Журнал обчислювальної та прикладної математики | 2022, № 2 | Journal of Numerical & Applied Mathematics

УДК 519.6
MSC 65Y05, 35R11

MULTITHREADING PERFORMANCE SIMULATING FRACTIONAL-ORDER MOISTURE TRANSPORT ON AMD EPYC

V. O. Bohaienko, A. V. Gladky
V. M. Glushkov Institute of Cybernetics of NAS of Ukraine, Kyiv, Ukraine,
E-mail: sevab@ukr.net, gladky@ukr.net

ЕФЕКТИВНІСТЬ БАГАТОПОТОЧНОГО РОЗПАРАЛЕЛЮВАННЯ ПРИ МОДЕЛЮВАННІ ВОЛОГОПЕРЕНЕСЕННЯ НА ОСНОВІ ДРОБОВО-ДИФЕРЕНЦІАЛЬНОЇ МОДЕЛІ НА AMD EPYC

В. О. Богаєнко, А. В. Гладкий
Інститут кібернетики ім. В. М. Глушкова НАН України, Київ, Україна,
E-mail: sevab@ukr.net, gladky@ukr.net

Abstract. The paper studies the performance of multithreaded parallel implementation of a finite-difference solver for a two-dimensional space-fractional generalization of Richards equation. For numerical solution we used implicit Crank-Nicholson scheme with L1-approximation of Caputo fractional derivative and TFQMR linear systems' solver. OpenMP implementation was tested on three CPUs — server Intel Xeon Bronze 3104 and AMD EPYC 7542 along with laptop AMD Ryzen 3 5300U. Testing results show that the proposed implementation can give close-to-linear acceleration when executing on up to 8 cores. On high-performance AMD EPYC maximal acceleration was achieved when 32-64 cores were used showing limited scalability of the algorithms on such a CPU.
Keywords: moisture transport, fractional derivatives, multithreading, irrigation.

Анотація. У статті досліджено продуктивність багатопоточної паралельної реалізації скінченно-різницевого алгоритму для розв'язання двовимірного дробово-диференціального за просторовою змінною узагальнення рівняння Річардса. Для чисельного розв'язання використано неявну схему Кранка-Ніколсона з L1-апроксимацією дробової похідної Капуто, а також алгоритм TFQMR для розв'язання систем лінійних алгебраїчних рівнянь. OpenMP-реалізація була протестована на трьох процесорах — серверних Intel Xeon Bronze 3104 та AMD EPYC 7542, а також на ноутбуковому AMD Ryzen 3 5300U. Результати тестування показують, що запропонована реалізація може надавати близьке до лінійного прискорення при виконанні на 8 чи менше процесорних ядрах. На високопродуктивних AMD EPYC максимальне прискорення було

досягнуто при використанні 32-64 ядер, що демонструє обмежену масштабованість алгоритмів на такому CPU.
Ключові слова: вологоперенесення, дробові похідні, багатопотоковість, зрошення.

1. Introduction

Simulation of moisture transport has numerous application in hydrology and agriculture. While in most cases practitioners apply models based on the classic Richards' differential equation [1] and corresponding software (e.g. HYDRUS) there are two fields where additional developments are needed from the standpoint of more accurate and fast simulation.

At first, let us mention that under certain conditions soils can be considered as media of fractal structure [2]. Mass and heat transport processes in such media are non-local and are often modelled using fractional differential equations [3]. Among the literature related to the fractional-differential generalization of Richards equation, we can refer to the paper [4] in which time-fractional equations with Caputo derivative [3] are studied.

The second field is decision support in the design of drip irrigation systems, especially subsurface ones, for which economic efficiency of irrigation depends on the quality of the choice of systems' parameters. A widely used class of algorithms for their automated selection consists of optimization algorithms superposed on moisture transport models with objective functions based on economic assessment of irrigation efficiency. Often (see, e.g. [5]) optimization is performed by time-consuming meta-heuristic methods making urgent the development of algorithms and the usage of computational systems aimed at acceleration of moisture transport simulation. The usage of fractional-differential models, numerical algorithms for which have higher order of computational complexity compared to the ones for integer-order models, additionally increase the need for such developments. While multithreaded implementations of classical Richards equation solvers are widely studied (see, e.g. [6]), similar studies for their fractional-order counterparts are unknown to the authors.

In this context, the paper is a continuation of previous authors' results on high-performance computing in, particularly fractional-order, modelling of mass and moisture transport [7, 8, 9] studying the performance of multithreaded parallel implementation of finite-difference solver for two-dimensional space-fractional generalization of Richards equation.

2. Mathematical model and numerical method

The space-fractional generalization of Richards equation stated in terms of water heads and derived similarly to the one-dimensional equation in [7] has the form

$$\begin{array}{c} C(h)\frac{\partial h}{\partial t} = D_x^{\alpha}(k_x(H)\frac{\partial H}{\partial x}) + D_z^{\beta}(k_z(H)\frac{\partial}{\partial z}) - S, \\ 0 \le x \le L_x,\ 0 \le z \le L_z,\ t \ge 0, 0 < \alpha < 1, \end{array} \quad (1)$$

$$
\begin{aligned}
D_x^\alpha H &= \tfrac{1}{2}(D_{x,l}^\alpha H + D_{x,r}^\alpha H),\\
D_{x,l}^\alpha H &= \tfrac{1}{\Gamma(1-\alpha)} \int\limits_0^x \tfrac{\partial H}{\partial x}(x-\xi)^{-\alpha} d\xi,\\
D_{x,r}^\alpha H &= \tfrac{1}{\Gamma(1-\alpha)} \int\limits_x^{L_x} \tfrac{\partial H}{\partial x}(\xi-x)^{-\alpha} d\xi,
\end{aligned}
\tag{2}
$$

where D_x^α is the Caputo fractional derivative with respect to the variable x (derivative with respect to z is denoted and defined similarly), $h(x,z,t) = \frac{P(x,z,t)}{\rho g}$ is the water head, m, $H(x,z,t) = \frac{P(x,z,t)}{\rho g} + z$ is the full moisture potential, m, $P(x,z,t)$ is the suction pressure, Pa, ρ is the density of water, kg/m^3, g is the acceleration of gravity, m/s^2, $C(h) = \frac{\partial\theta}{\partial h}$ is the differential soil moisture content, $\%/m$, $\theta(x,z,t)$ is the volumetric soil moisture content, %, $k_x(H)$, $k_z(H)$ are the hydraulic conductivities in the fractal dimensions m^α/s and m^β/s, $S(x,z,t)$ is the source function, $\%/s$, that simulates the extraction of moisture by plants' roots and its supply by subsurface irrigation.

Following [10] we assume that

$$k_x(H) = \sigma_x^{\alpha-1} k(H), \quad k_z(H) = \sigma_z^{\alpha-1} k(H),$$

where σ_x, σ_z are fractal dimention constants, m, and $k(H)$ is the hydraulic conductivity in non-fractal medium, m/s. For simplicity, as only performance aspects are considered, we further set

$$\sigma_x = \sigma_z = 1.$$

Boundary and initial conditions for Equation 1, the form of the source function and the configuration of solution domain are the same as the ones for the integer-order model described in [9].

Water retention curve of soil is described by van Genuchten's model [11] in the form

$$\theta(h) = \theta_0 + \frac{\theta_1 - \theta_0}{\left[1 + (10\alpha\,|h|)^n\right]^{1-1/n}}$$

while the dependency between hydraulic conductivity and moisture potential is represented according to Averyanov's model [12] in the form

$$k(H) = k_f \left(\frac{\theta(H-z) - \theta_0}{\theta_1 - \theta_0} \right)^\beta,$$

where k_f is the filtration coefficient, β is the fixed exponent.

The model (1), (2) is used to simulate the process of subsurface drip irrigation. We model the start of watering at the moment of time when the average moisture content of a root zone becomes less than the specified value. Watering continues until the average moisture content reaches some upper limit usually equal to field capacity. Moisture content is averaged according to the function of root system density with higher weight coefficients for the areas where root system is denser.

The used approach is described in more detail in [7, 9].

The numerical solution of the initial-boundary value problem for the model based on Equations (1), (2) is performed according to the implicit finite-difference

Crank-Nicholson scheme [13] on a uniform grid

$$\omega = \left\{ \ (x_i = ih_x, z_k = kh_z, t_j = j\tau): \ i = \overline{0,m}, \ k = \overline{0,n}, j = 0,1,2,... \right\}, \qquad (3)$$

where $h_x = L_x/m, h_z = L_z/n$ are the steps with respect to the space variables, τ is the step with respect to the time variable. Here and further, grid analogue of the sought-for function h and, similarly, the other functions is denoted as $h_{ik}^j = h(x_i, z_k, t_j)$.

On the grid (3) using the solution on the previous time step for non-local part of the derivative (2) we obtain the following 5-diagonal linear equations system given here without obvious discretizations of boundary conditions:

$$h_{i-1,k}^j A_{1,i,k}^{j-1} + h_{i,k-1}^j A_{2,i,k}^{j-1} + h_{i+1,k}^j B_{1,i,k}^{j-1} + h_{i,k+1}^j B_{2,i,k}^{j-1} - h_{i,k}^j \cdot R_{i,k}^{j-1} = \Phi_{i,k}^{j-1},$$

where

$$A_{1,i,k}^{j-1} = \frac{1}{4} D_x (k(H_{i-1,k}^{j-1}) + k(H_{i,k}^{j-1})), \ A_{2,i,k}^{j-1} = \frac{1}{4} D_z (k(H_{i,k-1}^{j-1}) + k(H_{i,k}^{j-1})),$$

$$B_{1,i,k}^{j-1} = \frac{1}{4} D_x (k(H_{i+1,k}^{j-1}) + k(H_{i,k}^{j-1})), \ B_{2,i,k}^{j-1} = \frac{1}{4} D_z (k(H_{i,k+1}^{j-1}) + k(H_{i,k}^{j-1})),$$

$$D_x = \frac{h_x^{-1-\alpha}}{\Gamma(2-\alpha)}, \ D_z = \frac{h_z^{-1-\beta}}{\Gamma(2-\beta)},$$

$$R_{i,k}^{j-1} = A_{1,i,k}^{j-1} + A_{2,i,k}^{j-1} + B_{1,i,k}^{j-1} + B_{2,i,k}^{j-1} + \frac{C(h_{i,k}^{j-1})}{\tau},$$

$$\Phi_{i,k}^{j-1} = -h_{i-1,k}^{j-1} A_{1,i,k}^{j-1} - h_{i,k-1}^{j-1} A_{2,i,k}^{j-1} - h_{i+1,k}^{j-1} B_{1,i,k}^{j-1} - h_{i,k+1}^{j-1} B_{2,i,k}^{j-1} +$$

$$+ (A_{1,i,k}^{j-1} + A_{2,i,k}^{j-1} + B_{1,i,k}^{j-1} + B_{2,i,k}^{j-1} - \frac{C(h_{i,k}^{j-1})}{\tau}) h_{i,k}^{j-1} - S_{i,k}^j -$$

$$- \Delta_{i,k}^{j-1} + D_x h_x (k(H_{i,k+1}^{j-1}) - k(H_{i,k}^{j-1})).$$

Here $\Delta_{i,k}^{j-1}$ describes the 'non-local' part of the fractional derivative and has the form

$$\Delta_{i,k}^{j-1} = \frac{1}{2} \sum_{l=1, l \neq i}^{m-1} \left(|i-l+1|^{1-\alpha} - |i-l|^{1-\alpha} \right) h_{xx,l,k}^{j-1} +$$

$$+ \frac{1}{2} \sum_{l=1, l \neq k}^{n-1} \left(|k-l+1|^{1-\beta} - |k-l|^{1-\beta} \right) h_{zz,i,l}^{j-1},$$

$$h_{xx,i,k}^{j-1} = 2D_x \left(h_{i-1,k}^{j-1} A_{1,i,k}^{j-1} + h_{i+1,k}^{j-1} B_{1,i,k}^{j-1} - h_{i,k}^{j-1} (A_{1,i,k}^{j-1} + B_{1,i,k}^{j-1}) \right),$$

$$h_{zz,i,k}^{j-1} = 2D_z (h_{i,k-1}^{j-1} A_{2,i,k}^{j-1} + h_{i,k+1}^{j-1} B_{2,i,k}^{j-1} - h_{i,k}^{j-1} (A_{2,i,k}^{j-1} + B_{2,i,k}^{j-1}) +$$

$$+ 0.5 h_x (k(H_{i,k+1}^{j-1}) - k(H_{i,k}^{j-1})).$$

The corresponding linear systems are solved by the TFQMR algorithm [14].

The initial time step length is taken to be equal to 1 s and changes during the simulation based on the hypothesis about the correlation between the time step and the condition number of a matrix along with the correlation of the condition number and the number of iterations of the solution algorithm — the step is multiplied on a given value (here equal to 1.25) when the number of iterations of the TFQMR algorithm exceeds a given maximum value (here 20).

The solution at the appropriate time step is then repeated. If the number of iterations is less when 1/3 of the maximum value, the length of the next time step increases the same way.

Let us also mention that the presented scheme's accuracy and convergence is not studied here as these matters have no impact on the main topic — the performance of the scheme's multithreaded implementation.

3. Multithreaded implementation

The solution procedure on one time step was implemented in multithreaded environment using OpenMP. All computations are performed in a single *parallel* block. Within the TFQMR algorithm all summations are implemented in parallel using *reduction* directive.

Scalar variables are updated in the *single* environment. Before starting solver's iterations the values of

$$A_{1,i,k}^{j-1}, \quad A_{2,i,k}^{j-1}, \quad B_{1,i,k}^{j-1}, \quad B_{2,i,k}^{j-1}, \quad R_{i,k}^{j-1}, \quad \Phi_{i,k}^{j-1}$$

are calculated in parallel. The values of

$$H_{xx,l,k}^{j-1}, \quad H_{zz,i,l}^{j-1}$$

are calculated in parallel before the calculation of $\Delta_{i,k}^{j-1}$.

4. Performance testing

Input data described in [9] were used for computational experiments. A single-layered soil model with the filtration coefficient of 15 cm/day was considered. Simulation domain was 10 m wide and 1 m deep. At its lower and lateral boundaries the free-flow boundary conditions were set. Irrigation was applied in the simulation then average moisture content in 0.5 m-deep layer of soil that model the root zone fell lower than 95% of field capacity and lasted until it became higher than 100% of field capacity. Fixed evapotranspiration was set to be equal to 5.1 mm/day.

The first series of simulations were performed for $m = 1000$, $n = 100$ for time up to 1 day for two cases: $\alpha = \beta = 1$ that corresponds to the classical integer-order model and $\alpha = \beta = 0.98$. The modelled dynamics of average moisture content in the root zone for these two cases is shown in Fig. 1.

Total irrigation volume for 1 day period here is 6.9% less for the fractional-order model compared to the classical one with four waterings proposed by both models. Interval between waterings was larger for the fractional-order model while their duration is lower.

Total water content in the simulation domain was at the end of the modelled period higher for the case of the classical model that means less irrigation water outflow below the root zone when soil structure has even slightly pronounced fractal properties.

Computions were performed on a node of SCIT-5 cluster of VM Glushkov Institute of Cybernetics with AMD EPYC 7542 CPUs, on a node of SCIT-45

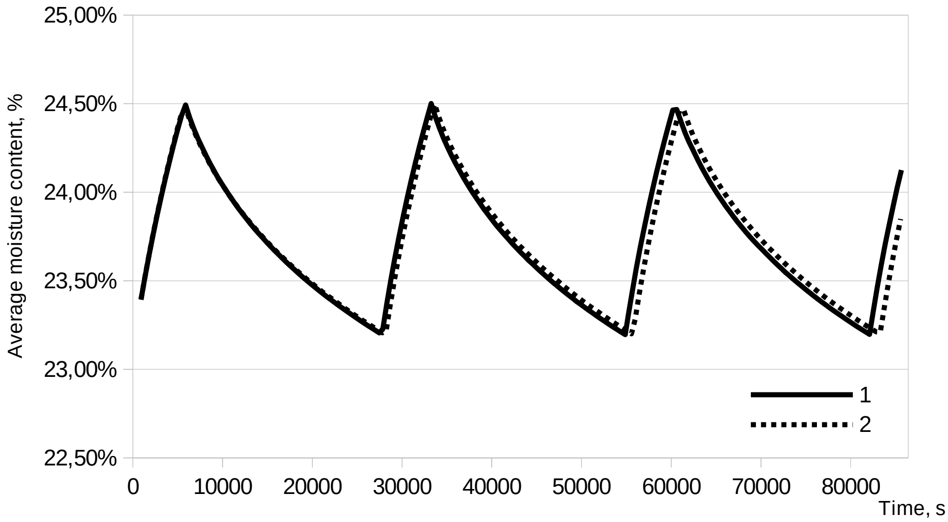

FIGURE 1. Dynamics of average moisture content: (1) — integer-order model, (2) — space-fractional model with $\alpha = \beta = 0.98$

TABLE 1. Execution time averaged among 5 runs, s

	$\alpha=\beta=1$			$\alpha=\beta=0.98$		
n threads	Ryzen	Xeon	EPYC	Ryzen	Xeon	EPYC
1	135	327	142,25	365	1105	484
2	121	179	81	239	573	256
4	89	154	67,25	156	341	180
8	79	115	66	127	223	125
16			62			117
32			47			94
64			66,5			81

cluster with Intel Xeon Bronze 3104 CPUs, and on a laptop with AMD Ryzen 3 5300U CPU. Execution times averaged among 5 runs are given in Tab. 1. In all cases each thread was executed in a separate core.

Here, differences in execution time in a single-threaded mode are roughly similar to single thread performance of the used CPUs reported on

www.cpubenchmark.net.

Fractional-order modelling was 2.7-3.4-times slower in a single-threaded mode. With the increase of thread number this difference decreased because of higher speed-up when fractional-order model was used.

Higher acceleration of computations in this case is due to lower ratio of execution time between parallely executable blocks and blocks that are executed

TABLE 2. Maximum acceleration and overall relative errors subject to the grid size parameter n

	$\alpha = \beta = 1$			$\alpha = \beta = 0.98$		
$n =$	50	100	150	50	100	150
Maximum acceleration	2.58	3.03	4.30	3.88	5.93	9.15
Overall relative error	9.95%	8.30%	5.43%	5.06%	3.03%	1.21%

in a single thread plus the needed synchronizations. The cause of the latter is the need to compute non-local part of the fractional derivative.

High amount of reduction operations in TFQMR algorithm leads to only 3-times maximal speed-up (32 threads, AMD EPYC, with further increase of thread number increasing execution time) achieved while solving integer-order problem while 6-times speed-up (64 threads, AMD EPYC) was achieved for the fractional-order one.

Running on 8 threads (the maximal number tested on all three CPUs) the highest acceleration of computation was measured on Intel Xeon CPU and the lowest — on AMD Ryzen 3.

It also should be noted that on AMD EPYC high coefficients of variation (>15%) of execution times among performed runs were observed when 8 and more cores were used. Other CPUs did not demonstrate such a behaviour.

The second series of simulations was aimed at testing multithreaded implementation's performance on AMD EPYC CPU when grid size changes ($n = 50, 100, 150$, $m = 10n$, number of threads $P = 1, 2, 4, 8, 16, 32, 64$).

Assuming that the time $t_1(P)$ spent on executing barrier synchronizations and single-threaded blocks is $t_1(P) \leq k_2 P$ we assess total execution time as

$$T(n, P) = k_1 \frac{n}{P} + k_2 P \tag{4}$$

where k_1, k_2 are the performance coefficients.

For each value of n we measured execution time $T_e(n, P)$ and computed the maximum acceleration of computations along with the overall relative error of the description of execution time changes subject to the number of threads

$$\varepsilon(n) = \frac{\sum\limits_i \left(T(n, P_i) - T_e(n, P_i)\right)^2}{\sum\limits_i T_e^2(n, P_i)}.$$

Coefficient values of the performance model (4) for a fixed n and a series of P values were determined by the least squares fitting. The computed values are given in Tab. 2.

Maximum acceleration here increased close-to-linearly with the increase of n. It should also be noted that due to the usage of the procedure of dynamic time step selection, number of time steps also increased with the increase of n making impossible direct comparison of execution times.

Overall relative error of their description by the classical model (4) lowered with the increase of n being within the 10% limit.

However, local relative errors were in some cases up to 50% for $P \geq 4$ due to high variability of execution times in these cases.

The ratio k_1/k_2 was in the narrow ranges $[1.08, 1.24]$ for the classical model and $[2.68, 3]$ for the fractional-order one that supports the applicability of the performance model (4) and additionally confirms higher computational complexity when solving fractional differential equations.

5. Conclusion

Regarding the obtained experimental results it could be stated that the proposed multithreaded implementation of the solver for the space-fractional two-dimensional moisture transport equation can be efficient when executing on up to 8 cores on modern CPUs.

However, its efficiency is limited (maximal acceleration was achieved when 32-64 cores were used) on high-performance AMD EPYC CPUs due to a large number of blocks that needs execution in a single thread in the used TFQMR linear solver.

Thus, regarding optimization problems solved by meta-heuristic algorithms on EPYC CPUs it could be efficient to employ both task and data level parallelism restricting the scale of the latter using the obtained performance estimates.

References

1. Richards L.A. Capillary conduction of liquids through porous mediums. *Physics*. 1931. Vol. 1. No. 5. P. 318–333.
2. Pachepsky Y., Timlin D. Water transport in soils as in fractal media. *Journal of Hydroogy*. 1988. Vol. 204. No. 1–4. P. 98–107.
3. Podlubny I. Fractional differential equations. New York: Academic Press, 1999.
4. Pachepsky Y., Timlin D., Rawls W. Generalized Richards' equation to simulate water transport in unsaturated soils. *Journal of Hydrology*. 2003. Vol. 272. P. 3–13.
5. Kandelous M.M, Kamai T., Vrugt J.A., Simunek J., Hanson B., Hopmans J.W. Evaluation of subsurface drip irrigation design and management parameters for alfalfa. *Agricultural Water Management*. 2012. Vol. 109. P. 81–93.
6. Egidi N., Gioia E., Maponi P., Spadoni L. A numerical solution of Richards equation: a simple method adaptable in parallel computing, *nternational Journal of Computer Mathematics*. 2020. Vol. 97. No. 1–2, P. 2–17.
7. Romashchenko M., Bohaienko V., Matiash T., Kovalchuk V., Krucheniuk A. Numerical simulation of irrigation scheduling using fractional Richards equation. *Irrigation Science*. 2021. Vol. 39. No. 3. P. 385–396.
8. Bohaienko V.O. Parallel algorithms for modelling two-dimensional non-equilibrium salt transfer processes on the base of fractional derivative model. *Fractional Calculus and Applied Analysis*. 2018. Vol. 21. No. 3. P. 654–671.
9. Romashchenko M., Bohaienko V., Bilobrova A. Two-dimensional mathematical modelling of soil water regime under drip irrigation (in Ukrainian). *Bulletin of Agricultural Science*. 2021. Vol. 99. No. 4. P. 59–66.

10. Gomez-Aguilar J.F., Miranda-Hernandez M., Lopez-Lopez M.G., Alvarado- Martinez V.M., Baleanu D. Modeling and simulation of the fractional space-time diffusion equation. *Communications in Nonlinear Science and Numerical Simulation.* 2016, Vol. 30. P. 115–127.
11. van Genuchten M.T. A closed-form equation for predicting the hydraulic conductivity of unsaturated soils. *Soil Sci. Soc. Am. J.* . 1980. Vol. 44. P. 886–900.
12. Averyanov S.F. Filtration from canals and its influence on groundwater regime (in Russian). Moscow: Kolos, 1982.
13. Samarskii A.A. The theory of difference schemes. New York (NY): CRC Press, 2001.
14. Freund R.W. A transpose-free quasi-minimal residual algorithm for non-hermitian linear systems. *SIAM J. Sci. Comput.* 1993. Vol. 14. No. 2. P. 470–482

Received: 20.08.2022 / Accepted: 25.09.2022