International Journal of Pure and Applied Mathematics
Volume 15 No. 4 2004, 419-437

PRECONDITIONING AND DOMAIN DECOMPOSITION SCHEMES TO SOLVE PDES

M.T. Darvishi

Department of Mathematics
Faculty of Science
Razi University
Kermanshah, IRAN
e-mail: darvishi@razi.ac.ir

Abstract: We present in this paper some new preconditioning methods in spectral collocation derivative approach to solve linear partial differential equations. Also, additional preconditioning schemes to reduce roundoff errors in computing derivatives using matrix vector multiplication method are introduced. Numerical results over some problems indicate the superiority of the new methods.

AMS Subject Classification: 65N55, 65M70
Key Words: pseudospectral, matrix vector multiplication, preconditioning

1. Introduction

Many real life problems are reduced to solve some type of ordinary or partial differential equations (ODEs or PDEs). Since these equations often do not have closed form solutions, researchers can only compute solutions approximately. One of the methods to solve ODEs and PDEs is the spectral collocation method or the pseudospectral method.

Spectral collocation methods have become increasingly popular for solving differential equations also they are very useful in providing highly accurate solutions to differential equations.

Received: May 10, 2004

In this method such an approximation $u_N(x)$ to $u(x)$ is presented that $u_N(x_i) = u(x_i)$ for some collocation points x_i. After setting u_N in the differential equation, we have to use derivative(s) of u_N at the collocation points. A straightforward implementation of the spectral collocation methods involves the use of spectral differentiation matrices to compute derivatives at the collocation points, in which if $\vec{\mathbf{u}} = \{u(x_i)\}$ is the vector consisting of values of $u(x)$ at the $N+1$ collocation points and $\vec{\mathbf{u'}} = \{u'(x_i)\}$ consists of values of the derivatives at the collocation points, then the collocation derivative matrix D is the matrix mapping $\vec{\mathbf{u}} \mapsto \vec{\mathbf{u'}}$. The entries of derivative matrix D can be computed analytically. To obtain optimal accuracy these matrices must be computed carefully. In [6]-[24] the authors describe the subject very well.

Some researchers have worked on the problem of reducing roundoff errors in Chebyshev collocation derivative methods. Baltensperger and Trummer [3] demonstrate that naive algorithms for computing these matrices suffer from severe loss of accuracy due to roundoff errors. Breuer and Everson [7] introduce a preconditioning to reduce roundoff error by making the value of the function on the boundaries vanish. In [20], the author attempts to combat roundoff by preconditioning the problem. Don and Solomonoff [12] try to reduce the roundoff error by using trigonometric identities to rewrite components of the derivative matrix. They also use the Tal-Ezer mapping to reduce the roundoff error [13]. Tang and Trummer in [23] use trigonometric identities and a flipping trick to reduce roundoff errors. A different approach is suggested by [4] and [1], [2].

It is shown that we can not compute high order derivatives in required accuracy by using available numerical methods. Some researchers such as Bernardi and Maday [5], Don and Solomonoff [12], [13] have worked on this issue. In [10], [11] some works to reduce roundoff error of Chebyshev collocation derivatives are presented. In [10] a new preconditioning method to reduce roundoff errors that occur when computing derivatives using Chebyshev collocation methods is investigated. In [10], [11] to enhance accuracy of differentiation, a domain decomposition approach is used. Some applications of new algorithms are presented in [9]. These include applications of new algorithms for solving some ODEs.

In this study, based on previous works, we provide further applications of the preconditioning and domain decomposition methods for solving linear PDEs by using pseudospectral method.

The remainder of the paper is organized as follows. In the following section we give the preconditioning and domain decomposition methods. Section 3 describes the pseudospectral method to solve a linear PDE. In Section 4, we introduce two alternative preconditioning methods. We will report the applications of these new preconditionings on ordinary and partial differential equations in a future paper.

2. Algorithms

Let $f(x)$ be a smooth function on $[-1,1]$. We interpolate $f(x)$ by constructing the N-order interpolation polynomial $g_j(x)$ such that $g_j(x_k)=\delta_{jk}$, that is

$$u(x)=\sum_{j=0}^{N} g_j(x)u_j\,, \tag{1}$$

where $u(x)$ is a polynomial of degree at most N and $u_j = f(x_j)$ for $j = 0,1,\ldots,N$. In fact, we have

$$f(x)=u(x)+E(x)\,, \tag{2}$$

where

$$E(x)=\frac{\pi(x)f^{(N+1)}(\eta)}{(N+1)!}\,, \tag{3}$$

with $\pi(x)=\Pi_{i=0}^{N}(x-x_i)$ and η depends on x and it is in the interval spanned by $x_0,x_1,\ldots,x_N$ and x (see [19]). It can be shown that [5] we have

$$g_j(x)=\frac{(-1)^{j+1}(1-x^2)T_N'(x)}{c_jN^2(x-x_j)},\quad j=0,1,\ldots,N, \tag{4}$$

where $c_0=c_N=2$ and $c_j=1,\ j=1,2,\ldots,N-1$.

In this paper, we use Chebyshev-Gauss-Lobatto points as the grid points. These points are defined as the extrema of the N-th order Chebyshev polynomial, i.e.,

$$T_N(x)=\cos(N\cos^{-1}x), \tag{5}$$

the extreme points of this polynomial are

$$x_j=\cos(\frac{j\pi}{N}),\quad j=0,1,\ldots,N. \tag{6}$$

In pseudospectral method the derivative of $u(x)$ at the collocation point x_j is computed by matrix vector multiplication. The entries of the Chebyshev derivative matrix $D=(D_{kj})$ are computed by taking the analytical derivative of $g_j(x)$ and evaluating it at the collocation points x_k for $j,k = 0,1,\ldots,N$, i.e., $D_{kj} = g_j'(x_k)$.

The formula for $D_{kj},\ k \neq j$, is commonly given as

$$D_{kj} = \frac{c_k}{c_j}\frac{(-1)^{j+k}}{(x_k - x_j)},$$

with corresponding modifications for $D_{kk},\ k \neq 0, N$, [16]. The largest elements in the matrix D are concentrated in the top-left and bottom-right corners. If x_1 is subject to a roundoff error ε then D_{01} contains an error that grows like $O(N^4\varepsilon)$. Don and Solomonoff [12] use trigonometric identities to reduce roundoff error. They introduce D_{kj}s as follows

$$\begin{aligned}
D_{kj} &= -\frac{c_k}{2c_j}\frac{(-1)^{j+k}}{\sin((k+j)\frac{\pi}{2N})\sin((k-j)\frac{\pi}{2N})}, && k \neq j\,,\\
D_{kk} &= -\frac{1}{2}\cos(\frac{k\pi}{N})(1+\cot^2(\frac{k\pi}{N})), && k \neq 0, N\,,\\
D_{00} &= -D_{NN} = \frac{2N^2+1}{6}.
\end{aligned} \tag{7}$$

Formula (7), which avoid differencing of nearly-equal numbers, have been introduced to reduce this source of error from $O(N^4\varepsilon)$ to $O(N^3\varepsilon)$. Don and Solomonoff [12] show that, even with utilization of (7), the error incurred in the evaluation of $D\mathbf{u}$ near $x = -1$ is significantly larger than at $x = 1$, even if $\mathbf{u}$ is symmetric (related to the accuracies achieved in evaluating $\sin(x)$ and $\sin(\pi - x)$ for small x). In other words, if k and j are small then D_{kj} can be computed accurately whilst if k and j are near N then the evaluation of D_{kj} is less accurate. This can be utilized by evaluating D_{kj} in the upper half of the matrix and then 'flipping' to take advantage of the symmetry property

$$D_{kj} = -D_{N-k,N-j}.$$

The derivative of u at x_k becomes

$$u_k' = (D\vec{\mathbf{u}})_k = \sum_{j=0}^{N} D_{kj}u_j, \quad k = 0,1,\ldots,N, \tag{8}$$

from (2) we have

$$f'(x_k) = u'(x_k) + E_k^{'}\,, \tag{9}$$

where

$$E_k^{'} = E_k^1 = \frac{\pi'(x_k)f^{(N+1)}(\gamma)}{(N+1)!}, \tag{10}$$

where γ is in the interval spanned by $x_0, x_1, \ldots, x_N$ (see [19]).

2.1. Preconditioning

In this part, we introduce an interesting preconditioning. From (7) for $k \neq j$ we have

$$|D_{kj}| = \frac{c_k}{2c_j} \frac{1}{|\sin((k+j)\frac{\pi}{2N})\sin((k-j)\frac{\pi}{2N})|},$$

since the value of $\sin((k-j)\frac{\pi}{2N})$ is near zero when k is near j, hence this causes $|D_{kj}|$ become large for k near j. This means entries of derivative matrix D, with large absolute values, are on a band near main diagonal. That is large values of $|D_{kj}|$ correspond to values of k near j. Therefore matrix vector multiplication method causes large roundoff error. In [10] to reduce roundoff error in k-th node, we defined $h_k(x)$ as follows:

$$h_k(x) = u(x) - u(x_k), \tag{11}$$

where $x_k = \cos(\frac{k\pi}{N})$. From (1) we have

$$h_k(x) = \sum_{j=0}^{N} g_j(x) h_k(x_j), \tag{12}$$

hence, from (8) the derivative of h_k at $x = x_k$ is as follows

$$h'_k(x_k) = \sum_{j=0}^{N} D_{kj} h_k(x_j), \tag{13}$$

or

$$u'(x_k) = \sum_{\substack{j=0 \\ j\neq k}}^{N} D_{kj}(u(x_j) - u(x_k)). \tag{14}$$

Therefore, by using this preconditioning, we can reduce the influence of large values of $|D_{kj}|$, in the matrix vector multiplication method.

Remark. Equation (14) is mathematically the same equation as equation (22) in [3]. Therefore, the preconditioning presented in [10] is another way of deriving the "so-called" negative sum trick (NST, see for example [3] for further comments on NST).

2.2. Domain Decomposition Scheme

In this part, we state and prove two important theorems. The first one shows the error propagation in computing high order derivatives of a function using matrix vector multiplication. In the second theorem we prove that in M-panel, P-point domain decomposition Chebyshev collocation method, by increasing the number of panels, error term approaches to zero.

We have the following relation to compute q-th order derivative:

$$f^{(q)}(x_k) = f_k^{(q)} = \sum_{j=0}^{N} D_{kj} f_j^{(q-1)} + E_k^q, \quad q = 1, 2, \ldots, \quad k = 0, 1, \ldots, N, \quad (15)$$

where

$$E_k^q = \frac{\pi'(x_k) f^{(N+q)}(\xi_q)}{(N+1)!}, \quad (16)$$

and ξ_q lies in the interval spanned by $x_0, \ldots, x_N$.

To be able to compute r-th order derivative of function f we have the following theorem.

Theorem 1. *If*

$$f_k^{(q)} = \sum_{j=0}^{N} D_{kj} f_j^{(q-1)} + E_k^q, \qquad q = 1, 2, \ldots,$$

then for $q = r$*;*

$$f_k^{(r)} = \sum_{j=0}^{N} D_{kj}^{(r)} f_j + \sum_{j=0}^{N} D_{kj}^{(r-1)} E_j^1 + \sum_{j=0}^{N} D_{kj}^{(r-2)} E_j^2 + \cdots + \sum_{j=0}^{N} D_{kj} E_j^{r-1} + E_k^r, \quad (17)$$

where $D^s = (D_{kj}^{(s)})$ and E_k^s is defined by (16) for $s = 1, \ldots, r$.

Proof. By induction on r, as for $r = 1$ by definition we have

$$f_k' = \sum_s D_{ks} f_s + E_k^1, \quad k = 0, \ldots, N,$$

and so

$$f_k'' = \sum_j D_{kj} f_j' + E_k^2, \quad k = 0, \ldots, N,$$

then

$$f_k'' = \sum_j D_{kj}[\sum_s D_{js} f_s + E_j^1] + E_k^2 = \sum_s [\sum_j D_{kj} D_{js}] f_s + \sum_j D_{kj} E_j^1$$
$$+ E_k^2 = \sum_s D_{ks}^{(2)} f_s + \sum_j D_{kj} E_j^1 + E_k^2 = \sum_j D_{kj}^{(2)} f_j + \sum_j D_{kj} E_j^1 + E_k^2,$$

as for Chebvshev-Gauss-Lobatto interpolation points we have, $D \cdot D = D^2$. Hence the relation holds for $r = 1$, 2. Assume that, the relation holds for $r-1$, i.e.;

$$f_k^{(r-1)} = \sum_{j=0}^N D_{kj}^{(r-1)} f_j + \sum_{j=0}^N D_{kj}^{(r-2)} E_j^1 + \sum_{j=0}^N D_{kj}^{(r-3)} E_j^2 + \cdots$$
$$+ \sum_{j=0}^N D_{kj} E_j^{r-2} + E_k^{r-1},$$

we have

$$f_k^{(r)} = \sum_s D_{ks} f_s^{(r-1)} + E_k^r\,,$$

by induction hypothesis

$$f_k^{(r)} = \sum_s D_{ks}[\sum_j D_{sj}^{(r-1)} f_j + \sum_j D_{sj}^{(r-2)} E_j^1 + \sum_j D_{sj}^{(r-3)} E_j^2 + \cdots$$
$$+ \sum_j D_{sj} E_j^{r-2} + E_s^{r-1}] + E_k^r = \sum_j [\sum_s D_{ks} D_{sj}^{(r-1)}] f_j$$
$$+ \sum_j [\sum_s D_{ks} D_{sj}^{(r-2)}] E_j^1 + \sum_j [\sum_s D_{ks} D_{sj}^{(r-3)}] E_j^2$$
$$+ \cdots + \sum_j [\sum_s D_{ks} D_{sj}] E_j^{r-2} + \sum_s D_{ks} E_s^{r-1}] + E_k^r$$
$$= \sum_{j=0}^N D_{kj}^{(r)} f_j + \sum_{j=0}^N D_{kj}^{(r-1)} E_j^1 + \sum_{j=0}^N D_{kj}^{(r-2)} E_j^2 + \cdots + \sum_{j=0}^N D_{kj} E_j^{r-1} + E_k^r,$$

as $D^p = D \cdot D^{p-1}$. Hence the theorem is proved. □

As (17) shows, the error propagates and causes accumulation of errors in computing r-th order ($r \geq 2$) derivative. Hence, increasing the order of differentiation in Chebyshev collocation method (CCM) causes reduction of accuracy. To deal with this issue we must reduce the errors incurred in derivatives of orders less than r. For this, a typical domain decomposition method is used. The basis of this technique is reduction of E_k^s $(s < r)$ using a domain decomposition method. The following domain decomposition method is used.

We divide $[-1,1]$ to M equal panels $[y_i, y_{i+1}]$, $\quad i = 0, 1, \dots, M-1$, such that

$$-1 = y_0 < y_1 < y_2 < \cdots < y_M = 1. \tag{18}$$

In each panel we use a P-point pseudospectral method such that $MP = N$. In the following theorem we show that in M-panel, P-point domain decomposition Chebyshev collocation method, by increasing the number of panels, the numerical solution converges to the exact solution.

Theorem 2. *If $f \in C^{P+1}[-1,1]$, then in the M-panel, P-point domain decomposition Chebyshev collocation method with collocation points $\{x_k\}_{k=0}^N$ we have*

$$\lim_{M\to\infty} E_{k,M} = 0\,,$$

where $E_{k,M}$ is the error of domain decomposition Chebyshev collocation method at x_k using M panels.

Proof. By relation (16) the error term of differentiating function f of (2) at x_k with r points (instead of N points) is

$$E_k = \frac{\pi'(x_k) f^{(r+1)}(\xi(x_k))}{(r+1)!}.$$

Now, we divide interval [-1,1] to M panels and use a P-point CCM in each subinterval $[y_i, y_{i+1}]$, where

$$y_i = -1 + \frac{2i}{M}, \quad i = 0, \dots, M,$$

also for $x \in [-1,1]$ and $z \in [y_i, y_{i+1}]$, consider

$$z = \frac{(y_{i+1} - y_i)x + (y_{i+1} + y_i)}{2}, \tag{19}$$

whence

$$f(z) = f(\frac{(y_{i+1} - y_i)x + (y_{i+1} + y_i)}{2}). \tag{20}$$

By (20) and (9) in CCM we have

$$\frac{d}{dx} f\left(\frac{(y_{i+1} - y_i)x + (y_{i+1} + y_i)}{2}\right) = \frac{y_{i+1} - y_i}{2} f'\left(\frac{(y_{i+1} - y_i)x + (y_{i+1} + y_i)}{2}\right). \tag{21}$$

Thus in each panel we have:

$$f'\left(\frac{(y_{i+1}-y_i)x+(y_{i+1}+y_i)}{2}\right)$$
$$= \frac{2}{y_{i+1}-y_i}\sum_j D_{kj}F_j + \frac{\pi'(x_k)(y_{i+1}-y_i)^P f^{(P+1)}(\xi(z_k))}{2^P(P+1)!}, \quad (22)$$

where $F_j = f(\frac{1}{M}x_j + \frac{2i+1}{M} - 1)$ and ξ belongs to the interval spanned by $x_0, \ldots, x_N$. Let $\Delta y = y_{i+1} - y_i$, $i = 0, \ldots, M-1$ and

$$E_{k,i} = \frac{\pi'(x_k)f^{(P+1)}(\xi(z_k))(\Delta y)^P}{2^P(P+1)!}, \quad i = 0, \ldots, M-1, \quad (23)$$

since $f(x) \in C^{P+1}[-1,1]$ there exist positive K such that for all $x \in [-1,1]$

$$|f^{(P+1)}(x)| \leq K, \quad (24)$$

also there exist positive L such that for all $x \in [-1,1]$ we have

$$|\pi'(x)| \leq L. \quad (25)$$

Therefore from (24) and (25) for $M \to \infty$ we have

$$|E_{k,i}| \leq \frac{LK}{M^P(P+1)!} \longrightarrow 0\,, \quad (26)$$

hence the theorem is proved. □

3. Application

In this section, we apply the preconditioning method and domain decomposition scheme on two particular partial differential equations. These equations are solved by the pseudospectral method in [21] and the stability of pseudospectral Chebyshev method is demonstrated for them in [15], [17]. The general form of these problems is:

$$u_t = S(x)u_x, \quad -1 \leq x \leq 1, \quad 0 < t, \quad (27)$$

with given boundary and/or initial conditions. In [21] the authors used the pseudospectral method. They only considered the solution at the collocation points. The collocation points are

$$x_j = \cos\frac{j\pi}{N}, \quad j = 0, \ldots, N. \quad (28)$$

To solve (27) by the pseudospectral method, we replace the derivative (in space) by a matrix multiplication as we described it in the previous section.

If $u_N(x,t) = v(x,t)$ is an approximation to $u(x,t)$, following [15] we can write

$$u(x,t) \simeq u_N(x,t) = \sum_{k=0}^{N} a_k(t) T_k(x), \tag{29}$$

where

$$a_k(t) = \frac{2}{Nc_k} \sum_{j=0}^{N} \frac{1}{c_j} v(x_j,t) \cos \frac{\pi j k}{N}, \quad k = 0, \ldots, N. \tag{30}$$

Let $u_N = v$, then

$$u(x,t) \simeq v(x,t) = \sum_{k=0}^{N} a_k(t) T_k(x).$$

The following cases for $S(x)$ are considered in [21]:

i) $S(x) = 1$,

ii) $S(x) = -x$.

In this paper we consider (27) with the above values of $S(x)$.

Case (i). $S(x) = 1$.

In this case, the resulting PDE is:

$$\begin{array}{lll} u_t = u_x, & -1 \leq x \leq 1, & 0 < t, \\ u(x,0) = f(x), & u(1,t) = g(t). & \end{array} \tag{31}$$

To discretize (31) in space we use the matrix vector multiplication method, that is at the collocation points we set

$$u_x = D\mathbf{u}, \tag{32}$$

where D is the differentiation matrix based on the collocation points (28). Hence, as v is an approximation to u from (31) and (32) at the collocation points, we have

$$v_t = D\mathbf{v}, \tag{33}$$

that is

$$(v_t)_k = (D\mathbf{v})_k$$

or

$$v_t(x_k,t) = \sum_{j=0}^{N} D_{kj} v(x_j,t), \quad k = 0, \ldots, N. \tag{34}$$

Sometimes, (34) is called semi-discrete approximation. If we set

$$v_t(x_k, t) = v'(x_k, t),$$

then (34) converts into

$$v'(x_k, t) = \sum_{j=0}^{N} D_{kj} v(x_j, t), \quad k = 0, \ldots, N. \tag{35}$$

Equations (35) form a system of ordinary differential equations in time. Therefore, to advance the solution in time, we can use any ODE solver such as the midpoint Adams-Bashforth or Runge-Kutta methods. Authors in [21] use the standard fourth-order Runge-Kutta formula, because it is closer to the high spatial accuracy of the spectral method than $second-order$ formulas. An alternative method is to use a spectral method in time [22]. However, it is difficult to generalize such methods to nonlinear problems, while Runge-Kutta methods extend trivially to nonlinear problems. Hence, following [21] we also use the fourth-order Runge-Kutta method to advance the solution in time.

As Runge-Kutta method is an explicit method, it is easy to impose boundary conditions at any stage of the algorithm. Further, because of stability considerations of explicit methods, there is a limit on the allowable time steps.

It should be noted that any differential equation in (35) is independent of the other equations. Let $v(x_k, t) = v(t)$ and $F(v, t) = \sum_{j=0}^{N} D_{kj} v(x_j, t)$, hence, for fixed k, any relation in (35) has the following form

$$\begin{cases} v'(t) = F(v, t), \\ v(t_0) = f(x), \end{cases} \tag{36}$$

where $t_0 = 0$ and $f(x)$ is a known function. Therefore, the standard fourth-order Runge-Kutta method is as follows

$$v(x_k, t_{n+1}) = v(x_k, t_n) + \tfrac{1}{6}(z_1 + 2z_2 + 2z_3 + z_4), \quad \begin{array}{l} k = 1, \ldots, N, \\ n = 0, 1, \ldots, \end{array}$$

where

$$
\begin{aligned}
z_1 &= \Delta t\, F(v(t_n), t_n) = \Delta t\, F(v_n, t_n) = \Delta t \sum_{j=0}^{N} D_{kj} v(x_j, t_n),\\
z_2 &= \Delta t\, F(v_n + \frac{z_1}{2}, t_n + \frac{\Delta t}{2}) = \Delta t \sum_{j=0}^{N} D_{kj}[v_n + \frac{z_1}{2}]\\
&= \Delta t \sum_{j=0}^{N} D_{kj}[v(x_j, t_n) + \frac{z_1}{2}],\\
z_3 &= \Delta t\, F(v_n + \frac{z_2}{2}, t_n + \frac{\Delta t}{2}) = \Delta t \sum_{j=0}^{N} D_{kj}[v_n + \frac{z_2}{2}]\\
&= \Delta t \sum_{j=0}^{N} D_{kj}[v(x_j, t_n) + \frac{z_2}{2}],\\
z_4 &= \Delta t\, F(v_n + z_3, t_n + \Delta t) = \Delta t \sum_{j=0}^{N} D_{kj}[v_n + z_3]\\
&= \Delta t \sum_{j=0}^{N} D_{kj}[v(x_j, t_n) + z_3],
\end{aligned}
$$

the initial condition is $v(x, 0) = f(x)$ and the boundary condition is $v(1, t) = g(t)$.

Solomonoff and Turkel [21] show that for stability, Δt must be chosen as $\Delta t = O(\frac{1}{N^2})$. They obtained some solutions for different values of Δt which satisfies this stability condition. The best result is for $\Delta t = \frac{\sqrt{2}}{N^2}$, which we also take to be the time step in this paper. Also, they took 1 as the final value for t, and we do likewise. Further, in order to measure the accuracy of the approximation, they chose $u(x, t) = f(x + t)$ for $f(x) = \sin(\pi x)$. Hence, the approximation can be compared pointwise with the analytic solution. The boundary data is then given by $g(t) = f(1 + t)$.

3.1. Preconditioning

In order to solve problem (31) by preconditioning method of Section 2, we substitute $v(x_j,t)$ by $v(x_j,t)-v(x_k,t)$ in the following relation

$$v_t(x_k,t)=\sum_{j=0}^{N} D_{kj}v(x_j,t). \tag{37}$$

Then, the preconditioned system has the following form

$$v_t(x_k,t)=\sum_{j=0}^{N} D_{kj}[v(x_j,t)-v(x_k,t)], \quad k=0,\dots,N. \tag{38}$$

We solve (38) by the fourth-order Runge-Kutta method. The Runge-Kutta formula for solving system (38) is similar to (36), if we replace $F(v,t)$ by

$$F(v,t)=\sum_{j=0}^{N} D_{kj}[v(x_j,t)-v(x_k,t)].$$

3.2. Domain Decomposition

We can write problem (31) as follows:

$$\begin{cases} u_t-u_x=0, & -1\le x\le 1, \quad 0<t, \\ u(x,0)=f(x), & u(1,t)=g(t). \end{cases} \tag{39}$$

Most collocation versions of spectral domain decomposition methods use variations or extensions of the patching technique originally suggested by Orszag [18]. Suppose that interval $\Omega=(a,b)$ is broken into two subdomains, $\Omega_1=(y_0,y_1)$ and $\Omega_2=(y_1,y_2)$, where $y_0=a$ and $y_2=b$. The solution to a given differential equation on Ω is denoted by u. Its restrictions to $\bar{\Omega}_1$ and $\bar{\Omega}_2$ are denoted by u_1 and u_2, respectively. The approximate solution on Ω is denoted by u^N and its restrictions to $\bar{\Omega}_1$ and $\bar{\Omega}_2$ by u_1^N and u_2^N, respectively. The collocation points in $\bar{\Omega}_1$ are denoted by $x_j^{(1)}$, $j=0,\dots,N$ and those in $\bar{\Omega}_2$ by $x_j^{(2)}$, $j=0,\dots,N$, where N is a known integer.

A patching method for linear problem (39) is as follows

$$\begin{aligned}
&\frac{\partial u_1^N}{\partial t} - \frac{\partial u_1^N}{\partial x}\Big|_{x=x_j^{(1)}} = 0, \quad j = 0, \ldots, N-1,\\
&\frac{\partial u_2^N}{\partial t} - \frac{\partial u_2^N}{\partial x}\Big|_{x=x_j^{(2)}} = 0, \quad j = 1, \ldots, N-1,\\
&u_2^N(y_2, t) = g(t),\\
&u_1^N(y_1, t) = u_2^N(y_1, t),\\
&\frac{\partial u_1^N}{\partial x}(y_1, t) = \frac{\partial u_1^N}{\partial x}(y_1, t)
\end{aligned}$$

(see [8]).

Case (ii). The second test problem in [21] setting $S(x) = -x$, is:

$$\begin{aligned}
&u_t = -xu_x, \qquad -1 \le x \le 1,\\
&u(x, 0) = f(x).
\end{aligned}$$

Solomonoff and Turkel [21] do not specify boundary condition at boundary of the domain. For this problem the characteristics face outwards at the boundaries, so boundary conditions are not imposed. In this problem the solution is given by

$$u(x, t) = f(xe^{-t}), \quad \text{for } f(x) = \sin \pi x.$$

They also show for the fourth-order Runge-Kutta scheme, time step must be $\Delta t = O(\frac{1}{N})$ and the best result is obtained when $\Delta t = \frac{\sqrt{2}}{8N}$. They also set final value of t equal to 10. In this paper, we consider these values for Δt and t.

3.3. Numerical Results

As in [9], these two problems are solved by four methods, namely,

1) Ordinary pseudospectral method (PSU).

2) Pseudospectral method along with preconditioning scheme (PSUPRE).

3) Ordinary pseudospectral method coupled with domain decomposition approach (DDPSU).

4) Preconditioning pseudospectral method along with domain decomposition method (DDPRE).

For each test problem the natural logarithm of the error is sketched as a function of the natural logarithm of the number of grid points. In domain

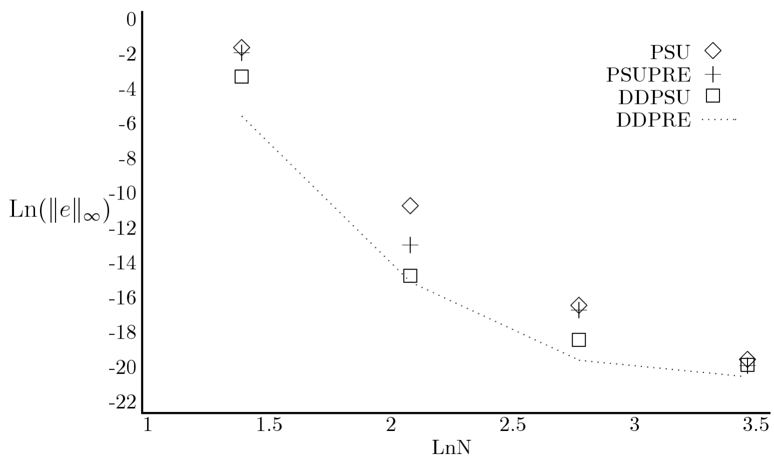

Figure 1: $\mathrm{Ln}(\|e\|_\infty)$ of error for the first test problem

decomposition approach a P-point, M-panel scheme has been used, in which $MP = N$. The error of this method is compared with an N-node scheme of the other algorithms.

Figure 1 and Figure 2 show the error curves. Both figures show the new algorithms work very well.

4. New Preconditioning Methods

As stated in Section 2, for $k \neq j$ as we have

$$|D_{kj}| = \frac{c_k}{2c_j} \frac{1}{|\sin((k+j)\frac{\pi}{2N})\sin((k-j)\frac{\pi}{2N})|}, \tag{40}$$

the entries of the derivative matrix D, with large absolute value, are on a band near main diagonal. In fact the values of $|D_{kj}|$ in (40) can be very large when k is near j. Particularly, if $|k-j| = 1$ the value of $|D_{kj}|$ is very large. This means large elements of the derivative matrix, in absolute value, are the following elements:

$$\begin{aligned} &|D_{k,k-1}|, \quad k = 1, \ldots, N, \\ &|D_{k,k+1}|, \quad k = 0, \ldots, N-1. \end{aligned}$$

Therefore, if these elements are multiplied by zero, the influence of these big elements in roundoff error, will be vanished. We define the following functions:

$$\begin{aligned} h_-(x) &= u(x) - u(x_{k-1}), \quad k = 1, \ldots, N, \\ h_+(x) &= u(x) - u(x_{k+1}), \quad k = 0, \ldots, N-1, \end{aligned}$$

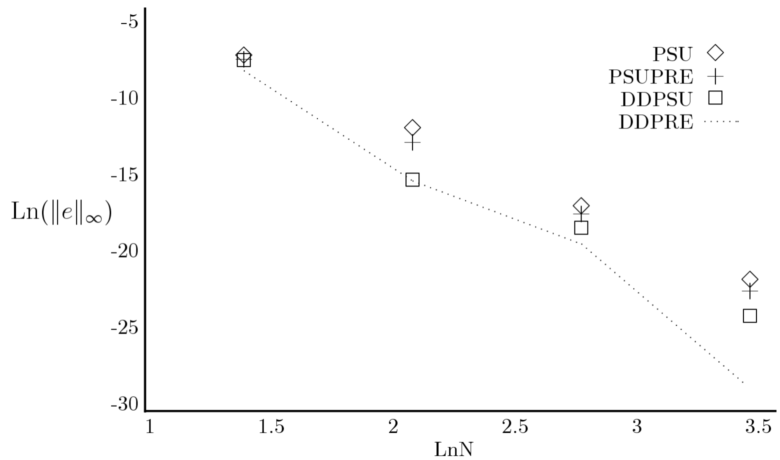

Figure 2: $\mathrm{Ln}(\|e\|_\infty)$ of error for the second test problem

where $x_k = \cos\frac{k\pi}{N}$, note that $u'(x) = h'_-(x) = h'_+(x)$. Similar (13), from these functions we have

$$h'_-(x_k) = \sum_{j=0}^{N} D_{kj} h_-(x_j), \quad k = 1, \ldots, N, \tag{41}$$

$$h'_+(x_k) = \sum_{j=0}^{N} D_{kj} h_+(x_j), \quad k = 0, \ldots, N-1, \tag{42}$$

or

$$u'(x_k) = \sum_{j=0}^{N} D_{kj}(u(x_j) - u(x_{k-1})), \quad k = 1, \ldots, N, \tag{43}$$

and

$$u'(x_k) = \sum_{j=0}^{N} D_{kj}(u(x_j) - u(x_{k+1})), \quad k = 0, \ldots, N-1. \tag{44}$$

If we want to use (43) we need to compute $u'(x_0)$ separately. Similarly if we want to use (44) we need to compute $u'(x_N)$ separately. Therefore, we propose the following formulas:

$$\begin{cases} u'(x_k) = \sum_{j=0}^{N} D_{kj}(u(x_j) - u(x_{k-1})), & k = 1, \ldots, N, \\ u'(x_0) = \sum_{j=0}^{N} D_{0j}(u(x_j) - u(x_1)), & \end{cases} \tag{45}$$

and

$$\begin{cases} u'(x_k) = \sum_{j=0}^{N} D_{kj}(u(x_j) - u(x_{k+1})), & k = 0, \ldots, N-1, \\ u'(x_N) = \sum_{j=0}^{N} D_{Nj}(u(x_j) - u(x_{N-1})). & \end{cases} \tag{46}$$

We call the preconditionings in (45) and (46) *left* and *right* preconditionings, respectively. The effect of these preconditionings are shown on test functions in [10], namely, $\sin 2x$, $\cos 2x$ and e^{3x}. The numerical results are reported in Table 1, in which the value of maximum error of the first order derivative as a function of number of collocation points, N, is demonstrated.

	left prec.	right prec.	left prec.	right prec.	left prec.	right prec.
N	$\sin(2x)$		$\cos(2x)$		$\exp(3x)$	
16	3.39395,-13	3.38951,-13	4.01901,-14	3.37035,-14	3.42231,-10	4.76774,-10
32	4.10783,-15	1.55431,-15	8.43769,-15	8.65974,-15	3.55271,-14	6.39488,-14
64	9.65894,-15	5.55112,-15	2.04281,-14	6.43929,-15	1.20792,-13	1.42109,-13
128	8.10463,-15	6.21725,-15	1.88738,-14	1.44329,-14	1.27898,-13	1.77636,-13
256	8.61533,-14	2.89768,-14	1.87406,-13	5.95080,-14	5.82645,-13	5.75540,-13
512	8.31557,-14	2.68674,-14	1.70086,-13	6.01741,-14	1.12976,-12	1.20082,-12

Table 1: Absolute maximum error of the first order derivatives of three test functions, by left and right preconditionings

By comparison of results in Table 1 and similar results of the preconditioning in [10], it can be seen that these new preconditionings are better than the old one in [10]. We will report the application of these new preconditionings on ODEs and PDEs in a future paper.

Acknowledgements

The author would like to thank Professor Richard Baltensperger for valuable comments and useful communications on this work.

References

[1] R. Baltensperger, J.P. Berrut, The errors in calculating the pseudospetral differentiation matrices for Chebvshev-Gauss-Lobatto points, *Comput. Math. Appl.*, **37** (1999), 41-48.

[2] R. Baltensperger, J.P. Berrut, Errata to: "The errors in calculating the pseudospetral differentiation matrices for Chebvshev-Gauss-Lobatto points, *Comput. Math. Appl.*, **37**, No. 1 (1999), 41-48", *Comput. Math. Appl.*, **38** (1999), 119.

[3] R. Baltensperger, M.R. Trummer, Spectral differencing with a twist, *SIAM J. of Sci. Comp.*, **24**, No. 5 (2003), 1465-1487.

[4] A. Bayliss, A. Class, B. Matkowsky, Roundoff error in computing derivatives using the Chebyshev differentiation matrix, *J. Comput. Phys.*, **116** (1995), 380-383.

[5] C. Bernardi, Y. Maday, Some spectral approximations of one-dimensional fourth-order problems, Progress in: *Approximation Theory* (Ed-s: P. Nevai, A. Pinkus) Academic press, Boston (1991).

[6] J.P. Boyd, *Chebyshev and Fourier Spectral Methods,* Lecture Notes in Engineering, **49**, Springer-Verlag, Berlin (1989).

[7] K.S. Breuer, R.S. Everson, On the errors incurred calculating derivatives using Chebyshev polynomials, *J. Comput. Physics*, **99**, 56-67 (1992).

[8] C. Canuto, A. Quarteroni, M.Y. Hussaini, T. Zang, *Spectral Method in Fluid Mechanics,* Springer-Verlag, New York (1988).

[9] M.T. Darvishi, New algorithms for solving ODEs by pseudospectral method, *Korean J. Comput. and Appl. Math.*, **7**, No. 2 (2000), 319-331.

[10] M.T. Darvishi, F. Ghoreishi, Error reduction for higher derivatives of Chebyshev collocation method using preconditioning and domain decomposition, *Korean J. Comput. and Appl. Math.*, **6**, No. 2 (1999), 421-435.

[11] M.T. Darvishi, F. Ghoreishi, Multidomain and preconditioning schemes for computing higher derivatives, *Far East J. of Mathematical Sciences*, **1**, No. 2 (1999), 297-316.

[12] W.S. Don, A. Solomonoff, Accuracy and speed in computing the Chebyshev collocation derivative, *SIAM J. of Sci. Comput.*, **16**, No. 4 (1995), 1253-1268.

[13] W.S. Don, A. Solomonoff, Accuracy enhancement for higher derivatives using Chebyshev collocation and a mapping technique, *SIAM J. of Sci. Comput.*, **18**, No. 4 (1997), 1040-1055.

[14] D. Funaro, *Polynomial Approximation of Differential Equations,* Springer-Verlag, Berlin (1992).

[15] D. Gottlieb, The stability of pseudospectral-Chebyshev methods, *Math. Comput.*, **36**, No. 153 (1981), 107-118.

[16] D. Gottlieb, M.Y. Hussaini, S.A. Orszag, Theory and application of spectral methods, In: *Spectral Methods for Partial Differential Equations* (Ed-s: R.G. Voigt, D. Gottlieb, M.Y. Hussaini), SIAM-CBMS, Philadelphia (1984), 1-54.

[17] D. Gottlieb, S.A. Orszag, E. Turkel, Stability of pseudospectral and finite-difference methods for variable coefficient problems, *Math. Comput.*, **37**, No. 156 (1981), 293-305.

[18] S.A. Orszag, Spectral methods for problems in complex geometries, *J. Comput. Phys.*, **37** (1980), 70-92.

[19] A. Ralston, *A First Course in Numerical Analysis*, McGraw-Hill (1965).

[20] E.E. Rothman, Reducing round-off error in Chebyshev pseudospectral computations, In: *High Performance Computing II*, (Ed-s: M. Durant, F. El Dabaghi), Elsevier-North Holland, Amsterdam (1991), 423-439.

[21] A. Solomonoff, E. Turkel, Global properties of pseudospectral methods, *J. Comput. Phys.*, **81** (1989), 239-276.

[22] H. Tal-Ezer, Spectral methods in time for hyperbolic equations, *SIAM J. Numer. Anal.*, **23**, No. 1 (1986), 11-26.

[23] T. Tang, M.R. Trummer, Boundary layer resolving pseudospetral methods for singular perturbation problems, *SIAM J. Sci. Comput. Phys.*, **17** (1996), 430-438.

[24] L.N. Trefethen, *Spectral Methods in MATLAB,* SIAM, Philadelphia (2000).