

COMPUTATION OF OPTIMAL MONOTONICITY PRESERVING GENERAL LINEAR METHODS

DAVID I. KETCHESON

ABSTRACT. Monotonicity preserving numerical methods for ordinary differential equations prevent the growth of propagated errors and preserve convex boundedness properties of the solution. We formulate the problem of finding optimal monotonicity preserving general linear methods for linear autonomous equations, and propose an efficient algorithm for its solution. This algorithm reliably finds optimal methods even among classes involving very high order accuracy and that use many steps and/or stages. The optimality of some recently proposed methods is verified, and many more efficient methods are found. We use similar algorithms to find optimal strong stability preserving linear multistep methods of both explicit and implicit type, including methods for hyperbolic PDEs that use downwind-biased operators.

1. INTRODUCTION

This work is concerned with the numerical solution of the initial value problem (IVP)

$$(1) \quad \mathbf{u}' = F(\mathbf{u}, t), \quad \mathbf{u}(t_0) = \mathbf{u}_0$$

where $\mathbf{u} \in \mathbb{R}^m$, and the function $F : \mathbb{R}^m \times \mathbb{R} \rightarrow \mathbb{R}^m$ is such that

$$(2) \quad \|\mathbf{u}(t + \Delta t)\| \leq \|\mathbf{u}(t)\| \quad \forall \Delta t \geq 0.$$

Here $\|\cdot\|$ may be a norm, or more generally any convex functional.

Throughout this work it is assumed that F satisfies a discrete analog of (2) when the forward Euler method is employed with a sufficiently small stepsize, i.e.,

$$(3) \quad \|\mathbf{u} + \Delta t F(\mathbf{u}, t)\| \leq \|\mathbf{u}\| \quad \text{for } 0 < \Delta t \leq \Delta t_{\text{FE}}, \quad \text{for all } (\mathbf{u}, t) \in \mathbb{R}^m \times \mathbb{R}.$$

Here, \mathbf{u}_n is a numerical approximation to $\mathbf{u}(t_0 + n\Delta t)$. We are interested in higher order accurate (multistage and/or multistep) numerical methods that preserve the monotonicity property

$$(4) \quad \|\mathbf{u}_n\| \leq \max\{\|\mathbf{u}_{n-1}\|, \dots, \|\mathbf{u}_{n-k}\|\} \quad \text{for } 0 \leq \Delta t \leq \Delta t_{\text{max}}$$

whenever the forward Euler condition (3) holds. Here k represents the number of previous steps used to compute the next solution value. The monotonicity property (4) is desirable in that it mimics property (2) of the true solution and prevents growth of errors.

Received by the editor May 13, 2008 and, in revised form, August 19, 2008.

2000 *Mathematics Subject Classification.* Primary 65L06.

This work was supported by a U.S. Department of Energy Computational Science Graduate Fellowship under grant number DE-FG02-97ER25308, and by AFOSR grant number FA9550-06-1-0255.

©2009 American Mathematical Society
Reverts to public domain 28 years from publication

Strong stability preserving (SSP) methods have been developed to satisfy the monotonicity property (4) for the IVP (1) whenever the forward Euler condition (3) is fulfilled. The monotonicity property is guaranteed under the maximum timestep $\Delta t_{\max} = c\Delta t_{\text{FE}}$, where the SSP coefficient c depends only on the numerical integration method (not on F). Considerable research effort has been devoted to finding methods with the largest value of c among various classes (see, e.g., [7, 14, 12]).

Often, the monotonicity property (4) is of interest when equation (1) results from a method of lines of semi-discretization of the hyperbolic conservation law

$$(5) \quad U_t + \nabla \cdot f(U) = 0.$$

In this case, the semi-discrete operator F (meant to approximate $-\nabla \cdot f$) typically involves an *upwind-biased* discretization. If the discretization is biased in the opposite (*downwind*) direction and negated¹, the resulting operator \tilde{F} is just the reverse-time version of F , and so will also satisfy the forward Euler condition, i.e.,

$$(6) \quad \|\mathbf{u} + \Delta t \tilde{F}(\mathbf{u}, t)\| \leq \|\mathbf{u}\| \text{ for } 0 < \Delta t \leq \Delta t_{\text{FE}}, \text{ for all } (\mathbf{u}, t) \in \mathfrak{R}^m \times \mathfrak{R}.$$

SSP methods with downwinding satisfy the monotonicity property (4) whenever the forward Euler conditions (3) and (6) are fulfilled, under the maximum timestep $\Delta t_{\max} = \tilde{c}\Delta t_{\text{FE}}$, where the SSP coefficient \tilde{c} depends only on the numerical method. For more details regarding methods with downwinding, see e.g. [8].

Several authors have considered the monotonicity property (4) in the simpler case that the function F in (1) is linear and autonomous [25, 17, 26, 19, 20, 5]. Then (1) simplifies to

$$(7) \quad \mathbf{u}' = \mathbf{L}\mathbf{u}, \quad \mathbf{u}(t_0) = \mathbf{u}_0$$

where $\mathbf{L} \in \mathfrak{R}^{m \times m}$ is a constant matrix, and the forward Euler condition (3) simplifies to the circle condition

$$(8) \quad \|\mathbf{I} + \Delta t \mathbf{L}\| \leq 1 \text{ for } 0 < \Delta t \leq \Delta t_{\text{FE}}.$$

Contractive methods have been developed to satisfy the monotonicity property (4) for the linear autonomous IVP (7) whenever the circle condition (8) is fulfilled. The monotonicity property is guaranteed under the maximum timestep $\Delta t_{\max} = R\Delta t_{\text{FE}}$, where the *threshold factor* R depends only on the numerical integration method (not on \mathbf{L}). Since (7) is a special case of (1), clearly $c \leq R$.

Important examples of the linear IVP (7) include semi-discretizations of the partial differential equations describing acoustics, linear elasticity, and electromagnetic waves. Optimally contractive methods may be used to integrate such semi-discretizations, including those with time-dependent source terms [6, 2]. For instance, they have been used for integration of Maxwell's equations [2, 21] and geometrical optics [3], using discontinuous Galerkin semi-discretizations. They are also useful for providing strong stability bounds when applied to spectral semi-discretizations [4, 10, 6].

In the present work we are mainly interested in finding methods with optimal values of R . Previous studies have investigated optimally contractive one-step, multistage methods [17, 26, 6] and one-stage, multistep methods [19, 20]. In the present work we investigate optimally contractive methods of general (i.e. multistage, multistep) type. The key to the present approach lies in reformulating the associated

¹Note that the definition of \tilde{F} here differs from the usual definition by a factor of -1 . This results in slightly simpler linear programming problems in Section 3.

nonlinear optimization problem as a series of linear programming (LP) feasibility problems to which highly efficient LP solvers can then be applied. We will see that this approach is also useful for efficiently finding linear multistep methods with optimal values of c and \tilde{c} .

In Section 2, we investigate optimal contractive general linear methods for the linear IVP. In Section 3, we investigate optimal strong stability preserving multistep methods for the general IVP. We conclude in Section 4.

2. OPTIMAL THRESHOLD FACTORS FOR EXPLICIT GENERAL LINEAR METHODS

2.1. **General linear methods.** When applied to the linear autonomous IVP (7), most numerical methods (including Runge-Kutta and multistep methods) take the form

$$(9) \quad \mathbf{u}_n = \psi_1(\Delta t \mathbf{L})\mathbf{u}_{n-1} + \psi_2(\Delta t \mathbf{L})\mathbf{u}_{n-2} + \cdots + \psi_k(\Delta t \mathbf{L})\mathbf{u}_{n-k}.$$

We will refer to (9) as a *general linear method* for (7) (see, e.g., [1, 11] for a fuller description of general linear methods). The present section deals with explicit methods, for which each function ψ_i is a polynomial whose degree is at most the number of stages of the method, denoted by s :

$$(10) \quad \psi_i = \sum_{j=0}^s a_{ij} z^j \quad 1 \leq i \leq k.$$

The method (9) approximates the solution of (7) to order p if

$$(11) \quad e^{kz} = \psi_1(z)e^{(k-1)z} + \psi_2(z)e^{(k-2)z} + \cdots + \psi_k(z) + \mathcal{O}(z^{p+1}) \text{ for } z \rightarrow 0.$$

Writing the exponential functions in (11) as Taylor series and equating coefficients of powers of z , we find the order conditions for order p in terms of the coefficients a_{ij} :

$$(12) \quad \sum_{i=1}^k \sum_{j=0}^q a_{ij} \frac{(k-i)^{q-j}}{(q-j)!} = \frac{k^q}{q!} \quad 1 \leq q \leq p.$$

In Section 3 we will consider multistep methods, which also take the form (9), but with ψ_i being linear functions (for explicit methods) or rational functions with linear numerator and denominator (for implicit methods).

2.2. **Absolute monotonicity.** When solving the linear IVP (7) by the general linear method (9), the timestep restriction for monotonicity preservation depends on the radius of absolute monotonicity of the functions ψ_i .

Definition 1 (Radius of absolute monotonicity). The radius of absolute monotonicity $R(\psi)$ of a function $\psi : \mathbb{R} \rightarrow \mathbb{R}$ is the largest value of r such that $\psi(z)$ and all of its derivatives exist and are nonnegative for $z \in (-r, 0]$.

When considering absolute monotonicity, it is often helpful to write a function in the form

$$(13) \quad \psi_i(z) = \sum_j \gamma_{ij} \left(1 + \frac{z}{r}\right)^j \quad \text{with } \gamma_{ij} = \frac{r^j}{j!} \psi_i^{(j)}(-r).$$

The coefficients in (13) have the following useful properties:

$$(14a) \quad \gamma_{ij} \geq 0 \iff r \leq R(\psi_i),$$

$$(14b) \quad \sum_{i,j} \gamma_{ij} = 1.$$

The following result may be viewed as a special case of [25, Theorem 3.5], generalized to convex functionals.

Theorem 1. *Assume the matrix \mathbf{L} and convex functional $\|\cdot\|$ satisfy the circle condition (8). Then the monotonicity property (4) holds for the solution of the linear autonomous IVP (7) by a consistent general linear method (9) if the timestep satisfies*

$$(15) \quad \Delta t \leq R\Delta t_{FE}$$

where the threshold factor $R = \min_i R(\psi_i(z))$.

Proof. Using (13) with $r = R$, we have

$$\begin{aligned} \|\mathbf{u}^{n+1}\| &= \left\| \sum_i \psi_i(\Delta t \mathbf{L}) \mathbf{u}_{n-i} \right\| \\ &= \left\| \sum_{i,j} \gamma_{ij} \left(I + \frac{\Delta t}{R} \mathbf{L} \right)^j \mathbf{u}_{n-i} \right\| \\ &\leq \sum_{i,j} \gamma_{ij} \left\| \left(I + \frac{\Delta t}{R} \mathbf{L} \right)^j \right\| \|\mathbf{u}_{n-i}\| \\ &\leq \sum_{i,j} \gamma_{ij} \|\mathbf{u}_{n-i}\| \leq \max_i \|\mathbf{u}_{n-i}\|. \end{aligned}$$

The first inequality follows from (14) and convexity of $\|\cdot\|$, while the second follows from the circle condition (8). \square

In [25], it is shown also that for \mathbf{L} corresponding to a first-order upwind discretization of the advection equation and $\|\cdot\| = \|\cdot\|_\infty$, the result of Theorem 1 is sharp (i.e., the strong stability bound (4) is violated for some \mathbf{u} whenever the timestep (15) is exceeded). This suggests that the timestep bound (15) may be important for semi-discretizations of hyperbolic PDEs.

We can now define the object of interest of the present work. Let $\Pi_{s,k,p}$ denote the set of all ordered sets of k polynomials (ψ_1, \dots, ψ_k) of degree at most s satisfying the order conditions (12) up to order p .

Definition 2. For given integers s, k, p , the *optimal threshold factor* $R_{s,k,p}$ is the largest threshold factor R among all explicit k -step, s -stage general linear methods of order at least p :

$$(16) \quad R_{s,k,p} = \sup \left\{ \min_i R(\psi_i) \mid (\psi_1, \dots, \psi_k) \in \Pi_{s,k,p} \right\}.$$

In the following sections, we will determine values of $R_{s,k,p}$ and corresponding optimal methods, using bisection on R . The results below justify this approach. The following lemma can be seen as a special case of [18, Lemma 3.1].

Lemma 1. *A polynomial $\psi(z)$ is absolutely monotonic at $z = -\xi < 0$ if and only if ψ has radius of absolute monotonicity $R(\psi) \geq \xi$.*

Proof. Suppose $R(\psi) \geq \xi$. Then ψ is absolutely monotonic on the interval $(-\xi, 0]$, so by continuity ψ is absolutely monotonic at $-\xi$.

On the other hand, suppose $\psi(z)$ is absolutely monotonic at $z = -\xi < 0$. Then writing ψ in form (13) with $r = \xi$, using (14a), and differentiating term-by-term shows that $\psi(z)$ is absolutely monotonic on $(-\xi, 0]$. \square

Lemma 1, combined with Definition 2, leads immediately to the following result, which will be useful in constructing our solution algorithm.

Corollary 1. *Let $r > 0$. Then $R_{s,k,p} \geq r$ if and only if there exists $(\psi_1, \dots, \psi_k) \in \Pi_{s,k,p}$ such that each ψ_i is absolutely monotonic at $-r$.*

2.3. Solution Algorithm. We now present an algorithm for finding optimal threshold factors $R_{s,k,p}$ and corresponding methods (ψ_1, \dots, ψ_k) for a given number of stages s , steps k , and order p . Corollary 1 indicates that $R_{s,k,p}$ can be found by bisection, as follows:

Optimization Algorithm
 Inputs: Positive integers s, k, p and real numbers ϵ and r_{\max} such that $R_{s,k,p} \leq r_{\max}$
 Output: r satisfying $R_{s,k,p} - \epsilon \leq r \leq R_{s,k,p}$

1. $r_{\min} := 0$.
2. Set $r := (r_{\max} + r_{\min})/2$.
3. Determine whether there exists $(\psi_1, \dots, \psi_k) \in \Pi_{s,k,p}$ such that each ψ_i is absolutely monotonic at $-r$. If so, set $r_{\min} := r$; otherwise set $r_{\max} := r$.
4. If $r_{\max} - r_{\min} < \epsilon$, set $r := r_{\min}$ and stop. Otherwise, return to step 2.

Two ingredients are necessary for the execution of this approach: a value for r_{\max} and a method to solve the feasibility problem in step 3. The following theorem provides the bound r_{\max} .

Theorem 2. *For any $s, k, p > 0$, the optimal threshold factor for explicit s -stage, k -step, order p general linear methods is at most equal to the number of stages; i.e., $R_{s,k,p} \leq s$.*

Proof. Take any $(\psi_1, \dots, \psi_k) \in \Pi_{s,k,p}$, and let R be the threshold factor of this method. Writing out explicitly the first two order conditions (i.e., (12) for $p = 0, 1$) gives

$$(17a) \quad \sum_{i=1}^k a_{i0} = 1,$$

$$(17b) \quad \sum_{i=1}^k (a_{i1} + (k - i)a_{i0}) = k.$$

Equating the right-hand sides of equations (10) and (13) gives the following relation between the coefficients a_{il} and γ_{ij} :

$$(18) \quad a_{il} = \frac{1}{l!R^l} \sum_{j=0}^s \gamma_{ij} \prod_{n=0}^{l-1} (j - n).$$

Substituting (18) in (17) yields

$$(19a) \quad \sum_{i=1}^k \sum_{j=0}^s \gamma_{ij} = 1,$$

$$(19b) \quad \sum_{i=1}^k \sum_{j=0}^s \gamma_{ij}(j + R(k - i)) = kR.$$

Subtracting ks times (19a) from (19b) gives

$$(20) \quad \sum_{i=1}^k \sum_{j=0}^s \gamma_{ij}(j + R(k - i) - ks) = k(R - s).$$

Since (for $1 \leq i \leq k, 0 \leq j \leq s$)

$$(21) \quad j + R(k - i) - ks = (j - s) + R(1 - i) + (R - s)(k - 1) \leq (R - s)(k - 1),$$

then

$$\begin{aligned} k(R - s) &= \sum_{i=1}^k \sum_{j=0}^s \gamma_{ij}(j + R(k - i) - ks) \\ &\leq (k - 1)(R - s) \sum_{i=1}^k \sum_{j=0}^s \gamma_{ij} \\ &= (k - 1)(R - s), \end{aligned}$$

which implies that $R \leq s$. □

An alternate proof is as follows. Let R denote the threshold factor of the method given by $(\psi_1, \dots, \psi_k) \in \Pi_{s,k,p}$. Consider the disk

$$(22) \quad D_R = \{z \in \mathbb{C} \mid |z + R| \leq R\}.$$

For $z \in D_R, |1 + z/R| \leq 1$, so using (13) and (14), we find $|\sum_i \psi_i(z)| \leq 1$. Hence the region of absolute stability for this method contains D_R . Then [13, Theorem 3.1] asserts that $R \leq s$.

2.4. The Feasibility Problem. We now show that the feasibility problem in step 3 of our algorithm above is a linear programming problem (LP). Using equations (12) and (18), the conditions for the method to be accurate to order p can be written as

$$(23) \quad \sum_{i=1}^k \sum_{j=0}^s \gamma_{ij} \sum_{l=0}^q \binom{q}{l} \frac{(k - i)^{q-l}}{R^l} \prod_{n=0}^{l-1} (j - n) = k^q \quad 1 \leq q \leq p.$$

Thus the feasibility problem can be stated very simply: Determine whether there exist $\gamma_{ij} \geq 0$ satisfying (23). Since (23) is linear in γ_{ij} , for a fixed value of R this is a linear programming feasibility problem. Set

$$(24a) \quad \mathbf{Y} = (\gamma_{10}, \dots, \gamma_{1s}, \gamma_{20}, \dots, \gamma_{2s}, \dots, \gamma_{k0}, \dots, \gamma_{ks})^T,$$

$$(24b) \quad \mathbf{k} = (1, k, k^2, \dots, k^p)^T,$$

$$(24c) \quad [\mathbf{B}(r)]_{q,m} = \sum_{l=0}^q \binom{q}{l} \frac{(k - i)^{q-l}}{r^l} \prod_{n=0}^{l-1} (j - n).$$

where $1 \leq i \leq k, 0 \leq j \leq s, 1 \leq q \leq p$, and $m = s(i - 1) + i + j$. Then the problem is given by the standard form feasibility LP (for given $s, k \geq 1, r > 0$):

LP 1. Determine whether there exists $\boldsymbol{\gamma} \geq 0$ such that $\mathbf{B}(r)\boldsymbol{\gamma} = \mathbf{k}$.

Modern and highly efficient LP solvers can be applied to LP 1. These solvers also return the coefficients describing (ψ_1, \dots, ψ_k) . In the present work, the LP solvers included in MATLAB and Maple have been used. Scripts are available from the SSP website [16].

2.5. Optimal Methods. In this section we present optimal methods found using the algorithm described above. Two important and simpler subclasses of general linear methods are one-stage (i.e., linear multistep) and one-step (i.e., Runge-Kutta) methods. Optimal polynomials corresponding to Runge-Kutta methods were investigated in [17]. The present algorithm was also applied to find such optimal polynomials, in [14]. Optimal linear multistep methods were considered in [19, 20, 10, 8]), and are investigated further in Section 3 of the present work.

We now consider the general case of multistage, multistep methods. Optimization of the threshold factor has not been previously attempted for such methods.

Table 1 gives optimal threshold factors for two-, three- and four-step methods with up to ten stages (for one-step methods, we recover the results presented in [17]).

Table 2 gives optimal threshold factors for two-, three- and four-stage methods with up to ten steps (for one-stage methods, we recover the results presented in [19]). In general, less is gained from additional steps than from additional stages. For instance, for first-order methods, the optimal threshold factor is the same regardless of the number of steps, but increases linearly with the number of stages.

Some of the particularly simple methods are described here. The optimal second-order two-step, s -stage method has $R_{s,2,2} = \sqrt{s(s - 1)}$ and is given by

$$(25) \quad \mathbf{u}_n = \frac{2(s + R) - 2}{2(s + R) - 1} \left(I + \frac{\Delta t \mathbf{L}}{R} \right)^s \mathbf{u}_{n-1} + \frac{1}{2(s + R) - 1} \mathbf{u}_{n-2}.$$

The optimal second-order k -step, two-stage method

$$(26) \quad \mathbf{u}_n = \frac{kR}{2 + R(k - 1)} \left(I + \frac{\Delta t \mathbf{L}}{R} \right)^2 \mathbf{u}_{n-k+1} + \frac{2 - R}{2 + R(k - 1)} \mathbf{u}_{n-k}$$

has $R_{2,k,2} = 2/(\sqrt{(k - 1)^2 + 1} - k + 2)$. The optimal third-order two-step, eight-stage method

$$(27) \quad \mathbf{u}_n = \frac{2}{3} \left(I + \frac{\Delta t \mathbf{L}}{R} \right)^8 \mathbf{u}_{n-1} + \frac{1}{3} \left(I + \frac{\Delta t \mathbf{L}}{R} \right)^8 \mathbf{u}_{n-2}$$

has $R_{8,2,3} = 6$. The optimal third order three-step, three-stage method

$$(28) \quad \mathbf{u}_n = \frac{3}{4} \left(I + \frac{\Delta t \mathbf{L}}{R} \right)^3 \mathbf{u}_{n-1} + \frac{1}{4} \left(I + \frac{\Delta t \mathbf{L}}{R} \right)^3 \mathbf{u}_{n-3}$$

has $R_{3,3,3} = 2$.

The methods are presented here in a form corresponding to (13). This form is useful for implementation because it can be shown (see [19, Lemma 2.4(ii)]) that any optimal method will have at most p nonzero coefficients γ_{ij} . The last two methods above are remarkable in that they have only $p - 1$ nonzero coefficients. They can be

TABLE 1. Threshold factors $R_{s,k,p}$ of optimal two-, three- and four-step general linear methods.

	s	P									
		1	2	3	4	5	6	7	8	9	10
k=2	1	1.0									
	2	2.0	1.414	0.732							
	3	3.0	2.449	1.651	1.284	0.654					
	4	4.0	3.464	2.507	2.118	1.620	1.217				
	5	5.0	4.472	3.385	2.929	2.355	1.977	1.447			
	6	6.0	5.477	4.229	3.775	3.071	2.614	2.027	1.465		
	7	7.0	6.481	5.093	4.662	3.649	3.137	2.567	2.099	1.501	
	8	8.0	7.483	6.0	5.433	4.274	3.737	3.078	2.641	2.180	1.562
	9	9.0	8.485	6.674	6.114	4.837	4.366	3.621	3.180	2.777	2.363
	10	10.0	9.487	7.352	6.797	5.461	4.979	4.239	3.796	3.335	3.000
k=3	1	1.0	0.500								
	2	2.0	1.618	1.113	0.822						
	3	3.0	2.637	2.0	1.586	1.123	0.466	0.051			
	4	4.0	3.646	2.617	2.241	1.728	1.466	1.083	0.716		
	5	5.0	4.651	3.393	3.060	2.489	2.215	1.879	1.655	1.342	
	6	6.0	5.653	4.229	3.897	3.284	2.994	2.582	2.291	2.011	1.668
	7	7.0	6.655	5.093	4.777	4.016	3.701	3.135	2.846	2.412	2.093
	8	8.0	7.657	6.0	5.624	4.633	4.307	3.708	3.393	2.898	2.515
	9	9.0	8.658	6.674	6.303	5.251	4.911	4.267	3.891	3.350	3.019
	10	10.0	9.659	7.352	6.985	5.884	5.557	4.794	4.446	3.822	3.513
k=4	1	1.0	0.667	0.333							
	2	2.0	1.721	1.243	0.934	0.542					
	3	3.0	2.732	2.0	1.684	1.223	0.928	0.555			
	4	4.0	3.737	2.617	2.331	1.883	1.664	1.406	1.188	0.602	0.325
	5	5.0	4.740	3.393	3.143	2.639	2.400	2.035	1.751	1.525	1.235
	6	6.0	5.742	4.229	3.975	3.403	3.124	2.676	2.437	2.123	1.903
	7	7.0	6.743	5.093	4.847	4.039	3.770	3.230	2.996	2.640	2.443
	8	8.0	7.744	6.0	5.723	4.633	4.384	3.801	3.572	3.197	2.969
	9	9.0	8.745	6.674	6.400	5.251	4.986	4.383	4.148	3.719	3.478
	10	10.0	9.745	7.352	7.081	5.884	5.628	4.987	4.742	4.237	4.005

implemented very efficiently by storing only the quantities $(I + \frac{\Delta t \mathbf{L}}{R})^s \mathbf{u}$ for each solution vector \mathbf{u} .

The second-order methods are of particular interest because they are also optimal second-order SSP methods in their respective classes. For instance, the optimal second-order two-stage methods have been proposed as SSP methods in [12], while the optimal second-order two-step methods have been proposed in [15]. The present results imply that these methods, which in some cases were obtained by nonlinear optimization, are indeed optimal, even among much larger classes of methods than those considered in [12, 15]. These results also imply the optimality of the third-order SSP general linear methods with up to three stages or three steps in [12, 15].

3. STRONG STABILITY PRESERVING LINEAR MULTISTEP METHODS

In the previous section, we investigated methods with optimal values of the threshold factor R , which governs the maximum timestep for monotonicity in the

TABLE 2. Threshold factors $R_{s,k,p}$ of optimal two-, three-, and four-stage general linear methods.

	k	1	2	3	4	P 5	6	7	8	9	10
s=2	1	2.0	1.0								
	2	2.0	1.414	0.732							
	3	2.0	1.618	1.113	0.823						
	4	2.0	1.721	1.243	0.934	0.542					
	5	2.0	1.781	1.243	0.984	0.674	0.434				
	6	2.0	1.820	1.243	1.028	0.796	0.575	0.263			
	7	2.0	1.847	1.243	1.063	0.833	0.691	0.413	0.174		
	8	2.0	1.867	1.243	1.089	0.876	0.750	0.484	0.308	0.010	
	9	2.0	1.883	1.243	1.109	0.905	0.765	0.573	0.395	0.186	0.021
	10	2.0	1.895	1.243	1.124	0.905	0.779	0.614	0.481	0.266	0.115
s=3	1	3.0	2.0	1.0							
	2	3.0	2.449	1.651	1.284	0.655	0.000				
	3	3.0	2.637	2.0	1.586	1.123	0.466	0.052			
	4	3.0	2.732	2.0	1.684	1.223	0.928	0.555	0.000		
	5	3.0	2.788	2.0	1.752	1.384	1.143	0.945	0.388	0.131	
	6	3.0	2.825	2.0	1.798	1.450	1.272	1.006	0.755	0.427	0.000
	7	3.0	2.851	2.0	1.831	1.467	1.301	1.018	0.851	0.656	0.257
	8	3.0	2.870	2.0	1.855	1.467	1.325	1.075	0.947	0.780	0.520
	9	3.0	2.885	2.0	1.873	1.467	1.339	1.121	0.990	0.870	0.645
	10	3.0	2.897	2.0	1.887	1.467	1.351	1.150	1.052	0.873	0.706
s=4	1	4.0	3.0	2.0	1.0						
	2	4.0	3.464	2.507	2.118	1.620	1.217	0.000			
	3	4.0	3.646	2.617	2.241	1.728	1.466	1.083	0.716		
	4	4.0	3.737	2.617	2.331	1.883	1.664	1.406	1.188	0.602	0.325
	5	4.0	3.791	2.617	2.390	1.979	1.802	1.465	1.271	1.025	0.815
	6	4.0	3.827	2.617	2.430	2.002	1.827	1.533	1.336	1.180	0.975
	7	4.0	3.852	2.617	2.459	2.002	1.848	1.603	1.448	1.260	1.130
	8	4.0	3.871	2.617	2.480	2.002	1.866	1.643	1.525	1.297	1.150
	9	4.0	3.886	2.617	2.496	2.002	1.881	1.658	1.537	1.309	1.179
	10	4.0	3.898	2.617	2.509	2.002	1.893	1.658	1.548	1.340	1.242

solution of the linear autonomous problem (7). We now investigate methods with optimal values of the SSP coefficient c (or \tilde{c}), which governs the maximum timestep for monotonicity in the solution of the nonlinear nonautonomous problem (1). For multistage methods, the problem of finding optimal SSP methods involves highly nonlinear order conditions, and cannot be solved by the technique used here. For linear multistep methods, however, the order conditions and the equations describing the SSP coefficient are linear in the coefficients of the method, and it is possible to solve the relevant optimization problem using bisection and solving a series of feasibility LPs. In each case, the appropriate LP is obtained by combining the order conditions and the inequalities that are necessarily satisfied by the SSP coefficient.

Note that, for explicit linear multistep methods, the quantities R and c are equal. For implicit linear multistep methods, the difference between the two is generally small (see Thm. 4.3 and Cor. 4.4 of [20]).

In the following, we consider implicit and explicit linear multistep methods, including methods with downwinding.

3.1. Explicit methods. Optimal explicit linear multistep methods have been studied previously [23, 24, 19, 9]. Known results include optimal explicit methods of up to seventh-order and twenty steps, as well as an algorithm for finding the optimal methods of arbitrary order with arbitrary number of steps. Using the present algorithm we have computed optimal methods of up to fiftieth-order. Note that, since $c = R$ for explicit linear multistep methods, this problem is just a special case of the problem discussed in the previous section, and the methods given here are optimal in terms of both threshold factor and SSP coefficient.

For simplicity, we use here the traditional notation for multistep methods, rather than the notation of the previous section. A k -step linear multistep method has the form

$$\mathbf{u}_n = \sum_{j=0}^{k-1} \alpha_j \mathbf{u}_{n-k+j} + \Delta t \beta_j F(\mathbf{u}_{n-k+j}).$$

The method is accurate to order p if

$$(29) \quad \sum_{j=0}^{k-1} (\alpha_j j^i + \beta_j i j^{i-1}) = k^i \quad (0 \leq i \leq p)$$

and has SSP coefficient $c \geq r$ if

$$(30) \quad \beta_j \geq 0 \quad \alpha_j - r\beta_j \geq 0 \quad (0 \leq j \leq k-1).$$

If (30) is satisfied for some $r > 0$, then clearly it is satisfied for any smaller positive value of r . This implies that bisection can be used to find c .

Introducing $\delta_j = \alpha_j - r\beta_j$, the feasibility problem in this case (equivalent to LP 1 with $s = 1$) takes the form (for given positive integers k, p with $p \leq k$)

LP 2. Given r , determine whether there exist β_j, δ_j such that

$$\beta_j, \delta_j \geq 0 \quad (0 \leq j \leq k-1),$$

$$\sum_{j=0}^{k-1} ((\delta_j + r\beta_j)j^i + \beta_j i j^{i-1}) = k^i \quad (0 \leq i \leq p).$$

Thus optimal methods may be found by applying the algorithm of Section 2.3, where the feasibility problem in step 3 is replaced by LP 2. Computed optimal values of the SSP coefficient for $1 \leq k \leq 50, 1 \leq p \leq 15$ are shown in Table 3. Again, these values are also optimal threshold factors for methods in their classes. In the notation of the previous sections, the optimal coefficients are values of $R_{1,k,p}$.

3.2. Explicit methods with downwinding. A k -step linear multistep method with downwinding takes the form

$$(31) \quad \mathbf{u}_n = \sum_{j=0}^{k-1} \alpha_j \mathbf{u}_{n-k+j} + \Delta t \beta_j F(\mathbf{u}_{n-k+j}) + \Delta t \tilde{\beta}_j \tilde{F}(\mathbf{u}_{n-k+j}).$$

The method is accurate to order p if

$$(32) \quad \sum_{j=0}^{k-1} (\alpha_j j^i + \beta_j i j^{i-1} - \tilde{\beta}_j i j^{i-1}) = k^i \quad (0 \leq i \leq p)$$

TABLE 3. SSP coefficients of optimal explicit linear multistep methods

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1.000														
2	1.000														
3	1.000	0.500													
4	1.000	0.667	0.333												
5	1.000	0.750	0.500	0.021											
6	1.000	0.800	0.583	0.165											
7	1.000	0.833	0.583	0.282	0.038										
8	1.000	0.857	0.583	0.359	0.145										
9	1.000	0.875	0.583	0.393	0.228										
10	1.000	0.889	0.583	0.421	0.282	0.052									
11	1.000	0.900	0.583	0.443	0.317	0.115									
12	1.000	0.909	0.583	0.460	0.345	0.175	0.018								
13	1.000	0.917	0.583	0.474	0.370	0.210	0.077								
14	1.000	0.923	0.583	0.484	0.390	0.236	0.116								
15	1.000	0.929	0.583	0.493	0.406	0.259	0.154	0.012							
16	1.000	0.933	0.583	0.501	0.411	0.276	0.177	0.044							
17	1.000	0.938	0.583	0.507	0.411	0.291	0.198	0.075							
18	1.000	0.941	0.583	0.513	0.411	0.304	0.217	0.106	0.003						
19	1.000	0.944	0.583	0.517	0.411	0.314	0.232	0.128	0.035						
20	1.000	0.947	0.583	0.521	0.411	0.322	0.246	0.148	0.063						
21	1.000	0.950	0.583	0.525	0.411	0.330	0.259	0.163	0.082						
22	1.000	0.952	0.583	0.528	0.411	0.337	0.269	0.177	0.100	0.010					
23	1.000	0.955	0.583	0.531	0.411	0.342	0.278	0.190	0.116	0.031					
24	1.000	0.957	0.583	0.534	0.411	0.347	0.286	0.201	0.131	0.048					
25	1.000	0.958	0.583	0.536	0.411	0.351	0.294	0.210	0.145	0.063					
26	1.000	0.960	0.583	0.538	0.411	0.354	0.301	0.220	0.155	0.078	0.012				
27	1.000	0.962	0.583	0.540	0.411	0.358	0.307	0.228	0.165	0.091	0.027				
28	1.000	0.963	0.583	0.542	0.411	0.360	0.312	0.234	0.174	0.105	0.042				
29	1.000	0.964	0.583	0.543	0.411	0.363	0.317	0.240	0.184	0.116	0.055				
30	1.000	0.966	0.583	0.545	0.411	0.365	0.319	0.246	0.191	0.125	0.066	0.002			
31	1.000	0.967	0.583	0.546	0.411	0.368	0.319	0.250	0.199	0.134	0.079	0.014			
32	1.000	0.968	0.583	0.548	0.411	0.370	0.319	0.255	0.205	0.142	0.089	0.026			
33	1.000	0.969	0.583	0.549	0.411	0.371	0.319	0.259	0.211	0.150	0.097	0.036			
34	1.000	0.970	0.583	0.550	0.411	0.373	0.319	0.262	0.216	0.157	0.106	0.047			
35	1.000	0.971	0.583	0.551	0.411	0.375	0.319	0.266	0.221	0.164	0.114	0.057	0.005		
36	1.000	0.971	0.583	0.552	0.411	0.376	0.319	0.269	0.225	0.169	0.121	0.066	0.016		
37	1.000	0.972	0.583	0.553	0.411	0.377	0.319	0.271	0.229	0.175	0.128	0.074	0.027		
38	1.000	0.973	0.583	0.554	0.411	0.378	0.319	0.274	0.233	0.180	0.134	0.082	0.036		
39	1.000	0.974	0.583	0.555	0.411	0.379	0.319	0.276	0.237	0.185	0.140	0.089	0.044		
40	1.000	0.974	0.583	0.555	0.411	0.380	0.319	0.278	0.240	0.189	0.146	0.096	0.052	0.003	
41	1.000	0.975	0.583	0.556	0.411	0.381	0.319	0.280	0.243	0.193	0.152	0.103	0.059	0.011	
42	1.000	0.976	0.583	0.557	0.411	0.382	0.319	0.281	0.246	0.197	0.156	0.109	0.066	0.020	
43	1.000	0.976	0.583	0.557	0.411	0.383	0.319	0.283	0.249	0.200	0.161	0.114	0.073	0.027	
44	1.000	0.977	0.583	0.558	0.411	0.384	0.319	0.284	0.251	0.203	0.165	0.119	0.079	0.034	
45	1.000	0.977	0.583	0.559	0.411	0.385	0.319	0.286	0.253	0.206	0.169	0.124	0.085	0.040	0.000
46	1.000	0.978	0.583	0.559	0.411	0.385	0.319	0.287	0.256	0.209	0.172	0.129	0.090	0.047	0.008
47	1.000	0.978	0.583	0.560	0.411	0.386	0.319	0.288	0.258	0.211	0.176	0.133	0.095	0.053	0.015
48	1.000	0.979	0.583	0.560	0.411	0.387	0.319	0.289	0.260	0.213	0.179	0.138	0.100	0.058	0.022
49	1.000	0.979	0.583	0.561	0.411	0.387	0.319	0.290	0.261	0.215	0.182	0.141	0.105	0.064	0.028
50	1.000	0.980	0.583	0.561	0.411	0.388	0.319	0.291	0.261	0.218	0.185	0.145	0.109	0.069	0.034

and has SSP coefficient $\tilde{c} \geq r$ if

$$(33a) \quad \beta_j, \tilde{\beta}_j \geq 0 \quad (0 \leq j \leq k - 1),$$

$$(33b) \quad \alpha_j - r(\beta_j + \tilde{\beta}_j) \geq 0 \quad (0 \leq j \leq k - 1).$$

Optimal SSP explicit linear multistep methods with downwinding have been studied previously [24, 10, 22, 8]. Previous searches for methods in this class were restricted to methods satisfying $\beta_j \tilde{\beta}_j = 0$. The following lemma shows that this restriction is automatically satisfied by optimal methods.

Lemma 2. *Any optimal SSP method of the form (31) has the property that $\beta_j \tilde{\beta}_j = 0$ for each j .*

Proof. Note that the order conditions (32) depend only on the difference $\beta_j - \tilde{\beta}_j$, while the inequality constraint (33b) can be written as (setting $r = \tilde{c}$)

$$\tilde{c} \leq \alpha_j / (\beta_j + \tilde{\beta}_j).$$

Suppose that an optimal method has $\beta_j > \tilde{\beta}_j > 0$ for $j \in J_1 \subset (0, 1, \dots, k - 1)$ and $\tilde{\beta}_j > \beta_j > 0$ for $j \in J_2 \subset (0, 1, \dots, k - 1)$. Then for $j \in J_1$ set $\beta_j^* = \beta_j - \tilde{\beta}_j$ and $\tilde{\beta}_j^* = 0$; for $j \in J_2$ set $\tilde{\beta}_j^* = \tilde{\beta}_j - \beta_j$ and $\beta_j^* = 0$. Then (α, β^*) satisfies (33) and (32) with a larger value of \tilde{c} , which is a contradiction. \square

Lemma 2 could be used to write the optimization problem in terms of fewer variables, as was done in [22, 8]. However, by retaining all of the $\beta_j, \tilde{\beta}_j$, the problem can be solved more easily. If (33) is satisfied for some positive value of \tilde{c} , then it holds for any smaller positive value. Hence we are again justified in using bisection to find optimal methods. The feasibility problem to be solved at each step is (with $\delta_j = \alpha_j - r(\beta_j + \tilde{\beta}_j)$):

LP 3. Given r , determine whether there exist $\beta, \tilde{\beta}, \delta$ such that

$$\beta_j, \tilde{\beta}_j, \delta_j \geq 0 \quad (0 \leq j \leq k - 1),$$

$$\sum_{j=0}^{k-1} \left((\delta_j + r\beta_j + r\tilde{\beta}_j)j^i + \beta_j i j^{i-1} - \tilde{\beta}_j i j^{i-1} \right) = k^i \quad (0 \leq i \leq p).$$

Optimal coefficients are known for methods with up to $k = 10$ steps and order $p = 6$ [8]. Optimal SSP coefficients for $1 \leq k \leq 50, 1 \leq p \leq 15$ are given in Table 4. For $p \leq 6, k \leq 10$, these values agree with those reported in [8]. The remaining values are new. Note that, for large values of k , there is little or no difference between the SSP coefficient for the optimal methods with and without downwinding.

3.3. Implicit methods. A k -step implicit linear multistep method has the form

$$\mathbf{u}_n - \Delta t \beta_k F(\mathbf{u}_n) = \sum_{j=0}^{k-1} \alpha_j \mathbf{u}_{n-k+j} + \Delta t \beta_j F(\mathbf{u}_{n-k+j}).$$

The method is accurate to order p if

$$(34) \quad \sum_{j=0}^{k-1} \alpha_j j^i + \sum_{j=0}^k \beta_j i j^{i-1} = k^i \quad (0 \leq i \leq p).$$

and has SSP coefficient $\tilde{c} \geq r$ if

$$(35) \quad \beta_j \geq 0, \quad \alpha_j - r\beta_j \geq 0 \quad (0 \leq j \leq k).$$

Again we can find optimal methods in this class using bisection, but with the following feasibility LP (where $\delta_j = \alpha_j - r\beta_j$):

LP 4. Given r , determine whether there exist δ_j, β_j such that

$$\beta_j, \delta_j \geq 0 \quad (0 \leq j \leq k - 1),$$

$$\sum_{j=0}^{k-1} (\delta_j + r\beta_j)j^i + \sum_{j=0}^k \beta_j i j^{i-1} = k^i \quad (0 \leq i \leq p).$$

SSP coefficients of optimal methods for $1 \leq k \leq 20, 1 \leq p \leq 8$ were computed in [20]. SSP coefficients of optimal methods for $1 \leq k \leq 50, 1 \leq p \leq 15$ are listed in Table 5.

TABLE 4. SSP coefficients for optimal explicit linear multistep methods with downwinding

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1.000														
2	1.000	0.500													
3	1.000	0.667	0.287												
4	1.000	0.750	0.415	0.159											
5	1.000	0.800	0.517	0.237	0.087										
6	1.000	0.833	0.583	0.283	0.131	0.046									
7	1.000	0.857	0.583	0.360	0.187	0.081	0.024								
8	1.000	0.875	0.583	0.394	0.223	0.107	0.044	0.013							
9	1.000	0.889	0.583	0.424	0.261	0.142	0.068	0.025	0.007						
10	1.000	0.900	0.583	0.447	0.299	0.175	0.086	0.037	0.013	0.003					
11	1.000	0.909	0.583	0.464	0.326	0.199	0.104	0.056	0.022	0.007	0.002				
12	1.000	0.917	0.583	0.477	0.351	0.215	0.131	0.069	0.030	0.012	0.004	0.000			
13	1.000	0.923	0.583	0.487	0.372	0.242	0.154	0.083	0.044	0.019	0.007	0.002	0.000		
14	1.000	0.929	0.583	0.496	0.391	0.266	0.171	0.096	0.055	0.024	0.010	0.004	0.001	0.000	
15	1.000	0.933	0.583	0.503	0.407	0.280	0.187	0.112	0.066	0.034	0.016	0.006	0.002	0.001	0.000
16	1.000	0.938	0.583	0.509	0.411	0.295	0.199	0.133	0.075	0.043	0.020	0.008	0.003	0.001	0.000
17	1.000	0.941	0.583	0.514	0.411	0.307	0.216	0.147	0.086	0.051	0.025	0.013	0.005	0.002	0.001
18	1.000	0.944	0.583	0.519	0.411	0.317	0.232	0.161	0.095	0.060	0.032	0.016	0.007	0.003	0.001
19	1.000	0.947	0.583	0.523	0.411	0.326	0.243	0.173	0.111	0.067	0.040	0.020	0.010	0.004	0.002
20	1.000	0.950	0.583	0.526	0.411	0.334	0.254	0.182	0.125	0.077	0.046	0.024	0.013	0.005	0.002
21	1.000	0.952	0.583	0.529	0.411	0.339	0.263	0.189	0.135	0.084	0.053	0.030	0.015	0.007	0.004
22	1.000	0.955	0.583	0.532	0.411	0.344	0.272	0.201	0.146	0.090	0.059	0.036	0.018	0.010	0.004
23	1.000	0.957	0.583	0.535	0.411	0.349	0.280	0.212	0.157	0.103	0.066	0.040	0.022	0.012	0.006
24	1.000	0.958	0.583	0.537	0.411	0.353	0.288	0.220	0.164	0.113	0.073	0.047	0.026	0.015	0.007
25	1.000	0.960	0.583	0.539	0.411	0.356	0.295	0.228	0.171	0.122	0.079	0.052	0.031	0.017	0.009
26	1.000	0.962	0.583	0.541	0.411	0.359	0.302	0.235	0.177	0.132	0.085	0.057	0.036	0.020	0.011
27	1.000	0.963	0.583	0.543	0.411	0.362	0.307	0.241	0.183	0.140	0.092	0.063	0.040	0.023	0.013
28	1.000	0.964	0.583	0.544	0.411	0.365	0.312	0.246	0.189	0.147	0.102	0.069	0.045	0.027	0.015
29	1.000	0.966	0.583	0.546	0.411	0.367	0.317	0.250	0.196	0.154	0.109	0.074	0.050	0.031	0.018
30	1.000	0.967	0.583	0.547	0.411	0.369	0.319	0.254	0.202	0.159	0.117	0.079	0.054	0.034	0.020
31	1.000	0.968	0.583	0.548	0.411	0.371	0.319	0.257	0.209	0.164	0.124	0.084	0.058	0.038	0.023
32	1.000	0.969	0.583	0.549	0.411	0.373	0.319	0.261	0.214	0.169	0.131	0.089	0.064	0.043	0.026
33	1.000	0.970	0.583	0.550	0.411	0.374	0.319	0.265	0.219	0.173	0.137	0.096	0.068	0.047	0.029
34	1.000	0.971	0.583	0.551	0.411	0.376	0.319	0.268	0.224	0.178	0.142	0.102	0.072	0.050	0.033
35	1.000	0.971	0.583	0.552	0.411	0.377	0.319	0.271	0.228	0.182	0.146	0.109	0.076	0.054	0.036
36	1.000	0.972	0.583	0.553	0.411	0.378	0.319	0.273	0.231	0.185	0.151	0.115	0.081	0.058	0.039
37	1.000	0.973	0.583	0.554	0.411	0.379	0.319	0.276	0.235	0.190	0.155	0.120	0.085	0.062	0.043
38	1.000	0.974	0.583	0.555	0.411	0.380	0.319	0.278	0.238	0.194	0.159	0.126	0.089	0.066	0.046
39	1.000	0.974	0.583	0.556	0.411	0.381	0.319	0.279	0.240	0.198	0.162	0.130	0.095	0.070	0.049
40	1.000	0.975	0.583	0.556	0.411	0.382	0.319	0.281	0.243	0.202	0.166	0.134	0.100	0.073	0.052
41	1.000	0.976	0.583	0.557	0.411	0.383	0.319	0.283	0.245	0.206	0.169	0.138	0.105	0.076	0.056
42	1.000	0.976	0.583	0.558	0.411	0.384	0.319	0.284	0.248	0.208	0.172	0.142	0.109	0.080	0.060
43	1.000	0.977	0.583	0.558	0.411	0.385	0.319	0.286	0.249	0.211	0.174	0.145	0.114	0.083	0.063
44	1.000	0.977	0.583	0.559	0.411	0.385	0.319	0.287	0.251	0.214	0.177	0.148	0.119	0.086	0.066
45	1.000	0.978	0.583	0.560	0.411	0.386	0.319	0.288	0.254	0.216	0.179	0.151	0.122	0.091	0.069
46	1.000	0.978	0.583	0.560	0.411	0.387	0.319	0.289	0.256	0.218	0.182	0.154	0.126	0.095	0.072
47	1.000	0.979	0.583	0.561	0.411	0.387	0.319	0.290	0.258	0.220	0.185	0.157	0.129	0.099	0.075
48	1.000	0.979	0.583	0.561	0.411	0.388	0.319	0.291	0.260	0.222	0.188	0.160	0.132	0.103	0.078
49	1.000	0.980	0.583	0.562	0.411	0.388	0.319	0.292	0.261	0.224	0.191	0.162	0.135	0.107	0.080
50	1.000	0.980	0.583	0.562	0.411	0.389	0.319	0.292	0.261	0.225	0.193	0.164	0.138	0.111	0.083

3.4. **Implicit methods with downwinding.** As a final application we consider the SSP properties of implicit linear multistep methods with downwinding, which take the form

$$\mathbf{u}_n - \Delta t \beta_k F(\mathbf{u}_n) - \Delta t \tilde{\beta}_k \tilde{F}(\mathbf{u}_n) = \sum_{j=0}^{k-1} \alpha_j \mathbf{u}_{n-k+j} + \Delta t \beta_j F(\mathbf{u}_{n-k+j}) + \Delta t \tilde{\beta}_j \tilde{F}(\mathbf{u}_{n-k+j}).$$

As far as we know, methods of this type have not been studied previously. The method is accurate to order p if

$$\sum_{j=0}^{k-1} \alpha_j j^i + \sum_{j=0}^k (\beta_j - \tilde{\beta}_j) i j^{i-1} = k^i \quad (0 \leq i \leq p)$$

TABLE 5. SSP coefficients of optimal implicit linear multistep methods

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	∞	2.000													
2	∞	2.000	1.000												
3	∞	2.000	1.500	1.000											
4	∞	2.000	1.667	1.243	0.667										
5	∞	2.000	1.750	1.243	0.796	0.500									
6	∞	2.000	1.800	1.243	0.929	0.660	0.300								
7	∞	2.000	1.833	1.243	1.006	0.784	0.468	0.197							
8	∞	2.000	1.857	1.243	1.052	0.868	0.550	0.345	0.006						
9	∞	2.000	1.875	1.243	1.084	0.905	0.642	0.443	0.206	0.024					
10	∞	2.000	1.889	1.243	1.106	0.905	0.690	0.533	0.295	0.127					
11	∞	2.000	1.900	1.243	1.123	0.905	0.733	0.580	0.393	0.203					
12	∞	2.000	1.909	1.243	1.136	0.905	0.764	0.625	0.444	0.295	0.093				
13	∞	2.000	1.917	1.243	1.147	0.905	0.781	0.662	0.485	0.353	0.171	0.042			
14	∞	2.000	1.923	1.243	1.155	0.905	0.795	0.692	0.526	0.402	0.238	0.103			
15	∞	2.000	1.929	1.243	1.162	0.905	0.806	0.714	0.551	0.438	0.303	0.163			
16	∞	2.000	1.933	1.243	1.168	0.905	0.815	0.719	0.578	0.468	0.342	0.221	0.076		
17	∞	2.000	1.938	1.243	1.174	0.905	0.823	0.719	0.593	0.493	0.368	0.273	0.128	0.028	
18	∞	2.000	1.941	1.243	1.178	0.905	0.829	0.719	0.609	0.518	0.397	0.299	0.173	0.072	
19	∞	2.000	1.944	1.243	1.182	0.905	0.835	0.719	0.623	0.535	0.420	0.328	0.223	0.111	
20	∞	2.000	1.947	1.243	1.186	0.905	0.839	0.719	0.631	0.550	0.443	0.356	0.253	0.159	0.034
21	∞	2.000	1.950	1.243	1.189	0.905	0.844	0.719	0.639	0.564	0.459	0.374	0.279	0.191	0.081
22	∞	2.000	1.952	1.243	1.192	0.905	0.847	0.719	0.646	0.577	0.475	0.394	0.302	0.221	0.120
23	∞	2.000	1.955	1.243	1.194	0.905	0.851	0.719	0.651	0.587	0.487	0.410	0.320	0.242	0.148
24	∞	2.000	1.957	1.243	1.197	0.905	0.853	0.719	0.656	0.595	0.497	0.426	0.336	0.264	0.178
25	∞	2.000	1.958	1.243	1.199	0.905	0.856	0.719	0.661	0.596	0.506	0.439	0.353	0.283	0.201
26	∞	2.000	1.960	1.243	1.201	0.905	0.858	0.719	0.665	0.596	0.514	0.450	0.368	0.300	0.221
27	∞	2.000	1.962	1.243	1.202	0.905	0.861	0.719	0.668	0.596	0.520	0.459	0.383	0.314	0.241
28	∞	2.000	1.963	1.243	1.204	0.905	0.862	0.719	0.671	0.596	0.527	0.468	0.392	0.327	0.257
29	∞	2.000	1.964	1.243	1.205	0.905	0.864	0.719	0.674	0.596	0.532	0.476	0.402	0.341	0.270
30	∞	2.000	1.966	1.243	1.207	0.905	0.866	0.719	0.676	0.596	0.536	0.482	0.413	0.352	0.283
31	∞	2.000	1.967	1.243	1.208	0.905	0.867	0.719	0.678	0.596	0.540	0.489	0.420	0.362	0.294
32	∞	2.000	1.968	1.243	1.209	0.905	0.869	0.719	0.679	0.596	0.543	0.495	0.426	0.372	0.305
33	∞	2.000	1.969	1.243	1.210	0.905	0.870	0.719	0.681	0.596	0.547	0.500	0.433	0.380	0.316
34	∞	2.000	1.970	1.243	1.211	0.905	0.871	0.719	0.682	0.596	0.549	0.505	0.438	0.388	0.327
35	∞	2.000	1.971	1.243	1.212	0.905	0.872	0.719	0.684	0.596	0.552	0.508	0.444	0.394	0.335
36	∞	2.000	1.971	1.243	1.213	0.905	0.873	0.719	0.685	0.596	0.554	0.508	0.448	0.400	0.343
37	∞	2.000	1.972	1.243	1.214	0.905	0.874	0.719	0.686	0.596	0.556	0.508	0.451	0.406	0.349
38	∞	2.000	1.973	1.243	1.215	0.905	0.875	0.719	0.687	0.596	0.557	0.508	0.455	0.411	0.356
39	∞	2.000	1.974	1.243	1.216	0.905	0.876	0.719	0.688	0.596	0.559	0.508	0.458	0.416	0.363
40	∞	2.000	1.974	1.243	1.217	0.905	0.877	0.719	0.689	0.596	0.560	0.508	0.461	0.420	0.368
41	∞	2.000	1.975	1.243	1.217	0.905	0.878	0.719	0.690	0.596	0.562	0.508	0.463	0.424	0.373
42	∞	2.000	1.976	1.243	1.218	0.905	0.879	0.719	0.691	0.596	0.563	0.508	0.465	0.428	0.377
43	∞	2.000	1.976	1.243	1.218	0.905	0.879	0.719	0.692	0.596	0.564	0.508	0.467	0.431	0.381
44	∞	2.000	1.977	1.243	1.219	0.905	0.880	0.719	0.692	0.596	0.565	0.508	0.469	0.434	0.385
45	∞	2.000	1.977	1.243	1.220	0.905	0.881	0.719	0.693	0.596	0.566	0.508	0.471	0.437	0.389
46	∞	2.000	1.978	1.243	1.220	0.905	0.881	0.719	0.694	0.596	0.567	0.508	0.473	0.439	0.391
47	∞	2.000	1.978	1.243	1.221	0.905	0.882	0.719	0.694	0.596	0.568	0.508	0.474	0.441	0.395
48	∞	2.000	1.979	1.243	1.221	0.905	0.882	0.719	0.695	0.596	0.568	0.508	0.475	0.444	0.397
49	∞	2.000	1.979	1.243	1.222	0.905	0.883	0.719	0.695	0.580	0.569	0.508	0.476	0.443	0.399
50	∞	2.000	1.980	1.243	1.222	0.905	0.883	0.719	0.696	0.573	0.570	0.508	0.477	0.443	0.402

and has SSP coefficient $\tilde{c} \geq r$ if

$$(36) \quad \beta_j, \tilde{\beta}_j \geq 0 \quad (0 \leq j \leq k-1),$$

$$(37) \quad \alpha_j - r(\beta_j + \tilde{\beta}_j) \geq 0 \quad (0 \leq j \leq k-1).$$

Setting $\delta_j = \alpha_j - r(\beta_j + \tilde{\beta}_j)$, the relevant LP feasibility problem is

LP 5. Given r , determine whether there exist $\beta_j, \tilde{\beta}_j, \delta_j$ such that

$$\beta_j, \tilde{\beta}_j, \delta_j \geq 0 \quad (0 \leq j \leq k-1)$$

$$\sum_{j=0}^{k-1} (\delta_j + r(\beta_j + \tilde{\beta}_j)) j^i + \sum_{j=0}^k (\beta_j - \tilde{\beta}_j) i j^{i-1} = k^i \quad (0 \leq i \leq p)$$

As for explicit methods with downwinding, it is straightforward to show that the optimal solution will always have $\beta_j \tilde{\beta}_j = 0$ for each j . SSP coefficients of optimal methods for $1 \leq k \leq 40, 1 \leq p \leq 15$ are given in Table 6.

TABLE 6. SSP coefficients of optimal implicit linear multistep methods with downwinding

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	∞	2.000													
2	∞	2.000	1.303												
3	∞	2.000	1.591	1.000											
4	∞	2.000	1.710	1.243	0.744										
5	∞	2.000	1.776	1.243	0.871	0.544									
6	∞	2.000	1.817	1.243	0.976	0.672	0.384								
7	∞	2.000	1.845	1.243	1.035	0.788	0.535	0.261							
8	∞	2.000	1.866	1.243	1.073	0.870	0.619	0.410	0.171						
9	∞	2.000	1.882	1.243	1.093	0.905	0.668	0.474	0.280	0.107					
10	∞	2.000	1.894	1.243	1.117	0.905	0.718	0.542	0.369	0.191	0.065				
11	∞	2.000	1.905	1.243	1.132	0.905	0.755	0.585	0.433	0.280	0.119	0.038			
12	∞	2.000	1.913	1.243	1.143	0.905	0.776	0.629	0.485	0.304	0.195	0.078	0.021		
13	∞	2.000	1.920	1.243	1.153	0.905	0.791	0.664	0.512	0.376	0.234	0.121	0.047	0.013	
14	∞	2.000	1.926	1.243	1.160	0.905	0.803	0.693	0.540	0.424	0.296	0.175	0.077	0.028	0.007
15	∞	2.000	1.931	1.243	1.167	0.905	0.813	0.714	0.570	0.455	0.342	0.227	0.116	0.050	0.014
16	∞	2.000	1.935	1.243	1.172	0.905	0.821	0.719	0.588	0.479	0.376	0.240	0.157	0.074	0.029
17	∞	2.000	1.939	1.243	1.177	0.905	0.828	0.719	0.605	0.497	0.404	0.266	0.199	0.103	0.047
18	∞	2.000	1.943	1.243	1.181	0.905	0.834	0.719	0.620	0.520	0.430	0.330	0.228	0.140	0.067
19	∞	2.000	1.946	1.243	1.185	0.905	0.839	0.719	0.630	0.536	0.445	0.354	0.263	0.170	0.089
20	∞	2.000	1.949	1.243	1.188	0.905	0.843	0.719	0.638	0.551	0.457	0.379	0.292	0.201	0.124
21	∞	2.000	1.951	1.243	1.191	0.905	0.847	0.719	0.645	0.565	0.472	0.396	0.313	0.227	0.148
22	∞	2.000	1.953	1.243	1.194	0.905	0.850	0.719	0.651	0.577	0.484	0.413	0.335	0.256	0.172
23	∞	2.000	1.956	1.243	1.196	0.905	0.853	0.719	0.656	0.587	0.495	0.424	0.352	0.276	0.196
24	∞	2.000	1.957	1.243	1.198	0.905	0.856	0.719	0.660	0.595	0.504	0.435	0.369	0.293	0.220
25	∞	2.000	1.959	1.243	1.200	0.905	0.858	0.719	0.664	0.596	0.513	0.443	0.383	0.308	0.240
26	∞	2.000	1.961	1.243	1.202	0.905	0.860	0.719	0.668	0.596	0.520	0.453	0.392	0.325	0.258
27	∞	2.000	1.962	1.243	1.203	0.905	0.862	0.719	0.671	0.596	0.526	0.460	0.402	0.339	0.274
28	∞	2.000	1.964	1.243	1.205	0.905	0.864	0.719	0.674	0.596	0.532	0.469	0.410	0.352	0.289
29	∞	2.000	1.965	1.243	1.206	0.905	0.866	0.719	0.676	0.596	0.536	0.477	0.417	0.362	0.301
30	∞	2.000	1.966	1.243	1.208	0.905	0.867	0.719	0.678	0.596	0.540	0.483	0.423	0.370	0.314
31	∞	2.000	1.967	1.243	1.209	0.905	0.869	0.719	0.679	0.596	0.543	0.489	0.430	0.379	0.326
32	∞	2.000	1.968	1.243	1.210	0.905	0.870	0.719	0.681	0.596	0.547	0.495	0.435	0.386	0.336
33	∞	2.000	1.969	1.243	1.211	0.905	0.871	0.719	0.682	0.596	0.549	0.500	0.439	0.392	0.343
34	∞	2.000	1.970	1.243	1.212	0.905	0.872	0.719	0.684	0.596	0.552	0.505	0.444	0.398	0.350
35	∞	2.000	1.971	1.243	1.213	0.905	0.873	0.719	0.685	0.596	0.554	0.508	0.447	0.403	0.356
36	∞	2.000	1.972	1.243	1.214	0.905	0.874	0.719	0.686	0.596	0.556	0.508	0.451	0.408	0.363
37	∞	2.000	1.973	1.243	1.215	0.905	0.875	0.719	0.687	0.596	0.558	0.508	0.455	0.413	0.369
38	∞	2.000	1.973	1.243	1.216	0.905	0.876	0.719	0.688	0.596	0.559	0.508	0.458	0.417	0.374
39	∞	2.000	1.974	1.243	1.216	0.905	0.877	0.719	0.689	0.596	0.560	0.508	0.461	0.421	0.378
40	∞	2.000	1.975	1.243	1.217	0.905	0.878	0.719	0.690	0.596	0.562	0.508	0.463	0.425	0.382

4. CONCLUSION

A new approach to determining optimal monotonicity preserving methods has been proposed and applied to several classes of methods. The algorithm efficiently computes optimal explicit methods for solution of linear autonomous systems, among the full class of general linear methods for which the theory of contractivity preservation was established in [25]. As special cases, known results from [17, 19, 20] are recovered.

Many of the resulting methods allow larger contractive timesteps for linear differential equations than do previously known methods, even when this is normalized by the cost of each step. Furthermore, the results verify the optimality of some known strong stability preserving methods.

A similar algorithm has been applied to find optimal SSP methods of linear multistep type. Extensive results have been obtained for explicit and implicit methods, with and without downwinding, including methods of up to fifteenth-order of accuracy. This is the first investigation of implicit SSP methods with downwinding.

To save space, only a few of the optimal methods have been provided. Optimal methods may easily be found using the scripts available from [16].

ACKNOWLEDGMENT

The author thanks Sigal Gottlieb and Randy LeVeque for helpful comments on early drafts of this work, and the anonymous referees for their suggestions that improved the paper.

REFERENCES

- [1] J.C. Butcher. *Numerical Methods for Ordinary Differential Equations*. Wiley, 2003. MR1993957 (2004e:65069)
- [2] M.-H. Chen, B. Cockburn, and F. Reitich. High-order RKDG methods for computational electromagnetics. *Journal of Scientific Computing*, 22-23:205–226, 2005. MR2142195 (2005m:65208)
- [3] Bernardo Cockburn, Jianliang Qian, Fernando Reitich, and Jing Wang. An accurate spectral/discontinuous finite-element formulation of a phase-space-based level set approach to geometrical optics. *Journal of Computational Physics*, 208:175–195, 2005. MR2144697 (2006a:65124)
- [4] D. Gottlieb and E. Tadmor. The CFL condition for spectral approximations to hyperbolic initial-boundary value problems. *Mathematics of Computation*, 56:565–588, 1991. MR1066833 (91k:65137)
- [5] Sigal Gottlieb. On high order strong stability preserving Runge-Kutta and multi step time discretizations. *Journal of Scientific Computing*, 25:105–127, 2005. MR2231945 (2007f:65029)
- [6] Sigal Gottlieb and Lee-Ad J. Gottlieb. Strong stability preserving properties of Runge-Kutta time discretization methods for linear constant coefficient operators. *Journal of Scientific Computing*, 18:83–109, 2003. MR1958936 (2003m:65161)
- [7] Sigal Gottlieb, David I. Ketcheson, and Chi-Wang Shu. High order strong stability preserving time discretizations. *Journal of Scientific Computing*, DOI: 10.1007/s10915-008-9239-z.
- [8] Sigal Gottlieb and Steven J. Ruuth. Optimal strong-stability-preserving time-stepping schemes with fast downwind spatial discretizations. *Journal of Scientific Computing*, 27:289–303, 2006. MR2285782 (2008j:65122)
- [9] Sigal Gottlieb and Chi-Wang Shu. Total variation diminishing Runge-Kutta schemes. *Mathematics of Computation*, 67:73–85, 1998. MR1443118 (98c:65122)
- [10] Sigal Gottlieb, Chi-Wang Shu, and Eitan Tadmor. Strong stability preserving high-order time discretization methods. *SIAM Review*, 43:89–112, 2001. MR1854647 (2002f:65132)
- [11] E. Hairer, S.P. Norsett, and G. Wanner. *Solving ordinary differential equations I: Nonstiff Problems*. Springer Series in Computational Mathematics. Springer, Berlin, 1993. MR1227985 (94c:65005)
- [12] C. Huang. Strong stability preserving hybrid methods. *Applied Numerical Mathematics*, 2008. doi: 10.1016/j.apnum.2008.03.030.
- [13] Rolf Jeltsch and Olavi Nevanlinna. Stability of explicit time discretizations for solving initial value problems. *Numerische Mathematik*, 37:61–91, 1981. MR615892 (82g:65042)
- [14] David I. Ketcheson. Highly efficient strong stability preserving Runge-Kutta methods with low-storage implementations. *SIAM Journal on Scientific Computing*, 30(4):2113–2136, 2008. MR2407154
- [15] David I. Ketcheson. Strong stability preserving two-step Runge-Kutta methods, 2008, in preparation.
- [16] David I. Ketcheson, Colin B. Macdonald, and Sigal Gottlieb. See the SSP website: <http://www.cfm.brown.edu/people/sg/ssp.html>.
- [17] J. F. B. M. Kraaijevanger. Absolute monotonicity of polynomials occurring in the numerical solution of initial value problems. *Numerische Mathematik*, 48:303–322, 1986. MR826471 (87c:65084)
- [18] J. F. B. M. Kraaijevanger. Contractivity of Runge-Kutta methods. *BIT*, 31:482–528, 1991. MR1127488 (92i:65120)
- [19] H W. J. Lenferink. Contractivity-preserving explicit linear multistep methods. *Numerische Mathematik*, 55:213–223, 1989. MR987386 (90f:65058)
- [20] H W. J. Lenferink. Contractivity-preserving implicit linear multistep methods. *Math. Comp.*, 56:177–199, 1991. MR1052098 (91i:65129)

- [21] Tiao Lu, Wei Cai, and Pingwen Zhang. Discontinuous Galerkin time-domain method for GPR simulation in dispersive media. *IEEE Transactions on Geoscience and Remote Sensing*, 43(1):72–80, 2005.
- [22] Steven J. Ruuth and Willem Hundsdorfer. High-order linear multistep methods with general monotonicity and boundedness properties. *Journal of Computational Physics*, 209:226–248, 2005. MR2145787 (2005m:65143)
- [23] J. Sand. Circle contractive linear multistep methods. *BIT*, 26:114–122, 1986. MR833836 (87h:65124)
- [24] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77:439–471, 1988. MR954915 (89g:65113)
- [25] M. N. Spijker. Contractivity in the numerical solution of initial value problems. *Numerische Mathematik*, 42:271–290, 1983. MR723625 (85b:65067)
- [26] J. A. van de Griend and J. F. B. M. Kraaijevanger. Absolute monotonicity of rational functions occurring in the numerical solution of initial value problems. *Numerische Mathematik*, 49:413–424, 1986. MR853663 (87m:65102)

DEPARTMENT OF APPLIED MATHEMATICS, UNIVERSITY OF WASHINGTON, SEATTLE, WASHINGTON 98195-2420

E-mail address: `ketch@amath.washington.edu`