

ITC 2/46

Journal of Information Technology
and Control
Vol. 46 / No. 2 / 2017
pp. 171-182
DOI 10.5755/j01.itc.46.2.17298
© Kaunas University of Technology

**Application of Real Ant Colony Optimization Algorithm to Solve
Space and Time Fractional Heat Conduction Inverse Problem**

Received 2016/12/20

Accepted after revision 2017/04/05

 <http://dx.doi.org/10.5755/j01.itc.46.2.17298>

Application of Real Ant Colony Optimization Algorithm to Solve Space and Time Fractional Heat Conduction Inverse Problem

Rafał Brociek, Damian Słota

Institute of Mathematics, Silesian University of Technology, Kaszubska 23, 44-100 Gliwice, Poland

Corresponding author: rafal.brociek@polsl.pl

This paper describes the method of solution of the space fractional and 2D time fractional heat conduction inverse problem. In this paper the authors consider two models – 1D space fractional heat conduction equation and 2D time fractional heat conduction equation with the initial-boundary conditions. To solve the inverse heat conduction problem, a functional defining the error of approximate solution must be minimized. To minimize this functional the Real Ant Colony Optimization (ACO) algorithm was used. In order to reduce the computational time, the calculations were performed in a parallel (multi-threaded) way. The paper presents examples to illustrate the accuracy and stability of the presented algorithm.

KEYWORDS: Ant Colony Optimization Algorithm, Inverse Problem, Identification, Time Fractional Heat Conduction Equation, Space Fractional Heat Conduction Equation.

Introduction

Inverse problems are very important issues in science, they have a wide application in signal processing, communication theory, physics and many other fields of engineering. In this paper the authors consider the space and time fractional heat conduction inverse problem which consists in reconstructing the boundary condition in the fractional heat conduction models, basing on the temperature measurements. In

papers [13-15] the heat conduction inverse problems with the classical derivative are considered, whereas in articles [4-6] the fractional heat conduction inverse problems are investigated.

The artificial intelligent algorithms, particularly the algorithms inspired by nature, are very popular in solving various practical and theoretical problems [1,

10, 13-15, 21, 35, 40-43]. The most popular and efficient algorithms inspired by nature are the following algorithms: Ant Colony algorithms [12, 36], Artificial Bee Colony algorithm [16-18, 33] and Firefly algorithm [35]. In many cases these types of algorithms provide better results than the conventional algorithms and, what is more, they are easy to implement. In case of optimization algorithms inspired by nature, another good feature of these algorithms is the fact that they do not need any requirements about minimized function, except the existence of the solution.

Fractional calculus is very useful to model many various types of physical and technical phenomena [8, 9, 11, 25, 26, 30, 34]. Application of fractional calculus can be found, for example, in electrical engineering [26], control theory [8, 11], mechanics [9]. In papers [30, 46] the authors consider the model of heat conduction in ceramic and composite medium. The models containing fractional derivative better describe the heat conduction process than the models with classical derivative. To solve fractional heat conduction inverse problem, we need first to solve the direct problem. In paper [29] Murio presents the numerical method of solving the time fractional diffusion equation with Dirichlet zero boundary conditions. Meerschaeert in paper [22] describes the numerical solution of the space fractional diffusion equation with boundary condition of the first kind, and in paper [23] the authors present the finite difference method for two-dimensional fractional dispersion equation. In both papers, as the fractional derivative, the Riemann-Liouville derivative was used. In paper [3] the author presents the numerical solution of time fractional heat conduction equation with Neumann and Robin boundary conditions, and in paper [7] the authors consider the space fractional heat conduction equation with mixed boundary conditions.

In papers [27, 28] Murio deals with the inverse problems of fractional order. Article [27] presents the solution of the time fractional inverse heat conduction problem with Caputo fractional derivative and in paper [28] the author reconstructs the heat flux in the fractional-diffusion heat conduction equation. Also in paper [24] the inverse diffusion problem is considered. The problem consists in determining the spatial coefficient and the order of derivative. The authors prove that under certain conditions the solution of the problem is unique. The proof is done by transforming

the solution to the solution of the wave equation. In paper [38] the inverse problems of fractional order are considered. The inverse source problem is transformed into a first kind Volterra integral equation. Further, the authors use the boundary element method and Tikhonov regularization to solve the Volterra integral equation of the first kind. Many other authors deal also with the various kinds of fractional inverse problems, see for example [2, 4-6, 20, 39, 44-46].

This paper describes an application of the parallel version of Real Ant Colony Optimization algorithm to reconstruct the heat flux at the boundary where the temperature distribution in measurement points is given. Two models are considered: 1D space fractional heat conduction equation and 2D time fractional heat conduction equation. To reconstruct the heat flux, a functional defining the error of approximate solution is minimized. In this purpose we use the Real Ant Colony Optimization algorithm, which inspiration is taken from the behavior of ant swarms, widely regarded as the very intelligent communities, especially because of their tactics in search for the shortest path connecting the anthill with the source of food. In order to speed up the solving procedures we used the parallelization of the ant algorithm which significantly reduced the computation time. The direct problem in the proposed approach was solved by applying the implicit finite difference method [3, 7, 22, 23]. The paper also includes some examples illustrating the accuracy and stability of the presented procedures.

Formulation of the problem

We consider two mathematical models of fractional heat conduction equation.

Model I

First of all we introduce the following space fractional heat conduction equation

$$c\rho \frac{\partial u(x,t)}{\partial t} = \lambda(x) \frac{\partial u^\alpha(x,t)}{\partial x^\alpha} \quad (1)$$

defined in region

$$D = \{ (x, t): x \in [a, b], t \in [0, t^*] \},$$

where c, ρ, λ denote the specific heat, density and thermal conductivity, respectively. Equation (1) is completed with the initial condition

$$u(x, 0) = f(x), \quad x \in [a, b], \tag{2}$$

and the boundary conditions of the second and third kind

$$-\lambda(a) \frac{\partial u(a,t)}{\partial x} = q(t), \quad t \in (0, t^*), \tag{3}$$

$$-\lambda(b) \frac{\partial u(b,t)}{\partial x} = h(t)(u(b, t^*) - u^\infty), \quad t \in (0, t^*), \tag{4}$$

where h is the heat transfer coefficient, q is the heat flux and u^∞ denotes the ambient temperature.

The space fractional derivative occurring in equation (1) is interpreted in the sense of the left-sided Riemann-Liouville derivative, which is defined by formula [34]:

$$\frac{\partial^\alpha u(x,t)}{\partial x^\alpha} = \frac{1}{\Gamma(n-\alpha)} \frac{\partial^n}{\partial x^n} \int_a^x u(s,t)(x-s)^{n-1-\alpha} ds, \tag{5}$$

where Γ is the Gamma function, $\alpha \in (n-1, n]$. In case of $\alpha \in (1, 2)$ equation (1) describes the phenomenon of super-diffusion, whereas for $\alpha = 2$ we get the differential equation with classical derivative. In this paper we investigate $\alpha \in (1, 2)$.

Model II

Now, let us consider the 2D time fractional heat conduction equation

$$c\rho \frac{\partial u^\alpha(x,y,t)}{\partial t^\alpha} = \lambda_1(x,y,t) \frac{\partial u^2(x,y,t)}{\partial x^2} + \lambda_2(x,y,t) \frac{\partial u^2(x,y,t)}{\partial y^2} + g(x,y,t) \tag{6}$$

defined in region

$$D = \{ (x, y, t): x \in [0, L_x], y \in [0, L_y],$$

$$t \in [0, t^*], L_x, L_y, t^* \in \mathbb{R}_+ \},$$

where $\alpha \in (0, 1)$, c is the specific heat, ρ is the density and $\lambda_1, \lambda_2 > 0$ for $(x, y, t) \in D$. To equation (6) we add the initial condition

$$u(x, y, 0) = f(x, y), \quad x \in [0, L_x], y \in [0, L_y], \tag{7}$$

and the Neumann (for $y=0, y=L_y$) and Robin (for $x=0, x=L_x$) boundary conditions:

$$-\lambda_2(x, 0, t) \frac{\partial u(x,0,t)}{\partial y} = q_1(t, x), \tag{8}$$

$$t \in [0, t^*], x \in [0, L_x],$$

$$-\lambda_2(x, L_y, t) \frac{\partial u(x,L_y,t)}{\partial y} = q_2(t, x), \tag{9}$$

$$t \in [0, t^*], x \in [0, L_x],$$

$$-\lambda_1(0, y, t) \frac{\partial u(0,y,t)}{\partial x} = h_1(t, y)(u(0, y, t) - u^\infty), \tag{10}$$

$$t \in [0, t^*], y \in [0, L_y],$$

$$-\lambda_1(L_x, y, t) \frac{\partial u(L_x,y,t)}{\partial x} = h_2(t, y)(u(L_x, y, t) - u^\infty), \tag{11}$$

$$t \in [0, t^*], y \in [0, L_y],$$

where h_1, h_2 are the heat transfer coefficients, q_1, q_2 are the heat fluxes and u^∞ is the ambient temperature. In this model, the fractional derivative with respect to time, occurring in equation (6) is the Caputo derivative defined by the following equation

$$\frac{\partial^\alpha u(x,t)}{\partial t^\alpha} = \frac{1}{\Gamma(n-\alpha)} \int_a^t \frac{\partial^n u(x,s)}{\partial s^n} (t-s)^{n-1-\alpha} ds, \tag{12}$$

where Γ is the Gamma function and $\alpha \in (n-1, n]$.

In case of Model I we assume that the function h , occurring in boundary condition (4), will depend on n parameters $a_i (i = 1, 2, \dots, n)$. In Model II we need to reconstruct functions h_1, h_2 occurring in the Robin boundary conditions. These functions also depend on n parameters $b_i (b_1, b_2, \dots, b_n)$. The considered inverse problem consists in restoring the parameters a_i (and, therefore, the boundary condition in Model I) and parameters b_i (in Model II). Additional information is delivered by the temperature measurements (values of function u) at the selected points inside region D . The known values of function u (the input data for inverse problem) at the selected points (x_i, t_k) (or (x_i, y_j, t_k) in case of Model II) of region D will be denoted as follows

$$u(x_i, t_k) = \hat{U}_{ik}, \quad i = 1, 2, \dots, N_1, \quad k = 1, 2, \dots, N_2, \quad (13)$$

(Model I)

$$u(x_i, y_j, t_k) = \hat{U}_{(ij)k}, \quad (ij) = 1, 2, \dots, N_1, \quad k = 1, 2, \dots, N_2, \quad (14)$$

(Model II)

where N_1 is the number of sensors and N_2 denotes the number of measurements at each sensor. Solving the direct problem for the fixed values of coefficients a_i (or b_i in case of Model II) we obtain the values approximating function u at the selected points $(x_p, t_k) \in D$ (or (x_p, y_p, t_k)). These values will be denoted by $U_{ik}(h)$ (or $U_{(ij)k}(h_1, h_2)$). In this way, basing on this computation and the input data, we create the following functional defining the error of approximate solution

$$F(h) = \sum_{i=1}^{N_1} \sum_{k=1}^{N_2} (U_{ik}(h) - \hat{U}_{ik})^2, \quad (15)$$

(for Model I)

$$F(h_1, h_2) = \sum_{(ij)=1}^{N_1} \sum_{k=1}^{N_2} (U_{(ij)k}(h_1, h_2) - \hat{U}_{(ij)k})^2, \quad (16)$$

(for Model II)

By minimization of these functionals we will reconstruct the heat transfer coefficients h, h_1, h_2 .

Method of solution

Direct problems, defined by equations (1)-(4) (Model I) and (6)-(11) (Model II), for the fixed values of h, h_1 and h_2 is solved by using the implicit finite difference method [3, 7, 23].

To reconstruct the heat transfer coefficients, it is necessary to minimize functionals (15) and (16). For this purpose we use the parallel version of Real Ant Colony Optimization (ACO) algorithm [5, 36]. The ACO algorithm was inspired by observation of the ant colonies behavior, widely regarded as the efficient and intelligent communities. Because it is a heuristic algorithm, the calculation need to be repeated a certain number of times. In this paper it is ten times. In order to reduce computation time, we adapted the algorithm for parallel computing. To describe the algorithm, we will use the following notation

F – minimized function, n – dimension (number of variables), nT – number of threads, $M = nT \cdot p$ – num-

ber of ants, I – number of iteration, L – number of pheromone spots, q, ξ – parameters of the algorithm.

Now we present the steps of the algorithm.

Initialization of the algorithm

- 1 Setting the input parameters of the algorithm L, M, I, nT, q, ξ .
- 2 Generating randomly L pheromone spots (solutions). Assigning them to set T_0 (starting archive).
- 3 Calculating the value of minimized function F for each pheromone spot and organizing the archive T_0 from the best solution to the worst.

Iterative process

- 4 Assigning probabilities to pheromone spots (solutions) according to the following formula

$$p_l = \frac{\omega_l}{\sum_{j=1}^L \omega_j} \quad l = 1, 2, \dots, L,$$

where wights ω_l are associated with l -th solution and expressed by formula

$$\omega_l = \frac{1}{qL\sqrt{2\pi}} \cdot e^{-\frac{(l-1)^2}{2q^2L^2}}.$$

- 5 Ant chooses a random l -th solution with probability p_l .
- 6 Ant transforms the j -th coordinate ($j = 1, 2, \dots, n$) of l -th solution s_j^l by proximity sampling with the probability density function (Gaussian function)

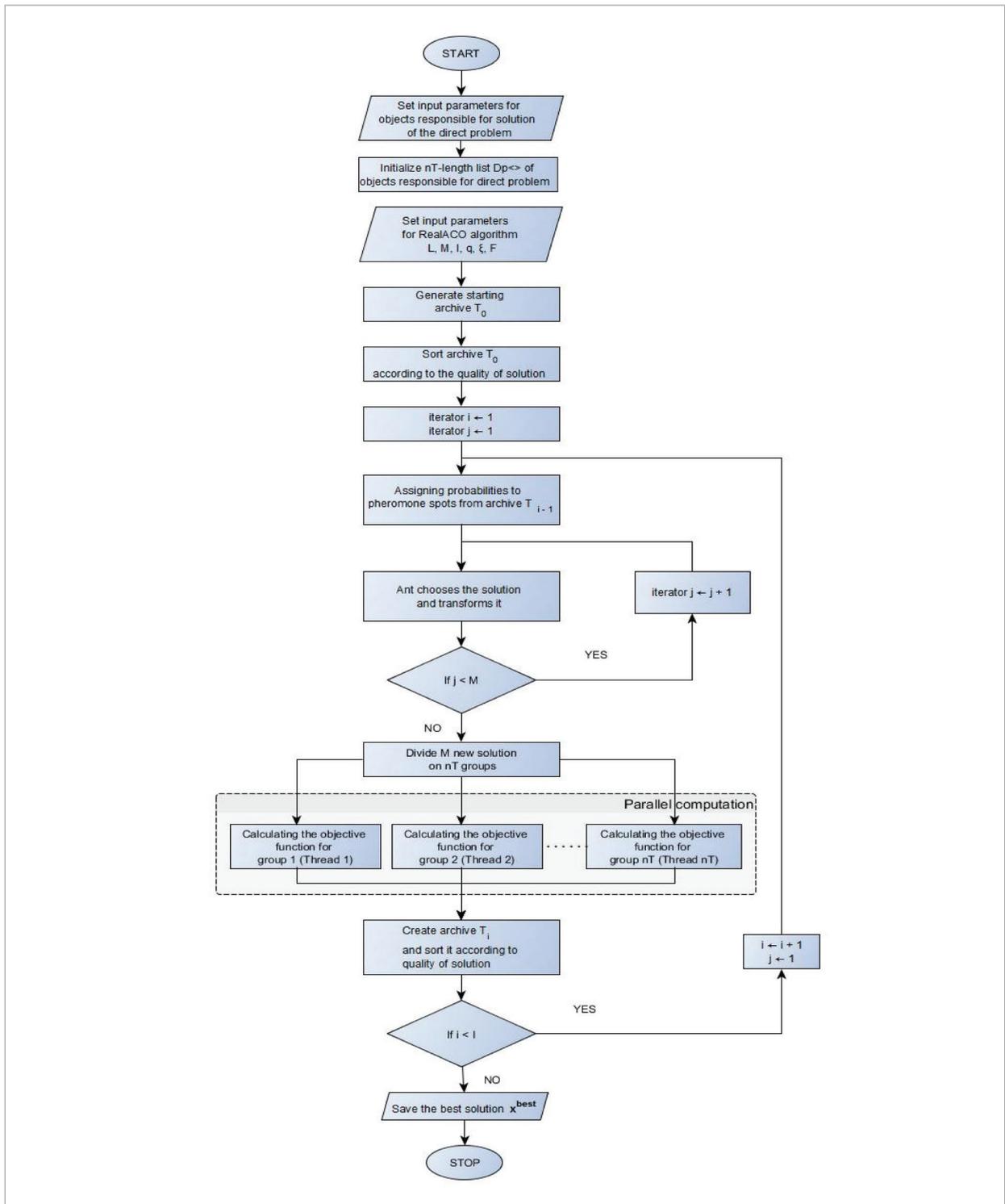
$$g(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

where $\mu = s_j^l, \sigma = \frac{\xi}{L-1} \sum_{p=1}^L |s_p^l - s_j^l|$.

- 7 Repeating steps 5-6 for each ant. Obtaining M new solutions (pheromone spots).
- 8 Dividing the new solutions on nT groups. Calculating the value of minimized function F for each new solution (parallel computing).
- 9 Adding the new solutions to the archive T_p , organizing the archive with respect to quality, removing M worst solutions.
- 10 Repeating steps 3 – 9 I times.

Figure 1

Control block diagram of the procedure reconstructing the boundary condition by using RealACO algorithm



Experimental results

The proposed algorithm was implemented in C# 5.0 on the computer with the following parameters: CPU: Intel Core i5-3230M 2.60GHz; OS: Microsoft Windows 10 Home; RAM: 8.00 GB. The multi-threaded calculations were performed by using the Task Parallel Library.

Example 1. We consider equation (1) (Model I) with the following data: $a = 1.8$, $t^* = 500$, $x \in [0,1]$, $c = 1000$, $\rho = 2680$, $\lambda = 240$, $u^\circ = 100$, $f(x) = 100x^2$, $q(t) = 0$. The unknown heat transfer coefficient depends on four parameters (which have to be reconstructed) in the following form

$$h(t) = \begin{cases} a_1, & t \in [0,100], \\ a_2, & t \in (100,200], \\ a_3, & t \in (200,350], \\ a_4, & t \in (350,500]. \end{cases}$$

The exact values of sought parameters a_1 , a_2 , a_3 and a_4 are equal to 2000, 1400, 800 and 250, respectively.

As a result of solving the direct problem for the exact heat transfer coefficient h , we obtain the values of temperature at the selected points in the grid of domain D . Then, from these values we select only those ones corresponding to the predetermined grid points (location of the thermocouple). These values simulate the temperature measurements. We call them the exact input data and denote by \hat{U}_{ik} . The grid used to generate these data was of size 200×1000 .

There is one measurement point $x_p = 0.18$ ($N_1 = 1$), the measurements from this point will be read every 1s and 2s ($N_2 = 501, 251$). In order to investigate the impact of measurement errors on the results of reconstruction and stability of the algorithm, the input data were perturbed by the pseudo-random error of sizes 1 and 2%.

In the process of reconstructing the boundary condition (minimizing the functional), the direct problem was solved many times. The grid used for this purpose was of size 150×500 and had different density than the grid used to generate the input data.

Minimum of functional (15) was searched by using the ACO algorithm. This algorithm is heuristic, therefore it is required to repeat calculations a certain number of times. In this paper, we assumed that the calculations for each case were repeated ten times.

Algorithm was adapted for parallel computations (multi-threaded calculations) which significantly reduced the computational time. In ACO algorithm, we set the following parameters

$nT = 4$, $M = 12$, $L = 8$, $I = 30$, $a_1 \in [1800, 2300]$, $a_2 \in [1200, 1700]$, $a_3 \in [500, 1000]$, $a_4 \in [100, 500]$.

Thus, the number of minimized function calls was equal to 368.

Table 1 presents the results of determining a_1, a_2, a_3, a_4 in dependence on the size of input data disturbance

Table 1

Results of calculations in case of measurements at every 1s, 2s (\bar{a}_i – restored value of a_i , δ_{a_i} – percentage relative error of a_i , σ – standard deviation ($i = 1, 2, 3, 4$))

Noise	\bar{a}_i	δ_{a_i} [%]	σ	\bar{a}_i	δ_{a_i} [%]	σ
	every 1s			every 2s		
0%	2003.22	0.17	7.78	2002.89	0.15	6.15
	1399.18	0.06	8.32	1400.90	0.07	7.33
	796.60	0.43	3.06	796.73	0.41	3.18
	245.11	1.96	3.34	245.35	1.86	2.34
1%	2002.85	0.15	5.21	2003.34	0.17	2.41
	1398.24	0.13	3.93	1398.39	0.12	7.04
	797.16	0.36	1.56	795.59	0.56	2.81
	246.41	1.44	1.57	246.42	1.44	0.88
2%	2004.94	0.25	2.13	2002.50	0.13	7.79
	1394.81	0.38	2.72	1401.83	0.14	4.62
	800.01	0.02	3.18	792.80	0.90	2.02
	244.44	2.23	3.72	245.08	1.97	3.52

at the measurement point $x_p = 0.18$ for measurements taken at every 1s and 2s. Generally, the obtained results are quite good. Except the error of parameter a_4 restoration, the other errors do not exceed the input data errors. Maximal error of parameter a_4 reconstruction is equal to 2.23%, in case of the other parameters the errors do not exceed 0.91%.

One of the main indicators of evaluating the obtained results is the error of reconstructing the temperature at the measurement point $x_p = 0.18$. Table 2 presents the errors of this reconstruction in case of measurements at every 1 and 2s. The average errors of temperature reconstructed in the measurement point are small. The relative average errors do not exceed 0.1% in each case. If the input errors are higher, then the average relative errors of temperature reconstruction slightly increase. These differences are however minimal.

Table 2

Errors of temperature reconstruction in measurement point $x_p = 0.18$ for measurements at every 1, 2s (Δ_{avg} – average absolute error, Δ_{max} – maximal absolute error, δ_{avg} – average relative error, δ_{max} – maximal relative error)

Noise	0%	1%	2%	0%	1%	2%
	every 1s			every 2s		
Δ_{avg} [K]	0.0070	0.0113	0.0175	0.0097	0.0116	0.0278
Δ_{max} [K]	0.0846	0.0846	0.0850	0.0846	0.0847	0.0845
δ_{avg} [%]	0.0424	0.0546	0.0663	0.0487	0.0524	0.0977
δ_{max} [%]	2.3406	2.3384	2.3508	2.3386	2.3413	2.3363

Figure 1 shows the relative errors of reconstructing the heat transfer coefficient h for measurements at every 1, 2s. This error was calculated according to the formula

$$\delta_h = \frac{\|\hat{h}(t) - h(t)\|}{\|h(t)\|} \cdot 100 [\%],$$

where $\hat{h}(t)$, $h(t)$ describe the reconstructed and exact heat transfer coefficient, respectively, and $\|\cdot\|$ denotes the norm defined by the following formula

$$\|f(t)\| = \left(\int_0^{t^*} |f(t)|^2 \right)^{\frac{1}{2}}.$$

These errors are minimal and smaller than 0.42%. For measurements every one second, relative error of reconstruction heat conduction coefficient for exact input data is slightly greater than in case of 1% perturbed input data. The reason for this it could be probabilistic character of Real Ant Colony Optimization algorithm.

Figure 2 presents the distribution of errors of the temperature reconstruction in measurement point $x_p = 0.18$ in case of measurements at every 2s.

We can see that the error of temperature reconstruction depends on perturbations of the input data. The input errors are larger, the errors of temperature reconstruction increase. Generally, the temperature is reconstructed very well.

Figure 2

Relative errors of reconstructing the heat transfer coefficient for various perturbations of input data and for measurements at every 1s and 2s

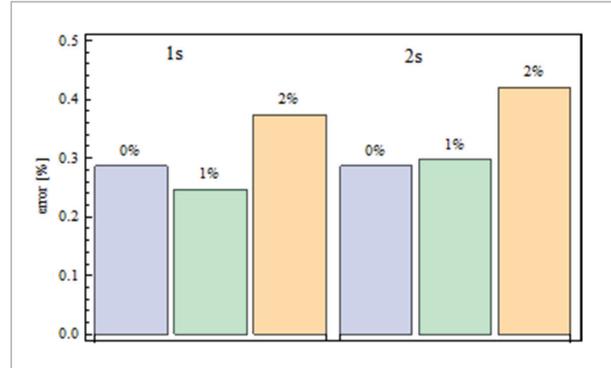
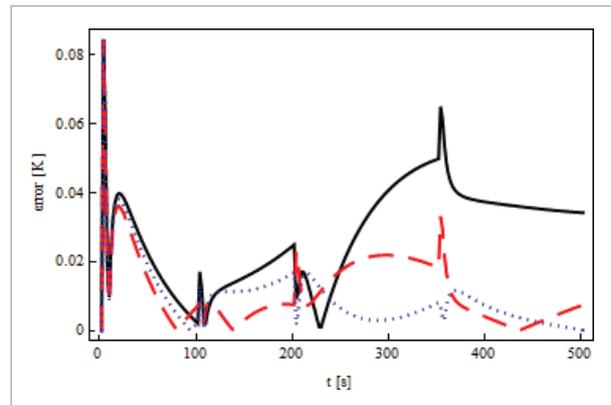


Figure 3

Distribution of errors of temperature reconstruction in measurement point $x_p = 0.18$ for measurements at every 2s and for various perturbations of input data (0% – dotted line, 1% – dashed line, 2% – solid line)



Example 2. In this example, we consider equation (6) (Model II) with the following data:

- $t^* = 100, x \in [0, 0.2],$
- $y \in [0, 0.2],$
- $c = 1000, \rho = 2680, a = 0.4,$
- $\lambda_1(x, y, t) = \lambda_2(x, y, t) = 240,$
- $u^x = 300, f(x, y) = 900,$
- $q_1(t, x) = q_2(t, x) = 0,$
- $g(x, y, t) = 0.$

Each of the unknown heat transfer coefficients depends on three parameters in the following form

$$h_1(t, y) = \begin{cases} b_1, & t \in [0, 30], \\ b_2, & t \in (30, 70], \\ b_3, & t \in (70, 100], \end{cases}$$

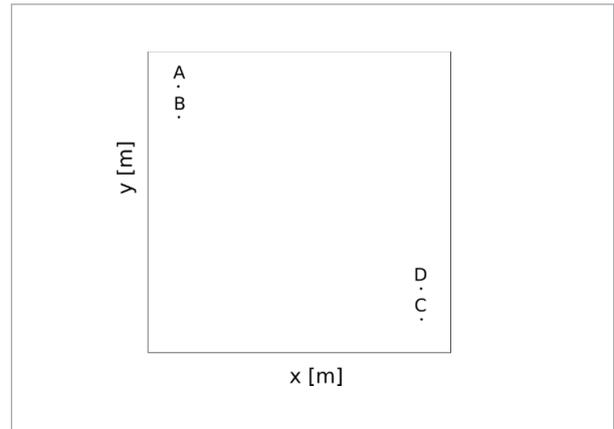
$$h_2(t, y) = \begin{cases} b_4, & t \in [0, 30], \\ b_5, & t \in (30, 70], \\ b_6, & t \in (70, 100]. \end{cases}$$

To restore the Robin boundary conditions, we need to find parameters b_1, b_2, \dots, b_6 . In this numerical experiment the exact values of parameters $b_i (i = 1, 2, \dots, 6)$ are equal to 1200, 800, 300, 900, 600, 150, respectively. In order to generate the input data, we used the grid of size $200 \times 200 \times 200$, but in process of solving the inverse problem we used the grid of size $100 \times 100 \times 100$. We obtained thus the measurements for four measurement points ($N_1 = 4$) with the following spatial coordinates : $A(0.004, 0.19)$, $B(0.004, 0.17)$, $C(0.196, 0.01)$, $D(0.196, 0.03)$. Distribution of the measurement points is presented in Figure 4.

The parameters of the algorithm are as follows:
 $M = 16, L = 8, I = 30, nT = 8,$

Figure 4

Distribution of the measurement points



$b_1 \in [1000, 1400], b_2 \in [600, 1000], b_3 \in [100, 500],$
 $b_4 \in [700, 1100], b_5 \in [400, 800], b_6 \in [50, 350].$

Table 3 presents the results of coefficients b_i reconstruction. If the input data are not affected by the pseudorandom error, the obtained results are good.

Table 3

Results of calculation in case of measurements at every 1s, 2s (\bar{b}_i – restored value of b_i , $\delta_{\bar{b}_i}$ – percentage relative error of b_i , σ – standard deviation ($i = 1, 2, 3, 4, 5, 6$))

Noise	\bar{a}_i	$\delta_{\bar{a}_i} [\%]$	σ	\bar{a}_i	$\delta_{\bar{a}_i} [\%]$	σ	\bar{a}_i	$\delta_{\bar{a}_i} [\%]$	σ
	every 1s			every 2s			every 4s		
0%	1203.46	0.29	2.82	1202.18	0.19	2.58	1201.77	0.15	4.78
	795.37	0.58	3.12	797.13	0.36	2.94	792.84	0.90	3.59
	300.48	0.16	3.03	293.33	2.23	4.16	299.59	0.14	3.41
	904.04	0.45	2.62	903.58	0.40	3.04	899.67	0.04	2.39
	597.71	0.38	2.97	599.43	0.10	2.61	597.77	0.38	3.73
	149.21	0.53	2.66	148.12	1.26	3.17	147.58	1.62	3.38
1%	1199.52	0.04	4.51	1187.69	1.03	3.41	1225.32	2.12	2.62
	796.77	0.41	3.10	811.48	1.44	2.03	798.24	0.22	3.63
	293.47	2.18	7.12	291.51	2.83	3.29	282.09	5.97	3.56
	903.13	0.35	4.82	900.55	0.07	3.59	917.64	1.97	5.79
	605.68	0.95	4.43	609.84	1.65	2.87	587.49	2.09	3.39
	150.31	0.21	5.78	160.77	7.19	2.55	153.12	2.09	4.79
2%	1182.44	1.47	3.26	1259.78	4.99	3.03	1186.09	1.16	2.35
	813.03	1.63	3.03	757.22	5.35	2.75	792.29	0.97	2.31
	302.47	0.82	3.11	295.87	1.38	8.74	326.24	8.75	4.99
	946.56	5.18	2.79	913.90	1.55	4.83	918.73	2.09	4.07
	624.32	4.06	1.94	613.14	2.20	5.76	555.50	7.42	5.17
	114.66	23.56	2.72	143.02	4.66	5.74	186.08	24.06	4.56

Most of the coefficients are restored with the errors not exceeding 0.6%. The only exceptions are in case of coefficients b_3, b_6 for the measurements taken at every 2s, and in case of coefficients b_2, b_6 for measurements taken at every 4s. However these errors are still acceptable. For the input data disturbed by 1% error, the best results were obtained for measurements read at every 1s. In this case, only the error of parameter b_3 reconstruction is greater than the error of input data. Considering the case of 2% input data error, we can see the different levels of reconstruction errors for various coefficients b_i . For example, the error of coef-

ficient b_3 reconstruction for measurements at every 1s is 0.82%, while for the measurements at every 2 and 4s these errors are, respectively, 1.38% and 8.75%.

Table 4 shows the errors of temperature reconstruction in the measurement points. In each case, the relative error does not exceed 0.33%. With the increase of disturbance input data, the errors of reconstruction temperature grow. For example, for the measurements at every 2s and the input data error of 0%, 1%, 2%, the maximal relative error of temperature restoration are equal to 0.1931%, 0.2182%, 0.3238%, respectively.

Table 4

Errors of temperature reconstruction in measurement points *A,B,C,D* for measurements at every 1, 2, 4s (Δ_{avg} – average absolute error, Δ_{max} – maximal absolute error, δ_{avg} – average relative error, δ_{max} – maximal relative error)

Noise	0%	1%	2%	0%	1%	2%	0%	1%	2%
	every 1s			every 2s			every 4s		
Δ_{avg} [K]	0.1168	0.1869	0.8028	0.1262	0.3920	0.8036	0.1284	0.4234	0.7543
Δ_{max} [K]	1.7559	1.8189	2.0914	1.7764	2.0077	3.0930	1.7830	1.5291	2.0332
δ_{avg} [%]	0.0128	0.0208	0.0899	0.0139	0.0431	0.0869	0.0140	0.0462	0.0839
δ_{max} [%]	0.1909	0.1977	0.2273	0.1931	0.2182	0.3238	0.1938	0.1613	0.2210

In Figures 5 and 6, we can see the errors of temperature reconstruction in measurement points A and D. In case of points B and C these errors are almost at the same level as in case of points A and D, respectively.

This is due to the fact that points A and B and points C and D are close to each other. The largest errors in case of point A are observed in the initial moment of time, and in case of point D about the time $t = 30$.

Figure 5

Distribution of errors of temperature reconstruction in measurement point A for measurements at every 1s and for various perturbations of input data (0% – blue dots, 1% – red dots, 2% – green dots)

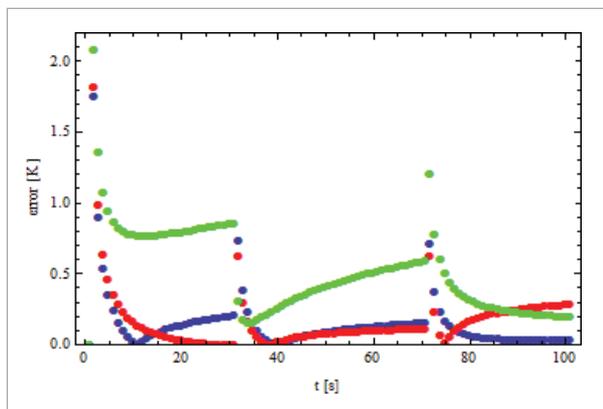
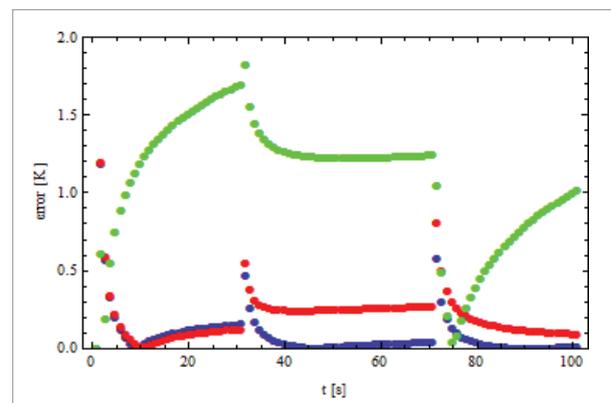


Figure 6

Distribution of errors of temperature reconstruction in measurement point D for measurements at every 1s and for various perturbations of input data (0% – blue dots, 1% – red dots, 2% – green dots)



Conclusions

In this paper the Real Ant Colony Optimization algorithm was used to solve the heat conduction inverse problem of fractional order. We considered two models – 1D space fractional heat conduction equation and 2D time fractional heat conduction equation. The heat transfer coefficient occurring in the Robin boundary condition was restored. In order to reconstruct it, the functional defining the error of approximate solution was minimized. The direct problem was solved using the finite difference method.

The obtained results are very good. In Example 1, the errors of heat transfer coefficient reconstruction in each considered case are less than 0.42 % and do not exceed the input data errors. More importantly, the

errors of temperature reconstruction at the measurement point are minimal, the average relative errors are smaller than 0.1 %.

In case of Example 2, the obtained results are also good. Although some of the parameters b_i are reconstructed with quite large error (e.g. b_6 for 2% input data), the error of temperature reconstruction are still small. In each considered case, the average relative error is less than 0.09%

It is worth mentioning that the used algorithm can be easily adapted to parallel computing which allows to reduce significantly the computation time.

In Example 1, by executing the algorithm for 4 threads the computations were performed nearly 3.8 times faster than without the multi-threaded approach.

References

1. Ardakani, M., Khodadad, M. Identification of thermal conductivity and the shape of an inclusion using the boundary elements method and the particle swarm optimization algorithm. *Inverse Problems in Science and Engineering*, 2009, 17, 855-870. <https://doi.org/10.1080/17415970902884136>
2. Battaglia, J. L., Cois, O., Puigsegur, L., Oustaloup, A. Solving an inverse heat conduction problem using a non-integer identified model. *International Journal of Heat and Mass Transfer*, 2001, 44, 2671-2680. [https://doi.org/10.1016/S0017-9310\(00\)00310-0](https://doi.org/10.1016/S0017-9310(00)00310-0)
3. Brociek, R. Implicit finite difference method for time fractional diffusion equations with mixed boundary conditions. *Zesz. Nauk. PŚ., Mat. Stosow*, 2014, 4, 73-87.
4. Brociek, R., Słota, D. Reconstruction of the boundary condition for the heat conduction equation of fractional order. *Thermal Science*, 2015, 19, 35-42. <https://doi.org/10.2298/TSCI15S1S35B>
5. Brociek R., Słota D., Application and comparison of intelligent algorithms to solve the fractional heat conduction inverse problem. *Information Technology and Control*, 2016, 45, 184-194. <https://doi.org/10.5755/j01.itc.45.2.13716>
6. Brociek, R., Słota, D., Wituła, R. Reconstruction of the thermal conductivity coefficient in the time fractional diffusion equation. In: K. Latawiec, M. Łukaniszyn, R. Stanisławski (Eds.), *Advances in Modelling and Control of Non-integer-Order Systems*, Vol. 320 of *Lecture Notes in Electrical Engineering*. Springer, 2015, 239-247. https://doi.org/10.1007/978-3-319-09900-2_22
7. Brociek, R., Słota, D. Implicit finite difference method for the space fractional heat conduction equation with the mixed boundary conditions. *Silesian Journal of Pure and Applied Mathematics*, 2016, 6, 125-136.
8. Caponetto, R., Dongola, G., Fortuna, I., Petras, I. *Fractional Order Systems: Modeling and Control Applications*. World Scientific Series on Nonlinear Science, Series A, World Scientific Publishing, 2010, 72. <https://doi.org/10.1142/7709>
9. Carpinteri, A., Mainardi, F. *Fractal and Fractional Calculus in Continuum Mechanics*. Springer, New York, 1997. <https://doi.org/10.1007/978-3-7091-2664-6>
10. Carvalho, A. R., Velho, H. F. D. C., Stephany, S., Souto, R. P., Sandri, J. C. B. S. Fuzzy ant colony optimization for estimating chlorophyll concentration profile in offshore sea water. *Inverse Problems in Science and Engineering*, 2008, 16, 705-715. <https://doi.org/10.1080/17415970802083276>
11. Das, S. *Functional Fractional Calculus for System Identification and Controls*. Springer, Berlin, 2008.
12. Dorigo, M., Stützle, T. *Ant Colony Optimization*. MIT Press, Cambridge, 2004.
13. Hetmaniok, E., Nowak, I., Słota, D., Zielonka, A. Determination of optimal parameters for the immune algorithm used for solving inverse heat conduction prob-

- lems with and without a phase change. *Numerical Heat Transfer, Part B*, 2012, 62(6), 462-478. <https://doi.org/10.1080/10407790.2013.730906>
14. Hetmaniok, E., Słota, D., Zielonka, A. Experimental verification of selected artificial intelligence algorithms used for solving the inverse Stefan problem. *Numerical Heat Transfer, Part B*, 2014, 66, 343-359. <https://doi.org/10.1080/10407790.2014.915680>
 15. Hetmaniok, E., Słota, D., Zielonka, A., Using the swarm intelligence algorithms in solution of the two-dimensional inverse Stefan problem. *Computers and Mathematics with Applications*, 2015, 69, 347-361. <https://doi.org/10.1016/j.camwa.2014.12.013>
 16. Karaboga, D., Basturk, B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 2007, 39, 459-471. <https://doi.org/10.1007/s10898-007-9149-x>
 17. Karaboga, D., Basturk, B. On the performance of artificial bee colony (ABC) algorithm. *Applied Soft Computing*, 2008, 8, 687-697. <https://doi.org/10.1016/j.asoc.2007.05.007>
 18. Karaboga, D., Akay, B. A comparative study of artificial bee colony algorithm. *Applied Mathematics and Computation*, 2009, 214, 108-132. <https://doi.org/10.1016/j.amc.2009.03.090>
 19. Klafter, J., Lim, S., Metzler, R. *Fractional dynamics. Recent advances*, World Scientific, New Jersey, 2012.
 20. Liu, J., Yamamoto, M. A backward problem for the time-fractional diffusion equation. *Journal of Applied Analysis*, 2010, 89, 1769-1788. <https://doi.org/10.1080/00036810903479731>
 21. Martišius, I., Birvinskas, D., Damaševičius, R., Jusas, V. EEG dataset reduction and classification using wave atom transform. *Lecture Notes in Computer Science – ICANN'2013*, 2013, 8131, 208-215.
 22. Meerschaert, M. M., Tadjeran, C. Finite difference approximations for fractional advection-dispersion flow equations. *Journal of Computational and Applied Mathematics*, 2006, 172, 65-77. <https://doi.org/10.1016/j.cam.2004.01.033>
 23. Meerschaert, M. M., Scheffler, H. P., Tadjeran, C. Finite difference method for two-dimensional fractional dispersion equation. *Journal of Computational Physics*, 2006, 211, 249-261. <https://doi.org/10.1016/j.jcp.2005.05.017>
 24. Miller, L., Yamamoto, M. Coefficient inverse problem for a fractional diffusion equation, *Inverse Problems*, 2013, 29(7), 075013, (8pp).
 25. Mitkowski, W., Kacprzyk, J., Baranowski, J. *Advances in the Theory and Applications of Non-integer Order Systems*. Springer Inter. Publ., Cham, 2013. <https://doi.org/10.1007/978-3-319-00933-9>
 26. Mitkowski, W., Skrucz, P. Fractional-order models of the supercapacitors in the form of RC ladder networks. *Bulletin of the Polish Academy of Sciences Technical Sciences*, 2013, 61, 581-587. <https://doi.org/10.2478/bpasts-2013-0059>
 27. Murio, D. Time fractional IHCP with Caputo fractional derivatives. *Computers and Mathematics with Applications*, 2008, 56, 2371-2381. <https://doi.org/10.1016/j.camwa.2008.05.015>
 28. Murio, D. Stable numerical solution of a fractional-diffusion inverse heat conduction problem. *Computers and Mathematics with Applications*, 2007, 53, 1492-1501. <https://doi.org/10.1016/j.camwa.2006.05.027>
 29. Murio, D. Implicit finite difference approximation for time fractional diffusion equations. *Computers and Mathematics with Applications*, 2008, 56, 1138-1145. <https://doi.org/10.1016/j.camwa.2008.02.015>
 30. Obrączka, A., Kowalski, J. Modeling the distribution of heat in the ceramic materials using fractional differential equations. In: M. Szczygieł (Ed.), *Materiały XV Jubileuszowego Sympozjum «Podstawowe Problemy Energoelektroniki, Elektromechaniki i Mechatroniki» PPEEm, 2012, Archiwum Konferencji PTETiS, Komitet Organizacyjny Sympozjum PPEE i Seminarium BSE, 2012, 32, 133-132 (in Polish).*
 31. Obrączka, A., Mitkowski, W. The comparison of parameter identification methods for fractional partial differential equation. *Solid State Phenomena*, 2014, 210, 265-270. <https://doi.org/10.4028/www.scientific.net/SSP.210.265>
 32. Oprzędkiewicz, K. Approximation method for a fractional order transfer function with zero and pole. *Archives of Control Sciences*, 2014, 24, 409-425.
 33. Özbakir, L., Baykasoglu, A., Tapkan, P. Bees algorithm for generalized assignment problem. *Applied Mathematics and Computation*, 2010, 215, 3782-3795. <https://doi.org/10.1016/j.amc.2009.11.018>
 34. Podlubny, I. *Fractional Differential Equations*. Academic Press, San Diego, 1999.
 35. Santos, A., Campos, V. H., Luz, E., Freitas, S., Grell, G., Gan, M. Firefly optimization to determine the precipitation field on South America. *Inverse Problems in Science and Engineering*, 2013, 21, 451-466. <https://doi.org/10.1080/17415977.2012.712531>
 36. Socha, K., Dorigo, M. Ant Colony Optimization in continuous domains. *European Journal of Operational Research*

- search, 2008, 185, 1155-1173. <https://doi.org/10.1016/j.ejor.2006.06.046>
37. Toksari, M. D. Ant colony optimization for finding the global minimum. *Applied Mathematics and Computing*, 2006, 176, 308-316. <https://doi.org/10.1016/j.amc.2005.09.043>
 38. Wei, T., Zhang, Z. Q. Reconstruction of time-dependent source term in time-fractional diffusion equation. *Engineering Analysis with Boundary Elements*, 2013, 37, 23-31. <https://doi.org/10.1016/j.enganabound.2012.08.003>
 39. Wang, J. G., Zhou, Y. B., Wei, T. A posteriori regularization parameter choice rule for the quasi-boundary value method for the backward time fractional diffusion problem. *Applied Mathematics Letters*, 2013, 26, 741-747. <https://doi.org/10.1016/j.aml.2013.02.006>
 40. Woźniak, M. Novel image correction method based on swarm intelligence approach. *Communications in Computer and Information Science*, Springer Publishing International, Switzerland, 2016, 639, 404-413. https://doi.org/10.1007/978-3-319-46254-7_32
 41. Woźniak, M., Połap, D. Flexible neural network architecture for handwritten signatures recognition. *International Journal of Electronics and Telecommunications*, De Gruyter Open Ltd., 2016, 62(2), 197-202. <https://doi.org/10.1515/ejtel-2016-0027>
 42. Woźniak, M., Połap, D., Napoli, C., Tramontana, E. Graphic object feature extraction system based on cuckoo search algorithm. *Expert Systems with Applications*, Elsevier, 2016, 66, 20-31. <https://doi.org/10.1016/j.eswa.2016.08.068>
 43. Woźniak, M., Damaševičius, R., Vasiljevas, M., Šalkevičius J. Human Activity Recognition in AAL Environments Using Random Projections. *Computational and Mathematical Methods in Medicine*, Hindawi Publishing Corporation, USA, 2016, Vol. 2016, 4073584:1-4073584:17. <https://doi.org/10.1155/2016/4073584>
 44. Yan, L., Yang, F. Efficient Kansa-type MFS algorithm for time-fractional inverse diffusion problems. *Computers and Mathematics with Applications*, 2014, 67, 1507-1520. <https://doi.org/10.1016/j.camwa.2014.02.008>
 45. Zheng, G., Wei, T. A new regularization method for the time fractional inverse advection-dispersion problem. *SIAM Journal on Numerical Analysis*, 2011, 49, 1972-1990. <https://doi.org/10.1137/100783042>
 46. Zhuang, Q., Yu, B., Jiang, X. An inverse problem of parameter estimation for time-fractional heat conduction in a composite medium using carbon-carbon experimental data. *Physica B*, 2015, 456, 9-15. <https://doi.org/10.1016/j.physb.2014.08.011>

Summary / Santrauka

This paper describes the method of solution of the space fractional and 2D time fractional heat conduction inverse problem. In this paper the authors consider two models – 1D space fractional heat conduction equation and 2D time fractional heat conduction equation with the initial-boundary conditions. To solve the inverse heat conduction problem, a functional defining the error of approximate solution must be minimized. To minimize this functional the Real Ant Colony Optimization (ACO) algorithm was used. In order to reduce the computational time, the calculations were performed in a parallel (multi-threaded) way. The paper presents examples to illustrate the accuracy and stability of the presented algorithm.

Straipsnyje aprašomas metodas, kurį taikant sprendžiama erdvės frakcinės ir 2D laiko frakcinės karščio laidumo inversijos problema. Autoriai aptaria du modelius – 1D erdvės frakcinę karščio laidumo lygtį ir 2D laiko frakcinę karščio laidumo lygtį su pradinėmis ribinėmis sąlygomis. Norint išspręsti inversinę karščio laidumo problemą, turi būti minimizuota funkcionalo apibrėžties apytikrio sprendimo klaida. Šiam funkcionalui minimizuoti naudojamas skruzdžių kolonijos optimizavimo algoritmas (angl. *Real Ant Colony Optimization (ACO)*). Tam, kad būtų sumažintas skaičiavimų laikas, jie atlikti lygiagrečiu (daugiagiju) būdu. Straipsnyje pateikta siūlomo algoritmo tikslumo ir stabilumo pavyzdžių.