

Thermodynamically consistent Darcy–Brinkman–Forchheimer framework in matrix acidization[★]

Yuanqing Wu¹, Jisheng Kou^{2,3}, Shuyu Sun^{4,*}, and Yu-Shu Wu⁵

¹ College of Mathematics and Statistics, Shenzhen University, Shenzhen, Guangdong 518048, PR China

² School of Civil Engineering, Shaoxing University, Shaoxing, Zhejiang 312000, PR China

³ School of Mathematics and Statistics, Hubei Engineering University, Xiaogan, Hubei 432000, PR China

⁴ Computational Transport Phenomena Laboratory, Division of Physical Science and Engineering, King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia

⁵ Department of Petroleum Engineering, Colorado School of Mines, 1600 Arapahoe Street, Golden, CO 80401, USA

Received: 29 May 2020 / Accepted: 17 November 2020

Abstract. Matrix acidization is an important technique used to enhance oil production at the tertiary recovery stage, but its numerical simulation has never been verified. From one of the earliest models, i.e., the two-scale model (Darcy framework), the Darcy–Brinkman–Forchheimer (DBF) framework is developed by adding the Brinkman term and Forchheimer term to the momentum conservation equation. However, in the momentum conservation equation of the DBF framework, porosity is placed outside of the time derivation term, which prevents a good description of the change in porosity. Thus, this work changes the expression so that the modified momentum conservation equation can satisfy Newton's second law. This modified framework is called the improved DBF framework. Furthermore, based on the improved DBF framework, a thermal DBF framework is given by introducing an energy balance equation to the improved DBF framework. Both of these frameworks are verified by former works through numerical experiments and chemical experiments in labs. Parallelization to the complicated framework codes is also realized, and good scalability can be achieved.

1 Introduction

Acidization is a useful technique for promoting or restoring oil production in reservoirs and can be classified as either fracture acidization or matrix acidization. In fracture acidization, a highly pressurized acid flow is injected into a well to physically enlarge the fractures and chemically dissolve the deposits that inhibit permeability. However, in matrix acidization, the pressure of the acid flow is not high enough to destroy the fractures, and thus, the acid flow can only enlarge the natural pores of the matrix. Both kinds of acidization attempt to enlarge the voids in reservoirs and ease the outflow of hydrocarbons from the subsurface matrix. Many previously published papers [1–3] have studied fracture acidization, but this work pays more attention to matrix acidization.

Theoretically, matrix acidization is a chemical dissolution–front instability problem. Many studies have focused

on the factors that influence matrix acidization, such as the mineral reactive surface area [4], mineral dissolution ratio [5], and solute dispersion [6]. Numerically, four major models have been proposed to investigate matrix acidization, including the capillary tube model [7], the network model [8–10], the dimensionless model [11, 12] and the two-scale model [13–15]. Since the two-scale model can better predict dissolution patterns and more accurately capture the formation of wormholes, this work focuses on the two-scale model. The two scales indicate the Darcy scale and pore scale. In each scale, there are a series of equations used to describe the progress of matrix acidization. In earlier literature, the momentum conservation equation in the Darcy scale was described by Darcy's law, so the two-scaled model was also named the Darcy framework in these works [15–18]. Later, due to the nature of matrix acidization, Wu *et al.* [19–22] concluded that the Darcy framework was insufficiently accurate to describe matrix acidization and provided the Darcy–Brinkman–Forchheimer (DBF) framework by adding the Brinkman term and Forchheimer term to the momentum conservation equation. Li *et al.* [23–25] further analyzed the numerical stability and accuracy of the DBF framework. However, the DBF framework still has a defect when processing the momentum

[★] This work is supported by the *Peacock Plan Foundation of Shenzhen* (No. 000255), the *National Natural Science Foundation of China* (No. 11601345) and the *Natural Science Foundation of SZU* (No. 2017059).

* Corresponding author: shuyu.sun@kaust.edu.sa

conservation equation, which degrades its reliability. This work reviews the DBF framework in more detail than [19] and focuses on this defect, which is covered in Section 2. Moreover, in the DBF framework, a pseudo parameter e is introduced into the mass conservation equation [26] to solve the linear system by an iterative solver HYPRE [27], which also degrades the reliability of the DBF framework. However, by replacing HYPRE with a direct solver MUMPS [28, 29], the introduction of e is not necessary, which is realized in this work. Furthermore, the flowchart of the simulation should also be changed accordingly. Thus, the new framework provided in this work is called the improved two-scale model based on the DBF framework, or the improved DBF framework for short.

However, the frameworks above only consider the mass conservation law and momentum conservation law, and the energy conservation law is not included, which is a major drawback. Temperature is a key variable in the energy conservation law and has a significant influence on thermodynamic parameters such as the surface reaction rate and the molecular diffusion coefficient. However, these thermodynamic parameters are deemed constants in those frameworks. Moreover, in real applications, matrix acidization is performed in a subsurface environment where the matrix is warmed by terrestrial heat. Thus, the temperature should be considered as a major factor in matrix acidization, which promotes the necessity and reasonability of introducing the energy conservation equation into these frameworks.

In addition to this work, many researchers have acknowledged the temperature issue and upgraded the two-scale model in their own ways. For example, Li *et al.* [30] introduced a heat transmission equation in the form of radial flows to the two-scale model and produced simulation results near the wellbore. Ma *et al.* [2] also developed a temperature-influenced model based on the two-scale model and used it to simulate matrix acidization in fractured carbonate rocks. Kalia and Glasbergen [31] studied cases when the temperatures of the acid fluid and the matrix were different and simulated acidization in the matrix under both adiabatic and nonadiabatic conditions. They concluded that the fluid temperature could be designed as a parameter to control matrix acidization. Although these endeavors attempted to consider the thermal effect on matrix acidization, all of them were based on the Darcy framework, which is not accurate enough to simulate matrix acidization, as mentioned above. As a result, their reliability was degraded. Therefore, this work provides a heat transfer model as an expansion of a more reasonable improved DBF framework and aims to output more reliable results. The new model is a thermodynamically consistent DBF framework, which can be called the thermal DBF framework for short. Different from [30], this work studies matrix acidization in the form of linear flows. Meanwhile, fractures in the matrix are not considered, and a general matrix is acidized, which is different from [2]. Inspired by [31], two cases are studied – when the temperatures of the acid flow and the matrix are the same and when they are different – and the results are verified against [31] and [32], respectively.

The work in [19] only realized a 2D parallel code of the DBF framework; the present study further realizes a 3D

parallel code of the DBF framework. In addition, the 2D and 3D parallel codes of the improved DBF framework and thermal DBF framework are also realized in the present work.

In the following discussions, the improved DBF framework is developed, and then the thermal DBF framework is provided. In the model verification section, the correctness of the improved DBF framework is checked by comparing its 2D and 3D results with existing works. Only when the correctness of the improved DBF framework is guaranteed can the reasonability of the thermal DBF framework be assured. After this, a series of thermal experiments are performed to investigate the temperature effect on matrix acidization. The performance of the 3D parallel code is evaluated at the end of this work.

2 Improved DBF framework and its solution scheme

The meanings of all the notations in the statements below are listed in Table 1. The Darcy framework has been widely used to simulate the matrix acidization procedure. At the pore scale, a group of semiempirical equations is provided to describe the relationship of parameters at the pore scale, such as porosity, permeability [33], and local mass-transfer coefficient. These equations have seen few changes during the study of matrix acidization simulation. However, at the Darcy scale, which is the other scale of the Darcy framework, these equations have seen many changes with continued study. Generally, there are three kinds of equations on the Darcy scale: momentum conservation equations, mass conservation equations, and concentration balance equations. The changes have mainly occurred in the momentum conservation equations, while the other two kinds of equations have remained more or less the same, although some small changes have been made according to the needs of different cases. Initially, the momentum conservation equation is represented by Darcy's law:

$$\nabla p + \frac{\mu}{K} \mathbf{u} = 0,$$

assuming that the permeability is homogeneous and isotropic. However, the applicability of this equation is limited to conditions where the Reynolds number $Re < 1$ [34] and the Darcy number $Da \ll 1$. At the beginning of matrix acidization, the porosity in the porous medium is not high, and Darcy's law can be leveraged to properly describe the flow of fluid in the porous medium. However, with the propagation of channels due to matrix acidization, the areas eroded by the channels become enlarged. In these channels, the porosity can grow and even approach the value of one, which creates high permeability in these areas. From the definition of the Darcy number, it can be determined that the Darcy number is much higher than one as a result of this change. Moreover, in high-permeability areas, the fluid velocity increases, which may lead to a high Reynolds number that can exceed the value of one, assuming that the viscosity and

Table 1. Nomenclature.

Notation	Meaning
p	Pressure
μ	Fluid viscosity
K	Permeability value
\mathbf{u}	Velocity vector
Re	Reynolds number
Da	Darcy number
ϕ	Porosity
ρ_f	Mass density of the fluid
F	Forchheimer coefficient
t	Time
\mathbf{g}	Gravity vector
C_f	Cup-mixing concentration of the acid
\mathbf{D}_e	Effective dispersion tensor
d_m	Molecular diffusion coefficient
d_l	Longitudinal dispersion coefficient
d_t	Transverse dispersion coefficient
α_{OS}	Constant depending on pore connectivity
λ_X	Constant depending on the structure of the medium
λ_T	Constant depending on the structure of the medium
r_p	Pore radius
\mathbf{E}	Orthogonal projection along the velocity
\mathbf{I}	Identity matrix
k_c	Local mass-transfer coefficient
a_v	Interfacial surface area per unit volume
C_s	Concentration of the acid at the fluid-solid interface
T	Temperature
$R(C_s, T)$	Reaction rate
α	Dissolving power of the acid
ρ_s	Mass density of the solid phase
k_s	Surface reaction rate
Sh	Sherwood number
Sh_∞	Asymptotic Sherwood number
S_c	Schmidt number
τ	Time step
$H_r(T)$	Reaction heat
λ_f	Heat conduction coefficient of the fluid phase
λ_s	Heat conduction coefficient of the solid phase
λ	Average heat conduction coefficient
θ_f	Heat capacity of the fluid phase
θ_s	Heat capacity of the solid phase
ϑ_f	Amount of heat per unit volume of the fluid phase
ϑ_s	Amount of heat per unit volume of the solid phase
ϑ	Total amount of heat per unit volume
E_g	Activation energy
R_g	Molar gas constant

mass density of the fluid remain more or less the same and that the particle size remains constant. These reasons indicate that Darcy's law is not suitable for use in matrix acidization simulations.

To address the issue, some corrections have to be made to Darcy's law to cope with the conditions of high permeability and high Reynolds numbers. The first correction is called the Brinkman correction. According to Darcy's law, the uniform velocity in a cross-sectional direction can be seen when the permeability is low. However, in a porous medium with high permeability, a no-slip condition should be considered instead of uniform velocity. The Brinkman correction introduces a viscous shear stress term to Darcy's law that can describe the no-slip well. The Brinkman-corrected Darcy's law can be expressed as,

$$\nabla p + \frac{\mu}{K} \mathbf{u} - \frac{\mu}{\phi} \nabla^2 \mathbf{u} = 0,$$

in which $\frac{\mu}{\phi} \nabla^2 \mathbf{u}$ is called the Brinkman term. In addition, under the condition of high Reynolds numbers, form drag can be much larger than viscous drag, which can be suitably described by a Forchheimer correction. In this correction, a term called the Forchheimer term, which is expressed as $\frac{\rho_f F}{\sqrt{K}} |\mathbf{u}| \mathbf{u}$, is added to Darcy's law. Combining both corrections together, Darcy's law can be modified as,

$$\nabla p + \frac{\mu}{K} \mathbf{u} - \frac{\mu}{\phi} \nabla^2 \mathbf{u} + \frac{\rho_f F}{\sqrt{K}} |\mathbf{u}| \mathbf{u} = 0,$$

with $F = \frac{1.75}{\sqrt{150\phi^3}}$ as the Forchheimer coefficient [35]. In the equation above, the right-hand side equals zero, which means that the sum of all the external forces imposed on the fluid is zero. However, this is not a general case. For cases in which the sum of all the external forces is not zero, the right-hand side of the equation should equal the product of mass density and acceleration. With a Eulerian expression of acceleration, the right-hand side of the equation can be written as,

$$-\frac{\rho_f}{\phi} \frac{\partial \mathbf{u}}{\partial t} - \frac{\rho_f}{\phi^2} \nabla \cdot \mathbf{u} \mathbf{u}.$$

If the gravity effect is considered, the momentum conservation equation in its final form can be written as,

$$\begin{aligned} \frac{\rho_f}{\phi} \frac{\partial \mathbf{u}}{\partial t} + \frac{\rho_f}{\phi^2} \nabla \cdot \mathbf{u} \mathbf{u} = & -\nabla p - \frac{\mu}{K} \mathbf{u} + \frac{\mu}{\phi} \nabla^2 \mathbf{u} \\ & - \frac{\rho_f F}{\sqrt{K}} |\mathbf{u}| \mathbf{u} + \rho_f \mathbf{g}. \end{aligned} \quad (1)$$

Equation (1) is the momentum conservation equation used in our previous work [19]. Because this kind of momentum conservation equation introduces the Brinkman term and Forchheimer term, this model is a two-scale model based on the Darcy–Brinkman–Forchheimer (DBF) framework, or the DBF framework for short. More details can be found in [19]. However, the framework cannot meet Newton's second law, since porosity ϕ is changed during the simulation procedure. It is noted that in the first term on the left-hand side of equation (1), ϕ is outside of the time derivative,

which indicates that ϕ does not change with time. This contradicts the true physical observation, and thus Newton's second law is also violated. Thus, ϕ should be placed inside the time derivative. After this operation, $\frac{\mathbf{u}}{\phi}$ can be deemed a new variable – the effective velocity. Accordingly, the second term on the left-hand side of equation (1) and the third term on the right-hand side of equation (1) are changed, which can be expressed as,

$$\rho_f \frac{\partial}{\partial t} \left(\frac{\mathbf{u}}{\phi} \right) + \rho_f \frac{\mathbf{u}}{\phi} \cdot \nabla \frac{\mathbf{u}}{\phi} = -\nabla p - \frac{\mu}{K} \mathbf{u} + \nabla \cdot \mu \nabla \frac{\mathbf{u}}{\phi} - \frac{\rho_f F}{\sqrt{K}} |\mathbf{u}| \mathbf{u} + \rho_f \mathbf{g}. \quad (2)$$

It is noted that the left-hand side of equation (2) is in reality the material derivative,

$$\rho_f \frac{\partial}{\partial t} \left(\frac{\mathbf{u}}{\phi} \right) + \rho_f \frac{\mathbf{u}}{\phi} \cdot \nabla \frac{\mathbf{u}}{\phi} = \rho_f \frac{D}{Dt} \left(\frac{\mathbf{u}}{\phi} \right).$$

The discussion above, taken from the theories of fluid dynamics, demonstrates that the new momentum conservation equation should describe matrix acidization more reasonably, and thus the new model is an “improved” two-scale model based on the DBF framework, or the improved DBF framework for short.

Since the flow in matrix acidization is assumed to be incompressible, the mass conservation equation can be expressed as,

$$\nabla \cdot \mathbf{u} = 0.$$

However, considering the local volume change in the matrix acidization procedure, the mass conservation equation should be modified as,

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \mathbf{u} = 0. \quad (3)$$

The concentration balance equation can be derived from the principle of species balance during matrix acidization. The balance of species can be achieved by accumulation, advection, diffusion, and reaction effects, which bring about a straightforward expression of the concentration balance equation,

$$\frac{\partial(\phi C_f)}{\partial t} + \nabla \cdot (\mathbf{u} C_f) = \nabla \cdot (\phi \mathbf{D}_e \cdot \nabla C_f) - k_c a_v (C_f - C_s). \quad (4)$$

In the equation, \mathbf{D}_e is a function of \mathbf{u} ,

$$\mathbf{D}_e = d_m \mathbf{I} + \|\mathbf{u}\| (d_l \mathbf{E} + d_t \mathbf{E}^\perp),$$

with,

$$d_l = \alpha_{OS} d_m + \frac{2\lambda_X \|\mathbf{u}\| r_p}{\phi},$$

$$d_t = \alpha_{OS} d_m + \frac{2\lambda_T \|\mathbf{u}\| r_p}{\phi}.$$

In the 3D condition,

$$\mathbf{E} = \frac{1}{\|\mathbf{u}\|^2} \begin{pmatrix} u_x^2 & u_x u_y & u_x u_z \\ u_y u_x & u_y^2 & u_y u_z \\ u_z u_x & u_z u_y & u_z^2 \end{pmatrix},$$

$$\mathbf{E}^\perp = \mathbf{I} - \mathbf{E}.$$

u_x , u_y , and u_z represent the x -direction, y -direction, and z -direction velocities, respectively. The 2D conditions are similar. In fact, the concentration balance equation is the other expression of the mass conservation law, in addition to the mass conservation equation.

In equations (2)–(4), the velocity vector \mathbf{u} , pressure p , and the cup-mixing concentration of the acid C_f are deemed unknowns to be solved for. However, the three equations cannot accomplish this since the values of the other variables are unknown. Thus, more auxiliary equations and other necessary assumptions are needed. Three additional equations are given as follows:

$$k_c (C_f - C_s) = R(C_s, T), \quad (5)$$

$$\frac{\partial \phi}{\partial t} = \frac{R(C_s, T) a_v \alpha}{\rho_s}, \quad (6)$$

$$C_s = \frac{C_f}{1 + \frac{k_s}{k_c}}. \quad (7)$$

These equations are derived from mathematical deduction and chemical experiments, and more details can be found in [15]. It should be emphasized that equation (7) can be put in equation (4) to substitute C_s when solving C_f . Moreover, since the mass density ρ_f and viscosity μ of the fluid will not change much during the matrix acidization procedure, they should be given constants for simplicity. It is also easy to understand that the dissolving power of the acid α and the mass density of the solid phase ρ_s can also be deemed as given constants. For the reason mentioned later, the surface reaction rate k_s is no longer deemed as a constant, as shown in [19]. Instead, it is a function of the temperature T in this work. As a result, the reaction rate also becomes a function of T and can be rewritten as $R(C_s, T)$, which is different from [19].

In addition, it is noted that some variables at the pore scale, such as porosity, permeability, the interfacial surface area per unit volume and the local mass-transfer coefficient, appear in the equations at the Darcy scale. Thus, their values should be known before we attempt to solve for the unknowns on the Darcy scale with the help of a series of equations on the pore scale. It is stipulated that the subscript 0 represents the initial value or reference value of the corresponding variable in the following equations, and all the initial values are known. First, three equations called the Carman–Kozeny correlation are provided as follows:

$$\frac{K}{K_0} = \frac{\phi}{\phi_0} \left(\frac{\phi(1 - \phi_0)}{\phi_0(1 - \phi)} \right)^2, \quad (8)$$

$$\frac{r_p}{r_0} = \sqrt{\frac{K\phi_0}{K_0\phi}}, \quad (9)$$

$$\frac{a_v}{a_0} = \frac{\phi r_0}{\phi_0 r_p}. \quad (10)$$

From these equations, it can be seen how the permeability, pore radius, and interfacial surface area per unit volume change with porosity. Thus, as long as the porosity is known, the values of these three variables can be computed from the Carman–Kozeny correlation. At this point, we can compute porosity. From equations (5) to (7), the equation below can be derived:

$$\frac{\partial \phi}{\partial t} = \frac{a_v \alpha C_f k_c k_s}{\rho_s (k_c + k_s)}. \quad (11)$$

The left-hand side of equation (11) describes the change in porosity with time, and its right-hand side includes many variables; except for a_v and k_c , all the other variables have no direct relationship with porosity. By using equations (8)–(10), a_v can be expressed as a function of ϕ ,

$$a_v = a_0 \frac{1 - \phi}{1 - \phi_0}. \quad (12)$$

Equation (11) can then be changed as,

$$\frac{\partial \phi}{\partial t} = \frac{a_0 \alpha C_f k_c k_s (1 - \phi)}{\rho_s (k_c + k_s) (1 - \phi_0)}. \quad (13)$$

Moreover, the local mass-transfer coefficient k_c can be calculated from the expression of the Sherwood number Sh , which is a dimensionless mass-transfer coefficient. The expression is given as follows:

$$Sh = \frac{2k_c r_p}{d_m} = Sh_\infty + 0.7 Re^{1/2} Sc^{1/3}. \quad (14)$$

On the right-hand side of equation (14), Sh_∞ is a given constant, and the Reynolds number Re can be expressed as,

$$Re = \frac{2\|\mathbf{u}\| r_p \rho_f}{\mu}. \quad (15)$$

The Schmidt number Sc is expressed as,

$$Sc = \frac{\mu d_m}{\rho_f}. \quad (16)$$

Thus, by leveraging equations (14)–(16) together, k_c can be calculated as,

$$k_c = \frac{d_m}{2r_p} \left(Sh_\infty + 0.7 \left(\frac{2\|\mathbf{u}\| r_p \rho_f}{\mu} \right)^{\frac{1}{2}} \left(\frac{\mu d_m}{\rho_f} \right)^{\frac{1}{3}} \right). \quad (17)$$

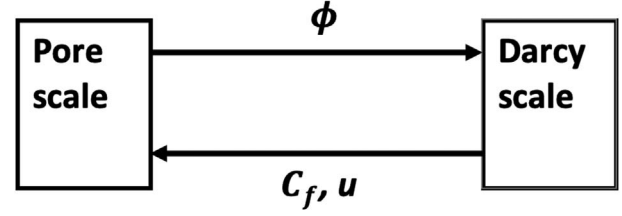


Fig. 1. The interaction of the pore scale and Darcy scale. The upper arrow indicates that the pore scale will affect Darcy scale by the pore-scale variable ϕ . As long as ϕ is changed, the pore-scale variables used in the Darcy scale are changed, which brings about the changes of the main variables \mathbf{u} , p , and C_f in the Darcy scale. The lower arrow indicates that the Darcy scale will affect the pore scale by the Darcy-scale variables C_f and \mathbf{u} , since the two variables will affect the value of ϕ directly. Once ϕ is changed, all the other variables in the pore scale will be changed accordingly.

Equation (17) includes the variable r_p , which is a function of porosity from equations (8) and (9),

$$r_p = r_0 \frac{\phi(1 - \phi_0)}{\phi_0(1 - \phi)}, \quad (18)$$

which means that k_c is in fact a function of porosity and velocity. Thus, equation (17) can substitute for k_c in equation (13), and then a new equation with porosity being the unknown can be derived. However, the new equation is too complex to derive an analytic formula of porosity, and therefore, the value of porosity has to be computed by a numerical scheme. Under that condition, a semi-implicit scheme is applied. In the following statements, the superscripts of the notations represent the time step. The porosity at time step τ is used in equation (18) to calculate the pore radius at time step τ , which is then put into equation (17) to compute k_c at time step τ . With the semi-implicit scheme, equation (13) can be rewritten as,

$$\frac{\phi^{\tau+1} - \phi^\tau}{\Delta t} = \frac{a_0 \alpha C_f^\tau k_c^\tau k_s (1 - \phi^{\tau+1})}{\rho_s (k_c^\tau + k_s^\tau) (1 - \phi_0)}, \quad (19)$$

by which $\phi^{\tau+1}$ can be computed easily.

From the discussion above, it can be seen that the main unknown to be solved at the pore scale is porosity ϕ . As long as the porosity value is obtained, the values of the other variables at the pore scale can be derived from a series of pore-scale equations. Furthermore, with the values of all the pore-scale variables, the main unknowns in the Darcy scale can be calculated by the Darcy-scale equations. In addition, from the computation of porosity, it can be seen that the variables \mathbf{u} and C_f in the Darcy scale will affect the porosity value. Therefore, the computations of the Darcy scale and pore scale are coupled with each other, with the Darcy-scale variables \mathbf{u} and C_f and the pore-scale variable ϕ being their interaction media, which can be shown in Figure 1.

The derivations of all the equations at both the Darcy scale and pore scale have been achieved in the former discussions, and then numerical schemes are applied to these derivations to solve for the variables. Since porosity ϕ plays a central role in the improved DBF framework, it should be computed from equation (19) with the semi-implicit scheme first. Then, with the computed porosity, permeability K and the interfacial surface area per unit volume a_v can be computed with equations (8) and (12), respectively. Next, equations (2) and (3) are combined together as a linear system and solved for the velocity \mathbf{u} and pressure p with the semi-implicit scheme. It is emphasized that the term $\frac{\partial \phi}{\partial t}$ in equation (3) is substituted by equation (6) when the linear system is solved. Moreover, with the update of porosity ϕ , the local mass-transfer coefficient is also updated by equations (17) to $k_c^{\tau+\frac{1}{2}}$ before solving the linear system. With the semi-implicit scheme, equations (2) and (3) can be rewritten as follows:

$$\begin{aligned} \rho_f \frac{\frac{\mathbf{u}^{\tau+1}}{\phi^{\tau+1}} - \frac{\mathbf{u}^\tau}{\phi^\tau}}{\Delta t} + \rho_f \frac{\mathbf{u}^\tau}{\phi^\tau} \cdot \nabla \frac{\mathbf{u}^{\tau+1}}{\phi^{\tau+1}} &= -\nabla p^{\tau+1} - \frac{\mu}{K^{\tau+1}} \mathbf{u}^{\tau+1} \\ + \nabla \cdot \mu \nabla \frac{\mathbf{u}^{\tau+1}}{\phi^{\tau+1}} - \frac{\rho_f F^{\tau+1}}{\sqrt{K^{\tau+1}}} |\mathbf{u}^\tau| \mathbf{u}^{\tau+1} + \rho_f \mathbf{g}, \\ \frac{a_v^{\tau+1} \alpha C_f^\tau k_c^{\tau+\frac{1}{2}} k_s^\tau}{\rho_s (k_c^{\tau+\frac{1}{2}} + k_s^\tau)} + \nabla \cdot \mathbf{u}^{\tau+1} &= 0. \end{aligned}$$

After that, since the velocity \mathbf{u} is updated, the local mass-transfer coefficient can be updated again by equations (17) to $k_c^{\tau+1}$. Last, the semi-implicit scheme is used to solve equation (4) for concentration C_f and then another linear system is formed. It is emphasized that equation (7) is put into equation (4) when the linear system is solved, which is shown as,

$$\begin{aligned} \frac{\phi^{\tau+1} C_f^{\tau+1} - \phi^\tau C_f^\tau}{\Delta t} + \nabla \cdot (\mathbf{u}^{\tau+1} C_f^{\tau+1}) \\ = \nabla \cdot (\phi^{\tau+1} \mathbf{D}_e^{\tau+1} \cdot \nabla C_f^{\tau+1}) \\ - k_c^{\tau+1} a_v^{\tau+1} \left(C_f^{\tau+1} - \frac{C_f^{\tau+1}}{1 + \frac{k_s^\tau}{k_c^{\tau+1}}} \right). \end{aligned} \quad (20)$$

In brief, the solution procedure can be described as a flowchart, which is shown in Figure 2.

From the flowchart, another difference of this work from our former work [19] can be seen. In the former work, for each iteration, the simulation begins with the computation of the variables in the Darcy scale, such as pressure, velocity, and concentration, and ends with the computation of the variables in the pore scale, such as porosity, permeability, and the interfacial surface area per unit volume. However, in the present work, the computation of the variables at the pore scale is performed ahead of the computation of the variables at the Darcy scale. The flowchart of this work is more reasonable, which can be demonstrated by the following statements. From equation (20), it is

observed that to obtain $C_f^{\tau+1}$, we have to know $\phi^{\tau+1}$, and therefore the computation of $\phi^{\tau+1}$ should occur before the computation of $C_f^{\tau+1}$. However, in our former work, the computation of $\phi^{\tau+1}$ occurs after the computation of $C_f^{\tau+1}$, which is not reasonable. With this philosophy, in the simulation of the improved DBF framework, the variables are first computed at the pore scale and then at the Darcy scale.

3 Thermal DBF framework and its solution scheme

Based on the improved DBF framework, a heat transfer model that considers the heat transmission process in matrix acidization is developed. The model is composed of the improved DBF framework and the energy conservation equation, which can be expressed as the governing equation of the temperature T ,

$$\begin{aligned} \frac{\partial \vartheta}{\partial t} + \nabla \cdot \left(\mathbf{u} \frac{\vartheta_f}{\phi} \right) &= \nabla \cdot \lambda \nabla T - p \nabla \cdot \mathbf{u} + \mu \nabla \frac{\mathbf{u}}{\phi} : \nabla \frac{\mathbf{u}}{\phi} + \frac{\mu}{K} |\mathbf{u}|^2 \\ &+ \frac{\rho_f F}{\sqrt{K}} |\mathbf{u}|^3 + a_v R(C_s, T) H_r(T), \end{aligned} \quad (21)$$

where,

$$\begin{aligned} \vartheta &= \vartheta_f + \vartheta_s, \\ \vartheta_f &= \phi \rho_f \theta_f T, \\ \vartheta_s &= (1 - \phi) \rho_s \theta_s T, \\ \lambda &= \phi \lambda_f + (1 - \phi) \lambda_s, \\ H_r(T) &= [-9702 + 16.97 T - 0.00234 T^2]. \end{aligned}$$

$H_r(T)$ is the reaction heat [30]. λ_f and λ_s are the heat conduction coefficients of the fluid and solid phase, respectively, and thus, λ is the average heat conduction coefficient between the two phases. θ_f and θ_s are the heat capacities of the fluid and solid phase, respectively. λ_f , λ_s , θ_f and θ_s are deemed constants in this work. ϑ_f and ϑ_s are the amounts of heat per unit volume of the fluid and solid phase, respectively, and thus, ϑ is the total amount of heat per unit volume. The total energy is a sum of the fluid energy and solid medium energy, which may vary with time. The energy transportation caused by the fluid flow occurs only between fluids in different spatial positions due to fluid flow. The heat conduction in the interiors of both fluids and solids and with each other are considered in the first term of the right-hand side of (21). The effects of the pressure and viscosity force of the fluid are described in the second and third terms of the right-hand side of (21). The effects of friction forces between fluids and solids are described in the fourth and fifth terms of the right-hand side of (21) based on the Darcy–Forchheimer framework. The chemical reaction can produce heat, which is considered in the last term of (21). All the terms constitute the source or sink of energy. It is emphasized that the temperature of the acid and matrix is assumed to become the same

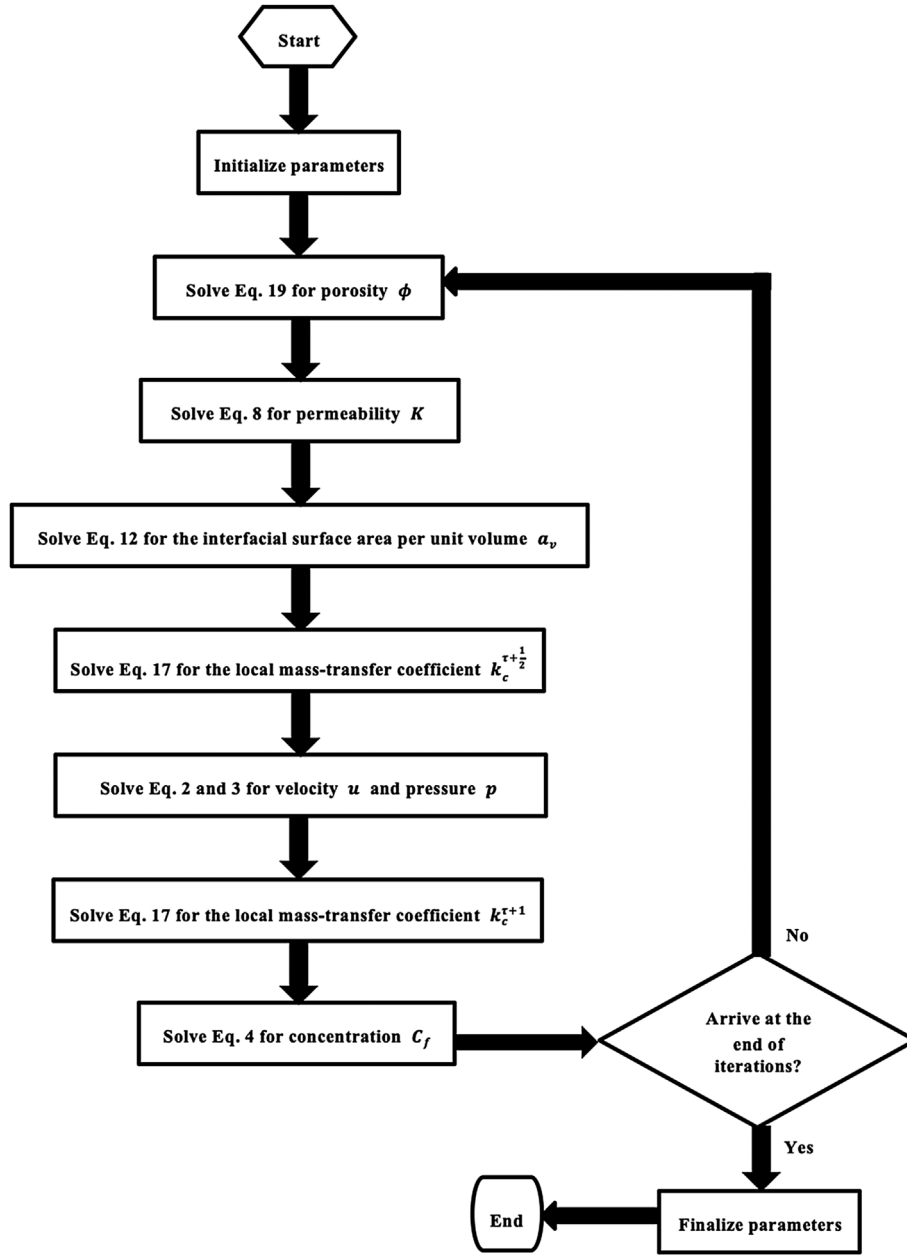


Fig. 2. The flowchart of the improved DBF framework.

immediately when acid is injected into the matrix, since the speed of heat transfer is much faster than the fluid speed. Thus, the temperature of the acid and matrix can be represented by a single notation T . In fact, differentiating between the acid temperature and matrix temperature brings challenges to theories and applications, and the details of the heat transfer between the acid and matrix must be studied thoroughly. Relevant work can be left to the future. Furthermore, the surface reaction rate k_s is deemed a function of the temperature T , which can be expressed as,

$$k_s = k_{s0} \cdot e^{\frac{E_g}{R_g}(\frac{1}{T_0} - \frac{1}{T})}, \quad (22)$$

in which k_{s0} is the surface reaction rate at temperature T_0 , E_g is the activation energy, and R_g is the molar gas constant [31]. As a result, the reaction rate also becomes a function of T , and its expression is rewritten as $R(C_s, T)$ in (21). Moreover, the molecular diffusion coefficient d_m is also affected by the temperature T , which can be expressed as [30],

$$d_m = d_{m0} \cdot e^{\frac{E_g}{R_g}(\frac{1}{T_0} - \frac{1}{T})}. \quad (23)$$

d_{m0} is the molecular diffusion coefficient at temperature T_0 . The heat transfer model induced from the improved DBF framework is called the thermal DBF framework for short.

After solving a series of equations in the improved DBF framework, the semi-implicit scheme can be used to solve equation (21) for the temperature T , which can be expressed as,

$$\begin{aligned} & \frac{(\phi^{\tau+1} \rho_f \theta_f T^{\tau+1} + (1 - \phi^{\tau+1}) \rho_s \theta_s T^{\tau+1}) - (\phi^\tau \rho_f \theta_f T^\tau + (1 - \phi^\tau) \rho_s \theta_s T^\tau)}{\Delta t} \\ & + \nabla \cdot (\mathbf{u}^{\tau+1} \phi^{\tau+1} \rho_f \theta_f T^{\tau+1}) \\ & = \nabla \cdot (\phi^{\tau+1} \lambda_f + (1 - \phi^{\tau+1}) \lambda_s) \nabla T^{\tau+1} - p^{\tau+1} \nabla \cdot \mathbf{u}^{\tau+1} + \mu \nabla \frac{\mathbf{u}^{\tau+1}}{\phi^{\tau+1}} : \\ & \nabla \frac{\mathbf{u}^{\tau+1}}{\phi^{\tau+1}} + \frac{\mu}{K^{\tau+1}} |\mathbf{u}^{\tau+1}|^2 + \frac{\rho_f F^{\tau+1}}{\sqrt{K^{\tau+1}}} |\mathbf{u}^{\tau+1}|^3 \\ & + k_c^{\tau+1} a_v^{\tau+1} \left(C_f^{\tau+1} - \frac{C_f^{\tau+1}}{1 + \frac{k_s}{k_c^{\tau+1}}} \right) H_r(T^\tau), \end{aligned}$$

and the third linear system is formed. It is noted that equation (7) is put into (21) in the above expression. When using the thermal DBF framework to simulate matrix acidization, the flowchart is shown in Figure 3. It can be seen in the figure that the molecular diffusion coefficient d_m and the surface reaction rate k_s are calculated first, followed by a series of computations in the improved DBF framework, and the computation of the temperature is performed last. It can also be seen that by using the variables ϕ , \mathbf{u} , p , and C_f , the improved DBF framework changes the main variable T in the energy conservation equation, while the energy conservation equation also changes the variables of the improved DBF framework by T , or k_s and d_m , which can be seen in Figure 4. From these results, it can be determined that the thermal DBF framework considers all three kinds of conservation laws: mass, momentum and energy, and should simulate matrix acidization more reasonably.

At the end of this section, the relationships among the Darcy framework, the DBF framework, the improved

4 Discretization and parallelization

Possible discretization methods include the multipoint flux approximation method and the hybrid finite volume method. The multipoint flux approximation method is “designed to give a correct discretization of the flow equations for general nonorthogonal grids as well as for general orientation of the principal directions of the permeability tensor” [36], while the hybrid finite volume method is “the ideal method for computing discontinuous solutions arising in compressible flows” [37]. Since this work considers orthogonal grids and incompressible flows, the finite difference method should be a better fit from all possible discretization methods. In the following discussion, the finite difference method is used to discretize the model, and the experimenting field approach [38–42] is used to compute the coefficients in the two linear systems. Although these approaches have been used in the 2D simulation of matrix acidization in our previous work [19], it is necessary to expand them to the 3D simulation, which is a focus of this work.

The equations used in the thermal DBF framework are discretized one by one according to the flowchart shown in Figure 3. For simplicity, suppose there is a 3D Cartesian grid. For equations (8), (12), (19), (22), and (23), every variable is imposed at the center of the cube. For equation (17), except the variable \mathbf{u} , the other variables are imposed at the center of the cube. However, $\|\mathbf{u}\|$ should also be imposed at the center of the cube. Generally, $\mathbf{u} = (u_x, u_y, u_z)$ and its three components are imposed on the faces of the cube. In other words, the x -direction velocity u_x is imposed on the x -direction face, which is vertical to the x -axis. The processes of the y -direction velocity u_y and z -direction velocity u_z are similar. Thus, for a cube with its x -coordinate ranging from i to $i + 1$, y -coordinate ranging from j to $j + 1$ and z -coordinate ranging from k to $k + 1$, there is,

$$\|\mathbf{u}\|_{(i+\frac{1}{2}, j+\frac{1}{2}, k+\frac{1}{2})} = \sqrt{\left(u_{x, i+\frac{1}{2}, j+\frac{1}{2}, k+\frac{1}{2}} - u_{x, i-\frac{1}{2}, j+\frac{1}{2}, k+\frac{1}{2}}\right)^2 + \left(u_{y, i+\frac{1}{2}, j+\frac{1}{2}, k+\frac{1}{2}} - u_{y, i+\frac{1}{2}, j-\frac{1}{2}, k+\frac{1}{2}}\right)^2 + \left(u_{z, i+\frac{1}{2}, j+\frac{1}{2}, k+\frac{1}{2}} - u_{z, i+\frac{1}{2}, j+\frac{1}{2}, k-\frac{1}{2}}\right)^2},$$

DBF framework and the thermal DBF framework can be summarized. Initially, the Darcy framework is provided to simulate matrix acidization and achieves great success. In the framework, the communication between the pore scale and Darcy scale advanced the progress of simulation. Then, considering the clear fluid area that cannot be described accurately by Darcy’s law, the Brinkman term and Forchheimer term are introduced to the momentum conservation equation, and the DBF framework is developed. However, this framework cannot obey Newton’s second law, and a modification is made, thus suggesting the improved DBF framework. Until now, all the frameworks have only considered two kinds of conservations, i.e., mass conservation and the momentum conservation, which is insufficient. Thus, based on the improved DBF framework, a third kind of conservation, energy conservation, is introduced in the thermal DBF framework. Their relationships can also be seen in Figure 5.

with the subscript representing the coordinate. Equation (2) is discretized on the faces of the cube, with the x -direction momentum equation discretized on the x -direction face, the y -direction momentum equation discretized on the y -direction face and the z -direction momentum equation discretized on the z -direction face. Thus, the porosity and permeability on the faces should be known. However, from the discussion above, it can be seen that the porosity and permeability are imposed at the center of the cube. Thus, the harmonic method must be applied to obtain their values on the faces. It is emphasized that the advection term in equation (2) is discretized with the upwind scheme. After performing all the operations, the x -direction momentum equation imposed on the x -direction face with its x -coordinate being i , y -coordinate ranging from j to $j + 1$ and z -coordinate ranging from k to $k + 1$ can be discretized as below:

$$\begin{aligned}
& \frac{\rho_f}{\Delta t} \left(\frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} - \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau}} \right) + \frac{\rho_f}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau}} \\
& \left(\bar{u}_y^{\tau} \times \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} - \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} \frac{u_{x,i-1,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i-1,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} + \right. \\
& \left. \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} \frac{u_{x,i,j-\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j-\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} - \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} \frac{u_{x,i,j+\frac{1}{2},k-\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k-\frac{1}{2}}^{\tau+1}} \right) \\
& \left(\bar{u}_z^{\tau} \times \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} - \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} \frac{u_{x,i,j+\frac{1}{2},k+1}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+1}^{\tau+1}} + \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} \frac{u_{x,i,j+\frac{1}{2},k-1}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k-1}^{\tau+1}} \right) \\
& = - \frac{p_{i+\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1} - p_{i-\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\Delta x} - \frac{\mu}{K_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1} \\
& + \mu \left(\frac{u_{x,i+1,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i+1,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} - \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} \frac{u_{x,i-1,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i-1,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} + \right. \\
& \left. \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} \frac{u_{x,i,j-\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j-\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} - \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} \frac{u_{x,i,j+\frac{1}{2},k-\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k-\frac{1}{2}}^{\tau+1}} \right) \\
& \left(\frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} \frac{u_{x,i,j+\frac{1}{2},k+1}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+1}^{\tau+1}} - \frac{u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}} \frac{u_{x,i,j+\frac{1}{2},k-1}^{\tau+1}}{\phi_{i,j+\frac{1}{2},k-1}^{\tau+1}} \right) \\
& - \frac{\rho_f F_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\sqrt{K_{i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}} \sqrt{\left(u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau} \right)^2 + \left(\bar{u}_y^{\tau} \right)^2 + \left(\bar{u}_z^{\tau} \right)^2} u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1} + \rho_f g_x,
\end{aligned}$$

in which \bar{u}_y and \bar{u}_z represent the y -direction average velocity and z -direction average velocity on the face, respectively. \bar{u}_y is computed as the average of the y -direction velocities on the four y -direction faces adjacent to the face. The computation of \bar{u}_z is similar. g_x is the x -direction component of \mathbf{g} . It is noted that in the above equation, it is assumed that,

$$u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau} > 0, \quad \bar{u}_y^{\tau} > 0 \quad \text{and} \quad \bar{u}_z^{\tau} > 0.$$

The discretization of the y -direction and z -direction momentum equations is similar. Under the Neumann boundary condition for pressure, the momentum conservation equation on the boundary degenerates to,

$$u = u_B,$$

in which u_B is the boundary normal velocity. The discretization of equation (3) is straightforward. Equation (4) is discretized at the center of the cube. For a cube with its x -coordinate being from i to $i+1$, y -coordinate being from j to $j+1$ and z -coordinate being from k to $k+1$, the left-hand side of equation (4) is discretized as,

$$\begin{aligned}
& \frac{\phi_{i+\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1} C_{f,i+\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1} - \phi_{i+\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}}^{\tau} C_{f,i+\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}}^{\tau}}{\Delta t} \\
& + \frac{u_{x,i+1,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1} C_{f,i+1,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1} - u_{x,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1} C_{f,i,j+\frac{1}{2},k+\frac{1}{2}}^{\tau+1}}{\Delta x} \\
& + \frac{u_{y,i+\frac{1}{2},j+1,k+\frac{1}{2}}^{\tau+1} C_{f,i+\frac{1}{2},j+1,k+\frac{1}{2}}^{\tau+1} - u_{y,i+\frac{1}{2},j,k+\frac{1}{2}}^{\tau+1} C_{f,i+\frac{1}{2},j,k+\frac{1}{2}}^{\tau+1}}{\Delta y} \\
& + \frac{u_{z,i+\frac{1}{2},j+\frac{1}{2},k+1}^{\tau+1} C_{f,i+\frac{1}{2},j+\frac{1}{2},k+1}^{\tau+1} - u_{z,i+\frac{1}{2},j+\frac{1}{2},k}^{\tau+1} C_{f,i+\frac{1}{2},j+\frac{1}{2},k}^{\tau+1}}{\Delta z},
\end{aligned}$$

with the upwind scheme being used to obtain the concentration value on the face of the cube, since the computed concentration is imposed at the center of the cube. If \mathbf{D}_e is written as,

$$\mathbf{D}_e = \begin{pmatrix} D_{xx} & D_{xy} & D_{xz} \\ D_{yx} & D_{yy} & D_{yz} \\ D_{zx} & D_{zy} & D_{zz} \end{pmatrix},$$

the first term of the right-hand side of equation (4) can be discretized as,

$$\begin{aligned}
& \frac{C^{\tau+1}_{f,i+\frac{3}{2},j+\frac{1}{2},k+\frac{1}{2}} - C^{\tau+1}_{f,i+\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}}}{\Delta x} \times \phi^{\tau+1}_{i+1,j+\frac{1}{2},k+\frac{1}{2}} \times D^{\tau+1}_{xx,i+1,j+\frac{1}{2},k+\frac{1}{2}} - \frac{C^{\tau+1}_{f,i+\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}} - C^{\tau+1}_{f,i-\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}}}{\Delta x} \times \phi^{\tau+1}_{i,j+\frac{1}{2},k+\frac{1}{2}} \times D^{\tau+1}_{xx,i,j+\frac{1}{2},k+\frac{1}{2}} \\
& \quad \Delta x \\
& + \frac{\frac{d\overline{C}_f}{dy}^{\tau+1}_{i+1,j+\frac{1}{2},k+\frac{1}{2}} \times \phi^{\tau+1}_{i+1,j+\frac{1}{2},k+\frac{1}{2}} \times D^{\tau+1}_{xy,i+1,j+\frac{1}{2},k+\frac{1}{2}} - \frac{d\overline{C}_f}{dy}^{\tau+1}_{i,j+\frac{1}{2},k+\frac{1}{2}} \times \phi^{\tau+1}_{i,j+\frac{1}{2},k+\frac{1}{2}} \times D^{\tau+1}_{xy,i,j+\frac{1}{2},k+\frac{1}{2}}}{\Delta x} \\
& + \frac{\frac{d\overline{C}_f}{dz}^{\tau+1}_{i+1,j+\frac{1}{2},k+\frac{1}{2}} \times \phi^{\tau+1}_{i+1,j+\frac{1}{2},k+\frac{1}{2}} \times D^{\tau+1}_{xz,i+1,j+\frac{1}{2},k+\frac{1}{2}} - \frac{d\overline{C}_f}{dz}^{\tau+1}_{i,j+\frac{1}{2},k+\frac{1}{2}} \times \phi^{\tau+1}_{i,j+\frac{1}{2},k+\frac{1}{2}} \times D^{\tau+1}_{xz,i,j+\frac{1}{2},k+\frac{1}{2}}}{\Delta x} \\
& + \frac{\frac{d\overline{C}_f}{dx}^{\tau+1}_{i+\frac{1}{2},j+1,k+\frac{1}{2}} \times \phi^{\tau+1}_{i+\frac{1}{2},j+1,k+\frac{1}{2}} \times D^{\tau+1}_{yx,i+\frac{1}{2},j+1,k+\frac{1}{2}} - \frac{d\overline{C}_f}{dx}^{\tau+1}_{i+\frac{1}{2},j,k+\frac{1}{2}} \times \phi^{\tau+1}_{i+\frac{1}{2},j,k+\frac{1}{2}} \times D^{\tau+1}_{yx,i+\frac{1}{2},j,k+\frac{1}{2}}}{\Delta y} \\
& + \frac{\frac{C^{\tau+1}_{f,i+\frac{1}{2},j+\frac{3}{2},k+\frac{1}{2}} - C^{\tau+1}_{f,i+\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}}}{\Delta y} \times \phi^{\tau+1}_{i+\frac{1}{2},j+1,k+\frac{1}{2}} \times D^{\tau+1}_{yy,i+\frac{1}{2},j+1,k+\frac{1}{2}} - \frac{C^{\tau+1}_{f,i+\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}} - C^{\tau+1}_{f,i+\frac{1}{2},j-\frac{1}{2},k+\frac{1}{2}}}{\Delta y} \times \phi^{\tau+1}_{i+\frac{1}{2},j,k+\frac{1}{2}} \times D^{\tau+1}_{yy,i+\frac{1}{2},j,k+\frac{1}{2}}}{\Delta y} \\
& + \frac{\frac{d\overline{C}_f}{dz}^{\tau+1}_{i+\frac{1}{2},j+1,k+\frac{1}{2}} \times \phi^{\tau+1}_{i+\frac{1}{2},j+1,k+\frac{1}{2}} \times D^{\tau+1}_{yz,i+\frac{1}{2},j+1,k+\frac{1}{2}} - \frac{d\overline{C}_f}{dz}^{\tau+1}_{i+\frac{1}{2},j,k+\frac{1}{2}} \times \phi^{\tau+1}_{i+\frac{1}{2},j,k+\frac{1}{2}} \times D^{\tau+1}_{yz,i+\frac{1}{2},j,k+\frac{1}{2}}}{\Delta y} \\
& + \frac{\frac{d\overline{C}_f}{dx}^{\tau+1}_{i+\frac{1}{2},j+\frac{1}{2},k+1} \times \phi^{\tau+1}_{i+\frac{1}{2},j+\frac{1}{2},k+1} \times D^{\tau+1}_{zx,i+\frac{1}{2},j+\frac{1}{2},k+1} - \frac{d\overline{C}_f}{dx}^{\tau+1}_{i+\frac{1}{2},j+\frac{1}{2},k} \times \phi^{\tau+1}_{i+\frac{1}{2},j+\frac{1}{2},k} \times D^{\tau+1}_{zx,i+\frac{1}{2},j+\frac{1}{2},k}}{\Delta z} \\
& + \frac{\frac{d\overline{C}_f}{dy}^{\tau+1}_{i+\frac{1}{2},j+\frac{1}{2},k+1} \times \phi^{\tau+1}_{i+\frac{1}{2},j+\frac{1}{2},k+1} \times D^{\tau+1}_{zy,i+\frac{1}{2},j+\frac{1}{2},k+1} - \frac{d\overline{C}_f}{dy}^{\tau+1}_{i+\frac{1}{2},j+\frac{1}{2},k} \times \phi^{\tau+1}_{i+\frac{1}{2},j+\frac{1}{2},k} \times D^{\tau+1}_{zy,i+\frac{1}{2},j+\frac{1}{2},k}}{\Delta z} \\
& + \frac{\frac{C^{\tau+1}_{f,i+\frac{1}{2},j+\frac{1}{2},k+\frac{3}{2}} - C^{\tau+1}_{f,i+\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}}}{\Delta z} \times \phi^{\tau+1}_{i+\frac{1}{2},j+\frac{1}{2},k+1} \times D^{\tau+1}_{zz,i+\frac{1}{2},j+\frac{1}{2},k+1} - \frac{C^{\tau+1}_{f,i+\frac{1}{2},j+\frac{1}{2},k+\frac{1}{2}} - C^{\tau+1}_{f,i+\frac{1}{2},j+\frac{1}{2},k-\frac{1}{2}}}{\Delta z} \times \phi^{\tau+1}_{i+\frac{1}{2},j+\frac{1}{2},k} \times D^{\tau+1}_{zz,i+\frac{1}{2},j+\frac{1}{2},k}}{\Delta z} .
\end{aligned}$$

The notation $\frac{d\overline{C}_f}{dy}^{\tau+1}$ stands for the average value of $\frac{dC_f}{dy}$ on the x -direction face with its x -coordinate being $i + 1$, which is calculated by the four $\frac{dC_f}{dy}$ on the four y -direction faces adjacent to the x -direction face. The meanings of the other similar notations are analogous. The discretization of the second term of the right-hand side of equation (4) is trivial and thus not given here. It is easy to see that the stencil pattern of T is the same as C_f ; thus, the discretization of equation (21) holds the same philosophy as that of equation (4), and the details are no longer given.

To use the experimenting field approach to compute the coefficients of the three linear systems, the unknowns to be computed can be divided into four fields: the velocity field, the pressure field, the concentration field, and the temperature field. If there is a 3D domain and it can be divided into eight cubes as shown in Figure 6, then the velocity field can be represented as arrows on the faces of the cubes, and the pressure field, concentration field and temperature field can be represented as points at the centers of the cubes. Each x -momentum, y -momentum, and z -momentum conservation equation can be discretized on each x -direction, y -direction,

and z -direction face, respectively. Each mass conservation equation, concentration balance equation, and energy conservation equation can be discretized at each center of the cube. This kind of grid is called a staggered grid in CFD. The experimenting field approach used in the 2D simulation [19] is expanded to the 3D case directly, and the details are no longer given in this work.

To capture the details of the configuration of the matrix after acidization, a fine 3D grid is needed in the simulation, which brings about a large number of cells in the 3D grid, and as a result, it is necessary to introduce parallelization in the simulation. In the first step, domain decomposition must be performed on the 3D domain. The main purpose of domain decomposition is to allocate the discretized equations to the processors. Suppose there is a 3D Cartesian grid with nx , ny , and nz cubes in the x -, y -, and z -directions, respectively, and there are np_x , np_y , and np_z processors in the x -, y -, and z -directions, respectively. nx , ny , and nz are supposed to be divisible by np_x , np_y , and np_z , respectively. Furthermore, it is stipulated that $\frac{nx}{np_x} \geq 2$, $\frac{ny}{np_y} \geq 2$, and $\frac{nz}{np_z} \geq 2$. For the processor with the coordinate (I, J, K) , the

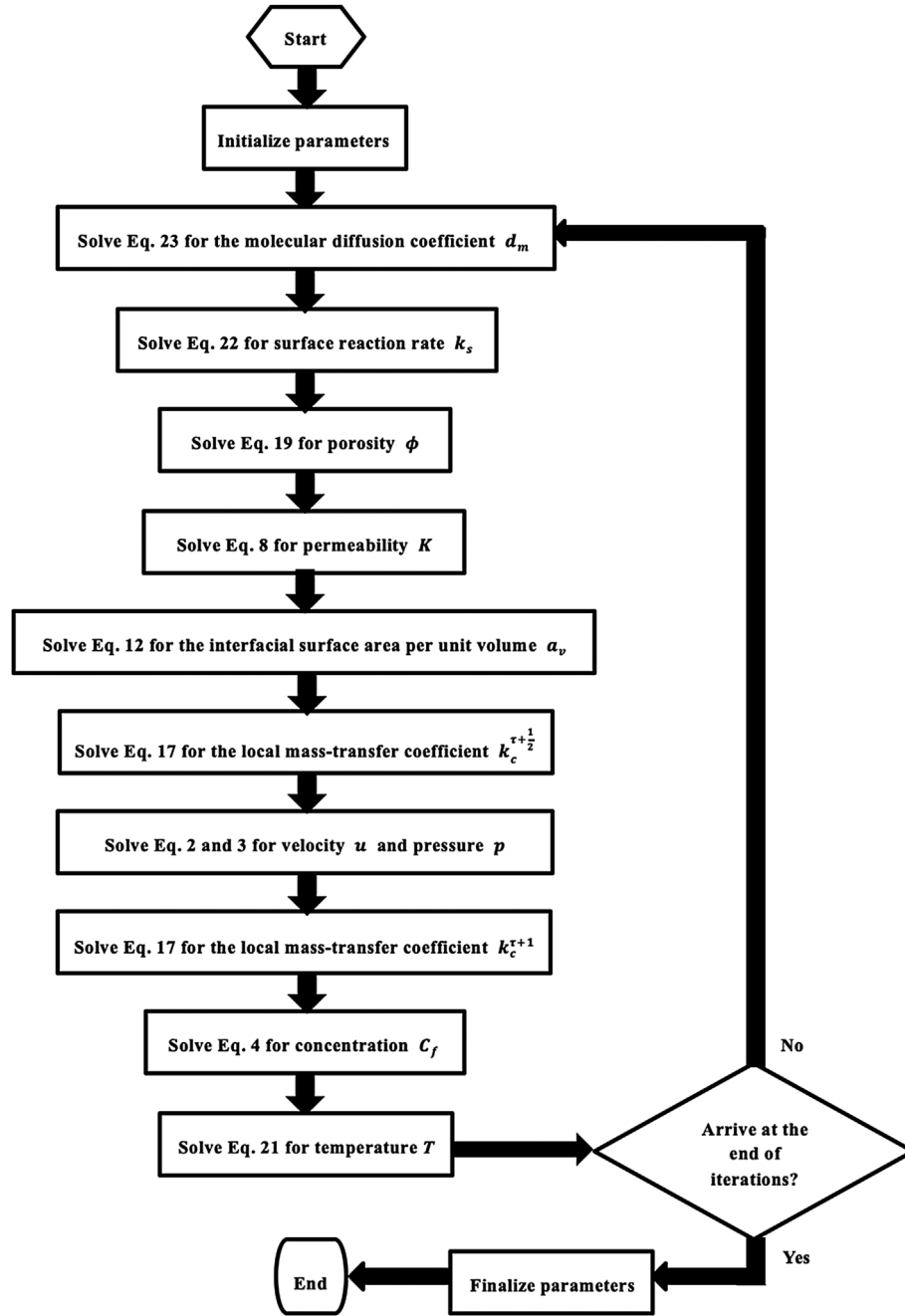


Fig. 3. The flowchart of the thermal DBF framework.

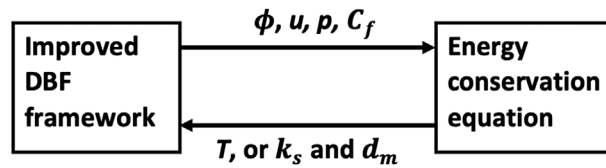


Fig. 4. The interaction of the improved DBF framework and the energy conservation equation.

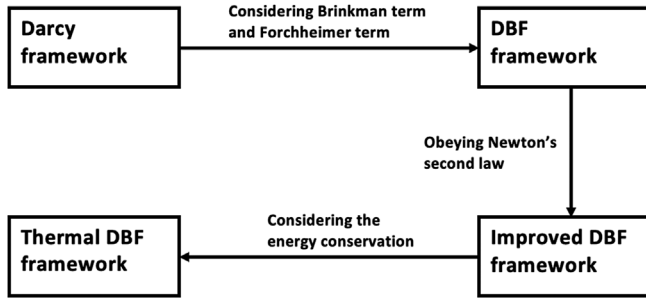


Fig. 5. The relationship among the Darcy framework, the DBF framework, the improved DBF framework, and the thermal DBF framework.

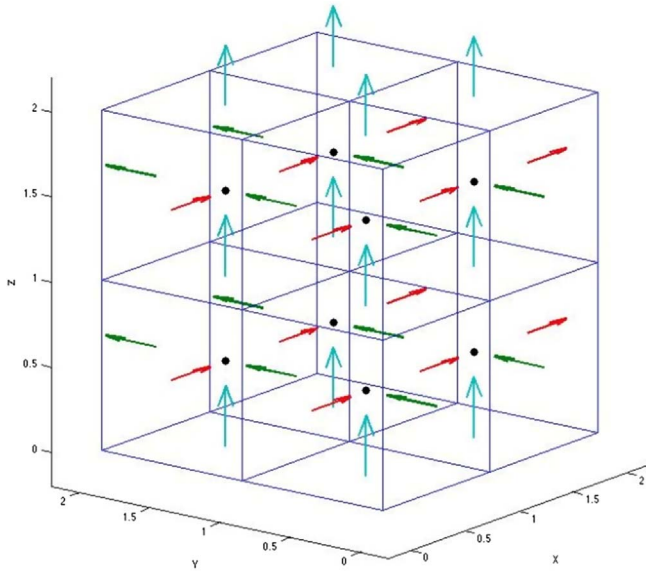


Fig. 6. Staggered grid.

following equations discretized at the centers of the cubes with the coordinate (i, j, k) are allocated to it,

$$(I - 1) \times \frac{nx}{np_x} + 1 \leq i \leq I \times \frac{nx}{np_x},$$

$$(J - 1) \times \frac{ny}{np_y} + 1 \leq j \leq J \times \frac{ny}{np_y},$$

$$(K - 1) \times \frac{nz}{np_z} + 1 \leq k \leq K \times \frac{nz}{np_z},$$

$$1 \leq I \leq np_x, \quad 1 \leq J \leq np_y, \quad 1 \leq K \leq np_z,$$

and the equations discretized on the x -direction faces with the coordinate (i, j, k) , which are the x -momentum conservation equations, are allocated to it as,

$$(I - 1) \times \frac{nx}{np_x} + 1 \leq i \leq I \times \frac{nx}{np_x} + \delta_x,$$

$$(J - 1) \times \frac{ny}{np_y} + 1 \leq j \leq J \times \frac{ny}{np_y},$$

$$(K - 1) \times \frac{nz}{np_z} + 1 \leq k \leq K \times \frac{nz}{np_z},$$

$$\delta_x = \begin{cases} 1, & \text{if } I = np_x \\ 0, & \text{otherwise} \end{cases},$$

$$1 \leq I \leq np_x, \quad 1 \leq J \leq np_y, \quad 1 \leq K \leq np_z.$$

The allocation strategy of the y -momentum conservation equations and the z -momentum conservation equations is similar. After the allocation of the discretized equations, the variables that are needed by the equations should also be allocated to the processors, with some variables being communicated among the processors. By the domain decomposition strategy, each discretized equation can be allocated to only one processor, with the benefits that it can keep load balance of the processors and reduce the communication cost among the processors.

After domain decomposition, a suitable parallel solver can be leveraged to solve the three linear systems. In the 2D parallel simulation of the work [19], the parallel solver HYPRE is used. There are many numerical algorithms that can be used in HYPRE, such as the Generalized Minimal RESidual method (GMRES) and the Algebraic Multigrid Method (AMG). However, few simple cases can be solved by these algorithms. As a result, more complicated cases have to be solved by the direct solver UMFPACK [43] in a serial code, which limits the application of the parallel code. The main issue of these complicated cases is that their condition number is too large. In this work, another parallel solver called MUMPS is used, which can solve complicated cases that HYPRE cannot solve. With the help of MUMPS, most of the cases for the improved DBF framework and the thermal DBF framework can be run in parallel, which makes the fine 3D simulation feasible. In MUMPS, a variant of the Gaussian elimination method – the multifrontal method – is used. This method can solve a large sparse system of equations in parallel: the equation system is first divided into independent subsets, which are called fronts, and then the fronts are processed on different processors simultaneously. It is emphasized that unlike HYPRE, which is an iterative solver, MUMPS is a direct solver, which makes the time step larger in the simulation. Moreover, the direct solver can solve the linear system directly, with no need to add a pseudo parameter e in the mass conservation equation, which is the case in the work [19]. There, e is introduced to ensure a linear system with an invertible coefficient matrix; otherwise, the iterative solver

HYPRE cannot solve it. However, the introduction of e changes the attribute of the flow in matrix acidization from incompressible to slightly compressible, which contradicts the real case and makes the DBF framework less reliable. Last, with the help of FORTRAN 90 and MPI, a series of 2D and 3D parallel codes are developed. In the following sections, these codes are used to run a series of numerical experiments on the Shaheen supercomputer [44].

5 Verification of the improved DBF framework

5.1 3D shear-driven cavity flows

It can be seen that the model of shear-driven cavity flows [45] is in fact a reduction of the improved DBF framework, since equations (2) and (3) reduce to the following two equations:

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{u}\mathbf{u} = -\nabla p + \frac{1}{Re} \nabla^2 \mathbf{u},$$

$$\nabla \cdot \mathbf{u} = 0,$$

which describes shear-driven cavity flows. Therefore, the 3D code of the improved DBF framework can be used to simulate 3D shear-driven cavity flows identically as long as some parameters are simplified, such as ignoring the concentration balance equation. In terms of 3D shear-driven cavity flows, a laminar incompressible flow is inside a unit cube cavity whose y -direction top surface is moved by an x -direction uniform velocity of 1 m/s, as shown in Figure 7. The Reynolds number (Re) is 100. The gravity effect is ignored. The grid has 20^3 cubes. The simulation results of stable flows are shown in Figures 8 and 9, respectively. The two figures display the velocity profiles of the x -direction component on the vertical centerline and the y -direction component on the horizontal centerline of the plane $z = 0.5$. The simulation results can be compared with Figure 6 in [45]. To the naked eye, we cannot clarify their differences, which proves the correctness of the 3D code of the improved DBF framework to some extent.

5.2 2D linear flows

The 2D linear flows in a previous work [17] are simulated again by the improved DBF framework, with more or less the same experimental parameters, which are shown in Table 2. In [17], the flows are simulated with the Darcy framework [15], but Navier-Stokes fluid dynamics are considered. Thus, among all the state-of-the-art models developed from the Darcy framework, the model is close to the improved DBF framework, and its results can be compared with the results from the improved DBF framework. It is noted that the subscript 0 represents the initial value. ϕ_0 represents the initial average porosity in the medium, with a heterogeneity magnitude of 0.03. The gravity effect is ignored. In the 2D simulation, there is a rectangular matrix of 0.1-m length (x -direction) and 0.04-m width

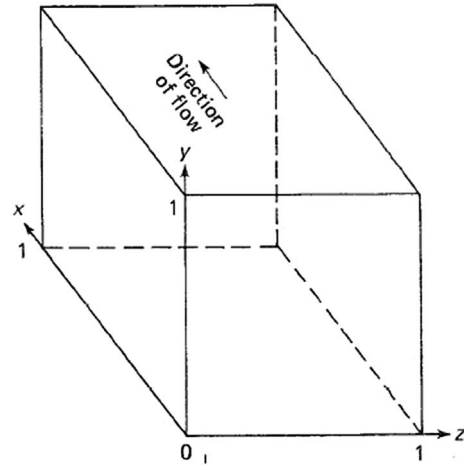


Fig. 7. 3D shear-driven cavity flow configuration and coordinate system [45].

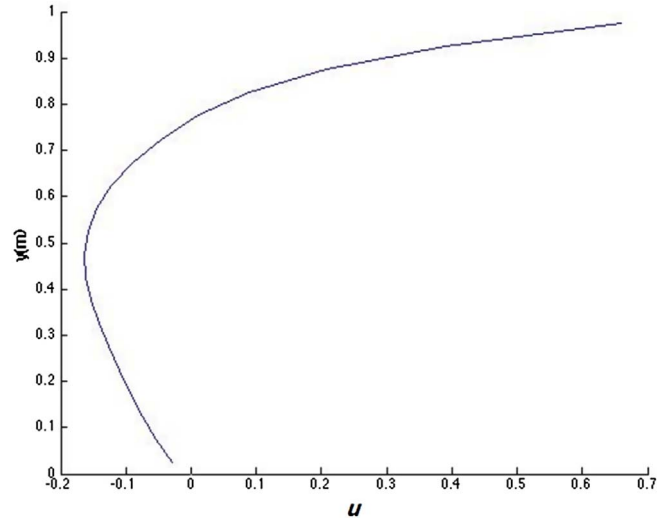


Fig. 8. Velocity profile of the x -direction component (u) on the vertical centerline of the plane $z = 0.5$.

(y -direction). Acid flow is injected into the matrix from the left boundary and goes out of the matrix from the right boundary, which means that the injected velocity is imposed on the left boundary, and the Dirichlet boundary condition for pressure is imposed on the right boundary. It is stipulated that the pressure imposed on the right boundary is the same as the initial pressure in the matrix. For concentration, the Dirichlet boundary condition is imposed on the left boundary, and the no-flux boundary condition is imposed on the right boundary. The upper and lower boundaries are closed for both pressure and concentration, which means that no-flow, no-flux boundary conditions are imposed. The acid concentration is initially zero in the matrix. The injected velocity of the acid flow of 0.5 M hydrogen chloride (HCl) on the left boundary is changed in the simulation, which leads to different

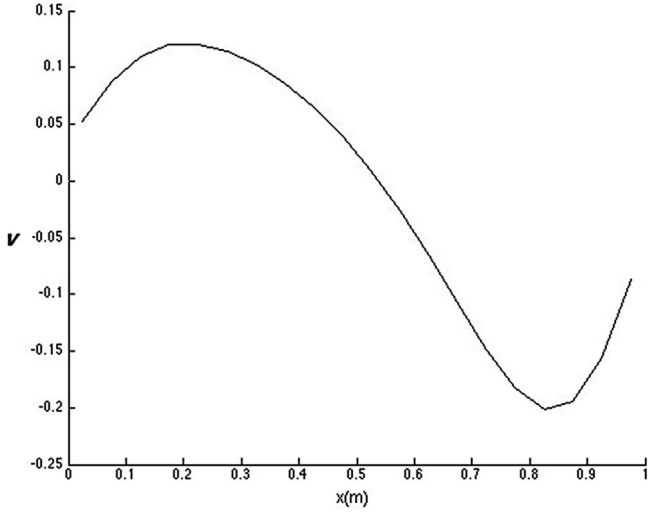


Fig. 9. Velocity profile of the y -direction component (v) on the horizontal centerline of the plane $z = 0.5$.

Table 2. Experimental parameters.

Parameter	Value
p_0	1.52×10^7 Pa
μ	1.0×10^{-3} kg/(m·s)
K_0	9.869233×10^{-16} m ²
$\bar{\phi}_0$	1.8×10^{-1}
ρ_f	1.01×10^3 kg/m ³
C_f	5.0×10^2 mol/m ³
d_m	3.6×10^{-9} m ² /s
α_{OS}	5.0×10^{-1}
λ_X	5.0×10^{-1}
λ_T	1.0×10^{-1}
r_0	1.0×10^{-6} m
a_{i0}	5×10^{-1} m ⁻¹
α	5.0×10^{-2} kg/mol
ρ_s	2.71×10^3 kg/m ³
k_s	2.0×10^{-3} m/s
Sh_∞	3.66
λ_f	5.8×10^{-1} W/(m·K)
λ_s	5.526 W/(m·K)
θ_f	4.184×10^3 J/(kg·K)
θ_s	2.0×10^2 J/(kg·K)
k_{s0}	2.0×10^{-3} m/s (298 K)
d_{m0}	3.6×10^{-9} m ² /s (298 K)
E_g	5.02416×10^4 J/mol
R_g	8.314 J/(K·mol)

configurations of the matrix after acidization. The grid has 180 cells in the x -direction and 72 cells in the y -direction, which is the same size as that in [17]. Since the author of [17] declared that their grid is fine enough to describe matrix acidization, our grid is also capable of doing so.

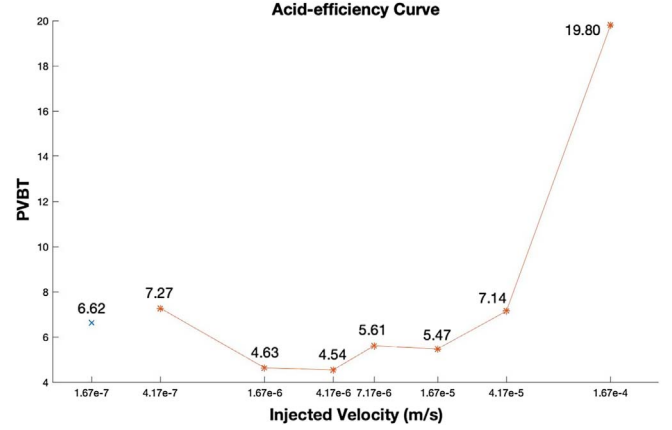


Fig. 10. Acid-efficiency curve of the 2D linear flows. The numbers beside the points represent the values of PVBT.

Table 3. Time steps for the injected velocities of the 2D linear flows.

Velocity (m/s)	Time step (s)
1.67×10^{-7}	1579
4.17×10^{-7}	643
1.67×10^{-6}	150
4.17×10^{-6}	60
7.17×10^{-6}	16
1.67×10^{-5}	16.5
4.17×10^{-5}	6.43
1.67×10^{-4}	1.5

The Pore Volumes to BreakThrough (PVBT) of different injected velocities are given in Figure 10, and the curve in the figure is called the acid-efficiency curve. Break-through is defined as the moment when the pressure drop across the medium drops to 1% of its initial value [46]. In Figure 10, it can be seen that the acid-efficiency curve in this work matches the corresponding curve in Figure 8 of [17] well when injected velocity ranges from 4.17×10^{-7} m/s to 1.67×10^{-4} m/s. When the injected velocity is 1.67×10^{-7} m/s, its corresponding PVBT is 6.62, which is not reasonable. The injected velocity of 1.67×10^{-7} m/s is very slow, which may indicate a face dissolution pattern. According to [16], under such conditions, approximately 100 million grid cells are needed to capture the face dissolution pattern accurately, which indicates that our grid is not fine enough to output accurate results. Unfortunately, due to the limits of supercomputing power, a simulation based on 100 million grid cells cannot be finished in a reasonable time, so it is not performed in this work. The minimum PVBT is 4.54, which is achieved at an injected velocity of 4.17×10^{-6} m/s.

A fixed time step is assumed for the simulations, and the time steps corresponding to the injected velocities are

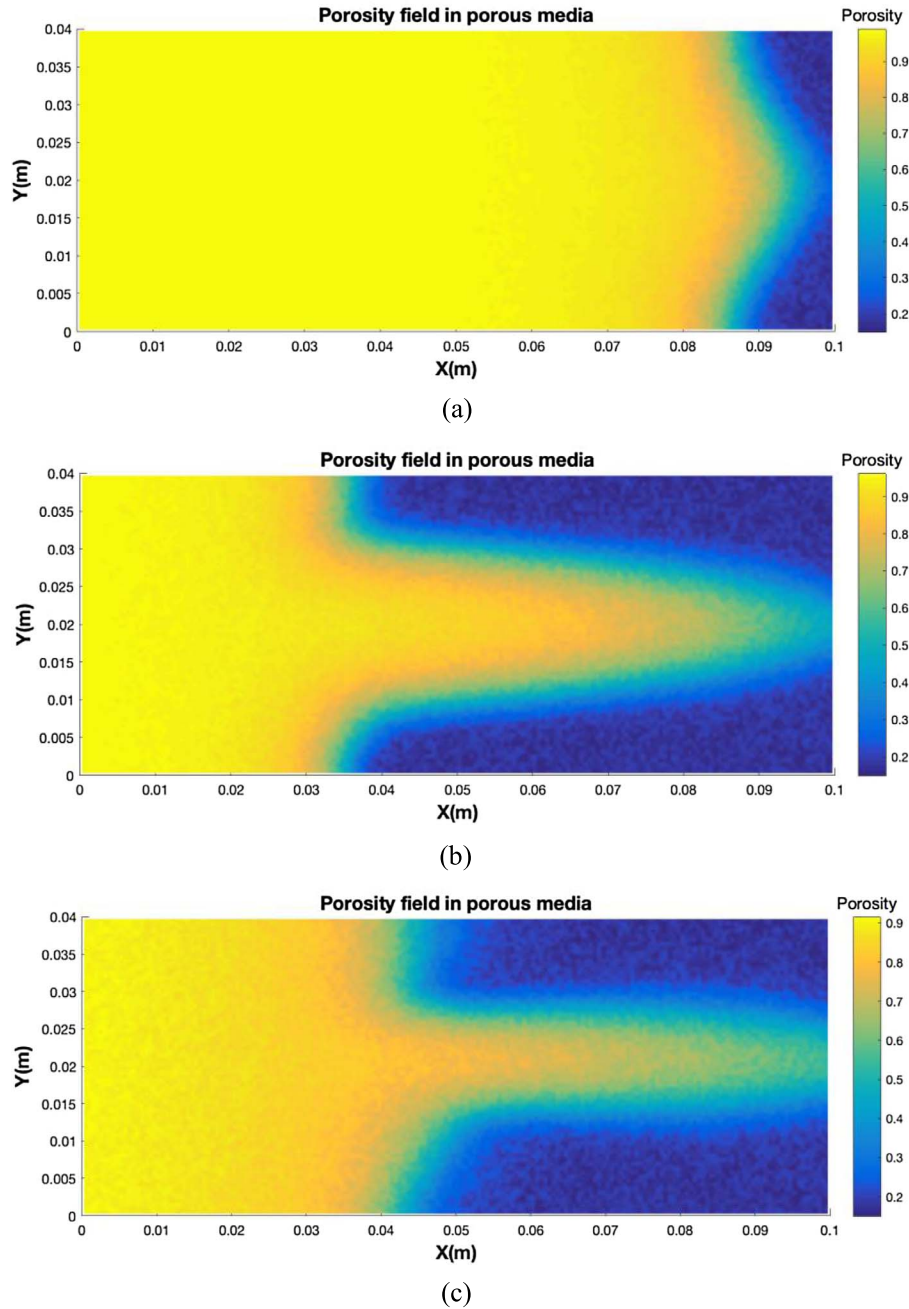


Fig. 11. Porosity profiles at breakthrough for five different injected velocities of the 2D linear flows. (a) $u_x = 4.17 \times 10^{-7}$ m/s, face dissolution. (b) $u_x = 1.67 \times 10^{-6}$ m/s, conical wormhole. (c) $u_x = 4.17 \times 10^{-6}$ m/s, dominant wormhole. (d) $u_x = 7.17 \times 10^{-6}$ m/s, ramified wormhole. (e) $u_x = 1.67 \times 10^{-5}$ m/s, uniform dissolution.

shown in Table 3. It is emphasized that all the time steps ensure that the Courant number is less than one. A sensitivity test is performed when each of the time steps is increased by two times to assure that the Courant number is less than one and all the values of PVBT are the same, which demonstrates that the simulation results in Figure 10 can be deemed the true results.

The porosity profiles at breakthrough corresponding to five different injected velocities are given in Figure 11.

In the figure, it can be seen that five dissolution patterns appear in their turns when the injected velocity increases.

5.3 3D linear flows

The simulation of 2D linear flows has achieved reasonable results, and it can be expanded to the simulation of 3D linear flows by adding another dimension to the matrix, with a length of 0.04 m. After this addition, a 3D matrix is created

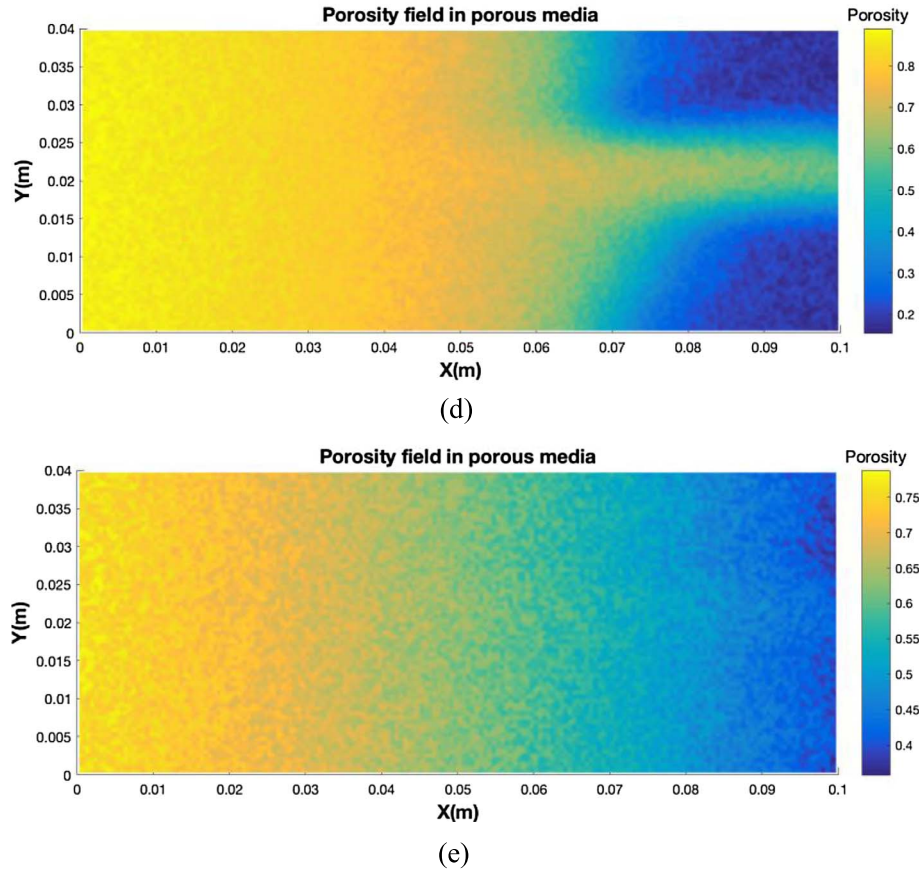


Fig. 11. (Continued)

Table 4. Time steps for the injected velocities of the 3D linear flows.

Velocity (m/s)	Coarse time step (s)	Fine time step (s)
1.04×10^{-7}	9322	4661
3.04×10^{-7}	3611	/
7.04×10^{-7}	984	492
1.04×10^{-6}	593.2	296.6
3.04×10^{-6}	237.2	118.6
7.04×10^{-6}	139	69.5
1.04×10^{-5}	91.6	45.8
1.04×10^{-4}	21.2	/

with a 0.1-m length in the z -direction and 0.04-m length in the x - and y -directions, respectively. According to [17], dissolution patterns from a conical wormhole to uniform dissolution can be captured accurately only when the grid has at least 180 cells in the z -direction and 72 cells in the x -direction and y -direction. However, the face dissolution pattern requires a finer grid. To ensure that the Courant number is less than one, the number of iteration steps can be large, which brings about a long simulation period. Even though

the code runs on Shaheen, at least one month is needed for the fastest case to achieve breakthrough. Thus, that kind of grid currently exceeds computing capacity, and a coarser grid is thus given in this work. The number of cells is divided by two in each dimension, and a coarser grid, with 90 cells in the z -direction and 36 cells in the x -direction and y -direction, is used to simulate the 3D linear flows. Although the coarser grid is not fine enough to capture all dissolution patterns accurately, the simulation results can still be used to verify the correctness of the 3D code of the improved DBF framework. The acid flow is injected into the matrix along the z -direction. The other experimental parameters and boundary conditions are the same as the 2D simulation.

The time steps for different injected velocities are shown in Table 4. There are two groups of time steps, with the values of the second column being two times the corresponding values of the third column, the purpose of which is to test the convergence of the results. All the time steps guarantee that the Courant number is less than one. It is emphasized that due to the limits of Shaheen, a code can run on the supercomputer for three days at most. Therefore, simulations for velocities of 3.04×10^{-7} m/s and 1.04×10^{-4} m/s at fine time steps are not performed since their simulation time to achieve breakthrough is beyond three days. The acid-efficiency curves for both groups of

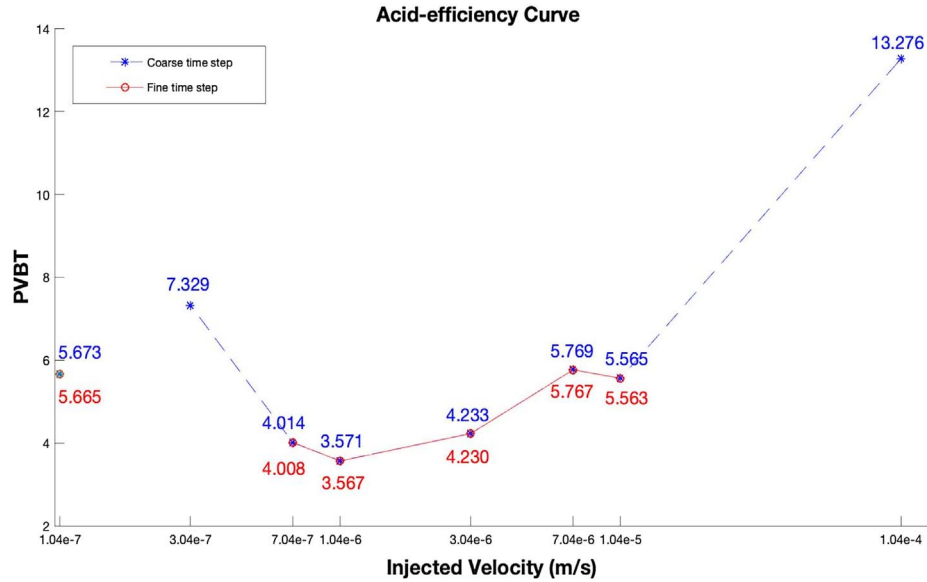


Fig. 12. Acid-efficiency curves of the 3D linear flows. The numbers beside the points represent the values of PVBT.

time steps are shown in Figure 12. The numbers beside the points represent the values of PVBT, with the blue points coming from coarse time steps and the red points coming from fine time steps. In the figure, it can be seen that the values of PVBT from coarse time steps are very close to those from fine time steps, which demonstrates the convergence of the results. Moreover, the values of PVBT tend to decrease with finer time steps. It can be expected that with finer grids, smaller PVBT values can be achieved. The minimum PVBT is 3.567, which is achieved at the injected velocity of 1.04×10^{-6} m/s. This coincides with the work [17], where the minimum PVBT is achieved at the injected velocity of $0.1 \text{ cm}^3/\text{min}$, which is the same as 1.04×10^{-6} m/s. However, in [17], the minimum PVBT is approximately two, which is smaller than ours due to the finer grid. Except for the points at the injected velocity of 1.04×10^{-7} m/s, the shape of the acid-efficiency curves matches the corresponding shape in the [17] well. The drop in PVBT values at that velocity is due to the inaccuracy of the simulation when the grid is not fine enough, which can also be seen in the simulation of 2D linear flows above. Furthermore, the minimum PVBT of 2D simulations is larger than that of 3D simulations, and the injected velocity of 2D simulations at which the minimum PVBT is achieved is also larger than that of 3D simulations, which conforms to the qualitative trends in [15].

These effects of the injected velocities on dissolution patterns are shown in Figure 13. It can be seen in the figure that five different dissolution patterns can be simulated.

6 Verification of the thermal DBF framework

6.1 Isothermal conditions

The correctness of the improved DBF framework is a major premise of the thermal DBF framework, which has been

verified in the last section. The correctness of the thermal DBF framework is discussed in this section. First, an experiment is carried out with isothermal conditions in which the injected acid temperature and the initial matrix temperature are the same. Since 2D experiments are eligible to verify the correctness of the model, the grid of the 2D linear flows is used again, with 180×72 cells in total. 3D experiments are left to future work. To compare the numerical results with the chemical results of the “effects of temperature” experiment in [32], three temperatures are chosen: 295 K, 323 K, and 353 K, which correspond to 22 °C, 50 °C, and 80 °C in [32]. The boundary conditions and initial conditions for pressure and concentration are the same as those in the 2D linear flow experiment above. In addition, for temperature, adiabatic conditions are applied, which means that except for the acid injection boundary (left boundary), all the other boundaries are adiabatic boundaries. The experimental parameters can be seen in Table 2. It is noted that the values of d_m and k_s in Table 2 are not used in the experiments of this section, since they are variables in the thermal DBF framework. All the parameters are more or less the same as those in the “effects of temperature” experiment of [32].

The values of PVBT for different injected velocities and temperatures are shown in Table 5. It can be seen in the table that when the temperature is 295 K, the minimal PVBT is 4.350, which is achieved at the optimal injected velocity of 2.67×10^{-6} m/s; when the temperature is 323 K, the minimal PVBT is 4.362, which is achieved at the optimal injected velocity of 9.17×10^{-6} m/s; when the temperature is 353 K, the minimal PVBT is 4.416, which is achieved at the optimal injected velocity of 4.17×10^{-5} m/s. To verify the convergence of the results, both coarse-time-step and fine-time-step results are computed. The coarse time step is two times the fine time step for every injected velocity. All the time steps can guarantee that the Courant number is less than one. It can be seen

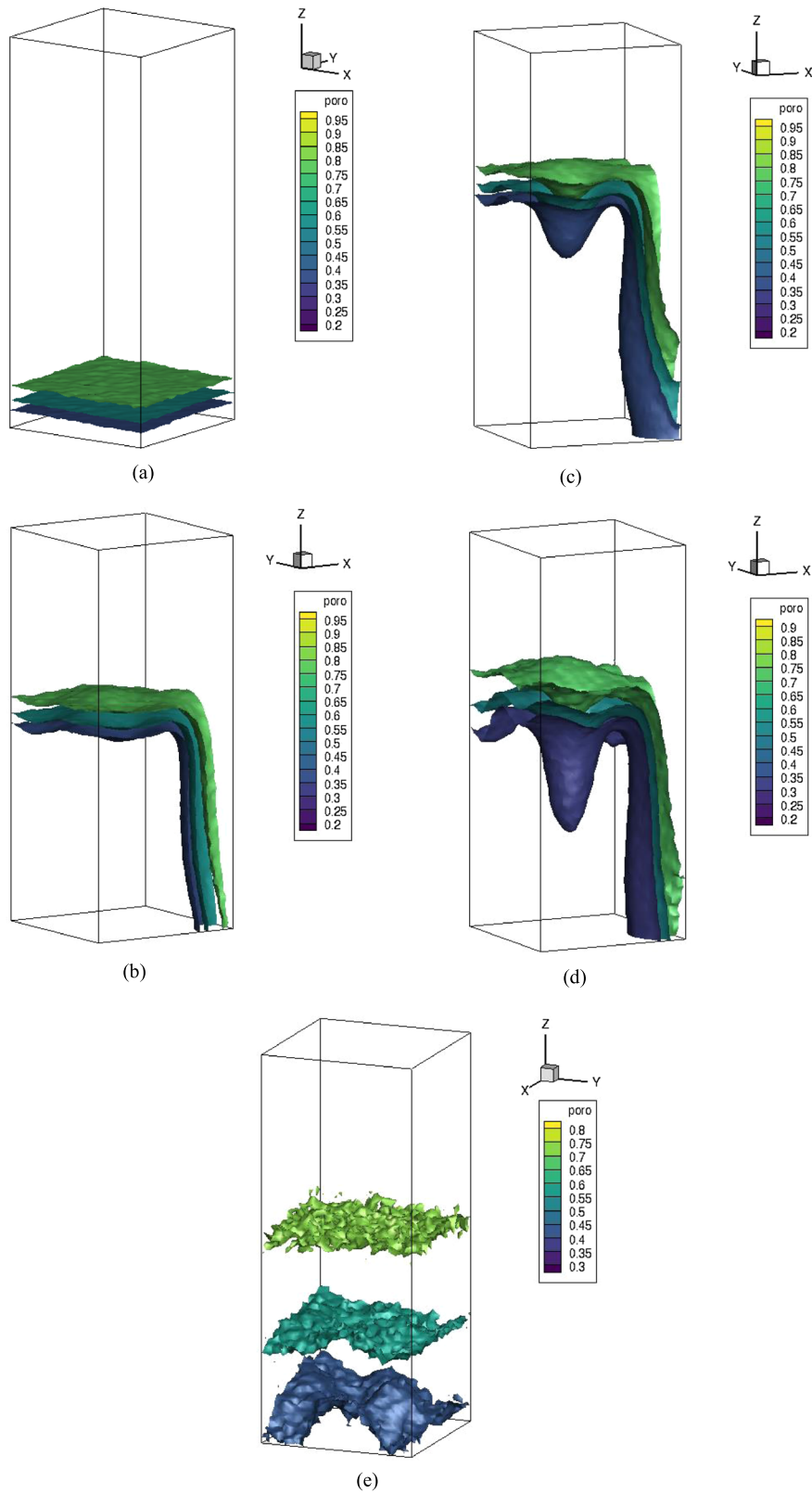
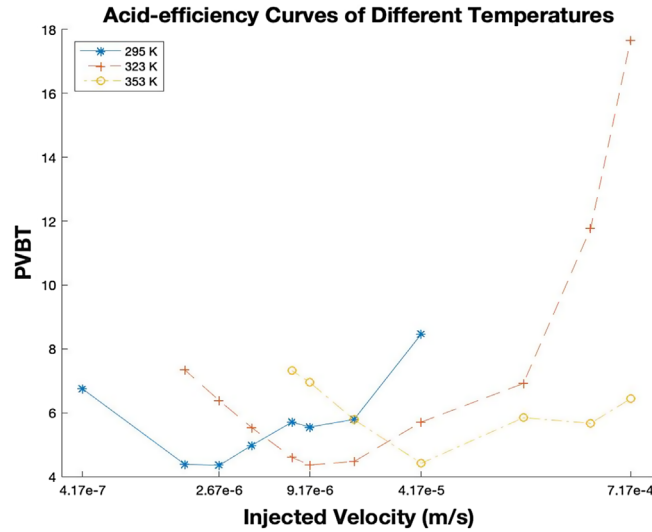


Fig. 13. Porosity isosurfaces at breakthrough for five different injected velocities of the 3D linear flows. (a) $u_z = 3.04 \times 10^{-7}$ m/s, face dissolution. (b) $u_z = 7.04 \times 10^{-7}$ m/s, conical wormhole. (c) $u_z = 1.04 \times 10^{-6}$ m/s, dominant wormhole. (d) $u_z = 3.04 \times 10^{-6}$ m/s, ramified wormhole. (e) $u_z = 1.04 \times 10^{-5}$ m/s, uniform dissolution.

Table 5. Values of PVBt. The first row represents injected velocities, and their unit is m/s.

	4.17×10^{-7}	1.67×10^{-6}	2.67×10^{-6}	4.17×10^{-6}	7.17×10^{-6}	9.17×10^{-6}	1.67×10^{-5}	4.17×10^{-5}	1.67×10^{-4}	4.17×10^{-4}	7.17×10^{-4}
295 K coarse time step	6.750	4.387	4.352	4.982	5.703	5.553	5.789	8.465	/	/	/
295 K fine time step	6.745	4.385	4.350	4.981	5.703	5.552	5.788	8.465	/	/	/
323 K coarse time step	/	7.348	6.387	5.530	4.602	4.364	4.473	5.708	6.917	11.771	17.666
323 K fine time step	/	7.344	6.383	5.528	4.601	4.362	4.471	5.708	6.916	11.771	17.665
353 K coarse time step	/	/	/	/	7.321	6.952	5.785	4.418	5.844	5.669	6.438
353 K fine time step	/	/	/	/	7.319	6.947	5.783	4.416	5.843	5.668	6.437

**Fig. 14.** Acid efficiency curves of different temperatures in isothermal conditions.

in the table that the differences of the coarse-time-step and fine-time-step results are very small, which means convergence is achieved. The values of PVBt from fine time steps constitute the acid-efficiency curves of different temperatures in Figure 14. Since these experimental parameters hint that the injected acid is 0.5 M HCl, the matrix is limestone; Figure 14 is compared with Figure 6 in [32]. In the two figures, it is evident that both the minimal PVBt and the optimal injected velocity increase when the temperature increases, which means that the numerical simulation results can be observed in labs. It is noted that when the injected velocity is below approximately 4.17×10^{-6} m/s, the values of PVBt increase with increasing temperature; when the injected velocity is above approximately 4.17×10^{-5} m/s, the values of PVBt decrease with increasing temperature, which means that the former is a mass-transfer controlled regime and the latter is a kinetically

controlled regime. The porosity profiles at breakthrough in the optimal injected velocity are given in Figure 15 for the three different temperatures. It can be seen in the figure that with the increase of the temperature, the diameter of the wormhole also increases, which matches the observation in Figure 7 of [32]. This also explains why the minimal PVBt value increases with increasing temperature. In fact, the transferring efficiency of HCl decreases due to increased acid consumption on the walls of the wormhole, which brings about the phenomenon above.

6.2 Nonisothermal conditions

Isothermal conditions are common in labs. However, in field cases, the injected acid temperature is often different from the initial matrix temperature. Due to the geothermal factor, the initial matrix temperature may be higher than

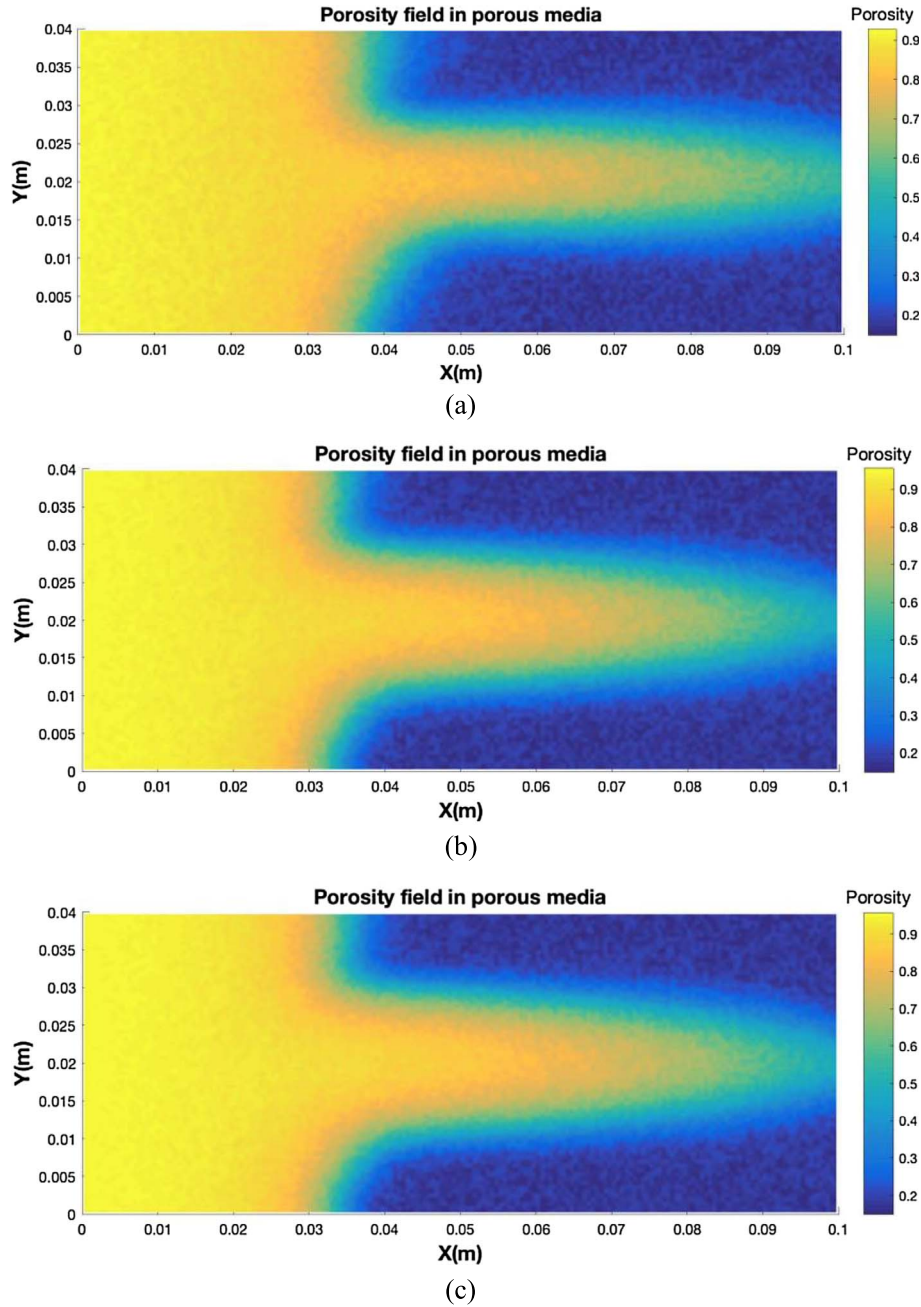


Fig. 15. Porosity profiles at breakthrough in the optimal injected velocity for three different temperatures. (a) 295 K, $u_x = 2.67 \times 10^{-6}$ m/s. (b) 323 K, $u_x = 9.17 \times 10^{-6}$ m/s. (c) 353 K, $u_x = 4.17 \times 10^{-5}$ m/s.

the injected acid temperature. Thus, to expand the simulations from labs to fields, nonisothermal conditions are considered. In the experiment, four cases with different combinations of the injected acid temperature and initial matrix temperature are simulated, with an injected velocity of 9.17×10^{-6} m/s, and the values of PVBT are shown in Table 6. The other experimental parameters are the same as the isothermal conditions. All the results are from the fine-time-step simulations. It can be seen in Table 6 that when the injected acid temperature is fixed at 323 K, the

Table 6. Values of PVBT for nonisothermal conditions.

Injected acid temperature (K)	Initial matrix temperature (K)	PVBT
295	323	5.552
353	323	6.948
323	295	4.362
323	353	4.362

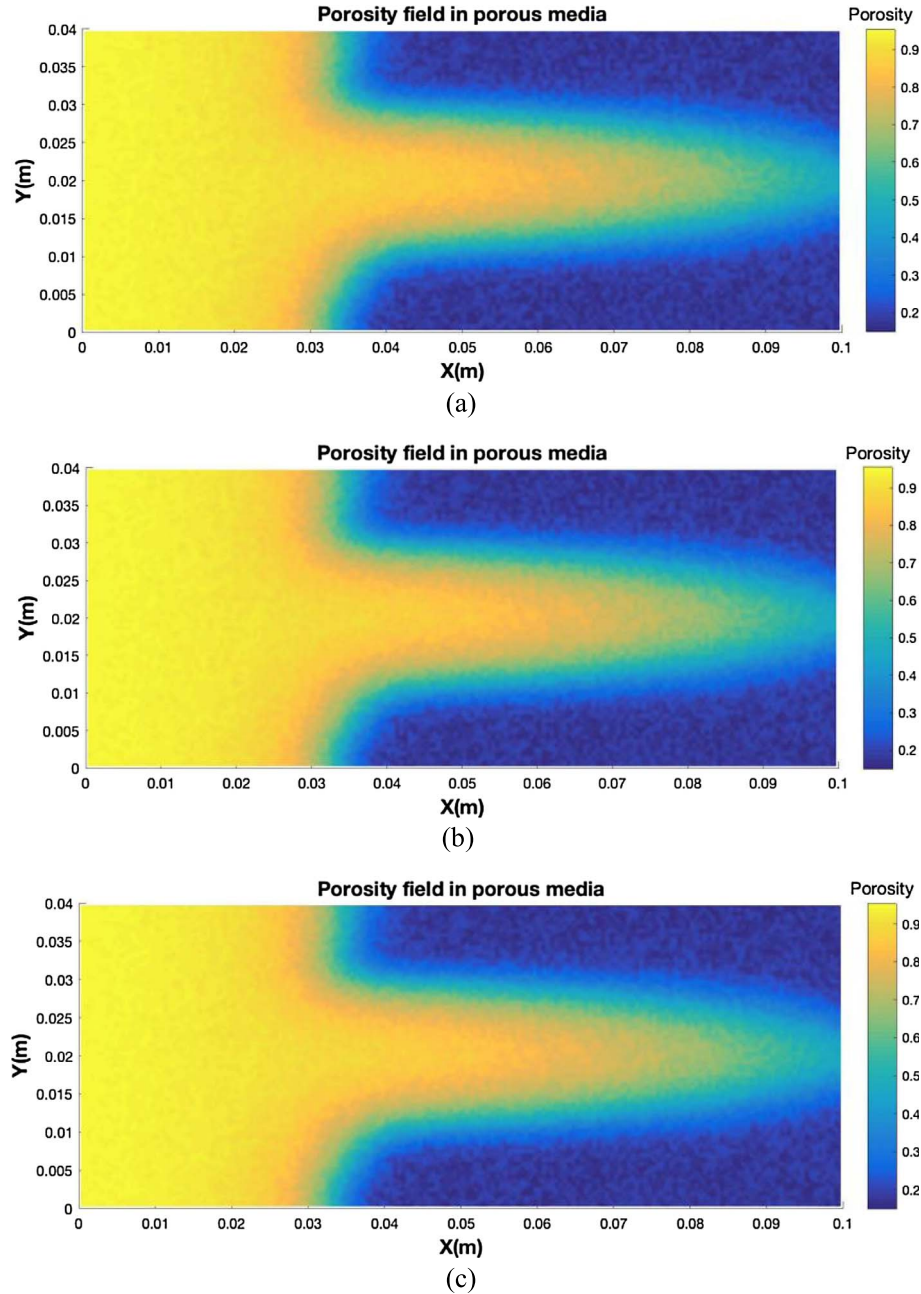


Fig. 16. Porosity profiles at breakthrough for three different initial matrix temperatures. The injected acid temperature is 323 K, and its injected velocity is 9.17×10^{-6} m/s. (a) 295 K. (b) 323 K (c) 353 K.

changed initial matrix temperatures will not change the values of PVBT. For three different initial matrix temperatures: 295 K, 323 K, and 353 K, the values of PVBT are the same as 4.362. However, when the injected acid temperature is changed, even though the initial matrix temperature is unchanged, the values of PVBT will be changed, which is evident from the first two rows in Table 6. From the two rows, it can be further seen that the two PVBT values are nearly the same as the corresponding values in the isothermal conditions, which also demonstrates that the injected acid temperature, instead of the initial matrix

temperature, has an effect on the PVBT value. Thus, the injected acid temperature governs the PVBT value and can be a design parameter in matrix acidization, which can also be concluded from [31]. The porosity profiles at breakthrough for three different initial matrix temperatures are given in Figure 16, where the same kind of porosity profiles can be seen clearly. Thus, Figure 16 demonstrates that the temperature of the initial matrix is not a key factor affecting matrix acidization once again.

To learn the reason why the injected acid temperature has such a significant effect on matrix acidization, the

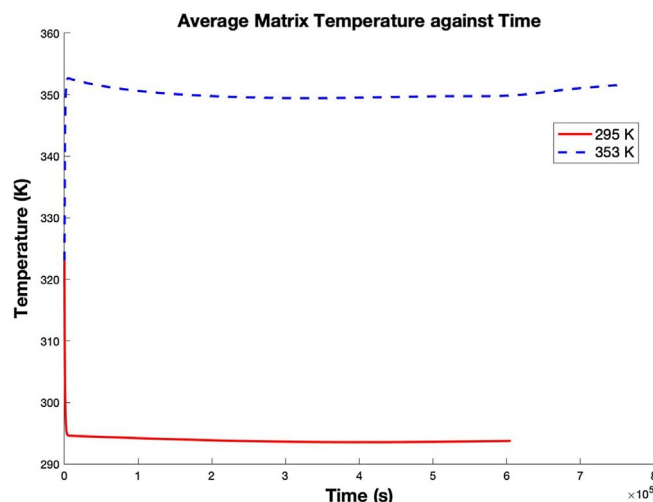


Fig. 17. Average matrix temperature against time. The solid curve represents the case where the temperature of the injected acid is 295 K, and the dashed curve represents the case where the temperature of the injected acid is 353 K. The initial matrix temperature is 323 K.

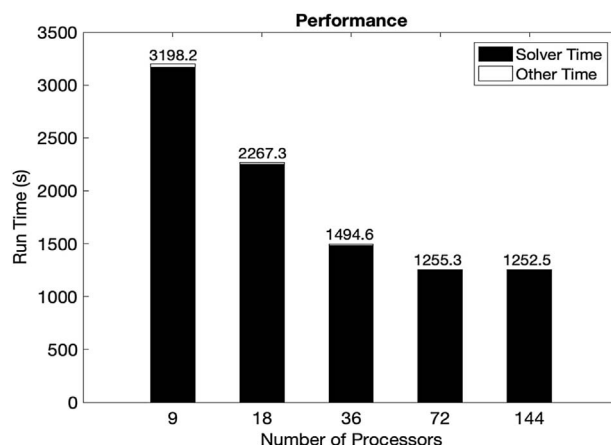


Fig. 18. Performance of the 3D parallel code. The numbers above the bars represent the run time, which is the sum of the solver time and the other time. It can be seen that for each bar, the percentage of solver time over run time >98%.

change in the average matrix temperature with time is investigated. The initial matrix temperature is set as 323 K, and the injected acid temperatures are 295 K and 353 K, respectively, which are also the cases represented by the first two rows in Figure 6. The history of the average matrix temperature from the beginning to breakthrough is shown in Figure 17. It can be seen in the figure that the matrix temperature becomes more or less the same as the injected acid temperature immediately after acidization begins and continues to be similar until breakthrough. This explains why the initial matrix temperature has almost no effect on matrix acidization.

After the discussion above, it is concluded that the thermal DBF framework can simulate reasonable numerical results, which are verified by the numerical and chemical experiments of other works. This indicates that the thermal

DBF framework can be an effective tool in the field of matrix acidization.

7 Performance evaluation

The performance of the 2D parallel code was evaluated in [19], and this work tries to evaluate the performance of the 3D parallel code. The test is performed on the 3D grid used above. Moreover, the number of iterations is set to 100 to save supercomputing resources. Meanwhile, since the sparsity pattern of the coefficient matrix of the linear system of the energy conservation equation is the same as that of the concentration balance equation, evaluating both of the linear systems is a redundancy. Therefore, only the linear system of the concentration balance equation is

evaluated, which means the performance of the 3D parallel code of the improved DBF framework is evaluated in this section. This evaluation result can foresee the performance of the 3D parallel code of the thermal DBF framework.

The experiment of the improved DBF framework in the verification section is performed again, and the performance results are shown in Figure 18. It can be seen in the figure that the solver time takes up most of the run time ($>98\%$), which is reasonable. Except for the solver code, the other parts of the code only perform simple operations such as filling arrays and communicating data among processors, which will not cost much time. Moreover, the run time decreases with the increase in the number of processors, which means a certain level speedup can be achieved. However, when the number of processors increases to 144, further speedup seems impossible. The similar scalability of the MUMPS solver can be seen in [47].

8 Conclusion and future work

Since Wu *et al.* [19] contributed the DBF framework to the field of matrix acidization, improvements to the framework have been ongoing. This work is such as endeavor and tries to correct a defect in the momentum conservation equation of the DBF framework and maintain the momentum conservation equation when the porosity is changed. Furthermore, by introducing a direct solver called MUMPS, the pseudo parameter e in the mass conservation equation can be deleted, which keeps the incompressible attribute of the acid flow in matrix acidization and thus makes the framework more reasonable. In addition, the simulation flowchart is also changed in this work, which is another correction to the DBF framework. After these revisions, the new framework can be called the improved DBF framework for short. The improved DBF framework is realized by 2D and 3D parallel codes with the help of MPI and FORTRAN 90 and verified by comparison with a series of previous works. It is emphasized that the 3D simulation results of the improved DBF framework are given for the first time in this work. The improved DBF framework can simulate similar numerical results with [17] and [45], which demonstrates its reliability.

The correctness of the improved DBF framework makes it feasible to develop a thermal DBF framework based on this verification. In addition to the mass conservation law and momentum conservation law, which are included in the improved DBF framework, the thermal DBF framework also considers the energy conservation law and thus introduces the energy balance equation to the improved DBF framework. Verification to the thermal DBF framework is done under isothermal conditions and nonisothermal conditions, and the numerical simulation results match the conclusions from other chemical and numerical experiments such as [31] and [32]. Therefore, the thermal DBF framework is reasonable and trustable.

Since the accuracy of matrix acidization simulation is highly dependent on the size of the grid, very fine grids are required for trustable results, which brings about the

need to develop parallel codes to finish simulations in reasonable time. However, parallelizing the improved DBF framework and thermal DBF framework is not an easy task due to their complex equation systems. With the help of MPI and FORTRAN 90 and the experimenting field approach, this work overcomes these difficulties and develops scalable parallel codes, which is another large contribution to the field of matrix acidization.

With the reliable improved DBF framework and thermal DBF framework, a series of numerical investigations on matrix acidization can be carried out in the future, and more reasonable results are expected.

Acknowledgments. For computer time, this research used the resources of the Supercomputing Laboratory at King Abdullah University of Science & Technology (KAUST) in Thuwal, Saudi Arabia.

References

- 1 Liu P., Yao J., Couples G.D., Ma J., Huang Z., Sun H. (2017) Modeling and simulation of wormhole formation during acidization of fractured carbonate rocks, *J. Pet. Sci. Eng.* **154**, 284–301.
- 2 Ma G., Yun C., Yan J., Huidong W. (2018) Modelling temperature-influenced acidizing process in fractured carbonate rocks, *Int. J. Rock Mech. Min. Sci.* **105**, 73–84.
- 3 Szymczak P., Kwiatkowski K., Jarosinski M., Kwiatkowski T., Osselin F. (2019) Wormhole formation during acidizing of calcite-cemented fractures in gas-bearing shales, *SPE J.* **24**, 795–810.
- 4 Zhao C., Hobbs B.E., Ord A., Hornby P., Peng S. (2008) Effect of reactive surface areas associated with different particle shapes on chemical-dissolution front instability in fluid-saturated porous rocks, *Transp. Porous Media* **73**, 75–94.
- 5 Zhao C., Hobbs B.E., Ord A., Peng S. (2010) Effects of mineral dissolution ratios on chemical-dissolution front instability in fluid-saturated porous media, *Transp. Porous Media* **82**, 317–335.
- 6 Zhao C., Hobbs B.E., Ord A. (2010) Theoretical analyses of the effects of solute dispersion on chemical-dissolution front instability in fluid-saturated porous media, *Transp. Porous Media* **84**, 629–653.
- 7 Buijse M.A. (2000) Understanding wormholing mechanisms can improve acid treatments in carbonate formations, *SPE Prod. Facil.* **15**, 168–175.
- 8 Hoefinger M.L., Fogler H.S. (1988) Pore evolution and channel formation during flow and reaction in porous media, *AIChE J.* **34**, 45–54.
- 9 Fredd C.N., Fogler H.S. (1998) Influence of transport and reaction on wormhole formation in carbonate porous media, *AIChE J.* **44**, 1933–1949.
- 10 Daccord G., Touboul E., Lenormand R. (1989) Carbonate acidizing: Toward a quantitative model of the wormholing phenomenon, *SPE Prod. Eng.* **4**, 63–68.
- 11 Daccord G., Lenormand R., Lietard O. (1993) Chemical dissolution of a porous medium by a reactive fluid – I. Model for the “wormholing” phenomenon, *Chem. Eng. Sci.* **48**, 1, 169–178.
- 12 Daccord G., Lietard O., Lenormand R. (1993) Chemical dissolution of a porous medium by a reactive fluid – II.

- Convection vs. reaction, behavior diagram. *Chem. Eng. Sci.* **48**, 1, 179–186.
- 13 Liu X., Ormond A., Bartko K., Ortoleva P. (1997) A geochemical reaction-transport simulator for matrix acidizing analysis and design, *J. Pet. Sci. Eng.* **17**, 181–197.
 - 14 Golfier F., Zarcone C., Bazin B., Lenormand R., Lasseux D., Quintard M. (2002) On the ability of a Darcy-scale model to capture wormhole formation during the dissolution of a porous medium, *J. Fluid. Mech.* **457**, 213–254.
 - 15 Panga M.K.R., Ziauddin M., Balakotaiah V. (2005) Two-scale continuum model for simulation of wormholes in carbonate acidization, *AIChE J.* **51**, 3231–3248.
 - 16 Maheshwari P., Balakotaiah V. (2013) Comparison of carbonate HCl acidizing experiments with 3D simulations, *SPE Prod. Oper.* **28**, 04, 402–413.
 - 17 Akanni O.O., Nasr-El-Din H.A., Gusain D. (2017) A computational Navier-Stokes fluid-dynamics-simulation study of wormhole propagation in carbonate-matrix acidizing and analysis of factors influencing the dissolution process, *SPE J.* **22**, 06, 2-049.
 - 18 De Oliveira T.J.L., De Melo A.R., Oliveira J.A.A., Pereira A.Z.I. (2012) Numerical simulation of the acidizing process and PVT extraction methodology including porosity/permeability and mineralogy heterogeneity, in: *SPE International Symposium and Exhibition on Formation Damage Control*, Society of Petroleum Engineers.
 - 19 Wu Y., Salama A., Sun S. (2015) Parallel simulation of wormhole propagation with the Darcy–Brinkman–Forchheimer framework, *Comput. Geotech.* **69**, 564–577.
 - 20 Wu Y. (2015) *Parallel reservoir simulations with sparse grid techniques and applications to wormhole propagation*. Diss.
 - 21 Kou J., Sun S., Wu Y. (2016) Mixed finite element-based fully conservative methods for simulating wormhole propagation, *Comput. Meth. Appl. Mech. Eng.* **298**, 279–302.
 - 22 Kou J., Sun S., Wu Y. (2019) A semi-analytic porosity evolution scheme for simulating wormhole propagation with the Darcy–Brinkman–Forchheimer model, *J. Comput. Appl. Math.* **348**, 401–420.
 - 23 Li X., Rui H. (2019) Superconvergence of a fully conservative finite difference method on non-uniform staggered grids for simulating wormhole propagation with the Darcy–Brinkman–Forchheimer framework, *J. Fluid Mech.* **872**, 438–471.
 - 24 Li X., Rui H., Chen S. (2019) A fully conservative block-centered finite difference method for simulating Darcy–Forchheimer compressible wormhole propagation, *Numer. Algorithms* **82**, 2, 451–478.
 - 25 Li X., Rui H. (2020) A fully conservative block-centered finite difference method for Darcy–Forchheimer incompressible miscible displacement problem, *Numer. Meth. Partial Differ. Equ.* **36**, 1, 66–85.
 - 26 Wang Y., Sun S., Gong L., Yu B. (2018) A globally mass-conservative method for dual-continuum gas reservoir simulation, *J. Nat. Gas Sci. Eng.* **53**, 301–316.
 - 27 Falgout R.D., Yang U.M. (2002) *hypre: A library of high performance preconditioners*, Springer, Berlin, Heidelberg.
 - 28 Amestoy P.R., Duff I.S., L'Excellent J.-Y., Koster J. (2001) A fully asynchronous multifrontal solver using distributed dynamic scheduling, *SIAM J. Mat. Anal. Appl.* **23**, 1, 15–41.
 - 29 Amestoy P.R., Guermouche A., L'Excellent J.-Y., Pralet S. (2006) Hybrid scheduling for the parallel solution of linear systems, *Parallel Comput.* **32**, 2, 136–156.
 - 30 Li Y., Liao Y., Zhao J., Peng Y., Pu X. (2017) Simulation and analysis of wormhole formation in carbonate rocks considering heat transmission process, *J. Nat. Gas Sci. Eng.* **42**, 120–132.
 - 31 Kalia N., Glasbergen G. (2010) Fluid temperature as a design parameter in carbonate matrix acidizing, in: *SPE Production and Operations Conference and Exhibition*, Society of Petroleum Engineers.
 - 32 Fredd C.N., Fogler H.S. (1999) Optimum conditions for wormhole formation in carbonate porous media: Influence of transport and reaction, *SPE J.* **4**, 03, 196–205.
 - 33 Wang Y., Sun S. (2016) Direct calculation of permeability by high-accurate finite difference and numerical integration methods, *Commun. Comput. Phys.* **20**, 2, 405–440.
 - 34 Wang Y. (2019) Reynolds stress model for viscoelastic drag-reducing flow induced by polymer solution, *Polymers* **11**, 10, 1659.
 - 35 Whitaker S. (1996) The Forchheimer equation: A theoretical development, *Transp. Porous Media* **25**, 1, 27–61.
 - 36 Aavatsmark I. (2002) An introduction to multipoint flux approximations for quadrilateral grids, *Comput. Geosci.* **6**, 3, 405–432.
 - 37 Šekutkovski B., Kostić I., Stefanović Z., Simonović A., Kostić O. (2015) A hybrid RANS-LES method with compressible k-omegaSSTAS turbulence model for high Reynolds number flow applications/Hibridna RANS-LES metoda s kompresibilnim k-omegaSSTAS turbulentnim modelom namjenjena analizi strujanja pri velikim Reynoldsovim brojevima, *Tehnicksi Vjesnik-Technical Gazette* **22**, 5, 1237–1246.
 - 38 Sun S., Salama A., El-Amin M.F. (2012) An equation-type approach for the numerical solution of the partial differential equations governing transport phenomena in porous media, in: *The International Conference on Computational Science, ICCS, June 4–6, Omaha, NE*.
 - 39 Salama A., Sun S., El Amin M.F. (2013) A multi-point flux approximation of the steady state heat conduction equation in anisotropic media, *ASME J. Heat Trans.* **135**, 1–6.
 - 40 Salama A., Li W., Sun S. (2013) Finite volume approximation of the three-dimensional flow equation in axisymmetric, heterogeneous porous media based on local analytical solution, *J. Hydrol.* **501**, 45–55.
 - 41 Salama A., Sun S., Wheeler M. (2014) Solving global problem by considering multitude of local problems: Application to flow in anisotropic porous media using the multipoint flux approximation, *J. Comput. Appl. Math.* **267**, 117–130.
 - 42 Salama A., Sun S., El Amin M.F. (2015) Investigation of thermal energy transport from an anisotropic central heating element to the adjacent channels: A multipoint flux approximation, *Ann. Nucl. Energy* **76**, 100–112.
 - 43 Davis T.A. (2004) Algorithm 832: UMFPACK V4. 3 – an unsymmetric-pattern multifrontal method, *ACM Trans. Math. Softw.* **30**, 196–199.
 - 44 Hadri B., Kortas S., Feki S., Khurram R., Newby G. (2015) Overview of the KAUST's Cray X40 system-Shaheen II, in: *Proceedings of the 2015 Cray User Group*.
 - 45 Ku H.C., Hirsh R.S., Taylor T.D. (1987) A pseudospectral method for solution of the three-dimensional incompressible Navier-Stokes equations, *J. Comput. Phys.* **70**, 2, 439–462.
 - 46 Kalia N., Balakotaiah V. (2009) Effect of medium heterogeneities on reactive dissolution of carbonates, *Chem. Eng. Sci.* **64**, 2, 376–390.
 - 47 Raju M.P. (2009) Parallel computation of finite element Navier-Stokes codes using MUMPS solver, *Int. J. Comput. Sci. Issu.* **4**, 2, 20–24.