Annals of University of Craiova, Math. Comp. Sci. Ser.
Volume 36(2), 2009, Pages 35–41
ISSN: 1223-6934

A Particle Cellular Automata Model for Fluid Simulations

Costin-Radu Boldea

Abstract. A new cellular-automaton model for fluid dynamics is introduced in this paper, that focus on discrete models based on point particles moving on a lattice in order to mimic a fully molecular dynamics. The CA model uses an easily implementable, deterministic pair of interaction rules. Therefore, we combine the advantage of the low computational cost of CA and its ability to mimic the realistic fluid dynamics to develop a new animating framework for computer graphics applications.

2000 Mathematics Subject Classification. Primary 68Q80; Secondary 37B15.
Key words and phrases. Cellular automata, Lattice gas models, Navier-Stokes equation.

1. Introduction

A majority of fluid animation methods in computer graphics use 2D/3D mesh based approaches that are mathematically motivated by the Eulerian methods of Finite Element (FE) and Finite Difference (FD), in conjunction with Navier-Stokes equations of fluids [5]. These works are based on a top down viewpoint of the nature: the fluid is considered as a continuous system subjected to Newton's and conservation Laws as well as state equations connecting the macroscopic variables of pressure P, density ρ and temperature T.

In the past decades, a new paradigm of simulation imposed the use Cellular Automata: ultra-discrete models at the place of the complex equations ([2],[6]). A cellular automaton is a large array of cells, like the squares of a checkerboard or the hexagons of a honeycomb, which can be projected onto a computer screen. On this lattice, dots hop from cell to cell, colliding and recoiling according to a few simple rules programmed into the computer. These are discrete models based on point particles that move on a lattice, according to suitable and simple rules in order to mimic a fully molecular dynamics. Particles can only move along the edges of the lattice and their interactions are based on simple collision rules. There is an exclusion principle that limits to one the number of particles that enter a given site (lattice node) in a given direction of motion. Such framework needs low computational resources for both the memory allocation and the computation itself.

In this paper we focus on fluid modeling through Lattice Gas Cellular Automata (LGCA) for computer graphics applications. Specifically we take a special LCGA, introduced by Frisch, Hasslacher and Pomeau, known as FHP model, and show its capabilities for computer graphics applications. By Chapman-Enskog expansion, a known multiscale technique in this area, it can be demonstrated that the Navier-Stokes model can be reproduced by FHP technique. However, there is no need to solve Partial Differential Equations (PDEs) to obtain a high level of description. Therefore, we combine the advantage of the low computational cost of LGCA and its ability to mimic the realistic fluid dynamics to develop a new animating framework

for computer graphics applications. Up to our knowledge, there are no references using FHP for fluid animation in Computer Graphics. In this work, we discuss the theoretical elements of our proposal and present some experimental results.

The paper is organized as follows. Section 2 offer a review of CFD for fluid animation. Section 3 describes the FHP model its multiscale analysis. The experimental results are presented on section 4. Conclusions are given on Section 5.

2. Navier-Stokes for Fluid Animation

The majority fluid models in computer graphics follow the Eulerian formulation of fluid mechanics; that is, the fluid is considered as a continuous system subjected to Newton's and conservation Laws as well as state equations connecting the macroscopic variables that define the thermodynamic state of the fluid: pressure P, density ρ and temperature T.

So, the mass conservation, also called continuity equation, is given by [5]:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0 \tag{1}$$

The linear momentum conservation equation, also called Navier-Stokes, can be obtained by applying the third Newton's Law to a volume element dV of fluid. It can be written as [5].:

$$\rho \left(\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} \right) = -\nabla P + \mathbf{F} + \mu \left(\nabla^2 \vec{u} + \frac{1}{3} \nabla \left(\nabla \cdot \vec{u} \right) \right) \tag{2}$$

where $\mathbf{F}$ is an external force field and μ is the viscosity of the fluid. Besides, the equation $\nabla \cdot \vec{u} = 0$ must be added to model incompressible fluids. Thus, if we combine this equations with expression (2) we obtain the Navier-Stokes equations for incompressible fluids (water, for example):

$$\rho \left(\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} \right) = -\nabla P + \mathbf{F} + \mu \nabla^2 \vec{u}, \tag{3}$$

$$\nabla \cdot \vec{u} = 0. \tag{4}$$

Also, we need an additional equation for the pressure field. This is a state equation which ties together all of the conservation equations for continuum fluid dynamics and must be chosen to model the appropriate fluid (*i.e.* compressible or incompressible). In the case of liquids, the pressure P is temperature insensitive and can be approximated by $P = P(\rho)$. Muller proposed an expression that have been used for fluid animation also [8]:

$$P = c^2 \rho \tag{5}$$

where c is the speed of sound in this fluid.

Equations (3)-(5) need initial conditions $(\rho(t = 0, x, y, z), \vec{u}(t = 0, x, y, z))$. Besides, in practice, fluid domain is a closed subset of the Euclidean space and thus the behavior of the fluid in the domain boundary - *boundary conditions* - must be explicitly given. For a fixed rigid surface S, one usual model is the no-sleep boundary condition that can be written as:

$$\vec{u}|_S = 0. \tag{6}$$

Also, numerical methods should be used to perform the computational simulation of the fluid because the fluid equations in general do not have analytical solution. Finite

Element (FE) and Finite Difference (FD) are known approaches in this field. Recently, the Lagrangian *Method of Characteristics* and the meshfree methods of *Smoothed Particle Hydrodynamics* (**SPH**) [8] and *Moving-Particle Semi-Implicit* (MPS) [9] have been also applied.

If the fluid is temperature sensitive, then an energy conservation law should be applied. The model comprises equations (3),(4),(6) as well as the following equation for temperature change and the *buoyant* force, respectively:

$$\frac{\partial T}{\partial t} = \lambda \nabla^2 T - \nabla \cdot (T\vec{u}), \tag{7}$$

$$\mathbf{F} = -\beta g (T_0 - T), \tag{8}$$

where λ is the diffusion coefficient, T_0 is a reference temperature and β is the coefficient of thermal expansion. The numerical method used in [5] is Finite Difference. This work can reproduce a hot gas behavior with some realism but has the limitation that the integration time step is constrained to:

$$\Delta t < \frac{h}{\|\vec{u}\|}, \tag{9}$$

where h is the mesh resolution. Besides, the restriction of equation (4) is not suitable for a compressible system like a gas.

For Computer Graphics applications, such approach is explored in [7] for real-time simulation and animation of phenomena involving convection, reaction-diffusion, and boiling. An extension of cellular automata known as the coupled map lattice (CML) is used for simulation. CML represents the state of a dynamic system as continuous values on a discrete lattice. In [7] the lattice values are stored in a texture, and pixel-level programming are used to implement simple next-state computations on lattice nodes and their neighbors. However, Navier-Stokes models are not considered and CML still uses continuous values for representations. That is also the case of Lattice Boltzmann models [4]. In this paper we propose the application of an even more simples model, the FHP one, for fluid simulation. It will be demonstrated how Navier-Stoke models can be reproduced by this method. FHP is described in the next section.

3. FHP and Navier-Stokes

The FHP was introduced by Frisch, Hasslacher and Pomeau [3] in 1986 and is a model of a two-dimensional fluid. It can be seen as an abstraction, at a microscopic scale, of a fluid. The FHP model describes the motion of particles traveling in a discrete space and colliding with each other. The space is discretized in a hexagonal lattice.

The microdynamics of FHP is given in terms of Boolean variables describing the occupation numbers at each site of the lattice and at each time step (i.e. the presence or the absence of a fluid particle). The FHP particles move in discrete time steps, with a velocity of constant modulus, pointing along one of the six directions of the lattice. The dynamics is such that no more than one particle enters the same site at the same time with the same velocity. This restriction is the *exclusion principle*; it ensures that six Boolean variables at each lattice site are always enough to represent the microdynamics.

In the absence of collisions, the particles would move in straight lines, along the direction specified by their velocity vector. The velocity modulus is such that, in a time step, each particle travels one lattice spacing and reaches a nearest-neighbor site.

In order to conserve the number of particles and the momentum during each interaction, only a few configurations lead to a non-trivial collision (i.e. a collision in which the directions of motion have changed). When exactly two particles enter the same site with opposite velocities, both of them are deflected by 60 degrees so that the output of the collision is still a zero momentum configuration with two particles. When exactly three particles collide with an angle of 120 degrees between each other, they bounce back to where they come from (so that the momentum after the collision is zero, as it was before the collision). Both two- and three-body collisions are necessary to avoid extra conservation laws. Several variants of the FHP model exist in the literature [1], including some with rest particles like models FHP-II and FHP-III. For all other configurations no collision occurs and the particles go through as if they were transparent to each other.

The full microdynamics of the FHP model can be expressed by evolution equations for the occupation numbers defined as the number, $n_i(\vec{r},t)$, of particle entering site $\vec{r}$ at time t with a velocity pointing along direction $\vec{c}_i$, where $i = 1, 2, \ldots, 6$ labels the six lattice directions. The numbers n_i can be 0 or 1.

We also define the time step as Δ_t and the lattice spacing as Δ_r. Thus, the six possible velocities $\vec{v}_i$ of the particles are related to their directions of motion by

$$\vec{v}_i = \frac{\Delta_r}{\Delta_t}\vec{c}_i. \tag{10}$$

Without interactions between particles, the evolution equations for the n_i would be given by

$$n_i(\vec{r}+\Delta_r\vec{c}_i, t+\Delta_t) = n_i(\vec{r},t) \tag{11}$$

which express that a particle entering site $\vec{r}$ with velocity along $\vec{c}_i$ will continue in a straight line so that, at next time step, it will enter site $\vec{r}+\Delta_r\vec{c}_i$ with the same direction of motion. However, due to collisions, a particle can be removed from its original direction or another one can be deflected into direction $\vec{c}_i$.

For instance, if only n_i and n_{i+3} are 1 at site $\vec{r}$, a collision occurs and the particle traveling with velocity $\vec{v}_i$ will then move with either velocity $\vec{v}_{i-1}$ or $\vec{v}_{i+1}$, where $i = 1, 2, \ldots, 6$. The quantity

$$D_i = n_i n_{i+3}(1-n_{i+1})(1-n_{i+2})(1-n_{i+4})(1-n_{i+5}). \tag{12}$$

indicates, when $D_i = 1$ that such a collision will take place. Therefore $n_i - D_i$ is the number of particles left in direction $\vec{c}_i$ due to a two-particle collision along this direction.

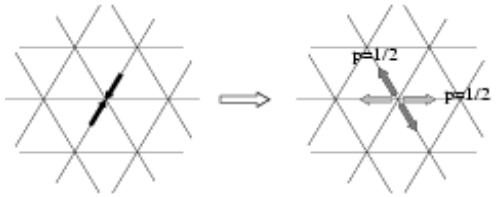

Figura 1: The two-body collision in the FHP.

Now, when $n_i = 0$, a new particle can appear in direction $\vec{c}_i$, as the result of a collision between n_{i+1} and n_{i+4} or a collision between n_{i-1} e n_{i+2}. It is convenient to introduce a random Boolean variable $q(\vec{r},t)$, which decides whether the particles are

deflected to the right ($q = 1$) or to the left ($q = 0$), when a two-body collision takes place. Therefore, the number of particle created in direction $\vec{c}_i$ is

$$qD_{i-1} + (1-q)\, D_{i+1}. \tag{13}$$

Particles can also be deflected into (or removed from) direction $\vec{c}_i$ because of a three-body collision. The quantity which express the occurrence of a three-body collision with particles n_i, n_{i+2} and n_{i+4} is

$$T_i = n_i n_{i+2} n_{i+4} \left(1 - n_{i+1}\right) \left(1 - n_{i+3}\right) \left(1 - n_{i+5}\right) \tag{14}$$

As before, the result of a three-body collision is to modify the number of particles in direction $\vec{c}_i$ as

$$n_i - T_i + T_{i+3}, \tag{15}$$

Thus, according to our collision rules, the microdynamics of a LGCA is written as

$$n_i \left(\vec{r} + \Delta_r \vec{c}_i, t + \Delta_t\right) = n_i \left(\vec{r}, t\right) + \Omega_i \left(n \left(\vec{r}, t\right)\right) \tag{16}$$

where Ω_i is called the collision term.

For the FHP model, Ω_i is defined so as to reproduce the collisions, that is

$$\Omega_i = -D_i + qD_{i-1} + (1-q)\, D_{i+1} - T_i + T_{i+3}. \tag{17}$$

Using the full expression for D_i and T_i, given by the Equations (12)-(14), we obtain,

$$\begin{aligned} &\Omega_i \\ &= -n_i n_{i+2} n_{i+4} \left(1 - n_{i+1}\right) \left(1 - n_{i+3}\right) \left(1 - n_{i+5}\right) \\ &+ n_{i+1} n_{i+3} n_{i+5} \left(1 - n_i\right) \left(1 - n_{i+2}\right) \left(1 - n_{i+4}\right) \\ &- n_i n_{i+3} \left(1 - n_{i+1}\right) \left(1 - n_{i+2}\right) \left(1 - n_{i+4}\right) \left(1 - n_{i+5}\right) \\ &+ (1-q)\, n_{i+1} n_{i+4} \left(1 - n_i\right) \left(1 - n_{i+2}\right) \left(1 - n_{i+3}\right) \\ &+ (1-q) \left(1 - n_{i+5}\right) \\ &+ q n_{i+2} n_{i+5} \left(1 - n_i\right) \left(1 - n_{i+1}\right) \left(1 - n_{i+3}\right) \left(1 - n_{i+4}\right). \end{aligned} \tag{18}$$

These equations are easy to code in a computer and yield a fast and exact implementation of the model.

4. Experimental Results

In this section we describe some experiments with FHP for bidimensional fluid simulation. Firstly, we highlight the simplicity of creating new configurations. Figure 2-a shows an initial configuration with zero density in the corners of the system. It is not required any extra mathematical machinery to deal with such density discontinuity because system rules do not undergo modifications. We get an interesting pattern formation presented on Figure 2-b. These patterns evolve to the "S" formations pictured on Figure 3.

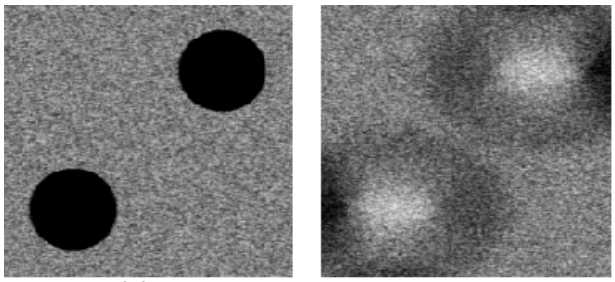

Figure 2: (a) Initial configuration. (b) Trasient pattern formation.

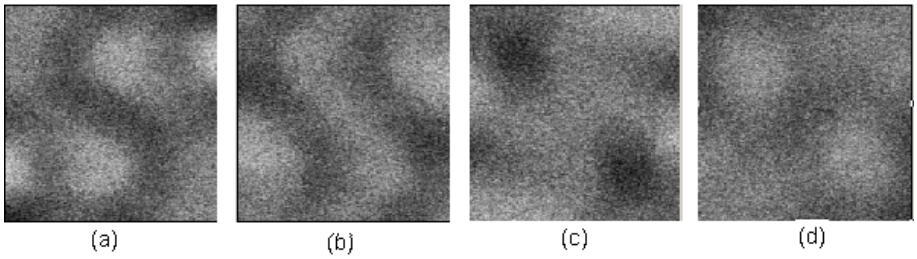

Figure 3: Evolution of the configuration pictured on Figure 2-a.

Besides, we can take advantage of the simplicity of the model for changing boundary. For a LGCA, there is no need to re-build the lattice. It is just a matter of finding the boundary cells of the lattice and apply the proper collision rules for particles entering the corresponding sites.

5. Conclusions

In this paper we propose the FHP model for fluid modelling in computer graphics applications. We discuss some experimental results using the implementation of this model. Further works are the the discussion about the theoretical elements of this proposal and the incorporation of external forces for visual effects generation.

References

[1] G. Doolen, *Lattice Gas Method for Partial Differential Equations*, Addison-Wesley, 1990.

[2] U. Frisch, D. D'Humières, B. Hasslacher, P. Lallemand, Y. Pomeau, and J.-P. Rivet, Lattice gas hudrodynamics in two and three dimension, *Complex Systems*, pages 649–707, 1987.

[3] U. Frisch, B. Hasslacher, and Y. Pomeau, Lattice-gas automata for the navier-stokes equation, *Phys. Rev.*, page 1505, 1986.

[4] J. Harting, J. Chin, M. Venturoli, and P. V. Coveney, Large-scale lattice boltzmann simulations of complex fluids: advances through the advent of computational grids, http://www.ica1.uni-stuttgart.de/ jens/pub/05/05-PhilTransReview.pdf, 2005.

[5] C. Hirsch, *Numerical Computation of Internal and External Flows: Fundamentals of Numerical Discretization*, John Wiley and Sons, 1988.

[6] T. Inamuro, T. Ogata, and F. Ogino, Numerical simulation of bubble flows by the lattice boltzmann method, *FUTURE GENERATION COMPUTER SYSTEMS*, **20(6)**:959–964, 2004.

[7] J. Mark, G. Harris, and all, Physically-based visual simulation on graphics hardware, *Graphics Hardware*, pages 1–10, 2002.

[8] M. Müller, D. Charypar, and M. Gross, Particle-based fluid simulation for interactive applications, in *Proceedings of ACM SIGGRAPH symposium on Computer animation*, 2003.
[9] S. Premoze, T. Tasdizen, J. Bigler, A. Lefohn, and R. Whitaker, Particle-based simulation of fluids, in *EUROGRAPHICS*, vol. 22, 2003.
[10] H. Bonsdorff, A comparison of the ordinary and a varying parameter exponential smoothing, *J. Appl. Prob.*, **26**, 784-792 (1989).
[11] B.L. Bowerman, R.T. O'Connell, *Forecasting and Time Series: An Applied Approach*, Belmont, CA, Duxbury Press, 1993.

(Costin Radu Boldea) Department of Computer Sciences, University of Craiova, Al.I. Cuza Street, No. 13, Craiova RO-200585, Romania, & Fax: 40-251412673
E-mail address: cboldea@central.ucv.ro