# Evolving Cellular Automata to Model Fluid Flow in Porous Media

Tina Yu
*ChevronTexaco Information Technology Company*
*6001 Bollinger Canyon Road*
*San Ramon, CA 94583*
*tiyu@chevrontexaco.com*

Seong Lee
*ChevronTexaco Exploration & Production Technology Company*
*6001 Bollinger Canyon Road*
*San Ramon, CA 94583*
*seon@chevrontexaco.com*

## Abstract

*Fluid flow in porous media is a dynamic process that is traditionally modeled using PDE (Partial Differential Equations). In this approach, physical properties related to fluid flow are inferred from rock sample data. However, due to the limitations posed in the sample data (sparseness and noise), this method often yields inaccurate results. Consequently, production information is normally used to improve the accuracy of property estimation. This style of modeling is equivalent to solving inverse problems.*

*We propose using a Genetic Algorithm (GA) as an inverse method to model fluid flow in a pore network Cellular Automaton (CA). This GA evolves the CA to produce specified flow dynamic responses. We apply this method to a rock sample data set. The results are presented and discussed. Additionally, the prospect of building the pore network CA machine is discussed.*

## 1 Introduction

Reservoir characterization and simulation play an important role in oil exploration and production. During the exploration stage, various data (e.g. rock samples, seismic data) are used to characterize reservoir properties. Based on these properties, gas and oil reserves can be estimated, which in turn determines whether or not to invest resources for oil production. During the production stage, the identified reservoir characteristics are used to simulate reservoir performance. The simulation results provide useful information for effective production operations (e.g., drilling, well locations and different recovery processes). When the geological properties of a reservoir are modeled accurately, good decisions can be made to generate profitable operations. The study of reservoir characterization and simulation is therefore very important in the petroleum industry.

The simulation of reservoir performance is based on the geological properties of a reservoir. However, the current simulation methods do not always give satisfactory results due to the uncertainty of the property descriptions, which are inferred using a small amount of core data (a core is a rock sample removed from a reservoir). As a result, there exist discrepancies between the inferred and the actual reservoir properties, no matter what geostatisitical method was used to infer those properties. To improve the accuracy of the reservoir model, production data are normally used to update/correct the reservoir properties as the production progresses. With a more accurate model, better decisions can be made to optimize production operations.

Using production data to identify reservoir properties is known as an inverse problem. In this problem, the reservoir model is a black box with unknown properties. Moreover, the inputs and outputs of the model are provided by the production data (normally the input is the amount of water/steam injected at the wells and the output is the amount of oil/gas/water produced at the wells). Based on the inputs and outputs, the task is to construct the black box reservoir model (see Figure 1).

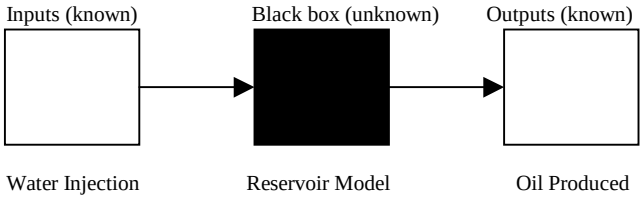


**Figure 1: Reservoir properties inverse problem.**

Inverse problems are difficult to solve for the following two reasons:

- Small variation in the input/output data may result in unbounded changes in the model estimation;
- The solution of the problem is non-unique; i.e., more than one model can satisfy the same set of input and output data.

It is therefore not uncommon to incorporate other information during the inverse process to construct a more accurate model. For example, Wen et al. [17] used pressure and breakthrough data in their inverse technique to construct a geostatistical aquifer model. Similarly, we use pressure drop information in our inverse method to model the properties of a rock core sample that exemplifies oil reservoirs.

There are many reservoir properties that have impacts on oil production, e.g. porosity, permeability and wettability. In this work, we focus on permeability, which is the primary influential factor of fluid flow in a reservoir. Moreover, we use a 3-D Cellular Automaton (CA) to represent the core sample model. The inverse task is to construct permeability distribution in the CA using the given fluid input/output data and pressure drop information. Using a Genetic Algorithm (GA) to solve this inverse problem, the CA is evolved to model the fluid flow pattern that matches the given fluid input/output data.

This approach is similar to that of Gregorio et al. [6]who applied a CA to model a complex bioremidation problem. However, we use a pore network CA to model multi-phase flow in porous media.

With the intent to model geological properties at reservoir scale, this work of modeling core scale data is a first step toward such goal. We report our investigation results and will apply the same method to more complicated reservoir scales in the future.

One major obstacle of reservoir scale modeling is the required computation time. It is not uncommon for one model simulation to take days to complete, which is unacceptable when the modeling method involves a population-based search algorithm, such as GAs. Previously, a massively parallel computer (CM5) has been used to address this issue [10]. However, a more effective approach is to build a CA machine where the simulation is done at hardware level. We will give a brief discussion of the prospect of the implementation of the CA machine at the end of the paper.

The paper is organized as follows. Section 2 provides background information. It first explains fluid dynamics in porous media and then gives a brief description of CA. Section 3 presents the pore network CA model. The CA local rules and time step calculation are also discussed. Section 4 describes the core sample modeled in this work. In Section 5, GA information is given. Section 6 analyzes the GA experimental results and Section 7 discusses related issues. Section 8 gives the prospect of hardware implementation of the pore network CA. Finally, conclusions and future work are provided in Section 9.

## 2 Background

### 2.1 Fluid Dynamics in Porous Media

*Single-phase* fluid flow (e.g. oil, gas or water) in porous media is generally described by Darcy's law [11]:

$$u = -\frac{k}{\mu}\nabla p \qquad (1)$$

Where $u$ is fluid velocity (flow rate), $k$ is permeability, $\mu$ is fluid viscosity, and $\nabla p$ is pressure gradient. For *homogenous* porous media, given $u$, $\mu$ and $\nabla p$ (which can be measured easily from a core sample in a laboratory), the permeability can be calculated using Equation (1). Permeability measures how easily a fluid can flow in porous media.

However, natural porous media (such as oil reservoir) are highly *heterogeneous*. It is not unusual for a reservoir to have permeability ($k$) that varies with several orders of magnitude between layered formations. In order to identify areas in the reservoirs that have high permeability, a large number of core sample data are required. This is not practical since core sampling is very expensive. As a result, stochastic or statistical methods are employed to estimate the permeability distribution in a reservoir, based on a limited number of core samples [9].

Using core sample data to estimate reservoir permeability distribution has the following problems:

- A core sample is normally about 2-3 cm, which is considerably smaller than the size of a reservoir (~10 km). The extrapolated results are therefore not always accurate.
- Even a core sample of 2-3 cm can contain large variability in permeability. The high heterogeneity within a core sample makes the measurement of its permeability difficult [19].

Besides heterogeneity, *multi-phase* flow modeling is another challenge in oil reservoir simulation. Oil reservoirs normally contain more than one type of fluid (oil, gas and water). Multi-phase flow is very difficult to analyze because each fluid has different physical properties and they can interact with either the rock surfaces or the other fluids. As a result, this problem becomes intractable when the first principles of physics are applied. For practical applications, a simple phenomelogical model is frequently used [11]:

$$u_i = -\frac{kk_{r,i}}{\mu_i}\nabla p_i \qquad (2)$$

In this equation, subscript $i$ denote phase (oil, gas or water) and $k_r$ is the relative permeability.

In order to apply equation 2 for multi-phase flow modeling, relative permeability information ($k_r$) is required. This is generally obtained through a fluid displacement experiment on a core sample. This method assumes that relative permeability is a function of saturation *only*. Such simplification lacks the support of rigorous proof. In fact, some experimental results have shown clearly that relative permeability depends on more than just saturation. Other variables, such as capillary force, fluid velocity, can influence relative permeability [11].

Another issue of using this extension of Darcy's law to model multi-phase flow is that it gives macroscopic information only, without any description of the actual pore structures. For microscopic analysis of multi-phase flow, other method, instead of Equation (2), is required [7].

Recently, Chen et al. [2] has calculated multi-phase flow phenomena with microscopic information for a few hundred pores scale. However, their method is not scalable easily to a larger number of pores. Various researchers therefore proposed different more scalable models. Among them, pore network model is the most promising. We therefore select this model to study multi-phase flows on core scale data. Once the method is justified, its application on reservoir scale data can be pursued.

### 2.2 Cellular Automata

A Cellular Automaton (CA) [16][18] is a spatially distributed model of N cells, each of which is in one of *k* states at time *t*. At time *t+1*, each cell updates its state following a set of rules. Moreover, the state of a cell at time *t+1* generally depends on its own state and the states of some number of neighboring cells at time *t*.

For example, Table 1(a) is a CA with 16 cells each has 2 possible states (0 and 1). Each cell has 4 neighbors (above, bellow, left and right). When applied with a majority rule, (the new state of a cell is the majority state of its current state and the states of its neighbors), the CA is updated with new sates shown in Table 1(b). Note that the CA is circulatory, hence every cell has 4 neighbors.

| | | | |
|---|---|---|---|
| 1 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 |

**(a)**

| | | | |
|---|---|---|---|
| 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |

**(b)**

**Table 1: A CA at time *t* (a) and time *t+1* (b).**

## 3 The Pore Network CA Model

In this work, the CA is a 3-D pore network model where each cell contains a pore and these pores are connected to each other with a tube (see Figure 2). Fluid flows from pores to pores through the tubes according to the laws of physics [1]. We assume fluid volume is exclusively contained in the pores and the flow friction occurs along the small capillary tube.

With those assumptions, the model can be generalized by implementing random connectivity between pores and random distribution of the pore sizes. In this paper we construct a simple network model that has regular connectivity (Cartesian with 6 neighbors). Moreover, all pores have the same size while the tube radius varies from one to another. This model hence has a heterogeneous distribution of permeability.

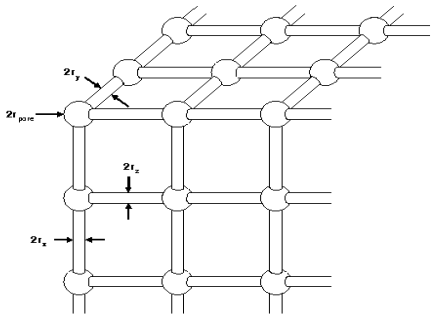

**Figure 2: A 3-D pore network model CA.**

The pore network CA is a particular kind of lattice gas CA [8][5], where each cell contains a particle and the interactions of particles are governed by simple rules of collision and propagation. However, a clee in the pore network CA contains a pore with fluid. The fluid displacement process is governed by two rules: accessibility of the invading fluid to the pore and the fluid displacement. Moreover, unlike the Boolean operation updating rules used in lattice gas CA, the rules in pore network CA muniplate real numbers (see Section 3.2).

In a sense, the granularity of pore network CA is coarse in comparsion to the lattice gas CA (porse vs. particle). The purpose of this extension of CA is to study macroscopic behavior of fluid flow in porous media (where lattice gas CA studies fluid flow in general with no consideration of the media).

### 3.1 Time-Step Calculation

Each pore has 3 possible states: (1) water, (2) oil, and (3) water and oil mixed. The time-step to update the CA states is not uniform, because the time for each pore to change its state varies, depending on its water/oil contents and flow rate. In principle, the CA should have all its states updated when any of the pores has changed its state. We have devised a sophisticated algorithm to calculate the time-step of the next state updating based on the CA's current states. The details of the algorithm are described in [10].

### 3.2 Rules for Fluid Motion and State Changes

Pore state changes are the results of encroachment of displacing fluids to the pores. The process of displacing a wetting[1] fluid with a non-wetting fluid is called imbibition (see Figure 3). Imbibition leads to the changes of fluid connectivity and flow rate. We apply three rules to update

[1] Wetting fluids have more affinity to pore surface than non-wetting ones.

pore states; two are related to fluid connectivity and one is related to flow rate.

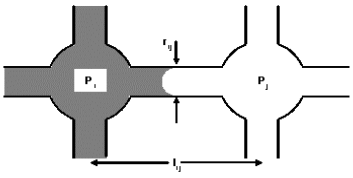

**Figure 3: Imbibition process.**

When two pores *i* and *j*, one containing wetting and the other containing non-wetting phase, the capillary force inside the *tube* is:

$$P_{t,ij} = \frac{2\sigma\cos\theta}{r_{ij}} \tag{3}$$

where $r_{ij}$ is the circular tube radius between the two pores, σ is the surface tension between two fluids and θ is their contact angle.

The capillary force inside a *pore* with wetting and non-wetting fluids is:

$$P_{p,i} = \frac{2\sigma\cos\theta}{r_{p,i}} \tag{4}$$

where $r_{p,i}$ is the effective radius of the pore.

At the initial state of an imbibition process, a non-wetting fluid occupies the tube. If there is a gravitational force due to fluid density differences, the hydrodynamic potential difference can differ depending on what fluid occupies the tube. Recall that although the fluid volume is contained in the pore, the gravitational potential difference is proportional to the distance between the two pores, which is equivalent to the tube length, $l_{ij}$. Hence, for imbibition to take place, the potential difference has to overcome the maximum barrier depending on the gravity direction and the flow direction. Therefore, the rules for the wetting fluid to flow from pore *i* to pore *j* are:

$$P_i - P_{p,i} - P_j + \rho g l_{ij} > -P_{t,ij} \tag{5}$$

$$P_i - P_{p,i} - P_j + P_{p,j} + \rho g l_{ij} > 0 \tag{6}$$

Here, *g* is gravity and ρ is fluid density, which can be either of the wetting or of non-wetting fluid, depending on which density gives a stricter criterion in Equation (5). Equation (5) specifies the potential criteria for a wetting fluid to flow through the *tube*. Equation (6) specifies the conditions for the flow of the wetting fluid between two *pores*.

When the flow criteria (Equations 5 & 6) are met, the flow rate between two pores is calculated from the pressure difference and the tube radius and length. This is based on Poiseuille law[4]:

$$Q_{\alpha,ij} = \frac{\pi r_{ij}^4}{8\mu_\alpha l_{ij}}(P_i - P_j) \tag{7}$$

where subscript α indicates the fluid phase (oil, gas or water).

In summary, a pore changes its state depending on whether it meets the fluid flow criteria (Equations 5 & 6). If imbibition takes place between two pores, their states are updated and the flow rate (Equation 7) is applied. This information (current states and flow rate) is used to calculate time-step for the next state updating.

### 3.3 Volume Conservation Principle

In this CA model, we assume the fluids are incompressible, which requires volume conservation (no volume change) for every pore. This means that the sum of the fluxes (defined in Equation 7) in all pore *i* should vanish:

$$\sum_{\alpha,j} Q_{\alpha,ij} = \sum_{\alpha,j} T_{\alpha,ij}(P_i - P_j) = 0 \tag{8}$$

Here,

$$T_{\alpha,ij} = \frac{\pi r_{ij}^4}{8\mu_\alpha l_{ij}} \tag{9}$$

Based on the principle of volume conservation, the pressure value for each pore is updated so that this principle is satisfied in every pore:

$$P_i^{new} = \frac{\sum_{\alpha,j} T_{\alpha,ij} P_j^{old}}{\sum_{\alpha,j} T_{\alpha,ij}} \tag{10}$$

## 4 The Core Sample Descriptions

We use a pore network CA with 1024 pores (32x8x4) to model a core sample. This core sample has a two-layer structure, where the top layer has its mean tube radius that is twice as big as that of the bottom layer (see Figure 4).

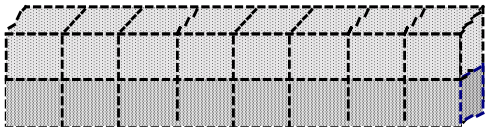

**Figure 4: a 2-layer core sample.**

As mentioned in Section 3, all pores have the same size. Hence, the modeling task is to construct the CA tubes such that it generates the give fluid flow pattern. Since it is very time consuming to model every tube that connects the 1024 pores (there are 1024 tubes and the number of possible combination is prohibitively large), we partition the core sample into 16 regions. In this way, only the mean and variance of the tube radii in each region are modeled. The actual tube radii are distributed based on these means and variances. The means and variances of the tube radii are given in Table 2. This simple test case can be extended later for more complex conditions.

| Regions | Mean tube radius | Variance |
|---|---|---|
| Top layer | 0.2 | 0.2 |
| Bottom layer | 0.1 | 0.2 |

**Table 2: The means and variances of tube radii.**

We collect the flow and pressure drop information of this core sample by conducting a fluid simulation with a fixed flow rate (q_inlet=q_outlet=1.0). Based on the simulation results, the fractional flow (*fw*) at the outlet and pressure drop between the inlet and outlet ($\Delta p$=p_inlet-p_outlet) are calculated. Fractional flow is the percentage of the inlet fluid that reaches the outlet. This value is between 0 and 1. Table 3 lists the 21 calculated *fw* and $\Delta p$ values.

## 5 Genetic Algorithms

We use a steady-state GA [4] to search for the tube radius combinations of the 16 regions that produce the fractional flow and pressure drop patterns in Table 3. This GA has 50% overlapping population; i.e., at each generation, the better half of the population remains while the worse half is replaced by newly created offspring. New offspring always make into the population, regardless of whether their fitness are better than the worse half of the original population or not.

The selection method is the traditional roulette wheel (fitness proportionate) selection. In this method, the probability of an individual to be chosen equals to the fitness of the individual divided by the sum of the finesses of all individuals in the population.

| Data Point | Fractional Flow (*fw*) | Pressure Drop ($\Delta p$) |
|---|---|---|
| 0 | 0 | 115.2633 |
| 1 | 0 | 109.3515 |
| 2 | 0 | 96.2072 |
| 3 | 0 | 80.6902 |
| 4 | 0 | 63.7544 |
| 5 | 0.541509 | 52.426 |
| 6 | 0.637602 | 45.2596 |
| 7 | 0.77412 | 41.0353 |
| 8 | 0.787548 | 38.6209 |
| 9 | 0.618791 | 38.7839 |
| 10 | 0.738425 | 36.0254 |
| 11 | 0.904544 | 34.7905 |
| 12 | 0.913995 | 34.11 |
| 13 | 0.921368 | 33.7028 |
| 14 | 0.873326 | 33.4896 |
| 15 | 0.969382 | 32.9226 |
| 16 | 0.922918 | 32.7587 |
| 17 | 0.959221 | 32.2171 |
| 18 | 0.98576 | 32.4237 |
| 19 | 0.986584 | 32.1066 |
| 20 | 0.985451 | 32.1268 |

**Table 3: Fractional flow and pressure drop data.**

The GA genome is an array of 32 real numbers. They are the tube radius mean and variance for the 16 regions. We bound the mean value to be between 0.01 and 0.3 and the variance value to be between 0 and 0.5. They are reasonable assumptions based on the real world reservoir data.

Two genetic operators are used to generate offspring: uniform crossover and Gaussian mutation. Uniform crossover picks gene values from two parents randomly to compose the offspring. Gaussian mutation changes a gene value to a new value based on a Gaussian distribution around the original value. The GA parameters are listed in Table 4.

| Parameters | Values |
|---|---|
| Population size | 50 |
| Number of generation | 50 |
| Crossover rate | 90% |
| Mutation rate | 1% |
| Replacement rate | 50% |

**Table 4: GA parameters.**

### 5.1 Fitness Functions

We made three different GA runs with slightly different assumptions and different sets of fractional flow and pressure drop information. In the first run, no extra assumption was made and the information of fraction flow and pressure drop is as described in Table 3. The fitness function for this run is defined as follow:

$$\frac{1}{N}\sum_{i}^{N}[(fw(i)-fw(i)_{meas})^2 + wt\ \left(\Delta p\ (i)-\Delta p_{meas}(i)\right)^2]$$

Where the subscript *meas* stands for measured value from the CA model and *wt* is the weight function to normalize the pressure drop measurement error. The weight function is defined as follow:

$$wt = 1/\Delta p_{meas}(0)^2$$

In the second run, the tube data for first two regions (entry point of the core sample) are specified. GA is to search for the tube radii values of the rest 14 regions (i.e. the genome size is 28 instead of 32). The fitness function for this run is the same as that as the first run.

In the third run, fractional flow (*fw*) and flow rate (*q*) for the two layers are also provided (in addition to the information listed in Table 3). They are listed in Table 5.

The fitness function for this run is defined as follow:

$$\frac{1}{N}\sum_{i}^{N}[(fw(i)-fw(i)_{meas})^2 + wt\ \left(\Delta p\ (i)-\Delta p_{meas}(i)\right)^2]$$
$$+[\left(fw_T(i)-fw_{T,meas}(i)\right)^2+\left(fw_B(i)-fw_{B,meas}(i)\right)^2]$$
$$+wt_1[\left(q_T(i)-q_{T,meas}(i)\right)^2+\left(q_B(i)-q_{B,meas}(i)\right)^2]$$

where the subscript *T* stands for top layer and *B* for bottom layer. The weight function $wt_1$ is defined as:

$$wt_1 = 4/q_{inlet}^2$$

where $q_{inlet}$ is 1.0 (see Section 4). The purpose of this weight function is to increase the importance of flow rate measure errors on the CA model. In this way, they have an equal weight as the fractional flow measure errors.

## 6 Results

The best models produced by each of the 3 GA runs are given in Tables 6-8. The model from the first run (Table 6) has its upper layer tube radii smaller than the lower layer, which is opposite to the core sample. This result is not surprising because the given flow and pressure information is from the entire core sample. No distinction can be made about the contribution of the upper and the lower layers based on these data.

| Data | fw (top) | fw(bottom) | q(top) | q(bottom) |
|---|---|---|---|---|
| 0 | 0 | 0 | 0.9469 | 0.0531 |
| 1 | 0 | 0 | 0.9449 | 0.0551 |
| 2 | 0 | 0 | 0.9446 | 0.0554 |
| 3 | 0 | 0 | 0.9445 | 0.0555 |
| 4 | 0 | 0 | 0.9442 | 0.0558 |
| 5 | 0.5597 | 0 | 0.9676 | 0.0324 |
| 6 | 0.6567 | 0 | 0.971 | 0.029 |
| 7 | 0.7921 | 0 | 0.9773 | 0.0227 |
| 8 | 0.8049 | 0 | 0.9784 | 0.0216 |
| 9 | 0.6368 | 0 | 0.9717 | 0.0283 |
| 10 | 0.7561 | 0 | 0.9766 | 0.0234 |
| 11 | 0.9196 | 0 | 0.9836 | 0.0164 |
| 12 | 0.9301 | 0 | 0.9827 | 0.0173 |
| 13 | 0.9376 | 0 | 0.9827 | 0.0173 |
| 14 | 0.8899 | 0 | 0.9814 | 0.0186 |
| 15 | 0.9837 | 0 | 0.9854 | 0.0146 |
| 16 | 0.939 | 0 | 0.9828 | 0.0172 |
| 17 | 0.9741 | 0 | 0.9847 | 0.0153 |
| 18 | 1 | 0 | 0.9858 | 0.0142 |
| 19 | 1 | 0.2149 | 0.9829 | 0.0171 |
| 20 | 1 | 0 | 0.9855 | 0.0145 |

**Table 5: Fractional flow and flow rate for two layers.**

Once the tube radii at the entry point are specified, both runs 2 and 3 produce models that have the same layer structure as that of the core sample (top layer has tube radii that are larger than those of the bottom layer). In other words, GA has identified the correct layer structure.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0.199 | 0.188 | 0.100 | 0.023 | 0.01 | 0.065 | 0.112 | 0.115 |
| 0.284 | 0.282 | 0.228 | 0.144 | 0.197 | 0.237 | 0.242 | 0.228 |

**Table 6: Best model from run 1(mean tube radii).**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0.2 | 0.181 | 0.198 | 0.227 | 0.249 | 0.175 | 0.203 | 0.191 |
| 0.1 | 0.053 | 0.182 | 0.066 | 0.091 | 0.137 | 0.061 | 0.060 |

**Table 7: Best model from run 2 (mean tube radii).**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0.2 | 0.183 | 0.194 | 0.174 | 0.253 | 0.197 | 0.275 | 0.195 |
| 0.1 | 0.032 | 0.121 | 0.043 | 0.050 | 0.101 | 0.136 | 0.101 |

**Table 8: Best model from run 3 (mean tube radii).**

The fractional flow and pressure drops generated by the 3 resulting models are presented in Figure 4. All models give flow and pressure drop patterns that are very close to that produced by the core sample. The CA model from run 3, which used extra flow information, gives the best results. This indicates that knowledge is helpful for GA to solve inverse problems.

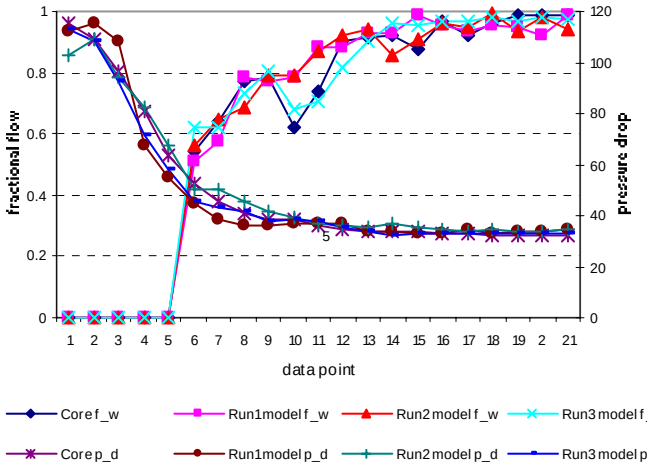


**Figure 4: Fractional flow and pressure drop results.**

## 7 Discussion

Inverse problems are usually ill posed; that is, the true answer is embedded in a class of solutions, any of which can produce the same data within the measurement error. To find the true solution, or at least an approximation to it, additional information must be brought to bear, to regularize the problem [13].

Many techniques have been applied to solve inverse problems. Among them, GA is a modern method used to solve geophysical inverse problems [12]. The results of this work show that GA is able to produce CA models that generate outputs, which are very close to the target outputs. However, none of these models matches the core structure exactly. To address the non-uniqueness issue of this inverse problem, we believe the regulation of the search space using other information (such as geostatisticial variogram) is required.

The GA population fitness information for run 3 is plotted in Figure 5. It shows that the population average fitness improves very fast during the first 5 generations. After that, the average fitness improves in a slow and steady pace. The population converges at generation 46 when average fitness is the same as the best and the worst fitness.

This fitness improvement pattern might be related to the genetic operators used in this GA. Uniform crossover seems to be aggressive in locating the high fitness area in the search space at the beginning of the run. Once the population is within such area, crossover does not have much impact (the genes are more or less similar although the population is not complete converged). At this point, Gaussian mutation (a local neighborhood search operator) gradually improves the population fitness.

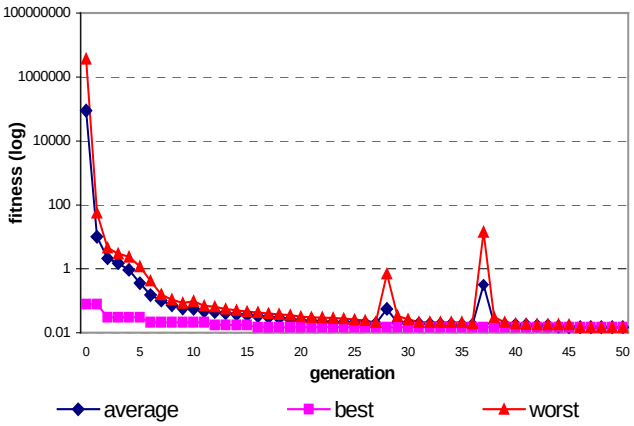


**Figure 5: Population fitness information for run 3.**

We do not know whether this is the best combination of genetic operators. In general, uniform crossover is more disruptive than one-point/two-point crossover. However, for small population size, which does not provide the necessary sampling accuracy, this disruption, gives the exploration needed for adaptive search [3]. In this problem, the size of the search space is unbounded (the 32 genes are real numbers). To get enough sampling accuracy, it needs a large population size, which is not possible because flow simulation is very time-consuming. We therefore choose uniform crossover and Gaussian mutation to work with a small population size (50).

Although this work is based on a core sample, the pore network CA can be scaled to a large number of pores for reservoir scale. However, since the simulation of flow pattern on reservoir scale is very time consuming, hardware CA will be crucial to the success of reservoir scale modeling. We therefore conducted a preliminary investigation on building the pore network CA machine and give our prospect in the following section.

## 8 A Prospect of Pore Network CA Machines

The first built CA machine is CAM, which is based on a 2-D lattices gas CA model (CAM-8 has a 3-D mesh architecture) [14]. As mentioned in Section 3, the pore network CA has different granularity from that of lattices gas CA. Moreover, the state-updating scheme used in CAM (table lookup) is not suitable for the pore network CA model. We therefore can not adapt CAM to implement our CA machines.

However, the fundamental mesh architecture of the CAM still has a direct relevancy to the pore network CA structure. Recently, Toffoli refers this type of computer architecture "programmable matter", which he believes can offer a performance gain of several orders of magnitude with respect to conventional computer on suitable applications [15]. One consideration of building such architecture is admitting physics (such as thermodynamics and quantum character) in logic design. This research is relevant to our work. We intend to follow a similar path to build our pore network CA machine.

## 9 Conclusions and Further Work

Modeling reservoir properties using inverse techniques is an important task in the petroleum industry. This work of using a pore network CA with a GA inverse method to model a sample core has produced some encouraging results.

- The CA models generate flow patterns that are very close to that of the core sample.
- The CA models have the same two layers structures as that of the core sample.

We continue this research in the following directions:

- Incorporate other geological information during the GA inverse process to improve the CA model accuracy.
- Apply the method to reservoir scale data, which has a larger size and more complex structure.
- Investigate hardware implementation of the CA machine to speed up the modeling process.

## Acknowledgments

The GA implementation is based on the GAlib genetic algorithm package, written by Matthew Wall at MIT.

## References

[1] M. Blunt and P. King, "Macroscopic Parameters form Simulations of Pore Scale Flow", *Physical Review A*, Vol. 42, pp. 4780-4787, 1990.

[2] S. Chen, K. Diemer, G. D. Doolen, K. Eggert, C. Fu, S. Gutman, B. Travis, "Lattice Gas Automata for Flow Through Porous Media", *Physica D*, 47,72-84. 1991.

[3] K. A. De Jong and W. M. Spears, "An Analysis of the Interacting Roles of Population Size and Crossover in Genetic Algorithms". In *International Workshop Parallel Problem Solving from Nature*, pages 38–47. 1990.

[4] K. A. De Jong, *An Analysis of the Behavior of a Class of Genetic Adaptive Systems*, Doctoral Thesis, Department of Computer and Communication Sciences, University of Michigan, Ann Arbor, 1975.

[5] G. D. Doolean, *Lattice Gas Methods for Partial Differential Equations*. Addison Wesley Longman, SFI Studies in the Sciences of Complexity. 1990.

[6] S. Di Gregorio, R. Serra and M. Villani, "Applying Cellular Automata to Complex Environmental Problems: The Simulation of the Biomediation of Contaminated Soils", *Theoretical Computer Sciences*, vol. 217, pp. 131-156, 1999.

[7] F.A.L. Dullien, *Porous Media: Fluid Transport and Pore Structure*, Academic Press, New York, 1979.

[8] U. Frisch, B. Hasslacher and Y. Pomeau, Lattice-Gas Automata for the Navier-Stokes Equation. *Physical Review Letters*, vol. 56, pp. 1505-1508, 1986.

[9] W.H. Isaaks and R.M. Srivastava. *An Introduction to Applied Geostatistics*, Oxford University Press, New York, 1989.

[10] S.H. Lee, L. Padmanabhan and H. Al-Sunaidi, "Simulation of Linear Displacement Experiments on Massively Parallel Computers", *SPE Journal*, vol.1, pp. 327-340, 1996.

[11] A.E. Scheidegger. *The Physics of Flow through Porous Media*, University of Toronto Press, 1974.

[12] M. Sen and P. Stoffa. *Global Optimization Methods in Geophysical Inversion*. Elsevier, 1995.

[13] A. Tarantola, *Inverse Problem Theory: Methods for Data Fitting and Model Parameter Estimation*, Elsevier, 1988.

[14] T. Toffoli and N. Margolus, *Cellular Automata Machines. A New Environment for Modeling*. MIT Press, Cambridge, MA, USA, 1987.

[15] T. Toffoli. Programmable matter methods. *Future Generation Computer Systems*, 16(2):187-201, 1999.

[16] J. Von Neumann, *Theory of Self-Reproducing Automata*, edited by A. W. Burks, University of Illinois, Urbana. 1966.

[17] X.-H. Wen, C.V. Deutsch and A.S. Cullick, "Construction of Geostatistical Aquifer Models Integrating Dynamic Flow and Tracer Data Using Inverse Technique", *J. Hydrology*, vol. 255, pp. 151-168, 2002.

[18] S. Wolfram. *Theory and Applications of Cellular Automata*. World Scientific. 1986.

[19] B. Xu, J. Kamath, Y.C. Yortsos and S.H. Lee, "Use of Pore-Network Models to Simulate Laboratory Corefloods in a Heterogeneous Carbonate Sample", *SPE Journal*, vol. 4, no. 3, pp. 179-185, 1999.