Visual Simulation of Freezing Ice with Air Bubbles

Takanori Nishino, Wakayama University
Kei Iwasaki, Wakayama University
Yoshinori Dobashi, Hokkaido University, JST CREST
Tomoyuki Nishita, The University of Tokyo

Abstract

Ice often contains small air bubbles that are dissolved air exhausted from water-ice interface. Depicting these air bubbles with various sizes enhances the realism of ice. This paper proposes a fast simulation method of freezing ice taking into account air bubbles. Our method introduces a physically-based calculation method of the sizes of air bubbles based on the freezing velocity and pressure. In contrast to the previous method, our method can represent numerous small bubbles that cannot be represented by using the resolutions of simulation grids. The computations of heat transfer, tracking of water-ice interface, and generation of bubbles are accelerated by implementing our method using CUDA. Our method achieves the freezing simulation and bubble generation at interactive frame-rates.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation;

Keywords: natural phenomena, visual simulation, ice, bubble, GPU

1 Introduction

Visual simulations of natural phenomena have become an important research subject in the field of computer graphics. In recent years, many methods have been proposed for simulating smoke, fire, clouds, and ice. In this paper, we focus on the freezing simulation of ice. Ice from tap water often contains small air bubbles with various sizes and shapes as shown in Fig. 1. These bubbles are dissolved air that are exhausted due to freezing and trapped in ice. These small air bubbles are important features to represent ice in computer graphics. To depict ice containing air bubbles, modeling numerous small air bubbles is a burden task. To address this, freezing simulation with air bubbles is required to model ice efficiently.

Several methods have been proposed for visual simulations of freezing. Kharitonsky et al. proposed a physically-based model of icicle growth [Kharitonsky and Gonczarowski 1993]. Kim and Lin presented a physically-based model of ice crystal growth [Kim and Lin 2003]. Kim et al. presented a hybrid algorithm for modeling ice formation [Kim et al. 2004], and they also simulated the growth of icicles by solving a thin film Stephan problem [Kim et al. 2006]. Although these methods can simulate freezing of ice, these methods do not take into account air bubbles in ice. Seipel et al. proposed a real-time rendering method of ice taking into account small air bubbles and cracks in ice [Seipel and Nivfors 2006]. This method depicts air bubbles and cracks in ice by using precomputed textures representing air bubbles and cracks.

Copyright © 2012 by the Association for Computing Machinery, Inc.
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions Dept, ACM Inc., fax +1 (212) 869-0481 or e-mail permissions@acm.org.

SIGGRAPH Asia 2012, Singapore, November 28 – December 1, 2012.
© 2012 ACM 978-1-4503-1757-3/12/0011$15.00

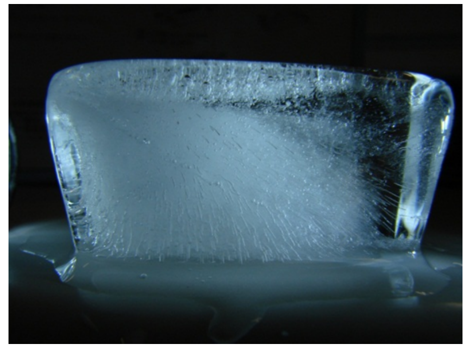

Figure 1: *Photograph of ice containing small air bubbles.*

Simulations of bubbles have been widely studied [Hong and Kim 2003; Zheng et al. 2006; Cleary et al. 2007; Thürey et al. 2007; Hong et al. 2008; Busaryev et al. 2012]. However, these methods mainly focus on the interactions between bubbles and fluids and do not deal with generation of bubbles due to freezing.

To address this problem, Madrazo et al. proposed a freezing simulation method taking into account air bubbles [Madrazo and Okada 2008; Madrazo et al. 2009]. This method simulates the movement of dissolved air exhausted from water-ice interface. Unfortunately, this method has several limitations. Firstly, Madrazo's method assumes that the freezing velocity is constant in time, while the freezing velocity is proportional to the gradient of temperature. Moreover, the freezing velocity affects the sizes of air bubbles. That is, small air bubbles are generated for high freezing velocity and large air bubbles are generated for low freezing velocity. Secondly, Madrazo's method does not describe the effect of the pressure on air bubbles while pressure affects the sizes of air bubbles [Yoshimura et al. 2007]. Finally, Madrazo's method generates one air bubble per grid cell. Therefore, high resolution grids are required to generate many small bubbles in ice. This results in an increase in the simulation time.

This paper proposes a fast freezing simulation method taking into account air bubbles to solve these problems. The contributions of our method are as follows:

- calculation method of sizes of air bubbles taking into account pressure and freezing velocity is introduced.
- numerous small air bubbles that are independent on resolutions of simulation grids can be generated.
- interactive freezing simulation with air bubbles is achieved by using our GPU implementation.

2 Freezing Simulation

Fig. 2 shows an overview of our simulation method. Our method consists of two processes: freezing simulation process and bubble generation process. The simulation space is subdivided by a grid. At each grid point $\mathbf{x}$, temperature T, heat value Q, density ρ, pressure p, flag that indicates ice or water, freezing velocity $\mathbf{v}$, value of level set function $\phi(\mathbf{x})$ that represents the signed distance from water-ice interface, and weight of dissolved air $S(\mathbf{x})$ are assigned. Our method assumes that the water in the container whose top is

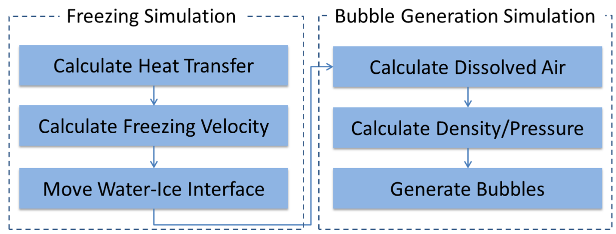

Figure 2: *Overview of our simulation method. This figure shows a single time step of our simulation.*

open freezes due to the surrounding cool air and our method does not consider the diffusion of water.

2.1 Heat Transfer

Our method simulates the heat transfers between water, ice, air, and the container. The heat transfer is calculated by :

$$\frac{\partial T}{\partial t} = a\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2}\right), \tag{1}$$

where T is the temperature, t is the time, a is the thermal diffusion constant (that depends on the flag of the grid point). Our method takes into account the latent heat of solidification. That is, the water with 273K temperature does not change to the ice until the latent heat of solidification $L\rho_{ice}V_c$ is released, where L is the latent heat of water, ρ_{ice} is the density of ice, and V_c is the volume of each grid cell.

2.2 Calculation of Freezing Velocity

As shown in Fig. 3(a), the freezing velocity $\mathbf{v}$ at the water-ice interface is calculated by

$$\mathbf{v} = \frac{K_{ice}}{L\rho}(\nabla T \cdot \mathbf{n})\mathbf{n}, \tag{2}$$

where K_{ice} is the thermal conductivity of ice, ρ is the density of water, ∇T is the gradient of the temperature, and $\mathbf{n}$ is the normal of the water-ice interface and calculated by the following equation.

$$\mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|}, \tag{3}$$

$$\nabla\phi = \left(\frac{\partial\phi}{\partial x}, \frac{\partial\phi}{\partial y}, \frac{\partial\phi}{\partial z}\right). \tag{4}$$

2.3 Moving Water-Ice Interface

The water-ice interface moves according to the freezing velocity calculated by Eq. (2). The water-ice interface is defined by the level set function $\phi(\mathbf{x})$. Level set function $\phi(\mathbf{x})$ is calculated by using the particle level set method [Enright et al. 2002]. The water-ice interface is extracted by using the marching cube method.

3 Generation of Bubbles

Dissolved air on the water-ice interface is pushed into the water since air is hard to be dissolved in ice. When the volume of dissolved air exceeds the saturated dissolved air, bubbles are generated. Most of generated bubbles are trapped in the water-ice interface and do not rise to upward [Maeno 1966]. Our method represents the air bubbles with spheres. The cylindrical air bubbles are represented by connecting several spheres.

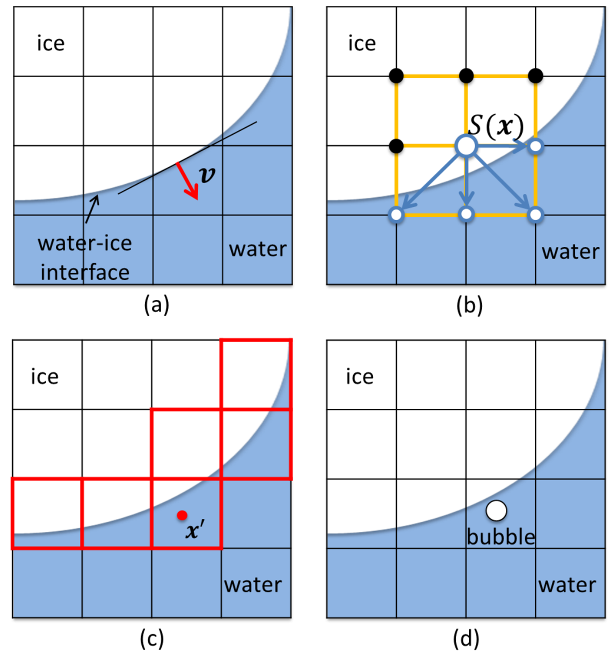

Figure 3: *Movement of water-ice interface and generation of bubble.*

3.1 Movement of Dissolved Air

Our method calculates the weight of dissolved air $S(\mathbf{x})$ at each grid point to simulate generation of bubbles. The initial value of dissolved air $S(\mathbf{x})$ at each grid point inside water is set to C_0V_c, where V_c is the volume of grid cell and C_0 is the dissolved air concentration of tap water and is set to $2.1 \times 10^{-2}[\mu g/mm^3]$ in our examples. If the grid point $\mathbf{x}$ changes its flag to ice, dissolved air $S(\mathbf{x})$ is uniformly distributed to neighboring grid points of water as shown in Fig. 3(b) and $S(\mathbf{x})$ is set to zero.

3.2 Density Calculation

Our method calculates the radii of air bubbles based on the pressure of water. To compute the pressure, the previous method [Becker and Teschner 2007] employs the Tait equation. Unfortunately, this method cannot be applied to the freezing simulation, since this method calculates the pressure by using the maximum speed of flow, which is considered as zero in the freezing simulation. Therefore, we propose a simple calculation model of pressure based on the measured compression ratio χ of water [National Astronomical Observatory Of Japan 2012].

To compute the pressure of water, our method first calculates the density of water as follows. The volume V_c of the grid cell for water expands to $1.09V_c$ when it freezes. Our method assumes that the volume expands vertically and the ratio of the downward expansion is set to c and that of the upward expansion is $0.09 - c$. Let V be the initial volume of water and n_{ice} grid points are considered as ice. Then the volume of water V_{water} compressed by the downward expansion of ice is calculated as $V - n_{ice}(1 + c)V_c$. The water density ρ is calculated as

$$\rho = \frac{n_{water}M_{water}}{V_{water}}, \tag{5}$$

where n_{water} is the number of grid points for water, M_{water} is the weight of water for each grid cell.

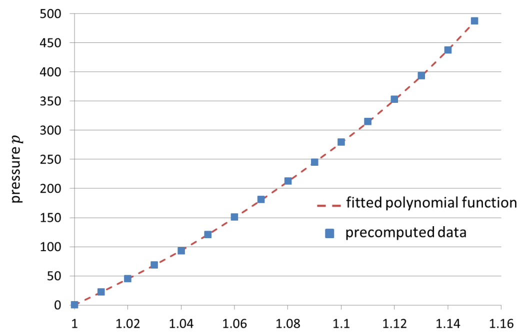

Figure 4: *Relationship between density and pressure. Our polynomial approximation (red dashed line) matches the precomputed data (blue square) well.*

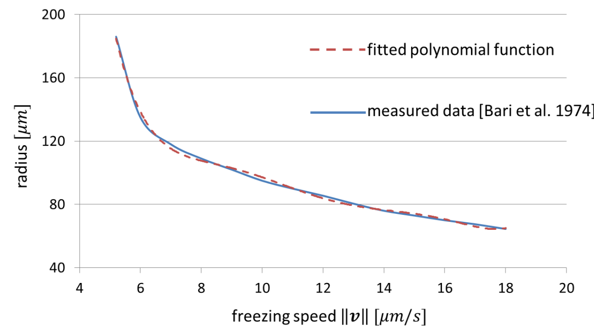

Figure 5: *Relationship between freezing speed $||\mathbf{v}||$ and radius of bubble. Our polynomial representation (red dashed line) matches the measured data (blue line) well.*

3.3 Pressure Calculation

To calculate pressure p by using density ρ, our method represents p with a polynomial function of ρ. In the preprocess, our method calculates ρ for various sampled pressure in order to fit p with a polynomial function of ρ as follows. The variation of volume dV due to the increase of the pressure from standard atmosphere p_0 to p is calculated by

$$dV = -v_0 \int_{p_0}^{p} \chi(p')dp', \tag{6}$$

where v_0 is the initial volume, $\chi(p')$ is the compression ratio at pressure p' [National Astronomical Observatory Of Japan 2012]. Then the density ρ at pressure p is calculated by

$$\rho = \rho_0 \frac{v_0}{v_0 + dV}, \tag{7}$$

where ρ_0 is the density of water at standard atmosphere p_0. Our method precomputes ρ for sampled pressure and fits p for polynomials of density ρ as

$$p = \begin{cases} 10^5(6.35\rho^4 - 27.1\rho^3 + 43.6\rho^2 - 31.2\rho + 8.33) & (p > 100) \\ 10^3(3.77\rho^2 - 4.37\rho + 1.60) & (p \leq 100). \end{cases}$$

Fig. 4 shows the relationship between our polynomial representation and precomputed values and demonstrates the accuracy of our polynomial representation.

3.4 Generation of Bubbles

To generate air bubbles, our method first searches grid cells that contain water-ice interfaces (red grid cells in Fig. 3(c)). Then the position $\mathbf{x}'$ where an air bubble is generated is randomly determined. The dissolved air $S(\mathbf{x}')$ at $\mathbf{x}'$ is interpolated by those of neighboring grid points of water.

Next, our method determines the radius of the air bubble. The radius of the bubble is calculated from the freezing velocity and the pressure at $\mathbf{x}'$. The freezing velocity and the pressure at $\mathbf{x}'$ are interpolated by those of 8 neighboring grid points. Our method calculates the radius of air bubble at standard pressure p_0 by using the measured data [Bari and Hallett 1974]. To do this, our method represents the radius of air bubble r' at p_0 with a polynomial function of freezing speed as follows.

$$\begin{aligned} r' &= 1.33 \times 10^{-3}||\mathbf{v}||^6 - 9.77 \times 10^{-2}||\mathbf{v}||^5 + 3.04||\mathbf{v}||^4 \\ &\quad -48.4||\mathbf{v}||^3 + 423||\mathbf{v}||^2 - 1937||\mathbf{v}|| + 3737. \end{aligned} \tag{8}$$

Fig. 5 shows the relationship between freezing speed $||\mathbf{v}||$ and radius. As shown in Fig. 5, our polynomial representation matches to the measured data well. Then our method considers the effect of the pressure at $\mathbf{x}'$. Due to the Boyle's law, the pressure and the volume of gas are inversely proportional. Therefore, the product of the volume of air bubble $\frac{4\pi}{3}r'^3$ and p_0 is equal to the product of the volume of air bubble $\frac{4\pi}{3}r^3$ and p. By using the Boyle's law, the radius r at p is calculated by $r = r'(p_0/p)^{1/3}$.

By using r, the volume of air V_{air} required to generate the air bubble is calculated as $V_{air} = \frac{4\pi}{3}r^3$. The weight of air W_{air} under pressure p is then calculated as $W_{air} = \rho_{air}V_{air}\frac{p}{p_0}$, where ρ_{air} is the density of air. The air bubble is generated at $\mathbf{x}'$, if the dissolved air $S(\mathbf{x}')$ exceeds both W_{air} and $S_{thres}\frac{p}{p_0}$, where S_{thres} is the saturated dissolved air. Our method uses S_{thres} as $C_{thres}V_c$ where C_{thres} is the saturated concentration of dissolved air at 273K, and is set to $3.84 \times 10^{-2} [\mu g/mm^3]$. At the neighboring grid points, the amount of air $\frac{W_{air}}{N_{water}}$ is subtracted from $S(\mathbf{x})$ at each grid point $\mathbf{x}$ for water, where N_{water} is the number of neighboring grid points for $\mathbf{x}'$.

4 Results

Fig. 6 shows a simulation of freezing ice cube. Water in the acrylic container freezes due to the surrounding cool air, and the dissolved air bubbles are exhausted from the water. We have implemented our algorithm using CUDA and tested it on a standard PC with Core i7-2700K CPU and GeForce 580GTX GPU. The images are rendered by using Pov-ray.

Our method assumes that the simulation scene consists of a single acrylic container (external size $70 \times 60 \times 70mm^3$, internal size $50 \times 50 \times 50mm^3$) and the container contains water (278K) in the refrigerator (258K). The grid resolution is 101^3 and the time step Δt is set to 0.1 sec except for the calculation of heat transfer ($\Delta t =$ 0.002 sec) due to the CFL condition. The average computational time for a single time step is 45.7 ms (21.9 frames per second) using our GPU implementation.

Fig. 7 shows a simulation of freezing water contained in a stanford bunny. The small air bubbles are generated around the water-ice interface.

5 Conclusion and Future Work

We have proposed a fast simulation method of freezing ice taking into account air bubbles. We presented a calculation method for

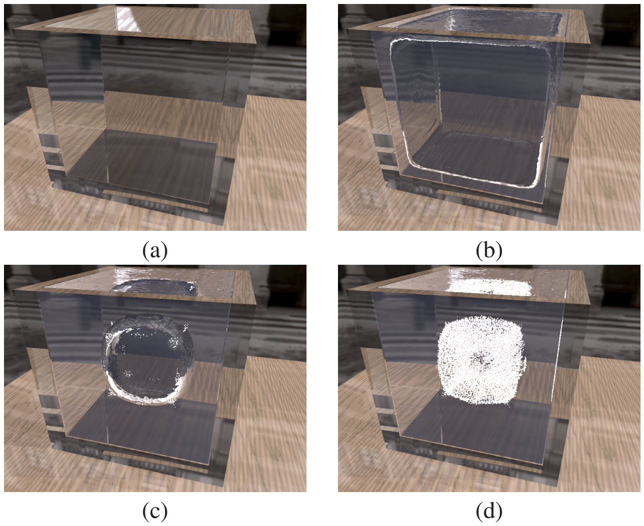

Figure 6: *Freezing simulation of water in the cubic acrylic container. Small air bubbles are generated in the center of the ice.*

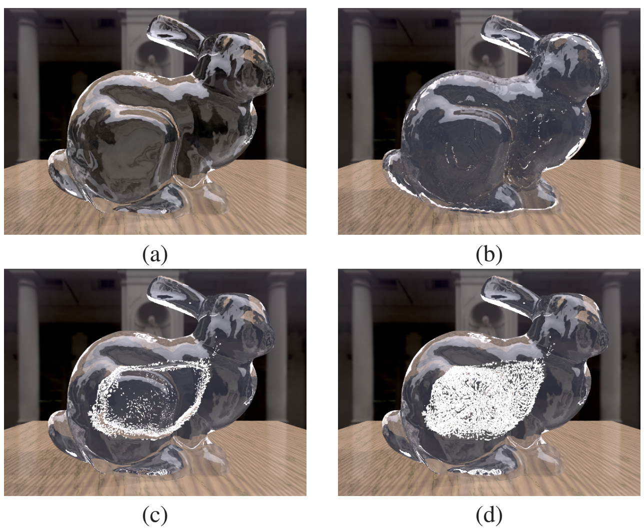

Figure 7: *Freezing simulation of water in the stanford bunny shape container.*

radii of air bubbles taking into account pressure and freezing speed of water-ice interface. Our method parallelized the simulation using CUDA and achieved interactive freezing simulation.

In future work, we would like to simulate the cracks in the ice to enhance the realism.

Acknowledgements

This research was partially supported by JST CREST.

References

BARI, S., AND HALLETT, J. 1974. Nucleation and growth of bubbles at an ice-water interface. *Journal of Glaciology*, 489–520.

BECKER, M., AND TESCHNER, M. 2007. Weakly compressible sph for free surface flows. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 209–217.

BUSARYEV, O., DEY, T. K., WANG, H., AND REN, Z. 2012. Animating bubble interactions in a liquid foam. *ACM Transactions on Graphics 31*, 4, 63:1–63:8.

CLEARY, P. W., PYO, S. H., PRAKASH, M., AND KOO, B. K. 2007. Bubbling and frothing liquids. *ACM Transactions on Graphics 26*, 3, 97:1–97:6.

ENRIGHT, D., FEDKIW, R., FERZIGER, J., AND MITCHELL, I. 2002. A hybrid particle level set method for improved interface capturing. *J. Comput. Phys. 183*, 1, 83–116.

HONG, J.-M., AND KIM, C.-H. 2003. Animation of bubbles in liquid. *Computer Graphics Forum 22*, 3, 253–262.

HONG, J.-M., LEE, H.-Y., YOON, J.-C., AND KIM, C.-H. 2008. Bubbles alive. *ACM Transactions on Graphics 27*, 3, 48:1–48:4.

KHARITONSKY, D., AND GONCZAROWSKI, J. 1993. A physically based model for icicle growth. *The Visual Computer 10*, 2, 88–100.

KIM, T., AND LIN, M. 2003. Visual simulation of ice crystal growth. In *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation 2003*, 86–97.

KIM, T., HENSON, M., AND LIN, M. 2004. A hybrid algorithm for modeling ice formation. In *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation 2004*, 305–314.

KIM, T., ADALSTEINSSON, D., AND LIN, M. 2006. Modeling ice dynamics as a thin-film stefan problem. In *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation 2006*, 167–176.

MADRAZO, C., AND OKADA, M. 2008. Physically based modeling of ice with bubbles. In *Eurographics 2008 short paper*, 13–16.

MADRAZO, C., TSUCHIYA, T., SAWANO, H., AND KOYANAGI, K. 2009. Air bubbles in ice by simulating freezing phenomena. *The Journal of the Society for Art and Science 8*, 1, 35–42.

MAENO, N. 1966. Air bubbles in ice, nucleated and trapped at an ice-water interface. *Low temperature science. Series A, Physical sciences 24*, 91–109.

NATIONAL ASTRONOMICAL OBSERVATORY OF JAPAN, Ed. 2012. *Chroronological Scientific Tables*. MARUZEN Co., Ltd.

SEIPEL, S., AND NIVFORS, A. 2006. Efficient rendering of multiple refractions and reflections in natural objects. In *Proceedings of the Annual SIGRAD Conference*, 1–6.

THÜREY, N., SADLO, F., SCHIRM, S., MÜLLER-FISCHER, M., AND GROSS, M. 2007. Real-time simulations of bubbles and foam within a shallow water framework. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 191–198.

YOSHIMURA, K., INADA, T., AND KOYAMA, S. 2007. Growth of gas bubbles and columns at ice-water interface at high pressure. Tech. rep., Fukuoka Industrial Technology Center.

ZHENG, W., YONG, J.-H., AND PAUL, J.-C. 2006. Simulation of bubbles. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 325–333.