2016 IEEE 2nd International Conference on Big Data Security on Cloud, IEEE International Conference on High Performance and Smart Computing, IEEE International Conference on Intelligent Data and Security

Large-scale Reservoir Simulations on Parallel Computers

Yan Chen*‡, Hui Liu*, Kun Wang*, Zhangxin Chen*, Yanfeng He*, Bo Yang*, and Peng Zhang †
* Dept. of Chemical and Petroleum Engineering, University of Calgary, Calgary, AB, Canada, T2N 1N4
Email: {yan.chen2, hui.j.liu, wang30, zhachen, yanfeng.he, yang6}@ucalgary.ca
† College of Engineering and Applied Sciences, Stony Brook University, NY, USA, 11794
Email: peng.zhang@stonybrook.edu
‡ College of Mathematics and Informatics, South China Agricultural University, Guangzhou, China, 510642

***Abstract*—This paper presents our work on simulation of large-scale black oil models on parallel computers. An in-house platform has been developed and a black oil simulator has been implemented based on this platform, which can handle the standard black oil model and oil-water model. Numerical methods and a new parallel preconditioner are introduced. The simulator uses MPI for communication among computation nodes and it is capable of simulating black oil models with hundreds of millions of grid cells. Numerical simulations show that our simulator has excellent scalability and it can speed simulations thousands of times faster.**

I. Introduction

Reservoir simulations, which can model the movement of oil and gas in a reservoir, are powerful tools in the oil and gas industry. The geological models of reservoirs can be complex and highly heterogeneous. Usually, these kinds of models are hard to simulate. When a model is large enough, a simulator may take days to finish one project. Efficient numerical methods, such as numerical methods for reservoir models and linear solvers, and fast reservoir simulators should be investigated.

Reservoir simulations have been hot research topics for decades. Various reservoir models and their solution methods have been developed [13], and various computer techniques have been applied [6], [20], [21], [22]. Chen et al. studied finite element methods and finite difference methods for the black oil, compositional and thermal models [13]. Nonlinear methods, linear solvers and preconditioners were also studied [13]. Kaarstad et al. [3] developed a parallel oil-water simulator, which handled two phases (oil and water) and two components (oil and water). Rutledge et al. [1] implemented a compositional simulator for parallel computers. This simulator used the IMPES (implicit pressure-explicit saturation) method, where only pressure unknowns were solved and other variables were derived from pressure unknowns. Shiralkar et al. [2] developed a parallel reservoir simulator, and it was designed to run on a variety of parallel systems. Killough and his collaborators [4] studied locally refinement techniques, which could improve accuracy and reduce calculations compared with global grid refinement. Dogru and his group [5] studied parallel black oil simulation using structured and unstructured grids. Their parallel simulator was highly efficient and it could calculate reservoir models with billions of grid cells. Zhang et al. developed a scalable general-purpose platform for parallel adaptive finite element and adaptive finite volume methods [6], [7], which provided unstructured grids, newest-vertex based bisection refinement for tetrahedral grids, various linear solvers, preconditioners and eigenvalue solvers. The package has been applied to CFD (computational fluid dynamics), Maxwell equations, electronic structure, and material problems. It was also applied to black oil simulations using discontinuous Galerkin methods and the black oil simulator had good scalability [18]. Chen et al. developed a parallel platform to support the implementation of the black oil, compositional and thermal models [21]. Guan et al. implemented a parallel black oil simulator and cases with hundreds of millions of grid cells were calculated [22]. Wheeler studied advanced numerical methods and parallel computing techniques for the black oil model [23]. It is well-known that most of simulation time is spent on the solution of linear systems from reservoir simulations, especially for the black oil model. Many preconditioning methods have been proposed, such as constrained pressure residual (CPR) methods [8], [9], multi-stage methods [10], multiple level preconditioners [17], fast auxiliary space preconditioners (FASP) [11], [12], and CPR-FPF methods [15].

A black oil simulator has been implemented. It can handle several different models, such as the standard black oil model, oil-water model, polymer flooding model, and dual-porosity and dual permeability model. This paper focuses on the standard black oil model and oil-water model. The standard black oil model has three components (oil, water and gas). The water component can only stay in the water phase, the oil component stays in the oil phase, and the gas component can stay both in the oil and gas phases. Each component has a mass conservation equation to describe. The system is a coupled nonlinear system, which is solved by Newton-Raphson methods. We choose structured grids and finite difference methods for the sake of simplicity. The simulator is studied on different parallel computers. Numerical experiments show that our simulator has excellent scalability and it can calculate problems with hundreds of millions of grid cells.

978-1-5090-2403-2/16 $31.00 © 2016 IEEE
DOI 10.1109/BigDataSecurity-HPSC-IDS.2016.20

IEEE computer society

II. Reservoir Simulation Models

The Darcy's law is applied for multi-phase in porous media. The law establishes a relationship among volumetric flow rates of a component, reservoir properties, fluid properties and phase pressure differences in a reservoir, which is described as:

$$Q = -\frac{\kappa A \Delta p}{\mu L}. \tag{1}$$

In above equation, κ is permeability of rock (or reservoir), A is cross-section area in some direction, such as x, y and z direction, Δp is pressure difference, μ is fluid viscosity, and L is length of a porous media in the direction. In three-dimensional space, the differential form of Darcy's law written as:

$$q = \frac{Q}{A} = -\frac{\kappa}{\mu}\nabla p. \tag{2}$$

The law is applied to traditional reservoir models, such as black oil model, compositional model, and thermal model.

A. Black Oil Model

The black oil model is assumed to be isothermal, so energy is not considered. By combining Darcy's law, mass conservation equation for each component can be described as:

$$\begin{cases} \frac{\partial}{\partial t}(\phi s_o \rho_o^o) & = \nabla \cdot (\frac{K K_{ro}}{\mu_o} \rho_o^o \nabla \Phi_o) + q_o, \\ \frac{\partial}{\partial t}(\phi s_w \rho_w) & = \nabla \cdot (\frac{K K_{rw}}{\mu_w} \rho_w \nabla \Phi_w) + q_w, \\ \frac{\partial (\phi \rho_o^g s_o + \phi \rho_g s_g)}{\partial t} & = \nabla \cdot (\frac{K K_{ro}}{\mu_o} \rho_o^g \nabla \Phi_o) + q_o^g \\ & \quad + \nabla \cdot (\frac{K K_{rg}}{\mu_g} \rho_g \nabla \Phi_g) + q_g. \end{cases} \tag{3}$$

For a phase α ($\alpha = o, w, g$), Φ_α is its potential, ϕ and K are porosity and permeability of a reservoir, and s_α, μ_α, p_α, ρ_α, $K_{r\alpha}$ and q_α are phase saturation, phase viscosity, phase pressure, phase density, relative permeability and well rate, respectively. ρ_o^o and ρ_o^g are density of oil component in oil phase and density of solution gas in oil phase.

B. Two-Phase Flow Model

This two-phase model ignores gas phase and can be read as a simplified model of the standard black oil model,, which is written as [13]:

$$\begin{cases} \frac{\partial}{\partial t}(\phi s_o \rho_o) & = \nabla \cdot (\frac{K K_{ro}}{\mu_o} \rho_o \nabla \Phi_o) + q_o \\ \frac{\partial}{\partial t}(\phi s_w \rho_w) & = \nabla \cdot (\frac{K K_{rw}}{\mu_w} \rho_w \nabla \Phi_w) + q_w. \end{cases} \tag{4}$$

C. Well Modeling

A source-sink model and Peaceman method [19] are applied to calculate well rates. For each perforation block m, its well rate for each component $q_{\alpha,m}$ is calculated by:

$$q_{\alpha,m} = W_i \frac{\rho_\alpha K_{r\alpha}}{\mu_\alpha}(p_h - p_\alpha - \rho_\alpha \wp (z_h - z)), \tag{5}$$

where p_h is the bottom hole pressure of a well, W_i is well index of the perforated block m, z_h is reference depth for bottom hole pressure p_h, z is depth of the perforated block m, and p_α is phase pressure of interested component. Various operation constraints may be applied to a well at different time period, such as fixed bottom hole pressure, fixed water rate, fixed oil rate and fixed liquid rate.

D. Nonlinear Methods

The inexact Newton method is applied to our simulation, which solves the linear systems approximately. Its algorithm is shown by Algorithm 1.

Algorithm 1 The inexact Newton Method

```
Given an initial guess $x^0$ and stopping criterion $\epsilon$, let $l = 0$
and assemble the right-hand side $b(x^l)$.
while $\|b(x^l)\| \geq \epsilon$ do
    Assemble the Jacobian matrix $J$.
    Find $\theta_l$ and $x$ such that
        $\|b(x^l) - J\delta x\| \leq \theta_l \|b(x^l)\|,$    (6)
    Let $l = l + 1$ and $x^l = x^{l-1} + \delta x$.
end while
$x^* = x^l$ is the solution of the nonlinear system.
```

The algorithm is the same as standard Newton methods except the choice of θ_l. The standard Newton method chooses a small value, such as 1e-6; in this case, the solution of the corresponding linear system is accurate enough. The termination criteria of the inexact Newton method is large compared with standard Newton method, such as 1e-2. And also, its value is set automatically. Three choices are listed as follows:

$$\theta_l = \begin{cases} \frac{\|b(x^l) - r^{l-1}\|}{\|b(x^{l-1})\|}, \\ \frac{\|b(x^l)\| - \|r^{l-1}\|}{\|b(x^{l-1})\|}, \\ \gamma \left(\frac{\|b(x^l)\|}{\|b(x^{l-1})\|} \right)^\beta, \end{cases} \tag{7}$$

where r^l is the residual of the lth iteration,

$$r^l = b(x^l) - J\delta x. \tag{8}$$

E. Preconditioner

Since linear systems, $Ax = b$, from black oil model are unsymmetric, Krylov solvers are employed. If proper ordering technique is applied, the matrix A has the following structure,

$$A = \begin{pmatrix} A_{pp} & A_{ps} & A_{pw} \\ A_{sp} & A_{ss} & A_{sw} \\ A_{wp} & A_{ws} & A_{ww} \end{pmatrix}, \tag{9}$$

where A_{pp} is the matrix coefficients corresponding to the pressure unknowns, A_{ss} is the matrix coefficients corresponding to the saturation unknowns, A_{ww} is the matrix coefficients corresponding to the well unknowns, and other matrices are coupled items. The system is hard to solve and many CPR-type preconditioners have been developed. We also design a set of parallel CPR-type methods for black oil model and compositional model [15], one of which is CPR-FPF method. F means RAS method (Restricted Additive Schwarz) is applied to linear system $Ar = f$, and P means AMG solver is used to solve linear $A_{pp}r_p = f_p$. The algorithm is described by Algorithm 2.

Algorithm 2 The CPR-FPF Preconditioner for preconditioning system $Ax = f$.

```
$x = R(A)^{-1}f.$
$r = f - Ax$
$x = x + \Pi_p AMG(A_{pp})^{-1}\Pi_r r.$
$r = f - Ax$
$x = x + R(A)^{-1}r.$
```

F. Platform

We have implemented a parallel platform to support reservoir simulator development. It provides grid, data, mapping, linear solver, well modeling, parallel input and output, keywords parsing and visualization [21].

Currently, regular structured hexahedral grids are applied, since they have simple geometry and grids are easy to generate. Each cell of a grid is a hexahedron as shown in Figure 1. Each cell has an integer coordinate (i, j, k) and each component (i, j, and k) is numbered along x-, y- and z-axis. Each cell also has a unique global index, which is calculated as

$$Id(i, j, k) = n_x * n_y * (n_z - k) + n_x * j + i. \tag{10}$$

We can see that the index numbered from the top layer to the bottom layer. This numbering style is employed by most commercial reservoir simulators.

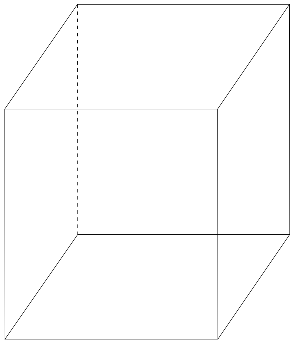

Fig. 1. A hexahedron

In our simulator, a grid is distributed in N_p MPI tasks and each MPI task owns a sub-grid. Let $\mathbb{G}$ be a structured grid, which has $N_g = n_x \times n_y \times n_z$ cells,

$$\mathbb{G} = \{B_1, B_2, \cdots, B_{N_g}\}, \tag{11}$$

where B_i is the i-th cell of $\mathbb{G}$. Let $\mathbb{G}_i$ be the sub-grid owned by the i-th MPI task. When we partition grid, these sub-grids satisfies the following conditions:

$$\begin{cases} \mathbb{G}_i \neq \emptyset \ (i = 1, \cdots, N_p) \\ \mathbb{G}_i \cap \mathbb{G}_j = \emptyset \ (i \neq j) \\ \cup \mathbb{G}_i = \mathbb{G} \ (i = 1, \cdots, N_p). \end{cases} \tag{12}$$

Each sub-grid is not empty, and each cell B_i belongs to some sub-grid. Its neighboring cells may belong to different sub-grids. When we discretize reservoir models, only information from neighboring cell is required, then communication pattern can be modeled by dual graph and communication volume can be approximated by cutting edges. The Hilbert space-filling curve method is employed to partition a grid.

Cell-centered data module is implemented to support finite difference methods and finite volume methods. The platform also has distributed-memory matrix and vector, and basic matrix-vector operations, such as

$$y = \alpha Ax + \beta y, \tag{13}$$

$$z = \alpha Ax + \beta y, \tag{14}$$

$$y = \alpha x + \beta y, \tag{15}$$

$$z = \alpha x + \beta y, \tag{16}$$

$$\alpha = \langle x, y \rangle, \tag{17}$$

$$\alpha = \langle x, x \rangle^{\frac{1}{2}}. \tag{18}$$

With these operations, parallel Krylov subspace solvers and preconditioners are developed, including GMRES, BiCGSTAB, Orthomin, RAS (Restricted Additive Schwarz) and AMG preconditioners.

III. Numerical Experiments

The systems are used for the numerical experiments. The first one is an Blue Gene/Q from IBM. The system, Wat2Q, is located in the IBM Thomas J. Watson Research Center. Each node has 32 computer cards (64-bit PowerPC A2 processor), which has 17 cores. One of them is for the operation system and the other 16 cores for computation. The system has 32,768 CPU cores for computation. The performance of each core is really low compared with Intel processors. However, the system has strong network relative to CPU performance, and the system is scalable. The second one is GPC from SciNet. It uses Intel Xeon E5540 CPU for computation and InfiniBand for communication. Each node has two CPUs and the system has 3,864 nodes (30,912 cores). The tests focus on scalability.

Example 1. *The case tests two-phase oil-water model with a refined SPE10 project and each cell is refined to 27 cells. The model has around 30 millions of cells. The stopping criteria for inexact Newton method is 1e-2 and maximal nonlinear iterations are 20. The linear solver is BiCGSTAB, and its maximal iterations are 100. The simulation time is 10 days. The case is run on IBM Blue Gene/Q. Numerical summaries are shown in Table I and scalability is presented in Fig 2.*

TABLE I
NUMERICAL SUMMARIES OF EXAMPLE 1

# procs	# Steps	# Newton	# Solver	Time (s)
128	40	295	2470	43591.87
256	39	269	2386	20478.49
512	40	260	2664	10709.86
1024	39	259	2665	5578.75

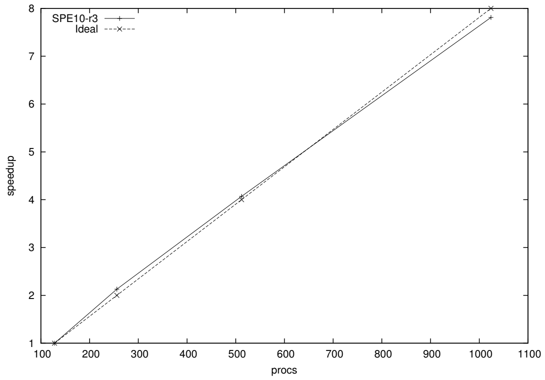

Fig. 2. Scalability of Example 1

This case uses up to 1024 MPI tasks and their speedups are compared with case that uses 128 MPI tasks. From Table I, we can see Newton method and linear solver are robust. When more MPI tasks are employed, fewer Newton iterations are required. And each Newton iteration terminates in around 10 linear iteration. The running time and Figure 2 show the parallel simulator has excellent scalability, which is almost ideal on IBM Blue Gene/Q.

Example 2. *The case tests two-phase oil-water model. A refined SPE10 project is used and each cell is refined to 64 cells. It has around 65 millions of grid cells. The stopping criteria for inexact Newton method is 1e-2 and maximal nonlinear iterations are 20. The linear solver is BiCGSTAB, and its maximal iterations are 100. The simulation time is 20 days. The case is run on GPC (General Purpose Cluster). Numerical summaries are shown in Table II and scalability is presented in Fig 3.*

TABLE II
NUMERICAL SUMMARIES OF EXAMPLE 2

# procs	# Steps	# Newton	# Solver	Time (s)	Eff
512	107	662	6971	26636.13	1.
1024	108	668	7427	13772.96	0.97

This case is larger than last example and is run on a different parallel system. Two different configurations are benchmarked. Again, Table II shows our nonlinear method and linear solver are robust. Each time step uses around 6.5 Newton iterations. Our linear solver and preconditioner are effective, which can solve a linear system in around 11 linear iterations. We assume the case with 512 MPI tasks has an efficiency of 1 and we can see the case with 1024 has an efficiency of 97%, which is good for parallel computing. The scalability is also demonstrated by Figure 3.

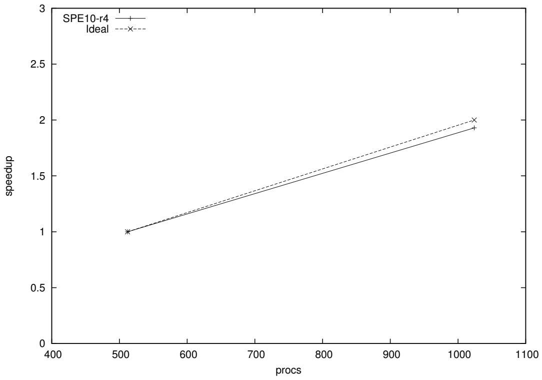

Fig. 3. Scalability of Example 2

Example 3. *The case tests two-phase oil-water model with a refined SPE10 project and each cell is refined to 125 cells. The model has around 140 millions of cells. The stopping criteria for inexact Newton method is 1e-2 and maximal nonlinear iterations are 20. The linear solver is BiCGSTAB, and its maximal iterations are 50. The simulation time is 10 days. The case is run on IBM Blue Gene/Q. Numerical summaries are shown in Table III and scalability is presented in Fig 4.*

TABLE III
NUMERICAL SUMMARIES OF EXAMPLE 3

# procs	# Steps	# Newton	# Solver	Time (s)
256	27	108	495	41127.23
512	27	105	515	19112.77
1024	27	102	572	9756.6
2048	26	101	625	4896.47

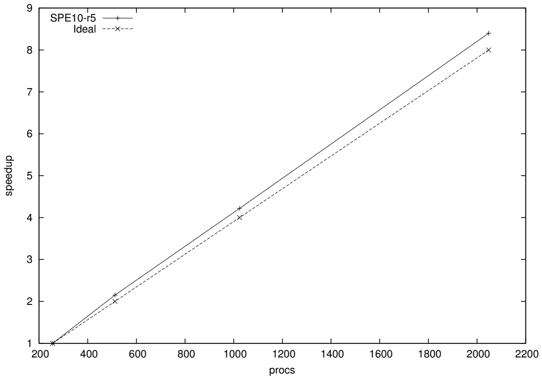

Fig. 4. Scalability of Example 3

From Table III, we can see Newton method is robust. The linear solver is also robust. However, when more MPI tasks are employed, its convergence becomes lower and lower. Average of linear iterations increases from 4.9 to 6.2. Even through, Figure 4 show the simulator has linear scalability.

Example 4. *This case tests the standard black oil model with a refined SPE10 geological model and each cell is refined to 8 cells. It has around 8.96 millions of grid cells. The stopping criteria for inexact Newton method is 1e-3 and maximal nonlinear iterations are 15. The linear solver is BiCGSTAB, and its maximal iterations are 100. The simulation time is 200 days. The case is run on GPC (General Purpose Cluster). Numerical summaries are shown in Table IV and scalability is presented in Figure 5.*

TABLE IV
NUMERICAL SUMMARIES OF EXAMPLE 4

# procs	# Steps	# Newton	# Solver	Time (s)
64	219	1444	23597	82305.88
128	214	1402	23355	41859.71
256	218	1453	26934	22024.53
512	214	1401	24579	11548

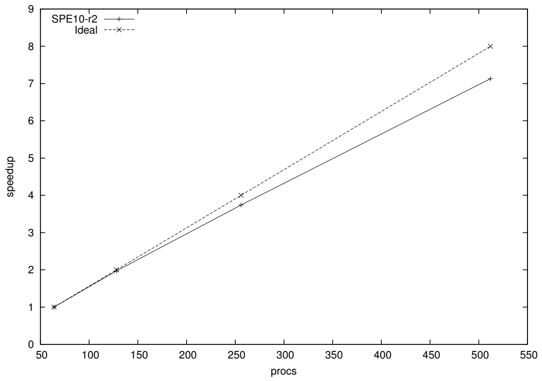

Fig. 5. Scalability of Example 4

This example tests the standard black oil model. Table IV shows Newton method and linear solver are robust. The running time and Figure 5 show the simulator on GPC has a linear scalability.

Example 5. *The case tests the standard black oil model using a refined SPE1 project. The model has 100 millions of cells. The stopping criteria for inexact Newton method is 1e-2 and maximal nonlinear iterations are 15. The linear solver is BiCGSTAB, and its maximal iterations are 20. The simulation time is 10 days. The case is run on IBM Blue Gene/Q. Numerical summaries are shown in Table V and scalability is presented in Fig 6.*

The case with 512 MPI tasks is the base case. From Table V, we can see Newton method and linear solver show good convergence. Figure 6 shows the simulator has excellent scalability and cases with 1024 MPI tasks and 2048 MPI tasks have super-linear scalability.

TABLE V
NUMERICAL SUMMARIES OF EXAMPLE 5

# procs	# Steps	# Newton	# Solver	Time (s)
512	27	140	586	11827.99
1024	27	129	377	5328.46
2048	26	122	362	2703.51
4096	27	129	394	1474.21

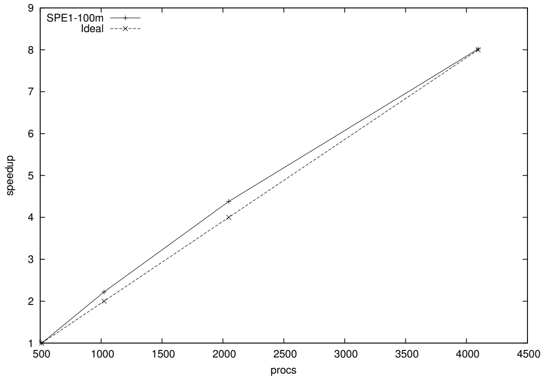

Fig. 6. Scalability of Example 5

Example 6. *The case tests one linear system from pressure equation and the size of the matrix is 3 billion. GMRES(30) solver is applied and it has fixed iterations of 90. The preconditioner is the RAS (Restricted Additive Schwarz) method. The case is run on IBM Blue Gene/Q. Numerical summaries are shown in Table VI and scalability is presented in Fig 7.*

TABLE VI
NUMERICAL SUMMARIES OF EXAMPLE 6

# procs	# Solver	Time (s)
512	90	918.91
1024	90	454.04
2048	90	227.05
4096	90	116.63

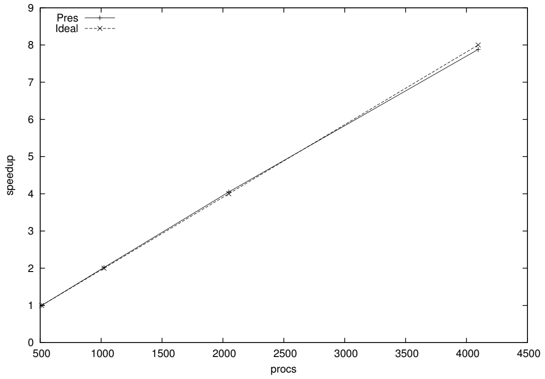

Fig. 7. Scalability of Example 6

This example tests the scalability of linear solver, preconditioner and SpMV. Table VI shows that when MPI tasks are doubled, running time is cut by half. The numerical results and Figure 7 demonstrate the simulator can model extremely large-scale reservoirs and it has excellent scalability.

IV. Conclusion

Parallel black oil simulations are studied in the paper. A standard black oil simulator and a simplified two-phase oil-water simulator are implemented. Numerical experiments show that our simulator has great scalability. The paper also demonstrates that parallel computing is a powerful tool for reservoir simulations.

Acknowledgements

The support of Department of Chemical and Petroleum Engineering, University of Calgary and Reservoir Simulation Group is gratefully acknowledged. The research is partly supported by NSERC/AIEE/Foundation CMG and AITF Chairs.

References

[1] Rutledge, J., Jones, D., Chen, W., and Chung, E., The Use of Massively Parallel SIMD Computer for Reservoir Simulation, SPE-21213, eleventh SPE Symposium on Reservoir Simulation, Anaheim, 1991.

[2] Shiralkar, G., Stephenson, R., Joubert, W., Lubeck, O., and van Bloemen Waanders, B., A production quality distributed memory reservoir simulator, SPE Reservoir Simulation Symposium. 1997.

[3] Kaarstad, T., Froyen, J., Bjorstad, P., Espedal, M., Massively Parallel Reservoir Simulator, SPE-29139, presented at the 1995 Symposium on Reservoir Simulation, San Antonio, Texas, 1995.

[4] Killough, J., Camilleri, D., Darlow, B., Foster, J., Parallel Reservoir Simulator Based on Local Grid Refinement, SPE-37978, SPE Reservoir Simulation Symposium, Dallas, 1997.

[5] Dogru, A., Fung, L., Middya, U., Al-Shaalan, T., Pita, J., A next-generation parallel reservoir simulator for giant reservoirs, SPE/EAGE Reservoir Characterization & Simulation Conference. 2009.

[6] Zhang, L.,, A Parallel Algorithm for Adaptive Local Refinement of Tetrahedral Meshes Using Bisection, Numer. Math.: Theory, Methods and Applications, 2009, 2, 65–89.

[7] Zhang, L., Cui, T., and Liu, H., A set of symmetric quadrature rules on triangles and tetrahedra, J. Comput. Math, 2009, 27(1), 89–96.

[8] Wallis, J., Kendall, R., and Little, T., Constrained residual acceleration of conjugate residual methods, SPE Reservoir Simulation Symposium, 1985.

[9] Cao, H., Schlumberger, T., Hamdi, A., Wallis, J., Yardumian, H., Parallel scalable unstructured CPR-type linear solver for reservoir simulation. SPE Annual Technical Conference and Exhibition. 2005.

[10] Al-Shaalan, T., Klie, H., Dogru, A., Wheeler, M., Studies of Robust Two Stage Preconditioners for the Solution of Fully Implicit Multiphase Flow Problems. SPE Reservoir Simulation Symposium. 2009.

[11] Hu, X., Liu, W., Qin, G., Xu, J., Zhang, Z., Development of a fast auxiliary subspace pre-conditioner for numerical reservoir simulators, SPE Reservoir Characterisation and Simulation Conference and Exhibition. 2011.

[12] Feng, C., Shu, S., Xu, J., and Zhang, C., A Multi-Stage Preconditioner for the Black Oil Model and Its OpenMP Implementation, 21st International Conference on Domain Decomposition Methods, 2012, France.

[13] Chen, Z., Huan, G., and Ma, Y., Computational methods for multiphase flows in porous media, Vol. 2. Siam, 2006.

[14] Falgout, R., and Yang, U., HYPRE: A library of high performance preconditioners, Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2002. 632-641.

[15] Liu, H., Wang, K., and Chen, Z., A family of constrained pressure residual preconditioners for parallel reservoir simulations, Numerical Linear Algebra with Applications, DOI: 10.1002/nla.2017.

[16] Christie, M., and Blunt, M., Tenth SPE comparative solution project: A comparison of upscaling techniques. SPE Reservoir Evaluation & Engineering 4.4 (2001): 308-317.

[17] Baohua Wang, Shuhong Wu, Qiaoyun Li, Xiaobo Li, Hua Li, Chensong Zhang, Jinchao Xu, A Multilevel Preconditioner and Its Shared Memory Implementation for New Generation Reservoir Simulator, SPE-172988-MS, SPE Large Scale Computing and Big Data Challenges in Reservoir Simulation Conference and Exhibition, 15-17 September, Istanbul, Turkey, 2014.

[18] Wang, K., Zhang, L., and Chen, Z., Development of Discontinuous Galerkin Methods and a Parallel Simulator for Reservoir Simulation, SPE-176168-MS, SPE/IATMI Asia Pacific Oil & Gas Conference and Exhibition, 20-22 October, Nusa Dua, Bali, Indonesia, 2015.

[19] Peaceman D., Interpretation of Well-Block Pressures in Numerical Reservoir Simulation, SPE-6893, 52nd Annual Fall Technical Conference and Exhibition, Denver, 1977.

[20] A. Sedighi, Y. Deng, P. Zhang, Fairness of Task Scheduling in High Performance Computing Environments, Scalable Computing: Practice and Experience, 15(3), pp: 273-285, 2014.

[21] Liu, H., Wang, K., Chen, Z., Jordan, K., Luo, J., Deng, H., A Parallel Framewrok for Reservoir Simulators on Distributed-memory Supercomputers, SPE-176045-MS,SPE/IATMI Asia Pacific Oil & Gas Conference and Exhibition, Nusa Dua, Indonesia, 20 C 22 October, 2015.

[22] Wenchao Guan, Changhe Qiao, Hongxuan Zhang, Chen-Song Zhang, Meipeng Zhi, Zhifan, Zhenying Zheng, Weicai Ye, Yongdong Zhang, Xiaozhe Hu, Zheng Li, Chunsheng Feng, Yuesheng Xu, Jinchao Xu, On Robust and Efficient Parallel Reservoir Simulation on Tianhe-2, SPE-175602-MS, SPE Reservoir Characterisation and Simulation Conference and Exhibition, 14-16 September, Abu Dhabi, UAE, 2015.

[23] M. Wheeler, Advanced Techniques and Algorithms for Reservoir Simulation, II: The Multiblock Approach in the Integrated Parallel Accurate Reservoir Simulator (IPARS), The IMA Volumes in Mathematics and its Applications, Springer New York, pp: 9-19, 2002.