Procedural Fluid Animation using Mirror Image Method

Jinho Park

Multimedia Department, Namseoul University, Korea

ABSTRACT

Physics based fluid animation schemes need large computation cost due to tremendous degree of freedom. Many researchers tried to reduce the cost for solving the large linear system that is involved in grid-based schemes. GPU based algorithms and advanced numerical analysis methods are used to efficiently solve the system. Other groups studied local operation methods such as SPH (Smoothed Particle Hydrodynamics) and LBM (Lattice Boltzmann Method) for enhancing the efficiency. Our method investigates this efficiency problem thoroughly, and suggests novel paradigm in fluid animation field. Rather than physics based simulation, we propose a robust boundary handling technique for procedural fluid animation. Our method can be applied to arbitrary shaped objects and potential fields. Since only local operations are involved in our method, parallel computing can be easily implemented.

Keywords: Fluid Animation, Vector Potential, Mirror Image Method, Boundary Condition and Procedural Method.

1. INTRODUCTION

Grid based approach to numerically solve the Navier-Stokes equations is most popular approach for producing realistic fluid animation. Projection step is the worst bottleneck in the grid based fractional step method. This bottleneck is due to global operation characteristic of the projection step. Our objective is to propose a grid-based method satisfying divergence free property without the projection method. Computational efficiency is an important theme in fluid simulation since its heavy computational load is burden to designers.

The notorious projection problem has been interested in many researchers. One group tried to reduce computation time itself for solving the linear system of extremely high dimension. GPU based algorithms and advanced numerical analysis methods are located in the group. The other group investigated local operation method such as SPH (Smoothed Particle Hydrodynamics)[1],[2] and LBM (Lattice Boltzmann Method)[3],[4] in spite of their quality degradation. Our method solves this problem thoroughly, and suggest novel paradigm in fluid simulation.

2. RELATED WORK

Many researchers have proposed efficient algorithms for linear system brought by incompressibility of fluid: GPU instead of CPU [5], preconditioned CGM [6] and Multigrid instead of general CGM [7]. Such methods improved the computational efficiency surprisingly. However, they did not remove the large scale linear system completely.

SPH is relatively recent approach to enhance the computation load of fluid simulation [2]. Basic SPH consists of only local operations, and are hard to satisfy incompressibility. Some improved SPH were proposed to match the incompressibility with high computational load.

Particular potential flows such as uniform flows, vortices, sources, and sinks are used to generate wind velocity fields [8, 9]. However, those approaches are restricted to laminar flows for irrotationality of the potential flows. Moreover, the boundary condition for arbitrary shaped geometry is not handled.

Fourier analysis is used to produce physically plausible turbulent velocity fields [10],[11]. The approaches cannot handle the arbitrary shaped boundaries and provide user interactions such as adjusting the magnitude of the turbulence.

Perlin's noise function [13]-[15] is widely used in computer graphics to generate random velocity fields; however, these fields cannot satisfy the divergence free condition which is a crucial property for realistic fluid behavior. Moreover, the Perlin noise based methods are hard to handle the arbitrary solid geometry. Procedural mechanisms by Lamorlette and Foster [16] generate all aspects of flame behavior including moving sources, combustion spread, flickering, separation and merging, and interaction with stationary objects.

Kniss and Hart [17] use the vector identity that the curl of any vector field is equal to zero for generating incompressible flow fields. Patel and Taylor [18] introduce an divergence free field generation technique based on noise function. Though the technique is efficient and adequate for artist control, it does not

consider boundary handling. Von Funck et al. [19] present an approach for shape deformations by constructing divergence-free vector fields which are based on vector calculus.

3. OUR METHOD

Given a potential field defined over a domain and an immersed object in the domain, our method handles boundary conditions including prevention of flow penetration into the object and slip of flow on the object boundary. For 2D flow generation, the potential field is given as a scalar function over a 2D domain. A vector field defined by the curl of the potential field depicts a fluid flow. From the vector identity that the divergence of a curl is zero, the flow is divergence free which is an essential component for realistic flow. Vector valued function is used as the potential field for 3D flow. Every argument for 2D flow such as the vector identity and divergence free can be equally applied to 3D case. Our method does not restrict the geometry or number of involving obstacles. It can be applied to both convex and concave objects. Such general boundary handling is the main contribution of our method. The last requirement for our solution is real-time performance. Unlike physically based fluid simulations, our fluid generation mechanism using potential field is spatiotemporally analytic: every flow behavior is known on whole timeline. Therefore, boundary handling must be also performed in real-time.

Fluid dynamics has a boundary treatment method known as mirror image [20],[21] for potential field with simple geometries such as flat wall and sphere. Mirror image method considers the boundary as a mirror. Given any potential field, a mirror image of the field is assumed to be located inside the boundary. For a flat wall and a source flow, the method places another source with the same strength on the opposite side of the wall. The two distances from wall are same. For vortex flow, a vortex rotating in opposite direction is used to satisfy the boundary condition of the flat wall. Superposition of a field and its mirror image field, specifically addition, results a potential field satisfying the boundary condition of wall.

The mirror image of a field for sphere boundary can be described by Kelvin transform [21] which describes the transformation between inside and outside of a sphere. Given a sphere with radius R, the mirror image x^* of a point x in $\mathbb{R}^n$ with respect to the sphere is defined as

$$x^* = \frac{R^2}{|x|^2} x \qquad (1)$$

The outside of the sphere is mapped into the inside, the conversion is also valid, and points on the boundary are mapped into themselves. The Kelvin transform f^* of a function f over $\mathbb{R}^n$ with respect to the sphere is

$$f^*(x^*) = \frac{|x|^{n-2}}{R^{2n-4}} f(x) = \frac{1}{|x^*|^{n-2}} f(x).$$

It is not straightforward to generalize the Kelvin transform for arbitrary shaped geometry. Flat wall can be considered as a sphere with infinite radius. Line segment between a point and its image by sphere or wall is perpendicular to the tangent at the intersection point between the line segment and the involving object. It is hard to design a mirroring transform for general geometry which analytically satisfies the perpendicularity.

We tweak the Kelvin transform to handle arbitrary shaped boundaries not only sphere. Our idea is to allow the radius R varying, not fixed. Many intersections between given arbitrary shaped object and a segment connecting x and center of mass for the object occurs. The alternative $R(x)$ is defined by the distance between the center of mass and the outmost intersection. For sphere, the intersection is unique for x and $R(x)$ is constant, the radius of sphere. The Kelvin transforms of point and function can be described with variable $R(x)$ (figure 1). With this modified Kelvin transform, exterior of an object is mapped into the interior of the object. Boundary is mapped into the boundary itself. Our idea is quite simple but, effective in boundary handling problem. As experiments show, the modified method works well for arbitrary shaped objects.
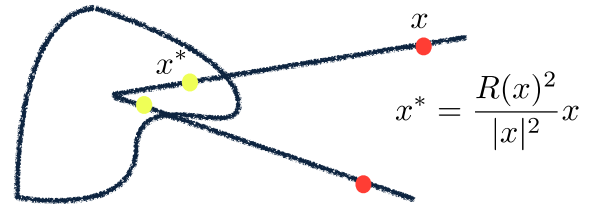

Fig. 1. Application of Kelvin transform to arbitrary objects: x^* is the transformed point inside the object of x.

Mirror image method based on our modified Kelvin transform works well for analytic boundaries. Here, analytic means the boundary is given by analytic function. Source field in figure 2(b) is $P(x, y) = \arctan\left(\frac{y-b}{x-a}\right)$, where (a, b) is the source position. Its mirrored field P^* is $\arctan\left(\frac{y+b}{x-a}\right)$ when $y = 0$ is mirror axis. Uniform flow field in figure 2(a) is $P(x, y) = y$ and its mirrored field P^* is $\frac{R^2}{|\vec{x}|^2} y$. The cardioid in figure 2(c) has Cartesian equation $(x^2 + y^2 + ax)^2 = a^2(x^2 + y^2)$, where a is a constant parameter. Radius R in equation 1 varies due to the position as shown in figure 1. In figure 2(d), two sources are used. Their own mirrored sources are evaluated individually. Linearly combined one of four fields shows boundary handling and two sources.
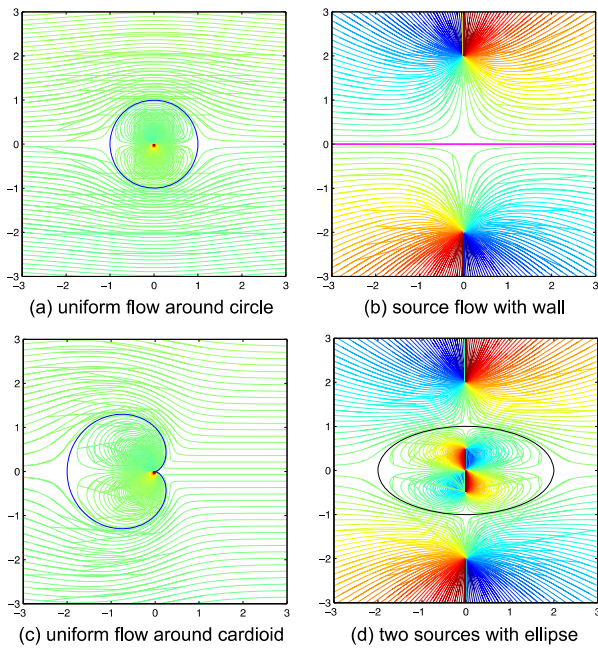
(a) uniform flow around circle   (b) source flow with wall

(c) uniform flow around cardioid   (d) two sources with ellipse

Fig. 2. Basic flows around various boundaries

4. EXPERIMENTAL RESULTS

Our method works well for multiple objects only with separable local computations. Figure 3 shows vortex field with two obstacles. Underlying potential field P is assumed as a bunch of vortices. For any point x, resulting potential $\hat{P}$ due to n objects is summation of mirrored fields: $\hat{P}(x) = P(x) - P_1^*(x) - P_2^*(x) - \cdots - P_n^*(x)$. Each mirrored field P_i^* is evaluated independently regardless of other mirrored field $P_j^*, j \neq i$. While computational cost for boundary handling is proportional to the number of involving objects, the cost can be reduced dramatically since our method does not involve any global operations.
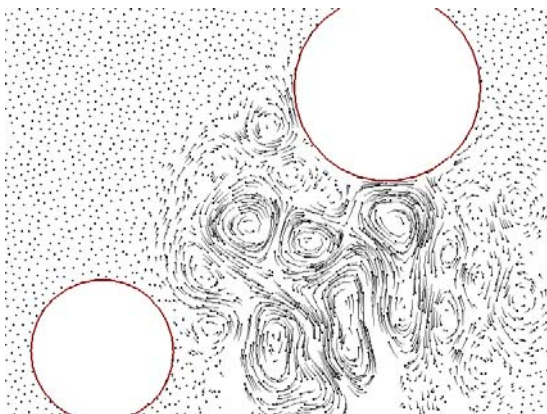

Fig. 3. Vortex flow

Figure 4 demonstrates our method can be applied to arbitrary shaped boundaries. For objects expressed analytically, the radius in Kelvin transform is also evaluated analytically. Intersection point between the object and line connecting the

origin and x is calculated in analytic way. Radius $R(x)$ in figure 1 is given as the distance between the origin and the intersection point. On the other hand, the radius in Kelvin transform for a polygonal object needs a numerical algorithm. For a polygonal object, the intersection check should be evaluated for every segment of the object. The computational cost for the check is proportional to the number of edges (faces for three dimensional case).
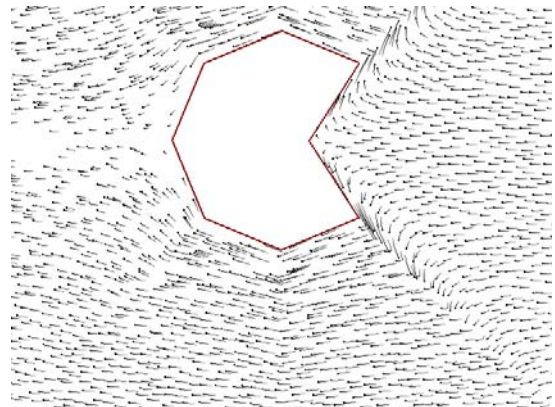

Fig. 4. Polygonal object

Curl noise [22] recently tackles the boundary problem for procedural flow. Main idea of curl noise is to make the potential function have a constant (zero in the paper) to match no slip boundary condition. For slip boundary condition, it modulates the potential with a ramp through zero based on distance to the closest boundary point. For no-slip boundary condition, it modulates the potential down to zero with a smoothed step function of distance, so that all the partial derivatives (and hence the curl) of the new potential are zero at the boundary. Their modulation satisfies the boundary condition on the exact boundary. The potential function in modification region smoothly varies from original potential function to the constant using Heaviside function. Then, normal and tangent gradient of potential function must be zero, and the velocity defined by the curl of potential is also zero on the boundary, this matches the no slip boundary condition. However, enforcement of zero potential on boundary changes potential in blend region. The velocity depends on gradients of potential. Such unreasonable enforcement occasionally changes flow direction (figure 5(a)). For single and simple object such as sphere, translation of a potential may hide the problem. However, for general and multiple cases, this opposite direction problem is inevitable. While additional equipment such as noise may cover the problem visually, this zero potential based tricks is far from complete analysis of the boundary problem. On the other hand, our method synthesizes the potential flow with correct boundary handling (figure 5(b)).
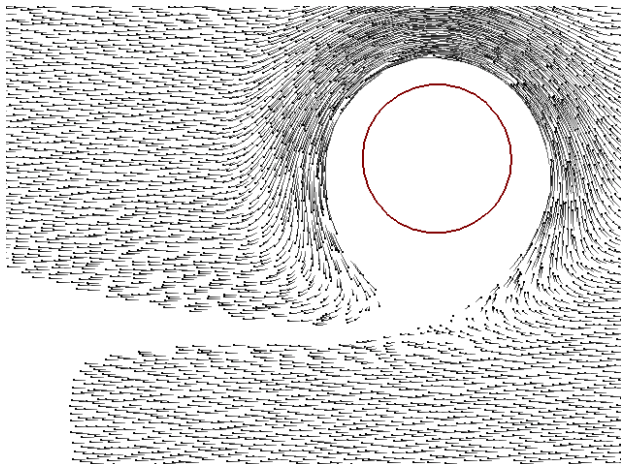
(a) Curl noise: Lower region of the circle has wrong flow with the opposite direction.
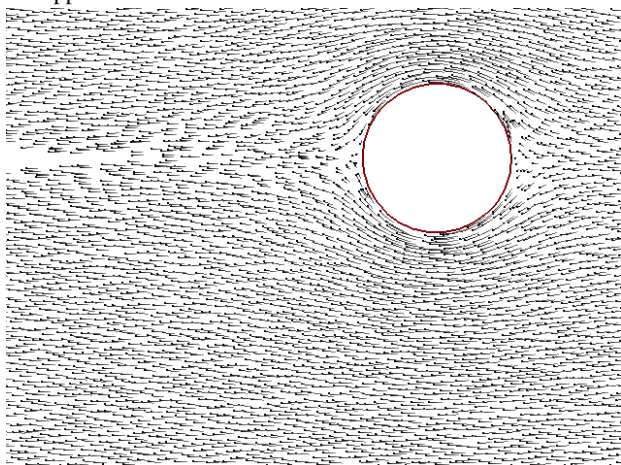
(b) Our method: the synthesized flow shows our method can handle the boundary condition well.

Fig. 5. Curl noise vs. Our method

Our method was implemented on a common system with 2.4 GHz Intel Core2 Quad Q6600 CPU and 8 GB RAM. Experiments shows real time performance: less than 0.01 sec/step for uniform flow with sphere (figure 5), 0.02 sec/step for vortex with two spheres (figure 3), and 0.03 sec/step for uniform flow with polygonal object (figure 4).

5. DISCUSSIONS AND CONCLUSION

To evaluate a velocity at a point, our method queries neighboring potentials since curl of potential is velocity. Particles in fluid domain are advected with the velocity field. Particles can penetrate into given objects due to numerical error. When such event occurs, our method removes the particle from the particle list. In computational geometry, the point-in-polygon (PIP) problem [23] asks whether a given point in the plane lies inside, outside, or on the boundary of a polygon. An early description of the problem in computer graphics shows two common approaches (ray casting and angle summation) in use as early as 1974.

Our method uses the center of mass of the object as the center.

For polygonal objects, the center of mass is simply the mean of the vertices. Experimental results show such simple strategy works well. However, it is not accurate strategy in the sense of fluid mechanics. We leave the problem of finding the accurate center as a future work.

Our modified Kelvin transform defines radius $R(x)$ by the distance between the center of mass and the outmost intersection. For polygonal object with finite line segments, we check all segments to calculate $R(x)$. Spatial partitioning such as binary space partitioning helps efficiency for complex objects with many line segments. Although such formulation makes visually plausible images in our experiments, it does not guarantee the quality of results for every geometry. For example, the discontinuity of the radius could be occurred for non-star shaped polygon such as alphabet E. In future, we will investigate smooth radius design.

We propose a robust boundary handling technique for procedural fluid animation. The handling technique is applied to arbitrary shaped objects and arbitrary potential field. Our work investigates the real-time fluid animation field and can be easily applied to parallel computing since no global operations are involved.

REFERENCES

[1] J.J. Monaghan, "An introduction to SPH", Computer Physics Communications, vol. 48, 1988, pp. 88–96.

[2] M. Muller, D. Charypar, and M. Gross, "Particle-based Fluid Simulation for Interactive Applications", *Proc. Eurographics/SIGGRAPH Symposium on Computer Animation,* 2003, pp. 154-159.

[3] S. Chen and G.D. Doolen, "Lattice Boltzmann Method for Fluid Flows", Annual Review of Fluid Mechanics, vol. 30, 1998, pp. 329-364.

[4] N. Thürey, K. Iglberger, and U. Rüde, "Free Surface Flows with Moving and Deforming Objects with LBM", *Proc. VMV'06*, 2006.

[5] J. Kruger and R. Westermann, "Linear algebra operators for GPU implementation on numerical algorithms", *Proc. ACM SIGGRAPH'05 Courses*, 2005.

[6] G.H. Golub and Q. Ye, "Inexact Preconditioned Conjugate Gradient Method with Inner-Outer Iteration", SIAM J. Sci. Comput. vol. 21, 1997, pp. 1305-1320.

[7] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, *Numerical Recipes: The Art of Scientific Computing*, Cambridge University Press, New York, 2007.

[8] K. Sims, "Particle animation and rendering using data parallel computation", *Proc. ACM SIGGRAPH'90*, 1990,

pp. 405–413.

[9] J. Wejchert and D. Haumann, "Animation aerodynamics", *Proc. ACM SIGGRAPH'91*, 1991, pp. 19–22.

[10] M. Shinya and A. Founier, "Stochastic motion: Motion under the influence of wind", *Proc. Eurographics'92*, 1992, pp. 119– 128.

[11] J. Stam and E. Fiume, "Turbulent wind fields for gaseous phenomena", *Proc. ACM SIGGRAPH'93*, 1993, pp. 369–376.

[12] J. Stam, "Stochastic dynamics: Simulating the effects of turbulence on flexible structures", Computer Graphics Forum, vol. 16, Sep. 1997, pp. 159-164.

[13] K. Perlin, "An image synthesizer", *Proc. ACM SIGGRAPH'85*, 1985, pp. 287–296.

[14] K. Perlin, "Improving noise", ACM Trans. Graph., vol. 21, no. 3, 2002, pp. 681–682.

[15] K. Perlin and F. Neyret, "Flow noise", *ACM SIG-GRAPH Technical Sketches and Applications*, 2001, p. 187.

[16] A. Lamorlette and N. Foster, "Structural modeling of flames for a production environment", *Proc. ACM SIGGRAPH'02*, 2002, pp. 729–735.

[17] J. Kniss and D. Hart, "Volume effects: modeling smoke, fire, and clouds*", ACM SIGGRAPH'04 courses*, 2004.

[18] M. Patel and N. Taylor, "Simple divergence-free fields for artistic simulation", Journal of graphics tools, vol. 10, no. 4, 2005, pp. 49–60.

[19] V.W. Funck, H. Theisel, and H.P. Seidel, "Vector field based shape deformations", ACM Trans. Graph., vol. 25, no. 3, 2006, pp. 1118–1125.

[20] F.M. White, *Fluid Mechanics (Chap. 8)*, Mc-Graw Hill, 2006.

[21] J.L. Doob, *Classical Potential Theory and Its Probabilistic Counterpart*, Springer-Verlag, 2001.

[22] R. Bridson, J. Houriham, and M. Nordenstam, "Curl-noise for procedural fluid flow", ACM Trans. Graph., vol. 26, no. 3, 2007.

[23] I.E. Sutherland, R.F. Sproull, R.A. Schumacker, "A Characterization of Ten Hidden-Surface Algorithms", ACM Computing Surveys, vol. 6, no. 1, 1974.

Jinho Park
He received his B.S. and M.S. degrees in applied mathematics in 1999 and 2001, respectively, and Ph.D. in Computer Science in 2007 from Korea Advanced Institute of Science and Technology. He is an assistant professor in the Department of Multimedia at Namseoul University, South Korea. His research interests include fluid animation and scientific visualization.