Smoothed-Particle Hydrodynamic Simulations of Viscous Fluid Flow

John Shaw, Michael Thompson, Dale Mashtare, and Rachael McGrath; Xerox Research Center Webster; Webster, New York, 14850

Abstract

The "Xerox Particle Simulation Environment" (XPSE) computer codes have been extended to enable three-dimensional fluid-flow simulations based on the Smoothed-Particle Hydrodynamic (SPH) technique. Individual ink drops (e.g., from an inkjet print head) or molten toner particles are discretized into several thousand small fluid elements. Each element samples localized material properties, such as volume mass density and temperature, averaged over a small region of space. Forces are computed on each element, which are then moved in accordance with the laws of Newtonian mechanics. The result is a time-dependent three-dimensional simulation of fluid flow including free-surface effects, self-consistent temperature and viscosity, and diffusion into porous media (e.g., paper). Multiple material types, such as different colored inks or toners, can be handled within the same simulation. Examples are drawn from early numerical experiments relating to the microscopic flow of gel-like materials.

Introduction

Using the *Xerox Particle Simulation Environment* (XPSE) [1-3], we investigate how mechanical and material properties of semi-fluid materials affect image quality in both electrographic and inkjet marking engines. For this, physically realistic simulations of microscopic fluid flow have proved invaluable.

In the electrographic case, we are interested in the fusing stage of the process. Here, solid plastic toner particles are heated and pressed onto a paper's surface using one or more cylindrical rolls. In addition to fusing to the paper, the semi-fluid particles tend to flow together with limited mixing. The subtle dynamics of this process is of particular interest when color images are being formed. In the case of inkjet printers, we are able to simulate the detailed mechanics of how solid-wax and gel inks flow and solidify onto (and into) paper as the material cools.

A brief overview of XPSE and the *Smoothed-Particle Hydrodynamics* method is presented. Realistic material properties and geometries (*e.g.,* particle and drop volumes) are used by the simulation code. The ultimate goal of these studies is to understand and improve upon the visual quality of printed images.

Overview of XPSE

XPSE is a set of C^{++} libraries and computer programs that were originally designed to enable the simulation of xerographic subsystems; such as erasure, charging, exposure, development, transfer, and fusing. The underlying code uses the particle-in-cell technique [4] to model individual toner particles in three spatial dimensions and time. Appropriate forces are calculated which describe the effects of collisions with other particles and geometric objects. The software architecture is fully object-orientated and can be thought of as an "operating system for particles". As the simulation time progresses, *events* (*e.g.,* particle-to-particle collisions) are detected, posted, and subsequently processed by registered *event handlers* (*e.g.,* the force between two colliding particles is computed).

XPSE provides a small three-dimensional CAD-like class library where all geometric objects (*e.g.,* blocks, plates, cylinders, and spheres) are "physically active". Moving donor and receiver surfaces are available for a number of development subsystem models. Simulations of transfer and fusing are evolving to include detailed air breakdown effects and pressure-driven flow of melted toner layers on paper. Many aspects of toner, carrier bead, and ion particles can be represented, including: stochastic size and charge distributions, inter-particle conduction, magnetic interactions (*i.e.,* for simulating the formation of magnetic brushes), particle-particle cohesion, and particle-boundary adhesion. The particle cohesion and adhesion models support a variety of force components such as: hard-core collisions, complex short-range forces due to charged surface patches and *Van der Waals* effects, induced electrostatic and magnetic polarization, and friction. Long-range electrostatic fields are solved on finite-element grids and blended with shorter-range forces that are calculated within the event-handler functions.

By assembling XPSE components (*e.g.,* finite-element grids, collections of toner particles, geometric objects, numerical field-solvers, *etc.*), it is possible to create digital simulators that emulate the behavior of specific pieces of hardware. These constructs may be thought of as *virtual fixtures*, and can be used by scientists and engineers to supplement experimentation on conventional *physical fixtures*. XPSE is suitable for problems where the number of cells and particles are on the order of 10^3 to 10^6. Run times vary widely with the problem being solved, but can range from minutes to tens of CPU hours on a modern PC/Linux workstation.

The focus of this paper is the extension of XPSE into the realm of fluid mechanics; specifically for the purpose of studying molten toner and gel-like ink flows on and into paper and similar substrates.

Smoothed Particle Hydrodynamics

Fluid flow is generally described by the Navier-Stokes equations or variations thereof [5, 6]. The solution of these coupled non-linear systems is challenging but well understood, primarily due to efforts of the aerospace industry over the past sixty years. Our interest is somewhat different than most aerospace problems, in that we wish to study rather small volumes of dense viscous fluid moving though air and along surfaces at low velocities. This is inherently a free-surface problem where the boundary between the fluid, air, and surface is part of the solution, rather than an imposed boundary condition.

Smoothed Particle Hydrodynamics (SPH) [7] is a numerical technique somewhat related to the *Finite-Element Method* (FEM) in that a set of discrete points sample physical quantities which are then formally interpreted using a set of basis functions. For convenience, we call the XPSE particle that corresponds to a

sampling point a *floxel* (*i.e.,* short for "fluid voxel"). By assuming the solution is defined by a sufficiently large set of floxels, one can formulate a set of equations that solve for the location of each particle based on a least-energy minimization principle, subject to a number of other physical constraints. However, unlike more conventional grid and mesh-based techniques, the relationship between floxels is not governed by a fixed topology. Unlike the case of the finite-element method, SPH interpolation functions usually encompass many sampling points; including first, second, and third nearest-neighbors. The general class of related numerical techniques is sometimes referred to as *mesh-free methods*.

The actual implementation of an SPH code looks much like a conventional particle code in that, at each discrete time step, a force is computed at the location of each floxel, and that is converted to an acceleration that is subsequently integrated into a new velocity and position vector. Thus the floxels move in time, carrying along all interpolated fluid properties. Where floxels don't exist, the fluid does not exist; thus the free-surface boundary problem is solved without undue effort. Furthermore, if floxels carry additional information such as material properties (*e.g.,* yellow vs. magenta ink), multi-phase flows with mixing are also accommodated in a very natural fashion.

One might note that SPH superficially resembles discrete-particulate flow (*e.g.,* powder flow) or the *Discrete Element Method* (DEM) [8], both of which have been used to model fluids. However SPH uses a variational Lagrangian formulation that the authors consider more appropriate for continuous systems. As a practical matter, our problem set generally involves viscous materials (*i.e.,* with kinematic viscosities on the order of 10 to 2000 *cP*, depending on the local operating temperature of the fluid) at relatively low speeds (*i.e.,* less than 3 *m/s*). Although turbulent flows are not expected, some mixing of different materials can occur. Other researchers have postulated optimized SPH algorithms for similar conditions [9]. XPSE implements SPH using the formulation of Monaghan [10] with the following important modifications:

- Floxels, which are effectively a type of XPSE particle, are assigned a fixed "hard-core size" that is determined to be consistent with the average density of the fluid. An efficient hard-core collision detection algorithm is used to prevent the sample points from approaching each other closer than this distance. Thus our SPH formulation, which would ordinarily only accommodate compressible flows, now imposes a constraint consistent with incompressible materials. It does this without the undue burden of having to implement an extremely high elastic modulus or very small time steps. If desired, this constraint can be "relaxed" to allow for moderate amounts of elastic compression, as seen in realistic materials.
- When floxels encounter regions designated as porous media (*i.e.,* paper), the underlying equations implement simple diffusion rather than Navier-Stokes. An empirical diffusion coefficient is assigned to this region, based on measurable substrate properties.
- Each floxel carries its own value of temperature. As floxels collide with each other and fixed boundaries, a thermal energy flux is calculated that is used to update the temperature of each floxel at each time step during the simulation. In this way, the microscopic temperature of the fluid is computed as a function of time and position. The local temperature directly influences the local viscosity of the fluid via a simple model or look-up table. This in turn feeds back into the equations of motion. Watkins *et. al.* [11] provides improved mechanisms for treating viscosity in SPH formulations. These ideas are not presently being used, but may be incorporated into future versions of XPSE.

Parallelization of XPSE's Particle Code

As might be expected from the above description of the SPH method, a great deal of computation must be done in order to study or solve realistic problems in an engineering environment. Although an individual toner particle or ink drop might "only" require $O(5000)$ floxels, we would like to simulate small solid patches, line edges, and half-tone dot patterns. Previous work with toner development simulations has indicated that being able to simulate a square millimeter area of substrate is sufficient for our needs. Even simulating a square millimeter requires *a lot* of particles and CPU time (*e.g.,* millions of particles and hours of CPU time).

XPSE uses a chaining-mesh [4] in order to calculate particle-particle collisions, which effectively makes it's algorithm scale close to $O(N)$ for particles and $O(T)$ for time. All particle-force and advection calculations have been parallelized using threading, which is suitable for modern shared-memory processors with multiple cores (*e.g.,* Intel's Nehalem Core series, such as the *i7*-920). The actual efficiency of our parallel code on large SPH problems approaches 90%; that is we observer a real speedup of approximately 3.5 times when running 4 to 8 hyperthreads on a 4-core *i7*-920 processor.

Our latest research effort focuses on using a GPU card, such as those produced by *nVidia Corporation* [12], for performing most the particle-force and advection calculations that occur during an SPH simulation. We are using the CUDA [13] programming environment for this purpose. Preliminary results indicate that speedups of $O(100)$ are possible over a single Intel processor; depending on the exact hardware and software optimizations used. Although more work is required, this technique looks very promising for enabling large-scale, physically realistic, imaging simulations with a relatively modest investment in computer hardware.

Simulation Results

Our simulations begin either with solid toner particles on paper, or with ink drops in mid flight (see figure 1). In either case, a toner particle or ink drop is discretized into approximately five-thousand floxels, which are arranged as a three-dimensional hex-close packed sphere. Multiple drops can be in flight as a function of time, thus a pattern can be laid down and tracked in a realistic manner. The exact time step that is used is problem dependent, but it is typically on the order of one microsecond. A simulation may run for ten's of milliseconds – and is usually terminated when the flow rate becomes sufficiently low so that little of interest is happening. In all cases, the temperature and viscosity of the fluid is continuously changing; often in a very complex manner.

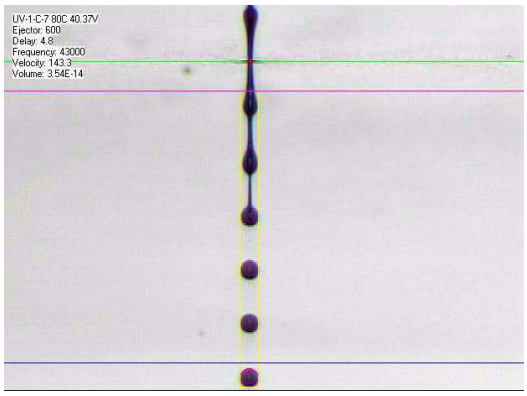

Figure 1: This image shows a set of inkjet drops being ejected from a single nozzle. Although each drop starts with a very long tail, it soon breaks off and becomes a near-perfect sphere due to its high surface tension. (photograph courtesy of Jing Zhou)

Case 1: Large Blob of Cooling Gel

In order to calibrate the simulation codes and related material properties, our initial studies concentrate on rather large "blobs" of gel-like material impacting a hard surface. XPSE can output an enormous amount of generated data, such as the complete state of every single "particle" at every time step. In practice, a small portion of the simulation data is saved to a file and visualized as an animation. Unfortunately, only very few snapshots of these animations can be shown in this paper.

Figure 2 shows a sequence of images for a simulated centimeter-sized blob of gel impacting a hard non-wettable surface. Each picture was captured at a different time in the simulation. The color represents temperature; with red corresponding to a high temperature and blue to a lower temperature. The local viscosity of the material is determined by look-up from an experimentally measured viscosity-temperature table. Note that each small "particle" in an image is representative of a floxel. This is purely an artifact of XPSE's visualization program, and the simulated fluid is mathematically smooth and continuous (*i.e.,* a floxel *is not* a particle of fluid!). Of particular interest is the rate of cooling of the drop and its final shape after it solidifies. If run long enough, the temperature of the blob approaches the temperature of the substrate. Unlike some other fluid-simulation codes, the contact angle is an output for us, not an input parameter. It is determined purely by the bulk properties of the material.

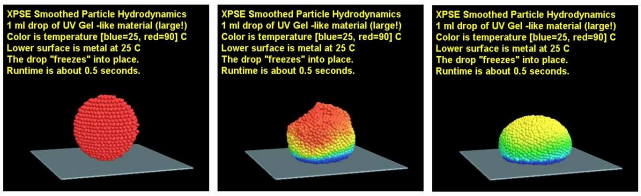

Figure 2: These three images show how a large (~1 cm) blob of gel-like material cools and solidified on contact with a cold metallic plate. The left-most image is the starting condition of a hot blob with low viscosity. It is dropped onto the plate with zero initial velocity, but it falls under gravity gaining kinetic energy. The middle image shows the blob at some later time. It is highly distorted as it is vibrating while it dissipates kinetic energy. The final image is the blob once it has frozen into place.

Case 2: Small Blob of Gel on a Wettable Surface

Figure 3 shows a similar sequence of images for a smaller millimeter-sized blob of gel impacting a hard wettable surface. As before, each picture was captured at a different time in the simulation process. The color scale, and hence local temperature, is similar to that in case 1. As the volume of the drop is small, it gains relatively little kinetic energy from its fall, so any vibrations are quickly dampened. As the drop spreads into a thing film, it reaches a uniform temperature as it freezes. We can adjust the degree that a material wets a surface by controlling how the local force between floxel particles and the surface. Roughly, if floxels are attracted to a surface, it is wettable. If they are repelled from a surface, it is not.

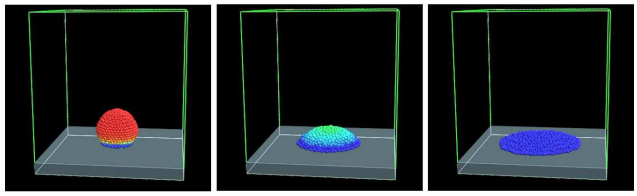

Figure 3: These three images show a smaller (~1 mm) blob of gel-like material solidifying on a hard wettable surface. As in figure 2, the time sequence runs from left-to-right.

Case 3: Small Blob of Gel Hitting a Moving Surface

Figure 4 shows a sequence of images for a millimeter-sized blob of gel impacting a moving surface. The drop has an initial downward speed of approximately 0.1 *m/s*, and the surface is moving from left to right at about 0.2 *m/s*. As the drop hits the surface, friction causes it to pick up a lateral component of momentum. As it cools and solidifies, it acquires an interesting "peanut shape" [14]. Note that this is due to conservation of angular momentum. It is unlikely to occur to any significant extent in a normal-sized inkjet drop, which would be much smaller (*e.g.,* microns in diameter).

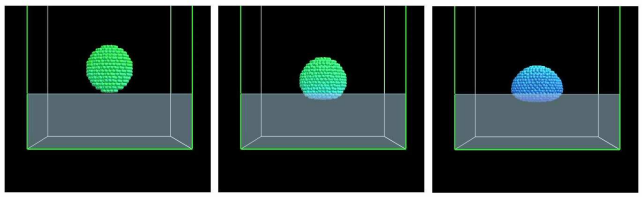

Figure 4: These three images show a small (~1 mm) blob of gel-like material impacting a moving surface. As before, the time sequence runs from left-to-right.

Case 4: Gel Diffusing into Paper

Figure 5 shows a sequence of images for a millimeter-sized blob of gel diffusing into a porous substrate. The drop has an initial downward speed of approximately 0.1 *m/s*. As the drop hits the surface, floxels are allowed to penetrate it's upper boundary. The equations of motion change from Navier-Stokes to simple linear diffusion. Given enough time and a suitably large diffusion coefficient, floxels in the paper will move out and down into the substrate. In the current implementation, the temperature of the substrate does not change, even though thermal energy from the

hot gel should cause it to (locally) rise. We hope to address this issue in the future.

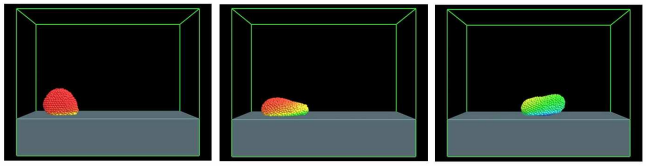

Figure 5: These three images show a small (~1 mm) blob of gel-like material impacting a moving surface. As before, the time sequence runs from left-to-right.

Case 5: Coalescence of Moving Drops

Figure 6 shows a sequence of images for three, millimeter-sized blobs of gel hitting a moving substrate while impacting with each other. The timing of the simulation is such that later drops hit earlier drops, and coalesce into a single large entity. Although the material properties of each drop is identical (which it need not be!), they are colored differently to show how mixing occurs. As the surface is non-wettable, and due to a somewhat random choice of the elastic properties of the gel, these drops tended to bounce as they hit the surface. In this simulation the temperature, and thus viscosity, of the material was held constant.

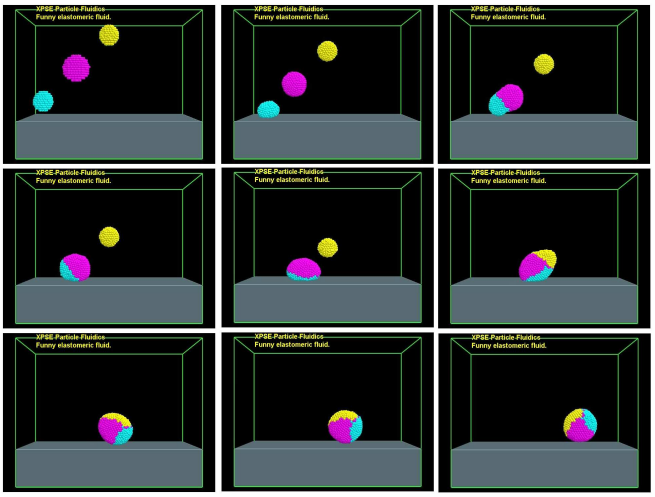

Figure 6: This sequence of images show three small (~1 mm) blobs of gel-like material impacting a moving surface and each other. The time sequence runs from left-to-right and top-to-bottom.

Conclusion

Through the use of physically realistic computer simulations, we can gain insight into the microscopic flow patterns of both molten toner and gel-like ink on porous surfaces. Although still in an early stage of development we believe that such techniques can supplement physical experimentation and aid in the optimization of electrographic and inkjet printing engines.

References

[1] J.G. Shaw and T. Retzlaff, *Particle Simulation of Xerographic Development*, IS&T Proc., 12th Int. Cong. On Digital Printing Technologies, pp. 283-287 (1996).

[2] J.G. Shaw and T. Retzlaff, *Pseudoforce Based Computation of Charged Particle Dynamics*, Proc. 20th Annual Meeting of the Adhesion Society, pp. 223-225 (1997).

[3] J.G. Shaw, D. Mashtare, P. Morehouse, M. Thompson and J. Knapp, *Image Noise in Dry-Powder Electrophotography*, IS&T Proc., 24th Int. Cong. On Digital Printing Technologies, pp. 467-470 (2008).

[4] R.W. Hockney and J.W. Eastwood, *Computer Simulation Using Particles*, IOP Publishing LTD (1988).

[5] H.L. Dryden, F.D. Murnaghan and H. Bateman, *Hydrodynamics*, Dover, New York (1956).

[6] L.D. Landau, E.M. Lifshitz, *Fluid Mechanics*, Pergamon Press, Oxford (1987).

[7] G.R. Liu and M.B. Liu, *Smoothed Particle Hydrodynamics: A Meshfree Particle Method*, World Scientific Publishing (2003).

[8] P. Azadi, N. Yan and R. Farnood, *Discrete element modeling of the transient heat transfer and toner fusing process in the Xerographic printing of collated papers*, Computers and Chemical Engineering, v. 32, pp. 3238-3245 (2008).

[9] L.G. Sigalotti, J. Klapp, E. Sira, Y. Melean and A Hasmy, *SPH Simulations of time-dependent Poiseuille flow at low Reynolds numbers*, Journal of Computational Physics, v. 191, pp. 622-638 (2003).

[10] J.J. Monaghan, *Smoothed Particle Hydrodynamics Code Basics*, Journal of the Korean Astronomical Society, v. 34, pp. 203-207 (2001).

[11] S.J. Watkins, A.S. Bhattal, N. Francis, J.A. Turner and A.P. Whitworth, *A new prescription for viscosity in Smoothed Particle Hydrodynamics*, Astronomy & Astrophysics Supplement Series 119, pp. 117-187 (1996).

[12] Tesla, *http://www.nvidia.com/object/tesla_computing_solutions.html*

[13] CUDA Zone, *http://www.nvidia.com/object/cuda_home_new.html*

[14] Tracy Nowak, private communication (2010).

Author Biography

John G. Shaw received a PhD in applied physics from the University of Manitoba in 1983. Since then he has worked in the Research and Technology Division of Xerox Corporation in Palo Alto, CA, Ithaca, NY (while visiting at Cornell University), and Webster, NY. His early career revolved around solid-state physics and amorphous-silicon technology. His recent work has focused on the development of toner adhesion models and particle-transport simulations.