

Yaroslav Sokolovskyy¹, Volodymyr Yarkun², Mariana Levkovych³¹ Computer Design Systems Department, Lviv Polytechnic National University, 12, S. Bandery str., Lviv, Ukraine, E-mail: yaroslav.i.sokolovskyy@lpnu.ua, ORCID 0000-0003-4866-2575² Software Engineering Department, Ukrainian National Forestry University, 103, Henerala Chuprynky str., Lviv, Ukraine, E-mail: yarkun@nltu.edu.ua, ORCID 0000-0002-4876-1111³ Computer Design Systems Department, Lviv Polytechnic National University, 12, S. Bandery str., Lviv, Ukraine, E-mail: mariana.v.levkovych@lpnu.ua, ORCID 0009-0004-6231-7799

PARALLEL ALGORITHM FOR SOLVING HEAT AND MASS TRANSFER IN FRACTAL MEDIA BASED ON THE SCHWARZ METHOD

Received: August 22, 2025 / Revised: September 12, 2025 / Accepted: September 15, 2025

© Sokolovskyy Y., Yarkun V., Levkovych M., 2025

<https://doi.org/10.23939/cds2025.02.114>

Abstract. Based on the Schwarz method, a parallel algorithm has been developed for partitioning a two-dimensional discretization domain into overlapping subdomains to solve a system of partial differential equations of fractional order that describes heat and mass transfer processes in anisotropic media with a fractal structure. Using the finite difference method, explicit difference schemes for temperature and moisture equations in subdomains are derived. Fractional derivatives in time and spatial coordinates are approximated using the Riemann – Liouville and Grunwald – Letnikov formulas, respectively. For parallel processing of subdomains, the formulation of local problems is organized, the formulation of local boundary and initial conditions within each subdomain is streamlined to ensure proper synchronization of the parallel algorithm, solution is updated in each subdomain, and information is transferred between subdomains to check the convergence of the method. Experimental results show that the proposed parallel algorithm has good scalability.

Keywords: parallel algorithm, domain decomposition, Schwarz method, media with fractal structure, heat and moisture transfer processes.

Introduction

This paper aims to develop a parallel algorithm based on the Schwarz method for the decomposition of the space-time domain of the heat and mass transfer problem described by time-dependent partial differential equations. To find a solution, high-resolution grids are often required, which implies a significant number of terms in the resulting system of equations, as well as the need to implement a parallel algorithm. Therefore, the development of an algorithm based on the Schwarz method for the decomposition of the space-time domain aims to reduce computational costs and accelerate the solution process by using modern multi-core and distributed computing systems.

Review of Modern Information Sources on the Subject of the Paper

High-performance computing architectures are becoming increasingly parallel, and the development of efficient and reliable algorithms that can fully exploit high degrees of parallelism remains a pressing issue in many scientific fields. In solving partial differential equations, significant progress has led to highly scalable domain decomposition methods [1], [2], [3] that exhibit high speedup and efficiency [4]. Domain decomposition methods are among the best approaches that provide the highest level of parallelism. Two grid levels of temporal discretization are mainly considered. The approximate solution of the PDE on a coarse grid provides initial values at each time subinterval, while more precise solutions are thus computed independently on a sufficiently fine time grid. In this sense, the parallel computational model can also be considered as an iterative predictor-corrector scheme [5].

Let us briefly recall some recent attempts to solve related spatiotemporal problems of fractal dimension. Fractal dimension indicates that the processes or objects under study have a complex, self-similar structure. In [2] a class of fully optimal methods based on spatiotemporal domain decomposition techniques was developed for certain boundary and distributed control problems in fluid flow and heat exchange. Existing research has demonstrated the reliability, parallel scalability, and efficiency of the proposed algorithms. Numerous results have been presented, obtained on multicore computers [5], [6], systems with several thousand processors [2], [3], and GPU parallelization using CUDA technologies [7]. In [8], [9] a class of corrected explicit-implicit GPU-based domain schemes is proposed to accelerate the solution of convection-dominated diffusion problems on GPUs. In [10], Schwarz algorithms with overlapping domains are considered for solving linear and nonlinear systems of equations arising from finite element discretization of elasticity problems on unstructured 3D meshes. It has been numerically demonstrated that Schwarz method-based algorithms with new coarse spaces exhibit high scalability on supercomputers, in terms of total computation time, for solving diffusion equations [11], [16] and elasticity problems discretized on large-scale meshes. These works explore many interesting aspects of space-time algorithms, but parallel computing is not their main concern.

The Schwarz method forms the foundation for many numerical algorithms and is widely used in modern mathematical computations [5], [7], [16]. It is an important tool for solving boundary value problems with fractal differential equations [10]. The fractal dimension indicates that the processes or objects under study have a complex, self-similar structure. In [17], a highly scalable domain decomposition solver is presented and studied for fully implicit solutions of internal flows in porous media with a fractional derivative. Additionally, a fast step-by-step approach is used for the time discretization of fractional derivatives, followed by the application of the parallel Newton-Krylov-Schwarz algorithm to solve the resulting discrete nonlinear system [18]. In [19] propose a parallel algorithm in three-dimensional space to speed up the computational speed. The developed algorithm achieves results 3.8 times faster while maintaining a localization error of only 3 %, proving its suitability for real-time decision-making. In [20] presents Schwarz methods as they were developed historically, illustrate how they can greatly enhance the performance of the solver, and show why one has to be cautious when testing them numerically. The paper [21] deals with the optimization of algorithms for solving complex fractional equations, with an emphasis on memory and parallel computing.

Objectives and Problems of Research

Parallelization of heat and mass transfer problems in fractal media allows to significantly reduce the computation time and increase the efficiency of these methods. Using the Schwarz method to divide the domain into overlapping subdomains ensures stability and accuracy of the solution. This allows to simultaneously process large computational grids on modern multi-core or cluster systems. This approach opens up opportunities for large-scale modeling of complex physical processes in practical engineering and scientific problems.

The object of this study is the heat and mass transfer processes in anisotropic media with a fractal structure, which are described by systems of fractional order differential equations.

The subject of the study is a parallel algorithm based on the Schwarz method, which implements the partition of a two-dimensional discretization domain into overlapping subdomains.

The main objective of the study is the development and analysis of a parallel algorithm based on the Schwarz method for heat and mass transfer problems in anisotropic fractal media. To achieve this objective, the following tasks are formulated:

- to develop a parallel computational algorithm based on the Schwarz domain decomposition method for solving fractional-order partial differential equations modeling heat and mass transfer in anisotropic media with fractal structure;
- to implement the partitioning of the two-dimensional discretization domain into overlapping subdomains to ensure efficient parallel processing;

– to apply finite difference methods to obtain explicit difference schemes for temperature and moisture transport equations using fractional derivatives based on the Riemann-Liouville and Grunwald-Letnikov approaches within subdomains;

– to streamline the formulation of local boundary and initial conditions within each subdomain to ensure correct synchronization of the parallel algorithm;

– to evaluate the efficiency and scalability of the proposed algorithm on parallel architectures.

The practical significance of this research is to create an algorithmic basis for effective modeling of heat and mass transfer processes and to ensure faster computational solution of heat and moisture transfer problems in complex fractal environments on multi-core systems. In addition, the application of the obtained parallel algorithm for accelerated analysis of physical processes in fractal structures will reduce computational costs compared to traditional methods and increase the performance of numerical modeling. The results of this research can be implemented in engineering, environmental and materials science simulation platforms, optimize the use of computational resources with the possibility of scaling.

Problem Statement

We consider a mathematical model of the heat and moisture transfer process in media with a fractal structure, which is described by a system of partial differential equations with fractional orders in time t and spatial coordinates x_1 and x_2 :

$$c\rho \frac{\partial^\alpha T}{\partial \tau^\alpha} = \varepsilon \rho_0 r \frac{\partial^\alpha U}{\partial \tau^\alpha} + \sum_{j=1}^2 \lambda_j \frac{\partial^\beta T}{\partial x_j^\beta}, \quad (1)$$

$$\frac{\partial^\alpha U}{\partial \tau^\alpha} = \sum_{j=1}^2 a_j \left(\frac{\partial^\beta U}{\partial x_j^\beta} + B \frac{\partial^\beta T}{\partial x_j^\beta} \right). \quad (2)$$

The initial conditions are as follows:

$$T|_{t=0} = T_0(x_1, x_2), U|_{t=0} = U_0(x_1, x_2), \quad (3)$$

where T and $V - T(t, x_1, x_2)$, $U(t, x_1, x_2)$, – the temperature and moisture content at the point (x_1, x_2) at the time t .

The boundary conditions on the boundaries $x_i = 0$ and $x_i = l_i$ ($i = 1, 2$) for determining temperature and moisture are as follows:

$$\lambda_i \frac{\partial^\nu T}{\partial x_i^\nu} \Big|_{x_i=0, l_i} + \rho_0 (1-\varepsilon) b_i (U|_{x_i=0, l_i} - U_{pi}) = \alpha_i (T|_{x_i=0, l_i} - t_{ci}), \quad (4)$$

$$a_i B \frac{\partial^\nu T}{\partial x_i^\nu} \Big|_{x_i=0, l_i} + a_i \frac{\partial^\nu U}{\partial x_i^\nu} \Big|_{x_i=0, l_i} = b_i (U_{pi} - U|_{x_i=0, l_i}), \quad (5)$$

where $(t, x_1, x_2) \in \Omega$, $\Omega = [0, t_1] \times [0, l_1] \times [0, l_2]$; ρ_0 – base density; ρ – density; ε – phase transition coefficient; a_1, a_2 – coefficients of moisture conductivity; r – specific heat of vaporization; λ_1, λ_2 – thermal conductivity coefficients; B – thermogradient coefficient; t_{c1}, t_{c2} – ambient temperature; c – specific heat capacity; U_{p1}, U_{p2} – relative humidity of the external environment; α_1, α_2 – heat exchange coefficient; β_1, β_2 – moisture exchange rate; α – fractional order of the time derivative ($0 < \alpha \leq 1$); ν, β – fractional exponents of the derivative in spatial coordinates ($0 < \nu \leq 1$), ($1 < \beta \leq 2$).

Main Material Presentation

a. Numerical solution.

To construct a numerical model for the problem of heat and moisture transfer in a medium with a fractal structure, we will use the Grunwald – Letnikov formulas [12] to determine the fractional derivative of the order and the Riemann – Liouville formulas for the fractional derivative α [6]. Specifically, the approximation of the Riemann – Liouville derivative:

$$\left. \frac{\partial^\alpha f}{\partial t^\alpha} \right|_{t^n} \approx \frac{f^{n+1} - \alpha f^n}{\Delta t^\alpha \Gamma(2-\alpha)}, (\Delta t = t^{n+1} - t^n), \quad (6)$$

where $\Gamma(\cdot)$ – gamma function.

To construct an explicit finite difference scheme, we introduce discrete time and spatial grids. The time grid $t_n = n\Delta t$, $n = 0, 1, \dots, N$. The spatial grid $x_{1,i} = i\Delta x_1$, $x_{2,j} = j\Delta x_2$, $i = 0, 1, \dots, M_1$; $j = 0, 1, \dots, M_2$. Let $T_{i,j}^n$ and $U_{i,j}^n$ denote the values of temperature and moisture at a given time t_n at the point $(x_{1,i}, x_{2,j})$. Then the explicit finite difference scheme for determining the temperature with respect to $T_{i,j}^{n+1}$ will take the following form:

$$T_{i,j}^{n+1} = \alpha T_{i,j}^n + \frac{\Delta t^\alpha \Gamma(2-\alpha)}{c\rho} + \left(\lambda_1 \frac{1}{\Delta x_1^\beta} \sum_{k=0}^m \omega_k^\beta (T_{i-k,j}^n - T_{i,j}^n) + \lambda_2 \frac{1}{\Delta x_2^\beta} \sum_{k=0}^m \omega_k^\beta (T_{i,j-k}^n - T_{i,j}^n) + \varepsilon \rho_0 r \frac{U_{i,j}^{n+1} - U_{i,j}^n}{\Delta t^\alpha} \right) \quad (7)$$

The explicit finite difference scheme for determining moisture $U_{i,j}^{n+1}$ is given by:

$$\begin{aligned} U_{i,j}^{n+1} = & \alpha U_{i,j}^n + \Delta t^\alpha \Gamma(2-\alpha) \left(a_1 \frac{1}{\Delta x_1^\beta} \sum_{k=0}^m \omega_k^\beta (U_{i-k,j}^n - U_{i,j}^n) + \right. \\ & + a_2 \frac{1}{\Delta x_2^\beta} \sum_{k=0}^m \omega_k^\beta (U_{i,j-k}^n - U_{i,j}^n) + a_1 B \frac{1}{\Delta x_1^\beta} \sum_{k=0}^m \omega_k^\beta (T_{i-k,j}^n - T_{i,j}^n) + \\ & \left. + a_2 B \frac{1}{\Delta x_2^\beta} \sum_{k=0}^m \omega_k^\beta (T_{i,j-k}^n - T_{i,j}^n) \right), \end{aligned} \quad (8)$$

where

$$\omega_k^\beta = \frac{(-1)^k \Gamma(\beta+1)}{\Gamma(\beta-k+1) \Gamma(k+1)}. \quad (9)$$

b. Application of the Schwarz method

The Schwarz method (or Schwarz method for partitioning a domain) is an iterative method used to solve boundary value problems for partial differential equations, elliptic equations in partial domains of complex domains [10]. Its basic idea is to partition the initial domain into subdomains on which the equations can be solved by simpler methods. The results from the subdomains are then used to construct a solution for the entire domain.

The main steps of the Schwarz method are:

- Partitioning the domain into several subdomains, which may overlap.
- On each subdomain, a partial problem is solved using boundary conditions obtained from the solutions of neighboring subdomains.
- The solution process is repeated iteratively until the solution on each subdomain stabilizes and reaches the required accuracy.

The Schwarz method is widely used in numerical methods, particularly in computational mechanics, in modeling physical phenomena [19] in engineering [14], mathematics [8], natural sciences [16], and other sciences [10], [11]. Due to its ability to efficiently handle complex domains and distribute computations between different processors, it is a key tool for solving large and complex mathematical problems. Due to the subdivision into subdomains, the Schwarz method is easily parallelized, allowing it to be utilized on multicore computers [5], [6], computers with GPU processors [7], [8], as well as in distributed cluster systems [10]. The method is adapted for various types of differential equations and boundary conditions [12]. The subdivision into subdomains enables working with regions of complex geometry, where traditional methods may be inefficient or difficult to apply [14].

The scheme for subdividing the domain into three overlapping subdomains is shown in Fig. 1. The overlapping ensures the correct transmission of boundary conditions between adjacent subdomains.

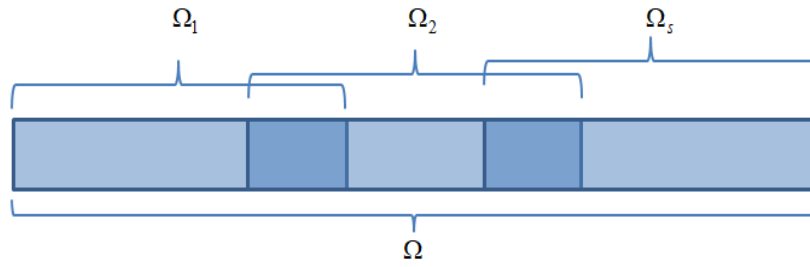


Fig. 1. Diagram of the overlapping subdomain partitioning

Modern versions of the Schwartz method use complex iterative and optimization schemes to improve the speed and accuracy of solutions. In particular, variants of the method [17] have been developed that include adaptive partitioning of domains, the use of different basis functions for each subdomain, and various techniques to accelerate the convergence of iterations [16].

We will subdivide the computational domain into S subdomains of equal size, as we will be using a multicore computer with s processors of equal power. To solve the system of differential equations, we use the explicit scheme Eqs. (6) and (7). The parallel algorithm based on the Schwarz method for subdividing the two-dimensional discretization domain into overlapping subdomains consists of the following steps:

1. Divide the domain Ω into S subdomains $\Omega_1, \Omega_2, \dots, \Omega_S$, that may overlap.
2. Formulate local problems for each subdomain $\Omega_i, i = 1, 2, \dots, S$ to determine the temperature T_i and moisture U_i . For subdomain Ω_i the equations for temperature and moisture are similar to Eqs. (1) and (2), replacing T with T_i and U with U_i .
3. For convenience, the initial and boundary conditions for each subdomain Ω_i are kept the same as in Eq. (3), Eq. (4), and Eq. (5), respectively.
4. Initiate a parallel iterative process to solve local problems in each subdomain.
5. Based on the known initial conditions, all processors simultaneously compute the values of $T_i|_{t=0}$ and $U_i|_{t=0}$ at the initial time step. The subsequent steps of the algorithm are performed similarly for each time step $t = 1, 2, \dots, m$. Taking into account the boundary conditions obtained from the previous time step from neighboring subdomains, at each time step in each subdomain, the equations for temperature Eq. (7) and moisture Eq. (8) will be solved. For the current time layer t , all processors calculate $T_i|_t$ and $U_i|_t$ in parallel. A similar calculation is organized for subdomains $\Omega_{i+1}, \dots, \Omega_S$.
6. Update boundary conditions on the subdomain boundaries Ω_i and Ω_{i+1} transferring information between neighboring subdomains. This involves exchanging boundary values between neighboring processors. The processor responsible for subdomain $\Omega_{i+1}, (i = 1, 2, \dots, S - 1)$ passes the increment in the time step of the desired function to its left neighbor.
7. Transition to the calculation of the next time layer.
8. Repeat the iterative process from step 5, where all processors simultaneously calculate the values of functions $T_i|_t$ and $U_i|_t$ in all subdomains Ω_i at the next time layer t . Iterations continue until the solution in each subdomain Ω_i stabilizes. After completing the parallel iterations, the results from all subdomains are merged into the arrays T_i and U_i .

c. Parallel discretization algorithm

The proposed algorithm can be represented by pseudo-code Fig. 2. We create one thread for each subdomain, where each thread computes the solution for its subdomain and synchronizes its results through a shared list results using a lock.

Algorithm 1 Parallel Iterative Process

```

1: Using parallel libraries and environment
2: Init variables and arrays, parameters and constants
3: Initialization functions:
4: fractional_derivative_time( $T, \alpha, dt$ ) {...}
5: fractional_derivative_space( $U, \beta, dx$ ) {...}
6: boundary_conditions( $T, U, T_{ci}, U_{pi}$ ) {...}
7: initialize_subdomain(rank,  $T, U, S$ ) {...}
8: update_subdomain( $T_{sub}, U_{sub}$ ) {...}
9: function PARALLEL_PROCESS(rank,  $T, U, results, lock$ )
10:   ( $T_{sub}, U_{sub}$ )  $\leftarrow$  initialize_subdomain(rank,  $T, U, S$ )
11:    $t \leftarrow 0$ 
12:   while  $t < T_{max}$  do
13:     ( $T_{new}, U_{new}$ )  $\leftarrow$  update_subdomain( $T_{sub}, U_{sub}$ )
14:     boundary_conditions( $T_{new}, U_{new}, t_{ci}, U_{pi}$ )
15:     lock.acquire()
16:      $results[rank] \leftarrow (T_{new}, U_{new})$ 
17:     lock.release()
18:      $T_{sub}, U_{sub} \leftarrow T_{new}, U_{new}$ 
19:      $t += dt$ 
20:   end while
21: end function
22: function MAIN_PROCESS
23:    $threads \leftarrow [new]$ 
24:    $results \leftarrow [i] * S$ 
25:    $lock \leftarrow threading.Lock()$ 
26:   for rank  $\leftarrow 0$  to  $S - 1$  do
27:      $thread \leftarrow parallel\_process, args=(rank, T, U, results, lock)$ 
28:      $threads.append(thread)$ 
29:      $thread.start()$ 
30:   end for
31:    $T_{final} \leftarrow np.vstack([results[rank][0] \text{ for rank in range}(S)])$ 
32:    $U_{final} \leftarrow np.vstack([results[rank][1] \text{ for rank in range}(S)])$ 
33: end function

```

Fig. 2. Pseudo-code of the parallel algorithm

Let's take a closer look at how each thread works. The thread initializes its subdomain by calling the `initialize_subdomain()` function, which returns the parts of the T and U arrays corresponding to that subdomain. Starting at time $t = 0$, the thread executes an iterative process until T_{max} is reached. At each iteration, the `update_subdomain()` function is called, which calculates new values of T and U for the subdomain using fractional derivatives in time and space. After updating the values, the thread sets boundary conditions by calling the `boundary_conditions()` function. The thread uses lock to write the results of its subdomain to a shared results array. The main program waits for all threads to finish using the `join()` method. After all threads are complete, the results of all subdomains are merged into the final arrays T_{final} and U_{final} .

After the calculations are complete, you can visualize the results as temperature and humidity distribution graphs.

Once the computations are done, the results can be visualized as graphs showing the distribution of temperature and moisture.

Results and Discussion

The temperature distribution graphs Fig. 3 and moisture distribution graphs Fig. 4 for a 2D domain divided into 4 parts demonstrate the state of moisture in a fractal-structured medium at the specified time $T/2$. The graphs illustrate how temperature and moisture are redistributed among the subdomains depending on the initial and boundary conditions as well as the anisotropic properties of the fractal-structured medium. The process of heat and moisture transfer from one part of the domain to another is evident, creating gradients. Dividing the domain into 4 parts allows for an accelerated investigation process and enables optimization of conditions to achieve the desired levels of temperature and moisture in the fractal-structured medium.

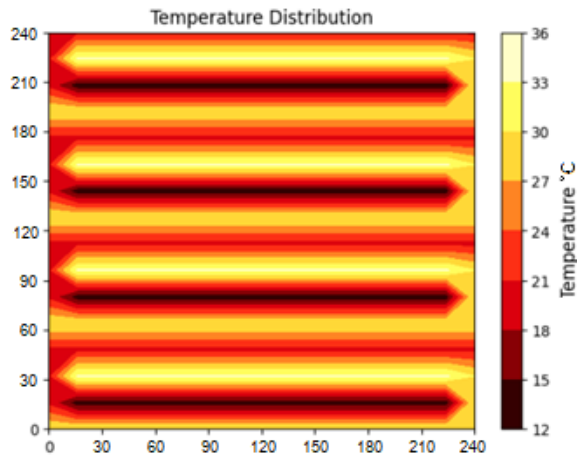


Fig. 3. Temperature distribution for a domain discretized into 4 parts

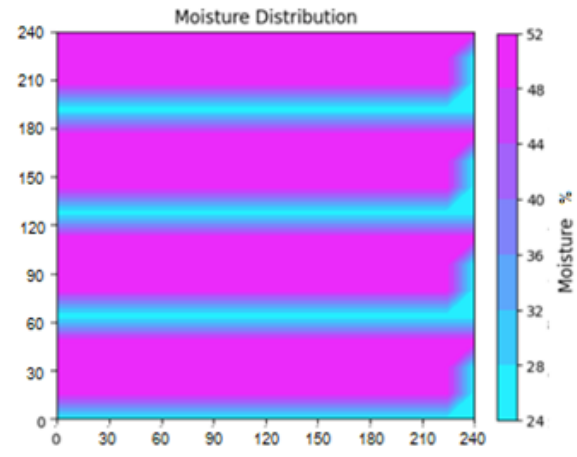


Fig. 4. Moisture distribution for a domain discretized into 4 parts

The following input parameters are defined: “Initial Ω size” – 50×50 nodes, “Grid step change” – 10, and “Number of experiments” – 100. Graphs showing the dynamics of parallel algorithm speedup will be constructed Fig. 5. The graph Fig. 5 shows how the speedup of the parallel algorithm increases rapidly with the number of nodes. The upward trend continues up to the 30th experiment. Starting from the 20th experiment, the algorithm becomes effective when using a computational grid size of 240×240 nodes. At this point, the speedup of the algorithm reaches approximately 4.4.

The graph Fig. 6 shows the performance graphs of the algorithm without partitioning the region and with partitioning the region into 4 subregions. It can be seen that the computation time of the parallel algorithm using the Schwartz method is significantly shorter than the execution time of the algorithm without partitioning the region.

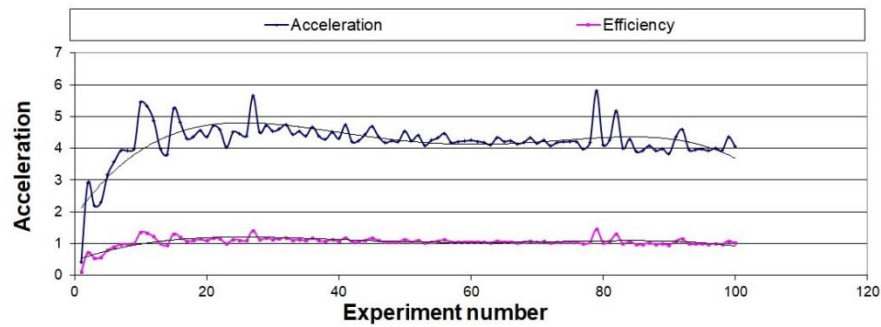


Fig. 5. Dynamics of Acceleration and Efficiency (CPU Core i3-8100)

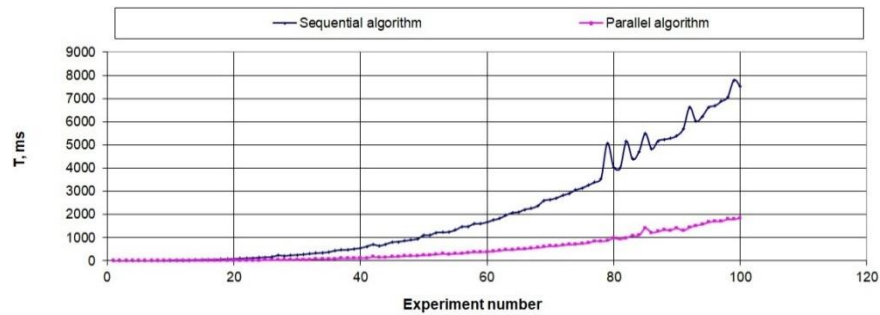


Fig. 6. Algorithm performance without partitioning the region and with partitioning the region into 4 parts

Table 1 presents the experimental results of the parallel algorithm. Given that the algorithm becomes effective starting from the 20th experiment Fig. 6, we analyze the obtained results for the domain discretized into 240×240 nodes. Here, T is the execution time (c), S is the speedup, and E is the efficiency.

Table 1

Experimental data for 240 nodes

240×240	Number of Threads				
	1	2	4	8	16
T	384.0	182.7	86.9	41.3	19.7
S		2.1	4.4	9.3	19.5
E		1.05	1.10	1.16	1.22

Execution time significantly decreases with the increase in the number of threads. With 16 threads, the execution time drops to 19.7 seconds, and the maximum speedup reaches 19.5. The efficiency is quite high, varying from 1.05 with 2 threads to 1.22 with 16 threads. This indicates that the algorithm utilizes resources even better than expected, with efficiency values exceeding 1. As the number of threads increases, the algorithm significantly reduces execution time, demonstrating very good scalability.

Table 2 presents the experimental results of the parallel algorithm for discretizing a 2D domain using the Schwarz method. The domain size Ω is 1050 nodes.

Table 2

Experimental data for 1050 nodes

1050×1050	Number of Threads				
	1	2	4	8	16
T	7562.3	3992.7	1736.0	935.6	437.3
S		1.9	4.4	8.1	17.3
E		0.95	1.09	1.01	1.08

Sequential execution of the algorithm took 7562.3 seconds. With 16 threads, the execution time decreases to 437.3 seconds. The maximum speedup of 17.3 is achieved with 16 threads. The efficiency is quite high, ranging from 0.95 with 2 threads to 1.09 with 4 threads. Speedup increases with the use of a larger number of subdomains, indicating effective parallelization of the heat and mass transfer problem in fractal-structured media described by differential equations of fractional order.

Conclusions

A parallel algorithm has been developed to solve a system of fractional-order partial differential equations describing heat and mass transfer processes in fractal-structured media. The parallel algorithm, based on the Schwarz method, utilizes a decomposition principle to distribute the overall computation across multiple subdomains. For parallel processing of subdomains, local problems are formulated, initial and boundary conditions are set for the subdomains, solution updates are managed within each subdomain, and information exchange between subdomains is implemented to check for method convergence. According to the conducted experiments, the algorithm's efficiency is close to one, indicating good scalability. As the number of threads increases, resource usage is optimal. Based on the Schwartz method, a parallel algorithm was developed to divide a two-dimensional discretization domain into overlapping subdomains for solving a system of fractional-order partial differential equations describing heat and mass transfer processes in anisotropic media with a fractal structure. It turns out to be effective. Of course, there is an upper limit to the number of experiments. One can perform hundreds of experiments and obtain good speedup and efficiency figures, even on very large grid sizes. The efficiency of the algorithm, when divided into 4 parts, reaches 1.2 on computers with 4 core processors with increasing number of experiments.

References

- [1] V. Dolean, P. Jolivet, and F. Nataf, *An Introduction to Domain Decomposition Methods: Algorithms, Theory and Parallel Implementation*. Master, France, 2015. DOI: [ffcel-01100932v6](https://doi.org/10.1007/978-1-4939-9326-6).
- [2] H. Yang and X. C. Cai, “Two-Level Space-Time Domain Decomposition Methods for Flow Control Problems”, *J. Sci. Comput.*, Vol. 70, pp. 717–743, 2017, <https://doi.org/10.1007/s10915-016-0263-0>.
- [3] F. Magoulès and G. Gbikpi-Benissan, “Asynchronous Parareal Time Discretization For Partial Differential Equations”, *SIAM J. Sci. Comput.*, Vol. 40, No. 6, 2018.
- [4] J.-L. Lions, Y. Maday, and G. Turinici, “Résolution d’EDP par un schéma en temps “pararéel”, *C. R. Math. Acad. Sci. Paris Ser. I*, Vol. 332, pp. 661–668, 2001.
- [5] Y. Sokolovskyy, V. Yarkun, and M. Levkovych, “Parallel Algorithm for Numerical Modeling of Anisotropic Heat and Mass Transfer in Fractal Media,” in 2023 17th Int. Conf. on the Experience of Designing and Application of CAD Systems (CADSM), Jaroslaw, Poland, 2023, pp. 39–43, <https://doi.org/10.1109/CADSM58174.2023.10076515>.
- [6] Y. Sokolovskyy, V. Shymanskyi, M. Levkovych, and V. Yarkun, “Mathematical Modeling of Heat and Moisture Transfer and Rheological Behavior in Materials with Fractal Structure Using the Parallelization of Predictor-Corrector Numerical Method,” in 2016 IEEE First Int. Conf. on Data Stream Mining & Processing (DSMP), Lviv, Ukraine, 2016, pp. 108–111, <https://doi.org/10.1109/DSMP.2016.7583518>.
- [7] Y. Sokolovskyy, A. Nechepurenko, T. Samotii, S. Yatsyshyn, O. Mokrytska, and V. Yarkun, “Software and Algorithmic Support for Finite Element Analysis of Spatial Heat-and-Moisture Transfer in Anisotropic Capillary-Porous Materials”, in IEEE 3rd Int. Conf. on Data Stream Mining and Processing (DSMP 2020), 2020, pp. 316–320, <https://doi.org/10.1109/DSMP47368.2020.9204175>.
- [8] A. Foadaddini, S. A. Zolfaghari, H. M. Darian, and H. Saadatfar, “A New GPU-Based Corrected Explicit-Implicit Domain Decomposition Scheme for Convection-Dominated Diffusion Problems”, *Computers & Mathematics with Applications*, Vol. 123, pp. 184–203, 2022, <https://doi.org/10.1016/j.camwa.2022.08.015>.
- [9] A. Foadaddini, S. A. Zolfaghari, H. M. Darian, et al., “An Efficient GPU-Based Fractional-Step Domain Decomposition Scheme for the Reaction – Diffusion Equation”, *Comput. Appl. Math.*, Vol. 39, No. 305, 2020, <https://doi.org/10.1007/s40314-020-01357-7>.
- [10] F. Kong and X.-C. Cai, “A Highly Scalable Multilevel Schwarz Method with Boundary Geometry Preserving Coarse Spaces for 3D Elasticity Problems on Domains with Complex Geometry”, *SIAM J. Sci. Comput.*, Vol. 38, No. 2, pp. C73–C95, <https://doi.org/10.1137/15M1010567>.
- [11] A. O. Lopushansky and H. P. Lopushanska, “Inverse Problem for the Fractional Diffusion Equation in Schwarz-Type Spaces”, *J. Math. Sci.*, Vol. 265, pp. 394–407, 2022, <https://doi.org/10.1007/s10958-022-06060-y>.
- [12] I. Podlubny, *Fractional Differential Equations*, Vol. 198 of Mathematics in Science and Engineering. Academic Press, San Diego, CA, USA, 1999, 340 p.
- [13] M. Gander and H. Zhang, “Schwarz Methods by Domain Truncation”, *Acta Numerica*, Vol. 31, pp. 1–134, 2022, <https://doi.org/10.1017/S0962492922000034>.
- [14] N. Ghosh, A. Reusken, and B. Stamm, “Schwarz Domain Decomposition Method Applied to the Conductor-Like Screening Model”, 2024.
- [15] G. Sanjuan, T. Margalef, and A. Cortés, “Wind Field Parallelization Based on Schwarz Alternating Domain Decomposition Method”, *Future Generation Computer Systems*, Vol. 82, 2017, <https://doi.org/10.1016/j.future.2016.12.041>.
- [16] Y. Sokolovskyy, V. Yarkun, M. Levkovych, O. Storozhuk, and I. Kapran, “Software and Algorithmic Aspects of Parallel Calculation of Non-Isothermal Moisture Transfer in Fractal-Structure Materials,” in 2021 IEEE XVIIth Int. Conf. on the Perspective Technologies and Methods in MEMS Design (MEMSTECH), Polyana, Ukraine, 2021, pp. 171–175, <https://doi.org/10.1109/MEMSTECH53091.2021.9467939>.
- [17] B. Shao, H. Yang, and H.-J. Zhao, “Scalable Fully Implicit Methods for Subsurface Flows in Porous Media with Fractional Derivative”, *Computers & Mathematics with Applications*, Vol. 134, pp. 55–65, 2023, <https://doi.org/10.1016/j.camwa.2023.01.003>.
- [18] X. Li, X. Jiang, F. Zeng, Z. Lin, S. Qin, and R. Chen, “Enhanced Parallel Computation for Time-Fractional Fluid Dynamics: A Fast Time-Stepping Method with Newton-Krylov-Schwarz Solver”, *Communications in Nonlinear Science and Numerical Simulation*, Vol. 133, 2024, <https://doi.org/10.1016/j.cnsns.2024.107952>.
- [19] L. Mochurad, “Implementation and Analysis of a Parallel Kalman Filter Algorithm for LiDAR Localization Based on CUDA Technology”, *Frontiers in Robotics and AI*, Vol. 11, 2024, <https://doi.org/10.3389/frobt.2024.1341689>.
- [20] M. Gander, “Schwarz methods over the course of time”, *ETNA. Electronic Transactions on Numerical Analysis*, Vol. 31, 2008, pp. 228–255.

[21] W. Zhang, W. Wei, X. Cai, "Performance modeling of serial and parallel implementations of the fractional Adams-Bashforth-Moulton method", *Fract. Calc. Appl. Anal.*, Vol. 17, pp. 617–637, 2014, <https://doi.org/10.2478/s13540-014-0189-x>

Ярослав Соколовський¹, Володимир Яркун², Мар'яна Левкович³

¹ Кафедра систем автоматизованого проєктування, Національний університет "Львівська політехніка", вул. С. Бандери, 12, Львів, Україна, E-mail: yaroslav.i.sokolovskyi@lpnu.ua, ORCID 0000-0003-4866-2575

² Кафедра інженерії програмного забезпечення, Національний лісотехнічний університет України, вул. Генерала Чупринки, 103, Львів, Україна, E-mail: yarkun@nltu.edu.ua, ORCID 0000-0002-4876-1111

³ Кафедра систем автоматизованого проєктування, Національний університет "Львівська політехніка", вул. С. Бандери, 12, Львів, Україна, E-mail: mariana.v.levkovych@lpnu.ua, ORCID 0009-0004-6231-7799

ПАРАЛЕЛЬНИЙ АЛГОРИТМ ДЛЯ ВИРІШЕННЯ ЗАДАТКІВ ТЕПЛОМАСОПЕРЕНОСУ У ФРАКТАЛЬНИХ СЕРЕДОВИЩАХ НА ОСНОВІ МЕТОДУ ШВАРЦА

Отримано: Серпень 22, 2025 / Переглянуто: Вересень 12, 2025 / Прийнято: Вересень 15, 2025

© Соколовський Я. *, Яркун В., Левкович М., 2025

Анотація. На основі методу Шварца розроблено паралельний алгоритм для розділення двовимірної області дискретизації на піддомен, які перекриваються, для розв'язання системи диференціальних рівнянь з частинними похідними дробового порядку, що описує процеси тепло- та масоперенесення в анізотропних середовищах з фрактальною структурою. За допомогою методу скінченних різниць отримано явні різниці схеми для рівнянь температури та вологості в піддоменах. Дробові похідні за часом та просторовими координатами апроксимуються за допомогою формул Рімана – Ліувілля та Грюнвальда – Летнікова відповідно. Для паралельної обробки піддоменів організовано формулювання локальних задач, оптимізовано формулювання локальних граничних та початкових умов у межах кожної піддоменної області для забезпечення належної синхронізації паралельного алгоритму, оновлено рішення в кожній піддоменній області та передано інформацію між піддоменними областями для перевірки збіжності методу. Експериментальні результати підтверджують хорошу масштабованість запропонованого паралельного алгоритму.

Ключові слова: паралельний алгоритм, декомпозиція домену, метод Шварца, середовища з фрактальною структурою, процеси тепло- та вологопереносу.

* Corresponding author



© The Author(s). This is an open access article distributed under the terms of the Creative Commons Attribution Licence 4.0 (<https://creativecommons.org/licenses/by/4.0/>)