Multi-Scale Modeling and Rendering of Granular Materials

Johannes Meng\textsuperscript{2,1} Marios Papas\textsuperscript{1,3} Ralf Habel\textsuperscript{1} 
Carsten Dachsbacher\textsuperscript{2} Steve Marschner\textsuperscript{4} Markus Gross\textsuperscript{1,3} Wojciech Jarosz\textsuperscript{1,5}\textsuperscript{*}

\textsuperscript{1}Disney Research Z"urich \textsuperscript{2}Karlsruhe Institute of Technology \textsuperscript{3}ETH Z"urich \textsuperscript{4}Cornell University \textsuperscript{5}Dartmouth College

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1.png}
\caption{We propose a multi-scale procedural approach for modeling granular materials. The user specifies the bounding shape for the aggregate material (top left), selects a pre-packed tile of grain bounding spheres (top middle), within which we instantiate randomly rotated copies of the selected exemplar grains (bottom left) according to the specified mixing ratios. The SANDCASTLE contains about 2 billion grains, each composed of approximately 200k triangles. We report the high-order / total render times in hours and the variance in parentheses. Our approach (top half) renders the high-order scattering over 12\times (50 vs. 628 hrs) faster than explicitly path tracing (EPT) the individual grains (bottom half) while providing visually indistinguishable results. The insets on the right provide equal time and equal variance comparisons.}
\end{figure}

\section{Abstract}

We address the problem of modeling and rendering granular materials—such as large structures made of sand, snow, or sugar—where an aggregate object is composed of many randomly oriented, but discernible grains. These materials pose a particular challenge as the complex scattering properties of individual grains, and their packing arrangement, can have a dramatic effect on the large-scale appearance of the aggregate object. We propose a multi-scale modeling and rendering framework that adapts to the structure of scattered light at different scales. We rely on path tracing the individual grains only at the finest scale, and—by decoupling individual grains from their arrangement—we develop a modular approach for simulating longer-scale light transport. We model light interactions within and across grains as separate processes and leverage this decomposition to derive parameters for classical radiative transport, including standard volumetric path tracing and a diffusion method that can quickly summarize the large scale transport due to many grain interactions. We require only a one-time precomputation per exemplar grain, which we can then reuse for arbitrary aggregate shapes and a continuum of different packing rates and scales of grains. We demonstrate our method on scenes containing mixtures of tens of millions of individual, complex, specular grains that would be otherwise infeasible to render with standard techniques.

\textbf{CR Categories:} 1.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Raytracing;

\textbf{Keywords:} physically based rendering, granular media

\section{Introduction}

In this paper we consider rendering materials consisting of large assemblies of macroscopic granules. Such granular materials are common in our everyday environment: sand, gravel, and snow; sugar, salt, ground spices, laundry detergent; ocean spray or bubbles in a carbonated beverage—any large pile or aggregate object consisting of randomly oriented grains in which the individual scatterers are discernible (see Figure 2). Common among all these examples is the potential for detailed appearance at the scale of grains, but smooth large-scale appearance of the aggregate due to multiple scattering between grains. Rendering granular materials accurately and efficiently at arbitrary scales remains an open problem. Individual grains can have complex shapes and complex scattering behavior, while at the same time they can have high albedo, so that long paths with many scattering events can remain important (see Figure 3).

Treating each individual grain as explicit geometry and simulating global light transport using path tracing [Kajiya 1986] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains. At the other extreme, the aggregate object could be interpreted as a continuous medium, the smooth, large-scale appearance of which may be well expressed by participating media rendering techniques [Cerezo et al. 2005] and its variants is a general solution, but is only practical for small collections of grains.
We model a granular material as a procedurally defined, tiled assembly of objects representing individual grains, and we use three models for the transport of light in the material. To capture the appearance of visible grains, the most detailed model explicitly path-traces the grain geometry (explicit path tracing, or EPT). To more efficiently capture larger-scale transport above the scale of grains, we approximate the granular material as a continuous medium and render using volumetric path tracing (VPT). To avoid the need to trace long paths, for scales above the mean free path of the volumetric medium we use a diffusion-based approximation.

Using these three levels of approximation in a practical rendering system requires solving two fundamental problems. First, we need to obtain parameters for each rendering technique that are visually consistent with the finest scale. We do this without any expensive scene-dependent precomputation that would preclude modifying the shape or composition of the aggregate object. To achieve this, we develop a stochastic “teleportation” model of light transport which accounts for intra-grain transport and combine it with analytic estimates for inter-grain propagation imported from the physics literature [Torquato 2001]. After a one-time precomputation for an individual grain, we can reuse the results to derive parameters for arbitrary aggregate shapes and a continuum of different packing rates and scales. Second, we need to determine when to use each of the three approximations to obtain maximum efficiency gain while remaining visually accurate, which we achieve using carefully designed heuristics based on how paths diverge in the medium and how deeply they penetrate the medium.

Our resulting multi-scale algorithm is able to efficiently render vast scenes consisting of millions to billions of individual grains (e.g. Figure 1 right).

2 Related Work

Aggregate scattering solutions. Condensing aggregate scattering behavior of complex geometric structures into more compact scattering functions has long been a research problem in computer graphics. Reflection from complex surfaces can be modeled using analytic micro-facet models [Torrance and Sparrow 1967], or by tabulating light scattered off real [Matusik et al. 2003] or virtual [Westin et al. 1992; Ashikhmin et al. 2000; Kimmel and Baranoski 2007; Sadeghi et al. 2012] surfaces. We also use a Monte Carlo approach to tabulate compact scattering functions for a single grain. Some work also expresses mesoscopic volumetric structure, propagation, and scattering using bidirectional texture functions [Dana et al. 1999; Filip and Haindl 2009], voxels [Kajiya and Kay 1989; Neyret 1998], or similar representations [Chen et al. 2004; Tong et al. 2005]. Pharr and Hanrahan [2000] formalized the notation of using aggregate...
We leverage work from sphere packings to inspire both a multi-work in hair [Moon and Marschner 2006; Zinke and Weber 2006; A particularly well-studied sub-problem is that of densely packed dry sand ranges from 0.55–0.61 [McWhorter and Sunada 1977], and using the standard RTE, which is valid at low volume fractions or a data-driven procedure to fit such “effective RTE properties” to medium (with statistically dependent scattering) is approximated. We likewise leverage an explicit geometric representation for granules at the finest scale, but assume procedurally instantiated, randomly oriented grains with arbitrarily complex geometry.

**Densely packed media.** The packing density or packing rate $f$ is the volume fraction that grains occupy relative to the background medium [Dullien 1991]. For suspended media, such as granules in water or bubbles in ice, arbitrarily low packing rates can be achieved. The classical RTE is a reasonable model for packing rates up to around 0.1 [Randrianalisoa and Baillis 2010]. However, non-point scattering effects start to significantly affect the aggregate appearance at higher packing rates. Unfortunately, many materials we are interested in rendering have significantly higher packing rates. Bagged white sugar has a packing rate of $\sim 0.46$ [Bubnik et al. 1998], dry sand ranges from 0.55–0.61 [McWhorter and Sunada 1977], and snow has packing rates from 0.1–0.7 [Cuffey and Paterson 2010].

A particularly well-studied sub-problem is that of densely packed hard spheres. So-called “poured random packings” result from pouring hard spheres into a bed, and induce packing rates in the range of 0.609–0.625; shaking the bed of spheres to densify can reach rates above 0.625 [Dullien 1991]. Song et al. [2008] recently derived an analytic maximum density of $\sim 0.634$ for this problem. We leverage work from sphere packings to inspire both a multi-scale modeling metaphor and to enable multi-scale rendering with minimal precomputation.

**Non-classical RTE methods outside graphics.** Understanding the optical and heat transfer properties of densely packed media is important in many fields, including thermal engineering, atmospheric sciences, and nuclear reactor physics. Accurate solutions can be obtained using a full wave approach [Foldy 1945] and solving Maxwell’s equations [Durant et al. 2007], but are computationally infeasible and not necessary for graphics-related problems.

A popular alternative is to make the “homogeneous phase approximation” [Randrianalisoa and Baillis 2009], where the densely packed medium (with statistically dependent scattering) is approximated using the standard RTE, which is valid at low volume fractions or for long-scale transport. Randrianalisoa and Baillis [2010] proposed a data-driven procedure to fit such “effective RTE properties” to the results of a Monte Carlo random walk through a discrete granular medium. This requires a separate, scene-dependent simulation, akin to Moon et al., which we avoid. Instead, we first model radiative transport using a non-classical process with “teleportation” and derive compound parameters from that.

Singh and Kaviany [1992] model dependent radiative transfer by precomputing a scattering function with positional offsets for a large, smooth, dielectric spherical particle. Replacing the standard phase function in the Discrete Ordinates Method [Chandrasekar 1960] with it allows them to account for intra-grain transport effects. We also precompute a scattering function with position offsets, but consider grains of arbitrary geometry and material properties. We use this to augment volumetric path tracing with stochastic “teleportation” through grains. This, combined with analytic estimates for inter-grain distances [Torquato 2001], allows us to model the free-flight distribution of a granular medium.

Donovan et. al. [2003] used a two-step process where they sampled chord lengths for transport between grains from a statistical distribution and computed intra-grain transport by instantiating grain geometry. Our teleportation model is very similar, but uses a statistical representation for intra-grain transport. We currently rely on a standard diffusion approximation of exponentially-distributed paths, though recent diffusion derivations for non-exponentially distributed paths [d’Eon 2013] might further improve our accuracy.

### 3 Granular Model and Algorithm Overview

We propose a multi-scale modeling and rendering framework for granular materials that adapts to the structure of scattered light at different scales. In this section we introduce our stochastic model for granular materials (Section 3.1) and describe our three distinct rendering approaches in succession, which rely on explicitly path tracing the grain geometry (EPT: Section 3.2), volumetrically path tracing a homogeneous medium (VPT: Section 3.3), and applying the diffusion approximation (DA: Section 3.4). We initially assume suitable parameters are provided for each method. In the sections that follow we show how to obtain parameters that make VPT and DA consistent with EPT at larger scales (Section 4) without requiring a per-scene precomputation, and explain our criteria for switching between the methods to obtain the best performance while remaining visually accurate (Section 5). Figure 4 illustrates the various inputs and stages of our approach.

#### 3.1 Stochastic Granular Model

The input to our random model for granular media (refer to green items in Figure 4) consists of a surface describing the boundary of the aggregate object, a model for the geometry and material properties of individual grains, numbers describing the size and packing rate of the grains composing the aggregate, and numbers defining the mixing weights if multiple grain types are used. Our basic assumption is that the interior of the bounding surface is filled with many non-overlapping grains that are randomly positioned and oriented. The goal of our granular model is to procedurally define a random but repeatable collection of individual grains that fill the medium and do not intersect one another, while keeping memory requirements modest.

**Tiled sphere packings.** To avoid explicitly modeling and storing the arrangement of millions of packed grains, we rely on a procedurally tiled infinite field of packed spheres to fill the volume of the aggregate shape. Each sphere contains one grain, and since the spheres do not overlap the grains cannot overlap. This approach greatly simplifies many parts of our system, but limits the maximum

The basic building block of our procedural sphere packing is a tile.

EPT consists of an aggregate mesh, packing rate & scale of spheres,

grains and less for non-spherical grains (see Section 2). This is a

dimensional tile index and the sphere id. This ensures consistent

used for each individual grain is a deterministic function of the three-

could be used to remove tiling artifacts completely if necessary.

some repetition artifacts, we found that in practice they hardly matter

= 0

tiles for various packing ratios

Figure 4: When rendering the granular material (bottom), primary
rays from the eye start out by explicitly path tracing (EPT) grains,
then after enough scattering events take place (Section 5), the path
is continued using volumetric path tracing (VPT), and eventually
terminated using a diffusion calculation (DA) to approximate the
contribution of all further interactions. The input (top, green) to
EPT consists of an aggregate mesh, packing & scale of spheres,
example grains, and mixing ratios of these grains. We combine grain
scattering statistics (blue)—calculated in a one-time, per-grain-type
preprocess—with the packing rate and mixing ratios to obtain RTE
parameters (orange) needed for the VPT and DA methods.

Figure 5: To intersect procedural granular media such as this bunny
of cubical grains (left), we consider all bounding spheres, from an
infinite tiled space, that reside inside a given aggregate boundary-

mesh. We perform intersection tests with randomly rotated,
instanced grain geometry (right) inside each bounding sphere.

attainable packing fraction to approximately 0.634 for spherical
grains and less for non-spherical grains (see Section 2). This is a
limitation especially if the grains are highly non-spherical, such as
gains of rice, as we will discuss in Section 7.

The basic building block of our procedural sphere packing is a tile,
a cuboid containing a number of non-intersecting spheres. We use the
algorithm of Skoge et al. [2006] to generate a dictionary of periodic
tiles for various packing ratios $f$. Figure 1 shows an example tile for
$f = 0.63$. Though tilings generated with these blocks may exhibit
some repetition artifacts, we found that in practice they hardly matter
due to the randomization involved in our grain instantiation which
we describe next. Aperiodic tiling [Peytavie et al. 2009] approaches
could be used to remove tiling artifacts completely if necessary.

**Randomized instantiation.** We treat each distinct sphere of the
infinite field as a location for instantiating a randomly chosen grain
from the set of provided exemplars. The exemplar grains are chosen
according to the user-specified mixing probabilities (see Figure 1).
The aggregate mesh provided by the user (assumed to be watertight
with consistent normals) defines the region in space where grains
are instantiated and each instance is randomly rotated within its
bounding sphere. Figure 5 illustrates this idea. The random seed
used for each individual grain is a deterministic function of the three-
dimensional tile index and the sphere id. This ensures consistent


grain instantiation for all rays when needed.


cuboid containing a number of non-intersecting spheres. We use the
algorithm of Skoge et al. [2006] to generate a dictionary of periodic
tiles for various packing ratios $f$. Figure 1 shows an example tile for
$f = 0.63$. Though tilings generated with these blocks may exhibit
some repetition artifacts, we found that in practice they hardly matter
due to the randomization involved in our grain instantiation which
we describe next. Aperiodic tiling [Peytavie et al. 2009] approaches
could be used to remove tiling artifacts completely if necessary.

**Randomized instantiation.** We treat each distinct sphere of the
infinite field as a location for instantiating a randomly chosen grain
from the set of provided exemplars. The exemplar grains are chosen
according to the user-specified mixing probabilities (see Figure 1).
The aggregate mesh provided by the user (assumed to be watertight
with consistent normals) defines the region in space where grains
are instantiated and each instance is randomly rotated within its
bounding sphere. Figure 5 illustrates this idea. The random seed
used for each individual grain is a deterministic function of the three-
dimensional tile index and the sphere id. This ensures consistent

grain instantiation for all rays when needed.


cuboid containing a number of non-intersecting spheres. We use the
algorithm of Skoge et al. [2006] to generate a dictionary of periodic
tiles for various packing ratios $f$. Figure 1 shows an example tile for
$f = 0.63$. Though tilings generated with these blocks may exhibit
some repetition artifacts, we found that in practice they hardly matter
due to the randomization involved in our grain instantiation which
we describe next. Aperiodic tiling [Peytavie et al. 2009] approaches
could be used to remove tiling artifacts completely if necessary.

**Randomized instantiation.** We treat each distinct sphere of the
infinite field as a location for instantiating a randomly chosen grain
from the set of provided exemplars. The exemplar grains are chosen
according to the user-specified mixing probabilities (see Figure 1).
The aggregate mesh provided by the user (assumed to be watertight
with consistent normals) defines the region in space where grains
are instantiated and each instance is randomly rotated within its
bounding sphere. Figure 5 illustrates this idea. The random seed
used for each individual grain is a deterministic function of the three-
dimensional tile index and the sphere id. This ensures consistent

grain instantiation for all rays when needed.


cuboid containing a number of non-intersecting spheres. We use the
algorithm of Skoge et al. [2006] to generate a dictionary of periodic
tiles for various packing ratios $f$. Figure 1 shows an example tile for
$f = 0.63$. Though tilings generated with these blocks may exhibit
some repetition artifacts, we found that in practice they hardly matter
due to the randomization involved in our grain instantiation which
we describe next. Aperiodic tiling [Peytavie et al. 2009] approaches
could be used to remove tiling artifacts completely if necessary.

**Randomized instantiation.** We treat each distinct sphere of the
infinite field as a location for instantiating a randomly chosen grain
from the set of provided exemplars. The exemplar grains are chosen
according to the user-specified mixing probabilities (see Figure 1).
The aggregate mesh provided by the user (assumed to be watertight
with consistent normals) defines the region in space where grains
are instantiated and each instance is randomly rotated within its
bounding sphere. Figure 5 illustrates this idea. The random seed
used for each individual grain is a deterministic function of the three-
dimensional tile index and the sphere id. This ensures consistent

grain instantiation for all rays when needed.


cuboid containing a number of non-intersecting spheres. We use the
algorithm of Skoge et al. [2006] to generate a dictionary of periodic
tiles for various packing ratios $f$. Figure 1 shows an example tile for
$f = 0.63$. Though tilings generated with these blocks may exhibit
some repetition artifacts, we found that in practice they hardly matter
due to the randomization involved in our grain instantiation which
we describe next. Aperiodic tiling [Peytavie et al. 2009] approaches
could be used to remove tiling artifacts completely if necessary.

**Randomized instantiation.** We treat each distinct sphere of the
infinite field as a location for instantiating a randomly chosen grain
from the set of provided exemplars. The exemplar grains are chosen
according to the user-specified mixing probabilities (see Figure 1).
The aggregate mesh provided by the user (assumed to be watertight
with consistent normals) defines the region in space where grains
are instantiated and each instance is randomly rotated within its
bounding sphere. Figure 5 illustrates this idea. The random seed
used for each individual grain is a deterministic function of the three-
dimensional tile index and the sphere id. This ensures consistent

grain instantiation for all rays when needed.

3Our procedure defines a grain to be inside the aggregate shape if its center
is within the aggregate boundary mesh. Instantiating grains for spheres that
lie entirely within the mesh is also possible with more costly overlap tests.

**Algorithm 1** Procedural Grain Instantiation

1: function INTERSECTGRAINS(Ray $r$, Float $t_{\text{min}}$, Float $t_{\text{max}}$)
2: while $t_{\text{min}} < t_{\text{max}}$ do
3: $t_v = \text{VoxelIndex}(r, t_{\text{min}})$
4: if PartiallyInside($t_v$) then
5: $o_v = \text{VoxelOrigin}(v)$
6: $o = \text{Origin}(r) - o_v$
7: if IntersectTile($o$, Direction($r$), $t_{\text{min}}$, $t_{\text{max}}$) then
8: $p = \text{BoundingSphereCenter}() + o_v$
9: if SurelyInside($p$) or PointInside($p$) then
10: if IntersectInstance($r$, $t_{\text{min}}$, $t_{\text{max}}$, $p$) then
11: return True
12: end if
13: end if
14: $t_{\text{min}} = \text{BoundingSphereExit}()$
15: end if
16: end if
17: $t_{\text{min}} = \text{FindNextVoxelT}(r, t_{\text{min}})$
18: end while
19: return False
20: end function

3.2 Explicit Path Tracing

For the finest level, we use an algorithm which explicitly path traces
against the actual geometry of the individual grains generated by
the procedural model just described. We dub this technique explicit
path tracing (EPT) and illustrate the expected input in Figure 4. The
core challenge with this approach is representing the vast collection
of grains and efficiently tracing rays through it.

**Ray tracing grains.** Even with instancing, storing the transfor-
mation matrices for the collection of billions of grain instances is
impractical. Instead, we procedurally instantiate and discard the
grains on demand during ray traversal of the scene. To facilitate this,
we voxelize the boundary mesh using two bits per voxel, marking
voxels as either “fully outside,” “fully inside,” or “partially inside.”
One voxel corresponds, in size and position, to one instance of the
base tile.

During ray tracing, we intersect the ray with the voxel grid to obtain
the first intersection with a voxel that is either “fully inside” or
“partially inside.” We then intersect the bounding spheres overlapping
the voxel. If the voxel is marked “fully inside,” we use the first
intersected bounding sphere. If it is marked “partially inside,” we
first have to determine if the bounding sphere’s center is contained
in the aggregate shape. We do this by tracing a ray from the sphere
center against the aggregate shape.

Upon hitting a bounding sphere, we select a random example
grain (see Randomized instantiation above) according to our mixing
weights and temporarily transform the ray into the (randomly
rotated) local coordinate system of the grain. We then intersect
the grain geometry to determine the final intersection point. If there is
no intersection, we proceed to the next bounding sphere intersected
by the ray (if any). Algorithm 1 lists pseudo-code for this procedure.

**Rendering.** With this grain intersection machinery in place, we
employ path tracing using next event estimation and multiple im-
portance sampling [Pharr and Humphreys 2010] as our finest level
rendering technique. Note that while next event estimation may be useful outside the aggregate object and close to its boundary, there are virtually no unblocked direct connections to light sources from within the granular material. Furthermore, most of our granular materials consist of grains with specular boundaries, rendering shadow rays useless and underpinning the need to switch to other approaches.

### 3.3 Volumetric Path Tracing

While EPT retains the necessary grain-level details present in low-order scattering, it quickly becomes impractical for higher-order transport when rays become incoherent. Luckily, for incoherent rays it is less important to precisely track which rays hit which grains, so we transition (Figure 4) to a continuous volumetric representation for medium-order transport, which we simulate using volumetric path tracing (VPT) [Kajiya 1986; Rushmeier 1988].

For this rendering approach we assume each bounding aggregate shape is filled with an index-matched homogeneous participating medium defined by its extinction coefficient \( \sigma_t \), single-scattering albedo \( \alpha_s \), and the scattering phase function \( \Phi(\cos \theta) \) (marked in orange in Figure 4). We discuss how to obtain these parameters from the properties of the granular packing in Section 4.

Apart from the fact that VPT does not need to instance or intersect grain geometry, transitioning to VPT allows us to leverage shadow connections from inside the granular medium, which was virtually impossible with EPT. As we will see in Section 6, this allows VPT to efficiently render scenes that are intrinsically hard for EPT, such as large aggregates of highly specular, non-absorbing grains.

### 3.4 Diffusion Approximation

While VPT has significantly lower variance than EPT, it can still be computationally expensive for highly scattering materials which require long transport paths. We therefore transition (Figure 4) to a rendering technique based on a fast diffusion approximation [Stam 1995; Jensen et al. 2001] to account for large-scale transport due to high-order multiple scattering. Diffusion allows us to effectively short-circuit the recursion of VPT and approximate long scattering paths directly.

We switch to diffusion by sampling a location on the boundary mesh and estimating the diffusion transport. We use a technique inspired by the method of Li et al. [2005], though we use d'Eon and Irving’s [2011] improved diffusion model instead of the classical dipole [Jensen et al. 2001], and integrate its contribution using Monte Carlo [Habel et al. 2013]. We also propose a different virtual source placement procedure which we found produces slightly better results than the approach described by Li et al., and we account for the finite thickness of the medium by using a multipole expansion [Donner and Jensen 2005]. We detail these changes in Appendix A. The parameters for this approach consist of the reduced medium parameters \( (\sigma_t', \sigma_s') \) which we obtain from the RTE parameters \( (\sigma_t, \alpha_s, \Phi) \) in the usual way [Jensen et al. 2001] using similarity theory.

### 3.5 Importance of Using a Hybrid Method

While it would be possible to use any of the aforementioned methods in isolation, we aim to combine them into a hybrid approach since we wish to render granular materials at multiple scales, potentially in a single shot (e.g., a sand dune that is close to the camera at the bottom of the frame, and far away at the top). In such scenarios, using EPT alone would likely be prohibitively expensive, while the more approximate techniques would not faithfully reproduce fine surface details where individual grains are discernible (see Figure 11). One of our core contributions is showing how to systematically combine these disparate methods and representations to ensure visual consistency between grains visible at vastly different scales both across the image or across time in an animation.

### 4 Precomputation and RTE Parameters

Our goal in this section is to derive RTE parameters \((\sigma_s, \sigma_t, \Phi)\) for a classical homogeneous medium which match the statistical scattering behavior of the discrete granular material. This task is non-trivial since granular materials with high packing rates do not actually satisfy a core assumption of the RTE—namely, that scatterers are sufficiently small and well-separated so that each interaction may be considered statistically independent.

To handle this seeming incompatibility, we first (Section 4.1) introduce a statistical, non-classical model of light transport which we call teleportation transport (TT). This intermediate model allows us to reason about the non-point scattering effects present in granular materials and will provide a stepping stone to derive classical RTE parameters (Section 4.2) for our VPT and DA approaches without the need for a scene-dependent precomputation.

#### 4.1 The Teleportation Transport Model

We wish to create a purely statistical model which characterizes paths constructed by path tracing the granular material as described previously in Section 3.2. While TT introduces several statistical independence assumptions, we design these to closely approximate the EPT behavior once rays have been sufficiently randomized by a series of explicit events.

With this analogy in mind, our TT model consists of two core steps which occur in alternation. The first step considers inter-grain transport by randomly deciding how far along a ray to move before the next interaction with a grain bounding sphere occurs. Upon hitting a bounding sphere, the second step considers intra-grain transport by scattering and “teleporting” the ray directly to a point where it exits the sphere. A deflection angle and spatial offset from the incoming point and direction are randomly selected to produce an outgoing point and direction from which the path is continued. This is done in such a way that the distribution of continuations is the same as is observed with the grain geometry, when averaged over all possible entry points and grain orientations. Note that this model is similar to a volumetric path tracing view of the RTE, but with the introduction of non-point scattering effects due to teleportation.

**Inter-grain transport.** To characterize inter-grain transport, we need to determine the distribution of free-flight distances from the point where a path exits one grain’s bounding sphere to the point where it enters the next grain’s. We could tabulate this distribution as Moon et al. [2007] did, by accumulating free-flight distances from many paths traced through a particular packing. We would, however, like to avoid a precomputation that depends on a particular packing.

We instead make use of prior work on transport in sphere packings—this is appropriate since the distances we are interested in are between interactions with bounding spheres, not with the grains themselves. Dixmier [1978] derived a simple model for free-flight distances in sphere packings, known in that context as outer chord-length distributions. He showed that these outer chord-lengths can be modeled by an exponential distribution

\[
p_\lambda(z) = \sigma_\phi e^{-\sigma_\phi z}, \quad \text{with} \quad \sigma_\phi = \frac{3}{4R \left(1 - f\right)},
\]

where \( R \) is the radius of the spheres, \( f \) is the sphere packing rate, and \( \lambda \) is the average distance between bounding sphere.
4.2 Classical RTE & Diffusion Parameters

Given the teleportation transport model just described, we wish to use it as an intermediate model to obtain RTE parameters ($\sigma_t, \alpha_s, \Phi$) describing an “equivalent” homogeneous participating media. Since our teleportation transport is a generalization of classical transport (with the addition of non-point scattering), there will inherently be some information loss in this conversion. In particular, classical transport has only one mechanism—propagation with exponential free-flight sampling dictated by $\sigma_t$—to affect the positions (distances) along rays. In TT, however, these distances are influenced not only by exponential inter-grain transport, but also by the intra-grain teleportation encoded in the TSDF. Our goal is therefore to extract all directional effects of the TSDF into the phase function $\Phi$ and combine the positional influences of the TSDF and the inter-grain transport into an effective extinction coefficient $\sigma_t$.

4.2.1 The Phase Function & Albedo

In the classical RTE, scattering interactions are assumed to happen at a single point. This amounts to assuming that outgoing direction $\vec{\omega}_o$ depends only on the incident direction $\vec{\omega}_i$. We can therefore extract the directional component of the TSDF and use this directly as a phase function $\Phi$. Due to symmetry of the random grain rotations, we additionally know that the phase function will depend solely on the cosine of the deflection angle $cos \theta = (\vec{\omega}_i, \vec{\omega}_o)$. This allows us to easily tabulate just a 1D distribution in the scatterometer setup by recording the dot product between the incident and outgoing directions of any ray that interacts with the grain. We found that tabulating this 1D distribution is practical, and gives superior results to fitting a simple parametric model like the Heney-Greenstein [1941]

Intra-grain transport. The goal of intra-grain transport is to statistically model the interaction of a random ray with a randomly oriented grain inside a bounding sphere. We could fully summarize the effect of a single grain interaction on a path by utilizing a virtual scatterometer setup akin to Westin et al.'s [1992] approach. This involves placing a single grain in a bounding sphere and tracing a collection of rays, distributed over directions and the cross section of the sphere. After interacting with the grain zero or more times, a path would either exit the bounding sphere or be absorbed. The points and directions upon entering $(x_i, \vec{\omega}_i)$ and exiting $(x_o, \vec{\omega}_o)$ of the bounding sphere (see illustration) characterize the scattering function $S(x_i, \vec{\omega}_i \rightarrow x_o, \vec{\omega}_o)$ of the grain (in the sense of scattering functions used in layer-oriented transport calculations [Pharr and Hanrahan 2000]). We call this function the teleportation scattering distribution function (TSDF), because it models scattering on a single grain by “teleporting” a photon from $x_i$ to $x_o$.

Like a BSSRDF [Jensen et al. 2001], the TSDF is parameterized over the complete set of rays entering and exiting a volume of space; in this way it is also similar to the BFSDF [Zinke and Weber 2007] used to describe scattering from fibers.

One could create a full rendering algorithm around the teleportation transport model, which would require storing the TSDF and sampling it and Equation (1) in alternation. We provide more details and an analysis of one prototype implementation in the supplemental material. In practice however, we found that we can obtain similar visual fidelity (see Figure 7) with significantly less implementation complexity by using the TT model only to obtain classical RTE parameters. This allows us to leverage standard RTE rendering methods, such as VPT and DA, in our approach. Additionally, deriving classical RTE parameters will require us to gather only a compact set of statistics of the TSDF from the virtual scatterometer, instead of exhaustively tabulating the full 8D function, as we discuss next.

4.2.1 The Phase Function & Albedo

In the classical RTE, scattering interactions are assumed to happen at a single point. This amounts to assuming that outgoing direction $\vec{\omega}_o$ depends only on the incident direction $\vec{\omega}_i$. We can therefore extract the directional component of the TSDF and use this directly as a phase function $\Phi$. Due to symmetry of the random grain rotations, we additionally know that the phase function will depend solely on the cosine of the deflection angle $cos \theta = (\vec{\omega}_i, \vec{\omega}_o)$. This allows us to easily tabulate just a 1D distribution in the scatterometer setup by recording the dot product between the incident and outgoing directions of any ray that interacts with the grain. We found that tabulating this 1D distribution is practical, and gives superior results to fitting a simple parametric model like the Heney-Greenstein [1941]

![Figure 6: We compare (top) Monte Carlo measurements against analytic estimates [Dixmier 1978] of the path segment lengths $p_b(z)$ between packed dielectric spheres in an infinite volume. We also compare (bottom) our full free-flight distribution $p_f(z)$ (Eq. 4) for three of our grain (mixtures) against Monte Carlo measurements. The packing rate is $f = 0.63$ for all grain types. The hit probability is $\beta = 0.53$ (Pink Salt), $\beta = 0.66$ (Flour), and $\beta = 0.63$ (Sand). The mean free paths are indicated by vertical lines.](image)

![Figure 7: Equal quality comparison (top) of a cube of glass beads, as shown in Moon et al. [2007, Fig. 9]. The last column shows individual low- and high-order contributions computed with EPT and VPT, respectively. MSE and variance vs. core hours for all three techniques (bottom).](image)
A ray may be absorbed in the interaction with a grain, in which case it never exits the grain bounding sphere, or it may experience a teleport step. During preprocessing, we estimate in our scatterometer simulation the mean teleport vector \( (x_o - x_i) \) of all rays that scatter with (and are not absorbed by) the grain. Due to symmetry of the random rotations, this mean teleport vector is always aligned with the original ray direction, and we denote its length \( \lambda_t \). By accounting for the probability of teleportation vs. absorption (the albedo \( \alpha_s \)), our final exponential free-flight distribution between exit points of grain interactions (refer to Figure 8) becomes:

\[
p_t(z) = \sigma t e^{-\sigma z}, \quad \text{with} \quad \sigma t = \frac{1}{\lambda t} = \frac{1}{\lambda_\beta + \alpha_s \lambda_s}.
\]

In Figure 6 (bottom), we compare \( p_t(z) \) for three grains from our example scenes to results obtained from explicitly path tracing grains in an infinite medium. Notably, the means of the distributions—indicated by vertical lines—match well, indicating that our analytic model is able to capture the average behavior of granular media, even though the true free-flight distribution is not an exponential.

### 4.2.3 Effective RTE and Diffusion Parameters

In summary, we extract just two average distributions, two probabilities, and a 1D distribution from our grain scatterometer simulations: \{\( \lambda_s, \lambda, \alpha_s, \beta, \Phi(\cos \theta) \)\}. In Table 1 we present the scalar statistics for the grains used in our results, and we visualize the phase function for two grains in the supplemental. From these, we obtain our effective RTE parameters by reusing the grain albedo \( \alpha_s \), and phase function \( \Phi(\cos \theta) \) directly, and by computing the extinction coefficient \( \sigma t \) using Equation 4. We compute effective RTE parameters for mixtures of different grains by computing a weighted average of their individual statistics. We then derive diffusion parameters by computing the mean scattering cosine \( g \) from \( \Phi(\cos \theta) \), and from that the reduced scattering coefficient \( \sigma t^* = (1 - g) \alpha t \), which gives a reduced extinction coefficient \( \sigma t^* = \sigma t^* + (\sigma t - \sigma t) \) and reduced albedo \( \alpha s^* = \alpha s^*/\sigma t \).

<table>
<thead>
<tr>
<th>glass</th>
<th>beads</th>
<th>flour</th>
<th>pink</th>
<th>salt</th>
<th>sand</th>
<th>snow</th>
<th>cane</th>
<th>brown</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \lambda_s )</td>
<td>0.00</td>
<td>0.87</td>
<td>0.97</td>
<td>0.85</td>
<td>0.86</td>
<td>0.77</td>
<td>0.77</td>
<td></td>
</tr>
<tr>
<td>( \lambda_t )</td>
<td>1.44</td>
<td>0.18</td>
<td>1.00</td>
<td>0.50</td>
<td>1.40</td>
<td>1.17</td>
<td>0.88</td>
<td></td>
</tr>
<tr>
<td>( \alpha_s )</td>
<td>1.00</td>
<td>1.00</td>
<td>0.99</td>
<td>0.68</td>
<td>1.00</td>
<td>1.00</td>
<td>0.76</td>
<td></td>
</tr>
<tr>
<td>( \beta )</td>
<td>1.00</td>
<td>0.66</td>
<td>0.53</td>
<td>0.63</td>
<td>0.65</td>
<td>0.71</td>
<td>0.71</td>
<td></td>
</tr>
</tbody>
</table>

### Validation.

In Figure 9, we validate our derivations by comparing the fluence and radiance fields produced within granular materials composed of dielectric spheres or snow grains. We visualize the illumination for various inclinations on a spherical detector surrounding a beam source inside the medium.

We obtain the data visualized in Figure 9 using a numerical simulation by tracing photons from a beam source in an infinite medium of unit radius grains. To ensure sufficient initial randomization, we initialize each photon at a random location and orientation in the infinite medium, and trace it until it has experienced 10 grain interactions. We take the resulting position and direction of the photon as the origin of the beam source, around which we center a detector sphere of a predefined radius (in our case, 7.5 or 15 mean free paths, which corresponds to 10 or 20 grain radii). We then continue tracing the photon until it exits the sphere.

Upon exiting the sphere, we parametrize the photon’s position and direction in the coordinate system shown in Figure 10: The inclination \( \theta \) describes how much the photon deviates from its initial direction, and the exitant direction \( \omega_o \) is expressed as its projection.
we describe our automatic criterion for transition between these

The fluence (Figure 9, top) computed using VPT with our RTE
various inclinations, albeit with some over-estimation of the forward
EPT result. This indicates that VPT could be used to precompute
scattering at small distances, but otherwise closely resembles the
traditional rendering scenario. The results show that our derived
parameters generally matches the EPT ground truth very well at
long-scale transport within granular materials. In isolation, however, neither VPT nor DA
can retain the high-frequency details of discernible grains, so next
we describe our automatic criterion for transition between these

rendering methods.

5 Switching between Rendering Techniques

We assume that primary rays are spawned outside the granular
material and begin rendering using explicit path tracing of the grain
group, so why it could be both beneficial and safe to switch from EPT to VPT
and DA, we consider the bundle of rays belonging to a pixel. As
long as these rays are somewhat coherent, EPT will perform reasonably
well. However, once the rays become statistically independent
by scattering in different directions, the variance from EPT would
become enormous since the chance of randomly hitting small light
sources tends to zero. Luckily, when rays are incoherent, there is
also no reason to keep track of which ray hits which grain and we
can safely switch to a smooth model like VPT or the DA. Our goal in
this section, is therefore to develop an automatic switching criterion
which allows us to use EPT when we need to—when rays are still
coherent on the pixel scale—and which switches to the smoother
VPT and DA techniques once rays become incoherent. Figure 11
shows that by intelligently combing our techniques with EPT, we
can retain small-scale details while also accurately approximating
the long-scale transport within granular materials.

EPT → VPT. To put this intuition into practice, we spawn a bundle
of $N = 16$ paths per pixel which are traced in lock-step through the
groups. Individual paths might be terminated (e.g. by escaping the
medium or due to Russian roulette), while others are traced further.
In this case, we simply continue with the bundle of remaining paths.
At each bounce $k$, we gauge the level of ray coherence by computing
the standard deviation $\sigma_k$ of the unterminated $N_k$ vertex positions
$(x_1, \ldots, x_{N_k})$. We switch to VPT when:

$$\sigma_k > \tau N_k / N$$  \hspace{1cm} (5)

Figure 10: The coordinate system used for plots in Figure 9. A
photon is traced from a beam source at location $p_i$ with direction
$\omega_i$. The photon follows a path (shown as a dotted line) until it exits
the detector sphere at $p_o$. The exitant direction $\omega_o$ is then projected
onto the blue tangent plane with normal $\omega_i$.

Figure 11: Diagonal comparison of proposed techniques (top left)
with EPT (bottom right) on a sphere composed of snow grains. We
compare four combinations of our methods: The first two combina-
tions show how VPT with and without diffusion compare against
EPT. The latter two show how preceding these methods with a few
bounces of EPT can accurately recover high-frequency detail.
where \( \tau \) is a user-specified multiple of the maximum grain radius \( r \); the \( N_\text{eff}/N \) factor allows bundles with few surviving rays to switch more quickly. For all our scenes we use a threshold of 4 grain radii. When \( \sigma_t \) exceeds this threshold, all paths in a bundle switch upon hitting the next grain. If only one ray of a bundle remains, we switch immediately at the next grain interaction.

To account for the fact that the VPT and DA techniques may converge faster than EPT, we can optionally dedicate it more computation time by probabilistically pruning paths when they attempt to switch. We achieve this using Russian roulette with an acceptance threshold \( P_a \). Later in this section, we show how to automatically compute the value of \( P_a \) which optimally balances the variance of the different techniques.

**VPT \( \rightarrow \) DA.** While diffusion can estimate multiple scattering very efficiently, it also introduces a number of approximations and assumptions (most notably the assumption of isotropic scattering and planar bounding geometry) that limit its accuracy in the general setting. For switching from VPT to DA, we adopt a criterion adapted from the work of Li et al. [2005] which aims to allow DA only when the approximations would not be too noticeable.

As in that work, our primary criterion for switching to diffusion is a minimum distance between the VPT path vertex \( x^k \) and the surface of the boundary mesh. While Li et al. used a threshold of one reduced mean free path, we use a threshold of \( d_{DA} = \min(1/\sigma^i_t, 0.5/\sigma_r) \) — that is, we switch if \( x^k \) is at least 1 reduced mean free path away from the boundary or at least half a diffuse mean free path. This allows us to accept diffusion connections more frequently for lower albedo materials which can be better represented using d’Eon and Irving’s improved diffusion model [2011]. Li et al. accelerate this distance calculation using a KD-tree filled with points uniformly sampled over the boundary mesh, but we found this to be impractical for optically thick granular media since the sampling would need to be excessively dense. Instead, we reuse the KD-tree acceleration structure of the aggregate mesh geometry, and perform a depth-first search to check whether any triangles overlap a sphere of radius \( d_{DA} \) centered at \( x^k \). The appendix details our diffusion source placement.

**High Order Acceptance Rate.** Since convergence of low-order paths computed with EPT is usually slower than convergence of high-order paths from VPT or DA, we can often reduce the total variance of the image by dedicating more sample budget to low-order paths without requiring each to also spend effort estimating high order scattering (which may already be converged). To accomplish this, we perform Russian roulette with an acceptance rate of \( P_a \leq 1 \) when switching between EPT and VPT. While we could set this manually as done by Moon et al. [2007], the optimal acceptance probability is scene and lighting dependent, making it difficult to estimate by hand. Instead, we propose to automatically compute the value of \( P_a \) that would minimize the variance of the final image.

Our approach requires a relatively inexpensive preprocess (taking about 1% of the total render time) during which we render a down-scaled image 1% of the original size (reduced by 10\( \times \) in both dimensions) with the same total number of samples per pixel as the final image. To ensure that the statistics of the switching criteria are not affected, we set the footprint of the camera ray packets to the size of the original (smaller) pixels. For each pixel in the low resolution preview we incrementally estimate the sample variance of the low-order \( V_L(x,y) \) and high-order \( V_H(x,y) \) contributions, and we also accumulate the total CPU time for computing these contributions as \( t_L(x,y) \) and \( t_H(x,y) \). The variance of the combined image as a function of the number of samples \( n \) and the acceptance rate \( P_a \) is approximately:

\[
V \approx \frac{1}{n} \left( \frac{V_L + V_H}{P_a} \right),
\]

where \( V_L \) and \( V_H \) are the averages across the image of \( V_L(x,y) \) and \( V_H(x,y) \), respectively. The total time needed for the image is approximately:

\[
t = n \left( t_L + P_a t_H \right),
\]

where \( t_L \) and \( t_H \) are the total CPU time needed for low order and high order paths respectively. By solving for the value of \( P_a \) which minimizes \( t \), we obtain:

\[
P_a = \sqrt{\frac{V_H}{V_H - V_L}}.
\]

In our results we report the render times both with \( P_a = 1 \) and with the optimal acceptance rate computed using Equation (8).

### 6 Results

We implemented our method in Mitsuba [Jakob 2010] as a new Integrator for our rendering algorithms and a new Shape primitive for our granular volumes. We rendered all results on a homogeneous cluster with nodes containing two 12-core Intel Xeon E5-2697v2 processors at 2.7 GHz with 64 GB RAM and report all render times in core-hours. For time related comparisons, we independently render many low sample count images across the machines and average the resulting floating-point images for the final result. In all results, RGB channels are rendered separately and then combined and tonemapped with an sRGB gamma curve.

For all our figures we estimate variance (in parentheses) by dividing the *time to unit variance* (ttuv) by the core hours used for the specific scene and method. Time to unit variance for each method is an estimate of the time needed to achieve a variance of 1, assuming \( 1/N \) variance reduction. We compute ttuv by rendering low sample count versions of the image, and then multiplying the average sample variance over all pixels with the average time needed to render a low sample count image.

In Table 2 we provide a summary of render times and analysis of variance for the main scenes shown in Figures 1, 12, and 13. We quantify the efficiency of each algorithm using ttuv. On the left half of the table we report ttuv and the resulting speedup compared to EPT for the overall render time, and on the right for only the high-order transport which we aim to accelerate with our approximations.

EPT+VPT provides a significant overall speedup over EPT in all our scenes, ranging from 2.1–30 \( \times \). This is largely due to VPT’s ability to create shadow connections to the light sources at various depths inside the granular medium, whereas EPT must rely on random chance for any grain with a dielectric boundary. Enabling our automatic acceptance rate calculation (\( P_a \)) provides further improvements, with speedups now ranging from 2.2–101 \( \times \). The optimal acceptance rate balances the computation between EPT and higher-order methods to minimize ttuv. Enabling diffusion provides significant additional speedup in scenes with low-absorption grains and long transport paths such as Snowman (Figure 13, 259 \( \times \) vs. 101 \( \times \) of EPT+VPT) at the cost of some visible bias. We attain these speedups in total render time primarily by reducing the computation time for the high-order scattering component. Measuring just the computation time spent on high-order scattering, our full method obtains dramatic speedups ranging from 241–7394 \( \times \).

In the rest of this section we will evaluate the accuracy and efficiency of our methods on our main scenes and scientific experiments.
Sandcastle. Figure 1 shows a sand castle scene containing about 2 billion grains, each modeled as a homogeneous medium enclosed in a dielectric boundary. The top half of the castle image is rendered using EPT+VPT+DA and the bottom half using EPT. For equal variance, we find that both our approximate methods render this scene about 2.2× faster than EPT with indistinguishable results. The cropped insets compare our approximate methods (middle) to the progress of EPT for the same CPU time (top) and for similar variance (bottom). The supplemental video contains a zoom sequence on this scene accompanied with visualizations of the switch depth as the camera approaches the grains.

While the high-order scattering speedup in the SANDCASTLE is over 200×, this results in only a modest overall speedup of 2.2×. This is primarily because the sand grains composing the castle have a low albedo of 0.68 (see Table 1), and due to Russian roulette few paths survive to our accelerated high-order approaches. This is compounded by the fact that our implementation currently enforces at least one grain interaction in EPT mode before switching to VPT. This allows us to reproduce the fresnel-like BRDF effects that occur at grazing angles with the aggregate, but also prevents us from completely side-stepping costly EPT for distant views. We hope to address this in future work.

Spices. In Figure 12 we render a scene with four very different granular materials: flour, pink salt, brown sugar and cane sugar. Our approximate methods accelerate the high-order scattering computation by over 120× compared to EPT, while introducing only barely perceptible bias. The visualization of the per-pixel switch depth from EPT to VPT shows that our threshold performs reasonably well: the highly coherent and high-albedo cane sugar switches later than the absorbing brown sugar or mildly absorbing pink salt. For the case of flour our method usually switches fairly quickly, except when the packet rays are trapped between the tightly packed opaque flour grains. Our visualization of individual components shows that our single, automatic switching criterion allows each rendering method to make a reasonable energy contribution to the final image.

Snowman. In Figure 13 we render a 1 m (top) and 10 cm (bottom) tall snowman. To achieve the miniaturization effect we increased the grain radius on the bottom row by a factor of 10. In the right-most images we show a closeup, rendered with our full method (EPT+VPT+DA), focused on the rim around the left eye. We render each snowman with three variations of our method. We show equal quality renders of the full light transport with EPT, EPT+VPT and EPT+VPT+DA. DA allows for a significant speedup over volumetric path tracing at the cost of bias which is most visible near creases. The black bordered insets (top right) show zoom-ins of the fallen grains. Our visualization of individual components shows that our approximate methods make a reasonable energy contribution to the final image.

Table 2: Time to unit variance in seconds for all methods. The value in parenthesis is the speedup relative to EPT alone. We report times for 100% acceptance probability ($P_a$) as well as our automatically computed optimal $P_a$. On the left we report times for rendering both low order and high order transport whereas on the right we report the timings and speedup only for the high order component.
The equal time insets (green) show unconverged results at roughly 106 and 85 hours for the large and small snowman, respectively. All speedups of our method are limited by the continued need to path bounding spheres. This inherently places a limit on how closely and intra-grain transport effects; however, it also imposes some practical limitations. We assume packings of spheres of a single radius and instantiate individual grains of smaller physical size within these bounding spheres. This inherently places a limit on how closely packed the actual grains may be. One partial workaround would be to use an “aggregate grain” which is itself a tight packing of irregularly shaped grains. While this could reduce the packing limitations for non-spherical grains such as rice, the empty space between aggregate grains would still limit the overall packing density. Leveraging work on ellipsoid packings [Donev et al. 2004] could prove fruitful for such randomly oriented oblong grains.

7 Conclusion & Discussion

In this paper we have shown how to approximate the large-scale behavior of a granular medium with a smooth radiative transport model. Normally, path tracing a high-albedo granular medium like snow is completely impractical because very long, very high-variance paths contribute much of the illumination. Our method uses smooth approximations to reduce both the variance (by enabling shadow connections) and the length (by using diffusion) of paths, greatly reducing the time spent on high-order scattering. The result is that render times are no longer dominated by long paths; instead the computation is spent rendering the visible structure of grains due to low-order scattering. As shown by our results, this makes it feasible to render many practically relevant scenes that would otherwise take unacceptably long to converge.

The speedups of our method are limited by the continued need to path trace the low-order contributions that create visible grain structure; in many cases we succeed in reducing the cost of high-order paths essentially to zero, so that in order to obtain further speedup one must turn to a new problem, that of rendering the glittery, structured low-order contributions faster without smoothing out the appearance.

Limitations in packing density. Relying on the well-studied properties of spherical packings allowed us to decouple inter-grain and intra-grain transport effects; however, it also imposes some practical limitations. We assume packings of spheres of a single radius and instantiate individual grains of smaller physical size within these bounding spheres. This inherently places a limit on how closely

Figure 13: For the tall (1 m, top) and tiny (10 cm, bottom) snowman we report the high-order / total render time in hours and the (variance). The equal time insets (green) show unconverged results at roughly 106 and 85 hours for the large and small snowman, respectively. All renderings in this figure simulate the full light transport. Any differences within the same row are due to bias and/or variance.

Generality and mixtures. While our method allows mixtures of different grain types in a single medium, we currently assume the mixing ratios are homogeneous at the scale of the aggregate. For some scenes, mid-scale heterogeneity, in which the ratios of particle types vary spatially, is desirable. Our method could be extended to this case by modulating the background medium’s extinction coefficient (Equation (1)) using a volumetric texture and computing correspondingly varying free-flight distances.

Appearance editing. We have proposed a bottom-up approach for specifying the appearance of granular materials where the grain properties and their packing rate dictate the large-scale appearance. An interesting alternative (akin to inverse bi-scale appearance design [Wu et al. 2013]) would be a top-down approach where the user specifies, or measures, the desired large-scale appearance, and the system proposes the individual grain properties and their arrangement from a dictionary of possibilities. Editing one scale independently while maintaining a fixed appearance at other scales would be a valuable appearance design tool for scenes containing granular media. Truly useful material design would require further improvements to rendering speed to obtain interactivity.

8 Acknowledgements

The authors are grateful to Maurizzio Nitti, who modeled the Sandcastle, Spices, and Snowman scenes along with individual grains, and Romain Prévost, who helped with the schematic in Appendix A.
The authors would like to thank Andrew Selle, Brent Burley and Chuck Tappan for fruitful discussions about granular media, as well as Nelson Max and Jan Novák for providing valuable feedback on early versions of the paper.

References


**A Diffusion Details**

Given a source location \( x^h_i \) generated by VPT—we generate the incident location \( x_i \) on the boundary by sampling a uniform random direction \( \omega_i \) and intersecting the aggregate mesh [Christensen et al. 2012]. We sample the direct illumination at \( x_i \) and set up our dipoles, where the positive receiver is the source location \( x^h_i \), and the negative receiver is a reflection of \( x^h_i \) about the extrapolated boundary a distance \( z_b(0) \) above the plane defined by the surface intersection \( x_i \) and its normal \( N_i \). One of our differences to Li et al. [2005] is that they define the reflection plane using \( x_i \) and the direction \( \omega_i \) (instead of the normal \( N_i \)). Another difference is that they assume \( \tau = 0 \) by construction, while we compute \( \tau \) as the projected distance of \( x^h_i \) on the plane defined by \( (x_i, N_i) \), thereby computing the correct distances for scenes adhering to the planar slab assumption. Additionally, instead of assuming a semi-infinite medium, we estimate a thickness \( d = z_r + z_a \) for a finite slab, where \( z_r \) is the scalar projection of \( (x_i - x^h_i) \) onto \( N_i \), and \( z_a \) the scalar projection of \( (x_a - x^h_i) \) onto \( N_a \), obtained with an additional ray shot in direction \( -\omega_i \) from \( x^h_i \). We define \( d_r = \sqrt{z_r^2 + \tau^2} \) and \( d_a = \sqrt{z_a^2 + \tau^2} \), with \( z_v = -z_r - z_b(0) \). We then derive the distances to the negative receiver and consecutive dipoles using the standard multi-pole expansion [Donner and Jensen 2005] but using the improved diffusion model [d’Eon and Irving 2011; Habel et al. 2013]. The definitions of the extrapolated boundaries at \( z_b(0) \) and \( z_b(d) \) can be found in [Donner and Jensen 2005]. We assume that the boundary of the granular medium is index matched (i.e. \( n = 1 \)), since our grains are suspended in vacuum. Finally, we resume VPT from \( x_i \). The auxiliary point and normal \( (x_a, N_a) \) are only used for computing the thickness of the approximate parallel slab and subsequent multi-pole mirroring.