IIEEJ Paper -- Special Issue on Visual Computing --

Multi-Sized Particle Sampling Method Based on Porosity Optimization in 2D Space

Xu WANG† , Makoto FUJISAWA†(*Member*) , Masahiko MIKAWA†

†University of Tsukuba

<**Summary**> This paper introduces a multi-sized particle sampling method within an arbitrary 2D shape using power tessellation. Our method aims to improve packing density as to sample as many particles as possible in a limited area. We propose a porosity-driven optimization technique to ensure no overlap between particles while increasing the packing density. With such properties, our method is applicable to physically-based simulations, such as the Discrete Element Method and its related framework. Additionally, this technology allows users to set the target particle size distribution by a predesigned cumulative distribution function and restrict the errors between 10% and 20% after the optimization. We demonstrate that our multi-sized particle sampling algorithms significantly improve packing density compared to Poisson disk sampling and SPH-based blue noise sampling.

Keywords: multi-sized particle sampling, porosity optimization, power tessellation, discrete element method

1. Introduction

Uniform particle sampling is a highly applicable technique in many graphics-related fields, such as image synthesis, rendering, geometry processing, and physically-based simulations. Notably, it has played an essential role in particle-based physics simulations in recent years. Meanwhile, several adaptive particle (multi-sized) sampling methods[1)–3)]are also implemented into the fluid simulation. However, it is not well suited for simulating granular materials, such as sand, because slight overlap can destabilize the simulation. Therefore, we aim to achieve multi-sized particle sampling in arbitrary 2D polygons and have it available for using in stable physics-based sand simulations.

This paper proposes a multi-sized particle sampling algorithm using weighted centroidal power tessellation. It computes the particle distribution by minimizing the porosity in a pre-defined 2D polygon. Our contributions are as follows:

- We present a novel multi-sized particle sampling method based on minimum porosity. Our method ensures no overlapping occurs between particles and effectively exploits the limited space to improve sampling density.
- Our method supports users to give a target particle size distribution using a predesigned cumulative distribution function and limits the error between 10% and 20% after the optimization.
- Multi-sized particles sampled by our method are available to the Discrete Element Method (DEM) and can realize stable physical simulations, whereas the particles sampled in the comparison experiments fail to perform stable physical simulations.

2. Related Work

2.1 Uniform particle sampling

Blue noise sampling is a well-known method used in computer graphics because of its ability to generate random uniform distributions. One of its patterns, Poisson disk sampling, has many rendering and geometry processing applications. Bridson[4)]proposed a fast Poisson disk sampling method that is effective and more easily implemented in arbitrary dimensions. While this method has a wide application, it is difficult to control the sampling density and ensure no overlapping between particles. Traditionally, uniform particle sampling within a 2D polygon is mainly implemented by dividing the plane as the uniform grids (Voxelization in 3D[5)]) and generating a particle in each uniform grid. However, it will result in a particle distribution that does not fit the boundary shape. As a solution to this problem, Schechter et al.[6)]proposed a relaxation-based sampling method to smoothen the particles on the boundary. Later, Jiang et al.[7)]applied the Smoothed Particle Hydrodynamics (SPH) method to achieve blue noise sampling, which can also re-

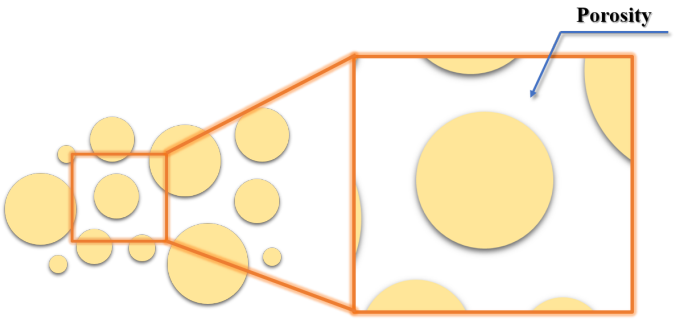


Fig. 1 Schematic diagram of porosity

alize uniform particle sampling by relaxing the boundary particles. Although these methods work well for uniform particle sampling, what we expect as multi-sized particle sampling and stable sand simulation cannot be fulfilled by simply extending their method.

2.2 Adaptive particle sampling

In order to achieve high-performance SPH-based fluid simulation, Adams et al.[1]presented a dynamic sampling method, which can dynamically adjust the size of the sampled particles by computing the distance between the medial axis and fluid surface. Jiang et al.[7]integrated the adaptive particle sampling method and proposed an SPH-based blue noise sampling to treat multi-sized particles. Winchenbach et al.[2]proposed an adaptive incompressible SPH-based fluid simulation method and an improved framework[3]adapted to higher volume ratios for simulating low-viscosity turbulent flows and achieving the high-resolution rendering of fluid surfaces. However, since these algorithms are all SPH-based, there will be overlapping between the multi-sized particles, which make the multi-sized sand simulation unstable.

2.3 Voronoi diagram and centroidal voronoi tessellation

The Voronoi diagram has been widely used in the area of computer graphics. It has many applications in physics-based simulations of computer animation, such as power particles[8], adaptive staggered power particles[9], power diagram-based high-resolution adaptive liquids[10]. Over time, several GPU-based algorithms diagram have been developed aiming to speed up the computation of the 3D Voronoi[11)–14)]. Among these methods, centroidal Voronoi tessellation (CVT) for reaching a specific density distribution is also one of the critical topics related to the Voronoi diagram. CVT can be regarded as one of the relaxation-based sampling methods. It has two stages of computation: 1. Randomly generates the point set. 2. Using Lloyd iterations to optimize the location of the points until convergence. However, directly using CVT will not cause the sampled points to have better properties of blue noise. Balzer et al.[15),16)]proposed a capacity-constrained Voronoi tessellation, which can generate high-quality blue noise properties and perfectly fit a known density function. According to the feature that the energy function of CVT has second-order smoothness, Liu et al.[17)]presented a faster convergence CVT method. Chen et al.[18)]proposed a parallel algorithm for CVT on GPU and verified the algorithm's performance, which can obtain good blue noise characteristics and high-speed computing. All these methods described above have difficulty in extending to multi-sized particle sampling. On the other hand, CVT was used to study the hierarchical data visualization presented by Balzer et al.[19)]. They called this approach "Voronoi Treemaps". Later, Nocaj et al.[20)]introduced a faster, more straightforward, and resolution-independent method for computing Voronoi treemaps. While this approach can generate multi-sized particles, the user needs to pre-define a large amount of data since its computation is based on hierarchical data. Besides, sampled results are singular with the same data.

2.4 Physics based granular material simulation

Granular material simulation-related studies have a long history in the computer graphics community[21)–25)]. Numerous works in recent years still focus on sand-water mixing simulation frameworks, such as SPH-based multi-phase multi-material flows[26),27)]and material point method (MPM) based simulations of a mixture of sand and water[28),29)]. Since the particle-particle approach (DEM) could better capture small-scale features at the particle level, several studies have also employed this method to make sandy-like animation, e.g., hybrid grains [30)]and seepage flow framework[31)].

3. Porosity Optimization Based Approach

This section will introduce our porosity optimization-based method of how to sample multi-sized particles.

3.1 Controllable porosity

Generally, a rock or soil's porosity is measured by the fraction of the total soil volume occupied by pore space per unit volume. As shown in **Fig.1**, the porosity ϕ is defined as:

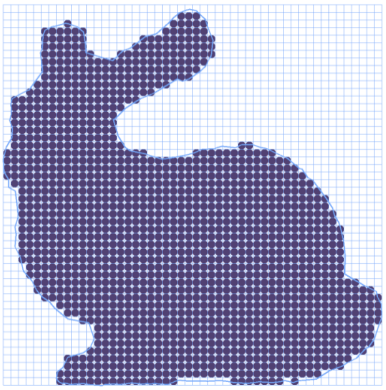


Fig. 2 Result of standard gridding approach based uniform particle sampling

$$\phi = 1 - \frac{V_{soil}}{V_{total}} \tag{1}$$

where V_{soil} is the volume of soil particles, and V_{total} presents a unit soil volume. The high ratio of soil particles in the unit volume will produce minor porosity. Hence we can improve the packing density within the unit volume by reducing the porosity and filling the limited space with more particles. Thanks to the controllability of the porosity, we can obtain higher packing densities and multi-sized particles by using a power diagram and porosity optimization.

3.2 Power diagram

There are n points $S = \{s_1, s_2, \ldots, s_n\}$ (s_i called Voronoi site) in the plane, Voronoi diagram is made by dividing the plane into n regions $\mathcal{R} = \{\mathcal{R}(s_1), \mathcal{R}(s_2), \ldots, \mathcal{R}(s_n)\}$ (each site $s_i \in S$ and its associated region $\mathcal{R}(s_i)$ are called Voronoi cell) and each region has only one point. Given an arbitrary 2D polygon as boundary Ω, Voronoi cell $\mathcal{R}(s_i)$ can be defined as:

$$\mathcal{R}(s_i) = \{p \in \Omega : \|p - s_i\| < \|p - s\| \quad \text{for each } s \in S - s_i\} \tag{2}$$

When assigning a weight w_i to each Voronoi site s_i, we call this Voronoi diagram as Power diagram which is defined as:

$$\mathcal{R}(s_i, w_i) = \{p \in \Omega : \|p - s_i\|^2 - w_i < \|p - s\|^2 - w \quad \text{for each } s \in S - s_i\} \tag{3}$$

When the weights w_i of all Voronoi site s_i are equal to zero, the power diagram is equivalent to a Voronoi diagram.

In contrast to the Voronoi diagram, the power diagram allows us to control the shape and size of each Voronoi cell in a relatively simple way. It means that we can more readily manipulate the size of the sampled particles and the porosity of the entire space. For controlling the size of Voronoi cell, there are many studies related to the capacity-constrained method. In these studies, some methods combined power diagrams with a false position method to obtain the different sizes of Voronoi cells[16] or integrated with Newton's method[32], L-BFGS method[33]. However, all these methods have a high computational cost and require optimization parameters tuning. As for our method, it is less complicated and more straightforward to control the size of the Voronoi cell. Though the precision is slightly inferior, it significantly improves packing density.

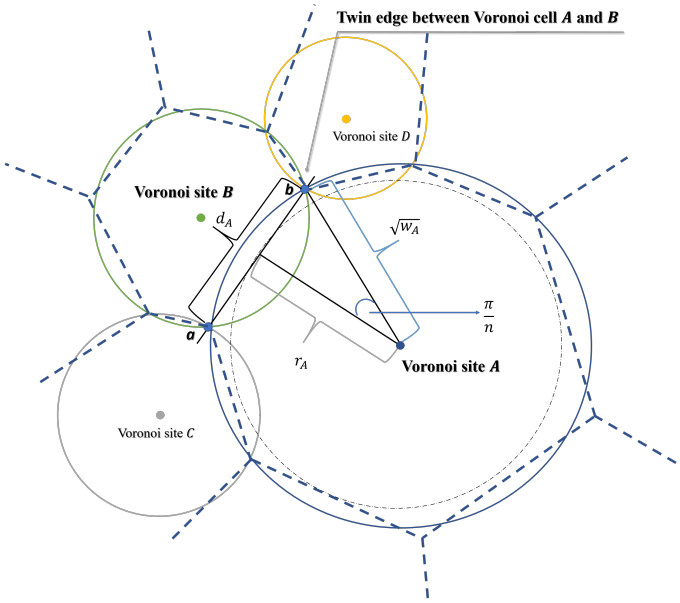


Fig. 3 A ideal case of the geometric relationship between the Voronoi cells and its weights

3.3 Requirements of multi-sized particle sampling

Using the standard gridding approach (**Fig.2**) as an example, when we generate particles in a unit cell, the minimum porosity is a fixed value of $1 - \frac{\pi}{4}$. It makes minimum porosity become a constant value and impossible to adjust dynamically.

According to the properties of the power diagram, the plane are divided into several Voronoi cells and each of them will be a convex polygon with a weight factor that can adjust its associated region. What inspires us is that it is possible to obtain sampling results with different porosity sizes by adjusting each Voronoi cell's weighting coefficients. During the Lloyd iteration, as changes in the weight coefficients of each Voronoi cell impact the shape and size of its associated region, it makes both the size of

particles and the pore size of its associated region changeable in each Voronoi cell. This property of the power diagram makes it possible to achieve multi-sized particle sampling within a given boundary. In order to achieve multi-sized particle sampling with a higher packing density, i.e., finding a minimum porosity sampling case, we summarize the requirements for the power diagram where the minimum porosity is obtainable, which we consider an ideal case:

Requirement 1: If each side of the Voronoi cell has the same length, then it is a regular polygon.

Requirement 2: If n Voronoi sites encircle a Voronoi site, its associated Voronoi cell is a regular n sided polygon.

Requirement 3: In the case of satisfying $n \to \infty$ and each side d of Voronoi cell has the same length, the shape of Voronoi cell is closer to the circle, porosity $\phi \to 0$.

Requirement 4: Each Voronoi site must be at the centroid of its associated Voronoi cell.

Notably, the ideal case described above is only a scenario envisioned for each Voronoi cell. It is impossible that the Voronoi cell can fully satisfy all the given four requirements locally or globally, but we can design the loss functions based on the requirements of an ideal case. Specifically, as the requirements are formulated for each Voronoi cell, the objective is to locally make each Voronoi cell meet the requirements and find a globally optimal solution.

3.4 Optimization

3.4.1 Objective function

To satisfy the requirements described in Section 3.3, we sketched **Fig.3** to represent an ideal case of the geometric relationship between the Voronoi cells and their weights. In this ideal case, Voronoi site A with B, C and D circling it satifies the conditions in **Fig.4**(b), which means that computing the length of twin edge $\overline{\text{ab}}$ between Voronoi sites A and B is sufficient to know the weight of any site. Notably, it doesn't mean that we ignore the weight of the other Voronoi site but it has been taken into account in the computing power diagram (the power diagram will be reconstructed before updating the weights of Voronoi sites, as detailed in Algorithm 1). As for computing power diagrams, we mainly rely on the method proposed by Aurenhammer[34]. They bring a general d-dimentional power diagram up to compute with a d+1 dimensional convex hull. Briefly, they transform a 2D weighted Voronoi site $s \in S$ into a sphere with coordinates (x_s, y_s) and radius $r = \sqrt{w_s}$ mapped into the 3D plane, and then intersect with other transformed Voronoi sites on a 3D plane and finally apply an inverse transformation project onto the 2D plane (details can be found in Aurenhammer[34]or Nocaj et al.[20]). The circles with the solid line in Fig.3 are the projection results, and their radii are the root of the weights of the associated Voronoi sites (i.e., the radius of Voronoi site A equal to $\sqrt{w_A}$).

Assuming that there exist n points around Voronoi site S_A, Ideally, the side length d_A of $\mathcal{R}(S_A)$ can be computed as follows:

$$d_A = 2\sqrt{w_A - r_A^2} \tag{4}$$

where w_A is the weight of Voronoi site S_A and r_A is the radius of Voronoi cell $\mathcal{R}(S_A)$'s maximum inscribed circle.

From the geometric properties of the power diagram, Equation 4 is only established in the case of Fig.4(b). When the weight of the Voronoi site is less than r_A^2, it will produce the case (a) or case (c) in Fig.4, which means that we will use a penalty function ψ (Section 3.4.2) instead of shape loss function $\Phi_{\mathcal{P}}$ for preventing two sites from being too close or too far apart.

To satisfy requirements 1, 2, and 3 in Section 3.3, we focus on information about the adjacent Voronoi sites rather than the shape of the target Voronoi cell. It is because each edge of the target Voronoi cell is generated using the weights of the neighboring Voronoi sites. We consider updating the weight of the target Voronoi site for each iteration while the weights of the neighboring Voronoi sites are fixed. Consequently, to keep each edge of the target Voronoi cell of the same length as possible, the update tendency of the target Voronoi site weight can be computed by incorporating the weights of all neighboring Voronoi sites. One can define the loss function that controls the shape part of the porosity optimization as follows:

$$\Phi_{\mathcal{P}}(w, r) = \sum_{i \in S} \sum_{j \in S-i} \left| w_i - w_j + r_j^2 - r_i^2 \right| \tag{5}$$

Meanwhile, we set a target particle radius distribution (Equation 10) in the initial stage to control the size of the sampled particles. According to this distribution function, we use the inverse transform method[35]to ensure that the generated target particle radii are uniformly dis-

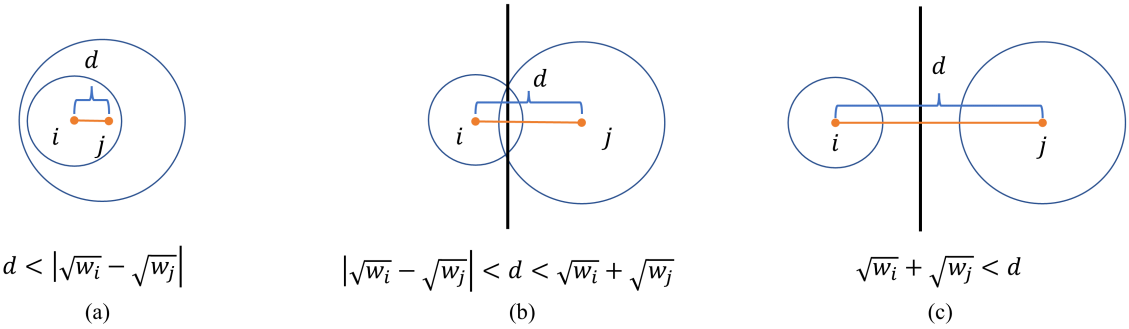


Fig. 4 Three possible cases between two neighboring Voronoi sites

tributed in each subinterval. Then we assign the generated target radius r_i to each Voronoi site s_i when generating the initial power diagram or dynamically appending particles (Section 3.4.3). Locally, the radius of the maximum inscribed circle in each Voronoi cell will approach the user's pre-set target radius. The global particle radius distribution obtained from multi-sized particle sampling will approximate the pre-defined particle radius distribution. Therefore, the area of each Voronoi cell in the ideal case (assume a regular n-sided polygon) can be computed by:

$$A_i^{\text{ideal}} = \frac{nr_i^2}{tan\frac{\pi}{n}} \tag{6}$$

We re-generate the power diagram with new weights with each iterative computation, and each Voronoi site corresponds to a new Voronoi cell. In order to achieve the target area for each newly generated Voronoi cell, we define the loss function that controls the area part of the porosity optimization as follows:

$$\Phi_{\mathcal{A}}(A) = \sum_{i \in S} \frac{A_i^{\text{ideal}} - A_i^{\text{current}}}{A^{\text{total}}} \tag{7}$$

where A_i^{current} is the area of Voronoi site S_i associated region in the current iteration step, and A^{total} is the area of the boundary polygon Ω, which is the sum of all Voronoi cell areas. Noticeably, the A_i^{current} is not modified directly by the cost function, but it depends on the shape of the corresponding Voronoi cell in the updated power diagram. In addition, A_i^{current} is also used for computing the porosity when we append new particles.

With joint the shape loss term $\Phi_{\mathcal{P}}$ and area loss term $\Phi_{\mathcal{A}}$, the final loss function we minimise can be defined as:

$$\Phi(w, r, A) = \alpha\Phi_{\mathcal{P}}(w, r) + \beta\Phi_{\mathcal{A}}(A) \tag{8}$$

where α and β are two weighted coefficients for adjusting global optimization. We apply Lloyd iteration to the power diagram to make each Voronoi cell meet the area requirement. Moreover, as the geometric relationship between the Voronoi cells and their weights is known, it allows us to optimize the shape of the Voronoi cells by adjusting the weights. Balzer and Deussen first proposed optimally meeting area requirements via Lloyd's method [19]. It was followed by "Voronoi Treemaps",[20]which also proved the practicability of this method. The core idea is that the weight of each Voronoi site is increased or decreased in proportion to the missing or redundant areas during Lloyd iteration. In our experiments, we found that this method is also applicable for controlling the shape of Voronoi cells, as shown in **Fig.5** for the optimization effect of porosity. It should note that we could perform L-BFGS method to optimize the power diagram[32),33)], but Lloyd's method is more straightforward, faster, and easier to implement. We would like to compare the impact of Lloyd's and L-BFGS methods on our experimental results in the future.

3.4.2 Constrained condition

While the Voronoi diagram ensures that each Voronoi cell only contains one Voronoi site, the power diagram may produce 3 cases, as shown in Fig.4, which shows that the Voronoi cell may contain zero or several Voronoi sites. Furthermore, the case of a Voronoi cell containing no Voronoi site or more than one Voronoi site will lead to the wrong results of our multi-sized particle sampling method. Our solution is not to perform the optimization described in Section 3.4.1 when the distance d_{ij} between Voronoi site i and j is greater than $\sqrt{w_i} + \sqrt{w_j}$ or less than $\left|\sqrt{w_i} - \sqrt{w_j}\right|$. Instead, we define a penalty function ψ for preventing two sites from being too close or too far apart, shown as follows:

$$\psi\left(w_i, d_{ij}, \sigma_k\right) = \sigma_k\left(d_{ij}^2 - w_i\right) \tag{9}$$

where $\sigma_k = k^2$ presents the penalty factor in k-th iter-

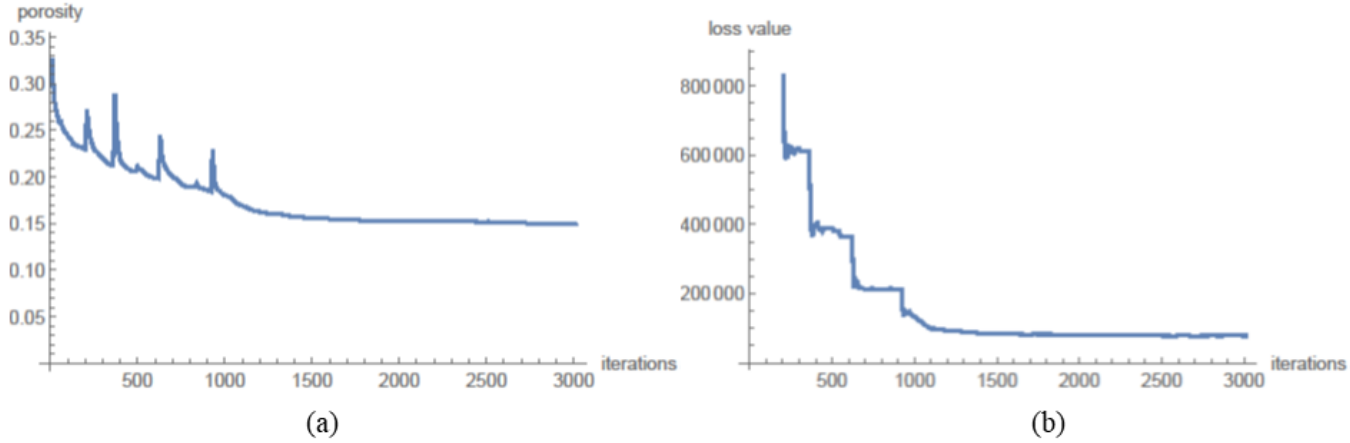


Fig. 5 The variation of porosity and loss value in the Bunny example

ation.

3.4.3 Dynamic particles appending

A fundamental issue with power diagram-based multi-sized particle sampling is that it requires searching for the optimal number of particles to fill the limited space. The reason is that generating a power diagram is needed to predetermine the number and weights of Voronoi sites. While the optimum number of particles for filling the entire space can be roughly estimated from the pre-defined particle distribution, the weights of each Voronoi site are difficult to compute in advance. To solve this problem, we propose a dynamic particle appending based on porosity. Specifically, we re-compute the porosity in the given boundary Ω at the end of each iteration and record it. Within the recorded data series, if the current iteration step is T_i, we will extract the porosity data change. Within a range from T_{i-k} to T_i (data length is k), we determine whether to append particles. We are using the least-squares method to fit the extracted porosity data and evaluate the slope of the fitted line. When it is lower than the threshold ϵ, we will append new particles according to the pre-defined particle radius distribution. We set $k = 20$ and $\epsilon = 1e{-}6$ in all our experiments and illustrate the process of dynamic particles appending method in **Fig.6**.

4. Implementation

This section will introduce the multi-sized particle sampling method described in Algorithm 1. The input to our algorithm is a boundary polygon Ω, composed of a series of 2D vertices and a list of target particle radii, and the list is generated from a user-defined particle radius distribution. The output is a power diagram consisting of sampled n Voronoi sites, where each site contains a particle radius. First, before the porosity optimization, we use the Lloyd iteration to ensure that each Voronoi site is located at the centroid of the Voronoi cell (line 22) followed by a search for neighboring Voronoi sites (line 4). We do not have to implement a specific neighbor search algorithm. The neighbor sites' information can be obtained from the QuickHull algorithm.

To address the 3 cases in the power diagram (Fig.4), the weights of the target and neighboring Voronoi sites determine whether to perform the objective function for porosity optimization or a penalty function (lines 6 to 12). Line 13 describes the degree to which the area of each Voronoi cell needs to be altered. Similarly, line 14 describes the degree of variation which is capable of making the shape of each Voronoi cell a regular n-sided polygon. Then we compute the values of the area and shape weights respectively (weights $w_i^{\mathcal{A}}$ and $w_i^{\mathcal{P}}$ are increased or decreased proportionally to $\Phi_{\mathcal{A}}^i$ and $\Phi_{\mathcal{P}}^i$)for each Voronoi cell that need to be updated for reaching the ideal case (lines 16 to 20). After updating the weighted factors, we need to recompute the power diagram and perform Lloyd iteration to make sure that each Voronoi site is located at the centroid of Voronoi cell (line 21 and line 22). Finally, we use "Dynamic Particles Appending Mechanism" to determine the convergence of the current porosity and whether the additional particles are needed (line 23). In all of our experiments, we set $\alpha = 1$ and $\beta = 1$ for a stable and effective optimization.

We have implemented our multi-sized particle sampling algorithm using C++ on AMD Ryzen 9 3900X CPU. In

Algorithm 1 Multi-Sized Particle Sampling Based on Porosity Optimization

Input: Boundary Polygon Ω, Target Particle Radius $R_1, R_2, \cdots, R_n$
Output: Power Diagram $\mathcal{R}(S)$, Voronoi Sites $s_i \in S$ with a particle radius

repeat
  $\mathcal{R}(S) \leftarrow$ Compute Power Diagram
  for each s_i in S **do**
    s_i Neighbor Searching(n_i: Number of Neighbor Sites $s_j \in S_i'$)
    $\Phi_{\mathcal{P}}^i, \Phi_{\mathcal{P}}^{\text{total}} \leftarrow 0$
    for each s_j in S_i' **do**
      if $(d_{ij} < |\sqrt{w_i} - \sqrt{w_j}|) \vee (\sqrt{w_i} + \sqrt{w_j} < d_{ij})$ **then**
        $\Phi_{\mathcal{P}}^i \mathrel{+}= \sigma_k(d_{ij}^2 - w_i)$
      else
        $\Phi_{\mathcal{P}}^i \mathrel{+}= \sum_j (\frac{\sum_j (w_j - r_j^2)}{n_i} - (w_j - r_j^2))$
      end if
      $\Phi_{\mathcal{P}}^{\text{total}} \mathrel{+}= \Phi_{\mathcal{P}}^i$
    end for
    $\Phi_{\mathcal{A}}^i = \frac{A_i^{\text{idea}} - A_i^{\text{current}}}{A^{\text{total}}}$ $(A_i^{\text{idea}} = \frac{n_i r_i^2}{\tan\frac{\pi}{n_i}})$
    $\Phi_{\mathcal{P}}^i = \frac{\Phi_{\mathcal{P}}^i}{\Phi_{\mathcal{P}}^{\text{total}}}$
  end for
  for each s_i in S **do**
    $w_i^{\mathcal{A}} = 1 - (\Phi_{\mathcal{A}}^i - 1)^2$
    $w_i^{\mathcal{P}} = 1 - (\Phi_{\mathcal{P}}^i - 1)^2$
    $w_i \mathrel{+}= \alpha w_i^{\mathcal{A}} + \beta w_i^{\mathcal{P}}$
  end for
  $\mathcal{R}(S) \leftarrow$ Compute Power Diagram
  Performing Lloyd iteration
  Performing Dynamic Particles Appending Mechanism
until $\Phi_{\text{global}} < \varepsilon$

line 2 of Algorithm 1, we use the quick hull algorithm[36] to compute the power diagram, which is stable, fast, and scalable for arbitrary dimensions. Additionally, we generate the straight skeleton[37]in each Voronoi cell and compute the position and radius of the maximum inscribed circle in the Voronoi cell based on the straight skeleton as the output of Algorithm 1.

5. Results

In the experiment, we set up the particle radius distribution at initialization by using a cumulative distribution function as follows:

$$F_{radius}(r) = \begin{cases} 0.5 & \text{if } 20 < r \leq 30 \\ 0.4 & \text{if } 30 < r \leq 80 \\ 0.1 & \text{if } 80 < r \leq 150 \end{cases} \tag{10}$$

At initialization, we randomly generate 100 points within the boundary polygon Ω and set the maximum iterations to 3000.

Figure 7 shows the results of our multi-sized particle sampling method at the initial stage, 200th iterations, 400th iterations, 2000th iterations, and 3000th iterations. We also recorded the values of the global porosity

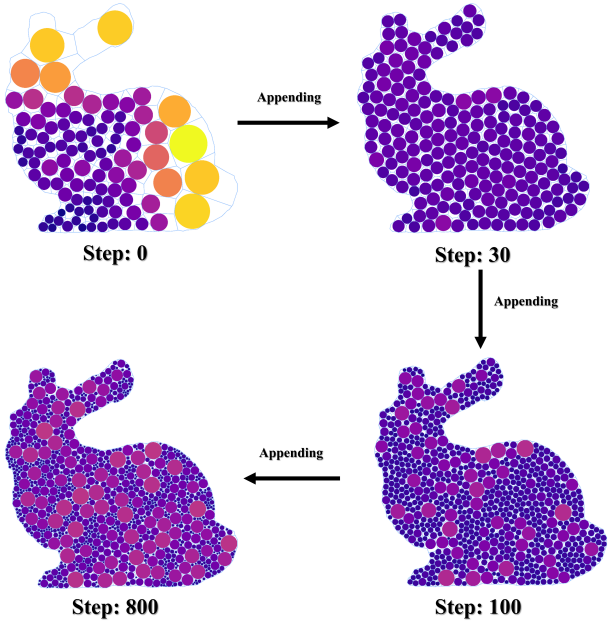


Fig. 6 Dynamic particles appending mechanism

(Fig.5(a)) and the loss value (Fig.5(b)) during 3000 iterations by taking Bunny as an example. It is worth noting here that in Fig.5(b), each time the loss value drops drastically, it is caused by triggering the dynamic particle appending mechanism.

Even though the porosity in Fig.5(a) also declines steadily when dynamic particle appending mechanism occurs, a sudden porosity increase occurs in the shortest time after each particle appending. This phenomenon is caused by the global porosity optimization having reached a steady state before adding the particles. Nevertheless, after appending new particles, since the shape optimization of each newly added Voronoi site is not sufficient, this will result in the global porosity sudden increase. Additionally, we measured the predefined particle size distribution and optimized particle size distribution over 3000 iterations in **Fig.8**. Intuitively, our optimized particle size distribution is similar to the predefined particle size distribution. Quantitatively, the error rate between optimized and predefined particle size distribution is roughly 15% according to RMSPE (Root Mean Squared Percentage Error).

Except for the distributions we used in Fig.7, we also experimented on whether it is possible to find a minor porosity under different pre-defined particle radius distributions. The top of **Fig.9** presents the multi-sized particle sampling results after 3000 iterations, while the bottom part corresponds to the pre-defined particle distribution. As a result of this experiment, setting a large

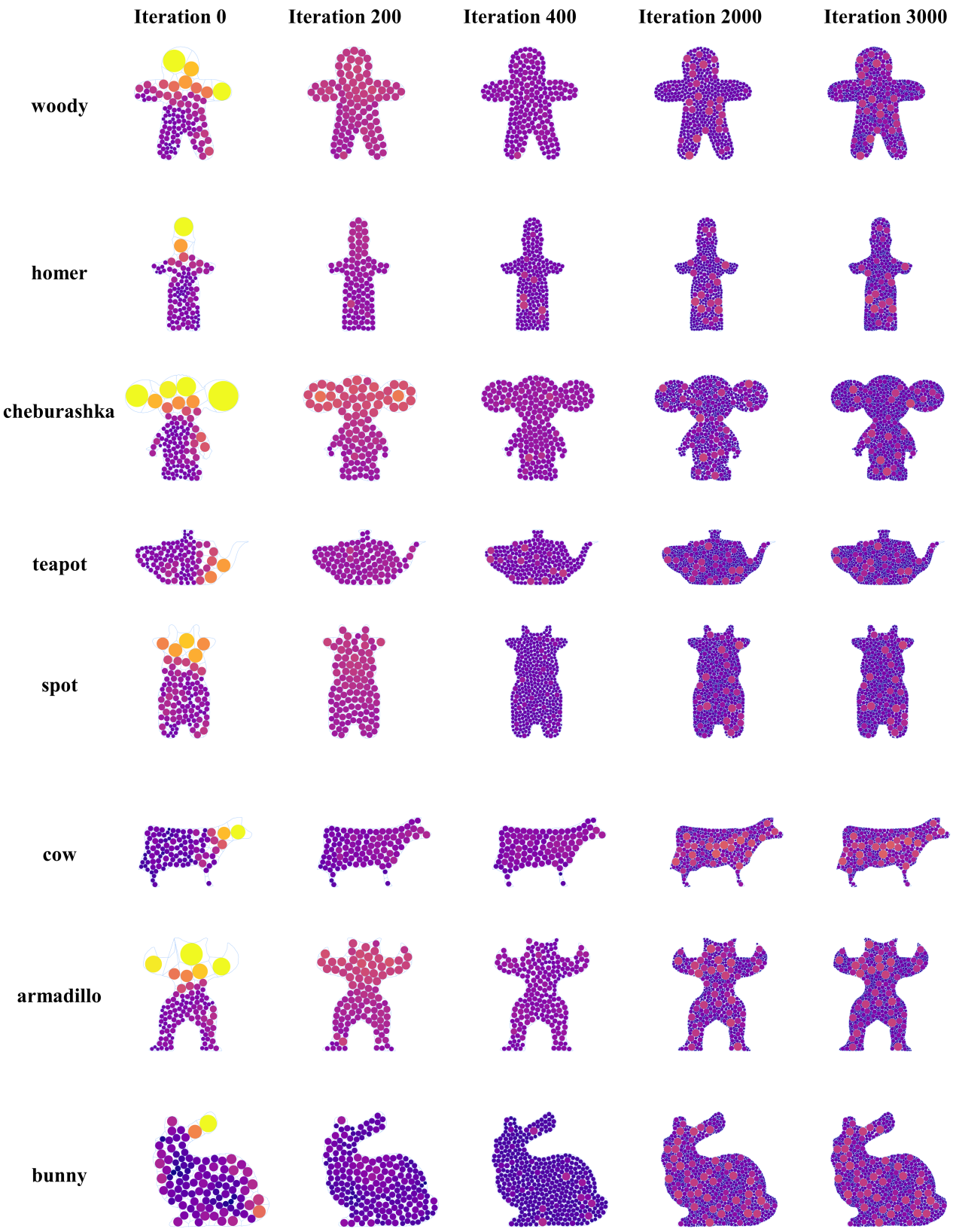


Fig. 7 Results of our multi-sized particle sampling method

number of the small size particles will be helpful to obtain a more negligible porosity.

As the right-most case in Fig.9 shows that our method allows to achieve a more complicated version of the parti-

cle distribution (containing three peaks), but in this case it will be hard to promote the packing density.

6. Discussion

Compared to uniform particle sampling method. Both standard gridding approach[5] and Poisson disk sampling[4] can keep no overlapping between particles. Nevertheless, neither of them can improve the packing density in a limited space while ensuring that the particles do not overlap with each other. In addition, as these two methods do not obtain a balanced representation at the boundary, the sampled particle clouds hardly preserve the polygon's original shape. Figure 7 demonstrates that the particles sampled by our method can ensure a smooth boundary and that there is no overlap between particles while increasing the packing density (i.e., the global porosity is effectively reduced, as shown in Fig.5(a)). SPH-based particle sampling methods[6),7] can provide a uniform distribution at the boundary and produce a dense cloud of particles. Unfortunately, due to the specialty of the SPH-based method, it is not ensured that there is no overlapping between the sampled particles. It could be applied to SPH-related algorithms, but it will produce unstable simulation results when applied to DEM-related methods. Compared to the SPH-based method[7], our algorithm could provide a stable DEM simulation, as shown in **Fig.10**.

Compared to multi-sized particle sampling method. Most of the adaptive particle sampling methods[1)–3)] focused on the appearance of detailed fluid surfaces or water splashes, and they had the property in which the particle radius became larger and closer to the center. As these methods are also based on SPH, they have the same issues as SPH-based blue noise sampling[7] . Furthermore, we also provide a comparison experiment with the Poisson disk-based multi-sized particle sampling in Fig.10, which fails to achieve stable simulation results due to the overlapping particle problem.

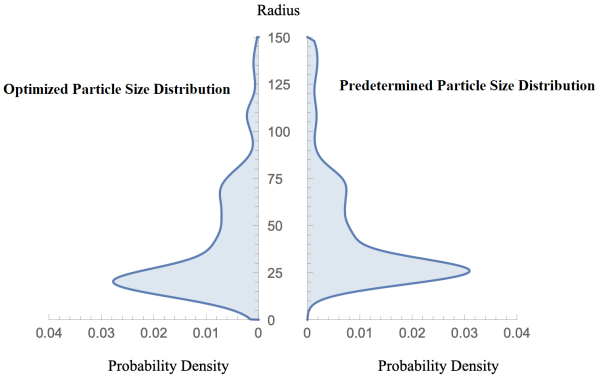


Fig. 8 Optimized and predetermined particle size distribution

Compared to porosity with different sampling method. While we illustrate the instability simulation when the Poisson disk sampling and SPH-based blue noise sampling methods are applied to DEM in Fig.10, the instability problem can be simply solved by detecting the overlapping parts between particles and modifying the particle radius. We admit that this approach is feasible but will induce a higher porosity, and the particle radius distribution is hard to control. Nevertheless, our approach ensures no overlap between particles and promotes the packing density (minor porosity). To prove that our method can produce a relatively more minor porosity, we adjust the size of each particle (the radius of each particle is reduced by 1/2 the amount of overlap) obtained by Poisson disk sampling and SPH-based blue noise sampling so that there is no overlap between particles. As shown in **Table 1**, we statistically measure the porosity of 11 different shapes of polygons respectively by using Poisson disk sampling, SPH-based blue noise sampling, and our method. Based on the results, the average porosity obtained by our sampling method is the most minor (0.165) and much better than the methods Poisson disk sampling (0.697) and SPH-based blue noise sampling (0.629).

7. Conclusion

This paper proposed to improve the packing density of particle sampling in a restricted space as well as keeping no overlap between particles. Our method allows users applying stable physics simulations based on DEM methods and related frameworks. We have introduced a multi-sized particle sampling method based on porosity optimization, which can obtain a particle distribution by minimizing the porosity in a pre-defined 2D polygon. The comparative experiments with alternative particle sampling methods demonstrate that we can deliver a stable DEM-based sand simulation.

While the experimental results have proven that our method can adapt to a variety of 2D polygons for multi-sized particle sampling, it still exists the case of undesirable sampling results as shown in **Fig.11**. These cases showed that if a particle with a large radius preset near a narrow area, it will make the effectiveness of local op-

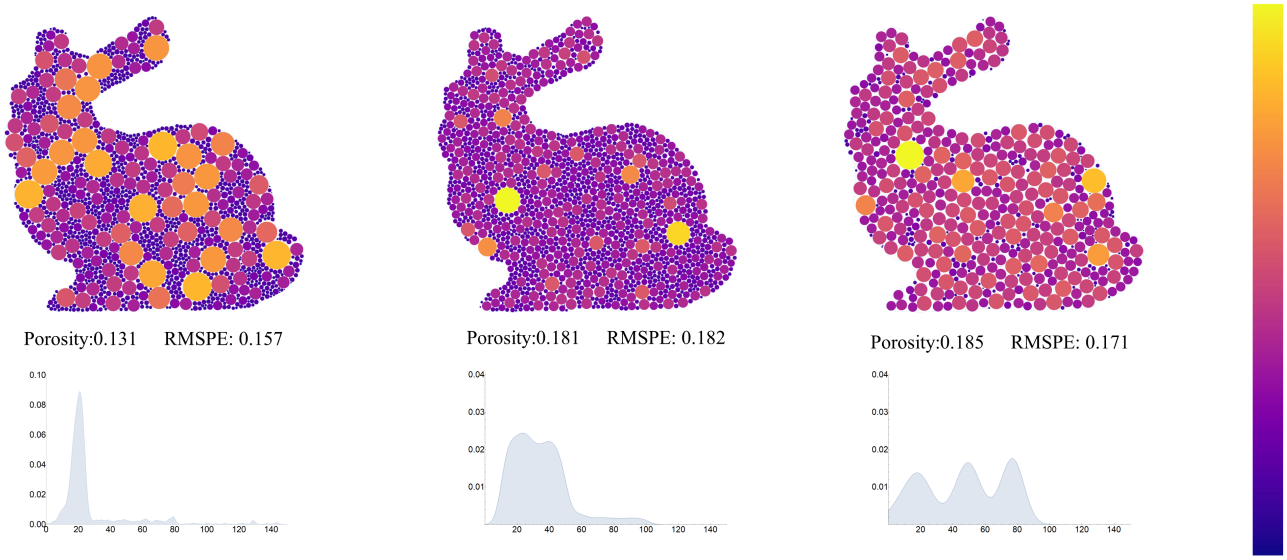


Fig. 9 Optimization results obtained from different pre-defined particle radius distributions

timization quite unfavorable. Moreover, while our approach is designed to meet the pre-defined particle radius distribution by adjusting Voronoi cells shape and size continuously, it lacks how to keep the balance of the global particle distribution, especially for the larger-sized particles. In the future, we would like to handle these problems through an extra constraint on the particle position distribution. Furthermore, we expect to explore a way to extend our method to 3D multi-sized particle sampling and adapt it to more physical simulation scenarios.

Acknowledgements

We would like to thank the anonymous reviewers for their valuable suggestions and comments. This work was supported by JSPS KAKENHI Grant Number JP20K11839 and JST SPRING Grant Number JPMJSP2124.

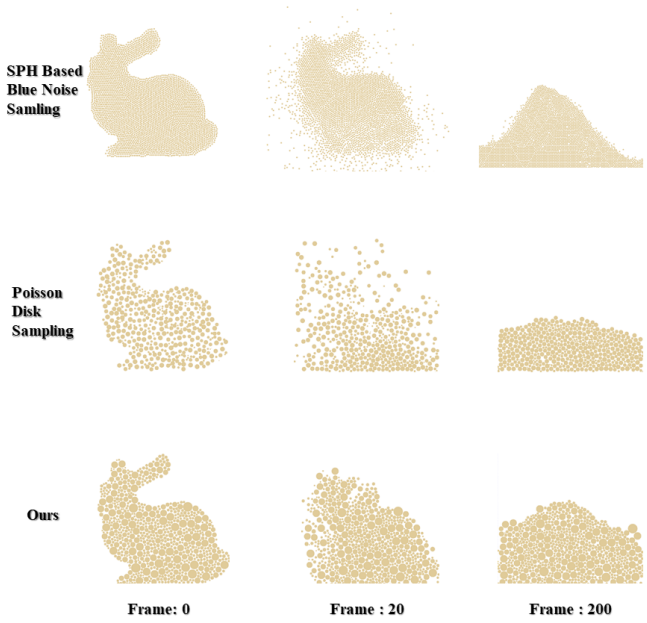


Fig. 10 Comparative experiments in sand simulation

References

1) B. Adams, M. Pauly, R. Keiser, L. J. Guibas: "Adaptively Sampled Particle Fluids", ACM Trans. on Graphics, Vol.26, No.3, pp.48:1-7 (2010).
2) R. Winchenbach, H. Hochstetter, A. Kolb: "Infinite Continuous Adaptivity for Incompressible SPH", ACM Trans. on Graphics, Vol.36, No.4, pp.102:1-10 (2017).
3) R. Winchenbach, A. Kolb: "Optimized Refinement for Spatially Adaptive SPH", ACM Trans. on Graphics, Vol.40, No.1, pp.8:1-15 (2021).
4) R. Bridson: "Fast Poisson Disk Sampling in Arbitrary Dimensions", Proc. of ACM SIGGRAPH Sketches, pp.22-es (2007).
5) M. Schwarz, H. P. Seidel: "Fast Parallel Surface and Solid Voxelization on GPUs", ACM Trans. on Graphics, Vol.29, No.6, pp.179:1-10 (2010).
6) H. Schechter, R. Bridson: "Ghost SPH for Animating Water", ACM Trans. on Graphics, Vol.31, No.4, pp.61:1-8 (2012).
7) M. Jiang, Y. Zhou, R. Wang, R. Southern, J. -J. Zhang: "Blue Noise Sampling Using an SPH-based Method", ACM Trans. on Graphics, Vol.34, No.6, pp.211:1-11 (2015).
8) F. De Goes, C. Wallez, J. Huang, D. Pavlov, M. Desbrun: "Power Particles: An Incompressible Fluid Solver Based on Power Diagrams", ACM Trans. on Graphics, Vol.34, No.4, pp.50:1-11 (2015).
9) X. Zhai, F. Hou, H. Qin, A. Hao: "Fluid Simulation with Adaptive Staggered Power Particles on GPUs", IEEE Trans. on Visualization and Computer Graphics, Vol.26, No.6, pp.2234-2246 (2018).
10) M. Aanjaneya, M. Gao, H. Liu, C. Batty, E. Sifakis: "Power Di-

Table 1 Table of porosity data with different particle sampling method

Example Name	Poisson Disk Sampling	SPH Based Blue Noise Sampling	Ours
Dragon	0.677	0.616	0.18
Horse	0.625	0.602	0.16
Lucy	0.605	0.613	0.172
Woody	0.703	0.598	0.161
Homer	0.708	0.655	0.172
Cheburashka	0.706	0.599	0.17
Teapot	0.713	0.635	0.164
Spot	0.772	0.653	0.168
Cow	0.704	0.649	0.17
Armadillo	0.695	0.604	0.174
Bunny	0.761	0.704	0.124
Average	**0.697**	**0.629**	**0.165**

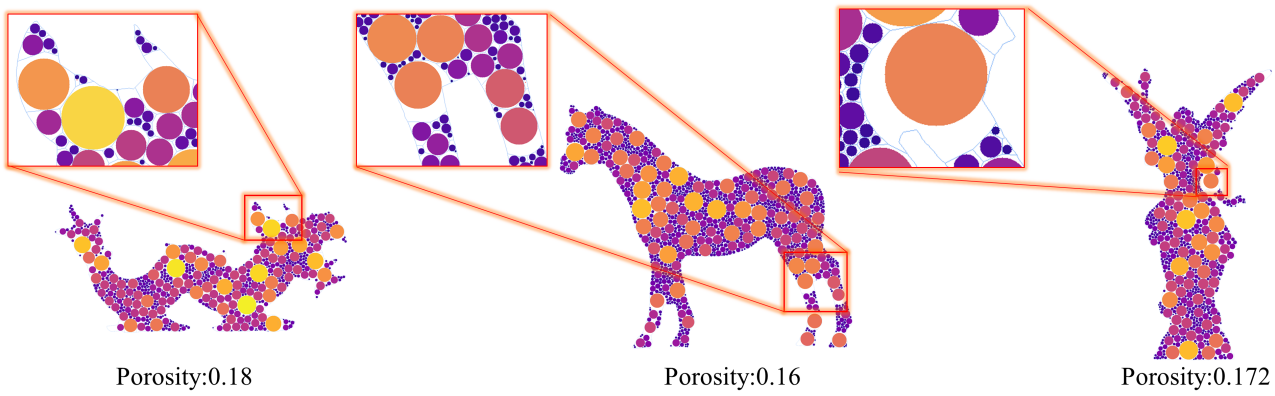


Fig. 11 Some of the failure cases

agrams and Sparse Paged Grids for High Resolution Adaptive Liquids", ACM Trans. on Graphics, Vol.36, No.4, pp.140:1-12 (2017).

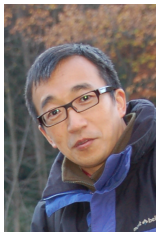
11) N. Ray, D. Sokolov, S. Lefebvre, B. Lévy: "Meshless Voronoi on the GPU", ACM Trans. on Graphics, Vol.37, No.6, pp.265:1-12 (2018).
12) X. Liu, L. Ma, J. Guo, D. -M. Yan: "Parallel Computation of 3D Clipped Voronoi Diagrams", IEEE Trans. on Visualization and Computer Graphics, Vol.28, No.2, pp.1363-1372 (2020).
13) J. Zheng, T. -S. Tan: "Computing Centroidal Voronoi Tessellation Using the GPU", Proc. of the ACM Symposium on Interactive 3D Graphics and Games, pp.14:1-9 (2020).
14) J. Basselin, L. Alonso, N. Ray, D. Sokolov, S. Lefebvre, B. Lévy: "Restricted Power Diagrams on the GPU", Computer Graphics Forum, Vol.40, No.2, pp.1-12 (2021).
15) M. Balzer, T. Schlömer, O. Deussen: "Capacity-Constrained Point Distributions: A Variant of Lloyd's Method", ACM Trans. on Graphics, Vol.28, No.3, pp.86:1-8 (2009).
16) M. Balzer: "Capacity-Constrained Voronoi Diagrams in Continuous Spaces", Proc. of the IEEE Symposium on Voronoi Diagrams, pp.79-88 (2009).
17) Y. Liu, W. Wang, B. Lévy, F. Sun, D. -M. Yan, L. Lu, C. Yang: "On Centroidal Voronoi Tessellation—Energy Smoothness and Fast Computation", ACM Trans. on Graphics, Vol.28, No.4, pp.101:1-17 (2009).
18) R. Chen, C. Gotsman: "Parallel Blue-noise Sampling by Constrained Farthest Point Optimization", Computer Graphics Forum, Vol.31, No.5, pp.1775-1785 (2012).
19) M. Balzer, O. Deussen, C. Lewerentz: "Voronoi Treemaps for the Visualization of Software Metrics", Proc. of the ACM Symposium on Software Visualization, pp.165-172 (2005).
20) A. Nocaj, U. Brandes: "Computing Voronoi Treemaps: Faster, Simpler, and Resolution-independent", Computer Graphics Forum, Vol.31, No.3, pp.855-864 (2012).
21) W. Rungjiratananon, Z. Szego, Y. Kanamori, T. Nishita: "Real-time Animation of Sand-water Interaction", Computer Graphics Forum, Vol.27, No.7, pp.1887-1893 (2008).
22) T. Lenaerts, B. Adams, P. Dutré: "Porous Flow in Particle-based Fluid Simulations", ACM Trans. on Graphics, Vol.27, No.3, pp.49:1-8 (2008).
23) T. Lenaerts, P.Dutré: "Mixing Fluids and Granular Materials", Computer Graphics Forum, Vol.28, No.2, pp.213-218 (2009).
24) R. Narain, A. Golas, M. -C. Lin: "Free-flowing Granular Materials with Two-way Solid Coupling", ACM Trans. on Graphics, Vol.29, No.6, pp.173:1-10 (2010).
25) G. Klár, T. Gast, A. Pradhana, C. Fu, C. Schroeder, C. Jiang, J. Teran: "Drucker-prager Elastoplasticity for Sand Animation", ACM Trans. on Graphics, Vol.35, No.4, pp.103:1-12 (2016).
26) T. Yang, J. Chang, B. Ren, M. -C. Lin, J. -J. Zhang, S. -M. Hu: "Fast Multiple-fluid Simulation Using Helmholtz Free Energy", ACM Trans. on Graphics, Vol.34, No.6, pp.201:1-11 (2015).
27) T. Yang, J. Chang, M. -C. Lin, R. R. Martin, J. -J. Zhang, S. -M. Hu: "A Unified Particle System Framework for Multi-phase, Multi-material Visual Simulations", ACM Trans. on Graphics, Vol.36, No.6, pp.224:1-13 (2017).
28) A. P. Tampubolon, T. Gast, G. Klár, C. Fu, J. Teran, C. Jiang, K. Museth: "Multi-species Simulation of Porous Sand and Water Mixtures", ACM Trans. on Graphics, Vol.36, No.4, pp.105:1-11 (2017).
29) M. Gao, A. Pradhana, X. Han, Q. Guo, G. Kot, E. Sifakis,

C. Jiang:"Animating Fluid Sediment Mixture in Particle-laden Flows", ACM Trans. on Graphics, Vol.37, No.4, pp.149:1-11 (2018).
30) Y. Yue, B. Smith, P. -Y. Chen, M. Chantharayukhonthorn, K. Kamrin, E. Grinspun:"Hybrid Grains: Adaptive Coupling of Discrete and Continuum Simulations of Granular Media", ACM Trans. on Graphics, Vol.37, No.6, pp.283:1-19 (2018).
31) X. Wang, M. Fujisawa, M. Mikawa:"Visual Simulation of Soil-Structure Destruction with Seepage Flows", Proc. of the ACM on Computer Graphics and Interactive Techniques, Vol.4, No.3, pp.41:1-18 (2021).
32) L. Zheng, Y. Yao, W. Wu, B. Xu, G. Zhang:"A Novel Computation Method of Hybrid Capacity Constrained Centroidal Power Diagram", Computers and Graphics, Vol.97, pp.108-116 (2021).
33) S. -Q. Xin, B. Lévy, Z. Chen, L. Chu, Y. Yu, C. Tu, W. Wang:"Centroidal Power Diagrams with Capacity Constraints: Computation, Applications, and Extension", ACM Trans. on Graphics, Vol.35, No.6, pp.244:1-12 (2016).
34) F. Aurenhammer:"Power Diagrams: Properties, Algorithms and Applications", SIAM Journal on Computing, Vol.16, No.1, pp.78-96 (1987).
35) J. M. Steele:Non-Uniform Random Variate Generation, Springer (1987)
36) C. B. Barber, D. P. Dobkin, H. Huhdanpaa:"The Quickhull Algorithm for Convex Hulls", ACM Trans. on Mathematical Software, Vol.22, No.4, pp.469-483 (1996).
37) P. Felkel, S. Obdrzalek:"Straight Skeleton Implementation", Proc. of Spring Conference on Computer Graphics, pp.210-218 (1998).

(Received March 10, 2022)
(Revised July 7, 2022)

Xu WANG
He received his BS degree in information management and information system from Shanghai University in 2016, and MS degree in media science from Tokyo University of Technology in 2020. He is currently a PhD candidate at Physics Based Computer Graphics Laboratory, University of Tsukuba. His research interests include physics-based simulation, style transfer, and all the relevant topics in computer graphics.

Makoto FUJISAWA (*Member*)
He is currently an Associate Professor in the Faculty of Library, Information and Media Science, University of Tsukuba since 2021. He received B.Eng., M.Eng., and Ph.D. degrees in mechanical engineering from Shizuoka University in 2003, 2005, and 2008 respectively. He worked for Nara Institute of Science and Technology from 2008 to 2010 and University of Tsukuba from 2011 to 2020 as an assistant professor. His research interests include computer graphics and physics simulation. He is a member of ACM, IEEE CS, IIEEJ, IPSJ and VRSJ.



Masahiko MIKAWA
He is currently an Associate Professor in the Faculty of Library, Information and Media Science, University of Tsukuba, Japan since 2006. He received B.Eng., M.Eng., and Ph.D. degrees from Osaka University in 1992, 1994 and 2001 respectively. He worked for NTT Access Network Systems Laboratories from 1994 to 2001, NTT Service Integration Laboratories from 2001 to 2003 and was a Lecturer in the Graduate School of Library, Information and Media Studies, University of Tsukuba from 2003 to 2006. He is a member of RSJ, SICE, SOFT and IEEE.