Proceedings of the Fifteenth International Conference on
Computational Structures Technology
Edited by: P. Iványi, J. Kruis and B.H.V. Topping
Civil-Comp Conferences, Volume 9, Paper 13.1
Civil-Comp Press, Edinburgh, United Kingdom, 2024
ISSN: 2753-3239, doi: 10.4203/ccc.9.13.1

Performance Evaluation of Iterative Solvers for Vectorized Quasi-Static Heat Conduction in Peridynamics

S. Kim[1], S. Jin[1,2] and J.-W. Hong[1]

[1] Department of Civil and Environmental Engineering, KAIST Daejeon, Republic of Korea
[2] Division of Engineering and Applied Science, California Institute of Technology Pasadena, United States of America

Abstract

Heat conduction analyses on discontinuities via Peridynamics require a large amount of calculations. In this study, we propose a vectorization process to solve the peridynamic governing equations for quasi-static heat conduction analyses and suggest the optimal iterative solver. The heat equation is expressed by using peridynamic differential operators, and simplified for thermally isotropic simulations. The governing equation represented with compressed sparse row matrices consists of an off-diagonal sparse matrix and two diagonal matrices. Using vectorized operations, these matrices are further split into matrices related to the geometry. Four iterative solvers such as BiCG, BiCGSTAB, GMRES, and LGMRES are applied to solve the vectorized equation, and LGMRES demonstrates the best convergence times with the least number of calculation steps for several types of geometries. The temperature fields yielded by LGMRES are in good agreement with the results by the finite element analyses in the quasi-static thermal condition. This proposed vectorization procedure and the optimized iterative solver on Peridynamics will be useful to simulate the fully coupled thermomechanics.

Keywords: peridynamics, quasi-static heat conduction, vectorized computation, iterative solver, peridynamic differential operator, discretization.

1 Introduction

Quasi-static heat conduction is a thermodynamic process that is almost in equilibrium at every moment and is considered when the temperature changes very slowly or the heat capacity is very small. Since this phenomenon can occur with the destruction of the structures, quasi-static heat conduction analyses near the discontinuities are needed to be studied. For example, cracked in concrete structures can be developed by freezing and thawing due to daily temperature changes, and the chips that separate from the metals during the cutting process might quickly reach thermal equilibrium [1].

Peridynamics is a non-local method that constructs a continuum with particles instead of meshing [2, 3] and can be applied for heat conduction analyses on discontinuities. To calculate the physical quantity of one node, neighboring nodes within the surrounding area called the horizon are used. Since the partial derivative terms of the classical governing equations are converted to integral forms concerning neighboring nodes, physical quantities on discontinuities can be calculated [4]. However, due to integration over neighboring nodes, Peridynamics needs more computational resources than the finite element method (FEM) which only considers the nodes that belong to each element [5]. This cost increases more as the number of nodes or the neighboring nodes in each horizon increases. Therefore, a method to improve the computational performance of quasi-static thermal analyses using Peridynamics is required.

The most computationally intensive task in the entire process is solving the discretized governing equations. The amount of the computation can be mitigated depending on the solver employed for the matrix equations. While exact solutions for linear equations can be achieved by directly inverting the matrix, the computational complexity increases rapidly with the matrix size. In contrast, iterative solvers start with an initial guess and iteratively refine the answer until a satisfactory solution is obtained. Although the solution may diverge, this approach can significantly reduce computation time [6]. This solver can demonstrates better performance with large and sparse matrices commonly used in scientific computing such as FEM, Peridynamics, and others. Several iterative solvers have been developed to solve various linear equations. BiConjugate Gradient (BiCG) extends the Conjugate Gradient (CG) method [7] used in commercial programs such as Ansys as well as LS-DYNA to solve asymmetric problems, and BiCG STABilized (BiCGSTAB) improves the stability and robustness of BiCG [8]. In addition, Generalized Minimal RESidual (GMRES) is specialized for solving asymmetric problems [9]. To save the calculation memory of GMRES, Loose GMRES (LGMRES) was also developed [10].

The computational load of fundamental matrix operations, such as multiplication and summation, is also notable for large matrices. To alleviate the memory usage and calculation cost, several methods have been proposed in various kinds of languages, including Fortran, Python, and Julia. Data access and CPU cache utilization can be optimized by storing matrix elements in contiguous blocks of memory [11]. In addition, large and sparse matrices can be condensed by compressed sparse row (CSR) or compressed sparse column (CSC) representations that record only nonzero elements. Mathematical operations involving matrices allocated in the memory can be optimized using vectorized calculations. Programming libraries for scientific computing such as Linear Algebra PACKage (LAPACK) in Fortran or SciPy in Python are optimized for such vectorized operations. The computation efficiency can be significantly improved by avoiding the use of nested loops.

This study describes a method to vectorize the governing equations to improve the computational performance for quasi-static heat conduction analysis using Peridynamics. Additionally, for each of the four iterative solvers, we compare the solution time for each solver using three initial guesses. To compare the influence of the shape of the problem being analyzed, two types of examples are used, and the spacing between nodes is changed to four types.

2 Quasi-static heat conduction using Peridynamics

A partial differential governing equations for heat conduction in a continuum is called the heat conduction equation or heat equation. For a thermally isotropic material in three-dimensional space, ignoring heat generation inside the body, this equation can be expressed as

$$\rho c_v \dot{T}\left(\mathbf{x}, t\right) = k_T \frac{\partial^2 T\left(\mathbf{x}, t\right)}{\partial x_i^2}. \tag{1}$$

In the tensor notation, the integer i varies from 1 to 3. The notation $T\left(\mathbf{x}, t\right)$ is the temperature of the node $\mathbf{x}$ at time t, and the material properties ρ, c_v, and k_T are density, specific heat at constant volume, and thermal conductivity, respectively.

Peridynamic differential operators (PDDOs) express the partial derivatives of an arbitrary field value approximated by Taylor series expansions (TSEs) as a linear combination of neighboring node values and can be used to convert the heat conduction equation to a peridynamic integral equation [12]. Using PDDO derived by the second-order TSE in three dimensions, the first- or the second-order spatial derivative for the field value $\phi\left(\mathbf{x}\right)$ are written as

$$\frac{\partial^{p_1+p_2+p_3}\phi\left(\mathbf{x}\right)}{\partial x_1^{p_1}\partial x_2^{p_2}\partial x_3^{p_3}} = \int_{\mathcal{H}_{\mathbf{x}}} g_2^{p_1p_2p_3}\left(\boldsymbol{\xi}\right)\left(\phi\left(\mathbf{x}'\right) - \phi\left(\mathbf{x}\right)\right)\mathrm{d}V', \tag{2}$$

where $\mathbf{x}'$ is position of neighboring node in the horizon $\mathcal{H}_{\mathbf{x}}$ and $\boldsymbol{\xi} = \left(\xi_1, \xi_2, \xi_3\right) =$

$\mathbf{x}' - \mathbf{x}$. A polynomial $g_2^{p_1p_2p_3}$ is called peridynamic function and defined as

$$g_2^{p_1p_2p_3}(\boldsymbol{\xi}) = \sum_{(q_1q_2q_3)\in S_2} a_{q_1q_2q_3}^{p_1p_2p_3}(\mathbf{x}) \cdot w_{q_1q_2q_3}(\|\boldsymbol{\xi}\|) \cdot \xi_1^{q_1}\xi_2^{q_2}\xi_3^{q_3}, \tag{3}$$

where the set of integer pairs is given as

$$S_2 = \{(100), (010), (001), (200), (020), (002), (110), (101), (011)\}. \tag{4}$$

The coefficient of each term $a_{q_1q_2q_3}^{p_1p_2p_3}(\mathbf{x})$ depends on the node, with details of calculations provided by Madenci *et al.* [13]. $w_{q_1q_2q_3}(\|\boldsymbol{\xi}\|)$ is a weighting function and has been used regardless of the direction of differentiation [14–16].

The heat conduction equation in three-dimensional space for a thermally isotropic material is reformulated by PDDO as

$$\rho c_v \dot{T}(\mathbf{x}, t) = k_T \int_{\mathcal{H}_\mathbf{x}} g_2^{\mathrm{II}}(\boldsymbol{\xi}) \left(T(\mathbf{x}', t) - T(\mathbf{x}, t)\right) \mathrm{d}V', \tag{5}$$

where

$$g_2^{\mathrm{II}}(\boldsymbol{\xi}) = g_2^{200}(\boldsymbol{\xi}) + g_2^{020}(\boldsymbol{\xi}) + g_2^{002}(\boldsymbol{\xi}). \tag{6}$$

In a quasi-static process that reaches equilibrium very slowly at each time step, the rate of change of temperature is close to zero. By approximating the peridynamic integral equation into the summation for each node, Eq. 5 can be discretized as

$$\mathbf{0} = k_T \cdot \mathbf{G} \cdot \mathbf{T}, \tag{7}$$

where

$$[\mathbf{G}]_{ij} = \begin{cases} g_2^{\mathrm{II}}(\boldsymbol{\xi}_{ij}) \Delta V_j & \text{for } j : \mathbf{x}_j \in \mathcal{H}_{\mathbf{x}_i}, \\ -\sum_{k:\mathbf{x}_k\in\mathcal{H}_{\mathbf{x}_i}} g_2^{\mathrm{II}}(\boldsymbol{\xi}_{ik}) \Delta V_k & \text{for } i = j, \\ 0 & \text{else,} \end{cases} \tag{8}$$

and where $\boldsymbol{\xi}_{ij} = (\xi_{ij,1}, \xi_{ij,2}, \xi_{ij,3}) = \mathbf{x}_j - \mathbf{x}_i$, ΔV_i is the finite volume occupied by the node $\mathbf{x}_i$, and the length of the temperature field vector $\mathbf{T}$ for all nodes is equal to the total number of nodes. To impose Dirichlet boundary conditions, Eq. 7 can be divided into boundary and body parts as

$$\mathbf{0} = \begin{bmatrix} \mathbf{G_{DD}} & \mathbf{G_{DB}} \\ \mathbf{G_{BD}} & \mathbf{G_{BB}} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{T_D} \\ \mathbf{T_B} \end{bmatrix}, \tag{9}$$

where subscriptions $\mathbf{D}$ and $\mathbf{B}$ represent Dirichlet boundary and the body nodes, respectively. Thus, the temperature field of the body part can be obtained by solving the linear equation written as

$$\mathbf{G_{BB}} \cdot \mathbf{T_B} = -\mathbf{G_{BD}} \cdot \mathbf{T_D}. \tag{10}$$

3 Implementation of vectorization and iterative solvers

In this section, we explain the methodology of constructing the discretized governing equations using Peridynamics through vectorized computations as well as evaluating the performance of each iterative solver employed for solving the linear equations for heat conduction.

3.1 Vectorization of peridynamic equations

The matrix $\mathbf{G}$ in Eq. 7 is a function of the displacement vector from each node to its neighbors and the volume occupied by each node. To decompose this matrix into matrices about the shape of the object, we rewrite this matrix as

$$\mathbf{G} = \left(\mathbf{g}^{\mathrm{Diag}} + \mathbf{g}^{\mathrm{OD}}\right) \cdot \mathrm{diag}(\mathbf{V}), \tag{11}$$

where

$$\left[\mathbf{g}^{\mathrm{Diag}}\right]_{ij} = -\frac{1}{\Delta V_i}\left[\sum_{k:\mathbf{x}_k \in \mathcal{H}_{\mathbf{x}_i}} g_2^{\mathrm{II}}\left(\boldsymbol{\xi}_{ik}\right) \Delta V_k\right] \cdot \delta_{ij}, \tag{12}$$

$$\left[\mathbf{g}^{\mathrm{OD}}\right]_{ij} = \begin{cases} g_2^{\mathrm{II}}\left(\boldsymbol{\xi}_{ij}\right) & \text{for } j : \mathbf{x}_j \in \mathcal{H}_{\mathbf{x}_i}, \\ 0 & \text{else,} \end{cases} \tag{13}$$

and the element of the vector $\mathbf{V}$ is the volume of the node ΔV_i. The function $\mathrm{diag}(\boldsymbol{v})$ returns a diagonal matrix from the vector $\boldsymbol{v}$, and δ_{ij} is Kronecker delta. The off-diagonal matrix g^{OD} can be divided into matrices along each derivative direction $(q_1q_2q_3)$ as

$$\mathbf{g}^{\mathrm{OD}} = \sum_{(q_1q_2q_3)\in S_2} \mathbf{g}^{\mathrm{OD}}_{q_1q_2q_3}, \tag{14}$$

where

$$\left[\mathbf{g}^{\mathrm{OD}}_{q_1q_2q_3}\right]_{ij} = a^{\mathrm{II}}_{q_1q_2q_3}\left(\mathbf{x}_i\right) \cdot w_{q_1q_2q_3}\left(\|\boldsymbol{\xi}_{ij}\|\right) \cdot \left(\xi_{ij,1}\right)^{q_1}\left(\xi_{ij,2}\right)^{q_2}\left(\xi_{ij,3}\right)^{q_3}, \tag{15}$$

and

$$a^{\mathrm{II}}_{q_1q_2q_3}\left(\mathbf{x}_i\right) = a^{200}_{q_1q_2q_3}\left(\mathbf{x}_i\right) + a^{020}_{q_1q_2q_3}\left(\mathbf{x}_i\right) + a^{002}_{q_1q_2q_3}\left(\mathbf{x}_i\right). \tag{16}$$

Using Eq. 15, $\mathrm{g}^{\mathrm{OD}}_{q_1q_2q_3}$ can be vectorized as

$$\mathbf{g}^{\mathrm{OD}}_{q_1q_2q_3} = \mathrm{diag}\left(\mathbf{a}^{\mathrm{II}}_{q_1q_2q_3}\right) \cdot \left(\mathbf{W}_{q_1q_2q_3} : \boldsymbol{\Xi}_1^{(q_1)} : \boldsymbol{\Xi}_2^{(q_2)} : \boldsymbol{\Xi}_3^{(q_3)}\right), \tag{17}$$

where

$$\left[\mathbf{a}^{\mathrm{II}}_{q_1q_2q_3}\right]_i = a^{\mathrm{II}}_{q_1q_2q_3}\left(\mathbf{x}_i\right), \; \left[\mathbf{W}_{q_1q_2q_3}\right]_{ij} = w_{q_1q_2q_3}\left(\|\boldsymbol{\xi}_{ij}\|\right), \; \left[\boldsymbol{\Xi}_k^{(q_k)}\right]_{ij} = \left(\xi_{ij,k}\right)^{q_k}, \tag{18}$$

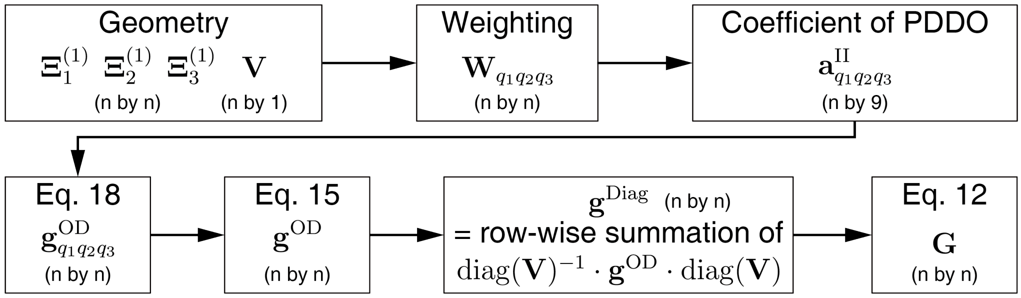

Figure 1: Procedures to construct peridynamic governing equations from matrices of the shape of an object using vectorized operations.

and the double dot product operator (:) in Eq. 17 performs element-wise multiplication between two matrices.

Since the weighting matrix $\mathbf{W}_{q_1q_2q_3}$ is related only to the distance to neighboring nodes, this matrix also can be easily constructed by vectorized operations. In this study, we adopt the following cubic spline function for all derivative directions $(q_1q_2q_3)$ as

$$w_{q_1q_2q_3}(\|\boldsymbol{\xi}\|) = \begin{cases} 2\left(\frac{\|\boldsymbol{\xi}\|}{\delta}\right)^3 - 3\left(\frac{\|\boldsymbol{\xi}\|}{\delta}\right)^2 + 1 & \text{for } 0 \le \|\boldsymbol{\xi}\| \le \delta, \\ 0 & \text{for } \delta < \|\boldsymbol{\xi}\|, \end{cases} \tag{19a}$$

where δ is the size of the horizon $\mathcal{H}_{\mathbf{x}}$.

The diagonal matrix $\mathbf{g}^{\mathrm{Diag}}$ can be obtained by adding all the elements of each row in the matrix $\mathrm{diag}(\mathbf{V})^{-1} \cdot \mathbf{g}^{\mathrm{OD}} \cdot \mathrm{diag}(\mathbf{V})$, and this row-wise summation can be easily calculated in the CSR matrix. Therefore, peridynamic linear equations for quasi-static heat conduction can be constructed using vectorized operations that are advantageous to the CSR matrix without accessing elements stored in the matrix or using nested loops. Figure 1 shows the process of constructing the discretized peridynamic equations from matrices that store the geometry of an object. The number in parentheses refers to the size of the matrix, and n is the total number of nodes.

3.2 Performance comparison of iterative solvers

For two heat conduction examples, the peridynamic heat conduction equation (Eq. 10) is solved using four iterative solvers to compare calculation performance. Figure 2 shows the shape and boundary condition of a plate with a hole and the L-shaped panel, respectively. The temperature of the node adjacent to the red side is set to 1 °C, and the blue part is 0 °C. The thicknesses of the plate and the panel are 25 mm and 100 mm, respectively.

To compare the performance depending on the size of the matrix, three distinct node spacings are chosen. The size of the horizon is fixed to four times the shortest

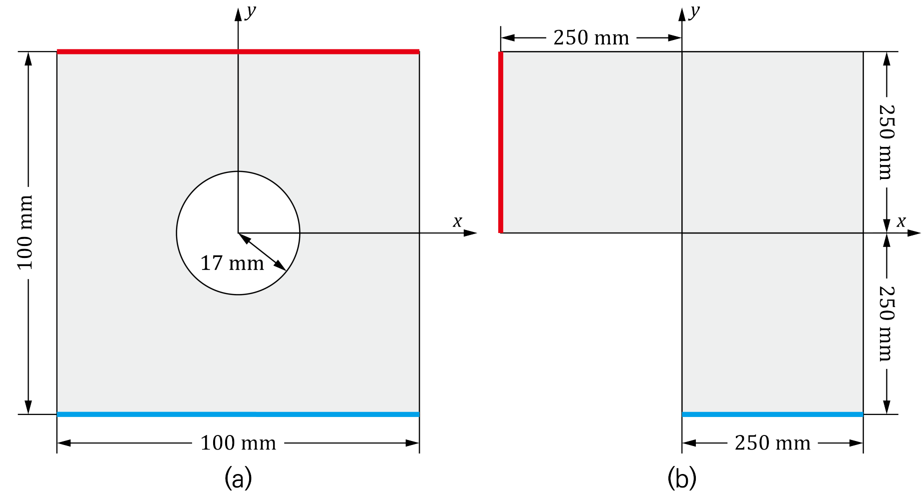

Figure 2: Two heat conduction examples: plate with a hole (a) and L-shaped panel model (b).

spacing of each case to maintain the number of neighboring nodes within the horizon. The solvers to be compared are BiCG, BiCGSTAB, GMRES, and LGMRES. A preconditioner to improve convergence speed is not used, and the initial guess of the solution for all nodes is set to 0.5°C which is the average value of the temperature boundary conditions. The version of Python is 3.11.3, and the analyses are performed in Intel Xeon Gold 6326 CPU with 256 GB RAM. The convergence condition of the solvers for the linear equation $\mathbf{A} \cdot \mathbf{x} = \mathbf{b}$ is set as

$$\|\mathbf{A} \cdot \mathbf{x} - \mathbf{b}\| \leq 10^{-5} \times \|\mathbf{b}\|, \tag{20}$$

where $\|\boldsymbol{v}\|$ is Euclidean norm of a vector $\boldsymbol{v}$. The solver with the shortest convergence time is determined as optimal, and the total number of time steps and the average time required for each step are then compared. In addition, the accuracy of the optimal solver is verified by comparing the temperature field obtained by Peridynamics with FEM results.

4 Results

In Table 1, n_B is the number of body node, t_c refers to the convergence time, and the highest values of performance indices are shown in bold for each case. Regardless of the objects or the sizes of the matrix, LGMRES demonstrates the shortest convergence time, up to 14.5 times faster than the subsequent shortest method, BiCGSTAB. The number of calculation steps is also the smallest, not exceeding six. The computation time for each step is the shortest for BiCG, but all except GMRES are similar. Figure 3 shows the temperature field of each example obtained by LGMRES, and these results are almost identical to the results of finite element analyses.

<table>
<tr><th>Example</th><th>n_B</th><th>Sparcity (%)</th><th>Solver</th><th>t_c (sec)</th><th>Steps</th><th>t_c/Steps</th></tr>
<tr><td rowspan="12">Plate
with a hole</td><td rowspan="4">4,560</td><td rowspan="4">98.18</td><td>BiCG</td><td>1.748</td><td>27</td><td>0.0647</td></tr>
<tr><td>BiCGSTAB</td><td>1.453</td><td>21</td><td>0.0692</td></tr>
<tr><td>GMRES</td><td>5.063</td><td>31</td><td>0.163</td></tr>
<tr><td>LGMRES</td><td>0.214</td><td>2</td><td>0.107</td></tr>
<tr><td rowspan="4">11,232</td><td rowspan="4">99.02</td><td>BiCG</td><td>16.479</td><td>33</td><td>0.499</td></tr>
<tr><td>BiCGSTAB</td><td>13.083</td><td>26</td><td>0.503</td></tr>
<tr><td>GMRES</td><td>34.562</td><td>39</td><td>0.886</td></tr>
<tr><td>LGMRES</td><td>1.499</td><td>3</td><td>0.500</td></tr>
<tr><td rowspan="4">23,800</td><td rowspan="4">99.56</td><td>BiCG</td><td>61.125</td><td>44</td><td>1.389</td></tr>
<tr><td>BiCGSTAB</td><td>47.390</td><td>34</td><td>1.394</td></tr>
<tr><td>GMRES</td><td>190.437</td><td>62</td><td>3.072</td></tr>
<tr><td>LGMRES</td><td>4.397</td><td>3</td><td>1.466</td></tr>
<tr><td rowspan="12">L-shaped
panel</td><td rowspan="4">9,280</td><td rowspan="4">99.70</td><td>BiCG</td><td>14.449</td><td>60</td><td>0.241</td></tr>
<tr><td>BiCGSTAB</td><td>11.483</td><td>47</td><td>0.244</td></tr>
<tr><td>GMRES</td><td>47.124</td><td>84</td><td>0.561</td></tr>
<tr><td>LGMRES</td><td>1.059</td><td>4</td><td>0.265</td></tr>
<tr><td rowspan="4">18,250</td><td rowspan="4">99.84</td><td>BiCG</td><td>62.804</td><td>74</td><td>0.849</td></tr>
<tr><td>BiCGSTAB</td><td>50.187</td><td>57</td><td>0.880</td></tr>
<tr><td>GMRES</td><td>254.976</td><td>130</td><td>1.961</td></tr>
<tr><td>LGMRES</td><td>3.609</td><td>4</td><td>0.902</td></tr>
<tr><td rowspan="4">47,600</td><td rowspan="4">99.94</td><td>BiCG</td><td>524.833</td><td>98</td><td>5.355</td></tr>
<tr><td>BiCGSTAB</td><td>391.037</td><td>71</td><td>5.508</td></tr>
<tr><td>GMRES</td><td>2397.561</td><td>201</td><td>11.928</td></tr>
<tr><td>LGMRES</td><td>26.991</td><td>5</td><td>5.398</td></tr>
</table>

Table 1: Comparison of iterative solver performance in peridynamic quasi-static heat conduction analyses across various geometries.

5 Conclusions

In this study, we present a process to vectorize the peridynamic governing equations for quasi-static heat conduction analyses. The heat equation discretized via peridynamic differential operators is used as the particle-based governing equation for isotropic thermal analyses. The matrix of a linear governing equation is expressed as a linear combination of one off-diagonal sparse matrix and two diagonal matrices. The off-diagonal matrix is only composed with the peridynamic functions of the differential operators. Using matrix multiplication and double dot product, the off-diagonal matrix is decomposed into a coefficient of the peridynamic function matrix, a weighting matrix, and matrices of relative displacement between each node and its neighbors. By combining these matrices with vectorized operations, the discretized governing equations consist only of the matrices regarding the geometry of the object.

To find the optimal iterative solver for large and sparse peridynamic equations,

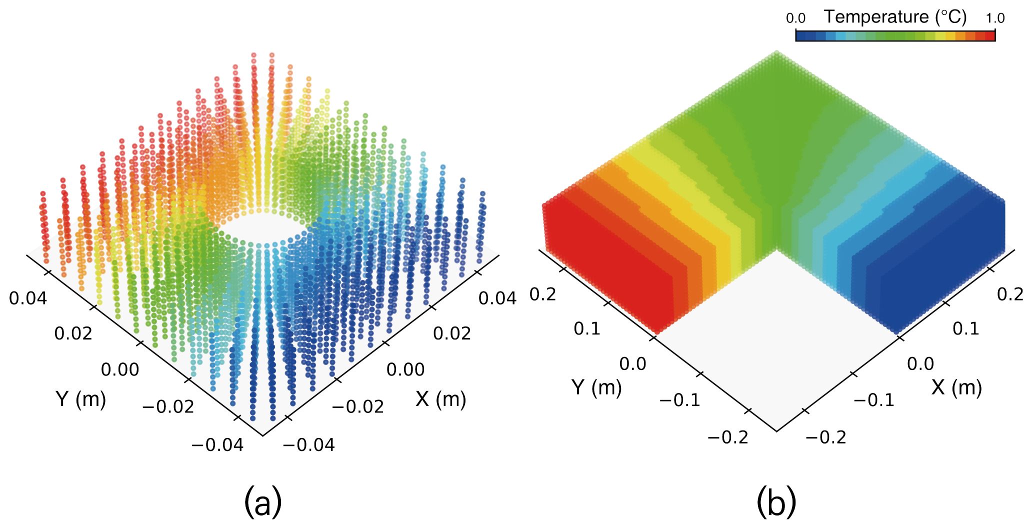

Figure 3: Temperature fields via peridynamic quasi-static heat conduction: plate with a hole (a) and L-shaped panel model (b).

we compare the convergence time, the number of steps, and the time required for each step of four methods: BiCG, BiCGSTAB, GMRES, and LGMRES. No preconditioner is applied and the average value of the boundary conditions is used as an initial guess of the solution. The temperature fields of two heat conduction examples are obtained, and the analyses are performed on the three different spacings. As a result, LGMRES solver shows the shortest convergence time and the smallest number of steps for all the geometries, and the time consumption of each step is similar to the times by other methods. The temperature fields obtained by this method are in good agreement with the results from the finite element analyses. Therefore, to achieve efficient solutions for quasi-static heat conduction problems, the LGMRES iterative solver proves to be an optimal choice. This vectorized framework for Peridynamics will be extended to fully coupled thermomechanics in future work.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2022R1A2C2091533).

References

[1] Q. Xia, "Quasi-static finite element modelling of thermal distribution and heat partitioning for the multi-component system of high speed metal cutting", Journal of Materials Processing Technology, 275, 116389, 2020.

[2] S.A. Silling, “A meshfree method based on the peridynamic model of solid mechanics”, Computers & structures, 83, 1526-1535, 2005.

[3] S. Jin, “Convergence study of stabilized non-ordinary state-based peridynamics for elastic and fracture problems”, Engineering Fracture Mechanics, 289, 109438, 2023.

[4] S. Jin, “Stabilized non-ordinary state-based peridynamics with irregular nodal distribution”, Mechanics Research Communications, 130, 104130, 2023.

[5] W. Liu, “A coupling approach of discretized peridynamics with finite element method”, Computer methods in applied mechanics and engineering, 245, 163-175, 2012.

[6] H.A. Van der Vorst, “Linear system solvers: sparse iterative methods”, Parallel numerical algorithms, Springer, 91-118, 1997.

[7] R. Fletcher, “Conjugate gradient methods for indefinite systems”, Numerical Analysis: Proceedings of the Dundee Conference on Numerical Analysis, 506, 73-89, 2006.

[8] H.A. Van der Vorst, “Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems”, SIAM Journal on scientific and Statistical Computing, 13(2), 631-644, 1992.

[9] C.C. Paige, “Solution of sparse indefinite systems of linear equations”, SIAM journal on numerical analysis, 12(4), 617-629, 1975.

[10] A.H. Baker, “A technique for accelerating the convergence of restarted GMRES”, SIAM Journal on Matrix Analysis and Applications, 26(4), 962-984, 2005.

[11] A. Silberschatz, “Operating system concepts”, Addison-Wesley Longman Publishing Co., Inc., 1991.

[12] E. Madenci, “Peridynamic differential operator and its applications”, Computer Methods in Applied Mechanics and Engineering, 304, 408-451, 2016.

[13] E. Madenci, “Peridynamic differential operator for numerical analysis”, Springer International Publishing, 2019.

[14] Z. Li, “Nonlocal steady-state thermoelastic analysis of functionally graded materials by using peridynamic differential operator”, Applied Mathematical Modelling, 93, 294-313, 2021.

[15] Y. Gao, “Multi-phase fluid flow simulation by using peridynamic differential operator”, Ocean Engineering, 216, 108081, 2020.

[16] V.R. Hosseini, “The peridynamic differential operator for solving time-fractional partial differential equations”, Nonlinear Dynamics, 109(3), 1823-1850, 2022.