Eurographics Symposium on Rendering (DL-only Track) (2020)
C. Dachsbacher and M. Pharr (Editors)

Supplemental Document

Cheng Zhang[1], Shuang Zhao[1]

[1]University of California, Irvine

1. Derivation of Algorithm 1

Each coefficient c_k^α in Eq. (11) in the main paper is given by a path integral of the form

$$c_k^\alpha(\beta_o, \sigma) = \int_{\Omega_k} \frac{f(\bar{\boldsymbol{x}})}{p(\bar{\boldsymbol{x}})}\, \mathrm{d}\bar{\boldsymbol{x}}, \tag{1}$$

where Ω_k denotes the space of light transport paths with exactly k volumetric scatterings. Then, it can be shown [KSZ*15] that

$$\frac{\partial c_k^\alpha}{\partial \sigma}(\beta_o, \sigma) = \int_{\Omega_k} \frac{f'(\bar{\boldsymbol{x}})}{p(\bar{\boldsymbol{x}})}\, \mathrm{d}\bar{\boldsymbol{x}}, \tag{2}$$

where f' is the partial derivative of f with respect to σ.

When performing our symbolic path tracing (Algorithm 1) to estimate the coefficients c_k^α, the path throughput T effectively keeps track of f/p when incrementally constructing the light transport path $\bar{\boldsymbol{x}}$. Thus, Eq. (2) can be estimated by $\partial T := [f'(\bar{\boldsymbol{x}})/f(\bar{\boldsymbol{x}})]\,T$.

Notice that $f(\bar{\boldsymbol{x}}) = \sigma^k \exp(-\ell\sigma)\, f_0(\bar{\boldsymbol{x}})$, where k is the number of volumetric scatterings, ℓ denotes the total length of path segments inside the grain, and f_0 captures all the other terms independent of grain optical density σ (e.g., BSDF and phase functions). Then, it can be verified that

$$\partial T = \frac{f'(\bar{\boldsymbol{x}})}{f(\bar{\boldsymbol{x}})}\, T = \frac{\frac{\partial}{\partial\sigma}\left[\sigma^k \exp(-\ell\sigma)\right] f_0(\bar{\boldsymbol{x}})}{\sigma^k \exp(-\ell\sigma)\, f_0(\bar{\boldsymbol{x}})}\, T = \left(\frac{k}{\sigma} - \ell\right) T, \tag{3}$$

giving Line 26 of Algorithm 1.

2. Extended Polynomials

Fitting τ. In §3.4, we face the problem of finding $\tau > 0$ such that $c_k \approx c_K \exp(\tau(K-k))$ for all $K < k \le K'$. This problem can be formulated as minimizing the L^2 difference in the logarithmic space:

$$\begin{aligned} E(\tau) &:= \sum_{i=1}^{K'-K} \left[\log(c_{K+i}) - \log(c_K \exp(-\tau i))\right]^2 \\ &= \left(\sum_{i=1}^{K'-K} i^2\right)\tau^2 - 2\left(\sum_{i=1}^{K'-K} i\,\Delta c_i\right)\tau + \left(\sum_{i=1}^{K'-K} \Delta c_i^2\right). \end{aligned} \tag{4}$$

To find the minimizer of Eq. (4), we simply differentiate E and set the derivative to zero, resulting in Eq. (22).

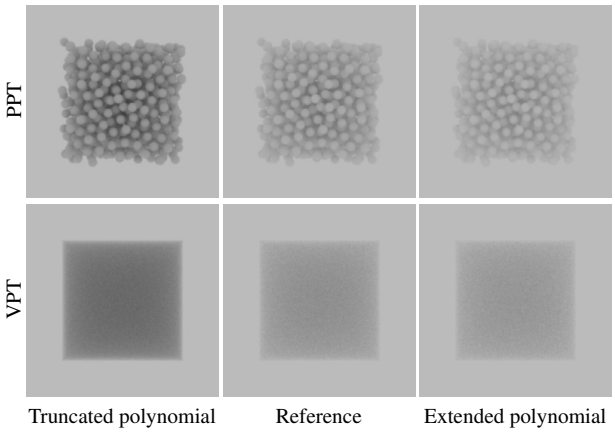

Figure 1: Rendering **highly scattering grains** requires using polynomials of very high orders to avoid severe energy loss. In this example where the grains have an albedo of 0.9995, using polynomials of order 30 (i.e., $K = 30$) yields noticeably darker results (middle vs. left). Our extended polynomial representation with $K = 29$ (that leads to the same model size) produces renderings matching the reference much more closely (middle vs. right).

Extended polynomials for derivatives. The coefficients $\{\partial c_k\}$ representing the derivatives of the GSDF and volume rendering parameters (with respect to grain density σ) can be handled in a similar way. Specifically, our experiments indicate that they can be well approximated with polynomials of degree five: for all $k > K$, $\partial c_k \approx \sum_{i=0}^5 u_i\,(k-K)^i$ for some $u_0, u_1, \ldots, u_5 \in \mathbb{R}$ (see Figure **??**-b). Then, $\sum_{k=K+1}^\infty \partial c_k\, a^k = a^K \sum_{i=0}^5 u_i \sum_{n=1}^\infty n^i a^n$, which can be calculated analytically since series of the form $\sum_{n=1}^\infty n^i a^n$ converges (as $0 \le a < 1$) and has close-form solutions.

3. Validations of extended polynomials

In main paper §3.1, we presented an extended polynomials formulation to efficiently handle highly scattering grains. Figure 1 shows a white furnace test using renderings generated with proxy path tracing (PPT) and volume path tracing (VPT) based on polynomial-valued GSDF and medium scattering parameters, respectively. When truncating the polynomials at degree $K = 30$, the results suffer from severe energy loss. In contrast, when storing polynomial

Algorithm 1 Estimating α_g^+ using symbolic and differentiated PT

function EstimateAlphaG+(β_o, σ)
  $\boldsymbol{c}^\alpha \leftarrow \{0\}^{K+1}$, $\partial\boldsymbol{c}^\alpha \leftarrow \{0\}^{K+1}$
  Randomly rotate the grain
  Initialize a ray $(\boldsymbol{r}, \boldsymbol{\omega})$ with incident angle β_o
  if the ray intersects the actual grain **then**
    $T \leftarrow 1$ ▷ Path throughput
    $k \leftarrow 0$ ▷ Number of scatterings
    $\ell \leftarrow 0$ ▷ Distance traveled inside the grain
    while ray $(\boldsymbol{r}, \boldsymbol{\omega})$ intersects the grain at $\boldsymbol{r}'$ **do**
      if line segment $(\boldsymbol{r}, \boldsymbol{r}')$ lies inside the grain **then**
        $t \leftarrow -\log(\text{rand}())/\sigma$
        $\ell \leftarrow \ell + \min(t,\ \|\boldsymbol{r}' - \boldsymbol{r}\|)$
      else
        $t \leftarrow \infty$
      end if
      if $t < \|\boldsymbol{r}' - \boldsymbol{r}\|$ **then** ▷ Volumetric scattering
        $k \leftarrow k + 1$
        $\boldsymbol{r}' \leftarrow \boldsymbol{r} + t\boldsymbol{\omega}$
        Draw a direction $\boldsymbol{\omega}'$ based on the grain's phase function
      else ▷ Interfacial scattering
        Draw a direction $\boldsymbol{\omega}'$ based on the grain's BSDF and scale T accordingly
      end if
      $\boldsymbol{r} \leftarrow \boldsymbol{r}'$, $\boldsymbol{\omega} \leftarrow \boldsymbol{\omega}'$
    end while
    $\boldsymbol{c}^\alpha[k] \leftarrow T$
    $\partial\boldsymbol{c}^\alpha[k] \leftarrow (k/\sigma - \ell)T$ ▷ Diff. throughput w.r.t. σ
  end if
  return $\boldsymbol{c}^\alpha$, $\partial\boldsymbol{c}^\alpha$
end function

coefficients up to $K = 29$ plus an additional term τ (which results in the same storage), much better accuracy can be achieved.

4. Figures of full resolution

In this section, we provide another version of Figure 9, 11 and 12, with all images shown in full resolution.

References

[KSZ*15] Khungurn P., Schroeder D., Zhao S., Bala K., Marschner S.: Matching real fabrics with micro-appearance models. *ACM Trans. Graph. 35*, 1 (2015), 1:1–1:26. 1

Algorithm 2 Computing ray-grain intersections

function IntersectBVH(ray, nd)
  if nd is a leaf node **then**
    IntersectTiles(ray, r)
  end if
  $\text{nd}_1 \leftarrow$ nd.left_child, $\text{nd}_2 \leftarrow$ nd.right_child
  if ray hits nd_2 before nd_1 **then**
    swap(nd_1, nd_2)
  end if
  IntersectBVH(ray, nd_1)
  IntersectBVH(ray, nd_2)
end function

function IntersectTiles(ray, nd)
  if node contain a single tile $\mathcal{T}_j$ **then**
    Assume $\mathcal{T}_j$ to contain grains with indices ranging from u to v from the template tile
    return IntersectTile(ray, kd.root, r_j, u, v)
  end if
  Split nd on the fly into nd_1 and nd_2
  if ray hits nd_2 before nd_1 **then**
    swap(nd_1, nd_2)
  end if
  IntersectTiles(ray, nd_1)
  IntersectTiles(ray, nd_2)
end function

Algorithm 3 Efficient importance sampling of $p_g^{\boldsymbol{x}}$

function SampleGSDFX(β_o, σ, a)
  for $i = 0$ to K **do** ▷ Compute $p_i = \text{sum}(\boldsymbol{P}_g^{\boldsymbol{x}}[:, i])$
    $p_i \leftarrow \left[\text{sum}(\boldsymbol{C}^{\boldsymbol{x}}[:, i]) + (\sigma - \sigma_0)\,\text{sum}(\partial\boldsymbol{C}^{\boldsymbol{x}}[:, i])\right] a^i$
  end for
  Draw $k \in \{0, \ldots, K\}$ with probability $p_k / \sum_t p_t$
  for $i = 0$ to $m - 1$ **do** ▷ Compute $q_i = \boldsymbol{Q}_g^{\boldsymbol{x}}[i, k]$
    $q_i \leftarrow \boldsymbol{C}^{\boldsymbol{x}}[i, k] + (\sigma - \sigma_0)\,\partial\boldsymbol{C}^{\boldsymbol{x}}[i, k]$
  end for
  Draw $j \in \{0, \ldots, m-1\}$ with probability $q_j / \sum_t q_t$
  Draw $\boldsymbol{x}_i$ uniformly from $\mathbb{S}_j$
  return $\boldsymbol{x}_i$
end function

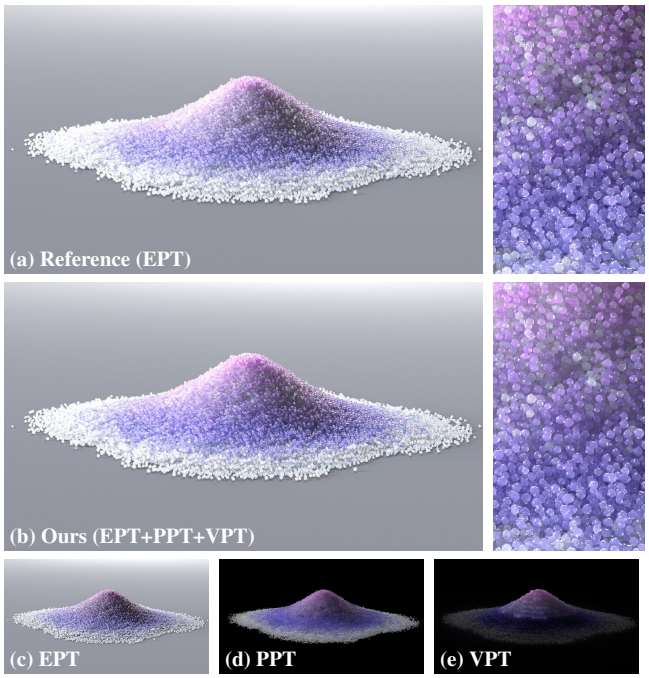

Figure 2: Figure 9 in the main paper

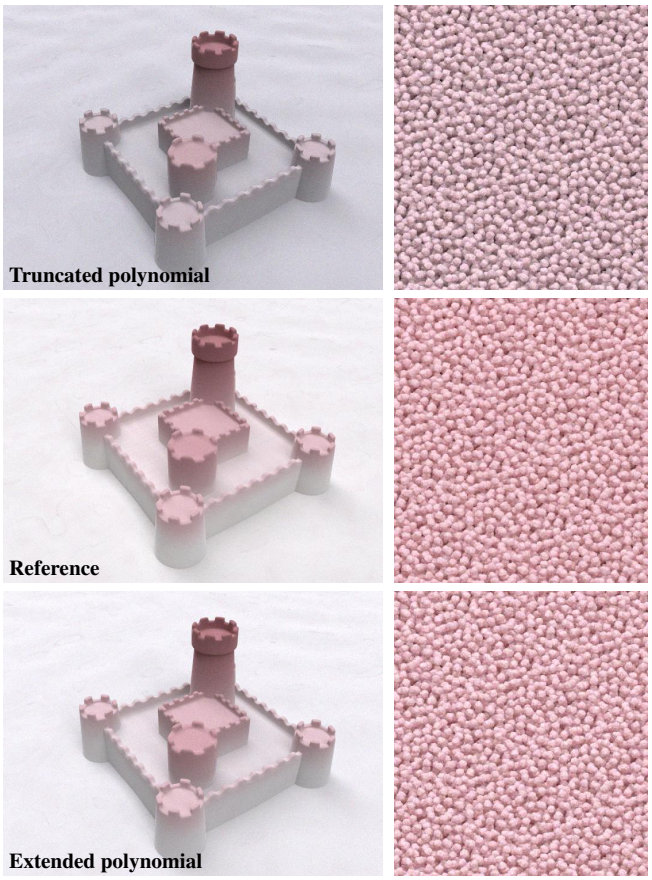

Figure 3: Figure 11 in the main paper

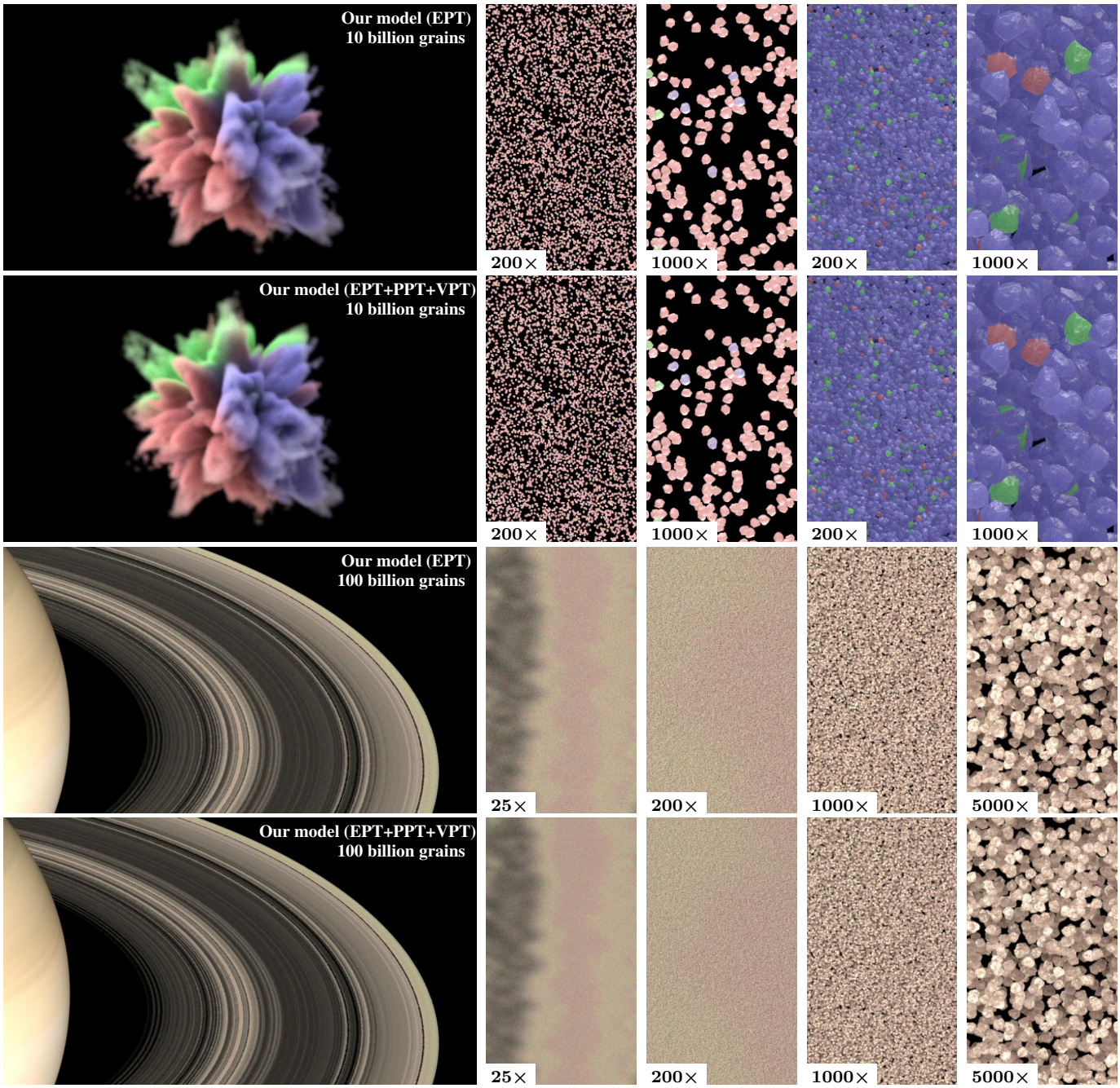

Figure 4: Figure 12 in the main paper