An algorithmic exploration of the existence of high-order summation by parts operators with diagonal norm

Nathan Albin^{a,*}, Joshua Klarmann^a

^aDepartment of Mathematics, Kansas State University, 138 Cardwell Hall, Manhattan, KS 66506

Abstract

This paper explores a common class of diagonal-norm summation by parts (SBP) operators found in the literature, which can be parameterized by an integer triple (s, t, r) representing the interior order of accuracy (2s), the boundary order of accuracy (t), and the dimension of the boundary closure (r). There is no simple formula for determining whether or not an SBP operator exists for a given triple of parameters. Instead, one must check that certain compatibility conditions are met: namely that a particular linear system of equations has a positive solution. Partly because of the complexity involved, not much is known about diagonal-norm SBP operators with 2s > 10.

By utilizing a new algorithm for answering the question "Does an SBP operator exist for the parameters (s, t, r)?", it is possible to explore the existence of SBP operators with high order accuracy, and previously unknown SBP operators with interior order of accuracy as large as 2s = 30 are found. Additionally, a method for optimizing the spectral radius of the SBP derivative is introduced, and the effectiveness of this method is explored through numerical experiment.

Keywords: high-order finite difference methods; summation by parts; diagonal energy norm

1. Introduction

The need for high-order numerical methods in the simulation of long-distance advection and wave propagation is well established. The introduction to a 1972 paper by Kreiss and Oliger [8], for example, provides a review of works dating back to the mid-1960's indicating that the inadequacy of second-order methods for applications in meteorology and oceanography was becoming apparent even then. The argument is based on the observation that the error in a numerical solution of a hyperbolic partial differential equation (PDE) scales roughly as a constant times $T\omega^{(p+1)}h^p$, where p is the order of accuracy of the numerical derivative operator, h is the spatial step size, ω is a characteristic frequency of the solution, and T is the final time to which the equation is to be solved. Thus, in order to maintain a given level of error, the spatial step size should be scaled as $h \sim \omega^{-(p+1)/p} T^{-1/p}$. For problems wherein ω and T are very large, it is essential that p be large as well, to avoid the need for overly small h.

For spatial derivative approximations based on finite differences, the combination of wide stencils to allow high-order accuracy and biased stencils to accommodate domain boundaries presents a particular challenge; it is very unlikely that an arbitrarily chosen high-order finite difference operator will lead to a stable solver for a hyperbolic initial-boundary value problem. Because of this, much research has been conducted in the search for stable and accurate finite difference schemes.

1.1. Summation by parts operators

Among the various types of high-order finite difference operators, the summation by parts (SBP) operators are unique in that their construction incorporates the construction of a natural discrete energy norm

*Corresponding author

Email addresses: albin@math.ksu.edu (Nathan Albin), jklarm@gmail.com (Joshua Klarmann)

that can be used to prove stability for PDE solvers. Numerical schemes based on SBP operators have proven effective in simulating a wide variety of physical phenomena, including fluid flow [13, 18, 19], elastic wave propagation [2, 12, 15], and orbiting binary black holes [14].

The basic idea behind SBP operators (see, e.g., References [4, 9, 16]) is straightforward. One seeks to build, simultaneously, a finite difference operator and an associated vector norm that mimic, in a semidiscrete setting, some continuum energy estimate for the PDE. The one-dimensional advection equation on a bounded interval provides the canonical example.

Consider the PDE

$$u_t + u_x = 0$$
 $x \in (0, 1), \quad t > 0,$ (1)

with suitable initial and boundary conditions. The energy

$$\mathcal{E}_u(t) = \|u(t,\cdot)\|_{L^2}^2 = \int_0^1 u(t,x)^2 \, dx,$$

has the property that, for u solving Equation (1), \mathcal{E}_u satisfies

$$\frac{d\mathcal{E}_u}{dt} = 2\int_0^1 u \, u_t \, dx = -2\int_0^1 u \, u_x \, dx = -\int_0^1 \frac{\partial}{\partial x} u^2 \, dx = u(t,0)^2 - u(t,1)^2.$$
(2)

Now, consider the following semi-discrete form of Equation (1). In what follows, for the sake of simplifying formulas, we deviate from convention and index arrays beginning at 0. Let $\{x_i\}_{i=0}^{n-1}$ be the grid of n equispaced nodes in [0, 1] with step size h = 1/(n-1), and let v(t) be the *n*-vector approximating u in the method-of-lines interpretation. That is, $v_i(t) \approx u(t, x_i)$ solves the semi-discrete equation

$$v_t + D_h v = 0, (3)$$

for some $n \times n$ finite difference operator D_h . Emulating the continuum case, let P_h be an $n \times n$ symmetric positive definite matrix, and define the energy

$$E_v(t) = \|v(t)\|_{P_h}^2 = v(t)^T P_h v(t).$$

If P_h and D_h together satisfy the condition

$$P_h D_h + D_h^T P_h = e_{n-1} e_{n-1}^T - e_0 e_0^T = Q,$$
(4)

with $\{e_i\}_{i=0}^{n-1}$ the canonical basis in \mathbb{R}^n , then it is straightforward to check that if v is a solution to Equation (3), then the energy satisfies

$$\frac{dE_v}{dt} = v_t^T P_h v + v^T P_h v_t = -v^T \left(P_h D_h + D_h^T P_h \right) v = v_0^2 - v_{n-1}^2$$

which is a semi-discrete analog of Equation (2). This property can be used to prove the stability of the fully discrete numerical solver.

Thus, the construction of an SBP first derivative operator (actually, an operator/norm pair) consists of constructing $n \times n$ matrices D_h and P_h with the following properties.

- (P1) D_h is a finite difference approximation of the first derivative.
- (P2) P_h and D_h together satisfy the energy condition (4).
- (P3) P_h is a positive definite matrix.

1.2. The parameters (s, t, r)

Making use of the scale-invariance of the derivative and norm, namely that $D_h = \frac{1}{h}D_1$ and $P_h = hP_1$, we now drop the subscript h and assume a step size of h = 1. In their general form, properties (P1)–(P3) form a large nonlinear system of equations and inequalities in the n^2 entries in D and the n(n+1)/2 entries in P. This is undesirable for three reasons: the equations are nonlinear, there are many of them, and their solution naturally depends on the grid size n.

Fortunately, each of these problems can be treated by standard methods [9, 20]. Let the positive integer triple (s, t, r) be given. For the remainder of the paper, P is assumed to have the block diagonal form

$$P = \begin{bmatrix} \tilde{P} & 0 & 0\\ 0 & I & 0\\ 0 & 0 & \tilde{P} \end{bmatrix} = P^T,$$
(5)

with \tilde{P} and \hat{P} symmetric positive definite $r \times r$ matrices. Furthermore, D is assumed to be a centered finite difference operator of order 2s in its interior n - 2r rows, and a finite difference operator of order t in its first r and last r rows. Under these assumptions, D can be factored as

$$D = \begin{bmatrix} \tilde{P}^{-1} & 0 & 0\\ 0 & I & 0\\ 0 & 0 & \hat{P}^{-1} \end{bmatrix} \begin{bmatrix} B & C_0 & 0\\ -C_0^T & \tilde{D} & -\hat{C}_0^T\\ 0 & \hat{C}_0 & \hat{B} \end{bmatrix},$$
(6)

where B and \hat{B} are $r \times r$ block matrices, C_0 and \hat{C}_0 are $r \times (n-2r)$ blocks of the forms

$$C_0 = \begin{bmatrix} C & 0 \end{bmatrix}$$
 and $\hat{C}_0 = \begin{bmatrix} 0 & \hat{C} \end{bmatrix}$

respectively, where $C, \hat{C} \in \mathbb{R}^{r \times s}$. For example, when s = 2 and r = 4, the matrix on the right-hand side of Equation (6) has a banded structure of the form

where the shaded block is the matrix C. Since the interior n - 2r rows of D are known, the remaining unknown quantities lie within \tilde{P} , \hat{P} , B and \hat{B} .

Remark 1.1. There are, naturally, a wide variety of modifications that can be made to this basic structure. For example, there is no need to require that the left and right closures are identical in size and order, nor does the interior method need to be a centered finite difference method. Moreover, the underlying computational grid need not be uniformly spaced [11], and the a much more general SBP framework has been developed [6]. Despite the wide variety of modifications available, however, it is still interesting to consider under what circumstances the conventional SBP first derivative operator can exist.

1.3. The question of existence

The main question of this paper can now be stated as follows.

Question 1. Let the positive integer triple (s, t, r) be given. Does there exist an SBP pair, P and D, satisfying (P1)–(P3) such that P has the form given in (5) and D has the form given in (6) with order of accuracy t in its first r and last r rows and order of accuracy 2s in its remaining interior rows?

We will say that an SBP operator for the triple (s, t, r) exists if the answer to Question 1 is affirmative, and that no such SBP operator exists if the answer is negative. The principal contributions of the present work are the following.

- We derive a set of compatibility conditions (similar to the conditions of Kreiss and Scherer [9]) on the triple (s, t, r) that are necessary and sufficient for the existence of an SBP operator (Section 2). These conditions decouple the problem of constructing SBP operators into a two-step process: first the norm P is constructed, if possible, and then the derivative matrix D is constructed. Compatibility conditions are given for the general block-norm setting (Theorem 1) and are specialized to the diagonal-norm setting (Theorem 2).
- Focusing on the operators with diagonal norm, we describe a deterministic algorithm for answering the existence question (Section 3). The algorithm is quite simple, comprising the solution of a linear system of equations, followed by the solution of a standard linear program.
- Next, we report the results of an automated search of the (s, t, r)-space, showing the existence of diagonal-norm SBP operators with orders of accuracy as large as 2s = 30 in the interior and t = 15 on the boundary. (The diagonal-norm operators of highest order accuracy in the literature are 2s = 10 and t = 5.)
- We follow this exploration with the description of a new algorithm for optimizing the SBP derivative operator (Section 5) and demonstrate the effectiveness of some newly constructed, high-order SBP operators by numerical experiment (Section 6).

2. Compatibility conditions

In this section, we present a derivation of necessary and sufficient compatibility conditions (similar to those given in [9]) designed explicitly to allow for the cases t < s and r > 2s. It is sufficient to treat the top rows of D; the bottom rows are treated analogously. Consider the three matrices $X \in \mathbb{R}^{r \times (t+1)}$, $\tilde{X} \in \mathbb{R}^{s \times (t+1)}$ and $Y \in \mathbb{R}^{r \times (t+1)}$ defined as

$$X_{ij} = i^j, \qquad \tilde{X}_{ij} = (r+i)^j, \qquad \text{and} \qquad Y_{ij} = ji^{j-1},$$

where, for convenience in dealing with the upper-left entries X_{00} and Y_{00} in what follows, we use the notation

$$0^0 = 1$$
 and $0 \cdot 0^{-1} = 0$.

(This is only a notational convenience to avoid the need to consider special cases in what follows. These conventions are never treated as computationally valid.)

With these definitions, D is a tth order derivative approximation in the first r rows if and only if

$$\tilde{P}^{-1}BX + \tilde{P}^{-1}C\tilde{X} = Y$$
 or equivalently $BX + C\tilde{X} = \tilde{P}Y$.

Moreover, D satisfies the energy condition (4) if and only if

$$B + B^T = -e_0 e_0^T, \qquad \hat{B} + \hat{B}^T = e_{r-1} e_{r-1}^T.$$

Splitting B into its symmetric and antisymmetric parts yields

$$B = B_1 + B_2, \quad B_1 = \frac{1}{2}(B + B^T) = -\frac{1}{2}e_0e_0^T, \qquad B_2 = \frac{1}{2}(B - B^T) = -B_2^T.$$

Thus, the equation for B and P can be written as

$$\frac{1}{2}(B - B^T)X = B_2 X = \tilde{P}Y - C\tilde{X} - B_1 X.$$
(7)

The form of Equation (7) suggests a solution strategy. First, determine conditions on the norm P—the compatibility conditions—which guarantee solvability of Equation (7) for B_2 . If such a norm exists, then necessarily a corresponding SBP derivative operator must exist. Conversely, if Equation (7) is not solvable

for any choice of norm \tilde{P} , then we may conclude that no SBP operator/norm pair exists for the given parameters.

Let $\mathcal{L} : \mathbb{R}^{r \times r} \to \mathbb{R}^{r \times (t+1)}$ be the linear operator with action $\mathcal{L}B = \frac{1}{2}(BX - B^TX)$. Then the standard solvability theorem of linear algebra states that Equation (7) can be solved if and only if the right-hand side is orthogonal to the nullspace of \mathcal{L}^* (with orthogonality and the adjoint defined in some inner product). It is convenient to use the standard inner product on $\mathbb{R}^{m \times n}$:

$$\langle U, V \rangle = \operatorname{trace}(UV^T) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} U_{ij} V_{ij}$$

Keeping in mind two fundamental properties of the matrix trace, namely that

$$\operatorname{trace}(U) = \operatorname{trace}(U^T)$$
 and $\operatorname{trace}(UV) = \operatorname{trace}(VU)$

whenever the dimensions of U and V are compatible with both products, we can compute the adjoint of \mathcal{L} as follows.

$$2 \langle \mathcal{L}B, A \rangle = \operatorname{trace}(BXA^T) - \operatorname{trace}(B^TXA^T) = \operatorname{trace}(BXA^T) - \operatorname{trace}(AX^TB) \\ = \operatorname{trace}(BXA^T) - \operatorname{trace}(BAX^T) = \operatorname{trace}(B(AX^T - XA^T)^T) = \langle B, AX^T - XA^T \rangle.$$

So $\mathcal{L}^*A = \frac{1}{2}(AX^T - XA^T)$. Thus, we have proved the following lemma.

Lemma 1. Equation (7) is solvable if and only if there exists a \tilde{P} such that $\langle \tilde{P}Y - C\tilde{X} - B_1X, A \rangle = 0$ for every $A \in \mathbb{R}^{r \times (t+1)}$ satisfying $AX^T = XA^T$.

This lemma provides the first form of the compatibility condition. An SBP operator for the parameter triple (s, t, r) exists if and only if one can find a positive definite matrix \tilde{P} such that the right-hand side of Equation (7) is orthogonal to the nullspace $N(\mathcal{L}^*)$ of \mathcal{L}^* , or equivalently, if and only if the right-hand side is orthogonal to each element of a basis for $N(\mathcal{L}^*)$. A basis is given by the following lemma.

Lemma 2. A matrix A satisfies $AX^T = XA^T$ if and only if A = XE for some $E = E^T \in \mathbb{R}^{(t+1)\times(t+1)}$.

Proof. Suppose $AX^T = XA^T$. Since X has full column rank, X^TX is invertible, and, thus,

$$A = AX^{T}X(X^{T}X)^{-1} = XA^{T}X(X^{T}X)^{-1} = XE.$$

To see that E is symmetric, observe that $XEX^T = AX^T = XA^T = XE^TX^T$ and so $X^TXEX^TX = X^TXE^TX^TX$. Since X^TX is invertible, $E = E^T$. The converse is straightforward.

Corollary 1. Equation (7) is solvable if and only if there exists a \tilde{P} such that

$$\left\langle \tilde{P}Y - C\tilde{X} - B_1X, X(e_p e_q^T + e_q e_p^T) \right\rangle = 0 \quad \text{for all} \quad p = 0, 1, \dots, t, \quad q = p, p+1, \dots, t.$$
(8)

Before forming the entire inner product in Equation (8), it is useful to compute a few separate terms. First, observe that for any matrix $Z \in \mathbb{R}^{r \times (t+1)}$

$$\langle Z, Xe_p e_q^T \rangle = \operatorname{trace}(Ze_q(Xe_p)^T) = \operatorname{trace}((Xe_p)^T Ze_q) = (Xe_p)^T Ze_q.$$

Now, considering the first term in the inner product in Equation (8), we find

$$\left\langle \tilde{P}Y, Xe_p e_q^T \right\rangle = (Xe_p)^T (\tilde{P}Ye_q) = \sum_{k=0}^{r-1} X_{kp} \sum_{\ell=0}^{r-1} \tilde{P}_{k\ell} Y_{\ell q} = \sum_{k=0}^{r-1} \sum_{\ell=0}^{r-1} qk^p \ell^{q-1} \tilde{P}_{k\ell}.$$

Turning to the second form in the right-hand side, we find

$$\left\langle C\tilde{X}, Xe_p e_q^T \right\rangle = (Xe_p)^T (C\tilde{X}e_q) = \sum_{k=0}^{r-1} X_{kp} \sum_{\ell=0}^{s-1} C_{k\ell} \tilde{X}_{\ell q} = \sum_{k=0}^{r-1} \sum_{\ell=0}^{s-1} k^p (r+\ell)^q C_{k\ell}.$$

Finally, we find

$$\left\langle B_1 X, X e_p e_q^T \right\rangle = (X e_p)^T (B_1 X e_q) = \sum_{k=0}^{r-1} X_{kp} \sum_{\ell=0}^{r-1} (B_1)_{k\ell} X_{\ell q} = -\frac{1}{2} \sum_{k=0}^{r-1} \sum_{\ell=0}^{r-1} k^p \ell^q \delta_{k0} \delta_{\ell 0} = -\frac{1}{2} \delta_{p0} \delta_{q0}.$$

Combining the above computations, we arrive at a system of (t+1)(t+2)/2 equations for \tilde{P} .

Theorem 1. An SBP operator with parameters (s, t, r) exists if and only if there is a positive definite matrix \tilde{P} satisfying

$$\sum_{k=0}^{r-1} \sum_{\ell=0}^{r-1} \left(qk^p \ell^{q-1} + pk^q \ell^{p-1} \right) \tilde{P}_{k\ell} = \sum_{k=0}^{r-1} \sum_{\ell=0}^{s-1} \left(k^p (r+\ell)^q + k^q (r+\ell)^p \right) C_{k\ell} - \delta_{p0} \delta_{q0} \tag{9}$$

for every p = 0, 1, ..., t and q = p, p + 1, ..., t.

2.1. SBP operators with diagonal norm

The remainder of the paper is restricted to the case that P_h is diagonal, which is quite natural due to the fact that these are the only SBP operators for which standard techniques exist for proving stability for PDEs with variable coefficients or on multi-dimensional curvilinear grids [17]. Although SBP operators with non-diagonal (block) norm have recently been shown to be stabilizable on curvilinear grids [10] by the addition of a tuned artificial damping term, the question of existence of diagonal-norm SBP operators remains an interesting open problem; no such operators with interior order greater than 10 exist in the literature. Under the assumption that P_h is diagonal, Theorem 1 simplifies somewhat.

Theorem 2. An SBP operator with parameters (s,t,r) and diagonal P_h exists if and only if there is a diagonal positive definite matrix \tilde{P} satisfying

$$\sum_{k=0}^{r-1} (p+q)k^{p+q-1}\tilde{P}_{kk} = \sum_{k=0}^{r-1} \sum_{\ell=0}^{s-1} \left(k^p (r+\ell)^q + k^q (r+\ell)^p\right) C_{k\ell} - \delta_{p0}\delta_{q0} \tag{10}$$

for every p = 0, 1, ..., t and q = p, p + 1, ..., t.

Remark 2.1. Since the left-hand side of Equation (10) is zero when p = q = 0, the SBP operator can only exist if $\sum_{k,\ell} C_{k\ell} = \frac{1}{2}$. If a generic row of the centered difference portion of D has the coefficients

$$\begin{bmatrix} \cdots & -\alpha_s & \cdots & \alpha_{-1} & 0 & \alpha_1 & \cdots & \alpha_s & \cdots \end{bmatrix}$$

then

$$\sum_{k,\ell} C_{k\ell} = \sum_{i=1}^{s} i\alpha_i,$$

so the p = q = 0 equation is simply the requirement that the centered difference operator evaluate derivatives of linear functions exactly, which will always be true in the present setting. The remaining t(t+3)/2 equations do involve the r unknowns in \tilde{P} .

Remark 2.2. Although Equation (10) appears to be an overdetermined system if r < t(t+3)/2, it is known (see [9, Theorem 2.1]) that, when r = s = 2t, Equation (10) has a unique solution. Moreover, by [7, Corollary 1], if a norm P exists, then the system must consistent with the requirement that the norm matrix P act as a 2s-order quadrature rule, providing a lower bound on the number of linearly independent equations that must be present for solvability. In this paper, we will not concern ourselves with locating the linearly independent equations since Equation (10) is sufficient for our purposes.

Remark 2.3. It is important that r be allowed to vary independently of s, since it appears (see Section 4.1) that if $s = t \ge 5$ and r = 2s, then the unique solution to Equation (10) is not positive and, therefore, that in general SBP operators do not exist for (s, t, r) = (s, s, 2s).

3. Algorithm for existence

This section describes a method for algorithmically deciding the answer to Question 1 for given parameters (s, t, r). That is, the algorithm presented here determines whether or not such an operator/norm pair exists, but does not completely construct one. The construction, assuming existence is known, is postponed until Section 5. The entire process that follows is performed in exact arithmetic (i.e., using rational numbers) since the solvability of Equation (7) requires that \tilde{P} be an exact solution to Equation (10). If \tilde{P} is only an approximate solution, then Equation (7) is not solvable. Of course, there should be approximate solutions to Equation (7) in this case, but the details of a finite precision implementation of this algorithm remains an open question (see Section 7). For the results presented in this paper, the Python library sympy [21] was used for exact arithmetic.

3.1. Solve the linear system

To initialize the algorithm, Equation (10) is put into the form of a matrix-vector equation Ax = b with $A = [(t+1)(t+2)/2] \times r$ matrix and x the vector of unknowns \tilde{P}_{kk} . This can be done in exact arithmetic by Gaussian elimination. As stated in Remarks 2.2 and 2.3, if r = 2s = 2t, Equation (10) has a unique solution. In this case, the answer to Question 1 is immediate. If x is positive, an SBP operator exists for the given parameters. If x has any non-positive entries, no SBP operator exists. This is exactly Theorem 2.1 of [9]. For other choices of (s, t, r), however, there is typically a solution manifold of the form

$$x = x_0 + Gy,$$

where G is $r \times v$, with v the number of degrees of freedom in the solution. In this case, more work is required to determine if there is a solution with positive entries.

3.2. Solve the LP problem

If the previous steps produced a manifold of solutions to the linear system in Equation (10), Question 1 is equivalent to asking whether there exists a $y \in \mathbb{R}^v$ such that $x_0 + Gy$ has all positive entries. To see how this problem can be solved algorithmically, consider the following optimization problem

maximize
$$\min_{i} x_i$$
,
subject to $x = x_0 + Gy$, (11)

and note that Equation (10) has a strictly positive solution if and only if the value of the optimization problem is strictly positive. The optimization problem can be algorithmically solved through a common technique that transforms the problem into a standard linear program (LP):

maximize
$$\eta$$
,
subject to $x_i \ge \eta$, $i = 0, 1, \dots, r-1$, (12)
 $x = x_0 + Gy$.

As an LP, Equation (12) can be solved by the simplex method, thus providing an algorithm for solving Equation (11). In this case, we conclude that the SBP operator exists if and only if the solution to Equation (12) is positive. From an implementation perspective, this is the most complex step, as it requires a simplex solver in exact arithmetic. We did not find an existing library for this, and so implemented our own simplex solver in Python.

Remark 3.1. By openness, it is clear that if a manifold of solutions exists and if there is one positive solution, then there are infinitely many positive solutions. In this case, it is not clear which choice of P is "best" in any particular sense. In this paper, we choose the P maximizing Equation (11) as a particular choice. This is not quite arbitrary, as described in Remark 5.1.

s	t	r	dof P_h	dof D_h	$\min_i x_i$
1	1	1	0	0	5.000e-01
2	2	4	0	0	3.541e-01
3	3	6	0	1	3.159e-01
4	4	8	0	3	2.575e-01
5	5	11	1	10	2.077e-01
6	6	14	2	21	9.683 e-03
7	7	19	5	55	1.907 e-01
8	8	23	7	91	4.652 e- 02
9	9	28	10		4.622 e-02
10	10	34	14		8.357e-02
11	11	40	18		3.907 e-02
12	12	47	23		5.286e-02
13	13	54	28		1.933e-02
14	14	62	34		1.863e-02
15	15	71	41		4.559e-02

Table 1: For the case t = s, the table reports the *smallest* value of r for which an SBP operator exists. When P_h is non-unique, the number of degrees of freedom in P_h is reported as "dof P_h ". The value of Equation (11) is reported as $\min_i x_i$. For the optimal P_h , the degrees of freedom of D_h is reported as "dof D_h ". Since the computation of the solution space in D_h is much more computationally taxing than that of P_h , only the cases with $s \leq 8$ include the dimension of the solution space for D_h .

4. Existence and nonexistence of SBP operators

This section presents some novel results based on the algorithm of the previous section. It is worth remarking that, although the algorithm is provably correct, the following computational results rely on computer-assisted proof. The numerators and denominators of the rational numbers involved are sufficiently large that we cannot hope to perform the Gaussian elimination and simplex method steps by hand except in a small number of cases. For example, the value for x_0 in the case s = t = 8, r = 23 is

$x_0 = \frac{83852077150009258297147}{299027329581685985280000}$

Although we have done our best to test our code and to compare with SBP results in the literature, the results presented in this text are nevertheless vulnerable to errors either in the sympy rational number manipulation routines, the Gaussian elimination routine or in the simplex solver. As an example, an earlier version of the code produced incorrect results from time to time due to some unexpected behavior in the symbolic operations of a particular commercial software tool. We have a high degree of confidence in the computational results presented in this paper, but certainly encourage their verification by others.

4.1. The smallest r for t = s

We first consider the case of an SBP operator of order 2s in the interior and t = s on the boundary. It can be readily seen that if a solution exists for a particular choice of (s, t, r), then this is also a solution for (s, t, r') for any r' > r. That is, if the answer to Question 1 is affirmative for a particular choice of parameters, the answer is also affirmative if the finite difference orders are left unchanged and the boundary closure size is increased. Thus, it is possible to perform a bisection search to locate the smallest r for which an SBP operator with the given choice s = t exists. Table 1 presents the results of this parameter sweep. The table also gives the dimension of the solution space of P_h and the value of Equation (11).

As stated previously, the present discussion does not concern the actual construction of an SBP operator, but merely the question of existence. However, once a suitable norm P_h is found, Equation (7) is then guaranteed solvable for the block B of an SBP difference operator D_h . Included in Table 1 is a column titled "dof D_h ", which reports (for some choices of s) the dimension of the solution set for D_h with the choice of P_h described in Remark 3.1. In Section 5, we discuss a method for choosing a particular D_h from this solution set.

s	t	r	dof P_h	$\min_i x_i$
1	1	2	0	5.000e-01
2	2	4	0	3.542 e-01
3	3	6	0	3.159e-01
4	4	8	0	2.575e-01
5	4	10	2	3.367 e-01
6	5	12	2	2.997 e- 01
7	6	14	2	9.682 e- 03
8	6	16	4	2.992e-01
9	6	18	6	3.207 e-01
10	$\overline{7}$	20	6	2.923e-01
11	$\overline{7}$	22	8	3.088e-01
12	8	24	8	2.504 e-01
13	8	26	10	2.980e-01
14	9	28	10	4.622 e- 02
15	9	30	12	2.858e-01

Table 2: For the case r = 2s, the table reports the *largest* value of t for which an SBP operator exists. When P_h is non-unique, the number of degrees of freedom in P_h is reported as "dof P_h ". The value of Equation (11) is reported as min_i x_i .

To our knowledge, the results for s > 5 are unknown in the literature. Of particular interest is the nonlinear dependence of r on s (see Remark 2.3). Based on previous results for the cases s = 2, 3 and 4, it might be expected that SBP operators exist for all r = 2s = 2t (the case considered in [9]). However, by the nature of the parameter sweep conducted here, we conclude that, for a given s, no SBP operator (with t = s) exists for r smaller than the value given in the table. The case s = 5, r = 11 has already been reported (see, e.g., References [5, 10]). However, while a footnote of Reference [5] states that the choice s = 5, r = 10 "did not result in a positive definite norm", no proof or explanation is given.

4.2. The largest t for r = 2s

From the previous results, it is clear that asking for r = 2s = 2t is, in general, too restrictive and, if we require that D_h have its boundary order of accuracy half as large as its interior order of accuracy, the size of the boundary closure must grow faster than 2s. As a second application of the algorithm, we consider the opposite question. Suppose we wish to have r = 2s. What is the largest boundary order of accuracy t for which an SBP operator exists? Table 2 presents the results of such a parameter study. Evidently, it is not difficult to find SBP operators with r = 2s, provided t is allowed to grow more slowly than s.

5. Optimization of the derivative operator

Another interesting observation about the results presented in Table 1 is that the number of degrees of freedom in D_h grows rapidly with increasing s. At the end of the algorithm described in Section 3 we are generally left, not with a single SBP operator, but with an entire linear manifold:

$$D_h = D_0 + \sum_j \xi_j D_j,\tag{13}$$

where j varies through all degrees of freedom. Although the primary concern of the present paper is the *existence* of SBP operators, the question of "Which is best?" is also important. As described in Reference [5], there are a wide range of options for defining "best". Rather than consider each of these, we focus on a particular objective function: minimizing the spectral radius $\rho(D_h)$. This objective function is interesting because it controls the CFL condition for explicit PDEs solvers. Moreover, it is unique among the objective functions considered in the reference in that it is non-convex in the parameters ξ_j of D_h , and therefore difficult to optimize globally. Hence the remark in Reference [5]: "Therefore, when we refer to minimizing



Figure 1: On the left, the spectral radius as a function of 1/h for the optimized s = t = 6, r = 14 SBP operator. On the right, the spectral radii as functions of 1/h for the optimized s = t = 6, r = 15 and s = t = 7, r = 19 SBP operators.

the spectral radius, we perform a numerical minimization and do not claim that we have actually found a global minimum."

In this paper, we suggest an alternative to minimizing the spectral radius directly. The key point is that, although $\rho(C)$ is not a convex function in the entries of C in general, it is convex for normal matrices C. And, although D_h is not a normal matrix in general, it is close in some sense to a normal matrix, because

$$\left(P_h D_h - \frac{1}{2}Q\right) + \left(P_h D_h - \frac{1}{2}Q\right)^T = 0$$

(see Equation (4)). Since the surrogate matrix $P_h D_h - \frac{1}{2}Q$ is skew-symmetric and therefore normal, its spectral radius agrees with its operator 2-norm and, thus, is a convex function of its entries. Moreover, in this norm

$$\rho(D_h) \le \|D_h\| \le \|P_h^{-1}\| \cdot \|P_h D_h\| \le \|P_h^{-1}\| \cdot \left(\|P_h D_h - \frac{1}{2}Q\| + \frac{1}{2}\right)$$

Provided $||P_h||$ is not too large, minimizing the norm of the surrogate matrix tends to make the spectral radius of D_h small. Defining

$$C_0 = P_h D_0 - \frac{1}{2}Q,$$
 $C_i = P_h D_i,$ and $C(\xi) = C_0 + \sum_j \xi_j C_j,$

the goal is to minimize $||C(\xi)||$ with respect to $\xi = (\xi_j)$. It turns out that this problem can be easily transformed into a Semidefinite Program (SDP) [3, Sec. 4.6.3], treatable by a number of standard solvers.

Our implementation of this idea is to choose a particular N (we chose N = 100 for our examples) and to numerically minimize the norm of the surrogate $||C(\xi)||$ with respect to ξ . Unlike in the previous section, there is no apparent need to perform this optimization in exact arithmetic. Instead, the elements of the C_i are evaluated in double precision and are used to set up the SDP, which is then solved through the cvxopt package [1]. Once the optimal ξ is found, the corresponding D_h is formed from Equation (13).

Using this technique on the case s = t = 7, r = 19 (with a 55-dimensional search space) we located an SBP operator such that $\rho(D_h) \approx 2/h$, as verified with several choices of h. Applying the same technique to the case s = t = 6, r = 14, however, did not produce suitable results, as might be expected from careful inspection of Table 1. In particular, the value $\min_i x_i$ associated with this case is a very small number, implying that $||P_h^{-1}||$ is large. When we attempted to minimize the surrogate matrix in this case the resulting SBP operator exhibited (numerically) the scaling $\rho(D_h) \approx 24/h$ —significantly larger than one might wish. This problem can be remedied as follows.

Recall that if the SBP equations are solvable for (s, t, r), then they are solvable for (s, t, r') for any r' > r. In general, increasing r leads to a larger number of degrees of freedom in both P_h and D_h . So, it



Figure 2: The ℓ^{∞} error in approximating the first derivative of the function $f(x) = e^x$ on the interval [0, 1] plotted against the number of sample points for several SBP operators suggesting the appropriate orders of convergence. The integer values in the legend refer to the boundary order (t = s) of the SBP operator. The gray lines denote 5th, 6th, and 7th order slopes respectively.

is reasonable to ask whether choosing r > 14 might improve the result in the case s = 6. With the choice s = 6, r = 15 there is an SBP operator with $\min_i x_i \approx 0.24$. In this case, the resulting SBP operator exhibits the scaling $\rho(D_h) \approx 1.92/h$. Thus, we conclude, that the smallest r for which an SBP operator exists is not necessarily the best choice of r. Apparently, it is useful to choose an r for which $||P_h^{-1}||$ is not too large.

Remark 5.1. It is interesting to note that, although the objective function in Equation (11) was not chosen specifically for this property, a side-effect of the algorithm described in this paper is to choose, among all possible P_h , the one with smallest $||P_h^{-1}||$.

Figure 1 shows the numerically computed spectral radii of the operators described in this section as functions of 1/h. The coefficients for the optimized s = 6 (r = 15) and s = 7 operators are included online as supplementary data for this paper, as described in Appendix A.

6. Numerical experiments

As a first test of the SBP operators, we consider a simple convergence study. Taking N uniformly spaced sample values of the function $x \mapsto e^x$ on the interval [0, 1], we compare the numerically computed derivative to the true derivative in ℓ^{∞} norm. Figure 2 shows the results of this study. The error is plotted against the number of grid points for three SBP operators. The integer values (5, 6, and 7) in the legend are the values of s for each operator. All operators have boundary order t = s and interior order 2s. The width of the closure is chosen to be as small as possible (see Table 1) except in the case t = s = 6, in which case r = 15is chosen as described in Section 5. Note in particular that P is selected as described in Section 3.2. When the SBP operator is not unique, D is selected via the optimization problem described in Section 5. The results in Figure 2 demonstrate that the constructed operators exhibit the appropriate orders of accuracy.

A more interesting test is to consider the operators' performance in solving the advection PDE

$$u_t + u_x = 0$$
 $x \in (0, 1000), t > 0$

with initial and boundary conditions

$$u(x,0) = 0,$$
 $u(0,t) = g(t) = \exp\left(-a(t+10)^2\right)$

where

$$a = \frac{-\log\left(10^{-16}\right)}{100}$$
11



Figure 3: (left) Solution to the advection example of Section 6 at time t = 1000. (right) Error as a function of CPU time for the advection example using a variety of SBP operators and Adams-Bashforth methods. Points labeled in the form s-q indicate the results using an SBP operator with interior order 2s and Adams-Bashforth time integration of order q.

The weight a in the Gaussian is designed so that the function is essentially supported (within a tolerance of 10^{-16}) in an interval of width 20. When the PDE is solved to time t = 1000, the Gaussian hump moves in from the left boundary and traverses the domain to end centered over the point x = 990 (essentially supported in the interval [980, 1000] (see Figure 3).

The purpose of the present study is to explore the best choice of method for solving the problem. The answer, of course, depends on the definition of *best* and also on the methods available for comparison. For this particular study, *best* shall mean that the method achieves a given accuracy in the shortest amount of CPU time. In this case, the CPU is a 1.6GHz desktop Intel Xeon processor running an advection solver written in Fortran. Spatial derivatives were approximated using SPB operators of interior order from 4 to 14 and temporal integration was performed by *q*th order Adams-Bashforth (AB*q*) with $q \in \{3, 4, 6, 7, 8\}$. (Second-and fifth-order AB are excluded from consideration because their stability regions include an insufficient amount of the imaginary axis.) These options are summarized in Table 3. As in the previous test, we selected SBP operators with t = s and with r as small as possible except for the s = 6 case, with P and D generated as described in Sections 3.2 and 5 respectively. The CFL scaling used to ensure stability is given by the product of cfl₁ and cfl₂ given in the table, with the time step size k and spatial step size h related via cfl₁·cfl₂k = h. The boundary conditions were enforced using a simultaneous approximation term (SAT) approach [4], with the semidiscrete equations taking the form

$$v_t + D_h v = -(v^T e_0 - g(t))P_h^{-1}e_0.$$

Asymptotically, with unlimited precision, it is clear that a higher-order method must always outperform a lower-order one. However, for any given problem and a given meaningful range of discretization step sizes in finite precision, it is not clear which choice of spatial or temporal order will achieve a given accuracy in the shortest time. In order to compare the methods, we performed a series of computations using each choice of SBP operator paired with each AB integrator, each for the choices $N \in$ $\{2000, 4000, 6000, 8000, 10000, 12000, 14000\}$ grid points on the interval [0, 1000]-giving a total of $6 \times 5 \times 7 =$ 210 computations. In each computation, the advection problem was solved to time t = 1000, and the maximum absolute error (computed on the computational grid) at this final time was recorded.

The results of the tests are shown in a scatter plot in Figure 3. To simplify the presentation, we have chosen to explicitly identify only three particular methods in the cloud of points. These methods are labeled with the form s-q in the legend, referring to the combination of ABq and the SBP operator with interior order of accuracy 2s. For this configuration, the SBP operator with s = t = 4 paired with AB4 is most effective for errors down to approximately 10^{-4} . If smaller errors are required, the s = t = 5 SBP operator paired

s	t	r	cfl_1	q	cfl_2
2	2	4	1.4	 3	1.39
3	3	6	1.6	4	2.38
4	4	8	1.8	6	8.93
5	5	11	1.9	7	17.5
6	6	15	2.0	8	34.1
$\overline{7}$	7	19	2.1		1

Table 3: SBP parameters (left) and Adams-Bashforth order (right) versus CFL multipliers for the advection example in Section 6. For SBP parameters (s, t, r) and Adams-Bashforth order q, the time step was selected so that $cfl_1 \cdot cfl_2 k = h$.

with AB6 performs well until approximately the error level 10^{-10} , where it is overtaken by the s = t = 7 SBP operator paired with AB6.

Although it is impossible to generalize the results of these tests to predictions for PDEs solvers in general, the outcome does illustrate a general principle suggested by common sense—there is no single *best* numerical method for all problems. Unsurprisingly, if moderate errors are tolerable, a lower-order method does better; the CFL conditions are less stringent than for high-order methods and the constants preceding the leading-order error terms tend to be smaller. As the error tolerance decreases, however, the asymptotic accuracy of the higher-order methods start to win out. It seems a difficult task to determine *a priori* which spatial and temporal order of accuracy will lead to the desired results with the least expenditure of computational time. However, as the example illustrates, there may be realistic instances when very high-order SBP operators are more efficient than lower-order ones.

7. Conclusion and future research

This paper introduces an algorithm that provably answers the question of existence of diagonal-norm SBP operators parameterized by a triple (s, t, r), and demonstrates the need to move away from the standard choice of (s, t, r) = (s, s, 2s) when $s \ge 5$. Our hope is that this approach will lead to further research in the field, and to this end, we conclude with a list of what we consider to be interesting directions of future research.

Floating point algorithms. As remarked in Section 3, the current algorithm relies crucially on the use of exact arithmetic. This appears to be the principal bottleneck preventing the discovery of even higherorder SBP operators than those presented in this text, since this representation leads to very large memory requirements and computationally expensive arithmetic operations. It would be interesting to know if there exists a similar algorithm for finding (approximate) SBP operators in floating point arithmetic. Even the use of a variable-precision library would be an improvement over the need for rational numbers.

Alternative energy norms. The algorithm described in Section 3.2 chooses, among all possible P_h , the one that maximizes $\min_i P_{ii}$. This choice is useful for two reasons. First, if the optimal value is non-positive, then we immediately conclude that no SBP operator exists. Moreover, as remarked in Section 5, this choice is good for the application of optimizing the spectral radius of the SBP derivative operator. On the other hand, if Equation (10) has a manifold of solutions and if one of these solutions is strictly positive, then, in fact, Equation (10) has an infinite number of strictly positive solutions. It is not clear how the choice of P_h might influence later steps of the algorithm.

Compact stencil sizes. As can be seen from Table 1, the closure size, r, appears to grow rapidly as s increases. For example, the 18th order (s = 9) centered operator requires at least 28 points for a 9th order closure. This derivative operator contains a 9th order derivative approximation with a stencil width of 28 + 9 = 37. It is not clear whether a method with such a wide stencil would actually be useful in applications, and it would be interesting to know if there are generalizations of the SBP framework that can reduce this size.

Acknowledgments

The authors are deeply indebted to Daniel Appelö, Jeremy Kozdon and Anders Petersson for their feedback and suggested improvements on early drafts of this manuscript. Joshua Klarmann's work on this project was sponsored by the McNair Scholars' Program and supported by the National Science Foundation under Award No. EPS-0903806 and matching support from the State of Kansas through the Kansas Board of Regents; further funding was received from a scholarship provided by the College of Arts and Sciences at Kansas State University.

Appendix A. Coefficients of new SBP operators

The coefficients for the new 6th- and 7th-order SBP operators described in Section 5 are included online as text files. The files $P_{-6-6-15.txt}$ and $D_{-6-6-15.txt}$ hold the coefficients for P_h and D_h of the 6th order method, respectively. While $P_{-7-7-19.txt}$ and $D_{-7-7-19.txt}$ hold the coefficients for P_h and D_h of the 7th order method.

The coefficients are scaled to the case h = 1. In the case of the norm matrices P_h , the data are stored in rows of 2 columns. Each row holds a pair (i, v) indicating that $\tilde{P}_{ii} = v$. Only the upper-left corner is given, since the lower-right corner can be obtained from symmetry. For the files containing D_h coefficients, each row contains a triple (i, j, v) indicating that $d_{ij} = v$. Again, only the upper-left corner is given.

- ANDERSEN, M. S., DAHL, J., AND VANDENBERGHE, L. CVXOPT: A Python package for convex optimization, Version 1.1.6, 2013. Available at http://cvxopt.org.
- [2] APPELÖ, D., AND PETERSSON, N. A. A stable finite difference method for the elastic wave equation on complex geometries with free surfaces. Commun. Comput. Phys. 5, 1 (2009), 84–107.
- [3] BOYD, S. P., AND VANDENBERGHE, L. Convex optimization. Cambridge university press, 2004.
- [4] CARPENTER, M. H., GOTTLIEB, D., AND ABARBANEL, S. Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: methodology and application to high-order compact schemes. J. Comput. Phys. 111, 2 (1994), 220–236.
- [5] DIENER, P., DORBAND, E. N., SCHNETTER, E., AND TIGLIO, M. Optimized high-order derivative and dissipation operators satisfying summation by parts, and applications in three-dimensional multi-block evolutions. J. Sci. Comput. 32, 1 (2007), 109–145.
- [6] FERNÁNDEZ, D. C. D. R., BOOM, P. D., AND ZINGG, D. W. A generalized framework for nodal first derivative summationby-parts operators. J. Comput. Phys. 266 (2014), 214–239.
- [7] HICKEN, J. E., AND ZINGG, D. W. Summation-by-parts operators and high-order quadrature. J. Comput. Appl. Math. 237, 1 (2013), 111–125.
- [8] KREISS, H.-O., AND OLIGER, J. Comparison of accurate methods for the integration of hyperbolic equations. *Tellus 24*, 3 (1972), 199–215.
- KREISS, H.-O., AND SCHERER, G. Finite element and finite difference methods for hyperbolic partial differential equations. In Mathematical Aspects of Finite Elements in Partial Differential Equations (1974), Academic Press, pp. 195–212.
- [10] MATTSSON, K., AND ALMQUIST, M. A solution to the stability issues with block norm summation by parts operators. J. Comput. Phys. 253 (2013), 418–442.
- [11] MATTSSON, K., ALMQUIST, M., AND CARPENTER, M. H. Optimal diagonal-norm sbp operators. J. Comput. Phys. 264 (2014), 91–111.
- [12] NILSSON, S., PETERSSON, N. A., SJÖGREEN, B., AND KREISS, H.-O. Stable difference approximations for the elastic wave equation in second order formulation. SIAM J. Numer. Anal. 45, 5 (2007), 1902–1936.
- [13] OSUSKY, M., HICKEN, J. E., AND ZINGG, D. W. A parallel Newton-Krylov-Schur flow solver for the Navier-Stokes equations using the SBP-SAT approach. In 48th AIAA Aerospace Sciences Meeting and Exhibit, Orlando, Florida, AIAA-2010-116 (2010).
- [14] PAZOS, E., TIGLIO, M., DUEZ, M. D., KIDDER, L. E., AND TEUKOLSKY, S. A. Orbiting binary black hole evolutions with a multipatch high order finite-difference approach. *Phys. Rev. D* 80, 2 (2009), 024027.
- [15] SJÖGREEN, B., AND PETERSSON, N. A. A fourth order accurate finite difference scheme for the elastic wave equation in second order formulation. J. Sci. Comput. 52, 1 (2012), 17–48.
- [16] STRAND, B. Summation by parts for finite difference approximations for d/dx. J. Comput. Phys. 110, 1 (1994), 47-67.
- [17] SVÄRD, M. On coordinate transformations for summation-by-parts operators. J. Sci. Comput. 20, 1 (2004), 29–42.
- [18] SVÄRD, M., CARPENTER, M. H., AND NORDSTRÖM, J. A stable high-order finite difference scheme for the compressible Navier-Stokes equations, far-field boundary conditions. J. Comput. Phys. 225 (July 2007), 1020–1038.
- [19] SVÄRD, M., AND NORDSTRÖM, J. A stable high-order finite difference scheme for the compressible Navier-Stokes equations: No-slip wall boundary conditions. J. Comput. Phys. 227, 10 (2008), 4805 – 4824.
- [20] SVÄRD, M., AND NORDSTRÖM, J. Review of summation-by-parts schemes for initial-boundary-value problems. J. Comput. Phys. 268 (2014), 17–38.

[21] SYMPY DEVELOPMENT TEAM. SymPy: Python library for symbolic mathematics, 2014. Available at http://www.sympy. org.