Visual Simulation of Glazed Frost Considering the Supercooled State

Tomokazu Ishikawa[*] *Yonghao Yue*[†] *Taichi Watanabe*[*] *Kei Iwasaki*[‡]
Yoshinori Dobashi[§] *Masanori Kakimoto*[*] *Kunio Kondo*[*] *Tomoyuki Nishita*[¶]

[*]Tokyo University of Technology [†]Columbia University (JSPS) [‡]Wakayama University/UEI Research
[§]Hokkaido University/UEI Research [¶]UEI Research/Hiroshima Shudo University

ABSTRACT

We propose a method for simulating glazed frost by computing heat transfers between water droplets and the surrounding air. Motions of air and water droplets are computed based on a FLIP (Fluid-Implicit-Particle)-based fluid solver. Glazed frost is a crystal clear ice and formed from supercooled raindrops that freeze when they hit object surfaces. We propose a method to create an animation of glazed frost formation by taking into account the heat transfer between particles and the outside grids. The time to freeze is calculated by considering the heat flux on the surface of the raindrop until freezing from the adhesion on the obstacles. We propose a technique which is highly compatible for a FLIP method to solve the heat conduction equation and we reproduce the formation of glazed frost.

1. INTRODUCTION

Glazed frost is a cluster of clear ice crystals and formed from supercooled raindrops that freeze when they hit object surfaces such as the ground and branches of trees (see Fig. 1). Simulation methods for formation of ice crystal, such as frost, on the surface of objects have been proposed by Kim et al. [1]. However, a supercooled state has to be considered for simulating freezing rain, and fluid simulation is required for reproducing the effect of raindrops running down on the ice surfaces. To our best knowledge, there has been no research presenting glazed frost by using a fluid simulation. We use the fluid simulation based on a FLIP method [2]. Hence, raindrops and obstacles are represented by particles which are used to calculate the advection term, and the update of velocity field is calculated by using grids except for advection term. We propose a method to create an animation of glazed frost formation by taking into account the heat transfer between particles and the grid.

2. RELATED WORK

In the computer graphics (CG) field, research to represent ice and snow began long ago. Kim et al. proposed a method for generating the crystal geometry of ice by solving a heat conduction equation in a two-dimensional plane [1]. The result of Kim et al. makes it clear that heat conduction plays an important part in creating ice crystals.

(a) Branches attached by glazed frost.

(b) Glazed frost attached on one side of branches.

Fig. 1: Glazed frost (photograph).

Takahashi and Fujishiro proposed a method for expressing a state of compacted snow [3]. In this method, they calculated an expression behavior of snow attaching to surface of object by introducing an effect of sintering into SPH (Smoothed Particle Method). In this paper, we propose to reproduce an icing phenomenon by calculating the ice buildup as an aggregate of particles and introducing a sticking power term into the equation.

Iwasaki et al. proposed a method to calculate ice-to-ice interaction at high speeds using SPH [4]. They showed that ice can be simulated as a rigid body at high speed using a GPU. However, although this method can express a process in which an icicle forms or water freezes in a blink, the supercooling phenomenon is not taken into consideration.

In our research, we adopt a FLIP method for representing a fluid phenomenon. The effectiveness and feasibility of the FLIP method in the field of CG has been suggested by Zhu et al. [2]. Recently, it has been actively improved for use in animation, and it is broadly utilized in various industries for reproducing a fluid state such as water or flame.

Numerical calculation of supercooled water has also been executed in the field of computational physics. In addition, a report has verified the thermal ambience of air against the actually observed glazed frost phenomenon [6]. In this paper, we confirm that temperature, humidity, wind velocity, and precipitation amount create the conditions for developing glazed frost. In

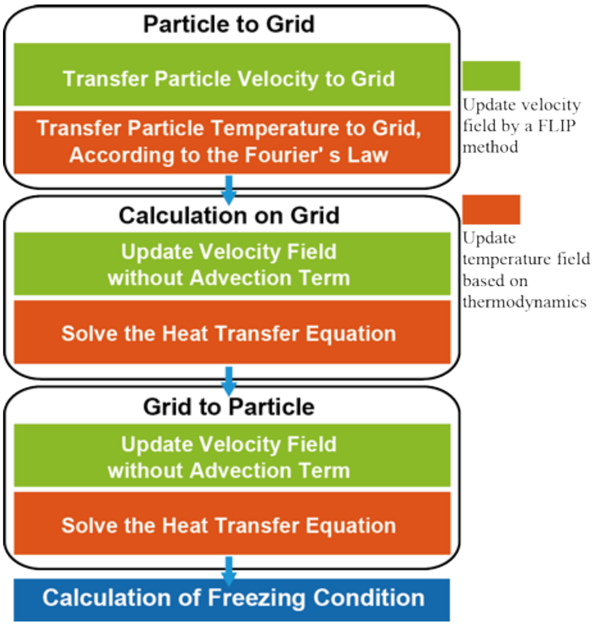


Fig. 2: Process flow in each timestep.

accordance with these facts, we propose a feasible model to realize a visual simulation of glazed frost.

3. PROPOSED SIMULATION METHOD

In this section, we explain the modeling of glazed frost and calculation of heat transfer using a FLIP method.

3.1. Fluid Calculation of Rainwater

The fluid behavior of rainwater is calculated by the Navier–Stokes equations. We solve the equations using the FLIP method, a hybrid method to solve the following Navier–Stokes equations using a grid and particles:

$$\nabla \cdot \mathbf{u} = 0, \tag{1}$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho}\nabla p + \mu \nabla^2 \mathbf{u} + \mathbf{F}, \tag{2}$$

where $\mathbf{u}$ is velocity, t is time, ρ is density, p is pressure, μ is dynamic coefficient of viscosity, and $\mathbf{F}$ is external force such as wind and adhesion by surface tension (details are described in Sec 3.3). In the FLIP method, updating the velocity field by the external force term and pressure term is calculated using a grid method, and the advection term is calculated using a particle method.

In our research, rainwater is represented as particles. Velocity v_i, temperature $T_{p,i}$, and state S_i are stored in the center of the i-th particle with a radius r_d. Particles entering the simulation space from the top represent raindrops. The size of the raindrop is determined by the following Marshall–Palmer distribution [5].

$$N(D) = N_0 \exp(-\lambda D), \tag{3}$$

where D is the size of the raindrops, and N_0 and λ are the initial and gradient parameters, respectively. A raindrop with particle size D is approximated by an appropriate number of particles that can be equal to the volume of a raindrop. That is, we determine a number of particles for raindrops of particle size D to a maximum of natural number n that satisfies the following nonstrict inequation:

$$n \leq \frac{D^3}{8{r_d}^3}. \tag{4}$$

The raindrops of small size D are not considered when the natural number n satisfying the above nonstrict inequation does not exist.

Water particles in a supercooled state are determined to be solidified or not by considering the amount of heat. Some water particles get solidified at the time of impact with an obstacle. Supercooled water that is not solidified at the time of impact can flow down. We also process both changing of ice to water and water to ice by ambient air temperature. Sublimation (changing from water vapor to ice) can result in ice crystal formation, but we do not consider it since the amount of ice crystal formation by sublimation is quite small. We represent objects covered by glazed frost (e.g., plants and branches) with particles that are referred to as object particles. The only particles updated by the FLIP method are the water particles; object particles and accreted ice particles are set as fixed boundaries in fluid calculation. Temperature should be maintained even for object particles, and then we calculate a heat transfer with the ambient air. Furthermore, we do not include any computation regarding a process for creating supercooled water, but we start a simulation in which all rain falling in the simulation space is considered to be in a supercooled state.

3.2. Heat Transfer in a FLIP Method

We propose a method to calculate heat conduction that is suitable for the FLIP method. The process flow during a single timestep is shown in Fig. 2.

We distribute velocity from particle to lattice in the first step of FLIP method. Heat is distributed from particle to grid as the velocity field is distributed. Heat transfer is not conducted between particles, but instead is calculated between lattices for computational efficiency. Heat is distributed from grid to particles after the heat transfer calculation. In addition, adhesion is calculated after supercooled water collides with object particles. The time from adhesion to a collided object's surface to being frozen is calculated with consideration of heat flux for glazed frost surface.

The heat distribution between particles and grid is calculated according to Fourier's law:

$$\Delta Q_i = -k_i A \frac{dT_i}{dx} = -k_i A \frac{T_{air} - T_{p.i}}{r_d}, \tag{5}$$

where ΔQ_i is heat quantity transferring between the i-th particle and grid including the particle, k_i is thermal conductivity, A is surface area of the particle, dx is the infinitesimal distance, dT_i is the temperature difference between the particle and ambient air, and T_{air} is temperature at the grid point near the particle. The temperature of each particle is updated by the heat quantity,

$$\Delta T_i = \frac{\Delta Q_i}{m_i c_i}, \tag{6}$$

where m_i is mass of particle i and c_i is specific heat of particle. Mass and specific heat of particle are set to different values depending on whether the particle represents a water droplet or an obstacle.

Heat transfer is calculated on the grid by solving heat transfer equations.

$$\frac{\partial T}{\partial t} = a\frac{\partial^2 T}{\partial \mathbf{x}^2}, \tag{7}$$

where a is thermal diffusivity coefficient and $\mathbf{x}$ is a position vector. After calculating heat transfer, the heat is distributed to the particles. At update of the velocity field, a timestep is set according to the Courant–Friedrichs–Lewy conditions. To use this timestep in the numerical solution of heat transfer equations, we employ the Crank–Nicolson method, an implicit scheme that can stably solve at arbitrary time intervals, to handle the heat transfer equation.

3.3. Calculation of adhesion force

For describing an ice accretion when rain in the supercooled state collides with an object such as plant or with already-formed glazed frost, we calculate a surface tension and interfacial tension for object surface. We take into account the surface tension and interfacial tension as external forces in Equation (2). These forces are calculated by the following equation.

$$\mathbf{F}_i^{sticking} = \sum_{j \in N_i} \kappa_j \frac{\mathbf{p}_j - \mathbf{p}_i}{\left\|\mathbf{p}_j - \mathbf{p}_i\right\|^2}, \tag{8}$$

where $\mathbf{p}_i$ is position of particle i and κ_i is coefficient of adhesion force. N_i is the set of particles which exists in a lattice with particle i and 6 lattices adjacent to it, and the adhesion should be considered with an influence from particle j with the accumulation. We improve the calculation efficiency due to collision detection limited in N_i.

3.4. Freezing Condition

The freezing time is calculated by considering the heat flux on the surface of the raindrop until it freezes from adhesion to the obstacles. The freezing speed is calculated by the following sum of heat flux:

$$Q = Q_s + Q_l + Q_f, \tag{9}$$

where Q_s is a sensible heat flux, Q_l is a latent heat flux, and Q_f is the heat flux necessary to freeze a water film. These quantities are calculated using the following equations:

$$Q_s = -\pi h_a \Delta T, \tag{10}$$

$$Q_l = -\pi L_e h_v \Delta \rho_v, \tag{11}$$

$$Q_f = L_f w, \tag{12}$$

where h_a is a heat exchange coefficient of air in Equation (10). In Equation (11), h_v is a water vapor exchange coefficient, L_e is evaporative latent heat, and $\Delta\rho_v$ is difference of water vapor concentration. In Equation (12),

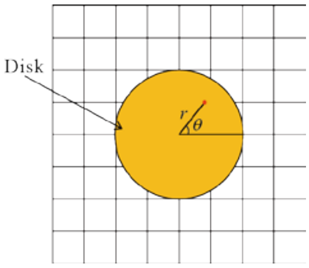


Fig. 3: Simulation space for verification experiment

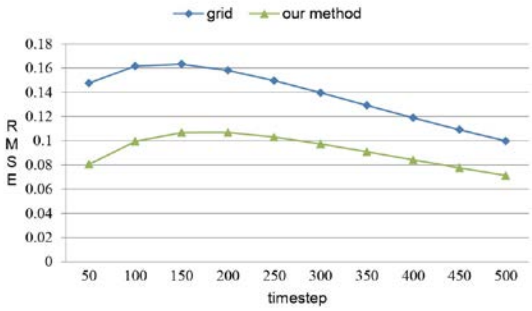


Fig. 4: Evaluation by RMSE between exact solution and calculation result.

Table 1: Spatial discretization and computation time.

	grid method	our method
number of grid	50× 50	50× 50
number of particles	---	1,000
computation time for one timestep	6.0 ms	8.2 ms

L_f is a latent heat for freezing and precipitation flux, and w is calculated by the following equation using the relationship of rain rate R and wind speed V.

$$w = \left\{ (R/3.6)^2 + \left(0.067R^{0.846}V\right)^2 \right\}^{\frac{1}{2}} \times 10^{-3}. \tag{13}$$

It is known that the smaller the sum of flux Q in negative value, the faster the freezing speed [6]. So we use the time integrals Q_s and Q_l and the required amount of heat H_f as the condition to determine whether the water freezes.

$$H_f \le \int \left| Q_s + Q_l \right| dt. \tag{14}$$

Integration calculation is limited to the time the water is attached to an obstacle or other glazed frost. The water does not freeze if it drifts by other raindrops and is affected by gravity or is not attached to the surface of glaze frost.

4. RESULTS AND CONCLUSION

For the simulation, we used C++ and the nVIDIA CUDA on a standard PC (CPU: Intel Core i7 3.2 GHz, RAM: 16.0 GB, GPU: nVIDIA GeForce GTX 580). For rendering the results, we used Autodesk 3ds Max.

First, we conducted experiments to verify the calculation of heat transfer by the proposed method. We calculated the error of numerical calculation in cases that we can exactly solve a heat conduction in two dimensions, such as the following: A circular object is located at the center of the simulation space, the transferred heat between the outside air and the object is calculated (see Fig. 3), and then the temporal change of the temperature is calculated. The exact solution at time t in polar coordinates (r, θ) on the circle in this case is calculated with the following formula:

$$T(r,\theta,t)=\sum_{n=0}^{\infty}\sum_{m=1}^{\infty}e^{-\mu_{m,n}t}J_n\left(\mu_{m,n}r\right)\left(A_{mn}\cos n\theta+B_{mn}\sin n\theta\right), \tag{15}$$

where J_n is n-th Bessel function and $\mu_{m,n}$ is m-th value from the smallest positive zero point of J_n. The proposed method discretizes the simulation space with particles and a grid. Table 1 compares space discretization and calculation time for both methods in the case of the temperature value with the exact solutionIn addition, the comparison of the RMSE (root-mean-square error) between the exact solution and our calculation result is shown in Fig. 4. The results confirm that the proposed method is comparably as stable as the numeric calculation with a grid method, and it can compute with a small error. Since the error in the exact solution is especially significant near the boundary, the boundary condition of the proposed method needs more consideration.

In Fig. 5, the brown particles represent tree branches and the deep blue particles represent ice (glaze). The number of particles used for the simulation is almost 20,000, including the objects such as fluid and branches. The grid resolution of the simulation space is 128 × 128 × 128, and the calculation time for 1 timestep is 20 msec. We show parameters used in our simulation in Table 2.

Fig. 6 shows rendering results produced by the mental ray rendering engine used with 3ds Max. The surfaces of the glazed frost are calculated by assigning particles representing glazes to anisotropic kernel density distributions proposed by Yu et al. [7] and extracting isosurfaces. The surface extraction took 30 seconds on average, and the rendering took almost 3 minutes for 1 frame.

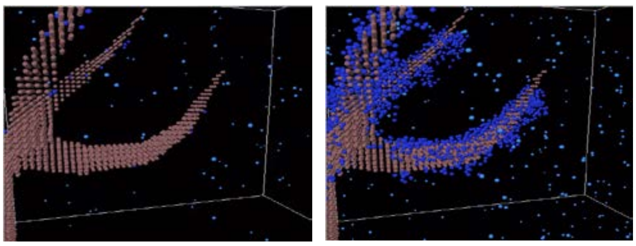
Supercooled water in some cases freezes instantly after collision, and in some cases gradually condenses in the process of running down objects. This effect can be expressed by calculating the thermal environment, sensible heat, and latent heat. Our method can reproduce the ice covering a branch.

By changing the parameter of the thermal environment, the situation was simulated that when the environmental temperature is higher than 0°C, water did not adhere at the top part of a branch and instead froze while it was flowing down (Fig. 7). Because the supercooled state is considered, the icing when colliding with the object surface and the fluid phenomenon of the change from ice to water can be calculated to the formation of icicles. Moreover, external forces such as wind can be considered using the FLIP method. The velocity field of the simulation space was defined where the wind blew from the right side of the image. The state in which more glaze is formed on the windward side of the tree branch can also be reproduced (Fig. 8).

We have proposed a technique that is highly compatible with a FLIP method to solve the heat conduction equation, and we used it to reproduce formation of glazed frost. By considering the supercooled conditions, our method can simulate formation of icicles by calculating ice buildup after collision with obstacles and the fluid phenomenon of changing of ice to water.

Table 2: Parameters in our simulation.

parameter	meaning	value	unit
dt	timestep	1.0	s
ρ	density	1.0×10^{3}	kg/m^3
μ	dynamic coefficient of viscosity	1.792×10^{-6}	m^2/s
N_0	initial value of Marshall Palmer distribution	8.0×10^{3}	mm^{-1}m^{-3}
λ	gradient of Marshall Palmer distribution	0.1192×10^{3}	m^{-1}
r_d	radius of particle	1.0×10^{-3}	m
k	thermal conductivity	0.569	W/(m・K)
c	specific heat	4217	J/(kg・K)
a	thermal diffusivity coefficient	0.1349×10^{-6}	m^2/s
κ	coefficient of adhesion force (liquid)	0.5	N
κ	coefficient of adhesion force (solid)	2.0	N
h_a	heat exchange coefficient of air	5.69	W/(m^2・K)
h_v	water vapor exchange coefficient	0.05	m/s
L_e	evaporative latent heat	2.498	J/kg
$\Delta\rho_v$	difference of water vapor concentration	1.61×10^{-3}	kg/m^3
L_f	latent heat for freezing	333.4×10^{3}	J/kg
R	precipitation intensity	25×10^{-3}	m/h
H_f	required heat quantity for freezing	333.4	J/m^3



(a) 10 timesteps (b) 7,000 timesteps

Fig. 5: Simulation results of generating glazed frost.

(a) 1,000 timesteps

(b) 4,000 timesteps

(c) 7,000 timesteps

(d) 10,000 timesteps

Fig. 6: Simulation and rendering results of generation of glazed frost.

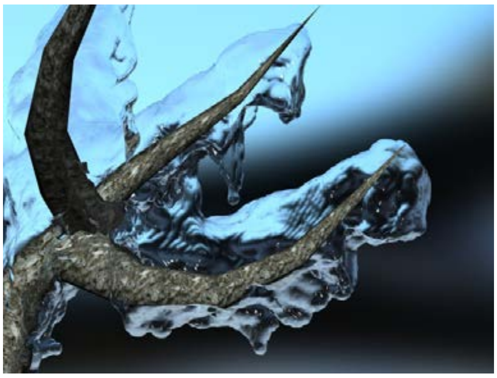


Fig. 7: Simulation result of formation of icicle shapes.

Fig. 8: Simulation result of glazed frost adhering on the windward side by the wind from right to left.

REFERENCES

[1] T. Kim, M. Henson and M. C. Lin, “A Hybrid Algorithm for Modeling Ice Formation”, Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp.305–314, 2004.

[2] Y. Zhu and R. Bridson, "Animating Sand as a Fluid", Proceedings of the ACM SIGGRAPH 2005, pp.965-072, 2005.

[3] T. Takahashi and I. Fujishiro, “Particle-based simulation of snow trampling taking sintering effect into account”, ACM SIGGRAPH 2012 Posters, No. 3, 2012.

[4] K. Iwasaki, H. Uchida, Y. Dobashi, T. Nishita, "Fast Particle-based Visual Simulation of Ice Melting", Computer Graphics Forum, Vol. 29, pp.2215-2223, 2010.

[5] J.S. Marshall and W.M. Palmer, "The distribution of raindrops with size," Journal of Meteorology, Vol. 5, pp. 165-166, 1948.

[6] K. F. Jones, "Ice accretion in freezing rain", Tech. rep., DTIC Document, 1996.

[7] J. Yu, G. Turk, "Reconstructing Surfaces of Particle-Based Fluids Using Anisotropic Kernels", Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp.217-225, 2010.

APPENDIX

A. COEFFICIENTS OF THE EXACT SOLUTION IN THE THERMAL DIFFUSION EQUATION OF THE DISK AREA

In Equation (15), J_n is n-th Bessel function and defined by the following equation,

$$J_n\left(z\right)=\sum_{k=0}^{\infty}\frac{\left(-1\right)^k}{k!\left(n+k\right)!}\left(\frac{z}{2}\right)^{n+2k}. \tag{A1}$$

Parameters in Equation (15) are the constants determined by the following equations.

$$A_{mn}=\frac{2}{\pi J_{n+1}\left(\mu_{m,n}\right)^2}\int_0^1\left(\int_{-\pi}^{\pi}T\left(r,\theta,0\right)J_n\left(\mu_{m,n}r\right)\cos n\theta d\theta\right)rdr, \tag{A2}$$

$$B_{mn}=\frac{2}{\pi J_{n+1}\left(\mu_{m,n}\right)^2}\int_0^1\left(\int_{-\pi}^{\pi}T\left(r,\theta,0\right)J_n\left(\mu_{m,n}r\right)\sin n\theta d\theta\right)rdr. \tag{A3}$$