The Thesis Committee for Masoud Behzadinasab

Certifies that this is the approved version of the following thesis:

Development and Application of a Parallel Chemical Compositional Reservoir Simulator

APPROVED BY

SUPERVISING COMMITTEE:

Supervisor: _____

Ofodike A. Ezekoye

Co-supervisor: _____

Kamy Sepehrnoori

Development and Application of a Parallel Chemical Compositional Reservoir Simulator

by

Masoud Behzadinasab, B.S.E.

Thesis

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Engineering

The University of Texas at Austin

August 2015

Dedication

To my parents

for everything they have done for me

for their unconditional love and support

Acknowledgements

Accomplishing this study would not have been possible without the support and contribution of many individuals. It is a great honor for me to thank some of those many, to whom I owe my deepest gratitude.

I would like to express my sincere gratitude to my supervisors, Dr. Kamy Sepehrnoori and Dr. Ofodike Ezekoye for their guidance, advice, and support throughout this study. I am privileged to have had an opportunity to work with Dr. Sepehrnoori. He helped me get through difficult times and provided emotional and technical guidance during my life in Austin. Moreover, I would like to thank the staff of both departments of Mechanical Engineering and the Petroleum and Geosystems Engineering for their technical and administrative support. I am also grateful for the financial support provided by the sponsors of the Reservoir Simulation Joint Industry Project (RSJIP) at the Center for Petroleum and Geosystems Engineering at The University of Texas at Austin.

I would like to thank all my friends, colleagues and officemates at The University of Texas at Austin. I am very thankful to Soroosh Sadeh, Abolfazl Hashemi, Moein Falahatgar, and Mahsa Ghasemi for their support during my tough times. My special gratitude goes to Dr. Hamid Reza Lashgari for his technical supports and insightful comments. I also acknowledge Ederson Grein for his helpful discussions and his technical contribution to development of the code. I am also thankful to Dr. Chowdhury Mamun, David Ede and Shannon Ede for careful review and comments on my thesis. I am very grateful to my friends, Dr. Aboulghasem Kazeminia, Dr. Mohsen Rezaveisi, Alireza Sanaei, Ali Goudarzi, Mahdi Haddad, Javad Rashidi, Arjang Shahriari, Rambod Tabasi, Babak Nasouri, Ali Abouie, Mohsen Taghavifar, Reza Ganjdanesh, Pooneh Hosseini, Mahmood Shakiba, Arash Shushtarian, Arjang Gandomkar, Bruno Batista Fernandes,

Hector Barrios, Beibit Bissakayev, Jose Sergio Cavalcante, Wei Yu, Wensi Fu, Elif Ozdingis, Yifei Xu, Wan Wei, and many other friends who I have not mentioned their names for all their support and kindness during my life in Austin.

Above all, I am extremely grateful to my parents, Homa and Mohammad, and my brother, Saeed, for their endless love and support through my entire life. Also, I wish to express my gratitude to my uncle, Hamid Rad, for his indispensable support during my life in the US.

Abstract

Development and Application of a Parallel Chemical Compositional Reservoir Simulator

Masoud Behzadinasab, M.S.E.
The University of Texas at Austin, 2015

Supervisors: Ofodike A. Ezekoye
Kamy Sepehrnoori

Simulation of large-scale and complicated reservoirs requires a large number of gridblocks, which requires a considerable amount of memory and is computationally expensive. One solution to remedy the computational problem is to take advantage of clusters of PCs and high-performance computing (HPC) widely available nowadays. We can run large-scale simulations faster and more efficiently by using parallel processing on these systems.

In this research project, we develop a parallel version of an in-house chemical flooding reservoir simulator (UTCHEM), which is the most comprehensive chemical flooding simulator. Every physical feature of the original code has been incorporated in the parallel code. The simulation results of several case studies are compared to the original code for verification and performance of the parallelization.

The efficiency of the parallelization is evaluated in terms of speedup using multiple numbers of processors. Consequently, we improve the parallel efficiency to carry out the

simulations by minimizing the communications among the processors by modifying the coding. The speedup results in comparison to linear speedup (considering the ideal speedup) indicate excellent efficiency. However, using large number of processors causes the simulator speedup to deviate from linear and the efficiency to decrease. The reason for the degradation is that the time devoted to communication between the processors increases with number of processors.

To the best of our knowledge, the parallel version of UTCHEM (UTCHEMP) is the first parallel chemical flooding reservoir simulator that can be effective in running large-scale cases. While it is not feasible to simulate large-scale chemical flooding reservoirs with millions of gridblocks in any serial simulator due to computer memory limitations, UTCHEMP makes simulation of such cases practical. Moreover, this parallel simulator can take advantage of multiple processors to run field-scale simulations with millions of gridblocks in few hours.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

In this chapter, the motivation behind the work, the scope of the thesis, and the main objectives are presented. Furthermore, the outline of the thesis is described.

1.1 MOTIVATION

Over the last 30 years, the scale of reservoir simulation problems has increased from hundreds of cells to millions of cells. Moreover, simulation of complicated enhanced oil recovery (EOR) processes involves a considerable amount of computational time and memory. Traditionally, supercomputers and expensive workstations were used for such simulations. Supercomputers are capable of running considerably large-scale problems; however, they are not economical and also not accessible to all users.

In order to improve the computational performance of reservoir simulators, considerable effort has been employed. Multi-scale methods and better linear system solvers, among other techniques, have been shown to be very important for such improvement; however, as the code is optimized, it is very difficult to achieve considerable time reduction by simply modifying the algorithms. Hence, consideration of the hardware architecture comes into play in order to make reservoir simulators faster.

One of the main factors affecting hardware performance is the CPU frequency, which means the rate at which a processor can complete a processing cycle. More operations can be performed and also the algorithm will run faster as the frequency increases. Figure 1-1 shows the frequency of several processors through the years. Obviously, the current processor architecture is saturating; hence no remarkable performance gain may occur unless there is a radical architecture change. For this reason, computer manufacturers adopted the strategy of using several simple processors instead of a single and probably complex processor. Every processor works only on a part of the

problem. Then the time consumed by the most loaded processor determines the overall computational performance. As more processors are used, each processor receives a smaller portion of the problem. Eventually, a large time reduction may be achieved. Ideally, the code runs N times faster if N processors are involved.

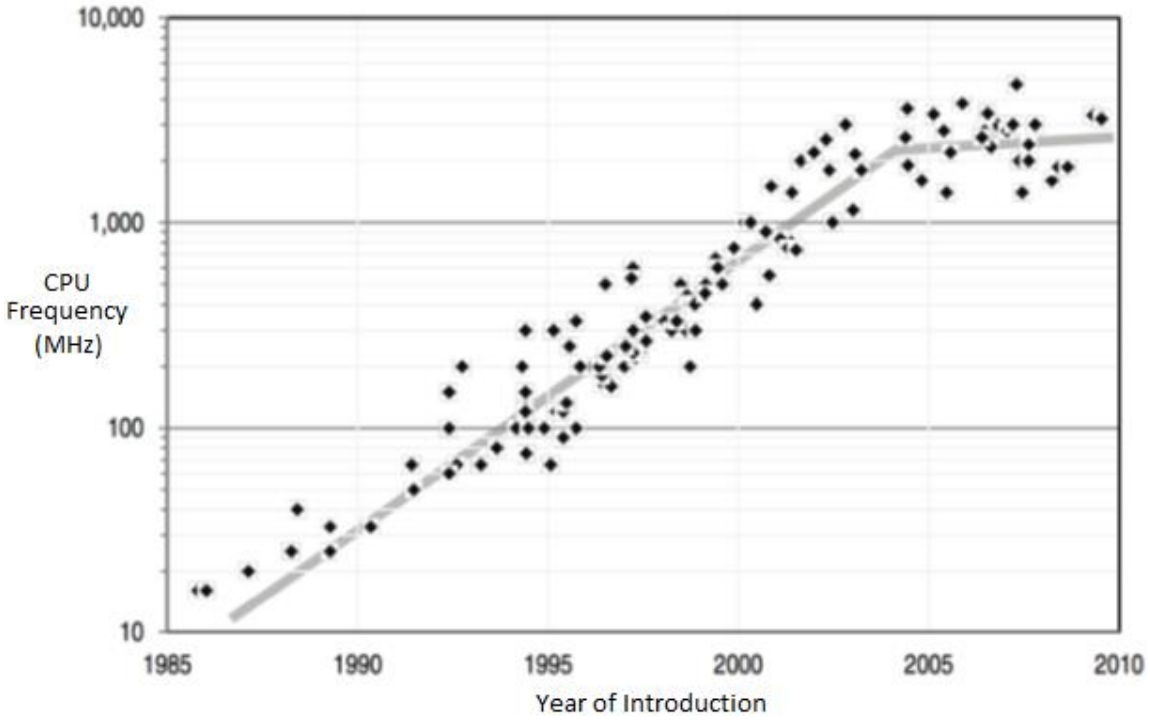

Figure 1-1-Frequency in MHz of several processors along the years (Shankland, 2012)

Development of parallel computer systems has attracted remarkable interest from reservoir simulation researchers in the application of parallel processing in this area. Parallel computers receive a lot of attention not only by having more than one processor, but also by providing a much larger available memory. Clusters of PCs have a memory module for each one of their nodes. Therefore, clearly, more memory is available when using more nodes. This is actually very interesting since a lot of simulation cases are not

practical because their models are so complex that it is not even possible to load them on a serial computer.

One should note, however, that using a parallel computer usually is not enough to profit from parallel computing. In fact, a code originally developed for serial computers may run even slower in a parallel computer since only a single – and probably simple – processor is allocated. It is also necessary that the code includes instructions about what operation each processor should execute. An egalitarian division of the operations optimizes the code performance. Otherwise, the overloaded processor dictates the overall performance.

Early attempts to apply parallel processing to reservoir simulation started in the late 1980s. Scott et al. discussed the application of parallel computers for reservoir simulation (Scott et al., 1987). Wheeler et al. (1990) presented a black oil simulator for distributed-memory parallel machines. Additional research was done on high performance parallel computing in reservoir simulation over the last 20 years.

1.2 RESEARCH OBJECTIVE

We started the development of UTCHEMP under the Integrated Parallel Accurate Reservoir Simulator (IPARS) framework (Parashar et al., 1997; Wang et al., 1997). Communications between the processors governed by MPI and memory allocations are optimized to achieve excellent parallel efficiencies.

Then we focus on verification of the parallel code against the original code. Several cases are studied to achieve this goal, such as water flooding, tracer tests, polymer flooding, surfactant/polymer flooding, ASP flooding, and gel treatment. In addition, efficiency of the parallel simulator is assessed in terms of speedup using various numbers of processors. Subsequently, we improve the coding and implementation in the simulator in order to

minimize the communications among the processors to improve the parallel efficiency to carry out the simulations.

Moreover, the old line-to-line format of the input file is modified to keyword-based format which makes usage of the simulator much simpler. In addition, other options are included, such as include files, inactive cells, and visual post-processing of the results using S3graph software (Sciencesoft, 2012).

1.3 REVIEW OF CHAPTERS

This thesis describes the development and application of a parallel chemical compositional reservoir simulator.

Chapter 2 provides the background for parallel processing: key definitions, review of parallel programming interfaces, and literature review of parallel reservoir simulation.

In Chapter 3, an overview of UTCHEM formulation is given.

In Chapter 4, we discuss the development methodology, new features, and capabilities of Parallel UTCHEM (UTCHEMP).

In Chapter 5, results for several case studies and parallel efficiency of the simulator are presented.

Chapter 6 presents the summary and the conclusions for the thesis and gives recommendations for future research.

Chapter 2: Background and Literature Review

In this chapter, we present the classification of algorithms, concept of parallel processing, parallel programming interfaces, and application of parallel processing in reservoir simulation.

2.1 ALGORITHM CLASSIFICATION

Generally speaking, algorithms can be classified based on task dependences (Grein, 2015):

1. Serial algorithms
2. Parallel algorithms
3. Serial-parallel algorithms (SPAs)
4. Nonserial-parallel algorithms (NSPAs)
5. Regular iterative algorithms (RIAs)

2.1.1 Serial algorithms

Serial algorithms involve sequential and ordered executions of tasks. The execution of a task must wait for the finalization of the previous one because of the data dependency

2.1.2 Parallel algorithms

On the other hand, in parallel algorithms, there is no data dependency between tasks. Hence, the tasks can be executed concurrently by several processors. The overloaded processor limits the overall performance of a parallel algorithm.

2.1.3 Serial-parallel algorithms (SPAs)

A serial-parallel algorithm involves grouped tasks in stages such that the tasks in each stage can be executed concurrently in parallel and the stages are executed sequentially. Clearly, an SPA becomes a parallel algorithm when there is only one stage.

Conversely, it becomes a serial algorithm when the number of tasks in each stage is one. Figure 2-1 shows an example of an SPA.

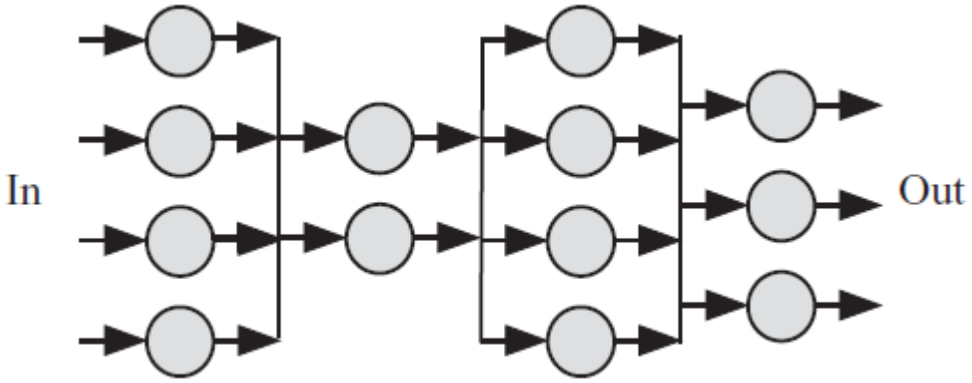

Figure 2-1-Example of a serial-parallel algorithm (Gebali, 2011)

2.1.4 Nonserial-parallel algorithms (NSPAs)

A nonserial-parallel algorithm does not fall into any of the above categories since it does not follow any pattern at all. Two types of constructs characterize an NSPA graph: nodes corresponding to the algorithm tasks and directed edges, which outline the direction of data flow amongst the nodes. Important information can be obtained from the graph, such as the amount of work to complete the algorithm (work), the maximum path length between any input node and any output node (depth), and the maximum number of nodes that can be processed in parallel (degree of parallelism).

2.1.5 Regular iterative algorithms (RIAs)

RIAs are those whose dependencies among the tasks show a fixed pattern, which might be very difficult to identify. Contrary to a serial algorithm, a parallel algorithm, or even an SPA, it is not trivial to explore the possible parallelization options of an RIA; however, they deserve special attention since they are very common in fields like signal, image and video processing, numerical simulation, and linear algebra applications.

2.2 THE CONCEPT OF PARALLEL PROCESSING

2.2.1 Definition

Parallel processing, in the context of computer science, is the simultaneous use of several processors to execute a program. Technically, the job is divided among different processors, and each of them works on its own task.

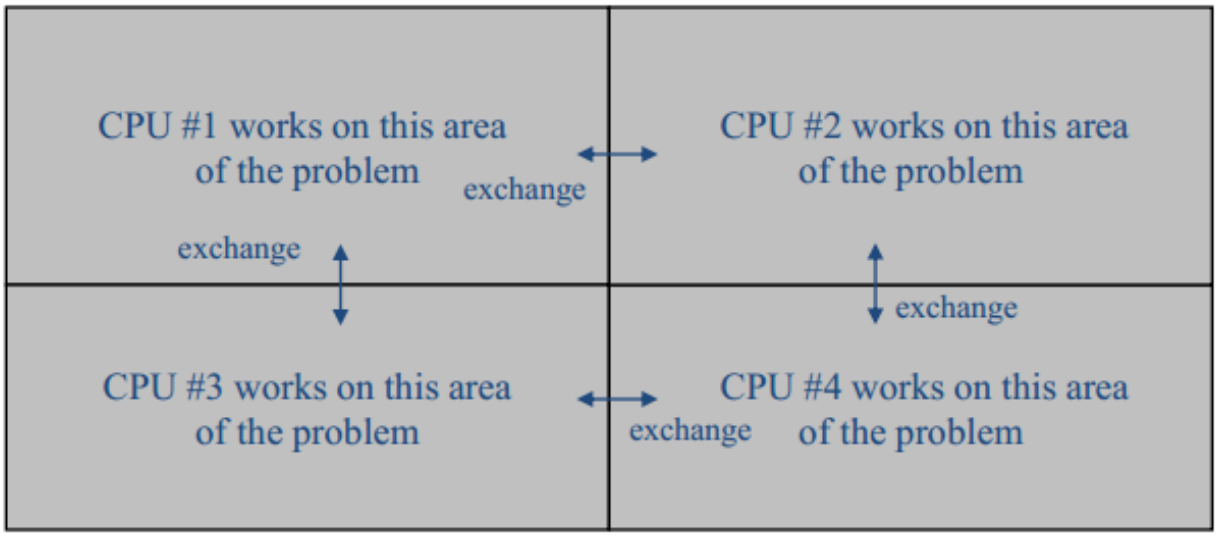

Figure 2-2-Definition of parallel processing (Willmore, 2012)

2.2.2 Purpose

Parallel computing can be beneficial in two ways. It can reduce wall-clock time, which is the time that elapses from the start to the end of a program. Moreover, we can take advantage of parallel processing in a memory allocation approach. It may not be possible to allocate large-scale problems in an ordinary computing node. Connecting multiple nodes together is cheaper than increasing memory of a single computing node.

Simulation of large-scale and complex problems with millions of gridblocks is required in the oil industry. As a consequence, parallel computing is very important in modern reservoir simulation.

2.2.3 Speedup and parallel efficiency

Speedup is one of the most common metrics to measure the advantages of parallel computing. It is defined as

$$S_N = \frac{T_1}{T_N} \tag{2.1}$$

where T_1 and T_N denote the computational time corresponding to the serial run (using one processor) and parallel run (using N processors), respectively. The computational time, in this context, is called wall-clock time.

In an ideal manner, if the code is fully parallelized, the communication time among processors and memory is negligible, and there is no overloaded processor, the speedup is linear; that is,

$$S_N = N. \tag{2.2}$$

However, in general, the speedup is sub-linear because the ideal conditions rarely happen. Normally, part of the code is serial, like input and output, which decreases the parallel efficiency. On the other hand, there are situations where we deal with super-linear speedup, i.e., the speedup is greater than the linear speedup. This might be the case where hardware issues such as memory bandwidth appear (Abate et al., 2001).

A good measurement of efficiency for a parallel implementation can be provided by speedup curve. This curve is generated when a problem is executed with different numbers of processors and the corresponding speedups are plotted versus the number of processors.

Another efficiency assessment is called parallel efficiency defined as

$$E_N = \frac{S_N}{N}. \tag{2.3}$$

Parallel efficiency is equal to 1, greater than 1, or less than 1 for linear, super-linear, and sub-linear speedup, respectively.

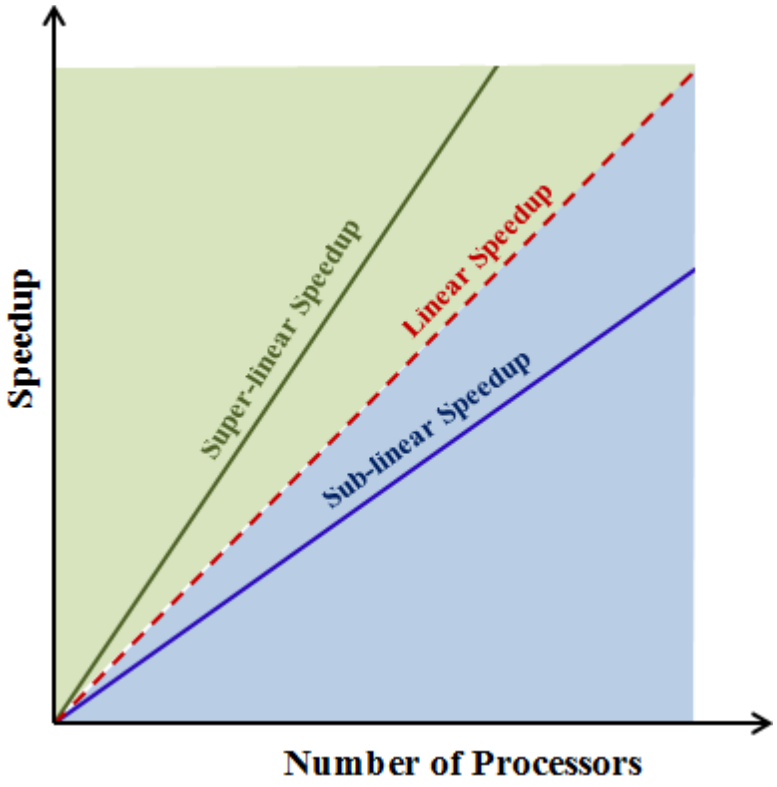

Figure 2-3-Speedup curve (Ghasemi Doroh, 2012)

2.2.4 Theoretical limits of parallel efficiency

In theory, parallel efficiency has some limitations. Part of an algorithm that cannot be parallelized decreases the efficiency, according to Amdahl's law (Amdahl, 1967). In other words, an algorithm is composed by a parallelizable fraction f and a serial fraction 1-f. This is the reason why linear speedup may not be gained. Considering that the execution of the parallelizable part is N times faster when using N processors, the computational time would be

$$T_N = f\frac{T_1}{N} + (1-f)T_1. \quad (2.4)$$

Hence, the theoretical speedup is

$$S_N = \frac{1}{f/N+(1-f)} \tag{2.5}$$

and the maximum speedup is found to be

$$S_{max} = \lim_{N\to\infty} S_N = \frac{1}{1-f} \tag{2.6}$$

Figure 2-4 depicts the speedup curve versus number of processors for different f.

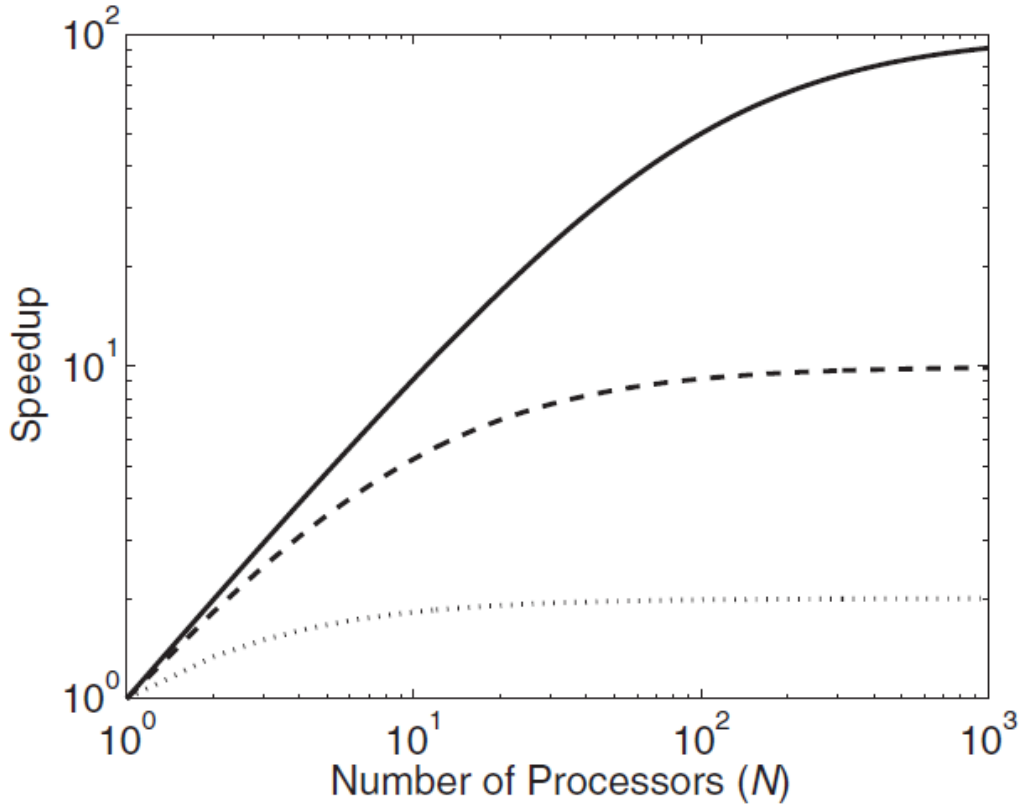

Figure 2-4-Speedup according to Amdahl's law. The solid line is for f=0.99; the dashed line is for f=0.9; and the dotted line is for f=0.5 (Gebali, 2011)

Speedup predictions made by Amdahl's law are pessimistic. Another common theoretical law, in this matter, is the Gustafson-Barsis'. Gustafson observed that as the problem size increases, the parallelism increases (Gebali, 2011). In the Gustafson-Barsis'

formula, the parallel computational time is taken as the reference. Then the execution time in serial would be

$$T_1 = (1 - f)T_N + fNT_N \tag{2.7}$$

giving the theoretical speedup as

$$S_N = 1 + f(N - 1) \tag{2.8}$$

As can be seen in Figure 2-5, speedup predictions from Gustafson-Barsis' law are much less pessimistic than Amdahl's law.

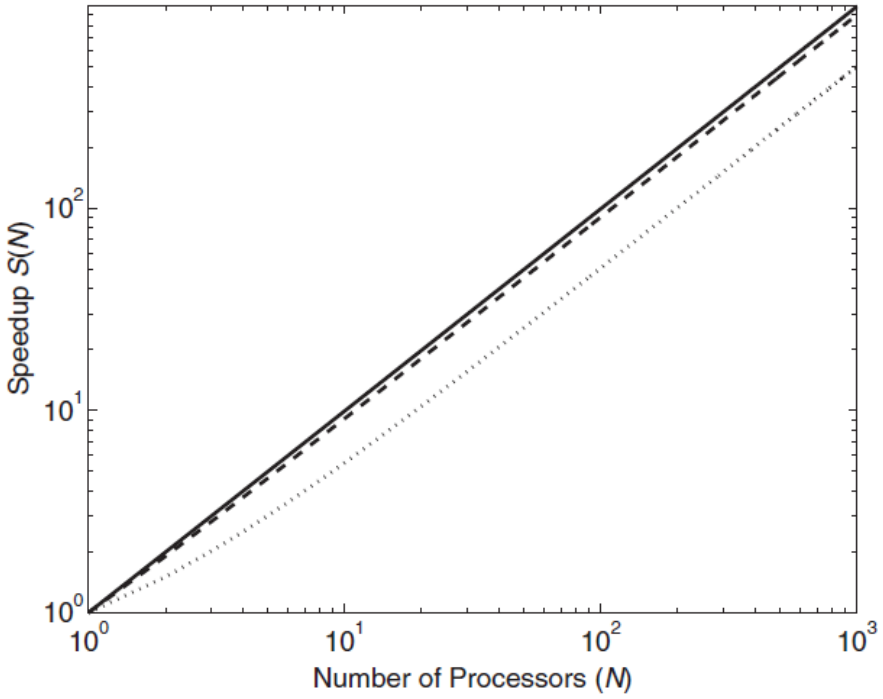

Figure 2-5-Speedup according to Gustafson-Barsis' law. The solid line is for f=0.99; the dashed line is for f=0.9; and the dotted line is for f=0.5 (Gebali, 2011)

2.3 PARALLEL ARCHITECTURES

Multiprocessing architecture refers to the way parallel processing can be performed. It has a significant effect on the manner used to make an algorithm parallel. A

processor's architecture should be chosen so that it is capable of performing the algorithm instructions assuring results' correctness. Moreover, an interconnection network is needed to enable the processors to communicate. This network, if not fast enough, could be the bottleneck for the software performance. Hence, reducing data exchange among processors is an important matter for the algorithm design if the interconnection network is known to have poor quality.

Parallel computers can be categorized in three main groups according to the multiprocessing architectures: shared, distributed, and hybrid memory.

2.3.1 Shared-Memory Architecture

In shared-memory architecture, all of the processors have access to the main memory and share it together. It allows them to communicate in an efficient manner via memory. This architecture, actually, is an extension of the single processor architecture. In the context of this thesis, processor alludes to CPU that is responsible for executing the assigned job. Figure 2-6 shows shared memory architecture.

There are pros and cons to use this architecture for parallelization of an algorithm. From the point of memory, a user-friendly way of programming is provided by the global shared memory. Also, data can be exchanged among the processors in a fast and uniform manner. The main disadvantage of shared memory architectures is that they are not scalable to large number of processors (Dongarra, 2003). Design of shared memory architectures becomes more challenging and expensive as the number of processors increases. Furthermore, a high memory machine can be very expensive.

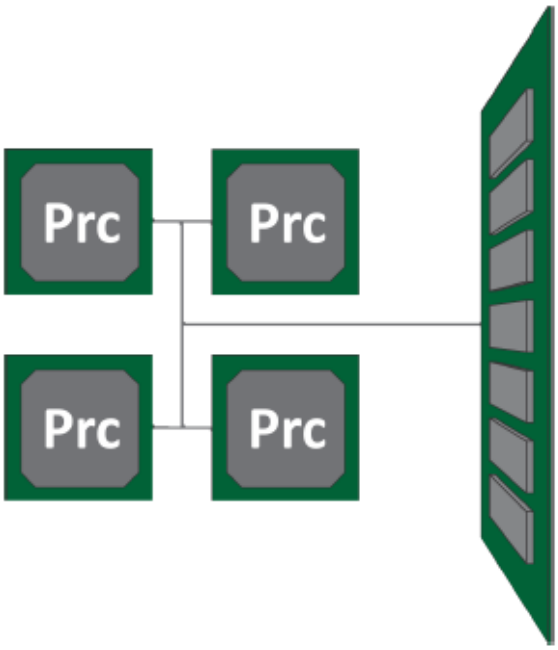

Figure 2-6-Shared memory architecture (Grein, 2015)

According to Gebali (2011), programming for shared memory parallel computers is not difficult. The programmer can treat the code as serial without considering the memory read operations which are hidden. On the other hand, memory write operations might require the data to be inaccessible until a thread has finished using it. Since processors simultaneously work and share the same memory storage, the programmer must ensure correct access to the global memory. Frequently, libraries based on OpenMP directives – discussed later – are used to handle synchronization and other affiliated operations.

2.3.2 Distributed-Memory Architecture

There are other multiprocessor systems in which all processors have their own memories. These systems are referred to as distributed-memory architectures. In these computers, an interconnection network is used to enable processors to communicate. In

order to improve the overall computational performance, it is important to favorably place the data among the memory modules. Consequently, fewer messages need to be sent among the processors. The Message Passing Interface (MPI) – discussed later – may be used as a language-independent message protocol. Figure 2-7 shows a distributed- memory architecture.

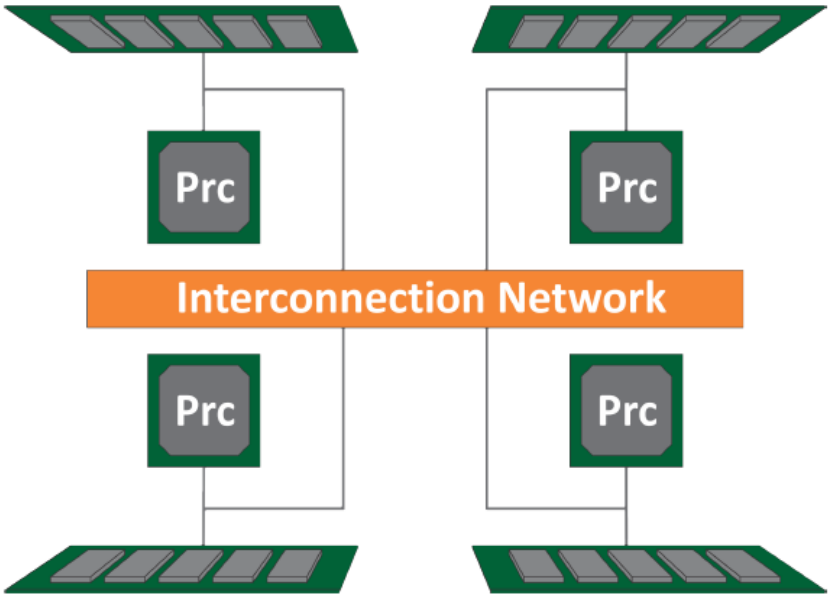

Figure 2-7-Distributed memory architecture (Grein, 2015)

2.3.3 Hybrid-Memory Architecture

Hybrid-memory systems profit from both shared and distributed memory computers. This architecture is similar to a distributed memory except that every node is a shared memory system. This approach enables the programmer to have high parallel efficiency within a node and then scale the program to a large number of processors. Figure 2-8 shows a hybrid-memory architecture.

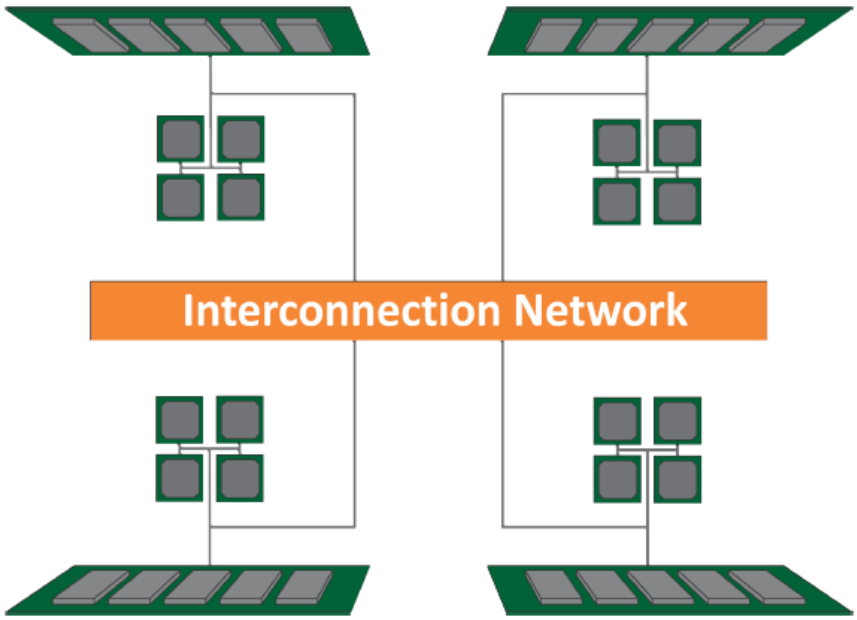

Figure 2-8-Hybrid memory architecture (Grein, 2015)

2.4 PARALLEL PROGRAMMING INTERFACES

An application programming interface (API) refers to a set of protocols, tools, and routines that can be beneficial in software applications development. The most common APIs, in the context of parallel programming, are the Message Passing Interface (MPI) and the Open Multi Processing (OpenMP). The former can be used in both distributed and shared-memory systems, while the latter has been designed to use in shared-memory parallel computers.

2.4.1 OpenMP

OpenMP has a set of compiler directives and environmental variables, as well as a runtime library, and can only be used in shared memory parallel computers. Parallelizing sequential programs implemented in Fortran, C, or C++ may be done through OpenMP.

OpenMP is considered a high level API, which means that many technical details, such as flow control and data decomposition, are left to the compiler and the programmer does not need to worry about them. It is not mandatory, in OpenMP, to make the whole code in parallel. This API allows the programmer to take advantage of its directives to parallelize the desired parts of the code. The rest of the code remains in sequential format. If OpenMP is not supported by a compiler, the directives will be interpreted as comments and thus ignored. Therefore, the application is parallel and sequential at the same time. Also it is noteworthy that the collection of OpenMP directives is relatively small. In other words, learning a whole new language is not needed in order to use OpenMP (Padua, 2011).

2.4.2 MPI

The Message Passing Interface (MPI) is a standard developed to improve the application of message passing mechanism. The dominant parallel programming language, nowadays, is MPI (Padua, 2011). MPI can be used in both shared and distributed memory systems because it treats communications among processors explicitly.

The processors, in the message passing mechanism, communicate by sending messages which is a two-sided operation; a processor sends a message, while another one receives it (Barney,. An MPI message is composed of two parts: the envelope and the message body. The former has four parts; *source* that is the processor sending the message, *destination* that is the processor receiving the message, the *communicator* that is the group of processors containing both source and destination, and *tag* which is the message classification. The latter term consists of three parts: *buffer* that is the data to be sent, *datatype* that is type of the message data, and *count* or number of items in the buffer.

2.5 PARALLEL PROCESSING APPLICATION IN RESERVOIR SIMULATION

Advanced reservoir simulation requires very detailed physical and geological models. This involves large numbers of gridblocks and is computationally expensive. Traditionally, powerful workstations and supercomputers were used to perform large- scale simulations. However, nowadays, as a result of a wide availability of clusters of PCs and also high performance computing (HPC) centers, an application of parallel programming provides a cheaper solution to the field-scale simulations.

Usually, most parts of a reservoir simulator are written in, or modified to, a parallel approach, and only a relatively small part of the code remains in serial format so that an excellent parallel efficiency is achievable. Moreover, roughly speaking, the time devoted to communication among the processors is relatively small compared to the computational time.

One of the first noticeable efforts in application of parallel processing in reservoir simulation is the parallel processing approach presented by Scott et al. (1987). They claimed that the most time consuming part of a simulator is its solver, and focused on parallelization of the solver. Also an application of distributed memory systems to the solution of compositional simulations was introduced by Killough et al. (1991). Moreover, a three dimensional parallel implicit reservoir simulator was developed by Wheeler et al. (1990). Excellent performance of parallel reservoir simulation was indicated by all of the aforementioned works. Therefore, the subject remained interesting to most of the simulator developers. Wang et al. (1997) presented a fully implicit parallel compositional reservoir simulator; they developed it using IPARS framework. A scalable parallel multi-purpose reservoir simulator was also developed by Chien et al. (1993).

Although a parallel simulator can be very advantageous, the development of such program may be very complicated (Schiozer, 1999). It could be easy to make a code

parallel using OpenMP, but its application is limited to one computing node. Parallelization using MPI, contrarily, is more complicated, but is not limited to one computing node. As a consequence, in the context of parallel reservoir simulators, MPI parallelization would be a common recommendation, and MPI was used for development of the code for this thesis.

Chapter 3: General Description of UTCHEM and UTCHEMP

3.1 INTRODUCTION TO UTCHEM

UTCHEM is a three-dimensional, multi-component, 4-phase (water, oil, microemulsion, and gas), compositional, variable temperature reservoir simulator developed at the University of Texas at Austin. It can be used to model chemical flooding processes for enhanced oil recovery purposes. This simulator takes into account complex phase behavior, chemical, and physical transformations and heterogeneity of the porous medium (User's Guide for UTCHEM-9.0, July 2000; Lashgari, 2014). Some of the major physical phenomena modeled by UTCHEM are as the following:

- Diffusion
- Dispersion
- Dilution Effects
- Adsorption for oil, surfactant, and polymer
- Capillary pressure
- Relative permeability
- Interfacial tension
- Hysteresis in relative permeability and capillary pressure
- Cation exchange
- Capillary trapping
- Aqueous reactions
- Phase density
- Dissolution/precipitation
- Compositional phase viscosity
- Phase behavior (pseudoquaternary)
- PH dependent surfactant adsorption
- Partitioning of chemical species between oil and water
- In-situ generation of surfactant from acidic crude oil
- Organic biodegradation capability
- Multiple organic species
- Dual porosity option for simple phase tracer flow
- Gel properties
- Tracer properties
- Temperature dependent properties
- Gas mobility reduction due to foam
- Mixed-wet oil/water capillary pressure and relative permeability

UTCHEM is applicable to groundwater:

- NAPL spill and migration in both saturated and unsaturated zones
- Partitioning inter-well test in both saturated and unsaturated zones of aquifers
- Remediation using surfactant/co-solvent/polymer
- Remediation using surfactant/foam
- Remediation using co-solvents
- Bioremediation
- Geochemical reactions (e.g., heavy metals and radionuclides)

In addition, its oil reservoir applications are:

- Waterflooding
- Single well, partitioning inter-well, and single well wettability tracer tests
- Polymer flooding
- Profile control using gel
- Surfactant flooding
- High pH alkaline flooding
- Microbial EOR
- Surfactant/foam and ASP/foam EOR

3.2 SOLUTION STRUCTURE IN UTCHEM

The solution method implemented in UTCHEM is IMPEC type. It means that pressure is solved implicitly and concentrations, saturations, and temperature are solved explicitly. The main solution procedure of UTCHEM at each time step is structured as the following:

- First, the pressure equation is solved implicitly. Pressure values of grid blocks get updated by solving a large sparse matrix. The matrix elements are determined by taking concentrations, saturations and other physical properties at the previous time step.
- After solving the pressure equation, other equations, such as concentrations, saturations, and energy balance equations, are solved explicitly using the updated pressure values.

The solution structure in UTCHEM is shown simplistically in Figure 3-1.

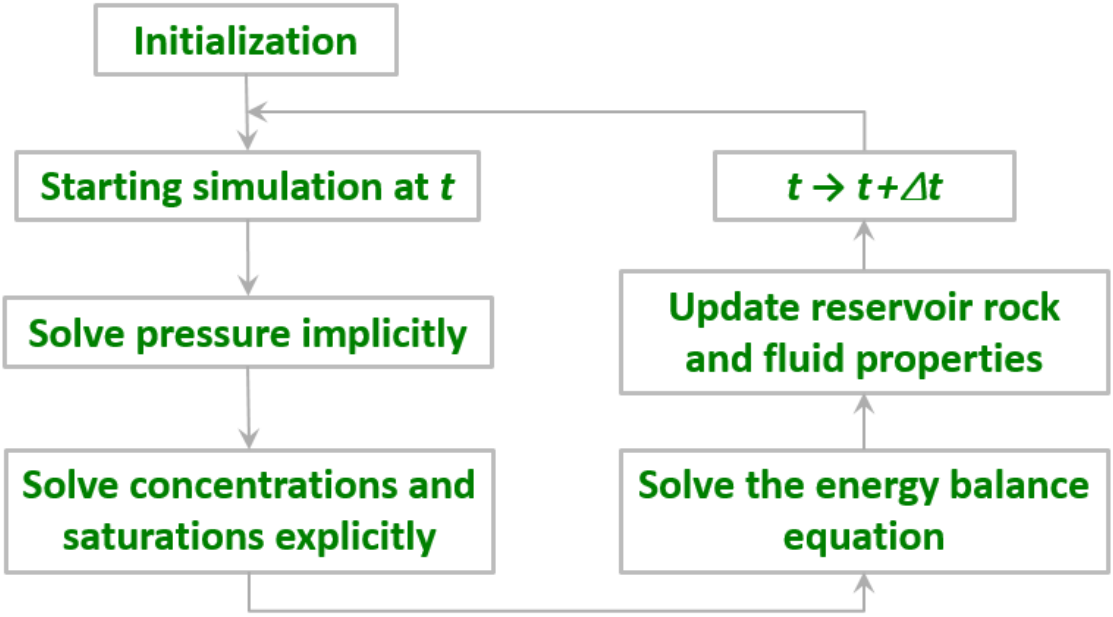

Figure 3-1 Solution structure in UTCHEM

3.3 MATHEMATICAL FORMULATION IN UTCHEM

The aqueous phase pressure, the concentrations, and the energy equations are the main equations to be solved (Technical Documentation for UTCHEM-9.0, July 2000).

The balance equations are as follows:

1. Mass balance equation for each species.
2. Aqueous phase pressure is calculated by an overall mass balance on volume-occupying components (water, oil, surfactant, co-solvent, and air). The other phase pressures are computed by adding capillary pressure.
3. Energy balance equation.

3.3.1 Mass Conservation Equations

The main assumption imposed on governing the flow equations are local thermodynamic equilibrium except for tracers and dissolution of organic component, slightly compressible soil and fluids, immobile solid phases, ideal mixing, Fickian dispersion, and Darcy's law. The boundary conditions implemented are no flow and no dispersive flux across the impermeable boundaries.

The mass equation for component K in a porous media in association with Darcy's law is given by

$$\frac{\partial}{\partial t}\left(\phi \tilde{C}_k \rho_k\right) + \vec{\nabla}.\left[\sum_{l=1}^{n_p} \rho_k \left(C_{kl}\vec{u}_l - \vec{\bar{\bar{D}}}_{kl}\right)\right] = R_k \tag{3.1}$$

where $\tilde{C}_k$ is the overall volume of component k per unit pore volume, i.e., overall concentration of k in all phases including the adsorbed phases.

$$\tilde{C}_k = \left(1 - \sum_{k=1}^{n_{CV}} \hat{C}_k\right) \sum_{l=1}^{n_p} S_l C_{kl} + \hat{C}_k \qquad for\ k = 1, \dots, n_c \tag{3.2}$$

n_{CV} is the total number of volume-occupying components (water, oil, surfactant, gas); n_p is the number of phases; $\hat{C}_k$ is the adsorbed concentration of species k; and ρ_k is the density of pure component k at a reference pressure P_R relative to its density at reference pressure P_{R0}. Ideal mixing and small and constant compressibilities C_k^o are assumed.

$$\rho_k = 1 + C_k^o (P_R - P_{R0}) \tag{3.3}$$

Porosity (ϕ) is also assumed to change linearly with pressure according to the expression

$$\phi = \phi_0\big(1 + C_r(P - P_{R0})\big) \tag{3.4}$$

where ϕ_0 is the referential porosity, evaluated at the reference pressure, and C_r is the rock compressibility presumably constant.

Fickian form is used to model the dispersive flux

$$\vec{\vec{D}}_{kl,x} = \phi S_l \vec{\vec{K}}_{kl}.\vec{\nabla}\mathrm{C_{kl}} \tag{3.5}$$

The dispersion tensor $\vec{\vec{K}}_{kl}$ which contains molecular diffusion is calculated as the following.

$$\vec{\vec{K}}_{klij} = \frac{D_{kl}}{\tau}\delta_{ij} + \frac{\alpha_{Tl}}{\phi S_l}|\vec{u}_l|\delta_{ij} + \frac{(\alpha_{Ll}-\alpha_{Tl})}{\phi S_l}\frac{u_{li}u_{lj}}{|\vec{u}_l|} \tag{3.6}$$

where τ is the tortuosity factor with the definition of being a value greater than one; α_{Ll} and α_{Tl} are phase l longitudinal and transverse dispersivities; u_{li} and u_{lj} are the components of Darcy flux of phase l in directions i and j; and δ_{ij} is the Kronecker delta function. The magnitude of vector flux for each phase is calculated as

$$|\vec{u}_l| = \sqrt{(u_{xl})^2 + \big(u_{yl}\big)^2 + (u_{zl})^2} \tag{3.7}$$

Using Darcy's law to find the phase flux as

$$\vec{u}_l = -\frac{k_{rl}\vec{\vec{k}}}{\mu_l}.(\vec{\nabla}P_l - \gamma_l\vec{\nabla}h) \tag{3.8}$$

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

The source term R_K in Equation (3.1) is composed by all rate terms for a particular species and can be expressed as

$$R_k = \phi\sum_{l=1}^{n_p} S_l\, r_{kl} + (1-\phi)r_{ks} + Q_k \tag{3.9}$$

where r_{kl} and r_{ks} are the reaction rates for species k in phase l and solid phase s, respectively, and Q_k is the injection/production rate for component k per bulk volume.

3.3.2 Pressure Equation

Summing up the mass balance equations over volume-occupying species and also using the definition of capillary pressure develop the pressure equation.

$$\phi C_t \frac{\partial P_1}{\partial t} + \vec{\nabla}.\vec{\vec{k}}.\lambda_{rTc}\vec{\nabla}P_1 = -\vec{\nabla}.\sum_{l=1}^{n_p}\vec{\vec{k}}.\lambda_{rlc}\vec{\nabla}h + \vec{\nabla}\sum_{l=1}^{n_p}\vec{\vec{k}}.\lambda_{rlc}\vec{\nabla}P_{cl1} + \sum_{k=1}^{n_{CV}} Q_k \qquad (3.10)$$

where $\lambda_{rlc} = \frac{k_{rl}}{\mu_l}\sum_{k=1}^{n_{CV}}\rho_k C_{kl}$ is relative mobility of phase l and $\lambda_{rTc} = \sum_{l=1}^{n_p}\lambda_{rlc}$ is the total relative mobility. C_t, total compressibility, is the volume-weighted sum of the rock matrix (C_r) and component compressibilities (C_k^o).

$$C_t = C_r + \sum_{k=1}^{n_{CV}} C_k^o \tilde{C}_k \qquad (3.11)$$

3.3.3 Energy Conservation Equation

The main assumption in the derivation of the energy conservation equation is that energy is a function of temperature only. Moreover, only heat conduction and advection contribute to the energy flux in reservoir or aquifer.

$$\frac{\partial}{\partial t}\left[(1-\phi)\rho_s C_{vs} + \phi\sum_{l=1}^{n_p}\rho_l S_l C_{vl}\right]T + \left(\vec{\nabla}.\sum_{l=1}^{n_p}\rho_l C_{pl} u_l T - \lambda_T\vec{\nabla}T\right) = q_H - Q_L \qquad (3.12)$$

where C_{vs} and C_{vl} are the rock and phase l heat capacities at constant volume, T is the reservoir temperature, C_{pl} is heat capacity of phase l at constant pressure, and λ_T is thermal conductivity. All of these parameters are assumed to be constant. Also q_H is the enthalpy source term per bulk volume and Q_L stands for the heat loss to underburden and overburden formations.

3.4 INTRODUCTION TO UTCHEMP

MPI is used to handle the parallel version of UTCHEM, named UTCHEMP. Hence, UTCHEMP can be operated in distributed, shared, or hybrid memory architectures. IPARS framework (Parashar et al., 1997) is the basis for development of this parallel code, and the code has been optimized in terms of parallel efficiency. All of the physical and numerical features of the original serial code are incorporated in the parallel code.

3.5 METHODOLOGY IN UTCHEMP

The approach used is based on parallelizing the part that has the highest computational cost. Clearly, this part is grid-related operations. This fact dictated using an approach based on domain decomposition. Gridblocks are divided among processors in an optimized manner such that the processors have roughly the same number of gridblocks. Each of the processors works only on the gridblocks in its domain, while the necessary communications are provided by MPI functions. Since all of the implementations are based on MPI, UTCHEMP can operate in both clusters of PCs and personal computers.

3.5.1 Domain decomposition

This parallelization work is only for structured grids. Dividing a structured domain is much easier than an unstructured one because the structured grid topology is strictly dependent of the coordinate system. In this work, grids are divided based on the y direction. In an (i,j,k) system, j_{m^-} and j_{m^+} are defined to specify the range of the gridblocks belonging to processor m in y direction, i.e., if a gridblock's coordinates are (i_x, j_x, k_x) and $j_{m^-} \leq j_x \leq j_{m^+}$, the gridblock belongs to the domain of processor m.

It is important to note that a grid division strictly based on the y direction may not be optimized. So for optimization purposes, values of j_{m^-} and j_{m^+} may change according to the k-coordinate. For instance, if a 5x5x2 model is simulated using two processors, the

first processor may take the first three gridblocks of y direction in the first horizontal layer but only two in the last one. Then each processor works on 25 gridblocks, and the grid is equally divided. An example of grid division is shown in Figure 3-2.

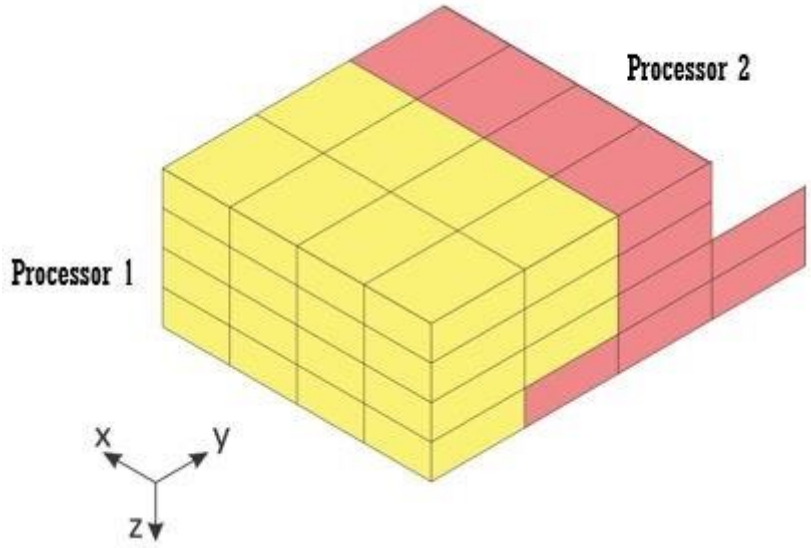

Figure 3-2 Example of grid division

3.5.2 Variables and Indexing

In the serial code, UTCHEM, NBL is the variable used to store the total number of gridblocks. However, in the parallel code, this variable stores the number of local active gridblocks (inactive gridblocks will be discussed later in this chapter).

Variables of UTCHEMP can be categorized into two groups; one consists of scalars or arrays that are not grid-dependent, and the other includes grid-dependent arrays. These grid-dependent arrays can be divided in two groups. The first group includes those that are not used in a stencil computation (computation that requires values at neighboring grid blocks). These arrays are allocated linearly to NBL. The second group consists of arrays used in stencil computations. For such arrays, a continuous indexing is not viable, and they are allocated in 3 dimensions (x,y,z) and include inactive gridblocks. Extra storage is

needed to store the values inside neighboring gridblocks that belong to a next or previous processor. The additional gridblocks are called ghost cells, and no computation is performed for them. Every time stencil computations are performed, MPI directives update the values at ghost cells. An example of ghost cells is shown in Figure 3-3.

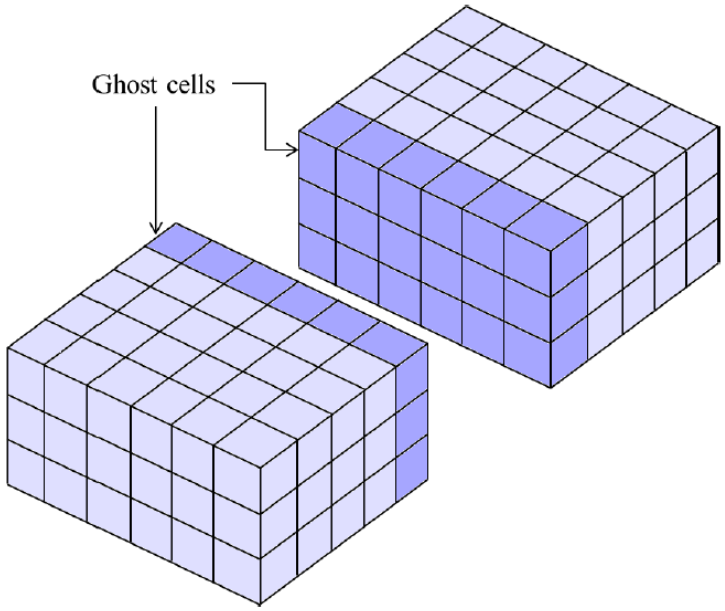

Figure 3-3 Example of ghost cells in a 6x8x3 grid

The first group of grid-dependent gridblocks with continuum indexation, mentioned earlier, would be labeled here as type 1, while the other group would be labeled as type 2. The (x,y,z) indices of a type 2 array are named I1, J1, and K1. For any processor, range of I1 is IL1-IL2, K1 is KL1-KL2, and J1 is JL1V(K1)- JL2V(K1). The difference in the latter reflects the nature of grid division, which was indicated earlier.

Every loop in UTCHEM that includes type 2 arrays should be modified in UTCHEMP, as illustrated in Figure 3-4. I is the continuum index used for type 1 arrays. Since the loop is coordinate-based, I index may be accessed using the function IJKPOS if needed. KEYOUT is also a variable indicating the type of a gridblock, and it is used to

drop inactive gridblocks from calculations. It is a type 2 array, and its value may be 1,0,or -1. If it is 1 for a gridblock, it means the gridblock is active. If a gridblock is inactive, KEYOUT has the value of 0 for that gridblock. KEYOUT -1 also refers to ghost cells. I1, J1, K1, and I are local to a processor, but they can be converted to global indices if necessary.

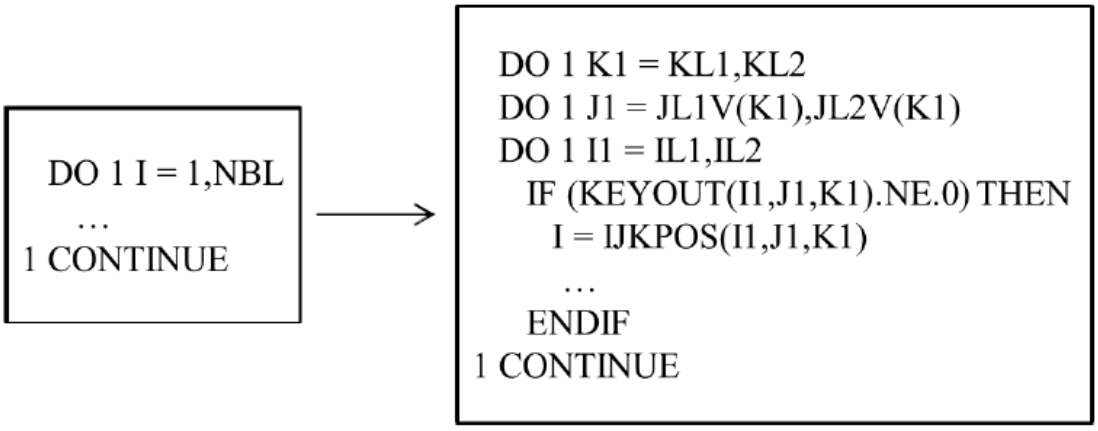

Figure 3-4 Modification of loops including type 2 arrays

3.5.3 Communications

Although a processor's task includes updating ghost cells, well management, solving the linear system of equations for pressure, time step selection, and so on; processors need to communicate for input/output matter. MPI routines handle all the communications.

3.6 IPARS FRAMEWORK

The Implicit Parallel Accurate Reservoir Simulator (IPARS) provides the framework for this parallelization (Parashar et al., 1997; Wang et al., 1997). This framework has several routines that facilitate execution of parallel-related operations. For instance, one of its routines updates type 2 array values, another one divides the gridblocks based on number of processors, and so on. Organization of UTCHEMP on this framework is shown in Figure 3-5.

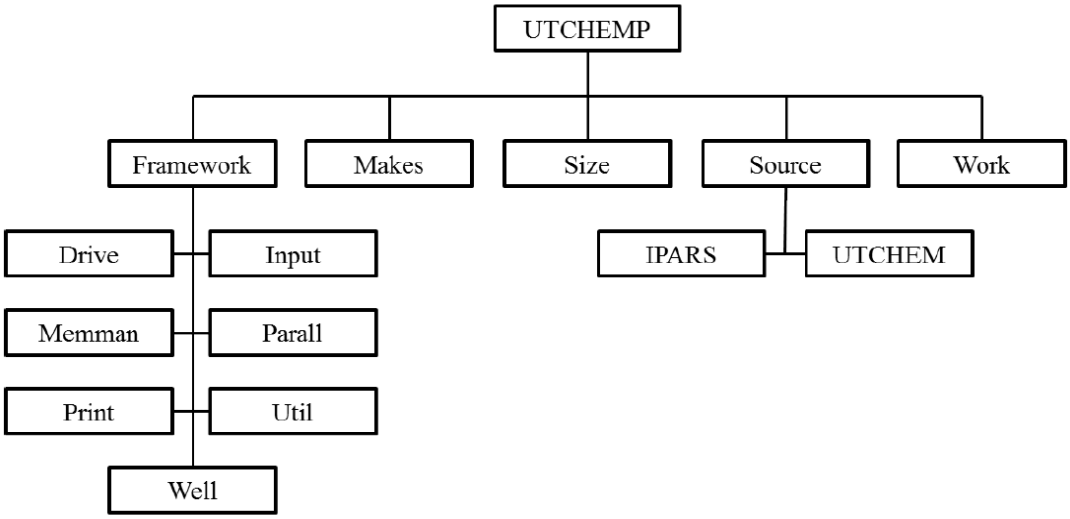

Figure 3-5 UTCHEMP organization on IPARS framework

The folder Framework keeps the main routines of IPARS. The main subroutine, named IPARS, is placed in the folder Drive. This subroutine directly or indirectly calls all the other subroutines of UTCHEMP. Input is the folder used to keep the subroutines responsible for reading the input file and initializing the simulation. The sub-folder Memman includes several functions written in C++ , mostly managing the dynamic allocation of arrays. Subroutines for processors' communications are placed in Parall. Print, Util, and Well also contain subroutines for exporting results to output files, general purpose subroutines, and well-related subroutines, respectively. There are two sub-folders

in folder Source, IPARS and UTCHEM. The latter includes almost all the subroutines that are being used in the serial UTCHEM as well as some new ones, while the former contains some additional IPARS subroutines mainly used for initializing the simulation.

The algorithm flowchart for UTCHEMP is illustrated in Figure 3-6. At the beginning of the simulation, the IPARS framework is initialized and input files are read. Subroutines INOUT and WELREAD are responsible for reading the input files. The next step would be to perform some additional initializations, mainly the variables originally presented in UTCHEM. Then the subroutine AAMAIN is called. This is the main subroutine in the serial UTCHEM that directly or indirectly calls all the other subroutines; however, in UTCHEMP, AAMAIN includes only the part of the code that is used to execute a time step. Parts of the code executed before or after any time step are moved to the IPARS framework or the new subroutines AAMAINI and AAMAINO. Finally, some additional output files are generated and the simulation is finalized.

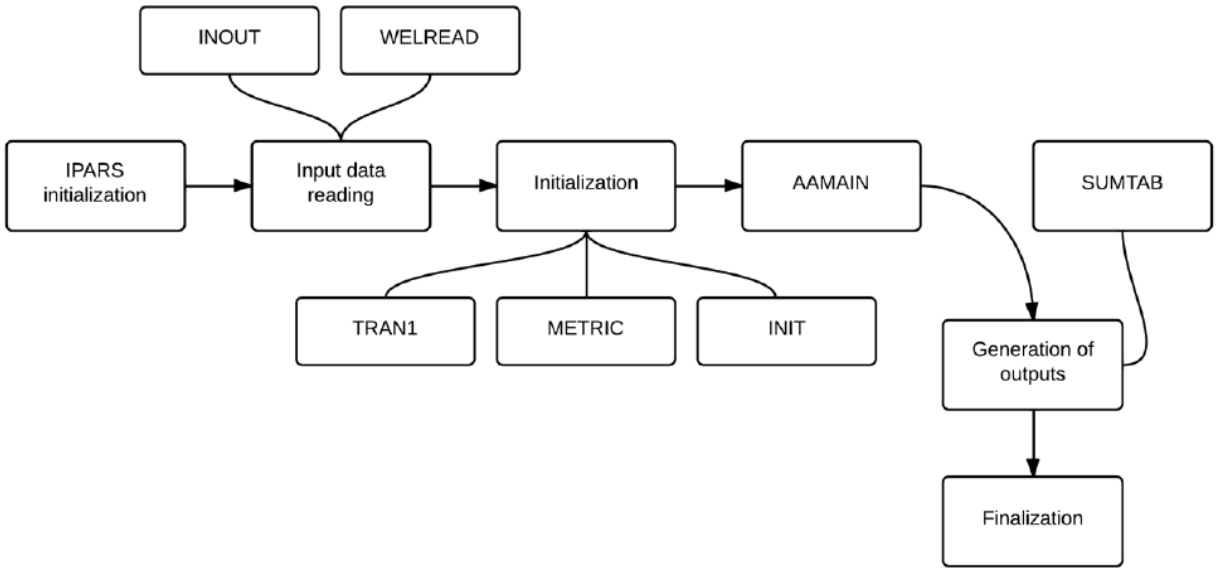

Figure 3-6 Algorithm flowchart for UTCHEMP

3.7 NEW FEATURES OF THE PARALLEL VERSION

The new features implemented in the parallel version are as follows:

- Optimized memory: UTCHEMP is able to simulate chemical EOR on large reservoirs up to 312,500 cells (8 components, 4 phase, 2 wells, ASP slug injection) using one processor on the Petros cluster, where each computing node has 16 GB of memory.
- PETSc, a high-performance parallel solver package, is used to solve the pressure equation in UTCHEMP.
- Inactive gridblocks are excluded in the parallel code.
- A new keyword-based input file format is included.

3.7.1 Inactive gridblocks treatment

There is a possibility that some parts of reservoir that have very low permeability and porosity such that fluid flow through them would be negligible. Gridblocks representing those parts are referred to as inactive gridblocks. Inactive gridblocks facilitate using structured grids. An example of inactive gridblocks is shown through Figure 3-7. These gridblocks make the grids closer to the reservoir geometry. Inactive gridblocks are not excluded in the serial UTCEHM and, instead, they are specified with very low values for porosity and permeability and are considered as water saturated. They are also involved in almost all the computations, which sometimes are physically inconsistent.

In contrast to the serial version, inactive gridblocks are totally excluded in this work, and they are not involved in any computation. For type 1 arrays, this is automatically done since their size is equal to the number of active gridblocks (the variable KEYOUT is used to prevent storing values for type 1 arrays from inactive gridblocks). For type 2 arrays, on the other hand, the value of KEYOUT is verified before every computation involving those kinds of arrays, as illustrated in Figure 3-4.

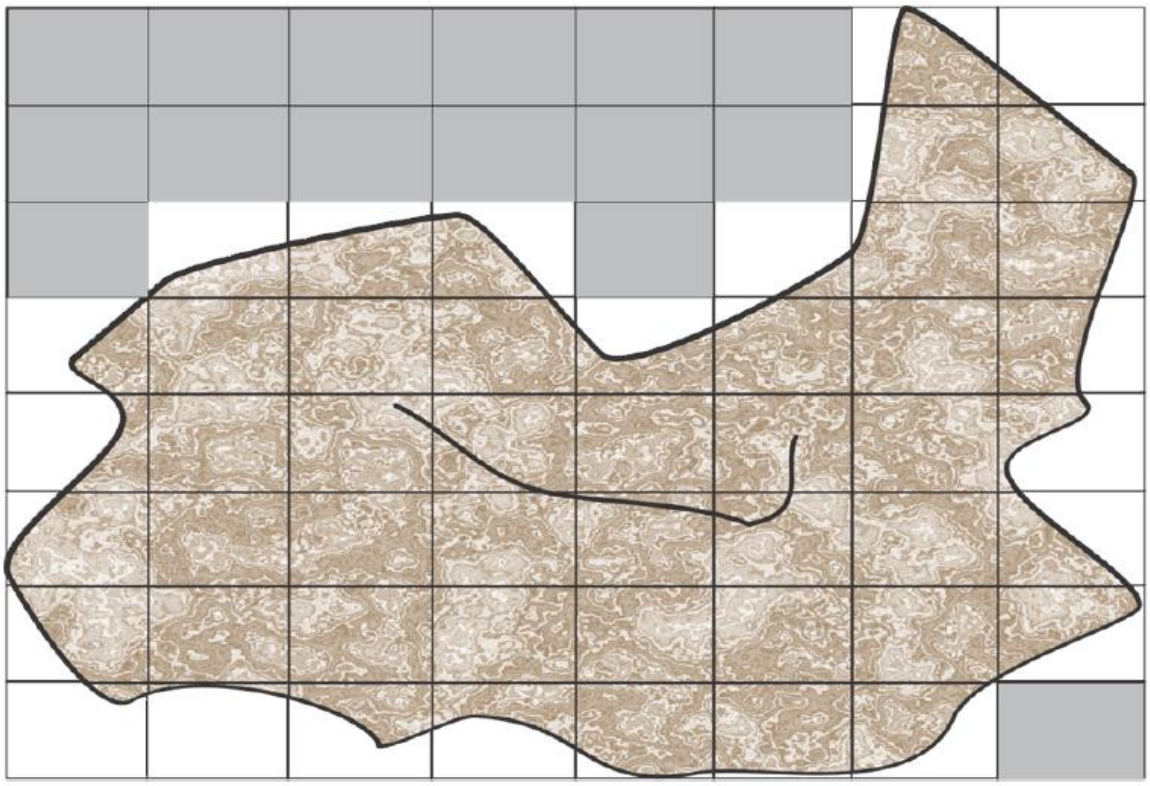

Figure 3-7 Example of inactive gridblocks. Inactive gridblocks are colored gray.

3.7.2 New input file format

The IPARS framework enables us to implement a much more flexible input file format compared to the original input file format. The older format is based on line-to-line reading, and there is one place where the datum of a certain variable can be placed. On the other hand, the new format is based on keywords. Each variable in the input file is associated to a keyword, and the keyword may be placed in the input file wherever the user desires.

Chapter 4: Verification

The main purpose of this chapter is to present several case studies in order to validate the simulator. In this regard, small cases are run by UTCHEMP using different numbers of processors and then the results are compared with the results for the original serial code. In order to make sure that the entire code is verified, different physical features from the simulator are tested.

The solver implemented in UTCHEMP to solve the linear system resulted from equation for the aqueous phase pressure is PETSc. However, UTCHEM uses a Jacobi Preconditioned Conjugate Gradient solver, named CHECK, in order to solve this equation. In order to make the comparison coherent, same values for tolerance are applied to both of these linear system solvers to check convergence.

Simulations for the serial UTCHEM were executed on a personal computer that has 16 GB of RAM and Intel core i7 CPU, while simulations for UTCHEMP were executed on Texas Advanced Computing Center (TACC), Lonestar Linux Cluster. This cluster consists of 1,888 compute nodes, with 12 processors per node. 24 GB of RAM is available for each node, and the frequency of cores is 3.33 GHz. Intel Fortran is the compiler used in this cluster, with the optimization flag O3. Consequently, due to the code optimizations because of that flag, it is normal if the obtained results do not perfectly match.

4.1 Case Study 1: Water Flooding

The first case is to study a water flooding process. Table 4-1 describes the reservoir and fluid properties. There are 17 wells operating in this reservoir; 4 injectors and 13 producers, and all of them are constrained to constant flow rate. The reservoir model is shown in Figure 4-1. The simulation time is 2526 days, and operation of all the wells does not change throughout the simulation.

Table 4-1 Reservoir and fluid properties description (Case study 1)

Case 1 (Water flooding)			
Dimensions (ft)	Length	3100	
	Width	4500	
	Thickness	25	
Number of gridblocks		4,185 (31x45x3)	
Number of components		5	
Max. number of phases		4	
Porosity (fraction)		0.1371	
Lateral permeability (md)	Top layer	75	
	Middle layer	500	
	Bottom layer	1250	
Vertical permeability (md)		5	
Irreducible water saturation (fraction)		0.1	
Residual oil saturation (fraction)		0.45	
Water viscosity (cp)		0.5	
Oil viscosity (cp)		2.7	
Initial reservoir pressure (psi)		150	
Number of Wells		17	4 Injectors
			13 Producers
Simulation time (days)		2526	

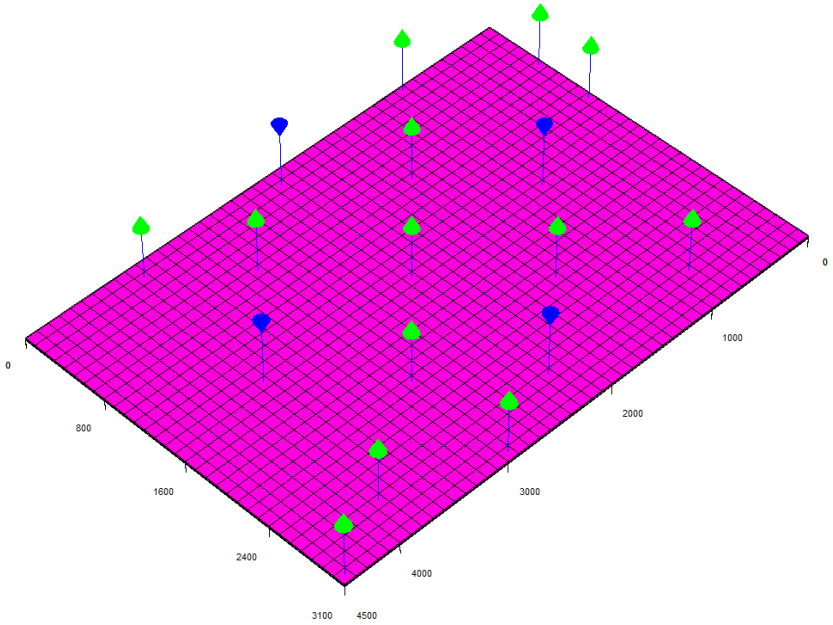

Figure 4-1 Reservoir model (Case study 1). Blue markers represent injection wells, and green ones are producers.

This case was simulated by UTCHEMP using 1, 2, 4, 8, 16, 32, and 64 processors. The results are perfectly matched with the results for UTCHEM. Figures 4-2, 4-3, and 4-4 provide the results for average reservoir pressure, oil cut, and recovered oil, respectively.

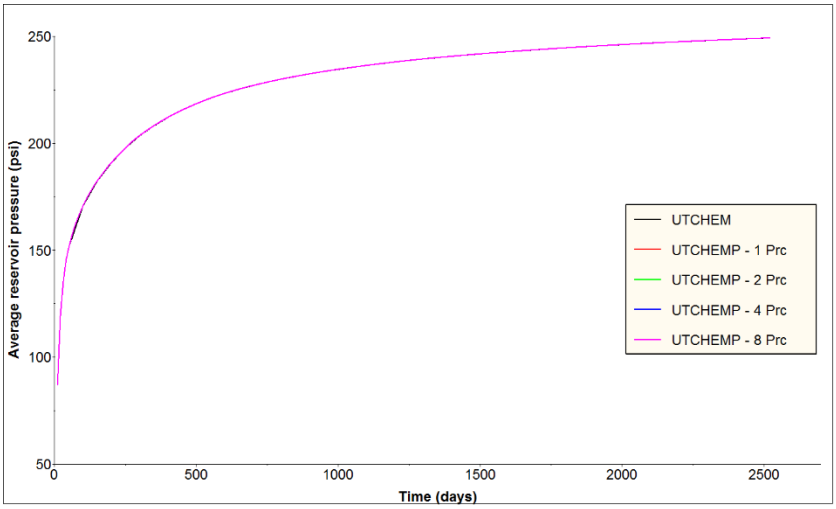

Figure 4-2 Average reservoir pressure (Case study 1)

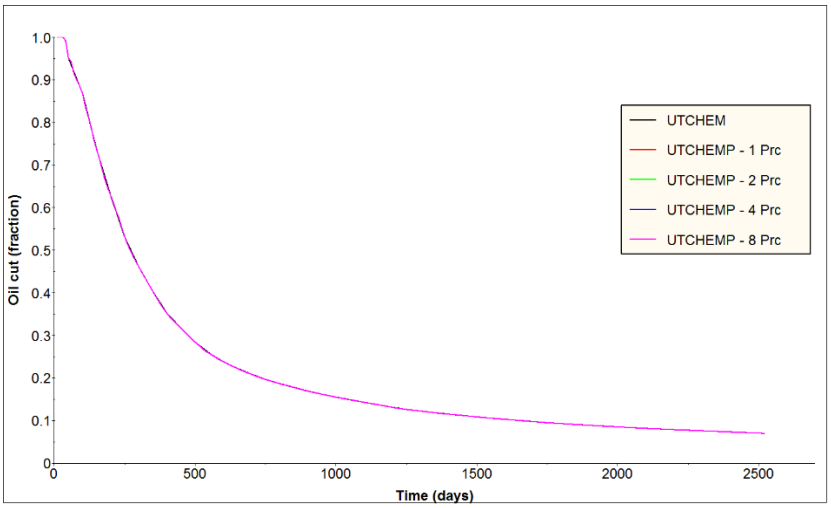

Figure 4-3 Oil cut (Case study 1)

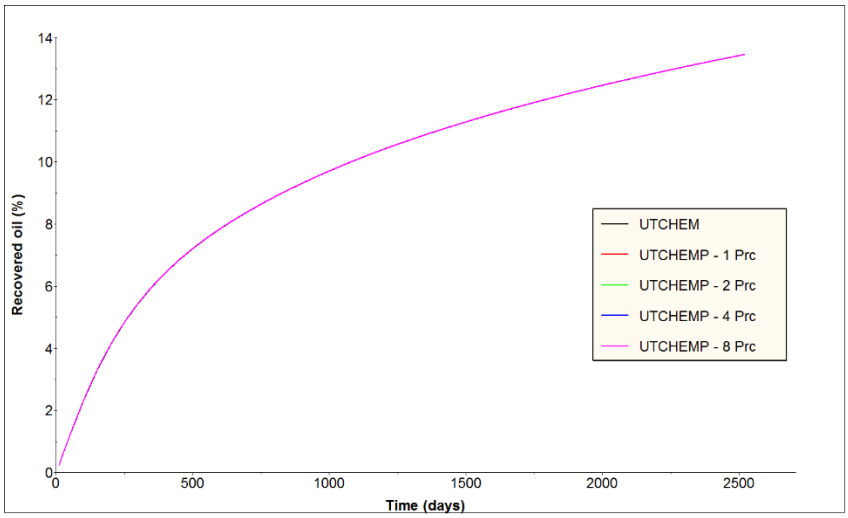

Figure 4-4 Recovered oil (Case study 1)

Distributions of pressure and water saturation at the end of the simulation are also shown in Figures 4-5 and 4-6 in order to further validate the code.

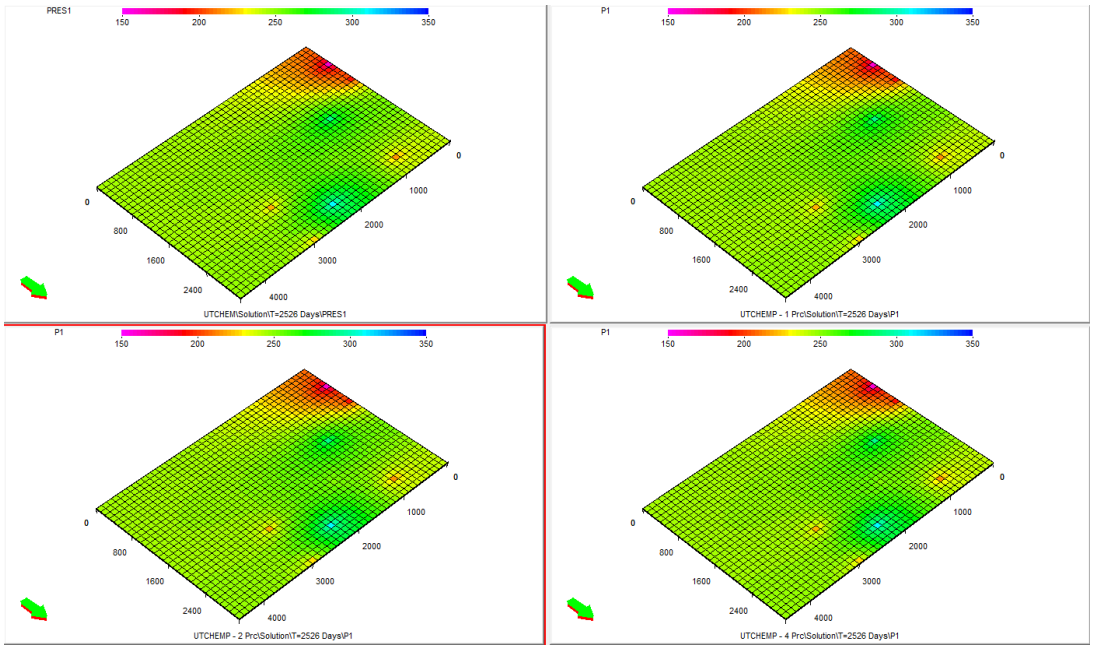

Figure 4-5 Pressure distribution at the end of simulation (Case study 1). Upper left shows the result for UTCHEM. Upper right, lower left, and lower right show the results for UTCHEMP using 1, 2, and 4 processors, respectively.

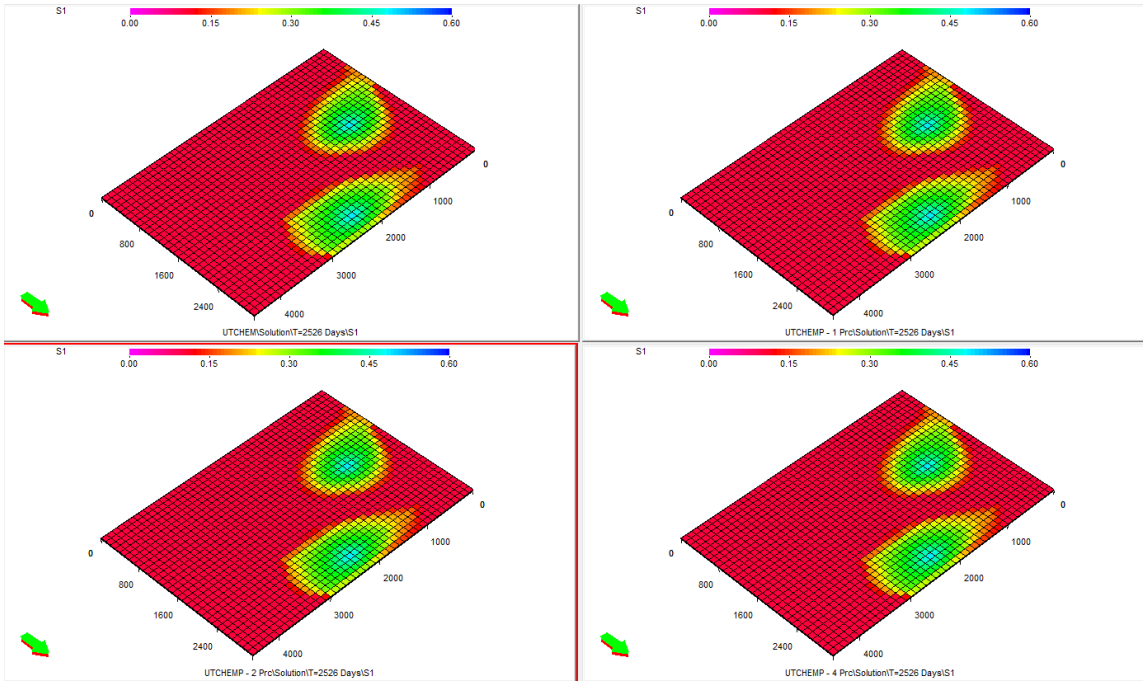

Figure 4-6 Water saturation distribution at the end of simulation (Case study 1). Upper left shows the result for UTCHEM. Upper right, lower left, and lower right show the results for UTCHEMP using 1, 2, and 4 processors, respectively.

4.2 CASE STUDY 2: GEL TREATMENT

In this case, gel treatment is studied. The IREACT flag in UTCHEMP is set to 1 so that gel reactions are used. Reservoir and fluid properties are provided in Table 4-2. Moreover, as can be seen in Figure 4-7, there are 1 injector and 3 producers operating here and they are constrained to constant pressure. Variable grid size is also used in this case. Although lateral and vertical permeabilities are different in this reservoir model, they are both uniform throughout the reservoir. However, the reservoir is heterogeneous in terms of porosity. The porosity distribution is shown in Figure 4-8. The simulation time is 816 days, and the bottom-hole pressures do not change with time. In the present case, the energy equation is considered and the IENG flag is set to 1.

Table 4-2 Reservoir and fluid properties description (Case study 2)

Case 2 (Gel treatment)			
Dimensions (ft)	Length	1100	
	Width	1000	
	Thickness	30	
Number of gridblocks		364 (14x13x2)	
Number of components		7	
Max. number of phases		4	
Porosity (fraction)		Heterogeneous 0.17 – 0.45	
Permeability (md)	Top layer	72	
	Bottom layer	900	
Initial water saturation (fraction)		0.3001	
Irreducible water saturation (fraction)		0.3	
Residual oil saturation (fraction)		0.26	
Water viscosity (cp)		0.7	
Oil viscosity (cp)		20	
Initial reservoir pressure (psi)		200	
Number of wells		4	1 Injector
			3 Producers
Initial reservoir temperature (°F)		103	
Injected fluid temperature (°F)		68	
Simulation time (days)		816	

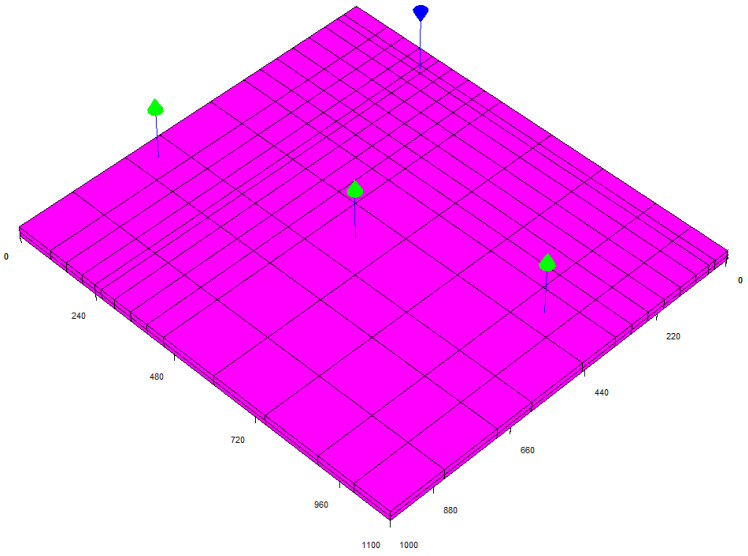

Figure 4-7 Reservoir model (Case study 2). Blue markers represent injection wells, and green ones are producers.

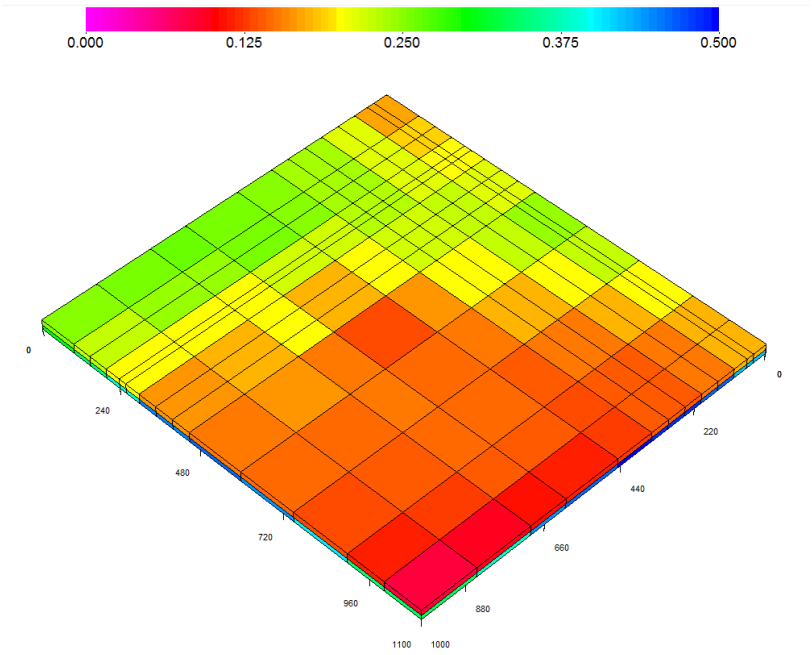

Figure 4-8 Porosity distribution (Case study 2)

The plots for average reservoir pressure, oil cut, and recovered oil are provided in Figures 4-9, 4-10, and 4-11, respectively. As the graphs show, the results are perfectly matched.

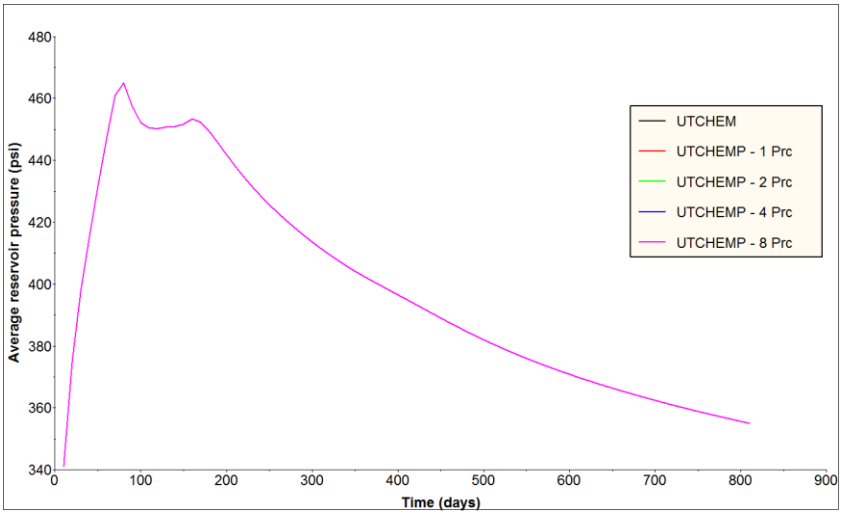

Figure 4-9 Average reservoir pressure (Case study 2)

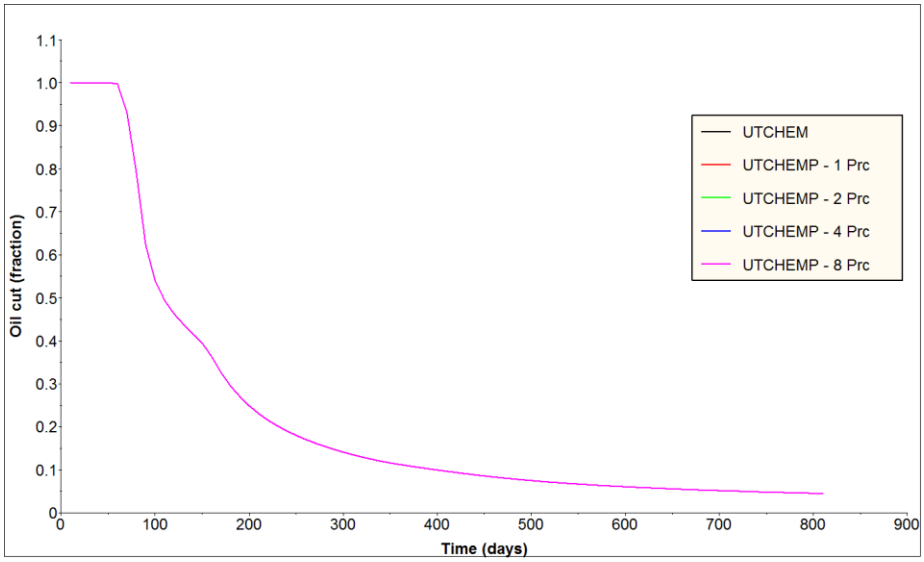

Figure 4-10 Oil cut (Case study 2)

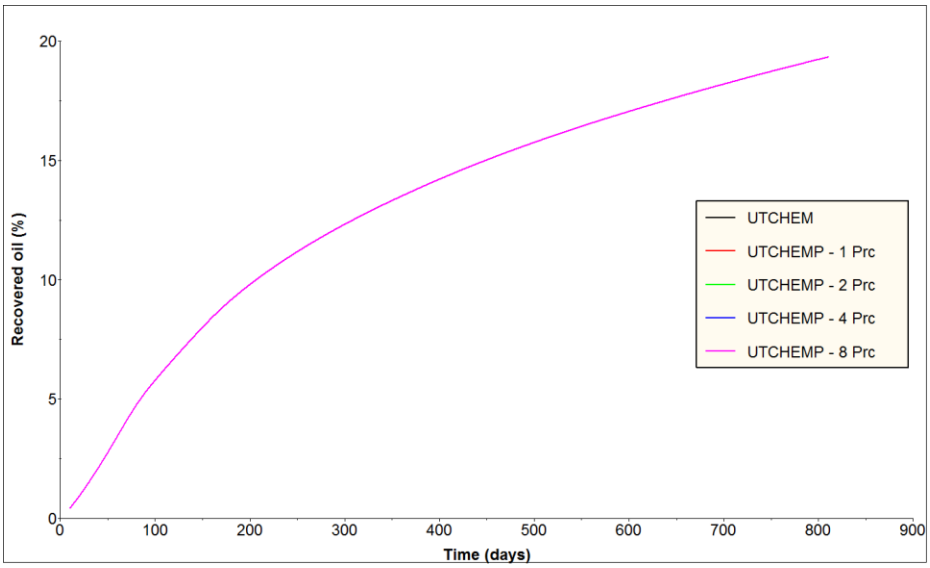

Figure 4-11 Recovered oil (Case study 2)

Furthermore, Figures 4-12 and 4-13 provide pressure and water saturation distribution at the end of the simulation. The results are in excellent agreement.

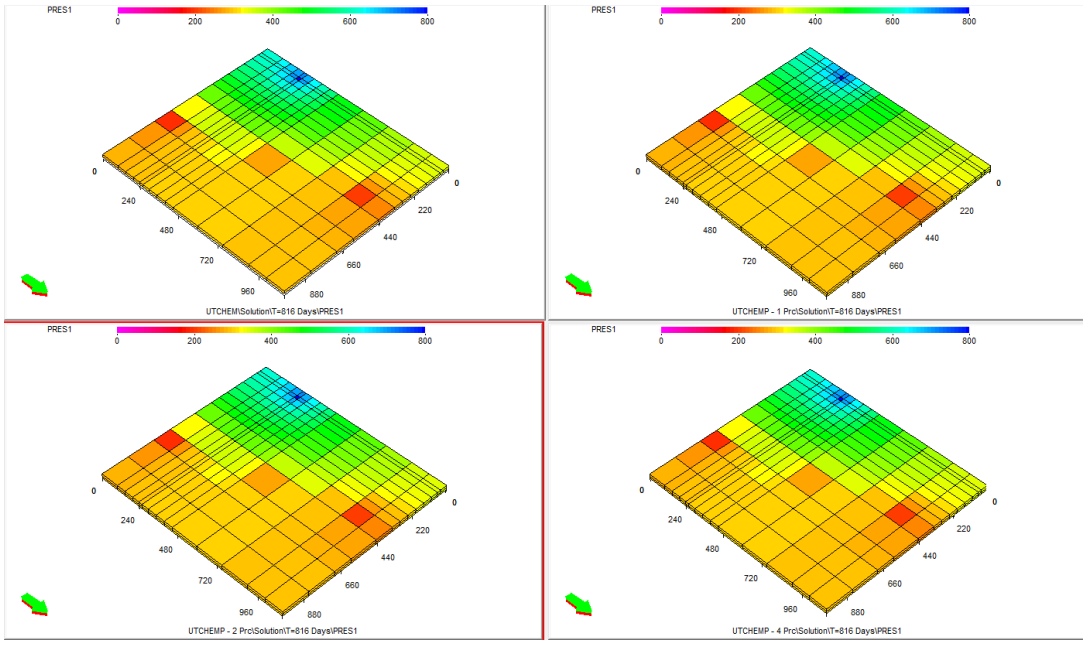

Figure 4-12 Pressure distribution at the end of simulation (Case study 2). Upper left shows the result for UTCHEM. Upper right, lower left, and lower right show the results for UTCHEMP using 1, 2, and 4 processors, respectively.

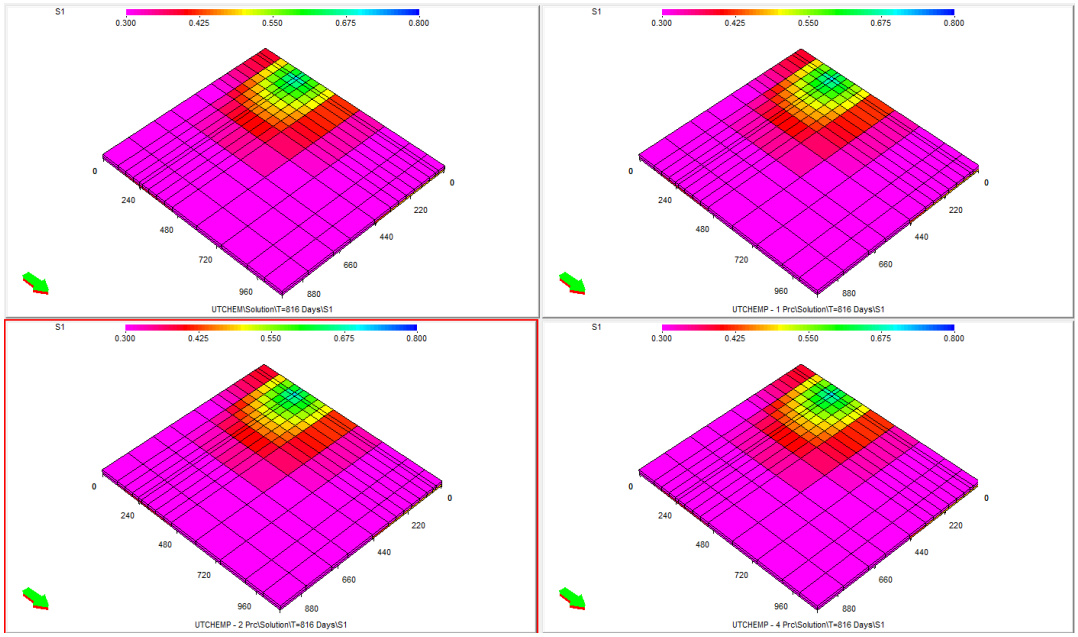

Figure 4-13 Water saturation distribution at the end of simulation (Case study 2). Upper left shows the result for UTCHEM. Upper right, lower left, and lower right show the results for UTCHEMP using 1, 2, and 4 processors, respectively.

4.3 CASE STUDY 3: POLYMER FLOODING

This case is a polymer flooding problem whose main features are presented in Table 4-3. 1 injector and 4 producers operate under constant pressure constraint in a five-spot scheme as can be seen in Figure 4-14. The reservoir is heterogeneous, and Figures 4-15 and 4-16 present the porosity and permeability distributions. Polymer concentration in the aqueous phase injected into the reservoir is 1000 ppm. Simulation time for this case is 1500 days. The injected fluid solution and the bottom hole pressures do not change with time.

Table 4-3 Reservoir and fluid properties description (Case study 3)

Case 3 (Polymer flooding)			
Dimensions (ft)	Length	1650	
	Width	1650	
	Thickness	10	
Number of gridblocks		675 (15x15x3)	
Number of components		6	
Max. number of phases		4	
Porosity (fraction)		Heterogeneous 0.1 – 0.5	
Permeability (md)		Heterogeneous 700 - 3000	
Initial water saturation (fraction)		0.38	
Irreducible water saturation (fraction)		0.2	
Residual oil saturation (fraction)		0.2	
Water viscosity (cp)		0.73	
Oil viscosity (cp)		40	
Initial reservoir pressure (psi)		100	
Number of Wells		5	1 Injector
			4 Producers
Injected polymer concentration (ppm)		1000	
Simulation time (days)		1500	

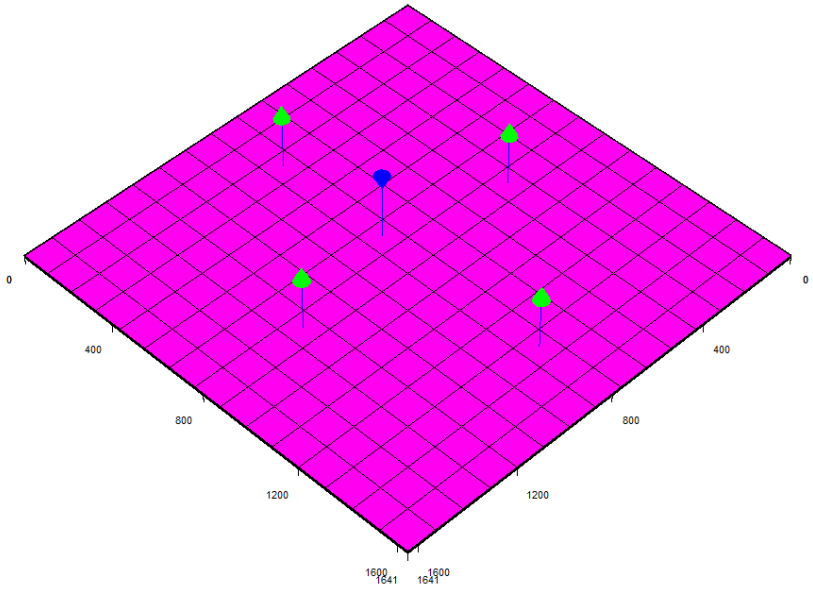

Figure 4-14 Reservoir model (Case study 3). Blue markers represent injection wells, and green ones are producers.

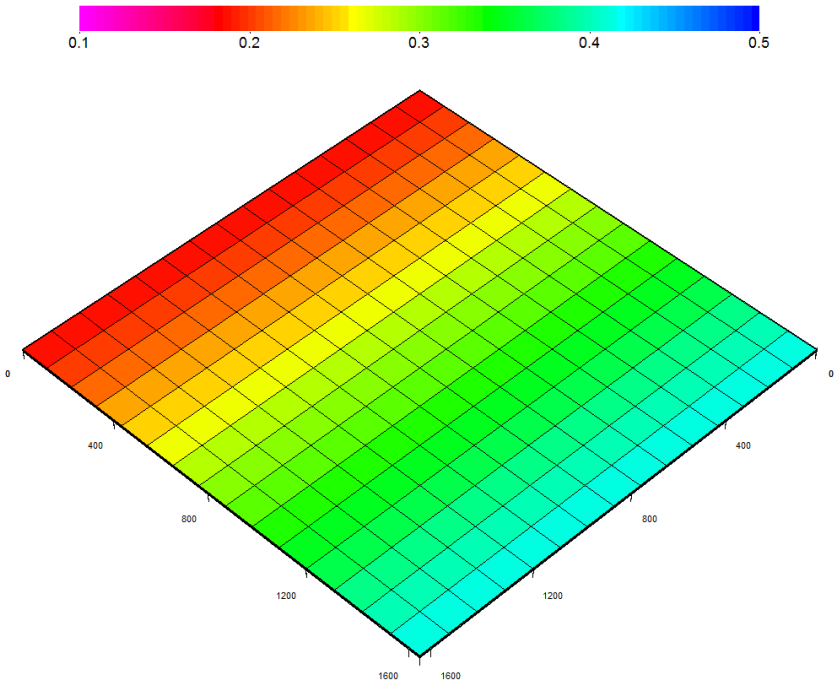

Figure 4-15 Porosity distribution (Case study 3)

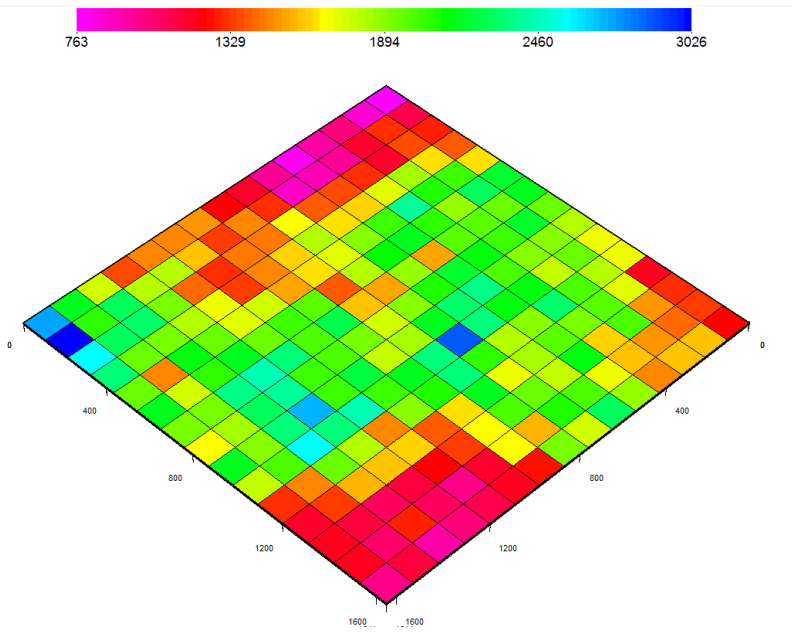

Figure 4-16 Permeability distribution (Case study 3)

Average reservoir pressure, oil cut, and recovered oil are shown in Figures 4-17, 4-18, and 4-19, respectively. Results are in perfect agreement.

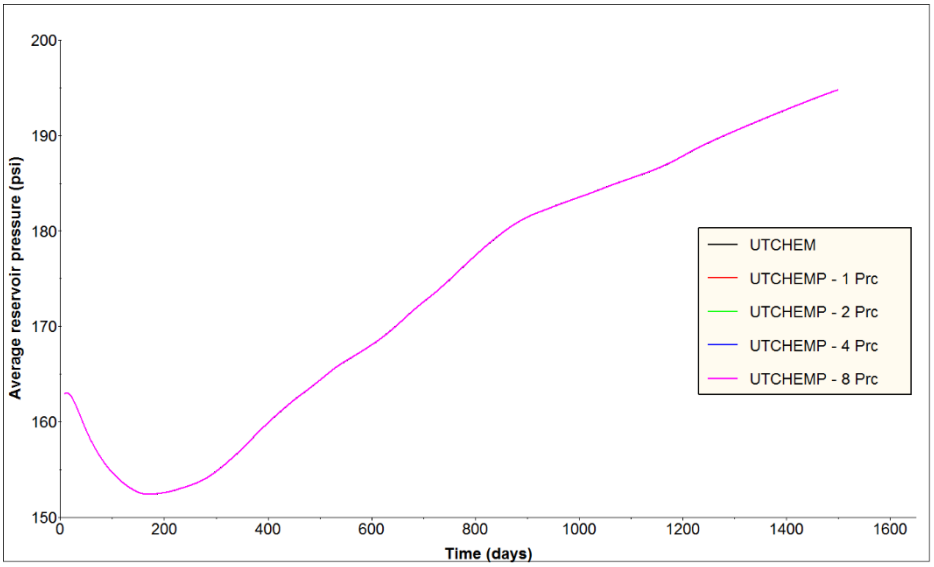

Figure 4-17 Average reservoir pressure (Case study 3)

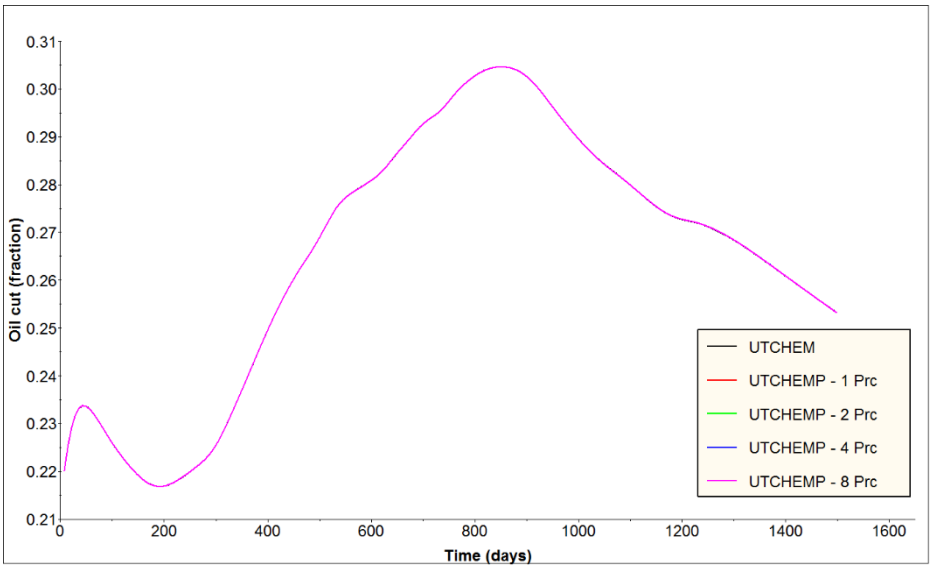

Figure 4-18 Oil cut (Case study 3)

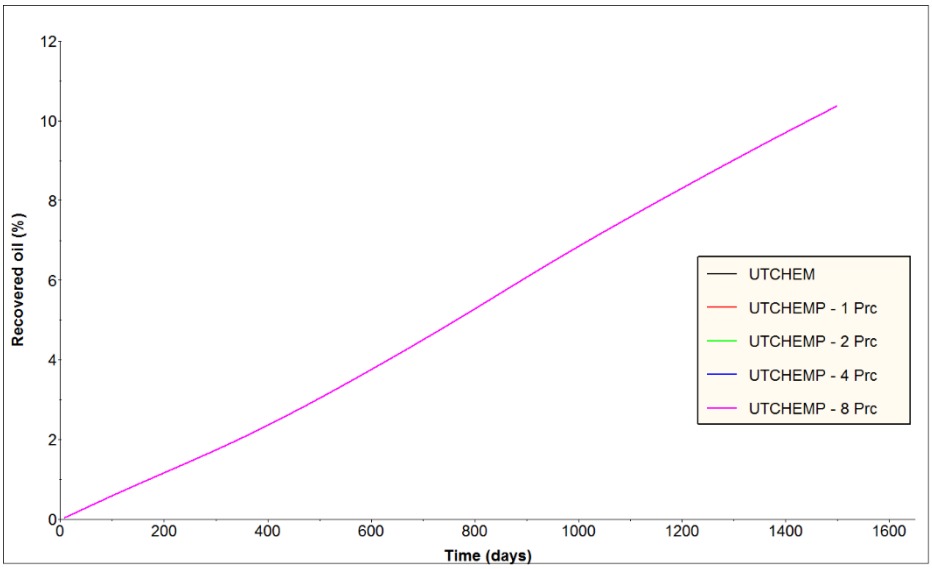

Figure 4-19 Recovered oil (Case study 3)

In addition, pressure and water saturation distributions at the end of the simulation are provided in Figures 4-20 and 4-21 to further verify the code. Results are in excellent agreement.

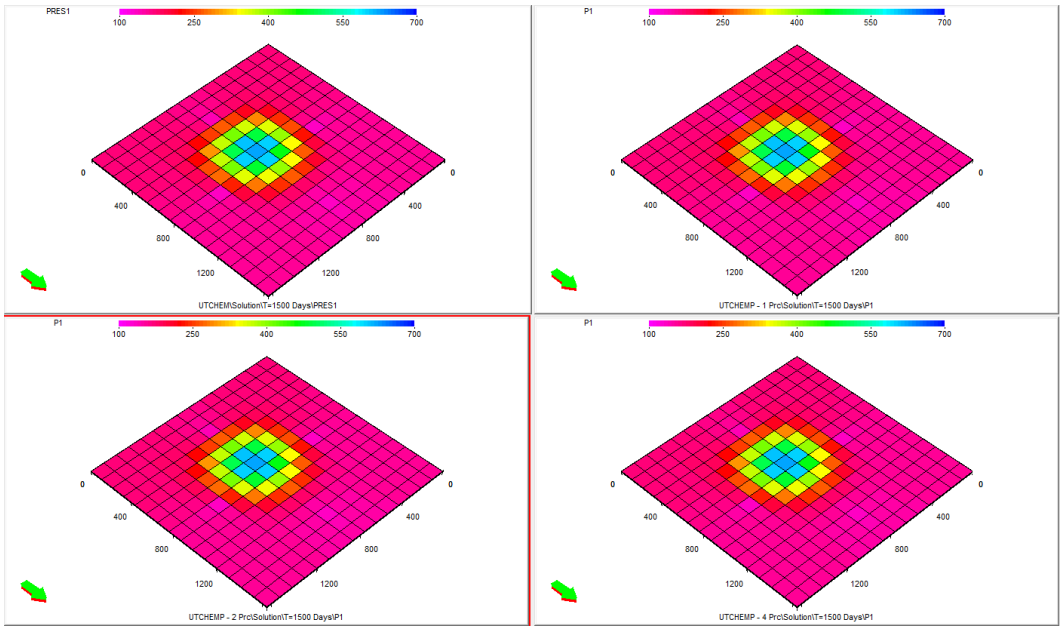

Figure 4-20 Pressure distribution at the end of simulation (Case study 3). Upper left shows the result for UTCHEM. Upper right, lower left, and lower right show the results for UTCHEMP using 1, 2, and 4 processors, respectively.

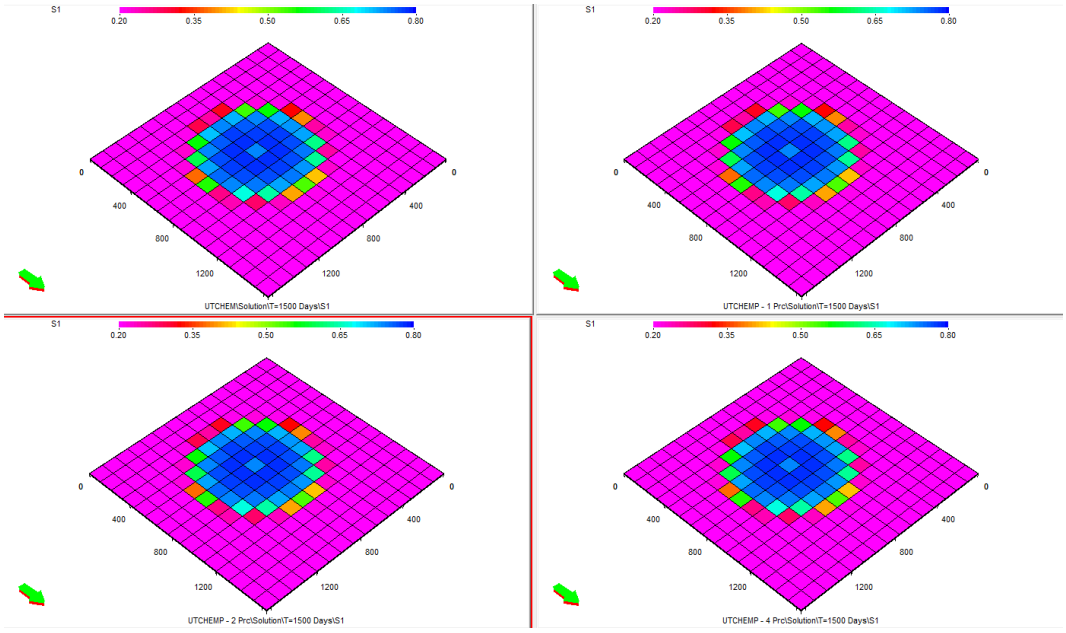

Figure 4-21 Water saturation distribution at the end of simulation (Case study 3). Upper left shows the result for UTCHEM. Upper right, lower left, and lower right show the results for UTCHEMP using 1, 2, and 4 processors, respectively.

4.4 CASE STUDY 4: SURFACTANT/POLYMER FLOODING

Surfactant/Polymer flooding (SP) process is studied in this case. The main features of this case are described in Table 4-4. There is a single injection well with constant rate flow injection and a single producer well constrained to constant bottom hole pressure whose positions are those of the classic five-spot pattern as can be seen in Figure 4-22. The simulation time is 1500 days and the injection condition does not change with time.

Table 4-4 Reservoir and fluid properties description (Case study 4)

Case 4 (Surfactant/Polymer flooding)			
Dimensions (ft)	Length	250	
	Width	250	
	Thickness	10	
Number of gridblocks		242 (11x11x2)	
Number of components		10	
Max. number of phases		4	
Porosity (fraction)		0.2	
Permeability (md)	X	500	
	Y	500	
	Z	50	
Initial water saturation (fraction)		0.65	
Irreducible water saturation (fraction)		0.37	
Residual oil saturation (fraction)		0.35	
Water viscosity (cp)		0.86	
Oil viscosity (cp)		4	
Initial reservoir pressure (psi)		700	
Number of Wells		2	1 Injector
			1 Producer
Injected polymer concentration (ppm)		500	
Injected surfactant concentration (volume fraction)		0.03	
Simulation time (days)		1500	

Comparison of the results for the execution of UTCHEM and UTCHEMP is shown in Figures 4-23, 4-24, and 4-25. It may be pointed out that the results do not match as they did in the previous cases. However, as can be seen in Figure 4-24, it is a highly oscillatory

case, and the small differences between the results are acceptable. To further justify the differences, it can be noted that while higher order finite difference method is generally used in UTCHEMP, we use lower order method for gridblocks in the processor's boundary. This approximation results in a negligible error, but simplifies the development.

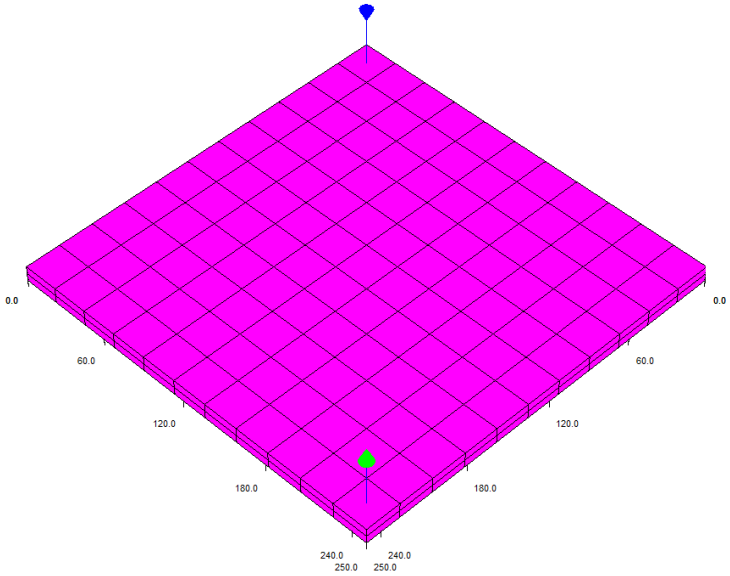

Figure 4-22 Reservoir model (Case study 4). Blue markers represent injection wells, and green ones are producers.

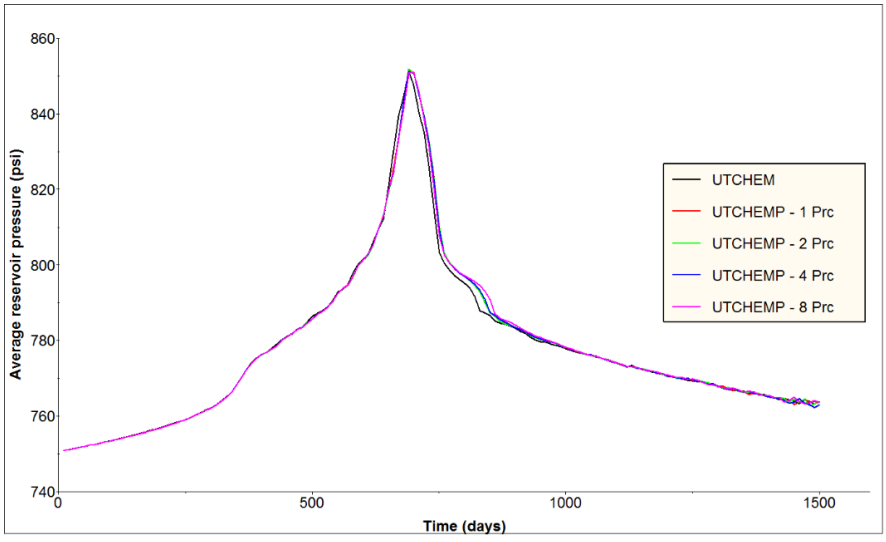

Figure 4-23 Average reservoir pressure (Case study 4)

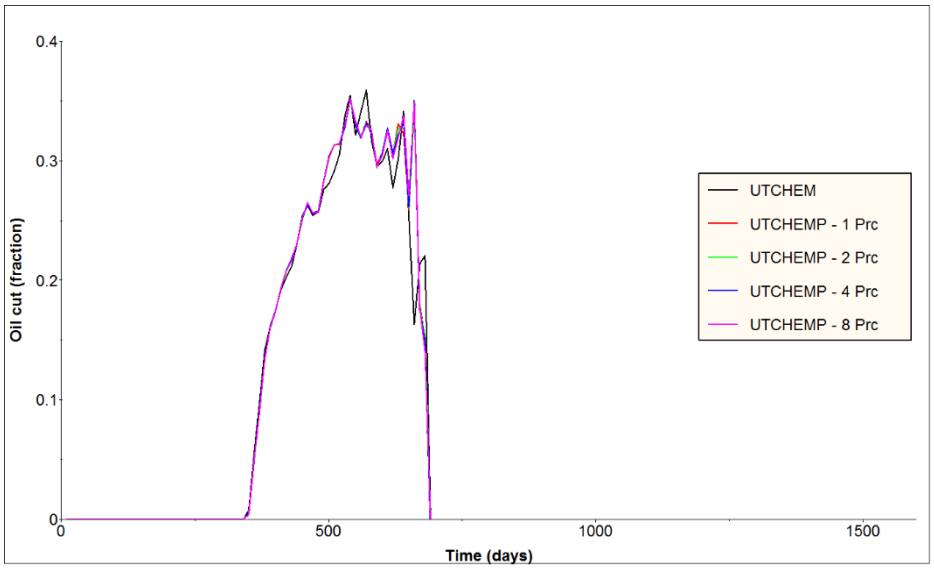

Figure 4-24 Oil cut (Case study 4)

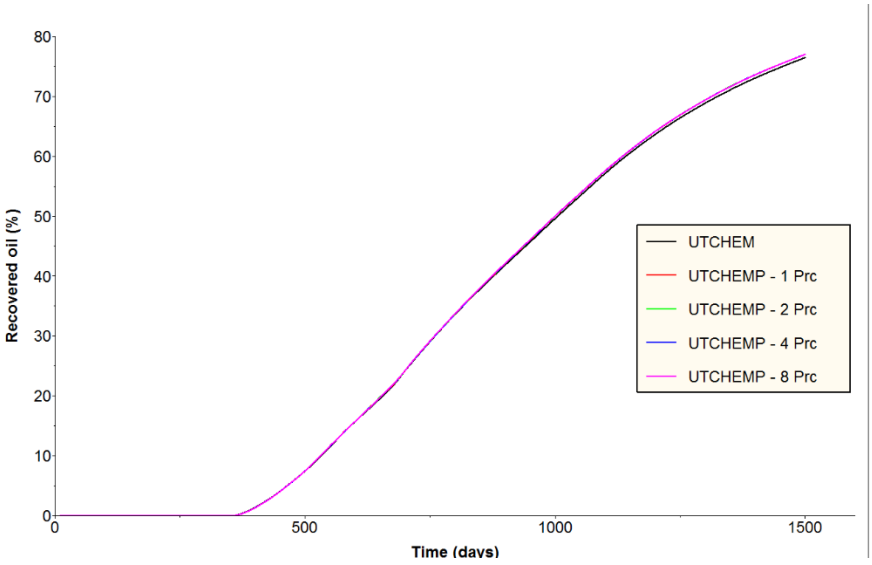

Figure 4-25 Recovered oil (Case study 4)

To further verify the simulator, pressure and water saturation distributions at the end of the simulation are shown in Figures 4-26 and 4-27.

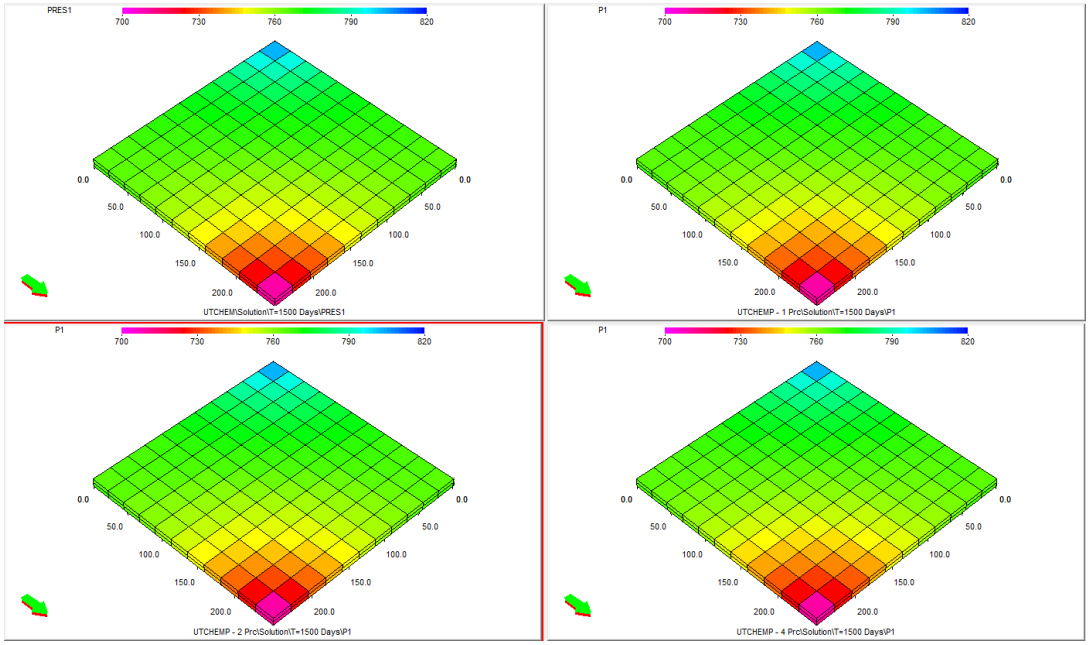

Figure 4-26 Pressure distribution at the end of simulation (Case study 4). Upper left shows the result for UTCHEM. Upper right, lower left, and lower right show the results for UTCHEMP using 1, 2, and 4 processors, respectively.

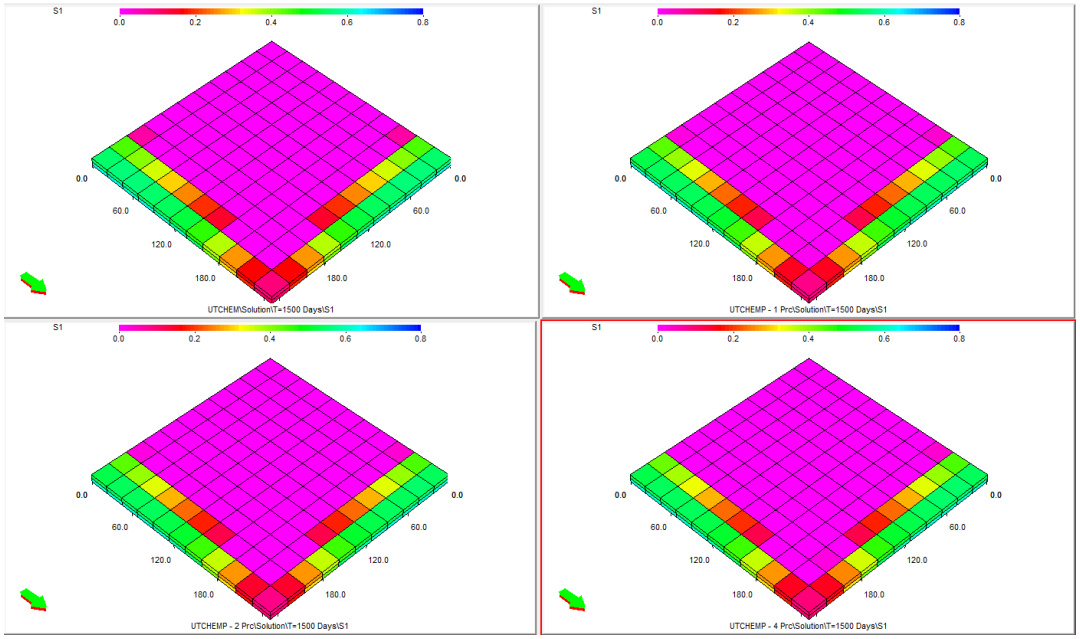

Figure 4-27 Water saturation distribution at the end of simulation (Case study 4). Upper left shows the result for UTCHEM. Upper right, lower left, and lower right show the results for UTCHEMP using 1, 2, and 4 processors, respectively.

4.5 CASE STUDY 5: ASP FLOOD

In this case, an alkaline/surfactant/polymer flooding (ASP) process is studied. The reservoir and the fluid properties are described in Table 4-5. The IREACT flag is set to 3 in order to use the geochemistry option with acidic crude. Four injectors and thirteen producers operate in this reservoir under constant rate restriction. Constant grid size is used to model the reservoir as visualized in Figure 4-28. Porosity is uniform throughout the reservoir; however, the reservoir is heterogeneous in terms of permeability. Lateral permeability distribution is shown in Figure 4-29. The vertical permeability, in each gridblock, is ten times lower than the lateral one. The initial water saturation distribution is also depicted in Figure 4-30.

Table 4-5 Reservoir and fluid properties description (Case study 5)

Case 5 (ASP flooding)			
Dimensions (ft)	Length	600	
	Width	600	
	Thickness	40	
Number of gridblocks		1,083 (19x19x3)	
Number of components		11	
Max. number of phases		3	
Porosity (fraction)		0.3	
Permeability (md)		Heterogeneous Lateral: 900 – 2300 Vertical: 90 - 230	
Initial water saturation (fraction)		Non-uniform 0.3 – 0.8	
Irreducible water saturation (fraction)		0.25	
Residual oil saturation (fraction)		0.15	
Water viscosity (cp)		0.46	
Oil viscosity (cp)		40	
Initial reservoir pressure (psi)		1780	
Number of Wells	17	4 Injectors	
		13 Producers	
Simulation time (days)		551	

The simulation time is 551 days. Injection of fluids into the reservoir is scheduled as follows (in each period, constant flow rate condition is applied on all the wells):

0 < time < 26 days: Alkaline accompanied by polymer is injected.

26 days < time < 51 days: A water solution of Alkaline and surfactant is injected.

51 days < time < 226 days: Alkaline, surfactant, and polymer are injected.

226 days < time < 276 days: Alkaline and polymer dissolved in water is injected.

276 days < time < 551 days: Brine water is injected into the reservoir.

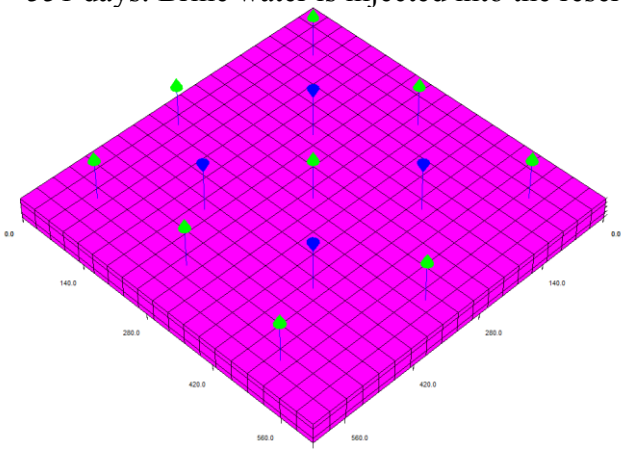

Figure 4-28 Reservoir model (Case study 5). Blue markers represent injection wells, and green ones are producers.

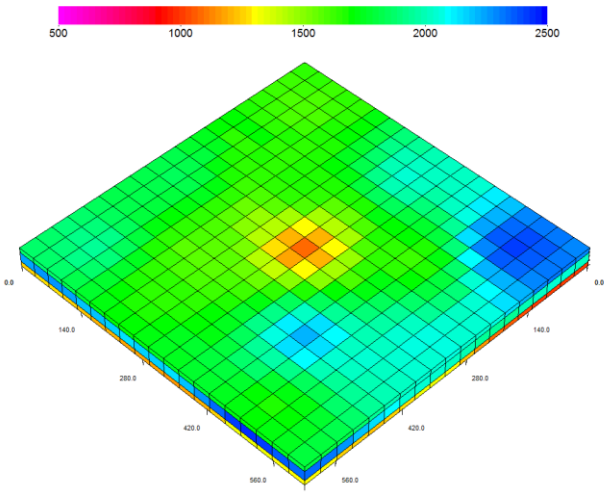

Figure 4-29 Lateral permeability distribution (Case study 5)

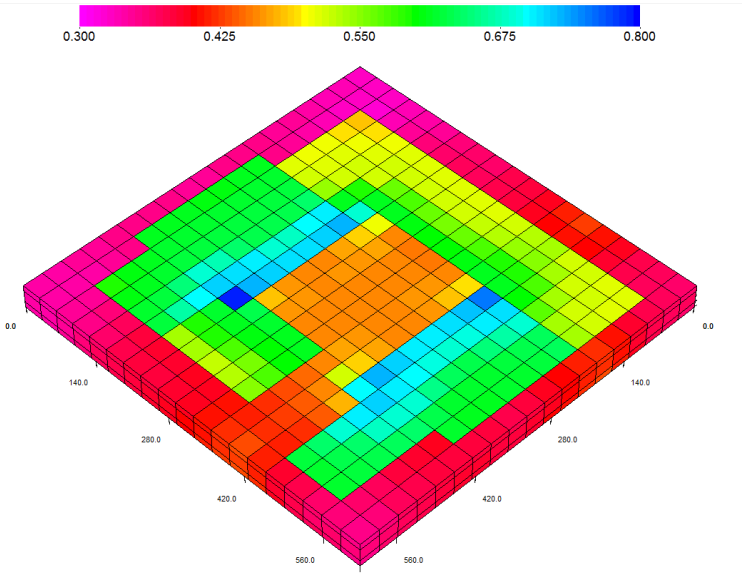

Figure 4-30 Initial water saturation distribution (Case study 5)

Simulation results are provided in Figures 4-31, 4-32, 4-33, 4-34, and 4-35. The results are in good agreement. Although the results do not perfectly match, the difference is very small and totally acceptable.

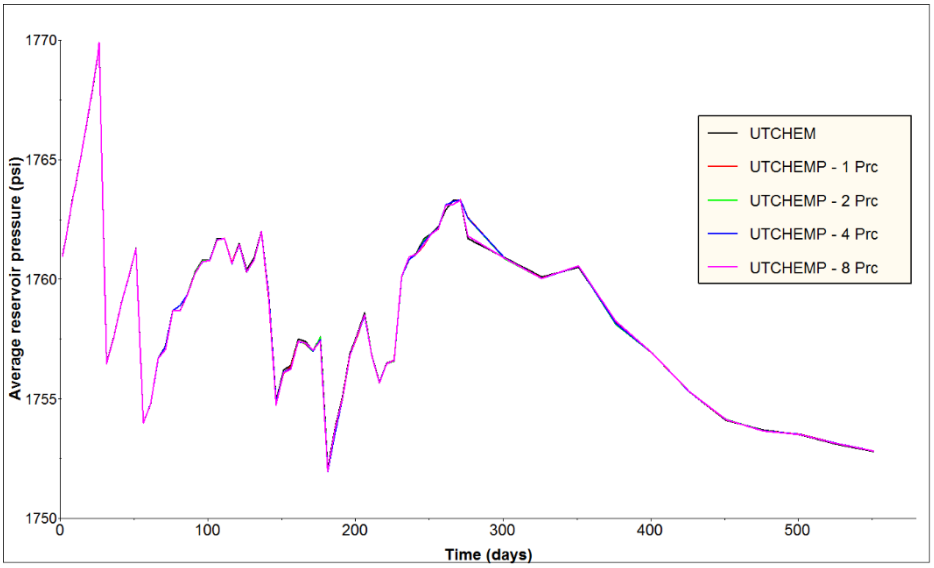

Figure 4-31 Average reservoir pressure (Case study 5)

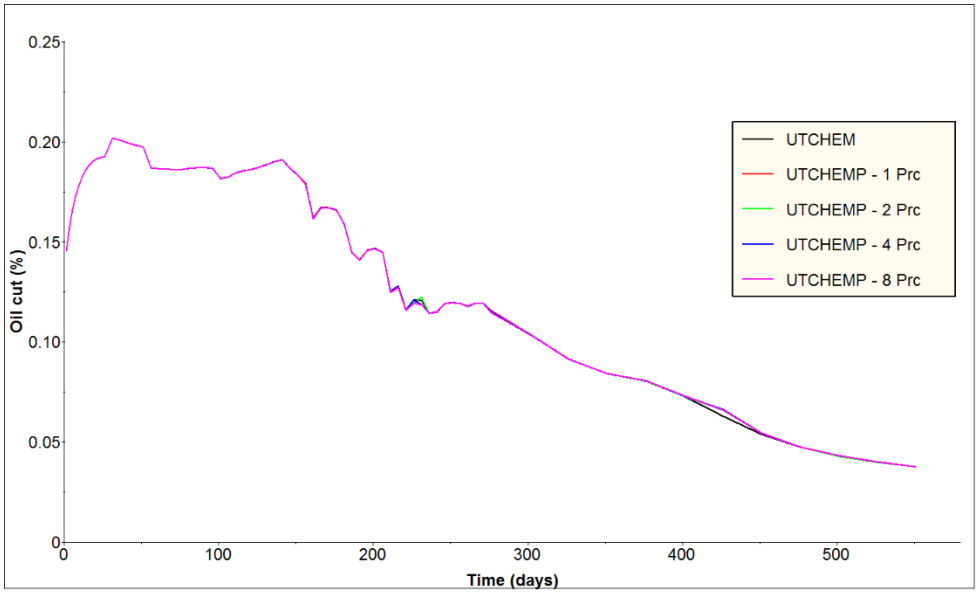

Figure 4-32 Oil cut (Case study 5)

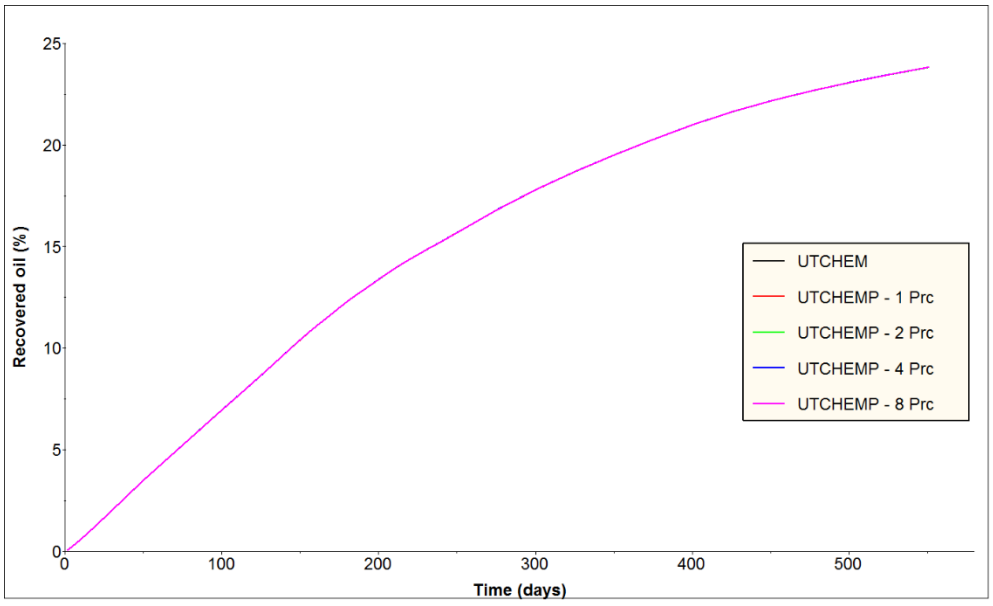

Figure 4-33 Recovered oil (Case study 5)

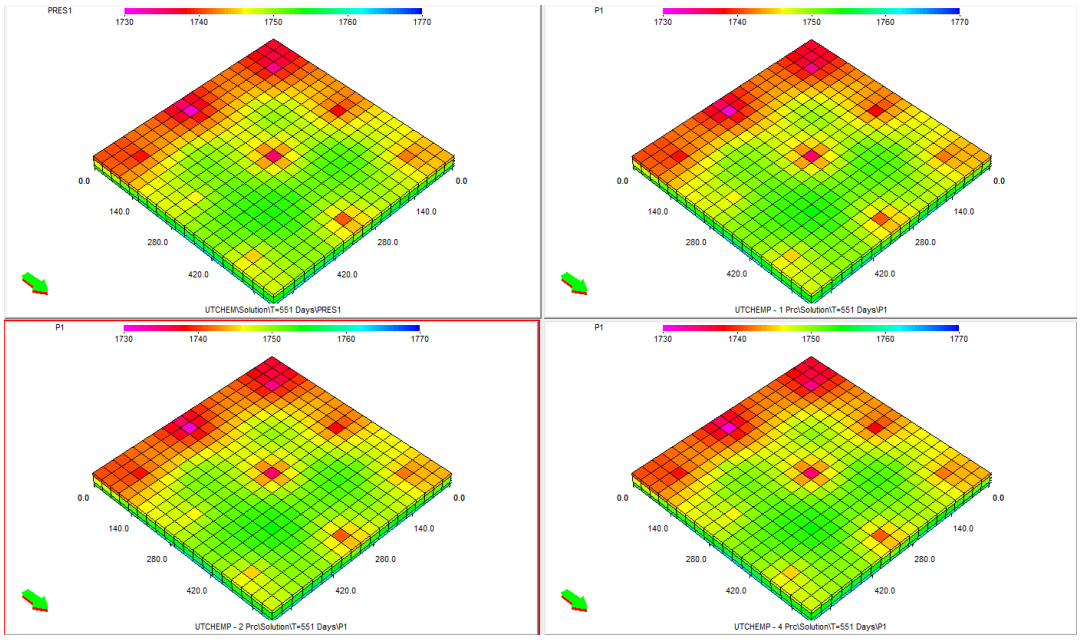

Figure 4-34 Pressure distribution at the end of simulation (Case study 5). Upper left shows the result for UTCHEM. Upper right, lower left, and lower right show the results for UTCHEMP using 1, 2, and 4 processors, respectively.

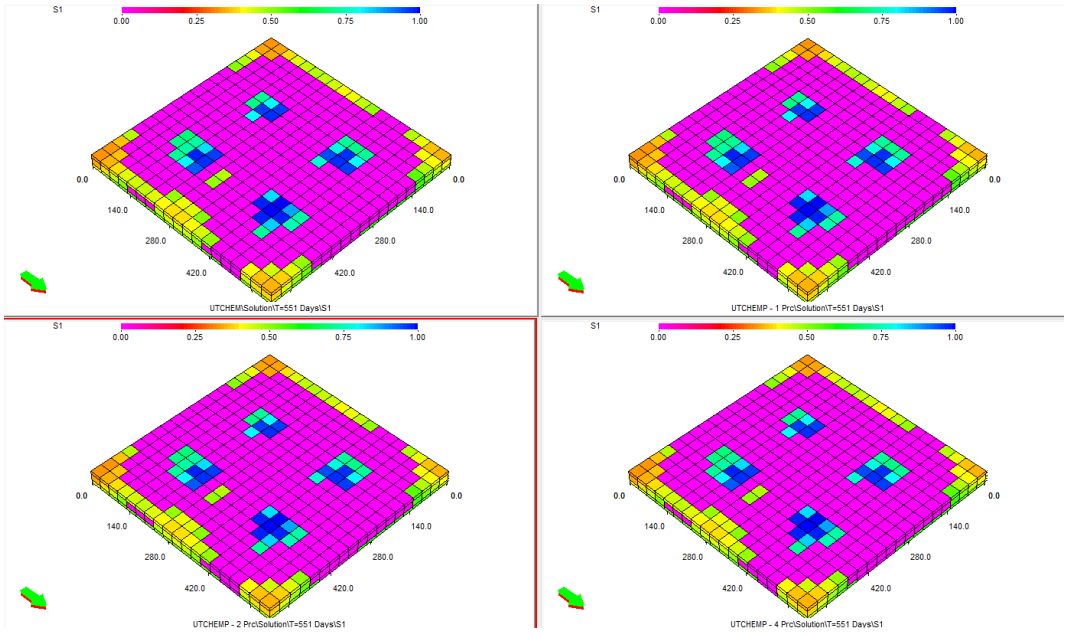

Figure 4-35 Water saturation distribution at the end of simulation (Case study 5). Upper left shows the result for UTCHEM. Upper right, lower left, and lower right show the results for UTCHEMP using 1, 2, and 4 processors, respectively.

Chapter 5: Application

According to the results presented in Chapter 4, the simulator is verified. Now in this chapter, we present the application of UTCHEMP to run large-scale cases. For that purpose, the speedup performance of the simulator is assessed first.

5.1 SPEEDUP PERFORMANCE

This part of the thesis is devoted to evaluate the parallel efficiency of UTCHEMP. For this purpose, the number of gridblocks compared to the verification part is sharply increased. Hypothetically speaking, parallel efficiency is expected to increase with number of gridblocks. The reason is that communication time increases with number of processors which decreases parallel efficiency. This effect, however, plays a less important role when large number of gridblocks is used in the simulations.

The maximum allowable execution time in TACC is limited to 24 hours. Thus, in order to achieve the best parallel efficiency in any case study, we tried to use the largest possible number of gridblocks such that the simulation can be done using one processor. We further provide the results for UTCHEMP using 2, 4, 8, 16, 32, and 64 processors in order to check the speedup performance of the code. To further check the effect of number of gridblocks on the speedup performance, speedup results are also provided for smaller numbers of gridblocks in each case.

In addition to presenting the speedup results, we also provide in detail the computational time spent on different sections of the code as follows:

- "Initialization" time mainly includes the computational time spent on initialization of the simulation and reading of input files.
- "Transmissibility calculation" time is the time spent for calculating and assembling the transmissibility matrix.

- "Concentration equation" time consists of the time spent on explicit calculations of concentrations, saturations, and some other physical properties, including densities and viscosities.
- "Salinity calculation" is the time spent for calculating effective salinity and redefining concentrations in terms of a pseudo-ternary.
- "Linear system solver" time refers to the computational time spent for solving the linear system of equations for computing the pressure implicitly.
- "Other" time mostly consists of the time for the communication among processors. It also includes the time spent for well management, time-step calculations, calculation of initial oil in place, and etc.

5.1.1 Case Study 6: Water Flooding

This case is similar to Case study 1, but is a larger case with much more gridblocks. Also some other characteristics of the model are different. The main features of this case are given in Table 5-1. In terms of geology, the reservoir is homogeneous. The reservoir model is also similar to Figure 4-1 but with more gridblocks. All of the seventeen operating wells are constrained to constant flow rate injection. The simulation time is set to 650 days, and during the entire simulation, the injection condition does not change.

The detailed computational times for this case using 1, 2, 4, 8, 16, 32, and 64 processors are also presented in Table 5-2 and are shown in Figure 5-1. It can be noted that the contribution of "Other" part increases with number of processors. The reason is that communication amongst processors increases with number of processors.

Table 5-1 Reservoir and fluid properties description (Case study 6)

Case 6 (Water flooding)			
Dimensions (ft)	Length	6000	
	Width	4000	
	Thickness	80	
Number of Gridblocks		240,000 (300x200x4)	
Number of components		5	
Max. number of phases		4	
Porosity (fraction)		0.1371	
Permeability (md)	X	800	
	Y	1000	
	Z	5	
Initial water saturation (fraction)		0.1	
Irreducible water saturation (fraction)		0.1	
Residual oil saturation (fraction)		0.45	
Water viscosity (cp)		0.5	
Oil viscosity (cp)		2.7	
Initial reservoir pressure (psi)		150	
Number of wells		17	4 Injectors
			13 Producers
Simulation time (days)		650	

Table 5-2 Computational times for Case study 6

	1 Prc	2 Prc	4 Prc	8 Prc	16 Prc	32 Prc	64 Prc
Initialization	6	6	7	8	9	12	16
Transmissibility calculation	1475	715	302	159	89	47	38
Concentration equation	6010	2971	1495	756	435	211	129
Salinity calculation	176	82	38	18	10	8	6
Linear system solver	59266	29555	16089	8095	4119	2069	1175
Other	549	413	324	279	236	151	156
Total	67482	33742	18255	9315	4898	2498	1520

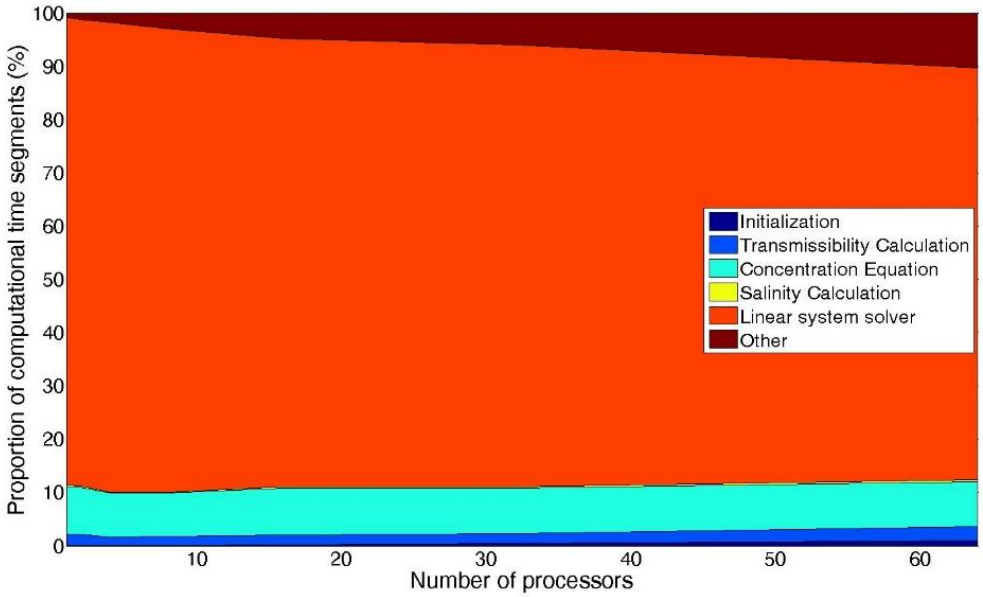

Figure 5-1 Contributions of different parts of the code to the total computational time (Case study 6)

Furthermore, Figures 5-2 and 5-3 show the corresponding speedup and parallel efficiency results. It can be seen that parallel efficiency decreases with number of processors. In other words, the speedup is close to linear speedup for smaller numbers of processors (up to 8 processors) and then it starts deviating from linear. As mentioned before, the reason behind this low efficiency for the large number of processors is due to the increase in communications among processors. Moreover, parallelization is more efficient for larger numbers of gridblocks. The reason is that for larger cases, the communication time has less effect on the speedup performance because it comprises a smaller portion of the total computational time.

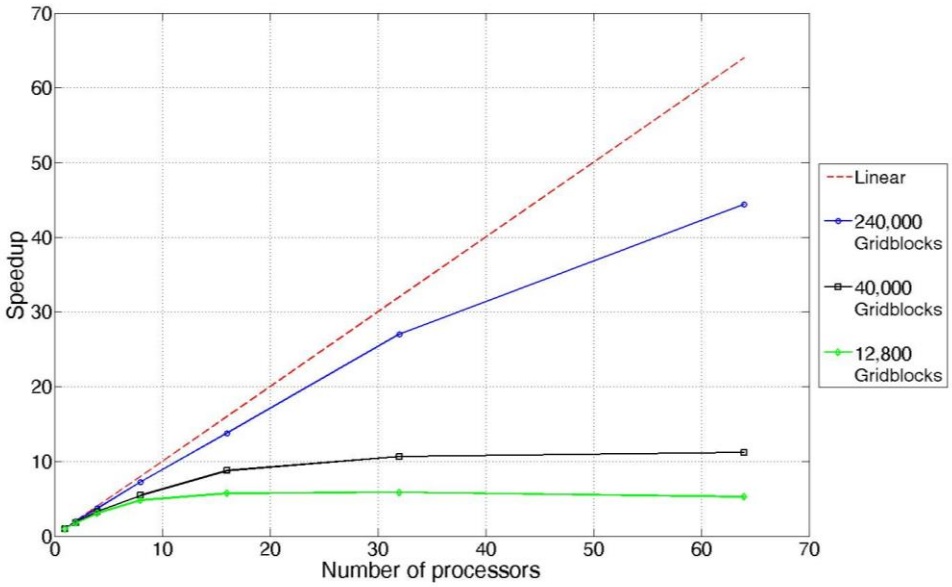

Figure 5-2 Speedup for different numbers of gridblocks (Case study 6)

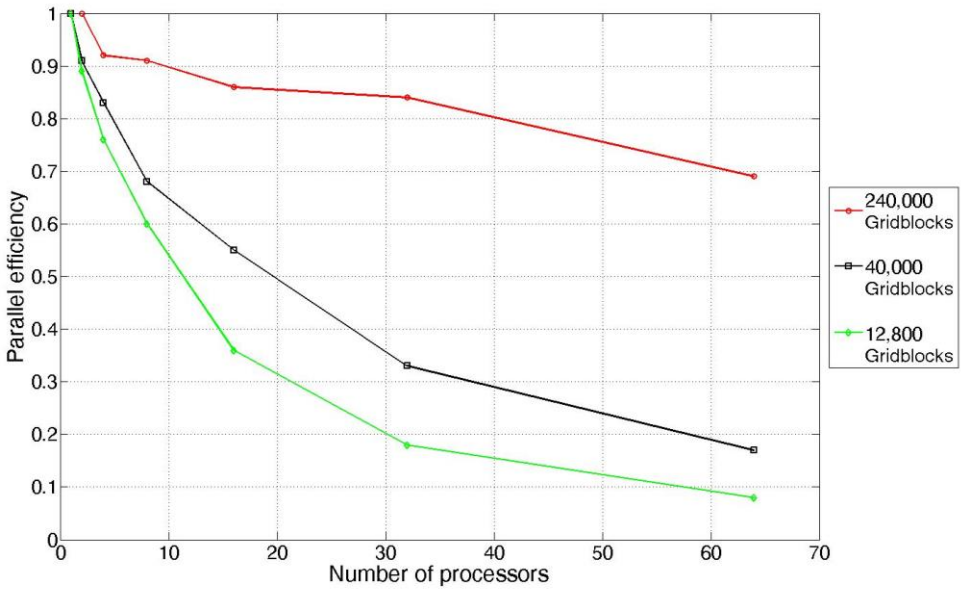

Figure 5-3 Parallel efficiency for different numbers of gridblocks (Case study 6)

5.1.2 Case Study 7: Gel Treatment

Case study 2 is modified here to simulate a gel treatment process for a larger reservoir model with many more gridblocks. The main features are provided in Table 5-3. The IREACT flag is set to 1 to include the gel reactions in the calculations. IENG flag is also set to 1 to include the energy balance equation. The reservoir model is also similar to Figure 4-7 but with finer gridblocks. All the three operating wells for the entire simulation time, which is 4000 days, are constrained to a fixed bottom hole pressure. Also the injected fluid temperature is lower than the initial reservoir temperature.

Table 5-3 Reservoir and fluid properties description (Case study 7)

Case 7 (Gel treatment)			
Dimensions (ft)	Length	20000	
	Width	8000	
	Thickness	150	
Number of Gridblocks		400,000 (200x200x10)	
Number of components		7	
Max. number of phases		4	
Porosity (fraction)		0.2	
Permeability (md)	X	500	
	Y	500	
	Z	50	
Initial water saturation (fraction)		0.3001	
Irreducible water saturation (fraction)		0.3	
Residual oil saturation (fraction)		0.26	
Water viscosity (cp)		0.7	
Oil viscosity (cp)		20	
Initial reservoir pressure (psi)		200	
Number of wells		4	1 Injector
			3 Producers
Initial reservoir temperature (°F)		103	
Injected fluid temperature (°F)		68	
Simulation time (days)		4000	

Details of the computational times for this case study are presented in Table 5-4 and visualized in Figure 5-4. The results are similar to the previous case.

Table 5-4 Computational times for Case study 7

	1 Prc	2 Prc	4 Prc	8 Prc	16 Prc	32 Prc	64 Prc
Initialization	9	9	8	9	10	13	9
Transmissibility calculation	1370	645	314	164	88	46	29
Concentration equation	5895	2908	1507	785	395	195	101
Salinity calculation	97	48	22	11	8	6	4
Linear system solver	41144	20298	10145	5189	2586	1305	643
Other	568	442	293	240	147	113	105
Total	49083	24350	12289	6398	3234	1678	891

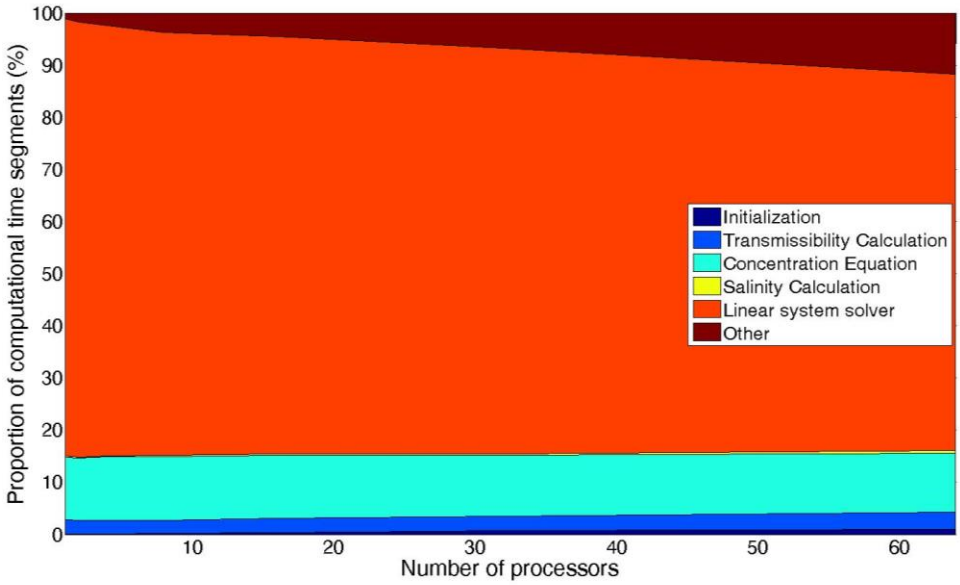

Figure 5-4 Contributions of different parts of the code to the total computational time (Case study 7)

Moreover, results for speedup and parallel efficiency are presented in Figures 5-5 and 5-6, respectively. We draw attention to the efficiency of higher than 1 when using 2 or 4 processors, in this case. It was mentioned in Chapter 2 that super-linear speedup occurs in some situations due to hardware issues.

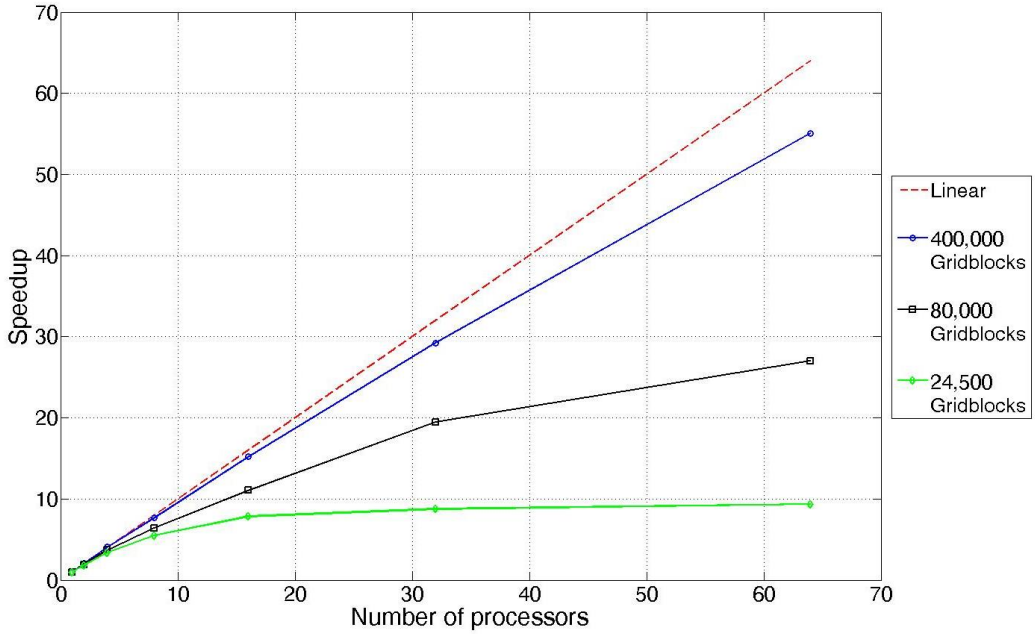

Figure 5-5 Speedup for different numbers of gridblocks (Case study 7)

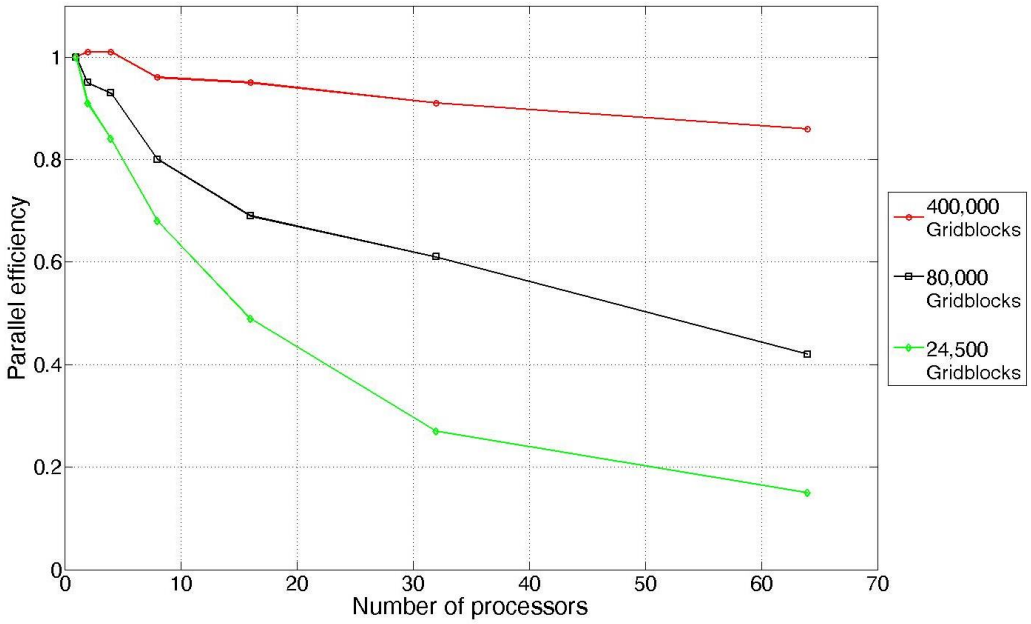

Figure 5-6 Parallel efficiency for different numbers of gridblocks (Case study 7)

5.1.3 Case Study 8: Polymer Flooding

This case is actually a modified version of Case study 3. Here the reservoir is much bigger and the number of gridblocks is much larger than Case study 3. Also the reservoir is homogeneous. The main characteristics of the reservoir and fluid properties can be found in Table 5-5. For this case study, the reservoir model is similar to the one in Figure 4-14 but with finer gridblocks. There are 5 operating wells in a five-spot pattern. The injection well is constrained to constant flow rate, while the producers are restricted to constant bottom hole pressure, and the production operation remains the same during the simulation time, which is 750 days.

Table 5-5 Reservoir and fluid properties description (Case study 8)

Case 8 (Polymer flooding)			
Dimensions (ft)	Length	3000	
	Width	3000	
	Thickness	300	
Number of Gridblocks		1,000,000 (500x500x4)	
Number of components		6	
Max. number of phases		4	
Porosity (fraction)		0.2	
Permeability (md)	X	500	
	Y	500	
	Z	30	
Initial water saturation (fraction)		0.38	
Irreducible water saturation (fraction)		0.2	
Residual oil saturation (fraction)		0.2	
Water viscosity (cp)		0.73	
Oil viscosity (cp)		40	
Initial reservoir pressure (psi)		100	
Number of wells		5	1 Injectors
			4 Producers
Injected polymer concentration (ppm)		1000	
Simulation time (days)		750	

Figures 5-6 and 5-7 show detailed computational times for this case study. Similar to the previous cases, the proportion of the "Other" time increases with the number of processors which is related to an increase in the processors' communications.

Table 5-6 Computational times for Case study 8

	1 Prc	2 Prc	4 Prc	8 Prc	16 Prc	32 Prc	64 Prc
Initialization	10	11	12	11	12	15	18
Transmissibility calculation	1101	583	301	148	74	39	28
Concentration equation	6780	3352	1699	845	419	232	140
Salinity calculation	144	70	33	18	10	7	8
Linear system solver	57050	28331	14092	7001	3509	1748	1109
Other	612	317	219	166	113	109	145
Total	65697	32664	16356	8189	4137	2150	1448

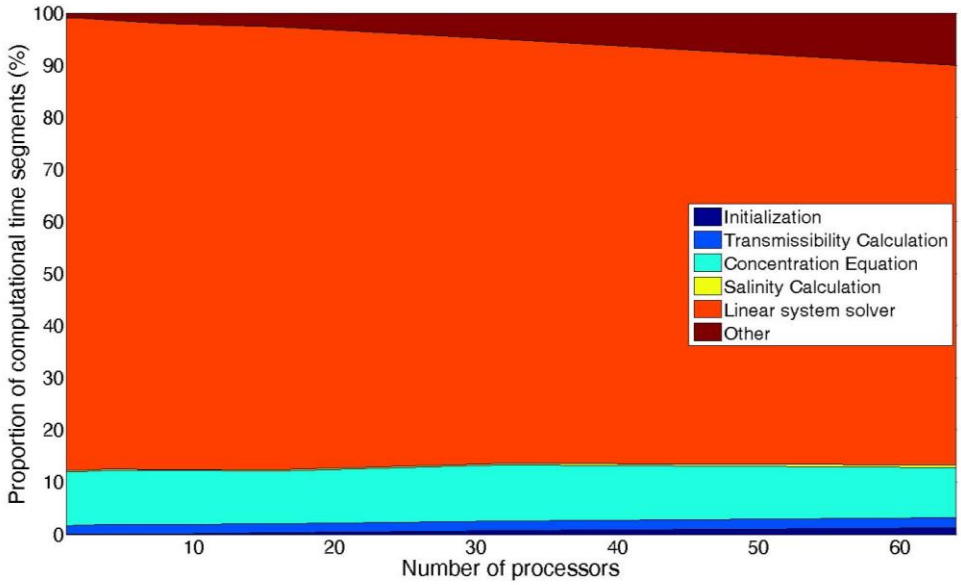

Figure 5-7 Contributions of different parts of the code to the total computational time (Case study 8)

Moreover, Figures 5-8 and 5-9 show the speedup and parallel efficiency results for this case study. The results agree with the results for the previous cases.

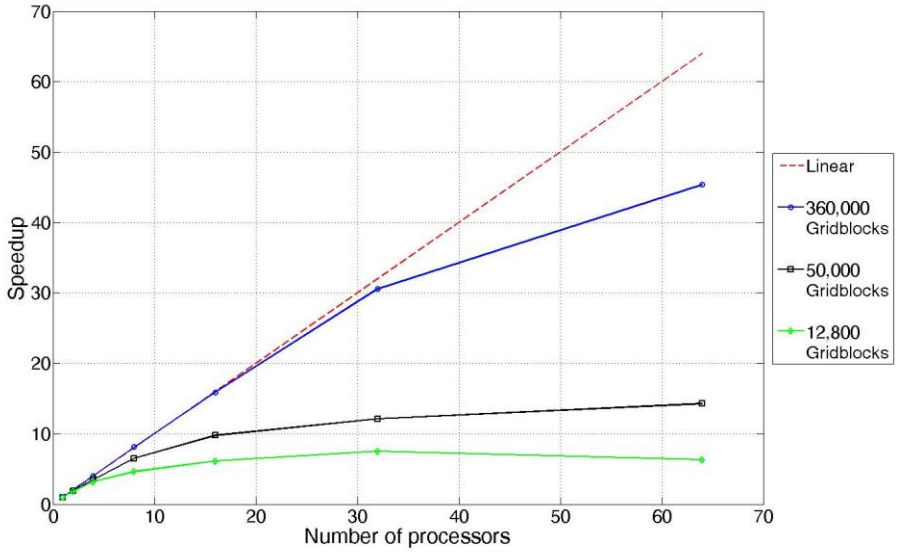

Figure 5-8 Speedup for different numbers of gridblocks (Case study 8)

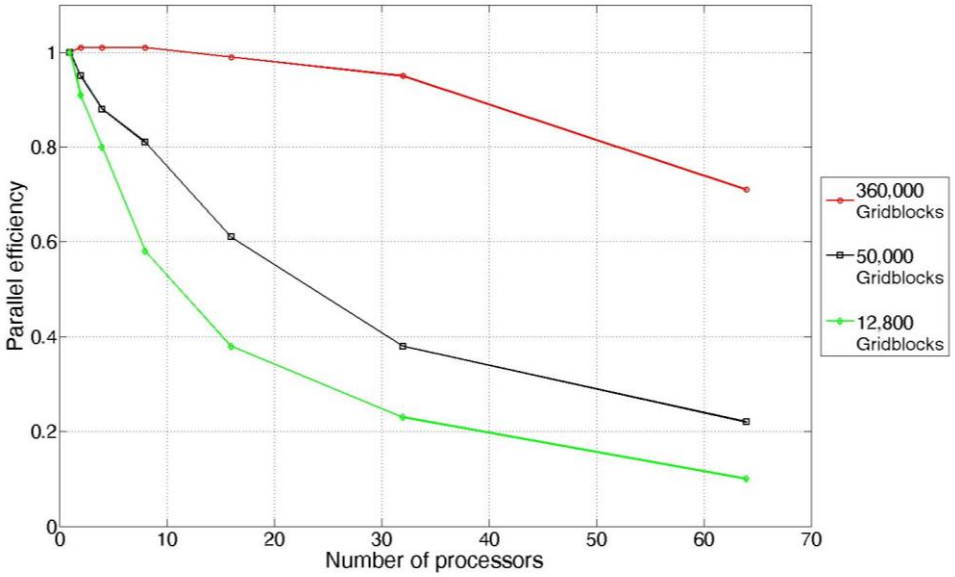

Figure 5-9 Parallel efficiency for different numbers of gridblocks (Case study 8)

5.1.4 Case Study 9: Surfactant/Polymer flooding

Some features of Case study 4 are modified, which can be seen in Table 5-7. Similar to Case study 4, two wells operate on this reservoir as can be seen in Figure 4-22. During the simulation time, which is 760 days, the injector is restricted to constant rate, while the producer operates under constant bottom hole pressure.

Table 5-7 Reservoir and fluid properties description (Case study 9)

Case 9 (Surfactant/Polymer flooding)			
Dimensions (ft)	Length	4000	
	Width	4000	
	Thickness	100	
Number of Gridblocks		640,000 (400x400x4)	
Number of components		10	
Max. number of phases		4	
Porosity (fraction)		0.2	
Permeability (md)	X	500	
	Y	500	
	Z	50	
Initial water saturation (fraction)		0.65	
Irreducible water saturation (fraction)		0.37	
Residual oil saturation (fraction)		0.35	
Water viscosity (cp)		0.86	
Oil viscosity (cp)		4	
Initial reservoir pressure (psi)		100	
Number of wells		2	1 Injector
			1 Producer
Injected polymer concentration (ppm)		500	
Injected surfactant concentration (volume fraction)		0.03	
Simulation time (days)		760	

The detailed computational times can be found in Table 5-8. Figure 5-10 also shows the contributions of different parts to the total computational time. In comparison to the previous results, the results are in agreement.

Table 5-8 Computational times for Case study 9

	1 Prc	2 Prc	4 Prc	8 Prc	16 Prc	32 Prc	64 Prc
Initialization	17	18	16	18	19	21	22
Transmissibility calculation	3241	1408	709	338	175	87	47
Concentration equation	12731	6489	3254	1664	822	402	209
Salinity calculation	730	364	189	96	50	28	16
Linear system solver	63224	31604	15602	7609	3789	1939	1119
Other	1150	876	586	414	312	212	203
Total	81093	40759	20356	10139	5167	2689	1616

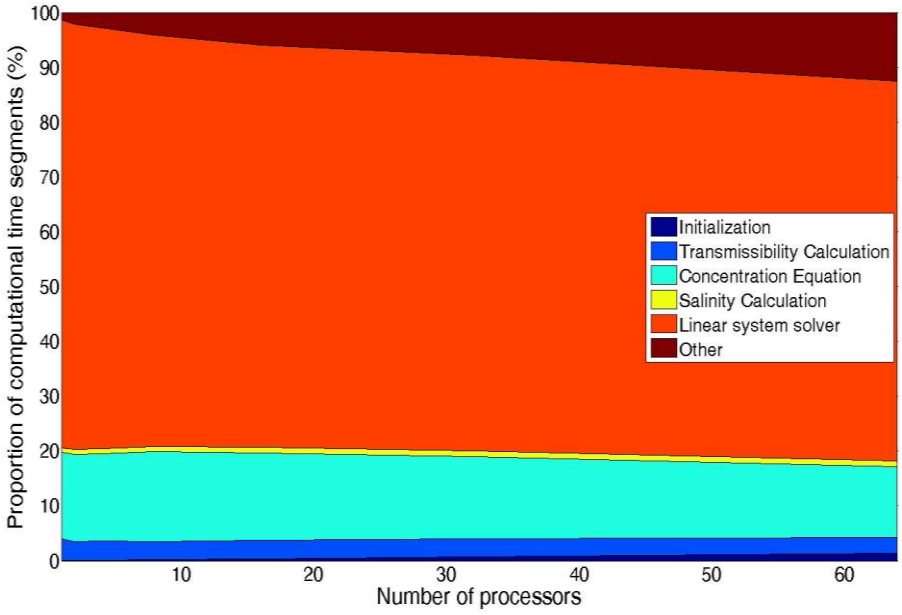

Figure 5-10 Contributions of different parts of the code to the total computational time (Case study 9)

Speedup and parallel efficiency are visualized in Figures 5-11 and 5-12. Similar to the previous case studies, parallel efficiency decreases with number of processors and increases with number of gridblocks.

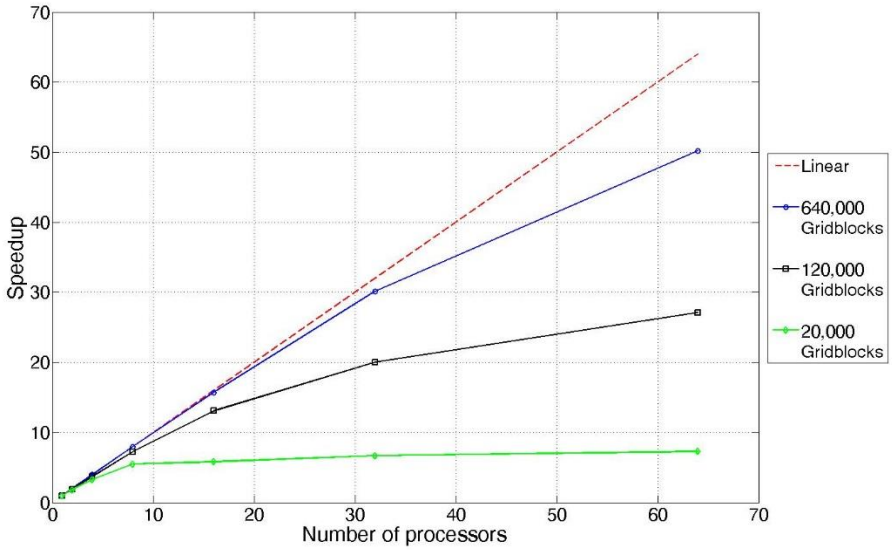

Figure 5-11 Speedup for different numbers of gridblocks (Case study 9)

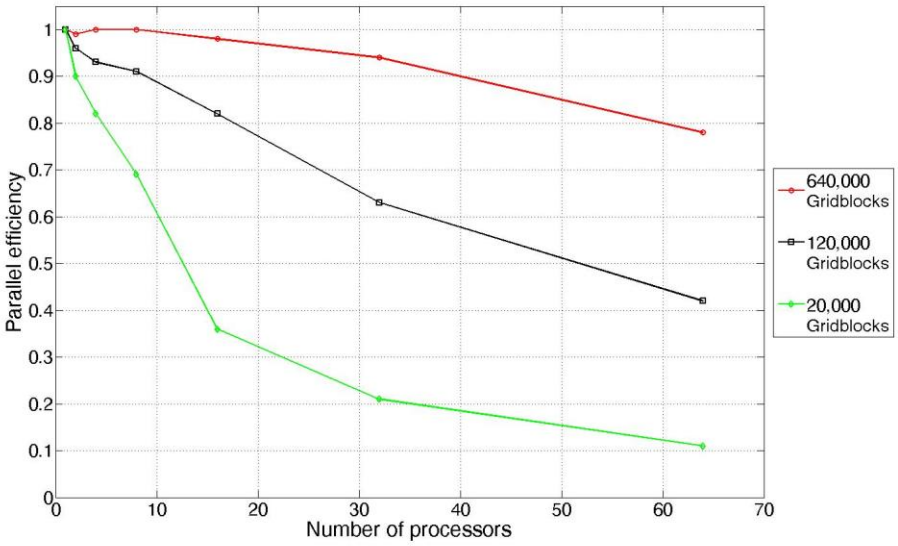

Figure 5-12 Parallel efficiency for different numbers of gridblocks (Case study 9)

5.1.5 Case Study 10: ASP Flood

Speedup performance of code is further tested for simulation results of an Alkaline/Surfactant/Polymer flood. Case study 5 is modified in this regard. The main features of the modified case can be found in Table 5-9. All of the operating wells are restricted to constant flow rate. The reservoir model is similar to Figure 4-28.

Table 5-9 Reservoir and fluid properties description (Case study 10)

Case 10 (ASP flooding)			
Dimensions (ft)	Length	3000	
	Width	3000	
	Thickness	150	
Number of Gridblocks		800,000 (200x200x6)	
Number of components		10	
Max. number of phases		3	
Porosity (fraction)		0.3	
Permeability (md)	Lateral	1600	
	Vertical	160	
Initial water saturation (fraction)		0.5	
Irreducible water saturation (fraction)		0.25	
Residual oil saturation (fraction)		0.15	
Water viscosity (cp)		0.46	
Oil viscosity (cp)		40	
Initial reservoir pressure (psi)		1750	
Number of wells		13	4 Injectors
			9 Producers
Simulation time (days)		475	

The simulation time is 475 days. The schedule of fluids injection to the reservoir is as follows (constant flow rate condition is applied on all the wells in each time period):

0 < time < 26 days: Alkaline and polymer dissolved in water is injected.

26 days < time < 51 days: A water solution of Alkaline and surfactant is injected.

51 days < time < 226 days: ASP solution is injected.

226 days < time < 276 days: Alkaline accompanied by polymer is injected.

276 days < time < 475 days: Brine water is injected into the reservoir.

Table 5-10 and Figure 5-13 present details of the computational times for this case study. As it was seen in the previous case studies, contribution of "Other" time increases with number of processors. It can also be mentioned that for this specific case, "Other" time includes the time for acidic crude reactions calculations (IREACT = 3), in addition to the other previously mentioned times. Hence, the contribution of "Other" time is more sharply compared to the other case studies here.

Table 5-10 Computational times for Case study 10

	1 Prc	2 Prc	4 Prc	8 Prc	16 Prc	32 Prc	64 Prc
Initialization	7	7	9	10	10	12	15
Transmissibility calculation	2741	1354	671	336	179	96	53
Concentration equation	12311	6179	3148	1685	875	472	288
Salinity calculation	570	301	175	108	62	37	24
Linear system solver	50275	25824	12978	6856	3460	1759	1038
Other	14313	7165	3767	2034	1055	613	511
Total	80217	40830	20748	11029	5641	2989	1929

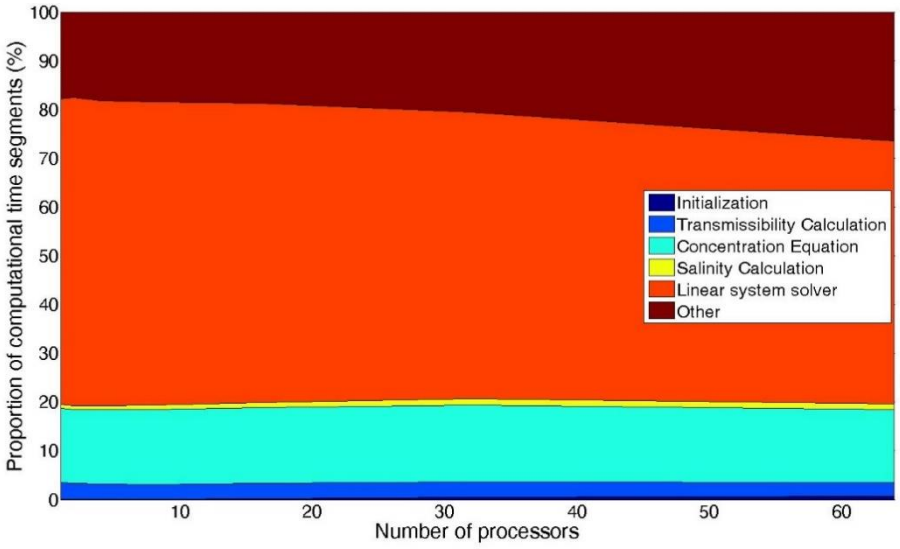

Figure 5-13 Contributions of different parts of the code to the total computational time (Case study 10)

Furthermore, Figures 5-14 and 5-15 show the results for speedup and parallel efficiency. Generally, the results are in agreement with the other case studies.

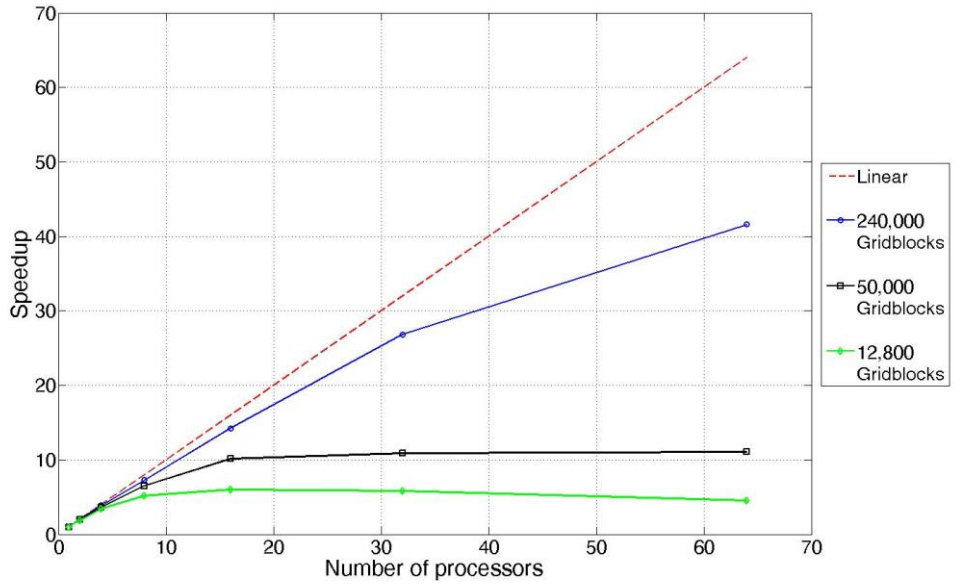

Figure 5-14 Speedup for different numbers of gridblocks (Case study 10)

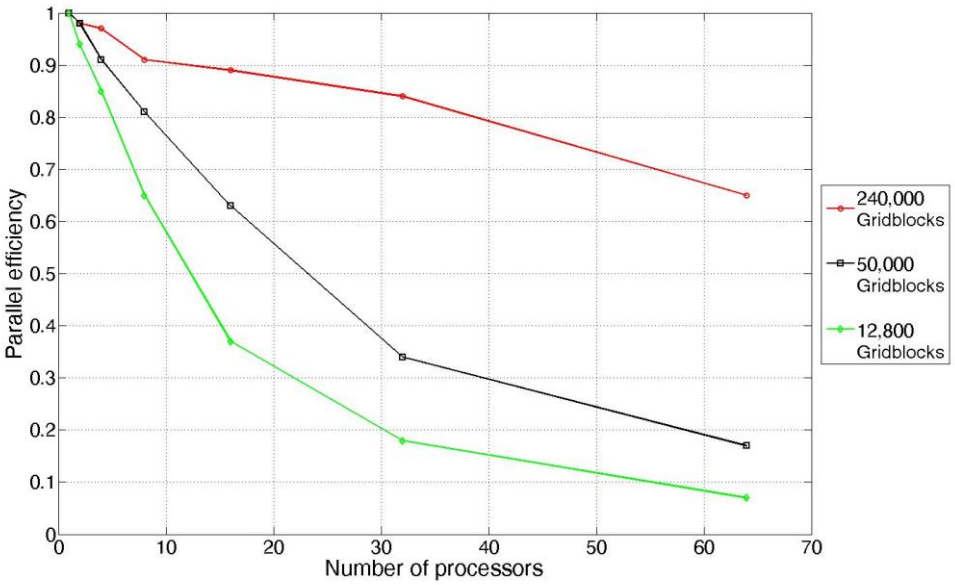

Figure 5-15 Parallel efficiency for different numbers of gridblocks (Case study 10)

5.2 LARGE-SCALE RESERVOIR SIMULATION

The simulation results for three large-scale cases are presented in this section.

5.2.1 Case Study 11: Polymer Flooding

The first large case study is a polymer flooding case. Approximately 1,750,000 gridblocks of the same size are used in this modeling. The reservoir and fluid properties are described in Table 5-11. In this reservoir, there are 41 wells operating with locations shown in Figure 5-16. The injectors are restricted to constant injection rate, while the producers are constrained to constant bottom hole pressure through the simulation time, which is 1200 days. Recovered oil at the end of the simulation is 17.6%.

Table 5-11 Reservoir and fluid properties description (Case study 11)

Case 11 (Polymer flooding)			
Dimensions (ft)	Length	3840	
	Width	3840	
	Thickness	48	
Number of Gridblocks		1,769,472 (384x384x12)	
Number of components		5	
Max. number of phases		4	
Porosity (fraction)		Heterogeneous 0.05 – 0.5	
Permeability (md)		Heterogeneous 50 - 1000	
Initial water saturation (fraction)		0.5	
Irreducible water saturation (fraction)		0.37	
Residual oil saturation (fraction)		0.35	
Water viscosity (cp)		0.86	
Oil viscosity (cp)		4	
Initial reservoir pressure (psi)		200	
Number of wells		41	16 Injectors
			25 Producers
Injection rate for each injection well (bbl/d)		2500	
Injected polymer concentration (ppm)		1500	
Bottom hole pressure for each producer well (psi)		200	
Simulation time (days)		1200	
Recovered oil at the end of simulation (%)		17.6	

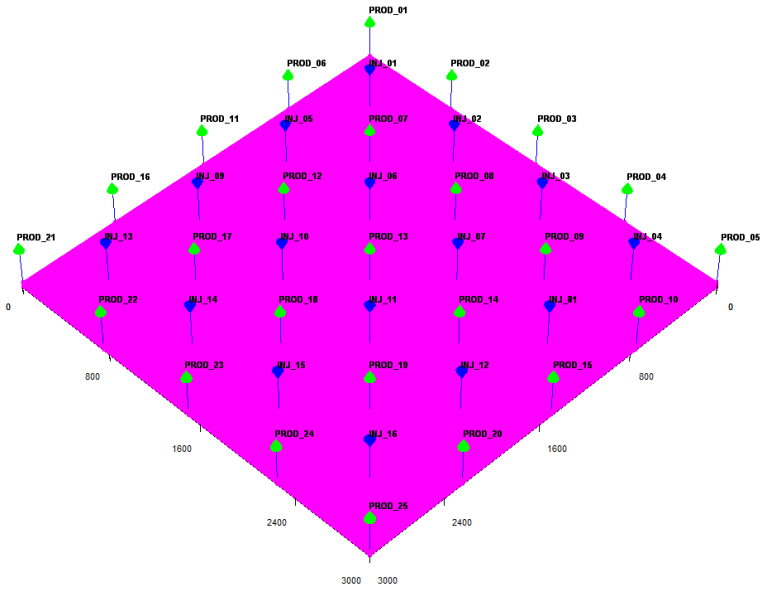

Figure 5-16 Well locations (Case Studies 11, 12, and 13). Blue markers represent injection wells, and green ones are producers.

Porosity and permeability of the reservoir are heterogeneous. Distribution of porosity is depicted in Figures 5-17 and 5-18, and the permeability distribution is shown in Figures 5-19 and 5-20. These distributions were created using Petrel.

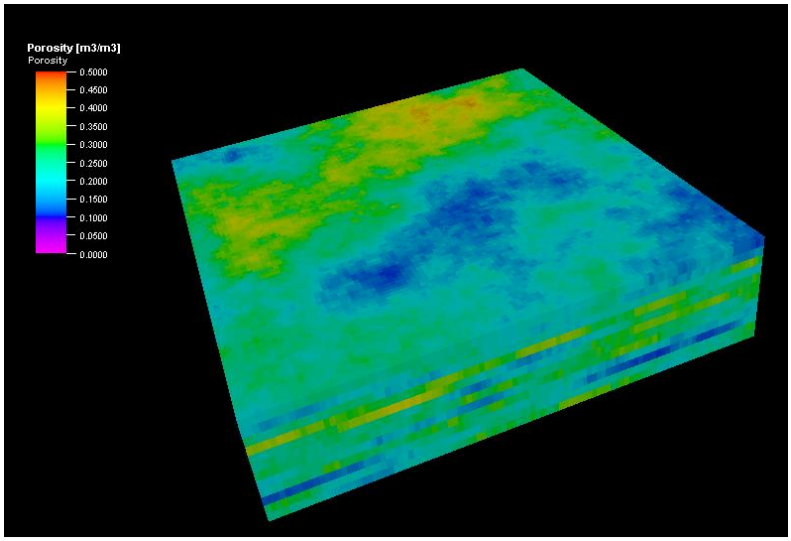

Figure 5-17 Porosity distribution (Case Study 11)

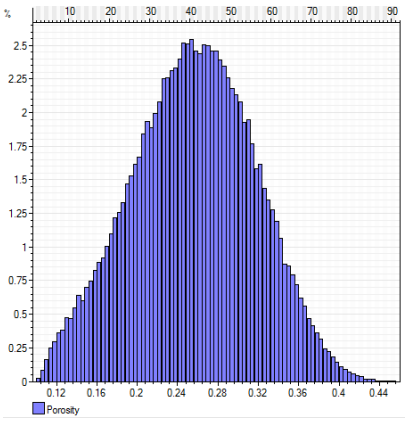

Figure 5-18 Normal distribution of porosity (Case Study 11)

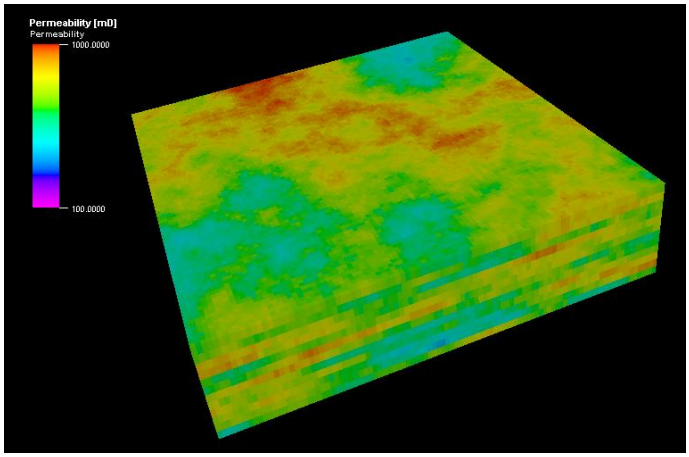

Figure 5-19 Permeability distribution (Case Study 11)

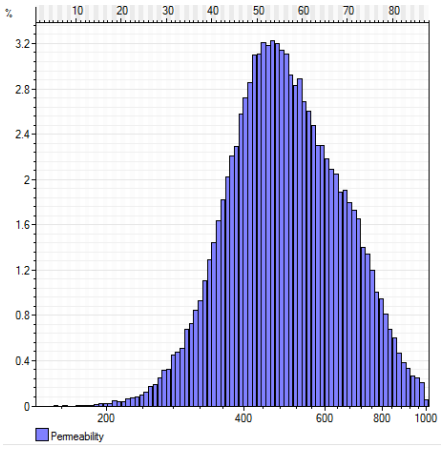

Figure 5-20 Log-normal distribution of permeability (Case Study 11)

It took 21 hours for the simulator to run this case using 56 processors. Some results including oil cut, average reservoir pressure, and oil production rate are plotted in Figures 5-21, 5-22, and 5-23, respectively.

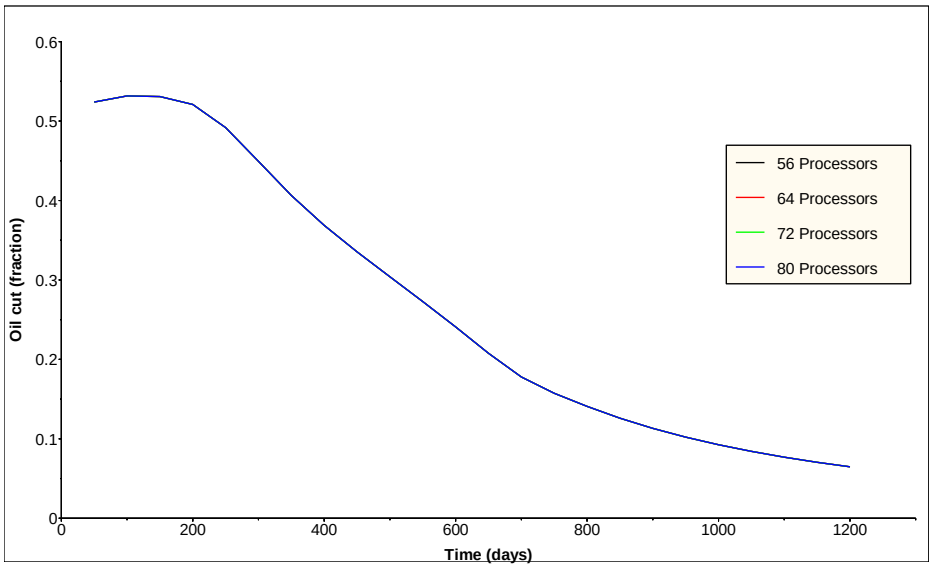

Figure 5-21 Oil cut (Case study 11)

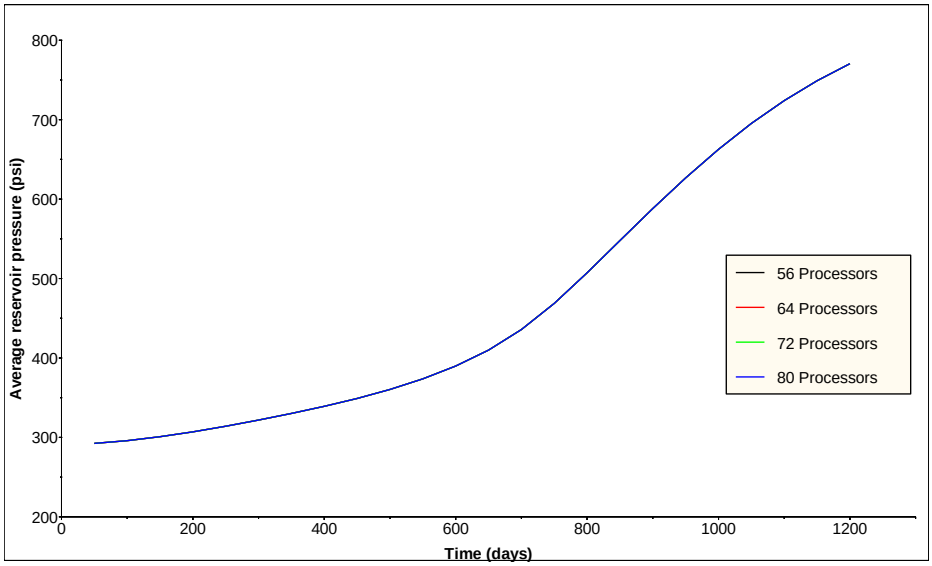

Figure 5-22 Average reservoir pressure (Case study 11)

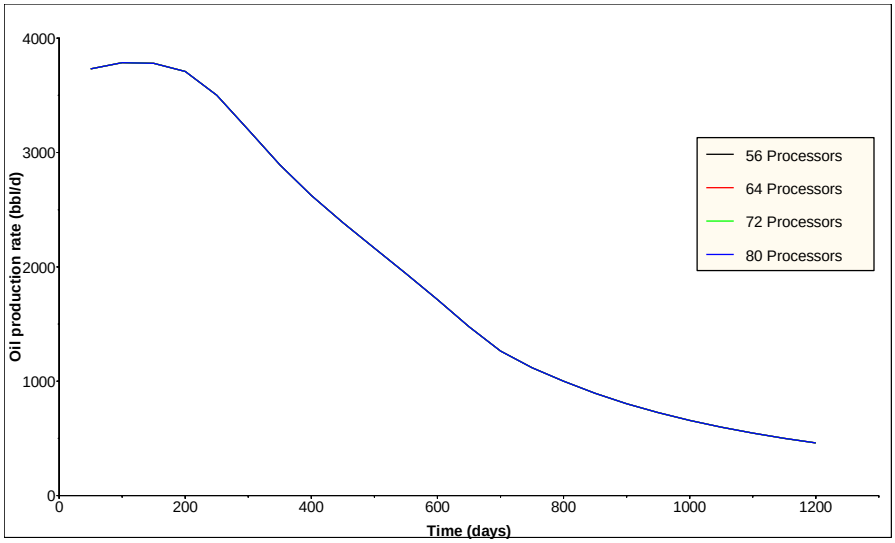

Figure 5-23 Oil production rate (Case study 11)

Speedup and parallel efficiency are shown in Figures 5-24 and 5-25, respectively. Because of the time limitation using the TACC cluster, the base case here involves 56 processors; the speedup performance using more processors is compared to that case. As it can be seen, speedup is close to linear.

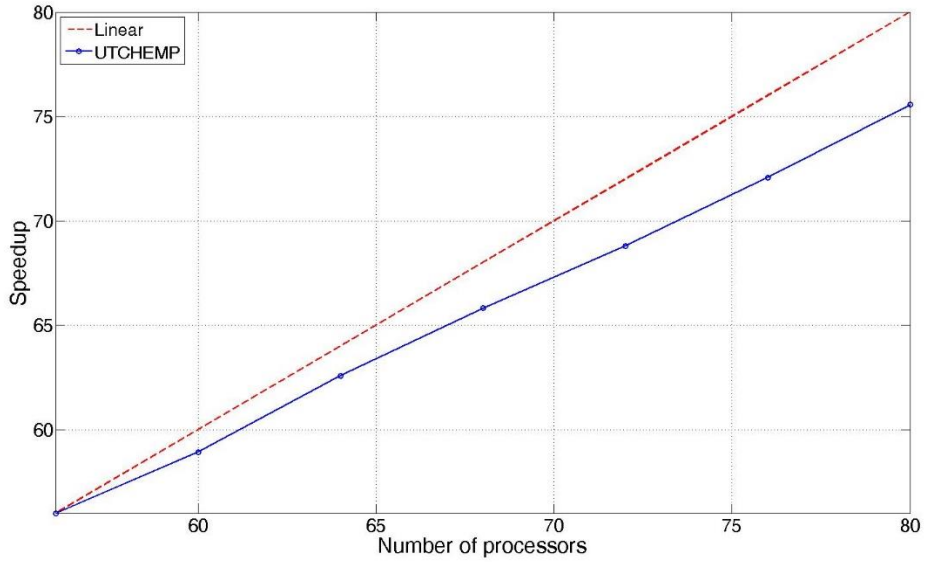

Figure 5-24 Speedup (Case study 11)

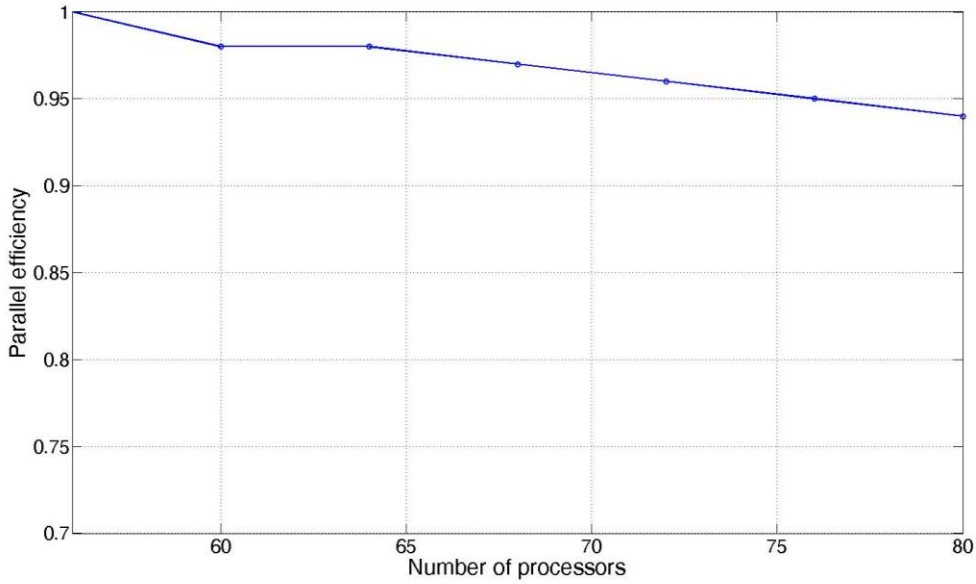

Figure 5-25 Parallel efficiency (Case study 11)

5.2.2 Case Study 12: Surfactant/Polymer Flooding

The second field-scale case is to model a surfactant/polymer flooding process. The number of gridblocks used for this study is 1,600,000. The main features are provided in Table 5-12. Constant gridblock size option is used in this case. The well pattern can be seen in Figure 5-16. The reservoir is homogeneous, and the ratio of lateral permeability to vertical equals 5. All of the operating injection wells are constrained to constant fluid injection rate, and the producers operate under the condition of constant bottom hole pressure. These conditions do not change during the entire simulation time, which is 1620 days. The oil recovery at the end of the simulation is 52.2%.

The computational run time using 56 processors is approximately 22 hours. Figures 5-26, 5-27, and 5-28 show oil cut, average reservoir pressure, and oil production rate for this case, respectively. As expected, the results for different numbers of processors match.

Table 5-12 Reservoir and fluid properties description (Case study 12)

Case 12 (Surfactant/Polymer flooding)			
Dimensions (ft)	Length	3000	
	Width	3000	
	Thickness	50	
Number of Gridblocks		1,600,000 (400x400x10)	
Number of components		6	
Max. number of phases		4	
Porosity (fraction)		0.2	
Permeability (md)	Lateral	500	
	Vertical	100	
Initial water saturation (fraction)		0.55	
Irreducible water saturation (fraction)		0.37	
Residual oil saturation (fraction)		0.35	
Water viscosity (cp)		0.86	
Oil viscosity (cp)		4	
Initial reservoir pressure (psi)		200	
Number of wells		41	16 Injectors
			25 Producers
Injection rate for each injection well (bbl/d)		2000	
Polymer concentration in the injected fluid (ppm)		1000	
Volume fraction of surfactant in the injected fluid		0.02	
Bottom hole pressure for each producer well (psi)		200	
Simulation time (days)		1620	
Recovered oil at the end of simulation (%)		52.2	

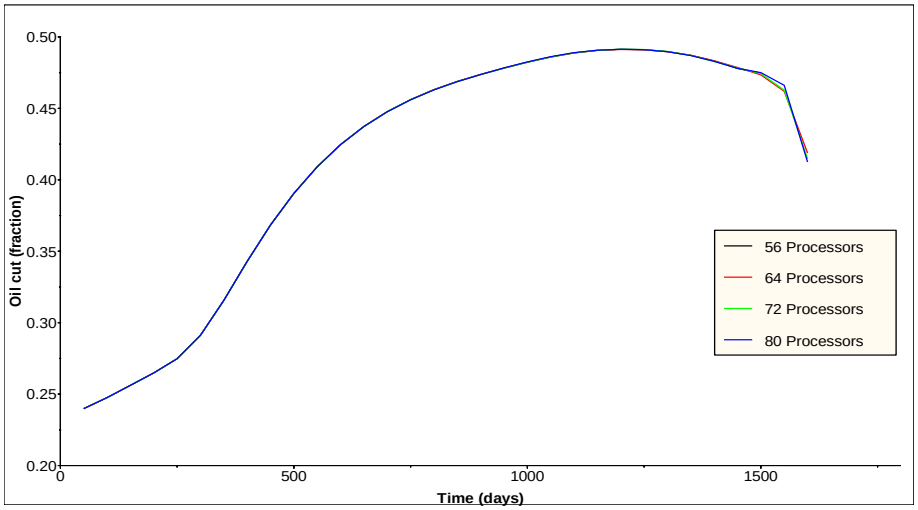

Figure 5-26 Oil cut (Case study 12)

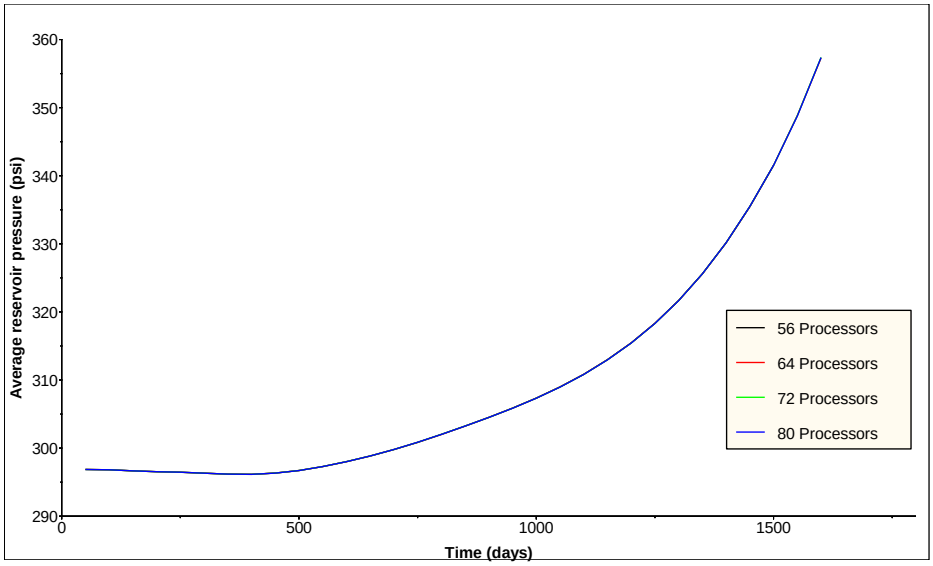

Figure 5-27 Average reservoir pressure (Case study 12)

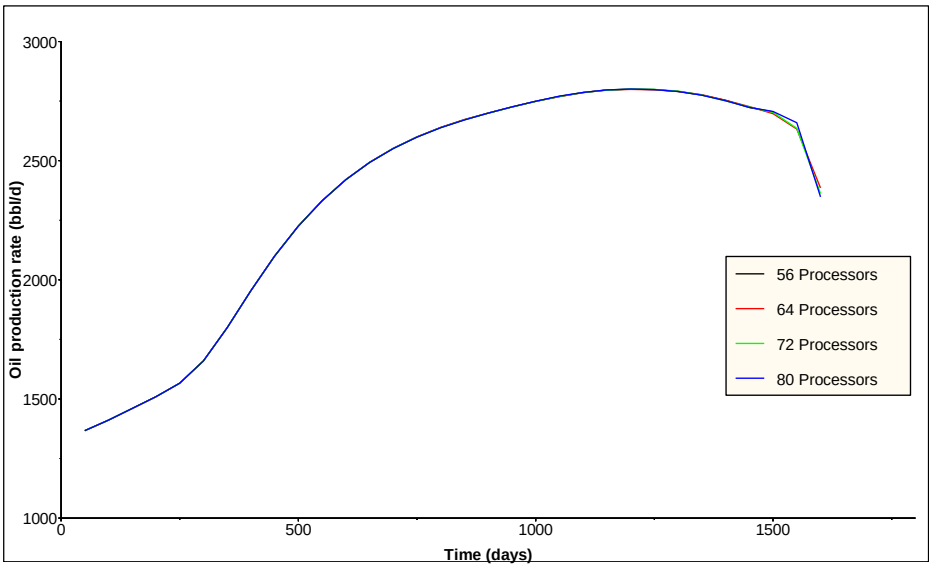

Figure 5-28 Oil production rate (Case study 12)

Figures 5-29 and 5-30 depict the speedup and parallel efficiency, respectively. We observe a super-linear speedup compared to the base case, which is 56 processors in this case. It was noted before that super-linear speedup can sometimes be achieved due to hardware issues.

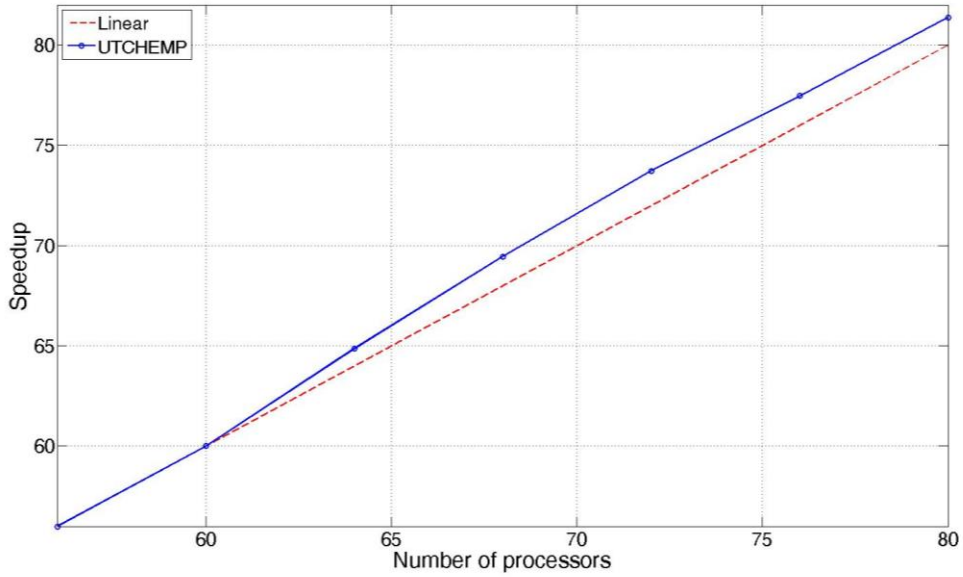

Figure 5-29 Speedup (Case study 12)

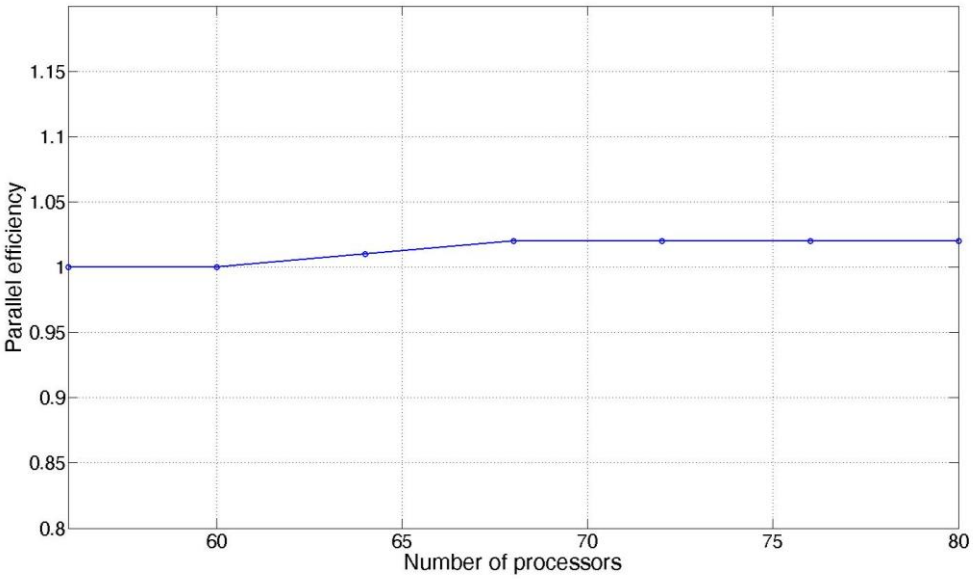

Figure 5-30 Parallel efficiency (Case study 12)

5.2.3 Case Study 13: Water Flooding

This is the largest case we tried to simulate. Two and a half million gridblocks of the same size are used to model a water flooding process. Table 5-13 describes the main features for this case study. Figure 5-16 visualizes the well pattern for this case. During the simulation time of 1500 days, all of the production wells are restricted to constant bottom hole pressures, while the injectors are constrained to constant fluid injection rates. Porosity and permeability are distributed uniformly over the reservoir. 47.6% of oil is recovered at the end of the simulation.

Table 5-13 Reservoir and fluid properties description (Case study 13)

Case 13 (Water flooding)			
Dimensions (ft)	Length	3500	
	Width	3500	
	Thickness	70	
Number of Gridblocks		2,500,000 (500x500x10)	
Number of components		4	
Max. number of phases		3	
Porosity (fraction)		0.2	
Permeability (md)	Lateral	500	
	Vertical	100	
Initial water saturation (fraction)		0.321	
Irreducible water saturation (fraction)		0.32	
Residual oil saturation (fraction)		0.3	
Water viscosity (cp)		0.86	
Oil viscosity (cp)		2.2	
Initial reservoir pressure (psi)		200	
Number of wells		41	16 Injectors
			25 Producers
Injection rate for each injection well (bbl/d)		2500	
Bottom hole pressure for each producer well (psi)		200	
Simulation time (days)		1500	
Recovered oil at the end of simulation (%)		47.6	

This simulation was completed in 22.5 hours using 24 processors. Simulation results including oil cut, average reservoir pressure, and oil production rate are shown in Figures 5-31, 5-32, and 5-33, respectively. Results for different numbers of processors perfectly match.

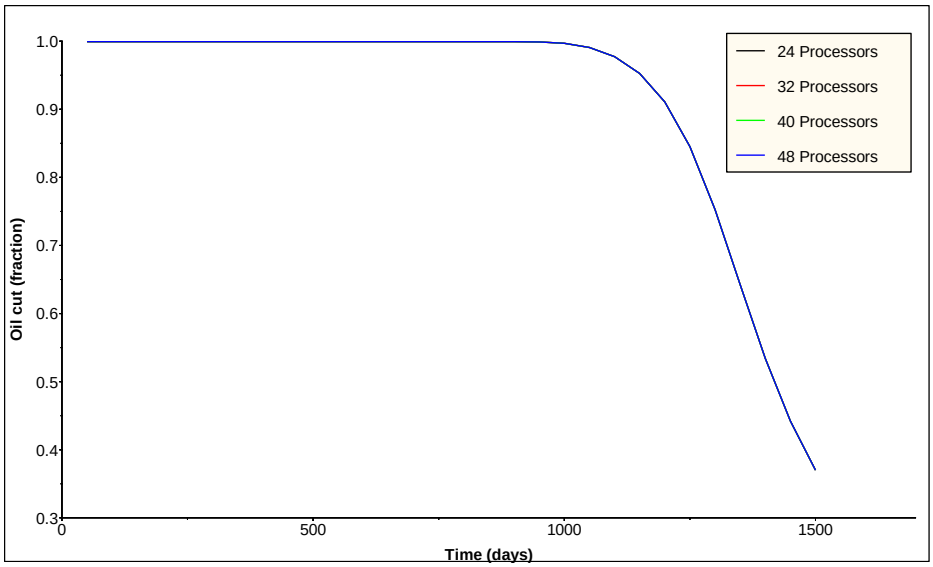

Figure 5-31 Oil cut (Case study 13)

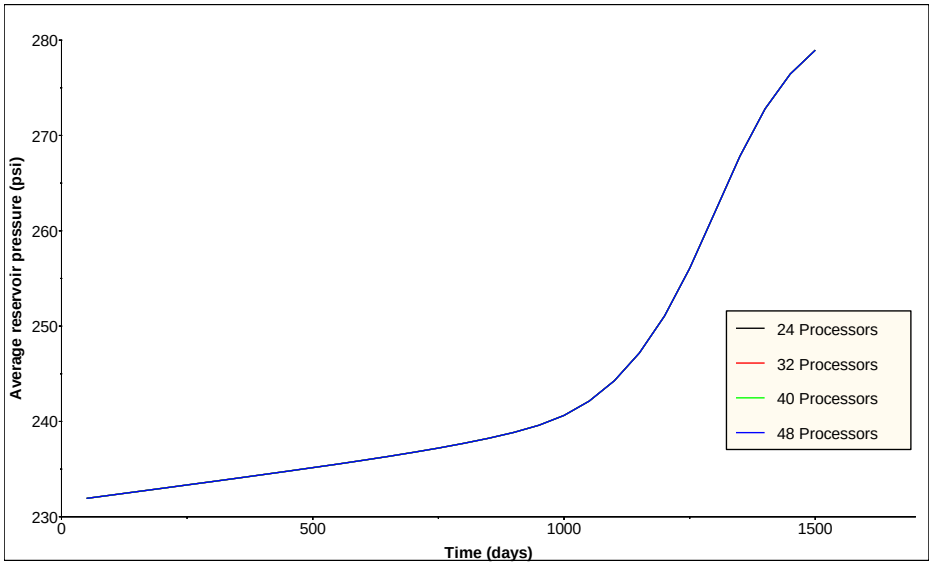

Figure 5-32 Average reservoir pressure (Case study 13)

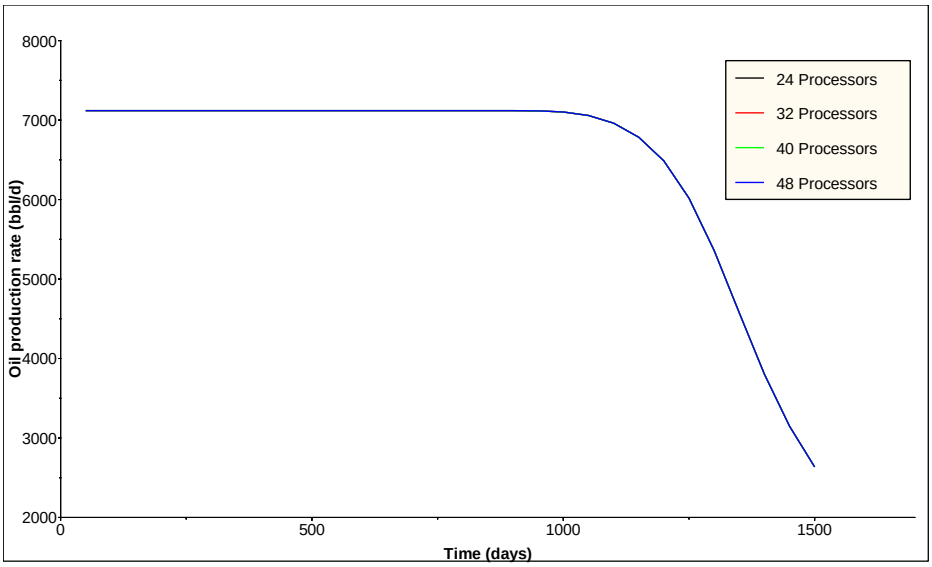

Figure 5-33 Oil production rate (Case study 13)

Moreover, speedup and parallel efficiency curves can be seen in Figures 5-34 and 5-35, respectively. A very good speedup performance is observed for this large-scale case.

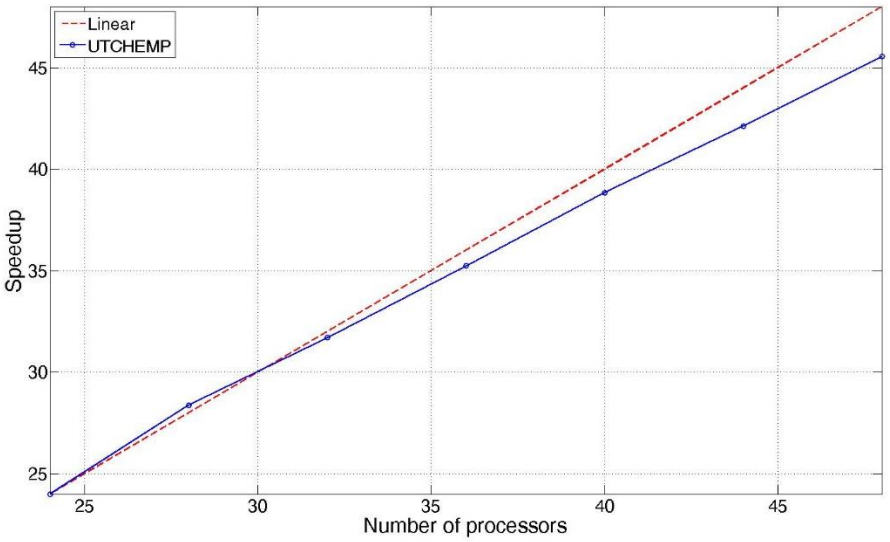

Figure 5-34 Speedup (Case study 13)

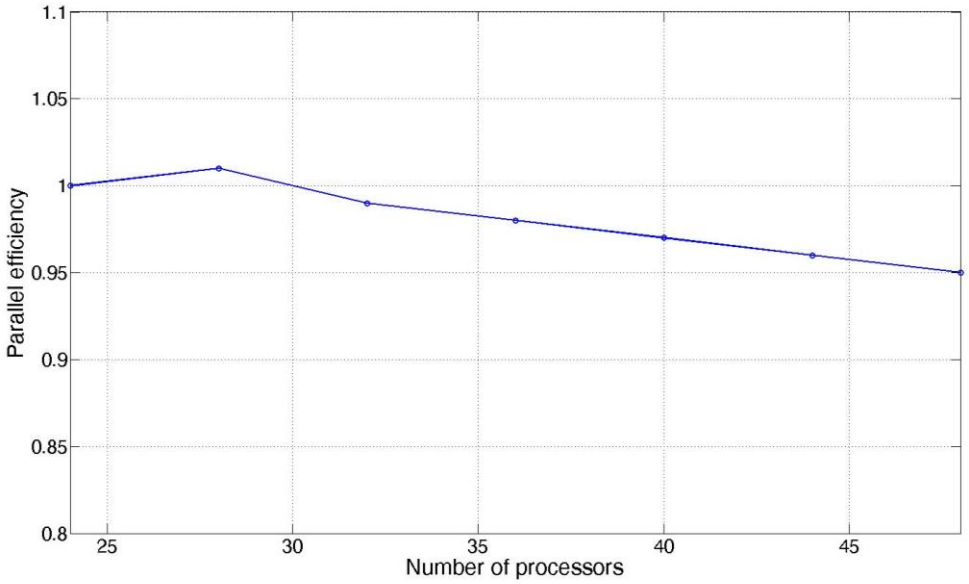

Figure 5-35 Parallel efficiency (Case study 13)

Chapter 6: Summary, Conclusions, and Recommendations

In this chapter, summary of the tasks performed in this work and conclusions of this thesis are provided. Moreover, several recommendations are presented for future extension of this work.

6.1 SUMMARY

In this work, the following tasks were performed.

- Parallel version of a chemical compositional reservoir simulator was developed using a domain decomposition methodology.
- PETSc is the parallel solver package used to solve the system of linear equations.
- IPARS framework was used in providing several operations for parallelization of the code.
- Several new features were included in the code, such as key-word based input file format and inactive gridblocks option.
- Execution of the parallel simulator was successfully performed in both distributed and shared-memory systems.
- Validation of the parallel code was successfully performed using various case studies.
- The performance of the parallel code was assessed in terms of speedup and parallel efficiency.
- Simulation of several large-scale problems, up to millions of gridblocks, was successfully performed by application of the parallel simulator.

6.2 CONCLUSIONS

The conclusions of this work are as follows:

- Application of parallel processing can be effective in reservoir simulation. Considerable amount of time and memory can be saved by parallelization of reservoir simulators.
- Parallelization of reservoir simulators in distributed-memory computers can be efficiently done using Message Passing Interface (MPI).
- Very good agreement of the results for the parallel code and the original code was obtained.
- Excellent speedup performance for the parallel code was observed by obtaining the speedup curves.
- Application of the parallel simulator becomes more efficient as the number of gridblocks increases. In other words, parallel efficiency increases with size of the problem.
- Parallel efficiency decreases as the number of processors increases due to increase in the communication among processors.
- It was observed that the time spent on solving the linear system of equation is the main contributor to the overall computational time.

6.3 RECOMMENDATIONS FOR FUTURE WORK

- Unstructured gird and corner point should be implemented in the code.
- X and Z directions should also be considered in domain decomposition. Then, for any given case, highest parallel efficiency possibly can be achieved by selecting the best possible decomposition approach, which may reduce the communication time.

- When a reservoir model includes a large number of randomly distributed inactive gridblocks, alternative efficient domain decomposition should be taken into account. Active gridblocks should roughly be distributed in equal numbers amongst the processors in that approach.
- Further research should be performed to implement other possible time-stepping methods, such as IMPSAT (implicit pressure and implicit saturation). Subsequently, the advantage of such time-stepping approaches in conjunction with parallel processing can be investigated by performing a comparison study.

Appendix A: Sample Input Data for UTCHEMP

A.1 SURFACTANT/POLYMER FLOODING (CASE STUDY 4)

```
TITLE(2) = "CASE 04"

DESCRIPTION() = "SURFACTANT/POLYMER FLOOD TEST, 11X11X2"
""
"LENGTH (FT) : 250            PROCESS : SURFACTANT/POLYMER"
"THICKNESS (FT) : 10          INJ. RATE (FT3/DAY) : 112.3"
"WIDTH (FT) : 250             COORDINATES : CARTESIAN"
"POROSITY : 0.20"
"GRID BLOCKS : 11X11X2"

OUTLEVEL 2
IOUTPUT 1

ISOLVER = 4


"            RESERVOIR DESCRIPTION              "


RUNNO = "CASE04"
OUTNO = "CASE04"

"SIMULATION FLAGS"
IMODE = 1
IMES = 2
IDISPC = 3
ICWM = 0
ICAP = 0
IBIO = 0
ICOORD = 1
ITREAC = 0
ITC = 0
IGAS = 0
IENG = 0

"NO. OF GRIDBLOCKS, UNIT"
NX() = 11
NY() = 11
NZ() = 2
IUNIT = 0

"GRIDBLOCKS DIMENSIONS"
DX() = 22.727
DY() = 22.727
DZ() = 5.0

"TOTAL NO. OF COMPONENTS, NO. OF TRACERS, NO. OF GEL COMPONENTS"
N = 11
NO = 0
```

NPHAS = 4
NTW = 3
NTA = 0
NGC = 0
NG = 0
NOTH = 0

"NAME OF SPECIES"
SPNAME(1) = "WATER"
SPNAME(2) = "OIL"
SPNAME(3) = "SURF."
SPNAME(4) = "POLYMER"
SPNAME(5) = "ANION"
SPNAME(6) = "CALCIUM"
SPNAME(7) = "ALCOHOL"
SPNAME(8) = "GAS(NO)"
SPNAME(9) = "TRACER 1"
SPNAME(10) = "TRACER 2"
SPNAME(11) = "TRACER 3"

"FLAG INDICATING IF THE COMPONENT IS INCLUDED IN CALCULATIONS OR NOT"
ICF() = 1 1 1 1 1 1 1 0 1 1 1

" OUTPUT OPTIONS "

"FLAG FOR PV OR DAYS TO PRINT OR TO STOP THE RUN"
ICUMTM = 0
ISTOP = 0
IOUTGMS = 0
IS3GRF = 1

"FLAG INDICATING IF THE PROFILE OF KCTH COMPONENT SHOULD BE WRITTEN"
IPRFLG() = 1 1 1 1 0 0 0 0 0 0 0

"FLAG FOR individual map files"
IPPRES = 1
IPSAT = 1
IPCTOT = 1
IPBIO = 0
IPCAP = 0
IPGEL = 0
IPALK = 0
IPTEMP = 0
IPOBS = 0

"FLAG for individual output map files"
ICKL = 1
IVIS = 1
IPER = 1
ICNM = 1
ICSE = 1
IHYSTP = 0

```
IFOAMP = 0
INONEQ = 0

"FLAG  for variables to PROF output file"
IADS = 0
IVEL = 0
IRKF = 0
IPHSE = 0


"              RESERVOIR PROPERTIES                "


"MAX. SIMULATION TIME ( DAYS)"
TMAX = 1500

"ROCK COMPRESSIBILITY (1/PSI), STAND. PRESSURE(PSIA)"
COMPR = 0
PSTAND = 0

"FLAGS INDICATING CONSTANT OR VARIABLE POROSITY, X,Y,AND Z PERMEABILITY"
IPOR1 = 0
IPERMX = 0
IPERMY = 0
IPERMZ = 0
IMOD = 0
ITRANZ = 0
INTG = 0

"POROSITY"
POR1() = 0.2

"X-PERMEABILITY (MILIDARCY)"
PERMX1() = 500.0

"Y-PERMEABILITY (MILIDARCY)"
PERMY1() = 500.0

"Z-PERMEABILITY (MILIDARCY)"
PERMZ1() = 50.0

"FLAG FOR CONSTANT OR VARIABLE DEPTH, PRESSURE, WATER SATURATION"
IDEPTH = 0
IPRESS = 0
ISWI = 0
ICWI = -1

"DEPTH (FT)"
EL1() = 0.0

"PRESSURE (PSIA)"
P1(,,,) = 700

"INITIAL WATER SATURATION"
```

```
S1(,,,1) = 0.65

"Salinity and divalent cation concentration of brine"
C50 = 0.4
C60 = 0.003


"              PHYSICAL PROPERTY DATA                "


"OIL CONC. AT PLAIT POINT FOR TYPE II(+)AND TYPE II(-), CMC"
C2PLC = 0.0
C2PRC = 1.0
EPSME = 0.0001
IHAND = 0

"FLAG INDICATING TYPE OF PHASE BEHAVIOR PARAMETERS"
IFGHBN = 0

"SLOPE AND INTERCEPT OF BINODAL CURVE AT ZERO, OPT., AND 2XOPT SALINITY FOR ALCOHOL 1"
HBNS70 0.131
HBNC70 0.1
HBNS71 0.191
HBNC71 0.026
HBNS72 0.363
HBNC72 0.028

"SLOPE AND INTERCEPT OF BINODAL CURVE AT ZERO, OPT., AND 2XOPT SALINITY FOR ALCOHOL 2"
HBNS80 0.0
HBNC80 0.0
HBNS81 0.0
HBNC81 0.0
HBNS82 0.0
HBNC82 0.0

"LOWER AND UPPER EFFECTIVE SALINITY FOR ALCOHOL 1 AND ALCOHOL 2"
CSEL7  0.177
CSEU7  0.344
CSEL8  0.0
CSEU8  0.0

"THE CSE SLOPE PARAMETER FOR CALCIUM AND ALCOHOL 1 AND ALCOHOL 2"
BETA6  0.8
BETA7  -2.0
BETA8  0.0

"FLAG FOR ALCOHOL PART. MODEL AND PARTITION COEFFICIENTS"
IALC  1.0
OPSK7O 0.0
OPSK7S 0.0
OPSK8O 0.0
OPSK8S 0.0
```

"NO. OF ITERATIONS, AND TOLERANCE"
NALMAX = 20
EPSALC = 0.0001

"ALCOHOL 1 PARTITIONING PARAMETERS IF IALC=1"
AKWC7 4.671
AKWS7 1.79
AKM7 48.
AK7 35.31
PT7 .222

"ALCOHOL 2 PARTITIONING PARAMETERS IF IALC=1"
AKWC8 0.0
AKWS8 0.0
AKM8 0.0
AK8 0.0
PT8 0.0

"IFT MODEL FLAG"
IFT = 0

"INTERFACIAL TENSION PARAMETERS"
G11 13.
G12 -14.8
G13 .007
G21 13.
G22 -14.5
G23 .010

"LOG10 OF OIL/WATER INTERFACIAL TENSION"
XIFTW 1.3

"MASS TRANSFER FLAG"
IMASS = 0
ICOR = 0

"WETTABILITY ALTERATION FLAGS"
IWALT = 0
IWALF = 0

"CAPILLARY DESATURATION PARAMETERS FOR PHASE 1, 2, AND 3"
ITRAP = 1
T11() = 1865.
T22() = 59074
T33() = 364.2

"RELATIVE PERM. FLAG (0:IMBIBITION COREY,1:FIRST DRAINAGE COREY)"
IPERM = 0
IRTYPE = 0

"FLAG FOR CONSTANT OR VARIABLE REL. PERM. PARAMETERS"
ISRW = 0
IPRW = 0
IEW = 0

"RES. SATURATION OF PHASES 1,2,AND 3 AT HIGH CAPILLARY NO."
S1RC = 0.0
S2RC = 0.0
S3RC = 0.0

"RES. SATURATION OF PHASES 1,2,AND 3 AT LOW CAPILLARY NO."
S1RW1() = .37
S2RW1() = .35
S3RW1() = .37

"ENDPOINT REL. PERM. OF PHASES 1,2,AND 3 AT HIGH CAPILLARY NO."
P1RC = 1.0
P2RC = 1.0
P3RC = 1.0

"ENDPOINT REL. PERM. OF PHASES 1,2,AND 3 AT LOW CAPILLARY NO."
P1RW1() = .11
P2RW1() = .95
P3RW1() = .11

"REL. PERM. EXPONENT OF PHASES 1,2,AND 3 AT HIGH CAPILLARY NO."
E13C = 1.0
E23C = 2.16
E31C = 1.0

"REL. PERM. EXPONENT OF PHASES 1,2,AND 3 AT LOW CAPILLARY NO."
E1W1() = 1.0
E2W1() = 2.16
E3W1() = 1.0

"WATER AND OIL VISCOSITY"
VIS1 = 0.86
VIS2 = 4.0

"VISCOSITY PARAMETERS"
ALPHAV() = 2.5 2.3 10. 1. 1.

"PARAMETERS TO CALCULATE POLYMER VISCOSITY AT ZERO SHEAR RATE"
AP1 81.
AP2 2700.
AP3 2500.

"PARAMETER TO COMPUTE CSEP,MIN. CSEP, AND SLOPE OF LOG VIS. VS. LOG CSEP "
BETAP 10.
CSE1 .01
SSLOPE .17

"PARAMETER FOR SHEAR RATE DEPENDENCE OF POLYMER VISCOSITY"
GAMMAC 20.
GAMHF 10.
POWN 1.8
IPMOD = 0
ISHEAR = 0
RWEFF = 0.5

GAMHF2 = 0.5

"FLAG FOR POLYMER PARTITIONING, PERM. REDUCTION PARAMETERS"
IPOLYM 1
EPHI3 1.
EPHI4 0.8
BRK 1000.
CRK 0.0186
RKCUT 10

"SPECIFIC WEIGHT FOR COMPONENTS 1,2,3,7,AND 8 , AND GRAVITY FLAG"
DEN1 .433
DEN2 .368
DEN23 0.368
DEN3 .42
DEN7 .346
DEN8 0.
IDEN 2

"FLAG FOR CHOICE OF UNITS (0:BOTTOMHOLE CONDITION , 1: STOCK TANK)"
ISTB = 0

"COMPRESSIBILITY FOR VOL. OCCUPYING COMPONENTS"
COMPC() = 0

"CONSTANT OR VARIABLE PC PARAM., WATER-WET OR OIL-WET PC CURVE FLAG"
ICPC = 0
IEPC = 0
IOW = 0

"CAPILLARY PRESSURE PARAMETERS"
CPC1() = 9.0
EPC1() = 2.0

"MOLECULAR DIFFUSION COEF. KCTH COMPONENT IN PHASE J"
D() = 0.0

"LONGITUDINAL AND TRANSVERSE DISPERSIVITY OF PHASE J"
ALPHAL() = 12.
ALPHAT() = .4

"FLAG TO SPECIFY ORGANIC ADSORPTION CALCULATION"
IADSO = 0

"SURFACTANT AND POLYMER ADSORPTION PARAMETERS"
AD31 1.5
AD32 .5
B3D 1000.
AD41 0.7
AD42 0.
B4D 100.
IADK 0
IADS1 0
FADS 0

```
REFK 500

"PARAMETERS FOR CATION EXCHANGE OF CLAY AND SURFACTANT"
QV 0.044
XKC .25
XKS .2
EQW 419.

"TRACER PARTITIONING COEFFICIENT (TK(IT),IT=1,NT)"
TK(1) = 0
TK(2) = 1
TK(3) = 0

"SALINITY DEPENDENCE PART. COEFF., ref. salinity"
TKS() = 0.
C5INI = 0.4

"RADIACTIVE DECAY COEFFICIENT (RDC(IT),IT=1,NT)"
RDC() = 0.

"TRACER RETARDATION COEFFICIENT (RET(IT),IT=1,NT)"
RET() = 0.


"              WELL DATA                "


"FLAG FOR RIGHT AND LEFT BOUNDARY"
IBOUND = 0
IZONE = 0


"TOTAL NUMBER OF WELLS, WELL RADIUS FLAG, FLAG FOR TIME OR COURANT NO."
NWELL = 2
IRO = 2
ITSTEP = 1
NWREL = 2

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN"
IDW(1) = 1
IW(1) = 1
JW(1) = 1
IFLAG(1) = 1
RW(1) = .5
SWELL(1) = 0.
IDIR(1) = 3
IFIRST(1) = 1
ILAST(1) = 2
IPRF(1) = 0

"WELL NAME"
WELNAM(1) = "INJECTOR"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
```

```
ICHEK(1) = 2
PWFMIN(1) = 0.0
PWFMAX(1) = 5000.
QTMIN(1) = 0.0
QTMAX(1) = 1000.

"INJ. RATE AND INJ. COMP. FOR RATE CONS. WELLS FOR EACH PHASE (L=1,3)"
QI(1,1) = 112.3
C_W(1,,1) =  .94  0.  0.03  .05  0.3  0.001  0.03  0.  1. 1.  1.

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN"
IDW(2) = 2
IW(2) = 11
JW(2) = 11
IFLAG(2) = 2
RW(2) = .5
SWELL(2) = 0.
IDIR(2) = 3
IFIRST(2) = 1
ILAST(2) = 2
IPRF(2) = 0

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
WELNAM(2) = "PRODUCER"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
ICHEK(2) = 2
PWFMIN(2) = 0.0
PWFMAX(2) = 5000.
QTMIN(2) = 0.0
QTMAX(2) = 50000.

"BOTTOM HOLE PRESSURE FOR PRESSURE CONSTRAINT WELL (IFLAG=2 OR 3)"
PWF(2) = 700

"CUM. INJ. TIME , AND INTERVALS (PV OR DAY) FOR WRITING TO OUTPUT FILES"
TINJ = 1550
CUMPR1 = 50
CUMHI1 = 50
WRHPV = 10
WRPRF = 5
RSTC = 1550

"FOR IMES=2 ,THE INI. TIME STEP,CONC. TOLERANCE,MAX.,MIN. COURANT NUMBERS"
DT 0.05
DCLIM 0.003
CNMAX 0.2
CNMIN 0.01
```

A.2 ASP FLOODING (CASE STUDY 5)

```
TITLE(2) = "3-D PILOT SCALE ASP FLOOD"

DESCRIPTION() = "ALKALINE/SURFACTANT/POLYMER FLOOD PILOT SCALE TEST "

OUTLEVEL 2
IOUTPUT 1

ISOLVER = 4


"            RESERVOIR DESCRIPTION           "


RUNNO = "CASE05"
OUTNO = "CASE05"

"SIMULATION FLAGS"
IMODE = 1
IMES = 2
IDISPC = 3
ICWM = 0
ICAP = 0
IREACT = 3
IBIO = 0
ICOORD = 1
ITREAC = 0
ITC = 0
IGAS = 0
IENG = 0

"NO. OF GRIDBLOCKS, UNIT"
NX() = 19
NY() = 19
NZ() = 3
IDXYZ = 2
IUNIT = 0

"GRIDBLOCKS DIMENSIONS"
DX() = 32.8
DY() = 32.8
DZ() = 10.   20.  10.

"TOTAL NO. OF COMPONENTS, NO. OF TRACERS, NO. OF GEL COMPONENTS"
N = 13
NO = 0
NPHAS = 3
NTW = 0
NTA = 0
NGC = 5
NG = 0
NOTH = 0
```

"NAME OF SPECIES"
SPNAME(1) = "WATER"
SPNAME(2) = "OIL"
SPNAME(3) = "SURF."
SPNAME(4) = "POLYM."
SPNAME(5) = "ANION"
SPNAME(6) = "CALC."
SPNAME(7) = "NONE"
SPNAME(8) = "NONE"
SPNAME(9) = "Mg"
SPNAME(10) = "CO3"
SPNAME(11) = "NA"
SPNAME(12) = "H+"
SPNAME(13) = "Acid"

"FLAG INDICATING IF THE COMPONENT IS INCLUDED IN CALCULATIONS OR NOT"
ICF() = 1 1 1 1 1 1 0 0 1 1 1 1 1

" OUTPUT OPTIONS "

"FLAG FOR PV OR DAYS TO PRINT OR TO STOP THE RUN"
ICUMTM = 0
ISTOP = 0
IOUTGMS = 2
IS3GRF = 1

"FLAG INDICATING IF THE PROFILE OF KCTH COMPONENT SHOULD BE WRITTEN"
IPRFLG() = 1 1 1 1 1 1 0 0 1 1 1 1 1

"FLAG FOR individual map files"
IPPRES = 1
IPSAT = 1
IPCTOT = 1
IPBIO = 0
IPCAP = 0
IPGEL = 0
IPALK = 1
IPTEMP = 0
IPOBS = 0

"FLAG for individual output map files"
ICKL = 0
IVIS = 0
IPER = 0
ICNM = 0
ICSE = 0
IHYSTP = 0
IFOAMP = 0
INONEQ = 0

"FLAG for variables to PROF output file"
IADS = 0

IVEL = 0
IRKF = 0
IPHSE = 0

" RESERVOIR PROPERTIES "

"MAX. SIMULATION TIME (DAYS)"
TMAX = 551

"ROCK COMPRESSIBILITY (1/PSI), STAND. PRESSURE(PSIA)"
COMPR = 0
PSTAND = 1740.45

"FLAGS INDICATING CONSTANT OR VARIABLE POROSITY, X,Y,AND Z PERMEABILITY"
IPOR1 = 0
IPERMX = 2
IPERMY = 3
IPERMZ = 3
IMOD = 0
ITRANZ = 0
INTG = 0

"POROSITY"
POR1() = 0.30

"X-PERMEABILITY (MILIDARCY)"
PERMX1() = 1648.1 1636.2 1634.9 1653.1 1659.0 1711.9 1817.3 1887.7 1941.6 1990.2
2017.1 2024.7 2054.8 2116.8 2200.1 2286.3 2317.7 2311.8 2283.6
1633.7 1614.1 1606.2 1623.7 1643.8 1704.1 1788.0 1913.3 1987.4 2024.0
2024.9 2017.7 2040.8 2113.5 2227.4 2338.4 2377.4 2357.3 2310.2
1629.7 1605.2 1593.0 1610.9 1664.9 1694.4 1776.4 1873.9 1990.9 2045.0
2011.4 1957.0 2008.3 2078.4 2228.9 2360.0 2410.0 2376.4 2314.0
1640.9 1618.3 1608.9 1628.4 1676.5 1682.3 1739.5 1816.9 1925.0 1988.3
1961.2 1900.5 1878.7 1944.8 2132.4 2286.4 2357.1 2333.3 2278.2
1666.8 1651.5 1648.4 1665.1 1656.9 1676.4 1688.0 1726.0 1786.6 1847.5
1842.8 1801.1 1772.6 1799.9 1901.4 2100.1 2204.1 2224.4 2204.5
1701.2 1692.2 1690.2 1653.6 1659.9 1667.0 1669.8 1666.4 1670.3 1677.8
1713.9 1700.1 1686.2 1669.6 1732.7 1870.0 2041.6 2103.8 2133.3
1702.8 1659.0 1646.0 1663.9 1654.4 1662.6 1673.0 1649.2 1585.6 1542.6
1592.2 1665.5 1651.9 1622.0 1676.7 1803.3 1887.0 2040.2 2075.4
1742.7 1738.9 1673.5 1644.4 1648.2 1644.6 1643.7 1589.7 1457.9 1413.8
1448.7 1600.3 1651.2 1657.2 1708.2 1802.5 1903.4 2022.5 2052.0
1782.7 1784.3 1723.8 1670.8 1620.7 1608.3 1566.3 1459.5 1310.4 1201.2
1305.4 1482.7 1645.5 1711.4 1770.1 1866.9 1954.2 2040.4 2053.4
1818.0 1824.9 1785.2 1717.3 1644.1 1559.9 1480.1 1415.1 1212.0 1085.0
1211.3 1444.8 1600.3 1722.8 1851.2 1941.0 2017.0 2061.2 2061.8
1831.8 1838.0 1810.0 1734.8 1653.0 1582.5 1507.2 1423.6 1280.7 1189.9
1301.2 1500.1 1690.2 1823.3 1893.6 1960.0 2034.4 2067.0 2054.6
1827.7 1826.8 1788.8 1722.1 1648.4 1590.0 1570.1 1519.4 1435.9 1390.1
1495.4 1670.5 1857.0 1956.3 1987.4 1988.8 2019.7 2054.9 2045.3
1818.5 1813.9 1787.9 1704.4 1640.3 1588.7 1577.0 1570.5 1556.5 1541.4
1658.5 1835.1 2030.4 2129.9 2094.4 2021.0 2038.8 2030.0 2020.6

1821.5 1816.5 1796.5 1714.2 1646.1 1586.2 1564.0 1581.7 1596.0 1620.6
1733.5 1919.8 2119.4 2216.0 2147.2 2032.6 2017.0 1994.7 1985.1
1842.2 1849.8 1839.6 1789.0 1680.4 1615.7 1597.4 1608.7 1626.7 1675.4
1759.9 1920.8 2066.4 2132.5 2090.3 1987.0 1954.4 1938.0 1937.7
1869.1 1896.4 1907.9 1855.6 1745.9 1656.3 1647.5 1641.9 1653.8 1691.0
1749.8 1850.5 1942.3 1960.2 1920.5 1845.5 1825.2 1860.6 1882.2
1883.6 1919.5 1943.0 1895.7 1767.8 1671.6 1649.0 1653.4 1666.6 1696.6
1730.8 1791.1 1877.4 1864.6 1770.5 1718.2 1730.9 1771.5 1831.8
1877.9 1904.1 1914.9 1872.4 1772.7 1689.5 1667.2 1676.9 1704.9 1744.1
1772.1 1807.1 1807.5 1763.6 1707.2 1672.0 1699.5 1752.3 1801.4
1859.5 1871.3 1867.8 1833.0 1771.1 1715.2 1702.6 1704.9 1721.3 1748.1
1757.9 1778.9 1775.3 1744.9 1717.9 1694.9 1711.4 1749.6 1790.2
2024.1 2034.6 2034.8 2014.7 1960.2 1898.9 1814.4 1767.9 1740.6 1750.5
1739.7 1763.2 1806.6 1870.5 1944.6 2017.3 2043.8 2041.4 2023.1
2038.6 2057.3 2065.2 2044.8 1980.1 1901.5 1829.8 1755.4 1728.1 1727.0
1726.6 1748.3 1794.3 1870.2 1968.2 2058.6 2088.9 2075.3 2043.3
2043.1 2066.7 2080.0 2057.8 1989.4 1892.0 1819.9 1755.0 1725.5 1717.0
1723.7 1743.9 1774.0 1847.8 1973.8 2075.8 2113.0 2089.4 2047.4
2033.4 2052.4 2060.5 2031.5 1956.8 1861.7 1805.6 1755.7 1731.7 1728.3
1725.8 1733.2 1748.0 1796.7 1905.7 2023.0 2075.6 2060.4 2025.5
2013.7 2018.8 2010.5 1970.2 1895.2 1827.2 1767.3 1753.4 1741.5 1745.1
1725.0 1709.9 1688.9 1709.2 1778.5 1891.1 1970.1 1989.3 1981.6
1997.1 1987.9 1963.5 1911.3 1841.3 1775.5 1743.9 1742.7 1747.1 1719.8
1721.4 1678.4 1629.4 1594.7 1649.1 1767.0 1863.5 1917.4 1954.5
2002.1 1992.5 1960.1 1890.3 1819.2 1752.8 1723.0 1743.8 1759.3 1742.4
1732.6 1695.0 1597.0 1538.0 1600.2 1725.9 1828.1 1901.9 1936.8
2018.3 2005.3 1965.4 1911.0 1835.4 1777.5 1749.1 1767.9 1779.9 1799.1
1772.5 1733.0 1646.9 1600.8 1650.4 1759.8 1860.1 1922.4 1951.5
2049.0 2045.8 2008.4 1949.6 1883.0 1826.9 1799.2 1804.7 1844.6 1861.1
1839.8 1784.1 1739.8 1718.4 1758.7 1845.0 1929.3 1978.7 1989.0
2086.9 2091.7 2065.8 2008.8 1959.7 1883.0 1865.5 1853.8 1875.7 1891.0
1869.1 1828.5 1803.4 1803.4 1855.8 1925.9 1999.9 2034.1 2025.9
2097.1 2105.0 2087.5 2035.8 1977.0 1933.2 1909.3 1872.3 1878.8 1880.4
1877.3 1854.8 1852.5 1908.6 1936.0 1984.1 2035.9 2055.0 2046.8
2094.4 2098.3 2079.3 2035.5 1978.4 1928.8 1904.9 1904.4 1880.0 1875.4
1884.5 1917.9 1929.5 1943.5 1966.3 1997.4 2031.3 2047.9 2046.3
2121.0 2115.3 2083.6 2035.3 1970.3 1896.7 1859.9 1882.4 1915.2 1885.2
1930.5 1953.2 1958.4 1956.2 1972.6 2003.4 2037.2 2048.9 2052.7
2151.7 2150.2 2135.6 2079.0 1984.1 1879.0 1816.0 1868.4 1920.8 1966.8
1986.1 1972.1 1963.3 1951.0 1973.3 2022.8 2066.1 2078.6 2080.5
2195.6 2209.4 2204.0 2155.0 2059.5 1968.2 1913.3 1946.4 1977.4 2019.2
2013.1 2015.5 1998.6 1987.5 2032.2 2111.1 2138.4 2142.0 2131.3
2236.5 2269.4 2283.9 2237.8 2152.0 2114.5 2126.7 2092.0 2069.4 2061.3
2047.4 2056.5 2067.1 2097.7 2182.7 2256.3 2270.4 2231.3 2196.0
2257.2 2295.8 2319.0 2280.2 2197.7 2179.7 2202.0 2163.6 2119.6 2098.0
2082.9 2105.9 2169.3 2259.8 2357.4 2410.2 2381.5 2325.4 2255.7
2254.7 2282.8 2294.8 2264.6 2207.2 2183.5 2183.4 2164.6 2149.0 2138.7
2132.0 2167.6 2230.4 2323.1 2419.8 2457.0 2428.2 2372.7 2320.7
2239.4 2254.4 2256.0 2235.1 2202.0 2180.5 2165.3 2150.4 2156.1 2154.6
2152.9 2192.7 2254.9 2333.2 2406.3 2432.2 2414.6 2373.7 2330.2
1557.9 1580.9 1584.6 1555.3 1489.2 1405.4 1296.3 1239.5 1209.1 1183.9
1140.6 1109.6 1071.3 1037.7 1015.4 1013.6 1009.2 1010.8 1016.6
1582.4 1620.1 1635.5 1603.3 1516.7 1407.6 1314.5 1225.6 1199.6 1189.7
1164.2 1121.1 1069.5 1028.7 1006.3 1003.9 1000.3 1003.3 1011.4

1588.0 1636.3 1661.0 1624.1 1517.6 1390.3 1296.1 1217.3 1196.6 1195.0
1176.5 1126.3 1055.4 1013.0 1005.9 999.4 997.0 1000.4 1010.2
1563.3 1607.2 1625.6 1578.6 1462.8 1339.9 1265.1 1205.6 1192.2 1185.4
1154.5 1098.6 1037.9 993.3 989.9 996.3 998.9 1003.7 1015.3
1509.3 1533.0 1529.2 1470.1 1355.8 1260.4 1193.1 1180.9 1177.3 1160.4
1109.2 1048.4 980.9 963.2 966.6 979.7 999.3 1013.1 1029.2
1442.9 1445.2 1423.1 1355.2 1265.5 1182.3 1142.3 1149.0 1160.6 1123.0
1076.5 1000.2 925.4 881.6 908.7 969.2 1000.3 1032.0 1043.3
1356.6 1380.4 1355.5 1295.0 1205.8 1143.4 1116.0 1133.0 1147.8 1131.3
1079.7 1002.5 890.9 834.0 880.7 967.1 1028.2 1059.2 1087.6
1313.4 1303.5 1282.3 1247.0 1198.7 1156.5 1133.7 1147.7 1152.9 1156.1
1116.5 1044.7 941.0 890.4 926.1 1003.3 1086.3 1130.7 1154.1
1296.4 1290.6 1273.9 1242.1 1203.6 1176.3 1157.4 1160.4 1203.6 1232.2
1194.4 1108.7 1032.8 997.5 1013.5 1092.8 1180.7 1229.4 1234.0
1298.0 1299.8 1285.1 1248.5 1208.2 1147.7 1152.7 1168.2 1234.3 1278.0
1232.7 1148.6 1082.0 1069.3 1096.2 1177.2 1273.9 1319.5 1301.8
1295.7 1308.0 1285.9 1227.3 1167.7 1131.0 1127.3 1135.5 1197.5 1232.1
1202.7 1139.7 1096.0 1114.1 1147.1 1225.7 1318.3 1355.0 1333.5
1284.2 1290.0 1264.8 1203.6 1131.1 1071.1 1048.2 1079.9 1117.5 1144.5
1143.0 1121.3 1087.7 1083.4 1125.1 1201.5 1288.2 1327.5 1316.0
1265.0 1259.5 1224.5 1169.9 1085.8 990.2 941.5 987.6 1065.1 1078.4
1110.1 1090.2 1056.9 1037.3 1073.4 1151.0 1226.3 1270.5 1277.3
1250.5 1241.4 1215.0 1164.3 1073.2 950.4 874.0 943.9 1026.4 1075.7
1092.0 1076.4 1037.1 1010.0 1046.2 1125.5 1188.4 1229.4 1245.4
1255.0 1255.8 1243.2 1196.9 1120.2 1029.1 973.6 1012.4 1053.4 1096.7
1094.2 1096.6 1065.7 1048.1 1089.9 1166.2 1198.9 1224.5 1235.4
1269.8 1286.2 1293.0 1256.7 1186.0 1160.4 1169.1 1139.8 1124.0 1122.5
1111.5 1110.2 1121.3 1135.6 1193.4 1245.3 1265.6 1249.6 1245.1
1278.9 1301.8 1317.0 1288.2 1224.3 1211.8 1232.0 1201.1 1164.8 1148.2
1133.4 1142.6 1157.0 1233.4 1290.8 1321.7 1313.8 1296.1 1262.5
1275.7 1292.6 1299.9 1276.9 1232.9 1215.1 1213.7 1196.2 1173.9 1157.2
1140.8 1159.5 1198.3 1258.2 1326.4 1346.0 1332.6 1309.7 1291.8
1264.6 1272.8 1272.3 1255.6 1229.7 1212.8 1195.0 1182.0 1176.9 1168.3
1155.5 1177.1 1215.3 1265.3 1320.5 1333.9 1326.4 1309.8 1294.3

PERMY1() = 0
PERMZ1() = 0

"Constant permeability multiplier for Y direction permeability"
FACTY = 1

"Constant permeability multiplier for Z direction permeability"
FACTZ = 0.1

"FLAG FOR CONSTANT OR VARIABLE DEPTH, PRESSURE, WATER SATURATION"
IDEPTH = 0
IPRESS = 2
ISWI = 2
ICWI = -1

"DEPTH (FT)"
EL1() = 4150.0

"PRESSURE (PSIA)"
P1(,,,1) = 1780

"INITIAL WATER SATURATION"
S1(,,,1) = 0.3295 0.3271 0.3272 0.3318 0.3403 0.3492 0.3570 0.3625 0.3675 0.3746
0.3838 0.3968 0.4117 0.4242 0.4097 0.3926 0.3798 0.3712 0.3649
0.3316 0.3280 0.3242 0.3321 0.3442 0.3560 0.3649 0.3690 0.3704 0.3743
0.3838 0.3950 0.4038 0.4083 0.4013 0.3895 0.3781 0.3719 0.3701
0.3343 0.3272 0.4804 0.4974 0.5068 0.5125 0.5158 0.5150 0.5138 0.5110
0.5160 0.5194 0.5302 0.5347 0.5218 0.5171 0.5124 0.5124 0.3757
0.3386 0.3333 0.4957 0.5048 0.5117 0.5167 0.5202 0.5194 0.5195 0.5194
0.5229 0.5340 0.5436 0.5485 0.5360 0.5196 0.5173 0.5160 0.3882
0.3433 0.3413 0.5041 0.5106 0.5167 0.5221 0.5633 0.5748 0.5622 0.5544
0.5604 0.5771 0.5923 0.5936 0.5725 0.5396 0.5199 0.5193 0.4066
0.3474 0.3482 0.5094 0.5149 0.5213 0.5946 0.6112 0.6089 0.6041 0.6013
0.6017 0.6051 0.6104 0.6126 0.5986 0.5682 0.5370 0.5366 0.4221
0.3501 0.3524 0.5123 0.5176 0.5463 0.6089 0.6858 0.5088 0.4601 0.4508
0.4514 0.4602 0.4949 0.7536 0.7075 0.6873 0.6392 0.6348 0.4117
0.3501 0.6095 0.6166 0.6239 0.6426 0.7064 0.7285 0.4884 0.4681 0.4559
0.4563 0.4659 0.4816 0.7218 0.7013 0.6682 0.6317 0.6263 0.4026
0.3501 0.6097 0.6171 0.6239 0.6310 0.6990 0.7169 0.4715 0.4657 0.4577
0.4573 0.4613 0.4653 0.7111 0.6941 0.6466 0.6221 0.6170 0.3943
0.3491 0.6078 0.6168 0.6238 0.6301 0.6805 0.7116 0.4602 0.4607 0.4577
0.4568 0.4570 0.4567 0.7055 0.6832 0.6328 0.6170 0.6151 0.3858
0.3484 0.6043 0.6167 0.6243 0.6306 0.6857 0.7123 0.4574 0.4587 0.4570
0.4569 0.4573 0.4566 0.7066 0.6889 0.6340 0.6165 0.6120 0.3799
0.3510 0.6090 0.6182 0.6255 0.6421 0.7029 0.7187 0.4660 0.4617 0.4572
0.4578 0.4628 0.4676 0.7139 0.6971 0.6536 0.6181 0.6147 0.3813
0.3530 0.6116 0.6191 0.6268 0.6739 0.7110 0.7307 0.4833 0.4659 0.4570
0.4577 0.4674 0.4885 0.7302 0.7061 0.6868 0.6277 0.6157 0.3841
0.3533 0.6114 0.6185 0.6267 0.6860 0.7173 0.7881 0.6328 0.6176 0.6125
0.6136 0.4584 0.5193 0.6971 0.7189 0.6858 0.6412 0.3949 0.3843
0.3528 0.3544 0.6120 0.6193 0.6496 0.6993 0.6159 0.6107 0.6027 0.5986
0.6011 0.4443 0.4567 0.4761 0.7074 0.6785 0.6352 0.3906 0.3815
0.3493 0.3492 0.6059 0.6144 0.6222 0.6679 0.5950 0.5939 0.5745 0.5554
0.5736 0.4305 0.4393 0.4405 0.6863 0.6404 0.6309 0.3818 0.3756
0.3449 0.3419 0.5952 0.6080 0.6166 0.6226 0.5373 0.5371 0.5229 0.5302
0.5487 0.4211 0.4261 0.4232 0.6434 0.6302 0.6245 0.3727 0.3681
0.3422 0.3425 0.3440 0.3568 0.3694 0.3804 0.3884 0.3913 0.3937 0.3986
0.4062 0.4164 0.4213 0.4115 0.6265 0.6195 0.6170 0.3642 0.3601
0.3399 0.3412 0.3457 0.3554 0.3655 0.3744 0.3809 0.3850 0.3901 0.3970
0.4069 0.4196 0.4299 0.4156 0.3916 0.3753 0.3646 0.3592 0.3522
0.3299 0.3280 0.3281 0.3336 0.3426 0.3515 0.3593 0.3650 0.3700 0.3770
0.3864 0.3990 0.4117 0.4214 0.4091 0.3946 0.3823 0.3743 0.3690
0.3314 0.3284 0.3247 0.3328 0.3451 0.3570 0.3663 0.3707 0.3724 0.3765
0.3861 0.3972 0.4053 0.4090 0.4025 0.3916 0.3804 0.3746 0.3732
0.3339 0.3271 0.4808 0.4972 0.5067 0.5127 0.5161 0.5154 0.5145 0.5120
0.5167 0.5206 0.5348 0.5384 0.5258 0.5177 0.5136 0.5135 0.3782
0.3388 0.3336 0.4956 0.5046 0.5116 0.5167 0.5203 0.5196 0.5201 0.5202
0.5262 0.5373 0.5458 0.5497 0.5377 0.5202 0.5179 0.5167 0.3901
0.3440 0.3421 0.5041 0.5105 0.5165 0.5219 0.5600 0.5759 0.5660 0.5589
0.5648 0.5798 0.5921 0.5931 0.5717 0.5408 0.5240 0.5207 0.4060
0.3485 0.3493 0.5095 0.5149 0.5210 0.5932 0.6099 0.6085 0.6044 0.6019
0.6022 0.6053 0.6104 0.6129 0.5982 0.5679 0.5394 0.5410 0.4185

0.3515 0.3537 0.5125 0.5176 0.5423 0.6075 0.6745 0.5059 0.4701 0.4601
0.4609 0.4711 0.5043 0.7724 0.7074 0.6868 0.6419 0.6397 0.4113
0.3516 0.6106 0.6171 0.6241 0.6416 0.7059 0.7276 0.4861 0.4752 0.4683
0.4686 0.4744 0.4837 0.7226 0.7008 0.6684 0.6350 0.6318 0.4040
0.3515 0.6109 0.6176 0.6242 0.6317 0.6997 0.7171 0.4769 0.4744 0.4708
0.4704 0.4724 0.4744 0.7111 0.6938 0.6487 0.6262 0.6193 0.3969
0.3505 0.6091 0.6173 0.6242 0.6304 0.6848 0.7124 0.4719 0.4724 0.4712
0.4704 0.4703 0.4690 0.7059 0.6871 0.6369 0.6184 0.6158 0.3893
0.3496 0.6058 0.6172 0.6246 0.6310 0.6902 0.7131 0.4705 0.4718 0.4709
0.4705 0.4704 0.4688 0.7070 0.6904 0.6385 0.6170 0.6133 0.3833
0.3524 0.6102 0.6186 0.6258 0.6455 0.7031 0.7189 0.4744 0.4730 0.4708
0.4709 0.4726 0.4740 0.7134 0.6968 0.6555 0.6208 0.6154 0.3843
0.3547 0.6126 0.6197 0.6271 0.6744 0.7106 0.7299 0.4830 0.4747 0.4700
0.4700 0.4737 0.4831 0.7268 0.7048 0.6855 0.6310 0.6163 0.3875
0.3550 0.6124 0.6192 0.6271 0.6856 0.7164 0.7798 0.6309 0.6174 0.6128
0.6140 0.4671 0.5024 0.6193 0.7159 0.6840 0.6429 0.3964 0.3877
0.3545 0.3564 0.6129 0.6201 0.6529 0.6989 0.6145 0.6095 0.6023 0.5989
0.6015 0.4435 0.4544 0.4646 0.7035 0.6772 0.6354 0.3920 0.3840
0.3510 0.3512 0.6072 0.6154 0.6229 0.6720 0.5946 0.5933 0.5765 0.5610
0.5809 0.4304 0.4382 0.4377 0.6836 0.6390 0.6312 0.3830 0.3784
0.3465 0.3439 0.5973 0.6093 0.6177 0.6233 0.5409 0.5394 0.5258 0.5355
0.5548 0.4216 0.4261 0.4221 0.6399 0.6302 0.6250 0.3745 0.3713
0.3437 0.3443 0.3464 0.3593 0.3720 0.3825 0.3908 0.3936 0.3961 0.4008
0.4081 0.4176 0.4211 0.4107 0.6268 0.6201 0.6179 0.3666 0.3642
0.3414 0.3431 0.3481 0.3582 0.3685 0.3774 0.3838 0.3889 0.3939 0.4004
0.4091 0.4192 0.4257 0.4135 0.3923 0.3772 0.3675 0.3631 0.3583
0.3302 0.3283 0.3286 0.3338 0.3422 0.3510 0.3589 0.3647 0.3699 0.3769
0.3867 0.4011 0.4171 0.4303 0.4158 0.3979 0.3831 0.3739 0.3665
0.3323 0.3290 0.3251 0.3332 0.3453 0.3570 0.3664 0.3711 0.3732 0.3773
0.3871 0.3991 0.4084 0.4136 0.4065 0.3950 0.3826 0.3760 0.3730
0.3351 0.3279 0.4802 0.4968 0.5064 0.5126 0.5162 0.5155 0.5145 0.5121
0.5167 0.5201 0.5331 0.5372 0.5279 0.5183 0.5144 0.5141 0.3797
0.3400 0.3345 0.4953 0.5044 0.5113 0.5165 0.5200 0.5193 0.5195 0.5194
0.5221 0.5316 0.5382 0.5422 0.5334 0.5217 0.5184 0.5173 0.3944
0.3450 0.3430 0.5039 0.5103 0.5161 0.5212 0.5465 0.5591 0.5516 0.5459
0.5510 0.5619 0.5765 0.5854 0.5574 0.5362 0.5247 0.5263 0.4138
0.3495 0.3503 0.5094 0.5146 0.5203 0.5722 0.6068 0.6064 0.6027 0.6001
0.6005 0.6037 0.6086 0.6109 0.5968 0.5563 0.5367 0.5442 0.4297
0.3524 0.3542 0.5126 0.5174 0.5309 0.6045 0.6535 0.4908 0.4619 0.4561
0.4567 0.4636 0.4994 0.7611 0.7061 0.6701 0.6375 0.6386 0.4184
0.3527 0.6106 0.6172 0.6239 0.6329 0.7039 0.7257 0.4794 0.4661 0.4620
0.4622 0.4661 0.4766 0.7216 0.6989 0.6548 0.6322 0.6310 0.4077
0.3526 0.6108 0.6176 0.6240 0.6305 0.6875 0.7157 0.4675 0.4661 0.4641
0.4638 0.4650 0.4664 0.7097 0.6865 0.6394 0.6254 0.6213 0.3988
0.3514 0.6091 0.6172 0.6238 0.6298 0.6710 0.7112 0.4647 0.4652 0.4647
0.4641 0.4639 0.4630 0.7049 0.6652 0.6322 0.6197 0.6163 0.3905
0.3506 0.6059 0.6170 0.6241 0.6303 0.6761 0.7119 0.4637 0.4648 0.4644
0.4640 0.4638 0.4627 0.7057 0.6702 0.6326 0.6175 0.6142 0.3841
0.3533 0.6101 0.6185 0.6253 0.6342 0.7008 0.7181 0.4661 0.4655 0.4642
0.4640 0.4649 0.4658 0.7120 0.6934 0.6427 0.6202 0.6159 0.3846
0.3558 0.6126 0.6197 0.6268 0.6644 0.7094 0.7296 0.4768 0.4667 0.4635
0.4633 0.4654 0.4762 0.7254 0.7026 0.6627 0.6277 0.6165 0.3871
0.3564 0.6125 0.6196 0.6271 0.6787 0.7159 0.7866 0.6319 0.6178 0.6123
0.6135 0.4604 0.5001 0.6263 0.7137 0.6797 0.6400 0.3967 0.3868

```
 0.3559  0.3581  0.6135  0.6204  0.6487  0.6988  0.6150  0.6101  0.6021  0.5974
 0.5998  0.4437  0.4542  0.4628  0.6995  0.6639  0.6352  0.3922  0.3832
 0.3525  0.3533  0.6080  0.6161  0.6231  0.6632  0.5949  0.5933  0.5683  0.5518
 0.5650  0.4311  0.4383  0.4368  0.6801  0.6366  0.6314  0.3832  0.3774
 0.3478  0.3459  0.5982  0.6101  0.6182  0.6236  0.5436  0.5401  0.5270  0.5338
 0.5497  0.4229  0.4271  0.4225  0.6373  0.6306  0.6257  0.3750  0.3702
 0.3444  0.3454  0.3479  0.3606  0.3730  0.3834  0.3919  0.3945  0.3973  0.4024
 0.4103  0.4207  0.4241  0.4136  0.6275  0.6212  0.6187  0.3670  0.3624
 0.3412  0.3430  0.3481  0.3580  0.3681  0.3769  0.3833  0.3882  0.3934  0.4008
 0.4116  0.4241  0.4350  0.4193  0.3951  0.3783  0.3680  0.3622  0.3543

"Salinity and divalent cation concentration of brine"
C50 = 0.0583
C60 = 0.0025


"            PHYSICAL PROPERTY DATA                  "


"OIL CONC. AT PLAIT POINT FOR TYPE II(+)AND TYPE II(-), CMC"
C2PLC = 0.0
C2PRC = 1.
EPSME = 0.0001
IHAND = 0

"FLAG INDICATING TYPE OF PHASE BEHAVIOR PARAMETERS"
IFGHBN = 0

"SLOPE AND INTERCEPT OF BINODAL CURVE AT ZERO, OPT., AND 2XOPT SALINITY FOR ALCOHOL 1"
HBNS70 0.0
HBNC70 .1
HBNS71 .0
HBNC71 .026
HBNS72 .0
HBNC72 .028

"SLOPE AND INTERCEPT OF BINODAL CURVE AT ZERO, OPT., AND 2XOPT SALINITY FOR ALCOHOL 2"
HBNS80 0.
HBNC80 0.
HBNS81 0.
HBNC81 0.
HBNS82 0.
HBNC82 0.

"LOWER AND UPPER EFFECTIVE SALINITY FOR ALCOHOL 1 AND ALCOHOL 2"
CSEL7  0.2
CSEU7  0.5
CSEL8  0.
CSEU8  0.

"THE CSE SLOPE PARAMETER FOR CALCIUM AND ALCOHOL 1 AND ALCOHOL 2"
BETA6  0.8
BETA7  -2.0
BETA8  0
```

"FLAG FOR ALCOHOL PART. MODEL AND PARTITION COEFFICIENTS"
IALC 1
OPSK7O 0
OPSK7S 0
OPSK8O 0
OPSK8S 0

"NO. OF ITERATIONS, AND TOLERANCE"
NALMAX = 20
EPSALC = .0001

"ALCOHOL 1 PARTITIONING PARAMETERS IF IALC=1"
AKWC7 4.671
AKWS7 1.79
AKM7 48.
AK7 35.31
PT7 .222

"ALCOHOL 2 PARTITIONING PARAMETERS IF IALC=1"
AKWC8 0
AKWS8 0
AKM8 0.
AK8 0
PT8 0

"IFT MODEL FLAG"
IFT = 0

"INTERFACIAL TENSION PARAMETERS"
G11 13.
G12 -14.8
G13 .007
G21 13.
G22 -14.95
G23 .010

"LOG10 OF OIL/WATER INTERFACIAL TENSION"
XIFTW 1.3

"MASS TRANSFER FLAG"
IMASS = 0
ICOR = 0

"WETTABILITY ALTERATION FLAGS"
IWALT = 0
IWALF = 0

"CAPILLARY DESATURATION PARAMETERS FOR PHASE 1, 2, AND 3"
ITRAP = 1
T11() = 1865.
T22() = 59074
T33() = 364.2

"RELATIVE PERM. FLAG (0:IMBIBITION COREY,1:FIRST DRAINAGE COREY)"
IPERM = 0
IRTYPE = 0

"FLAG FOR CONSTANT OR VARIABLE REL. PERM. PARAMETERS"
ISRW = 0
IPRW = 0
IEW = 0

"RES. SATURATION OF PHASES 1,2,AND 3 AT LOW CAPILLARY NO."
S1RW1() = .25
S2RW1() = 0.15
S3RW1() = 0.2

"ENDPOINT REL. PERM. OF PHASES 1,2,AND 3 AT LOW CAPILLARY NO."
P1RW1() = 0.2
P2RW1() = 0.95
P3RW1() = 0.2

"REL. PERM. EXPONENT OF PHASES 1,2,AND 3 AT LOW CAPILLARY NO."
E1W1() = 3
E2W1() = 2
E3W1() = 2

"RES. SATURATION OF PHASES 1,2,AND 3 AT HIGH CAPILLARY NO."
S1RC = 0.0
S2RC = 0.0
S3RC = 0.0

"ENDPOINT REL. PERM. OF PHASES 1,2,AND 3 AT HIGH CAPILLARY NO."
P1RC = 1.0
P2RC = 1.0
P3RC = 1.0

"REL. PERM. EXPONENT OF PHASES 1,2,AND 3 AT HIGH CAPILLARY NO."
E13C = 3
E23C = 2
E31C = 2

"WATER AND OIL VISCOSITY, RESERVOIR TEMPERATURE"
VIS1 = 0.46
VIS2 = 40
TSTAND = 0

"VISCOSITY PARAMETERS"
ALPHAV() = 0.5 0.5 0. .9 .7

"PARAMETERS TO CALCULATE POLYMER VISCOSITY AT ZERO SHEAR RATE"
AP1 38
AP2 1600.
AP3 0

"PARAMETER TO COMPUTE CSEP,MIN. CSEP, AND SLOPE OF LOG VIS. VS. LOG CSEP "
BETAP 2

CSE1 0.01
SSLOPE 0.175

"PARAMETER FOR SHEAR RATE DEPENDENCE OF POLYMER VISCOSITY"
GAMMAC 4
GAMHF 20
POWN 1.1
IPMOD = 0
ISHEAR = 0
RWEFF = 0.25
GAMHF2 = 0.5

"FLAG FOR POLYMER PARTITIONING, PERM. REDUCTION PARAMETERS"
IPOLYM 1
EPHI3 1
EPHI4 0.8
BRK 1000
CRK 0.0186
RKCUT 10

"SPECIFIC WEIGHT FOR COMPONENTS 1,2,3,7,AND 8 , AND GRAVITY FLAG"
DEN1 .433
DEN2 0.368
DEN23 0.368
DEN3 0.42
DEN7 0.346
DEN8 0
IDEN 2

"FLAG FOR CHOICE OF UNITS (0:BOTTOMHOLE CONDITION , 1: STOCK TANK)"
ISTB = 0

"COMPRESSIBILITY FOR VOL. OCCUPYING COMPONENTS"
COMPC() = 0

"CONSTANT OR VARIABLE PC PARAM., WATER-WET OR OIL-WET PC CURVE FLAG"
ICPC = 0
IEPC = 0
IOW = 0

"CAPILLARY PRESSURE PARAMETERS"
CPC1() = 9
EPC1() = 2.

"MOLECULAR DIFFUSION COEF. KCTH COMPONENT IN PHASE J"
D() = 0.0

"LONGITUDINAL AND TRANSVERSE DISPERSIVITY OF PHASE J"
ALPHAL() = 12
ALPHAT() = 0.4

"FLAG TO SPECIFY ORGANIC ADSORPTION CALCULATION"
IADSO = 0

"SURFACTANT AND POLYMER ADSORPTION PARAMETERS"
AD31 1
AD32 0.5
B3D 1000
AD41 0.7
AD42 0.
B4D 100.
IADK 0
IADS1 0
FADS 0
REFK 100

" Geochemical Properties "

IRSPS 2
IPHAD 1
EQW 419

PHC 7.0
PHT 13.0
PHT1 13.0
HPHAD 0

CSELP .2
CSEUP .4

IMIX 0

NELET 7
NFLD 18
NSLD 4
NSORB 4
NACAT 3
ICHRGE 1

NIAQ 7
NEX 1
NSLEL 4
NSURF1 12

NH 5
NNA 4
NCA 1
NMG 2
NCARB 3

NALU 0
NSILI 0
NOXY 0

NACD 6

```
ELEMNT(1) = "CALCIUM"
ELEMNT(2) = "MAGNESIUM"
ELEMNT(3) = "CARBON (AS CAR)"
ELEMNT(4) = "SODIUM"
ELEMNT(5) = "HYDROGEN (REA)"
ELEMNT(6) = "ACID (PETROLEUM)"
ELEMNT(7) = "CHLORINE"

ELCRG() = 2 2 -2 1 1 -1 -1

FLDSPS(1)  = "HYDROGEN ION "
FLDSPS(2)  = "SODIUM ION"
FLDSPS(3)  = "CALCIUM ION"
FLDSPS(4)  = "MAGENSIUM ION"
FLDSPS(5)  = "CARBONATE ION"
FLDSPS(6)  = "PETROLEUM ACID IN OIL"
FLDSPS(7)  = "WATER"
FLDSPS(8)  = "CALCIUM MONOHYDROXIDE ION"
FLDSPS(9)  = "MAGNESIUM MONOHYROXIDE ION"
FLDSPS(10) = "CA (HC03) +"
FLDSPS(11) = "MG (HCO3) +"
FLDSPS(12) = "PETRLEUM ACID ANION"
FLDSPS(13) = "HYDROXIDE ION"
FLDSPS(14) = "BICARBONATE ION"
FLDSPS(15) = "DISSOLVED CARBON MONOHYDROXIDE"
FLDSPS(16) = "AQUEOUS CALCIUM CARBONATE"
FLDSPS(17) = "AQUEOUS MAGNESIUM CARBONATE"
FLDSPS(18) = "PETROLEUM ACID IN WATER"

SLDSPS(1) = "CALCIUM CARBONATE(SOLID)"
SLDSPS(2) = "MAGNESIUM CARBONATE (SOLID)"
SLDSPS(3) = "CALCIUM HYDROXIDE (SOLID)"
SLDSPS(4) = "MAGNESIUM HYDROXIDE(SOLID)   (*"

SORBSP(1) = "SORBED HYDROGEN ION"
SORBSP(2) = "SORBED SODIUM ION"
SORBSP(3) = "SORBED CALCIUM ION"
SORBSP(4) = "SORBED MAGNESIUM ION        (*"

ACATSP(1) = "SURF. ASSOCIATED SODIUM ION"
ACATSP(2) = "SURF. ASSOCIATED CALCIUM ION"
ACATSP(3) = "SURF. ASSOCIATED MAGNESIUM ION"

NSORBX() = 4

AR(1,) = 0.  0.  1.  0.  0.  0.  0.  1.  0.  1.  0.  0.  0.  0.  0.  1.  0.  0
AR(2,) = 0.  0.  0.  1.  0.  0.  0.  0.  1.  0.  1.  0.  0.  0.  0.  0.  1.  0.
AR(3,) = 0.  0.  0.  0.  1.  0.  0.  0.  0.  1.  1.  0.  0.  1.  1.  1.  1.  0.
AR(4,) = 0.  1.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.
AR(5,) = 1.  0.  0.  0.  0.  1.  2.  1.  1.  1.  1.  0.  1.  1.  2.  0.  0.  1.
AR(6,) = 0.  0.  0.  0.  0.  1.  0.  0.  0.  0.  0.  1.  0.  0.  0.  0.  0.  1.

BR(1,) = 1.  0.  1.  0.
BR(2,) = 0.  1.  0.  1.
```

BR(3,) = 1. 1. 0. 0.
BR(4,) = 0. 0. 0. 0.
BR(5,) = 0. 0. 2. 2.
BR(6,) = 0. 0. 0. 0.

DR(1,) = 0. 0. 1. 0.
DR(2,) = 0. 0. 0. 1.
DR(3,) = 0. 0. 0. 0.
DR(4,) = 0. 1. 0. 0.
DR(5,) = 1. 0. 0. 0.
DR(6,) = 0. 0. 0. 0.

ER(1,) = 0. 1. 0.
ER(2,) = 0. 0. 1.
ER(3,) = 0. 0. 0.
ER(4,) = 1. 0. 0.
ER(5,) = 0. 0. 0.
ER(6,) = 0. 0. 0.

BB(1,) = 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(2,) = 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(3,) = 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(4,) = 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(5,) = 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(6,) = 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(7,) = 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(8,) = -1.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(9,) = -1.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(10,) = 1.0 0.0 1.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(11,) = 1.0 0.0 0.0 1.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(12,) = -1.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(13,) = -1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(14,) = 1.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(15,) = 2.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(16,) = 0.0 0.0 1.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(17,) = 0.0 0.0 0.0 1.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(18,) = 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(19,) = 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0
BB(20,) = 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0
BB(21,) = 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0
BB(22,) = 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0
BB(23,) = 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0
BB(24,) = 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0
BB(25,) = 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0

EXSLD(1,) = 0.0 0.0 1.0 0.0 1.0 0.0 0.0
EXSLD(2,) = 0.0 0.0 0.0 1.0 1.0 0.0 0.0
EXSLD(3,) = -2.0 0.0 1.0 0.0 0.0 0.0 0.0
EXSLD(4,) = -2.0 0.0 0.0 1.0 0.0 0.0 0.0

CHARGE() = 1.0 1.0 2.0 2.0 -2.0 0.0 0.0 1.0 1.0 1.0 1.0 -1.0 -1.0 -1.0 0.0 0.0 0.0 0.0

SCHARG(1,) = 1.0 1.0 2.0 2.0

EQK() = 0.1000000000000E+01 0.1000000000000E+01 0.1000000000000E+01
0.1000000000000E+01 0.1000000000000E+01 0.1000000000000E+01
0.1000000000000E+01 0.1205000000000E-12 0.3887100000000E-11
0.1412500000000E+12 0.5834500000000E+12 0.8547970968090E-14
0.1009300000000E-13 0.2138000000000E+11 0.3981100000000E+17
0.1584900000000E+04 0.4786300000000E+04 0.8547970968090E-04

EXK(1,) = 0.2622713901836E+03 0.1509475626956E+03 0.1460000000000E+08

EXEX(1,1,) = 0.0 2.0 -1.0 0.0 0.0 0.0 0.0 0.0 -2.0 1.0 0.0 0.0 0.0 0.0
EXEX(1,2,) = 0.0 2.0 0.0 -1.0 0.0 0.0 0.0 0.0 -2.0 0.0 1.0 0.0 0.0 0.0
EXEX(1,3,) = -1.0 1.0 0.0 0.0 0.0 0.0 0.0 1.0 -1.0 0.0 0.0 0.0 0.0 0.0

REDUC(1,) = -1.0 -1.0 0.0

EXCAI() = 0.4345986038364E-01

SPK() = 0.4748510000000E-08 0.6800000000000E-05 0.4731500000000E+23
0.5604500000000E+17

CHACAT() = 1.0 2.0 2.0

ACATK() = 0.4000000000000E+00 0.3400000000000E+00

EXACAT(1,) = 0.0 -2.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 2.0
-1.0 0.0
EXACAT(2,) = 0.0 -2.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 2.0
 0.0 -1.0

CI() = 0.4572988437942E-04 0.4599039207863E-06 0.9356875200263E-06

C5I = 0.8600000000000E-02
C6I = 0.2047362627532E-04

CELAQI() = 0.1073741607328E-03
0.5216798272134E-02 0.1445391370711E-01 0.1110242729979E+03 0.5287219302195E-04

CAC2I = 0.5090243963269E-01

CAQI() = 0.8967498053015E-11 0.1440818382273E-01 0.2215214219920E-05
0.4506915915486E-05 0.2143589526151E-02 0.4402071180698E-01
0.5548935469620E+02

CSLDI() = 0.1354334681257E+00 0.0000000000000E+00 0.0000000000000E+00
0.4214983717253E-02

CSORBI() = 0.2842693807677E-03 0.3128348856822E-01 0.2738914795667E-02
0.3207136421656E-02

C1I = 0.9999735646023E+00
C2I = 0.9751804675595E+00

ACIDIS = 0.1000000000000E-09
EQWPS = 0.5000000000000E+03

```
"              WELL DATA            "


"FLAG FOR RIGHT AND LEFT BOUNDARY"
IBOUND = 0
IZONE = 0

"TOTAL NUMBER OF WELLS, WELL RADIUS FLAG, FLAG FOR TIME OR COURANT NO."
NWELL = 13
IRO = 2
ITSTEP = 1
NWREL = 13

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(1) = 1
IW(1) = 17
JW(1) = 3
IFLAG(1) = 4
RW(1) = 0.49
SWELL(1) = 0.
IDIR(1) = 3
IFIRST(1) = 1
ILAST(1) = 3
IPRF(1) = 0
WELNAM(1) = "A1"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"PROD. RATE FOR EACH PHASE (L=1,3)"
ICHEK(1) = 0
PWFMIN(1) = 0.0
PWFMAX(1) = 3700
QTMIN(1) = 0.0
QTMAX(1) = 7100
QI(1,1) = -679.19

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(2) = 2
IW(2) = 10
JW(2) = 3
IFLAG(2) = 4
RW(2) = 0.49
SWELL(2) = 0.
IDIR(2) = 3
IFIRST(2) = 1
ILAST(2) = 3
IPRF(2) = 0
WELNAM(2) = "A2"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"PROD. RATE FOR EACH PHASE (L=1,3)"
ICHEK(2) = 0
PWFMIN(2) = 0.0
```

PWFMAX(2) = 3700
QTMIN(2) = 0.0
QTMAX(2) = 7100
QI(2,1) = -803.88

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(3) = 3
IW(3) = 14
JW(3) = 7
IFLAG(3) = 1
RW(3) = 0.49
SWELL(3) = 0.
IDIR(3) = 3
IFIRST(3) = 1
ILAST(3) = 3
IPRF(3) = 0
WELNAM(3) = "A3"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"INJ. RATE AND INJ. COMP. FOR RATE CONS. WELLS FOR EACH PHASE (L=1,3)"
ICHEK(3) = 0
PWFMIN(3) = 0.0
PWFMAX(3) = 3700
QTMIN(3) = 0.0
QTMAX(3) = 7100
QI(3,1) = 2035.89
C_W(3,,1) = 1. 0. 0.000 .0974 .015667 .0019 0. 0. .004774 .009122 0.01461 111.003360 0.0

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(4) = 4
IW(4) = 18
JW(4) = 11
IFLAG(4) = 4
RW(4) = 0.49
SWELL(4) = 0.
IDIR(4) = 3
IFIRST(4) = 1
ILAST(4) = 3
IPRF(4) = 0
WELNAM(4) = "A4"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"PROD. RATE FOR EACH PHASE (L=1,3)"
ICHEK(4) = 0
PWFMIN(4) = 0.0
PWFMAX(4) = 3700
QTMIN(4) = 0.0
QTMAX(4) = 7100
QI(4,1) = -928.32

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(5) = 5
IW(5) = 3
JW(5) = 3

IFLAG(5) = 4
RW(5) = 0.49
SWELL(5) = 0.
IDIR(5) = 3
IFIRST(5) = 1
ILAST(5) = 3
IPRF(5) = 0
WELNAM(5) = "A5"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"PROD. RATE FOR EACH PHASE (L=1,3)"
ICHEK(5) = 0
PWFMIN(5) = 0.0
PWFMAX(5) = 3700
QTMIN(5) = 0.0
QTMAX(5) = 7100
QI(5,1) = -850.24

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(6) = 6
IW(6) = 7
JW(6) = 7
IFLAG(6) = 1
RW(6) = 0.49
SWELL(6) = 0.
IDIR(6) = 3
IFIRST(6) = 1
ILAST(6) = 3
IPRF(6) = 0
WELNAM(6) = "A6"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"INJ. RATE AND INJ. COMP. FOR RATE CONS. WELLS FOR EACH PHASE (L=1,3)"
ICHEK(6) = 0
PWFMIN(6) = 0.0
PWFMAX(6) = 3700.
QTMIN(6) = 0.0
QTMAX(6) = 7100
QI(6,1) = 2197.99
C_W(6,,1) = 1. 0. 0.000 .0974 .015667 .0019 0. 0. .004774 .009122 0.01461 111.003360 0.0

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(7) = 7
IW(7) = 10
JW(7) = 10
IFLAG(7) = 4
RW(7) = 0.49
SWELL(7) = 0.
IDIR(7) = 3
IFIRST(7) = 1
ILAST(7) = 3
IPRF(7) = 0
WELNAM(7) = "A7"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"PROD. RATE FOR EACH PHASE (L=1,3)"
ICHEK(7) = 0
PWFMIN(7) = 0.0
PWFMAX(7) = 3700.
QTMIN(7) = 0.0
QTMAX(7) = 7100
QI(7,1) = -2088.94

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(8) = 8
IW(8) = 14
JW(8) = 14
IFLAG(8) = 1
RW(8) = 0.49
SWELL(8) = 0.
IDIR(8) = 3
IFIRST(8) = 1
ILAST(8) = 3
IPRF(8) = 0
WELNAM(8) = "A8"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"INJ. RATE AND INJ. COMP. FOR RATE CONS. WELLS FOR EACH PHASE (L=1,3)"
ICHEK(8) = 0
PWFMIN(8) = 0.0
PWFMAX(8) = 3700.
QTMIN(8) = 0.0
QTMAX(8) = 7100
QI(8,1) = 2323.00
C_W(8,,1) = 1. 0. 0.000 .0974 .015667 .0019 0. 0. .004774 .009122 0.01461 111.003360 0.0

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(9) = 9
IW(9) = 16
JW(9) = 18
IFLAG(9) = 2
RW(9) = 0.49
SWELL(9) = 0.
IDIR(9) = 3
IFIRST(9) = 1
ILAST(9) = 3
IPRF(9) = 0
WELNAM(9) = "A9"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"BOTTOM HOLE PRESSURE FOR PRESSURE CONSTRAINT WELL (IFLAG=2 OR 3)"
ICHEK(9) = 0
PWFMIN(9) = 0.0
PWFMAX(9) = 3700.
QTMIN(9) = 0.0
QTMAX(9) = 7100
PWF(9) = 1740.

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(10) = 10
IW(10) = 2
JW(10) = 11
IFLAG(10) = 4
RW(10) = 0.49
SWELL(10) = 0.
IDIR(10) = 3
IFIRST(10) = 1
ILAST(10) = 3
IPRF(10) = 0
WELNAM(10) = "A10"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"PROD. RATE FOR EACH PHASE (L=1,3)"
ICHEK(10) = 0
PWFMIN(10) = 0.0
PWFMAX(10) = 3700.
QTMIN(10) = 0.0
QTMAX(10) = 7100
QI(10,1) = -843.90

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(11) = 11
IW(11) = 7
JW(11) = 14
IFLAG(11) = 1
RW(11) = 0.49
SWELL(11) = 0.
IDIR(11) = 3
IFIRST(11) = 1
ILAST(11) = 3
IPRF(11) = 0
WELNAM(11) = "A11"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"INJ. RATE AND INJ. COMP. FOR RATE CONS. WELLS FOR EACH PHASE (L=1,3)"
ICHEK(11) = 0
PWFMIN(11) = 0.0
PWFMAX(11) = 3700.
QTMIN(11) = 0.0
QTMAX(11) = 7100
QI(11,1) = 2010.11
C_W(11,,1) = 1. 0. 0.000 .0974 .015667 .0019 0. 0. .004774 .009122 0.01461 111.003360 0.0

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(12) = 12
IW(12) = 9
JW(12) = 17
IFLAG(12) = 4
RW(12) = 0.49
SWELL(12) = 0.
IDIR(12) = 3
IFIRST(12) = 1

ILAST(12) = 3
IPRF(12) = 0
WELNAM(12) = "A12"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"PROD. RATE FOR EACH PHASE (L=1,3)"
ICHEK(12) = 0
PWFMIN(12) = 0.0
PWFMAX(12) = 3700.
QTMIN(12) = 0.0
QTMAX(12) = 7100
QI(12,1) = -611.97

"WELL ID,LOCATIONS,AND FLAG FOR SPECIFYING WELL TYPE, WELL RADIUS, SKIN, WELL NAME"
IDW(13) = 13
IW(13) = 3
JW(13) = 17
IFLAG(13) = 4
RW(13) = 0.49
SWELL(13) = 0.
IDIR(13) = 3
IFIRST(13) = 1
ILAST(13) = 3
IPRF(13) = 0
WELNAM(13) = "A13"

"ICHEK, MAX. AND MIN. ALLOWABLE BOTTOMHOLE PRESSURE AND RATE"
"PROD. RATE FOR EACH PHASE (L=1,3)"
ICHEK(13) = 0
PWFMIN(13) = 0.0
PWFMAX(13) = 3700.
QTMIN(13) = 0.0
QTMAX(13) = 7100
QI(13,1) = -693.95

"CUM. INJ. TIME , AND INTERVALS (PV OR DAY) FOR WRITING TO OUTPUT FILES"
TINJ = 26
CUMPR1 = 26
CUMHI1 = 26
WRHPV = 1
WRPRF = 5
RSTC = 30

"FOR IMES=2 ,THE INI. TIME STEP,CONC. TOLERANCE,MAX.,MIN. COURANT NUMBERS"
DT 0.01
DCLIM = 0.01
CNMAX = 0.1
CNMIN = 0.01

EndInitial

BeginTime 26

" CHANGES IN BOUNDARY CONDITIONS "

IBMOD = 0

IRO = 2
ITSTEP = 1
IFLAG() = 4 4 1 4 4 1 4 1 2 4 1 4 4

NWEL1 = 0

NWEL2 = 12
IDW2() = 1 2 3 4 5 6 7 8 10 11 12 13
QI(1,1) = -625.91
QI(2,1) = -942.54
QI(3,1) = 1994.57
C_W(3,,1) = 0.99574 0. 0.00426 .0000 .07168 .0034 0. 0. .0067 .3339 0.52517 111.0767 0.0
QI(4,1) = -1059.46
QI(5,1) = -829.07
QI(6,1) = 2173.97
C_W(6,,1) = 0.99574 0. 0.00426 .0000 .07168 .0034 0. 0. .0067 .3339 0.52517 111.0767 0.0
QI(7,1) = -2465.65
QI(8,1) = 2250.25
C_W(8,,1) = 0.99574 0. 0.00426 .0000 .07168 .0034 0. 0. .0067 .3339 0.52517 111.0767 0.0
QI(9,1) = -692.0
QI(10,1) = 1956.79
C_W(10,,1) = 0.99574 0. 0.00426 .0000 .07168 .0034 0. 0. .0067 .3339 0.52517 111.0767 0.0
QI(11,1) = -220.73
QI(12,1) = -795.53

TINJ = 51
CUMPR1 = 25
CUMHI1 = 25
WRHPV = 5
WRPRF = 25
RSTC = 50

DT 0.01
DCLIM = 0.001
CNMAX = 0.1
CNMIN = 0.00001

EndTime

BeginTime 51

" CHANGES IN BOUNDARY CONDITIONS "

IBMOD = 0

IRO = 2
ITSTEP = 1

IFLAG() = 4 4 1 4 4 1 4 1 2 4 1 4 4

NWEL1 = 0

NWEL2 = 12
IDW2() = 1 2 3 4 5 6 7 8 10 11 12 13
QI(1,1) = -619.07
QI(2,1) = -746.2
QI(3,1) = 2000.93
C_W(3,,1) = 0.99637 0. 0.00363 .0974 .04948 .0067 0. 0. .00831 .3351 0.3929 111.0839 0.0
QI(4,1) = -1071.65
QI(5,1) = -884.73
QI(6,1) = 2097.34
C_W(6,,1) = 0.99637 0. 0.00363 .0974 .04948 .0067 0. 0. .00831 .3351 0.3929 111.0839 0.0
QI(7,1) = -2041.19
QI(8,1) = 2250.25
C_W(8,,1) = 0.99637 0. 0.00363 .0974 .04948 .0067 0. 0. .00831 .3351 0.3929 111.0839 0.0
QI(9,1) = -1521.71
QI(10,1) = 2076.15
C_W(10,,1) = 0.99637 0. 0.00363 .0974 .04948 .0067 0. 0. .00831 .3351 0.3929 111.0839 0.0
QI(11,1) = -213.65
QI(12,1) = -696.41

TINJ = 226.0
CUMPR1 = 100
CUMHI1 = 100
WRHPV = 5
WRPRF = 100
RSTC = 200

DT 0.005
DCLIM = 0.0003
CNMAX = 0.1
CNMIN = 0.00001

EndTime

BeginTime 226.0

IBMOD = 0

IRO = 2
ITSTEP = 1
IFLAG() = 4 4 1 4 4 1 4 1 2 4 1 4 4

NWEL1 = 0

NWEL2 = 12
IDW2() = 1 2 3 4 5 6 7 8 10 11 12 13
QI(1,1) = -619.07
QI(2,1) = -746.2
QI(3,1) = 2000.93
C_W(3,,1) = 1. 0. .0 .05 .03586 .00665 0. 0. .00132 .0164 0.09 111.00336 0.
QI(4,1) = -1071.45

QI(5,1) = -844.73
QI(6,1) = 2097.34
C_W(6,,1) = 1. 0. .0 .05 .03586 .00665 0. 0. .00132 .0164 0.09 111.00336 0.
QI(7,1) = -2041.19
QI(8,1) = 2250.96
C_W(8,,1) = 1. 0. .0 .05 .03586 .00665 0. 0. .00132 .0164 0.09 111.00336 0.
QI(9,1) = -1521.71
QI(10,1) = 2076.15
C_W(10,,1) = 1. 0. .0 .05 .03586 .00665 0. 0. .00132 .0164 0.09 111.00336 0.
QI(11,1) = -213.65
QI(12,1) = -696.41

TINJ = 276.0
CUMPR1 = 49.5
CUMHI1 = 49.5
WRHPV = 5
WRPRF = 49.5
RSTC = 49.5

DT 0.005
DCLIM = 0.0008
CNMAX = 0.1
CNMIN = 0.00001

EndTime

BeginTime 276

" CHANGES IN BOUNDARY CONDITIONS "

IBMOD = 0

IRO = 2
ITSTEP = 1
IFLAG() = 4 4 1 4 4 1 4 1 2 4 1 4 4

NWEL1 = 0

NWEL2 = 12
IDW2() = 1 2 3 4 5 6 7 8 10 11 12 13
QI(1,1) = -619.07
QI(2,1) = -746.2
QI(3,1) = 2000.93
C_W(3,,1) = 1. 0. .0 .0 .0135 .00185 0. 0. .004774 .008 0.0146 111.00336 0.
QI(4,1) = -1071.45
QI(5,1) = -844.73
QI(6,1) = 2097.34
C_W(6,,1) = 1. 0. .0 .0 .0135 .00185 0. 0. .004774 .008 0.0146 111.00336 0.
QI(7,1) = -2041.19
QI(8,1) = 2250.96
C_W(8,,1) = 1. 0. .0 .0 .0135 .00185 0. 0. .004774 .008 0.0146 111.00336 0.
QI(9,1) = -1521.71

QI(10,1) = 2076.15
C_W(10,,1) = 1. 0. .0 .0 .0135 .00185 0. 0. .004774 .008 0.0146 111.00336 0.
QI(11,1) = -213.65
QI(12,1) = -696.41

TINJ = 551
CUMPR1 = 150
CUMHI1 = 150
WRHPV = 25
WRPRF = 150
RSTC = 551

DT 0.005
DCLIM = 0.0008
CNMAX = 0.1
CNMIN = 0.00001

EndTime

Nomenclature

C_k^o	compressibility of species k [$m^{-1}Lt^2$]
$\tilde{C}_k$	overall concentration of species k in the mobile and stationary phases [L^3/L^3]
$\hat{C}_k$	adsorbed concentration of species k [L^3/L^3]
C_{kl}	concentration of species k in phase l [L^3/L^3]
C_{pl}	constant pressure heat capacity of phase l [$QT^{-1}m^{-1}$]
C_r	rock compressibility [$m^{-1}Lt^2$]
C_{vl}	volumetric heat capacity of phase l [$QT^{-1}m^{-1}$]
C_{vs}	volumetric heat capacity of soil [$QT^{-1}m^{-1}$]
$\vec{\vec{D}}_{kl}$	diffusion coefficient of species k in phase l [L^2t^{-1}]
E_N	parallel efficiency
f	parallelizable fraction of code
h	depth [L]
$\vec{K}$	permeability tensor [L^2]
k_{rl}	relative permeability of phase l
N	number of processors
n_c	number of components
n_{cv}	number of volume-occupying components
n_p	number of phases
P	pressure [$mL^{-1}t^{-2}$]
P_R	reference pressure [$mL^{-1}t^{-2}$]
q_H	enthalpy source per bulk volume [$Qt^{-1}L^{-3}$]
Q_L	heat loss [$Qt^{-1}L^{-2}$]
R_k	total source/sink for species k [$mL^{-3}t^{-1}$]

S_l	saturation of phase l [L^3/L^3]
$S_{\max}$	maximum speedup according to Amdahl's law
S_N	speedup
t	time [t]
T	temperature [T]
T_1	serial run computational time [t]
T_N	parallel run computational time [t]
$\vec{u}_l$	Darcy flux [Lt^{-1}]
α_L	longitudinal dispersivity [L]
α_T	transverse dispersivity [L]
γ_l	specific weight for phase l
δ_{ij}	Kronecker delta function
λ_{rlc}	relative mobility of phase l [$m^{-1}Lt$]
λ_T	thermal conductivity [$Qt^{-1}T^{-1}L$]
λ_{rTc}	total mobility of phase l [$m^{-1}Lt$]
μ_l	viscosity of phase l [$mL^{-1}t^{-1}$]
ρ_l	density of phase l [mL^{-3}]
ρ_s	soil density [mL^{-3}]
τ	tortuosity factor
ϕ	porosity

Bibliography

Abate, J., Wang, P., and Sepehrnoori, K.: "Parallel Compositional Reservoir Simulation on Clusters of PCs," The International Journal of High Performance Computing Applications, Vol. 15, No. 1, Spring 2001, 13-21.

Amdahl, G.M.: "Validity of the Single Processor Approach to Achieving Large Scale Computing Capabilities," Spring Joint Computer Conf., 1967.

Barney, Blaise. "Introduction to parallel computing." Lawrence Livermore National Laboratory 6, no. 13, 2010.

Chien, M.C.H., and Northrup, E.J.: "Vectorization and Parallel Processing of Local Grid Refinement and Adaptive Implicit Schemes in a General Purpose Reservoir Simulator," Paper SPE 25258, SPE Reservoir Simulation Symposium, New Orleans, LA, U.S.A., Feb. 28-March 3, 1993.

Dongarra, J. J. *Sourcebook of Parallel Computing.* Morgan Kaufmann Publishers, 2003.

Gebali, F.: *Algorithms and Parallel Computing.* A John Wiley & Sons Publication, New Jersey, U.S.A., 2011.

Ghasemi Doroh, M.: "Development and Application of a Parallel Compositional Reservoir Simulator," M.SE. Thesis, The University of Texas at Austin, 2012.

Grein, E. A.: "Aplicação Da Computação Paralela Na Simulação De Reservatórios De Petróleo," M.SE. Thesis, Federal University of Santa Catarina, 2015.

Gustafson, J.L.: "Reevaluating Amdahl's Law," Communications of the ACM, Vol. 31, No. 5, May 1988.

Killough, J.E., and Bhogeswara, R.: "Simulation of Compositional Reservoir Phenomena on a Distributed-Memory Parallel Computer," Journal of Petroleum Technology, Nov. 1991, 1368-1374.

Lashgari, H. R. 2014. Development of a Four-Phase Thermal-Chemical Reservoir Simulator for Heavy Oil. PhD Dissertation. The University of Texas at Austin.

Lipovski, G.J., and Malek, M.: *Parallel Computing: Theory and Comparisons.* John Wiley & Sons Publication, U.S.A., 1987.

Padua, D., editor. *Encyclopedia of Parallel Computing.* Springer, 2011.

Parashar, M., Wheeler, J., Pope, G., Wang, K., and Wang, P.: "A New Generation EOS Compositional Reservoir Simulator: Part II – Framework and Multiprocessing," Paper SPE 37997, SPE Reservoir Simulation Symposium, Dallas, Texas, U.S.A., June 8-11, 1997.

PETSc: "PETSc User Manual," Revision 3.1, Mathematics and Computer Science Division, Argonne National Laboratory, March 2010.

Schiozer, D.J.: "Use of Reservoir Simulation, Parallel Computing and Optimization Techniques to Accelerate History Matching and Reservoir Management

Decisions," Paper SPE 53979, SPE Latin American and Caribbean Petroleum Engineering Conference, Caracas, Venezuela, April 21-23, 1999.

Scott, S.L., Wainwright, R.L., Raghavan, R., and Demuth, R.: "Application of Parallel (MIMD) Computers to Reservoir Simulation," Paper SPE 16020, SPE Reservoir Simulation Symposium, San Antonio, Texas, U.S.A., Feb. 1-4, 1987.

Shankland, S., "Keeping Moore's law ticking," October, 2012. www.cnet.com/news/on-the-moores-law-hot-seat-intels-mike-mayberry-q-a/

UTCHEM: "Technical Documentation for UTCHEM-9.0," The University of Texas at Austin, July 2000.

UTCHEM: "User's Guide for UTCHEM-9.0," The University of Texas at Austin, July 2000.

Wang, P., Yotov, I., Wheeler, M., Arbogast, T., Dawson, C., Parashar, M., and Sepehrnoori, K.: "A New Generation EOS Compositional Reservoir Simulator: Part I - Formulation and Discretization," Paper SPE 37979, SPE Reservoir Simulation Symposium, Dallas, Texas, U.S.A., Jun 8-11, 1997.

Wheeler, J.A., and Smith, R.A.: "Reservoir Simulation on a Hypercube," SPE Reservoir Engineering, Nov. 1990, 544-548.

Wikipedia: "Message Passing Interface," http://en.wikipedia.org/wiki/Message_Passing_Interface

Willmore, F.: “Introduction to Parallel Computing,” Texas Advanced Computing Center, The University of Texas at Austin, 2012.