Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows

Article *in* Computer Physics Communications · July 2019

DOI: 10.1016/j.cpc.2019.07.011

CITATION
1

READS
124

4 authors:

Haijian Yang
Hunan University
27 PUBLICATIONS **121** CITATIONS
SEE PROFILE

Shuyu Sun
King Abdullah University of Science and Technology
271 PUBLICATIONS **2,882** CITATIONS
SEE PROFILE

Yiteng Li
King Abdullah University of Science and Technology
11 PUBLICATIONS **24** CITATIONS
SEE PROFILE

Chao Yang
Peking University
77 PUBLICATIONS **768** CITATIONS
SEE PROFILE

Computer Physics Communications xxx (xxxx) xxx

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc

Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows

Haijian Yang [a], Shuyu Sun [b,*], Yiteng Li [b], Chao Yang [c]

[a] College of Mathematics and Econometrics, Hunan University, Changsha, Hunan 410082, PR China
[b] Physical Science and Engineering Division, King Abdullah University of Science and Technology, Thuwal, 23955-6900, Saudi Arabia
[c] School of Mathematical Sciences, Peking University, Beijing 100871, PR China

ARTICLE INFO

Article history:
Received 1 October 2018
Received in revised form 3 July 2019
Accepted 23 July 2019
Available online xxxx

Keywords:
Reservoir simulation
Fully implicit method
Complementarity problem
Parallel computing

ABSTRACT

The numerical simulation of multicomponent compressible flow in porous media is an important research topic in reservoir modeling. Traditional semi-implicit methods for such problems are usually conditionally stable, suffer from large splitting errors, and may accompany with violations of the boundedness requirement of the numerical solution. In this study we reformulate the original multicomponent equations into a nonlinear complementarity problem and discretize it using a fully implicit finite element method. We solve the resultant nonsmooth nonlinear system of equations arising at each time step by a parallel, scalable, and nonlinearly preconditioned semismooth Newton algorithm, which is able to preserve the boundedness of the solution and meanwhile treats the possibly imbalanced nonlinearity of the system. Some numerical results are presented to demonstrate the robustness and efficiency of the proposed algorithm on the Tianhe-2 supercomputer for both standard benchmarks as well as realistic problems in highly heterogeneous media.

1. Introduction

As an important model of reservoir simulation, compositional flow can be used to simulate the transport of subsurface fluids that is sensitive to compositional changes. Compared to the black oil model in which the hydrocarbon phases consist of two pseudo-components, the compositional model characterizes a reservoir fluid by a finite number of components [1,2]. As the most remarkable feature of compositional flow, fluid phase behavior is characterized by the real equation of state. A full compositional model can simulate the process involving considerable interphase component transfer, such as miscible flooding in oil reservoirs or CO_2 sequestration in saline aquifers. On the other hand, multicomponent single-phase flow, as the simplest compositional flow model, also has broad applications in various disciplines. One application is to simulate shale gas flow by coupling different mechanisms, such as Knudsen diffusion, slippage effect, adsorption and desorption, for accurate estimation of shale gas production. The compositional variety of produced gas complicates the processing technology [3,4], making it necessary to use the multicomponent flow model as well. Additionally, in groundwater hydrology, the multicomponent flow model can be used to investigate the transport of contaminants which are hazardous to groundwater systems. The increasing demand of shale gas, as well as the growing concern to groundwater contamination, motivates research on the simulation of multicomponent flow in porous media.

Popular approaches for the multicomponent compressible flow with n_c components are to manipulate the mathematical model into a parabolic pressure equation and the $n_c - 1$ nonlinear convection–diffusion equations for the mole fraction of the $n_c - 1$ components. This formulation of the problem is used when applying the classical IMplicit Pressure Explicit Concentration (IMPEC) or sequential time-stepping approaches [1,2,5–7], by which the pressure equation is firstly implicitly solved, and then the $n_c - 1$ transport equations are explicitly or implicitly solved. In this treatment, the conservation of mass holds for the $n_c - 1$ components which are computed from the associated transport equation if a mass-conservative spatial discretization is used. However, the mass conservation of the last n_c-th component might not be guaranteed, since its mole fraction is calculated by the volume constraint instead of solving its continuity equation. In addition, the numerical stability of these semi-implicit schemes is limited not only by the Courant–Friedrichs–Lewy (CFL) condition, but also more seriously by the splitting error from the decoupling fashion. This explicit and splitting transport update in the IMPEC schemes is only conditionally stable and may seriously degrade the overall simulation performance. Even worse, it often accompanies with the violation of the boundedness requirement

* Corresponding author.
E-mail addresses: haijianyang@hnu.edu.cn (H. Yang), shuyu.sun@kaust.edu.sa (S. Sun), yiteng.li@kaust.edu.sa (Y. Li), chao_yang@pku.edu.cn (C. Yang).

https://doi.org/10.1016/j.cpc.2019.07.011
0010-4655/

of the solution, especially for problems with strong nonlinearities from highly heterogeneous media. This feature is not desirable for extreme-scale simulations that require both high spatial resolution and long time integration. An effective approach to relax the time-step constraints is to simulate the multicomponent flow system rein a fully implicit fully coupled manner, which requires to solve all the equations simultaneously by using a nonlinear iterative method. In this study, we propose a new formulation for describing the continuity of the multicomponent flow in a porous medium, which naturally satisfies the mass conservation of all the n_c components. The presented formulation is based on the fully coupled approaches [8–13], which greatly relax the stability restriction of the traditional IMPEC or sequential time-stepping approaches and ensure that the pressure and all the mole fractions stay together throughout the computation. The method easily allows more physics to be added to the system without changing much of the algorithmic framework, and is especially friendly for simulating problems with increasingly complex physics, such as multiphase compositional or thermal processes.

A key to the success of a fully implicit time-stepping method is the design of a highly efficient nonlinear solver for solving the resultant nonlinear system at each fully implicit time step. Generally speaking, the nonlinear system is often solved in a monolithic way within the framework of inexact Newton (IN) methods [8, 14–16], in which the Newton's method is the basic nonlinear solver and a Krylov subspace iterative method is applied as the linear solver at each Newton step. The family of inexact Newton algorithms is easy to implement, general-purpose and has a rapid local convergence rate under certain conditions [15,17]. Therefore, a number of general-purpose Newton-based solvers have been designed for the numerical solution of nonlinear systems. However, a straightforward application of an inexact Newton method to the fully implicit simulation of multicomponent compressible flows often leads to nonphysical numerical oscillations, which can only be avoided by the substantial reduction of time step size. As a result, it can do great damage to the natural advantage of fully implicit methods, especially when solving problems with high nonlinearity for long time simulation at extreme scales. Recently, a new approach has been developed for preserving the physical boundedness of the solution by reformulating the original system of equations as a nonlinear complementarity problem (NCP) [18–21]. The framework can be rewritten into inequality constraints for the thermodynamic state as a nonsmooth and nonlinear systems by employing the complementary function, referred to as a C-function [22,23], which is solved by the family of semismooth Newton methods [23–26] and has been proven to be very efficient in broad applications [20,27, 28]. For reservoir simulation, although the semismooth Newton method applied to the NCP system is observed to have quadratic convergence for simple problems in porous media [29–31], it is reported that the highly heterogeneous media have negative effects on the convergence of Newton's iteration [12,21,32]. This situation becomes worse in this study that poor convergence and even divergence of the traditional nonlinear solver is observed due to the much complicated dynamics of the multicomponent compressible flows.

It has to be mentioned that such complexity results from a cubic equation of state, which governs the equilibrium distribution of fluid components among phases. However, in our previous work [13,21,26], either we assume full immiscibility or the thermodynamic equilibrium is simply embodied by the dissolved gas ratio and volatilized oil ratio if interphase mass transfer takes place between oil and gas phases. In contrast to the multicomponent flow models, fluid properties are often assumed either constant (e.g., oil viscosity and oil compressibility) or pressure-dependent (e.g. oil density and oil formation volume factor) in multiphase black-oil models [33]. In reality, most fluid properties are functions of its composition and pressure at the given temperature. The compositional change affects the phase behavior of reservoir fluid, which has a remarkable influence on the flow process involving considerable component transfer. Thus, the variation of fluid properties needs to be accurately characterized by introducing a cubic equation of state, such as Peng–Robinson equation of state and Soave–Redlich–Kwong equation of state. It is clear the fully implicit fully coupled formulation significantly increases the nonlinearity of the system and imposes a huge challenge on the nonlinear solver.

In this paper, we focus on designing nonlinearly preconditioned semismooth Newton (NPSN) algorithm to treat the nonlinearity of the system and at the same time preserve the boundedness of the numerical solution. Nonlinear preconditioning can be applied on the left or on the right of the nonlinear function [17,34,35]. The additive or multiplicative Schwarz preconditioned inexact Newton algorithm (ASPIN or MSPIN) [9,12, 36,37] is an example of the left preconditioning, which changes the function of the system to a more balanced system. On the other hand, the right nonlinear preconditioning can be applied by modifying the variables of the nonlinear function to enhance the quality of the initial guess [16,26,38,39]. The right preconditioning method is easier to implement than the left one, owing to the fact that the nonlinear function does not have to be replaced and more efficient and sophisticated linear solvers can be applied to the unchanged linear system. To the best of our knowledge, this is the first attempt to apply the nonlinear preconditioning technique in conjunction with the semismooth method for the multicomponent compressible flow in porous media. As we will show, the use of the right nonlinear preconditioner can help mitigate the lack of robustness observed in using the Newton algorithm for NCP formulation of the model problem. We numerically show that our new method is more robust and efficient for problems with highly heterogeneous media, and it also scales optimally with the number of processors on a supercomputer.

The paper is organized as follows. In Section 2, the complementarity formulation of multicomponent compressible flow in highly heterogeneous reservoir is presented, following which a fully implicit finite element scheme is provided for the temporal and spatial discretizations. In Section 3, we present a nonlinear positivity-preserving solver, including the semismooth Newton and the nonlinear preconditioning technique, for efficiently solving the resultant nonlinear system. We examine the efficiency and the parallel performance of the proposed method by carrying out numerical results on several test cases in Section 4. The paper is concluded in Section 5.

2. Mathematical model

The problem of interest in this study is the multicomponent compressible flow in a heterogeneous porous medium [1,2,5]. Suppose that the compressible flow consists of n_c components. Let Ω be the domain and T denotes the final simulated time. The mass conservation equations for the ith component through the porous medium can be expressed as

$$\begin{cases} \dfrac{\partial(\phi c_i)}{\partial t} + \nabla\cdot(c_i\mathbf{u}) = q_i, \ \text{ in } \Omega_T = \Omega\times(0,T) \\ c_i = z_i c, \ i = 1,2,\ldots,n_c, \end{cases} \tag{1}$$

where ϕ is the porosity of the medium, c is the overall molar density, c_i and z_i are the molar density and the mole fraction of component i respectively, and q_i is the external mass flow rate. The velocity $\mathbf{u}$ is described by the Darcy's law

$$\mathbf{u} = -\frac{\mathrm{K}}{\mu}(\nabla p - \rho\mathbf{g}), \tag{2}$$

Please cite this article as: H. Yang, S. Sun, Y. Li et al., Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.07.011.

where K is the absolute permeability tensor, p is the pressure, μ is the viscosity, ρ is the mass density, and $\mathbf{g}$ is the body force. Additionally, the mole fractions satisfy the following constraints

$$\begin{cases} \sum_{i=1}^{n_c} z_i = 1, \\ z_i \geq 0. \end{cases} \tag{3}$$

In this paper, the Peng–Robinson (PR) equation of state (EOS) is used to describe the phase molar density as a function of the composition, temperature and pressure:

$$\begin{cases} \rho = cM, \ c = \dfrac{p}{ZRT}, \\ Z^3 - (1-B)Z^2 + (A - 3B^2 - 2B)Z - (AB - B^2 - B^3) = 0, \end{cases} \tag{4}$$

where M is the molecular weight, T the temperature, R the gas constant, A and B are the PR-EOS parameters [2,5]. The boundary conditions for the coupled equations are assumed to be impervious, i.e., the normal component of the velocity vanishes $\mathbf{u}\cdot\mathbf{n} = 0$. The injection and production wells are respectively modeled as the inflow and the outflow boundaries [1].

For the simulation of multicomponent compressible flows, we use the so-called fully coupled approach [8,10,26], i.e., the system (1) is coupled with Eqs. (2), (3) and (4) into a monolithic model. For the solution of the coupled system, there are many ways to choose a set of primary variables, depending on the problem formulation and applications [40]. In this study, our primary variables are the pressure p of the flow and the mole fractions z_i of the $n_c - 1$ components. To be more precise, we use the mass conservation equation for the last component $i = n_c$, i.e., the equation

$$\frac{\partial \left(\phi c_{n_c}\right)}{\partial t} + \nabla \cdot \left(c_{n_c}\mathbf{u}\right) = q_{n_c},$$

as the formulation for the pressure variable. And the other $n_c - 1$ mass conservation equations in (1), i.e., $i = 1, 2, \ldots, n_c - 1$, are chosen as the formulations for the rest of mole fraction variables z_i. The value of the last mole fractions z_{n_c} is computed by the relation $z_{n_c} = 1 - \sum_{i=1}^{n_c-1} z_i$ in (3). The compressibility factor Z as an intermediate variable is obtained by solving the algebraic cubic equation (4) with the primary variable p. The molar density c_i $(i = 1, 2, \ldots, n_c)$, which are also regarded as the intermediate variables, are computed by using the relations in (1) and (4). All other quantities depend upon these unknowns and/or on the independent variables, time and position.

The preservation of physical bounds is one of the key requirements for the mathematical model (1), which will also be referred to as monotonicity for simplicity. In the context of reservoir simulation, the loss of boundedness can lead to negative solution or non-physical oscillations. By combining (3) with the boundedness of the mole fraction of component z_i, one can obtain a variational inequality problem for the multicomponent compressible flow problem by introducing the L^2-inner product $\langle \cdot, \cdot \rangle$ as follows:

Problem 1. Given $z_i(\cdot, 0) = z_i^0 \in H^1(\Omega)$ with $z_i^0 \geq 0$, find $z_i \in H^1(\Omega_T)$ such that $z_i \geq 0$ a.e. in $\Omega_T = \Omega \times [0, T]$ and

$$\left\langle \frac{\partial (\phi c_i)}{\partial t}, \mathcal{X}_i - z_i \right\rangle + \left\langle \nabla \cdot (c_i \mathbf{u}), \mathcal{X}_i - z_i \right\rangle - \left\langle q_i, \mathcal{X}_i - z_i \right\rangle \geq 0, \quad i = 1, 2, \ldots, n_c, \tag{5}$$

which has to hold for almost all t and all $\mathcal{X}_i \in H^1(\Omega)$ with $\mathcal{X}_i \geq 0$.

With the help of the Lagrange multiplier θ_i for the inequality constraint $z_i \geq 0$, we reformulate the inequality (5) into an equivalent form:

Problem 2. A function $z_i \in L^2\left(0, T; H^2(\Omega)\right) \cap H^1(\Omega_T)$, $i = 1, 2, \ldots, n_c$, solves the inequality (5) if there exists $\theta_i \in L^2(\Omega_T)$ such that

$$\begin{cases} \dfrac{\partial (\phi c_i)}{\partial t} + \nabla \cdot (c_i \mathbf{u}) - q_i - \theta_i = 0, \\ \theta_i \geq 0, \ z_i \geq 0, \\ \theta_i (z_i - 0) = 0, \end{cases} \tag{6}$$

where all equalities and inequalities have to hold almost everywhere.

Proof. Let η_i be a function in $L^2\left(0, T; H^2(\Omega)\right) \cap H^1(\Omega_T)$ with the restriction $\eta_i \geq 0$. Multiplying the first equation of (6) by $\eta_i - z_i$, we have

$$\int_{\Omega_T} \left(\frac{\partial (\phi c_i)}{\partial t} + \nabla \cdot (c_i \mathbf{u}) - q_i \right) (\eta_i - z_i) - \int_{\Omega_T} \theta_i (\eta_i - z_i) = 0. \tag{7}$$

Using the properties of $\eta_i \geq 0$ and the last two equations of (6), then we obtain that

$$\theta_i (z_i - \eta_i) \leq \theta_i (z_i - 0) = 0.$$

That is to say,

$$\theta_i (\eta_i - z_i) \geq 0.$$

It follows from (7) that

$$\int_{\Omega_T} \left(\frac{\partial (\phi c_i)}{\partial t} + \nabla \cdot (c_i \mathbf{u}) - q_i \right) (\eta_i - z_i) \geq 0.$$

Hence, the inequality (5) holds after the localization and the proof is completed. □

Based on the above reformulations, we obtain the following complementarity conditions for any $i \in \{1, 2, \ldots, n_c\}$ by eliminating the Lagrange multiplier θ_i in (6),

$$\begin{cases} z_i \geq 0, \ \dfrac{\partial (\phi c_i)}{\partial t} + \nabla \cdot (c_i \mathbf{u}) - q_i \geq 0, \\ (z_i - 0) \left(\dfrac{\partial (\phi c_i)}{\partial t} + \nabla \cdot (c_i \mathbf{u}) - q_i \right) = 0. \end{cases} \tag{8}$$

In fact, if the nonnegativity condition of the mole fraction $z_i \geq 0$ is replaced by $z_i \geq -\infty$, then the complementarity problem (8) is reduced to the original model problem (1). On the other hand, the complementarity problem can be seen as a type of mathematical optimization problem. It is the problem of optimizing (minimizing or maximizing) a functional subject to certain requirements (constraints); see related works [18–21,23] and references therein for more details.

We use the Raviart–Thomas mixed finite element method (MFEM) for the treatment of bulk fluid flow and use the upwind discontinuous Galerkin (DG) method for the treatment of the convection. We note that the lowest order Raviart–Thomas MFEM in a rectangular mesh can be implemented with a cell-center finite difference scheme by using a diagonal lumping approach. Furthermore, the lowest order upwind discontinuous Galerkin method can be viewed as the upwind finite volume method. In this study, the combined approach of Raviart–Thomas MFEM and upwind DG method is applied to discretize the model equations spatially, and the implicit Backward Euler scheme is used for the temporal integration. We partition the computational domain Ω into a finite number of cell elements. For any domain $D \subset \Omega$, we define $(\varphi, \psi)_D = \int \varphi\psi \, \mathrm{d}\mathbf{x}$ for any scalar function φ and ψ, and $(\boldsymbol{\varphi}, \boldsymbol{\psi})_D = \int \boldsymbol{\varphi}\cdot\boldsymbol{\psi} \, \mathrm{d}\mathbf{x}$ for any vector function $\boldsymbol{\varphi}$ and $\boldsymbol{\psi}$. Let E denote one of the cells in our partitioning $\mathcal{T}_h$, and ∂E denote the faces (or edges for two dimensional domains) of the cell E. The collection

Please cite this article as: H. Yang, S. Sun, Y. Li et al., Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.07.011.

of all internal faces is denoted by Γ_h. The vector space $\boldsymbol{V}_h$ and the scalar space W_h are defined as follows

$$\boldsymbol{V}_h = \{\boldsymbol{v} \in H(\mathrm{div}, \Omega) : \boldsymbol{v}|_E \in \mathrm{RT}_\kappa(E)\},$$
$$W_h = \left\{w \in L^2(\Omega) : w|_E \in \mathrm{Q}_\kappa(E)\right\},$$

where RT_κ and Q_κ represent the Raviart–Thomas space and the piecewise polynomial space of order κ respectively. We assume the velocity space vanishes on the boundaries of the domain, denoted by $\boldsymbol{V}_h^0 = \{\mathbf{u} \in \boldsymbol{V}_h : \mathbf{u} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega\}$. We consider only no-flow boundary condition in the formulation, but we note that the mass inflow and outflow can be implemented through the source and sink terms in a discrete model. Our spatially discretized algorithm reads: Find $p_h(\cdot, t) \in W_h$, $c_{i,h}(\cdot, t) \in W_h$, and $\mathbf{u}_h(\cdot, t) \in \boldsymbol{V}_h^0$ such that

$$\left(\frac{\partial}{\partial t}\left(\phi c_{i,h}\right), w\right)_\Omega - \sum_{E\in\mathcal{T}_h}\left(c_{i,h}\mathbf{u}_h, \nabla w\right)_E + \sum_{\gamma\in\Gamma_h}\left\langle c_{i,h}^*\mathbf{u}_h\cdot\mathbf{n}_\gamma, [w]\right\rangle_\gamma = (q_i, w)_\Omega, \ \forall w \in W_h, \tag{9}$$

$$\left(\mu\mathbf{K}^{-1}\mathbf{u}_h, \boldsymbol{v}\right)_\Omega - (p_h, \nabla\cdot\boldsymbol{v})_\Omega = (\rho\mathbf{g}, \boldsymbol{v})_\Omega, \ \forall \boldsymbol{v} \in \boldsymbol{V}_h^0, \tag{10}$$

where c_i^* is the molar density of component i in the upstream cell. It should be mentioned that the aforementioned work can be extended to any-order space. However, we only consider the lowest-order approximation with $\kappa = 0$ in the numerical examples of this paper. In that case, Q_0 is a piecewise constant space, leading to the removal of the second term $\sum_E (c_i\mathbf{u}, \nabla w)_{E\in\mathcal{T}_h}$ in (9).

The weak formulation of the transport equation over the entire domain (9) can be viewed as the summation of the local transport equation on each cell. The weak form of the transport equation on cell E can be derived by multiplying the test function w and integrating by parts

$$\int_E \frac{\partial\left(\phi_E c_{i,E}\right)}{\partial t}\omega_E - \int_E c_{i,E}\mathbf{u}\cdot\nabla\omega_E + \int_{\partial E} c_{i,E}^* w_E \mathbf{u}\cdot\mathbf{n} = \int_E q_{i,E}w_E,$$

with

$$c_{i,E}^* = \begin{cases} c_{i,E}, & \text{if } \mathbf{u}\cdot\mathbf{n} \geq 0, \\ c_{i,E'}, & \text{if } \mathbf{u}\cdot\mathbf{n} < 0, \end{cases}$$

where E' is the cell that shares the common edge or face with cell E. The velocity unknowns in the above system (9) and (10) can be eliminated using the trapezoidal quadrature rule. With the relation $c_i = cz_i$ and the summation constraint $\sum_i z_i = 1$, the primary unknowns of the system can be reduced to $(p, z_1, \ldots, z_{n_c-1})$.

For clarity of presentation, the above weak formulation (9)–(10) is stated without imposing the variational inequality constraints. We can, of course, combine (3) with the boundedness of the mole fraction of component z_i, and obtain a variational inequality-imposed spatial discretization of the multicomponent compressible flow problem, in a similar manner as we obtain (5), (6), and (8) previously.

3. Nonlinear boundedness-preserving solver

Let $\mathcal{I} = \{1, 2, \ldots, n\}$ be an index set with each index corresponding to an unknown component X_i and a nonlinear residual component F_i by applying the fully implicit scheme introduced in Section 2. Then the discretization of the complementarity formulation for the model problem is defined as follows. Find $X \in \mathbb{R}^n$ such that,

$$\begin{cases} X \geq 0, \\ F(x) \geq 0, \\ X^T F(X) = 0. \end{cases} \tag{11}$$

The nonlinear system (11) is solved by a nonlinear boundedness-preserving solver, which is based on the framework of semismooth Newton-type approaches, with the solution of the previous time step as the initial guess. Since the convergence behavior of the semismooth Newton method is problem-dependent, we present a class of nonlinear preconditioning techniques to further improve the convergence and the robustness of the semismooth Newton method.

3.1. Inexact Newton method

We briefly review the classical inexact Newton (IN) algorithm [8,14–16], which serves as the basic building-block of both the semismooth Newton method and its nonlinearly preconditioned version. For the discrete nonlinear system of equations

$$F(X) = 0, \tag{12}$$

where $F : \mathbb{R}^n \to \mathbb{R}^n$ is a given nonlinear vector-valued function arising from the residuals function in (11) given by $F = (F_1, F_2, \ldots, F_n)^T$ with $F_i = F_i(X_1, X_2, \ldots, X_n)^T$, and $X = (X_1, X_2, \ldots, X_n)^T$. The framework of the inexact Newton method is described in Algorithm 1.

Algorithm 1 Inexact Newton (IN) algorithm.

Choose an initial guess $X^0 \in \mathbb{R}^n$ and set $k = 0$.

1: **for** $k = 0, 1, 2, 3, \cdots$ until convergence **do**
2: Construct the Jacobian matrix $\nabla F(X^k)$.
3: Compute the inexact Newton direction d^k such that

$$\left\|\nabla F(X^k)d^k + F(X^k)\right\| \leq \max\left\{\eta_r\|F(X^k)\|, \eta_a\right\}. \tag{13}$$

4: Determine a step size $\lambda^k \in (0, 1]$ using a backtracking linesearch technique based on the merit function $\Psi(X) = \frac{1}{2}\|F(X)\|^2$.
5: Compute a new approximate solution $X^{k+1} = X^k + \lambda^k d^k$.
6: **end for**

In the inexact Newton algorithm, the computation of the Jacobian matrix is based on the assumption that the nonlinear residual function $F(X)$ is everywhere differentiable, and therefore it is simply an algebraic derivative of this nonlinear function. The Jacobian system (13) is solved inexactly by using a Krylov subspace method with a linear preconditioner [13,41,42]. The parameters $\eta_r \in [0, 1)$ and $\eta_a \in [0, 1)$ in (13) are, respectively, the relative tolerance and absolute tolerance, which are used to determine how accurately the Jacobian linear system needs to be solved; if these parameters are sufficiently small, the method is reduced to the exact Newton method. The Newton iteration remains valid until the following convergence criterion is satisfied

$$\|F(X^{k+1})\| \leq \max\{\varepsilon_r\|F(X^0)\|, \varepsilon_a\},$$

where ε_r (ε_a) is the relative (absolute) solver tolerance for the Newton iteration, respectively.

A major disadvantage of inexact Newton methods is that there is no guarantee the predicted solution stays within the physical range. To address this issue, we propose a family of semismooth Newton (SN) methods [23–26].

3.2. Semismooth Newton method

The semismooth Newton algorithm is based on a reformulation of (11) as a nonlinear and semismooth system of equations satisfying a semismooth property. A very popular approach to solve (11) is to transform it into a nonsmooth nonlinear equation

Please cite this article as: H. Yang, S. Sun, Y. Li et al., Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.07.011.

via the Fischer–Burmeister complementarity function, which is defined as

$$\varphi(a, b) = a + b - \sqrt{a^2 + b^2}.$$

Accordingly, solving the system (11) is equivalent to solving the following nonlinear system of equations

$$\mathcal{F}(X) = 0, \tag{14}$$

in which $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$ is componentwisely given by

$$\mathcal{F}_i(X) = \varphi\left(X_i - 0, F_i(X)\right), \ i = 1, 2 \ldots, n.$$

In the above nonsmooth system, $\mathcal{F}(X)$ usually becomes nondifferentiable at the points $(X_i, F_i(X)) = (0, 0)$. Hence, we construct a generalized Jacobian matrix using the concept of B-subdifferentiability [22,23]. let X^k be the current approximation at the kth Newton iteration, then the generalized Jacobian matrix at X^k has the following form

$$\nabla\mathcal{F}(X^k) = D_a(X^k) + D_b(X^k)\nabla F(X^k),$$

where $\nabla F(X^k)$ is the Jacobian matrix for the smooth system (12), the two nonnegative diagonal matrices $D_a(X^k)$ and $D_b(X^k)$ are defined as

$$D_a(X^k) = \mathrm{diag}(d_{a_1}, \ldots, d_{a_n}), \ D_b(X^k) = \mathrm{diag}(d_{b_1}, \ldots, d_{b_n}), \tag{15}$$

with

$$\begin{cases} (d_{a_i}, d_{b_i}) = \left(1 - \dfrac{a_i}{\sqrt{a_i^2 + b_i^2}}, 1 - \dfrac{b_i}{\sqrt{a_i^2 + b_i^2}}\right) \\ \quad \text{if } a_i^2 + b_i^2 \neq 0, \\ (d_{a_i}, d_{b_i}) \in \left\{(1 - \zeta_1, 1 - \zeta_2) \mid \zeta_1^2 + \zeta_2^2 \leq 1\right\} \quad \text{otherwise.} \end{cases}$$

Here, $a = X^k - 0$, $b = F(X^k)$, ζ_1 and ζ_2 are two positive constants. In practice, the positive constants are fixed to $\zeta_1 = 0$ and $\zeta_2 = 0$. If a Newton-type method with a generalized Jacobian matrix is applied to the resultant nonsmooth system of equations, it reduces to the family of semismooth Newton methods, as shown in Algorithm 2.

Algorithm 2 Semismooth Newton (SN) algorithm.

Choose an initial guess $X^0 \in \mathbb{R}^n$ and set $k = 0$.

1: **for** $k = 0, 1, 2, 3, \cdots$ until convergence **do**
2: Construct a generalized Jacobian matrix $\nabla\mathcal{F}(X^k)$.
3: Compute the inexact Newton direction d^k such that

$$\left\|\nabla\mathcal{F}(X^k)d^k + \mathcal{F}(X^k)\right\| \leq \max\left\{\eta_r\|\mathcal{F}(X^k)\|, \eta_a\right\}.$$

4: Determine a step size $\lambda^k \in (0, 1]$ using a backtracking linesearch technique based on the merit function $\Psi(X) = \frac{1}{2}\|\mathcal{F}(X)\|^2$.
5: Compute the new approximate solution $X^{k+1} = X^k + \lambda^k d^k$.
6: **end for**

The semismooth Newton algorithm has several well-known features: (a) Fast convergence. If the initial guess is close enough to the desired solution, then the convergence is very fast (quadratic). The fast convergence, or the so-called locally superlinear convergence, happens only if a good initial guess is available. Generally it is difficult to obtain such an initial guess for solving large sparse nonlinear systems of equations arising from discretization of variational inequality problem. (b) Nonrobustness. The damping factor λ^k is used to determine the step length that one should go in the selected search direction. When the nonlinearities in the system are well-balanced, a near quadratic convergence may be observed, even in the ideal case a full step is taken, i.e., $\lambda^k = 1$. However, when the resultant system is severely nonlinear and non-smooth, the size of λ^k could be extremely small, and in such a situation, the Newton direction becomes only a weak descent direction, thus it converges slowly or simply stagnates. Hence, in this work, we focus on the development of a nonlinear preconditioning technique to enhance the quality of the initial guess, and thereby improve the robustness of the outer semismooth Newton method.

Remark 1. In Algorithms 1 and 2, the step size λ^k is determined by the backtracking linesearch method [21,24,43] as follows: Compute $\bar{l}$ by the standard cubic line search algorithm [14,15] as the smallest l in $\{0, 1, 2, \ldots\}$ such that

$$\Psi\left(X^k + \gamma^{\bar{l}} d^k\right) \leq \Psi(X^k) + \sigma\gamma^{\bar{l}}\left(\nabla\Psi(X^k)\right)^T d^k,$$

where the parameter σ is fixed to $= 10^{-4}$. And then set $\lambda^k = \gamma^{\bar{l}}$.

3.3. Nonlinear preconditioning

To begin with, let us recall that linear preconditioning technique in conjunction with Krylov iterative methods for the solution of linear systems of equations $AX = b$. We can write the use of a Krylov method as $\mathbf{K}(A, X)$, where $A \in \mathbb{R}^{n\times n}$ is the matrix, $X \in \mathbb{R}^n$ is the initial solution, and $\mathbf{K}$ is the Krylov solver. Then the linear preconditioning takes the following two steps to recast the original problem,

$$AM^{-1}Y = b, \ Y = MX,$$

where one uses the linear preconditioner M^{-1} to transform the solution of the original linear problem X and then solves for the preconditioned solution $Y \in \mathbb{R}^n$ with the application of the Krylov method $\mathbf{K}$.

In the following, we take the basic notations of linear preconditioners from the linear system and apply them to the nonlinear case. Let $\mathcal{F}(X) = 0$ be a nonlinear system and $X^k \in \mathbb{R}^n$ be the numerical solution at the kth step, and suppose that $X^{k+1} = \mathbf{M}\left(\mathcal{F}(X^k), X^k\right)$ is the use of a nonlinear stationary solver $\mathbf{M}$ to obtain an approximate solution. Then the nonlinear preconditioner takes the following two steps to rebuild the original problem $\mathcal{F}(X) = 0$ in a similar fashion: (a) Preconditioning the nonlinear system with the solver $\mathbf{N}$ by

$$Y^k = \mathbf{N}\left(\mathcal{G}(X^k), X^k\right),$$

where $\mathcal{G}(X) = 0$ is a preconditioning system for the nonlinear preconditioning; (b) Solving the nonlinear system with the solver $\mathbf{M}$ by

$$X^{k+1} = \mathbf{M}\left(\mathcal{F}(Y^k), Y^k\right),$$

where $\mathcal{F}(Y)$ is a nonlinear function for the global update step provided that $Y \in \mathbb{R}^n$ is available. Under certain circumstances the recast system will have better conditioning and less severe nonlinearities than the original system [9,12,36]. In summary, the nonlinearly preconditioned semismooth Newton (NPSN) algorithm for (11) is described in Algorithm 3.

Remark 2. In Algorithm 3, the construction of the preconditioning system $\mathcal{G}(X) = 0$ is often problem-dependent. It is expected to be relatively cheap to implement and meanwhile effective to improve the performance of global Newton iterations. In this study, for the simulation of the multicomponent compressible flow, we suppose that a function $F(X)$ in (12) to build the preconditioning system $\mathcal{G}(X)$, i.e. $\mathcal{G}(X) = F(X)$. In this case, the system $F(X) = 0$ is a nonlinear and smooth system, which is solved by the inexact Newton algorithm as introduced in Algorithm 1.

Please cite this article as: H. Yang, S. Sun, Y. Li et al., Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.07.011.

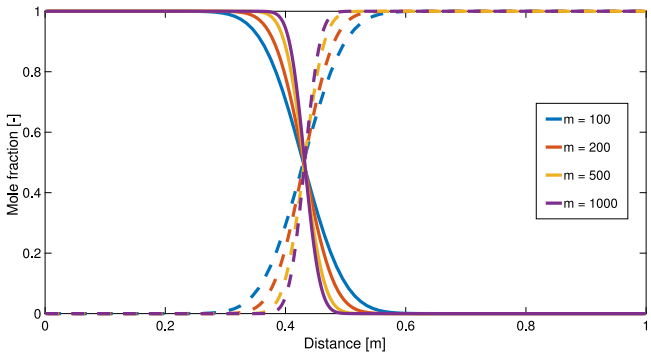


Fig. 1. Mole fraction profiles of C_1 (solid line) and C_3 (dash line) on 100, 200, 500 and 1000 uniform grids, respectively.

Algorithm 3 Nonlinearly Preconditioned Semismooth Newton (NPSN) algorithm.

Choose an initial guess $X^0 \in \mathbb{R}^n$ and set $k = 0$.

1: **for** $k = 0, 1, 2, 3, \cdots$ until convergence **do**

2: Nonlinearly preconditioning step:

a. Construct the preconditioning system based on the function $\mathcal{G}(X)$.
b. Evaluate $Y^k = \mathbf{N}\left(\mathcal{G}(X^k), X^k\right)$ with a nonlinear iterative method **N**.
c. Update $X^k = Y^k$.

3: Semismooth Newton step:

a. Construct the global nonlinear system based on the function $\mathcal{F}(X)$.
b. Evaluate $\mathbf{M}\left(\mathcal{F}(Y^k), Y^k\right)$ with the semismooth Newton method **M**.
c. Update $X^{k+1} = \mathbf{M}\left(\mathcal{F}(Y^k), Y^k\right)$.

4: **end for**

Remark 3. The nonlinear preconditioning step does not have to be applied at every outer Newton iteration. Hence, in practice the nonlinear preconditioner is activated only when the control parameter N_{switch} is triggered, when, for example, the global Semismooth Newton iteration is difficult to converge. This means that we first take several Semismooth Newton iterations as shown in Algorithm 2, then solve the preconditioning nonlinear system by the inexact Newton method, and continue this process until convergence. In particular, if the backtracking linesearch step size λ^k cannot be obtained in global Newton iterations, we directly switch to the nonlinear preconditioning step.

4. Numerical experiments

In the numerical experiments, we focus on (a) the verification of the numerical accuracy of the fully implicit solver; (b) a comparison of the performance of the classical semismooth Newton method and its nonlinearly preconditioned version; (c) the parallel performance of the proposed method.

4.1. Numerical validation with 1D tests

In this subsection, the proposed fully implicit model is validated by a number of one-dimensional examples. The length of the computational domain is 1 m with constant porosity and permeability. Initially, there exists pure propane (C_3) in the domain. We inject pure methane (C_1) from the left end and produce the fluid mixture at the right end. The molar density of the injected fluid is computed using the Peng–Robinson equation of state. It is worth noting that the temperature is high enough so that both displacing and displaced fluids exist in the vapor phase. Furthermore, gravity is ignored in the tests. The timestep size $\Delta t = 0.05$ s and the total number of time steps is 2000. Table 1 presents some physical parameters and relevant data.

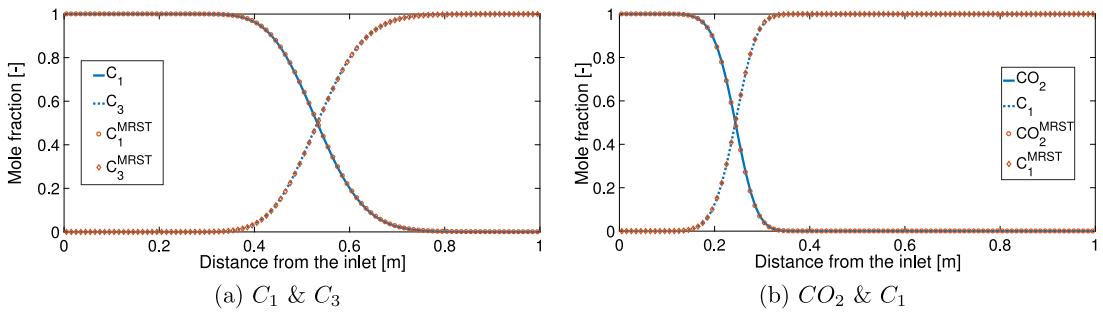
Table 1
Relevant data for Case-1.

Parameters	Value
Domain dimensions [m]	1
Temperature [K]	554.8
Permeability [mD]	100
Porosity [-]	0.2
Mole fraction of initial gas [-]	0.0 C_1, 1.0 C_3
Mole fraction of injected gas [-]	1.0 C_1, 0.0 C_3
Initial pressure [Bar]	1.0
Injection rate [m/s]	0.002

We first analyze the grid-size sensitivity of the multicomponent single-phase flow at four different grid numbers: m = 100, m = 200, m = 500 and m = 1000. For simplicity, it is assumed the viscosity of the fluid mixture is constant with $\mu = 0.015$ cp. Although viscosity is a function of pressure, temperature and mole composition, the insignificant variation of the viscosity makes little difference in the compositional flow at the given condition. We note that our simulation result matches the result with variable viscosity. Fig. 1 demonstrates the distribution of C_1 and C_3 at $t = 100$ s. As the grid is refined, the numerical dispersion gradually vanishes. It is observed solutions of higher mesh resolution ($m = 500$ and $m = 100$) have little difference, implying the solution of the finer mesh matches the real solution.

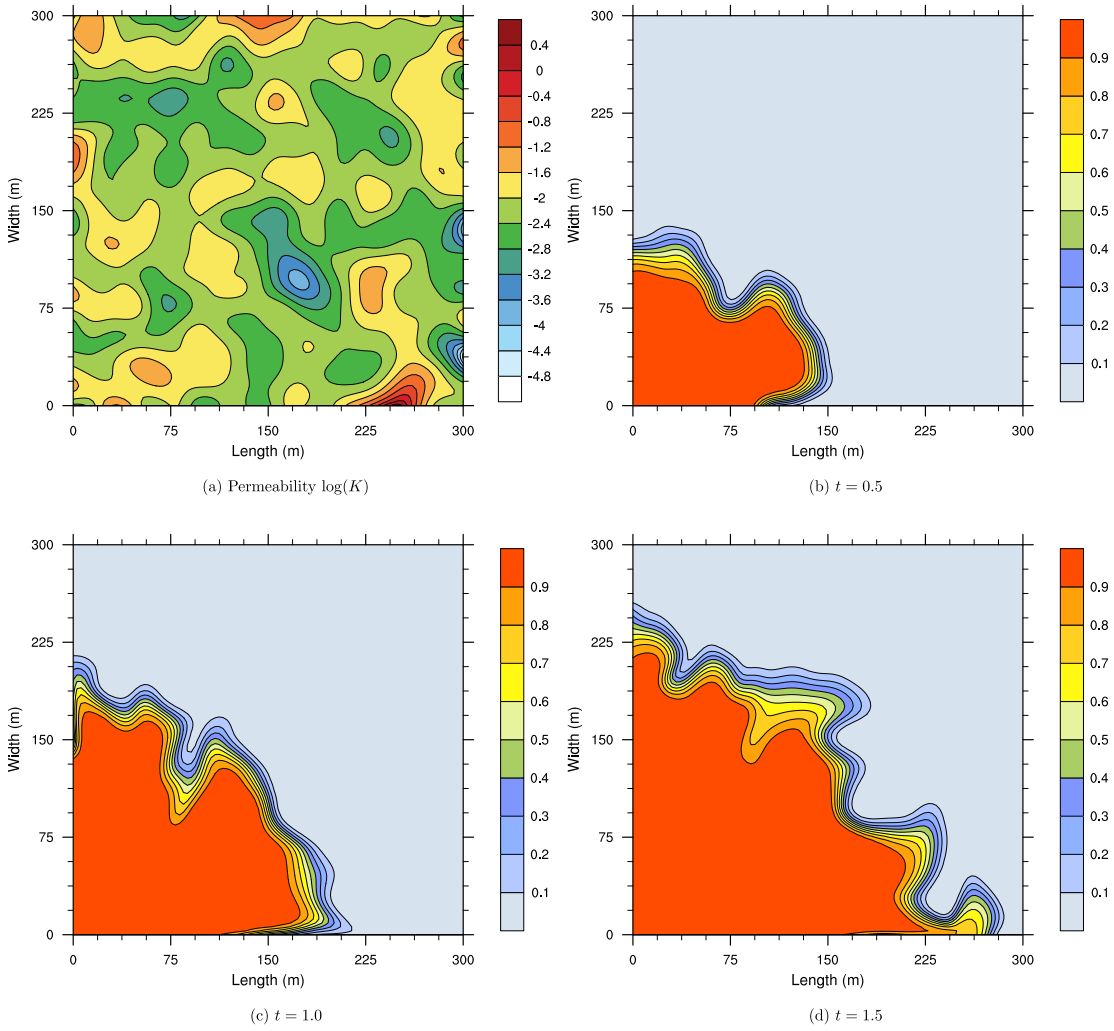
In order to verify the validity of the fully implicit approach further, our simulation results are compared with those computed by MRST (MATLAB Reservoir Simulation Toolbox) [44], which provides several one-dimensional benchmarks for the numerical simulation of multicomponent single-phase flow. In contrast to the mole-based formulation in this work, MRST adopts the mass-based formulation [45]. The viscosity of the fluid mixture is estimated by the Lohrenz, Bray and Clark correlation [46]. All the other parameters are same as those in the grid sensitivity analysis. As seen in left of Fig. 2, our simulation result, shown as solid and dot lines, matches perfectly with the simulation result of MRST, represented by circle and diamond symbols. The right of Fig. 2 presents another example in which pure carbon dioxide (CO_2) is injected to displace C_1. A random Gaussian permeability distribution is generated to increase the complexity of the test example, while all the other parameters remain unchanged. With the nonlinearly preconditioned semismooth Newton algorithm, our solver converges rapidly to the solution that strictly preserves the boundedness within the physical range. It is clear the fully implicit simulator successfully characterizes the flow behavior of components in the fluid mixture.

Please cite this article as: H. Yang, S. Sun, Y. Li et al., Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.07.011.



(a) C_1 & C_3 (b) CO_2 & C_1

Fig. 2. Mole fraction profiles of fluid components on a homogeneous domain (left) and a heterogeneous domain (right).



(a) Permeability $\log(K)$ (b) $t = 0.5$

(c) $t = 1.0$ (d) $t = 1.5$

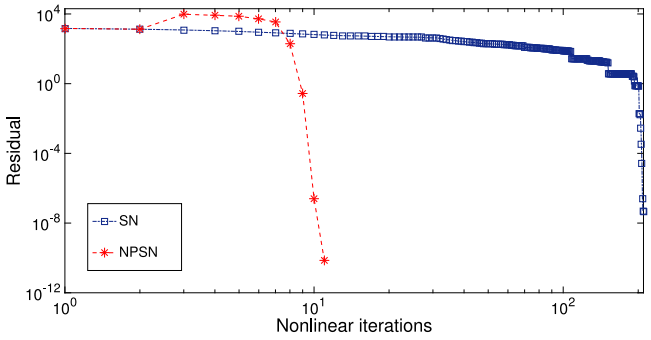
Fig. 3. Permeability field with the logarithmic scale and the molar density profiles for C_1.

4.2. Problems in highly heterogeneous porous media

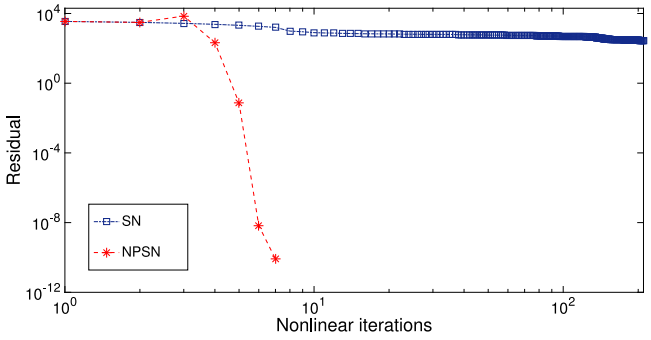
The aim of this subsection is to show the reliability of the proposed fully implicit simulator for solving problems in highly heterogeneous porous media. We first test the proposed method with a quarter five-spot problem [2,5,47] with random permeabilities. In the test, the reservoir domain is $[0, 300m] \times [0, 300m]$ and the permeability is generated again by a geostatistical model using the open source code MRST, as shown in the top left panel of Fig. 3. The reservoir is initially filled by pure propane (C_3), and methane (C_1) is injected with the flow rate 0.1 PV/year at the inflow boundary (the lower left corner), and the gas mixture is produced on the outflow boundary (the upper right corner). The temperature is fixed to $554.8K$ during the simulation. The values of physical parameters and other relevant data are given in Table 2. The contour plots of the molar density profiles at different times are shown in Fig. 3. In the test we set the time step to $\triangle t = 10^{-3}$ and run the simulation on a 100×100 mesh. We can see that the fully implicit approach successfully resolves the evolution of the molar density distributions.

We then present a comparison of the nonlinearly preconditioned semismooth Newton algorithm with its classical version. As introduced in Section 3, in the proposed fully implicit simulator, at each time step a nonlinear system needs to be solved

Please cite this article as: H. Yang, S. Sun, Y. Li et al., Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.07.011.



(a) At first time step



(b) At second time step

Fig. 4. Nonlinear residual histories for Case-3 at the first and second time steps.

Table 2
Relevant data for Case-3.

Parameters	Value
Domain dimensions [m^2]	300 × 300
Temperature [K]	554.8
Permeability [mD]	In Fig. 3
Porosity [-]	0.2
Mole fraction of initial gas [-]	0.0 C_1, 1.0 C_3
Mole fraction of injected gas [-]	1.0 C_1, 0.0 C_3
Initial pressure [Bar]	3.0
Injection rate [PV/year]	0.1

Table 3
Relevant data for Case-4.

Parameters	Value
Domain dimensions [ft^2]	2200 × 1200
Temperature [Kelvin]	554.8
Permeability [mD]	In Fig. 5
Porosity [-]	In Fig. 5
Mole fraction of initial gas [-]	0.0 C_1, 1.0 C_3
Mole fraction of injected gas [-]	1.0 C_1, 0.0 C_3
Initial pressure [Bar]	3.0
Injection rate [PV/year]	0.01

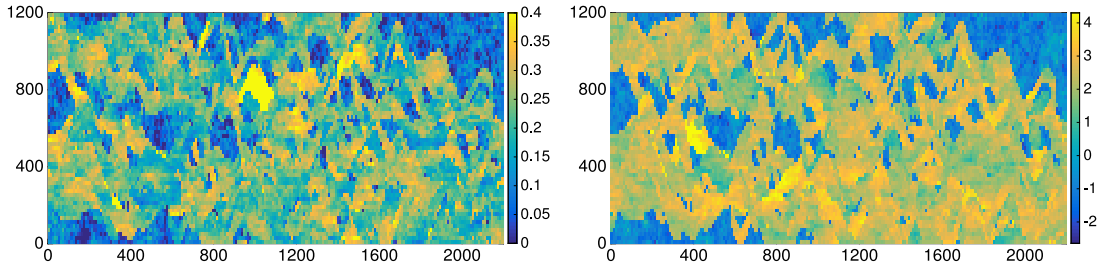
by the family of semismooth methods to maintain the predicted solution to stay within the physical range. However, this traditional nonlinear iterative method often fails to achieve the desired convergent rate due to the high nonlinearity of the system accompanied by the violation of the boundedness requirement of the mole fractions. Hence, in the study we use the nonlinearly preconditioning technique to fix this issue. In Fig. 4, we compare the histories of the nonlinear residual norms of the NPSN and SN methods for Case-3 on a 100 × 100 mesh. One can see that the nonlinear system is extremely hard to solve. As a result, the traditional semismooth Newton method takes many nonlinear iterations to finally converge or even diverge. Also, the performance of SN behaves similarly: the nonlinear residuals stagnate around without any improvement after many Newton iterations. The hard convergence of the nonlinear iteration for the SN method is often caused by the bad quality of the initial guess. On the other hand, with the help of the nonlinear preconditioner, the nonlinear system becomes much easier to solve and the number of nonlinear iterations for the NPSN method reduces quickly.

In next test case (Case-4), we import porosity and permeability data from the 85th layer of the Tenth SPE Comparative Project [48], which is a classic and challenging benchmark problem for reservoir due to the highly heterogeneous permeability and porosity. As shown in Fig. 5, the permeability is characterized by variations of more than six orders of magnitude and is ranged from 6.65×10^{-4} md to 2×10^4 md and the value of the porosity belongs to $[0, 0.5]$. The reservoir domain is 2200 ft × 1200 ft. There is one injection well at the center and there are four production wells at the four corners. The reservoir is initially filled by propane (C_3), and methane (C_1) is injected with the flow rate 0.01 PV/year. The rock and fluid properties are provided in Table 3. In Fig. 6, we display the composition distribution of C_1 and the pressure. The problem is solved on a 220 × 60 mesh and the simulation is ended at $t = 3$ with a fixed time step size $\triangle t = 10^{-3}$ year. Again, we can see that the nonlinearly preconditioned semismooth Newton method is clearly superior to the classical approach in terms of the number of nonlinear iterations, as shown in Fig. 7.

4.3. Parallel scalability tests

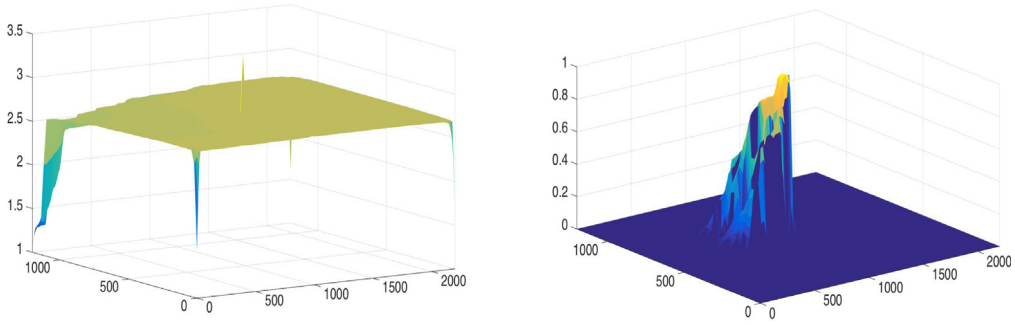
Scalability is an important topic in parallel computing. And an ideal scalability is hard to achieve, especially when solving large-scale problems with many processors. In this subsection,

Please cite this article as: H. Yang, S. Sun, Y. Li et al., Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.07.011.



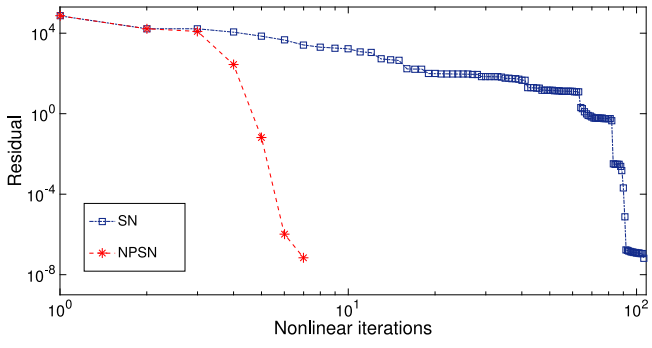
(a) Porosity ϕ (b) Permeability $\log(K)$

Fig. 5. Permeability field for the bottom layer of the SPE10 model in the simulation.

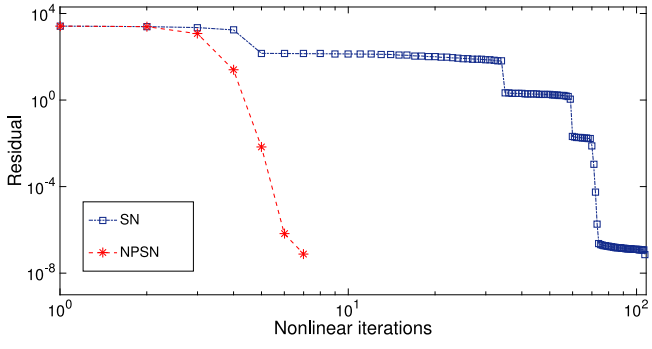


(a) Pressure p (b) C_1 mole fraction

Fig. 6. Pressure and molar density profiles for the bottom layer of the SPE10 model.



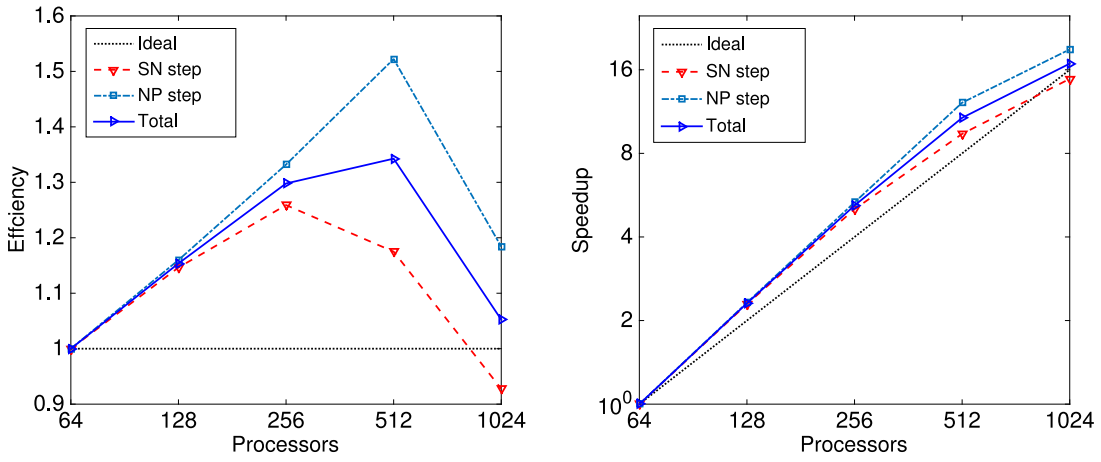
(a) At first time step



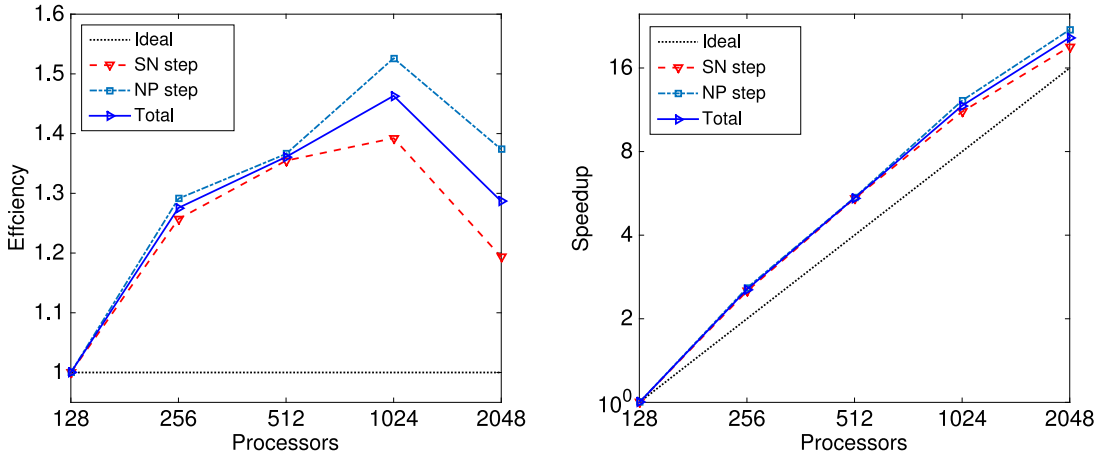
(b) At second time step

Fig. 7. Nonlinear residual histories for SPE10 at the first and second time steps.

Please cite this article as: H. Yang, S. Sun, Y. Li et al., Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.07.011.



(a) A 2048×1024 mesh



(b) A 4096×2048 mesh

Fig. 8. Scalability and efficiency results with different number of processor cores N_p and different grid sizes.

we numerically investigate the scalability and parallel efficiency of the proposed algorithm in terms of the computing time and number of iterations. The scalability *Speedup* and the parallel efficiency E_f, which are both calculated based on the computational time with the smallest processors number (N_{min}), are respectively defined as follows:

$$Speedup = \frac{T(N_{min})}{T(N_p)}, \quad E_f = \frac{T(N_{min}) \times N_{min}}{T(N_p) \times N_p},$$

where $T(N_p)$ denotes computational time with N_p processors. We implement the algorithms by using the Portable, Extensible Toolkit for Scientific computation (PETSc) library [49]. The numerical tests are carried out on the Tianhe-2 supercomputer. The computing nodes of Tianhe-2 are interconnected via a proprietary high performance network, and there are two 12-core Intel Ivy Bridge Xeon CPUs and 24 GB local memory in each node. In the numerical experiments, we use all 24 CPU cores in each node and assign one subdomain to each core. In the proposed NPSN method, we use an absolute tolerance of 10^{-6} for the global semismooth Newton step, and an absolute tolerance of 10^{-4} for the nonlinearly preconditioning (NP) step. All linear systems are solved by the Schwarz preconditioned GMRES method [13,41,42,50] with absolute and relative tolerances of 10^{-6} and 10^{-4}, respectively. Throughout all tables in this subsection, "N_p" stands for the number of processor cores, "N. It" is the average number of Newton iterations per time step, "L. It" is the average number of the preconditioned GMRES iterations per Newton iteration, and "Time" is the total computing time in seconds.

For the scalability tests, we conduct the numerical simulations in a 520 m×270 m domain [26,47]. The permeabilities are defined as

$$K = \begin{cases} 1 \text{ md} & 30 + 60i \le y \le 60 + 60i, \ i = 0, 1, 2, 3, \\ 100 \text{ md} & \text{otherwise}, \end{cases}$$

where (x, y) represents the position in the two-dimensional Cartesian coordinate system. The injection and production wells are located at the right and left sides, respectively. The injection rate is 0.01 PV/year. The rock and fluid are the same as Case-3. Table 4 shows the both numbers of nonlinear and linear iterations as the number of processor cores are gradually increased. Also shown in the figure is the computing time that is broken down into two parts: the semismooth Newton and nonlinearly preconditioning. The simulation is stopped after five implicit time steps with a fixed time step $\triangle t = 10^{-5}$ year using two fixed grid sizes: 2048×1024 and 4096×2048. The largest simulation consists of $4096 \times 2048 \times 2 = 16{,}777{,}216$ degrees of freedom. We can see that the number of nonlinear iterations per time step is independent of the number of processor cores, and the number of linear iterations grows very slow with the increase of the number of processor cores, which implies that the total number of nonlinear iterations and the average number of linear iterations are

Please cite this article as: H. Yang, S. Sun, Y. Li et al., Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.07.011.

Table 4
Test results using different numbers of processors N_p and grid sizes for NPSN.

Grid size	N_p	N. It	L. It	Time	N. It	L. It	Time	Time
		SN step			NP step			NPSN
2048 × 1024	64	4.0	5.7	28.2	5.2	4.9	34.1	62.3
	128	4.0	5.7	12.3	5.2	5.1	14.7	27.0
	256	4.0	5.9	5.6	5.2	5.3	6.4	12.0
	512	4.0	7.4	3.0	4.4	5.9	2.8	5.8
	1024	4.0	7.0	1.9	5.2	5.9	1.8	3.7
4096 × 2048	128	4.0	8.3	80.2	5.2	7.2	98.9	179.1
	256	4.0	8.9	31.9	5.2	7.4	38.3	70.2
	512	4.0	9.3	14.8	5.4	7.7	18.1	32.9
	1024	4.0	10.0	7.2	5.2	8.2	8.1	15.3
	2048	4.0	10.9	4.2	5.2	9.3	4.5	8.7

insensitive to the number of processor cores. In addition to that, the results reported in Fig. 8 show that the semismooth Newton method, combined with nonlinear preconditioning technique, can achieve an ideal parallel scalability with up to 2048 processor cores.

5. Conclusions

In this work, we have developed the family of nonlinearly preconditioned semismooth Newton methods based on the Fischer–Burmeister function to solve the discrete complementarity nonlinear systems, arising from the variational inequality of the transport of multicomponent compressible fluids in porous media. Numerical examples in one and two dimensions show that the proposed fully implicit simulator is significantly more robust and efficient than traditional inexact Newton methods, in terms of the property of positivity-preserving and the number of nonlinear iterations, even in the case of realistic discontinuous highly varying permeabilities. Since our approach follows the natural advantages of the fully coupled scheme, a particular emphasis of the proposed reservoir simulator is its potential to allow more physics to be added easily to the system without changing much of the algorithmic framework on a supercomputer.

Acknowledgments

The authors would like to express their appreciation to the anonymous reviewers for the invaluable comments that have greatly improved the quality of the manuscript. The work was supported in part by the National Natural Science Foundation of China (11571100 and 11871069).

References

[1] Z. Chen, G. Huan, Y. Ma, Computational Methods for Multiphase Flows in Porous Media, SIAM, Philadelphia, 2006.
[2] A. Firoozabadi, Thermodynamics of Hyrocarbon Reservoirs, McGraw-Hill, New York, 1999.
[3] K.A. Bulba, P.E. Krouskop, Oil Gas J. 107 (2009) 50–55.
[4] Y. Wang, Y. Wang, Z. Cheng, Polymers 11 (2019) 596.
[5] H. Hoteit, A. Firoozabadi, Water Resour. Res. 41 (2005) W11412.
[6] Y. Wang, S. Sun, B. Yu, Energies 10 (2017) 1380.
[7] Y. Wang, S. Sun, L. Gong, B. Yu, J. Nat. Gas Sci. Eng. 53 (2018) 301–316.
[8] X.-C. Cai, W.D. Gropp, D.E. Keyes, R.G. Melvin, D.P. Young, SIAM J. Sci. Comput. 19 (1998) 246–265.
[9] F.-N. Hwang, X.-C. Cai, J. Comput. Phys. 204 (2005) 666–691.
[10] D.A. Knoll, D.E. Keyes, J. Comput. Phys. 193 (2004) 357–397.
[11] S. Lacroix, Y.V. Vassilevski, J.A. Wheeler, M.F. Wheeler, SIAM J. Sci. Comput. 25 (2003) 905–926.
[12] L. Liu, D.E. Keyes, SIAM J. Sci. Comput. 37 (2015) A1388–A1409.
[13] H. Yang, S. Sun, Y. Li, C. Yang, Comput. Methods Appl. Mech. Engrg. 330 (2018) 334–350.
[14] J.E. Dennis, R.B. Schnabel, Numerical Methods for Unconstrained Optimization and Nonlinear Equations, SIAM, Philadelphia, 1996.
[15] S.C. Eisenstat, H.F. Walker, SIAM J. Optim. 4 (1994) 393–422.
[16] H. Yang, F.-N. Hwang, X.-C. Cai, SIAM J. Sci. Comput. 38 (2016) A2756–A2778.
[17] P. Brune, M. Knepley, B. Smith, X. Tu, SIAM Rev. 57 (2015) 535–565.
[18] L. Blank, H. Garcke, L. Sarbu, V. Styles, Numer. Methods Partial Differential Equations 29 (2013) 999–1030.
[19] I. Ekeland, R. Témam, Convex Analysis and Variational Problems, SIAM, Philadelphia, 1999.
[20] R. Glowinski, Numerical Methods for Nonlinear Variational Problems, Springer-Verlag, Berlin, 2008.
[21] H. Yang, S. Sun, C. Yang, J. Comput. Phys. 332 (2017) 1–20.
[22] F. Facchinei, J. Soares, SIAM J. Optim. 7 (1997) 225–247.
[23] C. Kanzow, Opt. Meth. Software 19 (2004) 309–325.
[24] S.J. Benson, T.S. Munson, Optim. Methods Softw. 21 (2006) 155–168.
[25] H. Yang, X.-C. Cai, J. Sci. Comput. 47 (2011) 258–280.
[26] H. Yang, C. Yang, S. Sun, SIAM J. Sci. Comput. 38 (2016) B593–B618.
[27] D. Kinderlehrer, G. Stampacchia, An Introduction To Variational Inequalities and their Applications, in: Pure and Applied Mathematics, New York-London, 1980.
[28] J.F. Rodrigues, Obstacle Problems in Mathematical Physics, North-Holland, Amsterdam, 1987.
[29] I. Ben Gharbia, J. Jaffré, Math. Comput. Simulation 99 (2014) 28–36.
[30] J. Chang, K.B. Nakshatrala, Comput. Methods Appl. Mech. Engrg. 320 (2017) 287–334.
[31] N.K. Mapakshi, J. Chang, K.B. Nakshatrala, J. Comput. Phys. 359 (2018) 137–163.
[32] J.O. Skogestad, E. Keilegavlen, J.M. Nordbotten, J. Comput. Phys. 234 (2013) 439–451.
[33] J.A. Trangenstein, J.B. Bell, SIAM J. Appl. Math. 49 (1989) 749–783.
[34] X.-C. Cai, X. Li, SIAM J. Sci. Comput. 33 (2011) 746–762.
[35] H. De Sterck, A. Howse, SIAM J. Sci. Comput. 38 (2016) A997–A1018.
[36] X.-C. Cai, D.E. Keyes, SIAM J. Sci. Comput. 24 (2002) 183–200.
[37] L. Liu, D.E. Keyes, R. Krause, SIAM J. Sci. Comput. 40 (2018) A1171–A1186.
[38] V. Dolean, M.J. Gander, W. Kheriji, F. Kwok, R. Massin, SIAM J. Sci. Comput. 38 (2016) A3357–A3380.
[39] J. Huang, C. Yang, X.-C. Cai, SIAM J. Sci. Comput. 38 (2016) A1701–A1724.
[40] Y.-S. Wu, P.A. Forsyth, J. Contam. Hydrol. 48 (2001) 277–304.
[41] Y. Saad, Iterative Methods for Sparse Linear Systems, second ed., SIAM, Philadelphia, 2003.
[42] A. Toselli, O. Widlund, Domain Decomposition Methods-Algorithms and Theory, Springer, Berlin, 2005.
[43] H.-J. Zhao, H. Yang, J. Comput. Appl. Math. 337 (2018) 37–50.
[44] K. Lie, S. Krogstad, I. Ligaarden, J. Natvig, H. Nilsen, B. Skaflestad, Comput. Geosci. 16 (2012) 297–322.
[45] K.-A. Lie, An Introduction to Reservoir Simulation Using MATLAB: User guide for the Matlab Reservoir Simulation Toolbox (MRST), SINTEF ICT, Oslo, 2016.
[46] J. Lohrenz, B.G. Bray, C.R. Clark, J. Pet. Technol. 16 (1964) 1–171.
[47] H. Hoteit, A. Firoozabadi, Adv. Water Resour. 31 (2008) 56–73.
[48] M. Christie, M. Blunt, SPE Reserv. Eval. Eng. (2001) 308–317.
[49] S. Balay, J. Brown, K. Buschelman, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, L.C. McInnes, B.F. Smith, H. Zhang, PETSc Users Manual, Argonne National Laboratory 2018.
[50] H. Yang, C. Yang, X.-C. Cai, J. Sci. Comput. 61 (2014) 258–280.

Please cite this article as: H. Yang, S. Sun, Y. Li et al., Parallel reservoir simulators for fully implicit complementarity formulation of multicomponent compressible flows, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.07.011.