ with nodes $X$ and weights $W = \{w_n\}_{n=1}^N$. Following Theorem 4.1, we can construct a diagonal-norm MFSBP operator $D_k = P^{-1} Q_x$ on the nodes $X$ with a diagonal norm matrix

$$
P = \text{diag}(w_1, \ldots, w_N).
$$

In general, we found such a positive and $\partial_k(\mathcal{F}^2)$-exact volume quadrature using the $N_x$ surface points from constructing the mimetic boundary matrix $B_x$ (see subsection 5.1), adding $N_i$ points in the interior of $\Omega$, and then using the POC algorithm. See Appendix A.2 for more details.

**Remark 5.4.** Consider a discretized problem that necessitates using MFSBP operators in different directions. For instance, in the $x$-direction, we have $D_x = P^{-1} Q_x$, while in the $y$-direction, we have $D_y = P^{-1} Q_y$. The norm operator $P$ must be
the same for both MFSBP operators to ensure energy stability, which we demonstrate in subsection 6.3. In such scenarios, \( P \) must correspond to a volume quadrature that is positive and concurrently exact for both \( \partial_x(F^2) \) and \( \partial_y(F^2) \). The two-dimensional function spaces that we examine later in sections 7 and 8 satisfy the condition \( \partial_x(F^2) = \partial_y(F^2) \). This congruence significantly simplifies the construction procedure. On the other hand, if \( \partial_x(F^2) \neq \partial_y(F^2) \), then the union of \( \partial_x(F^2) \) and \( \partial_y(F^2) \) encompasses a larger function space than each individual one and a larger number of quadrature points might be necessary.

5.3. The matrix \( Q_\xi \) and the MFSBP operator \( D_\xi = P^{-1}Q_\xi \)

In the last step, we construct the matrix \( Q_\xi \) from which we get the desired MFSBP operator as \( D_\xi = P^{-1}Q_\xi \). To this end, \( Q_\xi \) is decomposed into its symmetric and anti-symmetric parts, \( Q_\xi \) and \( Q_A \). Condition (iii) in Definition 3.2 then yields

\[
Q_\xi = Q_A + \frac{1}{2}B_\xi. \tag{5.5}
\]

Furthermore, the accuracy condition (i) in Definition 3.2 implies

\[
Q_A V = PV_\xi - \frac{1}{2}B_\xi V. \tag{5.6}
\]

Here, \( V_\xi = [(f_1)_\xi, \ldots, (f_K)_\xi] \) is the Vandermonde matrix for the derivatives in \( \xi \)-direction of the basis functions \( f_1, \ldots, f_K \) of the function space \( \mathcal{F} \subset C^1(\Omega) \). It remains to construct an anti-symmetric matrix \( Q_A \) that satisfies (5.6). Once such \( Q_A \) is found, we get \( Q \) by (5.5). Finding an anti-symmetric solution \( Q_A \) of (5.6) can be formulated as a classical Procrustes problem [44, 53, 59], with many available solution procedures. For instance, [43, 18, 36] used a least-squares approach to determine \( Q_A \). In our implementation, we used the POCM algorithm to find an anti-symmetric \( Q_A \) that satisfies (5.6) for efficiency reasons. We refer to Appendix A.3 for more details on how the POCM algorithm can be used to find \( Q_A \).

5.4. The surface and interior nodes

The only restriction on the surface points is that a positive and \( F^2 \)-exact surface quadrature has to exist for them, while the only limitation on the inner points is that a positive and \( \partial_\xi(F^2) \)-exact volume quadrature has to exist for the union of the inner and surface points. To this end, it was proved in [31, 32] that a positive and \( \mathcal{G} \)-exact quadrature can be found whenever (i) \( \mathcal{G} \) includes constants and (ii) sufficiently many points are used, where \( \mathcal{G} \) is \( F^2 \) and \( \partial_\xi(F^2) \) in the case of the surface and volume quadrature, respectively. It was numerically observed in [30–32] that the number of points has to be proportional to the squared dimension of \( \mathcal{G} \). Finally, the proof presented in [31, 32] allows for any class of equidistributed points [51], including equidistant points and low discrepancy points used in quasi-Monte Carlo methods [6, 19], e.g., Halton points [41]. In our implementation, we used equidistant surface points and inner points that are Halton-distributed. More precisely, we started by using a single surface point on each piecewise smooth boundary part and successively increased this number by one until we found a positive and \( F^2 \)-exact surface quadrature. Afterward, starting from using only the surface points (no inner points), we successively increased the number of inner (Halton) points by one until we found a positive and \( \partial_\xi(F^2) \)-exact volume quadrature.

Remark 5.5 (Halton points). The main idea behind Halton points is to generate uniformly distributed and well-spaced points throughout the unit cube. This is achieved by using different prime base numbers for each dimension and creating a sequence of numbers in each dimension that avoids large gaps or clusters. To generate Halton points within a bounded domain \( \Omega \), we initially create these points within a cube that encloses \( \Omega \), subsequently utilizing only the subset of points that lie on \( \Omega \). For a more comprehensive understanding of Halton points, see [41, 51]. We stress that using Halton points, while beneficial, is not a prerequisite for our method of constructing MFSBP operators.

6. Mimetic properties of MFSBP operators

Here, we demonstrate that (i) it is necessary to include constants in the function space \( \mathcal{F} \) for conservation and that (ii) most other mimetic results for polynomial MSBP operators [43, 18] also hold for our MFSBP operators. For simplicity, we illustrate this for the scalar linear advection equation in two dimensions.

6.1. The linear advection equation

The initial boundary value problem (IBVP) for the scalar linear advection equation with constant coefficients is

\[
\begin{align*}
\partial_t u + a \partial_x u + b \partial_y u &= 0, & \forall (x, y) \in \Omega, \ t > 0, \\
u(t, x, y) &= g(t, x, y), & \forall (x, y) \in \Gamma_-, \ t \geq 0.
\end{align*} \tag{6.1}
\]
where \( \Gamma_- = \{ (x, y) \in \partial \Omega \mid an_x + bn_y < 0 \} \) is the inflow part of the boundary of \( \Omega \) and \( \Gamma_+ = \Gamma \setminus \Gamma_- \) is the outflow part. Here, \( n_x \) and \( n_y \) are the first and second component of the outward pointing unit normal \( \mathbf{n} = [n_x, n_y]^T \) for the boundary \( \Gamma \). Let \( D_x = P^{-1}Q_x \) and \( D_y = P^{-1}Q_y \) be MFSBP operators, approximating the partial derivatives in the canonical directions, \( \partial_x \) and \( \partial_y \), with mimetic boundary operators \( B_x \) and \( B_y \). The MFSBP-SAT semi-discretization of (6.1) is formally given by

\[
\mathbf{u}_t + aD_x \mathbf{u} + bD_y \mathbf{u} = P^{-1}(aB_x + bB_y)(\mathbf{u} - \mathbf{g}).
\]

Here, \( \mathbf{u} = [u_1, \ldots, u_N]^T \) are the nodal values of the numerical solution at the nodes \( S = \{ \mathbf{x}_n \}_{n=1}^N \) and \( \mathbf{g} = [g_1, \ldots, g_N]^T \) is the boundary data vector with

\[
g_n = \begin{cases} 
g(t, \mathbf{x}_n) & \text{if } \mathbf{x}_n \in \Gamma_- , \\
u_n & \text{if } \mathbf{x}_n \in \Gamma_. \end{cases}
\]

Note that (ii) in Definition 4.4 ensures that \( B_x \) and \( B_y \) only act on surface entries corresponding to points on the boundary \( \Gamma^2 \). The right-hand side of (6.2) is a simultaneous approximation term (SAT) that weakly enforces the BC by “forcing” the numerical solution toward the boundary data \( g \) at the inflow boundary part. See [75,17,18] and references therein for more details on SATs.

Remark 6.1. In general, each SAT term on the right-hand side of (6.2) involves a free penalty parameter \( \sigma \), for instance, \( \sigma aP^{-1}B_x(\mathbf{u} - \mathbf{g}) \) instead of \( aP^{-1}B_x(\mathbf{u} - \mathbf{g}) \). We have fixed the penalty parameter to \( \sigma = 1 \), which was demonstrated to minimize boundary data errors in [36, Appendix A] (also see [3, page 11]).

6.2. Conservation

The exact solution of (6.1) satisfies the following conservation property:

\[
\frac{d}{dt} \int_{\Omega} u \, d\mathbf{x} = -\left( \int_{\Gamma_+} u(\partial_x u + bn_y) \, ds + \int_{\Gamma_-} g(\partial_x u + bn_y) \, ds \right). \tag{6.4}
\]

This means that the total amount of the quantity \( u \) is neither created nor destroyed inside the domain and only changes due to the flux across the boundaries. We now address how conservation can be mimicked in the discrete case using MFSBP operators. The left-hand side of (6.4) is approximated by \( \mathbf{1}^T \mathbf{P} \mathbf{u}_t \). Substituting the MFSBP-SAT semi-discretization (6.2), we get

\[
\mathbf{1}^T \mathbf{P} \mathbf{u}_t = -a \mathbf{1}^T \mathbf{P} D_x \mathbf{u} - b \mathbf{1}^T \mathbf{P} D_y \mathbf{u} + \mathbf{1}^T(aB_x + bB_y)(\mathbf{u} - \mathbf{g}). \tag{6.5}
\]

The MFSBP properties transform (6.5) to

\[
\mathbf{1}^T \mathbf{P} \mathbf{u}_t = a(D_x \mathbf{1})^T \mathbf{P} \mathbf{u} + b(D_y \mathbf{1})^T \mathbf{P} \mathbf{u} - \mathbf{1}^T(aB_x + bB_y)\mathbf{g}. \tag{6.6}
\]

The last term on the right-hand side of (6.6) approximates the right-hand side of (6.4). However, (6.6) also contains the additional volume terms \( a(D_x \mathbf{1})^T \mathbf{P} \mathbf{u} \) and \( b(D_y \mathbf{1})^T \mathbf{P} \mathbf{u} \), while no such terms are present in (6.4). To avoid artificial construction or destruction of the quantity \( u \),

\[
D_x \mathbf{1} = D_y \mathbf{1} = \mathbf{0} \tag{6.7}
\]

must hold. This is satisfied by construction for polynomial MSBP operators since non-zero constants are polynomials of degree zero. For MFSBP operators, we can ensure (6.7) by requiring the MFSBP operators to be exact for constants, i.e., \( 1 \in \mathcal{F} \). In this case, (6.6) becomes

\[
\mathbf{1}^T \mathbf{P} \mathbf{u}_t = -\mathbf{1}^T(aB_x + bB_y)\mathbf{g}, \tag{6.8}
\]

which is the discrete analog to (6.4).

\[\text{We can therefore assign arbitrary values to the entries in } \mathbf{g} \text{ that correspond to points in the interior of } \Omega.\]
6.3. Energy stability

The exact solution of (6.1) is energy-bounded since it satisfies

$$\frac{d}{dt} \|u\|_{L_2}^2 = - \int_{\Gamma_+} (a n_x + b n_y) u^2 \, ds - \int_{\Gamma_-} (a n_x + b n_y) g^2 \, ds,$$

(6.9)

where $\Gamma_+$ and $\Gamma_-$ are the outflow and inflow boundary part, respectively. A similar relation to (6.9) is desired in the discrete setting to establish discrete energy stability for the numerical solution. We observe that the left-hand side (6.9) is approximated by $\frac{d}{dt} u^T P u$. Substituting the MFSBP-SAT semi-discretization (6.2), we get

$$\frac{1}{2} \frac{d}{dt} u^T P u = \frac{1}{2} \frac{d}{dt} u^T P u = -a u^T P D_x u - b u^T P D_y u + u^T (a B_x + b B_y) (u - g).$$

(6.10)

The MFSBP properties transform (6.10) to

$$\frac{1}{2} \frac{d}{dt} u^T P u = a u^T D_x^T P u + b u^T D_y^T P u - u^T (a B_x + b B_y) g.$$  

(6.11)

Since $u^T D_x^T P u$ and $u^T D_y^T P u$ are real numbers, they are equal to their transposes, and summing (6.10) and (6.11) yields

$$\frac{d}{dt} u^T P u = u^T (a B_x + b B_y) (u - 2g).$$

(6.12)

Assume that $B_x$ and $B_y$ are mimetic boundary operators (see Definition 4.4). Using (i), (ii), and (iii) in Definition 4.4, the right-hand side of (6.12) reduces to

$$\frac{d}{dt} u^T P u = \sum_{m=1}^M u_m \left( (v_x)_m a n_x(x_m) + (v_y)_m b n_y(x_m) \right) (u_m - 2 g_m).$$

(6.13)

where $(b_x)_m = (v_x)_m a n_x(x_m)$ and $(b_y)_m = (v_y)_m b n_y(x_m)$ are the non-zero diagonal elements of $B_x$ and $B_y$, respectively. Note that the sum in (6.13) only includes the surface nodes $\{x_m\}_{m=1}^M \subset \Gamma$. Suppose that $B_x$ and $B_y$ are associated with the same positive and $F^2$-exact surface quadrature, i.e. $v_m = (v_x)_m = (v_y)_m$. Moreover, for ease of notation, we denote

$$\tau_m = v_m \left[ a n_x(x_m) + b n_y(x_m) \right].$$

(6.14)

Note that $\tau_m \geq 0$ if $x_m \in \Gamma_+$ and $\tau_m \leq 0$ if $x_m \in \Gamma_-$. We can therefore decompose the sum in (6.15) into two parts,

$$\frac{d}{dt} u^T P u = \sum_{x_m \in \Gamma_+} \tau_m u_m (u_m - 2 g_m) + \sum_{x_m \in \Gamma_-} \tau_m u_m (u_m - 2 g_m),$$

(6.15)

where the first contains the surface points on the outflow boundary part ($x_m \in \Gamma_+$) and the second contains the surface points on the inflow boundary part ($x_m \in \Gamma_-$). By construction of $g$ in (6.3), we have $g_m = u_m$ in the first sum and $g_m = g(t, x_m)$ in the second sum. Hence, (6.15) becomes

$$\frac{d}{dt} u^T P u = -\sum_{x_m \in \Gamma_+} \tau_m u_m^2 + \sum_{x_m \in \Gamma_-} \tau_m u_m (u_m - 2 g(t, x_m)).$$

(6.16)

Note that $u_m (u_m - 2 g(t, x_m)) = [u_m - g(t, x_m)]^2 - g(t, x_m)^2$, and we can therefore rewrite (6.16) as

$$\frac{d}{dt} u^T P u = -\sum_{x_m \in \Gamma_+} \tau_m u_m^2 - \sum_{x_m \in \Gamma_-} \tau_m g(t, x_m)^2 + \sum_{x_m \in \Gamma_-} \tau_m [u_m - g(t, x_m)]^2.$$  

(6.17)

Recall that $\tau_m = v_m \left[ a n_x(x_m) + b n_y(x_m) \right]$. Hence, the first two sums on the right-hand side of (6.17) are the discrete analog to (6.9). The additional third sum in (6.17) corresponds to a non-negative damping term that stabilizes the semi-discretization and vanishes with increasing accuracy.

7. Examples of MFSBP operators

We now exemplify the construction of MFSBP operators on a triangular and circular reference element $\Omega \subset \mathbb{R}^2$. In both cases, we consider the non-polynomial and polynomial function spaces

$$\mathcal{F}_1 = \text{span} \left\{ 1, x, y, x^2, xy, y^2, x^3, x^2 y, xy^2, y^3 \right\},$$

$$\mathcal{F}_2 = \text{span} \left\{ 1, x, y, \sin(\omega(x + y)), \cos(\omega(x + y)) \right\},$$

(7.1)
where \( \omega > 0 \) is a parameter that remains to be determined. Although the polynomial function space \( \mathcal{F}_1 \) has a higher dimension than the trigonometric function space \( \mathcal{F}_2 \), we found the trigonometric one to yield significantly more accurate numerical solutions for the specific tests carried out in subsection 8.1 and subsection 8.3. Note that for both spaces, \( \delta_i(\mathcal{F}_1^2) = \delta_i(\mathcal{F}_2^2), i = 1, 2 \). Hence, it suffices to find a positive and \( \delta_i(\mathcal{F}_1^2) \)-exact volume quadrature, for which we use equidistributed Halton points. Although using Halton points ensures the existence of the desired volume quadratures (see [32]), they are not necessarily optimal in the sense that the number of quadrature points can be notably larger than the dimension of the function space \( \delta_i(\mathcal{F}_1^2) \). Constructing optimal (so-called Gaussian) quadratures for general multi-dimensional function spaces is an open problem. Our ambition is to further investigate the optimization and efficiency of our MFSBP operators. However, such efforts need to be tailored to specific function spaces and will be carried out in future work.

Remark 7.1. In the upcoming numerical tests, we deliberately choose examples that favor the non-polynomial operators in order to exemplify the flexibility of using other function spaces as in the MFSBP operators.

7.1. MFSBP operators on triangles

We start by exemplifying the construction of MFSBP operators on the triangular domain in Fig. 2, given by

\[
\Omega = \left\{ (x, y) \in \mathbb{R}^2 \mid 0 \leq x \leq 1, 0 \leq y \leq 1 - x \right\}.
\]

Fig. 2. The surface and inner points used to construct the polynomial \( \mathcal{F}_1 \)-exact MSBP operator and (b) The 24 surface and 12 inner points used to construct the trigonometric \( \mathcal{F}_2 \)-exact MFSBP operator.
7.2. MFSBP operators on the circle

Next, we exemplify the construction of polynomial $\mathcal{F}_1$-exact and trigonometric $\mathcal{F}_2$-exact MFSBP operators, with the function spaces $\mathcal{F}_1$ and $\mathcal{F}_2$ as in (7.1), for the circular domain

$$\Omega = \left\{ \mathbf{x} \in \mathbb{R}^2 \mid \|\mathbf{x} - \mathbf{c}\| \leq r \right\}$$

(7.3)

with center $\mathbf{c} = [1/2, 1/2]^T$ and radius $r = 1/2$, see Fig. 3. The boundary of the disc consists of a single smooth part with a smoothly varying outward pointing unit normal vector $\mathbf{n}$. For the polynomial function space, we found a positive and $\mathcal{F}_2$-exact surface quadratures using 12 equidistant points on the surface. To next find a positive and $\partial_n(\mathcal{F}_2^2)$-exact volume quadrature, following the construction procedure in section 5, we had to add another 25 points in the interior of $\Omega$, using a total number of 37 points. Fig. 3 illustrates the position of these points. We could not find desired volume quadratures using fewer Halton-distributed points in the interior of $\Omega$. For the trigonometric function space, we found a positive and $\mathcal{F}_2^2$-exact surface quadratures and a positive and $\partial_n(\mathcal{F}_2^2)$-exact volume quadrature using the same points as for the polynomial $\mathcal{F}_1$-exact MSBP operator. Although the polynomial and trigonometric MFSBP operators use the same points, we found the trigonometric $\mathcal{F}_2$-exact MFSBP operator to yield significantly more accurate results for our numerical tests in subsection 8.4.

Remark 7.2. Although non-polygonal elements are rarely used in the context of SE-like methods for hyperbolic conservation laws, they are crucial in mesh-free RBF methods [23,25]. In the recent work [35], we established a stability theory for global RBF methods using FSBP operators. The MFSBP operators introduced here will allow us to extend this theory to the genuine multi-dimensional case, which will be addressed in future work, together with the extension to local RBF methods.

8. Numerical results

We focus on the homogeneous and inhomogeneous linear advection equation because it provides prototypical examples for which non-polynomial approximation spaces are advantageous. Future work will consider more sophisticated problems. We used the explicit strong stability preserving (SSP) Runge–Kutta (RK) method of third order using three stages (SSPRK(3,3)) [70] for all subsequent numerical tests. Furthermore, all tests are performed for conforming triangular grids. All tests were run with the elements scaled to be of unit length. The Julia code used to generate the numerical tests presented here is open access and can be found on GitHub.\(^3\)

\(^3\) See https://github.com/simonius/MFSBP.
8.1. The linear advection equation

Consider the IBVP for the linear advection equation with constant coefficients,
\[ \partial_t u + a \partial_x u + b \partial_y u = 0, \quad \forall (x, y) \in \Omega, \quad t > 0, \]
\[ u(0, x, y) = u_0(x, y), \quad \forall (x, y) \in \Omega, \]
\[ u(t, x, y) = g(t, x, y), \quad \forall (x, y) \in \Gamma_-, \quad t \geq 0, \]
where \( \Omega = \{0, 1\}^2 \) and \( \Gamma_- = \{(x, y) \in \partial \Omega \mid an_x + bn_y < 0\} \) is the inflow part of the boundary of \( \Omega \). In our numerical tests, we triangulated \( \Omega \) by generating \( K^2 \) equally-sized squares and then decomposing each square into two equally-sized triangles. Given MFSBP operators \( D_x = P^{-1} Q_x \) and \( D_y = P^{-1} Q_y \) with mimetic boundary operators \( B_x \) and \( B_y \), we used the multi-block semi-discretization
\[ \frac{du}{dt} + aD_x u + bD_y u = P^{-1} B_x (a u - f_x^{\text{num}}) + P^{-1} B_y (b u - f_y^{\text{num}}), \]
where \( f_x^{\text{num}} \) and \( f_y^{\text{num}} \) are the numerical flux functions in the \( x \)- and \( y \)-direction, respectively, coupling neighboring blocks and weakly enforcing the boundary condition \( u(t, x, y) = g(t, x, y) \) at the inflow part of the computation domain \( \Omega \). Recall that the boundary operators \( B_x \) and \( B_y \) include the components of the outward pointing unit normal \( n = [n_x, n_y]^T \). We used the classical local Lax–Friedrichs (LLF) flux [52], i.e.,
\[ f_x^{\text{num}} = \frac{a}{2} (u_i + u_o) - \text{sign}(n_x) \frac{c_{\text{max}}}{2} (u_o - u_i), \]
\[ f_y^{\text{num}} = \frac{b}{2} (u_i + u_o) - \text{sign}(n_y) \frac{c_{\text{max}}}{2} (u_o - u_i), \]
where \( u_i \) and \( u_o \) are the inner and outer states at the two sides of the interface of the present block. Furthermore, \( c_{\text{max}} \) is an upper bound of the advection speed, ensuring that the LLF fluxes in (8.3) are monotonic. For the linear advection equation, we chose \( c_{\text{max}} = \max[|a|, |b|] \).

We consider (8.1) with constant velocity coefficients \( a = b = 1 \), smooth initial condition \( u_0(x, y) = \sin(\pi(x + y)) \), and inflow boundary data \( g(x, y, t) = \sin(\pi(x + y) - (a + b)t) \) at the inflow part of the boundary \( \Gamma_- \). The exact reference solution is
\[ u(x, y, t) = u_0(x - at, y - bt) = \sin(\pi(x + y) - 2\pi t) \]
for \( a = b = 1 \). Fig. 4 illustrates the numerical solutions and their point-wise errors at time \( t = 1 \) for the polynomial \( \mathcal{F}_1 \)-exact and trigonometric \( \mathcal{F}_2 \)-exact MFSBP operators. Both function spaces are described in (7.1), where we chose \( \omega = \pi \) as the frequency parameter in \( \mathcal{F}_2 \). Recall from (7.1) that the polynomial function space \( \mathcal{F}_1 \) is ten-dimensional, while the trigonometric one \( \mathcal{F}_2 \) is five-dimensional. Still, comparing the errors in Figs. 4a and 4c, the trigonometric \( \mathcal{F}_2 \)-exact MFSBP operator yields significantly more accurate results. We triangulated \( \Omega \) by generating \( K^2 \) equally-sized squares with \( K = 10 \) and then decomposing each square into two equally-sized triangles.

We next provide a brief error analysis, further comparing the polynomial \( \mathcal{F}_1 \)-exact and trigonometric \( \mathcal{F}_2 \)-exact MFSBP operators. To this end, we performed the same test for the IBVP (8.1) with different numbers of triangular blocks. Fig. 5 reports on the corresponding errors. Thereby, for \( K = 3, \ldots , 10 \), we again considered the domain \( \Omega = [0, 1]^2 \), triangulated by first partitioning \( \Omega \) into \( K^2 \) equally-sized squares and then decomposing each square into two equally-sized triangles. The errors in Fig. 5 were calculated as
\[ E = \left( \frac{1}{N|T|} \sum_{T \in \mathcal{T}} \sum_{n=1}^{N} \left| u_n(T) (x_n^T, y_n^T, t) - u_n^{(T)}(t) \right|^2 \right)^{1/2}, \]
where \( \mathcal{T} \) is the collection of all triangular elements and \( u_n(T) (x_n^T, y_n^T, t) \) and \( u_n^{(T)}(t) \) denote the nodal value of the reference and numerical solution at the \( N \) grid points in the elements \( T \in \mathcal{T} \), respectively. The errors reported in Fig. 5 were obtained for a fixed time step size \( \Delta t = 10^{-3} \). We used a fixed time step size to focus solely on the errors of the spatial semi-discretizations while making errors introduced by the SSPRK(3,3) time integration method negligible. Fig. 5 demonstrates that the polynomial \( \mathcal{F}_1 \)-exact MSBP operator yields third- to fourth-order convergence. At the same time, we observe fifth-order convergence for the trigonometric \( \mathcal{F}_2 \)-exact MFSBP operator. A possible explanation for the observed, higher than expected, convergence rate of the trigonometric MFSBP operator is the vanishing residual of the operator,
\[ \|D_x V - V_x\|_2, \]
Fig. 4. Numerical solutions and their point-wise errors of the IBVP (8.1) at $t = 1$ using the polynomial $F_1$-exact and trigonometric $F_2$-exact MFSBP operators. We triangulated $\Omega$ by generating $K^2$ equally-sized squares with $K = 10$ and then decomposing each square into two equally-sized triangles. Although $F_1$ is ten-dimensional and $F_2$ is five-dimensional, the trigonometric $F_2$-exact MFSBP operator yields significantly more accurate results. The contour plots do not fully cover the corners of the computational domain due to the used plotting routine. In particular, to circumvent the issue of multivalued normals (see Remark 5.1), we refrained from positioning nodes at the corners of the triangular reference element, thereby forgoing any extrapolation to these points.

Fig. 5. Errors of the numerical solutions to the IBVP (8.1) at $t = 1$ using the polynomial $F_1$-exact (plusses along the dotted lines) and trigonometric $F_2$-exact (dots along the straight line) MFSBP operators, along the residual (8.6) of the trigonometric MFSBP operator (diamonds along the dashed line), for an increasing number of triangular blocks. The trigonometric MFSBP operator yields significantly smaller errors than the polynomial one.
which is shown in Fig. 5 to converge to zero with a similar order. Here, $\|A\|_2 = \sqrt{\sum_{n,m=1}^{N} |A_{n,m}|^2}$ is the Frobenius norm of the matrix $A \in \mathbb{R}^{N \times N}$. A more detailed investigation of the convergence rate of the proposed MFSBP operators will be provided in future works.

**Remark 8.1.** We have conducted the above experiments with various advection directions, diverging from the previously used $a = b = 1$. The different advection directions do not noticeably influence the performance of the MFSBP-based scheme. This is because the solution is a trigonometric function and is therefore well represented by the function space $\mathcal{F}_2$, regardless of the advection direction.

### 8.2. Including a radial basis function

We now revisit the test problem described in subsection 8.1, this time employing an MFSBP operator on the triangle that is exact for a function space including a Gaussian RBF. This demonstration serves two objectives: Firstly, it provides an example of a non-polynomial function space, beyond trigonometric functions. Secondly, it underscores the notion that the efficacy of MFSBP-based methodologies relies on the degree to which the function space $\mathcal{F}$ encapsulates the true underlying solution. To elaborate, note that the solution for the previously considered test problem in subsection 8.1 was a trigonometric function. Consequently, the trigonometric function space $\mathcal{F}_2$ mirrors this characteristic. In contrast, we now examine a four-dimensional function space including an RBF function, given by

$$\mathcal{F}_3 = \text{span} \left\{ 1, x, y, \exp \left( -\|x - x_0\|^2 / d^2 \right) \right\}.$$  \hfill (8.7)

This function space operates on the reference triangle, with the RBF centered at $x_0 = (1/3, 1/3)$ and displaying a characteristic diameter of $d = 1/5$. Despite our ability to construct an $\mathcal{F}_3$-exact MFSBP operator (see Remark 8.2 for more details on the construction), the resultant numerical solution is less accurate than that derived using the trigonometric $\mathcal{F}_2$-exact MFSBP or the polynomial $\mathcal{F}_1$-exact MSBP operator from before. This disparity in accuracy is depicted in Fig. 6, which presents the numerical solution and point-wise error for the $\mathcal{F}_2$-exact MFSBP operator, juxtaposed against the test problem from Fig. 4d. As before, we triangulated $\Omega$ by generating $K^2$ equally-sized squares with $K = 10$ and then decomposing each square into two equally-sized triangles. As anticipated, we see that the $\mathcal{F}_3$-exact MFSBP operator yields less accurate results than the trigonometric $\mathcal{F}_2$-exact MFSBP as well as the polynomial $\mathcal{F}_1$-exact MSBP operator. This can be attributed to $\mathcal{F}_3$'s less effective representation of the true underlying solution.

**Remark 8.2.** We found a positive and $\mathcal{F}_3$-exact surface quadrature using seven equidistant points on each linear part. (Again, no nodes were placed on the corners of the triangle to avoid undefined normal vectors.) Observe that $\hat{a}_x(\mathcal{F}_2^3) = \hat{a}_y(\mathcal{F}_2^3)$ and it therefore suffices to additionally find a positive $\hat{a}_x(\mathcal{F}_3^3)$-exact volume quadrature to get an MFSBP operator. To find such a volume quadrature, we used 9 Halton-distributed points in the triangle’s interior.
8.3. The inhomogeneous linear advection equation

We next consider the IBVP for the inhomogeneous linear advection equation with constant coefficients,

\[
\begin{align*}
\partial_t u + a \partial_x u + b \partial_y u &= s(u, x, y), \quad \forall (x, y) \in \Omega, \ t > 0, \\
u(0, x, y) &= u_0(x, y), \quad \forall (x, y) \in \Omega, \\
u(t, x, y) &= g(t, x, y), \quad \forall (x, y) \in \Gamma_-, \ t \geq 0,
\end{align*}
\]

(8.8)

with source term \(s(u, x, y) = \omega \cos(\omega x) \sin(\omega y) + \omega \sin(\omega x) \cos(\omega y) + \sin(\omega x) \sin(\omega y) - u\). We consider (8.8) on \(\Omega = [0, 1]^2\) and with constant velocity coefficients \(a = b = 1\). The generated grid was again constructed by first partitioning \(\Omega\) into \(K^2\) equally-sized squares before decomposing each square into two equally-sized triangles. We used zero initial and inflow boundary conditions, \(u_0 \equiv 0\) and \(g \equiv 0\). For these choices, (8.8) admits the trigonometric steady state solution \(u(x, y) = \sin(\omega x) \sin(\omega y)\). We solved (8.8) again using the MFSBP semi-discretization (8.2). This time, we compare the polynomial \(F_1\)-exact MSBP operator (see (7.1) for the definition of \(F_1\)) with a trigonometric \(F_4\)-exact MFSBP operator in the triangular domain (7.2). Here, \(F_4\) is a seven-dimensional mixed polynomial-trigonometric function space defined as

\[
F_4 = \text{span} \{1, x, y, \sin(\omega x) \sin(\omega y), \cos(\omega x) \cos(\omega y), \sin(\omega x) \cos(\omega y), \cos(\omega x) \sin(\omega y)\},
\]

(8.9)

where we set \(\omega = 2\pi\) in our numerical tests. For the triangle, we found a positive and \(F_4^2\)-exact surface quadrature using 12 equidistant points on each linear part. (Again, no nodes were placed on the corners of the triangle to avoid undefined normal vectors.) Observe that \(\delta_h(F_4^2) = \delta_h(F_3^2)\) and it therefore suffices to additionally find a positive \(\delta_h(F_4^2)\)-exact volume quadrature to get an MFSBP operator. To find such a volume quadrature, we used 24 Halton-distributed points in the triangle’s interior.

Fig. 7 illustrates the numerical solutions and their point-wise errors at time \(t = 2\) for the polynomial \(F_1\)-exact and trigonometric \(F_4\)-exact MFSBP operators. Although the polynomial space \(F_1\) is ten-dimensional and the trigonometric space \(F_4\) is seven-dimensional, we see in Fig. 7 that the trigonometric MSFBP operator yields a significantly more accurate numerical solution. We triangulated \(\Omega\) by generating \(K^2\) equally-sized squares with \(K = 10\) and then decomposing each square into two equally-sized triangles.

We again follow up with a brief error analysis. To this end, we performed the same test for the IBVP (8.8) with different numbers of triangular blocks. Fig. 8 reports on the corresponding errors of the polynomial \(F_1\)-exact and trigonometric \(F_4\)-exact MFSBP operators. As before, the polynomial \(F_1\)-exact MSBP operator yields third- to fourth-order convergence. At the same time, in all cases, the trigonometric \(F_4\)-exact MFSBP operator yields errors around \(10^{-10}\), which change only slightly as the number of blocks increases. We suspect that the reason for this is that the trigonometric space \(F_4\) can exactly represent the exact steady-state solution. Hence, we obtain the exact solution module round-off errors using the \(F_4\)-exact MFSBP operator. Increasing the number of blocks does not significantly decrease these round-off errors. Similar results were reported in [78] for DG methods based on non-polynomial approximation spaces—although without the SBP property and the resulting discrete stability of the scheme.

8.4. Advection on a circular disk

We end this section by demonstrating the geometric flexibility of MFSBP operators. To this end, we consider the IBVP (8.1) for the linear advection equation from subsection 8.1 with constant velocities \(a = b = 1\) on the circular disk

\[
\Omega = \left\{ \mathbf{x} \in \mathbb{R}^2 \ | \ ||\mathbf{x} - \mathbf{c}|| \leq r \right\}
\]

(8.10)

with center \(\mathbf{c} = [1/2, 1/2]^T\) and radius \(r = 1/2\). Also see Fig. 3. We compare the polynomial \(F_1\)-exact and trigonometric \(F_2\)-exact MFSBP operators already discussed in subsection 7.2. Fig. 9 illustrates the reference solution and the point-wise errors numerical solutions at time \(t = 1\) for the polynomial \(F_1\)-exact and trigonometric \(F_2\)-exact MFSBP operators. Once again, the trigonometric MSFBP operator yields a significantly more accurate numerical solution than the polynomial MSBP operator.

9. Concluding thoughts

We introduced the concept of diagonal-norm MFSBP operators, thereby generalizing existing one-dimensional FSBP operators [36] to multi-dimensional geometries and polynomial MSBP operators [43] to general function spaces. Our new MFSBP operators allow us to systematically develop stable and high-order accurate numerical methods that adapt to the characteristic behavior of the underlying solution. In this first paper, we focused on first establishing their theoretical foundation, including their mimetic properties and existence, and provided a general construction procedure. In particular, we showed that most mimetic properties of polynomial SBP operators carry over to the more general class of MFSBP operators. We stress that for MFSBP operators to mimic conservation, the associated approximation space needs to include constants, which polynomial MSBP operators satisfy by construction.
Fig. 7. Numerical solutions and their point-wise errors of the IBVP (8.8) at $t = 2$ using the polynomial $\mathcal{F}_1$-exact and trigonometric $\mathcal{F}_4$-exact MFSBP operators. Although $\mathcal{F}_1$ is ten-dimensional and $\mathcal{F}_4$ is seven-dimensional, the trigonometric $\mathcal{F}_4$-exact MFSBP operator yields significantly more accurate results. We triangulated $\Omega$ by generating $K^2$ equally-sized squares with $K = 10$ and then decomposing each square into two equally-sized triangles. The contour plots do not cover the corners of the computational domain due to the used plotting routine.

Fig. 8. Errors of the numerical solutions to the IBVP (8.8) at $t = 2$ using the polynomial $\mathcal{F}_1$-exact (plusses along the dotted lines) and trigonometric $\mathcal{F}_4$-exact (dots along the straight line) MFSBP operators for an increasing number of triangular blocks. The trigonometric MFSBP operator yields significantly smaller errors than the polynomial one.
Furthermore, we connected the existence of MFSBP operators and positive quadratures that are exact for certain, in general, non-polynomial function spaces. Building upon the theoretical existence investigation, we derived a general construction procedure for MFSBP operators. An essential part of this procedure is that a positive and $\partial_\xi (F^2)$-exact quadrature must exist if an $F$-exact MFSBP operators approximating the directional derivative $\partial_\xi$ is desired. While the existence and construction of such quadratures were established in [32] using a simple least squares approach, the resulting quadratures are not necessarily optimal in the sense that the number of quadrature points is notably larger than the dimension of the function space for which they are exact. Constructing optimal (so-called Gaussian) quadratures for general multi-dimensional function spaces is an open problem. We will further investigate the optimization and efficiency of our MFSBP operators. However, such efforts need to be tailored to specific function spaces and will be carried out in future work.

While we demonstrated the advantage of using MFSBP operators to solve numerical PDEs for different linear problems, where prior knowledge of which approximation space should be used is readily available, a more exhaustive numerical study will be provided in future work. Future research efforts also include investigating the CFL limits and dispersion properties of MFSBP operators (similar to [27]), strategies for adaptively changing the approximation space (see [78]), stabilization techniques, such as split forms [75,17,29], artificial dissipation [56,65,64], and other shock-capturing procedures, in combination with their application to discontinuous problems.

CRediT authorship contribution statement

Jan Glaubitz: Conceptualization, Writing – Original Draft, Review & Editing, Project administration
Simon-Christian Klein: Conceptualization, Writing – Original Draft, Review & Editing, Software, Visualization
Jan Nordström: Conceptualization, Writing – Original Draft, Review & Editing
Philipp Öffner: Conceptualization, Writing – Original Draft, Review & Editing

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

A link to Github with the codes used is given in the paper.

Acknowledgements

This research was supported by AFOSR #F9550-18-1-0316, the US DOD (ONR MURI) grant #N00014-20-1-2595, the US DOE (SciDAC program) grant #DE-SC0012704, Vetenskapsrådet Sweden grant 2018-05084 VR and 2021-05484, the Swedish e-Science Research Center (SeRC), and the Gutenberg Research College, JGU Mainz. Furthermore, it was supported through the program “Oberwolfach Research Fellows” (#2235p) by the Mathematisches Forschungsinstitut Oberwolfach in 2022. We also thank Maximilian Winkler for helpful discussions on the POCS algorithm.

Appendix A. The Projection Onto Convex Sets algorithm

The Projection Onto Convex Sets (POCS) algorithm, sometimes also called the alternating projection algorithm, is a method to find a point in the intersection of two closed convex sets. Given the two overlapping convex sets $C$ and $D$, a point $x_0$ in $C$, and two projections $P_C$ and $P_D$ on $C$ and $D$, the Iteration is:

$$x_{n+1} = P_D(P_C(x_n))$$

for $n = 0, 1, 2, \ldots$. This algorithm is used, for example, in signal processing for image reconstruction.
the POCs algorithm finds a point \( x \in C \cap D \) by alternatingly projecting onto the sets \( C \) and \( D \). Given an arbitrary initial point \( x_0 \), the POCs algorithm produces a sequence of points \( (x_k)_{k \in \mathbb{N}} \) with

\[
x_{k+1} = \text{proj}_D \left( \text{proj}_C (x_k) \right),
\]

where \( \text{proj}_C \) and \( \text{proj}_D \) are projections onto \( C \) and \( D \), respectively. Fig. 1 illustrates the POCs algorithm for the case of \( C \) and \( D \) being (affine) linear spaces. See [76,40,21] for more details on the POCs algorithm.

**A.1. The POCs algorithm and surface quadratures**

In subsection 5.1, we pointed out the necessity of finding positive and \( \mathcal{F}^2 \)-exact surface quadrature on each smooth boundary part \( \Gamma_j \), \( j = 1, \ldots, J \), to then construct a mimetic boundary operator. We can find such quadratures using the POCs algorithm. To this end, let \( \{f_i\}_{i=1}^L \) be a basis of \( \mathcal{F}^2 \). The exactness conditions (4.6) can then be formulated as a linear system for the weight vector \( v = [v_1, \ldots, v_M]^T \),

\[
Av = b,
\]

where the coefficient matrix \( A \) and the right-hand side vector \( b \) are given by

\[
A = \begin{bmatrix}
(\xi \cdot n(x_1^{(j)}) f_1(x_1^{(j)}) & \ldots & (\xi \cdot n(x_M^{(j)}) f_1(x_M^{(j)})) \\
\vdots & \ddots & \vdots \\
(\xi \cdot n(x_1^{(j)}) f_1(x_1^{(j)}) & \ldots & (\xi \cdot n(x_M^{(j)}) f_1(x_M^{(j)}))
\end{bmatrix}, \quad
b = \begin{bmatrix}
(I^{(1)})_{1} f_1 \\
\vdots \\
(I^{(L)})_{1} f_1
\end{bmatrix}.
\]

If \( M > L \) and \( A \) has linearly independent rows, then (A.2) has infinitely many solutions, which form an affine linear subspace of \( \mathbb{R}^M \). In particular, the solution space is a closed convex set and can therefore be used as \( C \) in the POCs algorithm, i.e.,

\[
C = \left\{ v \in \mathbb{R}^M \mid Av = b \right\}.
\]

The second restriction on the surface quadrature is positivity, i.e., \( v_m > 0 \) for all \( m = 1, \ldots, M \). This motivates us to use

\[
D = \left\{ v \in \mathbb{R}^M \mid v_m \geq 0, \ m = 1, \ldots, M \right\}
\]

as the second closed convex set in the POCs algorithm. We restrict the \( v_m \)'s to be non-negative instead of positive to ensure that \( D \) is a closed convex set. While this does only ensure that the surface quadrature is non-negative, we can go over to a positive quadrature by removing all zero weights and the corresponding points. The orthogonal projections onto \( C \) and \( D \) are

\[
\text{proj}_C(v) = v + A^+(b - Av),
\]

\[
[\text{proj}_D(v)]_m = \max(v_m, 0), \quad m = 1, \ldots, M,
\]

where \([\text{proj}_D(v)]_m\) denotes the \( m \)-th component of \( \text{proj}_D(v) \in \mathbb{R}^M \) and \( A^+ \) is the Moore–Penrose pseudo-inverse of \( A \). Since \( A \) has linearly independent rows, \( AA^T \) is invertible and \( A^+ = A^T(AA^T)^{-1} \) constitutes a left inverse. See [44,21] for more details.

**Remark A.1.** It was proved in [31,32] that a positive and \( \mathcal{F}^2 \)-exact least-squares quadrature is found whenever (i) \( \mathcal{F}^2 \) includes constants and (ii) sufficiently many points are used. This implies that, under the same assumptions, the intersection of the closed and convex sets \( C \) and \( D \) is non-empty, and the POCs algorithm is ensured to converge to the desired weights of a positive and \( \mathcal{F}^2 \)-exact surface quadrature.

**Remark A.2.** If a basis of \( \mathcal{F}^2 \) is not readily available, the linear system (A.2), encoding \( \mathcal{F}^2 \)-exactness of the desired surface quadrature, can also be formulated using a generating set of \( \mathcal{F}^2 \). If \( \{f_k\}_{k=1}^K \) is a basis of \( \mathcal{F} \), then \( \{g_k\}_{k=1}^K \) with \( g_{i+K(j-1)} = f_j f_i, \ i, j = 1, \ldots, K \), is a natural generating set. The price we have to pay for the gained simplicity is that \( \{g_k\}_{k=1}^K \) contains more elements than the basis of \( \mathcal{F}^2 \), which makes the linear system (A.2) larger and therefore computationally more expensive to solve. However, in our implementation, we observed the additional costs as insignificant compared to the overall computational costs.
A.2. The POCS algorithm and volume quadratures

As described in subsection 5.2, we need to find a positive and \( \partial_\xi (\mathcal{F}^2) \)-exact volume quadrature before we can construct an FSBP operator. To this end, we proceed similarly as in Appendix A.1 and use the POCS algorithm. Given a basis \( \{ f_i \}_{i=1}^K \) of \( \partial_\xi (\mathcal{F}^2) \), the exactness conditions (4.2) can again be formulated as the linear system

\[
Aw = b
\]  
(A.7)

for the weight vector \( w = [w_1, \ldots, w_N]^T \), where

\[
A = \begin{bmatrix} f_1(\mathbf{x}_1) & \cdots & f_1(\mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ f_L(\mathbf{x}_1) & \cdots & f_L(\mathbf{x}_N) \end{bmatrix}, \quad b = \begin{bmatrix} I[f_1] \\ \vdots \\ I[f_L] \end{bmatrix}.
\]  
(A.8)

Like before, this yields the closed convex set

\[
C = \left\{ w \in \mathbb{R}^N \mid Aw = b \right\}.
\]  
(A.9)

The second restriction on the volume quadrature is positivity, i.e., \( w_n > 0 \) for all \( n = 1, \ldots, N \). However, because of conditioning considerations, we further restrict the quadrature weights to lie above a certain threshold \( w_{\text{min}} \) and choose

\[
D = \left\{ w \in \mathbb{R}^N \mid w_n \geq w_{\text{min}}, \ n = 1, \ldots, N \right\}.
\]  
(A.10)

The motivation for introducing the threshold \( w_{\text{min}} \) is that it bounds the condition number of the inverse diagonal norm matrix by \( \| P^{-1} \| \leq 1/w_{\text{min}} \). Without such a threshold, a single small weight \( w_n \approx 0 \) could yield the inverse norm matrix to have an undesirable high condition number, which might translate into an ill-conditioned MFSBP operator \( D_\xi = P^{-1}Q_\xi \).

In our implementation, we choose \( w_{\text{min}} = \frac{1}{10N} \), although we do not claim the optimality of this choice. The orthogonal projections onto \( C \) and \( D \) are given by

\[
\text{proj}_C(w) = w + A^+ (b - Aw), \\
\text{proj}_D(w) = \max(w_n, w_{\text{min}}), \quad n = 1, \ldots, N.
\]  
(A.11)

Similar arguments as in Remark A.1 ensure that the intersection of \( C \) and \( D \) is non-empty and that the POCS algorithm converges to a positive and \( \partial_\xi (\mathcal{F}^2) \)-exact volume quadrature.

Remark A.3. Following up on the discussion in Remark A.2, the linear system (A.7) can also be formulated using a generating set of \( \partial_\xi (\mathcal{F}^2) \), if no basis is readily available. If \( \{ f_k \}_{k=1}^K \) is a basis of \( \mathcal{F} \), then \( \{ g_{i+k(j-1)} \}_{i=1}^K \) with \( g_{i+k(j-1)} = \partial_\xi (f_i f_j) \), \( i, j = 1, \ldots, K \), spans \( \partial_\xi (\mathcal{F}^2) \).

A.3. Using POCS for constructing \( Q_A \)

We now comment on some computational details for constructing the anti-symmetric matrix \( Q_A \) discussed in subsection 5.3, which has to satisfy the exactness condition (5.6). We can determine such a matrix \( Q_A \) using the POCS algorithm. To this end, note that the exactness condition (5.6) and the restriction to anti-symmetric \( Q_A \)'s (\( Q_A^\top = -Q_A \)) define nonempty closed convex sets,

\[
C = \left\{ Q_A \in \mathbb{R}^{N \times N} \mid Q_A V = PV_\xi - \frac{1}{2} B_\xi V \right\},
\]  
(A.12)

\[
D = \left\{ Q_A \in \mathbb{R}^{N \times N} \mid Q_A + Q_A^\top = 0 \right\},
\]  
(A.12)

in the space of \( N \times N \) matrices. If the intersection between the two convex sets \( C \) and \( D \) is non-empty, then the POCS algorithm finds a point in the intersection by alternatingly projecting onto \( C \) and \( D \). The orthogonal projections onto the sets in (A.12) are given by

\[
\text{proj}_C(Q_A) = Q_A + \left( PV_\xi - \frac{1}{2} B_\xi V - Q_A V \right) V^+,
\]

\[
\text{proj}_D(Q_A) = \left( Q_A - Q_A^\top \right) /2,
\]  
(A.13)

respectively. Here, \( V^+ \) is the Moore–Penrose pseudo-inverse of \( V \). Since \( V \) has linearly independent columns, \( V^T V \) is invertible and \( V^+ = (V^T V)^{-1} V^T \) constitutes a right inverse. See [44,21] for more details.
A.4. Robust implementation

In some cases, the coefficient matrix $A$ in the exactness conditions $Aw = b$, see (A.2) and (A.7), was observed to be ill-conditioned. We formulated and solved the exactness conditions for a reduced discrete orthonormal basis in such cases. Starting from a potentially over-complete spanning set $\{g_i\}_{i=1}^L$ of a function space $\mathcal{G} = \mathcal{F}^2$ or $\mathcal{G} = \partial_k (\mathcal{F}^2)$, we will now describe an algorithm that computes an orthonormal basis spanning approximately the same space. We start by computing the coefficient matrix $A$ in the exactness conditions (A.2) and (A.7) using the original spanning set $\{g_i\}_{i=1}^L$. We then determine the singular value decomposition (SVD) of $A$. Recall that the SVD of $A \in \mathbb{R}^{L \times N}$ with $L \leq N$ is given by

$$A = U \Sigma V^T,$$

where $U \in \mathbb{R}^{L \times L}$ and $V \in \mathbb{R}^{N \times N}$ are orthogonal matrices and $\Sigma \in \mathbb{R}^{L \times N}$ is a rectangular diagonal matrix with non-negative singular values $\sigma_1 \geq \cdots \geq \sigma_L$ on the diagonal. Let $\varepsilon > 0$ be a fixed threshold that remains to be determined. We can now “compress” the original set $\{g_i\}_{i=1}^L$ in the sense that a sizeable rank-deficient collection of these functions is replaced with an orthonormal basis $\{h_m\}_{m=1}^M$ whose dimension is the numerical rank of the spanning set up to precision $\varepsilon$. The basis is orthonormal in the sense that $h_m^T h_n$ is equal to one if $m = n$ and zero otherwise. We get the orthonormal basis functions as

$$h_m(x) = \frac{1}{\sigma_m} \sum_{i=1}^N u_{i,m} g_i(x), \quad m = 1, \ldots, M,$$

where $M$ is the largest integer such that $\sigma_M \geq \varepsilon$. If $A$ was invertible and $\varepsilon = 0$, this construction is equivalent to multiplying both sides of the SVD $A = U \Sigma V^T$ from the left-hand side with $\Sigma^{-1}U^T$, yields

$$V^T = \Sigma^{-1}U^T A.$$

The functions $h_1, \ldots, h_M$ constitute an orthonormal basis for the span of the input functions $g_1, \ldots, g_L$ to precision $\varepsilon$. Since the basis is orthonormal, the condition number of the resulting coefficient matrix $A_r$ is equal to one. The new basis can be used to calculate a new corresponding right side $b_r$ and therefore allows us to represent the exactness constraints in a stable manner. A crucial part is selecting the threshold $\varepsilon$. Too large values discard too many dimensions and lead to a matrix $A_r$ that describes fewer exactness conditions than the original one. At the same time, too small values inflate the norm of $b_r$. In our implementation, we solve $A_r w = b_r$ for $\varepsilon \in 10^{-16}, \ldots, 10^{-5}$ and ultimately select the resulting weight vector $w$, that minimizes $\|A_r w - b\|_2$, i.e., the weights that best-fit the original exactness conditions.

References