doi 10.26089/NumMet.v21r212

SOFTWARE SUITE FOR NUMERICAL SIMULATION OF THERMOPOROELASTIC MEDIUM WITH DAMAGE

A. S. Meretin[1]

In this paper we consider a software suite for numerical simulation of thermoporoelastic medium evolution with damage. The model is a modification of the Biot model for thermoporoelastic medium evolution and can be used for simulation of stress-strain medium behavior, fluid flow, non-isothermal effects and damage. Damage of the medium is simulated within the framework of continuum damage mechanics where state of the medium is described by a damage parameter. This parameter describes degree of medium destruction. Evolution of the parameter is defined by kinetic equation. The computational algorithm is based on the finite element method. Taylor-Hood finite elements with second order of displacements approximation and first order of pressure and temperature approximation are used. The system of equations is solved by using "monolithic" approach without iteration coupling between groups of equations. Results of rock damage evolution due to thermal treatment are presented.

Keywords: thermoporoelasticity, Biot model, damage, thermodynamic consistency principle, finite element method

1. Introduction. Studying the damage occurring in materials under thermomechanical stress is a highly relevant field of research in various areas of physics, which has an important applied value in metallurgy, construction, hydrocarbon production and other industries. In particular, one of the top priority tasks in oil engineering is the development of various techniques to increase oil recovery in low-permeability reservoirs. One such technique is thermal recovery, in which a thermal fluid is pumped into the reservoir to increase oil mobility. The thermal fluid is pumped under high pressure and has a much higher temperature than the reservoir. This results in various processes occurring in the reservoir as a result of injection, such as rock deformation, filtration and changes in physical and chemical properties of the reservoir fluids, changes in the temperature field, and rock damage. Modeling the thermal recovery process, therefore, requires proper accounting of the interaction between all the effects mentioned above.

There are currently a number of models describing the evolution of poroelastic medium taking into account its damage. The main disadvantages of these models are that some of them are not generally appropriate from the thermodynamic point of view [1, 2], while others [3] are more of a theoretical nature and are not suitable for applied calculations.

This article presents a thermodynamically consistent mathematical model to describe the evolution of thermoporoelastic medium taking into account deformation, fluid flow and non-isothermal processes, as well as the damage of the medium. A computational algorithm is described for the model, which is implemented as a software package in the *C++* language, and examples of applied computations are provided.

2. Mathematical model. One of the classical models used commonly to describe the evolution of a poroelastic medium is the Biot model [4]. In this model, the medium is represented as a combination of two interrelated continuums: deformable solid phase ("skeleton") and mobile liquid phase ("fluid").

In its classical formulation, the system of Biot equations consists of the mass conservation law, which describes the filtration of fluid in a porous medium, and the momentum conservation law, which describes changes in the stress-strain state. In the case of thermo-poroelastic medium, simulating non-isothermal effects requires introducing an additional energy balance equation in the system. These equations express the fundamental laws of continuous medium mechanics. To describe the features of specific media, the system of equations must be supplemented with constitutive relations that impose restrictions on the material behavior under external

[1] Moscow Institute of Physics and Technology (State University), 141701, Moscow Region, Dolgoprudny, 9 Institutskiy per., Russia, e-mail: meretin.as@cet-mipt.ru

stresses. A number of axioms [5] exist that describe the basic principles of building the constitutive relations theory. One of these basic principles is the principle of thermodynamic consistency, which states that constitutive relations must be compliant with the second law of thermodynamics for any sequence of process states.

To simulate the medium damage, it is necessary to expand the Biot model by introducing additional assumptions about the process of damage evolution. There are two main approaches to mathematical description of processes occurring during material damage. The first approach [6] considers damage as the development of a finite set of large-scale cracks. Each crack is assumed to have certain boundaries and to grow when the fracture criteria are met. This approach is commonly used in practice to describe the development of a finite (small) number of isolated cracks, for example, in simulating the hydrofracturing process in oil and gas fields.

The second approach used in this paper is based on continuum damage mechanics [7, 8]. In this approach, damage is seen as a reduction in the "effective" strength properties of the material, caused by the development of numerous micro-fractures and pores. It is assumed that the "degree of damage" of the material is described by an additional parameter (usually of tensor nature), called the damage parameter. The evolution of this parameter is determined by the kinetic equation given, which describes the change in the damage parameter depending on the current state of the medium. A detailed review of the existing models of damage parameter evolution is given in [9].

The mathematical model described in detail in [9] is used below. Let's consider a representative elementary volume Ω with the boundary $\partial\Omega$, consisting of two continuums: a deformable skeleton ("s") and a single-phase weakly compressible fluid ("f"). Assuming that spatial displacements are small and the influence of external and inertial forces is negligible, the system of equations in the model is recorded in the form

$$\begin{cases} \dfrac{\partial m_f}{\partial t} + \operatorname{div}(\rho_f \boldsymbol{w}) = 0, \\ \operatorname{div}\boldsymbol{\sigma} = 0, \\ \dfrac{\partial\left(m_s e_s + m_f e_f\right)}{\partial t} + \operatorname{div}\left(\rho_f e_f \boldsymbol{w}\right) = \operatorname{div}\left(-p\boldsymbol{w}\right) - \operatorname{div}(\boldsymbol{q}), \\ \boldsymbol{w} = -\dfrac{\boldsymbol{k}}{\mu}\operatorname{grad}(p), \\ \boldsymbol{q} = -\boldsymbol{\kappa}\operatorname{grad}(T), \\ D = D(\chi), \end{cases} \tag{1}$$

where m_f is the mass of the fluid in the elementary volume of the medium, ρ_f is the fluid density, $\boldsymbol{w}$ is the flow rate, $\boldsymbol{\sigma}$ is the full stress tensor, e_α is the specific internal energy of phase α, $\boldsymbol{q}$ is the total heat flow density, $\boldsymbol{k}$ is the positively determined permeability tensor, μ is the fluid viscosity, $\boldsymbol{\kappa}$ is the positively determined heat conductivity tensor, T is the temperature.

The last equation in system (1) describes the evolution of damage parameter D (in this paper D is assumed to be a scalar value) as a function of the current state of medium χ (instantaneous damage kinetics). In general, the kinetic equation for the damage rate parameter can be presented using the finite damage kinetics in the form of $\tau\dfrac{dD}{dt} = F(\chi)$, where τ is the relaxation time, F is a certain function dependent on the state of the medium [3]. The specific type of dependency used in this article will be described below.

The primary variables for the system of equations (1) are $\chi = \{\boldsymbol{\varepsilon}, p, T, D\}$. The constitutive relations are used to complete the system of equations (1). When deriving the constitutive relations, the main criteria limiting their formula is the compliance with the thermodynamic consistency principle. The method of deriving the constitutive relations is presented in detail in [9]. In this paper, the constitutive relations are presented in their final form, as used further in the numerical algorithm:

$$\begin{aligned} \Delta\boldsymbol{\sigma} &= \boldsymbol{C}(1-D) : \Delta\boldsymbol{\varepsilon} - \boldsymbol{b}\Delta p - \boldsymbol{C} : \boldsymbol{\alpha}_T \Delta T - \boldsymbol{C} : \varepsilon^0 : \Delta D, \\ \Delta m_f &= \rho_f \boldsymbol{b} : \Delta\boldsymbol{\varepsilon} + \rho_f \frac{\Delta p}{M} - \alpha_m \rho_f \Delta T, \\ \Delta\frac{1}{\rho_f} &= -\frac{1}{\rho_f}\frac{1}{K_f}\Delta p + \frac{1}{\rho_f}\alpha_f \Delta T. \end{aligned} \tag{2}$$

Here $\boldsymbol{C}$ is the 4th rank elastic coefficient tensor, $\boldsymbol{b}$ is the Biot coefficient, $\boldsymbol{C} : \alpha_T$ is the thermoelastic coefficient tensor, M is the Biot modulus, α_m is the thermal expansion coefficient, $C_{p\alpha}$ is the thermal capacity

of phase α, K_f is the fluid bulk modulus, $\boldsymbol{\varepsilon} = 1/2\left(\operatorname{grad}\boldsymbol{\xi} + \operatorname{grad}\boldsymbol{\xi}^T\right)$ is the small-deformation tensor, $\boldsymbol{\xi}$ is the skeleton displacement vector, f^0 is a certain reference state of the parameter f, and $\Delta f = f - f^0$.

The system of constitutive relations (2) describes the dependence of the corresponding values on the state of the medium χ. For example, the first expression describes the change in combined stress upon deformation of the skeleton (first summation term), the change in pore pressure (second term), temperature (third term), and medium damage (fourth term). The expression for the internal energy of the skeleton is

$$\Delta E_s = \Delta E_{s\boldsymbol{\varepsilon}} + \Delta E_{sp} + \Delta E_{sT} + \Delta E_{sD},$$

where

$$\begin{aligned}
\Delta E_{s\boldsymbol{\varepsilon}} &= \left[\boldsymbol{\sigma}^0 + \frac{1}{2}\boldsymbol{C} : \Delta\boldsymbol{\varepsilon} + p^0\boldsymbol{b} + T^0\boldsymbol{C} : \boldsymbol{\alpha}_T\right]\Delta\boldsymbol{\varepsilon},\\
\Delta E_{sp} &= \left[\frac{1}{N}\left(p - \frac{1}{2}\Delta p\right) - \alpha_\varphi\left(T - \frac{1}{2}\Delta T\right)\right]\Delta p,\\
\Delta E_{sT} &= \left[-\alpha_\varphi\left(p - \frac{1}{2}\Delta p\right) + \frac{C_{ps}}{T^0}\left(T - \frac{1}{2}\Delta T\right)\right]\Delta T,\\
\Delta E_{sD} &= -\left[Y^0 + \boldsymbol{C}\left(\varepsilon^0 + \frac{1}{2}\Delta\boldsymbol{\varepsilon}\right)\Delta\boldsymbol{\varepsilon}\right]\Delta D.
\end{aligned} \tag{3}$$

The expression for the internal energy of the fluid is

$$\Delta E_f = \left[\frac{\phi}{K_f}\left(p - \frac{1}{2}\Delta p\right) - \phi\alpha_f\left(T - \frac{1}{2}\Delta T\right)\right]\Delta p - \left[\phi\alpha_f\left(p - \frac{1}{2}\Delta p\right) - \frac{C_{pf}}{T^0}\left(T - \frac{1}{2}\Delta T\right)\right]\Delta T.$$

3. Computational algorithm. The system of equations (1) represents a bound problem for an elliptic equation (law of momentum conservation) and two parabolic equations (mass and energy conservation laws). In practice, various methods are used to solve this class of problems, such as finite volume method, finite element method and boundary integral equation method. In this case, the equations themselves can be solved both jointly and iteratively. In the latter case, convergence of the iteration process depends directly on the choice of the equation binding algorithm [10].

In this paper, the finite element method is used to solve the system of equations (1), and the equations are solved jointly. The key unknowns are displacement $\boldsymbol{\xi}$, pore pressure p, and temperature T. Tetrahedral Taylor-Hood elements [11] were chosen as finite element type, which have the second order of approximation by displacement and the first order by pressure and temperature. This type of finite elements ensures stability of the solution to poroelasticity problems at the expense of satisfying the Ladyzhenskaya–Babuška–Brezzi condition [12].

Let's consider the weak formulation of the problem. Suppose an area V_α of sufficiently smooth (vector) functions is defined within region Ω, where $\boldsymbol{\xi} \in V_{\boldsymbol{\xi}}, p \in V_p, T \in V_T$, and the Dirichlet conditions ($\partial\Omega_D$) or Neumann conditions ($\partial\Omega_N$) are defined for each parameter at the boundary $\partial\Omega$ of the area. Let us introduce a set of test functions $\boldsymbol{v}_\alpha$ ($\alpha = \{\boldsymbol{\xi}, p, T\}$) within the area $V_\alpha^0 \subset V$, where $V_\alpha^0 = \{\boldsymbol{v}_\alpha \in V_\alpha : \boldsymbol{v}_\alpha|_{\partial\Omega_D} = 0\}$. By multiplying each equation in the system (1) by its test function on the left, we arrive to

$$\begin{aligned}
&\int\limits_\Omega \boldsymbol{v}_{\boldsymbol{\xi}}^T \operatorname{div}\boldsymbol{\sigma}\, d\Omega = 0,\\
&\int\limits_\Omega v_p\left[\frac{\partial m_f}{\partial t} + \operatorname{div}(\rho_f\boldsymbol{w})\right]d\Omega = 0,\\
&\int\limits_\Omega v_T\left[\frac{\partial E_s}{\partial t} + \frac{\partial E_f}{\partial t} + \operatorname{div}\left(\frac{1}{\phi}E_f\boldsymbol{w}\right) + \operatorname{div}(p\boldsymbol{w}) + \operatorname{div}(\boldsymbol{q}_T)\right]d\Omega = 0.
\end{aligned} \tag{4}$$

Applying the Green formula to the system of equations (4), we get the weak formulation of the equations:

$$\begin{gathered}\int\limits_{\Omega}(\boldsymbol{L}\boldsymbol{v}_{\boldsymbol{\xi}})^{T}\boldsymbol{\sigma}d\Omega-\int\limits_{\partial\Omega_{N}}\boldsymbol{v}_{\boldsymbol{\xi}}^{T}\tilde{\boldsymbol{t}}dS=0,\\ \int\limits_{\Omega}v_{p}\frac{\partial m_{f}}{\partial t}d\Omega-\int\limits_{\Omega}(\operatorname{grad}v_{p})^{T}\boldsymbol{w}d\Omega+\int\limits_{\partial\Omega_{N}}v_{p}\tilde{q}dS=0,\\ \int\limits_{\Omega}v_{T}\left[\frac{\partial E_{s}}{\partial t}+\frac{\partial E_{f}}{\partial t}\right]d\Omega-\int\limits_{\Omega}(\operatorname{grad}v_{T})^{T}\frac{1}{\phi}E_{f}\boldsymbol{w}d\Omega-\int\limits_{\Omega}(\operatorname{grad}v_{T})^{T}p\boldsymbol{w}d\Omega-\int\limits_{\Omega}(\operatorname{grad}v_{T})^{T}\boldsymbol{q}d\Omega+\\ +\int\limits_{\partial\Omega_{N}}v_{T}\frac{1}{\phi}E_{f}\tilde{q}dS+\int\limits_{\partial\Omega_{N}}v_{T}p\tilde{q}dS+\int\limits_{\partial\Omega_{N}}v_{T}q_{T}dS=0,\end{gathered}\tag{5}$$

where $\tilde{\boldsymbol{t}},\tilde{q},\tilde{q_{T}}$ are the given stress vector, fluid flow and heat flow across the boundary $\partial\Omega_{N}$, $\boldsymbol{L}$ is the differential operator of the form

$$\boldsymbol{L}=\begin{bmatrix}\dfrac{\partial}{\partial x}&0&0&0&\dfrac{\partial}{\partial z}&\dfrac{\partial}{\partial y}\\ 0&\dfrac{\partial}{\partial y}&0&\dfrac{\partial}{\partial z}&0&\dfrac{\partial}{\partial x}\\ 0&0&\dfrac{\partial}{\partial z}&\dfrac{\partial}{\partial y}&\dfrac{\partial}{\partial x}&0\end{bmatrix}^{T},$$

and strain and stress tensors are represented in Voigt vector notation:

$$\boldsymbol{\varepsilon}=\left[\varepsilon_{x},\varepsilon_{y},\varepsilon_{z},\gamma_{yz},\gamma_{xz},\gamma_{xy}\right]^{T},\quad\gamma_{ij}=2\varepsilon_{ij},\quad\boldsymbol{\sigma}=\left[\sigma_{x},\sigma_{y},\sigma_{z},\sigma_{yz},\sigma_{xz},\sigma_{xy}\right]^{T}.$$

Suppose the initial stress tensor $\boldsymbol{\sigma}^{0}$ is constant for the entire volume and the initial strain $\boldsymbol{\varepsilon}^{0}$ equals zero. Then, after substituting all constitutive relations and differentiating the first equation we arrive from (5) to

$$\begin{gathered}\int\limits_{\Omega}(\boldsymbol{L}\boldsymbol{v}_{\boldsymbol{\xi}})^{T}:\boldsymbol{C}(1-D):\frac{\partial\boldsymbol{\varepsilon}(\boldsymbol{\xi})}{\partial t}d\Omega-\int\limits_{\Omega}(\boldsymbol{L}\boldsymbol{v}_{\boldsymbol{\xi}})^{T}\boldsymbol{b}\frac{\partial p}{\partial t}d\Omega-\int\limits_{\Omega}(\boldsymbol{L}\boldsymbol{v}_{\boldsymbol{\xi}})^{T}:\boldsymbol{C}:\boldsymbol{\alpha}_{T}\frac{\partial T}{\partial t}d\Omega=\int\limits_{\partial\Omega_{N}}\boldsymbol{v}_{\boldsymbol{\xi}}^{T}\frac{\partial\tilde{\boldsymbol{t}}}{\partial t}dS,\\ \int\limits_{\Omega}v_{p}\boldsymbol{b}:\boldsymbol{L}\frac{\partial\boldsymbol{\xi}}{\partial t}d\Omega+\int\limits_{\Omega}v_{p}\frac{1}{M}\frac{\partial p}{\partial t}d\Omega-\int\limits_{\Omega}v_{p}\alpha_{m}\frac{\partial T}{\partial t}d\Omega+\int\limits_{\Omega}(\operatorname{grad}v_{p})^{T}\frac{\boldsymbol{k}}{\mu}\operatorname{grad}(p)d\Omega=-\int\limits_{\partial\Omega_{N}}v_{p}\tilde{q}d\Omega,\\ \int\limits_{\Omega}v_{T}\left[\boldsymbol{\sigma}^{0}+\boldsymbol{C}(1-D):\boldsymbol{\varepsilon}+p^{0}\boldsymbol{b}+T^{0}\boldsymbol{C}:\boldsymbol{\alpha}_{T}\right]\frac{\partial\boldsymbol{\varepsilon}}{\partial t}d\Omega+\int\limits_{\Omega}v_{T}\left[\frac{1}{M}pd\Omega-\alpha_{m}T\right]\frac{\partial p}{\partial t}d\Omega+\\ +\int\limits_{\Omega}v_{T}\left[-\alpha_{m}p+\frac{C_{ps}+C_{pf}}{T^{0}}T\right]\frac{\partial T}{\partial t}d\Omega-\int\limits_{\Omega}v_{T}\left[\frac{1}{2}\frac{dD}{dt}\boldsymbol{C}:\boldsymbol{\varepsilon}\right]\boldsymbol{\varepsilon}d\Omega+\\ +\int\limits_{\Omega}(\operatorname{grad}v_{T})^{T}\left\{\frac{1}{\phi}E_{f}^{0}+\left[\frac{1}{K_{f}}p-\alpha_{f}T\right]\Delta p-\left[\alpha_{f}p-\frac{C_{pf}}{\boldsymbol{\phi}}\right]\Delta T\right\}\frac{\boldsymbol{k}}{\mu}\operatorname{grad}\boldsymbol{\phi}_{p}d\Omega+\\ +\int\limits_{\Omega}(\operatorname{grad}v_{T})^{T}p\frac{\boldsymbol{k}}{\mu}\operatorname{grad}pd\Omega-\int\limits_{\Omega}(\operatorname{grad}v_{T})^{T}\boldsymbol{q}_{T}d\Omega=-\int\limits_{\partial\Omega_{N}}v_{T}\rho_{f}e_{f}\tilde{q}dS-\int\limits_{\partial\Omega_{N}}v_{T}p\tilde{q}dS-\int\limits_{\partial\Omega_{N}}v_{T}\tilde{q}_{T}dS.\end{gathered}\tag{6}$$

Let's consider spatial approximation of the system of equations (6). For this purpose, we introduce a set of shape functions $\phi_{i}^{(\alpha)}$, so that given an arbitrary function f, we get

$$f=\sum_{i=1}^{N_{\alpha}}\phi_{i}^{(\alpha)}f_{i},\quad f=\boldsymbol{\xi},p,T.$$

Then, taking into account the type of constitutive relations (2), the system of equations (5) can be written in the matrix form

$$\begin{bmatrix}-\boldsymbol{A}_{\xi\xi}&\boldsymbol{A}_{\xi p}&-\boldsymbol{A}_{\xi T}\\ \boldsymbol{A}_{\xi p}^{T}&\boldsymbol{A}_{pp}&\boldsymbol{A}_{pT}\\ \boldsymbol{A}_{T\xi}&\boldsymbol{A}_{Tp}&\boldsymbol{A}_{TT}\end{bmatrix}\begin{bmatrix}\partial\boldsymbol{\xi}/\partial t\\ \partial p/\partial t\\ \partial T/\partial t\end{bmatrix}+\begin{bmatrix}0&0&0\\ 0&\boldsymbol{B}_{pp}&0\\ \boldsymbol{B}_{T\xi}&\boldsymbol{B}_{Tp}&\boldsymbol{B}_{TT}\end{bmatrix}\begin{bmatrix}\boldsymbol{\xi}\\ p\\ T\end{bmatrix}=\begin{bmatrix}-\partial\boldsymbol{f}_{\xi}/\partial t\\ \boldsymbol{f}_{p}\\ \boldsymbol{f}_{T}\end{bmatrix},\tag{7}$$

where

$$\boldsymbol{A}_{\xi\xi} = \int\limits_{\Omega} \left(\boldsymbol{L}\boldsymbol{\phi}_{\boldsymbol{\xi}}\right)^T \boldsymbol{C}(1-D)\left(\boldsymbol{L}\boldsymbol{\phi}_{\boldsymbol{\xi}}\right) d\Omega, \quad \boldsymbol{A}_{\xi p} = \boldsymbol{A}_{p\xi}^T = \int\limits_{\Omega} \left(\boldsymbol{L}\boldsymbol{\phi}_{\boldsymbol{\xi}}\right)^T \boldsymbol{b}\boldsymbol{\phi}_p d\Omega,$$

$$\boldsymbol{A}_{\xi T} = -\int\limits_{\Omega} \left(\boldsymbol{L}\boldsymbol{\phi}_{\boldsymbol{\xi}}\right)^T \boldsymbol{C}(1-D) : \boldsymbol{\alpha}_T \boldsymbol{\phi}_T d\Omega, \quad \boldsymbol{B}_{pp} = \int\limits_{\Omega} (\operatorname{grad}\boldsymbol{\phi}_p)^T \frac{\boldsymbol{k}}{\mu} (\operatorname{grad}\boldsymbol{\phi}_p) d\Omega, \tag{8}$$

$$\boldsymbol{A}_{pp} = \int\limits_{\Omega} {\boldsymbol{\phi}_p}^T \frac{1}{M} \boldsymbol{\phi}_p d\Omega, \quad \boldsymbol{A}_{pT} = -\int\limits_{\Omega} \boldsymbol{\phi}_p^T \alpha_m \boldsymbol{\phi}_T d\Omega,$$

$$\boldsymbol{A}_{T\xi} = \int\limits_{\Omega} \boldsymbol{\phi}_T^T \left(\boldsymbol{\sigma}^0 + \boldsymbol{C}(1-D) : \boldsymbol{\varepsilon} + p^0 \boldsymbol{b} + T^0 \boldsymbol{C} : \boldsymbol{\alpha}_T\right) \left(\boldsymbol{L}\boldsymbol{\phi}_{\boldsymbol{\xi}}\right) d\Omega,$$

$$\boldsymbol{A}_{Tp} = \int\limits_{\Omega} \boldsymbol{\phi}_T^T \left(\frac{1}{M} p - \alpha_m T\right) \boldsymbol{\phi}_p d\Omega, \quad \boldsymbol{A}_{TT} = \int\limits_{\Omega} \boldsymbol{\phi}_T^T \left[-\alpha_m p + (C_{ps} + C_{pf}) \frac{T}{T^0}\right] \boldsymbol{\phi}_T d\Omega,$$

$$\boldsymbol{B}_{Tp} = \int\limits_{\Omega} \operatorname{grad} {\boldsymbol{\phi}_T}^T \left\{\frac{1}{\phi} E_f^0 + \left[\frac{1}{K_f} p - \alpha_f T\right] \Delta p - \left[\alpha_f p - \frac{C_{pf}}{\boldsymbol{\phi}}\right] \Delta T\right\} \frac{\boldsymbol{k}}{\mu} \operatorname{grad} \boldsymbol{\phi}_p d\Omega,$$

$$\boldsymbol{B}_{T\xi} = \int\limits_{\Omega} \operatorname{grad} {\boldsymbol{\phi}_T}^T \left[\frac{1}{2} \frac{dD}{dt} \boldsymbol{C} : \boldsymbol{\varepsilon}\right] \left(\boldsymbol{L}\boldsymbol{\phi}_{\boldsymbol{\xi}}\right) d\Omega, \quad \boldsymbol{B}_{TT} = \int\limits_{\Omega} \operatorname{grad} {\boldsymbol{\phi}_T}^T \boldsymbol{\kappa} \operatorname{grad} \boldsymbol{\phi}_T d\Omega,$$

$$\boldsymbol{f}_{\xi} = \int\limits_{\partial\Omega_N} \boldsymbol{\phi}_{\boldsymbol{\xi}}^T \tilde{\boldsymbol{t}} dS, \quad \boldsymbol{f}_p = -\int\limits_{\partial\Omega_N} \boldsymbol{\phi}_p^T \tilde{q} d\Omega, \quad \boldsymbol{f}_T = -\int\limits_{\partial\Omega_N} \boldsymbol{\phi}_T^T \rho_f e_f \tilde{q} dS - \int\limits_{\partial\Omega_N} \boldsymbol{\phi}_T^T p \tilde{q} dS - \int\limits_{\partial\Omega_N} \boldsymbol{\phi}_T^T \tilde{q}_T dS.$$

To approximate the system of equations in time, a fully implicit scheme was used for displacement $\boldsymbol{\xi}$, pressure p and temperature T. Some parameters such as damage D, permeability $\boldsymbol{k}$, fluid viscosity μ_f are functions of the current state of the system, so they were calculated using parameters from the explicit layer. Newton's method was used to solve a nonlinear system of equations. At each Newtonian iteration, the system of linear equations was solved using BiConjugate Gradient Stabilized (BiCGStab) [13]. Incomplete LU factorization with single-level filling (ILU(1)) was used as a preconditioner. A number of approaches were used to ensure stability of the finite problem, such as the diagonalization of mass matrices [14] and rearrangement of rows and columns according to the Cuthill–McKee algorithm [15].

4. The software package. As a rule, if computations need to take into account deformation, flow, non-isothermal effects and damage simultaneously, several software modules are used and connected iteratively. For example, a flow simulator is launched at every step in the computation, and the resulting pore pressure distribution data is sent to the geomechanical simulator.

The computational algorithm described in this work allows considering different effects simultaneously. For practical computations, this algorithm is implemented as a software module written in *C++*. The main components of the module are preprocessor, computational kernel and postprocessor, which are described in detail below.

The software module starts operation by launching the preprocessor unit, which is responsible for reading and preparing input data for the model. The input data is presented in a series of text files, each describing a specific data block in a fixed format. As the first step, the module reads the computational grid. Since the algorithm in use can be applied to computing complex three-dimensional problems, tetrahedrons are used as elements of the computational grid. As was mentioned earlier, the elements have a second-order approximation for displacement and first-order approximation for pressure and temperature. This means that in addition to the 4 main nodes, intermediate nodes need to be defined on the edges of each tetrahedron. For simplicity, these intermediate nodes are set in the centers of the edges. Therefore, each element of the grid has a total of 10 nodes. Their local enumeration is provided in Figure 1.

To define a computational grid uniquely, it is necessary to set the coordinates of all its nodes and enumerate all the tetrahedrons with an index of the nodes they are composed of in a global enumeration. To construct a simple computational grid in this format, we wrote a *MATLAB* script that uses the built-in function to triangulate a given set of points using the Delaunay algorithm. This computational grid generator receives input data on the shape and size of the computational region, as well as the computational grid step (which can be variable); at the output, it generates a set of text files containing information on the coordinates of the

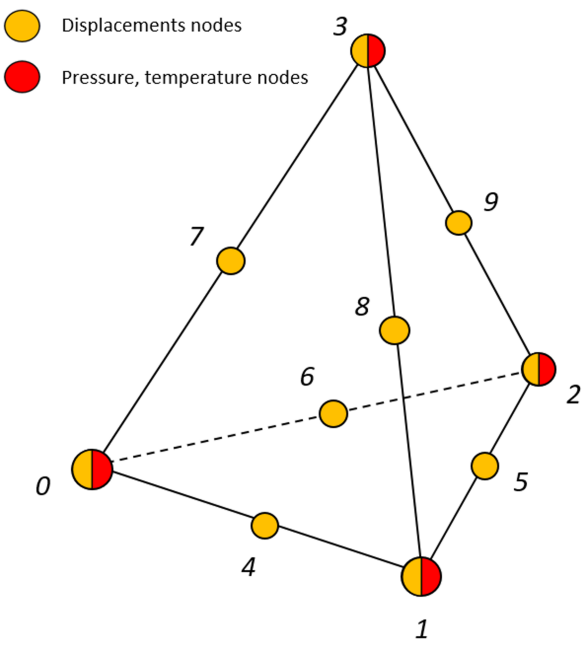

Figure 1. The numbering of tetrahedral element nodes

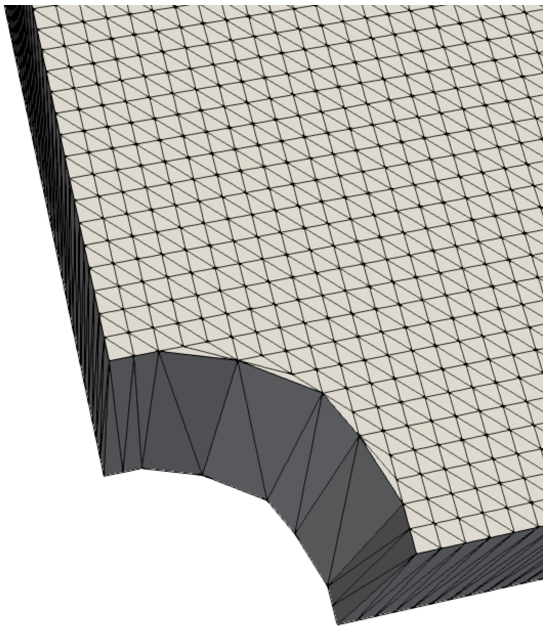

Figure 2. Sample Computational Grid

computational nodes and the list of the nodes making up each cell in the computational grid. An example of the computational grid is shown in Figure 2. Note that the program can work with any other grid generators that accept the data described above as inputs.

A complete description of the computational model, in addition to the grid, must include information on the values of the model parameters in each cell and the complete set of initial and boundary conditions. The model parameters are defined as a data set in the "keyword" — "parameter value" format. The list of key parameters includes geomechanical properties of the rock (Young's Modulus, Poisson's ratio), Biot coefficient, compressibility parameters (Biot modulus for the skeleton and fluid bulk modulus), skeleton and fluid density, reservoir porosity and permeability properties, thermophysical properties (heat capacity, thermal compression ratios), as well as initial pressure conditions. A single value can be assigned to the entire model, or data can be entered for each specific node element.

The software module allows setting boundary values of the first, as well as second kind. Boundary conditions of the first kind are set for individual nodes located at the corresponding boundary in the following format: number of the corresponding variable (displacement vector component, pressure or temperature) within the unknowns vector, and its value. The boundary condition of the second kind is set for the corresponding triangular faces. For each face, the numbers of the nodes shaping that face is specified, as well as the value of the stress vector acting on the face and the values of the mass and heat flow through the face.

After a complete description of all input data for the model, they are sent to the computational kernel, which calculates the solution to the system of equations (7) at each specific moment in time. As mentioned earlier, the system of nonlinear equations is solved using Newton's method. At each Newtonian iteration, the Jacobi matrix is assembled and the right hand side is calculated. The Jacobi Matrix is filled in blocks, by calculating volume integrals (8) for each element of the computational grid. The integrals containing shape functions for displacement are calculated using Gaussian quadrature rules of the second order for tetrahedral elements. In other integrals, the shape functions are of the first order only, so the integral can be calculated analytically. *Eigen* library [16] is used for storing and performing operations on matrices. Since Jacobian is a sparse matrix, the non-zero values of matrix elements are stored in an array of triplets during its assembly, with each triplet containing information about the element value and its position in the matrix.

After assembling the Jacobian and calculating the right hand side, boundary conditions are applied. For each variable for which the boundary condition of the first kind is defined, the corresponding row in the Jacobian and the right part is zeroed out, and the matrix diagonal is set to 1. This means that this component of the vector of unknowns does not increase, i.e. its value at any given moment in time equals the initial value (which

is set in the initial terms of the problem). Surface integrals $f_{\boldsymbol{\xi}}, f_p, f_T$ (8) are calculated for the boundary conditions of the second kind and added to the right hand side.

The condition for convergence of the iterative algorithm is meeting the following conditions simultaneously:

$$||\Delta \boldsymbol{x}||_\infty < \varepsilon_x, \quad ||\boldsymbol{R}||_\infty < \varepsilon_R,$$

where $\Delta \boldsymbol{x}$ is the incremental size of the unknowns at the current iteration, $\boldsymbol{R}$ is the value of non-linear divergence, $\varepsilon_{x,R}$ are the accuracy parameters. If a certain time interval requires more than 5 iterations, the interval is reduced by half, and the iteration algorithm is restarted from the beginning. In case of successful convergence, the algorithm proceeds with the next time interval, gradually increasing the time intervals.

At each Newtonian iteration, a system of linear equations is solved. After preliminary preparation of the system matrix, as described in the previous section, a linear solver is launched. The BiConjugate Gradient Stabilized (BiCGStab) method is used as a solver. It is implemented in the *HYPRE* (Parallel High Performance Preconditioners) library [17] together with ILU(1) preconditioner from the same library. The relative tolerance of the linear solver is set toε_{lin}.

After calculating the time interval, the program recalculates some parameters of the model which explicitly depend on the current state of the medium. The damage parameter is calculated first. The damage parameter is set for the entire element and depends on the condition of the medium. To calculate the damage parameter, all the relevant data is interpolated to the center of the element, after which the value calculated by a formula is assigned to the entire element. Since in many models the damage parameter depends on the primary values of the stress or strain tensor, eigenvalues of the respective matrix are used to calculate the primary values using the *Eigen* library functions.

In addition to damage, the software module developed recalculates permeability of the medium and physical properties of the fluid (such as viscosity). These values are also calculated explicitly at the end of the iteration, by using formulas that will be presented in the next section.

After all computations related to the current time interval are completed, the data is sent to the postprocessor. The postprocessor exports the main results (pressure, temperature, strain, stress, damage parameter, permeability, energy components, etc.) at each moment in time for visualization and analysis. *Paraview* software [18] is used for visualization, which receives a *.vtu* input file with unstructured grid data and the property distribution on the grid. The *VTK* (Visualization Toolkit) library [19] is used to generate this file. The final flowchart of the entire process is provided in Figure 3.

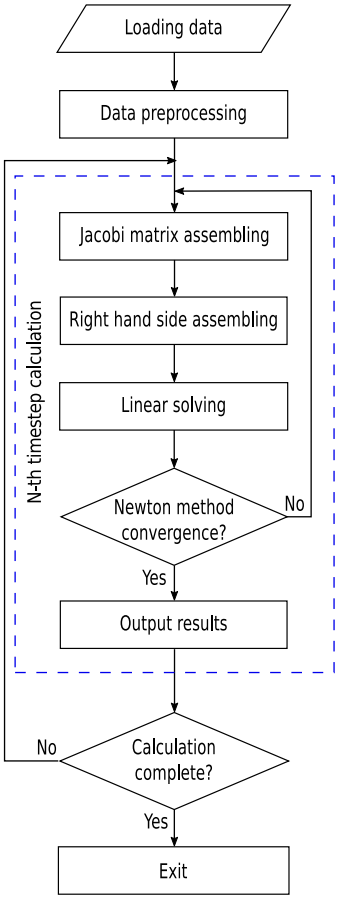

Figure 3. Software module operation flowchart

5. Computation results. In this section we provide an example of the software module work, using the sample problem of thermal impact on a reservoir saturated with fluid. The following effects were taken into account in addition to the standard ones when solving the problem: medium damage, changes in reservoir permeability upon deformation, changes in fluid viscosity depending on pressure and temperature.

The explicit dependence of the damage rate on the rock deformation [20] is used as the formula linking damage to the medium parameters:

$$D = \begin{cases} 0, & \text{if } \tilde{\varepsilon} < \tilde{\varepsilon_c}, \\ \dfrac{D_{\text{off}}}{\tilde{\varepsilon}_{\text{off}} - \tilde{\varepsilon}_c}\tilde{\varepsilon} - D_{\text{off}}\dfrac{\varepsilon_c}{\tilde{\varepsilon}_{\text{off}} - \tilde{\varepsilon}_c}, & \text{if } \tilde{\varepsilon}_c \le \tilde{\varepsilon} \le \tilde{\varepsilon}_{\text{off}}, \\ D_{\text{lim}} - (D_{\text{lim}} - D_{\text{off}})\dfrac{\tilde{\varepsilon}_{\text{off}}}{\tilde{\varepsilon}} & \text{if } \tilde{\varepsilon} > \tilde{\varepsilon}_{\text{off}}, \end{cases} \tag{9}$$

here $\tilde{\varepsilon}$ is calculated using the formula:

$$\tilde{\varepsilon} = \sqrt{\sum_{i=1}^{3} \langle \varepsilon_i \rangle^2}, \quad \langle \varepsilon_i \rangle = \frac{\varepsilon_i + |\varepsilon_i|}{2}.$$

where ε_i are the principal strains.

The dependence [21]

$$k = k_0 \exp\left[-\beta\left(\tilde{\sigma} - \alpha p\right)\right], \tag{10}$$

where k_0 is the initial permeability value, and $\tilde{\sigma} = 1/3(\sigma_x + \sigma_y + \sigma_z)$ is the average stress, is used to simulate the dependence of permeability on the medium state parameters.

The Beggs and Robinson correlation [22], widely utilized in oil engineering, is used to describe the changes in fluid viscosity depending on reservoir conditions:

$$\mu = 10^X - 1, \quad X = 10^Z T^{-1.163}, \quad Z = 3.0324 - 0.02023\gamma_0, \tag{11}$$

where γ_0 is the specific fluid density measured in ${}^0API$ (a dimensionless unit of measurement describing the ratio of fluid density to water density). Formula

$$\gamma_0 = \frac{141.5}{\rho_o/1000 + 131.5},$$

where ρ_o is the fluid density in kg/m^3, is used to convert fluid density from the SI international system to ${}^0API$.

As a specific example of the software application, let's consider the problem of damage zone development near the injection well upon injection of the thermal fluid under pressure exceeding that in the reservoir. This problem is somewhat similar to the problem of spontaneous fracture development during hydrofracturing. Of course, the formulation being considered does not describe the development of a fracture as a standalone object: it requires the use of other models that are qualitatively different from the one under consideration. However, qualitative behavior of the damage area can be reasonably expected to have a number of features typical for "real" hydraulic fractures. In particular, the damage area has a peculiar (flattened) shape and orientation with respect to the principal stress directions. It should be noted also that the purpose of the computations presented below is not to model the dynamics of the damage area in a meaningful applied context. The computations presented below demonstrate the main possibilities of the mathematical model, algorithms and software implementation used.

The model under consideration has $50 \times 50 \times 5$ m dimensions and consists of five layers. An injection well is located at the center of the model, injecting fluid at a constant pressure of 800 bar and temperature of 400^0. The initial reservoir pressure is 200 bar, temperature is 100^0, full (compressive) stresses in directions x, y and z, respectively, are 300, 550 and 700 bar. The reservoir and fluid parameters are presented in the table 1.

Table 1
Input parameter values for the model

Parameter	Value
Young's Modulus, E	20 GPa
Poisson's Ratio, ν	0.3
Biot Modulus, N	10 GPa
Fluid bulk modulus, K_f	3.3 GPa
Biot coefficient, b	0.79
Permeability, k	$1 \cdot 10^{-16}$ m^2
Porosity, φ	0.1
Viscosity, μ	1 mPa $\cdot$ s
Skeleton density, ρ_s	2100 kg/m^3
Fluid density, ρ_s	1000 kg/m^3
Skeleton thermal volume expansion coefficient, α_s	$1 \cdot 10^{-6}$ 1/K
Fluid thermal volume expansion coefficient, α_f	$1 \cdot 10^{-4}$ 1/K
Skeleton specific heat capacity, c_{ps}	1000 J/(kg $\cdot$ K)
Fluid specific heat capacity, c_{pf}	4200 J/(kg $\cdot$ K)
Effective thermal conductivity, κ	2 W/(m $\cdot$ K)

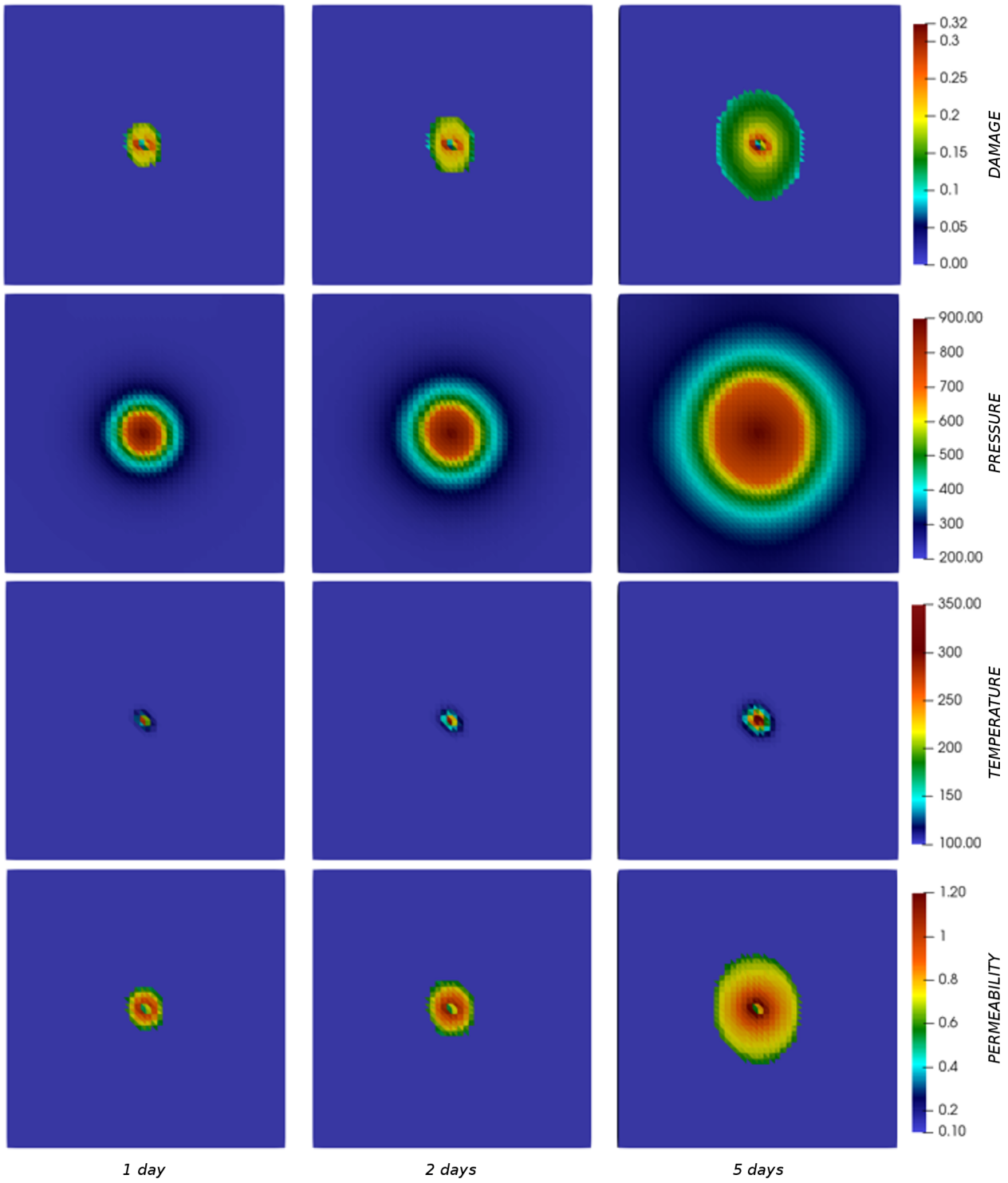

Figure 4. Distribution of damage parameter D (row 1), pressure p (row 2), temperature T (row 3), permeability k (row 4) after 1 day (left), 2 days (center) and 5 days (right)

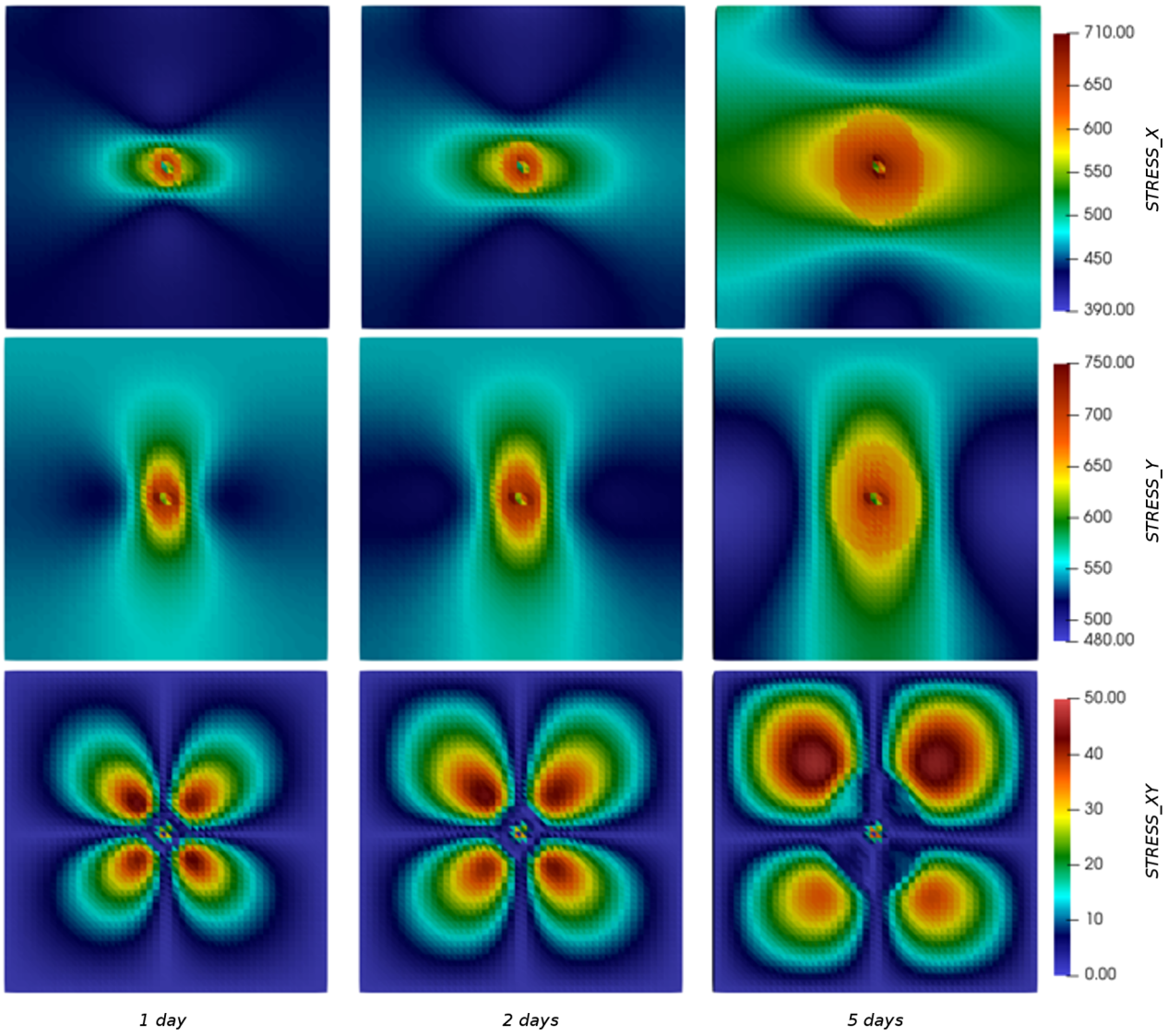

Figure 5. Distribution of the Lateral Components of Stress Tensor σ_{xx}, σ_{yy} and σ_{xy} after 1 day (left), 2 days (center) and 5 days (right)

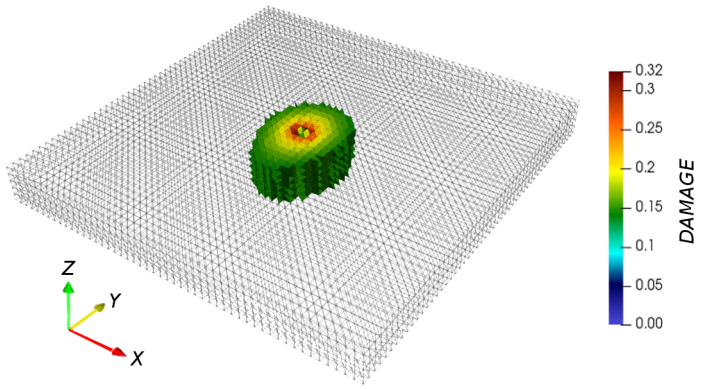

Figure 6. The area of the greatest damage

To ensure a more accurate assessment of the damage front advance associated with the growth of cracks in the reservoir, additional terms need to be introduced to the equation describing the damage parameter (9). According to [23], this condition includes a requirement that damage parameter must change if the maximum principal components of the effective stress tensor exceeds a certain threshold value:

$$\boldsymbol{\sigma}_{\text{eff}} = \boldsymbol{C} : \boldsymbol{\varepsilon} > \boldsymbol{\sigma}_{\max}.$$

The value of $\boldsymbol{\sigma}_{\max}$ in our calculations was assumed to be 100 bar.

The fluid injection time in our simulation was 5 days. The damage parameter, pressure and temperature distribution at different points in time is shown in Figure 4. The lateral components of the stress tensor are presented in Figure 5.

In the center of the model, an elliptical area can be distinguished (Figure 6), corresponding to the diffuse damage of the medium around the well. The area grows in size to 30×20 m after 5 days. The average value of the damage parameter in this area is 0.2, and the permeability has increased about 12 times in this area. A sharp transition from the damage area to intact area is associated with the introduction of the criteria for the threshold value of the effective stress tensor as presented above.

For this calculation, we assessed the contribution of various effects into the stress-strain state of the reservoir. Expression (2) for the total stress tensor can be expressed as

$$\Delta\boldsymbol{\sigma} = \boldsymbol{\sigma}_{\boldsymbol{\varepsilon}} - \boldsymbol{\sigma}_p - \boldsymbol{\sigma}_T - \boldsymbol{\sigma}_D,$$

where $\boldsymbol{\sigma}_{\boldsymbol{\varepsilon}} = \boldsymbol{C} : \Delta\boldsymbol{\varepsilon}$ is the strain component, $\boldsymbol{\sigma}_p = \boldsymbol{b}\Delta p$ is the filtration component, $\boldsymbol{\sigma}_T = \boldsymbol{C} : \boldsymbol{\alpha}_T \Delta T$ is the thermal component, $\boldsymbol{\sigma}_D = D \cdot \boldsymbol{C} : \Delta\boldsymbol{\varepsilon}$ is the damage-related component.

Figure 7 shows the distribution of each component's share after 30 days, where each component of the full stress tensor was normalized to the value

$$\Sigma = |\boldsymbol{\sigma}_{\boldsymbol{\varepsilon}}| + |\boldsymbol{\sigma}_p| + |\boldsymbol{\sigma}_T| + |\boldsymbol{\sigma}_D|.$$

It follows from Figure 7 that deformation and filtration processes are the main contributors to the stress-strain state. The contribution of fracturing in the affected area is about 15%, and thermal effects account for less than 10%.

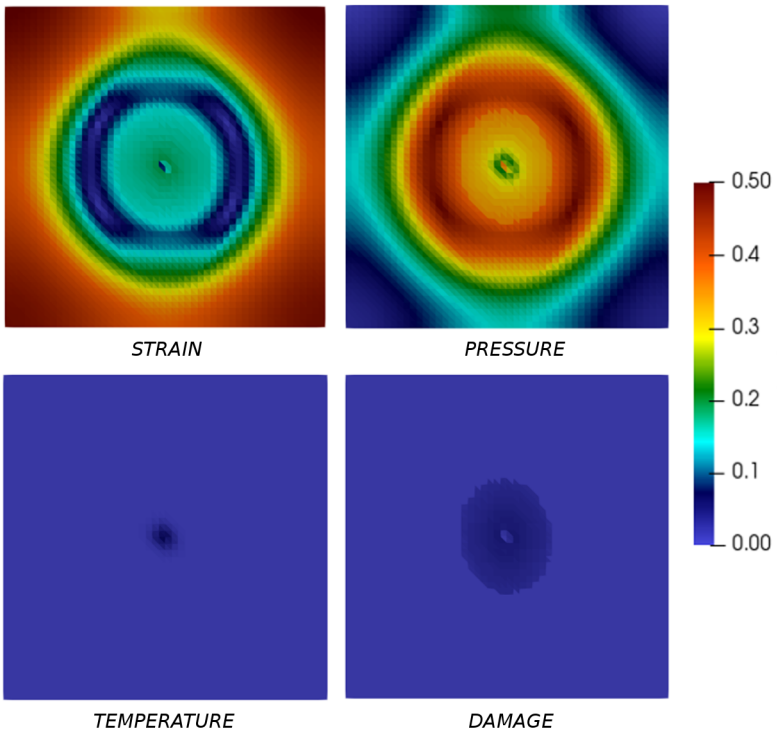

Figure 7. Contribution of various parameters to the change in average stress (in fractions) over the period of 30 days

Computation statistics. The computations were performed on a desktop computer with an Intel Core i7 processor with 8 logical cores. Computation of the linear system of equations was paralleled (via *HYPRE* library functions) using OMP (Open Multi-Processing) technology. The full computation time was 37 minutes, the number of nodes in the model was 112,211, the average amount of RAM used is 3 GB. The average number of linear iterations is 30, the number of Newtonian iterations is between 3 and 6.

6. Conclusion. This article provides a description of a software package for simulating thermo-poroelastic medium evolution taking into account its damage, the mathematical model used and computational algorithm are considered. The mathematical model is a system of equations including the mass, momentum and energy conservation laws, complemented with thermodynamically consisted constitutive relations. The computational algorithm is based on the finite element method. The system of equations is solved jointly without iterative binding. The work of the software package is demonstrated using the example problem of fracture development when injecting hot fluid into the reservoir. The degree of reservoir fracturing is analyzed in this context, along with the contribution of various effects to the change in the stress-strain conditions.

References

1. Krajcinovic D., Fonseka G. U. The continuous damage theory of brittle materials, part 1: general theory. *Journal of applied Mechanics*, 1981, V. 48, No. 4, P. 809–815.

2. Murakami S. Continuum damage mechanics: a continuum mechanics approach to the analysis of damage and fracture. Springer Science and Business Media, 2012, V. 185.

3. Kondaurov V. I, Fortov V. E. Fundamentals of condensed medium fluid mechanics. M.: MFTI, 2002, 336 p.

4. Biot, M. A. General theory of three dimensional consolidation. *Journal of Applied Physics*, 12, pp. 155–164. 1941

5. Noll W. A mathematical theory of the mechanical behavior of continuous media. *Archive for rational Mechanics and Analysis.* 1958, V. 2, No. 1, P. 197–226.

6. Griffith A. A. VI. The phenomena of rupture and flow in solids. *Philosophical transactions of the royal society of London.* Series A, containing papers of a mathematical or physical character, 1921, V. 221, No. 582–593, P. 163–198.

7. Kachanov L. M. Time to failure under creep conditions. *Izv AN USSR*, 1958, No 8, p. 26–31

8. Rabotnov Yu. N. About the mechanism of long destruction. *Fracture problems of durability of materials and designs.* M.: Publishing house of Academy of Sciences of the USSR. 1959. P. 5–7.

9. Meretin A. S., Savenkov E. B. Mathematical model for coupled flow and damage in thermoporoelastic medium. *Keldysh Institute Preprints*, 2019, No 58, 38 p.

10. Kim J., Tchelepi H. A., Juanes R. Stability, Accuracy and Efficiency of Sequential Methods for Coupled Flow and Geomechanics. *SPE Paper 119084*, 2009.

11. Taylor C., Hood P. A. A numerical solution of the Navier-Stokes equations using the finite element technique. *Computers and Fluids*, 1973, V. 1, No. 1, p. 73–100.

12. Brezzi F., Fortin M. Mixed and Hybrid Finite Elements Methods. Springer, 1991.

13. Saad Y. Iterative methods for sparse linear systems. Siam, 2003. V. 82.

14. Neuman S. P. Saturated-unsaturated seepage by finite elements. *J. HYDRAUL. DIV., PROC., ASCE.* 1973.

15. Cuthill E., McKee J. Reducing the bandwidth of sparse symmetric matrices. *In Proc. 24th Nat. Conf. ACM*, pages 157-–172, 1969.

16. Eigen library. URL: http://eigen.tuxfamily.org/. Cited September 14, 2020.

17. HYPRE: Scalable Linear Solvers and Multigrid Methods. URL: https://computation.llnl.gov/projects/hypre-scalable-linear-solvers-multigrid-methods. Cited September 14, 2020.

18. ParaView. URL: https://www.paraview.org/. Cited September 14, 2020.

19. VTK — The Visualization Toolkit. URL: https://www.vtk.org/. Cited September 14, 2020.

20. Pogacnik J., O'Sullivan M., O'Sullivan J. A Damage Mechanics Approach to Modeling Permeability Enhancement in Thermo-Hydro-Mechanical Simulations. *Proceedings of 39th Workshop on Geothermal Reservoir Engineering*, Stanford University, 2014, P. 24-26.

21. Tang C. A. et al. Coupled analysis of flow, stress and damage (FSD) in rock failure. *International Journal of Rock Mechanics and Mining Sciences*, 2002, V. 39, No. 4, P. 477-489.

22. Beggs H. D. et al. Estimating the viscosity of crude oil systems *Journal of Petroleum technology*, 1975, V. 27, №. 09, P. 1140-1141.

23. Sun F., Jia P., Xue S. Continuum Damage Modeling of Hydraulic Fracture from Perforations in Horizontal Wells. *Mathematical Problems in Engineering*, 2019

Accepted
February 10, 2020