A multilevel local grid refinement method to study the impact of polymer rheology on polymer injectivity of oil fields

Haishan Luo, Mojdeh Delshad, Zhitao Li, Amir Shahmoradi

Center for Petroleum and Geosystems Engineering, The University of Texas at Austin,
200 E. Dean Keeton St., Austin, TX 78712, USA

haishan.luo@utexas.edu delshad@mail.utexas.edu

Summary

Polymer injectivity is an important factor to evaluate the project economics of chemical flooding in oil fields. Nevertheless, as flow rate varies rapidly near injectors, polymer viscosity is significantly changed due to the polymer non-Newtonian rheological behavior. As a result, polymer viscosity near wellbore is difficult to calculate with practically acceptable gridblock size, leading to significant inaccuracy in polymer injectivity predictions. To reduce the impact of polymer rheology highly concerned in chemical EOR simulations, we develop an efficient multilevel local grid refinement (LGR) method that provides a higher resolution of the flows within the near-wellbore region. A novel numerical scheme is proposed to accurately solve the pressure equation and concentration equations on the multilevel grid for both homogeneous and heterogeneous reservoir cases. The block list and connections of the multilevel grid are generated via an efficient and extensible algorithm. Field case simulation results indicate that the proposed LGR is consistent with the analytical injectivity model and achieves the closest results with respect to the full grid refinement, which considerably improves the accuracy of solutions compared with the original grid. Besides polymer injectivity calculations, the LGR method is applicable for other problems in need of near-wellbore treatment, such as fractures near wells.

Keywords: polymer rheology, polymer injectivity, chemical EOR, local grid refinement, non-Newtonian flow

Introduction

Polymer flooding has become one of the most widely used enhanced oil recovery (EOR) methods because of its adaptability to a wide range of oil viscosity [1], relative simplicity for operations [2], and applicability to offshore [3], etc. For polymer flooding as well as most other chemical flooding processes such as surfactant-polymer flood, alkaline-surfactant-polymer flood, and alkaline-cosolvent-polymer flood, polymer injectivity is a key index for reservoir management, e.g., deciding the upper limit of polymer injection rate to optimize the project economics [4]. Factors affecting polymer injectivity include polymer degradation [4-5], induced fracturing near the injector [4,6], crosslinking polymer to form gel [7], and especially, polymer rheology [8-10]. Polymer solution is a non-Newtonian fluid whose viscosity is non-linearly related to the flow rate or the in-situ shear rate. For example, hydrolyzed polyacrylamide (HPAM)

solutions exhibit pseudoplastic behavior at low shear rates and dilatant behavior at high shear rates when flowing through porous media as shown in **Fig. 1** [8]. This leads to a complex relationship between the pressure drop and local velocity. Consequently, polymer injectivity is often erroneously calculated from the numerical simulations using gridblock size practical for full field simulations where flow rate decreases drastically from the wellbore [9,11]. It is crucial for the numerical simulator to capture near-wellbore polymer rheology more accurately, so as to improve the estimation of injection rate, shorten the project life, enhance the economics, and prevent or carefully design the injection induced fractures depending on the operators' decisions [12-13], etc.

The inaccuracy in calculated polymer injectivity mainly results from that flow rate is smeared within a coarse well gridblock. This is especially severe for common reservoir simulations in which the grid block size is used up to several dozens of feet while the wellbore radius is only about 0.5 ft. In-situ shear rates reach as high as 10^4 s^{-1} near wellbore and decrease sharply to about 1-10 s^{-1} within a well gridblock. To eliminate the grid effects, several empirical or analytical models were proposed based on effective properties of the well blocks. For instance, Sharma *et al.* [9] proposed to use an effective well radius to calculate the shear rate and match the polymer injectivity from very fine-grid simulation results; Li and Delshad [11] proposed an effective viscosity using mathematical integration of in-situ viscosity by assuming a radial velocity distribution within the well block. However, these approaches are not rigorous for other near-well effects except for polymer rheology, e.g., skin, polymer permeability reduction, injection induced fractures near the wellbore, which are often encountered during injection of polymer solutions. Therefore, in order to have a more accurate polymer injectivity that is adaptive to most reservoir conditions, it is necessary to refine the simulation grids. However, grid refinement for the whole reservoir model leads to excessive computational costs. It is thus important to develop a local grid refinement (LGR) technique (or similar unstructured gridding approaches), such as shown in **Fig. 2**, so that the grid refinement is only applied to the regions wherever needed.

Local grid refinement and similar unstructured gridding approaches have continuously played an important role in reservoir simulations. Successful applications can be found in water flood [14], miscible gas flood [15], steam flood [16-17], etc. LGR methods are classified into cell-based and patch-based approaches [18], while the former is more frequently used in simulations of porous media flow. Therefore, in the scope of this paper we only discuss the cell-based LGR approach. Forsyth and Sammon [19] developed an LGR algorithm with a rigorous analysis of discretization of flow equations upon the composite grid geometry. However, the accuracy of their numerical scheme is reported to be large because a direct subtraction of the pressures of two adjacent blocks is used to calculate the Darcy velocity across the block interface [20]. Nacul *et al*. [21] proposed an LGR technique using domain decomposition method, in which overlapping boundaries are used for the subdomains. Karimi-Fard and Durlofsky [22] presented an unstructured LGR method, and the well block is fully refined and solved in the fine scale to determine the effective properties that can be used for coarse-grid simulations over the reservoir domain.

In this paper, we propose an LGR method applied to Chemical EOR simulations, especially more accurate calculation of polymer rheological viscosity (polymer injectivity) under different reservoir conditions. Meanwhile, for a necessary complement to the scope of LGR approaches, this paper presents details on the numerical schemes to couple the mass conservation equations on the multilevel grid, as well as the indexing to the gridblocks and interfaces. In short, the proposed method includes the following features:

a) An efficient numerical scheme is developed to calculate the velocity and the mass flux across the block interface between different grid levels of the composite grid, which is also applied in the heterogeneous cases.

b) The algorithm on how to index the gridblock list and gridblock connections under the LGR composite grid is presented in details. The numerical computations under the LGR grid structure can benefit from this data management, which may also be extended to the classical unstructured grid and provide a good basis for the successive simulator development.

This paper is organized as follows. In the next two sections, we will give the mass balance equations and the chemical flood simulation models. The subsequent section presents the details of the proposed efficient LGR algorithm. We will then test several examples simulated with different levels of refinement to demonstrate the improvement in numerical results. The LGR simulations are also compared to those using the analytical injectivity model proposed by Li and Delshad [11].

Mathematical model

In this section, we briefly present the mathematical framework of the University of Texas Chemical Flooding simulator, UTCHEM (Delshad *et al*. [23]) and formulations for modeling polymer rheology and injectivity. UTCHEM is a three-dimensional multi-phase multi-component compositional simulator with the capability of modeling geochemical reactions, complex phase behavior, etc. The governing balance equations include: (1) the mass conservation equation for each species; (2) the pressure equation obtained by summing up all mass conservation equations for all volume-occupying species; (3) the energy conservation equation which will not be discussed here.

Mass conservation equations

We write the mass conservation equation for each component κ,

$$\frac{\partial}{\partial t}\left(\phi \tilde{C}_\kappa \rho_\kappa\right) + \vec{\nabla} \cdot \left[\sum_{l=1}^{n_p} \rho_\kappa \left(C_{\kappa l}\vec{u}_l - \vec{\tilde{D}}_{\kappa l}\right)\right] = R_\kappa \tag{1}$$

where ϕ is the porosity, ρ_κ is the density of component κ, $C_{\kappa l}$ is the concentration of component κ in phase l, and $\vec{u}_l$ is the Darcy flux of phase l which is calculated using Darcy's Law:

$$\vec{u}_l = -\frac{k_{rl}\vec{\vec{k}}}{\mu_l} \cdot \left(\vec{\nabla} P_l - \gamma_l \vec{\nabla} h\right) \tag{2}$$

where $\vec{\vec{k}}$ is the intrinsic permeability tensor, k_{rl} is the relative permeability, μ_l is the viscosity, γ_l is the specific weight of phase *l*, and *h* represents the vertical depth.

$\tilde{C}_\kappa$ is the overall concentration of component κ in the mobile and stationary phases expressed as

$$\tilde{C}_\kappa = \left(1 - \sum_{\kappa=1}^{n_{cv}} \hat{C}_\kappa\right) \sum_{l=1}^{n_p} S_l C_{\kappa l} + \hat{C}_\kappa \qquad \text{for } \kappa = 1, \ldots, n_c \tag{3}$$

where S_l is the saturation of phase *l*, n_{cv} is the total number of volume occupying components, and $\hat{C}_\kappa$ is the adsorbed concentration of component κ. In UTCHEM, the liquid phase *l* includes aqueous (*l*=1), oleic (*l*=2), and microemulsion (*l*=3).

$\vec{\tilde{D}}_{\kappa l}$ is the dispersive flux which is assumed to have a Fickian form:

$$\vec{\tilde{D}}_{\kappa l} = \phi S_l \vec{\vec{K}}_{\kappa l} \cdot \vec{\nabla} C_{\kappa l} \tag{4}$$

where the dispersion tensor $\vec{\vec{K}}_{\kappa l}$ is calculated as:

$$\vec{\vec{K}}_{\kappa l ij} = \frac{D_{\kappa l}}{\tau}\delta_{ij} + \frac{\alpha_{Tl}}{\phi S_l}|\vec{u}_l|\delta_{ij} + \frac{(\alpha_{Ll} - \alpha_{Tl})}{\phi S_l}\frac{u_{li}u_{lj}}{|\vec{u}_l|} \tag{5}$$

where $D_{\kappa l}$ is the molecular diffusion, τ is the tortuosity factor of the porous media, α_{Ll} and α_{Tl} are phase *l* longitudinal and transverse dispersivities, and δ_{ij} is the Kronecker delta function.

R_κ is the source term which is a combination of all rate terms for a particular component κ. It may be expressed as

$$R_\kappa = \phi \sum_{l=1}^{n_p} S_l r_{\kappa l} + (1 - \phi) r_{\kappa s} + Q_\kappa \tag{6}$$

where $r_{\kappa l}$ and $r_{\kappa s}$ are the reaction rates for component κ in phase *l* and solid phase *s* respectively, and Q_κ is the injection/production rate for component κ per bulk volume.

Pressure equation

Summing the mass balance equations from Eq. (1) over all the volume-occupying components, substituting Eq. (2) and using aqueous phase pressure as a reference pressure, we obtain the pressure equation:

$$\phi C_t \frac{\partial P_1}{\partial t} + \vec{\nabla} \cdot \vec{\vec{k}} \cdot \left(\sum_{l=1}^{n_p} \lambda_{rlc}\right) \vec{\nabla} P_1 = \vec{\nabla} \cdot \left(\sum_{l=1}^{n_p} \vec{\vec{k}} \cdot \lambda_{rlc} \left(\vec{\nabla} P_{cl1} - \gamma_l \vec{\nabla} h\right)\right) + \sum_{l=1}^{n_{cv}} Q_\kappa \tag{7}$$

where P_{cl1} is the capillary pressure between phase *l* and phase 1 (the aqueous phase), λ_{rlc} is the relative mobility expressed as

$$\lambda_{rlc} = \frac{k_{rl}}{\mu_l} \sum_{l=1}^{n_{cv}} \rho_\kappa \, C_{\kappa l} \tag{8}$$

and C_t represents the total compressibility which is the volume-weighted sum of the rock matrix (C_r) and component compressibilities (C_κ^0):

$$C_t = C_r + \sum_{l=1}^{n_{cv}} C_\kappa^0 \, \tilde{C}_\kappa \tag{9}$$

where $\phi = \phi_R[1 + C_r(P_R - P_{R0})]$.

Rheological viscosity of polymer solution

Non-Newtonian polymer rheology (shear-thinning behavior) is modeled using Meter's equation [24]:

$$\mu_{app} = \mu_\infty + \frac{\mu_p^0 - \mu_\infty}{1 + \left(\frac{\dot{\gamma}_{eff}}{\dot{\gamma}_{1/2}}\right)^{P_\alpha - 1}} \tag{10}$$

where μ_{app} is the apparent viscosity of polymer solution; μ_∞ is the polymer solution viscosity at infinite shear rate which is assumed to be brine viscosity; $\dot{\gamma}_{1/2}$ is the shear rate at which the apparent viscosity is the average of μ_∞ and μ_p^0; P_α is a fitting parameter. For synthetic polymer, e.g. HPAM, polymer solutions are shear-thinning at intermediate shear rates and shear-thickening (dilatant) behavior at high rates. To remediate the deficiency of Meter's equation, Delshad *et al*. [8] developed a comprehensive polymer viscosity model which covers the whole shear-rate regime. The apparent viscosity consists of two parts:

$$\mu_{app} = \mu_{sh} + \mu_{el} \tag{11}$$

where the shear-thinning model uses Carreau [25]:

$$\mu_{sh} = \mu_\infty + \left(\mu_p^0 - \mu_\infty\right)\left[1 + \left(\lambda_1 \dot{\gamma}_{eff}\right)^2\right]^{(n_1 - 1)/2} \tag{12}$$

and the shear-thickening model is:

$$\mu_{el} = \mu_{max}\left\{1 - exp\left[-\left(\lambda_2 \tau \dot{\gamma}_{eff}\right)^{n_2 - 1}\right] \right\} \tag{13}$$

where a_1, a_2 and τ are all fitting model parameters obtained by matching experimental data; $\mu_{\max}$ is given as

$$\mu_{max} = \mu_b\left(AP_{11} + AP_{22} ln C_p\right) C_{SEP}^{S_p} \tag{14}$$

where $C_{SEP}^{S_p}$ is the polymer viscosity dependence on salinity and hardness; AP_{11} and AP_{22} are fitting parameters. When AP_{11} and AP_{22} are zero, the comprehensive polymer viscosity model reduces to the Carreau model.

The effective shear rate ($\dot{\gamma}_{eff}$) correlates viscosity measured in a viscometer to an apparent in-situ viscosity in porous media and is defined using a capillary bundle model [26] as:

$$\dot{\gamma}_{eff} = C\left(\frac{3n + 1}{4n}\right)^{\frac{n}{n-1}} \frac{4|\vec{u}_w|}{\sqrt{8\bar{k} k_{rw} \phi S_w}} \tag{15}$$

where n is the slope of the linear portion of bulk polymer viscosity vs. shear rate and plotted on a log-log scale (bulk power law index); $\vec{u}_w$ is the Darcy flux of the aqueous polymer solution; $\overline{k}$ is the average permeability; k_{rw} is the aqueous phase relative permeability; S_w is the aqueous phase saturation; ϕ is the

porosity. *C* is a shear correction factor used to explain the deviation of porous medium from ideal capillary bundle model [27-28] and should be a function of permeability, porosity, and polymer molecule properties.

Analytical polymer injectivity model

According to Peaceman's well model [29], the relationship between injection rate Q_{inj} and the pressure difference between injector and well block $(P_{inj} - P_{wb})$ can be expressed by:

$$Q_{inj} = I\left(P_{inj} - P_{wb}\right) \tag{16}$$

where I is well injectivity:

$$I = \frac{2\pi h\sqrt{k_x k_y}}{ln\left(\frac{r_o}{r_w}\right) + s} \sum_{l=1,np} \frac{k_{rl,wb}}{\mu_{l,wb}} \tag{17}$$

where h represents the thickness of the wellblock, r_o represents the Peaceman equivalent radius, r_w is the well radius, s is well skin, and $k_{rl,wb}$ and $\mu_{l,wb}$ are the relative permeability and viscosity of phase l of wellblock, respectively.

In traditional simulation models, the polymer solution viscosity ($\mu_{w,wb}$) of wellblock is directly calculated from Eq. (10) or Eq. (11), using the averaged shear rate of the block. Thus, the shear rate is smeared and consequently gives significant error in well injectivity depending on the flow rate and the size of the gridblocks.

To overcome this limitation, Li and Delshad [11] proposed a rigorous analytical injectivity model to calculate the equivalent apparent viscosity of polymer solution based on the assumption that after conversion of coordinates to account for the effects of non-square grids and anisotropic permeability, radial flow dominates the near-wellbore region, i.e.,

$$u(\bar{r}) = \frac{Q_{inj}}{2\pi h\bar{r}} \tag{18}$$

where $\bar{r}$ is the distance from the wellbore after conversion of coordinates.

It can then be derived that the equivalent apparent viscosity of polymer solution has the following expression:

$$\bar{\mu}_{w,wb} = \frac{\int_{r_w}^{\bar{r}_o} \mu_{app}(r)\frac{dr}{r}}{ln\left(\frac{\bar{r}_o}{r_w}\right)} \tag{19}$$

in which $\mu_{app}(r)$ adopts the form of Eq. (10) or Eq. (11) using the shear rate calculated from the local velocity expressed by Eq. (18). For the detailed derivation, one can refer to Li and Delshad [11].

UTCHEM flowchart

UTCHEM uses FVM method (finite volume method) and IMPEC approach (implicit pressure explicit concentration). The flowchart of the simulator is in **Fig. 3**.

In each time step, the simulator first solves the pressure equation (Eq. (7)) implicitly, and then solves the concentration equations for each component (Eq. (1)) explicitly using a third-order scheme with a flux

limiter. After that, phase behavior calculations will be performed if surfactant is present. In the last step, properties are updated by taking into account the water reactions and polymer adsorption, as well as other chemical and physical changes. All the newly updated variables and properties will be provided for the initial values of the next time step. This continues until it reaches the final time.

Local grid refinement algorithm

Current form of the UTCHEM simulator is developed based on structured grid, and the use of local grid refinement (LGR) will transform the grid from structured to unstructured as the connections between blocks are no longer regular. This makes it necessary to change the original data structure and computational model for solving the pressure equation and concentration equations.

To adapt the original computational structure to LGR and to maintain a good memory management, we design a new flowchart for UTCHEM in **Fig. 4**. Compared to the original flowchart shown in Fig. 3, this new algorithm automatically generates the block list and connections according to the well location and refinement levels after the initialization step. An LGR module is also used to replace the original modules for solving the pressure equation and concentration equations. The other parts remain unchanged because those calculations are block-based and not relevant to the grid structure. Features of the LGR algorithm will be presented in the following two subsections.

Block list and connections

Computations with unstructured grid and LGR are normally based on a block list which gives the indices of gridblocks or cell numbering, and connections which give the indices of block interfaces linking to a pair of adjacent blocks. Considering LGR has a special grid topology composed of rectangular blocks at different levels, we developed a fast algorithm to generate the block list and connections illustrated in **Fig. 5** with a 2D example case. The domain was originally covered by 2 coarse blocks; and then it is refined to 8 blocks. The numbering of the block list is advanced by each coarse block. For each coarse block, the numbering starts first along the x-direction and then the y-direction.

Different from the common unstructured grid, the connections in our LGR algorithm are divided into two types: x-direction connections (marked in red in **Fig. 5**) and y-direction connections (marked in blue in **Fig. 5**). A summary of the block list and connections is given in **Table 1**.

The indexing of block list and connections facilitates the search for neighboring blocks, and the assignment of properties evaluated at the block interfaces, such as transmissibility, velocity, mass flux, etc., using the list of connections.

Coupling of governing equations

As IMPEC scheme is used, pressure equation and concentration equations are solved separately during the computations.

Coupling of pressure equation

The pressure equation needs to be solved implicitly and the calculation of the velocities across the block interfaces of the composite grid is a common issue. Let us take the block connection in **Fig. 6** as an example. The lengths of the coarse block are Δx and Δy and the lengths of the fine blocks are half. For the sake of simplicity to describe our approach, we assume in **Fig. 6** isotropic permeabilities without gravity effect, and define λ as the total fluid mobility, i.e., $\lambda = k_{abs}\sum_{l=1,N_p}\frac{k_{rl}^{ups}}{\mu_l}$, where k_{rl}^{ups} is the relative permeability of phase *l* defined on the block interface with an upstream scheme. The upstream scheme to obtain k_{rl}^{ups} is the same as that to obtain the upstream concentration, C_κ^{ups}, which we will explain in the next subsection.

To calculate the fluxes across the block interfaces, such as $u_{(m)}$ and $u_{(n)}$, early approach [19] used the pressures at the block centers to obtain the pressure difference in Dacry's Law. However, It was pointed out that it generated high truncations [20]. Gerritsen and Lambers [30] proposed in their anisotropic grid adaptivity method to use bilinear interpolation to get the pressure of the auxiliary points (such as $P_{(i1)}$ and $P_{(i2)}$ in Fig. 6) for calculating the interfacial velocity using Darcy's Law. This method proves to be second-order accurate when solving the pressure equation for homogeneous cases. However, the accuracy of bilinear interpolation is insufficient for heterogeneous cases because the discontinuity of the pressure gradient across the block interface is not taken into account. Actually, handling heterogeneity is an important factor to weigh up the reliability of the numerical scheme. As far as we know, there has not been a rigorous numerical scheme in the scope of the cell-centered finite volume method for accurately coupling the pressure equations with the LGR composite grid.

In **Appendix A**, we derive a simple but efficient numerical scheme to couple the pressure equations for the blocks with different grid levels. The expression of the velocities across the interface is as follows:

$$\begin{cases} u_{(m)} = -T_{(m)}\left[\left(\dfrac{P_{(j)}+P_{(k)}}{2}-P_{(i)}\right)\dfrac{u^0_{(mn)}}{u^0_{(mn)}+u^0_{(jk)}}+\left(\dfrac{\lambda_{(j)}P_{(j)}+\lambda_{(k)}P_{(k)}}{(\lambda_{(j)}+\lambda_{(k)})}-P_{(i)}\right)\dfrac{u^0_{(jk)}}{u^0_{(mn)}+u^0_{(jk)}}\right] \\ u_{(n)} = -T_{(n)}\left[\left(\dfrac{P_{(j)}+P_{(k)}}{2}-P_{(i)}\right)\dfrac{u^0_{(mn)}}{u^0_{(mn)}+u^0_{(jk)}}+\left(\dfrac{\lambda_{(j)}P_{(j)}+\lambda_{(k)}P_{(k)}}{(\lambda_{(j)}+\lambda_{(k)})}-P_{(i)}\right)\dfrac{u^0_{(jk)}}{u^0_{(mn)}+u^0_{(jk)}}\right] \end{cases} \quad (20)$$

where the meanings of $T_{(m)}$, $T_{(n)}$, $u^0_{(mn)}$, and $u^0_{(jk)}$ are given in **Appendix A**.

This numerical scheme has the following advantages:

- It has a simple form as it does not require any additional information from other blocks except for the current three connected blocks.
- It is easy to use as it does not need any interpolation/extrapolation.
- It is based on the continuity of mass flux across the interfaces and it is rigorously self-consistent under the homogeneous condition or the condition that fine-block permeabilities are identical. The latter condition is often met for most LGR applications when the permeabilities of the refined blocks are directly from the coarse block permeability.

Coupling of mass conservation equations

To guarantee the numerical stability, upstream schemes are mainly used to solve mass conservation equations. In the UTCHEM simulator, there are several options for the upstream schemes: first-order upstream scheme, second-order upstream scheme, and a third-order upstream scheme named as Leonard's scheme [31]. Because higher-order upstream schemes are more accurate to integrate the concentration equations, we only discuss about how to couple the concentration equations using the Leonard's scheme in this paper. Under the structured grid in **Fig. 7**, the Leonard's scheme to calculate the mass flux across the interface at $i+\frac{1}{2}$ is expressed by

$$f_{\left(i+\frac{1}{2}\right)}=\begin{cases} u_{\left(i+\frac{1}{2}\right)}\left[C_{(i)}-\dfrac{\Delta x_{(i)}(C_{(i-1)}-C_{(i)})}{3\left(\Delta x_{(i)}+\Delta x_{(i-1)}\right)}-\dfrac{2\Delta x_{(i)}(C_{(i)}-C_{(i+1)})}{3\left(\Delta x_{(i)}+\Delta x_{(i+1)}\right)}\right] & if\ u_{\left(i+\frac{1}{2}\right)}>0 \\ u_{\left(i+\frac{1}{2}\right)}\left[C_{(i+1)}-\dfrac{\Delta x_{(i+1)}(C_{(i)}-C_{(i+1)})}{3\left(\Delta x_{(i)}+\Delta x_{(i+1)}\right)}-\dfrac{2\Delta x_{(i+1)}(C_{(i+1)}-C_{(i+2)})}{3\left(\Delta x_{(i+2)}+\Delta x_{(i+1)}\right)}\right] & if\ u_{\left(i+\frac{1}{2}\right)}<0 \end{cases} \tag{21}$$

where *C* represents the component concentration.

For the LGR grid, we take the block combination in **Fig. 8** as one example. In this case, because the block center points are not in the same line, we utilize bilinear interpolation to obtain the concentration values C' at the auxiliary points, e.g., *i1*, *i2*. After that, we extend Leonard's scheme to this case:

$$f_{(m)}=\begin{cases} u_{(m)}\left[C'_{(i1)}-\dfrac{\Delta x_{(i)}(C_{(i1-)}-C'_{(i1)})}{3\left(\Delta x_{(i)}+\Delta x_{(i-)}\right)}-\dfrac{2\Delta x_{(i)}(C'_{(i1)}-C_{(j)})}{3\left(\Delta x_{(i)}+\Delta x_{(j)}\right)}\right] & if\ u_{(m)}>0 \\ u_{(m)}\left[C_{(j)}-\dfrac{\Delta x_{(j)}(C'_{(i1)}-C_{(j)})}{3\left(\Delta x_{(i)}+\Delta x_{(j)}\right)}-\dfrac{2\Delta x_{(j)}(C_{(j)}-C_{(j+)})}{3\left(\Delta x_{(j+)}+\Delta x_{(j)}\right)}\right] & if\ u_{(m)}<0 \end{cases} \tag{22}$$

$$f_{(n)}=\begin{cases} u_{(n)}\left[C'_{(i2)}-\dfrac{\Delta x_{(i)}(C_{(i1-)}-C'_{(i2)})}{3\left(\Delta x_{(i)}+\Delta x_{(i-)}\right)}-\dfrac{2\Delta x_{(i)}(C'_{(i2)}-C_{(k)})}{3\left(\Delta x_{(i)}+\Delta x_{(k)}\right)}\right] & if\ u_{(n)}>0 \\ u_{(n)}\left[C_{(k)}-\dfrac{\Delta x_{(k)}(C'_{(i2)}-C_{(k)})}{3\left(\Delta x_{(i)}+\Delta x_{(k)}\right)}-\dfrac{2\Delta x_{(j)}(C_{(k)}-C_{(k+)})}{3\left(\Delta x_{(k+)}+\Delta x_{(k)}\right)}\right] & if\ u_{(n)}<0 \end{cases} \tag{23}$$

where $f_{(m)}$ and $f_{(n)}$ are mass fluxes across the interfaces *m* and *n*.

Case study

To validate the LGR method proposed in this paper, we tested four simulation examples. These examples show the comparisons of the simulation results using LGR method with those using analytical polymer well model, and full grid refinement (FGR) where the whole model has the smallest grid size of the LGR.

Case 1: Polymer flooding in a 2D homogeneous reservoir

We start with a base case for polymer flooding. The polymer solution is assumed shear-thinning. Adsorption and permeability reduction are also considered. The reservoir and well descriptions are given in **Table 2**. The basic grid used for simulation is 15×15×1, and the grid with 4-level refinement is plot in **Fig. 9**. It shows that the well block is refined to 8×8 finest blocks and several transitional blocks connect the original coarse blocks and finest blocks.

In this case, the injection rate is constant so that the injector pressure varies with different polymer viscosities and thus well injectivities. **Fig. 10** shows the comparison of the injector wellbore pressures using different grids or well models. For water flooding period, there are not obvious differences of the injector wellbore pressures among the different simulations. However, when shifted to polymer flooding, it is observed that the wellbore injector pressures are remarkably differentiated using different grids or well models because of polymer rheology. It also shows that using the original grid leads to the highest injector pressure. The reason is that averaged viscosity within the well block area is artificially amplified due to a lower smeared flux rate caused by the coarse block size, which leads to an over-prediction of injection pressure that triggers the pressure limit, for example, in the case that the facility's injection pressure limit is 6000 Psi. In contrary, with grid refinement around the well block, the injector pressure gradually decreases and calculates a much "safer" injector pressure. Also, the variation of the injector pressure shrinks with the increase of the level of grid refinement, showing a convergence trend. Because the simulation results using 3-level and 4-level LGR are relatively close and further refinement may lead to excessive computational times, we regard the simulation result of 4-level LGR as the reference result to evaluate other simulations. Of course, it should be more precise to use the fully refined grid as the reference. Nevertheless, **Fig. 10** shows that 4-level FGR gives very similar injection pressure as the 4-level LGR. We also use the analytical injectivity model [11] and we observe that the simulated injector pressure is between the results of 3-level LGR and 4-level LGR. This result is more accurate than the case without grid refinement, and shows the consistency with the LGR results.

In **Table 3,** the CPU time using the FGR is almost 60 times more than the LGR, showing that the LGR is very helpful to improve the computational efficiency for the polymer flooding case while capturing accurate injectivity for non-Newtonian polymer flow.

Case 2: Polymer flooding in a 2D reservoir with a fracture near the injector

This case is aimed to analyze the behavior of the LGR in the presence of a planar fracture near the injector, which may often be encountered during the polymer injection projects [32-33]. We assume that the reservoir has the same condition as case 1 except that there is a fracture initiated from the injector well block. We show in **Fig. 11** a 4-level LGR grid with the permeability field (the permeability of the fracture is assumed to be 10 Darcy). In this case, it is obviously not proper to use the coarse grid as well as the analytical injectivity model, which could not describe the flow in the fractures. Therefore, it is obligatory to refine the gridblocks.

Fig. 12 shows the simulation results under the three conditions: LGR without fracture, LGR with fractures, and FGR with fractures. The LGR with fractures leads to a significantly smaller injector pressure compared to the LGR without fracture, which shows the importance of accounting for fractures near the injectors. We also show that the pressure curve of FGR is close to that of LGR, which proves the agreement of the results between using the two types of grids.

Case 3: Polymer flooding in a 3D heterogeneous reservoir

We study a polymer flooding field case. The polymer solution is assumed shear-thinning. Adsorption and permeability reduction are also considered. The reservoir and well descriptions are given in **Table 4**. The permeability field and well locations are exhibited in **Fig. 13**. The relevant grid with 4-level LGR is in **Fig. 14**. Some wells are deviated so that the local grid refinement is expanded in the x-y plane.

In this case, the injection pressure is fixed so that the injection rate varies with well injectivity. **Fig. 15** shows the simulated overall injection rate using the original grid, 4-level LGR, and analytical injectivity model. It is observed that we achieve higher injection rate with grid refinement compared to the original grid, which is consistent with the fact of polymer rheology. This is significant since we need to accurately calculate how high of polymer viscosity can be injected and the predicted BHP pressure for cases that the operators do not plan to inject polymer above the frac gradient. In this case, we lack the results of FGR because of the excessive simulation time. It is observed that the analytical model slightly overestimates the overall injection rate compared to 4-level LGR.

Case 4: A pilot of Alkaline Co-solvent Polymer (ACP) flood

The reservoir is a sandstone reservoir at a depth of approximately 1,000 ft which has gone through water flood for several years. The average oil saturation before ACP flood is approximately 44.3%. The pilot area includes 6 inverted 7-spot well patterns. The polymer solution is assumed shear-thinning. The reservoir and well descriptions are given in **Table 5**. The permeability field and well locations are shown in **Fig. 16**. The relevant grid with a 3-level LGR is in **Fig. 17**. Injectors are operated at constant injection rates.

Fig. 18 shows the BHP of Injector ECN-105i using different grids and well models (we note that other injectors have similar pressure profiles). For the original 42×37×5 coarse grid, the BHP is the highest. When the grid is locally refined near the injectors, it shows that the BHP significantly decreases, and the pressure change is more gradual, showing a significant improvement for estimating the injector BHP using the LGR. 3-level LGR gives a much smaller BHP compared to the 2-level LGR. The results from the 2-level and 3-level LGR are in very good agreement with the relevant FGR results, respectively, while taking much fewer CPU times (**Table 6**). Even though more CPU time is needed using the LGR compared to the coarse grid indicated by **Table 6**, one needs to consider the significant improvement in accuracy to balance the cost of computational time. The analytical injectivity model gives a similar trend but different profile of the pressure compared to using the original grid and the LGR grids, because there is a skin for the well while the analytical injectivity model originates assuming skin equals to 0. It indicates that the analytical polymer injectivity model is not always useful for the field cases.

Summary and Conclusions

We have developed an efficient local grid refinement (LGR) algorithm to improve the accuracy of numerically estimating the near wellbore solutions when dealing with complex rheology of polymer or emulsion solutions. We present the algorithm to generate the block list and connections and propose an

efficient numerical scheme to couple the pressure and mass conservation equations using the LGR composite grid and with consideration of heterogeneous reservoir properties. Several numerical examples are carried out focusing on the polymer flooding, reservoir with fractures near injection wells, and ACP flooding. The simulation results reveal that the LGR is able to obtain more accurate polymer injectivity compared to using the coarse grid and the analytical injectivity model. The LGR can deal with more complex and realistic reservoir conditions such as fractures and skin. CPU time is significantly reduced using LGR compared to full grid refinement (FGR). It offers a reliable and efficient solution to handle the general concern of reservoir simulations for the shear-dependent polymer rheology in chemical flooding projects.

Acknowledgements

The authors would like to thank Dr. K. Sepehrnoori and Dr. F. Marcondes for their valuable suggestions.

References

[1]Wassmuth, F., Green, K., Hodgins, L., and Turta, A. 2007. Polymer flood technology for heavy oil recovery. Paper presented at the *Canadian International Petroleum Conference, 12-14 June, Calgary, Alberta, Canada.*

[2]Mohammadi, H., and Jerauld G. 2012. Mechanistic modeling of the benefit of combining polymer with low salinity water for enhanced oil recovery. Paper SPE 153161 presented at the *SPE Improved Oil Recovery Symposium, 14-18 April, Tulsa, Oklahoma, USA*.

[3]Morel, D., Vert, M., Jouenne, S., Gauchet, R., and Bouger, Y. 2010. First polymer injection in deep offshore field Angola: recent advances on Dalia/Camelia field case. Paper SPE 135735 presented at the *SPE Annual Technical Conference and Exhibition, 19-22 September, Florence, Italy*.

[4]Seright, R., Seheult, J., and Talashek, T. 2009. Injectivity characteristics of EOR polymers. *SPE Reservoir Evaluatio & Engineering*, *12*(5): 783-792.

[5]Zaitoun, A., Makakou, P., Blin, N., Al-Maamari, R. S., Al-Hashmi, A. A. R., and Abdel-Goad, M. 2012. Shear stability of EOR polymers. *SPE Journal*, 17(02): 335-339.

[6]van den Hoek, P.J., Mahani, H., Sorop, T.G., Brooks, A.D., Zwaan, M., Sen S., Shuaili K., and Saadi, F. 2012. Application of injection fall-off analysis in polymer flooding. Paper SPE 154376 presented at *the 74th EAGE Conference & Exhibition incorporating SPE Europec, 4-7 June, Copenhagen, Denmark.*

[7]Goudarzi, A., Zhang, H., Varavei, A., Hu, Y., Delshad, M., Bai, B., and Sepehrnoori, K. 2013. Water management in mature oil fields using preformed particle gels. Paper SPE 165356 presented at the *SPE Western Regional & AAPG Pacific Section Meeting, 2013 Joint Technical Conference ,19-25 April, Monterey, California, USA*.

[8]Delshad, M., Kim, D., Magbagbeola, O., Huh, C., Pope, G., and Tarahhom, F. 2008. Mechanistic interpretation and utilization of viscoelastic behavior of polymer solutions for improved polymer-flood efficiency. Paper SPE 113620 presented at the *SPE Symposium on Improved Oil Recovery, 20-23 April, Tulsa, Oklahoma, USA*.

[9]Sharma, A., Delshad, M., Huh, C., and Pope, G. 2011. A Practical Method to Calculate Polymer Viscosity Accurately in Numerical Reservoir Simulators. Paper SPE 147239 presented at the *SPE Annual Technical Conference and Exhibition, 30 October-2 November, Denver, Colorado, USA*.

[10]Kulawardana, E., Koh, H., Kim, D. H., Liyanage, P., Upamali, K., Huh, C., Weerasooriya, U., and Pope, G. 2012. Rheology and transport of improved eor polymers under harsh reservoir conditions. Paper SPE 154294 presented at the *SPE Improved Oil Recovery Symposium, 14-18 April, Tulsa, Oklahoma, USA*.

[11]Li, Z., and Delshad, M. 2014. Development of an analytical injectivity model for non-newtonian polymer solutions. *SPE Journal*, 19(3): 381-389.

[12]Gadde, P., and Sharma, M. 2001. Growing injection well fractures and their impact on waterflood performance. Paper SPE 71614 presented at the *SPE Annual Technical Conference and Exhibition, 30 September-3 October, New Orleans, Louisiana*.

[13]Lee, K., Huh, C., and Sharma, M. 2011. Impact of fractures growth on well injectivity and reservoir sweep during waterflood and chemical eor processes. Paper SPE 146778 presented at the *SPE Annual Technical Conference and Exhibition, 30 October-2 November, Denver, Colorado, USA*.

[14]Oliveira, D. F., & Reynolds, A. 2014. An Adaptive Hierarchical Multiscale Algorithm for Estimation of Optimal Well Controls. *SPE Journal*, (Preprint).

[15]Suicmez, V., van Batenburg, D., Matsuura, T., Bosch, M., and Boersma, D. 2011. Dynamic Local Grid Refinement for Multiple Contact Miscible Gas Injection. Paper presented at the *International Petroleum Technology Conference, 15-17 November, Bangkok, Thailand*.

[16]Nilsson, J., Gerritsen, M., and Younis, R. 2005. An adaptive, high-resolution simulation forsteam-injection processes. Paper SPE 93881 presented at the *SPE Western Regional Meeting, 30 March-1 April, Irvine, California*.

[17]Christensen, J. R., Darche, G., Dechelette, B., Ma, H., and Sammon, P. H. 2004. Applications of dynamic gridding to thermal simulations. Paper SPE 86969 presented at the *SPE International Thermal Operations and Heavy Oil Symposium and Western Regional Meeting, 16-18 March, Bakersfield, California.*

[18]Berger, M. J., and Oliger, J. 1984. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of computational Physics*, 53(3): 484-512.

[19]Forsyth, P. A., and Sammon, P. H. 1986. Local mesh refinement and modeling of faults and pinchouts. *SPE Formation Evaluation*, *1*(3): 275-285.

[20]Rasaei, M. R., and Sahimi, M. 2009. Upscaling of the permeability by multiscale wavelet transformations and simulation of multiphase flows in heterogeneous porous media. *Computational Geosciences*, 13(2): 187-214.

[21]Nacul, E. C., Lepretre, C., Pedrosa, O.A., Girard, P., and Aziz, K. 1990. Efficient use of domain decomposition and local grid refinement in reservoir simulation. Paper SPE 20740 presented at the *SPE Annual Technical Conference and Exhibition, 23-26 September, New Orleans, Louisiana*.

[22]Karimi-Fard, M., and Durlofsky, L. 2012. Accurate resolution of near-well effects in upscaled models using flow based unstructured local grid refinement. *SPE Journal*, *17*(4): 1084-1095.

[23]Delshad, M., Pope, G. A., and Sepehrnoori, K. 1996. A compositional simulator for modeling surfactant enhanced aquifer remediation, 1 formulation. *Journal of Contaminant Hydrology*, *23*(4): 303-327.

[24]Meter, D. M., and Bird, R. B. 1964. Tube flow of non-Newtonian polymer solutions: PART I. Laminar flow and rheological models. AIChE Journal, 10(6): 878-881.

[25]Carreau, P. J. 1968. *Rheological equations from molecular network theories*. University of Wisconsin--Madison.

[26]Cannella, W. J., Huh, C., and Seright, R. S. 1988. Prediction of xanthan rheology in porous media. Paper SPE 18089 presented at the *SPE Annual Technical Conference and Exhibition, 2-5 October, Houston, Texas*.

[27]Sorbie, K. S. 1991. *Polymer-Improved Oil Recovery*. Blackie and Son Ltd. Glasgow and London.

[28]Wreath, D., Pope, G. A., and Sepehrnoori, K. 1990. Dependence of polymer apparent viscosity on the permeable media and flow conditions. *In Situ*, 14(3).

[29]Peaceman, D. W. 1983. Interpretation of well-block pressures in numerical reservoir simulation with nonsquare grid blocks and anisotropic permeability. *SPE Journal*, 23(3): 531-543.

[30]Gerritsen, M., and Lambers, J. V. 2008. Integration of local-global upscaling and grid adaptivity for simulation of subsurface flow in heterogeneous formations. *Computational Geosciences*, 12(2), 193-208.

[31]Saad, N. 1989. *Field-scale simulation of chemical flooding*. Texas Univ., Austin, TX (USA).

[32]Clemens, T., Deckers, M., Kornberger, M., Gumpenberger, T., and Zechner, M. 2013. Polymer solution injection-near wellbore dynamics and displacement efficiency, pilot test results, Matzen Field, Austria. Paper SPE 164904 presented at the *EAGE Annual Conference & Exhibition incorporating SPE Europec, 10-13 June, London, UK*.

[33]Manichand, R. N., Let, M. S., Kathleen, P., Gil, L., Quillien, B., and Seright, R. S. 2013. Effective propagation of HPAM solutions through the tambaredjo reservoir during a polymer flood. *SPE Production & Operations*, 28(04): 358-368.

Appendix A: A novel numerical scheme to solve the pressure equation on the multilevel grid

To simplify the complex flow in the composite blocks as shown in **Fig. 6**, we decompose the flow pattern into two flow patterns (for two-dimensional case) with main flow directions along x and y, which is shown in **Fig. 19**. For each pattern, it is assumed a pressure drop along the main flow direction while making the lateral sides impermeable. Then, we split the flow domain into two parallel parts, such as part A and part B in the x-direction main flow pattern and part C and part D in the y-direction main flow pattern. According to the continuity of mass flux across the interfaces *m* and *n*, we draw the pressure curves through the two parts of each pattern shown in **Fig. 19** by neglecting the cross flows between the two parts. Then, we investigate the expressions of $u_{(m)}$ and $u_{(n)}$ for each pattern based on the pressure curves.

For the flow pattern with a flow direction along x, according to the geometric knowledge, the length of line *i1-i2* equals to the length of line *j-k* because they are both half the length of line *m-n*. It is easy to infer that line *i1-j* is equal to and parallel to line *i2-k*. As a result, the line connecting point *i* and the point at the center of line *j-k* is equal to and parallel to the previous two lines. Therefore, we can obtain the following relationship:

$$P_{(i1)} - P_{(j)} = P_{(i2)} - P_{(k)} = P_{(i)} - \frac{P_{(j)} + P_{(k)}}{2} \tag{24}$$

Using Darcy's Law, we have the expressions of $u_{(m)}$ and $u_{(n)}$ according to the continuity of mass flux across the interfaces *m* and *n*:

$$\begin{aligned} u_{(m)} &= -T_{(m)}\left(\frac{P_{(j)} + P_{(k)}}{2} - P_{(i)}\right) \\ u_{(n)} &= -T_{(n)}\left(\frac{P_{(j)} + P_{(k)}}{2} - P_{(i)}\right) \end{aligned} \tag{25}$$

where $T_{(m)}$ and $T_{(n)}$ are transmissibilities on the interfaces *m* and *n* defined by

$$T_{(m)} = \frac{4\lambda_{(i)}\lambda_{(j)}}{\left(\lambda_{(i)} + 2\lambda_{(j)}\right)\Delta x}$$
$$T_{(n)} = \frac{4\lambda_{(i)}\lambda_{(k)}}{\left(\lambda_{(i)} + 2\lambda_{(k)}\right)\Delta x} \tag{26}$$

For the flow pattern with a flow direction along y, according to the geometric relation, the length of line *i1-j* equals to the length of line *i2-k* because they are both half the length of line *i-l*. Consequently, the following relationship can be obtained:

$$P_{(i1)} - P_{(j)} = P_{(i2)} - P_{(k)} = P_{(i)} - \frac{\lambda_{(j)}P_{(j)} + \lambda_{(k)}P_{(k)}}{(\lambda_{(j)} + \lambda_{(k)})} \tag{27}$$

Using Darcy's Law, we have the expressions of $u_{(m)}$ and $u_{(n)}$ according to the continuity of mass flux across the interfaces *m* and *n*:

$$u_{(m)} = -T_{(m)}\left(\frac{\lambda_{(j)}P_{(j)} + \lambda_{(k)}P_{(k)}}{(\lambda_{(j)} + \lambda_{(k)})} - P_{(i)}\right)$$
$$u_{(n)} = -T_{(n)}\left(\frac{\lambda_{(j)}P_{(j)} + \lambda_{(k)}P_{(k)}}{(\lambda_{(j)} + \lambda_{(k)})} - P_{(i)}\right) \tag{28}$$

Comparison between Eq. (25) and Eq. (28) shows that the expressions of $u_{(m)}$ and $u_{(n)}$ are different under the two flow patterns. Nevertheless, they can achieve accordance under the homogeneous condition or the condition that the total mobilities of the fine blocks are identical, i.e. $\lambda_{(j)} = \lambda_{(k)}$, which indicates that the expressions are rigorously self-consistent under these conditions.

For the heterogeneous condition, we need to choose appropriate expressions of $u_{(m)}$ and $u_{(n)}$ from Eq. (25) and Eq. (28) by considering the direction of the main flow. A decent estimation to the velocity across the interface can be a weighting of the x-direction velocity and the y-direction velocity weighted by the magnitude of the local velocities along x and y, i.e., $u^0_{(mn)}$ and $u^0_{(jk)}$, where

$$u^0_{(mn)} = \frac{\left(\left|u^0_{(m)}\right| + \left|u^0_{(n)}\right|\right)}{2} \tag{29}$$

and $u^0_{(jk)}$ is the flux across the interface of blocks *j* and *k* shown in Fig. 6, which is expressed by

$$u^0_{(jk)} = -\frac{4\lambda_{(j)}\lambda_{(k)}}{\left(\lambda_{(j)} + \lambda_{(k)}\right)\Delta y}\left(P_{(k)} - P_{(j)}\right) \tag{30}$$

where the superscript 0 for the velocities represents the last time step.

Accordingly, we design the following numerical scheme to calculate $u_{(m)}$ and $u_{(n)}$:

$$\begin{cases} u_{(m)} = -T_{(m)}\left[\left(\dfrac{P_{(j)} + P_{(k)}}{2} - P_{(i)}\right)\dfrac{u^0_{(mn)}}{u^0_{(mn)} + u^0_{(jk)}} + \left(\dfrac{\lambda_{(j)}P_{(j)} + \lambda_{(k)}P_{(k)}}{(\lambda_{(j)} + \lambda_{(k)})} - P_{(i)}\right)\dfrac{u^0_{(jk)}}{u^0_{(mn)} + u^0_{(jk)}}\right] \\ u_{(n)} = -T_{(n)}\left[\left(\dfrac{P_{(j)} + P_{(k)}}{2} - P_{(i)}\right)\dfrac{u^0_{(mn)}}{u^0_{(mn)} + u^0_{(jk)}} + \left(\dfrac{\lambda_{(j)}P_{(j)} + \lambda_{(k)}P_{(k)}}{(\lambda_{(j)} + \lambda_{(k)})} - P_{(i)}\right)\dfrac{u^0_{(jk)}}{u^0_{(mn)} + u^0_{(jk)}}\right] \end{cases} \tag{31}$$

This numerical scheme is achieved based on the assumption that the cross flows between the separated parts for each flow pattern is neglected. Actually, the piece-wise pressure curves shown in **Fig. 19** may be bended when there are across flow vertical to the main flow direction. Nevertheless, this numerical scheme has a lot of advantages that will be discussed in the text.

Captions

Fig. 1. Rheological relation between viscosity and shear rate for polymer solution.

Fig. 2. Schematic of multilevel local grid refinement.

Fig. 3. Flowchart of UTCHEM.

Fig. 4. Flowchart of UTCHEM using LGR module.

Fig. 5. An example of block list and connections.

Fig. 6. Schematic block connection and the position of pressure and velocity.

Fig. 7. Schematic of a third-order upstream scheme (Leonard's scheme) for structured grid.

Fig. 8. Schematic of Leonard's scheme for an LGR case.

Fig. 9. The mesh of 4-level local grid refinement for Case 1.

Fig. 10. Comparison of injection pressure using different grids or well models for Case 1.

Fig. 11. Permeability field of a fractured reservoir using an LGR grid.

Fig. 12. Comparison of injector BHP using different grids or well models for Case 2.

Fig. 13. Permeability distribution and well locations for Case 3.

Fig. 14. Well locations and 4-level local grid refinement.

Fig. 15. Comparison of overall injection rate using different grids or well models for Case 3.

Fig. 16. Permeability distributions and well locations for Case 4.

Fig. 17. Schematic of the 3-level local grid refinement for Case 4.

Fig. 18. Comparison of BHP for injector ECN-105i using different grids and well models for Case 4.

Fig. 19. Decomposition of the flow pattern into two with main flow directions along x and y; and the approximate pressure curves through the two separated parts (A and B for x-direction pattern, C and D for y-direction pattern) of each pattern.

Table 1. Mutual indexing of block list and connections

Block No.	Connections (x-direction)	Connections (y-direction)
1	1	1
2	1,2	2
3	2,5,3	3
4	4	1,4
5	4,5	2,5
6	6	4,5
7	6,7	3
8	3,7	-

Connection No. (x-direction)	Block pair
1	1,2
2	2,3
3	3,8
4	4,5
5	5,3
6	6,7
7	7,8

Connection No. (y-direction)	Block pair
1	1,4
2	2,5
3	3,7
4	4,6
5	5,6

Table 2. Reservoir and well descriptions (Case 1)

Model description	Values
Reservoir size	450 ft × 450 ft × 10 ft
No. of gridblocks	15×15×1
Simulation time	365 days
Number of components	3
Permeability in X or Y direction	300 md
Initial water saturation	0.35
Wells	1 injector; 1 producer
Injection rate	500 ft^3/day
Producer BHP	1000 psi
Water injection	0-150 days and 270-365 days
Polymer injection	150-270 days (0.3 wt%)

Table 3. CPU times for Case 1 using LGR and FGR methods

Grid	4-level LGR (471 gridblocks)	4-level FGR (14400 gridblocks)
CPU time (hr)	0.15	8.5

Table 4. Reservoir and well descriptions (Case 3)

Model description	Values
No. of gridblocks	17×21×25
No. of components	6
Total simulation time (PV)	0.32
Polymer injection	0~0.16 PV (0.2 wt%)
Water injection	0.16~0.32 PV
BHP (psi)	Injectors 4500; Producers 700

Table 5. Reservoir and well descriptions (Case 4)

Model description	Values	
Reservoir dimension	5,512 ft × 4,856 ft × 98 ft	
No. of gridblocks	42×37×5	
No. of components	12	
Total simulation time	7300 days	
Optimum salinity	0.26 meq/mL	
Wells	6 injectors; 22 producers	
ACP injection	0~3650 days	1.5 wt% Co-solvent 0.275 wt% Polymer
Polymer injection	3650~7300 days	0.225 wt% Polymer

Table 6. CPU times for Case 4 using original grid, LGR, and FGR

Grid	Original grid (7770 gridblocks)	3-level LGR (13185 gridblocks)	3-level FGR (124320 gridblocks)
CPU time (hr)	0.4	5.8	71

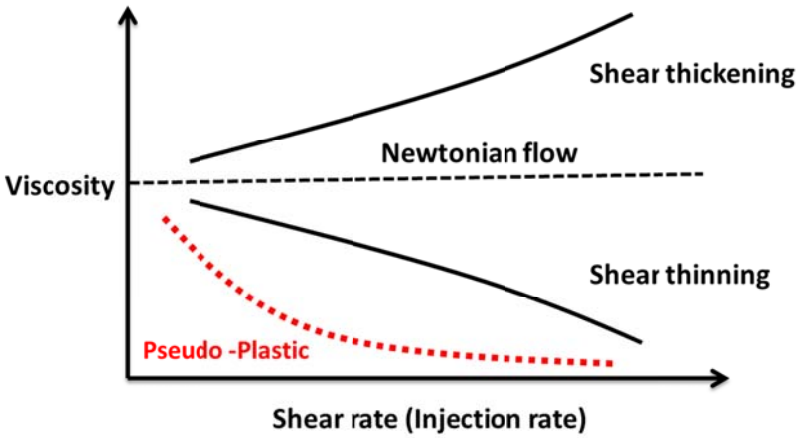

Fig. 1. Rheological relation between viscosity and shear rate for polymer solution.

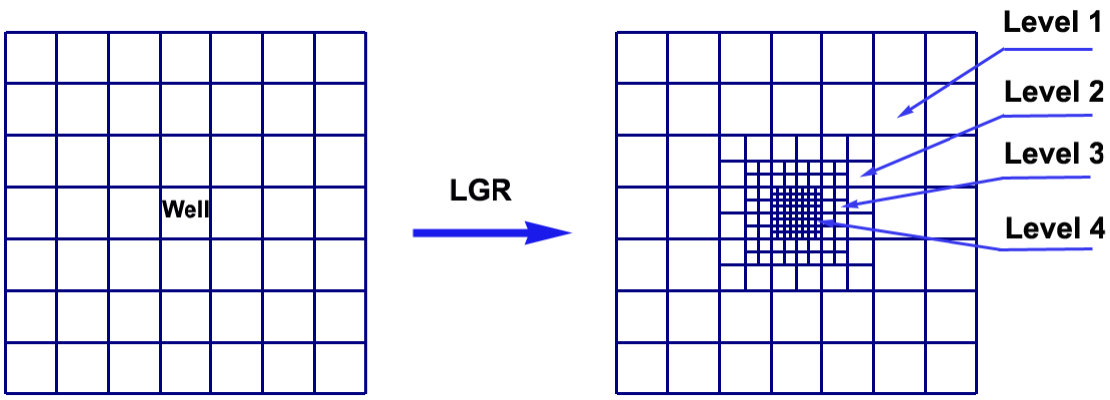

Fig. 2. Schematic of multilevel local grid refinement.

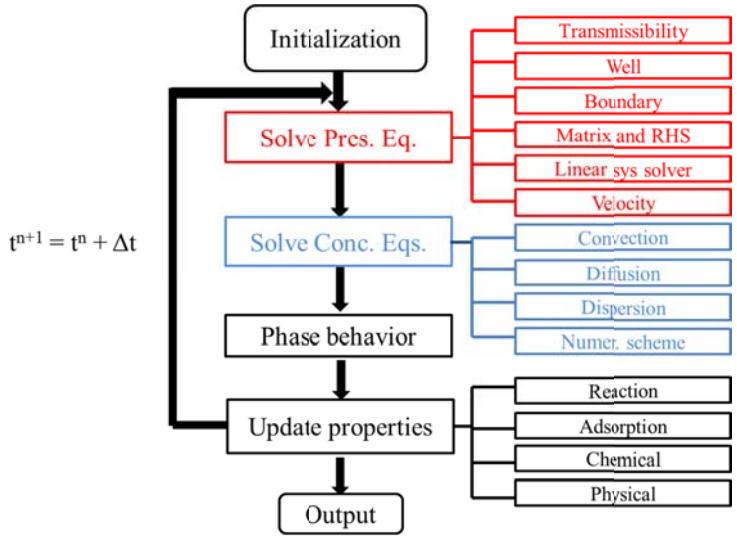

Fig. 3. Flowchart of UTCHEM.

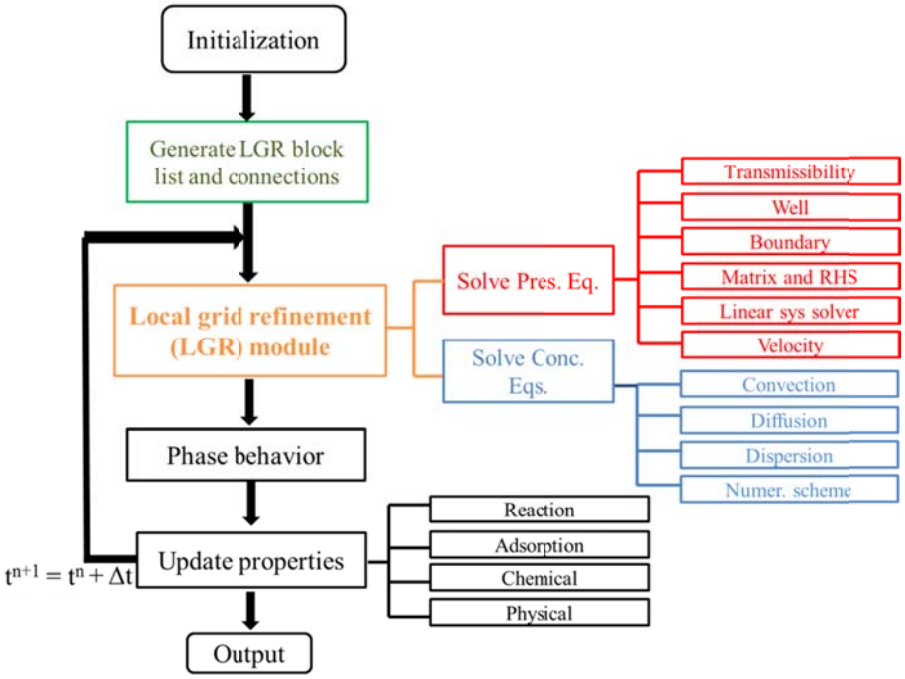

Fig. 4. Flowchart of UTCHEM using LGR module.

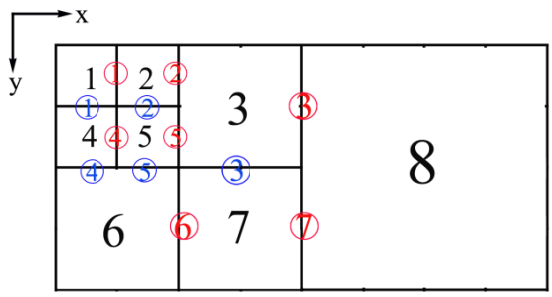

Fig. 5. An example of block list and connections.

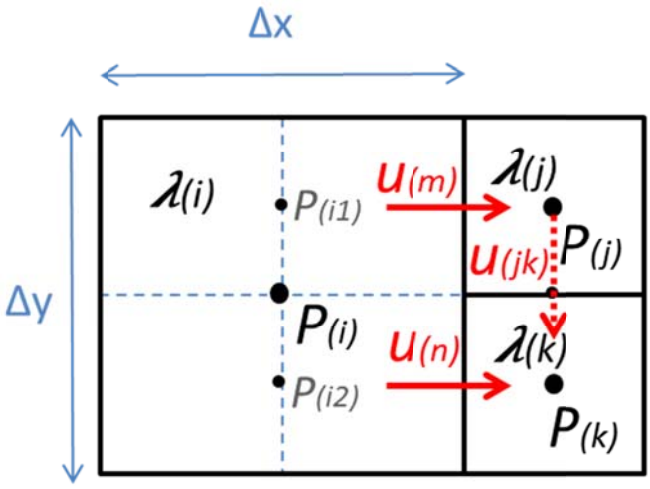

Fig. 6. Schematic block connection and the position of pressure and velocity.

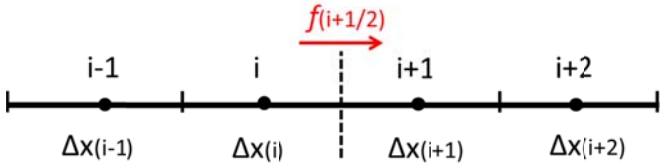

Fig. 7. Schematic of a third-order upstream scheme (Leonard's scheme) for structured grid.

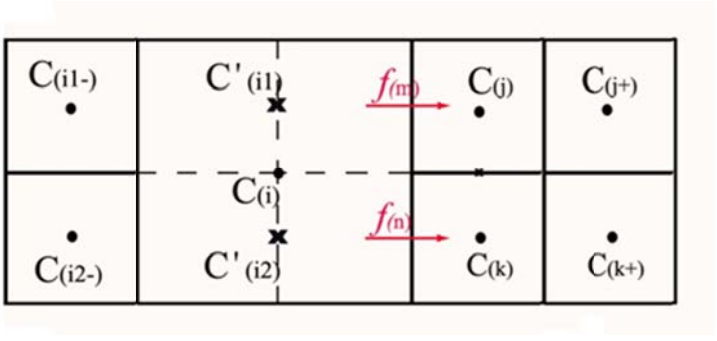

Fig. 8. Schematic of Leonard's scheme for an LGR case.

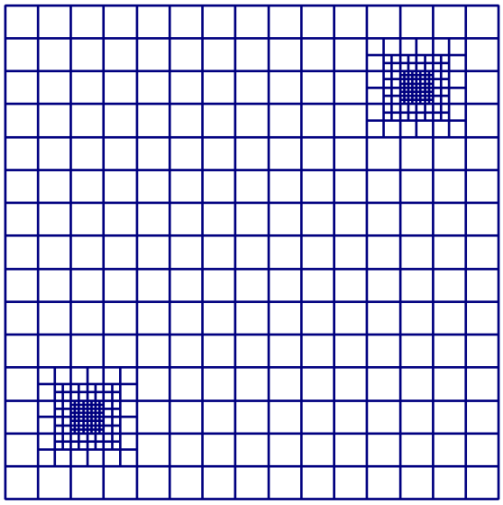

Fig. 9. The mesh of 4-level local grid refinement for Case 1.

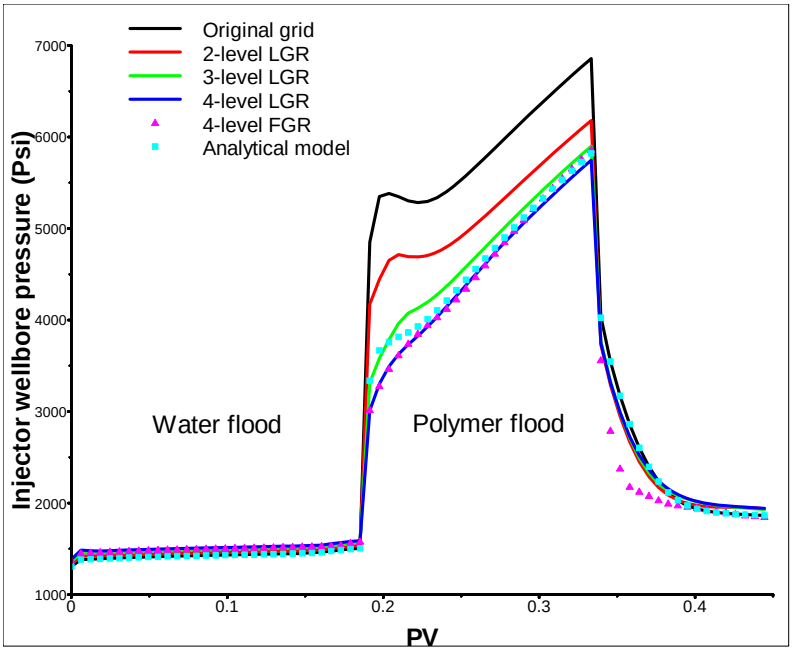

Fig. 10. Comparison of injection pressure using different grids or well models for Case 1.

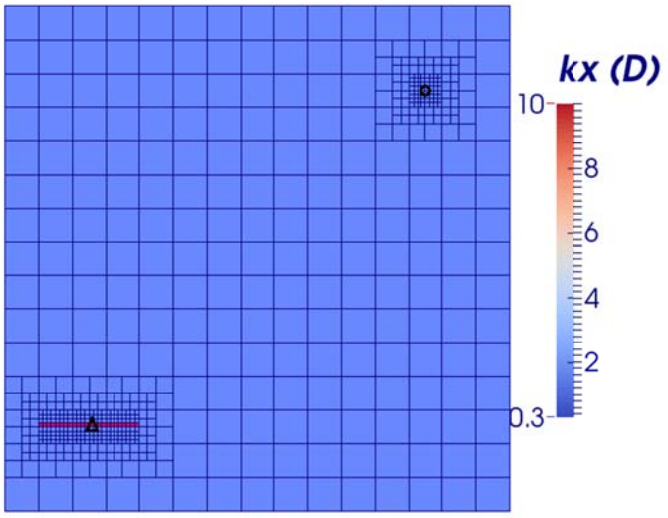

Fig. 11. Permeability field of a fractured reservoir using an LGR grid.

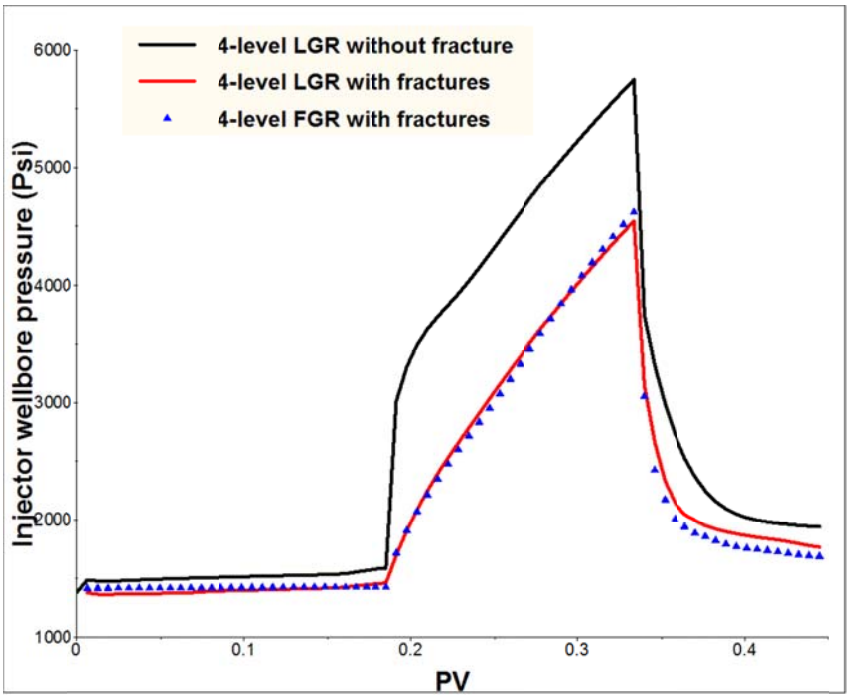

Fig. 12. Comparison of injector BHP using different grids or well models for Case 2.

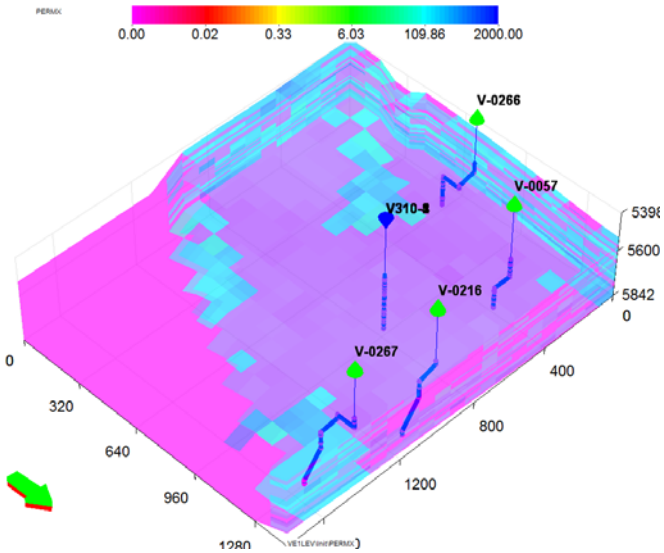

Fig. 13. Permeability distribution and well locations for Case 3.

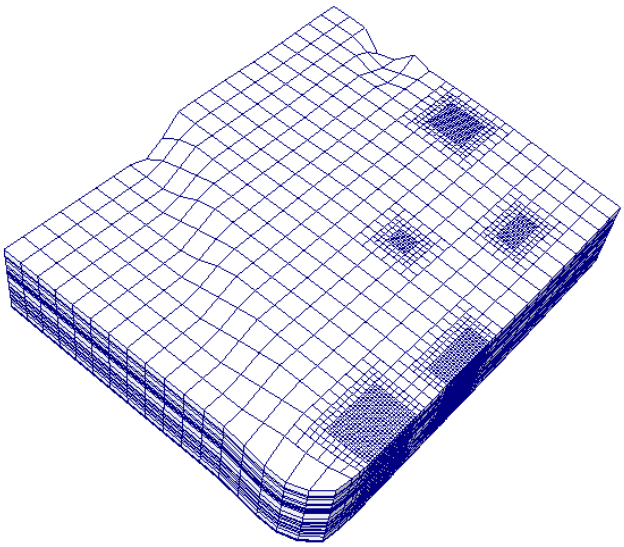

Fig. 14. Well locations and 4-level local grid refinement.

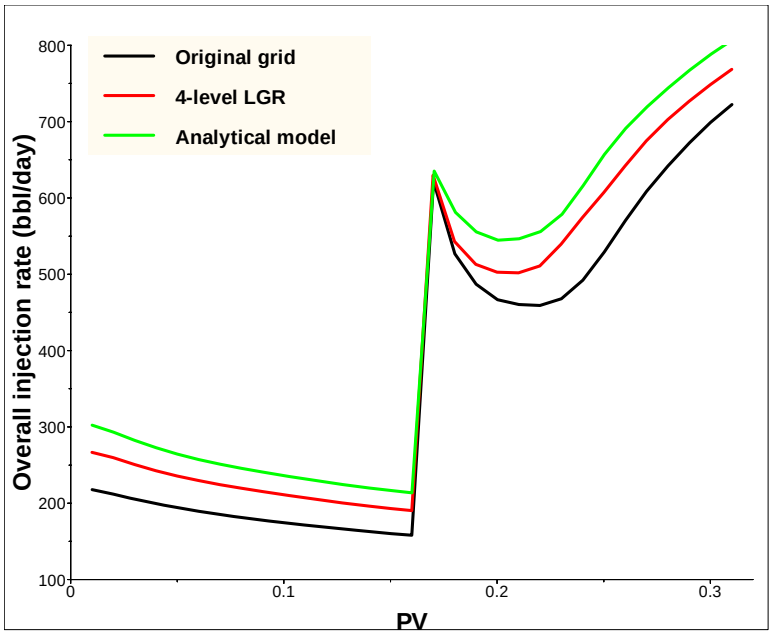

Fig. 15. Comparison of overall injection rate using different grids or well models for Case 3.

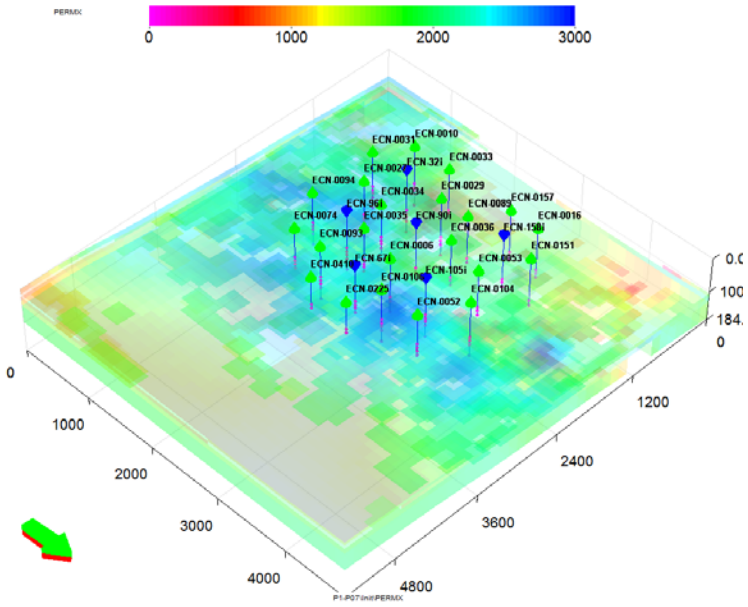

Fig. 16. Permeability distributions and well locations for Case 4.

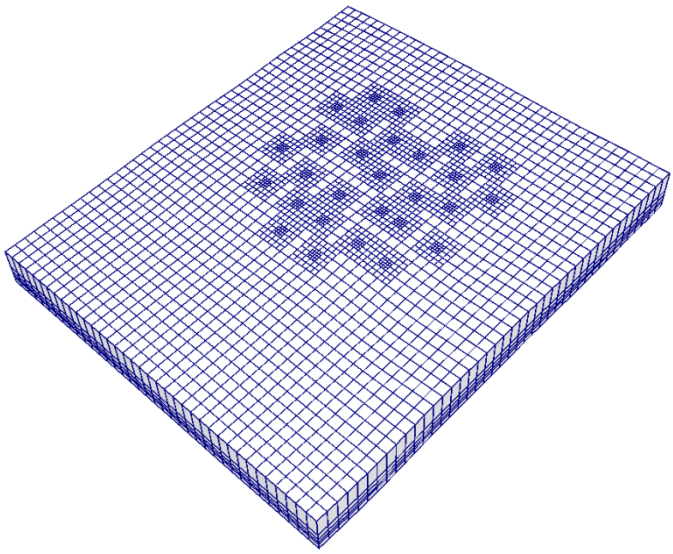

Fig. 17. Schematic of the 3-level local grid refinement for Case 4.

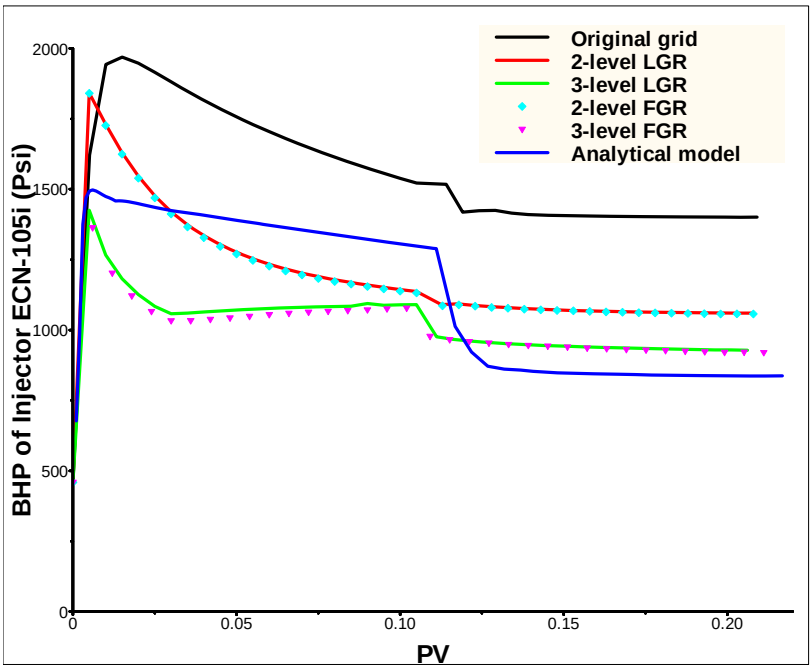

Fig. 18. Comparison of BHP for injector ECN-105i using different grids and well models for Case 4.

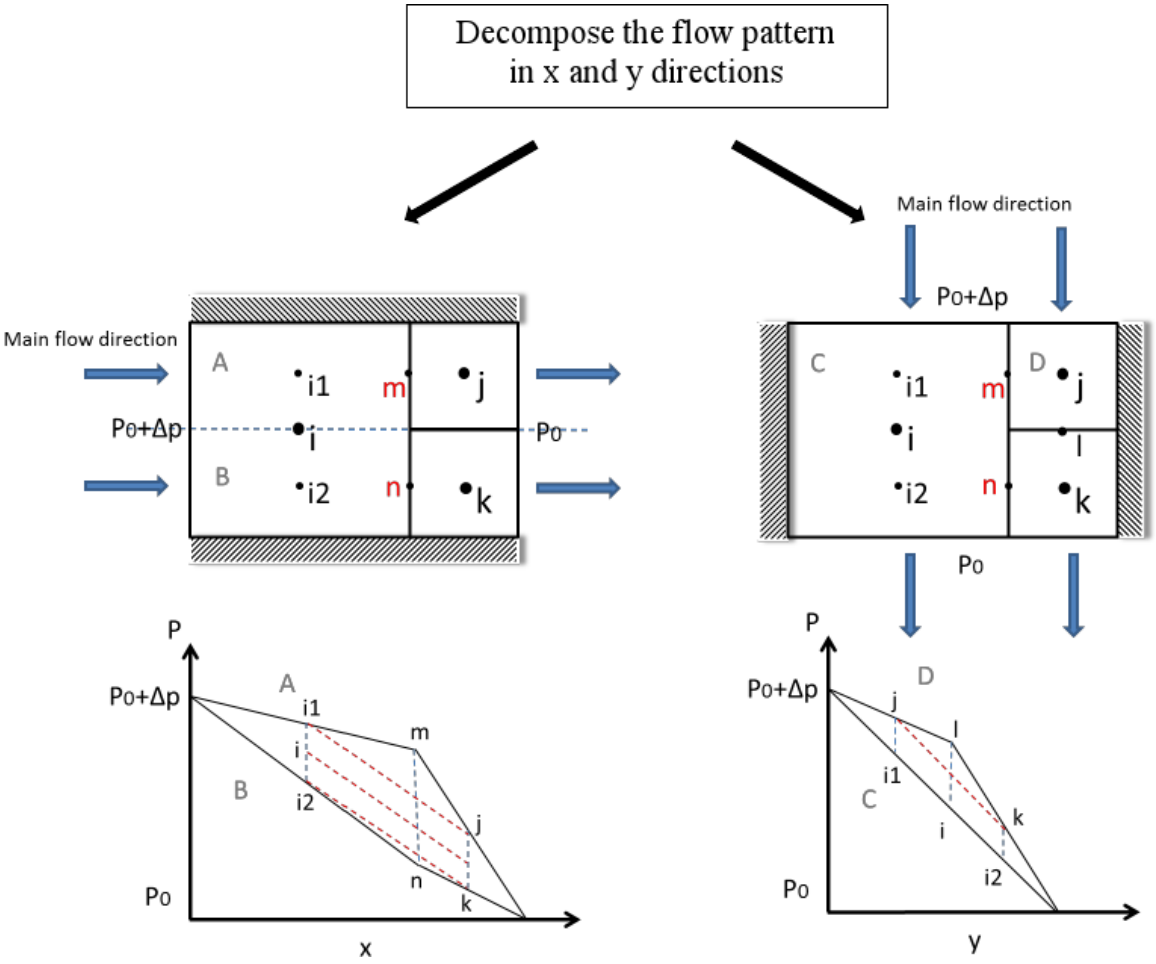

Fig. 19. Decomposition of the flow pattern into two with main flow directions along x and y; and the approximate pressure curves through the two separated parts (A and B for x-direction pattern, C and D for y-direction pattern) of each pattern.