

Article

One Approach to Numerical Modeling of the Heat and Mass Transfers of Two-Phase Fluids in Fractured-Porous Reservoirs

Yuliya O. Bobreneva ^{1,*}, Yury Poveshchenko ², Viktoriia O. Podryga ², Sergey V. Polyakov ²,
Ravil M. Uzyanbaev ^{1,3,*}, Parvin I. Rahimly ², Ainur A. Mazitov ¹ and Irek M. Gubaydullin ^{1,3,*}

¹ Institute of Petrochemistry and Catalysis, Russian Academy of Sciences, Pr. Oktyabrya Street 141, Ufa 450075, Russia; mazitov.ainur13@gmail.com

² Keldysh Institute of Applied Mathematics, Russian Academy of Sciences, Miusskaya Square 4, Moscow 125047, Russia; hecon@mail.ru (Y.P.); pvictoria@list.ru (V.O.P.); polyakov@imamod.ru (S.V.P.); pervin@rehimli.info (P.I.R.)

³ Department of Digital Technologies and Modeling, Graduate School of Information and Social Technologies, Ufa State Petroleum Technological University, Kosmonavtov Street 1, Ufa 450062, Russia

* Correspondence: yu.o.bobreneva@gmail.com (Y.O.B.); ravil-11@mail.ru (R.M.U.); irekmars@mail.ru (I.M.G.)

Abstract: The work is devoted to numerical modeling of the processes of heat and mass transfer of a two-phase fluid in the environment of a production well, which is necessary for monitoring the development of fractured-porous reservoirs. This work proposes an efficient approach to constructing a solution to the problem. To solve the problem, a model of the “double medium” type is used, where the pore part of the reservoir is considered as the first medium, and the system of natural fractures is considered as the second medium. For the resulting mathematical model, the difference schemes with time weights are constructed based on the algorithm of splitting by physical processes, which ensure the correctness and consistency of fluxes in the fracture system and the pore reservoir. In the numerical solution, the approximations of differential operators obtained in the framework of the finite difference method are used. For the parallel implementation of the developed numerical approach, the domain decomposition method and the matrix sweep algorithm are chosen. The program implementation is made using the MPI standard. Computational experiments are carried out, the results of which confirm the effectiveness of the developed numerical algorithm and its parallel implementation. In numerical experiments, the distributions of pressure and temperature near an operating production well are obtained, on the basis of which it is possible to adjust the operation of wells in order to increase production.

Keywords: mathematical modeling; two-phase filtration; fractured-porous reservoir; pressure; temperature; implicit finite difference schemes; matrix sweep method

MSC: 65Y05; 00A71; 35M10; 65N06; 76S05



Citation: Bobreneva, Y.O.; Poveshchenko, Y.; Podryga, V.O.; Polyakov, S.V.; Uzyanbaev, R.M.; Rahimly, P.I.; Mazitov, A.A.; Gubaydullin, I.M. One Approach to Numerical Modeling of the Heat and Mass Transfer of Two-Phase Fluids in Fractured-Porous Reservoirs. *Mathematics* **2023**, *11*, 3991. <https://doi.org/10.3390/math11183991>

Academic Editors: Leonid B. Sokolinsky and Mikhail Zymbler

Received: 3 August 2023

Revised: 1 September 2023

Accepted: 6 September 2023

Published: 20 September 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Many processes of heat and mass transfer occurring in nature are accompanied by the transfer of the mass of one substance to the mass of another substance [1,2]. These processes are of great practical importance in various fields. One of these areas is the oil and gas industry, where the use of scientific approaches to improve development technologies is an integral part.

To date, the scale of hydrocarbon production has significantly increased, new fields with complex geological conditions are being put into development. And each new open object most often has degraded filtration properties, which requires new approaches to development. In this regard, the level of requirements for the study of the movement of liquids (gas, oil and water) in operated reservoirs has significantly increased.

Oil fields are confined to uplifts of sedimentary rocks, which are accumulations of grains of minerals interconnected as a result of geological processes [3]. In carbonate rocks, the pore system is characterized by strong heterogeneity, while the system of secondary voids is more developed. Secondary voids include fractures caused by tectonic stresses and caverns resulting from various chemical reactions [4,5]. The length and dimensions of fractures and cavities may exceed the dimensions of primary pores. All of the above gives the features of the well-known theory of fluid filtration in reservoirs. One of them is the need to simultaneously consider processes in different media, namely, in two different systems (a system of natural fractures and a pore part of a reservoir (matrix)) with different reservoir properties, which numerically can reach a difference of several orders of magnitude [6,7].

Information about the reservoir consists of data from the study of rock samples, hydrodynamic studies of wells, analysis of the results of oil and formation water samples, and the history of field development [1]. But with such a variety of data, the information obtained is limited and not always sufficient for unambiguously building a reservoir model. At the same time, conducted experiments at wells incur large economic costs, as well as large production losses due to shutdowns of wells for research. Under such conditions, the task of the study is to establish qualitative patterns, quantitative relationships that are resistant to variation in the initial data and expand the totality of information that cannot be obtained experimentally [8]. Therefore, the solution of practical problems of the modern oil and gas industry is possible only with the help of numerical modeling, which requires the use of the most modern developments and fast solutions.

The process considered in this work is based on the well-known equations of hydrodynamics [2,9]. For traditional filtration models of the “single medium” type, effective algorithms have been developed, both direct ones for modeling the filtration process, and methods for interpreting hydrodynamic research data to identify model parameters. However, the situation is significantly more complicated when using models related to the “dual medium” [10,11], which is associated with practical features (the presence of two systems—the pore part of the reservoir and fractures) of the process under consideration, which complicate the model.

The programs for analysis and interpretation of results that exist both in Russia and abroad do not allow for a full range of calculations and are not always computationally efficient, which limits the range of technical and economic development problems solved with their help. Therefore, there is a need to create a fast calculation tool based on effective numerical algorithms that will allow solving operational planning problems that arise during field development, which include an express assessment of the required shutdown duration before the study, as well as a thorough study of the behavior of fluid dynamics processes at various reservoir parameters.

The difficulty of solving the problem lies in the large number of variables that affect the ongoing processes, in the nonlinearity of the basic equations of hydrodynamics, heat and mass transfer, in the impossibility of obtaining all information about the course of the process due to the high laboriousness of experimental studies [12].

Numerical modeling of such processes is also associated with great difficulties, similarly associated with a large number of unknowns and the use of small time steps [12,13]. The possibilities of modern computer technologies make it possible to implement tasks of this kind. To successfully solve the problem, an efficient numerical algorithm is needed, with the possibility of running on the architecture of large supercomputers.

In our previous works [14,15], the isothermal process of mass transfer of a two-phase fluid in a fractured-porous reservoir was considered in a one-dimensional setting. An original implicit finite-difference scheme on a non-uniform spatial grid and a parallel algorithm for solving the problem were proposed. The obtained calculation results confirmed the efficiency of the proposed algorithm and its parallel implementation.

In this work, we consider a non-isothermal process of heat and mass transfer of a two-phase fluid in a carbonate reservoir of a fractured porous type. A new mathematical model has been constructed for it, which additionally includes the energy equation. For this

model, an original difference scheme with splitting by physical processes has been developed. The scheme ensures the correctness and consistency of fluxes between the natural fracture system and the pore reservoir. Within the framework of the program implementation, the proposed numerical algorithm was parallelized using the MPI standard and the domain decomposition technique. With the help of the developed program, a series of computational experiments was carried out. A feature of the performed calculations is the use of field and geophysical data on the production characteristics and the filtration and capacitance properties of the formations. The goals of the calculations were to determine: (a) the optimal number of processes for solving the system of difference equations; and (b) the pressure and temperature dynamics depending on formation permeability values. The results of numerical experiments confirmed the correctness and efficiency of the developed numerical approach.

2. Problem Statement

The process of heat and mass transfer of a two-phase fluid in a carbonate reservoir (see Figure 1) is considered, where there are two pore systems—a system of fractures and the pore part of the reservoir (matrices). Figure 1 presents a fragment of the reservoir. Each of the systems is characterized by its own reservoir parameters (k, ϕ), the difference between which is achieved from one to several orders.

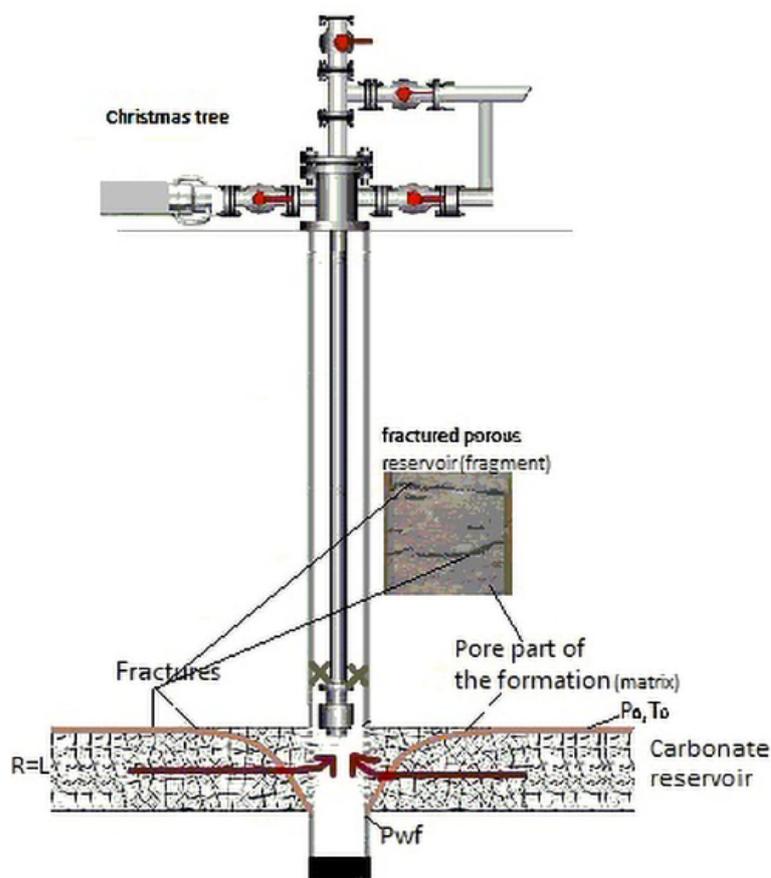


Figure 1. Scheme of the well and reservoir.

At the initial moment of time, the formation is not disturbed, i.e., initial formation pressures (P_0) and temperature (T_0) are set. The well is put into operation with constant bottom hole pressure (P_{wf}). At the moment of putting into operation, due to the created depression, the inflow of fluid from the formation to the well begins. The fluid flow occurs only along the fractures, and the pore part of the reservoir (matrix) is a certain capacity of fluid accumulation and is taken into account by introducing special functions [16]. When

the pressure in the fracture system decreases lower than in the pore part (matrix), fluid flows from the matrix into the fractures.

It is assumed that the reservoir is homogeneous and isotropic. The flow of a weakly compressible fluid in a system of fractures is within the validity of Darcy’s law. It is necessary to reproduce the dynamics of pressure and temperature during the operation of the well at different times and distances from the well at different values of fracture and matrix permeability. Figure 1 shows a scheme of a well and a reservoir.

The Christmas tree in Figure 1 is the system of mechanisms and devices designed to seal the mouth of pumping and fountain wells and their mutual isolation; fractures are the secondary voids caused by tectonic stresses, through which fluid flows; the pore part of the formation (matrices) is the rock that has physical properties that allow it to accumulate water, oil and gas, as well as filter and release them in the presence of a pressure drop; the carbonate reservoir is the type of reservoir, which mainly consists of limestone and dolomite, with a developed system of secondary voids; $R = L$ is the distance to which reservoir disturbances reach during well operation; P_w^f is the bottom hole pressure; and P_0 and T_0 are the initial reservoir pressure and temperature.

The mathematical description of the filtration processes is presented on the basis of the classical laws of continuum mechanics [1,2].

$$\frac{\partial(\phi^\alpha \rho_o^\alpha S_o^\alpha)}{\partial t} + \nabla(\rho_o^\alpha \vec{U}_o^\alpha) + q_o^\alpha = 0, \quad q_o^m = -q_o^f = -\rho_o^m \sigma \lambda_o^m (P^f - P^m), \quad (1)$$

$$\frac{\partial(\phi^\alpha \rho_w^\alpha S_w^\alpha)}{\partial t} + \nabla(\rho_w^\alpha \vec{U}_w^\alpha) - q_w^\alpha = 0, \quad q_w^m = -q_w^f = -\rho_w^m \sigma \lambda_w^m (P^f - P^m), \quad (2)$$

$$\begin{aligned} \frac{\partial}{\partial t} [(\phi^f \rho_o^f S_o^f \varepsilon_o^f + \phi^m \rho_o^m S_o^m \varepsilon_o^m + \phi^f \rho_w^f S_w^f \varepsilon_w^f + \phi^m \rho_w^m S_w^m \varepsilon_w^m) + (1 - \phi^f - \phi^m) \rho_s \varepsilon_s] + \\ \text{div}[\rho_o^f \varepsilon_o^f \vec{U}_o^f + \rho_w^f \varepsilon_w^f \vec{U}_w^f] + \text{div}[P^f (\vec{U}_o^f + \vec{U}_w^f)] + \text{div}[\vec{W}^f + \vec{W}^m + \vec{W}_s] = 0, \end{aligned} \quad (3)$$

$$\lambda_o^m = \frac{k^m k_{ro}(S_o^m)}{\mu_o}, \quad \lambda_w^m = \frac{k^m k_{rw}(S_w^m)}{\mu_w}. \quad (4)$$

Here, $\alpha = f, m$, where f is the fracture system, m is the matrix system, $i = o, w$, where o is the oil, w is the water, P^f is the formation pressure in the fracture system (Pa), P^m is the formation pressure in the matrix (Pa), ϕ^f is the porosity in the fracture system, ϕ^m is the porosity in the matrix, ρ_o^α is the density of oil (g/m^3), ρ_w^α is the density of water (g/m^3), S_i^f is the saturation of oil or water in the fracture system, S_i^m is the saturation of oil or water in the matrix, \vec{U}_i^α is the flow velocity of oil or water, q_i^α is the function of redistribution of the fluid between the matrix and the fractures, σ is the coefficient of fractured rock ($1/m^2$), ε_i^α is the energy of oil/water, s is rock skeleton, ρ_s, ε_s is the density and energy of the system, k^α is the absolute permeability (m^2), k_{rw} and k_{ro} are the relative phase permeabilities of water and oil, μ_o is the viscosity of oil (Pa·s), and μ_w is the viscosity of water (Pa·s).

Let us introduce notations:

$$\begin{aligned} \vec{W}^f = -(\phi^f [S_w^f \eta_w^f + (1 - S_w^f) \eta_o^f]) \nabla T, \quad \vec{W}^m = -(\phi^m [S_w^m \eta_w^m + (1 - S_w^m) \eta_o^m]) \nabla T, \\ \vec{W}_s = -[1 - \phi^f - \phi^m] \eta_s \nabla T, \quad \vec{W} = \vec{W}^f + \vec{W}^m + \vec{W}_s, \end{aligned} \quad (5)$$

where T is the temperature (K), and $\eta_w^f, \eta_w^m, \eta_s$ are the thermal conductivity coefficients in the system of fracture, matrix and skeleton.

The flow of fluid from the pore part of the reservoir into fractures is described by the following functions:

$$q_o^m = -q_o^f = -\frac{\sigma \rho_o^m k^m k_{ro}(S_o^m)}{\mu_o} (P^f - P^m), \quad q_w^m = -q_w^f = -\frac{\sigma \rho_w^m k^m k_{rw}(S_w^m)}{\mu_w} (P^f - P^m). \quad (6)$$

The generalized Darcy’s law is applied, according to which the filtration velocities of oil and water are equal to:

$$\vec{U}_o^\alpha = -\frac{k^\alpha k_{ro}(S_o^\alpha)}{\mu_o} grad P^\alpha, \vec{U}_w^\alpha = -\frac{k^\alpha k_{rw}(S_w^\alpha)}{\mu_w} grad P^\alpha. \tag{7}$$

Here, $\alpha = f, \vec{U}_o^m = \vec{U}_w^m = 0$.

We note that the investigated well is not affected by wells of the environment, while the reservoir is presented as infinite. In this regard, the right boundary condition is represented by constant pressure and temperature at the boundary, the left boundary condition is set by constant bottom hole pressure and temperature. For the initial condition, reservoir pressures and temperatures in the matrix and fractures at the initial time are used.

Problems (1)–(7) are a complex system of equations of mathematical physics of a mixed type. An important circumstance is that the problem formulated in the form of conservation laws, with a common matrix of the system with respect to water saturation, pressure and temperature, has mixed hyperbolic and parabolic properties. The direct use of such a system for the purposes of determining the dynamics of the above variables and constructing an implicit difference scheme required for calculating parabolic equations with large time steps is difficult. Accordingly, the numerical solution is not a trivial problem. To solve this problem for the initial equations, the method of splitting by physical processes [9,12,17,18] is used, where the equations are separated into the piezoconductive part and with respect to saturation transfer.

System (1)–(7), after some algebraic transformations, will be presented in the following form:

$$\begin{aligned} & \frac{S_w^f}{\rho_w^f} [\phi^f \rho_w^f]_t + \frac{(1 - S_w^f)}{\rho_o^f} [\phi^f \rho_o^f]_t + DIG^f = 0, \\ DIG^f &= \frac{1}{\rho_w^f} div(\rho_w^f \vec{U}_w^f) + \frac{1}{\rho_o^f} div(\rho_o^f \vec{U}_o^f) + \frac{q_w^f}{\rho_w^f} + \frac{q_o^f}{\rho_o^f}, \end{aligned} \tag{8}$$

$$\frac{S_w^m}{\rho_w^m} [\phi^m \rho_w^m]_t + \frac{(1 - S_w^m)}{\rho_o^m} [\phi^m \rho_o^m]_t + DIG^m = 0, \quad DIG^m = \frac{q_w^m}{\rho_w^m} + \frac{q_o^m}{\rho_o^m}, \tag{9}$$

$$\begin{aligned} & \phi^f \left(S_w^f \rho_w^f \frac{\partial \varepsilon_w^f}{\partial t} + (1 - S_w^f) \rho_o^f \frac{\partial \varepsilon_o^f}{\partial t} \right) + \phi^m \left(S_w^m \rho_w^m \frac{\partial \varepsilon_w^m}{\partial t} + (1 - S_w^m) \rho_o^m \frac{\partial \varepsilon_o^m}{\partial t} \right) + \\ & \frac{\partial}{\partial t} (1 - \phi^f - \phi^m) \rho_s \varepsilon_s + DIG_\varepsilon^f + DIG_\varepsilon^m + div \vec{W}_s = 0, \\ DIG_\varepsilon^f &= \left[div(\rho_w^f \varepsilon_w^f \vec{U}_w^f) - \varepsilon_w^f div(\rho_w^f \vec{U}_w^f) \right] + \left[div(\rho_o^f \varepsilon_o^f \vec{U}_o^f) - \varepsilon_o^f div(\rho_o^f \vec{U}_o^f) \right] + \\ & + div \left[P^f (\vec{U}_w^f + \vec{U}_o^f) \right] + div \vec{W}^f + (-\varepsilon_w^f q_w^f - \varepsilon_o^f q_o^f) = 0, \\ DIG_\varepsilon^m &= div \vec{W}^m + (-\varepsilon_w^m q_w^m - \varepsilon_o^m q_o^m), \\ div \vec{W}^f &= - \left(\phi^f [S_w^f \eta_w^f + (1 - S_w^f) \eta_o^f] \right), div \vec{W}^m = - \left(\phi^m [S_w^m \eta_w^m + (1 - S_w^m) \eta_o^m] \right). \end{aligned} \tag{10}$$

3. Difference Scheme Construction

A one-dimensional statement of Problems (1)–(7) is considered. An implicit finite-difference scheme on a uniform grid is used to solve partial differential equations [15,19,20]. Note that the approach presented below will be the basis for the problem in the multidimensional case. The use of a non-uniform grid is also possible for this statement.

$$\bar{W}_h = (x_i = ih, i = 0, 1, \dots, N, x_0 = 0, x_N = L), \quad \bar{W}_\tau = (t_j = k\tau, k = 0, 1, \dots, M), \tag{11}$$

where x_i are the coordinates of grid nodes in space, h is the grid step in radius, t_j are the coordinates of grid nodes in time, τ is the grid step in time, and N and M are the number of nodes in space and time, respectively. The grid values (pressure and saturation) are determined in x_i . The $i + 1/2$ -th cell of a one-dimensional grid Ω_i is understood as a segment $[x_i, x_{i+1}]$. $h_{i+\frac{1}{2}} = x_{i+1} - x_i$, $\hat{h}_i = (h_{i+\frac{1}{2}} + h_{i-\frac{1}{2}})/2$.

Differential Equations (1)–(3) are approximated by their grid analogues [19,20]:

$$(\bar{\phi}^\alpha \rho_o^\alpha S_o^\alpha)_t + DIN(\rho_o^\alpha U_o^\alpha)^\sim + q_o^{\alpha\sim} = 0, q_o^{m\sim} = -q_o^{f\sim} = -\rho_o^{m\sim} \bar{\sigma}^\sim \lambda_o^{m\sim} (P^{f\sim} - P^{m\sim}), \tag{12}$$

$$(\bar{\phi}^\alpha \rho_w^\alpha S_w^\alpha)_t + DIN(\rho_w^\alpha U_w^\alpha)^\sim + q_w^{\alpha\sim} = 0, q_w^{m\sim} = -q_w^{f\sim} = -\rho_w^{m\sim} \bar{\sigma}^\sim \lambda_w^{m\sim} (P^{f\sim} - P^{m\sim}), \tag{13}$$

$$\begin{aligned} & [(\bar{\phi}^f \rho_o^f S_o^f \varepsilon_o^f + \bar{\phi}^m \rho_o^m S_o^m \varepsilon_o^m + \bar{\phi}^f \rho_w^f S_w^f \varepsilon_w^f + \bar{\phi}^m \rho_w^m S_w^m \varepsilon_w^m) + \overline{(1 - \phi^f - \phi^m)} \rho_s \varepsilon_s]_t + \\ & + DIN[(\varepsilon_o^{f(\delta_{1f})})_{up}(\rho_o^f U_o^f)^\sim + (\varepsilon_w^{f(\delta_{1f})})_{up}(\rho_w^f U_w^f)^\sim] + DIN[P^f(U_o^f + U_w^f)]^\sim + \\ & + DIN[(W^f)^\sim + (W^m)^\sim + (W_s)^\sim] = 0, \\ & (W^f)^\sim_{k+\frac{1}{2}} = -[\phi^f(S_w^f \eta_w^f + (1 - S_w^f) \eta_o^f)]^\sim_{k+\frac{1}{2}} \frac{(\hat{T}_{k+1} - \hat{T}_k)}{h_{k+\frac{1}{2}}}, \\ & (W^m)^\sim_{k+\frac{1}{2}} = -[\phi^m(S_w^m \eta_w^m + (1 - S_w^m) \eta_o^m)]^\sim_{k+\frac{1}{2}} \frac{(\hat{T}_{k+1} - \hat{T}_k)}{h_{k+\frac{1}{2}}}, \\ & (W_s)^\sim_{k+\frac{1}{2}} = -[(1 - \phi^f - \phi^m) \eta_s]^\sim_{k+\frac{1}{2}} \frac{(\hat{T}_{k+1} - \hat{T}_k)}{h_{k+\frac{1}{2}}}. \end{aligned} \tag{14}$$

Here, $\bar{\phi} = \hat{h}\phi$, $\overline{(1 - \phi^f - \phi^m)} = \hat{h} - \bar{\phi}^f - \bar{\phi}^m$, $\bar{\sigma}^\sim = \hat{h}\sigma^\sim$, δ_{1f} is weight by time, a^\sim denotes the approximation of the grid function a between the layers in time t and \hat{t} , $\hat{t} = t + \tau$, τ is the time step, and up denotes an upstream approximation of the expression, $(DINa)_i = (a_{i+\frac{1}{2}} + a_{i-\frac{1}{2}})$, $\delta_{1f} = \frac{\sqrt{(\phi^f)^\wedge}}{\sqrt{(\phi^f)^\wedge} + \sqrt{(\phi^f)}}$, $\delta_{1m} = \frac{\sqrt{(\phi^m)^\wedge}}{\sqrt{(\phi^m)^\wedge} + \sqrt{(\phi^m)}}$.

After splitting, the system (12)–(14) has the form:

$$\begin{aligned} \frac{F^f}{\tau} &= \frac{(S_w^f)^{(\delta_{1f})}}{(\rho_w^f)^{(\delta_{1f})}} [\bar{\phi}^f \rho_w^f]_t + \frac{(1 - S_w^f)^{(\delta_{1f})}}{(\rho_o^f)^{(\delta_{1f})}} [\bar{\phi}^f \rho_o^f]_t + DIG^{f\sim} = 0, \\ DIG^{f\sim} &= \frac{1}{(\rho_w^f)^{(\delta_{1f})}} DIN(\rho_w^f U_w^f)^\sim + \frac{1}{(\rho_o^f)^{(\delta_{1f})}} DIN(\rho_o^f U_o^f)^\sim + \frac{q_o^{f\sim}}{(\rho_o^f)^{(\delta_{1f})}} + \frac{q_w^{f\sim}}{(\rho_w^f)^{(\delta_{1f})}}, \end{aligned} \tag{15}$$

$$\begin{aligned} \frac{F^m}{\tau} &= \frac{(S_w^m)^{(\delta_{1m})}}{(\rho_w^m)^{(\delta_{1m})}} [\bar{\phi}^m \rho_w^m]_t + \frac{(1 - S_w^m)^{(\delta_{1m})}}{(\rho_o^m)^{(\delta_{1m})}} [\bar{\phi}^m \rho_o^m]_t + DIG^{m\sim} = 0, \\ DIG^{m\sim} &= \frac{q_o^{m\sim}}{(\rho_o^m)^{(\delta_{1m})}} + \frac{q_w^{m\sim}}{(\rho_w^m)^{(\delta_{1m})}}, \end{aligned} \tag{16}$$

$$\begin{aligned} -\frac{\Phi_{\varepsilon k}}{\tau} &= (\bar{\phi}^f)^{(1-\delta_{1f})} \left([S_w^f \rho_w^f]^{(\delta_{1f})} (\varepsilon_w^f)_t + [(1 - S_w^f) \rho_o^f]^{(\delta_{1f})} (\varepsilon_o^f)_t \right) + \\ & + (\bar{\phi}^m)^{(1-\delta_{1m})} \left([S_w^m \rho_w^m]^{(\delta_{1m})} (\varepsilon_w^m)_t + [(1 - S_w^m) \rho_o^m]^{(\delta_{1m})} (\varepsilon_o^m)_t \right) + \left(\overline{(1 - \phi^f - \phi^m)} \rho_s \varepsilon_s \right)_t + \\ & + DIG_\varepsilon^{f\sim} + DIG_\varepsilon^{m\sim} + DINW_s^\sim = 0, \\ DIG_\varepsilon^{f\sim} &= \left(DIN \left[(\varepsilon_w^{f(\delta_{1f})})_{up}(\rho_w^f U_w^f)^\sim \right] - \varepsilon_w^{f(\delta_{1f})} DIN(\rho_w^f U_w^f)^\sim \right) + \end{aligned}$$

$$\begin{aligned}
 & + \left(\text{DIN} \left[(\varepsilon_o^f)^{(\delta_{1f})} \right]_{up} (\rho_o^f U_o^f)^\sim \right) - \varepsilon_o^f (\delta_{1f}) \text{DIN} (\rho_o^f U_o^f)^\sim + \\
 & + \text{DIN} [P^f (U_w^f + U_o^f)]^\sim + \text{DIN} W^{f\sim} - (\varepsilon_w^f)^{(\delta_{1f})} q_w^{f\sim} - (\varepsilon_o^f)^{(\delta_{1f})} q_o^{f\sim}, \\
 & \text{DIG}_\varepsilon^{m\sim} = \text{DIN} W^{m\sim} - (\varepsilon_w^m)^{(\delta_{1m})} q_w^{m\sim} - (\varepsilon_o^m)^{(\delta_{1m})} q_o^{m\sim}.
 \end{aligned} \tag{17}$$

As a result of the linearization of Equation (16), we obtain a relationship between the increments of pressures in the matrix (δP^f) and in the fracture (δP^m), temperature (δT) and the residual (Φ^{ms}).

$$\delta P^m = \pi_m^s \delta P^f - \Phi^{ms} - \frac{\Theta_{T_m}^s}{\Theta_{P_m\tau}^s} \delta T, \tag{18}$$

where

$$\pi_m^s = \frac{\tau}{\Theta_{P_m\tau}^s} \left(\frac{(\rho_w^m \bar{\sigma} \lambda_w^m)^s}{(\rho_w^m)^{(\delta_{1m})^\approx}} + \frac{(\rho_o^m \bar{\sigma} \lambda_o^m)^s}{(\rho_o^m)^{(\delta_{1m})^\approx}} \right), \quad \Phi^{ms} = \frac{F^{ms}}{\Theta_{P_m\tau}^s}, \tag{19}$$

$$\begin{aligned}
 \Theta_{P_m\tau}^s &= \Theta_{P_m}^s + \tau \left(\frac{(\rho_w^m \bar{\sigma} \lambda_w^m)^s}{(\rho_w^m)^{(\delta_{1m})^\approx}} + \frac{(\rho_o^m \bar{\sigma} \lambda_o^m)^s}{(\rho_o^m)^{(\delta_{1m})^\approx}} \right), \\
 \Theta_{P_m}^s &= \frac{(S_w^m)^{(\delta_{1m})^\approx}}{(\rho_w^m)^{(\delta_{1m})^\approx}} \left(\bar{\phi}^m \rho_w^m \right)_{P_m}^{ts} + \frac{(1 - S_w^m)^{(\delta_{1m})^\approx}}{(\rho_o^m)^{(\delta_{1m})^\approx}} \left(\bar{\phi}^m \rho_o^m \right)_{P_m}^{ts},
 \end{aligned} \tag{20}$$

$$\Theta_{T_m}^s = \frac{(S_w^m)^{(\delta_{1m})^\approx}}{(\rho_w^m)^{(\delta_{1m})^\approx}} \left(\bar{\phi}^m \rho_w^m \right)_T^{ts} + \frac{(1 - S_w^m)^{(\delta_{1m})^\approx}}{(\rho_o^m)^{(\delta_{1m})^\approx}} \left(\bar{\phi}^m \rho_o^m \right)_T^{ts}. \tag{21}$$

Here, a^\approx means that the meanings of the value of the implicit layer in time are taken at a known iteration. The index s , denoting a known computed iteration, has the same sense. The prime denotes the derivative with respect to thermodynamic variables (pressure or temperature). Here, F^{ms} is taken from (17) with the meanings of the values of the implicit layer in time from the s -th iteration. Further, the linearized system of Equations (15) and (17), taking into account Equation (18), will have the form:

$$\begin{aligned}
 & - (A_{pk}^{11} \delta P_{k-1}^f + A_{pk}^{12} \delta T_{k-1}) + (C_{pk}^{11} \delta P_k^f + C_{pk}^{12} \delta T_k) - (B_{pk}^{11} \delta P_{k+1}^f + B_{pk}^{12} \delta T_{k+1}) = \Phi_{pk}^1, \\
 & - (A_{ek}^{21} \delta P_{k-1}^f + A_{ek}^{22} \delta T_{k-1}) + (C_{ek}^{21} \delta P_k^f + C_{ek}^{22} \delta T_k) - (B_{ek}^{21} \delta P_{k+1}^f + B_{ek}^{22} \delta T_{k+1}) = \Phi_{ek}^2,
 \end{aligned} \tag{22}$$

where the matrices have the following form:

$$A_{pk}^{12} = 0, B_{pk}^{12} = 0, \tag{23}$$

$$\begin{aligned}
 A_{pk}^{11} &= \frac{\tau}{[(\rho_w^f)^{(\delta_{1f})}]_k^\approx} \left(\frac{1}{h_{k-\frac{1}{2}}} \left(\frac{\rho_w^f k^f}{\mu_w^f} \right)_{k-\frac{1}{2}}^s k_{rw(k-\frac{1}{2})}^{ups} \right) + \\
 & + \frac{\tau}{[(\rho_o^f)^{(\delta_{1f})}]_k^\approx} \left(\frac{1}{h_{k-\frac{1}{2}}} \left(\frac{\rho_o^f k^f}{\mu_o^f} \right)_{k-\frac{1}{2}}^s k_{ro(k-\frac{1}{2})}^{ups} \right),
 \end{aligned} \tag{24}$$

$$\begin{aligned}
 B_{pk}^{11} &= \frac{\tau}{[(\rho_w^f)^{(\delta_{1f})}]_k^\approx} \left(\frac{1}{h_{k+\frac{1}{2}}} \left(\frac{\rho_w^f k^f}{\mu_w^f} \right)_{k+\frac{1}{2}}^s k_{rw(k+\frac{1}{2})}^{ups} \right) + \\
 & + \frac{\tau}{[(\rho_o^f)^{(\delta_{1f})}]_k^\approx} \left(\frac{1}{h_{k+\frac{1}{2}}} \left(\frac{\rho_o^f k^f}{\mu_o^f} \right)_{k+\frac{1}{2}}^s k_{ro(k+\frac{1}{2})}^{ups} \right),
 \end{aligned} \tag{25}$$

$$\begin{aligned}
 C_{pk}^{11} = & \left[\frac{(S_w^f)^{(\delta_{1f}) \approx}}{(\rho_w^f)^{(\delta_{1f}) \approx}} \left(\bar{\phi}^f \rho_w^f \right)_{P_f}^{ts} + \frac{(1 - S_w^f)^{(\delta_{1f}) \approx}}{(\rho_o^f)^{(\delta_{1f}) \approx}} \left(\bar{\phi}^f \rho_o^f \right)_{P_f}^{ts} \right]_k + \\
 & + \frac{\tau}{[(\rho_w^f)^{(\delta_{1f}) \approx}]_k} \left(\frac{1}{h_{k+\frac{1}{2}}} \left(\frac{\rho_w^f k^f}{\mu_w^f} \right)_{k+\frac{1}{2}}^s k_{rw(k+\frac{1}{2})}^{ups} + \frac{1}{h_{k-\frac{1}{2}}} \left(\frac{\rho_w^f k^f}{\mu_w^f} \right)_{k-\frac{1}{2}}^s k_{rw(k-\frac{1}{2})}^{ups} \right) + \\
 & + \frac{\tau}{[(\rho_o^f)^{(\delta_{1f}) \approx}]_k} \left(\frac{1}{h_{k+\frac{1}{2}}} \left(\frac{\rho_o^f k^f}{\mu_o^f} \right)_{k+\frac{1}{2}}^s k_{ro(k+\frac{1}{2})}^{ups} + \frac{1}{h_{k-\frac{1}{2}}} \left(\frac{\rho_o^f k^f}{\mu_o^f} \right)_{k-\frac{1}{2}}^s k_{ro(k-\frac{1}{2})}^{ups} \right) + \\
 & + \left(\frac{\tau}{(\rho_w^f)^{(\delta_{1f}) \approx}} (\rho_w^m \bar{\sigma} \lambda_w^m)^s (1 - \pi_m^s) \right)_k + \left(\frac{\tau}{(\rho_o^f)^{(\delta_{1f}) \approx}} (\rho_o^m \bar{\sigma} \lambda_o^m)^s (1 - \pi_m^s) \right)_k, \tag{26}
 \end{aligned}$$

$$\begin{aligned}
 C_{pk}^{12} = & \left(\frac{\Theta_{T_m}^s}{\Theta_{P_{m\tau}}^s} \left(\frac{(\rho_w^m \bar{\sigma} \lambda_w^m)^s}{(\rho_w^f)^{(\delta_{1f}) \approx}} + \frac{(\rho_o^m \bar{\sigma} \lambda_o^m)^s}{(\rho_o^f)^{(\delta_{1f}) \approx}} \right) \tau + \right. \\
 & \left. + \left(\frac{(S_w^f)^{(\delta_{1f}) \approx}}{(\rho_w^f)^{(\delta_{1f}) \approx}} \left(\bar{\phi}^f \rho_w^f \right)_T^{ts} + \frac{(1 - S_w^f)^{(\delta_{1f}) \approx}}{(\rho_o^f)^{(\delta_{1f}) \approx}} \left(\bar{\phi}^f \rho_o^f \right)_T^{ts} \right) \right)_k, \tag{27}
 \end{aligned}$$

$$\Phi_{pk}^1 = -F^f s - \left(\frac{(\rho_w^m \bar{\sigma} \lambda_w^m)^s}{(\rho_w^f)^{(\delta_{1f}) \approx}} + \frac{(\rho_o^m \bar{\sigma} \lambda_o^m)^s}{(\rho_o^f)^{(\delta_{1f}) \approx}} \right) \tau \Phi^{ms}. \tag{28}$$

Here, $F^f s$ is taken from (14) with the meanings of the values of the implicit layer in time from the s -th iteration. Then, the index $k \pm 1/2$ is replaced by $k \pm 1/2$.

$$A_{\epsilon k}^{21} = \tau \left(\frac{1}{h_{-\frac{1}{2}}} \left(\frac{P^f k^f}{\mu_w^f} \right)_{-\frac{1}{2}}^s (k_{rw}^{up})_{-\frac{1}{2}}^s + \frac{1}{h_{-\frac{1}{2}}} \left(\frac{P^f k^f}{\mu_o^f} \right)_{-\frac{1}{2}}^s (k_{ro}^{up})_{-\frac{1}{2}}^s \right), \tag{29}$$

$$B_{\epsilon k}^{21} = \tau \left(\frac{1}{h_{\frac{1}{2}}} \left(\frac{P^f k^f}{\mu_w^f} \right)_{\frac{1}{2}}^s (k_{rw}^{up})_{\frac{1}{2}}^s + \frac{1}{h_{\frac{1}{2}}} \left(\frac{P^f k^f}{\mu_o^f} \right)_{\frac{1}{2}}^s (k_{ro}^{up})_{\frac{1}{2}}^s \right), \tag{30}$$

$$C_{\epsilon k}^{21} = C_{\epsilon k}^{f21} + C_{\epsilon k}^{m21} + C_{\epsilon k}^{s21} + C_{\epsilon P_k}^{f21}, \tag{31}$$

where

$$C_{\epsilon k}^{f21} = \left((\bar{\phi}^f)^{(1-\delta_{1f})} \left([S_w^f \rho_w^f]^{(\delta_{1f})} \left[- \left(\frac{P^f}{\rho_w^f} \right)'_{P_f} \right] + [(1 - S_w^f) \rho_o^f]^{(\delta_{1f})} \left[- \left(\frac{P^f}{\rho_o^f} \right)'_{P_f} \right] \right) \right) \approx, \tag{32}$$

$$\begin{aligned}
 C_{\epsilon k}^{m21} = & \left((\bar{\phi}^m)^{(1-\delta_{1m})} \left([S_w^m \rho_w^m]^{(\delta_{1m})} \left[- \left(\frac{P^m}{\rho_w^m} \right)'_{P_m} \pi_m^s \right] + \right. \right. \\
 & \left. \left. + [(1 - S_w^m) \rho_o^m]^{(\delta_{1m})} \left[- \left(\frac{P^m}{\rho_o^m} \right)'_{P_m} \pi_m^s \right] \right) \right) \approx, \tag{33}
 \end{aligned}$$

$$C_{\epsilon k}^{s21} = \left(\left((1 - \phi^f - \phi^m) \rho_s \left[- \left(\frac{P^s}{\rho^s} \right)' \right] \right)_{P_s} \left[(P_s)'_{P_f} + (P_s)'_{P_m} \pi_m^s \right] \right) \approx, \tag{34}$$

$$C_{\epsilon pk}^{f21} = \tau \left(\frac{1}{h_{-\frac{1}{2}}} \left(\frac{P^f k^f}{\mu_w^f} \right)_{-\frac{1}{2}}^s (k_{rw}^{up})_{-\frac{1}{2}}^s + \frac{1}{h_{-\frac{1}{2}}} \left(\frac{P^f k^f}{\mu_o^f} \right)_{-\frac{1}{2}}^s (k_{ro}^{up})_{-\frac{1}{2}}^s \right) +$$

$$+\tau\left(\frac{1}{h_{\frac{1}{2}}}\left(\frac{P^f k^f}{\mu_w^f}\right)^s (k_{rw}^{up})_{\frac{1}{2}}^s + \frac{1}{h_{\frac{1}{2}}}\left(\frac{P^f k^f}{\mu_o^f}\right)^s (k_{ro}^{up})_{\frac{1}{2}}^s\right), \tag{35}$$

$$A_{\epsilon k}^{22} = \frac{\tau}{h_{-\frac{1}{2}}}\left(\phi^f [S_w^f \eta_w^f + (1 - S_w^f) \eta_o^f]\right)_{-\frac{1}{2}}^{\approx} + \frac{\tau}{h_{-\frac{1}{2}}}\left(\phi^m [S_w^m \eta_w^m + (1 - S_w^m) \eta_o^m]\right)_{-\frac{1}{2}}^{\approx} + \frac{\tau}{h_{-\frac{1}{2}}}\left((1 - \phi^f - \phi^m) \eta_s\right)_{-\frac{1}{2}}^{\approx}, \tag{36}$$

$$B_{\epsilon k}^{22} = \frac{\tau}{h_{\frac{1}{2}}}\left(\phi^f [S_w^f \eta_w^f + (1 - S_w^f) \eta_o^f]\right)_{\frac{1}{2}}^{\approx} + \frac{\tau}{h_{\frac{1}{2}}}\left(\phi^m [S_w^m \eta_w^m + (1 - S_w^m) \eta_o^m]\right)_{\frac{1}{2}}^{\approx} + \frac{\tau}{h_{\frac{1}{2}}}\left((1 - \phi^f - \phi^m) \eta_s\right)_{\frac{1}{2}}^{\approx}. \tag{37}$$

Here, it is supposed to be $P^s = (\phi^f P^f + \phi^m P^m) / (\phi^f + \phi^m)$. Next, we denote the specific heat capacity at constant pressure by C_p and have the following:

$$C_{\epsilon k}^{22} = C_{\epsilon k}^{f22} + C_{\epsilon k}^{m22} + C_{\epsilon k}^{s22} + C_{\eta k}^{f22} + C_{\eta k}^{m22} + C_{\eta k}^{s22}, \tag{38}$$

where

$$C_{\epsilon k}^{f22} = \left((\bar{\phi}^f)^{(1-\delta_{1f})} \left([S_w^f \rho_w^f]^{(\delta_{1f})} \left[c_{\rho w}^f - \left(\frac{P^f}{\rho_w^f} \right)'_T \right] + [(1 - S_w^f) \rho_o^f]^{(\delta_{1f})} \left[c_{\rho o}^f - \left(\frac{P^f}{\rho_o^f} \right)'_T \right] \right) \right)^{\approx}, \tag{39}$$

$$C_{\epsilon k}^{m22} = \left((\bar{\phi}^m)^{(1-\delta_{1m})} \left([S_w^m \rho_w^m]^{(\delta_{1m})} \left[c_{\rho w}^m - \left(\frac{P^m}{\rho_w^m} \right)'_{P_m} - \frac{\Theta_{T_m}^s}{\Theta_{P_m}^s} \right] + [(1 - S_w^m) \rho_o^m]^{(\delta_{1m})} \left[c_{\rho o}^m - \left(\frac{P^m}{\rho_o^m} \right)'_{P_m} - \frac{\Theta_{T_m}^s}{\Theta_{P_m}^s} \right] \right) \right)^{\approx}, \tag{40}$$

$$C_{\epsilon k}^{s22} = \left(\left((1 - \phi^f - \phi^m) \rho_s \left[c_P^s T - \left(\frac{P^s}{\rho^s} \right)' \right] \right)'_T \right)^{\approx}. \tag{41}$$

Here, we accept $(P^s)'_{P_m} = \phi^m / (\phi^f + \phi^m)$ and, with this in mind, $(P^s)'_T = [\phi^m / (\phi^f + \phi^m)] [-\Theta_{T_m}^s / \Theta_{P_m}^s]$.

$$C_{\eta k}^{f22} = \frac{\tau}{h_{\frac{1}{2}}}\left(\phi^f [S_w^f \eta_w^f + (1 - S_w^f) \eta_o^f]\right)_{\frac{1}{2}}^{\approx} + \frac{\tau}{h_{-\frac{1}{2}}}\left(\phi^f [S_w^f \eta_w^f + (1 - S_w^f) \eta_o^f]\right)_{-\frac{1}{2}}^{\approx}, \tag{42}$$

$$C_{\eta k}^{m22} = \frac{\tau}{h_{\frac{1}{2}}}\left(\phi^m [S_w^m \eta_w^m + (1 - S_w^m) \eta_o^m]\right)_{\frac{1}{2}}^{\approx} + \frac{\tau}{h_{-\frac{1}{2}}}\left(\phi^m [S_w^m \eta_w^m + (1 - S_w^m) \eta_o^m]\right)_{-\frac{1}{2}}^{\approx}, \tag{43}$$

$$C_{\eta k}^{s22} = \frac{\tau}{h_{\frac{1}{2}}}\left((1 - \phi^f - \phi^m) \eta_s\right)_{\frac{1}{2}}^{\approx}, \tag{44}$$

$$\Phi_{\epsilon k}^2 = \Phi_{\epsilon k} + \Phi_{\epsilon k}^{s2} + \Phi_{\epsilon k}^{m2}. \tag{45}$$

Here, $\Phi_{\epsilon k}$ is taken from Equation (17), with the meanings of the values of the implicit layer in time from the s -th iteration.

We take

$$\Phi_{\epsilon k}^{s2} = \left(-\frac{\phi^m}{\phi^m + \phi^f} (1 - \phi^f - \phi^m) \left[\frac{F^{ms}}{\Theta_{Pm\tau}^s} \right] \right) \approx, \tag{46}$$

$$\begin{aligned} \Phi_{\epsilon k}^{m2} = & \left((\bar{\phi}^m)^{(1-\delta_{1m})} \left([S_w^m \rho_w^m]^{(\delta_{1m})} \left[-\left(\frac{P^m}{\rho_w^f}\right)'_{Pm} \frac{F^{ms}}{\Theta_{Pm\tau}^s} \right] + \right. \right. \\ & \left. \left. + [(1 - S_w^m) \rho_o^m]^{(\delta_{1m})} \left[-\left(\frac{P^m}{\rho_w^f}\right)'_{Pm} \frac{F^{ms}}{\Theta_{Pm\tau}^s} \right] \right) \right) \approx. \end{aligned} \tag{47}$$

The resulting Equation (22) is represented by a system of linear algebraic equations, which is solved using a matrix sweep method on each time layer.

4. Matrix Sweep

The implicit in pressure and implicit in saturation (IMPIS) method [9] was used to numerically solve the system of equations of two-phase filtration in a fractured-pore type collector (8)–(10). The first block with respect to the pressure and temperature equations was solved using an implicit finite-difference scheme; after finding these parameters in the matrix and fractures, they proceeded to calculate the second block with respect to saturation.

As a result of the approximation of partial derivatives by the corresponding finite differences, we obtain a system of linear algebraic Equations (SLAE) (10) with a matrix of a block-tridiagonal structure [19]. For the convenience of solving, we write the SLAE (22) in the canonical form:

$$-A_i y_{i-1} + C_i y_i - B_i y_{i+1} = F_i, \quad 1 \leq i \leq N - 1, \tag{48}$$

$$\begin{aligned} A_i = & \begin{pmatrix} a_{i,0,0} & a_{i,0,1} \\ a_{i,1,0} & a_{i,1,1} \end{pmatrix}, B_i = \begin{pmatrix} b_{i,0,0} & b_{i,0,1} \\ b_{i,1,0} & b_{i,1,1} \end{pmatrix}, C_i = \begin{pmatrix} c_{i,0,0} & c_{i,0,1} \\ c_{i,1,0} & c_{i,1,1} \end{pmatrix}, F_i = \begin{pmatrix} F_{i,0} \\ F_{i,1} \end{pmatrix}, y_i = \begin{pmatrix} y_{i,0} \\ y_{i,1} \end{pmatrix}; \\ & C_0 y_0 - B_0 y_1 = F_0, \quad C_N y_N - A_N y_{N-1} = F_N. \end{aligned}$$

Here, A, B, C are the square matrices (blocks) of dimension 2 by 2 (the first line is the piezoconductive part, the second line is the energy part, numbering from 0); y are the desired pressure δP and temperature δT , and F are the vectors of dimension 2×1 .

To solve System (48), we used the matrix sweep method, which is similar to the sweep method for scalar three-point equations [19]. To implement the method, operations on matrices are required: addition, multiplication and transposition, which were implemented as separate functions. The matrix sweep method itself is moved to a separate function, because during the calculation, it will be accessed on each layer in time (iteration). The solution consists of two steps, namely moving forward and moving backward:

$$\begin{aligned} \alpha_0 = & C_0^{-1} B_0, \beta_0 = C_0^{-1} F_0, \quad \alpha_i = (C_i - A_i \alpha_{i-1})^{-1} B_i, \tag{49} \\ \beta_i = & (C_i - A_i \alpha_{i-1})^{-1} (F_i + A_i \beta_{i-1}), \quad i = 1, 2, \dots, N; \\ y_N = & \beta_N, \quad y_i = \alpha_i y_{i+1} + \beta_i, \quad i = N - 1, N - 2, \dots, 0; \\ \alpha_i = & \begin{pmatrix} \alpha_{i,0,0} & \alpha_{i,0,1} \\ \alpha_{i,1,0} & \alpha_{i,1,1} \end{pmatrix}, \quad \beta_i = \begin{pmatrix} \beta_{i,0} \\ \beta_{i,1} \end{pmatrix}. \end{aligned}$$

The formulated problem is very time-consuming; this is manifested in the modeling of filtration processes in long reservoirs. To do this, it is necessary to use grids with a large number of nodes, as a result of which the computation time increases significantly. To solve this problem, parallelization of the algorithm is used.

The matrix sweep parallelization algorithm is based on the well-known scalar sweep parallelization algorithm, and is implemented in program language C using the MPI standard [21]. But it is worth noting that, compared with scalar sweep, the coefficients of

the equation in this case are matrices and vectors, respectively; it is necessary to allocate more memory for variables with the help of which the collective interaction of processes takes place.

To describe the parallel algorithm, we introduced a uniform partition of the set of grid node numbers $\Omega = 0, 1, \dots, N$ into adjacent non-intersecting subsets ($m = 0, \dots, p - 1$ is the logical number of the MPI process).

One of the stages of parallelizing the sweep is the solution of a short problem:

$$-\tilde{A}_i y_{i-1} + \tilde{C}_i y_i - \tilde{B}_i y_{i+1} = \tilde{F}_i, i \in \tilde{\Omega}, \quad (50)$$

where the index $i \pm 1$ is understood as the transition to the corresponding neighboring element from the set $\tilde{\Omega} = \{i_2^{(0)}, i_1^{(1)}, i_2^{(1)}, i_1^{(2)}, \dots, i_1^{(p-2)}, i_2^{(p-2)}, i_1^{(p-1)}\}$. $\tilde{A}, \tilde{B}, \tilde{C}, \tilde{F}$ are the coefficients of the equation, which are found from the coefficients of Equation (48) and the basis functions.

To solve a short problem, all MPI processes carry out a collective exchange of coefficients; the details of the algorithm are presented in [15]. This algorithm is implemented using the MPI_Allreduce function, the parameters of which are presented below.

For a scalar sweep, the call to this function has the following form:

MPI_Allreduce(dd,ee,4*ncp,MPI_DOUBLE,MPI_SUM,MPI_COMM_WORLD), where the function parameters are as follows: dd is the address of the beginning of the input buffer; ee is the address of the beginning of the receive buffer; 4*ncp is the number of elements in the input buffer; MPI_DOUBLE is the type of elements in the input buffer; MPI_SUM is the operation by which the reduction is performed; and MPI_COMM_WORLD is the communicator.

For a matrix sweep with a matrix dimension of 2×2 , the call to this function has the following form:

MPI_Allreduce(dd,ee,14*ncp,MPI_DOUBLE,MPI_SUM,MPI_COMM_WORLD).

Here, in the arrays dd and ee, there are three matrices, each with four elements, and one vector with two elements. The number of elements (A, B, C, F) from Equation (48) is 14.

Each internal process needs 14×2 (left and right borders) = 28 elements of the dd array. The zero and last processes have 14 elements, respectively, i.e., the number of elements in the send buffer changes.

The calculations were performed on the K100 hybrid supercomputer at the Center of Collective Usage of KIAM RAS [22].

To check the quality of parallelization with a change in the number of processes, a number of calculations were carried out for a small number of calculated time steps. For example, as a result of running the program on a grid with 1000 spatial steps and 70,000 time steps in serial mode, the calculation time is 25 s; in parallel mode on 16 processes, the calculation time is 2.8 s, while the parallelization efficiency is 0.558. This means that the overhead in the parallel variant is slightly less than half the estimated time. However, this is offset by an acceleration of almost nine times.

In the case of a multidimensional boundary value problem in rectangular domains, it is possible to effectively use the parallel sweep algorithm at the stages of implementation of iterative or time schemes with a factorized operator [23,24]. The examples are solving the multidimensional Poisson equation by the alternating direction method or solving the multidimensional heat equation using locally one-dimensional difference schemes.

5. Results

Based on the developed module, a series of computational calculations was carried out. We note that the input parameters are taken and varied on the basis of field studies at wells and various carbonate deposits in Russia. To verify the model, the calculated data were compared with the field data.

The following condition was considered for the formulation under consideration. The well has been idle for a long time. It is assumed that the pressure and temperature in the

formation around the well have recovered to the initial values: $P_0^f = 36$ MPa, $P_0^m = 36$ MPa, $T_0 = 96$ °C. Wells in the environment are not affected. Then, the well in question is put into operation with a constant bottomhole pressure $P_{wf}^f = P_{wf}^m = 28$ MPa. At the moment of start-up, a depression funnel is formed in the reservoir and fluid begins to flow into the well. It is necessary to investigate the behavior of pressure and temperature around the operating well at various permeability values. The following parameters were used in the calculation: $P_0^f = 36$ MPa, $P_0^m = 36$ MPa, $P_{wf}^f = 28$ MPa, $P_{wf}^m = 28$ MPa, $\phi^f = 0.01$, $\phi^m = 0.1$, $\rho_o = 740$ kg/m³, $\rho_w = 1118$ kg/m³, $S_w^f = 0.4$, $S_w^m = 0.4$, $\mu_w^f = \mu_w^m = 0.67 \times 10^{-3}$ Pa · s, $\mu_o^f = \mu_o^m = 0.86 \times 10^{-3}$ Pa · s, $\sigma = 0.6$ 1/m², $k_{rw} = 0.03(S_w)^2 + 0.002(S_w) + 0.0002$, $k_{ro} = 7.7(S_w)^4 - 12.07(S_w)^3 + 6.9(S_w)^2 - 1.8(S_w) + 0.2$, $k^f = 1 \times 10^{-12}$ m², $k^m = 1 \times 10^{-15}$ m².

The computational domain starts at a distance of 0.1 m from the borehole axis, and ends at a distance of 1 m from this axis. The dimensions of the computational grid are determined by the constant spatial step $h = 0.0009$ m and the total number of steps $N = 1000$, which is 0.9 m. The initial time step is set equal to $\tau = 0.1$, and varies depending on the number of iterations.

The change in permeability has the greatest effect on the pressure and temperature distribution front. Therefore, in the calculations, we will vary the values of absolute permeability in fractures. Figure 2 shows the dynamics of the pressure stabilization curves near the well for three absolute permeabilities, corresponding to 1×10^{-12} m², 1×10^{-13} m², 1×10^{-15} m². We note that under given conditions with permeability, the pressure drops faster. In this regard, for reservoirs with high permeability $k^f = 1 \times 10^{-12}$ m², it is necessary to select the characteristics of the well operation so that the pressure sinks more slowly over a short period of operation, otherwise this will lead to rapid depletion of the reservoir.

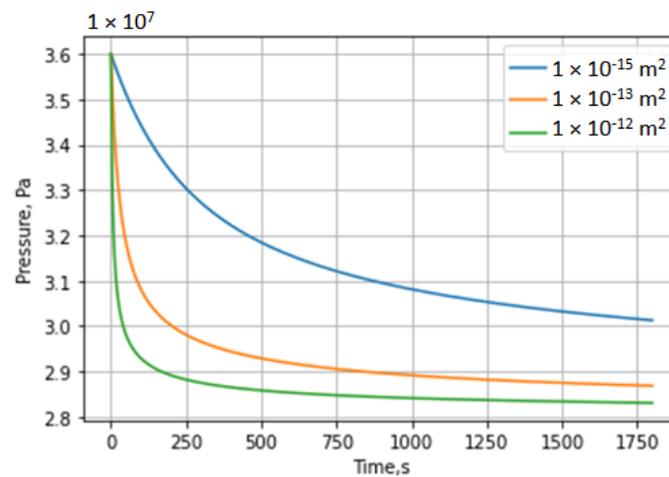


Figure 2. Pressure curves for different absolute permeabilities in fractures.

Figure 3 shows the dependence of pressure on the radius for various values of absolute permeability in fractures at the final time. Based on these calculations, it is possible to determine the area of pressure distribution for various time values. We note that at high permeability, the pressure distribution has a larger radius around the production well, which indicates a larger drainage zone.

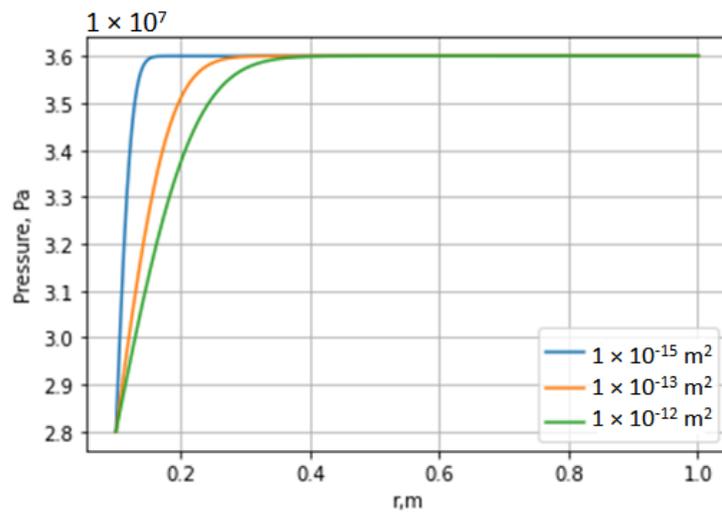


Figure 3. Pressure curves for different absolute permeabilities in fractures at the end time.

Figures 4 and 5 show temperature versus time and distance, respectively. Note that the rate of temperature propagation in comparison with pressure is negligible. However, there is a difference. The radius of change in the temperature front will be slower than with pressure.

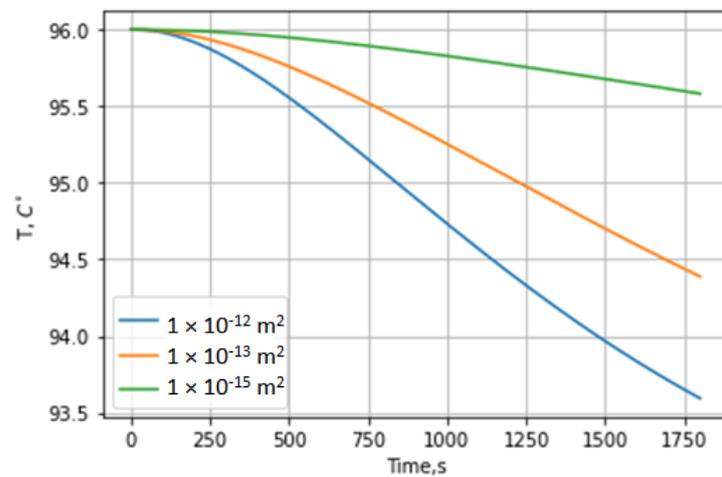


Figure 4. Dynamics of temperature curves for different absolute permeabilities in fractures.

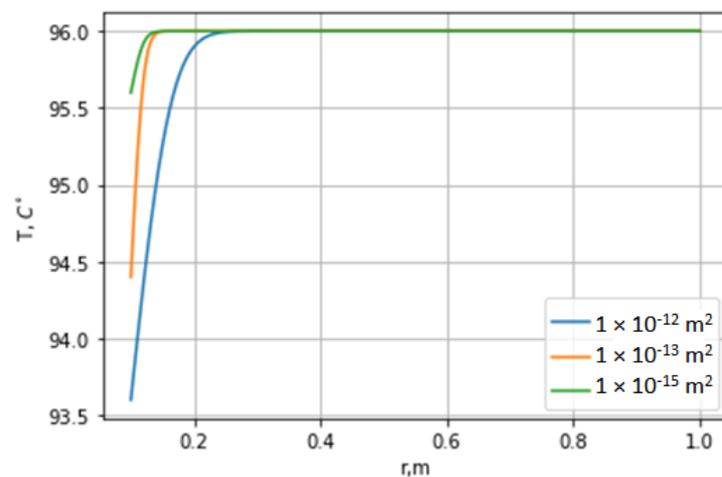


Figure 5. Temperature curves for different absolute permeabilities in fractures.

To study the properties of the matrix sweep parallelization algorithm and compare it with a sequential code, we use such characteristics as speedup coefficients (S_m) and efficiency coefficients (E_m).

$$S_m = \frac{T_1}{T_m}, \quad (51)$$

$$E_m = \frac{S_m}{m} = \frac{T_1}{m \cdot T_m}, \quad (52)$$

where S_m is the speedup, E_m is the efficiency, T_m is the execution time of the parallel program on m processes, and T_1 is the execution time of the sequential program.

Table 1 shows the speedup and efficiency factors depending on the number of processes. We see that the speedup grows, and the efficiency decreases moderately.

Table 1. The speedup and efficiency of the parallel algorithm.

Number m of Processes	Speedup S_m	Efficiency E_m
1	1.0000	1.0000
4	3.3276	0.8319
8	4.8764	0.6095
10	5.9349	0.5934
12	7.1202	0.5933
14	7.9783	0.5699
16	8.9286	0.5580

The analysis of Table 1 showed that the matrix sweep algorithm accelerates the calculation time, which confirms the validity of using multiprocessor calculations for the problem under consideration.

6. Conclusions

The mathematical model of the mass transfer process taking into account the non-isothermal nature of a two-phase fluid in a fractured-porous reservoir based on a dual porosity model was developed. Based on the algorithm of splitting by physical processes, a difference scheme with time weights was constructed, which ensured the correctness and consistency of fluxes between the system of natural fractures and the pore reservoir. For the numerical solution of the problem, an original implicit finite-difference scheme on a spatial grid was proposed. The resulting system of equations was solved using a parallel matrix sweep algorithm. A series of computational experiments was carried out, the results of which confirmed the effectiveness of the developed numerical algorithm and its parallel implementation. Let us pay attention to the advantages of the proposed algorithm in comparison with a completely implicit scheme: three systems of equations of a smaller dimension were solved, instead of one. In all calculations, it was found that such an approach, when the system was split by physical processes, and the groups of equations were solved implicitly, ensured the reliability of the calculation in the investigated range of parameters and acceptable speed, which was quite suitable for solving practical problems. The calculated data were compared with field data; the error did not exceed 5%. The coefficients of speedup and efficiency for a different number of processes were presented. The optimal number of processes was obtained for fixed grid parameters. The result of the modeling was the distribution of pressure and temperature in a fractured-porous reservoir near an operating production well with varying permeability values in fractures. It was noted that the higher the permeability in the system, the faster the pressure dropped during the well operation. These calculations will allow us to monitor and regulate the work of wells in order to increase production.

Author Contributions: Conceptualization, Y.O.B., R.M.U. and I.M.G.; methodology, Y.P.; formal analysis, S.V.P.; software, R.M.U. and A.A.M.; validation, Y.O.B. and R.M.U.; investigation, V.O.P.; writing—original draft preparation, Y.O.B. and R.M.U.; writing—review and editing, P.I.R.; supervision, Y.P., V.O.P. and S.V.P.; data curation, Y.O.B. and I.M.G. All authors have read and agreed to the published version of the manuscript.

Funding: The work of Bobreneva Yu.O. (conceptualization, validation, original draft preparation, data curation), Uzyanbaev R.M. (conceptualization, software, validation, original draft preparation), Mazitov A.A. (software) and Gubaydullin I.M. (conceptualization, data curation) was supported by the Russian Science Foundation (project №21-71-20047). The URL for information about the project: <https://rscf.ru/en/project/21-71-20047/> (accessed on 2 August 2023). The work of Poveshchenko Yu.A. (methodology, supervision), Podryga V.O. (investigation, supervision), Polyakov S.V. (formal analysis, supervision) and Rahimly P.I. (review and editing) was carried out within the framework of the state assignment of KIAM RAS. Numerical experiments were performed on the hybrid supercomputer K100 installed in the Supercomputer Centre of Collective Usage of KIAM RAS.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations and symbols are used in this manuscript:

CPU	Central Processing Unit
MPI	Message Passing Interface
m	Number of OpenMP threads
S_m	Speedup coefficient
T_m	Execution time

References

- Barenblatt, G.I.; Entov, V.M.; Ryzhik, V.M. *Movement of Fluids and Gases in Natural Reservoirs*; Nedra: Moscow, Russia, 1984; p. 211.
- Chekalyuk, E.B. *Thermodynamics of Oil Reservoir*; Nedra: Moscow, Russia, 1965.
- Kotyakhov, F.I. *Physics of Oil and Gas Reservoirs*; Nedra: Moscow, Russia, 1977.
- Aguilera, R. *Naturally Fractured Reservoirs*; T. Pennwell Corporation: Tulsa, OK, USA, 1980.
- Denk, S.O. *Problems of Fractured Productive Objects*; Electronic Publishing: Perm, Russia, 2004.
- Cholovsky, I.P. *Handbook: Oil and Gas Geologist's Companion*; Nedra: Moscow, Russia, 1989.
- Golf-Racht, T.D. *Fundamentals of Fractured Reservoir Engineering*; Elsevier Scientific Publishing Company: Amsterdam, The Netherlands; Oxford, UK, 1982.
- Paskonov, V.M.; Polezhaev, V.I.; Chudov, L.A. *Numerical Modeling of Heat and Mass Transfer Processes*; Nedra: Moscow, Russia, 1984.
- Aziz, H.; Settari, E. *Mathematical Modeling of Reservoir Systems*; Institute for Computer Research: Moscow-Izhevsk, Russia, 2004.
- Rao, X.; Xin, L.; He, Y.; Fang, X.; Gong, R.; Wang, F.; Zhao, H.; Shi, J.; Xu, Y.; Dai, W. Numerical simulation of two-phase heat and mass transfer in fractured reservoirs based on projection-based embedded discrete fracture model (pEDFM). *J. Pet. Sci. Eng.* **2022**, *208*, 109323. [[CrossRef](#)]
- Vasiliev, V.I.; Vasil'eva, M.V.; Grigoriev, A.V.; Prokopiev G.A. Mathematical modeling of the problem of two-phase filtration in inhomogeneous fractured porous media using the dual porosity model and the finite element method. *Uchenye Zap. Kazan. Univ. Ser. Phys. Math. Sci.* **2018**, *160*, 165–182.
- Bobreneva, Y.O.; Rahimly, P.I.; Poveshchenko, Y.A.; Podryga, V.O.; Enikeeva, L.V. On one method of numerical modeling of piezoconductive processes of a two-phase fluid system in a fractured-porous reservoir. *J. Phys. Conf. Ser.* **2021**, *2131*, 0220021. [[CrossRef](#)]
- Bobreneva, Y.O. Modeling the piezoconductivity process of a two-phase fluid system in a fractured-porous reservoir. *Math. Model. Comput. Simul.* **2022**, *4*, 645–653. [[CrossRef](#)]
- Uzyanbaev, R.; Bobreneva, Y.; Poveshchenko, Y.; Podryga, V.; Polyakov, S. *Modeling of Two-Phase Fluid Flow Processes in a Fractured-Porous Type Reservoir Using Parallel Computations*; Communications in Computer and Information Science; Springer: Cham, Switzerland, 2022; Volume 1618, pp. 276–292.
- Uzyanbaev, R.M.; Poveshchenko, Y.A.; Podryga, V.O.; Polyakov, S.V.; Bobreneva, Y.O.; Gubaydullin, I.M. Analysis of parallel algorithm efficiency for numerical solution of mass transfer problem in fractured-porous reservoir. In *Russian Supercomputing Days*; Springer International Publishing: Cham, Switzerland, 2022; Volume 13708, pp. 33–47.
- Warren, J.E.; Root, P.J. The behavior of naturally fractured reservoirs. *J. Soc. Petrol. Eng.* **1963**, *3*, 245–255. [[CrossRef](#)]
- Marchuk, G.I. *Splitting Methods*; Nauka: Moscow, Russia, 1988.

18. Alekseeva, N.; Podryga, V.; Rahimly, P.; Coffin, R.; Pecher, I. Mathematical modeling of gas hydrates dissociation in porous media with water-ice phase transformations using differential constrains. *Mathematics* **2022**, *10*, 3470. [[CrossRef](#)]
19. Samarskii, A.A. *Introduction to the Theory of Difference Schemes*; Nauka: Moscow, Russia, 1971.
20. Samarskii, A.A.; Nikolaev, E.S. *Numerical Methods for Grid Equations*; Nauka: Moscow, Russia, 1978; p. 592.
21. MPI. A Message-Passing Interface Standard. 2023. Available online: <http://www.mpi-forum.org/docs> (accessed on 2 August 2023).
22. Center for Collective Usage of the Keldysh Institute of Applied Mathematics of RAS. Available online: <http://ckp.kiam.ru> (accessed on 2 August 2023).
23. Chekalin, A.N.; Shevchenko, V.A. Numerical study of some difference schemes on a model problem of two-phase plane filtration. In *Proceedings of the III All-Union Seminar "Numerical Solution of Problems of Multiphase Incompressible Fluid Filtration"*; Konovalov, A.N., Ed.; Computing Center of the Siberian Branch of the USSR Academy of Sciences: Novosibirsk, Russia, 1977; pp. 214–226.
24. Akimova, E.N. Parallelization of the matrix sweep algorithm. *Mat. Model.* **1994**, *6*, 61–67.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.