Journal of Statistical Physics, Vol. 75, Nos. 5/6, 1994

Microscopic-Based Fluid Flow Simulation of Invasion on a Two-Dimensional Lattice. II. Mobilization and Cohesion

W. G. Wilson,[1] W. G. Laidlaw,[2] and D. A. Coombe[3]

Received July 15, 1993

An algorithm for modeling secondary invasion processes in porous media is presented. Mobilization of trapped defender fluid is accomplished through interfacial interaction rules. Cohesive forces ate also included within the defender phase. A series of simulation runs are performed using two-dimensional lattices and examined to determine optimal conditions for secondary invasions that sweep the trapped defender phase from the porous medium.

KEY WORDS: Random walk; porous media; fluid mobilization; secondary recovery.

1. INTRODUCTION

In this paper we extend previous algorithms designed to simulate fluid flow invasion in porous media[1] to the more general process of defender mobilization with arbitrary defender-phase surface tension. This is not a new problem[2] and has been attacked in diverse ways, ranging from numerical solutions of hydrodynamic equations on a discrete network[3-5] to random walker approaches.[6] All of these approaches are valid; however, we are drawn to the random walk algorithm due to its "coarse-grained" philosophy which replaces detailed calculations in an inherently stochastic process with "simple" stochastic rules, yet produces spatially and temporally rich solutions. The work we present here is inspired by the

[1] Department of Biological Sciences, University of California at Santa Barbara, Santa Barbara, California 93106.

[2] Department of Chemistry University of Calgary, Calgary, Alberta, Canada T2N 1N4; laidlaw@acs.ucalgary.ca.

[3] Computer Modelling Group, 3512 33rd St. N-W, Calgary, Alberta, Canada T2L 2A6.

work of Kadanoff on the evolution of an interface,[6] but we have modified the algorithm to incorporate two additional features: bulk motion in the invader phase through the explicit inclusion of invader fluctuations, and a reverse invasion, where the invader phase can be displaced (locally) by the actions of the defender phase.

One caveat with the methodology presented here is that momentum is never conserved either explicitly or implicitly, setting the algorithm apart from the lattice-gas hydrodynamic approaches of Frisch *et al.*[7] and Rothman and Keller.[8] Consequently only systems obeying the diffusion equation, such as the Hele Shaw cell[9] and flow through porous media,[2] can be accurately represented.

2. PROBLEMS

In this section we discuss an algorithm for fluid flow through porous media (more generally, diffusive systems) that contain features of displacement, mobilization, trapping, and cohesion. In the first part mobilization rules will be explained in detail. Rules that govern cohesion are then discussed as an extension of the mobilization rules.

Our starting point is a previous work[1] where we described a microscopically based algorithm for fluid flow invasion by implementing two "species" of random walkers. The two species of walkers represented the two phases, invader and defender, and in turn could have different diffusivities (a different number of steps per unit simulation time). The interactions between the walkers at the interface of the phases were contained in "rules"; for example, the rule that governed displacement allowed the invader walkers to step onto unoccupied defender sites (displacing the defender phase), then continue as walkers in the invader phase. The resultant dynamics ranged from a stable displacement at favorable viscosity ratios (specified as the inverse diffusivity ratio) to viscous fingering at adverse viscosity ratios. Although we retain the notion of different species of walkers (with different diffusivities), the present algorithm undertakes rather significant modifications of the rules for interfacial interactions in order to improve invasion efficiency and to permit mobilization of bypassed defender fluid.

2.1. Mobilization

The goal is to mobilize defender fluid; for example, Fig. 1 shows a defender-phase blob imbedded in an invader-saturated network. During a given physical displacement process the defender will, at some time, move out of one pore (that pore immediately being filled with invader phase)

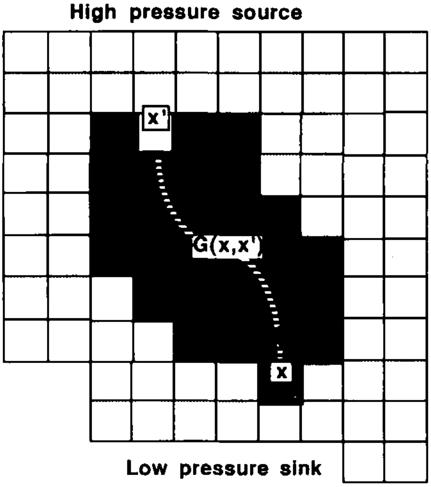

Fig. 1. Schematic diagram representing a trapped "blob" of defender fluid (in gray). The invader displaces defender and occupies a pore at x' and subsequently a displacement event occurs at x where defender displaces invader and occupies the pore.

and, a short time later, move into another pore, marking the initial net displacement of the defender blob and the beginning of its mobilization as an intact blob. If the displacement process were repeated with an identical initial defender blob geometry, we argue that the inherent stochasticity of the process will likely result in several differences from the first run. These differences include the time at which the first pore is evacuated, the specific pore that is evacuated, the time at which the second pore is entered, and the specific pore that is entered by the blob. However, if many runs were performed, there would be a correlation between the location of a high-pressure gradient at the blob's surface and the location of the first evacuated pore. Thus there is an inherent stochasticity associated with every run which, when averaged over many experimental configurations, i.e., when an ensemble average is performed, would generate the expected results.

The appendix contains the arguments, essentially those of Kadanoff, which give substance to the replacement of an ensemble of deterministic runs by the random walk. We will now discuss the rules representing this replacement.

Consider a network with fixed boundary pressures within which sits a defender blob. Invader walkers are launched, at the boundary, with a probability controlled by the boundary pressure. These walkers represent a pressure fluctuation originating from a point source and propagate with diffusivity D_i (i denotes invader). When an invader walker encounters the invader–defender interface, the invader walk may be terminated and a defender walk commenced from that point. This event is, in the first instance, conditional on the availability of an empty site (i.e., not occupied by a walker) for a newly launched defender walker. If this condition is met, then, with a "forward probability" ϕ, the invader walk is terminated and a defender walker launched from the empty site, otherwise the invader walk continues in the invader phase. A successful defender walk initiation is accompanied by the replacement of the defender phase with invader phase in the site at which the invader walk terminated. The defender walk within the defender blob is continued with diffusivity D_d until the surface of the blob is reached (here d denotes defender). At this point the defender walk must be coupled to the pressure. High invader pressure is denoted by invader occupancy and hence, if the site adjacent to the surface is occupied (i.e., the local invader pressure is too great), the defender phase is deemed unable to push into the invader. The defender walk then continues, within the defender blob, until the interface is reached elsewhere, and deposition is tested again. When the defender walker encounters a boundary site which is not occupied by an invader walker (i.e., the local pressure is low) then, with a "backward probability" β, the defender walker terminates and an invader walker is launched. A successful invader walk initiation is accompanied by the replacement of the invader phase with defender phase in the site at which the defender walk terminated. Although it is evident that the pressure at the source boundary effectively controls the overall mobilization, it is also clear that there is an inherent stochasticity in our rules that mimics the stochasticity in the experimental system and would agree with the formal calculation in the limit of many realizations.

2.2. Cohesion

Cohesion acts to minimize the interfacial area, resulting in larger, rounder defender-phase regions during the mobilization process. The algorithm that incorporates cohesion is quite simple, given the mobilization algorithm. First, lattice sites are labeled according to a majority rule of the neighbor sites' occupancy. Each site (on a two-dimensional square lattice) has four nearest neighbors (nn) and four next nearest neighbors (nnn). A given site is labeled invader-favored if three or four of the nn are invader-phase-occupied. If only two nn sites are invader-occupied and a majority of the nnn sites are invader-occupied, the site is also labeled as "invader-favored." All other configurations are labeled as "defender-favored." This labeling is completely independent of the phase that presently occupies the site. Cohesion acts to remove defender phase from invader-favored sites and subsequently place this fluid into sites that are defender-favored, but occupied by invader phase. To this end a surface walker (with the same diffusivity as the defender walker) is created, with a probability σ per unit time step, at an invader-favored but defender-occupied site. Upon creation of the surface walker the invader-favored site becomes invader-occupied. The surface walker's propagation terminates when it happens upon a defender-favored but invader-occupied site; at this point the site becomes defender-occupied and the surface walker is annihilated.

The displacement of the surface walker propagates according to a Green's function, which, in turn, is solved by a random walk of "surface walkers" as discussed in the appendix. Thus the only difference between the surface tension algorithm and the mobilization algorithm are the sources and sinks of the random walkers.

In summary, then, in the entire simulation there are three types of walkers. *Invader walkers* represent a pressure transient in the invader phase, *defender walkers* represent a pressure transient in the defender phase, and cohesive forces demand a third walker, the "*surface walkers*."

Besides the diffusivity, fundamental to each walker, there are three parameters in the algorithm controlling the evolution of the defender–invader interface. The parameter ϕ represents the probability of the "forward" process defined here as a process in which an invader walker at the interface enters a defender-filled pore. The parameter β controls the "backward" process of defenders displacing invaders, and the final parameter σ is the probability of surface tension-driven displacements. All of these are expressed as rates, or events per unit time step. The parameters ϕ and β could depend on a variety of fluid and substrate properties, but we treat them as phenomenological, independent parameters. For example, if β is set to zero, the implication is that the defender phase is never able to dislodge the invader phase from a pore. Likewise, if the surface tension parameter σ is set to zero, there are no cohesive effects in the defender fluid. As we shall see, it is only for certain parameter combinations that displacement and efficient mobilization occur.

3. RESULTS

3.1. Primary Invasions

We will first consider the algorithm for primary invasions (i.e., a situation where an invader enters a region initially *fully occupied* by defender fluid) with an adverse viscosity ratio, and we will examine how the various parameters affect the anticipated viscous fingering. To probe the effect of the surface parameter σ, runs at values of σ from zero to $\sigma = 0.8$ were carried out with, in each case, the forward probability held at $\phi = 0.4$ and the backward probability held at $\beta = 0.0$. The effect of the backward probability was tested for values of $\beta = 0.0$ from zero to $\beta = 0.1$ while holding the forward parameter at $\phi = 0.4$ and the surface parameter at $\sigma = 0.2$. In all these cases the invader to defender diffusivity ratio was set to 10:1. The runs were performed on a 128 × 128 square lattice with periodic boundaries on the sides, with the source of invader walkers maintained at 50% occupancy and the sink held at 0% occupancy. Whenever defender or surface walkers touched the source or sink they were removed (0% occupancy). The general result (such as depicted in Fig. 3a) was, as the adverse viscosity ratio would suggest, strongly fingered with about 35% residual defender. The details of the residual defender distribution did depend on the parameters σ and β. Increasing the surface probability σ increased the finger width, whereas increasing β, the defender phase's ability to displace the invader, resulted in the shattering of the fingers. Since adjusting the values of σ and β alone did not materially affect the overall mobilization, the role of the forward probability ϕ was also investigated.

3.2. Blob Mobilization

We investigated this question using a system initialized with a single, rectangular defender "blob" imbedded in an invader-saturated network (such trapped blobs constitute much of the 35% residual). The results for four cases are illustrated in Fig. 2. The forward probability was set at

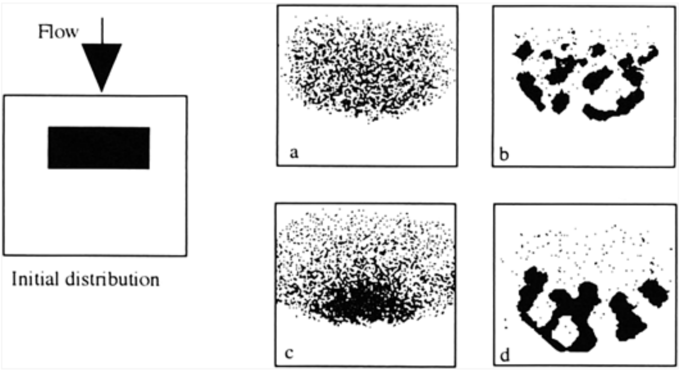

Fig. 2. Blob mobilization for various parameter combinations. All runs are performed at an adverse diffusivity ratio of 10:1 and forward probability $\phi = 0.1$. The backward (β) and surface probabilities (σ) are (a) $\beta = 0.05$ and $\sigma = 0.01$, (b) $\beta = 0.05$ and $\sigma = 1.0$, (c) $\beta = 0.5$ and $\sigma = 0.01$, and (d) $\beta = 0.5$ and $\sigma = 1.0$.

$\phi = 0.1$ with (a) $\beta = 0.05$ and $\sigma = 0.01$, (b) $\beta = 0.05$ and $\sigma = 1.0$, (c) $\beta = 0.5$ and $\sigma = 0.01$, and (d) $\beta = 0.5$ and $\sigma = 1.0$. All runs were again performed at an adverse diffusivity ratio of 10:1. All runs performed at low surface tensions result in a very shattered distribution of defender fluid that remains largely immobilized, while large values facilitate mobilization. Having the backward probability greater than the forward probability also promotes blob mobilization.

The optimal parameter values can be recognized as dependent, in part, on the invader and defender diffusivities. If a disconnected blob is to move downstream as a coherent entity, the leading edge must have the same speed as the trailing edge, otherwise the blob is unstable. The speed of the trailing edge is determined by the product of the invader walker density w_i, the invader diffusivity D_i, the forward probability, and the defender hole density h_d, or $w_i D_i \phi h_d$. Similarly, the speed of the leading edge is given by $w_d D_d \beta h_i$. Setting these speeds equal gives

$$\frac{\beta}{\phi} = \frac{D_i}{D_d} \frac{w_i h_d}{w_d h_i}$$

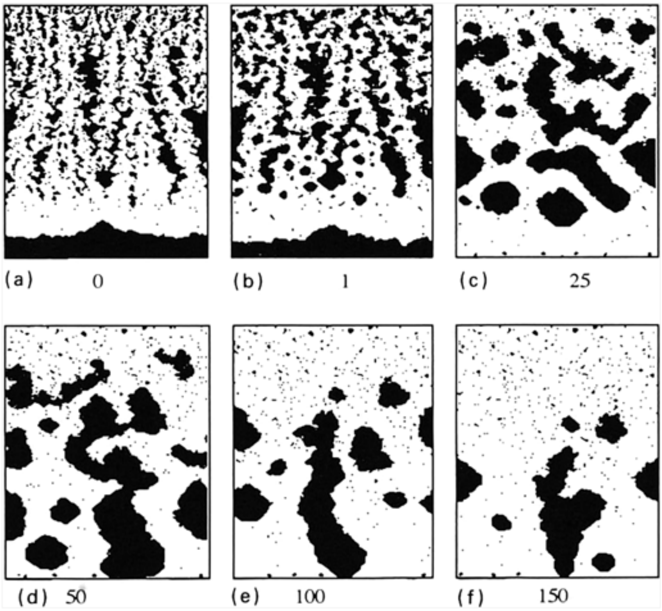

Fig. 3. Snapshots in time (units of $K = 1000$ simulation steps) of the secondary invasion process. The primary invasion [panel (a)] was performed with an adverse diffusivity ratio of 10:1, $\phi = 0.4$, $\beta = 0.2$, and $\sigma = 0.2$, giving a badly fingered displacement. A second invader phase, miscible with the first, and having $\phi = 0.1$, $\sigma = 1.0$, and $\beta = 0.5$, is brought to the source line and the secondary invasion commences.

Thus, the backward to forward parameter ratio is proportional to the invader to defender diffusivity ratio, in our case 10:1.

3.3. Secondary Invasions

The ultimate goal of a secondary invasion is the mobilization of trapped fluid resulting from a less-than-optimal primary invasion (cf. Section 3.2). The result of such a less-than-optimal primary invasion is illustrated in Fig. 3a for a 128×200 lattice. This run was obtained for an adverse diffusivity ratio of 10:1 (with $\phi = 0.4$, $\beta = 0.2$, and $\sigma = 0.2$) and gives a badly fingered displacement. As a result, about 35% of the defender (black) is still in place. The configuration of Fig. 3a is now taken as the starting configuration for a secondary invasion with a miscible second phase having $\phi = 0.1$, $\sigma = 1.0$, and $\beta = 0.5$ (the parameters used in Fig. 2d) placed at the source. Configurations after $1K$, $25K$, $50K$, $100K$, and $150K$ ($K = 1000$) time steps are shown in Figs. 3b–3f, respectively. As illustrated in Fig. 3b, after only 1000 time steps the secondary invader has swept through the entire network with significant coalescence of the defender fluid. After $25K$ time steps further coalescence has taken place, and after $50K$ time steps large blobs of defender are moving downstream. After $150K$ time steps much of the defender has been mobilized and is in the process of being moved off the grid.

Comparison of the first and last panels of Fig. 3 shows that these parameter values produce a very effective secondary flood. The increased cohesion is particularly crucial in producing these results. In addition, patience is required! For example, the primary invasion required approximately $4K$ time steps, while it takes roughly $200K$ time steps to remove the residual defender fluid.

4. SUMMARY

We have discussed an algorithm useful in the simulation of hydrodynamic and other processes where momentum conservation is not applicable. Our motivation arises from flow through porous media, which includes such diverse applications as environmental remediation and enhanced oil recovery. Since momentum is not conserved, problems can be framed as solutions to Laplace's equation subject to the appropriate boundary conditions. Random walks behave according to Laplace's equation, hence a connection between random walk algorithms and flow through porous media problems is attainable. The algorithm discussed here is based on using integral equations as propagators of pressure fluctuations, following the work of Kadanoff.[6] We have extended Kadanoff's algorithm to incorporate two additional features: bulk motion in the invader phase through the explicit inclusion of invader fluctuations, and the incorporation of reverse invasion, where the invader phase can be displaced (locally) by the actions of the defender phase. The algorithm presented here implements three random walkers, with three independent parameters that control interaction rates at the invader–defender interface. The three include invader, defender, and surface walkers. Invader fluid displacing defender fluid is controlled by the forward probability, the reverse is controlled by the backward probability, and defender cohesion is controlled by the surface probability.

We have tested the simulation for both primary and secondary invasion processes as well as blob mobilization. In the case of primary invasion, the algorithm produced the appropriate range of behavior from viscous fingering to stable displacement. However, in the case of high defender cohesion the fingers were thickened, being much less fractal in character, and in the case of high reverse invasion rates the fingers shattered drastically.

When we applied the algorithm to blob mobilization, we observed that the best mobilization occurred with high surface probability. Otherwise the blob shattered rather drastically, resulting in little downstream movement. More importantly, we found that mobilization is enhanced when the forward to backward probability ratio is proportional to the invader to defender diffusivity ratio and is accompanied by a high surface cohesion parameter.

We then performed a secondary invasion simulation using the optimal set of run parameters found in the blob mobilization tests and used the residual defender configuration from a primary invasion as a starting condition. Excellent recovery of the residual defender fluid was achieved, but at the expense of a very long flood time when compared with the primary invasion times.

APPENDIX

The usual model for flow of a bulk fluid is simply $\mathbf{v} = -\nabla p$ and for steady flow of incompressible fluids a Laplace equation for pressure, $\nabla^2 p(\mathbf{x}) = 0$, is also satisfied. Unfortunately neither equation need be true at a boundary and appropriate statements are required for each situation. We are particularly interested in the displacement of an interface between two fluids due to the appearance of a pressure fluctuation at the surface. For example, a pressure fluctuation may result in an intrusion as shown at the point $\mathbf{x}'$ in Fig. 1, with the blob distending, at some later time, into a pore at $\mathbf{x}$.

Rather than solve Laplace's equation subject to the boundary conditions directly, we could instead solve Laplace's equation for the Green's function $\nabla^2 G(\mathbf{x}, \mathbf{x}') = \delta(\mathbf{x} - \mathbf{x}')$, where $\mathbf{x}$ ranges over the blob's interior and surface, whereas $\mathbf{x}'$ only ranges over the surface (since there are no interior pressure sources). The Green's function describes the pressure at a point $\mathbf{x}$ due to a unit pressure source at the surface point $\mathbf{x}'$. The net internal pressures due to the imposed external pressures on the blob are then[10]

$$P(\mathbf{x}) = \int_{\text{surface}} P(\mathbf{x}') \frac{\partial G(\mathbf{x}, \mathbf{x}')}{\partial n'} \, da' \tag{A1}$$

Given the pressure everywhere on the blob's surface, the pressure everywhere inside the blob is obtainable if we can solve for the Green's function.[6]

It is well known from examining the discrete version of Laplace's equation,

$$\begin{aligned} G(x, y, t+1) = \tfrac{1}{4}[&G(x+1, y, t) + G(x-1, y, t) \\ &+ G(x, y+1, t) + G(x, y-1, t)] \end{aligned} \tag{A2}$$

which also governs the update of occupancy probabilities of a random walk, that a random walk solves Laplace's equation[11,12] for an arbitrary blob geometry. Equation (A2) can thus be used to solve for the pressure fields directly at steady state with the resultant pressure at each location being consistent with the applied boundary conditions and Laplace's equation.

Alternatively, a unit occupancy probability could be placed at the surface point $\mathbf{x}'$ and Eq. (A2) solved for the value of the Green's function as a function of $\mathbf{x}$, obtaining the final pressures using Eq. (A1). This is accomplished by a random walker through the following algorithm. Start a random walker at $\mathbf{x}'$ and let it walk over the network. Whenever it steps on an interior site $\mathbf{x}$, increment a counter for that site by one, and whenever the walker steps onto a boundary site, terminate the walk and remove the walker. This latter condition enforces the boundary condition that the Green's function goes to zero at the boundary except at the point source. The occupancy counter for each site, after averaging over many walkers, is proportional to $G(\mathbf{x}, \mathbf{x}')$. Kadanoff argues that repeating this process for all surface sites $\mathbf{x}'$ generates the entire Green's function.[6] The pressures could then be calculated, using Eq. (A1), to redistribute small bits of the blob.

ACKNOWLEDGMENTS

Support in the form of an operating grant from the Natural Sciences and Engineering Research Council of Canada is gratefully acknowledged, as is a grant from the Alberta Oil Sands Technology Research Authority.

REFERENCES

1. W. G. Wilson and W. G. Laidlaw, *J. Stat. Phys.* **66**:1165 (1992).
2. C. M. Marle, *Multiphase Flow in Porous Media* (Gulf Publishing Co., 1981); M. Sahimi, *Rev. Mod. Phys.* **65**:1393 (1993).
3. A. C. Payatakes, *J. Fluid Mech.* **164**:305 (1986).
4. R. Lenormand and C. Zarcone, *Phys. Rev. Lett.* **54**:2226 (1985).
5. R. Lenormand, E. Touboul, and C. Zarcone, *J. Fluid Mech.* **189**:165 (1988).
6. L. P. Kadanoff, *J. Stat. Phys.* **39**:267 (1985).
7. U. Frisch, B. Hasslacher, and Y. Pomeau, *Phys. Rev. Lett.* **56**:1505 (1986).
8. D. H. Rothman and J. M. Keller, *J. Stat. Phys.* **52**:1119 (1988).
9. J. H. S. Hele Shaw, *Nature* **58**:34 (1898).
10. J. D. Jackson, *Classical Electrodynamics*, 2nd ed. (Wiley, New York, 1975).
11. T. A. Witten and L. M. Sander, *Phys. Rev. Lett.* **47**:1400 (1981).
12. L. Patterson, *Phys. Rev. Lett.* **52**:1621 (1984).

Communicated by D. Stauffer