Jurnal Teknologi, 45(F) Dis. 2006: 67 – 82

SPEED UP OF LATTICE GAS AUTOMATA SIMULATION OF POLYMER FLOW USING A CLUSTER SYSTEM

MUHAMMAD TAUFIQ FATHADDINA[1], & MARIYAMNI AWANG[2*]

Abstract. The performance of polymer flooding may be estimated from microscopic behaviour of polymer particles in the pores of porous media. Lattice gas automata method is a modelling method that has been used by researchers to study flow behaviour on a particle scale. It had been also used to simulate polymer flow in porous media for studying microscopic interactions between rock and polymer. However, the computation time was too lengthy for even a 5 cm long porous media sample. On this study, parallel computing using standalone computers and a cluster system was investigated in an effort to decrease the computation time of modelling polymer flooding using lattice gas automata. From the results, speedup due to parallel processing was greater than 3.75 times for four processors and less. Differences in the estimations of displacement efficiency and saturations between sequential and parallel programming were less then three percent. It was concluded that parallel programming was successfully used to speed up computations in polymer flooding without causing significant variations in the results.

Keywords: Parallel computation; lattice gas automata; polymer displacement

Abstrak. Prestasi banjiran polimer boleh dianggar daripada tingkah laku mikroskopik partikel polimer dalam pori-pori media berliang. Kaedah kekisi gas automata adalah suatu kaedah pemodelan yang telah digunakan oleh penyelidik-penyelidik untuk mengkaji kelakuan aliran pada skala pori. Ia juga boleh digunakan untuk menyelaku aliran polimer dalam media berliang untuk mengkaji interaksi antara batuan dengan polimer. Namun, masa penghitungan terlalu panjang, walaupun untuk sampel media yang panjangnya 5 cm. Dalam kajian ini, penghitungan selari menggunakan komputer berdiri sendiri and sistem gugusan disiasati dalam usaha untuk mengurangkan masa penghitungan pemodelan banjiran polimer menggunakan kekisi gas automata. Daripada hasil kajian, speedup yang berasaskan pengolahan selari didapati lebih daripada 3.75 kali ganda, untuk empat pemprosesan atau kurang. Perbezaan dalam anggaran kecekapan penyesaran dan penepuan antara pengaturcaraan berjujukan dengan pengaturcaraan selari adalah kurang daripada tiga peratus. Oleh itu, dapatlah disimpulkan bahawa pengaturcaraan selari telah berjaya digunakan untuk melajukan penghitungan banjiran polimer tanpa menyebabkan variasi yang bererti daripada hasil keputusan.

Kata kunci: Pengiraan selari; kekisi gas automata; penyesaran polimer

1.0 INTRODUCTION

In polymer flooding, the reactions between polymer molecules, hydrocarbon fluids, water and rock surfaces affect the efficiency of displacement. Modelling and simulation

[1] Petroleum Engineering Department, Universitas Trisakti, Jakarta
[2] Petroleum Engineering Department, Universiti Teknologi Malaysia, 81310 Skudai, Johor Darul Ta'zim, Malaysia
*Corresponding author: Tel: 07-5535558, Fax: 07-5581463

on a pore scale is suitable for studies on adsorption, gelling and displacement efficiency because the interactions of the molecules determine the behavior on the macroscopic scale. Lattice gas automata (LGA) is a cellular automata technique that has been used in many disciplines to model fluid flow on a particle scale.

Frisch *et al.*, [1] introduced LGA method. In their model, a triangular lattice with hexagonal symmetry was applied. Their model allows particle movements in six directions. When particles meet at a site, they collide according to the determined collision rules that locally conserve mass and momentum. On the macroscopic scale, the Navier-Stokes equation [1 – 4] is reproduced from LGA particle flow.

Balasubramanian *et al.*, [5] found that the particle movements of the model flowing between parallel plates with no-slip boundary conditions gave a good agreement with theoretical Poiseuille flow equation. Lattice gas automata model was also applied for simulating fluid flow in a porous medium with complex boundary conditions by Rothman [6] who found that the model satisfied the Darcy equation. The immiscible lattice gas automata (ILGA) model had also been developed to model immiscible fluids, where surface tension between the fluids was introduced [7 – 13]. Fathaddin and Awang [14] introduced several additional collision rules to simulate polymer adsorption process in porous media. In addition, ILGA has also been applied for simulating polymer displacement in porous media at laboratory scale [15].

A higher order of size is necessary in order to study polymer flooding so that one is able to compare with laboratory scale porous media. A simulation of a 10 × 5 cm model was carried out. However, it took from one to three days using a personal computer. Parallel computing is suitable for LGA modelling because of its inherently parallel nature. Consequently, parallel computers were considered for this work. There are two major developments of modern computer. The massively parallel processors (MPPs) are the most powerful computers, which combine a few hundred to a few thousand CPUs in a single large cabinet connected to hundreds of gigabytes of memory. However, the operating or maintenance costs of the MPPs are usually high unlike the cluster system that is a cheaper alternative to accelerate the simulation rate. A cluster system is a set of computers connected by a network. When several MPPs were arranged in a cluster system, unequaled computational power resulted [16]. Previous researchers recognized the necessity of parallel computers for LGA simulation [1, 6, 11, 17, 18]. However, the computation time reduction for LGA using parallel computers has not been reported.

In this research, the lattice gas automata simulations of polymer displacement were conducted using both standalone computers and a cluster system to determine the speedup by comparing the CPU time of a sequential program with the CPU time of the parallel program. Results from parallel computation were also compared with the sequential program to test for any variation caused by additional collision rules.

2. 0 PARALLEL COMPUTATION

Two operating systems i.e. Linux and Windows were applied in these simulations. Since PVM [19] was only compatible with Linux, the operating system was used for parallel simulation in a cluster. Meanwhile, Windows was used for parallel simulation using standalone computers.

For parallel computation, the model was divided into several regions along the direction of fluid flow. Two boundary cases were considered. In case one, the regions were non-communicating, and therefore the flow channels were contained in each region by dividing the porous medium through the solid. Since the porous medium was heterogeneous, the regions had different pore structure. The simulation of each region was conducted independently using a standalone computer and the completion time of the simulation depended on the slowest region to simulate. In case two, division was made through the flow channels or by dividing the pore space. Consequently, the boundaries of each region were the middle of the flow channels and each region communicated at the flow boundaries. A cluster system was used to simulate this system. In a real porous medium, a mixture of these two boundaries will occur in porous medium. By conducting simulation using the two extreme models, the real behavior can be estimated to be within the range studied.

2.1 Hardware and Software

Table 1 summarizes the hardware and the operating systems. The standalone machines were personal computers with Windows ME operating system. The cluster consisted of nodes operating with a Linux 2.4 version and PVM for the parallel protocol. A computer represents a node. The Message Passing Interface (MPI) was used to provide communication through a network that has a communication speed capacity of 100 Mbit/s.

Table 1 Specification of computers

Computer types	**A (Standalone)**	**B (Cluster)**
Processor	Pentium 4; 2.4 GHz	Pentium 4; 2.0 GHz
Random Access Memory	256 MB	256 MB
Operating System	Windows ME	Linux Kernel 2.4
Fortran Compiler	Power Station	(g77) 3.3.1

The parallel program consisted of a master program running on one node that controlled slave programs installed on the other nodes. Both master and slave programs were written in FORTRAN. Initialization, controls, synchronization, instructions, collection and display of results resided on the master node, while actual computation of simulation resided on the slave nodes. Each slave node simulated a region and was able to receive and send data to the master node as well as two adjacent slave nodes that represented adjacent regions. All slave nodes began the same task at any particular time, which meant propagation or collision step started simultaneously in all regions. When the all nodes had completed the previous task, another round of instructions was received from the master.

2.2 Collision Rules for Communicating Model

Figure 1 shows the propagation of particles across the boundary. Assuming no collisions occur at boundary sites, a particle that reaches a boundary site will continue moving with the same velocity into another region. Collision between the fluid particles at other sites is governed by the collision rules of Frisch-Hasslacher-Pomeau (FHP) III

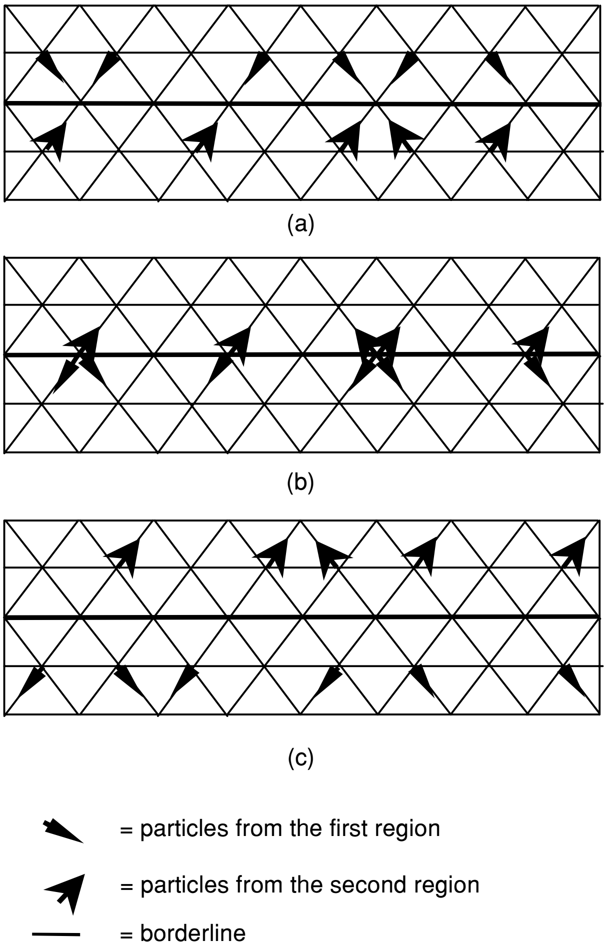
(a)
(b)
(c)
= particles from the first region
= particles from the second region
= borderline

Figure 1 Propagations of particles across the region boundary

model of lattice gas automata method [18]. Figure 1 illustrates the movements of particles across a boundary. The particles move from the neighbour sites of the boundary sites (Figure 1(a)). When the particles reach the boundary sites (Figure 1(b)), they continue moving with the same velocity since no collision occur at the sites (Figure 1(c)).

The number and direction of particles that cross the boundary are determined by the boundary particles, i.e. particles at sites adjacent to the boundary. These particles provide the communication between adjacent regions since the particles that leave one region will be the same particles that enter another region and vice versa. The boundary sites are co-owned by adjacent regions. Therefore, when the lattice model is divided into regions, the same boundary sites are included both in the two adjacent regions.

2.3 Formulations

Parallel performance can be measured by speedup S (P, N) and parallel efficiency, E (P, N). Speedup is defined as

$$S(P,N)=\frac{T(1,N)}{T(P,N)} \qquad \text{... 1}$$

Parallel efficiency is defined as

$$E(P,N)=\frac{S(P,N)}{P} \qquad \text{... 2}$$

Where $T(P, N)$ is the time required for a given algorithm to solve problem of size N on P processor.

Oil and polymer dynamic viscosities for FHP-III model are as follows:

$$\mu_o=\rho_o\frac{1}{28}\frac{1}{d_o(1-d_o)}\frac{1}{1-8d_o\frac{(1-d_o)}{7}}-\frac{1}{8} \qquad \text{... 3}$$

$$\mu_p=\rho_p\frac{1}{28}\frac{1}{d_p(1-d_p)}\frac{1}{1-8d_p\frac{(1-d_p)}{7}}-\frac{1}{8} \qquad \text{... 4}$$

Porosity is defined as the ratio of void space to the bulk volume. In equation form:

$$\phi=\frac{\text{void space}}{\text{bulk volume}} \qquad \text{... 5}$$

In lattice gas, effective porosity is defined as the number of lattice sites representing void sites divided by the total number of lattice sites. In equation form:

$$\phi = \frac{\sum_r s_v(r)}{\sum_r s_v(r) + \sum_r s_s(r)} = \frac{\sum_r s_v(r)}{\sum_r s(r)} \qquad \ldots 6$$

Saturation of any fluid in the reservoir is defined as the fraction of the pore space occupied by the fluid. If only one fluid fills up the void space, the saturation of the fluid is equal to one. If two fluids are present, oil and polymer, saturation of each fluid is less than one. In mathematical expressions, saturations of the oil and polymer are respectively given by:

$$S_o = \frac{\text{oil volume}}{\text{total pore volume}} \qquad \ldots 7$$

$$S_p = \frac{\text{polymer solution volume}}{\text{total pore volume}} \qquad \ldots 8$$

In lattice gas, oil saturation is defined as the total numbers of oil particles divided by the total numbers of oil and polymer particles. In equation form, oil saturation is given by:

$$S_o = \frac{\sum_i N_{oi}(r,t)}{\sum_i N_{oi}(r,t) + \sum_i N_{pi}(r,t)} \qquad \ldots 9$$

Moreover, polymer solution saturation is given by:

$$S_p = \frac{\sum_i N_{pi}(r,t)}{\sum_i N_{pi}(r,t) + \sum_i N_{pi}(r,t)} \qquad \ldots 10$$

Residual fluid saturation is defined as the remaining fluid saturation in porous media after sweeping by displacing fluid. The residual oil saturation, S_{or}, can be expressed as follows:

$$S_{or} = \frac{\text{residual oil volume}}{\text{total pore volume}} \qquad \ldots 11$$

In lattice gas, the residual displaced oil saturation is defined as the total numbers of the remaining oil particles divided by the total numbers of oil and polymer solution particles after displacement. In equation form, the residual displaced oil saturation is given by:

$$S_{or} = \frac{\sum\limits_i N_{oi}\left(r,t_{dis}\right)}{\sum\limits_i N_{oi}\left(r,t_{dis}\right)+\sum\limits_i N_{wi}\left(r,t_{dis}\right)} \qquad \text{... 12}$$

In LGA form, equations of oil and water relative permeabilities [20] are as follows:

$$k_{ro}= k_{ro}^{o}\left(\frac{1}{r}\sum_{r=1}^{n}\frac{1-\dfrac{\sum\limits_i N_{pi}\left(r,t\right)}{\sum\limits_i N_{oi}\left(r,t\right)+\sum\limits_i N_{pi}\left(r,t\right)}-\dfrac{\sum\limits_i N_{oi}\left(r,t_{dis}\right)}{\sum\limits_i N_{oi}\left(r,t_{dis}\right)+\sum\limits_i N_{pi}\left(r,t_{dis}\right)}}{1-\dfrac{\sum\limits_i N_{oi}\left(r,t_{dis}\right)}{\sum\limits_i N_{oi}\left(r,t_{dis}\right)+\sum\limits_i N_{pi}\left(r,t_{dis}\right)}-\dfrac{\sum\limits_i N_{wi}\left(r,o\right)}{\sum\limits_i N_{oi}\left(r,o\right)+\sum\limits_i N_{wi}\left(r,o\right)}}\right)^{n_o}$$

... 13

$$k_{rw}= k_{rw}^{o}\left(\frac{1}{r}\sum_{r=1}^{n}\frac{1-\dfrac{\sum\limits_i N_{pi}\left(r,t\right)}{\sum\limits_i N_{oi}\left(r,t\right)+\sum\limits_i N_{pi}\left(r,t\right)}-\dfrac{\sum\limits_i N_{wi}\left(r,o\right)}{\sum\limits_i N_{oi}\left(r,o\right)+\sum\limits_i N_{wi}\left(r,o\right)}}{1-\dfrac{\sum\limits_i N_{oi}\left(r,t_{dis}\right)}{\sum\limits_i N_{oi}\left(r,t_{dis}\right)+\sum\limits_i N_{pi}\left(r,t_{dis}\right)}-\dfrac{\sum\limits_i N_{wi}\left(r,o\right)}{\sum\limits_i N_{oi}\left(r,o\right)+\sum\limits_i N_{wi}\left(r,o\right)}}\right)^{n_w}$$

... 14

The values of $k_{rw}{}^{o}$, $k_{ro}{}^{o}$, n_w and n_o obtained from simulation and experimental results were 0.2154, 0.6357, 4.28 and 1.10 respectively.

Displacement efficiency, D_{eff}, is defined as the volume of produced oil divided by the total volume of oil at initial condition. In lattice gas, the displacement efficiency is defined as the number of produced oil particles from porous medium divided by the total numbers of oil particles in the porous medium at initial condition. Displacement efficiency is given by:

$$D_{eff} = \frac{\sum_i N_{o_i}(r,0) - \sum_i N_{o_i}(r,t_{dis})}{\sum_i N_{o_i}(r,0)} \qquad \text{... 15}$$

The regions of porous media in parallel computation are assumed as parallel flow in a porous medium with horizontal layers. Therefore, the averaging of the rock and fluid properties of all regions are estimated by arithmetic approximation equations. Arithmetic average values of porosity, saturation and relative permeability are respectively calculated using Equations (16) to (18):

$$\overline{\phi} = \frac{\sum_{r=1}^{n} \phi_r h_r}{\sum_{r=1}^{n} h_r} \qquad \text{... 16}$$

$$\overline{S} = \frac{\sum_{r=1}^{n} S_r h_r}{\sum_{r=1}^{n} h_r} \qquad \text{... 17}$$

$$\overline{k_r} = \frac{\sum_{r=1}^{n} k_{r_r} h_r}{\sum_{r=1}^{n} h_r} \qquad \text{... 18}$$

while the average value of displacement efficiency is calculated using the following equation:

$$\overline{D}_{eff} = \frac{\sum_{r=1}^{n}\left(\sum_i N_{o_i}(r,0) - \sum_i N_{o_i}(r,t_{dis})\right)_r}{\sum_{r=1}^{n}\left(\sum_i N_{o_i}(r,0)\right)} \qquad \text{... 19}$$

3.0 SIMULATION RUNS

Simulations of polymer displacement assuming no communication between the regions were conducted on computer type A as given in Table 1. The porous medium was

5 cm by 4 cm as shown in Figure 2. White bricks represent solid, white areas represent displacing polymer, gray areas represent displaced oil, and black dots represent adsorbed LGA polymer particles. No-slip boundary conditions were applied for colliding fluid particles at solid wall and collision rules for polymer adsorption process were applied [14]. Initial polymer concentration was 1000 ppm. Simulation terminated after one pore volume of displacing polymer was injected. Communicating flow was tested on the 5 cm by 4 m model as shown in Figure 3. Simulations were conducted using a cluster system with specifications as given in Table 1. Both systems, communicating and no-communicating flows were simulated with one (whole system), two, or four regions.

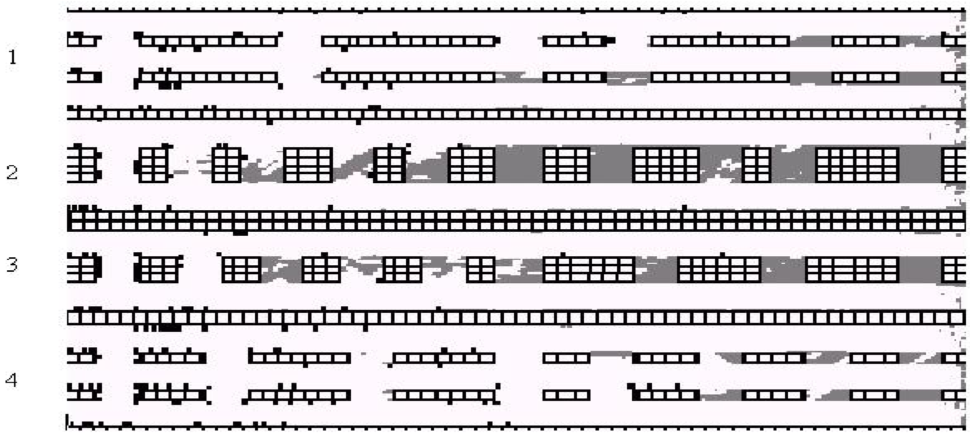

Figure 2 A porous medium consists of four non-communicating channels

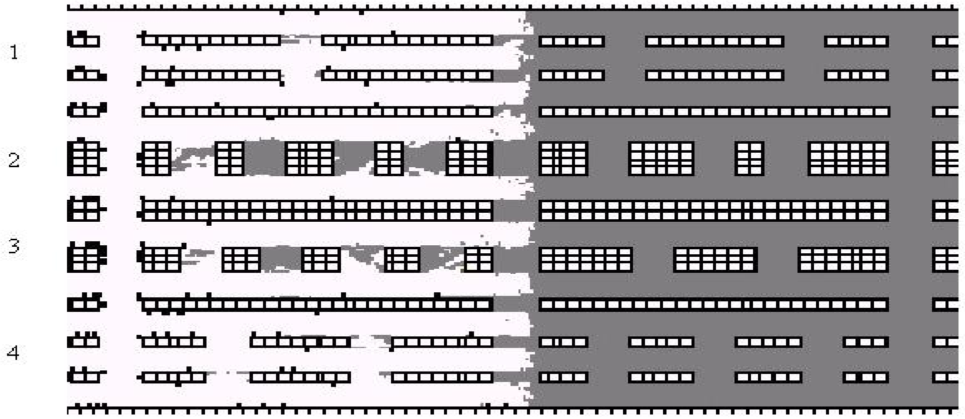

Figure 3 A porous medium consists of four communicating channels

4.0 RESULTS AND DISCUSSION

4.1 Non-communicating Flow

The average porosity, initial water saturation (S_{wi}) and residual oil saturation (S_{or}) of the model are given in Table 2. The average porosities of the two and four regions were equal to 0.634515 that was the same as that of the whole porous medium. This means that although every region had different porosity, the division into regions did not change the percentage of pore space and solid material overall. Porosity was determined at initial condition. The same average porosity for all divided cases indicates that separating the flow did not alter the porosity estimation.

Table 2 Properties of polymer displacement for non-communicating flow system

Parameter	Non-communicating System			Communicating System		
	5×4 cm^2	5×2 cm^2	5×1 cm^2	5×4 cm^2	5×2 cm^2	5×1 cm^2
ϕ, fraction	0.63452	0.63452	0.63452	0.65247	0.65247	0.65247
Δ, %	0	0	0	0	0	0
S_{wi}, fraction	0.05550	0.05550	0.05558	0.05561	0.05558	0.05562
Δ, %	0	0	0.149	0	0.0536	0.0180
S_{or}, fraction	0.32803	0.31953	0.32003	0.30960	0.32080	0.30238
Δ, %	0	2.5907	2.4376	0	3.6180	2.332

Similarly initial water saturation is determined at the initial condition. Locations of water and oil particles were randomly arranged by the random number generator. For instance, ten random numbers were generated. One of the numbers was set for water particles. The computer program determined whether a site was occupied by water or oil, based on probability. In this case, initial water saturation was expected to be 10%. The difference between the expected and defined saturations was most likely due to the random numbers.

The average initial water saturation of whole, two and four regions were 0.055498, 0.055498 and 0.05558 as given in Table 2, which gave differences of 0 % and 0.149 % for the two and four regions compared to the whole region.

Most oil particles were trapped in channels, which were perpendicular to the mainstream direction. The average saturations of the whole, two and four regions are 0.328025, 0.319527 and 0.320029, respectively. The differences of the residual oil saturation of whole porous medium curve to those of two and four regions are 2.5907% and 2.4376 % respectively.

Displacement efficiencies for the whole, two, and four regions are shown in Figure 4. The curves of two and four regions merge using Equation (16). The average differences

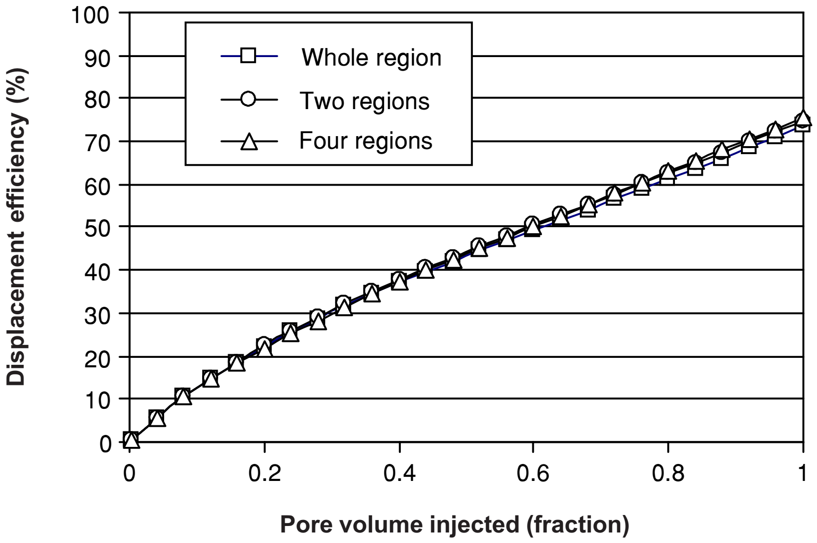

Figure 4 Comparisons of displacement efficiency for non-communicating flow system

of the whole region curve to two and to four regions are 1.8548% and 1.6410%, respectively.

The computation time of various processor numbers for polymer displacement in porous media with non-communicating channels is depicted in Figure 5. The computation time reduced as processor number increased. Speedup and parallel efficiency of the processors are given in Table 3. The table shows that the performance

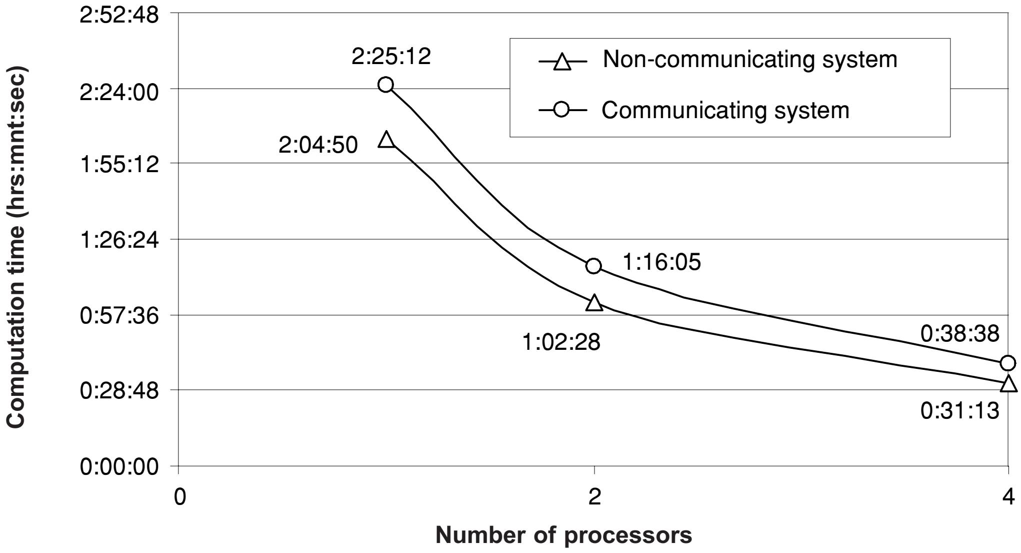

Figure 5 Computation time for various numbers of processors

Table 3 Performance of non-communicating and communicating flow systems

Number of processor	Non-communicating system		Communicating system	
	$S(P,N)$	$E(P,N)$	$S(P,N)$	$E(P,N)$
1	1	1	1	1
2	1.9984	0.9992	1.9084	0.9542
4	3.9989	0.9997	3.7584	0.9396

approached ideal condition because no time was needed for communication and so parallel efficiency $E(P,N)$ was equal to one and speedup $S(P,N)$ of two and four processors is 2 and 4, respectively.

The division of porous medium along the solid sites in this case did not influence the LGA method. Random number had several functions in the LGA models. The random number was used to determine the direction of entered particles from the source at the left boundary. Another random number was applied to determine the alternative out-state configuration of particles after collision in every time step. Another random number was applied to determine the distribution of oil and water particles in the polymer displacement model. Other random numbers were generated to determine whether a particle adsorbed in the model of polymer displacement. In FORTRAN, random numbers are generated randomly for a coordinate point at the same sequence. This means that the same sequence of random numbers will be reintroduced when the computer program is rerun. Division of the medium into some regions will change the coordinate of the regions; hence, it affects the random numbers. The differences of the results between undivided and divided porous media were probably due to random numbers. In general, the simulation results of the divided porous media give good agreement with those of the whole porous medium.

4.2 Communicating Flow

The simulation results of porosity, initial water and residual oil saturations of the whole medium of 5×4 cm^2, two regions of 5×2 cm^2 and four regions of 5×1 cm^2 are given in Table 2. As in the non-communicating case, the division of the porous medium into multiple regions did not cause any change in the estimated average porosity and less than 0.0651% difference in the initial water saturation estimation of the whole model. However higher differences were observed in the residual oil saturation estimation which gave up to 3.618 % error. Nevertheless, the differences were still within acceptable levels.

Figure 6 shows the displacement efficiency after one pore volume polymer was injected. Similarly, the differences of the whole medium curve to two and to four regions were small *i.e.* 0.8449% and 2.2147%, respectively.

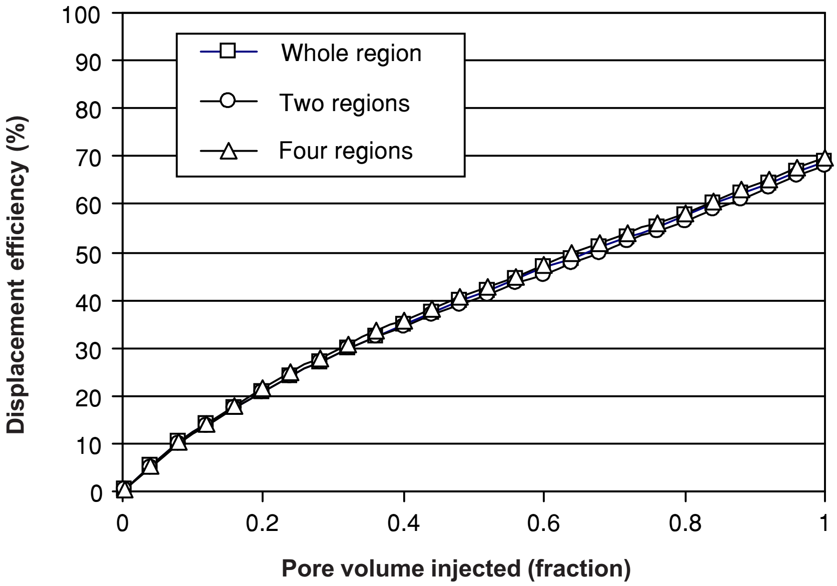

Figure 6 Comparisons of displacement efficiency for communicating flow system

The computation time of various processor numbers for polymer displacement in porous media with communicating channels is depicted in Figure 5. As expected, the addition of processor reduced computation time. Each processor computed a small task and so the more processors there were, less time was needed to complete the simulation. Hence, optimum condition could be planned to achieve the expected computation time.

The performance of a cluster system is measured by speedup and parallel efficiency. Ideally, the speedup for two processors is two or the parallel efficiency of the cluster is equal to one. However, because of communication load, the speedup was less than the number of processor or the parallel efficiency of the cluster was less than one (Table 3). The parallel efficiency decreased as the number of processor increased. The communication time for instructions from master node depended on the number of slave nodes. Therefore, instructions from the master took longer time. In addition, the simulation took longer time for synchronization process, since the slave processors took different times to perform their tasks.

The differences of speedup between undivided and divided porous media for this case (Section 4.2) could be due to both the random numbers and the communicating time between processors. Meanwhile, the differences of the speedup between whole and divided porous media for the previous case (Section 4.1) are only resulted from the random numbers.

Overall, whether the system was communicating or non-communicating, the differences in the estimated parameters were less than 3.62% and the speedup was at least 93% from the ideal speedup.

5.0 CONCLUSIONS

Based on the simulation results and analyses shown above, several conclusions are made as follows:

(1) Comparisons of the simulation results of rock and fluid properties show that the parallel simulations of LGA models gave good agreement with the single simulations, where maximum difference of all parameters for non-communicating and communicating flow systems was 3.62%.

(2) Parallel computation reduced computation time. The performance of non-communicating flow system met the ideal condition where the efficiency was equal to one and the speedup was equal to the processor number.

(3) The performance of communicating flow system i.e. speedup and parallel efficiency reduced as processor number increased but to no less than 93%. The performance of communicating flow system estimated by LGA was lower than that of non-communicating flow system.

(4) The differences and non-ideal behavior were most likely due to the random number generator and communication time.

ACKNOWLEDGEMENTS

We gratefully acknowledge the Government of Malaysia on the Intensification of Research Priority Areas (IRPA) Vote 72027 programme and Universiti Teknologi Malaysia that supported this research. M.T.F would also like to thank Mr. Mochamad Tito Julianto from Institut Pertanian Bogor, who helped conduct the parallel simulation experiments.

NOMENCLATURE

d	= mean density per link, gr cm^{-3}
h	= width of region, cm
k_r	= relative permeability
S	= saturation, fraction
r	= site position
t	= time, second
μ	= viscosity, cp
ϕ	= porosity, fraction
ρ	= density, gr cm^{-3}

Subscripts

i	= initial condition
o	= oil
p	= polymer

r	= region
w	= water
v	= void space
s	= solid material

REFERENCES

[1] Frisch, U., B. Hasslacher, and Y. Pomeau. 1986. Lattice-Gas Automata for the Navier-Stokes Equation. *Physical Review Letters*. 56(14): 1505.

[2] Wolfram, S. 1986. Cellular Automaton Fluids: Basic Theory. *Journal of Statistical Physics*. 45: 471 – 526.

[3] Gunstensen, A. K., D. H. Rothman, S. Zaleski, and G. Zanetti. 1991. Lattice Boltzmann Model of Immiscible Fluids. *Physical Review A*. 43(8): 4320 – 4327.

[4] Waite, M. E. 1998. Fluid Flow in Discrete Fractures: an Experimental and Lattice Gas Automata Modeling Study. Ph.D Thesis. The University of Colorado.

[5] Balasubramanian, K., F. Hayot, and W. F. Saam. 1987. Darcy's Law From Lattice-Gas Hydrodynamics. *Physical Review A*. 36(5): 2248 – 2253.

[6] Rothman, D. H. 1988. Cellular-Automaton Fluids: a Model for Flow in Porous Media. *Geophysics*. 509 – 518.

[7] Burgess, D., F. Hayot, and W. F. Saam. 1988. Model for Surface Tension in Lattice-Gas Hydrodynamics. *Physical Review A*. 38(7): 3589 – 3592.

[8] Clavin, P., P. Lallemand, Y. Pomeau, and G. Searby. 1988. Simulation of Free Boundaries in Flow Systems by Lattice Gas Model. *Journal of Fluid Mechanics*. 188: 437 – 464.

[9] Rothman, D. H., and J. L. Keller. 1988. Immiscible Cellular-Automaton Fluids. *Journal of Statistical Physics*. 52: 1119 – 1127.

[10] Somers, J. A., and P. C. Rem. 1991. Analysis of Surface Tension in Two Phase Lattice Gases. *Physica D*. 47: 39 – 46.

[11] Chen, S., G. D. Doolen, K. Eggert, D. Grunau, and E. Y. Loh. 1991. Local Lattice-Gas Model for Immiscible Fluids. *Physical Review A*. 43(12): 7053 – 7056.

[12] Malevanets, A., and R. Kapral. 1998. Continuous-Velocity Lattice Gas Model for Fluid Flow. *Europhys. Lett.* 44: 552 – 558.

[13] Hashimoto, Y., Y. Chen, and H. Ohashi. 2000. Immiscible Real-Coded Lattice Gas. *Computer Physics Communications*. 129: 56 – 62.

[14] Fathaddin, M. T., and M. Awang. 2004. Lattice Gas Automata Simulation of Adsorption Process of Polymer in Porous Media. *International Journal of Engineering*. 17(4): 329 – 338.

[15] Fathaddin, M. T., M. Awang, and B. Satiyawira. 2005. Estimation of Mobility Ratio and Adsorption Effects in Polymer Displacement Using Lattice Gas Automata. Symposium X Indonesian Petroleum Engineers Association. 156 – 164.

[16] Geist, A., A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, and V. Sunderam. 1994. PVM: Parallel Virtual Machine - A Users' Guide and Tutorial for Networked Parallel Computing. The Massachusetts Institute of Technology Press.

[17] Starr, F. W., S. T. Harrington, B. M. Boghosian, and H. E. Stanley. 1996. Interface Roughening in a Hydrodynamic Lattice-Gas Model with Surfactant. *Physical Review Letters*. 77: 3363 – 3366.

[18] Buick, J. M. 1997. Lattice Boltzmann Methods in Interfacial Wave Modelling. Ph.D Thesis. The University of Edinburgh.

[19] Geist, A., A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, and V. Sunderam. 1994. PVM 3 User's Guide and Reference Manual. The Applied Mathematical Sciences Research Program of the Office of Energy Research U.S. Department of Energy. The National Science Foundation and the State of Tennessee.

[20] Saad, N., A. S. Cullick, and M. M. Honarpour. 1995. Effective Relative Permeability in Scale-Up and Simulation. *Society of Petroleum Engineers*. SPE 29592: 451 – 463.