J. Numer. Math., Vol. 0, No. 0, pp. 1–22 (2006)
 Prepared using jnm.sty *[Version: 02.02.2002 v1.2]*

Reuse of Standard Preconditioners for Higher–Order Time Discretizations of Parabolic PDEs

Kent–Andre Mardal,* and Trygve Kastberg Nilssen†

29th March 2006

Abstract — In this work we study a preconditioned iterative method for some higher–order time discretizations of linear parabolic partial differential equations. We use the Padé approximations of the exponential function to discretize in time and show that standard solution algorithms for lower–order time discretization schemes, such as Crank–Nicolson and implicit Euler, can be reused as preconditioners for the arising linear system. The proposed preconditioner is order optimal with respect to the discretization parameters.

Keywords: preconditioning, higher–order time discretizations, parabolic PDEs

1. INTRODUCTION

In this paper we study preconditioners for higher–order time discretizations of linear parabolic partial differential equations. Optimal solution algorithms for lower–order discretizations such as the backward Euler or the Crank–Nicolson schemes are well known, see e.g., [1,11,13], and the main point here is that these algorithms can be reused as preconditioners for higher–order time discretizations.

The ”same” reasoning has been used with success for higher–order spatial discretizations. Already in 1985, preconditioners based on lower–order finite difference or finite element discretizations were reused for the spectral element discretization of the same equation, [3,5]. There are many later works on this subject. Although higher–order time discretizations have been extensively studied cf. [13], preconditioning methods for the arising linear systems have not been investigated much. Only BDF and DIRK methods provide higher–order accuracy in time by solving a linear system with the same matrix as in lower–order methods. The problem with BDF methods is that they can not handle temporal adaptivity which is crusial in many applications. DIRK methods are not stable enough for large time stepping in parabolic equations. To our knowledge only a few authors have adressed solution algorithms for other higher–order time discretizations [2,7,9,10].

The outline of the paper is as follows. The main idea is introduced in Section 2. Then the needed properties of Padé approximation and preconditioning are briefly

*Simula Research Laboratoy, P.O.Box 134, 1325 Lysaker, Norway

†Simula Research Laboratoy, P.O.Box 134, 1325 Lysaker, Norway (now at Scandpower Petroleum Technology, P.O.Box 113, Gåsevikvn. 2-4, 2027 Kjeller, Norway)

reviewed in Section 3 and Section 4, respectively. Section 5 describes the proposed preconditioner in detail, and proves order optimality with respect to the discretization parameters. In Section 6 we give a more precise bound on the condition number of the preconditioned system.

2. PRELIMINARIES

In this paper we consider a solution algorithm for discretizations of linear parabolic partial differential equations. For simplicity, but without loss of generality, we start with the homogenous model problem

$$\begin{aligned} \frac{\partial u}{\partial t} &= \Delta u, \quad \text{in } \Omega, t \in (0,T), \\ u &= 0, \quad \text{on } \partial\Omega, t \in (0,T), \\ u &= u_0, \quad \text{in } \Omega, t = 0. \end{aligned}$$

This equation is discretized in space, resulting in the following ODE system to be solved,

$$\frac{du_{h,p}}{dt} = A_{h,p}u_{h,p}, \quad t \in (0,T), \tag{2.1}$$

$$u_{h,p} = u_0, \quad t = 0, \tag{2.2}$$

where $A_{h,p}$ is the discrete Laplacian (a matrix), $u_{h,p}$ is the unknown (a vector) and h is characteristic for the mesh size and p is the polynomial degree. In the following we will drop the subscripts h and p.

It is known that the linear ODE system (2.1)-(2.2) can be solved with any order of accuracy with the following time stepping scheme

$$Q_{kj}(\Delta tA)u^n = P_{kj}(\Delta tA)u^{n-1}, \tag{2.3}$$

where Δt is the time stepping parameter and the two polynomials Q_{kj} and P_{kj} are the $(k,j)-$ Padé approximation to the exponential function.

We will come back to the specific structure of Q_{kj} and P_{kj} later, but remark that systems on the following form has to be solved at each time step,

$$(I - q_1\Delta tA + q_2\Delta t^2A^2 - \cdots + (-1)^j q_j \Delta t^j A^j)u^n = b.$$

Here j is the order of the polynomial. Instead of considering a preconditioner based on the polynomial Q_{kj} we want to reuse standard solution algorithms for lower–order time discretizations of the equation,

$$(I - \Delta tA)u = b. \tag{2.4}$$

Such algorithms have been studied extensively and order optimal algorithms have been found for most spatial discretization methods. Hence, we do not assume anything on the spatial discretization. Let $B_{\Delta t}$ be defined as

$$B_{\Delta t} = (I - \Delta tA)^{-1},$$

and we assume that the evaluation of $B_{\Delta t}$ on a vector is an order optimal process. Then we will demonstrate that

$$B_\tau^j$$

is a good preconditioner for

$$(I - \Delta t q_1 A + q_2 \Delta t^2 A^2 - \cdots + (-1)^j q_j \Delta t^j A^j),$$

where $\tau = \sqrt[j]{q_j}\Delta t$ and q_i are the coefficients in the Padé approximation described in the next section.

The proposed preconditioner works just as well for the inhomogeneous parabolic equations. In fact, c.f. [13], only the right–hand side of (2.3) needs to be altered to account for the inhomogeneous case.

3. PADÉ APPROXIMATION

Here we briefly review the basics of the Padé approximation c.f. [13]. The polynomials are given by:

$$\begin{aligned} P_{kj}(\Delta tA) &= \sum_{i=0}^{k} \left(\begin{array}{c} k \\ i \end{array} \right) \frac{(k+j-i)!}{(k+j)!} (\Delta tA)^i, \\ Q_{kj}(\Delta tA) &= P_{jk}(-\Delta tA). \end{aligned}$$

Notice that P_{kj} and Q_{kj} are polynomials of order k and j respectively. It can be shown that

$$e^{\Delta tA} - Q_{kj}^{-1}(\Delta tA) P_{kj}(\Delta tA) = \frac{(-1)^j j!k!}{(j+k)!(j+k+1)!} (\Delta tA)^{j+k+1} + \mathscr{O}\left((\Delta tA)^{j+k+2} \right),$$

which means that (2.3) is locally $j+k+1$ order accurate and globally $j+k$ order accurate. For a given k we need to choose j such that $k \leqslant j \leqslant k+2$ for the method to be A–stable. We will only consider A–stable Padé approximations in this work. For more information about Padé approximations of the exponential function and stability requirements, see [8,13].

4. PRECONDITIONING

Here we briefly review the basics of preconditioning adapted to the given model problem (2.3). The matrix $Q_{kj}(\Delta tA)$ is symmetric and positive definite (SPD), because $-A$ is. We also make a SPD preconditioner, R^j. The method of choice for SPD problems with SPD preconditioners is the preconditioned Conjugate Gradient method (PCG). Given that R^j and $Q_{kj}(\Delta tA)$ are spectrally equivalent, that is

$$c_0(R^j u, u) \leqslant (Q_{kj}(\Delta tA)u, u) \leqslant c_1(R^j u, u), \quad \forall u,$$

then the condition number of $R^{-j}Q_{kj}(\Delta tA)$, $\varkappa(R^{-j}Q_{kj}(\Delta tA)) \leqslant \frac{c_1}{c_0}$. We will show that c_0 and c_1 are independent of Δt and A. PCG will then converge to a fixed convergence criterion in a number of iterations which is bounded independent of Δt and A.

The number of floating point operations needed by the matrix–vector product of a polynomial in A, $P(A)$, is $\mathcal{O}(jN)$, where N is the number of non zeroes in the matrix and j is the order of the time discretization polynomial. To see this, let $P(A) = \sum_{i=0}^{j} q_i A^i$, then

$$P(A)u = \sum_{i=0}^{k} q_i A^i u = (q_0 + A(q_1 + A(q_2 + A(\cdots))))u.$$

The evaluation and storage of the preconditioner R^{-1} is assumed to be $\mathcal{O}(N)$, and therefore the evaluation of R^{-j} is $\mathcal{O}(jN)$.

5. THE OPTIMAL PRECONDITIONER

In this section we will see that the reuse of lower–order (standard) solution algorithms result in a preconditioner which is independent of Δt and A. The analysis, when using certain lower–order preconditioners, reduces to the consideration of polynomials. We will prove the desired properties analytically and supply with numerical experiments.

As mentioned earlier, we will reuse solution algorithms for

$$R = (I - c\Delta tA).$$

In fact, the proposed preconditioner is of the form

$$R^j = (I - c\Delta tA)^j,$$

where c is determined such that the highest order term of $R^j(\Delta tA)$ equals the highest order term of $Q_{kj}(\Delta tA)$. This is done by choosing

$$R_{kj}(\Delta tA) = \left(I - \sqrt[j]{\frac{j!}{(j+k)!}}\Delta tA \right)^j.$$

Notice also that the lowest order terms of $Q_{kj}(\Delta tA)$ and $R_{kj}(\Delta tA)$ are equal.

We introduce the notation

$$r_i = \binom{j}{i} \left(\frac{k!}{(j+k)!} \right)^{\frac{i}{j}} \quad \text{and} \quad q_i = \binom{j}{i} \frac{(j+k-i)!}{(j+k)!},$$

which gives

$$R_{kj}(\Delta tA) = \sum_{i=1}^{j} r_i(-\Delta tA)^i \quad \text{and} \quad Q_{kj}(\Delta tA) = \sum_{i=1}^{j} q_i(-\Delta tA)^i.$$

We have dropped the subscripts j and k in both r_i and q_i for notational simplicity.

The preconditioned system then reads,

$$R_{kj}^{-1}(\Delta tA)Q_{kj}(\Delta tA)u^n = R_{kj}^{-1}(\Delta tA)P_{kj}(\Delta tA)u^{n-1},$$

where $R_{kj}^{-1}(\Delta tA) = (R_{kj}(\Delta tA))^{-1}$. Our first lemma shows that the two operators $R_{kj}(\Delta tA)$ and $Q_{kj}(\Delta tA)$ are spectrally equivalent independent of A and Δt, but possibly dependent on k and j.

Lemma 5.1. *The polynomials R_{kj} and Q_{kj} are spectrally equivalent independent of A and Δt,*

$$c_0(Q_{kj}(\Delta tA)v,v) \leqslant (R_{kj}(\Delta tA)v,v) \leqslant c_1(Q_{kj}(\Delta tA)v,v), \quad \forall v. \tag{5.1}$$

Moreover,

$$c_1 \leqslant \max_{i\in[0,j]} \frac{r_i}{q_i} \quad \textit{and} \quad c_0 = 1.$$

Proof. We start by using the fact that A and the polynomials of A have the same eigenvectors. This leads to the following eigenvalues of R_{kj} and Q_{kj},

$$\begin{aligned} Q_{kj}(\Delta tA)v_\ell &= \sum_{i=1}^{j} q_i(-\Delta tA)^i v_\ell = \left(\sum_{i=1}^{j} q_i(\Delta t\lambda_\ell)^i\right) v_\ell = Q_{kj}(-\Delta t\lambda_\ell)v_\ell, \\ R_{kj}(\Delta tA)v_\ell &= \sum_{i=1}^{j} r_i(-\Delta tA)^i v_\ell = \left(\sum_{i=1}^{j} r_i(\Delta t\lambda_\ell)^i\right) v_\ell = R_{kj}(-\Delta t\lambda_\ell)v_\ell, \end{aligned}$$

where λ_ℓ is the eigenvalue of $-A$ that corresponds to the eigenvector v_ℓ.

A straightforward calculation shows that the spectral equivalence (5.1) can be stated in terms of the eigenvalues of R_{kj} and Q_{kj}

$$\begin{array}{rcl} c_0(Q_{kj}(\Delta tA)v,v) & \leqslant (R_{kj}(\Delta tA)v,v) \leqslant & c_1(Q_{kj}(\Delta tA)v,v), \quad \forall v, \\ & \Updownarrow & \\ c_0(Q_{kj}(\Delta tA)v_\ell,v_\ell) & \leqslant (R_{kj}(\Delta tA)v_\ell,v_\ell) \leqslant & c_1(Q_{kj}(\Delta tA)v_\ell,v_\ell), \quad \forall \ell, \\ & \Updownarrow & \\ c_0 Q_{kj}(-\Delta t\lambda_\ell) & \leqslant R_{kj}(-\Delta t\lambda_\ell) \leqslant & c_1 Q_{kj}(-\Delta t\lambda_\ell), \quad \forall \ell, \end{array}$$

where $\Delta t\lambda_\ell \in (0,\infty)$.

Let $x \in (0,\infty)$, then we need to consider

$$c_0 \sum_{i=1}^{j} q_i x^i \leqslant \sum_{i=1}^{j} r_i x^i \leqslant c_1 \sum_{i=1}^{j} q_i x^i.$$

Because x, $\{r_i\}$ and $\{q_i\}$ are positive we can check each term separately,

$$c_{0,i} q_i x^i \leqslant r_i x^i \leqslant c_{1,i} q_i x^i,$$

and take the minimum and maximum of $c_{0,i}$ and $c_{1,i}$ to find the estimates for c_0 and c_1,

$$c_0 = \min_{i\in[0,j]} c_{0,i} = \min_{i\in[0,j]} \frac{r_i}{q_i} \quad \text{and} \quad c_1 = \max_{i\in[0,j]} c_{1,i} = \max_{i\in[0,j]} \frac{r_i}{q_i}.$$

In the following we show that $\frac{r_i}{q_i} \geqslant 1$. We have that

$$\begin{aligned}
\frac{r_i}{q_i} &= \frac{\left(\frac{k!}{(j+k)!}\right)^{\frac{i}{j}}}{\frac{(j+k-i)!}{(j+k)!}} \\
&= \frac{(j+k)(j+k-1)\cdots(j+k-i+1)}{((j+k)(j+k-1)\cdots(k+1))^{\frac{i}{j}}} \\
&= \left(\frac{(j+k)^j(j+k-1)^j\cdots(j+k-i+1)^j}{(j+k)^i(j+k-1)^i\cdots(k+1)^i}\right)^{\frac{1}{j}}.
\end{aligned}$$

There are i times j terms in both the numerator and denominator and these are ordered such that they decrease towards the right. The crucial point is that when numbered in this way, the l'th term in the numerator is always larger or equal to the l'th term in the denominator. Therefore, $c_0 \geqslant 1$. Equality is obtained because $r_0/q_0 = 1$ (and $r_j/q_j = 1$). Hence, $c_0 = 1$ and $c_1 \leqslant \max_i \frac{r_i}{q_i}$ and the proof is complete. □

Remark 5.1. Lemma 5.1 gives that the condition number of the preconditioned system is bounded by

$$\varkappa\left(R_{kj}^{-1}(\Delta t A) Q_{kj}(\Delta t A)\right) < \max_{i\in[0,j]} \frac{r_i}{q_i}.$$

Remark 5.2. Note that there are two bounds in Lemma 5.1:

$$\varkappa\left(R_{kj}^{-1}(\Delta t A) Q_{kj}(\Delta t A)\right) \leqslant \max_{x<0} \frac{R_{kj}(x)}{Q_{kj}(x)}, \tag{5.2}$$

$$\max_{x<0} \frac{R_{kj}(x)}{Q_{kj}(x)} < \max_{i\in[0,j]} \frac{r_i}{q_i}. \tag{5.3}$$

In (5.2) the inequality is sharp, but the inequality in (5.3) is not sharp. However, the sharpness of (5.3) is increasing with increasing values of j and k.

$j\backslash k$	j	$j-1$	$j-2$
2	1.07	1.10	1.17
3	1.16	1.20	1.28
4	1.26	1.31	1.40
5	1.37	1.43	1.52
6	1.49	1.56	1.66
7	1.62	1.70	1.81
8	1.76	1.85	1.97
9	1.92	2.02	2.14
10	2.08	2.20	2.34

Table 1. Upper bound on the condition number for various values of j and k, based on the bound $\varkappa\left(R_{kj}^{-1}(\Delta tA)Q_{kj}(\Delta tA)\right) \leqslant \max_{x<0} \frac{R_{kj}(x)}{Q_{kj}(x)}$.

5.1. Numerical results

In this subsection we show what Lemma 5.1 means in practice, and demonstrate the sharpnes of the bounds. In Table 1 we have shown a bound on the condition number of the preconditioned system based on

$$\varkappa\left(R_{kj}^{-1}(\Delta tA)Q_{kj}(\Delta tA)\right) \leqslant \max_{x<0} \frac{R_{kj}(x)}{Q_{kj}(x)},$$

where the polynomial degrees range from 2 to 10. The condition number of the preconditioned system seems to be increasing slightly, but even for the 10th order polynomial the condition number seems acceptable.

To test the sharpness of the bound (5.2) we have made an 1D example with a standard second order finite difference approximation of the Laplace operator, i.e.

$$A = \frac{1}{h^2}\text{tridiag}(1,-2,1). \tag{5.4}$$

In Table 2 we choose $j = k = 4$ and show

$$E_{44} = \max_{x<0} \frac{R_{44}(x)}{Q_{44}(x)} - \varkappa\left(R_{44}^{-1}(\Delta tA)Q_{44}(\Delta tA)\right),$$

for various values of h and Δt. From Lemma 5.1 we know that E_{44} is a non–negative number. We observe that E_{44} seems to approach zero as h and Δt approach zero.

$h=\Delta t$	$\max_{x<0}\frac{R_{44}(x)}{Q_{44}(x)}$	$\varkappa$	E_{44}
$1/125$	1.2584	1.2461	0.0123
$1/250$	1.2584	1.2523	0.0061
$1/500$	1.2584	1.2551	0.0033
$1/1000$	1.2584	1.2568	0.0015

Table 2. An illustration of the sharpness of the inequality in (5.2).

$j\backslash k$	j	$j-1$	$j-2$
2	1.15	1.22	1.41
3	1.23	1.30	1.44
4	1.37	1.45	1.58
5	1.48	1.57	1.70
6	1.62	1.72	1.85
7	1.76	1.86.	2.01
8	1.92	2.03	2.18
9	2.09	2.22	2.38
10	2.27	2.41	2.58

Table 3. Upper bound on the condition number for various values of j and k, based on the bound $\varkappa\left(R_{kj}^{-1}(\Delta tA)Q_{kj}(\Delta tA)\right) < \max_{i\in[0,j]}\frac{r_i}{q_i}$.

In Table 3 we show a bound on the condition number of the preconditioned system based on the bound

$$\varkappa\left(R_{kj}^{-1}(\Delta tA)Q_{kj}(\Delta tA)\right) < \max_i \frac{r_i}{q_i},$$

where the polynomial degrees range from 2 to 10. These bounds are not sharp, but the sharpness seems to increase with increased values of j and k.

The numerical experiments motivate the following lemma, which will be proved in the next section.

Lemma 5.2. *An upper bound on the condition number of the preconditioned*

system is given by

$$\varkappa\left(R_{kj}^{-1}(\Delta tA)Q_{kj}(\Delta tA)\right) < \gamma\cdot(1.09)^j, \tag{5.5}$$

where

$$\begin{aligned}
&\gamma = 0.98 \quad \textit{for} \quad j = k,\\
&\gamma = 1.11 \quad \textit{for} \quad j = k+1,\\
&\gamma = 1.62 \quad \textit{for} \quad j = k+2.
\end{aligned}$$

The constant 1.09 is an approximation of $\frac{1}{4}e^{\frac{4}{e}}$*, which is the precise constant.*

5.2. Numerical tests for the Padé approximation scheme

To see the benefit of using higher–order schemes we now show some numerical experiments for the heat equation in one dimension with homogeneous Dirichlet conditions. We solve

$$\begin{aligned}
u_t &= u_{xx}, \quad x\in(0,1), \quad t\in(0,T) && (5.6)\\
u(0,t) &= u(1,t) = 0, \quad t\in(0,T) && (5.7)\\
u(x,0) &= \sin(\pi x), \quad x\in(0,1). && (5.8)
\end{aligned}$$

with $T = 0.2$. We use $M = \frac{T}{\Delta t}$ temporal discretization points and $N = \frac{1}{h}$ spatial discretization points.

In Table 4 we show the relative L^2 error, i.e.

$$\frac{\|u^M - u(T)\|_{L^2}}{\|u(T)\|_{L^2}},$$

when an implicit Euler scheme is used for the time stepping, i.e.

$$u^n = (I-\Delta tA)^{-1}u^{n-1}.$$

Here A is a standard second order finite difference approximation of the Laplace operator, see (5.4). The calculations for each time level cost about $5N$ floating point operations for this scheme. The total work is therefore about $5NM$ multiplications/divisions.

In Table 5 we show the same as in Table 4 for the Crank–Nicolson scheme, i.e.

$$u^n = \left(I-\frac{\Delta t}{2}A\right)^{-1}\left(I+\frac{\Delta t}{2}A\right)u^{n-1}.$$

The work load for one time step is here about $8N$, and the total work for this method is therefore about $8NM$. We mention that the fact that the Crank–Nicolson scheme

$M\backslash N$	40	80	160	320	640
2 560	1.72e-3	1.00e-3	8.23e-4	7.76e-4	7.64e-4
5 120	1.34e-3	6.27e-4	4.43e-4	3.96e-4	3.84e-4
10 240	1.15e-3	4.37e-4	2.52e-4	2.06e-4	1.94e-4
20 480	1.06e-3	3.42e-4	1.56e-4	1.10e-4	9.90e-5
40 960	1.01e-3	2.95e-4	1.10e-4	6.33e-5	5.15e-5
81 920	9.89e-4	2.71e-4	8.64e-5	3.95e-5	2.77e-5
163 840	9.77e-4	2.59e-4	7.45e-5	2.76e-5	1.58e-5
327 680	9.72e-4	2.53e-4	6.85e-5	2.17e-5	9.89e-6

Table 4. Relative L^2 error for an implicit Euler scheme in time and second order finite difference approximation in space.

$M\backslash N$	160	320	640	1 280	2 560	5 120
80	3.75e-5	8.43e-5	9.61e-5	9.91e-5	9.99e-5	1.00e-4
160	3.75e-5	9.28e-6	2.10e-5	2.40e-5	2.47e-5	2.49e-5
320	5.63e-5	9.49e-6	2.30e-6	5.26e-6	6.01e-6	6.19e-6
640	6.10e-5	1.41e-5	2.38e-6	5.75e-7	1.31e-6	1.50e-6
1 280	6.22e-5	1.53e-5	3.56e-6	5.98e-7	1.43e-7	3.29e-7
2 560	6.25e-5	1.56e-5	3.85e-6	8.91e-7	1.49e-7	3.61e-8

Table 5. Relative L^2 error for a Crank–Nicolson scheme in time and second order finite difference approximation in space.

$M\backslash N$	20	40	80	160	320
5	7.81e-5	6.79e-5	6.72e-5	6.72e-5	6.72e-5
10	1.51e-5	4.92e-6	4.22e-6	4.17e-6	4.16e-6
20	1.12e-5	1.01e-6	3.09e-7	2.63e-7	2.57e-7
40	1.09e-5	7.71e-7	6.58e-8	1.94e-8	1.63e-8
80	1.09e-5	7.56e-7	5.06e-8	4.22e-9	1.24e-9

Table 6. Relative L^2 error for the (2,2)–Padé approximation in time (fourth order accurate) and a fourth order finite difference approximation in space.

is not stiffly accurate explains why the error some places increaces with increased N while M is fixed.

In Table 6 we do the same for the (2,2)–Padé approximation in time, i.e.

$$u^n = \left(I - \frac{\Delta t}{2}A + \frac{\Delta t^2}{12}A^2\right)^{-1}\left(I + \frac{\Delta t}{2}A + \frac{\Delta t^2}{12}A^2\right)u^{n-1}.$$

Here A is a fourth order finite difference approximation of the Laplace operator based on the approximation

$$u_{xx}(x) \approx \frac{1}{12h^2}(-u(x-2h) + 16u(x-h) - 30u(x) + 16u(x+h) - u(x+2h)).$$

For each time level we must evaluate a 9–diagonal matrix and solve a linear system with a 9–diagonal matrix. The workload for evaluating the 9–diagonal matrix is about $9N$. The linear system is solved with the preconditioned Conjugate Gradient method, which in all our experiments used one iteration to reach discretization error. Thus the CG method requires one evaluation of a 9–diagonal matrix ($9N$), inverting the matrix $(I - \frac{\Delta t}{\sqrt{12}}A)$ two times ($22N$), two vector inner–products ($2N$) plus two scalar–vector products ($2N$). Before starting the time integration we form the systems $I - \frac{\Delta t}{2}A + \frac{\Delta t^2}{12}A^2$ and $I + \frac{\Delta t}{2}A + \frac{\Delta t^2}{12}A^2$ and this has a cost of $27N$. The total cost of this discretization method is therefore approximately $44NM + 27N$.

Example 5.1

Solve (5.6)–(5.8) with the accuracy requirement

$$\frac{\|u^M - u(T)\|_{L^2}}{\|u(T)\|_{L^2}} < 10^{-5}$$

for the three methods. Considering the Tables above we see that the Euler method requires a resolution with $M = 327\,680$ time levels and $N = 640$ spatial discretization points, which gives about $1.0 \cdot 10^9$ operations. Further Crank–Nicolson requires

$M = 160$ and $N = 320$, which gives about $4.1 \cdot 10^5$ operations, and the (2,2)–Padé approximation requires $M = 10$ and $N = 40$, which gives about $1.9 \cdot 10^4$ operations.

Example 5.2

Solve (5.6)–(5.8) with the accuracy requirement

$$\frac{\|u^M - u(T)\|_{L^2}}{\|u(T)\|_{L^2}} < 10^{-7}$$

for the three methods. The Euler scheme requires too many operations, Crank–Nicolson requires about $1.0 \cdot 10^8$ operations, and the (2,2)–Padé approximation requires about $1.4 \cdot 10^5$ operations. This means that the (2,2)–Padé approximation is about 750 times faster than the Crank–Nicolson scheme.

5.3. Inexact preconditioners

In this paper, we have considered the case where R was inverted exactly. In this subsection we consider an inexact preconditioner $\hat{R}$, which commutes with R, i.e., $R\hat{R} = \hat{R}R$. In this case, we have the following bound on the preconditioned system

$$\begin{aligned}
\varkappa\left(\hat{R}^{-j}Q_{kj}\right) &= \varkappa\left(\hat{R}^{-j}R^jR^{-j}Q_{kj}\right) \\
&= \|\hat{R}^{-j}R^jR^{-j}Q_{kj}\|\|(\hat{R}^{-j}R^jR^{-j}Q_{kj})^{-1}\| && (5.9)\\
&\leqslant \|\hat{R}^{-j}R^j\|\|R^{-j}Q_{kj}\|\|(\hat{R}^{-j}R^j)^{-1}\|\|(R^{-j}Q_{kj})^{-1}\| && (5.10)\\
&\leqslant \|\hat{R}^{-1}R\|^j\|R^{-j}Q_{kj}\|\|(\hat{R}^{-1}R)^{-1}\|^j\|(R^{-j}Q_{kj})^{-1}\| && (5.11)\\
&= \varkappa\left(\hat{R}^{-1}R\right)^j\varkappa\left(R^{-j}Q_{kj}\right). && (5.12)
\end{aligned}$$

Here (5.9) follows by the definition of the condition number, (5.10) follows by a standard inequality valid for the matrix norm, (5.11) follows by the same inequality and the commutaion of $\hat{R}$ and R and (5.12) follows by the definition of the condition number again.

This formula shows that the condition number of the preconditioned system is bounded, independent of the spatial discretization method and Δt. As seen in the previous subsections the last factor of (5.12) is relatively small, even for large values of j. Further, if $\hat{R}$ is a good preconditioner, $\varkappa\left(\hat{R}R\right)$ will be close to one, and the condition number for the preconditioned system remains relatively small for increasing j values.

6. PROOF OF LEMMA 5.2

This section deals with the proof of Lemma 5.2. The proof is rather long and technical. We have that

$$\varkappa\left(R_{kj}^{-1}(\Delta tA)Q_{kj}(\Delta tA)\right) < \max_i \frac{r_i}{q_i}.$$

It is therefore enough to show that

$$\frac{r_i}{q_i} \leqslant \gamma \cdot 1.09^j, \quad i \in [0, j],$$

with

$$\begin{aligned} \gamma &= 0.98 \quad \text{for} \quad j = k, \\ \gamma &= 1.11 \quad \text{for} \quad j = k+1, \\ \gamma &= 1.62 \quad \text{for} \quad j = k+2. \end{aligned}$$

This bound clearly apply for the values of j and k in Table 3. Therefore it is enough to prove Lemma 5.2 for values of j and k larger than in Table 3.

6.1. Bounding $\frac{r_i}{q_i}$ with Stirling's approximation

We have that

$$\frac{r_i}{q_i} \quad = \quad \frac{\left(\frac{k!}{(j+k)!}\right)^{\frac{i}{j}}}{\frac{(j+k-i)!}{(j+k)!}}.$$

Stirling's approximation of the factorial function is given by

$$\frac{n^n}{e^n}\sqrt{2\pi n}e^{\frac{1}{12n+1}} < n! < \frac{n^n}{e^n}\sqrt{2\pi n}e^{\frac{1}{12n}}, \quad \forall n \geqslant 1,$$

see [6,12], which enables us to bound

$$\begin{aligned} \frac{r_i}{q_i} \quad &< \quad \left(\frac{\frac{k^k}{e^k}\sqrt{2\pi k}e^{\frac{1}{12k}}}{\frac{(j+k)^{j+k}}{e^{j+k}}\sqrt{2\pi(j+k)}e^{\frac{1}{12(k+j)+1}}}\right)^{\frac{i}{j}} \frac{\frac{(j+k)^{j+k}}{e^{j+k}}\sqrt{2\pi(j+k)}e^{\frac{1}{12(k+j)}}}{\frac{(j+k-i)^{j+k-i}}{e^{j+k-i}}\sqrt{2\pi(j+k-i)}e^{\frac{1}{12(k+j-i)+1}}} \\ &= \quad \left(\frac{j+k}{j+k-i}\right)^{j+k+\frac{1}{2}-i}\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}i} e^{\frac{i}{12kj}-\frac{i}{(12(k+j)+1)j}+\frac{1}{12(k+j)}-\frac{1}{12(k+j-i)+1}} \qquad (6.1) \\ &= \quad \left(1-\frac{i}{j+k}\right)^{i-j-k-\frac{1}{2}}\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}i} e^{\frac{i}{12kj}-\frac{i}{(12(k+j)+1)j}+\frac{1}{12(k+j)}-\frac{1}{12(k+j-i)+1}} \\ &= \quad F(i,j,k)G(i,j,k), \end{aligned}$$

where

$$\begin{aligned} F(i,j,k) \quad &= \quad \left(1-\frac{i}{j+k}\right)^{i-j-k-\frac{1}{2}}\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}i}, \qquad (6.2) \\ G(i,j,k) \quad &= \quad e^{\frac{i}{12kj}-\frac{i}{(12(k+j)+1)j}+\frac{1}{12(k+j)}-\frac{1}{12(k+j-i)+1}}. \end{aligned}$$

(6.1) follows by reduction and collecting factors on the form $\frac{j+k}{j+k-i}$, $\frac{k}{j+k}$ and e.

It is straightforward to show that $G(i,j,k) < 1.0011$ for $k > 8$, and that $\lim_{j,k\to\infty} G(i,j,k) = 1$, $\forall i \in [0,j]$. The rest of this proof aims at bounding $F(i,j,k)$. For notational simplicity we write $F = F(i,j,k)$.

6.2. Maximizing F

In order to maximize this function over $i \in [0, j]$, we calculate

$$\frac{\partial F}{\partial i} = \left(\ln\left(1 - \frac{i}{j+k}\right) + 1 + \frac{1}{2(j+k)\left(1-\frac{i}{j+k}\right)} + \ln\left(\frac{k}{j+k}\right)\frac{2k+1}{2j} \right) F.$$

It can be seen from later calculations that the first factor has only one zero. Observing that $\frac{\partial F}{\partial i}(i=0) > 0$ and $F > 0$, for $i \in [0, j]$, we get that F is maximized when $\frac{\partial F}{\partial i} = 0$. Thus we have to find the zero of the first term

$$\ln\left(1 - \frac{i}{j+k}\right) + 1 + \frac{1}{2(j+k)\left(1-\frac{i}{j+k}\right)} + \ln\left(\frac{k}{j+k}\right)\frac{2k+1}{2j} = 0.$$

Make the substitution

$$x = \frac{1}{2(j+k)\left(1-\frac{i}{j+k}\right)},$$

which means that

$$i = j + k - \frac{1}{2x}. \tag{6.3}$$

Note that $i \in [0, j] \Rightarrow x \in (0, 1)$.

With this substitution we have to solve

$$\ln\left(\frac{1}{2(j+k)x}\right) + 1 + x = \ln\left(\frac{j+k}{k}\right)\frac{2k+1}{2j},$$

which implies

$$\frac{e^x}{x} = \frac{1}{h(j,k)}, \tag{6.4}$$

where

$$h(j,k) = \frac{e}{2(j+k)}\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}}. \tag{6.5}$$

Note that (6.4) has only one solution for $x \in (0,1)$, since $\frac{e^x}{x}$ is monotone. Notice also that $h(j,k) > 0$ and $h(j,k) \to 0$ when $j,k \to \infty$. In the following we write $h = h(j,k)$ for notational simplicity.

6.3. Lambert W–function

To solve (6.4) we introduce the Lambert W–function, see e.g. [4], because the solution of a nonlinear equation on the form $\frac{e^x}{x} = \frac{1}{h}$, $h \in \mathscr{R}$, can generally not be expressed with standard elementary functions. The Lambert W–function,*

$$W : [-\frac{1}{e}, \infty) \to [-1, \infty)$$

is defined to be the inverse function of the function

$$W \mapsto e^W W.$$

The series expansion of $W(x)$ is given by (see [4])

$$W(x) = \sum_{n=1}^{\infty} \frac{(-n)^{n-1}}{n!} x^n, \quad |x| < \frac{1}{e}. \tag{6.6}$$

The solution of $\frac{e^x}{x} = \frac{1}{h}$ can in fact be written

$$x = -W(-h). \tag{6.7}$$

To see this, notice that $W(x)$ is the inverse function of $e^W W$. Therefore,

$$e^{W(x)} W(x) = x \quad \Rightarrow \quad W(x) = x e^{-W(x)}.$$

Using this property together with (6.7), we get

$$\begin{aligned} \frac{e^x}{x} &= \frac{e^{-W(-h)}}{-W(-h)} \\ &= \frac{e^{-W(-h)}}{-(-h)e^{-W(-h)}} \\ &= \frac{1}{h}, \end{aligned}$$

which proves that (6.7) is the solution of (6.4).

In the following we show two properties of W which will be useful later. The first property is

$$\left(-\frac{W(-h)}{h} \right)^{-\frac{1}{2W(-h)}} = e^{\frac{1}{2}}. \tag{6.8}$$

This is proved by using the fact that

$$e = \left(\frac{W(x)}{x} \right)^{-\frac{1}{W(x)}},$$

*The Lambert W–function is available to any accuracy in Maple and Mathematica.

which follows from the definition of W.

The second property is

$$-\frac{1}{W(-h)} \quad < \quad \frac{1}{h} - 1. \tag{6.9}$$

This property is proved in the following. Define the function

$$\alpha(x) = \frac{x}{W(x)} = e^{W(x)}, \quad x < 0.$$

A Taylor series expansion of $\alpha(x)$ is

$$\alpha(x) = \alpha(0) + \alpha'(0)x + \frac{1}{2}\alpha''(\xi)x^2, \quad \xi \in (x,0). \tag{6.10}$$

Using (6.6) we see that $\alpha(0) = e^{W(0)} = 1$. Further the first order derivative of $\alpha(x)$ is

$$\alpha'(x) = e^{W(x)}W'(x),$$

and we get $\alpha'(0) = e^{W(0)}W'(0) = 1$. The second order derivative of $\alpha(x)$ is

$$\begin{aligned}\alpha''(x) &= e^{W(x)}\left[(W'(x))^2 + W''(x)\right] \\ &= -e^{W(x)}\frac{(W(x))^2}{(1+W(x))^3x^2} < 0, \quad \text{for} \quad x < 0.\end{aligned}$$

where we have used that $W'(x) = \frac{W(x)}{(1+W(x))x}$ and that $W(x) > -1$ and $e^{W(x)} > 0$. Inserting into (6.10) we get

$$\alpha(x) < 1 + x, \quad x < 0,$$

and the second property (6.9) follows by setting $x = -h$.

6.4. Maximizing F (continued)

With these properties of the Lambert function we continue to maximize F. We substitute back by inserting (6.7) into (6.3), and find that the extreme value for F is obtained when

$$i = j + k + \frac{1}{2W(-h)}.$$

Therefore

$$
\begin{aligned}
F(i,j,k) &\leqslant F(j+k+\frac{1}{2W(-h)},j,k) \\
&= \left(2(j+k)h\left(\frac{W(-h)}{-h}\right)\right)^{\frac{1}{2}}(2(j+k)h)^{-\frac{1}{2W(-h)}}\left(\frac{W(-h)}{-h}\right)^{-\frac{1}{2W(-h)}} \\
&\quad \left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}(j+k)}\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}\frac{1}{2W(-h)}} \qquad (6.11) \\
&= \left(e\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}}\left(\frac{W(-h)}{-h}\right)\right)^{\frac{1}{2}} e^{-\frac{1}{2W(-h)}}\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}\frac{-1}{2W(-h)}} e^{\frac{1}{2}} \\
&\quad \left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}(j+k)}\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}\frac{1}{2W(-h)}} \qquad (6.12) \\
&< \left(e\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}}\left(\frac{W(-h)}{-h}\right)\right)^{\frac{1}{2}} e^{\frac{1}{2h}-\frac{1}{2}}\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}(j+k)} e^{\frac{1}{2}} \qquad (6.13) \\
&= \left(e\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}}\left(\frac{W(-h)}{-h}\right)\right)^{\frac{1}{2}} e^{\frac{1}{2h}}\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}(j+k)}, \qquad (6.14)
\end{aligned}
$$

where (6.11) follows by simplifying and expanding F, (6.12) follows by the definition of h, (6.5), and the first property of W, (6.8), and (6.13) follows by reduction and the second property of W, (6.9).

We have that $\frac{W(-h)}{-h}$ is bounded because W is bounded for negative arguments and $W'(0)=1$. Thus we see that the first factor of (6.14), $\left(e\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}}\left(\frac{W(-h)}{-h}\right)\right)^{\frac{1}{2}}$, is bounded independent of j and k.

In the following we shall see that the last two factors of (6.14), $e^{\frac{1}{2h}}\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}(j+k)}$, are bounded by exponential functions in j (or equivalently in k). First we define $d=j-k$. This leads to

$$
\begin{aligned}
e^{\frac{1}{2h}} &= e^{\frac{j+k}{e}\left(\frac{j+k}{k}\right)^{\frac{2k+1}{2j}}} \\
&= e^{\frac{(2k+d)^2}{ek}\left(\frac{k}{2k+d}\right)^{\frac{2d-1}{2k+2d}}} \qquad (6.15) \\
&= e^{\frac{4}{e}j}e^{\frac{d^2}{ek}}e^{\frac{(2k+d)^2}{ek}\left(\left(\frac{k}{2k+d}\right)^{\frac{2d-1}{2k+2d}}-1\right)}, \qquad (6.16)
\end{aligned}
$$

where (6.15) follows from $\frac{2k+1}{2j}=1-\frac{2d-1}{2k+2d}$, and (6.16) follows by the trick $e^{xy}=e^x e^{x(y-1)}$ and the fact that $\frac{(j+k)^2}{ek}=\frac{4j}{e}+\frac{d^2}{ek}$.

Remember that, because of stability requirements, we are only interested in the three cases $j=k$, $j=k+1$ and $j=k+2$, which means that $d=0$, $d=1$ and $d=2$ respectively.

Now we study the last factor of (6.16). Note that

$$\lim_{k\to\infty} e^{\frac{(2k+d)^2}{ek}\left(\left(\frac{k}{2k+d}\right)^{\frac{2d-1}{2k+2d}}-1\right)} = 2^{\frac{2}{e}(1-2d)}. \tag{6.17}$$

Consider the sign of the last factor of the exponent, i.e.

$$\left(\frac{k}{2k+d}\right)^{\frac{2d-1}{2k+2d}} - 1.$$

This is positive for $d = 0$ and negative for $d = 1,2$. By this observation we can see that the last factor of (6.16) is a decreasing function of k for $d = 0$ and an increasing function of k for $d = 1,2$. Thus the limit (6.17) is an upper bound for (6.16) when $d = 1,2$.

For the last factor of (6.14) we bound

$$\begin{aligned}
\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}(j+k)} &= \left(\frac{1}{2+\frac{d}{k}}\right)^{2k-d+\frac{2k+d+d^2}{2k+2d}} && (6.18)\\
&< \left(\frac{1}{2+\frac{d}{k}}\right)^{2k-d+1} && (6.19)\\
&= \left(\frac{1}{2}\right)^{2j-3d+1}\left(1+\frac{d}{2k}\right)^{-2j+3d-1}, && (6.20)
\end{aligned}$$

where (6.18) follows by expanding the exponent and the base, (6.19) follows by $\frac{2k+d+d^2}{2k+2d} > 1$, which is valid for all integers d and (6.20) is an expansion.

To summarize, we now get an upper bound on F by inserting (6.16) and (6.20) into (6.14)

$$\begin{aligned}
F(i,j,k) < &\left(\frac{1}{4}e^{\frac{4}{e}}\right)^j\left(e\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}}\left(\frac{W(-h)}{-h}\right)\right)^{\frac{1}{2}} e^{\frac{d^2}{ek}}\\
&e^{\frac{(2k+d)^2}{ek}\left(\left(\frac{k}{2k+d}\right)^{\frac{2d-1}{2k+2d}}-1\right)}\left(\frac{1}{2}\right)^{-3d+1}\left(1+\frac{d}{2k}\right)^{-2j+3d-1}, \qquad (6.21)
\end{aligned}$$

where all factors except the first is bounded independently from j. The above result shows that F is bounded by a function on the form

$$F < \tilde{\gamma}\left(\frac{1}{4}e^{\frac{4}{e}}\right)^j, \tag{6.22}$$

where $\tilde{\gamma}$ is a constant independent of j and $\frac{1}{4}e^{\frac{4}{e}} \approx 1.088960317$. Since $G(i,j,k)$ is bounded independent of j and k, the first part of Lemma 5.1, (5.5), is proved. In the the following we determine the constant γ for the three cases $j = k$, $j = k+1$ and $j = k+2$.

6.5. Determining the constants

In order to determine the smallest possible value for γ we split up into the three interesting cases, i.e. $j = k$, $j = k+1$ or $j = k+2$. We utilize that the lemma is already proved for $k \leqslant 8$, and that all factors are decreasing in j and k.

For $j = k$ ($d = 0$) we get

$$\begin{aligned}
\frac{r_i}{q_i} &< F(i,k,k)G(i,k,k) \\
&< \left(\frac{1}{4}e^{\frac{4}{e}}\right)^j \sqrt{e\frac{1}{2}\left(\frac{W(-h(11,11))}{-h(11,11)}\right)}e^{\frac{44}{e}(2^{\frac{1}{22}}-1)} \\
&\quad \frac{1}{2}\max_i G(i,11,11) \\
&< 0.98\cdot(1.09)^j,
\end{aligned}$$

where we have used that $\frac{1}{2}^{\frac{2k+1}{2k}}$ is increasing function of k with the limit $\frac{1}{2}$, $\frac{W(-h(k,k))}{-h(k,k)}$ and $e^{\frac{4k}{e}\left(2^{\frac{1}{2k}}-1\right)}$ are decreasing functions of k and that the inequality is checked for $j,k \leqslant 10$.

For $j = k+1$ ($d = 1$) we get

$$\begin{aligned}
\frac{r_i}{q_i} &< F(i,k+1,k)G(i,k+1,k) \\
&< \left(\frac{1}{4}e^{\frac{4}{e}}\right)^j \sqrt{e\frac{1}{2}\left(\frac{W(-h(11,10))}{-h(11,10)}\right)} \\
&\quad e^{\frac{1}{10e}}\left(\frac{1}{2}\right)^{\frac{2}{e}} 2^2\left(1+\frac{1}{20}\right)^{-20}\max_i G(i,11,10) \\
&< 1.11\cdot(1.09)^j,
\end{aligned}$$

where we have used that $\frac{k}{2k+1}^{\frac{2k+1}{2k+2}}$ and $e^{\frac{(2k+1)^2}{ek}\left(\left(\frac{k}{2k+1}\right)^{\frac{1}{2k+2}}-1\right)}$ are increasing functions of k with limits $\frac{1}{2}$ and $2^{-\frac{2}{e}}$ respectively, $\frac{W(-h(k+1,k))}{-h(k+1,k)}$, $e^{\frac{1}{ek}}$, $\left(1+\frac{1}{2k}\right)^{-2k}$ and $\max_i G(i,k+1,k)$ are decreasing functions of k and that the inequality is valid up to $j = 10$ and $k = 9$.

For $j = k+2$ ($d = 2$) we get

$$\begin{aligned}
\frac{r_i}{q_i} &< F(i,k+2,k)G(i,k+2,k) \\
&< \left(\frac{1}{4}e^{\frac{4}{e}}\right)^j \sqrt{e\left(\frac{9}{20}\right)^{\frac{19}{22}}\left(\frac{W(-h(11,9))}{-h(11,9)}\right)} \\
&\quad e^{\frac{4}{e^9}}2^{-\frac{6}{e}}2^5\left(1+\frac{2}{18}\right)^{-17}\max_i G(i,11,9) \\
&< 1.62\cdot(1.09)^j,
\end{aligned}$$

where we have used that $e^{\frac{(2k+2)^2}{ek}\left(\left(\frac{k}{2k+2}\right)^{\frac{3}{2k+4}}-1\right)}$ is an increasing function of k with limit $2^{-\frac{6}{e}}$, $\frac{k}{k+2}^{\frac{2k+1}{2k+4}}$, $\frac{W(-h(k+1,k))}{-h(k+1,k)}$, $e^{\frac{4}{ek}}$, $\left(1+\frac{2}{2k}\right)^{-2k+1}$ and $\max_i G(i,k+2,k)$ are decreasing functions of k and that the inequality is valid up to $j = 10$ and $k = 8$.

6.6. The bound when $j,k \to \infty$

Above we have used that $k > 8$. In this subsection we determine the constants when $j,k \to \infty$. First we calculate the limits of the factors of (6.21)

$$\begin{aligned}
&\lim_{j,k\to\infty}\left(\frac{k}{j+k}\right)^{\frac{2k+1}{2j}} = \frac{1}{2}, \\
&\lim_{j,k\to\infty} h(j,k) = 0, \\
&\lim_{h\to 0}\frac{W(-h)}{-h} = 1, \\
&\lim_{j,k\to\infty} e^{\frac{d^2}{ek}} = 1, \\
&\lim_{j,k\to\infty}\left(1+\frac{d}{2k}\right)^{-2k+d-1} = e^{-d}.
\end{aligned}$$

Together with (6.17) we obtain from (6.21)

$$F < \left(\frac{1}{4}e^{\frac{4}{e}}\right)^j e^{\frac{1}{2}-d}2^{3d-\frac{3}{2}+\frac{2}{e}(1-2d)}, \tag{6.23}$$

when $j,k \to \infty$. Since $\lim_{j,k\to\infty} G(i,j,k) = 1$, we get

$$\begin{aligned}
\frac{r_i}{q_i} &< \left(\frac{1}{4}e^{\frac{4}{e}}\right)^j e^{\frac{1}{2}} 2^{\frac{2}{e}-\frac{3}{2}} \approx 0.9709 \left(\frac{1}{4}e^{\frac{4}{e}}\right)^j, \quad \text{when} \quad j = k \to \infty, \\
\frac{r_i}{q_i} &< \left(\frac{1}{4}e^{\frac{4}{e}}\right)^j e^{-\frac{1}{2}} 2^{-\frac{1}{2}-\frac{2}{e}} \approx 1.0302 \left(\frac{1}{4}e^{\frac{4}{e}}\right)^j, \quad \text{when} \quad j = k+1 \to \infty, \\
\frac{r_i}{q_i} &< \left(\frac{1}{4}e^{\frac{4}{e}}\right)^j e^{-\frac{3}{2}} 2^{\frac{1}{2}-\frac{6}{e}} \approx 1.0933 \left(\frac{1}{4}e^{\frac{4}{e}}\right)^j, \quad \text{when} \quad j = k+2 \to \infty.
\end{aligned}$$

Finally, we mention that the upper bound on F given in (6.23) probably can be extended to be valid for the condition number of the preconditioned system, i.e.

$$\varkappa\left(R_{kj}^{-1}(\Delta t A)Q_{kj}(\Delta t A)\right) < \left(\frac{1}{4}e^{\frac{4}{e}}\right)^j e^{\frac{1}{2}+k-j} 2^{3j-3k-\frac{3}{2}+\frac{2}{e}(1+2k-2j)},$$

for all $j \in [k, k+2]$. This inequality is valid for the numbers in Table 1, and can be chekced to be valid for larger values of j and k by using the bound (5.2).

7. CONCLUDING REMARKS

In this paper we have proposed a preconditioner for the system arising when using Padé approximation to obtain higher–order time discretization for the heat equation. The method is applicable to linear inhomogeneous parabolic equations. We have shown that our preconditioner will give a preconditioned system with a condition number independent of the discretization parameters. Finally, we have proved that the condition number will remain reasonably low even when the order of the time discretization is very high.

REFERENCES

1. R. E. Bank and T. Dupont, An optimal order process for solving finite element equations, *Math. Comp.* (1981) **36**, 35–51.
2. J. H. Bramble and P. H. Sammon, Efficient higher order single step methods for parabolic problems: Part i, *Math. Comp.* (1980) **35**, 655–677.
3. C. Canuto and A. Quarteroni, Preconditioned minimal residual methods for Chebyshev spectral calculations, *J. Comput. Phys.* (1985) **60**, 315–337.
4. R.M. Corless, G.H. Gonnet, D.E.G. Hare, D.J. Jeffrey, and D.E. Knuth, On the Lambert W function, *Advances in Computational Mathematics* (1996) **5**, 329–359.
5. M. O. Deville and E. H. Mund, Chebyshev pseudospectral solution of second–order elliptic equation with finite element preconditioning, *J. Comput. Phys.* (1985) **60**, 517–533.
6. W. Feller, *An Introduction to Probability Theory and Its Applications*, Wiley, 3nd edition, 1966.
7. B. Gustafsson and W. Kress, Deferred correction methods for initial value problems, *BIT* (2001) **41**, 986–995.

8. E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II - Stiff and Differential-Algebraic Problems*, Springer Verlag, 2nd edition, 1996.
9. W. Kress and B. Gustafsson, Deferred correction methods for initial boundary value problems, *J. Scientific Computing* (2002) **17**, 241–251.
10. J. Van Lent and S. Vandewalle, *Methods for Implicit Runge–Kutta and Boundary Value Method Discretizations of Parabolic PDEs, SIAM J. Sci. Comput.* (2005) **27(1)**, 67–92.
11. Maxim A. Olshanskii and Arnold Reusken, On the convergence of a multigrid method for linear reaction-diffusion problems, *Computing* (2000) **65(3)**, 193–202.
12. H. Robbins. A remark on Sterling's formula, *Amer. Math. Monthly* (1955) **62**, 26–29.
13. V. Thomée, *Galerkin Finite Element Methods for Parabolic Problems*, Springer-Verlag, 2nd edition, 1997.