

A 2-Level Domain Decomposition Preconditioner for KKT Systems with Heat-Equation Constraints

Eric C. Cyr

1 Introduction

This paper develops a new domain-decomposition method for solving the KKT system with heat-equation constraints. This effort is driven by the quadratic optimization problem of the form

$$\begin{aligned} \min_z \quad & \frac{1}{2} \int_0^T \|u - \tilde{u}\|_{L^2(\Omega)}^2 dt + \frac{\omega}{2} \int_0^T \|z\|_{L^2(\Omega)}^2 dt \\ \text{s.t.} \quad & \partial_t u - \nu \nabla \cdot \nabla u = z, & x \in \Omega \subset \mathbb{R}^2, t \in [0, T] \\ & u(x, t) = 0, \quad x \in \partial\Omega, t \in [0, T], & u(x, 0) = u_0(x), \quad x \in \Omega \end{aligned} \quad (1)$$

This quadratic PDE-constrained optimization problem finds a control z such that the solution u to the heat equation matches the target \tilde{u} . The spatial domain is Ω , the time interval is $[0, T]$, and the heat conductivity is ν . Uniform homogenous boundary conditions are assumed for all time, and the initial condition is prescribed by u_0 .

Many nonlinear methods use a series of quadratic approximations of the form represented by Eq. 1 to solve PDE-constrained optimization problems (see for instance sequential quadratic programming methods [8, 15, 27]). There have been several studies focused on developing scalable preconditioners for the saddle-point system that arises from the first-order necessary conditions. Often preconditioners for saddle-point systems take the form of approximate factorization block preconditioners [3]. These were explored for KKT systems in [4, 5]. Our work relies heavily on the block preconditioners from the Wathen group [23, 24, 25].

This effort focuses on transient PDE constraints where the size of the system scales with the number of spatial unknowns times the number of time steps, resulting in substantial computational effort. To alleviate this, a number of efforts have proposed accelerating the time solve using adaptive space-time discretiza-

Eric C. Cyr
Sandia National Laboratories, Albuquerque, NM, 87123, e-mail: eccyr@sandia.gov

tions [16, 17], parareal [12, 20, 27], multigrid approaches [6, 7, 10, 13, 19], block preconditioning [23, 24], and domain decomposition methods [14].

Our approach is also built on block preconditioning ideas. A difference is that our technique exploits an observation that the Schur-complement of the KKT system is elliptic in time (see [11, 18]). This allows us to leverage existing two level domain decomposition approaches for elliptic systems to improve the parallel scalability of the block preconditioner. Good performance is achieved by algorithmic choices that ensure the forward and backward in time integrators can be applied on the fine level.

2 Discrete system and block preconditioner

In this article the PDE in Eq. 1 will be discretized on a 2D Cartesian grid using first order backward Euler in time, and a second order finite difference stencil in space. A row of the discrete space-time system for the heat equation satisfies:

$$u_{i,j}^{n+1} - u_{i,j}^n + \Delta t \nu \left(\frac{-u_{(i+1)j}^{n+1} + 2u_{ij}^{n+1} - u_{(i-1)j}^{n+1}}{\Delta x^2} + \frac{-u_{i(j+1)}^{n+1} + 2u_{ij}^{n+1} - u_{i(j-1)}^{n+1}}{\Delta y^2} \right) = \Delta t z_{ij}^{n+1}. \quad (2)$$

Here i, j are the interior space indices defined over $1 \dots n_x - 1$ and $1 \dots n_y - 1$. The exterior indices are eliminated using the homogenous boundary conditions. The superscript time index n runs from $0 \dots N_t$. Each n is referred to below as a *time-node*. The control variable z 's index matches the implicit index on u , therefore z^{n+1} is associated with the n th time interval. For a single time interval, Eq. 2 rewritten in matrix form is

$$J_{(n+1)(n+1)} u^{n+1} + J_{(n+1)n} u_n + L_{(n+1)(n+1)} z^{n+1} = 0, \quad (3)$$

and the global space-time system is

$$Ju + Lz = f. \quad (4)$$

The right hand side f includes contributions from the initial conditions. The matrix J is block lower triangular and the matrix L is block diagonal.

The linear system whose solution solves the quadratic optimization problem from Eq. 1 is the celebrated KKT system $K\mathbf{u} = \mathbf{f}$ where

$$K = \begin{bmatrix} M_u & J^T \\ \omega M_z & L^T \\ J & L \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} u \\ z \\ w \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} f_u \\ f_z \\ f \end{bmatrix}. \quad (5)$$

The final row is the discrete form of the PDE constraint, enforced by the Lagrange multiplier w . We will also refer to w as the adjoint solution. M_u and M_z are identity matrices scaled by $\Delta t \Delta x \Delta y$. The matrix K is a saddle point matrix, whose structure is frequently observed in numerical optimization. Many effective block preconditioners

have been developed for this class of matrix [2, 3, 4]. We focus on the block preconditioning approach developed by Wathen and collaborators for solving linearized PDE-constrained optimization problems [23, 24, 25].

We write a block LDU factorization of the matrix K

$$K = \begin{bmatrix} I & & \\ JM_u^{-1} & \omega^{-1}LM_z^{-1} & I \end{bmatrix} \begin{bmatrix} M_u & & \\ & \omega M_z & \\ & & -S \end{bmatrix} \begin{bmatrix} I & M_u^{-1}J^T \\ & I & \omega^{-1}M_z^{-1}L^T \\ & & I \end{bmatrix} \quad (6)$$

where the Schur-complement is $S = JM_u^{-1}J^T + \frac{1}{\omega}LM_z^{-1}L^T$. Following [23], K is preconditioned using the block diagonal operator

$$P = \begin{bmatrix} M_u & & \\ & \omega M_z & \\ & & \hat{S} \end{bmatrix}, \text{ where } \hat{S} = \hat{J}M_u^{-1}\hat{J}^T, \hat{J} = J + \omega^{-1/2}L. \quad (7)$$

This preconditioner leverages the result in [21], and approximately inverts the block diagonal in the LDU factorization. The matrix \hat{J} used in the approximate Schur complement \hat{S} is block lower triangular (similar to J), a fact that we will exploit below. The choice of \hat{S} integrates the state Jacobian and the effects of the regularization parameter. In [23, 24] and [26], this approximation is developed and shown to lead to robust performance with respect to ω .

3 Two-level domain decomposition Schur-complement

We propose a new domain decomposition approach for approximately inverting \hat{S} . This is motivated by the observation that the operator S is elliptic in time (see [18] and [11]). For simplicity, we show this discretely using only the term $JM_u^{-1}J^T$. Consider the ODE $\partial_t y = -y$ discretized over three time steps with forward Euler: $y^{n+1} - y^n + \Delta t y^n = 0$. With $M_u = I$, the Schur-complement \hat{S} is

$$\begin{bmatrix} 1 & & \\ -1 + \Delta t & 1 & \\ & -1 + \Delta t & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 + \Delta t & \\ & 1 & -1 + \Delta t \\ & & 1 \end{bmatrix} = \begin{bmatrix} 1 & & & \\ -(1 - \Delta t) & 2(1 - \Delta t) + \Delta t^2 & & \\ & -(1 - \Delta t) & 2(1 - \Delta t) + \Delta t^2 & \\ & & & 1 \end{bmatrix}. \quad (8)$$

Examining the second row it is clear the operator has a 1D Laplacian stencil in time, with a positive perturbation on the diagonal. To take advantage of this ellipticity, we will apply existing domain decomposition approaches to the \hat{S} operator. This *ellipticity principle* enables scalable performance of a preconditioned Krylov method. We also impose an *efficiency constraint* that the computational kernels in our preconditioner use the time integration method for the state and adjoint unknowns.

The *ellipticity principle* is realized by considering a restricted additive Schwarz (RAS) method with N_D subdomains (see [9]). Each subdomain contains the spatial unknowns associated with a subset of time steps. For instance, if there are $N_t = 9$ time steps, then for $N_D = 3$ the subdomains contain time-nodes $\{1, 2, 3\}$, $\{3, 4, 5, 6\}$, and $\{6, 7, 8, 9\}$ (the 0th time-node is the excluded initial condition). Notice that the time-nodes are overlapped but the time steps are not. With these subsets, boolean operators R_s are defined that restrict a space-time vector to the time-nodes in a subdomain, giving the RAS preconditioner

$$\hat{S}_{\text{RAS}}^{-1} = \sum_{s=1}^{N_D} R_s^T D_s \left(R_s \hat{J} M_u^{-1} \hat{J}^T R_s^T \right)^{-1} R_s \quad (9)$$

where D_s is the (boolean) partition of unity matrix (Defn. 1.11 of [9]). RAS is known to lead to effective preconditioners for elliptic problems and can be extended to a multi-level schemes. While the *ellipticity principle* is exploited in $\hat{S}_{\text{RAS}}^{-1}$, the explicit formation of the product $\hat{J} M_u^{-1} \hat{J}^T$ does not satisfy the *efficiency constraint*.

To satisfy the *efficiency constraint* note that the range of $\hat{J}^T R_s^T$ is nonzero on time-nodes in the s^{th} subdomain and one time-node earlier. For instance, if the subdomain contains nodes $\{3, 4, 5, 6\}$ then the range is nonzero on $\{2, 3, 4, 5, 6\}$. Let Q_s be a new extended restriction operator whose action produces a space-time vector for time-nodes that are nonzero in the range of $\hat{J}^T R_s^T$. Further, choose Q_s so that

$$Q_s = \begin{bmatrix} W_s \\ R_s \end{bmatrix} \text{ and } Q_s Q_s^T = I. \quad (10)$$

The operator W_s restricts the space-time vector to the time-nodes contained in the earlier time step relative to the s^{th} subdomain. Because Q_s is a restriction operator that represents the nonzero range of $\hat{J}^T R_s^T$, we have that $Q_s^T Q_s \hat{J}^T R_s^T = \hat{J}^T R_s^T$. Recalling that M_u is diagonal, we can rewrite the subdomain solve in $\hat{S}_{\text{RAS}}^{-1}$ as

$$R_s \hat{J} M_u^{-1} \hat{J}^T R_s^T = R_s \hat{J} Q_s^T (Q_s M_u^{-1} Q_s^T) Q_s \hat{J}^T R_s^T \quad (11)$$

Using the constraint in Eq. 10, we have the additional identities $R_s = R_s Q_s^T Q_s$ and $R_s^T = Q_s^T Q_s R_s^T$. This permits a final rewrite of the operator in Eq. 11

$$R_s \hat{J} M_u^{-1} \hat{J}^T R_s^T = R_s Q_s^T \hat{J}_s M_s^{-1} \hat{J}_s^T Q_s R_s^T \quad (12)$$

where $\hat{J}_s = Q_s \hat{J} Q_s^T$ and $M_s^{-1} = Q_s M_u^{-1} Q_s^T$. The inverse action of $\hat{J}_s M_s^{-1} \hat{J}_s^T$ is easily computed in a matrix free way on the time-nodes in the extended subdomain. Motivated by this equivalence, we define a new one-level preconditioner

$$\hat{S}_{\text{RASQ}}^{-1} = \sum_{s=1}^{N_D} R_s^T D_s \left(R_s Q_s^T \hat{J}_s^{-T} M_s \hat{J}_s^{-1} Q_s R_s^T \right) R_s. \quad (13)$$

The term in parentheses is different from the term inverted in Eq. 9. The difference is that the inverse computed in Eq. 9 is constrained to have a zero initial condition

outside of the subdomain. This revised operator satisfies our *efficiency constraint* as computing \hat{J}_s^{-1} and \hat{J}_s^{-T} is done using the time integration method.

To obtain scalability with respect to the number of subdomains a coarse grid correction is required. Again leveraging the elliptic nature of the Schur-complement we consider the Nicolaidis coarse space developed for solving the Poisson problem [22]. Following the presentation in [9], the Nicolades coarse space is defined as

$$Z = \begin{bmatrix} D_1 R_1 \Phi_1 & & \dots & \\ & D_2 R_2 \Phi_2 & \ddots & \\ \vdots & & \ddots & \\ & & \dots & D_{N_D} R_{N_D} \Phi_{N_D} \end{bmatrix}, \text{ where } \Phi_s = \begin{bmatrix} 1 & -1 \\ 1 & -1 + \frac{2}{N_s} \\ \vdots & \vdots \\ 1 & -1 + 2\frac{N_s}{N_s} \end{bmatrix} \quad (14)$$

The columns of Φ_s form a constant and linear basis over the N_s subdomain time-nodes. The coarse restriction $R_0 = Z^T$ is used in the definition of the coarse operator $\hat{S}_0 = R_0 \hat{J} M_u^{-1} \hat{J}^T R_0^T$. The coarse solve is applied in a multiplicative way

$$\hat{S}_{2\text{-level}}^{-1} = \hat{S}_{\text{RASQ}}^{-1} R_0^T \hat{S}_0^{-1} R_0. \quad (15)$$

Due to the structure of R_0 the coarse operator \hat{S}_0 can be constructed in parallel. This does represent a violation of the *efficiency constraint* to be addressed by future work.

4 Numerical experiments

To demonstrate this approach we discretize the quadratic optimization problem from Eq. 1 as described in Sec. 2. The 2D spatial domain is $\Omega = (0, 1) \times (0, 2) \subset \mathbb{R}^2$, and the time domain is $[0, 1]$. The initial conditions and target solutions are

$$u_0(x, y) = -xy(x-1)(y-2), \quad \tilde{u}(x, y, t) = \sin(2.0\pi t) \sin(2.0\pi x) \sin(2.0\pi y). \quad (16)$$

The regularization parameter ω varies over five orders of magnitude. Experiments were run with 9×9 , 17×17 , and 33×33 mesh points. Qualitatively, variation with number of spatial points was not a factor in the convergence. This is not surprising as the implicit operator in space is inverted with a direct solve. As a result, the computations below are all for the case of 17×17 mesh points. Recall that homogeneous boundary conditions are removed, giving 15×15 unknowns in each time step. The linear system $K\mathbf{u} = \mathbf{f}$ (Eq. 5) is solved using right preconditioned GMRES from PyAMG [1] iterated until a relative residual tolerance of 10^{-6} is achieved.

Figure 1 presents three weak-scaling studies ranging from 100 to 3200 time steps. The number of time steps per subdomain is fixed at 80 in the left plot, 20 in the center plot, and 5 in the right plot. For the case of 80 steps, the fewest number of time steps is 200 (the minimum number of time steps evenly divisible by 80 in the chosen sequence). The plots, show the number of iterations as a function of time

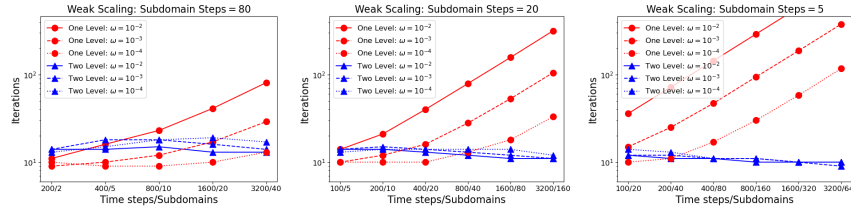


Fig. 1 Three weak scaling studies for different numbers of time-steps per subdomain. The two level scheme (triangles) has flat iteration counts for regardless of the number of time steps, the subdomain size, and the regularization parameter. Asymptotically the one level method shows a strong dependence with respect to the number of subdomains and time steps.

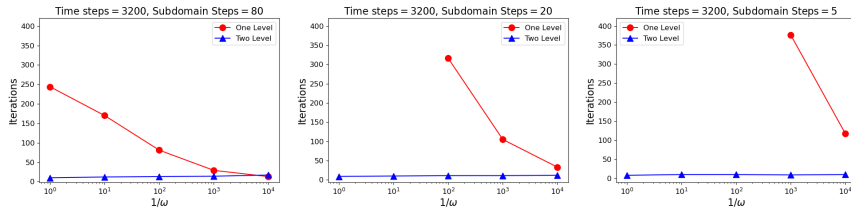


Fig. 2 This plot demonstrates the robustness of the two level scheme (triangle markers) with respect to the regularization parameter ω . Note that for many cases the one level scheme (circle markers) did not converge in the 420 iterations (the maximum allowed), thus those values are omitted.

step count for GMRES preconditioned using P from Eqn 7 with Schur complement approximations \hat{S}_{RASQ}^{-1} for the one level case (circle markers), and $\hat{S}_{2-level}^{-1}$ for the two level case (triangle markers). Different values for the regularization parameter ω are indicated using solid (10^{-2}), dashed (10^{-3}) or dotted (10^{-4}) lines. These plots demonstrate that the performance of the two level method is independent of both the number of subdomains, and the number of time steps. Further, independence holds regardless of the value of the regularization parameter. As anticipated the one level method has substantial growth with the number of time steps, and variability with the regularization parameter. However, it is worth noting that dependent on the number of subdomains and the size of the regularization parameter the one level method may be faster despite its lack of scalability. For instance, when using 40 subdomains and a regularization parameter of 10^{-4} the one-level method takes the same number of iterations but lacks the synchronization and added cost of the two level method.

The scaling with respect to the regularization parameter is investigated in Figure 2. In these plots the preconditioned iteration counts are plotted as a function of the inverse regularization parameter. Data points are excluded when the number of iterations exceeded the maximum iteration count for GMRES (in this case 420). Here again the two level method scales well, yielding essentially flat iteration counts as a function of the regularization parameter. The one level method shows strong dependence on ω , though it improves dramatically for smaller values.

5 Conclusion

In this paper, motivated by results in block preconditioning and the elliptic-in-time nature of the KKT system, we develop a two level domain decomposition preconditioner that facilitates a parallel-in-time solver for the discrete optimality system constrained by the heat equation. While limited in their breadth, initial results for this approach show excellent scalability with respect to the number of time steps, subdomains, and the regularization parameter. Future work will focus on achieving improved scaling by including more levels in the hierarchy, and applying this technique to a broader class of problems and discretizations.

Acknowledgements The author is indebted anonymous reviewers whose comments resulted in a significant strengthening of the paper. The author also acknowledges support from the SEA-CROGS project and the Early Career program, both funded by the DOE Office of Science. This article has been authored by an employee of National Technology & Engineering Solutions of Sandia, LLC under Contract No. DE-NA0003525 with the U.S. Department of Energy (DOE). The employee owns all right, title and interest in and to the article and is solely responsible for its contents. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this article or allow others to do so, for United States Government purposes. The DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan <https://www.energy.gov/downloads/doe-public-access-plan>. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

References

1. Bell, N., Olson, L. N., and Schroder, J. PyAMG: Algebraic multigrid solvers in python. *Journal of Open Source Software* **7**(72), 4142 (2022).
2. Benzi, M. and Golub, G. H. A preconditioner for generalized saddle point problems. *SIAM Journal on Matrix Analysis and Applications* **26**(1), 20–41 (2004).
3. Benzi, M., Golub, G. H., and Liesen, J. Numerical solution of saddle point problems. *Acta numerica* **14**, 1–137 (2005).
4. Benzi, M., Haber, E., and Taralli, L. A preconditioning technique for a class of PDE-constrained optimization problems. *Advances in Computational Mathematics* **35**(2), 149–173 (2011).
5. Biros, G. and Ghattas, O. Parallel Lagrange–Newton–Krylov–Schur methods for PDE-constrained optimization. Part I: The Krylov–Schur solver. *SIAM Journal on Scientific Computing* **27**(2), 687–713 (2005).
6. Borzi, A. Multigrid methods for parabolic distributed optimal control problems. *Journal of Computational and Applied Mathematics* **157**(2), 365–382 (2003).
7. Borzi, A. and Schulz, V. Multigrid methods for PDE optimization. *SIAM review* **51**(2), 361–395 (2009).
8. Byrd, R. H., Curtis, F. E., and Nocedal, J. An inexact SQP method for equality constrained optimization. *SIAM Journal on Optimization* **19**(1), 351–369 (2008).
9. Dolean, V., Jolivet, P., and Nataf, F. *An Introduction to Domain Decomposition Methods*. Society for Industrial and Applied Mathematics, Philadelphia, PA (2015). <https://epubs.siam.org/doi/pdf/10.1137/1.9781611974065>.

10. Falgout, R. D., Maldague, J.-M., Meyers, I., Munar, J., Neville, E., and Overman, T. TrimGRIT: An Extension of Multigrid Reduction in Time for Constrained Optimization. In: *PinT 2020 – 9th Parallel-in-Time Workshop* (2020).
11. Gander, M. J. and Kwok, F. Schwarz methods for the time-parallel solution of parabolic control problems. In: *Domain decomposition methods in science and engineering XXII*, 207–216. Springer (2016).
12. Gander, M. J., Kwok, F., and Salomon, J. PARAOPT: A parareal algorithm for optimality systems. *SIAM Journal on Scientific Computing* **42**(5), A2773–A2802 (2020).
13. Günther, S., Gauger, N. R., and Schroder, J. B. A non-intrusive parallel-in-time approach for simultaneous optimization with unsteady PDEs. *Optimization Methods and Software* **34**(6), 1306–1321 (2019).
14. Heinkenschloss, M. A time-domain decomposition iterative method for the solution of distributed linear quadratic optimal control problems. *Journal of Computational and Applied Mathematics* **173**(1), 169–198 (2005).
15. Heinkenschloss, M. and Ridzal, D. A matrix-free trust-region SQP method for equality constrained optimization. *SIAM Journal on Optimization* **24**(3), 1507–1541 (2014).
16. Langer, U., Steinbach, O., Tröltzsch, F., and Yang, H. Space-time finite element discretization of parabolic optimal control problems with energy regularization. *SIAM Journal on Numerical Analysis* **59**(2), 675–695 (2021).
17. Langer, U., Steinbach, O., Tröltzsch, F., and Yang, H. Unstructured space-time finite element methods for optimal control of parabolic equations. *SIAM Journal on Scientific Computing* **43**(2), A744–A771 (2021).
18. Lewis, R. M. and Nash, S. G. Model problems for the multigrid optimization of systems governed by differential equations. *SIAM Journal on Scientific Computing* **26**(6), 1811–1837 (2005).
19. Lin, S. *Multilevel-in-Time Methods for Optimal Control of PDEs and Training of Recurrent Neural Networks*. Ph.D. thesis, Rice University (2022).
20. Maday, Y. and Turinici, G. A parareal in time procedure for the control of partial differential equations. *Comptes Rendus Mathématique* **335**(4), 387–392 (2002).
21. Murphy, M. F., Golub, G. H., and Wathen, A. J. A note on preconditioning for indefinite linear systems. *SIAM Journal on Scientific Computing* **21**(6), 1969–1972 (2000).
22. Nicolaidis, R. A. Deflation of Conjugate Gradients with Applications to Boundary Value Problems. *SIAM Journal on Numerical Analysis* **24**(2), 355–365 (1987).
23. Pearson, J. W., Stoll, M., and Wathen, A. J. Regularization-robust preconditioners for time-dependent PDE-constrained optimization problems. *SIAM Journal on Matrix Analysis and Applications* **33**(4), 1126–1152 (2012).
24. Pearson, J. W., Stoll, M., and Wathen, A. J. Preconditioners for state-constrained optimal control problems with moreau–yosida penalty function. *Numerical Linear Algebra with Applications* **21**(1), 81–97 (2014).
25. Rees, T., Dollar, H. S., and Wathen, A. J. Optimal solvers for PDE-constrained optimization. *SIAM Journal on Scientific Computing* **32**(1), 271–298 (2010).
26. Schiela, A. and Ulbrich, S. Operator preconditioning for a class of inequality constrained optimal control problems. *SIAM Journal on Optimization* **24**(1), 435–466 (2014).
27. Ulbrich, S. Generalized SQP methods with “parareal” time-domain decomposition for time-dependent PDE-constrained optimization. In: *Real-time PDE-constrained optimization*, 145–168. SIAM (2007).