INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 2, Supp , Pages 153–160

LARGE-SCALE RESERVOIR SIMULATIONS USING PC-CLUSTERS

GUOZHONG ZHAO, ZHILIN YIN, YONG WU, AND JINGXIA CHEN

Abstract. Going through the development more than forty years, the overall water-cut to Daqing Oilfield has almost reached 90%. But there is still considerable residual oil in the place. Reservoir engineers want to know the residual oil spatial distribution and how to dig it. This requires large-scale reservoir simulation within limited time. Enlarged scale and highly expected efficiency need higher technical capability for reservoir simulation. By using PC-Cluster technique developed in recent years, large-scale reservoir simulation can be carried out at a relatively low cost. The first PC-Cluster used for reservoir simulation in Daqing Oilfield was designed and built. Based on this developing environment, the serial black oil simulator was parallelized by using the SLES components in PETSc. Then this parallel simulating technique was applied in seven oil production districts of Daqing Oilfield, where the PC-Clusters were configured and the parallel black oil simulator PBRS2.1 we had developed was installed and good results were achieved. In this paper, the hardware and system software configuration of PC-Clusters built is briefly introduced, the idea and method for parallelizing the serial black oil simulator is discussed, and the simulation study at seven typical field blocks and their application results are described and presented.

Key Words. residual oil, large-scale reservoir simulation, PC-Linux, PETSc, parallel black reservoir simulation (PBRS).

1. Introduction

Since reservoir simulation came to be used it had always followed the computer's development to satisfy technical requirements of the oil exploitation industry, with the problem scale being larger, the simulator's main purpose expands to research fine distribution of fluid under ground from the past trends of studying the whole reservoir performance, so that the expenditure for every simulation is bigger and bigger. The parallel computation environment (shared and distributed) came forth ten years ago, synchronously some developers of reservoir simulator began to research how the serial simulator was parallelized, and the commercial version came onto the market later. Taking account of the application convenience and the computation cost, it is necessary to us to parallel the existing serial simulator, and then the independent parallel simulation technique is performed to satisfy the requirements of large scale reservoir simulation in our Oilfield.

In the procedure of reservoir numerical simulation, the computation can be divided into coupled and uncoupled parts. Parallelization of uncoupled part only involves the program technique, the important thing to do is on data decomposition by regions, and most of the serial source code for this part can be adopted for

parallel program. The coupled part is mostly executed in the process for solving linear equations, where the parallel solving method we use must be different from the serial case. Therefore, the key work is the development and implementation of the parallel solving method for large, sparse and unsymmetrical linear system. By the use of the differential equation parallel solver package-PETSc(Portable, Parallel, Extended Toolkit for Scientific Computation) coming forth from Argonne National Lab of America, for the developer of reservoir simulators, it is possible to realize that the existing serial simulator is parallelized quickly. Therefore, in terms of the use of PETSc's options on the LINUX PC-cluster, we made the serial black oil simulator be parallelized and developed the parallel black oil simulator PBRS, and successfully applied it in seven oil production areas of the current reservoir studies in Daqing.

2. The hardware and system software configurations of PC-Clusters in Daqing

The scale of PC-Cluster is from several nodes to thousands, if we keep extensibility, the more nodes, higher the expense. The first PC-Cluster to be built in Daqing was mainly used for experiment and software development, its function was primary, and its performance was secondary. In order to enhance the probability of success, we reduced the cost as much as possible, so the number of nodes is not large. Hardware and system configuration of the integrated PC-Clusters is as follows:

Hardware configuration

(1) Node: one master (control) node, eight slave (computation) nodes;
(2) CPU: Intel Pentium III 800EB or higher;
(3) Memory: 2GB for master node, 1GB per slave node;
(4) Network card: two Intel 100/1000M PC cards, teaming, for master node, one Intel 100/1000M PC card per slave node;
(5) Switch: 24/12Port 100/1000M Switch;
(6) Hard disk: 18GB inside for every node, 200GB RAID connected to master node;
(7) Display: 21 inches display linked to master node.

System software configuration:

(1) Linux operating system RH7.1 or higher;
(2) MPI (Message Passing Interface) 1.2;
(3) The differential equation parallel solver package-PETSc (Portable, Parallel-Extended Toolkit for Scientific Computation) from American National Lab.

3. Parallel solving strategies in PBRS software

Most computation examples indicated that it took 98 percent of all simulation time when we used serial simulator to calculate Jacobian coefficients and to solve the linear system coupled by grid equations and well equations. In order to obtain good parallel efficiency, the computation and data involved in the two steps above must be distributed into every parallel node. It is possible to realize parallel computing for the other portions, but we have to consider that the higher communication expense is not worth the candle.

In order to reduce the workload of parallel coding, it is convenient to adopt Master/Slave parallel solution strategies. A more brief account of it is as follows: slave process takes charge of Jacobian computation, performs the calculating task of linear system and couples grid equations with well equations to solve linear

system, master process is in charge of the other work such as input, output, well management and the solving process controls, etc..

The parallel system works on the popular standard MPI interface (PETSc must be supported by MPI), which increased the communication efficiency between processes and source code could be migrated across different system environment.

4. Data distribution and memory management

Data used to a running reservoir simulator includes scalar data and grid array data, and in principle the latter must be allocated to slave processes to be stored locally, as it takes most memory space. Because of the output, several important grid arrays such as pressure, saturation and gas-oil ratio, all are still stored in the master process. Here the scalar data also includes a few small arrays that are independent of the grid number and well number, such as relative permeability and PVT table, etc.. It is negligible for the occupied memory, so they can repeatedly be stored in every process.

4.1. Data partition. After obtaining the partition instruction from user, it is easy to obtain the partition scheme. If we do only a little coding, the local grid number of every region can be set with approximate scale each other, this will benefit loading balance. In view of the need for automatic data exchange between regions, the indices of region borderline are set first along one direction and then along another direction. Some information may be used later must be accurately stored after partition, such as regions, position of region borderline, its size and indices, etc. In addition, the wells in every region are also ordered again, and the relationship between its local and global indices must be reserved.

4.2. Memory management. After the master process is started, we read the restart file and allocate for enough memory to read all primary information, necessary backup data is stored, dynamic memory is released and a new one requested, until the data stored in every region has been transported and then the backup data is imported again. Dynamic memory of the slave process is requested to satisfy its minimum needs.

5. Parallelizing of Jacobian Computation

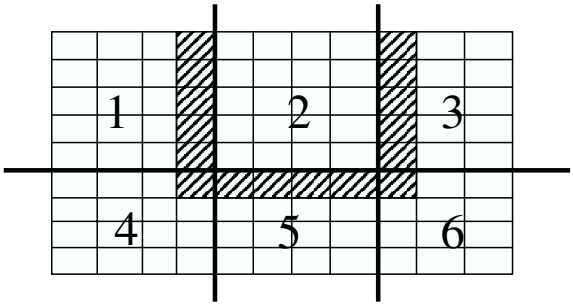
Jacobian computation can be accurately parallelized. Because the Jacobian elements of boundary grid equation are related to the boundary grid data of the neighbor subregions, the grid system of every subregion can be extended outwards from the inner border during this period, so current dummy grid system includes all the grids in relation to Jacobian calculation in this subregion.

If NXD(I) and NYD(I) denote actual grid numbers in subregion along I and J direction, NXV(I) and NYV(I) denote dummy grid numbers in subregion along I and J direction, then region 2(the grids with bias as background are extended dummy grids) in FIGURE 1 shows as follows:

$$\begin{aligned} &\text{NXD(2)=4,} && \text{NYD(2)=5,} \\ &\text{NXV(2)=6,} && \text{NYV(2)=6.} \end{aligned}$$

Each subregion is regarded as an individual model to use Jacobian calculation source code of primary serial program. After array data was set according to natural order in dummy grid system, for I subregion, we only need to use NXV(I) and NYV(I) to replace primary NX and NY. In FIGURE 1, NX=11, NY=9.

FIGURE 1. Dummy grid system of Jacobian calculation in subregion.



This can avoid the necessity of transferring information frequently in the process of Jacobian computation, but the whole computation load is increased in contrast to serial calculation. It can only be eliminated if logic filter is added in code, here it is necessary to set a local integer array, after Jacobian calculation is finished it is easy to map from dummy grid to actual grid for use in the latter linear system solving.

6. Parallel solving linear system with PETSc

PETSc is a extensible, large scale parallel solving software package for Scientific Computation, it can be run on many kinds of operating system, it is fit for parallel solving of partial differential equations [1]. It consists of some basic tools and many components included in data object, data and grid management, linear equation solver(SLES), nonlinear equation solver(SNES) and differential equation solver(TS). Data object mainly includes vectors(VEC) and matrices (MAT). SLES mainly includes the KSP and PC components for subspace methods and preconditioners. The user can use part or all of these components to develop parallel application according to their own needs.

6.1. Local setting of linear system. In order to obtain better parallel efficiency during parallel reservoir simulation using SLES in PETSc, the key is how to assemble matrices. According to the data partition scheme previously mentioned, row elements derived from grid and well equations for a subregion are stored in corresponding process, in this way it can ensure there is no data transfer between subregions during local setting of matrices.

The developer and releaser of PETSc strongly suggests that users use two integer arrays (D-NNZ and O-NNZ) to let the setting function get the location of nonzero elements on diagonal block and non-diagonal block of the matrix. In this way, it was easy for us to compress and store data based on rows in the procedure of setting, and then we could get higher parallel solving efficiency. The first thing to solve this problem is to use logic trace for the linear system setting procedure, and exactly pass the location of nonzero elements, then produce a subroutine that run in advance to set D-NNZ and O-NNZ arrays. We need not run this routine every time we solve the problem, we only need to run it before the first Newton iteration at any time step (including the first time step for this simulation) when well production or injection status changes.

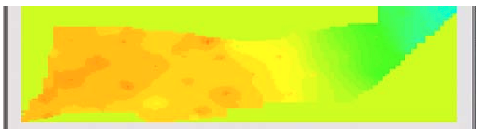


FIGURE 2. Pressure field of the third layer serial computing for 2,618 days with single CPU.

6.2. Choice of parallel components of PETSc. There are many subspace iteration methods and serial preconditioners in SLES, but there are only two components for parallel preconditioners, block Jacobian (BJACOBI) and addition Schwarz (ASM). The different combinations of these components are used to solve different problems. Numerical simulation examples indicate that the combination of two subspace iterations for incomplete LU preconditioners and GMRES [2] and BCGS [3] works well to reservoir simulation problem, other combinations can not compete to it. Because BJACOBI is only a special case of ASM with no overlay, it is adequate to choose ASM. Some options are given for end users to set concrete parameters, and then the users can try to choose the parameters in detail to the actual problems.

7. Communication between processes

Data transfer will occur between the master process and each slave process. When a slave process is started, the data independent of time is first received from the master process in one-shot time. The master process must receive the variables of grid pressure, saturation and gas-oil ratio from a slave process for use with material balance analysis and possible print output before each time step is ended. In every Newton iterative step, the local maximal absolute value of the residual of the finite differential equation and the unknown change must be transferred to the master process, the iterative control data will be transferred to slave process again after the master gathers this data.

Communication is also necessary between processes in which the neighbor regions exist. During each Newton iterative step, the neighboring subregions must transfer unknown changes in boundary grids for the use of variable update in subregion and Jacobian computation in the next Newton iterative step.

8. Parallel examples and performance analysis on the first PC-Cluster in Daqing

We have tested PBRS software with four different actual models with 2,000, 210,000, 440,000 and 1,160,000 cells, the result indicates:

(1) Taking the model with 210,000 cells for example, we compared the parallel computation result by 8 CPUs and the serial computation result by single CPU for 2,618 days. The difference of maximum balance errors of oil, gas, and water is -0.0016, -0.0016 and 0.0033, respectively; the difference of maximum single well daily oil production, daily gas production and water cut is $0.07m^3$/day, $-8m^3$/day and 0.02%, respectively; the MAP plots of pressure field and saturation field (showed by FIGURE 2 to FIGURE 5) was too similar to distinguish the difference by sight, they also do not depend on region decomposition. Considering that they are all numerical solutions, the results are correct in the numerical solution meaning.

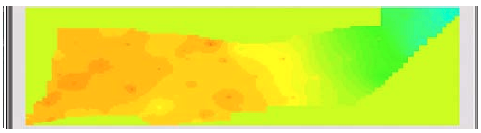


FIGURE 3. Pressure field of the third layer parallel computing for 2,618 days with 8 CPUs.

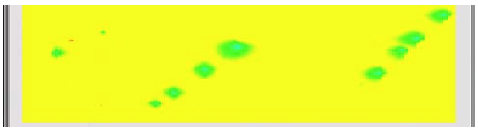


FIGURE 4. Saturation field of the forth layer serial computing for 2,618 days with single CPU.

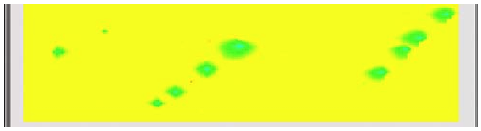


FIGURE 5. Pressure field of the forth layer parallel computing for 2,618 days with 8 CPUs.

TABLE 1. Parallel efficiency and acceleration ratio of 210,000 cells.

Number of CPUs	Time steps	Number of Newton iterative	CPU time /s	Efficiency $E_p = S_p/p$	Acceleration ratio $S_p = T_1/T_p$
1	320	5	14224		
2	368	5	8759	82.0%	1.63
4	396	5	4508	79.1%	3.61
8	189	5	2386	74.5%	5.96
16	208	5	1395	64.0%	10.2

(2) The computation capability of assembled PC-cluster and developed parallel black oil simulator PBRS reaches a million cells.

(3) The running speed of single CPU exceeds that of Origin2K parallel computer, data communications between nodes in PC-cluster depend on the network, on which the speed is lower than Origin2K, but the whole computation speed of model with more than million cells is higher than Origin2K, for example, for the model with 1160,000 cells, it needs 70 hours to compute with ten CPUs in Origin2K, but it only takes about 42 hours on the PC-cluster we built first.

(4) Data communication of the PC-cluster depends on the network, with the number of CPU and model scale increasing, data quantity increases, but the efficiency and acceleration ratio decreases. It is impossible to run the models with 440,000 and 1,160,000 cells with single CPU, so it is impossible to compare the efficiency and acceleration ratio. TABLE 1 shows parallel efficiency and acceleration ratio of 210,000 nodes.

TABLE 2. The characterization of reservoir simulation in seven typical field blocks.

Name of block	Areas /km^2	Number of layers	Number of wells	Exploitation history (put into production)	Number of grids
6∼16 well regions of the north block of Lamadian oil field	8.2	91	279	28 (in 1974)	472,017
The east of the third north block of Sabei oil field	10.56	81	684	39 (1963)	753,300
The east of the first north block of Sazhong oil field	9.46	50	514	42 (1960)	550,368
The east of the second south block of Sanan oil field	5.5	65	251	38 (1964)	455,000
X4∼6 regions of the third north block of Xingbei oil field	6.9	99	308	36 (1966)	635,283
The east of X10∼11 block of Xingnan oil field	12.92	56	369	31 (1971)	804,272
The east block of Tainan oil field	12.5	22	135	20 (1982)	204,600

9. Actual examples of seven typical field blocks

We respectively chose a typical block from the first to the seventh oil production districts in Daqing Oilfield in 2002. To each typical block we built a geological model consisting of single sand layers based on fine geology studies, and carried out reservoir simulations including up to 40-year history matching and effect prediction on a series of development adjustment solutions to be optimized, TABLE 2 shows the brief summary.

The total area has reached 66km^2, which is about 5% of the old oil production districts. The number of wells involved has reached 2540, which is about 10% of the old oil production districts. The total OOIP of research blocks has reach $2.9 \times 10^8 t$.

We used single sand sediment models as simulation zones in vertical grid partition for the seven field blocks, it kept consistency with the fine geology studies. We adopted a uniformity rectangle grid system in plane grid partition, which is designed in the view of existed well pattern and of possible fill-in well. The design goal meets the requirement of simulation precision and we reduced the grid number as much as possible and took little account of finer and finer grids. But the description for the target reservoirs has reached the topmost fine level in reservoir simulation history in Daqing.

10. The conclusions

(1) The integrated techniques of PC-cluster for large scale reservoir simulation have been tried in Daqing, and one PC-cluster has been built for use in developing parallel simulation software.
(2) The approach of parallel simulation technique has been explored, namely, network and MPI application software is used as communication tools on

the LINUX PC-cluster. Through the use of parts of the PETSc components, we have achieved parallelization for the existing serial simulator.

(3) During the process of research, we resolved a series of parallel key problems about region decomposition strategy, Jacobian coefficient parallel calculation, well management parallel consideration, linear solver building and the management of input and output, etc.

(4) We have achieved the development of parallel black oil simulation software PBRS2.1 with independent copyright and partially broke away from dependence on the commercial one. The popular application of reservoir simulation and development technique has been accelerated in our oilfield, and a steady foundation has been built in order that the kernel components of reservoir simulator can be designed in Petrochina.

(5) The application has been used in actual work situations and it is possible for large-scale numerical simulation technique to be widely applied in Daqing Oilfield. The parallel simulation technique can provide finer and more reliable basis in order to determine development and adjustment projects. From this point of view, the economic benefit that PC-clusters bring to us is indirect and tremendous, in the way of economy, the enormous expenses have been decreased by applying and developing this kind of computer hardware and software.

References

[1] S. Balay, W. D. Gropp. L. C. McInnes and B. F. Smith, Efficient Management of Parallelism in Object-Oriented Numerical Software Libraries, p. 163–202, Modern Software Tools in Scientific Computing, E. Arge etc. Ed., Birkhauser Press, 1997.

[2] Youcef Saad and Martin H. Schultz. GMRES, A generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J. Sci. Stat. Comput., 7:856–869, 1986.

[3] H. A. van der Vorst. BiCGSTAB: A fast and smoothly converging variant of BiCG for the solution of nonsymmetric linear systems. SIAM J. Sci. Stat. Comput., 13:631–644, 1992.

Exploration & Development Research Institute of Daqing Oilfield Co. Ltd., Heilongjiang, China