2023 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
MODELING SIMULATION AND SOFTWARE (MS2) TECHNICAL SESSION
AUGUST 15-17, 2023 - NOVI, MICHIGAN

DYNAMIC SNOWFALL SCENE SIMULATIONS FOR AUTONOMOUS VEHICLE SENSOR PERFORMANCE

Sergey N. Vecherin, Molly E. Tedesche, Michael W. Parker

U. S. Army ERDC - Cold Regions Research and Engineering Laboratory, Hanover, NH

ABSTRACT

Cold regions are becoming increasingly more important for off-road vehicle mobility, including autonomous navigation. Most of the time, these regions are covered by snow, and vehicles are forced to operate under active snowfall conditions. In such scenarios, realistic and effective models to predict performance of on-board sensors during snowfalls become of paramount importance. This paper describes a stochastic approach for two-dimensional numerical simulation of dynamic snow scenes that eventually will be used for driving condition visualization and vehicle sensor performance predictions. The model captures realistic snow particle size distribution, terminal near-surface particle speeds, and adequately describes interactions with wind.

Citation: S. N. Vecherin, M. E. Tedesche, M. W. Parker, "Dynamic Snowfall Scene Simulations for Autonomous Vehicle Sensor Performance", In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 15-17, 2023.

1. INTRODUCTION

Off-road autonomous vehicles are currently being considered to support US military defense operations in the Arctic, as well as in other northern areas of the world. In these cold climate regions, snowfall contributes a substantial portion of annual precipitation, while snow and ice persist as ground cover for much of the year. Autonomous vehicles, such as those being studied for US defense purposes, can be fully or partially unmanned, and in some cases, may perform better than a human driver by rapidly sensing and reacting to terrain changes [1]. Onboard technology can be used for look-ahead sensing to classify terrain in front of the vehicle, which informs autonomous controllers to adapt for upcoming driving conditions and optimize route finding [2].

Vehicle mounted sensors, such as Light Detection and Ranging (LiDAR), that uses laser pulses to measure variable surface heights, can be effective tools for autonomous vehicle operation. However, route finding for off-road vehicle autonomy is extremely challenging in cold regions

DISTRIBUTION A. Approved for public release; distribution unlimited. OPSEC #: CRREL 03/13/2023-1.

dominated by snow and ice cover, where surface textures vary greatly due to rapidly changing conditions [3]. The sensor performance worsens further during an active snowfall event. Therefore, dynamic winter weather simulations are key to advancing the development, testing, and use of autonomous vehicles in the Arctic and other cold climates. A dynamic winter weather simulation capability will allow a multitude of simulation environments to be generated including scenarios that would otherwise be challenging to encounter in nature. This coupled with high fidelity sensor simulations will allow scientists to study how these winter scenarios affect sensors and the subsequent autonomous operation of the optionally manned or unmanned ground vehicle.

Modeling snow for dynamic winter weather simulations can be categorized into three phases, including: (1) generation of snowflakes in the atmosphere, (2) mechanics and visualization of falling particles, and (3) snowpack accumulation and metamorphism. In the atmospheric phase of snow modeling, the shape and formation of individual snowflakes, comprised of clusters of ice crystals, are dependent on meteorology [4]. In the propagation phase, falling snow can be visualized using either a conceptual approach [5] or more involved particle system techniques [6]. In the final phase, snow accumulation patterns are affected by interactions with, or interception by, the ground surface or other obstacles [7].

In this study, we explore the second component of snow modeling, dynamic snowfall propagation, as a key component of winter scene simulations used for estimating autonomous vehicle onboard sensor performance. Specifically, we model snow particle size distribution across two dimensions (2D), as the first step of a simulation to be scaled into three dimensions (3D). Additionally, our snow particle size distribution model is the baseline component of an ongoing snowfall scene simulation project that includes realistic snowflake fall velocity, crystal type, and axial ratio or fall orientation in order to increase simulation fidelity, if desired.

In terms of simulating dynamic snowfall events and environment interactions with autonomous vehicle sensors, there are four basic techniques. These include: (1) "off the shelf" or video gaming approaches like in Unreal Engine 4 (UE4), (2) physics based models (high fidelity simulations), (3) statistical or stochastic approaches modeling effects of key parameters without the complex particle interactions, and (4) empirical sensor performance degradation models (Figure 1).

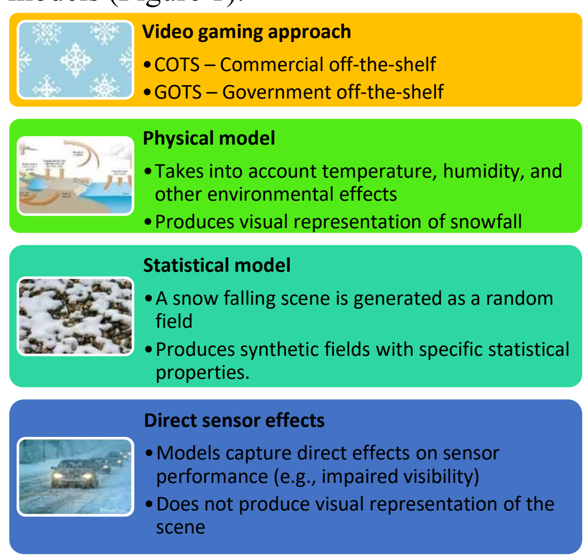

Figure 1: Potential dynamic snowfall simulation approaches.

In the video game approach, interactive unmanned vehicle simulators leverage open source gaming technology for near real-time virtual autonomous navigation environments. This method uses multimedia visualization techniques from the commercial video-game industry to run commercial off-the-shelf (COTS) or governmental off-the-shelf (GOTS) hardware [8].

Physical snowfall models simulate the physical processes of snow crystal formation, snowflake aggregation, snowfall, and accumulation, by considering meteorological parameters and energy transfer [9, 10]. Physical models typically provide the best fidelity, but also require enormous computational resources and time, which may not be practical for real-time simulations aboard autonomous vehicles.

Empirical snowfall models utilize direct simulation of sensor performance as a function of environmental conditions, and where each sensor technology is considered separately [11]. This could include LiDAR, infrared, hyperspectral, and visual methods. While the direct measurement approach is computationally efficient, it may be subject to biases specific to an experiment, such as formations of snowflake type or size which where specific to experiment location or time.

In the stochastic approach, statistical data for various snowfall characteristics are generated for snowflake formation processes. This approach provides an alternative to the computationally heavy task of modeling snow particle microphysics and energy transfer, and can be calibrated to visualize realistic, naturally occurring snowfall scenes in a more efficient manner [12]. Ultimately, results of this work will be used in the Engineer Research and Development Center (ERDC) environmental sensing engine (ESE) software, where there are high fidelity sensor models specifically designed for interacting with environmental conditions. This realistic interaction between the sensor and environment is critical for development of autonomous vehicle navigation algorithms.

2. METHOD

This section describes the current efforts of the modeling that is focused on the two-dimensional simulation. This can be visualized as an observer (or a sensor) watching a snowfall from a window (e.g., a driver viewing thought a windshield while driving during a snowfall). The stochastic approach described in this paper strikes a balance between numerical efficiency (no high-performance computing is required) and realism and is based on a few key considerations.

First, complex microphysics of forming each individual snowflake is omitted. Instead, snowflakes are taken as an ensemble of snow particles already existing, and keeping coming, near the ground surface. The simulation architecture allows for specifying different snowflake types, both regular and irregular shapes, such as solid or thin plates, needles, prisms and, dendrites. However, to simplify the prediction of lidar or radar scattering and for the sake of proving the concept, the examples in this paper consider only regular spheres that have random diameters distributed in accordance with an experimentally determined probability density function (PDF).

Second, complex physics of snowflake propagation from the upper atmosphere to the ground, including possible upward motions due to upward thermal drafts and other atmospheric effects leading to formation of specific snowflake shape and size are also omitted. Such an approach is considered in [13] and [14] and leads to heavy computations using HPC for days or weeks even for moderate spatial extensions of a simulation scene. Instead, as was observed in [13] in their high-fidelity HPC computations, all particles reach constant terminal velocities in the vicinity of the ground. The terminal velocities were empirically found to relate to the particle size [15]. Therefore, in our stochastic approach, the particles appear at the top line of a screen in accordance with the Poisson distribution.

Third, to realistically model wind effects, our model does not attempt to solve the flow dynamics equations (non-linear Navier-Stokes), as in [16], which requires HPC. Instead, a wind vector (potentially including a fully heterogeneous 2D wind filed) is added to the terminal particle velocities near the surface directed vertically downwards. After that, their position in the screen is updated at every time step using simple kinematic rules.

Figure 2 summarizes the described steps.

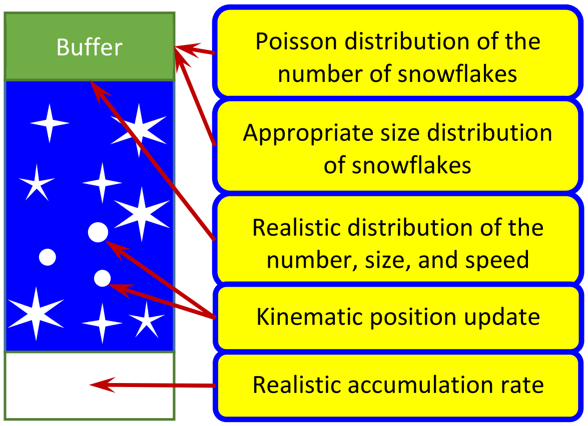

Figure 2: Logical chart of the proposed stochastic approach.

The model starts by creating a specified number of particles, N, in the time buffer containing particles before they reach a visible modeling screen. The particles are uniformly distributed over duration of the buffer and the screen spatial aperture taking into account snowflake rate, f [particles/m/s]. Such a realization leads to the Poisson distribution of particles appearance in time on the first line of the visible area (the top of the blue area in Figure 2). More precisely, the probability P that k particles will appear during a time interval Δt over the simulation screen x-aperture l_x is given by

$$P(n = k) = \frac{\lambda^k}{k!} e^{-\lambda}, \qquad (1)$$

where the expected number of particles λ is proportional to the snowflake rate f:

$$\lambda = f l_x \Delta t. \qquad (2)$$

Should a particle appear on the top of the simulation screen, its location along the aperture length is chosen in accordance with the random uniform distribution. Standard algorithms for uniform random generator use pseudo-random sequences that are not ideally random. As a result, the full screen aperture length may not be uniformly sampled by a limited number of particles, allowing some unrealistic voids. To improve the efficiency of the sampling, in this paper, the Sobol's sequence [17], which is additionally scrambled in order [18, 19].

To realistically simulate snowfall particles, the PDF of their size distribution should be specified. The existing literature describes two schools of thought on this topic. The first one is concerned with the mass and energy transportation during snowfalls and, therefore, they study the diameter of water droplets that have the equivalent water content as a snowflake or the size of ice crystals during blizzards [16, 20, 21]. According to their studies, the size of the particles can be described by the gamma PDF with the most prevailing size of the particles about 0.2 mm. The second one is focused on measuring the actual size of snowflakes during a snowfall, as it appears to an observer on the ground, using modern distrometers [15, 22, 23]. In these studies, it appears that the falling particle size can also be described by the gamma distribution, but with predominant particle size of about 2 mm, with some instances encountering particles as large as 8 mm in diameter. This second school of thought is more pertinent to our research goal, which is to simulate realistic snowfall scenes for visual perception and sensor interaction. Therefore, the gamma PDF $\Gamma(a, b)$ with the shape parameter $a = 2$ and the scale parameter $b = 1$, shown in Figure 3, for the particle diameter size was chosen to fit their findings.

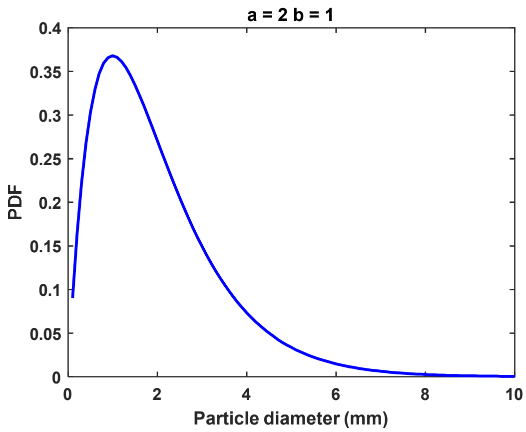

Figure 3: Gamma PDF of falling particle diameter used in simulation.

To create *N* particles in the buffer with the diameters described by the gamma distribution, we generate *N* random numbers z uniformly distributed in the [0, 1] range first, following the procedure described above using scrambled Sobol's sequence. Then, we apply the inverse cumulative gamma distribution to generate the diameters d in accordance to the chosen PDF [24]:

$$d = \mathrm{invCDF}_{\Gamma(a,b)}(z). \tag{3}$$

The next step in the model is to assign the terminal speed to the particles. As was well established in prior research [15, 22, 23, 25], the observed terminal speed of snowflakes relates to their diameter. It was shown that the median terminal speed in the vicinity of the ground surface weakly depends on snowflake type and varies in the range [1, 2] m/s [15]. Specifically, the median terminal speeds in our model relate to their diameter by the power law:

$$v = 0.84d^{0.36}, \tag{4}$$

where the particle speed is in m/s and the constants are found empirically in [15]. According to Table 3 in [15], this power law fits majority of experimental observations for all particle types better than the alternatively considered saturated exponential function.

Once the terminal speeds of falling particles are known, we proceed with the wind interaction. It is assumed that the wind can be imposed as an independent filed on the particle velocities that, otherwise, would fall straight downwards. From this perspective, for each time step t_i, one needs to add the current particle velocity vector with the wind vector and update particle positions. This can be described by the following procedure:

$$\begin{aligned} &t_i = i\Delta t, \qquad i = 0,1,\dots,I \\ &\mathbf{u}_{i+1} = \mathbf{u}_0 + \mathbf{w}_i,\ \mathbf{u}_0 = (0, v), \\ &\mathbf{r}_{i+1} = \mathbf{r}_i + \Delta t \mathbf{u}_{i+1} \end{aligned} \tag{5}$$

where i is the time evolution index, t_i is the current time, Δt is the time increment, $\mathbf{u}_i = (u_{xi}, u_{yi})$ is the current 2D velocity of the particle in the Cartesian coordinates (*x, y*) with the *y*-axis pointing downwards (note that the initial velocity of a particle equals to the median terminal speed with zero *x*-component), $\mathbf{w}_i = (w_{xi}, w_{yi})$ is the current wind velocity, and $\mathbf{r}_i = (x_i, y_i)$ is the current location of a particle in the simulation screen. In order to accommodate the wind effects in the *x*-direction, the same procedure with the time buffer as described above was implemented relative to the *x*-axis.

3. SIMULATION RESULTS

This section provides examples of the theory described in the previous section with detailed notes on numerical realization.

For the simulation, the size of the modeling screen was chosen to be 1 m by 1 m in the Cartesian coordinates with the *y*-axis pointing downwards. The number of particles in the time buffer, wind speed, and whether or not to have size differentiation of the particles are user-specified parameters. For the examples considered below, the number of particles in the time buffer was set to $N = 100$ particles. The duration of the buffer, T_b, is set up to accommodate these *N* particles:

$$T\_b = Nl_x/f, \tag{6}$$

where l_x is the horizontal aperture of the simulation screen, i.e., the length of the *x*-axis, $l_x = 1$ m.

The time increment was chosen in such a way that the simulation smoothly reproduces flowing particles in the simulation screen:

$$\Delta t = l_y/150, \tag{7}$$

where l_y is the vertical aperture of the simulation screen, $l_y = 1$ m. The simulation duration was set to $T = 5$ s.

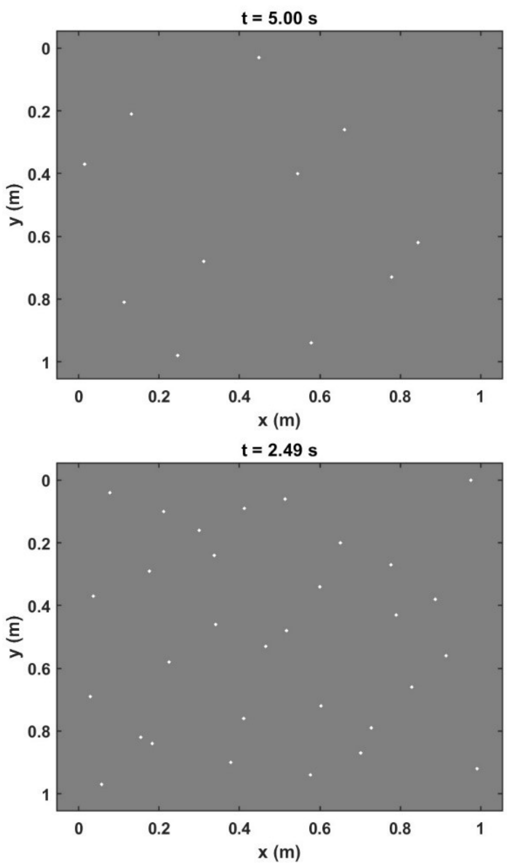

Figure 4: Selected frames from the simulated snowfall. Top: snowflake rate $\boldsymbol{f = 15}$ pp/m/s. Bottom: snowflake rate $\boldsymbol{f = 45}$ pp/m/s.

Figure 4 shows selected frames from the simulations with no wind and no particle size differentiation for two distinct snowflake rates of 15 and 45 particles per meter per second. As mentioned in Introduction, no particle type was specified, although the model architecture can easily accommodate this if needed.

Figure 5 shows examples of the simulations with distinct modeling parameters. The top plot depicts a single frame from the time evolving snowfall with the particle size distribution given by Equation (3). The physical diameters of the particles were recalculated in the marker size using a linear map. The middle plot shows the traces of the snow particles in 50 consecutive time frames of the simulation with particle size differentiation, but no speed differentiation. Instead, for the middle plot results, all particles were forced to have the same vertical terminal speed of 1.5 m/s. The wind was set to blow in the *x*-direction also with the constant speed of 1.5 m/s. Under such conditions, the traces of snow particles should represent straight lines, strictly aligned along the main diagonal of the simulation screen and strictly parallel to each other despite particle sizes, because all particles have the same speed in both *x*- and *y*-directions. The color of the traces in the middle and bottom plots represent time evolution. Blue and red colors indicate the first and last frames being depicted, respectively, that is, particles move from the upper left corner to the bottom right one. One can see that the character of particle traces in the middle plot matches exactly to the physical predictions. Finally, the bottom plot depicts results for the same set up as the middle plot except with particle speed differentiation turned on in accordance with Equation (4). In this case, larger particles (thicker traces) have higher terminal speeds along the *y*-axis, and, therefore, their traces should have steeper slopes than those of smaller particles (thinner traces). The bottom plot validates these observations.

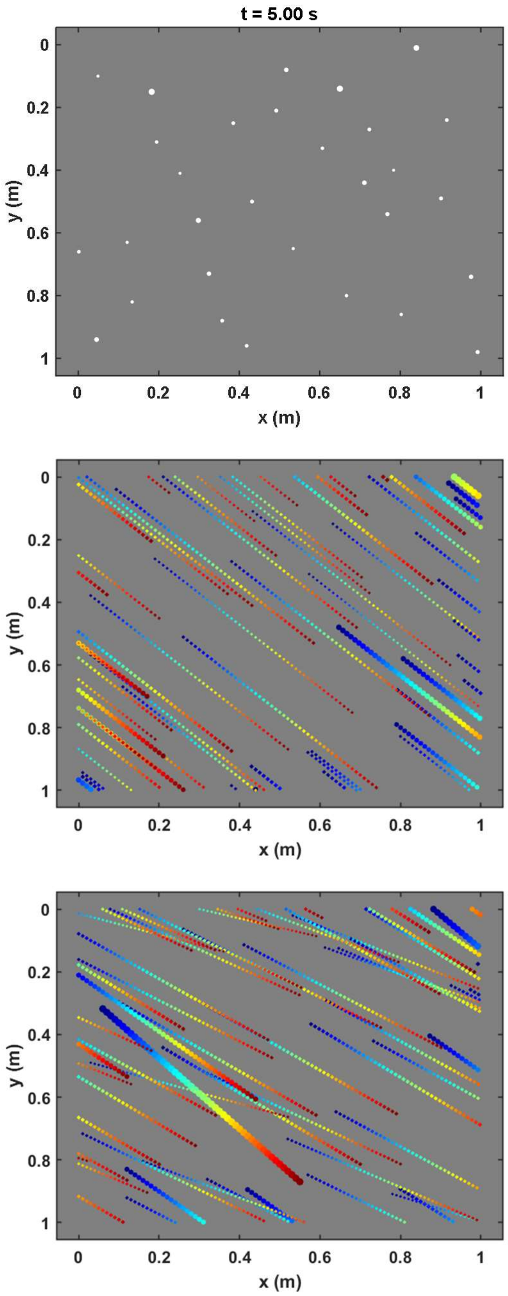

Figure 5: Snowfall models. Top: a snapshot of snowfall with particle size differentiation. Middle: last 50 frames of snow fall with the constant horizontal wind $\boldsymbol{w_x} = \mathbf{1.5}$ m/s and no particle differentiation by speed. Bottom: the same as the middle but with the differentiation of particles by their terminal speed.

4. CONCLUSIONS

Computer simulations are being used increasingly wider for simulation of adverse weather conditions for vehicle mobility. This is true for both, human-operated (road visibility) and autonomous navigation (sensor performance). As such, a virtual environment becomes a popular solution for such a purpose. One of the most challenging conditions for vehicle navigation is the snow weather condition in general and a snowfall in particular. For realistic predictions of road visibility and sensor payload operation, effective simulation tools are needed to reproduce snowfalls. A few approaches exist to this end, but some of them are extremely high fidelity and require HPC even for modest simulation scenes. Some others, having excellent technical capabilities from the software engineering point of view, lacking connection to characteristic snowfall parameters adequately describing the nature. In this paper, we describe a stochastic model that strikes a golden middle. On the one hand, it does not attempt to solve dynamic equations for the particles and the wind flow in real time, replacing this stage by physical insights and statistical information obtained from empirical data. On the other hand, it is realistic enough to reproduce naturally occurring snowflake rates, particle size distribution, particle speed distribution, snow accumulation rate, and can handle wind interactions with sufficient fidelity to provide a representative environment for driving simulators and sensor performance virtual testbeds. The model is scalable to the three-dimensional case. In that case, the only practical limitation of the model will be the memory size for all particles. The actual computational load of the model is rather modest. Future efforts will be devoted to develop a 3D model and integrate it with the sensor simulation and testing software.

5. ACKNOWLEDGEMENTS

This study was conducted for the U.S. Army Corps of Engineers under Autonomous Vehicle-Terrain Interactions PE 622145 / Project BG2 / Task 03, "Dynamic Winter Weather in Real-Time M&S." The permission to publish has been granted by the Director, Cold Regions Research and Engineering Laboratory.

6. REFERENCES

[1] M. Parker *et al.*, "Exploring cold regions autonomous operations," *Journal of Terramechanics,* vol. 96, pp. 159-165, 2021.

[2] O. Welling, A. Meyer, S. Vecherin, and M. Parker, "Determining Where a Vehicle Can and Cannot Safely Drive," in *Ground Vehicle Systems Engineering and Technology Symposium (GVSETS), NDIA*, Novi, MI, 2021, pp. 1-12.

[3] S. Vecherin, A. Meyer, B. Guinn, T. Letcher, and M. Parker, "Simulation of Snow Roughness for Autonomous Vehicle Numerical Modeling," presented at the Ground Vehicle Systems Engineering and Technology Symposium (GVSETS), Novi, Mi, 2022.

[4] K. G. Libbrecht, "The physics of snow crystals," *Reports on progress in physics,* vol. 68, no. 4, p. 855, 2005.

[5] Y. J. Noh, G. Liu, E. K. Seo, J. R. Wang, and K. Aonashi, "Development of a snowfall retrieval algorithm at high microwave frequencies," *Journal of Geophysical Research: Atmospheres,* vol. 111, no. D22, 2006.

[6] R. R. Braham Jr, "Snow particle size spectra in lake effect snows," *Journal of Applied Meteorology and Climatology,* vol. 29, no. 3, pp. 200-207, 1990.

[7] N. Hedstrom and J. Pomeroy, "Measurements and modelling of snow interception in the boreal forest," *Hydrological Processes,* vol. 12, no. 10-11, pp. 1611-1625, 1998.

[8] M. M. Rohde *et al.*, "An interactive physics-based unmanned ground vehicle simulator leveraging open source gaming technology: progress in the development and application of the virtual autonomous navigation environment (VANE) desktop," in *Unmanned Systems Technology XI*, 2009, vol. 7332: SPIE, pp. 373-385.

[9] J. Tan and X. Fan, "Particle system based snow simulating in real time," *Procedia Environmental Sciences,* vol. 10, pp. 1244-1249, 2011.

[10] Y. Amarasinghe, W. Zhang, R. Zhang, D. M. Mittleman, and J. Ma, "Scattering of terahertz waves by snow," *Journal of Infrared, Millimeter, and Terahertz Waves,* vol. 41, pp. 215-224, 2020.

[11] R. Tao *et al.*, "Snow particle size distribution from a 2-D video disdrometer and radar snowfall estimation in east China," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 59, no. 1, pp. 196-207, 2020.

[12] K. Gunn and J. Marshall, "The distribution with size of aggregate snowflakes," *Journal of Atmospheric Sciences,* vol. 15, no. 5, pp. 452-461, 1958.

[13] I. Saltvik, A. C. Elster, and H. R. Nagel, "Parallel methods for real-time visualization of snow," in *Applied Parallel Computing. State of the Art in Scientific Computing: 8th International Workshop, PARA 2006, Umeå, Sweden, June 18-21, 2006, Revised Selected Papers 8*, 2007: Springer, pp. 218-227.

[14] R. Eidissen, "Utilizing gpus for real-time visualization of snow," Citeseer, 2009.

[15] E. Barthazy and R. Schefold, "Fall velocity of snowflakes of different riming degree and crystal types," *Atmospheric research,* vol. 82, no. 1-2, pp. 391-398, 2006.

[16] M. Nemoto and K. Nishimura, "Numerical simulation of snow saltation and suspension in a turbulent boundary layer," *Journal of Geophysical Research: Atmospheres,* vol. 109, no. D18, 2004.

[17] I. M. Sobol, "On the distribution of points in a cube and the approximate evaluation

of integrals," *Zh. Vychisl. Matem. Matematich. Fiz.,* vol. 7, no. 4, pp. 784-802, 1967.

[18] J. Matoušek, "On the L2-discrepancy for anchored boxes," *J. Complexity,* vol. 14, no. 4, pp. 527-556, 1998, doi: //dx.doi.org/10.1006/jcom.1998.0489.

[19] H. S. Hong and F. J. Hickernell, "Algorithm 823: Implementing scrambled digital sequences," *ACM T. Math. Software,* vol. 29, no. 2, pp. 95-109, 2003.

[20] W. Budd, "The drifting of nonuniform snow particles1," *Studies in Antarctic* meteorology*,* vol. 9, pp. 59-70, 1966.

[21] R. Schmidt, "Vertical profiles of wind speed, snow concentration, and humidity in blowing snow," *Boundary-Layer* Meteorology*,* vol. 23, no. 2, pp. 223-246, 1982.

[22] J. Zikmunda, "Fall Velocities of Spatial Crystals. and Aggregates," *Journal of Atmospheric Sciences,* vol. 29, no. 8, pp. 1511-1515, 1972.

[23] J. D. Locatelli and P. V. Hobbs, "Fall speeds and masses of solid precipitation particles," *Journal of Geophysical Research,* vol. 79, no. 15, pp. 2185-2197, 1974.

[24] R. A. Johnson and D. W. Wichern, *Applied multivariate statistical analysis* (no. 8). Prentice Hall Upper Saddle River, NJ, 2002.

[25] M. A. Rice, B. B. Willetts, and I. McEwan, "An experimental study of multiple grain-size ejecta produced by collisions of saltating grains with a flat bed," *Sedimentology,* vol. 42, no. 4, pp. 695-706, 1995.