

A Parallel Solver for a Preconditioned Space-Time Boundary Element Method for the Heat Equation

Stefan Dohr, Michal Merta, Günther Of, Olaf Steinbach, and Jan Zapletal

1 Introduction

In this note we describe a parallel solver for the discretized weakly singular space-time boundary integral equation of the spatially two-dimensional heat equation. The global space-time nature of the system matrices leads to improved parallel scalability in distributed memory systems in contrast to time-stepping methods where the parallelization is usually limited to spatial dimensions. We present a parallelization technique which is based on a decomposition of the input mesh into submeshes and a distribution of the corresponding blocks of the system matrices among processors. To ensure load balancing, the distribution is based on a cyclic decomposition of complete graphs [8, 9]. In addition, the solution of the global linear system requires an efficient preconditioner. We present a robust preconditioning strategy which is based on boundary integral operators of opposite order [6, 14].

The parallelization of the discretized space-time integral equation in distributed and shared memory is discussed in [5]. Here, we extend the parallel solver to the preconditioned system. We demonstrate the method for the spatially two-dimensional case. However, the presented results, particularly the parallelization in distributed memory and the stability results for the preconditioner, can be used to extend the method to the three-dimensional problem.

Let $\Omega \subset \mathbb{R}^2$ be a bounded domain with a Lipschitz boundary $\Gamma := \partial\Omega$ and $T > 0$. As a model problem we consider the initial Dirichlet boundary value problem for the heat equation

Stefan Dohr, Olaf Steinbach, Günther Of
Institute of Applied Mathematics, Graz University of Technology,
Steyrergasse 30, 8010 Graz, Austria
e-mail: dohr@math.tugraz.at, o.steinbach@tugraz.at, of@tugraz.at

Michal Merta, Jan Zapletal
IT4Innovations and Department of Applied Mathematics, VSB – Technical University of Ostrava,
17. listopadu 2172/15, 708 00 Ostrava-Poruba, Czech Republic
e-mail: michal.merta@vsb.cz, jan.zapletal@vsb.cz

$$\alpha \partial_t u - \Delta_x u = 0 \text{ in } Q := \Omega \times (0, T), \quad u = g \text{ on } \Sigma := \Gamma \times (0, T), \quad u = u_0 \text{ in } \Omega \quad (1)$$

with a heat capacity constant $\alpha > 0$, the given initial datum $u_0 \in L^2(\Omega)$, and the boundary datum $g \in H^{1/2, 1/4}(\Sigma)$. An explicit formula for the solution of (1) is given by the representation formula for the heat equation [1], i.e. for $(x, t) \in Q$ we have

$$\begin{aligned} u(x, t) &= (\widetilde{M}_0 u_0)(x, t) + (\widetilde{V} w)(x, t) - (Wg)(x, t) \\ &= \int_{\Omega} U^*(x - y, t) u_0(y) \, dy + \frac{1}{\alpha} \int_{\Sigma} U^*(x - y, t - \tau) w(y, \tau) \, ds_y \, d\tau \\ &\quad - \frac{1}{\alpha} \int_{\Sigma} \frac{\partial}{\partial n_y} U^*(x - y, t - \tau) g(y, \tau) \, ds_y \, d\tau, \end{aligned} \quad (2)$$

with $w := \partial_n u$ and U^* denoting the fundamental solution of the two-dimensional heat equation given by

$$U^*(x - y, t - \tau) = \begin{cases} \frac{\alpha}{4\pi(t - \tau)} \exp\left(\frac{-\alpha|x - y|^2}{4(t - \tau)}\right) & \text{for } \tau < t, \\ 0 & \text{otherwise.} \end{cases}$$

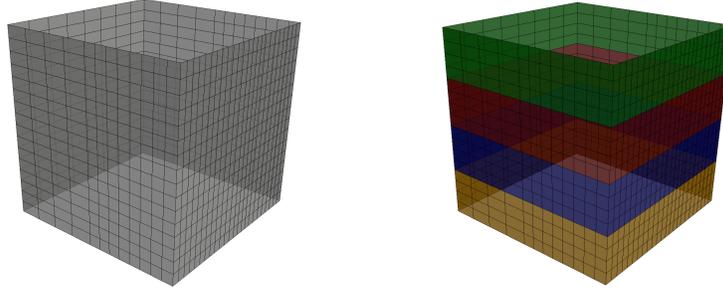
The yet unknown Neumann datum $w \in H^{-1/2, -1/4}(\Sigma)$ can be found by applying the interior Dirichlet trace operator to (2) and solving the resulting weakly singular boundary integral equation

$$g(x, t) = (M_0 u_0)(x, t) + (Vw)(x, t) + \left(\frac{1}{2}I - K\right)g(x, t) \quad \text{for } (x, t) \in \Sigma. \quad (3)$$

The operators in (3) are obtained by composition of the heat potentials in (2) with the Dirichlet trace operator. The ellipticity [2] and boundedness of the single-layer operator $V: H^{-1/2, -1/4}(\Sigma) \rightarrow H^{1/2, 1/4}(\Sigma)$ together with the boundedness of the double-layer operator $K: H^{1/2, 1/4}(\Sigma) \rightarrow H^{1/2, 1/4}(\Sigma)$ and the initial Dirichlet operator $M_0: L^2(\Omega) \rightarrow H^{1/2, 1/4}(\Sigma)$ ensure unique solvability of (3).

We consider a space-time tensor product decomposition of Σ [2, 10, 11] and use the Galerkin method for the discretization of (3). For a triangulation $\Gamma_h = \{\gamma_i\}_{i=1}^{N_\Gamma}$ of the boundary Γ and a decomposition $I_h = \{\tau_k\}_{k=1}^{N_I}$ of the time interval $I := (0, T)$ we define $\Sigma_h := \{\sigma = \gamma_i \times \tau_k : i = 1, \dots, N_\Gamma; k = 1, \dots, N_I\}$, i.e. $\Sigma_h = \{\sigma_\ell\}_{\ell=1}^N$ with $N = N_\Gamma N_I$. In the two-dimensional case the space-time boundary elements σ are rectangular. A sample decomposition of the space-time boundary of $Q = (0, 1)^3$ is shown in Fig. 1a.

We use the space $X_h^{0,0}(\Sigma_h) := \text{span}\{\varphi_\ell^0\}_{\ell=1}^N$ of piecewise constant basis functions and the space $X_h^{1,0}(\Sigma_h) := \text{span}\{\varphi_i^{10}\}_{i=1}^N$ of functions that are piecewise linear and globally continuous in space and piecewise constant in time for the approximations of the Cauchy data w and g , respectively. The initial datum u_0 is discretized by using the space of piecewise linear and globally continuous functions $S_h^1(\Omega_h) := \text{span}\{\varphi_j^1\}_{j=1}^{M_\Omega}$, which is defined with respect to a given triangulation $\Omega_h := \{\omega_i\}_{i=1}^{N_\Omega}$ of the domain Ω .



(a) Tensor product decomposition.

(b) Submeshes.

Fig. 1: Sample space-time boundary decompositions for $Q = (0, 1)^3$ [5].

This leads to the system of linear equations

$$V_h \mathbf{w} = \left(\frac{1}{2}M_h + K_h\right)\mathbf{g} - M_h^0 \mathbf{u}_0 \quad (4)$$

with

$$\begin{aligned} V_h[\ell, k] &:= \langle V\varphi_k^0, \varphi_\ell^0 \rangle_{L^2(\Sigma)}, & K_h[\ell, i] &:= \langle K\varphi_i^{10}, \varphi_\ell^0 \rangle_{L^2(\Sigma)}, \\ M_h^0[\ell, j] &:= \langle M_0\varphi_j^1, \varphi_\ell^0 \rangle_{L^2(\Sigma)}, & M_h[\ell, i] &:= \langle \varphi_i^{10}, \varphi_\ell^0 \rangle_{L^2(\Sigma)}, \end{aligned}$$

for $i, k, \ell = 1, \dots, N$ and $j = 1, \dots, M_\Omega$. Due to the ellipticity of the single-layer operator V the matrix V_h is positive definite and therefore (4) is uniquely solvable.

2 Operator Preconditioning

The boundary element discretization is done with respect to the whole space-time boundary Σ and since we want to solve (4) without an application of time-stepping schemes to make use of parallelization in time, we need to develop an efficient iterative solution technique. The linear system (4) with the positive definite but non-symmetric matrix V_h can be solved by a preconditioned GMRES method. Here we will apply a preconditioning technique based on boundary integral operators of opposite order [14], also known as operator or Calderon preconditioning [6].

First, we introduce the hypersingular operator D , which is defined as the negative Neumann trace of the double layer potential W in (2), i.e. $(Dv)(x, t) = -\partial_n(Wv)(x, t)$ for $(x, t) \in \Sigma$. The single-layer operator $V: H^{-1/2, -1/4}(\Sigma) \rightarrow H^{1/2, 1/4}(\Sigma)$ and the hypersingular operator $D: H^{1/2, 1/4}(\Sigma) \rightarrow H^{-1/2, -1/4}(\Sigma)$ are both elliptic [2] and the composition $DV: H^{-1/2, -1/4}(\Sigma) \rightarrow H^{-1/2, -1/4}(\Sigma)$ defines an operator of order zero. Thus, following [6], the Galerkin discretization of D allows the construction of a suitable preconditioner for V_h . While the discretization of the single-layer operator V is done with respect to $X_h^{0,0}(\Sigma_h)$, for the Galerkin discretization of the hypersingular

operator D we need to use a conforming trial space $Y_h = \text{span} \{\psi_i\}_{i=1}^N \subset H^{1/2,1/4}(\Sigma)$, see also [4] for the spatially one-dimensional problem.

Theorem 1 ([6, 14]) *Assume that the discrete stability condition*

$$\sup_{0 \neq v_h \in Y_h} \frac{\langle \tau_h, v_h \rangle_{L^2(\Sigma)}}{\|v_h\|_{H^{1/2,1/4}(\Sigma)}} \geq c_1^M \|\tau_h\|_{H^{-1/2,-1/4}(\Sigma)} \quad \text{for all } \tau_h \in X_h^{0,0}(\Sigma_h) \quad (5)$$

holds. Then there exists a constant $c_\kappa > 1$ such that $\kappa(M_h^{-1}D_hM_h^{-\top}V_h) \leq c_\kappa$ where, for $k, \ell = 1, \dots, N$,

$$D_h[\ell, k] = \langle D\psi_k, \psi_\ell \rangle_\Sigma, \quad M_h[\ell, k] = \langle \varphi_k^0, \psi_\ell \rangle_{L^2(\Sigma)}.$$

Thus we can use $C_V^{-1} = M_h^{-1}D_hM_h^{-\top}$ as a preconditioner for the matrix V_h . For the computation of the matrix D_h we use an alternative representation of the associated bilinear form which is attained by applying integration by parts, see [2, Theorem 6.1]. Note that the boundary element space Y_h is chosen to have the same dimension as $X_h^{0,0}(\Sigma_h)$ and thus, M_h is a square matrix. It remains to define a suitable boundary element space Y_h such that the mass matrix M_h is invertible and that the stability condition (5) is satisfied. In what follows we will discuss a possible choice.

We assume that the decompositions Γ_h and I_h are locally quasi-uniform. For the given boundary element mesh Γ_h we construct a dual mesh $\tilde{\Gamma}_h := \{\tilde{\gamma}_\ell\}_\ell^{N_\Gamma}$ according to [7, 13] and assume, that $\tilde{\Gamma}_h$ is locally quasi-uniform as well. For the discretization of the operator D we choose $Y_h = X_h^{1,0}(\tilde{\Sigma}_h) \subset H^{1/2,1/4}(\Sigma)$, which denotes the space of functions that are piecewise linear and globally continuous in space and piecewise constant in time, defined with respect to the decomposition $\tilde{\Gamma}_h$ and I_h , respectively. In order to prove the stability condition (5) we establish the $H^{1/2,1/4}(\Sigma)$ -stability of the $L^2(\Sigma)$ -projection $\tilde{Q}_h^{1,0}: L^2(\Sigma) \rightarrow Y_h \subset L^2(\Sigma)$ defined by

$$\langle \tilde{Q}_h^{1,0}v, \tau_h \rangle_{L^2(\Sigma)} = \langle v, \tau_h \rangle_{L^2(\Sigma)} \quad \text{for all } \tau_h \in X_h^{0,0}(\Sigma). \quad (6)$$

The Galerkin-Petrov variational problem (6) is uniquely solvable since the trial and test spaces satisfy a related stability condition [3]. When assuming appropriate local mesh conditions of Γ_h and $\tilde{\Gamma}_h$, see [12, 13], we are able to establish the stability of $\tilde{Q}_h^{1,0}: H^{1/2,1/4}(\Sigma) \rightarrow H^{1/2,1/4}(\Sigma)$, see [3] for a detailed discussion. Hence, there exists a constant $c_S > 0$ such that

$$\left\| \tilde{Q}_h^{1,0}v \right\|_{H^{1/2,1/4}(\Sigma)} \leq c_S \|v\|_{H^{1/2,1/4}(\Sigma)} \quad \text{for all } v \in H^{1/2,1/4}(\Sigma). \quad (7)$$

The stability estimate (7) immediately implies the stability condition (5). Hence the condition number $\kappa(C_V^{-1}V_h)$ with $C_V^{-1} = M_h^{-1}D_hM_h^{-\top}$ is bounded.

3 Distributed Memory Parallelization

Distributed memory parallelization of the solver is based on the scheme presented in [8, 9] for spatial problems. In [5] we have extended the approach to support time-dependent problems for the heat equation. Let us briefly describe the method and refer the more interested readers to the above-mentioned papers.

To distribute the system among P processes the space-time mesh Σ is decomposed into P slices in the temporal dimension (see Fig. 1b) which splits the matrices $A \in \{V_h, K_h, D_h\}$ into $P \times P$ blocks

$$A = \begin{bmatrix} A_{0,0} & 0 & \cdots & 0 \\ A_{1,0} & A_{1,1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A_{P-1,0} & A_{P-1,1} & \cdots & A_{P-1,P-1} \end{bmatrix}.$$

The matrices are block lower triangular with lower triangular blocks on the main diagonal due to the properties of the fundamental solution and the selected discrete spaces. We aim to distribute the blocks among processes such that the number of shared mesh parts is minimal and each process owns a single diagonal block. For this purpose we consider each block $A_{i,j}$ as an edge (i, j) of a complete graph K_P on P vertices. The distribution problem corresponds to finding a suitable decomposition of K_P into P subgraphs G_0, G_1, \dots, G_{P-1} . In [5, 8] we employ a cyclic decomposition algorithm – first, a generator graph G_0 on a minimal number of vertices (corresponding to blocks to be assembled by the process 0) is constructed; the remaining graphs G_1, \dots, G_{P-1} are obtained by a clock-wise rotation of G_0 along vertices of K_P placed on a circle. An example of the generating graph and the corresponding matrix decomposition for four processes is depicted in Fig. 2. In the case of the initial matrix M_h^0 we distribute block-rows of the matrix among processes. Similarly, since the matrix M_h is block diagonal, each process owns exactly one block of the matrix.

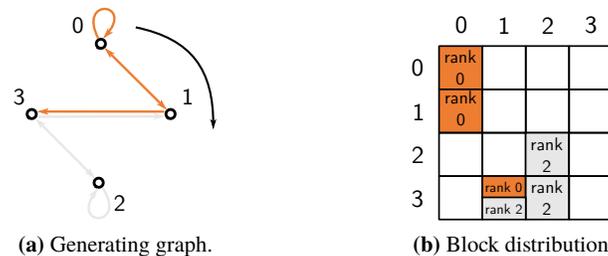


Fig. 2: Distribution of the system matrix blocks among four processes [5].

In addition to the distributed memory parallelization by MPI, the assembly of the matrices is parallelized and vectorized in shared memory using OpenMP [5, 15]. Therefore, in our numerical experiments we usually employ hybrid parallelization

using one MPI process per CPU socket and an appropriate number of OpenMP threads per process.

4 Numerical Experiments

The presented examples refer to the initial Dirichlet boundary value problem (1) in the space-time domain $Q := (0, 1)^3$. The heat capacity constant is set to $\alpha = 10$. We consider the exact solution

$$u(x, t) := \exp\left(-\frac{t}{\alpha}\right) \sin\left(x_1 \cos \frac{\pi}{8} + x_2 \sin \frac{\pi}{8}\right) \quad \text{for } (x, t) = (x_1, x_2, t) \in Q$$

and determine the Dirichlet datum g and the initial datum u_0 accordingly. The linear system (4) is solved by the GMRES method with a relative precision of 10^{-8} .

Operator Preconditioning

As a preconditioner we use the discretization $C_V^{-1} = M_h^{-1} D_h M_h^{-\top}$ of the hypersingular operator D in the space $X_h^{1,0}(\tilde{\Sigma}_h)$, while the Galerkin discretization of the integral equation (3) is done with respect to $X_h^{0,0}(\Sigma_h)$. Instead of using M_h in the preconditioner we computed a lumped mass matrix. Thus, the matrix becomes diagonal and the inverse can be applied efficiently.

The example corresponds to a globally uniform boundary element mesh with the mesh size $h = \mathcal{O}(2^{-L})$. Table 1 shows the iteration numbers of the non-preconditioned and preconditioned GMRES method. As expected, the iteration numbers of the preconditioned version are bounded due to the boundedness of $\kappa(C_V^{-1} V_h)$. For numerical results in the case of an adaptive refinement we refer to [3].

Table 1: Iteration numbers of the non-preconditioned GMRES method (It.) and the preconditioned GMRES method (It. prec.) in the case of uniform refinement. N denotes the number of boundary elements on level L .

L	N	It.	It. prec.
2	64	14	17
3	256	19	18
4	1 024	24	20
5	4 096	35	20
6	16 384	50	20
7	65 536	67	20
8	262 144	91	19
9	1 048 576	122	19

Table 2: Assembly of D_h for 65 536, 262 144, and 1 048 576 space-time elements.

nodes ↓ mesh →	D_h assembly [s]			D_h speedup			D_h efficiency [%]		
	65k	262k	1M	65k	262k	1M	65k	262k	1M
1	184.1	—	—	1.0	—	—	100.0	—	—
2	92.0	—	—	2.0	—	—	100.1	—	—
4	46.8	—	—	3.9	—	—	98.4	—	—
8	23.8	373.6	—	7.7	1.0	—	96.7	100.0	—
16	11.8	186.1	—	15.6	2.0	—	97.3	100.4	—
32	5.9	91.9	—	31.0	4.1	—	96.7	101.7	—
64	3.0	47.0	747.0	60.5	7.9	1.0	94.6	99.3	100.0
128	—	24.0	376.9	—	15.6	2.0	—	97.3	99.1
256	—	—	193.5	—	—	3.9	—	—	96.5

Scalability in Distributed Memory

The numerical experiments for the scalability were executed on the Salomon cluster at IT4Innovations National Supercomputing Center in Ostrava, Czech Republic. The cluster is equipped with 1008 nodes with two 12-core Intel Xeon E5-2680v3 Haswell processors and 128 GB of RAM. Nodes of the cluster are interconnected by the InfiniBand 7D enhanced hypercube network.

We tested the assembly of the BEM matrix D_h . Computation times for the assembly of the matrices V_h, K_h, M_h^0 , the related matrix-vector multiplication, and the evaluation of the solution in Q can be found in [5]. Strong scaling of the parallel solver was tested using a tensor product decomposition of Σ into 65 536, 262 144 and 1 048 576 space-time surface elements. We used up to 256 nodes (6 144 cores) of the Salomon cluster for the computations and executed two MPI processes per node. Each MPI process used 12 cores for the assembly of the matrix blocks.

In Table 2 the assembly times for D_h including the speedup and efficiency are listed. We obtain almost optimal parallel scalability. Note that the number of nodes is restricted by the number of elements of the temporal decomposition I_h . Conversely, for fine meshes we need a certain number of nodes to store the matrices.

5 Conclusion

In this note we have described a parallel space-time boundary element solver for the two-dimensional heat equation. The solver is parallelized using MPI in the distributed memory. The distribution of the system matrices is based on [5, 8, 9]. The space-time boundary mesh is decomposed into time slices which define blocks in the system matrices. These blocks are distributed among MPI processes using the graph decomposition based scheme. For a detailed discussion on shared memory

parallelization see [5]. Moreover, we have introduced an efficient preconditioning strategy for the space-time system which is based on the use of boundary integral operators of opposite order. The preconditioner was then distributed with the presented parallelization technique.

The numerical experiments for the proposed preconditioning strategy confirm the theoretical findings, i.e. the boundedness of the iteration number of the iterative solver. We also tested the efficiency of the parallelization scheme for the preconditioner. The results show almost optimal scalability.

Acknowledgements The research was supported by the project ‘Efficient parallel implementation of boundary element methods’ provided jointly by the Ministry of Education, Youth and Sports (7AMB17AT028) and OeAD (CZ 16/2017). SD acknowledges the support provided by the International Research Training Group 1754, funded by the German Research Foundation (DFG) and the Austrian Science Fund (FWF). JZ and MM further acknowledge the support provided by The Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPS II) project ‘IT4Innovations excellence in science – LQ1602’ and the Large Infrastructures for Research, Experimental Development and Innovations project ‘IT4Innovations National Supercomputing Center – LM2015070’.

References

1. Arnold, D.N., Noon, P.J.: Boundary integral equations of the first kind for the heat equation. In: Boundary elements IX, Vol. 3 (Stuttgart, 1987), pp. 213–229. Comput. Mech., Southampton (1987)
2. Costabel, M.: Boundary integral operators for the heat equation. *Integral Equations and Operator Theory* **13**, 498–552 (1990). DOI: [10.1007/BF01210400](https://doi.org/10.1007/BF01210400)
3. Dohr, S., Niino, K., Steinbach, O.: Preconditioned space–time boundary element methods for the heat equation. In preparation
4. Dohr, S., Steinbach, O.: Preconditioned space-time boundary element methods for the one-dimensional heat equation. In: Domain Decomposition Methods in Science and Engineering XXIV., *Lect. Notes Comput. Sci. Eng.*, vol. 125, pp. 243–251. Springer, Cham (2018). DOI: [10.1007/978-3-319-93873-8](https://doi.org/10.1007/978-3-319-93873-8)
5. Dohr, S., Zapletal, J., Of, G., Merta, M., Kravcenko, M.: A parallel space–time boundary element method for the heat equation. *Comput. Math. Appl.* **78**(9), 2852–2866 (2019). DOI: [10.1016/j.camwa.2018.12.031](https://doi.org/10.1016/j.camwa.2018.12.031)
6. Hiptmair, R.: Operator Preconditioning. *Comput. Math. Appl.* **52**, 699–706 (2006)
7. Jerez-Hanckes, C., Hiptmair, R., Urzua, C.: Mesh-Independent Operator Preconditioning for Boundary Elements on Open Curves. *SIAM J. Numer. Anal.* **52**, 2295–2314 (2014)
8. Kravcenko, M., Merta, M., Zapletal, J.: Distributed fast boundary element methods for Helmholtz problems. *Appl. Math. Comput.* **362**, 1–15 (2019). DOI: [10.1016/j.amc.2019.06.017](https://doi.org/10.1016/j.amc.2019.06.017)
9. Lukas, D., Kovar, P., Kovarova, T., Merta, M.: A parallel fast boundary element method using cyclic graph decompositions. *Numer. Algorithms* **70**(4), 807–824 (2015). DOI: [10.1007/s11075-015-9974-9](https://doi.org/10.1007/s11075-015-9974-9)
10. Messner, M., Schanz, M., Tausch, J.: A fast Galerkin method for parabolic space–time boundary integral equations. *J. Comput. Phys.* **258**, 15–30 (2014). DOI: [10.1016/j.jcp.2013.10.029](https://doi.org/10.1016/j.jcp.2013.10.029)
11. Noon, P.: The Single Layer Heat Potential and Galerkin Boundary Element Methods for the Heat Equation. Thesis, University of Maryland (1988)
12. Steinbach, O.: On the stability of the L_2 projection in fractional Sobolev spaces. *Numer. Math.* **88**, 367–379 (2001)

13. Steinbach, O.: On a generalized L_2 projection and some related stability estimates in Sobolev spaces. *Numer. Math.* **90**, 775–786 (2002)
14. Steinbach, O., Wendland, W.: The construction of some efficient preconditioners in the boundary element method. *Adv. Comput. Math.* **9**, 191–216 (1998)
15. Zapletal, J., Merta, M., Maly, L.: Boundary element quadrature schemes for multi- and many-core architectures. *Comput. Math. Appl.* **74**(1), 157–173 (2017). DOI: [10.1016/j.camwa.2017.01.018](https://doi.org/10.1016/j.camwa.2017.01.018)