

Advances in Water Resources
Manuscript Draft

Manuscript Number: ADWR-15-232

Title: An implicit numerical model for multicomponent compressible two-phase flow in porous media

Article Type: Research Paper

Keywords: Implicit scheme, MFE, FV, two-phase flow

Cover Letter

April 16, 2015

Editor,

Advances in Water Resources

Dear Editor,

This is the second paper as part of three papers. Part 1 was published in Advanced in Water Resources late last year. The third paper which is in progress combines the formulaⁿ from the first two papers to present a new formulaⁿ and efficient algorithm in two-phase flow in fractured media. We published a related paper on incompressible immiscible flows in fractured media in Advances in Water Resources in 2008.

Best regards,

Abbas Firoozabadi

Professor

Highlights (for review)

-
- 1- New Implicit discretization of the transport equations in compressible multicomponent two-phase flow
 - 2- Gravity is included in the new formulation
 - 3- Compositional variations is evaluated from change of species in a constant volume domain.
 - 4- New implicit discretization has very little numerical dispersion.
 - 5- Algorithm is much more robust than commercial codes.

An implicit numerical model for multicomponent compressible two-phase flow in porous media

Ali Zidane¹, Abbas Firoozabadi^{1,2}

(1) RESEARCH ENGINEERING RESEARCH INSTITUTE, PALO ALTO, CALIFORNIA, USA
(2) YALE UNIVERSITY, NEW HAVEN, USA

Submitted to *Journal of Advances in Water Resources*

1
2
3
4 **Abstract**
5
6

7 We introduce a new implicit approach to model multicomponent compressible two-phase flow in
8 porous media with species transfer between the phases. In the implicit discretization of the
9 species transport equation in our formulation we calculate for the first time the derivative of the
10 molar concentration of component i in phase α ($c_{\alpha,i}$) with respect to the total molar concentration
11 (c_i) under the conditions of a constant volume V and temperature T . The species transport
12 equation is discretized by the finite volume (FV) method. The fluxes are calculated based on
13 powerful features of the mixed finite element (MFE) method which provides the pressure at grid-
14 cell interfaces in addition to the pressure at the grid-cell center. The efficiency of the proposed
15 model is demonstrated by comparing our results with existing implicit compositional models.
16 Our algorithm has a low numerical dispersion despite the fact it is based on first-order space
17 discretization. The algorithm is very robust.

18
19
20
21
22
23
24
25
26
27
28
29
30 **Keywords:** Implicit scheme, MFE, FV, two-phase, Peng-Robinson, Newton-Raphson
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

1
2
3
4
5

1. Introduction

6 In two-phase compositional flow, most of the numerical models in the literature are based
7 on explicit approximation of the species balance equation [1]. The nonlinearity between the
8 phase molar and/or mass concentration and the total concentration in the implicit method
9 requires some complicated derivatives that do not appear in an explicit scheme. In the explicit
10 scheme, one of the major drawbacks is time step selection based on the Courant-Freidrichs-Levy
11 condition, known as the CFL condition. The CFL condition requires the time step to be less than
12 the necessary time for flow to pass through one mesh block. The impact of this condition
13 becomes severe when one grid block has a relatively small size compared to the size of the
14 simulation domain. Numerical modeling of fractured reservoirs is a clear example of small and
15 large grid cells. Most of the numerical models that deal with fractures require either to have
16 small mesh elements to explicitly model the fractures (e.g. single-porosity models) [2, 3, 4], or to
17 have very small elements near the fractures (e.g. Cross-Flow-Equilibrium approach) [5, 6, 7, 8].
18 In this case the numerical simulation becomes –computational wise- expensive, if an explicit
19 scheme is used in the numerical model. In compositional multiphase flow, the use of an explicit
20 method is preferred for the following reasons: i) the significant numerical dispersion usually
21 consorted with the implicit approximation, ii) the complexity of applying a straightforward
22 Newton method to solve the nonlinear system of equations in the compositional multiphase flow.
23 However, the importance of using implicit scheme arises when the use of an explicit scheme is –
24 CPU wise- expensive as mentioned above. This will be demonstrated in the core of this
25 manuscript when we compare computational cost of both implicit and explicit schemes using the
26 same gridding and then reducing the size of one or multiple grid blocks to study how the CFL
27 condition would affect the CPU time in the explicit scheme. Evidently, the CPU cost of the
28 implicit scheme will not be affected by the size of the grid block as much as by the number of the
29 grid blocks instead.

30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

Despite the fact the implicit approximations are incorporated in all of the commercial models, very few publications could be found when an implicit scheme is used for compressible multiphase flow. Most of the literature goes back to early publications. Coats [9] presented a fully implicit scheme for compositional multiphase flow. In his model the set of unknowns consist of the pressure, the saturations and the phase mole fractions. These variables are referred to as natural-type variables. Fussel and Fussel [10] presented a formulation that uses a different

1
2
3
4 set of variables based on phase compositions. These variables are referred to as the mass
5 variables. Later, Chien et al. [11] proposed using the equilibrium ratios as a set of unknowns
6 beside the pressure and the overall concentrations rather than saturations and phase
7 compositions.
8

9
10 In Fig.1 we show a comparison of our model to two different commercial models that we
11 denote by CM-1 and CM-2. The comparison is based on a modified 3-component mixture
12 example presented in [9]. The modified example is a 3-component, two-phase flow in 1-D.
13 Results show a high numerical dispersion of CM-1, even when going to very fine mesh
14 compared to CM-2. Only CM-2, after removing the over and under-shoots, is able to give
15 comparable results to our model. We should note that we also compared the results to a third
16 commercial model (CM-3). We do not show the results of CM-3 since it does not converge when
17 a more refined mesh than 120 grid-blocks is used in the modified Coats example. In 2D, the
18 effect of numerical dispersion becomes gleaming with the commercial models. To demonstrate
19 the low numerical dispersion of our model, we compare results of 1D and 2D examples (with our
20 model) to an explicit higher-order method (Discontinuous Galerkin method) in section 4.
21
22 The most recent work on compositional multiphase implicit scheme is reported in [12]. These
23 authors chose the set of unknowns based on the overall molar density of species and the pressure.
24 The phase compositions are updated in a post-processing step using constant volume and
25 temperature flash routines. To demonstrate the efficiency of our model, we compare the CPU
26 time in three different examples taken from [12] as will be shown later.
27
28

29 In this paper we present a new model that solves implicitly the species balance equation.
30 The species transport equation is discretized by using a finite volume (FV) approximation. The
31 Newton-Raphson method is used to solve implicitly the species transport equation. The
32 calculation of the derivatives in our formulation will be discussed later in detail. The total flux is
33 calculated by the hybridized mixed finite element method (MFE). The latter provides accurate
34 calculation of the velocity field even in highly heterogeneous media when compared to the
35 traditional finite element and finite volume methods [1, 5, 7, 13, 14, 15, 16, 17, 18, 19, 20]. The
36 strength of the MFE method is from the calculation of the pressure inside a finite element and the
37 traces of the pressures at the interfaces of each finite element in the computational domain. The
38 fluxes of each phase are deduced from the total flux by using the phase mobility coefficient as
39
40

What are
CM1/2 based
on?
Explain!

1
2
3
4 will be discussed later. In this paper the effect of gravity is taken into account. In two-phase
5 flow, the calculation of the phase fluxes is not trivial with gravity effect. The difficulty arises
6 from up-streaming the mobility of each phase. In absence of gravity, both phases are in the same
7 direction as the total flux. With gravity, however, counter current flow may develop at the finite
8 element interfaces due to the density contrast. Updating the phase mobilities based on the values
9 at the previous time step is not consistent. A phase could appear/disappear from one time step to
10 another. To resolve this complexity, we have developed an efficient method to upstream the
11 phase derivatives based on the updated phase mobilities and phase fluxes at the current time step.
12 In this paper we use the same upstreaming technique of the phase mobilities as [1, 7]. Once
13 evaluated, the phase fluxes are then coupled with our developed upstreaming technique of the
14 phase derivatives discussed in details in section 3.

15
16
17 The compressible behavior of each phase is described by the Peng-Robinson equation of
18 state [21]. The computation of composition of each phase is provided by the equality of
19 fugacities of each component in both (vapor and liquid) phases. This is based on the assumption
20 of local thermodynamic equilibrium; this calculation is commonly known in the literature by
21 flash calculation [22]. For a given pressure P and temperature T the flash calculation is
22 performed at each finite element of the computational domain and the calculation is known as
23 PT -flash calculation. However, when an implicit scheme is used to solve the species transport
24 equation, the derivative of concentration of each component of each phase is computed with
25 respect to the total concentration at constant volume V and temperature T . In this work we will
26 show how the derivative can be calculated without the VT -flash.
27
28 In addition to the new derivatives that appear in the species transport equation in the implicit
29 scheme, we couple the volumetric fluxes to the species equation differently from the work of [9,
30 12] and others. The pressure in our formulation is calculated in a preprocessing step in order to
31 update the fluxes based on the converged values of the molar densities. We believe that this
32 approach reduces the number of iterations per time step when compared to the implicit update of
33 the pressures with the molar densities and compositions. The numerical examples in this work
34 will be compared to a higher-order explicit method, the Discontinuous Galerkin (DG) method.
35
36 The aim of this work is to show that our implicit model can produce accurate results even when
37 compared to the higher-order methods such as DG. We believe that the implicit scheme can be
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

more efficient than the explicit scheme where the CFL condition has a severe constraint on the time step as we mentioned above.

The rest of the paper is organized as follows: in the next section we provide the differential equations describing the multicomponent compressible two-phase flow in porous media. Then we present the discretization of the pressure and the species mass balance equations. We present six numerical examples to demonstrate the efficiency and accuracy of the proposed algorithm.

2. Mathematical model

2.1 Species balance equation

The mass balance of component i in compressible two-phase (gas and oil) flow of n_c -component mixture is given by the following equations:

$$\emptyset \frac{\partial c z_i}{\partial t} + \nabla \cdot \left(\sum_{\alpha} c_{\alpha} x_{i,\alpha} v_{\alpha} \right) = F_i, \quad i = 1 \dots n_c \text{ in } \Omega \times (0, \tau) \quad (1)$$

The above equation is subject to the following constraints:

$$\sum_{i=1}^{n_c} z_i = \sum_{i=1}^{n_c} x_{i,\alpha} = 1 \quad \forall \alpha \quad (2)$$

In the above equations, \emptyset denotes the porosity, v_{α} the velocity of phase α , c the overall molar density of the mixture; z_i and F_i are the overall mole fraction and the sink/source term of component i in the mixture, respectively. c_{α} is the molar density of phase α and $x_{i,\alpha}$ is the mole fraction of component i in phase α . Ω is the computational domain and τ denotes the simulation time and n_c is the number of components. We neglect diffusion in Eq. (1).

The velocity for each phase α is given by the Darcy law:

$$v_{\alpha} = -\frac{\mathbf{K} k_{r\alpha}}{\mu_{\alpha}} (\nabla p - \rho_{\alpha} \mathbf{g}) = -\lambda_{\alpha} \mathbf{K} (\nabla p - \rho_{\alpha} \mathbf{g}), \quad \alpha = o, g \quad (3)$$

where \mathbf{K} is the absolute permeability, $k_{r\alpha}$, μ_{α} and ρ_{α} are the relative permeability, dynamic viscosity and mass density of phase α respectively, with $\lambda_{\alpha} = k_{r\alpha}/\mu_{\alpha}$; p is the pressure and \mathbf{g} is the gravitational acceleration. To calculate the relative permeabilities we use Stone's model [23, 24], and to find the phase viscosities we follow Lohrenz et al [25].

1
2
3
4 2.2 Pressure equation

5
6 The pressure equation is calculated using the concept of total volume balance [26, 27] given by:

	$\phi C_t \frac{\partial p}{\partial t} + \sum_{i=1}^{n_c} \bar{V}_i \nabla \cdot \left(\sum_{\alpha} c_{\alpha} x_{i,\alpha} \mathbf{v}_{\alpha} \right) = \sum_{i=1}^{n_c} \bar{V}_i F_i$	(4)
--	--	-----

7
8 where C_t is the total compressibility and \bar{V}_i is the total partial molar volume of component i [see
9
10
11
12
13 e.g. 22].

14
15 The local thermodynamic equilibrium implies the equality of the fugacities of each component in
16
17 the two phases:

	$f_{o,i}(T, p, x_{j,o}) = f_{g,i}(T, p, x_{j,g}), \quad i = 1, \dots n_c; j = 1, \dots n_c - 1$	(5)
--	---	-----

18
19 The phase and volumetric behavior are modeled by the Peng-Robinson equation of state [21]:

	$c_{\alpha} = \frac{N_{\alpha}}{V_{\alpha}} = \frac{p}{Z_{\alpha} RT}, \quad \rho_{\alpha} = c_{\alpha} \sum_{i=1}^{n_c} x_{i,\alpha} M_i$	(6)
--	--	-----

$$Z_{\alpha}^3 - (1 - B_{\alpha})Z_{\alpha}^2 + (A_{\alpha} - 3B_{\alpha}^2 - 2B_{\alpha})Z_{\alpha} - (A_{\alpha}B_{\alpha} - B_{\alpha}^2 - B_{\alpha}^3) = 0$$

20
21 where N_{α} , V_{α} , Z_{α} , c_{α} , ρ_{α} , are the number of moles, volume, compressibility factor, molar and
22
23 mass density of phase α . M_i is the molar weight of component i . R is the universal gas constant
24
25 and T is the temperature. A_{α} and B_{α} are the parameters of the PR-EOS which depend on
26
27 pressure, temperature and composition of each phase [see e.g. 22].

28
29 The saturation of each phase is calculated from

	$S_{\alpha} = \frac{c}{c_{\alpha}} \omega_{\alpha}$	(7)
--	---	-----

30
31 where $\omega_{\alpha} = N_{\alpha} / \sum_{\beta} N_{\beta}$. Using Eq.(7), the saturation constraint could be then written in the
32
33 following form:

	$1 - c \left(\frac{\omega_o}{c_o} + \frac{\omega_g}{c_g} \right) = 0$	(8)
--	--	-----

34
35 The above equation is used as a criterion for the selection of a time step.

36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65 β : remaining phases? or all phases?
 ↳ mole/mol
 fraction

$$\omega_{\alpha} = \frac{N_{\alpha}}{N_1 + N_2 + \dots + N_{np}}, \text{ not } N_{\alpha}$$

1
2
3
4
5

3. Numerical discretization

3.1 Discretization of the transport equation

In the following we show how we solve implicitly the species transport equation. Using a FV integration in Eq.(1) we get:

$$G_{K,i} = \emptyset |K| \frac{cz_{i,K}^{n+1} - cz_{i,K}^n}{\Delta t} + \sum_{E \in \partial K} \sum_{\alpha} \left[\widetilde{c_{\alpha} x_{i,\alpha}}^{n+1} q_{\alpha,K,E} \right] |K| F_i = 0 \quad (9)$$

where $|K|$ is the surface area of the finite element K , $\widehat{c_\alpha x_{i,\alpha}}$ is the upstream value of $c_\alpha x_{i,\alpha}$, and $q_{\alpha,K,E}$ is the normal flux of phase α at the interface E of element K .

In the above equation, $G_{K,i}$ is a function of $cz_{i,K}$ and the surrounding elements of the element K .

In the Newton-Raphson method one needs to evaluate the term $\partial G_{K,i} / \partial c z_i^{n+1}$, that is:

$$\frac{\partial G_{K,i}}{\partial c z_i^{n+1}} = \frac{\emptyset |K|}{\Delta t} + \sum_{E \in \partial K} \sum_{\alpha} \left\{ \frac{\partial \widetilde{c_{\alpha} x_{t,\alpha}}^{n+1}}{\partial c z_i^{n+1}} q_{\alpha,K,E} \right\} \quad (10)$$

The two major complexities in the above equation are: i) the evaluation of the term $\partial c_{\alpha} x_{i,\alpha} / \partial cz_i$, and ii) upstream the derivative $\widetilde{\partial c_{\alpha} x_{i,\alpha}} / \partial cz_i$ if one or more of the surrounding elements are in the upstream direction with respect to element K . In the next section we will show how to calculate these derivatives and how to upstream a derivative if the surrounding elements are in different phases than the element K .

3.1.1 Evaluation of the derivative $\partial c_\alpha x_{i,\alpha} / \partial cz_i$

We represent $c_{\alpha}x_i$ by c_i and $c_{\alpha}x_{i,\alpha}$ by $c_{\alpha,i}$. From the definition of the molar density of component i in phase α , $c_{\alpha,i}$, the derivative of $c_{\alpha,i}$ ($= n_{\alpha,i}/V_\alpha$) with respect to the total molar density c_i could be written in the following form:

$$\frac{\partial c_{\alpha,i}}{\partial c_i} = \frac{V_\alpha \partial n_{\alpha,i}/\partial c_i - n_{\alpha,i} \partial V_\alpha/\partial c_i}{V_\alpha^2} \quad (11)$$

In order to calculate $\partial c_{\alpha,i} / \partial c_i$ one needs to evaluate $\partial n_{\alpha,i} / \partial c_i$ and $\partial V_\alpha / \partial c_i$. To do so we use: i) thermodynamic equilibrium based on fugacities, and ii) constant volume at each grid cell element in the computational domain.

At constant volume and temperature (the conditions inside each grid-cell) the fugacity of each component i in phase α , $f_{\alpha,i}$ is a function of the number of moles of all the components in the

same phase. The variation of the fugacity $f_{\alpha,i}$ with respect to each component l is written as follows:

$$\left(\frac{\partial f_{\alpha,i}}{\partial n_l} \right)_{n_l} = \sum_j \left(\frac{\partial f_{\alpha,i}}{\partial n_{\alpha,j}} \right)_{p,n_{\alpha,j}} \left(\frac{\partial n_{\alpha,j}}{\partial n_l} \right)_{n_l} + \left(\frac{\partial f_{\alpha,i}}{\partial p} \right)_{n_\alpha} \left(\frac{\partial p}{\partial n_l} \right)_{n_l} \quad (12)$$

where the vector $\mathbf{n}_l = (n_1, n_2, \dots, n_{l-1}, n_{l+1}, \dots, n_c)$. The total volume index is dropped for simplicity.

Using Eq.(12) for both phases with the equality of fugacities in Eq.(5) we obtain

$$\sum_j \left(\frac{\partial f_{g,i}}{\partial n_{g,j}} + \frac{\partial f_{o,i}}{\partial n_{o,j}} \right) \frac{\partial n_{o,j}}{\partial n_l} + \left(\frac{\partial f_{o,i}}{\partial p} - \frac{\partial f_{g,i}}{\partial p} \right) \frac{\partial p}{\partial n_l} = \frac{\partial f_{g,i}}{\partial n_{g,l}} \quad (13)$$

At element level, the total volume is the summation of volumes of the oil and gas phases; it is equal to the size of the finite element grid cell:

$$V_t = V_o + V_g \quad (14)$$

The variation of the volumes with respect to variation of moles of component i could be then written as:

$$\left(\frac{\partial V_t}{\partial n_i} \right)_{n_i} = \left(\frac{\partial V_o}{\partial n_i} \right)_{n_i} + \left(\frac{\partial V_g}{\partial n_i} \right)_{n_i} = 0 \quad (15)$$

From the equation of state for each phase :

$$\begin{aligned} \left(\frac{\partial V_\alpha}{\partial n_i} \right)_{n_i} &= RT \left[-\frac{N_\alpha Z_\alpha}{p^2} \frac{\partial p}{\partial n_i} + \frac{1}{p} \left(N_\alpha \frac{\partial Z_\alpha}{\partial n_i} + Z_\alpha \frac{\partial N_\alpha}{\partial n_i} \right) \right] \\ \left(\frac{\partial Z_\alpha}{\partial n_i} \right)_{n_i} &= \sum_{j=1}^{n_c} \left(\frac{\partial Z_\alpha}{\partial n_{\alpha,j}} \right)_{p,n_{\alpha,j}} \left(\frac{\partial n_{\alpha,j}}{\partial n_i} \right)_{n_i} + \left(\frac{\partial Z_\alpha}{\partial p} \right)_{n_1} \left(\frac{\partial p}{\partial n_i} \right)_{n_i} \end{aligned} \quad (16)$$

Using Eq.(16) in Eq.(15) and after some arrangements:

$$\begin{aligned} &\left(-\frac{N_g Z_g}{p^2} - \frac{N_o Z_o}{p^2} + \frac{N_g}{p} \frac{\partial Z_g}{\partial p} + \frac{N_o}{p} \frac{\partial Z_o}{\partial p} \right) \frac{\partial p}{\partial n_i} + \frac{1}{p} N_g \frac{\partial Z_g}{\partial n_{g,i}} + \frac{1}{p} Z_g \\ &+ \frac{1}{p} \left(\sum_{j=1}^{n_c} \left(N_o \frac{\partial Z_o}{\partial n_{o,j}} + Z_o - N_g \frac{\partial Z_g}{\partial n_{g,j}} - Z_g \right) \frac{\partial n_{o,j}}{\partial n_i} \right) = 0 \end{aligned} \quad (17)$$

Eq.(13) and Eq.(17) form a system of equations for each component i in the mixture which are used to calculate $\partial n_{\alpha,i}/\partial n_i$ and $\partial p/\partial n_i$. Once these derivatives are evaluated we use them in Eq.(11) to find $\partial c_{\alpha,i}/\partial c_i$ knowing that for every quantity X :

$$\frac{\partial X}{\partial c_i} = \frac{\partial X}{\partial n_i} V_t$$

(18)

Once $\partial c_{\alpha,i}/\partial c_i$ are evaluated, the phase fluxes $q_{\alpha,K,E}$ are used to upstream the derivative $\widetilde{\partial c_{\alpha}x_{i,\alpha}}/\partial cz_i$ at each interface of element K . However, the upstream depends on the number of phases inside the adjacent element of K that we denote by K' as follows (this applies for all the adjacent elements of K):

- i) If $q_{\alpha,K,E} > 0$ then $\widetilde{\partial c_{\alpha}x_{i,\alpha}}/\partial cz_i = (\partial c_{\alpha}x_{i,\alpha}/\partial cz_i)_K$; the subscript K denotes that the derivative is evaluated in element K .
- ii) If $q_{\alpha,K,E} < 0$ then four different cases should be considered:
 - 1) When elements K and K' are both in two-phase then we simply upstream each phase derivative from K' to K . Hence $\widetilde{\partial c_{\alpha}x_{i,\alpha}}/\partial cz_i = (\partial c_{\alpha}x_{i,\alpha}/\partial cz_i)_K$,
 - 2) If the element K is in two phase and K' is in single phase then the derivative of the existing phase in K' is set to one and the derivative of the absent phase is set to zero. This is readily deduced from the fact that if there is only oil in K' then the molar concentration of the oil phase is the total molar concentration of component i and hence its derivative is unity. The zero value of the absent phase derivative follows the same logic.
 - 3) If the element K is in single phase and K' is in two phase then the opposite procedure in (2) is followed.
 - 4) If both elements are in single phase, then it is a combination of (2) and (3) together.

We have examined the derivative $(\partial c_{\alpha}x_{i,\alpha}/\partial cz_i)_{V,T}$ for a number of conditions; it varies in the range of -1.2 to 1.2. A detailed plot that shows the variation of the derivatives as a function of PVI is presented in Appendix A. It is the small variation that may have a profound effect on convergence in our formulation.

3.2 Discretization of the total and phase fluxes

The total flux is approximated by the lowest order Raviart-Thomas space (RT_0). The resulting velocity with the MFE has a marked improvement compared to traditional FD methods [28],

29]. With the MFE method, the total velocity in each grid-cell K is written in terms of the normal
 fluxes across each interface E of K as follows:

$$\nu = \sum_{\alpha} \nu_{\alpha} = \sum_{E \in \partial K} q_{K,E} w_{K,E} \quad (19)$$

where $w_{K,E}$ is the RT_0 basis function across edge E of element K and $q_{K,E}$ is the normal flux at
 interface E and is calculated through the average cell pressure of K and the traces of pressure at
 interfaces of K as follows:

$$q_{K,E} = \alpha_{K,E} p_K - \sum_{E' \in \partial K} \beta_{K,E,E'} t p_{K,E'} - \gamma_{K,E} \quad (20)$$

The coefficients $\alpha_{K,E}$, $\beta_{K,E,E'}$ and $\gamma_{K,E}$ depend on the geometrical shape of the element and the
 mobility. For more details about these coefficients and the MFE formulation the reader may refer
 to [15, 16, 30, 31, 32, 33].

Once the total velocity is evaluated we can calculate the velocity of each phase independently by
 using Eq.(3) and Eq. (19) :

$$\nu_{\alpha} = f_{\alpha}(\nu - G_{\alpha}) \quad (21)$$

with

$$f_{\alpha} = \frac{\lambda_{\alpha}}{\sum_{\beta} \lambda_{\beta}} \text{ and } G_{\alpha} = \begin{cases} \lambda_o (\rho_o - \rho_g) g, & \text{if } \alpha = g \\ \lambda_g (\rho_g - \rho_o) g, & \text{if } \alpha = o \end{cases} \quad (22)$$

3.3 Discretization of the pressure equation

Inserting Eq.(21) into Eq.(4) and integrating over a finite element K we obtain after using
 Gauss's theorem on the divergence term :

$$\emptyset |K| C_t \frac{\Delta p}{\Delta t} + \sum_{i=1}^{n_c} \sum_{\alpha} \sum_{E \in \partial K} \bar{V}_{i,K} \int_E c_{\alpha} x_{i,\alpha} f_{\alpha} (\nu - G_{\alpha}) \cdot \mathbf{n}_E = \sum_{i=1}^{n_c} \bar{V}_{i,K} F_{i,K} \quad (23)$$

The coefficient $c_{\alpha} x_{i,\alpha}$, f_{α} and \bar{V}_i are evaluated at the element center without considering higher-order spatial approximation. This is due to the fact that the pressure is a smooth function of space in the scope of this work. The same upwind technique that is used in species transport equation

1
2
3
4 (based on the direction of the phase flux) cannot be used here since the phase fluxes are not
5 known yet.
6
7
8
9

10 **4. Numerical examples**
11

12 In the following we present six numerical examples with the number of species varying from 2 to
13 10 in structured grids to investigate the efficiency of the proposed model. In the first example the
14 domain is in 1-D with 5 components. In the rest of the examples the domain is 2-D and the
15 number of components varies between 2, 8 and 10. An Intel Core-i5 PC, 3 GHZ CPU, 4
16 GB RAM is used in all the runs.
17
18

19 **4.1 Example 1: 5-component mixture in 1-D**
20

21 In this 1-D example, methane is injected at the left side of a 50-m long domain. Production is at
22 the right side. The domain is initially saturated with a mixture of equal mole fractions of
23 $C_2/C_3/C_4/C_5$. The higher number of components implies larger matrix inversions (see Eq. 10) at
24 each time step. At each iteration the number of matrix inversions is equal to the number of
25 components times the number of grid blocks in the computational domain. The relevant data of
26 the domain are given in Table 1. Three different mesh refinements are used with 50, 100 and
27 200 grid blocks. The overall compositions of the 5 components after 60% PV injection are
28 plotted in Fig.2 with the 200 grid blocks for both the implicit and DG methods. There is
29 agreement for all of the components between our implicit results and the DG results. The CPU
30 cost of the implicit scheme and the high-order explicit scheme with the same gridding is shown
31 in Table 2. Obviously, the explicit scheme is more efficient when the CFL condition does not put
32 a serious limitation on the time step. However, in Table 2 we show the CPU time ratio of the
33 implicit scheme over the explicit. Even with these coarse meshes, the CPU ratio reduces from 4.6
34 to 3.6 when the number of elements increases to 200. Despite the fact that the matrix inversion is
35 more expensive when the number of elements increases, the CFL condition with a more refined
36 mesh affects the CPU time. For further examination, we pick up one grid block (first block on
37 the left side) of the 200-element mesh in both models (DG and implicit) and divide its size by
38 two, then divide it by ten (i.e. one grid block has the size of half of the rest of the elements in the
39 first case, and in the second case one grid block has one tenth of the size of the rest of the
40 elements). We refer to these cases in Table 2 by cut-2 and cut-10. When the size of one grid
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

1
2
3
4 block reduces to half and to one tenth of the original 200 elements mesh, the CPU time ratio
5 reduces from 2.5 down to 1.4 with cut-10. Furthermore, if we reduce the size of one grid block
6 50 times (cut-50, Table 2) the implicit scheme is now more efficient than the explicit scheme
7 with almost the same level of accuracy (Fig.2). Here we should note that: 1) The results from the
8 original and the 3 cuts (cut-2, cut-10, and cut-50) are exactly the same as shown in Fig.2; 2) The
9 CPU time for the implicit scheme is the same in the three cuts, the 0.2 and 0.3 sec are just
10 numerical fluctuations; 3) having one element that is smaller by a factor of 50 and even 100 than
11 the rest of the elements in the computational domain, is very common in fractured reservoir
12 simulation. If one describes flow in small grids by the implicit method and larger grids by the
13 explicit method, the method is called the adaptive implicit method.

23 4.2 Example 2: 2-component mixture in 2-D

24 In this example the domain is $50 \times 50 \text{ m}^2$. Methane is injected at the bottom left corner to displace
25 the initially saturated propane to the production well at the top right corner of the domain. The
26 relevant data of this example are given in Table 3. We examine 5 different mesh refinements
27 from a very coarse mesh of 36 elements to 100, 400, 1600 and 3600 elements (Fig.3). In Fig.4
28 we show the composition (overall mole fraction) profile of methane at 70% PV injection with
29 different mesh refinement. The result of the fine mesh with 3600 elements is compared to the
30 DG explicit method with the same mesh refinement; there is good agreement (Fig.4 c, d). The
31 CPU time for different mesh refinements is shown in Table 4. Results show that as the size of the
32 grid block decreases (more refined mesh) the difference between the explicit scheme and the
33 implicit model reduces. With 3600-element fine mesh the implicit scheme becomes faster than
34 the DG explicit scheme. This is due to the effect of the CFL condition with the explicit scheme
35 and to the efficiency of our scheme. The average number of Newton iterations per time step in
36 this example is 2.75.

50 4.3 Example 3: 8-component mixture in 2-D

51 The domain in this example is a $50 \times 50 \text{ m}^2$; it is initially saturated with the 8-component oil. CO_2
52 is injected at the bottom left corner to displace the oil to the opposite corner. The relevant data
53 for this example are given in Table 5 and the compositions of the initial and injected components
54 are given in Table 6. The PR-EOS parameters for each component are given in Table 7 and the
55 symmetrical binary interaction parameters matrix is given in Table 8. Similar to the previous
56

example, we use different mesh refinements, and show the results for the more refined mesh of 3600 elements. The overall compositions of the first and fourth components and the gas saturation are shown in Fig.5. The composition profiles of the several components and the gas saturation demonstrate the accuracy of our scheme when compared to the higher-order DG method. Evidently, the CPU time with the 8-component mixture is more expensive than the two-component mixture since the number of matrix inversions at each iteration is directly related to the number of components and the number of elements (grid blocks). However, in our implicit model, going from a 2-component mixture to an 8-component mixture, the CPU time increases by a factor of 4.2. With the 8-components mixture, the implicit method requires around 16 min (Table 9) to converge to the results shown in Fig.5 compared to 3.8 minutes with a 2-component mixture. On the other hand, the CPU time of the explicit DG scheme is 14 min if the same mesh refinement is used due to the CFL constraint on a refined mesh. In case of a more refined mesh with the explicit scheme, the use of our implicit model becomes a natural choice. The efficiency of our model is compared with two different commercial codes (CM-1) and (CM-2). Since the runs of our model and the two commercial codes are performed in different machines, the CPU time would not give clear demonstration of the efficiency. Therefore, we choose to compare three parameters; the total number of time steps, the total number of Newton iterations and the average number of Newton iterations per time step during the whole simulation. Results show that our model is more efficient than CM-1 simulator, in terms of the total number of time steps and the total number of Newton iterations for the three different mesh refinements that have been used. With the refined mesh of 3600 elements, the average number of Newton iterations per time step is 2.91 compared to 2.88 in our model (Table 10). A total number of 102 time steps are required by CM-2 with the refined mesh of 3600 elements, compared to 162 time steps in our model. As a result the average number of iterations per time step is 2.85 with CM-2 compared to 2.88 with our model. To demonstrate the low numerical dispersion in our model we compare the results to an explicit higher-order DG method (Fig.5); results show the contour profiles in our model and in the explicit DG are about the same. The CPU time with our model in this example is 16 min for a simulation time of 2 years, and with a total of 3600 elements.

4.4 Example 4: 10-component mixture in a 2-D domain

We consider a $500 \times 150 \text{ m}^2$ reservoir that is initially saturated with 10 components oil. The relevant data of the domain, the initial and injected compositions are the same as example-3. The

1
2
3
4 8-component mixture of the last example is adapted to 10-component mixture by dividing the C₁
5 component and the C₂-C₃ component into C₁-a, C₁-b and C₂-a, C₂-b, respectively. In this case the
6 phase behavior will not be affected but the performance of the model will be influenced by the
7 additional two components and the fine gridding used in this larger domain. The domain is
8 discretized by 160 grids in the x-direction and 48 grids in the y-direction what makes a total of
9 7680 structured grid elements. With this discretization, both the horizontal and vertical elements
10 have a specific length of 3.125 m. For reference we show in Fig.6 the composition of CO₂ and
11 the gas saturation at different PV injections. For brevity we do not show the overall compositions
12 for the rest of the components. The efficiency of our model in this example is compared to the
13 higher-order DG method. The CPU time required in our model is about 33 min when it is in the
14 order of 41 min with DG and the average number of Newton iteration per step is 2.82 in our
15 model (Table 11).
16
17
18
19
20
21
22
23
24
25
26

4.5 Example 5: 2-component mixture in 2-D with gravity

27 In this example we consider a vertical domain i.e. the gravity effect is taken into account. The
28 input data of this example are the same as example 2. With gravity, counter current flow may
29 develops at the finite element interfaces, and this could add more restrictions on the time step. In
30 this example we demonstrate the efficiency of our model when the gravitational effect is taken
31 into account. In this 2-component mixture, the CPU time is 125 sec. It is 51 sec without gravity
32 with the same 1600 mesh refinement. The average number of newton iterations per time step
33 increases from 2.7 iterations/Δt without gravity to 2.92 iterations/Δt with gravity. However, this
34 increase is expected due to the fact that in absence of gravity, when a phase appears/disappears,
35 the phase flux is always in the same direction as the total flux (whether if it is influx or out-flux).
36 In an implicit update, this means, more elements contribute to the molar density update (in fact to
37 the derivative of the molar density) at the interface where the flux is evaluated. With gravity, the
38 two adjacent elements of one interface could then contribute to the calculation (as discussed in
39 section 3) and hence more iterations are required to reach convergence. The gas saturation and
40 the composition of the injected methane are shown for different PVI in Fig.7. With a more
41 refined mesh of 3600 elements (results not shown) the CPU time becomes 6 min for a simulation
42 time of 2 years.
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

4.6 Example 6: 8-component mixture in 2-D with gravity

In the last example we consider injection of CO₂ into a vertical domain saturated with 8-component oil. Dimensions and properties of the domain are the same as example 3. The high number of components affects CPU time. More components require more matrix inversions. With gravity the complexity of the flow increases, hence the number of iterations increases at each time step, as discussed earlier. Together, the high number of components and the gravity effect should have a significant effect on the CPU time. However, we demonstrate in this example the efficiency of our model even in this complicated scenario. The average number of newton iterations per time step is 3.21 iterations compared to 2.72 iterations without gravity. Comparing the CPU time to [12] in this example, the authors report that the CPU time for this 8-component mixture with gravity is 96.7 hours for a simulation time of 1.36 years and total of 3200 elements. In our model, the CPU time in this example is 32 min for a simulation time of 2 years, and a mesh refinement of 3600 elements. The results from our code are the same as in [12] (Fig. 8).

We should note that the total number of iterations per time step could be reduced in this example if we limit the maximum allowed time step by a factor of ten to become 10^{-3} years; the CPU time in this case increases to 52 min. For reference we show in Fig.8 the gas saturation and the compositions for different components after 60 % PVI.

5. Conclusions

In this work we have introduced an efficient numerical model for implicit treatment of the species transport equation in compositional two-phase flow in porous media. Our new formulation is faster than the existing implicit models in the literature. The efficiency of our model is also comparable and in general superior to existing commercial codes that are based on implicit models. However, the numerical dispersion for our model is less than in the commercial models we have tested even when compared to explicit higher order method. The efficiency of our model is due to two factors:

- i) The calculation of the derivatives of the molar concentration at constant volumes of each component in each phase with respect to the total molar concentration. The derivatives in all of our examples, varies in the range [-1.2, 1.2]. The small variation of the nonlinear

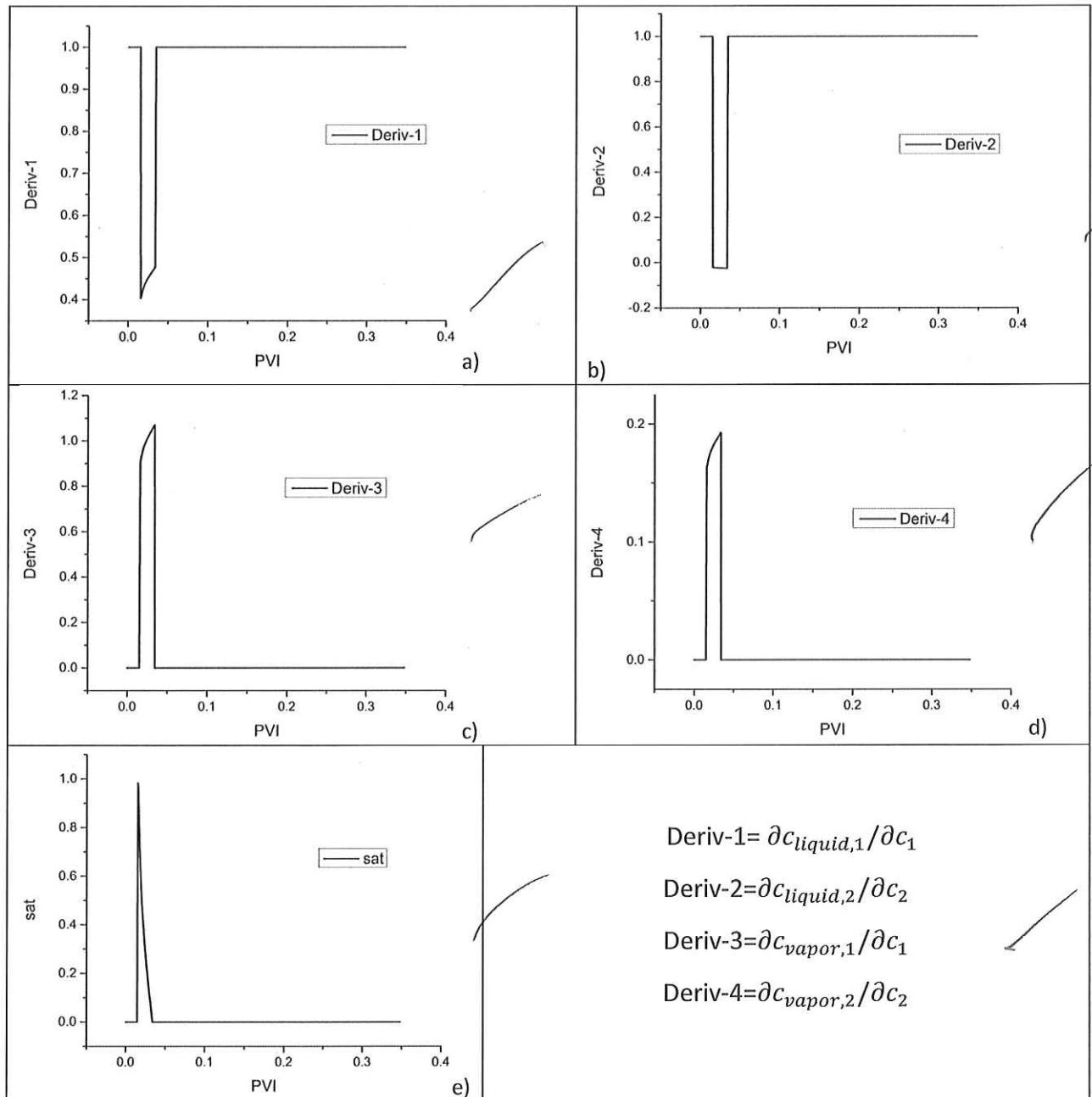
1
2
3
4 coefficient within the Newton method results in a fast convergence at each time step.
5
6 And this reduces significantly the total number of Newton iterations, hence, CPU time.
7
8 We also believe that the small range of variation of the calculated derivatives is related
9 to the low level of numerical dispersion observed in our model compared to existing
10 implicit codes.
11
12

- 13 ii) The way we use to update the phase fluxes is based on the converged solution of the
14 molar concentrations and mole fractions. This coupling between the species transport
15 equation and the phase fluxes, reduces the number of iterations at each time step
16 compared to the existing implicit formulations.
17
18

19 In all the examples we have run the maximum number of iterations per time step did not exceed
20
21 5. The average number of iteration without gravity is generally less than 3 and with gravity is
22 generally less than 4.
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

1
2
3
4 **Appendix A**

5
6 To show the variation of the derivatives we use example 2 and replace the number of
7 components by 2 instead of 5 for simplicity. In Fig. A1 we show the variation of the derivatives
8 as a function of PVI for a C₁ (species 1) injection into a domain saturated with C₃ (species 2).
9
10
11



59 Fig. A1: Variation of the derivatives (a,b,c,d) and gas saturation (e) as a function of PVI
60
61

1
2
3
4 **References**
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

- [1] Hoteit H, Firoozabadi A. Compositional modeling by the combined discontinuous Galerkin and mixed methods. *SPE J*, 2006;11(1):19–34.
- [2] Noorishad J, Mehran M. An upstream finite element method for solution of transient transport equation in fractured porous media. *Water Resour Res* 1982;18(3):588–96.
- [3] Baca R, Arnett R, Langford D. Modeling fluid flow in fractured porous rock masses by finite element techniques. *Int J Num Meth Fluids* 1984;4:337–48.
- [4] Granet S, Fabrie P, Lemmonier P, Quintard M. A single phase flow simulation of fractured reservoir using a discrete representation of fractures. In: Proceedings of the 6th European conference on the mathematics of oil recovery (ECMOR VI), September 8–11, Peebles, Scotland, UK; 1998.
- [5] Hoteit H, Firoozabadi A. Multicomponent fluid flow by discontinuous Galerkin and mixed methods in unfractured and fractured media. *Water Resour Res* 2005;41(11):W11412.
- [6] Hoteit H, Firoozabadi A. Compositional modeling of discrete-fractured media without transfer functions by the discontinuous Galerkin and mixed methods. *SPE J* 2006b;11(3):341–52.
- [7] Moortgat J, Firoozabadi A. Higher-order compositional modeling of three-phase flow in 3D fractured porous media based on cross-flow equilibrium, *Journal of Computational Physics* 250 , 2013, 425–445.
- [8] Moortgat J, Firoozabadi A. Higher-order compositional modeling with Fickian diffusion in unstructured and anisotropic media, *Advances in Water Resources* 33, 2010, 951–968.
- [9] Coats KH. An equation of State compositional model, *SPEJ*, 1980, V.20, N5
- [10]Fussell LT, Fussel DD. An iterative technique for compositional reservoir models, *SPEJ*, August 1979, pp211-220
- [11] Chien MCH, Lee ST, Chen WH. A new fully implicit compositional simulator, *SPE 13385*, 1985

- 1
2
3
4 [12] Polívka O, Mikyška J. Compositional modeling in porous media using constant volume
5 flash and flux computation without the need for phase identification ,Journal of Computational
6 Physics 272, 2014, 149–169
7
8
9
- 10 [13] Younes A, Fahs M, Ahmed S. Solving density driven flow problems with efficient spatial
11 discretizations and higher-order time integration methods. Adv Water Res 32, 2009;340-352.
12
13
- 14 [14]Ackerer P, Younes A. Efficient approximations for the simulation of density driven flow in
15 porous media. Advances in Water Resources, 2008, 31 (1), pp. 15-27.
16
17
- 18 [15] Zidane A, Firoozabadi A. An efficient numerical model for multicomponent compressible
19 flow in fractured porous media; Advances in Water Resources 74, 2014, 127–147
20
21
- 22 [16] Zidane A, Zechner E, Huggenberger P, Younes A. On the effects of subsurface parameters
23 on evaporite dissolution (Switzerland). Journal of Contaminant Hydrology, 160, 2014, 42–52
24
25
- 26 [17] Zidane A, Younes A, Huggenberger P, Zechner E. The Henry semi- analytical solution for
27 saltwater intrusion with reduced dispersion, WaterResour. Res, doi: 10.1029/2011WR011157,
28 2012
29
30
- 31 [18] Raviart P, Thomas J. A mixed hybrid finite element method for the second order elliptic
32 problem. Lectures notes in mathematics, vol. 606. New York: Springer-Verlag; 1977
33
34
- 35 [19] Brezzi F, Fortin M. Mixed and hybrid finite element methods. Environmental engineering.
36 New York: Springer-Verlag; 1991.
37
38
- 39 [20] Chavent G, Jaffré J. Mathematical models and finite elements for reservoir simulation.
40 Studies in mathematics and its applications. North-Holland: Elsevier; 1986.
41
42
- 43 [21] Peng DY, Robinson DB. A New Two-Constant Equation of State. Industrial and
44 Engineering Chemistry: Fundamentals 15, 1976, pp. 59-64.
45
46
- 47 [22] Firoozabadi A. Thermodynamics of hydrocarbon reservoirs. USA: McGraw-Hill
48 Professional; 1999
49
50
- 51 [23] Stone H. Probability model for estimating three - phase relative permeability, J. Petrol.
52 Technol., 1970,214, 214-218.
53
54
- 55 [24] Stone H. Estimation of three - phase relative permeability and residual oil data, J. Can.
56 Petrol. Technol., 1973, 12, 53-61.
57
58
- 59
60
61
62
63
64
65

- 1
2
3
4 [25] Lohrenz J, Bray BG, Clark C R. Calculating viscosities of reservoir fluids from their
5 compositions, J. Petrol. Technol., 1964 ,16(10), 1171-1176.
6
7
8 [26] Acs G, Doleschall S, Farkas E. General purpose compositional model, SPE J. 25 (4), 1985,
9 543–553.
10
11
12 [27] Watts JW. A compositional formulation of the pressure and saturation equations, SPE
13 Reserv. Eng. 1 (3), 1986, 243–252.
14
15 [28] Darlow B, Ewing R, Wheeler M. Mixed finite element method for miscible displacement
16 problems in porous media, SPE J, 1984, 24, 391–398.
17
18 [29] Ewing RE, Lazarov RD, Wang J. Superconvergence of the velocity along the Gauss lines in
19 mixed finite element methods, SIAM J. Numer. Anal., 1991, 28, 1015–1029.
20
21
22 [30] Chavent G, Roberts JE. A unified physical presentation of mixed, mixed-hybrid finite
23 element method and standard finite difference approximations for the determination of velocities
24 in water flow problems. Adv.Water Resour, 1991. 14(6):329-348.
25
26
27 [31] Mose R, Siegel P, Ackerer P, Chavent G. Application of the mixed hybrid finiteelement
28 approximation in a groundwater-flow model – luxury or necessity. Water Resour Res
29 1994;30(11):3001–12.
30
31
32
33 [32] Brezzi F, Fortin M. Mixed and hybrid finite element methods. New York: Springer- Verlag;
34 1991.
35
36
37 [33] Chavent G, Cohen G, Jaffré J, Eymard R, Guéritot DR, Weill L. Discontinuous and mixed
38 finite elements for two-phase incompressible flow. SPE Reservoir Eng 1990;5(4):567–75.
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

Figures

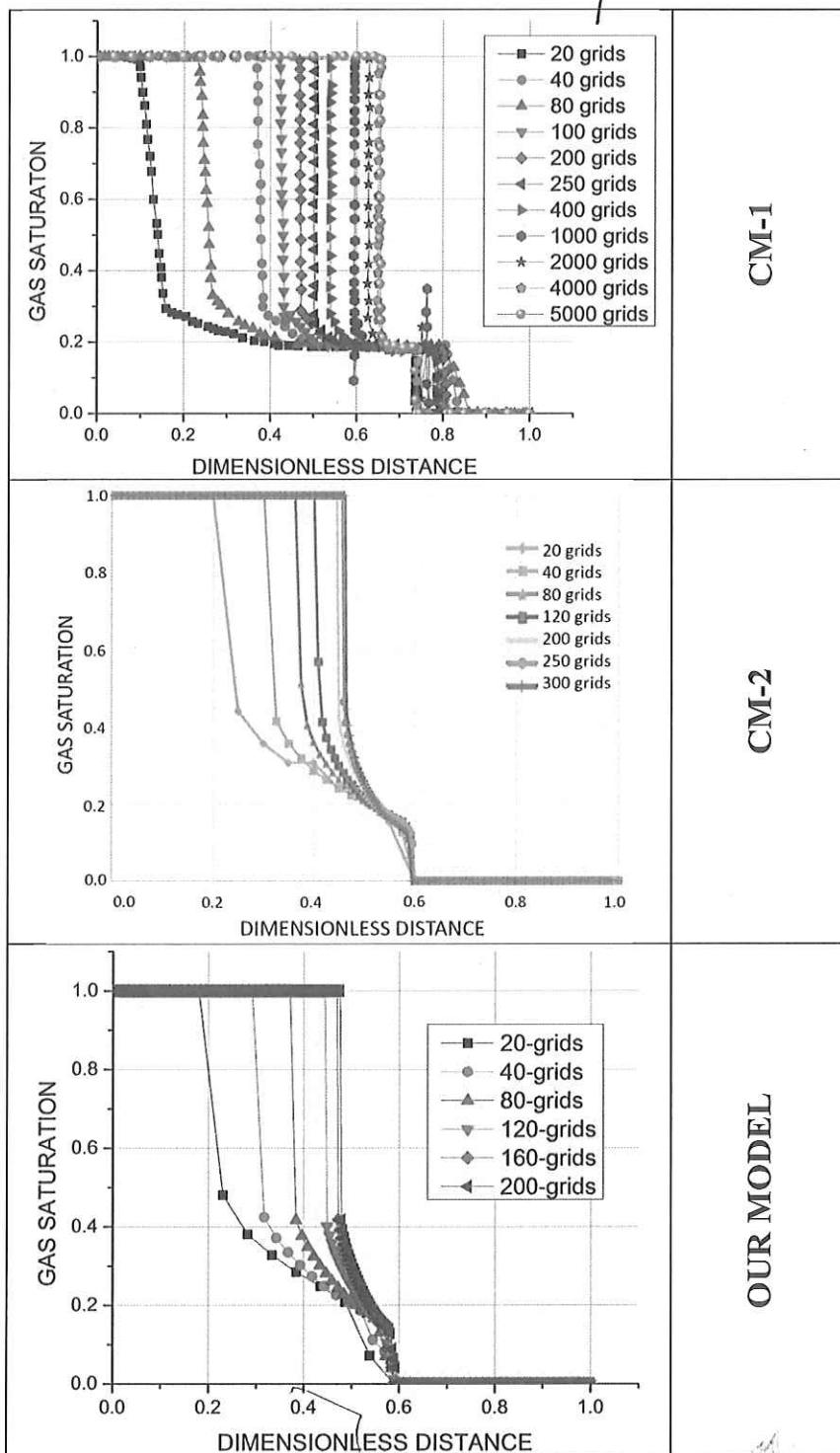


Fig.1: Comparison of our model to CM-1 and CM-2 with different mesh refinements; modified Coats example

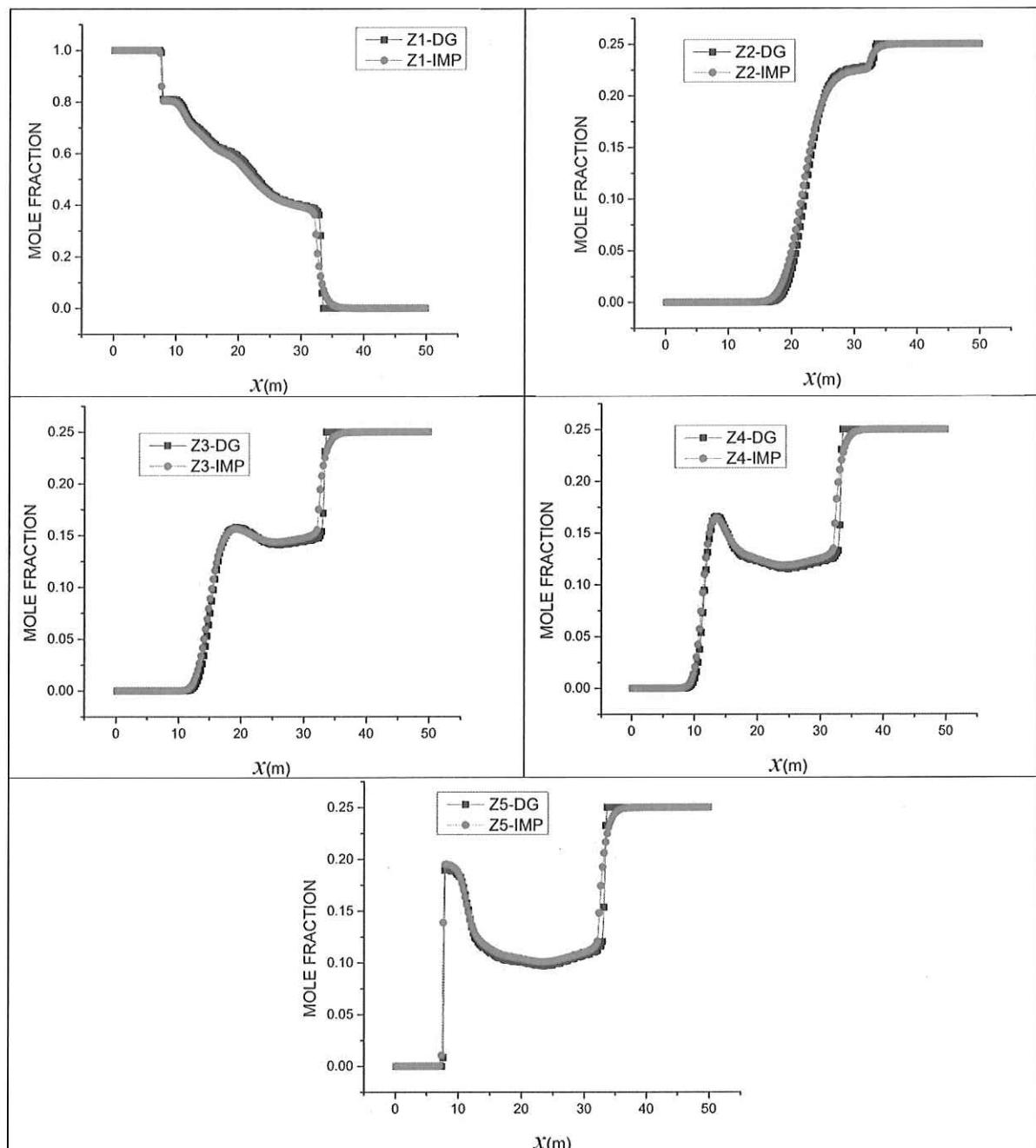


Fig.2: Overall mole fractions of the 5 components in our algorithm and the DG method at 60% PVI.

Example 1

200 grid blocks

→ Is this assumed to be validation
test case as well? Aren't there any
semi-analytical solution for comparison?
preliminary

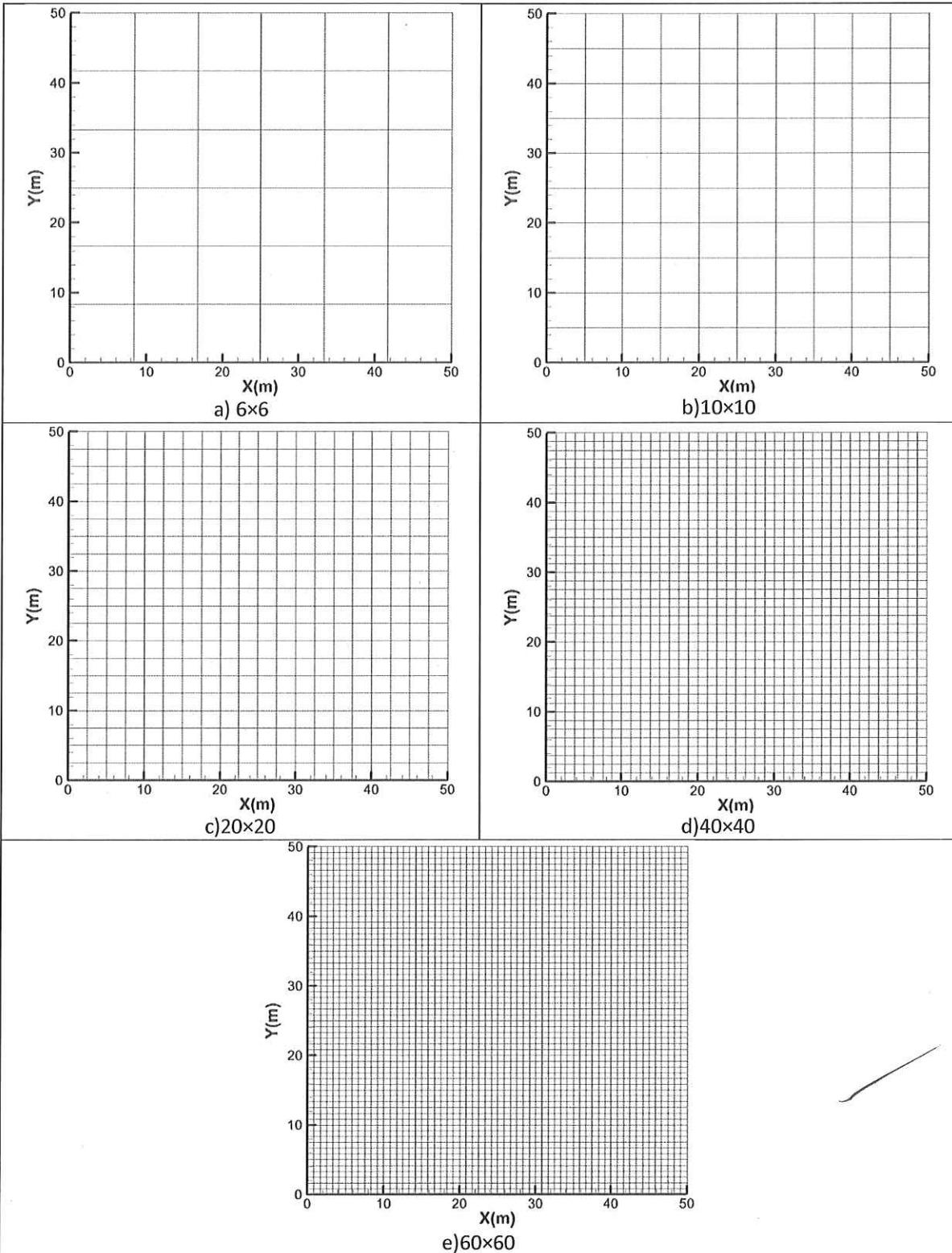


Fig.3: Mesh refinements used. Example 2

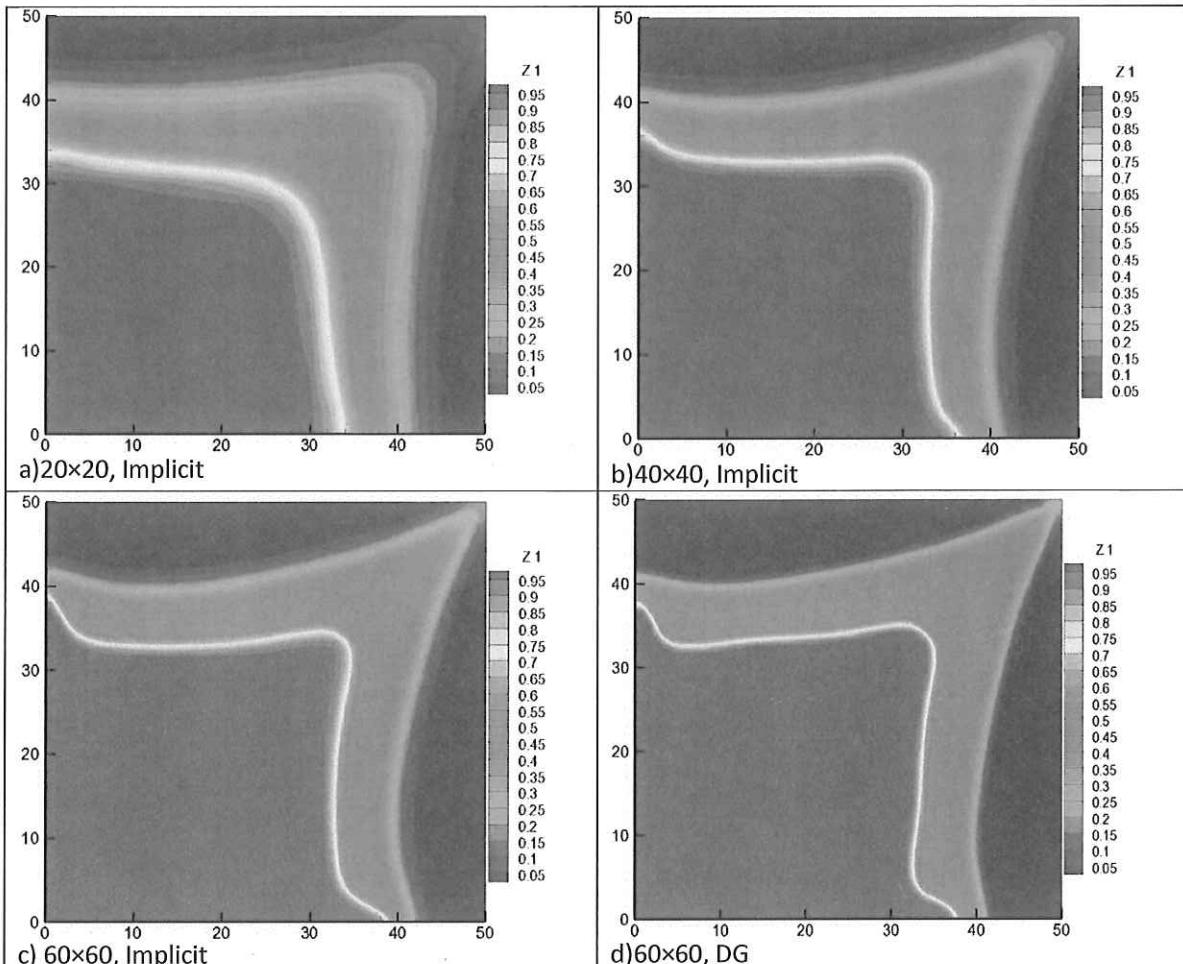


Fig.4: Methane overall mole fractions in different mesh refinements in the implicit scheme (a, b, c) and in the DG explicit scheme (d) at 85% PVI. Distances in meter; Example 2.

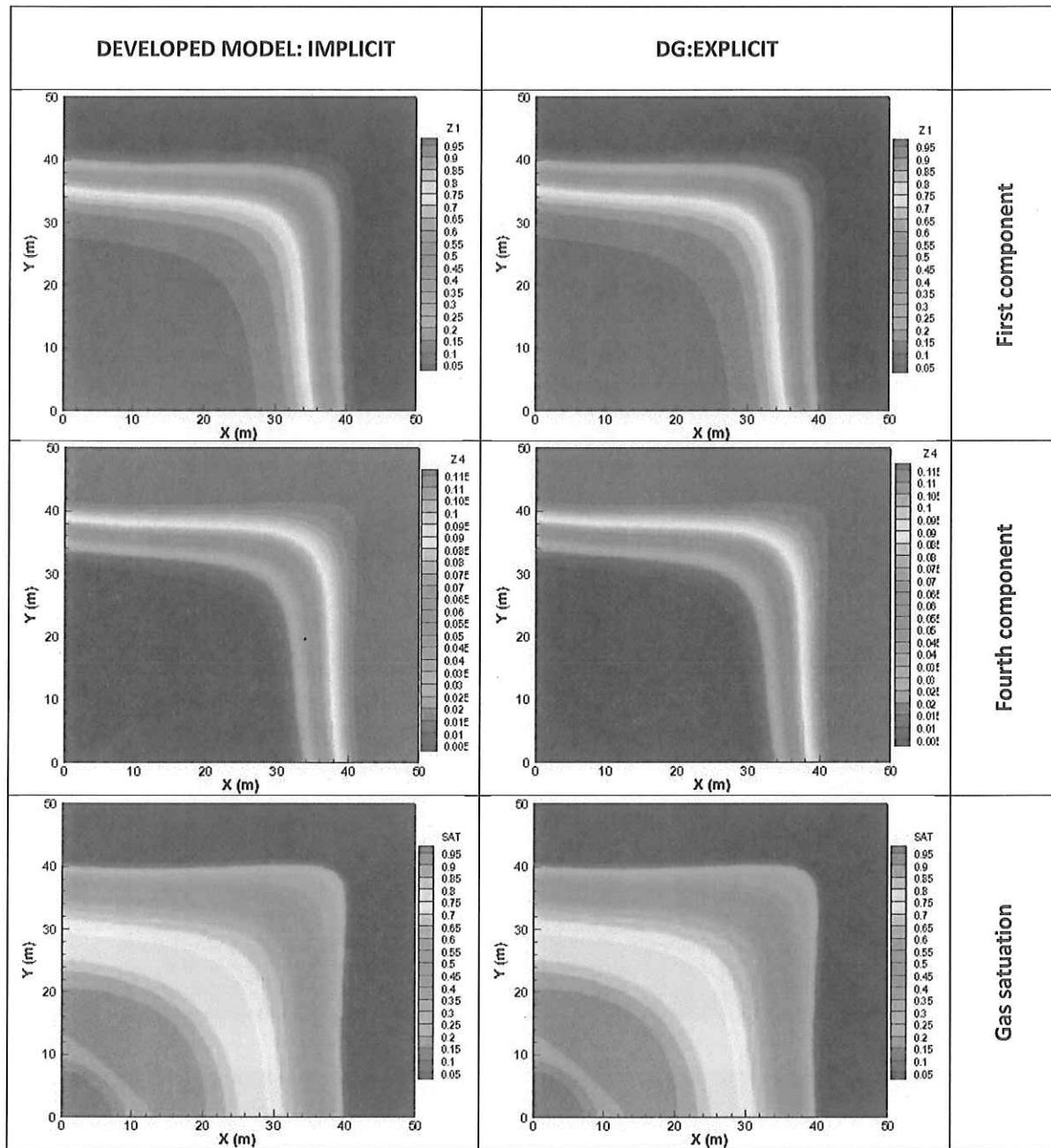


Fig.5: Overall mole fractions of components 1, 4, and the gas saturation profiles in compared to the DG results at 65% PVI. Distances in meter; Example 3.

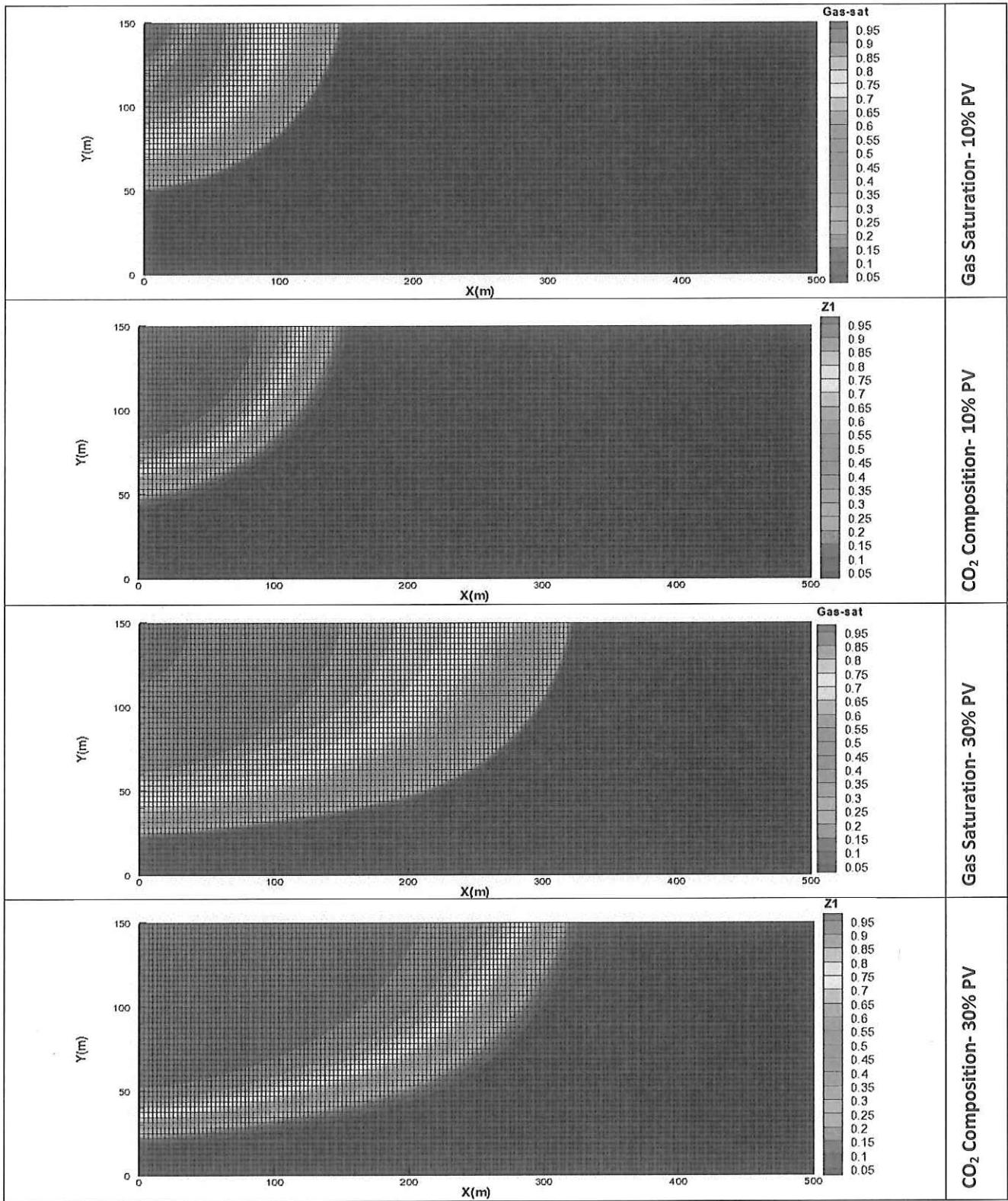


Fig.6: Gas saturation and overall mole fraction of CO₂ at different PVI. Example 4

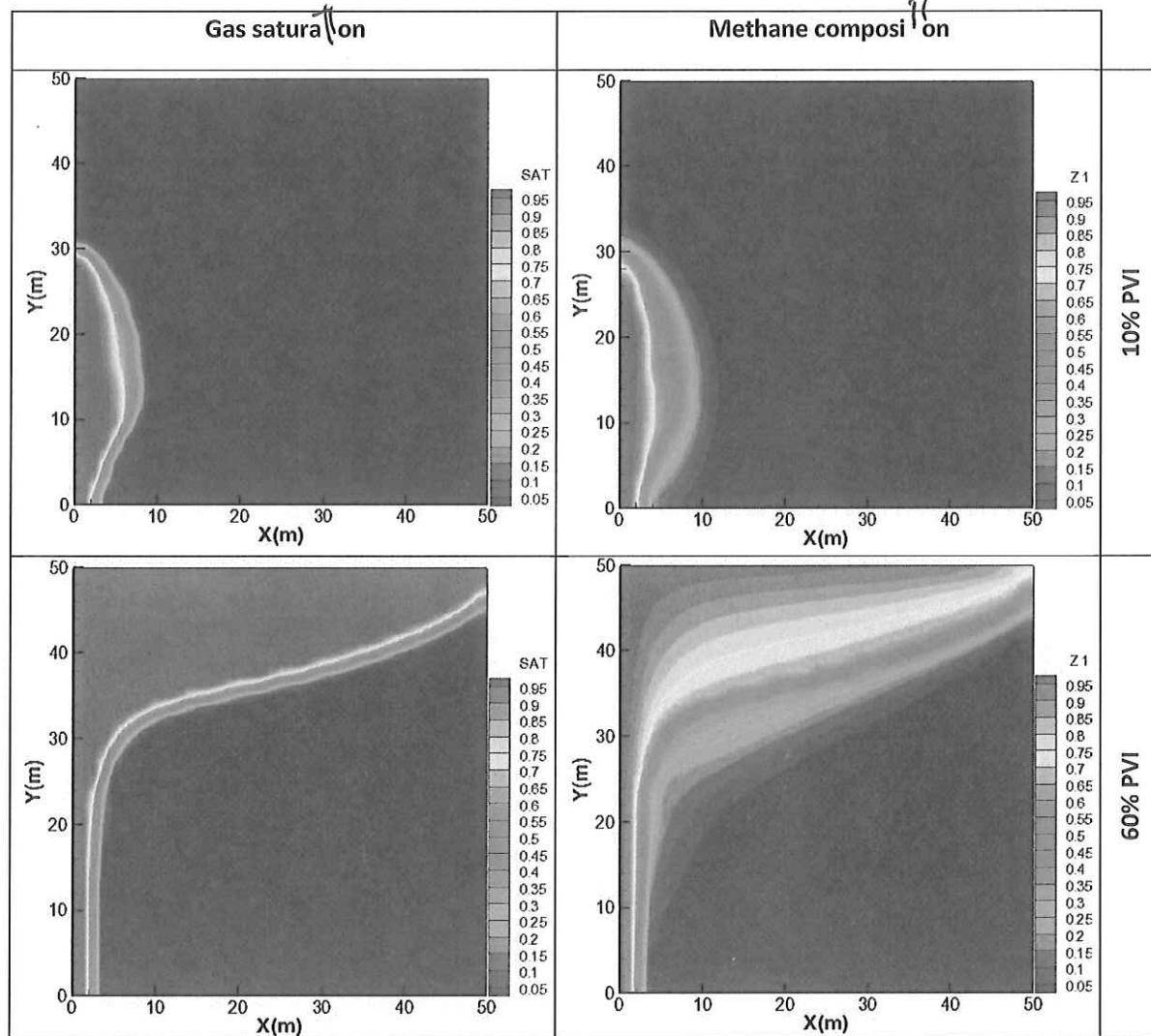


Fig. 7: Gas saturation (left) and composition of injected methane at 10% and 60% PVI; Example 5



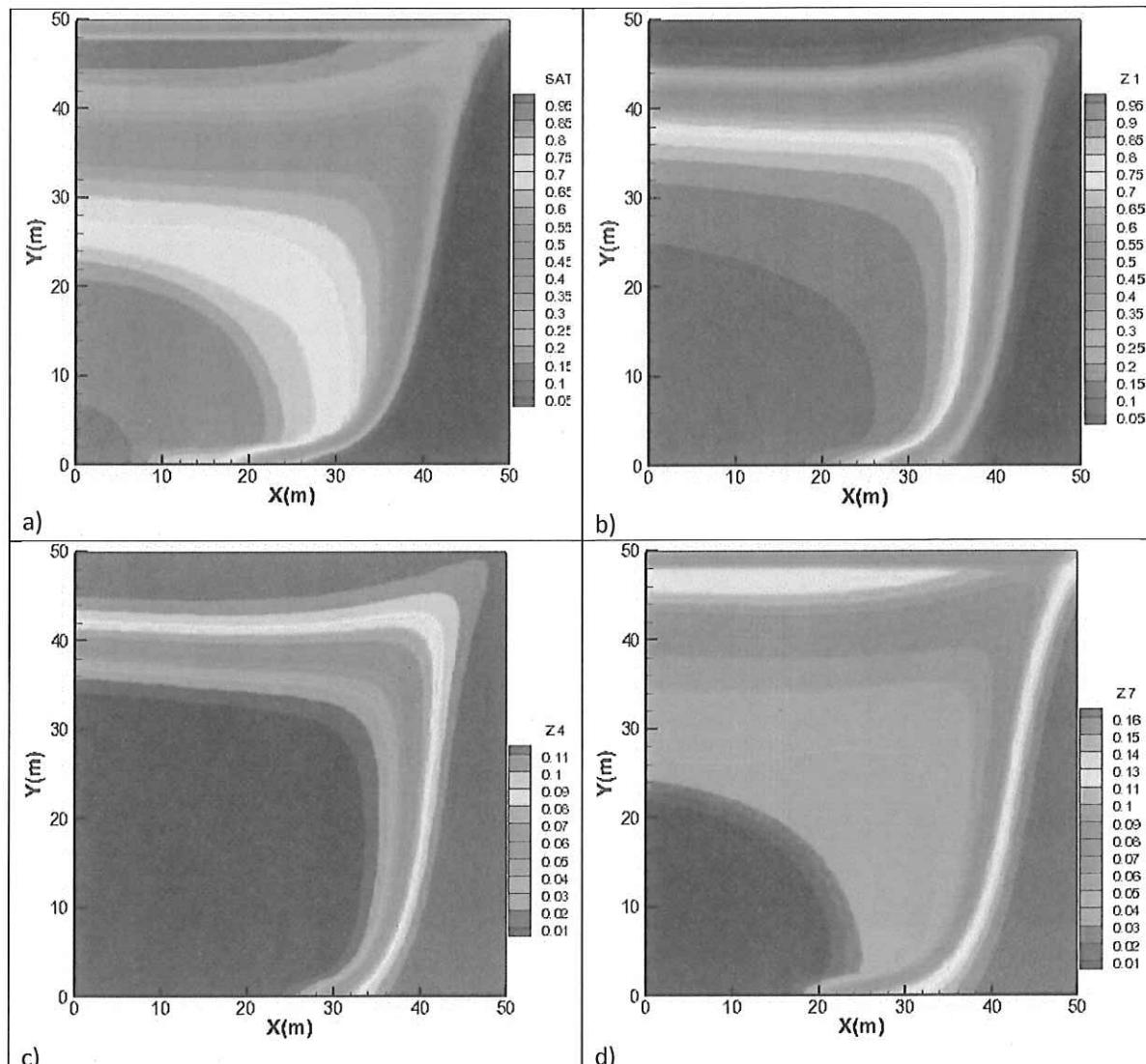


Fig. 8: Gas saturation (a) and compositions of the first (b), fourth (c) and seventh (d) components at 60% PVI; Example 6



1
2
3
4
5

Tables

Porosity	0.2
Permeability	10 mD
Injection rate	10 Pv/yr
Temperature	311 K
Pressure	69 bar
Injected fluid	1.0 C1
Initial fluid	0.25 C2/0.25 C2/0.25 C3/0.25C4
Relative permeability (oil and gas) coefficients	Endpoint=1, power=2
Residual saturations	0

18 Table 1: Relevant data of the domain and initial conditions ($C_1 / C_2 / C_3 / C_4 / C_5$ in mole fraction); Example
19
20
21

Nbr. of elements	DG	Imp.	Ra o
50	1.2	5.6	4.6
100	3.3	12.4	3.8
200	5.7	20.5	3.6
Cut-2	8.12	20.7	2.5
Cut-10	14.7	20.9	1.4
Cut-50	23.4	21.2	0.9

31 Table 2: CPU time (sec) for the implicit and the DG explicit scheme; Example 1
32
33

Porosity	0.2
Permeability	10 mD
Injection rate	31.025 Pv/yr
Temperature	311 K
Pressure	69 bar
Injected fluid	1.0 C1
Initial fluid	1.0 C2
Relative permeability (oil and gas) coefficients	Endpoint=1, power=2
Residual saturations	0

44 Table 3: Relevant data of the domain and initial conditions (C_1 / C_2 are in mole fraction); Example 2
45
46
47
48

Nbr. of elements	DG	Imp.
36	0.5	1
100	1.5	4
400	7	22
1600	32	51
3600	304	228

56 Table 4: CPU time (sec) for the implicit and the DG explicit scheme; Example 2
57
58
59
60
61
62
63
64
65

Scalability

Porosity	0.2
Permeability	10 mD
Injection rate (at atmospheric conditions)	97.3309 PV/yr
Temperature	403.15 K
Pressure	276 bar
Relative permeability (oil and gas) coefficients	Endpoint=1, power=2
Residual saturations	0

Table 5: Relevant data of the domain; Example 3

Component	Initial in the domain	Injected
CO2	0.0086	1.
N2	0.0028	0.
C1	0.4451	0.
C2-C3	0.1207	0.
C4-C5	0.0505	0.
C6-C10	0.1328	0.
C11-C24	0.1660	0.
C25+	0.0735	0.

Table 6: Initial and injected compositions (mole fractions); Example 3

Component	Acentric factor	Tc[K]	Pc[bar]	M.weight[g/mol]	Vc [m3/kg]
CO2	0.239	304.14	7.375E+01	44	0.00214
N2	0.039	126.21	3.39E+01	28	0.00321
C1	0.011	190.56	4.599E+01	16	0.00615
C2-C3	0.11783	327.81	4.654E+01	34.96	0.00474
C4-C5	0.21032	435.62	3.609E+01	62.98	0.00437
C6-C10	0.41752	574.42	2.504E+01	110.21	0.00425
C11-C24	0.66317	708.95	1.502E+01	211.91	0.00443
C25+	1.7276	891.47	0.76E+01	462.79	0.00417

Table 7: Relevant parameters for Peng-Robinson EOS; Example 3

	CO2	N2	C1	C2-C3	C4-C5	C6-C10	C11-C24	C25+
CO2	0.0							
N2	0.	0.0						
C1	0.15	0.1	0.0					
C2-C3	0.15	0.1	0.0346	0.0				
C4-C5	0.15	0.1	0.0392	0.0	0.0			
C6-C10	0.15	0.1	0.0469	0.0	0.0	0.0		
C11-C24	0.15	0.1	0.0635	0.0	0.0	0.0	0.0	
C25+	0.08	0.1	0.1052	0.0	0.0	0.0	0.0	0.0

Table 8: The symmetric binary interaction parameter matrix; Example 3

	CPU (sec)	Newt/ Δt
DG	842	-
Proposed model	963	2.88

Table 9: Performance of our model and DG with 3600 grid blocks; Example 3

Elements	CM-1			CM-2			Proposed model		
	# ΔT	#Newt	Newt/ Δt	# ΔT	#Newt	Newt./ Δt	# ΔT	#Newt	Newt./ Δt
400	68	192	2.82	26	67	2.56	71	186	2.62
1600	133	374	2.82	102	270	2.64	121	329	2.72
3600	149	433	2.91	102	291	2.85	162	466	2.88

Table 10: Performance of our model and the CM-1 and CM-2 commercial simulators; Example 3

	CPU (sec)	Newt/ Δt
DG	2465	-
Proposed model	1982	2.82

Table 11: Performance of our model and DG with 7680 grid blocks; Example 4

