Time-parallel iterative solvers for parabolic evolution equations

Martin Neumüller, Iain Smears

Time-parallel iterative solvers for parabolic evolution equations*

Martin Neumüller[†] and Iain Smears[‡]

March 6, 2019

Abstract

We present original time-parallel algorithms for the solution of the implicit Euler discretization of general linear parabolic evolution equations with time-dependent self-adjoint spatial operators. Motivated by the inf-sup theory of parabolic problems, we show that the standard nonsymmetric time-global system can be equivalently reformulated as an original symmetric saddle-point system that remains inf-sup stable with respect to the same natural parabolic norms. We then propose and analyse an efficient and readily implementable parallel-in-time preconditioner to be used with an inexact Uzawa method. The proposed preconditioner is non-intrusive and easy to implement in practice, and also features the key theoretical advantages of robust spectral bounds, leading to convergence rates that are independent of the number of time-steps, final time, or spatial mesh sizes, and also a theoretical parallel complexity that grows only logarithmically with respect to the number of time-steps. Numerical experiments with large-scale parallel computations show the effectiveness of the method, along with its good weak and strong scaling properties.

Key words: Parabolic partial differential equations, parallel algorithms, preconditioners, convergence, parallel complexity.

Contents

[†]Institute of Computational Mathematics, Johannes Kepler University Linz, 4040 Linz, Austria (neumueller@numa.uni-linz.ac.at)

[‡]Department of Mathematics, University College London, 25 Gordon Street, London WC1E 6BT, United Kingdom (i.smears@ucl.ac.uk)

1   Introduction

The usual approach to the numerical solution of parabolic partial differential equations (PDE) involves the sequential solution of discrete systems obtained by a time-stepping scheme. In many cases, a large number of time-steps might be required, which can lead to long computation times. The large number of cores in present-day high-performance computers enables the use of time-parallel algorithms as an alternative to the sequential approach. By treating many time-steps simultaneously and in parallel, these specialized algorithms aim to compute the numerical solution to a desired accuracy in a shorter computational time than can be achieved by the sequential approach.

Starting with Nievergelt [33] in 1964, a wide variety of time-parallel algorithms have been proposed [5, 8, 9, 13, 17, 20, 23, 26, 31, 43], see also the review [16]. One of the central questions that any time-parallel algorithm must address is how to efficiently propagate information about the solution across the whole time interval. In the parareal method [3, 19, 27, 29], this is done by iterating between the sequential solution of problems on a coarse temporal grid with parallel solves on the fine temporal grid. Instead of a single coarse temporal grid, time and space-time multigrid methods [13, 18, 22, 24, 25] use a hierarchy of coarser grids in both space and time. For linear problems, many of these algorithms can be seen as convergent iterative solvers for a large time-global nonsymmetric linear system. Recall that the available theory for the design and analysis of iterative methods for general nonsymmetric systems is currently rather more limited than for their symmetric counterparts [30, 42]. Therefore, the study of iterative methods for parabolic problems appears significant within the wider context of the solution of large nonsymmetric linear systems.

Recently, two independent works [1, 39] proposed to treat the nonsymmetry of linear systems coming from parabolic problems using approaches based on the inf-sup condition. In particular, the first step in the approach from [39], which is developed further here, is to build a left-preconditioner that is based on the mapping from trial functions to their optimal test functions in the analysis of the inf-sup condition. The left-preconditioner then leads to equivalent symmetric reformulations that are stable in the same norms and spaces (or their discrete analogues) as those that appear naturally in the analysis of well-posedness of the problem. Note that the stability of these reformulations distinguishes this approach from classical ones such as forming

the normal equations. Convergent iterative solvers can then be built out of preconditioners for the symmetric positive definite systems associated to the norms and spaces appearing in the inf-sup condition.

This approach to preconditioning and solving the system is conceptually distinct from those in many previous works in several ways, one of which is that it explicitly features preconditioners that handle the appropriate spatial dual-norm on the time derivative. For illustration, consider momentarily the heat equation $\partial_t u - \Delta u = f$ in $\Omega \times (0, T)$, where $f \in L^2(0, T; H^{-1}(\Omega))$, with, for simplicity, homogeneous Cauchy–Dirichlet conditions at the boundary and initial time. Then, following [28, Chap. 3], the solution space S is the space of functions in $L^2(0, T; H_0^1(\Omega)) \cap H^1(0, T; H^{-1}(\Omega))$ that vanish at $t = 0$. Furthermore, we have the following identity

$$\|u\|_S = \sup_{v \in L^2(0,T;H_0^1(\Omega)) \backslash \{0\}} \frac{B(u,v)}{\|v\|_A} \quad \forall\, u \in S, \tag{1.1}$$

where $B(u, v) := \int_0^T \langle \partial_t u, v \rangle + (\nabla u, \nabla v)_\Omega \, \mathrm{d}t$ is the bilinear form for a weak formulation of the problem, where $\|u\|_S^2 := \int_0^T \|\partial_t u\|_{H^{-1}(\Omega)}^2 + \|\nabla u\|_\Omega^2 \, \mathrm{d}t + \|u(T)\|_\Omega^2$ is the norm on S, and where $\|v\|_A^2 := \int_0^T \|\nabla v\|_\Omega^2 \, \mathrm{d}t$, with $\|\cdot\|_\Omega$ denoting the L^2-norm over Ω (for proof, see for instance [11]). For each trial function $u \in S$, there is an optimal choice of test function $v \in L^2(0, T; H_0^1(\Omega))$ that achieves the supremum in (1.1); in the discrete setting, this mapping between trial and optimal test functions leads to a left-preconditioner that symmetrizes the system in a stable way. The complete solution algorithm combines this with additional preconditioners tied to the discrete versions of the norms $\|\cdot\|_S$ and $\|\cdot\|_A$, i.e. that handle explicitly the norm on the time derivative (see Sections 2 and 4 below for details).

The formulation of the inf-sup theorem for general abstract linear problems in Hilbert spaces was first stated by Nečas [32], and was later applied in the error analysis of finite element methods by Babuška in [2]. See also [7] for saddle-point problems. More recently, for parabolic problems, we note that inf-sup stability has previously found application in several contexts, such as a priori error analysis [40], a posteriori analysis [11, 12], and reduced-basis methods [41]. We also refer the reader to the textbooks [10, 38] for the inf-sup theorem for general linear equations in Banach spaces, and its application to parabolic problems.

In this work, we present some original time-parallel algorithms for the solution of the implicit Euler discretization of general parabolic evolution equations with self-adjoint spatial operators. The first contribution is to show that the discrete systems admit a similar inf-sup analysis to (1.1), which allows us to find equivalent symmetric systems that are stable with respect to the discrete counterparts of the norms $\|\cdot\|_S$ and $\|\cdot\|_A$ above. In particular, we obtain an equivalent symmetric saddle-point formulation that is well-suited for preconditioned iterative solvers, such as the inexact Uzawa method [6, 45] or the preconditioned MINRES method [34]. The second contribution is to propose robust and efficient time-parallel preconditioners for these linear systems, resulting fast convergence of the iterative solvers. Specifically, the preconditioner for the first variable of the saddle-point system is block-diagonal with respect to the time-steps, and the Schur complement preconditioner for the second variable is also block-diagonalized under the Discrete Sine Transform (DST) in time. Thus, after application of the DST, the time-parallelism is essentially trivial, and significantly simplifies the treatment of the dual norm of the time derivatives. The DST can be implemented through parallel Fast Fourier Transforms (FFT), which have a low parallel complexity and are often relatively cheap in practice compared to the spatial solvers. The transformations of the temporal basis via the DST constitutes the main mechanism for exchange of information over time.

The main features of the algorithm can be summarized as follows.

3

Convergence theory. Applying the proposed time-parallel preconditioners to standard solvers, such as the inexact Uzawa method, leads to robust convergence rates that depend only on the efficiency of the spatial preconditioners and on the quasi-uniformity of the problem. As a result, the convergence of the algorithm is independent of the number of time-steps, the spatial mesh size, and the final time. The main step in the analysis is a proof of the robust spectral equivalence of the Schur complement and its preconditioner through the DST.

Parallel complexity. Since we are primarily interested here in time-parallelism, we study the dependence of the parallel complexity on N the number of time-steps. We show that each iteration of the method has a parallel complexity of $O(\log N)$ when sufficiently many processors are available. To put this result in context, note that the standard parareal method using the implicit Euler method in the coarse and fine solvers achieves at best a parallel complexity of order $O(\sqrt{N})$, as shown in [4]. This result should be considered in light of the logarithmic-order lower bounds on the optimal achievable parallel complexity, see [26, 44].

Treatment of time-dependent spatial operators and non-uniform time-steps. The spatial operators can be time-dependent and the time-step lengths may vary, under the quasi-uniformity condition (2.2) below. It is worth noting that the time-dependence of the operators precludes an analysis based on reducing the problem to the scalar ODE case through spatial eigenvector decompositions, which is a common approach in the literature on time-parallel algorithms.

Simplicity of implementation. On the practical side, the parallel implementation of the proposed method has the advantage of being non-intrusive with respect to the spatial solvers and preconditioners, so black-box existing spatial solvers can be re-used. This means that significant spatial parallelism can be straightforwardly included, see for instance the numerical experiments below. The parallelization in time requires only parallel implementations of the one-dimensional FFT, which are available in libraries such as FFTW3 [15].

This article is organized as follows. The discrete parabolic problem is presented in Section 2, where we propose an equivalent stable symmetric saddle-point formulation that is the starting point for our approach. We then consider an inexact Uzawa method as a representative iterative solver in Section 3, along with a convergence theorem that motivates the need for a spectrally equivalent Schur complement preconditioner. Section 4 then details the construction of parallelisable preconditioner along with the key spectral bounds. This is followed by the bounds on parallel complexity in Section 5. The analysis of the spectral bounds is taken up in Sections 6 and 7. Finally, we present numerical experiments with large scale parallel computations in Section 8, before presenting our conclusions.

2   Discrete parabolic problem

For $T > 0$, consider a partition of the time interval $(0, T)$ into disjoint time-step intervals $I_n = (t_{n-1}, t_n)$, with $0 = t_0 \leq t_{n-1} < t_n \leq t_N = T$ for each $1 \leq n \leq N$. Let $\tau_n := t_n - t_{n-1}$ denote the time-step lengths for each $n = 1, \ldots, N$. For a given finite dimensional space $\mathbb{V}$, let M and A_n, $n = 1, \ldots, N$, be symmetric positive definite matrices on $\mathbb{V}$.

Consider the discretization of an abstract parabolic evolution equation by the implicit Euler method

$$M(u_n - u_{n-1}) + \tau_n A_n u_n = \tau_n f_n, \tag{2.1}$$

where $u_n \in \mathbb{V}$ for each $n = 1, \ldots, N$. At the first time-step $n = 1$, the term u_0 is replaced by some given initial datum $u_I \in \mathbb{V}$. In applications to second-order parabolic PDEs, the matrices M and A_n typically represent the mass and stiffness matrices obtained by some spatial discretization method.

4

The matrices M and A_n, $n = 1, \ldots, N$, induce the inner-products $(\cdot, \cdot)_M$ and $(\cdot, \cdot)_{A_n}$ and the norms $\|\cdot\|_M$ and $\|\cdot\|_{A_n}$ on $\mathbb{V}$. To simplify the notation, we shall identify functions in $\mathbb{V}$ with their vector representations, so for instance we shall write $\|v\|_M^2 = v^\top M v$ for all $v \in \mathbb{V}$. We assume that there exists a symmetric positive definite matrix A along with positive constants $\tau > 0$ and $\alpha \geq 1$, such that

$$\frac{1}{\alpha} \tau A \leq \tau_n A_n \leq \alpha \tau A \quad \forall n = 1, \ldots, N, \tag{2.2}$$

where the inequalities are in the sense of the partial ordering of positive semi-definite symmetric matrices (i.e. $A \leq B$ if and only if $B - A$ is positive semi-definite). In other words, we assume that the matrices $\tau_n A_n$, $n = 1, \ldots, N$, are uniformly spectrally equivalent to the matrix τA; this amounts to a non-degeneracy and quasi-uniformity assumption. For instance, this assumption is guaranteed if the temporal grid is quasi-uniform and all matrices A_n are spectrally equivalent to A; however (2.2) is a somewhat weaker assumption in general. We stress that (2.2) is not a CFL-type restriction on the time-step sizes. The constant τ and matrix A will be needed in the algorithm below, and, ideally, they should be chosen to make the constant α as close to 1 as possible. In practice, some simple choices would be to select τ and A among the time-steps $\{\tau_n\}$ and operators A_n given by the problem, or to consider an average.

2.1 Inf-sup stability of the discrete problem

First, we express (2.1) in time-global form by gathering the solution values into the vector $\mathbf{u} = [u_1, \ldots, u_N] \in \mathbb{V}^N$, with $\mathbb{V}^N = \mathbb{V} \times \cdots \times \mathbb{V}$, which leads to the nonsymmetric system

$$\mathbf{B}\mathbf{u} = \mathbf{f}, \quad \mathbf{B} := \mathbf{K} + \mathbf{A}, \tag{2.3}$$

where $\mathbf{A} := \operatorname{diag}\{\tau_n A_n\}_{n=1}^N$ is the block-diagonal matrix with entries $\tau_n A_n$ along the diagonal, and where

$$\mathbf{K} := K \otimes M, \quad K := \begin{pmatrix} 1 & & & \\ -1 & 1 & & \\ & -1 & 1 & \\ & & & \ddots \end{pmatrix}, \tag{2.4}$$

with $\otimes$ denoting is the Kronecker product of matrices. Furthermore, the right-hand side in (2.3) is given by $\mathbf{f} = [\tau_1 f_1 + M u_I, \tau_2 f_2, \ldots, \tau_N f_N]$.

We now introduce the matrix $\mathbf{S}$ defined by

$$\mathbf{S} := \mathbf{B}^\top \mathbf{A}^{-1} \mathbf{B} = \mathbf{K}^\top \mathbf{A}^{-1} \mathbf{K} + \mathbf{K} + \mathbf{K}^\top + \mathbf{A}. \tag{2.5}$$

The matrix $\mathbf{S}$ is symmetric positive definite, since $\mathbf{B}$ is known to be nonsingular. Furthermore, it is easy to see that, for any $\mathbf{u} \in \mathbb{V}^N$,

$$\|\mathbf{u}\|_{\mathbf{S}} = \sup_{\mathbf{v} \in \mathbb{V}^N \setminus \{0\}} \frac{\mathbf{v}^\top \mathbf{B}\mathbf{u}}{\|\mathbf{v}\|_{\mathbf{A}}}, \tag{2.6}$$

where the supremum is achieved by taking $\mathbf{v} = \mathbf{P}\mathbf{u}$, where $\mathbf{P} := \mathbf{A}^{-1}\mathbf{B}$, and where $\|\cdot\|_{\mathbf{A}}$ and $\|\cdot\|_{\mathbf{S}}$ are the norms associated with $\mathbf{A}$ and $\mathbf{S}$ defined above. The identity in (2.6) represents the discrete analogue of (1.1). In particular, we will see later in Section 6 how the variational formulation of the implicit Euler scheme offers a deeper conceptual understanding of (2.6) and its relation to the continuous setting.

2.2 Equivalent reformulations

The starting point for the derivation of the algorithm is to consider two equivalent reformulations of (2.3). First, observe that the matrix $\mathbf{P}$, which maps trial functions to optimal test functions in (2.6), can be rewritten as $\mathbf{P} = \mathbf{A}^{-1}\mathbf{K} + \mathbf{I}$, with $\mathbf{I} = \mathrm{Id}_{N \times \dim \mathbb{V}}$ the identity matrix of dimension $N \times \dim \mathbb{V}$. Moreover, we have $\mathbf{S} = \mathbf{P}^{\top}\mathbf{B}$, so the matrix $\mathbf{S}$ can be seen as the product of $\mathbf{B}$ with the left-preconditioner $\mathbf{P}^{\top}$. Therefore, (2.3) is equivalent to the left-preconditioned system

$$\mathbf{S}\mathbf{u} = \mathbf{g}, \quad \mathbf{g} \coloneqq \mathbf{P}^{\top}\mathbf{f}. \tag{2.7}$$

Thus (2.7) amounts to left-preconditioning (2.3) with the mapping between trial functions to the optimal test-functions.

It is possible, in theory, to apply standard iterative solvers, such as a preconditioned conjugate gradient (CG) method, to (2.7) since $\mathbf{S}$ is symmetric positive definite. However, this would explicitly require the action of $\mathbf{A}^{-1}$ at each iteration, which can be expensive to compute. To overcome this issue, we introduce the auxiliary variable $\mathbf{p} \in \mathbb{V}^{N}$ defined by the equation $\mathbf{A}\mathbf{p} = \mathbf{K}\mathbf{u} - \mathbf{f}$. Then, a simple calculation shows that the matrix $\mathbf{S}$ in (2.7) is the Schur complement of the symmetric indefinite system

$$\boldsymbol{\mathcal{A}}\boldsymbol{u} = \boldsymbol{g}, \quad \boldsymbol{\mathcal{A}} \coloneqq \begin{bmatrix} \mathbf{A} & -\mathbf{K} \\ -\mathbf{K}^{\top} & -\left(\mathbf{K} + \mathbf{K}^{\top} + \mathbf{A}\right) \end{bmatrix}, \quad \boldsymbol{u} \coloneqq \begin{bmatrix} \mathbf{p} \\ \mathbf{u} \end{bmatrix}, \quad \boldsymbol{g} \coloneqq \begin{bmatrix} -\mathbf{f} \\ -\mathbf{f} \end{bmatrix}. \tag{2.8}$$

The solution $\mathbf{u} \in \mathbb{V}^{N}$ of (2.3) is the second component of the solution $\boldsymbol{u} \in \mathbb{V}^{N} \times \mathbb{V}^{N}$ of (2.8), and we see that $\mathbf{p} = -\mathbf{u}$ from (2.3). The matrix $\boldsymbol{\mathcal{A}}$ is of saddle-point type, with dimension $2 \times \dim \mathbb{V} \times N$ and is block-sparse, since $\mathbf{A}$ is block-diagonal, $\mathbf{K}$ is block lower-triangular and bidiagonal, and $\mathbf{K} + \mathbf{K}^{\top}$ is block-tridiagonal.

We propose to compute the solution of the problem by applying preconditioned iterative solvers to the system (2.8), such as the inexact Uzawa method. The advantage of (2.8) over (2.7) is that it allows for iterative solvers that merely approximate the action of $\mathbf{A}^{-1}$ with preconditioners. The advantage of (2.8) over the original formulation (2.3) is that it is symmetric, whilst remaining inf-sup stable with respect to the same norms as (2.3). Indeed, it is possible to show, using for instance the bounds in [37], that, for any $\boldsymbol{u} \in \mathbb{V}^{N} \times \mathbb{V}^{N}$,

$$\frac{1}{2}\left(\sqrt{5} - 1\right) \|\boldsymbol{u}\|_{*} \leq \sup_{\boldsymbol{v} \in \mathbb{V}^{N} \times \mathbb{V}^{N} \setminus \{0\}} \frac{\boldsymbol{v}^{\top}\boldsymbol{\mathcal{A}}\boldsymbol{u}}{\|\boldsymbol{v}\|_{*}} \leq \frac{1}{2}\left(\sqrt{5} + 1\right) \|\boldsymbol{u}\|_{*}. \tag{2.9}$$

where the norm $\|\cdot\|_{*}$ on $\mathbb{V}^{N} \times \mathbb{V}^{N}$ is defined by

$$\|\boldsymbol{v}\|_{*}^{2} \coloneqq \|\mathbf{q}\|_{\mathbf{A}}^{2} + \|\mathbf{v}\|_{\mathbf{S}}^{2} \quad \forall \boldsymbol{v} = [\mathbf{q}, \mathbf{v}] \in \mathbb{V}^{N} \times \mathbb{V}^{N}. \tag{2.10}$$

We then see that the norm on the second variable in (2.10) is the same as in the left-hand side of (2.6).

3 Inexact Uzawa method

There is a range of iterative methods for solving saddle-point systems such as (2.8). We consider here the inexact Uzawa method

$$\begin{aligned} \mathbf{p}_{j+1} &= \mathbf{p}_{j} + \widetilde{\mathbf{A}}^{-1}\left(\mathbf{K}\mathbf{u}_{j} - \mathbf{A}\mathbf{p}_{j} - \mathbf{f}\right), \\ \mathbf{u}_{j+1} &= \mathbf{u}_{j} + \omega\widetilde{\mathbf{H}}^{-1}\left(\mathbf{f} - \mathbf{K}^{\top}\mathbf{p}_{j+1} - \left[\mathbf{K} + \mathbf{K}^{\top} + \mathbf{A}\right]\mathbf{u}_{j}\right), \end{aligned} \tag{3.1}$$

where $\widetilde{\mathbf{A}}$ and $\widetilde{\mathbf{H}}$ are respectively preconditioners for $\mathbf{A}$ and $\mathbf{S}$, where $\omega > 0$ is a damping parameter, and where $\boldsymbol{u}_0 = [\mathbf{p}_0, \mathbf{u}_0]$ is an initial guess. In practice, it is most natural to choose $\widetilde{\mathbf{A}}$ to be of block-diagonal form

$$\widetilde{\mathbf{A}} := \operatorname{diag}\{\tau_n \widetilde{A}_n\}_{n=1}^N, \tag{3.2}$$

where, for each $n = 1, \ldots, N$, the matrix $\widetilde{A}_n$ is symmetric positive definite. The application of $\widetilde{\mathbf{A}}^{-1}$ is then trivially parallel with respect to the time-steps. In practice, the approximation $\widetilde{A}_n$ is usually defined implicitly in terms of its inverse $\widetilde{A}_n^{-1}$ that represents the action of a standard solver for spatial problems, such as a small number of multigrid V-cycles for example. For the analysis, we assume that there exists $\rho_{\mathbf{A}}$ with $0 \le \rho_{\mathbf{A}} < 1$, such that each $\widetilde{A}_n$ is a convergent approximation of A_n, i.e.

$$\|I - \widetilde{A}_n^{-1} A_n\|_{\widetilde{A}_n} \le \rho_{\mathbf{A}} \quad \forall n = 1, \ldots, N. \tag{3.3}$$

The following theorem gives sufficient conditions for convergence of the inexact Uzawa method. Let the norm $\|\cdot\|_{\boldsymbol{\mathcal{D}}}$ be defined on $\mathbb{V}^N \times \mathbb{V}^N$ by

$$\|\boldsymbol{v}\|_{\boldsymbol{\mathcal{D}}}^2 := \omega \rho_{\mathbf{A}} \|\mathbf{q}\|_{\widetilde{\mathbf{A}}}^2 + \|\mathbf{v}\|_{\widetilde{\mathbf{H}}}^2. \tag{3.4}$$

Theorem 3.1 (Convergence). *Suppose that (3.3) holds and that $\widetilde{\mathbf{H}}$ is symmetric positive definite. Let $0 < \lambda_{\min} \le \lambda_{\max}$ be constants such that $\lambda_{\min}\widetilde{\mathbf{H}} \le \mathbf{S} \le \lambda_{\max}\widetilde{\mathbf{H}}$. Let the sequence of iterates $\{\boldsymbol{u}_j\}_{j=0}^\infty$ be defined by (3.1), and let the quantities σ_- and σ_+ be defined by*

$$\sigma_- := \frac{1}{2}\left[(1 - \rho_{\mathbf{A}})(1 - \omega\lambda_{\min}) + \sqrt{4\rho_{\mathbf{A}} + (1 - \rho_{\mathbf{A}})^2(1 - \omega\lambda_{\min})^2}\right], \tag{3.5a}$$

$$\sigma_+ := \frac{1}{2}\left[(1 + \rho_{\mathbf{A}})(1 + \omega\lambda_{\max}) - 2 + \sqrt{4\rho_{\mathbf{A}} + [(1 + \rho_{\mathbf{A}})(1 + \omega\lambda_{\max}) - 2]^2}\right]. \tag{3.5b}$$

Let $\rho_U := \max\{\sigma_-, \sigma_+\}$. Then we have

$$\|\boldsymbol{u} - \boldsymbol{u}_{j+1}\|_{\boldsymbol{\mathcal{D}}} \le \rho_U \|\boldsymbol{u} - \boldsymbol{u}_j\|_{\boldsymbol{\mathcal{D}}} \quad \forall j \ge 0. \tag{3.6}$$

If the damping parameter $\omega > 0$ is chosen such that

$$\omega\,\lambda_{\max} < 2\,\frac{1 - \rho_{\mathbf{A}}}{1 + \rho_{\mathbf{A}}}, \tag{3.7}$$

then $\rho_U < 1$ and the inexact Uzawa method is convergent.

The proof of Theorem 3.1 is given in Section **??** below. Thus, it is seen that the convergence rate of the inexact Uzawa method can be bounded by ρ_U which depends only on $\rho_{\mathbf{A}}$, ω and on the spectral equivalence between $\widetilde{\mathbf{H}}$ and $\mathbf{S}$. If $\widetilde{\mathbf{H}}$ is spectrally equivalent to $\mathbf{S}$, i.e. with uniform bounds on $\lambda_{\min}$ and $\lambda_{\max}$ independent of the discretization parameters, e.g. the number of time-steps, then the convergence will be robust with respect to these parameters. We will construct in Section 4 below a spectrally equivalent preconditioner $\widetilde{\mathbf{H}}$ that verifies the assumptions in Theorem 3.1, provided only that (2.2) and (3.3) hold, and that suitable preconditioners for certain associated spatial matrices are available.

Remark 3.1 (Condition on the damping parameter). The condition (3.7) is essentially equivalent to the condition given in [45, Thm 4.3], although written in a different form. This assumption is also rather natural, since if $\rho_{\mathbf{A}}$ approaches zero, then the inexact Uzawa method approaches a preconditioned Richardson iteration for $\mathbf{S}$ and (3.7) approaches the standard condition that $\omega < 2/\lambda_{\max}$ for guaranteeing the contraction of the iteration matrix $\mathbf{I} - \omega\widetilde{\mathbf{H}}^{-1}\mathbf{S}$.

Remark 3.2 (Norm of the principal variable). Since we are primarily interested in the second variable in the system (2.8), and since the preconditioners $\widetilde{\mathbf{H}}$ considered here will be shown to be spectrally equivalent to $\mathbf{S}$, we see that the associated norm for the second variable in $\|\cdot\|_{\boldsymbol{\mathcal{D}}}$ is equivalent to $\|\cdot\|_{\mathbf{S}}$, which is natural for the problem, owing to (2.6). In the literature, various other norms have been used in the analysis of parallel algorithms for parabolic equations; a popular choice appears to be the max-norm $\max_{1 \le n \le N} \|v_n\|_M$, see for instance [19]. It turns out that the norm $\|\cdot\|_{\mathbf{S}}$ considered here is stronger than the max-norm; indeed, Corollary 6.2 below shows that, for every $\mathbf{v} \in \mathbb{V}^N$, with $\mathbf{v} = [v_1, \dots, v_N]$, we have the bound $\max_{1 \le n \le N} \|v_n\|_M \le \frac{1}{\sqrt{2}} \|\mathbf{v}\|_{\mathbf{S}}$, without any additional unknown constant. However, in general, this bound has no robust converse, i.e. the norm $\|\cdot\|_{\mathbf{S}}$ is strictly stronger than the max-norm.

Remark 3.3 (Norm of the auxiliary variable). We see in (3.4) that the parameters $\rho_{\mathbf{A}}$ and ω appear in association with the norm for the auxiliary variable $\mathbf{q}$ in the definition of the norm $\|\cdot\|_{\boldsymbol{\mathcal{D}}}$. This can be explained as follows: if one of $\rho_{\mathbf{A}}$ or ω is very small, then it is clear that the error $\mathbf{u} - \mathbf{u}_{j+1}$ for the principal variable is not significantly influenced by the previous error $\mathbf{p} - \mathbf{p}_j$ in the auxiliary variable; this is reflected in the bound (3.6). In the case where $\rho_{\mathbf{A}}$ approaches zero, then the inexact Uzawa method approaches the preconditioned Richardson iteration for $\mathbf{S}$, and the error $\mathbf{u} - \mathbf{u}_{j+1}$ becomes in the limit independent of $\mathbf{p} - \mathbf{p}_j$. For this reason, we see that the presence of the parameters $\rho_{\mathbf{A}}$ and ω in norm $\|\cdot\|_{\boldsymbol{\mathcal{D}}}$ is rather natural for the analysis of the inexact Uzawa method.

Remark 3.4 (Other iterative methods). Although we consider here the inexact Uzawa method for solving the system (2.8), this is by no means the only possible choice. For instance, one alternative is the MINRES algorithm [34]. Given a symmetric positive definite preconditioner, MINRES minimizes a preconditioned residual norm over a Krylov subspace. For example, a suitable yet simple choice of preconditioner for MINRES here would be the block diagonal matrix $\begin{bmatrix} \widetilde{\mathbf{A}} & \\ & \widetilde{\mathbf{H}} \end{bmatrix}$. Then, provided that $\widetilde{\mathbf{H}}$ is spectrally equivalent to $\mathbf{S}$, we may use the spectral bounds in [37] and convergence theory in [21] to show the robust convergence of MINRES. Notice that unlike the inexact Uzawa method above, MINRES does not require a suitably chosen damping parameter ω to have guaranteed convergence. Moreover, the condition (3.3) can be relaxed to the weaker assumption of spectral equivalence.

Proof of Theorem 3.1. The proof essentially follows the approach of [45], which gives sufficient conditions for convergence of inexact Uzawa methods in the context of general saddle-point problems with zero lower diagonal block. However, the saddle-point matrix $\boldsymbol{\mathcal{A}}$ defined in (2.8) has nonzero lower diagonal block, so, strictly speaking, we must check that the approach in [45] can be extended to cover the present situation. Therefore, in this subsection, we adapt the main steps from [45] for the sake of completeness. To remain brief, we do not attempt to give as general a treatment as the one in [45].

Let $\boldsymbol{\mathcal{M}}$ denote the iteration matrix of the inexact Uzawa method, i.e. $\boldsymbol{e}_{j+1} = \boldsymbol{\mathcal{M}} \boldsymbol{e}_j$ for each $j \ge 0$, where $\boldsymbol{e}_j \coloneqq \boldsymbol{u} - \boldsymbol{u}_j$. A simple calculation shows that

$$\boldsymbol{\mathcal{M}} = \begin{bmatrix} \mathbf{I} - \widetilde{\mathbf{A}}^{-1}\mathbf{A} & \widetilde{\mathbf{A}}^{-1}\mathbf{K} \\ -\omega\widetilde{\mathbf{H}}^{-1}\mathbf{K}^{\top}\left(\mathbf{I} - \widetilde{\mathbf{A}}^{-1}\mathbf{A}\right) & \mathbf{I} - \omega\widetilde{\mathbf{H}}^{-1}\widetilde{\mathbf{S}} \end{bmatrix}, \tag{3.8}$$

where $\widetilde{\mathbf{S}} \coloneqq \mathbf{K}^{\top}\widetilde{\mathbf{A}}^{-1}\mathbf{K} + \mathbf{K} + \mathbf{K}^{\top} + \mathbf{A}$. Therefore, we can obtain (3.6) by showing that $\|\boldsymbol{\mathcal{M}}\|_{\boldsymbol{\mathcal{D}}} \le \rho_U$, where $\|\boldsymbol{\mathcal{M}}\|_{\boldsymbol{\mathcal{D}}}$ denotes the operator norm of $\boldsymbol{\mathcal{M}}$ with respect to the norm $\boldsymbol{\mathcal{D}}$ from (3.4).

Lemma 3.2. *Assume that* (3.3) *holds, that* $\omega > 0$*, and that* $\widetilde{\mathbf{H}}$ *is symmetric positive definite. Then we have the bound* $\|\boldsymbol{\mathcal{M}}\|_{\boldsymbol{\mathcal{D}}} \le \max_{\mu \in \Sigma} |\mu|$ *where* $\Sigma \subset \mathbb{R}$ *denotes the set of eigenvalues of the*

generalized symmetric eigenvalue problem:

$$\boldsymbol{\mathcal{N}}\boldsymbol{v} = \mu\boldsymbol{\mathcal{E}}\boldsymbol{v}, \quad \boldsymbol{\mathcal{N}} := \begin{bmatrix} \widetilde{\mathbf{A}} & -\mathbf{K} \\ -\mathbf{K}^\top & \widetilde{\mathbf{S}} - \frac{1}{\omega}\widetilde{\mathbf{H}} \end{bmatrix}, \qquad \boldsymbol{\mathcal{E}} := \begin{bmatrix} \frac{1}{\rho_\mathbf{A}}\widetilde{\mathbf{A}} & \\ & \frac{1}{\omega}\widetilde{\mathbf{H}} \end{bmatrix}. \tag{3.9}$$

Proof. Let $\boldsymbol{\mathcal{D}} := \begin{bmatrix} \omega\rho_\mathbf{A}\widetilde{\mathbf{A}} & \\ & \widetilde{\mathbf{H}} \end{bmatrix}$ denote the matrix inducing the norm $\|\cdot\|_{\boldsymbol{\mathcal{D}}}$ in (3.4). A simple calculation shows that the matrix $\boldsymbol{\mathcal{M}}$ can be factorized as $\boldsymbol{\mathcal{M}} = -\boldsymbol{\mathcal{R}}\boldsymbol{\mathcal{N}}\boldsymbol{\mathcal{Q}}$, where $\boldsymbol{\mathcal{N}}$ is as in (3.9), and $\boldsymbol{\mathcal{R}} := \begin{bmatrix} \widetilde{\mathbf{A}}^{-1} & \\ & \omega\widetilde{\mathbf{H}}^{-1} \end{bmatrix}$, $\boldsymbol{\mathcal{Q}} := \begin{bmatrix} \widetilde{\mathbf{A}}^{-1}\mathbf{A}-\mathbf{I} & \\ & \mathbf{I} \end{bmatrix}$. It is furthermore easy to check that $\omega\boldsymbol{\mathcal{E}}^{-1} = \boldsymbol{\mathcal{R}}^\top\boldsymbol{\mathcal{D}}\boldsymbol{\mathcal{R}}$, and that $\boldsymbol{\mathcal{Q}}^\top\boldsymbol{\mathcal{E}}\boldsymbol{\mathcal{Q}} \le \frac{1}{\omega}\boldsymbol{\mathcal{D}}$ as a consequence of (3.3). Therefore, we find that

$$\begin{aligned} \|\boldsymbol{\mathcal{M}}\|_{\boldsymbol{\mathcal{D}}}^2 &= \sup_{\boldsymbol{v}\in\mathbb{V}^N\times\mathbb{V}^N\setminus\{0\}} \frac{\boldsymbol{v}^\top\boldsymbol{\mathcal{Q}}^\top\boldsymbol{\mathcal{N}}\boldsymbol{\mathcal{E}}^{-1}\boldsymbol{\mathcal{N}}\boldsymbol{\mathcal{Q}}\boldsymbol{v}}{\boldsymbol{v}^\top(\frac{1}{\omega}\boldsymbol{\mathcal{D}})\boldsymbol{v}} \\ &\le \sup_{\boldsymbol{w}\in\mathbb{V}^N\times\mathbb{V}^N\setminus\{0\}} \frac{\boldsymbol{w}^\top\boldsymbol{\mathcal{N}}\boldsymbol{\mathcal{E}}^{-1}\boldsymbol{\mathcal{N}}\boldsymbol{w}}{\boldsymbol{w}^\top\boldsymbol{\mathcal{E}}\boldsymbol{w}} = \|\boldsymbol{\mathcal{E}}^{-1}\boldsymbol{\mathcal{N}}\|_{\boldsymbol{\mathcal{E}}}^2, \end{aligned}$$

where we obtain the inequality in second line above by substituting $\boldsymbol{w} = \boldsymbol{\mathcal{Q}}\boldsymbol{v}$ and using the bound $\boldsymbol{\mathcal{Q}}^\top\boldsymbol{\mathcal{E}}\boldsymbol{\mathcal{Q}} \le \frac{1}{\omega}\boldsymbol{\mathcal{D}}$ given above. Since the matrix $\boldsymbol{\mathcal{E}}^{-1}\boldsymbol{\mathcal{N}}$ is symmetric with respect to the $\boldsymbol{\mathcal{E}}$-inner product, we see that $\|\boldsymbol{\mathcal{E}}^{-1}\boldsymbol{\mathcal{N}}\|_{\boldsymbol{\mathcal{E}}} = \max_{\mu\in\Sigma}|\mu|$ where Σ denotes the set of eigenvalues in (3.9). This implies that $\|\boldsymbol{\mathcal{M}}\|_{\boldsymbol{\mathcal{D}}} \le \max_{\mu\in\Sigma}|\mu|$. $\qquad\square$

The next step in the proof of Theorem 3.1 is to bound the eigenvalues $\mu \in \Sigma$ defined by (3.9).

Lemma 3.3. *Assume that* (3.3) *holds, that* $\omega > 0$, *and that* $\widetilde{\mathbf{H}}$ *is symmetric positive definite. Let* $0 < \lambda_{\min} \le \lambda_{\max}$ *be constants such that* $\lambda_{\min}\widetilde{\mathbf{H}} \le \mathbf{S} \le \lambda_{\max}\widetilde{\mathbf{H}}$. *If* $\mu \in \Sigma$ *is a negative eigenvalue of* (3.9), *then* $|\mu| \le \sigma_-$ *where* σ_- *is defined* (3.5a). *If* $\mu \in \Sigma$ *is a positive eigenvalue, then* $|\mu| \le \sigma_+$ *with* σ_+ *defined in* (3.5b).

Proof. We start by showing the bound $|\mu| \le \sigma_-$ for any negative eigenvalue $\mu < 0$. First, let $\lambda = -\mu = |\mu|$; then, since $(1+\frac{\lambda}{\rho_\mathbf{A}})\widetilde{\mathbf{A}}$ is positive definite, we may use the eigenvalue problem (3.9) to find that there exists a nonzero $\mathbf{v} \in \mathbb{V}$ such that

$$\frac{\lambda}{\omega}\widetilde{\mathbf{H}}\mathbf{v} = \frac{1}{\omega}\widetilde{\mathbf{H}}\mathbf{v} - \widetilde{\mathbf{S}}\mathbf{v} + \frac{\rho_\mathbf{A}}{\rho_\mathbf{A}+\lambda}\mathbf{K}^\top\widetilde{\mathbf{A}}^{-1}\mathbf{K}\mathbf{v}.$$

By taking the inner product with $\mathbf{v}$ and applying the inequalities (3.3) and $\widetilde{\mathbf{S}} \ge (1-\rho_\mathbf{A})\lambda_{\min}\widetilde{\mathbf{H}}$, we eventually find that λ satisfies the inequality

$$\lambda^2 - \lambda(1-\rho_\mathbf{A})(1-\omega\lambda_{\min}) \le \rho_\mathbf{A}, \tag{3.10}$$

from which we deduce that $|\mu| = \lambda \le \sigma_-$ as claimed.

Next we consider the case of positive eigenvalues $\mu > 0$. First note that $\rho_\mathbf{A} < \sigma_+$; therefore we need only consider the case of $\mu > \rho_\mathbf{A}$, otherwise if $\mu \le \rho_\mathbf{A}$ then $\mu < \sigma_+$ and there is nothing left to show. If $\mu > \rho_\mathbf{A}$, then $(1 - \mu/\rho_\mathbf{A})\widetilde{\mathbf{H}}$ is nonsingular, and we find that there is a nonzero $\mathbf{v} \in \mathbb{V}$ such that

$$\frac{(1+\mu)}{\omega}\widetilde{\mathbf{H}}\mathbf{v} = \widetilde{\mathbf{S}}\mathbf{v} + \frac{\rho_\mathbf{A}}{\mu-\rho_\mathbf{A}}\mathbf{K}^\top\widetilde{\mathbf{A}}^{-1}\mathbf{K}\mathbf{v}.$$

We then apply the inequality $\widetilde{\mathbf{S}} \le (1+\rho_\mathbf{A})\lambda_{\max}\widetilde{\mathbf{H}}$ to find eventually that

$$\mu^2 - \mu\left[(1+\rho_\mathbf{A})(1+\omega\lambda_{\max}) - 2\right] \le \rho_\mathbf{A}, \tag{3.11}$$

which implies the inequality $\mu \le \sigma_+$ as claimed. $\qquad\square$

9

Proof of Theorem 3.1 The combination of Lemmas 3.2 and 3.3 implies that $\|\mathcal{M}\| \leq \rho_U$ where $\rho_U := \max\{\sigma_-, \sigma_+\}$; this implies the bound (3.6) on the sequence of iterates. We next show that $\rho_U < 1$ if (3.7) holds. Indeed, it is easy to check that $\sigma_- < 1$ for all $\omega\lambda_{\min} > 0$, and $\sigma_+ < 1$ if and only if $\rho_{\mathbf{A}} + (1 + \rho_{\mathbf{A}})(1 + \omega\lambda_{\max}) - 2 < 1$, which is equivalent to the condition in (3.7). □

4 Schur complement preconditioner

We have seen in the previous section that a convergent iterative solver for (2.8) can be obtained provided that we have at our disposal a spectrally equivalent preconditioner for the Schur complement $\mathbf{S}$. In this section, we propose such a preconditioner that is well-suited for parallel computations. We shall denote this preconditioner by $\mathbf{H}$ in the case of exact spatial solvers, and by $\widetilde{\mathbf{H}}$ in the practical case of approximate spatial solvers. To motivate our construction, we consider the following example.

Example 4.1 (Uniform time-steps with constant coefficients). Consider momentarily the case where $\tau_n = \tau$ and $A_n = A$ for all $n = 1, \ldots, N$, i.e. where the time-steps and spatial operators are constant in time. Then the Schur complement matrix $\mathbf{S}$ is

$$\mathbf{S} = \frac{1}{\tau}K^\top K \otimes MA^{-1}M + (K + K^\top) \otimes M + \tau\mathrm{Id}_N \otimes A,$$

where K is from (2.4), where Id_N is the $N \times N$ identity matrix and where $\otimes$ denotes the Kronecker product. It is then easy to see that the matrices $K^\top K$ and $K + K^\top$ are both symmetric positive definite, since

$$K^\top K = \begin{pmatrix} 2 & -1 & & \\ -1 & 2 & -1 & \\ & \ddots & \ddots & -1 \\ & & -1 & 1 \end{pmatrix}, \quad K + K^\top = \begin{pmatrix} 2 & -1 & & \\ -1 & 2 & -1 & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix}. \tag{4.1}$$

As a special case of the results of Section 6, we will see that $\mathbf{S}$ is spectrally equivalent to a simpler matrix where the middle term $(K + K^\top) \otimes M$ is dropped. Since this simpler matrix involves only a sum of two Kronecker products of matrices, it can be block-diagonalized with respect to time. The key observation is that the matrix $K^\top K$ has explicitly known (generalized) eigenvalues and eigenvectors, which are related to discrete Sine transforms (DST), which are well-suited for time-parallelism. This suggests using the DST in time to obtain a block-diagonal and thus time-parallel preconditioner. □

To define the preconditioners, we use the type-III DST, represented by the matrix $\mathbf{\Phi}$ that maps $\mathbf{u} \in \mathbb{V}^N$ to $\hat{\mathbf{u}} = \mathbf{\Phi}\,\mathbf{u} \in \mathbb{V}^N$, where $\hat{\mathbf{u}} = [\hat{u}_1, \ldots, \hat{u}_N]$ is defined by

$$\hat{u}_k := \frac{2}{N}\sum_{n=1}^{N}\beta_n u_n \sin\left(\frac{(2k-1)n\pi}{2N}\right), \quad k = 1, \ldots, N, \tag{4.2}$$

where

$$\beta_n := \begin{cases} 1 & \text{if } 1 \leq n < N, \\ \frac{1}{2} & \text{if } n = N. \end{cases} \tag{4.3}$$

The inverse map $\mathbf{\Phi}^{-1}$ that satisfies $\mathbf{u} = \mathbf{\Phi}^{-1}\hat{\mathbf{u}}$ is simply given by the type-II DST

$$u_n = \sum_{k=1}^{N}\hat{u}_k \sin\left(\frac{(2k-1)n\pi}{2N}\right), \quad n = 1, \ldots, N. \tag{4.4}$$

10

We stress that the actions of the transformations are with respect to the temporal components of the vectors $\mathbf{u}$ and $\hat{\mathbf{u}}$, since each term in the sums of (4.2) and (4.4) is a vector in $\mathbb{V}$. Thus the DST used here represents a change of the temporal basis. Furthermore, the actions of the matrices $\mathbf{\Phi}$ and $\mathbf{\Phi}^{-1}$ (and their transposes) can all be implemented efficiently through recursive splittings of the summations, leading to fast implementations akin to the FFT.

The ideal preconditioner $\mathbf{H}$ is defined by

$$\mathbf{H} := \mathbf{\Phi}^{\top} \hat{\mathbf{H}} \mathbf{\Phi}, \quad \hat{\mathbf{H}} := \frac{N}{2\tau} \operatorname{diag} \left\{ H_k A^{-1} H_k \right\}_{k=1}^{N}, \tag{4.5}$$

where τ and A are as in (2.2), and where the matrices H_k are defined by

$$H_k := \mu_k M + \tau A, \quad \mu_k := 2\sin\left(\frac{(2k-1)\pi}{4N}\right) \quad \forall\, k = 1, \ldots, N. \tag{4.6}$$

Notice that $\mu_k > 0$ for each $1 \le k \le N$. The inversion of $\mathbf{H}$ can be performed by composition of $\mathbf{\Phi}^{-T}, \hat{\mathbf{H}}^{-1}$ and $\mathbf{\Phi}^{-1}$. As mentioned above, the actions of $\mathbf{\Phi}^{-T}$ and $\mathbf{\Phi}^{-1}$ can be computed by fast DST algorithms akin to the FFT, and the application of $\hat{\mathbf{H}}^{-1}$ simply requires the solution of linear systems for weighted implicit Euler steps and can be parallelized over the blocks $k = 1, \ldots, N$.

4.1 Approximations

The analysis of iterative solvers given below will allow for approximations to be made in the application of the inverse of $\mathbf{H}$. More precisely, we consider approximations $\widetilde{\mathbf{H}}$ of $\mathbf{H}$ given by

$$\widetilde{\mathbf{H}} := \mathbf{\Phi}^{\top} \hat{\mathbf{H}}_{\mathrm{approx}} \mathbf{\Phi}, \quad \hat{\mathbf{H}}_{\mathrm{approx}} := \frac{N}{2\tau} \operatorname{diag} \left\{ \widetilde{H}_k A^{-1} \widetilde{H}_k \right\}_{k=1}^{N}, \tag{4.7}$$

where, for each $k = 1, \ldots, N$, the symmetric positive definite matrix $\widetilde{H}_k$ represents an approximation of H_k. We have in mind cases where each matrix $\widetilde{H}_k$ is obtained from a standard solver for the matrices H_k, for example by multigrid or domain decomposition methods. For the analysis, we shall assume that there exist positive constants γ and Γ such that, for all $k = 1, \ldots, N$,

$$\gamma\, \widetilde{H}_k A^{-1} \widetilde{H}_k \le H_k A^{-1} H_k \le \Gamma\, \widetilde{H}_k A^{-1} \widetilde{H}_k. \tag{4.8}$$

By comparing (4.7) with (4.5), it is then clear that the matrices $\mathbf{H}$ and $\widetilde{\mathbf{H}}$ are spectrally equivalent, with $\gamma \widetilde{\mathbf{H}} \le \mathbf{H} \le \Gamma \widetilde{\mathbf{H}}$.

4.2 Spectral bounds for the Schur complement preconditioners

Our main result is that the preconditioner $\mathbf{H}$ defined in (4.5), and its approximation $\widetilde{\mathbf{H}}$ defined in (4.7), are spectrally equivalent to $\mathbf{S}$ with bounds showing the explicit dependence of the constants on α, γ and Γ.

Theorem 4.1 (Spectral equivalence). *Assume* (2.2). *Then, for any number of time-steps N and any symmetric positive definite matrices M and $\{A_n\}_{n=1}^{N}$, the matrices $\mathbf{H}$ and $\mathbf{S}$ are spectrally equivalent with the following bounds*

$$\frac{1}{2\alpha}\, \mathbf{H} \le \mathbf{S} \le 3\alpha\, \mathbf{H}. \tag{4.9}$$

Furthermore, if (4.8) *also holds, then $\widetilde{\mathbf{H}}$ defined in* (4.7) *is spectrally equivalent to $\mathbf{S}$ with the following bounds*

$$\frac{\gamma}{2\alpha}\, \widetilde{\mathbf{H}} \le \mathbf{S} \le 3\alpha\Gamma\, \widetilde{\mathbf{H}}. \tag{4.10}$$

The proof of Theorem 4.1 is the subject of Sections 6 and 7. Theorem 4.1 implies that we can take $\lambda_{\min} = \frac{\gamma}{2\alpha}$ and $\lambda_{\max} = 3\alpha\Gamma$ in Theorem 3.1.

Remark 4.1 (Robust convergence). The combination of Theorems 3.1 and 4.1 leads to bounds on the convergence rate of the method that depend only on the the constants in the assumptions (2.2), (3.3), (4.8), and on ω. In many practical applications, these assumptions are satisfied with uniformly bounded constants independent of parameters that determine the spatial matrices M and A_n, such as the spatial mesh size, in which case the convergence will be robust.

Remark 4.2 (Extension to locally refined time-steps). The quasi-uniformity assumption (2.2) is required in the analysis of the preconditioner $\mathbf{H}$ owing to the DST. The inf-sup stability (2.2) and the stable symmetric reformulation (2.8) do not depend on (2.2), and therefore it appears that various extensions may be possible in future work, provided suitable modifications are made in the construction of $\mathbf{H}$. The inclusion of spatial adaptivity, where the space $\mathbb{V}$ may change between time-steps, represents also a further challenge for future work.

5 Parallel complexity

Following [25, 26], the notion of parallel complexity is understood here as the theoretical computational cost assuming the availability of sufficiently many processors, and ignoring communication costs. It is therefore of interest as an intrinsic property of the given algorithm. Since we are primarily interested in the time-parallelism of the algorithm, we shall focus on the dependence on the number of time-steps N.

In order to treat the costs related to spatial operations related to $\mathbb{V}$ in a general way, we introduce the following elementary constants.

- Let $C_{\mathbb{V}}^{\mathrm{add}}$ denote the cost of additions and subtractions of vectors in $\mathbb{V}$; more precisely, $C_{\mathbb{V}}^{\mathrm{add}}$ is the maximal cost of the operation $(v, w, c) \mapsto v + c\,w$, where $v, w \in \mathbb{V}$ and $c \in \mathbb{R}$.

- Let $C_{\mathbb{V}}^{\mathrm{mult}}$ denote the maximal cost of performing a matrix vector product $(L, v) \mapsto L\,v$, where $v \in \mathbb{V}$ and the matrix L is one of M, A or A_n, $n = 1, \ldots, N$.

- Let $C_{\mathbb{V}}^{\mathrm{prec}}$ denote the maximal cost of performing the action of the spatial preconditioners, i.e. the cost of the matrix-vector product $(L, v) \mapsto L\,v$ where L is one of the $\widetilde{A_n}^{-1}$ or $\widetilde{H_k}^{-1}$, for $n, k = 1, \ldots, N$.

Given that the proposed algorithm allows the re-use of existing spatial solvers, it is clear that in many applications there can be significant spatial parallelism as well; see the experiments in Section 8.6. We distinguish the costs of these different operations since in practice they may be rather different; for instance, we expect that $C_{\mathbb{V}}^{\mathrm{add}}$ will be smaller than $C_{\mathbb{V}}^{\mathrm{mult}}$ or $C_{\mathbb{V}}^{\mathrm{prec}}$.

We can now analyse the parallel complexity of the inexact Uzawa method using $\widetilde{\mathbf{H}}$ as defined in (4.7). For fixed constants $\rho_{\mathbf{A}}$, ω, α, γ and Γ, the convergence rate of the algorithm is robust with respect to the number of time-steps. Therefore, for any $\varepsilon > 0$, at most $O(\log \varepsilon^{-1})$ iterations are required to achieve a relative reduction of the residual by a tolerance ε. The total cost is then based on the number of iterations required multiplied by the cost per iteration. Each iteration of the inexact Uzawa method (3.1) requires

- A fixed number of matrix vector products with $\mathbf{K}$, $\mathbf{A}$ and $\mathbf{K}^{\top}$, each of which has parallel complexity $O(C_{\mathbb{V}}^{\mathrm{mult}} + C_{\mathbb{V}}^{\mathrm{add}})$, independently of N, since each of these matrices is block-sparse.

- A fixed number of vector additions/subtractions on $\mathbb{V}^N \times \mathbb{V}^N$, with parallel complexity $O(C_{\mathbb{V}}^{\mathrm{add}})$, independently of N.

- The application of $\widetilde{\mathbf{A}}^{-1}$, which has parallel complexity $O(C_{\mathbb{V}}^{\mathrm{prec}})$, independently of N, since $\widetilde{\mathbf{A}}$ is block diagonal.

- The application of $\widetilde{\mathbf{H}}^{-1}$, which we discuss further below.

The matrix-vector product with $\widetilde{\mathbf{H}}^{-1}$ involves the application of the DST transformations related to $\boldsymbol{\Phi}^{-1}$ and $\boldsymbol{\Phi}^{-\top}$, and the application of the block diagonal matrix $\hat{\mathbf{H}}_{\mathrm{approx}}^{-1}$ from (4.7). It is clear that the application of $\hat{\mathbf{H}}_{\mathrm{approx}}^{-1}$ has parallel complexity $O(C_{\mathbb{V}}^{\mathrm{mult}} + C_{\mathbb{V}}^{\mathrm{prec}})$, and is independent of N. It remains only to consider the parallel complexity of the DST. It is clear that if N is, for example, an integer power of 2, then the DST has parallel complexity of $O(C_{\mathbb{V}}^{\mathrm{add}}(\log N + 1))$, as shown by recursive splitting of the summation in (4.4). From a theoretical perspective, the same parallel complexity bound can also be achieved for general N, since a DST of general length N can be obtained by Bluestein's method, which involves two zero-padded discrete Fourier transforms with length equal to a power of two of same order as N; see [36] for further details.

In summary, the parallel complexity of each inexact Uzawa iteration is then bounded by

$$O\left(C_{\mathbb{V}}^{\mathrm{add}}(\log N + 1) + C_{\mathbb{V}}^{\mathrm{mult}} + C_{\mathbb{V}}^{\mathrm{prec}}\right). \tag{5.1}$$

It is thus seen that each iteration has a parallel complexity that grows at most logarithmically with N. The strong decoupling of the method between time and space can also be seen through the fact that the terms involving $C_{\mathbb{V}}^{\mathrm{mult}}$ and $C_{\mathbb{V}}^{\mathrm{prec}}$ are independent of N. Note that the terms of order $\log N$ are not necessarily dominant in actual computations, since $C_{\mathbb{V}}^{\mathrm{prec}}$ and $C_{\mathbb{V}}^{\mathrm{mult}}$ are often significantly larger than $C_{\mathbb{V}}^{\mathrm{add}}$; indeed, in our experiments, the cost of the DST is significantly lower than the cost of solving the associated spatial problems, see Section 8.4 below for further details.

Remark 5.1 (Comparison with the parareal method). As mentioned above, it is known that the best parallel complexity of the parareal algorithm using the implicit Euler scheme grows as $\sqrt{N}$, see [4]. More precisely, in the current notation, it can shown to be of order $O((C_{\mathbb{V}}^{\mathrm{prec}} + C_{\mathbb{V}}^{\mathrm{add}} + C_{\mathbb{V}}^{\mathrm{mult}})\sqrt{N})$. It is then seen that the difference with (5.1) is not only in the order of dependence on N, but also in the associated constants due to the spatial problems.

6 Inf-sup stability of the implicit Euler method

As mentioned in the introduction and in Section 2.1, the analysis of the proposed algorithm will make use of the variational formulation the implicit Euler method.

6.1 Time-global variational formulation

We now present the variational formulation of (2.3). Define the space

$$\mathbb{V}_\tau := \oplus_{n=1}^{N} \mathcal{P}_0(I_n; \mathbb{V}), \tag{6.1}$$

where $\mathcal{P}_0(I_n; \mathbb{V})$ denotes the set of $\mathbb{V}$-valued functions that are constant-in-time over each time-step interval I_n, for $n = 1, \ldots, N$. In other words, a function $v \in \mathbb{V}_\tau$ if and only if v is a piecewise constant function on each time-step I_n, with $v|_{I_n} \in \mathbb{V}$. A basis can be constructed for $\mathbb{V}_\tau$ by considering the tensor product between a basis of $\mathbb{V}$ and a basis for $\oplus_{n=1}^{N} \mathcal{P}_0(I_n; \mathbb{R})$ the space of real-valued piecewise-constant functions. Thus, a standard choice for the temporal basis of $\mathbb{V}_\tau$ is given as follows: for any $v \in \mathbb{V}_\tau$, we have $v = \sum_{n=1}^{N} v_n \chi_n$, where the coefficients $v_n \in \mathbb{V}$, and where χ_n is the indicator function of I_n, for each $n = 1, \ldots, N$; thus, $v_n = v|_{I_n}$ the restriction of v to I_n.

Define the reconstruction operator $\mathcal{I}\colon \mathbb{V}_\tau \to H^1(0,T;\mathbb{V}) \cap \oplus_{n=1}^N \mathcal{P}_1(I_n;\mathbb{V})$ by

$$(\mathcal{I}v)(t) := v_n - \frac{t_n - t}{\tau_n}(\!|v|\!)_{n-1}, \quad t \in (t_{n-1}, t_n], \tag{6.2}$$

where $\mathcal{P}_1(I_n;\mathbb{V})$ denotes the set of piecewise affine functions on each time-step I_n, and where $(\!|\cdot|\!)$ denotes the jump operator defined by $(\!|v|\!)_{n-1} := v_n - v_{n-1}$, with the convention that $v_0 = 0$ for all $v \in \mathbb{V}_\tau$ to simplify the notation. It follows that $\mathcal{I}$ defines a linear operator on $\mathbb{V}_\tau$, and that, for any $v \in \mathbb{V}_\tau$, the function $\mathcal{I}v$ is a piecewise-affine continuous function in time, with $\mathcal{I}v(t_n) = v_n$ for each $n = 1, \ldots, N$. Thus the function $\mathcal{I}v \in H^1(0,T;\mathbb{V})$, and $\mathcal{I}v$ has a weak temporal derivative $\partial_t \mathcal{I}v \in \mathbb{V}_\tau$ with $(\partial_t \mathcal{I}v)|_{I_n} = \frac{1}{\tau_n}(v_n - v_{n-1})$ for all $n = 1, \ldots, N$. Additionally, the function $\mathcal{I}v$ also satisfies the initial condition $\mathcal{I}v(0) = 0$.

It is easy to see that the discrete problem (2.3) can then be equivalently rewritten as: find $u \in \mathbb{V}_\tau$ such that

$$b(u,v) = \ell(v) \quad \forall\, v \in \mathbb{V}_\tau, \tag{6.3}$$

where the bilinear form $b(\cdot,\cdot)$ and linear functional $\ell(\cdot)$ are defined by

$$b(u,v) := \sum_{n=1}^N \int_{I_n} (\partial_t \mathcal{I}u, v)_M + (u,v)_{A_n}\, \mathrm{d}t, \quad \ell(v) := (u_I, v_1)_M + \sum_{n=1}^N \int_{I_n} (f_n, v)_M\, \mathrm{d}t, \tag{6.4}$$

where we recall that u_I is the given initial datum.

6.2 Inf-sup stability

As explained above, for many parabolic equations, the natural norm for the temporal derivative is a dual norm induced by the spatial differential operator of the problem. In the discrete setting, these dual norms admit the following characterizations. First, we introduce the dual norms $\|\cdot\|_{MA_n^{-1}M}$, for each $n = 1, \ldots, N$, and $\|\cdot\|_{MA^{-1}M}$ on $\mathbb{V}$, defined by

$$\|w\|_{MA_n^{-1}M} := \sup_{v \in \mathbb{V}\backslash\{0\}} \frac{(w,v)_M}{\|v\|_{A_n}}, \quad \|w\|_{MA^{-1}M} := \sup_{v \in \mathbb{V}\backslash\{0\}} \frac{(w,v)_M}{\|v\|_A} \qquad \forall\, w \in \mathbb{V}. \tag{6.5}$$

Since M is positive definite, and since $\mathbb{V}$ is finite dimensional, it is clear that $\|\cdot\|_{MA^{-1}M}$ and $\|\cdot\|_{MA_n^{-1}M}$ define norms on $\mathbb{V}$. The notation $\|\cdot\|_{MA_n^{-1}M}$, is justified by the fact that this norm is induced by the inner product of the matrix $MA_n^{-1}M$, since it is straightforward to show that $\|w\|_{MA_n^{-1}M}^2 = w^\top MA_n^{-1}Mw$ and $\|w\|_{MA^{-1}M}^2 = w^\top MA^{-1}Mw$ for all $w \in \mathbb{V}$, where we again identify $w \in \mathbb{V}$ with its vector representation. It is easy to show that the spectral equivalence (2.2) implies that $\frac{1}{\tau_n}MA_n^{-1}M$ and $\frac{1}{\tau}MA^{-1}M$ are also spectrally equivalent:

$$\frac{1}{\alpha}M(\tau A)^{-1}M \le M(\tau_n A_n)^{-1}M \le \alpha\, M(\tau A)^{-1}M. \tag{6.6}$$

We introduce the following norms on the space $\mathbb{V}_\tau$

$$\|v\|_{\mathbb{A}}^2 := \sum_{n=1}^N \int_{I_n} \|v\|_{A_n}^2\, \mathrm{d}t, \tag{6.7a}$$

$$\|u\|_{\mathbb{S}}^2 := \sum_{n=1}^N \int_{I_n} \|\partial_t \mathcal{I}u\|_{MA_n^{-1}M}^2 + \|u\|_{A_n}^2\, \mathrm{d}t + \|u_N\|_M^2 + \sum_{n=1}^N \|(\!|u|\!)_{n-1}\|_M^2, \tag{6.7b}$$

14

for all functions u and v in $\mathbb{V}_\tau$, where the jump operators $(\!|\cdot|\!)$ were defined in Section 6.1. It is easy to see that $\mathbf{A}$ is the inner-product matrix (Gram matrix) of the inner product that induces the norm $\|\cdot\|_{\mathbb{A}}$. In other words, $\|\mathbf{v}\|_{\mathbf{A}} = \|v\|_{\mathbb{A}}$ for any $v \in \mathbb{V}_\tau$ with vector representation $\mathbf{v}$ with respect to the standard basis of $\mathbb{V}_\tau$. The norm $\|\cdot\|_{\mathbb{S}}$ is induced by the coercive bilinear form $s(\cdot,\cdot)\colon \mathbb{V}_\tau \times \mathbb{V}_\tau \to \mathbb{R}$ defined by

$$s(u,v) := \sum_{n=1}^{N} \int_{I_n} (\partial_t \mathcal{I}u, \partial_t \mathcal{I}v)_{MA_n^{-1}M} + (u,v)_{A_n} \, \mathrm{d}t + j(u,v), \tag{6.8}$$

where the symmetric bilinear form $j(\cdot,\cdot)$ is defined by

$$j(u,v) := (u_N, v_N)_M + \sum_{n=1}^{N} ((\!|u|\!)_{n-1}, (\!|v|\!)_{n-1})_M. \tag{6.9}$$

The following theorem shows that $\mathbf{S}$ is the matrix representation of $s(\cdot,\cdot)$, thereby establishing the equivalence between the discrete inf-sup identity in (2.6) and its variational representation given below, and thus showing the connection to the continuous one in (1.1).

Theorem 6.1. *The matrix $\mathbf{S}$ is the matrix representation of the bilinear form $s(\cdot,\cdot)$ in the standard basis of $\mathbb{V}_\tau$. Therefore, (2.6) is equivalent to*

$$\|u\|_{\mathbb{S}} = \sup_{v \in \mathbb{V}_\tau \setminus \{0\}} \frac{b(u,v)}{\|v\|_{\mathbb{A}}}. \tag{6.10}$$

Proof. First, note that it is clear that $s(\cdot,\cdot)$ defines an inner-product that induces the norm $\|\cdot\|_{\mathbb{S}}$, since it is clear that $s(v,v) = \|v\|_{\mathbb{S}}^2$ for any $v \in \mathbb{V}_\tau$. The equivalence of (6.10) and (2.6) will follow immediately after it is shown that $\mathbf{S}$ is the matrix for $s(\cdot,\cdot)$ under the standard basis of $\mathbb{V}_\tau$. Hence, we need only show that $\mathbf{v}^\top \mathbf{S} \mathbf{u} = s(u,v)$ for any $u, v \in \mathbb{V}_\tau$ with respective vector representations $\mathbf{u}$ and $\mathbf{v} \in \mathbb{V}^N$.

To begin, define the operator $P\colon \mathbb{V}_\tau \to \mathbb{V}_\tau$ that maps a function v to its optimal test function, namely

$$Pv|_{I_n} := A_n^{-1}M(\partial_t \mathcal{I}v)|_{I_n} + v|_{I_n} \quad \forall\, v \in \mathbb{V}_\tau. \tag{6.11}$$

As explained above, the time derivative $\partial_t \mathcal{I}v$ is indeed in $\mathbb{V}_\tau$ owing to the facts that $\mathcal{I}v_h \in H^1(0,T;\mathbb{V})$ and $\mathcal{I}v$ is piecewise affine. Therefore, we indeed have $Pv \in \mathbb{V}_\tau$ for all $v \in \mathbb{V}_\tau$. It is easy to check that the matrix representation of the operator P is the matrix $\mathbf{P}$ defined in Section 2.1. Furthermore, recall that $\mathbf{S} = \mathbf{P}^\top \mathbf{B}$. Therefore, if $u, v \in \mathbb{V}_\tau$ have respective vector representations $\mathbf{u}$ and $\mathbf{v} \in \mathbb{V}^N$, then $\mathbf{v}^\top \mathbf{S} \mathbf{u} = b(u, Pv)$. We now simply show that $b(u, Pv) = s(u,v)$ by calculation: for arbitrary $u, v \in \mathbb{V}_\tau$, $b(u, Pv) = \sum_{n=1}^{N} \int_{I_n} (\partial_t \mathcal{I}u, Pv)_M + (u, Pv)_{A_n} \, \mathrm{d}t$. By expanding the terms and simplifying, we obtain

$$b(u, Pv) = \sum_{n=1}^{N} \int_{I_n} (\partial_t \mathcal{I}u, \partial_t \mathcal{I}v)_{MA_n^{-1}M} + (u,v)_{A_n} + (\partial_t \mathcal{I}u, v)_M + (u, \partial_t \mathcal{I}v)_M \, \mathrm{d}t, \tag{6.12}$$

where we have made use of the identities $\left(\partial_t \mathcal{I}u, A_n^{-1}M \partial_t \mathcal{I}v\right)_M = (\partial_t \mathcal{I}u, \partial_t \mathcal{I}v)_{MA_n^{-1}M}$ and $\left(u, A_n^{-1}M \partial_t \mathcal{I}v\right)_{A_n} = (u, \partial_t \mathcal{I}v)_M$ for any $u, v \in \mathbb{V}_\tau$. To complete the proof that $s(u,v) = b(u, Pv)$, it remains only to show that

$$\sum_{n=1}^{N} \int_{I_n} (\partial_t \mathcal{I}u, v)_M + (u, \partial_t \mathcal{I}v)_M \, \mathrm{d}t = j(u,v), \tag{6.13}$$

15

where $j(\cdot,\cdot)$ is defined as in (6.9). Recall that $\partial_t \mathcal{I}u|_{I_n} = \frac{1}{\tau_n}(u_n - u_{n-1})$ for each $1 \leq n \leq N$, so we obtain $\int_{I_n}(\partial_t \mathcal{I}u, v)_M + (u, \partial_t \mathcal{I}v)_M \, \mathrm{d}t = (\langle\!| u |\!\rangle_{n-1}, \langle\!| v |\!\rangle_{n-1})_M + (u_n, v_n)_M - (u_{n-1}, v_{n-1})_M$, where we obtained the identity by adding and subtracting $(u_n - u_{n-1}, v_{n-1})_M$. Simplifying the telescoping sum then leads to (6.13). This completes the proof that $s(u,v) = b(u, Pv)$ and that $\mathbf{S}$ is the matrix representing $s(\cdot,\cdot)$ in the standard basis of $\mathbb{V}_\tau$. $\qquad\square$

A consequence of Theorem 6.1 is that the norm $\|\cdot\|_{\mathbb{S}}$ for functions in $\mathbb{V}_\tau$ coincides with the norm $\|\cdot\|_{\mathbf{S}}$ for $\mathbb{V}^N$, i.e. we have $\|v\|_{\mathbb{S}} = \|\mathbf{v}\|_{\mathbf{S}}$ for all $v \in \mathbb{V}_\tau$, where $\mathbf{v} \in \mathbb{V}^N$ is the vector representation of v.

Remark 6.1 (High-order time-stepping methods). Theorem 6.1 extends straightforwardly to the family of discontinuous Galerkin time-stepping methods, provided that $\mathcal{I}$ is appropriately defined, as shown in [39].

We now prove the max-norm bound mentioned previously in Remark 3.2.

Corollary 6.2 (Max-norm bound). *For any $u \in \mathbb{V}_\tau$, with vector representation $\mathbf{u} \in \mathbb{V}^N$, we have*

$$\max_{1 \leq n \leq N} \|u_n\|_M \leq \frac{1}{\sqrt{2}} \|u\|_{\mathbb{S}} = \frac{1}{\sqrt{2}} \|\mathbf{u}\|_{\mathbf{S}}. \tag{6.14}$$

Proof. Let $1 \leq n \leq N$ be arbitrary. Define the test function $v \in \mathbb{V}_\tau$ by $v_m = u_m$ for all $m \leq n$, and $v_m = 0$ for all $m > n$. Then, it is straightforward to show that $b(u,v) = \frac{1}{2}\|u_n\|_M^2 + \frac{1}{2}\sum_{m=1}^n \|\langle\!| u |\!\rangle_{m-1}\|_M^2 + \|v\|_{\mathbb{A}}^2$. We then use the inf-sup identity (6.10) and Young's inequality to find that $|b(u,v)| \leq \frac{1}{4}\|u\|_{\mathbb{S}}^2 + \|v\|_{\mathbb{A}}^2$, which yields the bound $\|u_n\|_M \leq \frac{1}{\sqrt{2}}\|u\|_{\mathbb{S}}$; this completes the proof since n was arbitrary. $\qquad\square$

7 Spectral equivalence of the Schur complement preconditioner

We now analyse the spectral equivalence between the preconditioner $\mathbf{H}$, defined in (4.5), and the Schur complement matrix $\mathbf{S}$, leading to the proof of Theorem 4.1. We will use below the auxiliary bilinear form $s_D \colon \mathbb{V}_\tau \times \mathbb{V}_\tau \to \mathbb{R}$ defined by

$$s_D(u,v) \coloneqq \sum_{n=1}^N \left\{ \int_{I_n} (\partial_t \mathcal{I}u, \partial_t \mathcal{I}v)_{MA_n^{-1}M} \, \mathrm{d}t + \beta_n \int_{I_n} (u,v)_{A_n} \, \mathrm{d}t \right\}, \tag{7.1}$$

where the weights $\{\beta_n\}_{n=1}^N$ are defined in (4.3). Thus it is seen that $s_D(\cdot,\cdot)$ is closely related to $s(\cdot,\cdot)$, where the jump bilinear form $j(\cdot,\cdot)$ has been removed, and a weight is included in one of the terms. This weight is needed later for the analysis of the DST below, see Remark 7.1. Next, we show that $s_D(\cdot,\cdot)$ and $s(\cdot,\cdot)$ are spectrally equivalent with fully robust constants; thus $s_D(\cdot,\cdot)$ represents the dominant terms in $s(\cdot,\cdot)$.

Lemma 7.1 (Spectral equivalence). *Let s_D be defined in (7.1). Then, we have*

$$s_D(v,v) \leq s(v,v) \leq 3\, s_D(v,v) \quad \forall\, v \in \mathbb{V}_\tau. \tag{7.2}$$

Proof. The lower bound $s_D(v,v) \leq s(v,v)$ for all $v \in \mathbb{V}_\tau$ follows directly from the fact that $j(v,v) \geq 0$ as shown by (6.9). Now, it follows from the Cauchy–Schwarz inequality that $j(v,v) \leq$

$2 \sum_{n=1}^{N} \int_{I_n} \|\partial_t \mathcal{I}v\|_{MA_n^{-1}M} \|v\|_{A_n} \mathrm{d}t$ as a result of the definition of $\|\cdot\|_{MA_n^{-1}M}$ in (6.13). Then, Young's inequality implies that

$$s(v,v) \leq \sum_{n=1}^{N} \left\{ \left(1 + \frac{1}{\beta_n}\right) \int_{I_n} \|\partial_t \mathcal{I}v\|_{MA_n^{-1}M}^2 \, \mathrm{d}t + (1 + \beta_n) \int_{I_n} \|v\|_{A_n}^2 \, \mathrm{d}t \right\}$$

$$= \sum_{n=1}^{N} \left(1 + \frac{1}{\beta_n}\right) \left\{ \int_{I_n} \|\partial_t \mathcal{I}v\|_{MA_n^{-1}M}^2 \, \mathrm{d}t + \beta_n \int_{I_n} \|v\|_{A_n}^2 \, \mathrm{d}t \right\},$$

which implies $s(v,v) \leq 3\, s_D(v,v)$ for all $v \in \mathbb{V}_\tau$, since $1 + \beta_n^{-1} \leq 3$ for all $n \leq N$. $\qquad\square$

7.1 Discrete Sine Transform

Let the piecewise constant real-valued functions $\{\varphi_k\}_{k=1}^{N} \subset \oplus_{n=1}^{N} \mathcal{P}_0(I_n; \mathbb{R})$ be defined by

$$\varphi_k|_{I_n} = \varphi_k^n := \sin\left(\frac{(2k-1)n\pi}{2N}\right), \qquad n = 1, \ldots, N. \tag{7.3}$$

It is important to note that the functions φ_k are globally supported in time. Moreover, we also define $\varphi_k^0 := 0$ for all $1 \leq k \leq N$. These functions are linearly independent and form a basis of $\oplus_{n=1}^{N} \mathcal{P}_0(I_n; \mathbb{R})$, so any function $v \in \mathbb{V}_\tau$ can be written in the form $v = \sum_{k=1}^{N} \hat{v}_k \varphi_k$ with coefficients $\hat{v}_k \in \mathbb{V}$ for each $k = 1, \ldots, N$. The following result shows several basic properties of the basis induced by $\{\varphi_k\}_{k=1}^{N}$.

Lemma 7.2 (Discrete orthogonality). *The functions $\{\varphi_k\}_{k=1}^{N}$ defined by (7.3) form an orthogonal basis of $\oplus_{n=1}^{N} \mathcal{P}_0(I_n; \mathbb{R})$ in the following discrete inner-products: for any $1 \leq k, j \leq N$, we have*

$$\sum_{n=1}^{N} \beta_n \varphi_k^n \varphi_j^n = \frac{N}{2}\delta_{kj}, \quad \sum_{n=1}^{N} (\varphi_k^n - \varphi_k^{n-1})(\varphi_j^n - \varphi_j^{n-1}) = \frac{N}{2}\mu_k^2 \delta_{kj}, \tag{7.4}$$

where δ_{kj} is the Kronecker delta, where the $\{\beta_n\}_{n=1}^{N}$ are defined in (4.3), and where the positive real numbers μ_k are defined in (4.6). For any function $v \in \mathbb{V}_\tau$, we have the change of basis formulas given in (4.2) and (4.4).

We will make use of the following result from [35, Theorem 4].

Lemma 7.3 (Pearson & Wathen). *Let M and A be arbitrary symmetric positive definite matrices and let $\lambda \geq 0$ be an arbitrary nonnegative real number. Then, we have*

$$\frac{1}{2} \leq \frac{v^\top (MA^{-1}M + \lambda A)\, v}{v^\top (M + \sqrt{\lambda}A)A^{-1}(M + \sqrt{\lambda}A)\, v} \leq 1 \quad \forall\, v \in \mathbb{V} \setminus \{0\}.$$

7.2 Proof of Theorem 4.1

Let τ and let A be as in (2.2). As an intermediary step, we will use the bilinear form $s_\dagger(\cdot, \cdot) \approx s_D(\cdot, \cdot)$ defined on $\mathbb{V}_\tau$ by

$$s_\dagger(u,v) := \sum_{n=1}^{N} \left\{ \int_{I_n} \frac{\tau_n}{\tau}(\partial_t \mathcal{I}u, \partial_t \mathcal{I}v)_{MA^{-1}M} \, \mathrm{d}t + \beta_n \int_{I_n} \frac{\tau}{\tau_n}(u,v)_A \, \mathrm{d}t \right\}. \tag{7.5}$$

The bilinear forms $s_D(\cdot, \cdot)$ and $s_\dagger(\cdot, \cdot)$ are spectrally equivalent, since (2.2) and (6.6) imply that $\frac{1}{\alpha} s_\dagger(v,v) \leq s_D(v,v) \leq \alpha\, s_\dagger(v,v)$ for all $v \in \mathbb{V}_\tau$. Therefore, these inequalities and Lemma 7.1

17

imply that $\frac{1}{\alpha} s_\dagger(v, v) \leq s(v, v) \leq 3\,\alpha\, s_\dagger(v, v)$ for all $v \in \mathbb{V}_\tau$. Using Theorem 6.1, it is then seen that the matrix $\mathbf{S}_\dagger$ that represents $s_\dagger(\cdot, \cdot)$ in the standard basis of $\mathbb{V}_\tau$ satisfies

$$\frac{1}{\alpha}\,\mathbf{S}_\dagger \leq \mathbf{S} \leq 3\alpha\,\mathbf{S}_\dagger. \tag{7.6}$$

Next we establish the connection between $\mathbf{S}_\dagger$ and $\mathbf{H}$ as follows. Let u and $v \in \mathbb{V}_\tau$ be arbitrary; since v and $\partial_t \mathcal{I} v$ are piecewise constant on each time-step, and since $\partial_t \mathcal{I} v|_{I_n} = \frac{1}{\tau_n}(v_n - v_{n-1})$ for each n, we can simplify (7.5) to obtain

$$s_\dagger(u, v) = \sum_{n=1}^{N} \left\{ \frac{1}{\tau}(u_n - u_{n-1}, v_n - v_{n-1})_{MA^{-1}M} + \beta_n \tau (u_n, v_n)_A \right\}.$$

Therefore, Lemma 7.2 implies that the change of basis to $\{\varphi_k\}_{k=1}^{N}$ gives

$$s_\dagger(u, v) = \frac{N}{2} \sum_{k=1}^{N} \left\{ \frac{\mu_k^2}{\tau}(\hat{u}_k, \hat{v}_k)_{MA^{-1}M} + \tau(\hat{u}_k, \hat{v}_k)_A \right\} \quad \forall\, u,\, v \in \mathbb{V}_\tau, \tag{7.7}$$

where $\{\hat{u}_k\}_{k=1}^{N}$ and $\{\hat{v}_k\}_{k=1}^{N}$ denote the coefficients of the basis expansion of u and v with respect to $\{\varphi_k\}_{k=1}^{N}$, as in (4.4). In matrix notation, the identity (7.7) shows that

$$\mathbf{S}_\dagger = \mathbf{\Phi}^\top \hat{\mathbf{D}}\, \mathbf{\Phi}, \qquad \hat{\mathbf{D}} := \frac{N}{2} \operatorname{diag}\left\{ \frac{\mu_k^2}{\tau} MA^{-1}M + \tau A \right\}_{k=1}^{N}, \tag{7.8}$$

where we recall that $\mathbf{\Phi}$ is the matrix representation of the change of basis to $\{\varphi_k\}_{k=1}^{N}$. In other words, the change of basis to $\{\varphi_k\}_{k=1}^{N}$ block-diagonalizes $\mathbf{S}_\dagger$.

To complete the proof, recall that $\mathbf{H}$ is defined by $\mathbf{H} = \mathbf{\Phi}^\top \hat{\mathbf{H}} \mathbf{\Phi}$, where $\hat{\mathbf{H}}$ is defined in (4.5). We therefore have the change of basis identities $\mathbf{v}^\top \mathbf{S}_\dagger \mathbf{v} = \hat{\mathbf{v}}^\top \hat{\mathbf{D}} \hat{\mathbf{v}}$ and $\mathbf{v}^\top \mathbf{H} \mathbf{v} = \hat{\mathbf{v}}^\top \hat{\mathbf{H}} \hat{\mathbf{v}}$ for any $\mathbf{v} \in \mathbb{V}^N$. Hence, we can apply Lemma 7.3 block-by-block to the matrices $\hat{\mathbf{D}}$ and $\hat{\mathbf{H}}$, and we deduce that

$$\frac{1}{2}\mathbf{H} \leq \mathbf{S}_\dagger \leq \mathbf{H}. \tag{7.9}$$

Therefore, we obtain (4.9) from (7.6) and (7.9). Using the assumption (4.8), it is straightforward to show from the definition of $\mathbf{H}$ in (4.5) and $\widetilde{\mathbf{H}}$ in (4.7) that $\gamma \widetilde{\mathbf{H}} \leq \mathbf{H} \leq \Gamma \widetilde{\mathbf{H}}$. Using these inequalities, we obtain the spectral equivalence (4.10) of $\mathbf{S}$ and $\widetilde{\mathbf{H}}$ from the equivalence (4.9) between $\mathbf{S}$ and $\mathbf{H}$. $\qquad\square$

Remark 7.1 (Weight terms). It might appear desirable to avoid the weights β_n in the bilinear forms $s_D(\cdot, \cdot)$ and $s_\dagger(\cdot, \cdot)$, which would have the advantage of tightening the constant in the upper bound (4.9) from 3α to 2α. Moreover, the basis that block-diagonalizes the un-weighted version of $s_\dagger(\cdot, \cdot)$ is known explicitly through the functions $\{\psi_k\}_{k=1}^{N}$ where $\psi_k|_{I_n} := \sin\left(\frac{(2k-1)\,n\,\pi}{2N+1}\right)$ for each $n = 1, \ldots, N$. However, the appearance of the term $2N+1$ in the denominators seems to be inconvenient for the implementation of the fast DST, which is why it is not considered further.

8 Numerical experiments

We now study the efficiency, robustness and parallel scaling of the proposed method using a range of example problems in one, two, and three space dimensions.

8.1 Condition numbers of Schur complement preconditioner

First, we assess the sharpness of the bounds in Theorem 4.1, by computing numerically the extremal eigenvalues of the matrix $\mathbf{H}^{-1}\mathbf{S}$. For simplicity, we consider the one-dimensional heat equation on the spatial domain $\Omega = (0,1)$, with $T = 1$, discretized by $P1$ FEM in space on a uniform mesh, and by the implicit Euler method with a uniform time-step size $\tau = T/N$. In this case, the matrices M and $A_n = A$ represent respectively one dimensional mass and stiffness matrices. Note that the reason for choosing in this experiment some low dimensional spatial problems is to guarantee the high accuracy of the eigenvalue solver. For this problem, the assumption (2.2) holds with $\alpha = 1$. Therefore, the bound (4.9) shows that $1/2 \leq \lambda_{\min}$ and $\lambda_{\max} \leq 3$, where $\lambda_{\min}$ and $\lambda_{\max}$ denote respectively the minimal and maximal eigenvalues of $\mathbf{H}^{-1}\mathbf{S}$. This is in agreement with the results in Table 1, which suggest that the lower bound on the eigenvalues is indeed sharp, although the optimal upper bound appears rather to be $\lambda_{\max} \leq 2$, leading to condition numbers $\kappa(\mathbf{H}^{-1}\mathbf{S}) \leq 4$ in these experiments. Table 1 leads to some further predictions. For instance, the results for $\lambda_{\max}$ suggest that the damping parameter condition (3.7) simplifies here to $\omega < (1 - \rho_{\mathbf{A}})/(1 + \rho_{\mathbf{A}}) \leq 1$.

$h = 1/64$	$N = 4$	$N = 8$	$N = 16$	$N = 32$	$N = 64$	$N = 128$	$N = 256$	$N = 512$	$N = 1024$
$\lambda_{\min}$	0.8099	0.7080	0.6270	0.5728	0.5402	0.5223	0.5129	0.5081	0.5056
$\lambda_{\max}$	1.9999	1.9998	1.9996	1.9993	1.9986	1.9972	1.9944	1.9888	1.9780
$\kappa(\mathbf{H}^{-1}\mathbf{S})$	2.4693	2.8248	3.1893	3.4906	3.6994	3.8237	3.8885	3.9145	3.9122

$h = 1/128$	$N = 4$	$N = 8$	$N = 16$	$N = 32$	$N = 64$	$N = 128$	$N = 256$	$N = 512$	$N = 1024$
$\lambda_{\min}$	0.8099	0.7079	0.6270	0.5728	0.5402	0.5223	0.5129	0.5081	0.5056
$\lambda_{\max}$	2.0000	2.0000	1.9999	1.9998	1.9996	1.9993	1.9986	1.9972	1.9944
$\kappa(\mathbf{H}^{-1}\mathbf{S})$	2.4694	2.8250	3.1897	3.4916	3.7014	3.8278	3.8967	3.9310	3.9445

Table 1: Extremal eigenvalues and condition numbers of the matrix $\mathbf{H}^{-1}\mathbf{S}$ for the tests of Section 8.1.

8.2 Robustness with respect to time-steps, mesh-sizes, and approximate spatial solvers

We now study the robustness of the preconditioners with respect to variations in the time-steps, mesh-size, and also with respect to approximations in the spatial solvers that defined $\widetilde{\mathbf{A}}$ and $\widetilde{\mathbf{H}}$. In this experiment, we consider the two dimensional heat equation on the spatial domain $\Omega = (0,1)^2$, with $T = 1$, and initial condition $u(0) = \sin(\pi x)\sin(\pi y)$. The problem is discretized in space by $P1$ FEM on a uniform mesh of sizes $h = 2^{-k}$, $k = 3, \ldots, 6$, and in time by the implicit Euler method with uniform time-steps. Again, for this problem, we also have $\alpha = 1$. The preconditioners $\widetilde{\mathbf{A}}$ and $\widetilde{\mathbf{H}}$ employ a fixed number of geometric multigrid (MG) V-cycles for the spatial solvers; i.e. the matrices $\widetilde{A}_n^{-1}$ and $\widetilde{H}_k^{-1}$ are defined as the application of either one or two MG V-cycles with Jacobi smoothers. We then apply the inexact Uzawa method for these various choices of preconditioners, where we fix $\omega = 0.9$ for the damping parameter. In order to give a fair comparison of these different preconditioners, we compute at each iteration $\|\mathbf{u} - \mathbf{u}_j\|_{\mathbf{S}}$ for the exact error in the principal variable. Figure 1 shows the convergence histories of the inexact Uzawa method for these choices of spatial solvers, as well as for direct solvers. Here we fix $N = 512$ and $h = 1/64$. It is seen that the convergence rate does not depend significantly on the approximation of the spatial inverses.

Given the results in Figure 1, we now concentrate on the case of one V-cycle, and we now vary the mesh sizes and time-steps to verify the robustness with respect to the problem parameters.
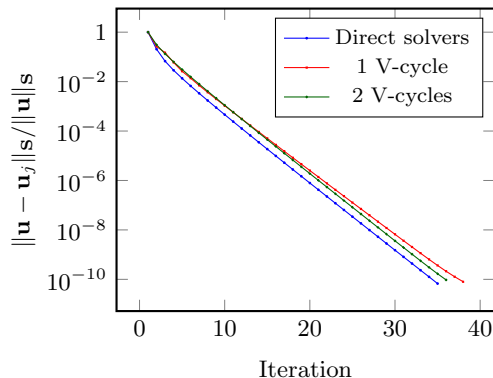
19

Figure 1: Convergence histories of the inexact Uzawa method for different choices of spatial solvers in Section 8.2: direct solvers, one V-cycle and two V-cycles.

	$h = 1/8$	$h = 1/16$	$h = 1/32$	$h = 1/64$
$N = 128$	20	21	21	21
$N = 256$	21	22	22	22
$N = 512$	22	22	22	22
$N = 1024$	22	22	22	22

Table 2: Number of inexact Uzawa iterations required to satisfy $\|\mathbf{u} - \mathbf{u}_j\|_{\mathbf{S}} < 10^{-6}\|\mathbf{u}\|_{\mathbf{S}}$ in the experiments of Section 8.2 (one MG V-cycle in space).

The results are given in Table 2, which presents the number of iterations required to achieve a relative error of 10^{-6} in the $\mathbf{S}$-norm of the error for the principal variable. It is thus seen that the resulting method is robust with respect to the mesh size and number of time-steps. It is also robust with respect to the ratio of mesh and time-step sizes, as expected.

8.3   Optimization of the damping parameter

We now consider the effect of the damping parameter on the contraction rate ω. With the same experiment as in Section 8.2, we compute the approximate contraction factors for a range of values of $\omega \in [1/2, 1]$. Here, we fix we now fix $N = 512$ and $h = 1/32$, and one multigrid V-cycle is used to for the spatial solvers. Since the norm $\|\cdot\|_{\mathcal{D}}$ depends on ω in (3.6), we compute here the approximate contraction factors as the maximum value of $\|\mathbf{u} - \mathbf{u}_{j+1}\|_{\mathbf{S}}/\|\mathbf{u} - \mathbf{u}_j\|_{\mathbf{S}}$ over a sufficient number of iterations required to reduce the error by a factor of 10^{-10}. The results are presented in Figure 2.

8.4   Time-parallel computations

We now present our main set of numerical experiments, involving weak and strong scaling studies with large-scale parallel computations. All the parallel computations in this work were performed on the Vulcan BlueGene/Q Supercomputer in Livermore, California. We start with weak and strong scaling tests of the time-parallelism. We solve the three dimensional heat equation on the spatial domain $\Omega = (0, 1)^3$, with $T = 0.1$. The right hand side and the initial condition are chosen such that the exact solution is $u(t, x, y, z) = e^{-3\pi^2 t} \sin(\pi x) \sin(\pi y) \sin(\pi z)$. The spatial
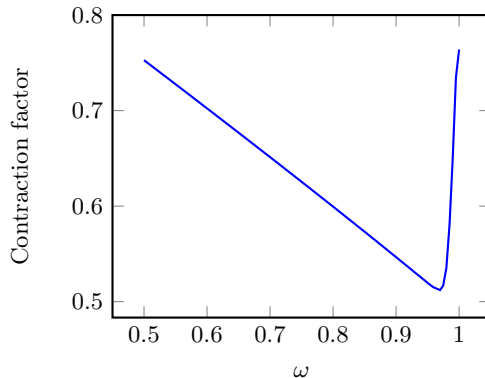
Figure 2: Approximate contraction factors for varying values of the damping parameter ω. The optimal value in this experiment is $\omega \approx 0.97$.

discretization uses lowest-order hexahedral finite elements on a fixed uniform mesh with 4096 elements. Uniform time-steps are used in time. For the weak scaling test, we assign 16 time steps to each core; each core then has a fixed problem size of 157 216 unknowns. For the strong scaling test we fix the number of time steps $N = 65\,536$, which results overall in a global linear system of dimension 643 956 736. For the spatial solvers, we use an algebraic multigrid method (AMG) provided by the library *hypre* [14], and the parallel DST (based on the FFT) is provided by the library FFTW3 [15]. We further use GMRES as an acceleration method for the inexact Uzawa method.

Tables 3 and 4 report the results of the weak and strong scaling tests, including the total number of iterations to reach a residual tolerance of 10^{-8}, the time spent per iteration and the total time of the computation. The last two columns in each table show the percentages of the total time that are spent inside calls to the parallel FFT and the spatial solvers. In the weak scaling test, we see that for small values of N, the number of iterations initially increases up to a maximum of 22, after which it remains constant; this is explained by the pre-asymptotic behaviour of the condition numbers reported in Table 1 above. Most importantly, the time per iteration remains essentially constant, so the small increase in total time is due to the iteration count. In this sense the test shows very good weak scaling of the method. Table 4 shows the good strong scaling of the method, where a doubling of the number of processors reduces the computational time by a factor often close to two. The last two columns of both Tables 3 and 4 show that the use of the FFT amounts to only a small part of the total computational time, whereas the AMG calls are dominant, with some additional time being spent outside of both FFT and AMG calls, for instance to handle some of the vector operations outside the preconditioners.

8.5 Time-parallel computations with variable coefficient

In this example we consider the same setup as in Section 8.4, but now we consider the equation $\partial_t u = \mu(t)\Delta u$ with a discontinuous time-dependent diffusion coefficient

$$\mu(t) := \begin{cases} 2 & t \in [\,2k\Delta t, (2k+1)\Delta t\,) \\ \frac{1}{2} & t \in [\,(2k+1)\Delta t, 2(k+1)\Delta t\,) \end{cases}, \quad k \in \mathbb{N}_0, \quad \Delta t = \frac{T}{1024}.$$

21

procs	N	dofs	iter	time/iter	total time	time FFT (%)	time AMG (%)
1	16	157 216	15	1.87	28.00	0.9%	84.5%
2	32	314 432	15	1.85	27.75	1.5%	83.4%
4	64	628 864	15	1.81	27.16	1.7%	82.8%
8	128	1 257 728	15	1.77	26.60	1.9%	82.4%
16	256	2 515 456	15	1.78	26.72	2.1%	82.1%
32	512	5 030 912	15	1.79	26.78	2.3%	82.0%
64	1 024	10 061 824	16	1.79	28.66	3.0%	81.3%
128	2 048	20 123 648	19	1.81	34.35	4.1%	79.8%
256	4 096	40 247 296	20	1.81	36.11	4.2%	79.5%
512	8 192	80 494 592	21	1.80	37.88	4.2%	79.3%
1 024	16 384	160 989 184	22	1.81	39.77	4.4%	79.0%
2 048	32 768	321 978 368	22	1.82	40.10	5.3%	78.3%
4 096	65 536	643 956 736	22	1.87	41.09	7.4%	76.4%

Table 3: Weak scaling test of Section 8.4. Computational times in seconds.

procs	N	dofs	iter	time/iter	total time	time FFT (%)	time AMG (%)
16	65 536	643 956 736	22	310.18	6823.88	3.9%	72.9%
32	65 536	643 956 736	22	155.68	3425.04	4.1%	72.9%
64	65 536	643 956 736	22	78.66	1730.53	4.8%	72.4%
128	65 536	643 956 736	22	39.98	879.52	5.5%	72.0%
256	65 536	643 956 736	22	20.89	459.60	7.1%	70.5%
512	65 536	643 956 736	22	10.76	236.82	7.3%	70.9%
1024	65 536	643 956 736	22	5.65	124.22	6.8%	72.3%
2048	65 536	643 956 736	22	3.13	68.79	7.0%	74.1%
4096	65 536	643 956 736	22	1.87	41.09	7.4%	76.4%

Table 4: Strong scaling test of Section 8.4. Computational times in seconds.

Thus the matrices $A_n = \mu(t_n)A$, $n = 1, \ldots, N$ are not constant anymore and the assumption (2.2) is satisfied with $\alpha = 2$. We repeat the weak scaling tests as in the previous subsection. This leads to similar results, with only a slight increase in the number of iterations due to the larger value of α in the spectral bound (4.10).

8.6 Space-time parallel computations

We now consider space-time parallelism, where additional cores are used to apply the spatial solvers in parallel over space. The spatial parallelism here is provided by the library *hypre*. We consider the same right hand side, initial condition and space-time domain as in the previous example. For the approximations in space we again use lowest order hexahedral elements, where we decompose the spatial domain $\Omega = (0,1)^3$ into 262 144 elements, and we use 4096 time steps for the time discretization. This results in a global linear system for 2 249 728 000 unknowns. With the same solver settings as in Section 8.4, we use varying numbers of processors in space and time, denoted respectively by p_x and p_t, thereby resulting in a total of $p_x p_t$ processors.

Table 6 presents the total computational times in seconds for these different processor configurations. We observe almost perfect scaling with respect to the time-parallelization, whereas the space-parallelization stagnates beyond 128 processors. The best result was obtained for $p_x = 64$ and $p_t = 2048$, i.e. with 131 072 processors overall, resulting in a time to solution of 15.96 seconds. Concerning the spatial parallelism, we used the default library settings in *hypre* without further tuning. A possible reason for the observed behaviour in terms of the spatial parallelism is that the spatial solvers adapt the smoothers inside AMG to the number of cores, with more effective

procs	N	dofs	iter	time/iter	total time	time FFT (%)	time AMG (%)
1	16	157 216	17	1.90	32.33	0.9%	84.7%
2	32	314 432	18	1.88	33.79	1.4%	83.5%
4	64	628 864	25	1.80	45.03	1.6%	82.1%
8	128	1 257 728	23	1.78	40.86	1.9%	81.8%
16	256	2 515 456	19	1.82	34.60	2.2%	81.7%
32	512	5 030 912	19	1.84	35.03	3.3%	80.8%
64	1 024	10 061 824	21	1.84	38.69	3.9%	80.0%
128	2 048	20 123 648	27	1.81	49.00	3.8%	79.4%
256	4 096	40 247 296	31	1.82	56.28	4.0%	78.6%
512	8 192	80 494 592	33	1.81	59.71	4.1%	78.4%
1 024	16 384	160 989 184	34	1.81	61.68	4.3%	78.2%
2 048	32 768	321 978 368	34	1.83	62.25	5.2%	77.5%
4 096	65 536	643 956 736	33	1.87	61.76	7.1%	76.0%

Table 5: Weak scaling test of Section 8.5. Computational times in seconds.

		procs w.r.t. space p_x					
		16	32	64	128	256	512
	4	12 158.70	7 000.47	4 381.72	2 925.62	2 132.41	2 107.73
	8	6 721.02	3 911.30	2 437.63	1 654.01	1 219.39	1 170.38
	16	4 016.91	3 522.05	1 459.71	1 007.60	728.52	703.79
procs w.r.t. time p_t	32	2 203.77	1 946.12	822.15	565.93	421.31	418.68
	64	1 212.84	9 04.27	429.03	304.47	238.31	245.17
	128	667.20	468.11	220.43	162.00	130.97	135.74
	256	341.14	232.08	117.75	85.76	70.97	74.36
	512	172.21	119.18	59.54	44.76	37.58	
	1 024	84.94	60.44	30.12	23.07		
	2 048	44.92	31.73	15.96			
	4 096	27.94	21.29				

Table 6: Space-time parallel test of Section 8.6. Computational times in seconds. The total number of processors is $p_t p_x$, with up to 131 072 processors in total. The linear system involves 2 249 728 000 unknowns.

sequential smoothers available on lower core counts. This leads to an additional reduction of the total number of iterations for small p_x.

Conclusion

We have presented an original method for the time-parallel solution of parabolic problems. The inf-sup stability of the discrete problem lead to the derivation of an original symmetric saddle-point reformulation of the problem that remains stable with respect to the natural norms of the problem. The saddle-point system can then be solved efficiently by iterative methods, such as the inexact Uzawa method considered here. We proposed an easily implementable non-intrusive time-parallel preconditioner for the Schur complement of the system, and proved robust spectral bounds with respect to key discretization parameters. The robustness, efficiency and parallel performance of the proposed method were shown both theoretically and experimentally in large scale parallel computations.

More broadly, the approach for preconditioning nonsymmetric systems as pursued in this

work, namely the construction of preconditioners based on the inf-sup stability of the problem, should not be strictly limited to the parabolic PDEs considered here, since inf-sup stability is equivalent to well-posedness for general linear operators. However, it is natural to expect that the practical details for developing efficient solvers will be specific to each problem and their discretizations.

References

[1] R. ANDREEV, *Wavelet-in-time multigrid-in-space preconditioning of parabolic evolution equations*, SIAM J. Sci. Comput., 38 (2016), pp. A216–A242.

[2] I. BABUŠKA, *Error-bounds for finite element method*, Numer. Math., 16 (1971), pp. 322–333.

[3] G. BAL, *On the convergence and the stability of the parareal algorithm to solve partial differential equations*, in Domain decomposition methods in science and engineering, vol. 40 of Lect. Notes Comput. Sci. Eng., Springer, Berlin, 2005, pp. 425–432.

[4] G. BAL AND Y. MADAY, *A "parareal" time discretization for non-linear PDE's with application to the pricing of an American put*, in Recent developments in domain decomposition methods (Zürich, 2001), vol. 23 of Lect. Notes Comput. Sci. Eng., Springer, Berlin, 2002, pp. 189–202.

[5] M. A. BOTCHEV AND H. A. VAN DER VORST, *A parallel nearly implicit time-stepping scheme*, J. Comput. Appl. Math., 137 (2001), pp. 229–243.

[6] J. H. BRAMBLE, J. E. PASCIAK, AND A. T. VASSILEV, *Analysis of the inexact Uzawa algorithm for saddle point problems*, SIAM J. Numer. Anal., 34 (1997), pp. 1072–1092.

[7] F. BREZZI, *On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers*, Rev. Française Automat. Informat. Recherche Opérationnelle Sér. Rouge, 8 (1974), pp. 129–151.

[8] A. J. CHRISTLIEB, R. D. HAYNES, AND B. W. ONG, *A parallel space-time algorithm*, SIAM J. Sci. Comput., 34 (2012), pp. C233–C248.

[9] M. EMMETT AND M. L. MINION, *Toward an efficient parallel in time method for partial differential equations*, Commun. Appl. Math. Comput. Sci., 7 (2012), pp. 105–132.

[10] A. ERN AND J.-L. GUERMOND, *Theory and practice of finite elements*, vol. 159 of Applied Mathematical Sciences, Springer-Verlag, New York, 2004.

[11] A. ERN, I. SMEARS, AND M. VOHRALÍK, *Guaranteed, locally space-time efficient, and polynomial-degree robust a posteriori error estimates for high-order discretizations of parabolic problems*, SIAM J. Numer. Anal., 55 (2017), pp. 2811–2834.

[12] A. ERN, I. SMEARS, AND M. VOHRALÍK, *Equilibrated flux a posteriori error estimates in $L^2(H^1)$-norms for high-order discretizations of parabolic problems*, IMA Journal of Numerical Analysis, (2018). Available online at http://dx.doi.org/10.1093/imanum/dry035.

[13] R. D. FALGOUT, S. FRIEDHOFF, T. V. KOLEV, S. P. MACLACHLAN, AND J. B. SCHRODER, *Parallel time integration with multigrid*, SIAM Journal on Scientific Computing, 36 (2014), pp. C635–C661.

[14] R. D. FALGOUT AND U. M. YANG, *hypre: A library of high performance preconditioners*, in Computational Science — ICCS 2002, P. M. A. Sloot, A. G. Hoekstra, C. J. K. Tan, and J. J. Dongarra, eds., Berlin, Heidelberg, 2002, Springer Berlin Heidelberg, pp. 632–641.

[15] M. FRIGO AND S. G. JOHNSON, *The design and implementation of FFTW3*, Proceedings of the IEEE, 93 (2005), pp. 216–231. Special issue on "Program Generation, Optimization, and Platform Adaptation".

[16] M. J. GANDER, *50 Years of Time Parallel Time Integration*, Springer International Publishing, Cham, 2015, pp. 69–113.

[17] M. J. GANDER AND S. GÜTTEL, *PARAEXP: a parallel integrator for linear initial-value problems*, SIAM J. Sci. Comput., 35 (2013), pp. C123–C142.

[18] M. J. GANDER AND M. NEUMÜLLER, *Analysis of a new space-time parallel multigrid algorithm for parabolic problems*, SIAM J. Sci. Comput., 38 (2016), pp. A2173–A2208.

[19] M. J. GANDER AND S. VANDEWALLE, *Analysis of the parareal time-parallel time-integration method*, SIAM J. Sci. Comput., 29 (2007), pp. 556–578.

[20] E. GILADI AND H. B. KELLER, *Space-time domain decomposition for parabolic problems*, Numer. Math., 93 (2002), pp. 279–313.

[21] A. GREENBAUM, *Iterative methods for solving linear systems*, vol. 17 of Frontiers in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.

[22] W. HACKBUSCH, *Parabolic multigrid methods*, in Computing methods in applied sciences and engineering, VI (Versailles, 1983), North-Holland, Amsterdam, 1984, pp. 189–197.

[23] T.-T.-P. HOANG, J. JAFFRÉ, C. JAPHET, M. KERN, AND J. E. ROBERTS, *Space-time domain decomposition methods for diffusion problems in mixed formulations*, SIAM J. Numer. Anal., 51 (2013), pp. 3532–3559.

[24] G. HORTON, *The time-parallel multigrid method*, Comm. Appl. Numer. Methods, 8 (1992), pp. 585–595.

[25] G. HORTON AND S. VANDEWALLE, *A space-time multigrid method for parabolic partial differential equations*, SIAM J. Sci. Comput., 16 (1995), pp. 848–864.

[26] G. HORTON, S. VANDEWALLE, AND P. WORLEY, *An algorithm with polylog parallel complexity for solving parabolic partial differential equations*, SIAM J. Sci. Comput., 16 (1995), pp. 531–541.

[27] J.-L. LIONS, Y. MADAY, AND G. TURINICI, *Résolution d'EDP par un schéma en temps "pararéel"*, C. R. Acad. Sci. Paris Sér. I Math., 332 (2001), pp. 661–668.

[28] J.-L. LIONS AND E. MAGENES, *Non-homogeneous boundary value problems and applications. Vol. I*, Springer-Verlag, New York-Heidelberg, 1972. Die Grundlehren der mathematischen Wissenschaften, Band 181.

[29] Y. MADAY AND G. TURINICI, *The parareal in time iterative solver: a further direction to parallel implementation*, in Domain decomposition methods in science and engineering, vol. 40 of Lect. Notes Comput. Sci. Eng., Springer, Berlin, 2005, pp. 441–448.

[30] J. MÁLEK AND Z. STRAKOŠ, *Preconditioning and the conjugate gradient method in the context of solving PDEs*, vol. 1 of SIAM Spotlights, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2015.

[31] E. MCDONALD, J. PESTANA, AND A. WATHEN, *Preconditioning and iterative solution of all-at-once systems for evolutionary partial differential equations*, SIAM J. Sci. Comput., 40 (2018), pp. A1012–A1033.

[32] J. NEČAS, *Sur une méthode pour résoudre les équations aux dérivées partielles du type elliptique, voisine de la variationnelle*, Ann. Scuola Norm. Sup. Pisa (3), 16 (1962), pp. 305–326.

[33] J. NIEVERGELT, *Parallel methods for integrating ordinary differential equations*, Comm. ACM, 7 (1964), pp. 731–733.

[34] C. C. PAIGE AND M. A. SAUNDERS, *Solutions of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.

[35] J. W. PEARSON AND A. J. WATHEN, *A new approximation of the Schur complement in preconditioners for PDE-constrained optimization*, Numer. Linear Algebra Appl., 19 (2012), pp. 816–829.

[36] R. B. PELZ, *Parallel FFTs*, in Parallel numerical algorithms (Hampton, VA, 1994), vol. 4 of ICASE/LaRC Interdiscip. Ser. Sci. Eng., Kluwer Acad. Publ., Dordrecht, 1997, pp. 245–266.

[37] J. PESTANA AND A. J. WATHEN, *Natural preconditioning and iterative methods for saddle point systems*, SIAM Rev., 57 (2015), pp. 71–91.

[38] C. SCHWAB, *p- and hp-finite element methods*, Numerical Mathematics and Scientific Computation, The Clarendon Press Oxford University Press, New York, 1998. Theory and applications in solid and fluid mechanics.

[39] I. SMEARS, *Robust and efficient preconditioners for the discontinuous Galerkin time-stepping method*, IMA J. Numer. Anal., 37 (2017), pp. 1961–1985.

[40] F. TANTARDINI AND A. VEESER, *The L^2-projection and quasi-optimality of Galerkin methods for parabolic equations*, SIAM J. Numer. Anal., 54 (2016), pp. 317–340.

[41] K. URBAN AND A. T. PATERA, *An improved error bound for reduced basis approximation of linear parabolic problems*, Math. Comp., 83 (2014), pp. 1599–1615.

[42] A. J. WATHEN, *Preconditioning*, Acta Numer., 24 (2015), pp. 329–376.

[43] D. E. WOMBLE, *A time-stepping algorithm for parallel computers*, SIAM J. Sci. Statist. Comput., 11 (1990), pp. 824–837.

[44] P. H. WORLEY, *Limits on parallelism in the numerical solution of linear partial differential equations*, SIAM J. Sci. Statist. Comput., 12 (1991), pp. 1–35.

[45] W. ZULEHNER, *Analysis of iterative methods for saddle point problems: a unified approach*, Math. Comp., 71 (2002), pp. 479–505.