Higher Order Ray Marching

Adolfo Muñoz

Universidad de Zaragoza
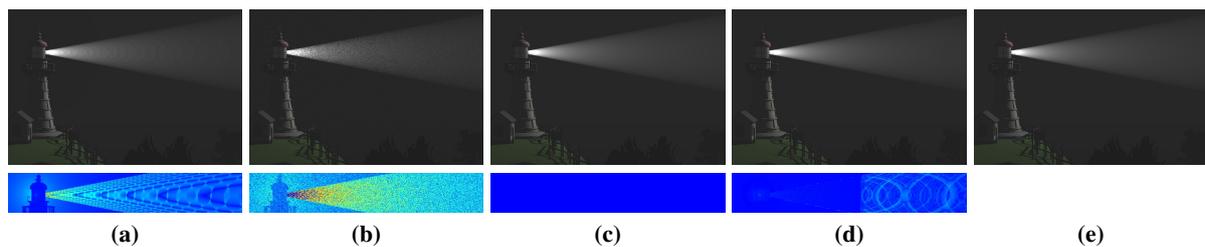

Figure 1: *Render of participating media with equiangular sampling and several numerical solvers.* **Top row:** *Rendered image.* **Bottom row:** *Absolute error vs. ground truth in false color (blue is lowest, red is highest).* **(a)** *Standard ray marching, rectangle quadrature rule (17s). The image shows banding at the cone of light.* **(b)** *Monte Carlo integration (19s). The image shows high frequency noise.* **(c)** *Bogacki-Sampine method: an order three embedded (adaptive) initial value problem solver (1m 15s).* **(d)** *Nested Simpson quadrature rule: an order two nested (adaptive) quadrature rule (17s).* **(e)** *Monte Carlo ground truth (5h).*

Abstract
Rendering participating media is still a challenging and time consuming task. In such media light interacts at every differential point of its path. Several rendering algorithms are based on ray marching: dividing the path of light into segments and calculating interactions at each of them. In this work, we revisit and analyze ray marching both as a quadrature integrator and as an initial value problem solver, and apply higher order adaptive solvers that ensure several interesting properties, such as faster convergence, adaptiveness to the mathematical definition of light transport and robustness to singularities. We compare several numerical methods, including standard ray marching and Monte Carlo integration, and illustrate the benefits of different solvers for a variety of scenes. Any participating media rendering algorithm that is based on ray marching may benefit from the application of our approach by reducing the number of needed samples (and therefore, rendering time) and increasing accuracy.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: —Color, shading, shadowing, and texture I.3.7 [Computer Graphics]: —Ray tracing

1. Introduction

A ray of light that traverses a participating medium is altered in several ways at differential level: its radiance may be scattered away or absorbed, and its power may increase due to medium emission and in-scattering. All these interactions happen at every differential point of the path of light. As a consequence, light simulation becomes a daunting task. How to simulate a phenomenon that changes at every differential point of the volume within a reasonable time frame? This question inspires a very active field of research, in which each of the different participating media rendering

techniques finds its own compromise between accuracy and simulation time.

Rendering physically-accurate participating media involves solving (or approximating) the Radiative Transfer Equation (RTE) [Cha60]. There are several interesting approaches, one of the most widely used techniques being ray marching [PH89,Jen01]. Its main idea is to divide the path of light into uniform segments that approximate all the differential interactions by a single sample. Depending on the size of each of those segments, the render becomes more accurate (short segments) or the simulation takes shorter time (large

segments). The optimal compromise between time and accuracy happens at a specific step size, which could potentially vary between different regions of the same scene.

Another option is to integrate the RTE using a Monte Carlo quadrature method [PM93]. In this case, the trade-off between accuracy and render time depends on the number of Monte Carlo samples: given N samples (render time being proportional to the number of samples), variance becomes proportional to $1/\sqrt{N}$. In contrast with ray marching, Monte Carlo is *unbiased*. Lately, the RTE has been formulated as an initial value problem and solved using several numerical methods [Muñ12]. Higher order Runge-Kutta methods, while still showing the same step-size trade-off between accuracy and render time, present faster convergence rates than lower order ones.

Our work analyzes how higher order methods, both quadrature rules and initial value problem (IVP) solvers, are capable of simulating participating media, and how they perform compared to ray marching or Monte Carlo integration. We show how each of these methods leads to a new ray marching technique, and study their applicability for a number of scenarios. Furthermore, we improve the performance of such methods by showing how to include importance sampling strategies into the simulation.

Several of the analyzed methods are adaptive, where the number of samples increases in regions that require it for an accurate solution, while the number of samples decreases when they are not required for the desired threshold accuracy, speeding up simulation time. The adaptation of such solvers is not related to per-medium heuristics [Jen01] nor to light distribution [JZJ08]. Instead, they adapt to the mathematical definition of light contribution along the path (including both medium and light distribution). This adaptability is of great help in scene setups with visible shadows or highly heterogeneous media, in which standard ray marching leads to banding artifacts and Monte Carlo yields high frequency noise (see Figure 1).

Standard ray marching and Monte Carlo integration are particular cases of our approach. Any algorithm that includes any of those techniques may include other methods as well. It is the first time that such numerical methods have been used for rendering participating media.

2. Previous work

Participating media rendering. Many previous papers have dealt with participating media rendering [GJJD09], either based on ray tracing [KvH84], Monte Carlo simulations [PM93] or other methods [RT87]. In practice, all of them approximate the Radiative Transfer Equation [Cha60] or find ways to solve it. Recent research deals with reformulating the RTE for specific scenarios to reduce its complexity (single point light and single scattering [PSS11]) or devise new sampling strategies [KF12]. Other participating media

rendering algorithms are based on additional representations of light interactions such as point particles (photons) [Jen01] or segments, such as virtual ray lights [NNDJ12b] or photon beams [JZJ08, JNSJ11, NNDJ12a].

Ray marching. Ray marching is one of the key techniques for rendering participating media [PH89]. Since then, it has been adapted to algorithms such as volumetric photon mapping [Jen01]. Some authors have proposed adaptive ray marching techniques, that adjust their step size to better fit the scene's properties. They can be based on medium heuristics [Jen01] or to the distribution of light samples in the volume [JZJ08]. In contrast, our algorithm provides a generic framework that, using an adaptive numerical method, adapts to both media properties and light distribution without the need of specific per-scene heuristics.

Importance sampling. Monte Carlo based stochastic methods are also widely used for rendering participating media. As our work, the key idea of these methods is to find a sampling strategy that reconstructs light transport in participating media from the minimal number of samples. Some strategies consist on finding and using a probability distribution function that, given a specific heuristic, such as the optical thickness of the media [Col68, RSK08] (even adapting the strategy to heterogeneous media with varying optical thickness [WMHL65]), or the incident radiance from a point light source [KF12], distribute samples and weights according to it. Our work shows how to apply such importance sampling strategies on top of any numerical method. More advanced stochastic methods involve incorporating several importance sampling strategies at once, either based on a multiple importance sampling strategy [KF12], or on a joint probability distribution along multiple light bounces [GKH*13].

Numerical solvers. There has been a lot of research regarding both initial value problem solvers [Gea71, CL85, PTVF07] and quadrature rules [SS66, PTVF07]. In computer graphics, initial value problem solvers have not been widely used for rendering, with few exceptions [GSMA06, Muñ12]. However, some quadrature rules are very often applied, such as Monte Carlo quadrature for path tracing algorithms [Vea97] or rectangle quadrature rules for ray marching [PH89]. The application of other higher order quadrature rules has been rare, with again few exceptions [JLSJ11].

3. Radiative Transfer

When light traverses a participating medium, it interacts with it at every differential point of the path, in three possible ways: it may get absorbed, scattered or emitted by the medium. The equation that defines this behavior is the Radiative Transfer Equation [Cha60]:

$$\frac{\partial L(\mathbf{x}, \omega)}{\partial t} = \sigma_a(\mathbf{x}) L_e(\mathbf{x}, \omega) - \sigma_t(\mathbf{x}) L(\mathbf{x}, \omega)$$
$$+ \sigma_s(\mathbf{x}) \int_\Omega p(\mathbf{x}, \omega', \omega) L(\mathbf{x}, \omega') \, d\omega' \qquad (1)$$
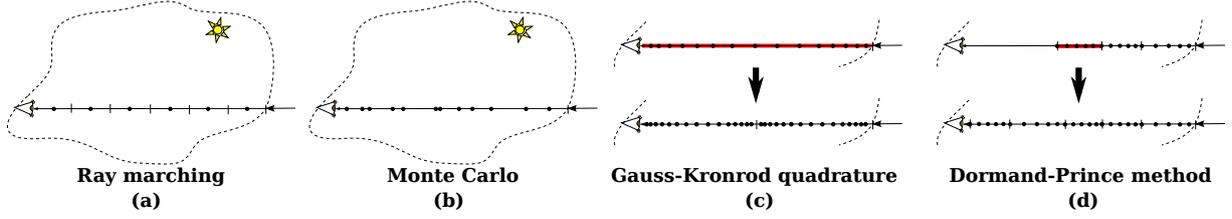
Figure 2: *Ray marching techniques (steps separated by vertical lines, samples marked as dots). (a) Standard ray marching: path is split into uniform segments, each of them evaluated using one sample. (b) Monte Carlo: samples are randomly chosen. (c) Gauss-Kronrod: the whole path is integrated with fifteen samples, but if the error is higher than a tolerance (marked in red), the path is split in two and the same rule is recursively applied. (d) Dormand-Prince: five samples per step. A step is rejected when its error is higher than a tolerance (marked in red). Step size is modified according to the error even if accepted.*

where $\frac{\partial L(\omega_o)}{\partial t}$ represents the differential variation of radiance along the path of light t, $\mathbf{x}$ is the differential point at which the interaction occurs, ω represents the direction followed by light and ω' represents the direction of other light paths that reach the differential point $\mathbf{x}$ (Ω is the domain of integration, a sphere). The rest of the symbols represent the properties of each medium:

- σ_a is the absorption coefficient, energy that is absorbed by the medium at every differential step.
- σ_s is the scattering coefficient, energy scattered by particles in the medium at every differential step.
- $\sigma_t = \sigma_a + \sigma_s$ is the extinction coefficient, energy that is either absorbed or out-scattered.
- $p(\mathbf{x}, \omega', \omega)$ is the phase function, that defines the angular distribution of light scattering.
- L_e is the medium's emission.

In order to render participating media, we need to solve Equation 1 and obtain the radiance $L(\mathbf{x}, \omega)$. Equation 1 is a linear ordinary differential equation, and therefore has an analytical integral solution:

$$
\begin{aligned}
L(\mathbf{x_0}, \omega) = {} & T_r(\mathbf{x_0}, \mathbf{x_t}) L(\mathbf{x_t}, \omega) \\
& + \int_0^t T_r(\mathbf{x_0}, \mathbf{x_s}) \sigma_a(\mathbf{x_s}) L_e(\mathbf{x_s}, \omega) ds \\
& + \int_0^t T_r(\mathbf{x_0}, \mathbf{x_s}) \sigma_s(\mathbf{x_s}) L_i(\mathbf{x_s}, \omega) ds
\end{aligned}
\tag{2}
$$

where $\mathbf{x_0}$ is the origin point of the ray, $\mathbf{x_s} = \mathbf{x_0} + t\mathbf{x_s}$ is a differential point of interaction at the position parametrized by s and t is the distance which the light comes from. $T_r(\mathbf{x_0}, \mathbf{x_t})$ is named *transmittance* and accounts for all the light that has traversed the medium between $\mathbf{x_0}$ and $\mathbf{x_t}$ without getting extinguished due to the medium's properties. $L_i(\mathbf{x_s}, \omega)$ represents the in-scattered radiance (energy coming from different light paths). They are defined as follows:

$$
T_r(\mathbf{x_0}, \mathbf{x_t}) = e^{\int_0^t -\sigma_t(\mathbf{x_s})ds}
\tag{3}
$$

$$
L_i(\mathbf{x_s}, \omega) = \int_\Omega p(\mathbf{x_s}, \omega', \omega) L(\mathbf{x_s}, \omega') d\omega'
\tag{4}
$$

Both in differential (Equation 1) and integral form (Equation 2), L_i involves both single and multiple scattering. Mul-

tiple scattering (light that has bounced several times in the medium) can be sampled for instance using Monte Carlo integration or photon mapping [Jen01]. Single scattering can be computed by tracing shadow rays from the sample point to the light sources. Light that comes directly from a light source travels a distance across the medium, and therefore must be also extinguished according to T_r. The computation of L_i must be done per-sample and is usually linked to the efficiency of the simulation. In case of homogeneous media, T_r is an analytical term:

$$
T_r(\mathbf{x_0}, \mathbf{x_s}) = e^{-\sigma_t ||\mathbf{x_s} - \mathbf{x_0}||}
\tag{5}
$$

Rendering participating media involves solving the RTE, either its integral form (Equation 2, the most common option, analyzed in Section 4.1) or its differential form (Equation 1, analyzed in Section 4.3). Each method translates into a new ray marching strategy, of sampling light along the path, as it is shown in Figure 2: each technique selects a different set of sample points and weights in order to converge to the solution. Adaptive techniques may reject some samples (marked in red) based on an error estimation heuristic and an adaptation strategy, and select new sample points accordingly.

4. Solving the Radiative Transfer Equation

4.1. Quadrature rules

Equation 2 can be numerically calculated by a quadrature rule, which approximates the integral by a sum of weighted function evaluations. For instance, the third term of Equation 2 would be approximated as follows:

$$
\begin{aligned}
& \int_0^t T_r(\mathbf{x_0}, \mathbf{x_s}) \sigma_s(\mathbf{x_s}) L_i(\mathbf{x_s}, \omega) ds \approx \\
& \Sigma_{k=1}^n w_k T_r(\mathbf{x_0}, \mathbf{x_k}) \sigma_s(\mathbf{x_k}) L_i(\mathbf{x_k}, \omega)
\end{aligned}
\tag{6}
$$

from a set of n sample points $\mathbf{x_k}$, each of them weighed by a specific factor w_k. The sample points $\{\mathbf{x_k}\}$ and weights $\{w_k\}$ are specific values of each quadrature rule. Each function evaluation at each $\mathbf{x_k}$ requires calculating T_r and L_i, which are again integrals. In case of homogeneous media, T_r (see Equation 5) is analytical. However, for inhomogeneous

media, the transmittance T_r can be computed by applying the same quadrature rule that is applied to the integral RTE (Equation 2) to the exponent in Equation 3. In the case of single scattering, L_i becomes an addition (for all light sources) but T_r must be also applied to light coming directly from the light sources to the sample point.

Standard ray marching is the application of a rectangle quadrature rule to Equation 2: the sample points are uniformly distributed along the path of light, and the weight for each of them is homogeneous Δs, which is the step size. By increasing the number of segments, image quality gets improved (step size Δs is reduced) but simulation time also increases. Figure 2(a) illustrates this. Monte Carlo method is also a quadrature rule, with sample points $\mathbf{x_k}$ randomly distributed according to a *probability distribution function* (*pdf*) and $w_k = \frac{1}{pdf(x_k)}$, as shown in Figure 2(b). In this work we also test Simpson rule as representative of a higher order quadrature rule.

4.2. Nested quadrature

Nested quadrature rules apply a higher order and a lower order rule simultaneously, in order to estimate both a solution and an estimation of the error:

$$\hat{F^H} = \sum_{i=1}^{n} w_i^H f\left(\mathbf{x_i^H}\right) \qquad \hat{F^L} = \sum_{i=1}^{m} w_i^L f\left(\mathbf{x_i^L}\right) \qquad (7)$$

where $\hat{F^H}$ and $\hat{F^L}$ represent the estimations from the higher order and the lower order quadrature rules, respectively ($n > m$), and $e = ||\hat{F^H} - \hat{F^L}||$ is the estimated error (although other method-specific error metrics can be found in the literature). The corresponding weights and $\{w_i^H\}$ and $\{w_i^L\}$, as well as the sample points $\{\mathbf{x_i^H}\}$ and $\{\mathbf{x_i^L}\}$ are specific to each method. Furthermore, the lower order method samples are a subset of the higher order methods samples ($\{\mathbf{x_i^L}\} \subset \{\mathbf{x_i^H}\}$), so just with n sample points we get both the estimation of the integral F^L and the error e. When the error is above a certain tolerance *tol* ($e > tol$), the estimation F^L is rejected and the integration range is split in two. Then, the quadrature rule is applied recursively to each sub-segment (see Figure 2(c) for an overview). In practice, it is convenient to set up maximum and minimum step sizes, so the adaptive method is able to sample high frequency details and is not stalled indefinitely.

The tolerance is a fundamental parameter for error control. In this work we test the **Nested Simpson rule** (standard Simpson and trapezoid quadrature rules) and the **Gauss-Kronrod method** [Pat68]. While there are others in the literature, these two are a representative subset as they have different orders.

4.3. Initial value problem solvers

As previous work [Muñ12], our work takes advantage of the differential form of the Radiative Transfer Equation (Equa-

tion 1). Equation 1 has the form $y'(t) = f(y,t)$ (considering that t is the distance along the path of light and y the radiance itself). Furthermore, the radiance that enters the medium $y_0 = L(\mathbf{x_0}, \omega)$ is also known (it comes from the surface). Therefore, the RTE is in fact an initial value problem (IVP) and, as such, there are several methods that can numerically solve it, such as all Runge-Kutta methods [Gea71, CL85, PTVF07]. General Runge-Kutta methods are defined as follows:

$$y_{i+1} = y_i + \sum_{j=1}^{n} b_j k_i^j \qquad t_{i+1} = t_i + h \qquad (8)$$

$$k_i^j = hf\left(t_i + c_j s_i, y_i + \sum_{l=1}^{j-1} a_{jl}\right) \qquad (9)$$

where h is the step size and n is the number of function evaluations per step, which is usually related to the *order* of the method. The values $\{b_j\}$, $\{c_j\}$ and $\{a_{jl}\}$ are specific to each method, and are usually arranged in a *Butcher's tableau* [Gea71]. The Butcher's tableau of different methods can easily be found on specialized literature [PTVF07]. Euler's method is the most simple Runge-Kutta method, with $n = 1$, $c_1 = 0$ and $b_1 = 1$.

We also need to account for the transmittance T_r for the the radiance that comes directly from the light sources to each sample point. In the case of homogeneous media this is an analytical term (Equation 5), and for heterogeneous media we solve the following IVP using the same solver than for the RTE:

$$\frac{\partial L(\mathbf{x}, \omega)}{\partial t} = -\sigma_t(\mathbf{x}) L(\mathbf{x}, \omega) \qquad (10)$$

which is just the extinction term of the RTE (Equation 1). Note that Equation 3 would be the integral form (analytical solution) of Equation 10 that defines the transmittance T_r. In this work, we analyze **Euler's method** for side-by-side comparison with the rectangle rule (standard ray marching), and the standard **fourth order Runge-Kutta method** (RK4). As the RTE is a linear ordinary differential equation (Equation 1), the RK4 method requires only two samples per step instead of the usual four [PTVF07]. RK4 is included in our tests because it is a standard high order not embedded numerical method.

4.4. Embedded Runge-Kutta methods

An embedded Runge-Kutta method consists of two different order methods which share the same $\{c_j\}$ and $\{a_{jl}\}$ values with different $\{b_j\}$ weights:

$$y_{i+1} = y_i + \sum_{j=1}^{n} b_j k_i^j \qquad y_{i+1}^* = y_i + \sum_{j=1}^{n} b_j^* k_i^j \qquad (11)$$

where y_i and y_i^* are the two estimations from the two methods, and $\{b_j\}$ and $\{b_j^*\}$ are the corresponding weight coefficients. The calculation of the k_i^j values (which involves

function evaluations, as seen in Equation 9) is common for both methods, so therefore we get both the per-step estimation and error with the same number of function evaluations (similar to nested quadrature rules, see Section 4.2). y_i^* only serves for the purpose of error estimation $e_i = ||y_i - y_i^*||$.

If the per-step estimated relative error e_i is greater than a specified tolerance *tol*, the step is rejected and recalculated with a new step size. Even if the step is accepted, the step size is modified. We use the following default step adaptation strategy for all embedded methods, although other configurations may be explored as future work [PTVF07]:

$$h_{i+1} = 0.99 \left(\frac{tol}{e_i + 10^{-2}} \right) h_i \qquad (12)$$

As with nested quadrature rules, we also set global upper and lower bounds for the step size. In this work, we consider the order three **Bogacki-Shampine method** [BS89] and the order five **Dormand-Prince method** [DP80]. As the RTE is linear a Dormand-Prince step requires five function evaluations instead of the usual six (as it happens with the RK4 method). The analysis of both methods not only gives insight about embedded methods but also enables comparison between two methods of different order.

5. Importance sampling

Importance sampling is a general technique that enables Monte Carlo estimators to choose the integration samples $\{\mathbf{x_k}\}$ according to a *probability distribution function* (*pdf*), so that statistically more samples are taken in the most relevant regions of the integrand. Different *pdf* lead to different sampling strategies along the domain of integration, preferably minimizing the number of samples required to obtain an accurate result. Identifying the right *pdf* for the right integrand involves factoring as much knowledge of the integrand as possible in the *pdf* formula, and is a topic of interest for the rendering community on its own. In this section we show how, by means of a change of variable, we adapt several importance sampling strategies to any quadrature rule or initial value problem solver.

Distance sampling [RSK08]: The key idea of this strategy is to locate fewer samples in regions where extinction makes them unnoticeable. In order to include such strategy into our approach (based either on initial value problem solvers or quadrature rules) we apply the following change of variable:

$$u = e^{-s\sigma_t^{ref}} \qquad (13)$$

where σ_t^{ref} is a per-ray *reference extinction coefficient*, which we calculate as the average (for all wavelengths) of the extinction coefficient at the origin of the ray that traverses the medium. By applying Equation 13 to both the differential and the scattering term of the integral form of the RTE

(Equations 1 and 2) we get:

$$\frac{\partial L(\mathbf{x},\omega)}{\partial u} = -\frac{\sigma_a(\mathbf{x'_u})L_e(\mathbf{x'_u},\omega) - \sigma_t(\mathbf{x'_u})L(\mathbf{x'_u},\omega) + \sigma_s(\mathbf{x})L_i(\mathbf{x},\omega)}{u\sigma_t^{ref}} \qquad (14)$$

$$\int_1^{e^{-t\sigma_t^{ref}}} -\frac{T_r(\mathbf{x_0},\mathbf{x'_u})\sigma_s(\mathbf{x'_u})L_i(\mathbf{x'_u},\omega)}{u\sigma_t^{ref}} du \qquad (15)$$

where $\mathbf{x'_u} = \mathbf{x_0} - \frac{\log u}{\sigma_t^{ref}}\omega$ is the point in the path of light according to the change of variable. Equation 14 can be solved as an IVP (see Section 4.3) and Equation 15 by means of a quadrature rule (see Section 4.1).
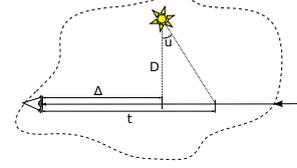

Figure 3: *Geometrical variables involved in equiangular sampling*

Equiangular sampling [KF12]: Given a participating medium illuminated by a point light, this strategy locates more samples close to the light source. Again, a change of variable enables us to take advantage of this sampling strategy. Given a light source at distance *D* to the ray:

$$u = \tan^{-1}\left(\frac{s - \Delta}{D} \right) \qquad (16)$$

where Δ is the distance from the origin of the ray to the projection of the position of the point light source to the ray (see Figure 5). This leads to the following transformations of both the differential form and the scattering term of the integral form of the RTE:

$$\frac{\partial L(\mathbf{x},\omega)}{\partial u} = D(\tan^2(u)+1)(\sigma_a(\mathbf{x})L_e(\mathbf{x},\omega) - \sigma_t(\mathbf{x})L(\mathbf{x},\omega)) + \sigma_s(\mathbf{x})L_i(\mathbf{x},\omega)) \quad (17)$$

$$\int_{\tan\left(\frac{-\Delta}{D}\right)}^{\tan\left(\frac{t-\Delta}{D}\right)} D(\tan^2(u)+1)T_r(\mathbf{x_0},\mathbf{x'_u})\sigma_s(\mathbf{x'_u})L_i(\mathbf{x'_u},\omega)du \quad (18)$$

where $\mathbf{x'_u} = \mathbf{x_0} - (\tan(u)*D+\Delta)\omega$ is again the point in the path of light according to the change of variable.

For fixed step size methods (either quadrature rules or IVP solvers) and Monte Carlo integration, importance sampling locates more samples in targeted regions. For adaptive methods this change of variable serves as a prior for initial sample placement. Notice that Monte Carlo importance sampling is actually mathematically equivalent to this approach, in which the probability distribution function is the derivative of the corresponding change of variable (Equations 13 and 16, respectively). Other importance sampling strategies that deal with the traversal of participating media may be also expressed as a change of variable and therefore could be adopted by higher order ray marching techniques.
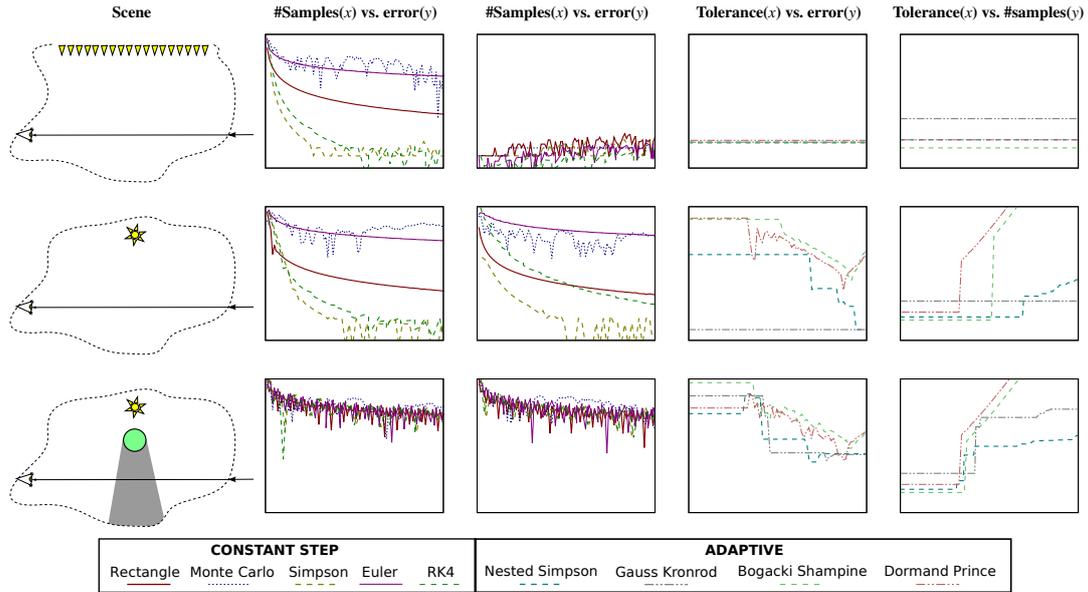
Figure 4: *Error analysis under three different scenarios, each one of them on each row, represented by the image on the left. The analysis is done for the single represented ray.* **Scenarios from top to bottom:** *a single directional light, a single point light and a single point light with an occluder.* **From left to right:** *scene and ray representation; number of samples vs. error for constant step methods (without importance sampling); number of samples vs. error for constant step methods, with specific importance sampling (distance for the directional light scenario, equiangular for both point light scenarios); tolerance vs. error for adaptive methods with specific importance sampling; tolerance vs. number of samples for adaptive methods with specific importance sampling.*

6. Results and discussion

Each of the solvers that have been introduced in previous sections shows a different behavior when applied to render participating media. In order to better understand each method's trade-off between number of samples and accuracy of the resulting radiance, we set up three simple scenarios. For each of them, we compare the radiance given by a ray that traverses the corresponding participating medium with a ground truth. We calculate the ground truth using a Monte Carlo quadrature rule with 100000 samples (for that single ray), and compare it to the result of the specific numerical method. For a L ground truth solution and a $\hat{L}$ estimated radiance, our error metric is relative: $\frac{||L-\hat{L}||}{max(||L||,||\hat{L}||)}$.

These three scenarios are: a single directional light, a single point light and a single point light with an occluder. For this test (and most results in this section) multiple scattering is neglected and just single scattering is computed. The errors at multiple scattering simulation would be impossible to disambiguate from each solver's error, complicating the analysis. Furthermore, multiple scattering tends to *blur* the apparent radiance, therefore smoothening discontinuities that are pretty relevant for the performance analysis of ray marching techniques (discontinuities are actually a worst-case scenario, as discussed below in the text). Figure 4 shows the result of these tests. Instead of analyzing rendering time

for a single ray, we count the number of function evaluations (samples) required by each method. This metric is more reliable, as the bottleneck in participating media rendering comes from the visibility tests and the indirect light illumination, and the required time per sample is scene dependent.

Importance sampling. Applying distance importance sampling for the first scenario (top row, a single directional light) is the optimal sampling strategy (the *pdf* equals the integrand): even with a single sample any numerical method gives an accurate solution. Therefore, the error yielded by constant step methods (second column) is owed just to numerical precision. Furthermore, adaptive methods converge in a single step, no matter the tolerance (fourth column). In general scenarios with directional light sources, however, this optimal behavior would be altered either by occluders and heterogeneities on media coefficients.

Equiangular sampling helps the convergence of some quadrature rules such as Monte Carlo and rectangle rule for the second scenario. However, it seems that it hinders the performance of IVP solvers. This is due to the fact that IVP solvers are generally weak to *stiff equations*: differential equations that for some reason present a numerically unstable behavior. It seems that the change of variable for equiangular sampling (Equation 16) may be leading to a stiff equa-
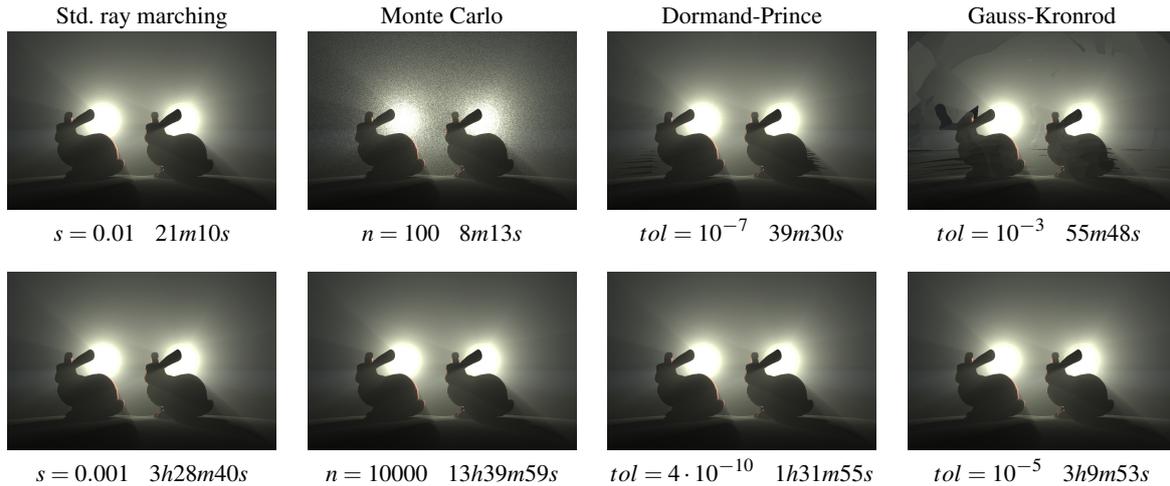
Std. ray marching	Monte Carlo	Dormand-Prince	Gauss-Kronrod
$s = 0.01$ $21m10s$	$n = 100$ $8m13s$	$tol = 10^{-7}$ $39m30s$	$tol = 10^{-3}$ $55m48s$
$s = 0.001$ $3h28m40s$	$n = 10000$ $13h39m59s$	$tol = 4 \cdot 10^{-10}$ $1h31m55s$	$tol = 10^{-5}$ $3h9m53s$

Figure 5: *Test of the performance of different solvers in a medium with very noticeable lighting discontinuities (shadows). From left to right: traditional ray marching (*s *is the step size), Monte Carlo (*n *is the number of samples), Dormand-Prince solver and Gauss-Kronrod solver (*tol *represents tolerance). Top row: sub-sampled render. Notice that, except Monte Carlo (that is unbiased), all solvers lead to perceivable structured errors, with different patterns (more regular in the case of standard ray marching). Bottom row: Accurate render.*

tion under certain circumstances. This can also be perceived on the performance of adaptive IVP solvers.

This stiffness is further illustrated in Figure 6 for the case of distance importance sampling. Monte Carlo greatly reduces high frequency noise, and it also seems that the rectangle rule (standard ray marching) yields less artifacts than with uniform sampling. However, Euler's method is importantly penalized: the underwater scene is quite large compared to the extinction coefficient (the medium is optically thick), fact that increases equation stiffness for distance sampling and compromises the performance of IVP solvers. Figure 7 also illustrates the effect of equation stiffness.

Shadows. For the top two scenarios, as expected, higher order methods converge faster to the accurate solution than lower order ones (requiring less samples). However, shadows (third scenario) alter this behavior. An IVP solver (or quadrature rule) of order n assumes that the differential equation (or the integrand) is C^n (continuous derivative up to order n). However, most scene setups imply that neither the RTE nor its integral formulation are actually C^n: media coefficients are not necessarily continuous up to any order (not even order 0), and lighting from single scattering may be discontinuous due to shadows. It can be proved that an order n solver applied to an order m equation shows an order m performance if $m < n$ [EJNT88]. The bottom row of Figure 4 analyzes a scenario with sharp shadows (order 0), and shows how constant step methods show the same performance than Monte Carlo (order 0). Adaptive methods, however, can overcome such discontinuities by casting more samples. Shadows are a major source of variance.

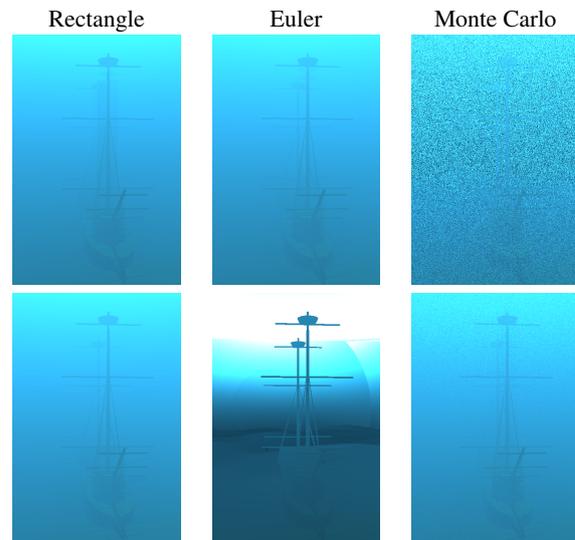Figure 5 shows how discontinuities affect participating

Rectangle	Euler	Monte Carlo

Figure 6: *Underwater scene. Rectangle, Euler and Monte Carlo solvers, using 40 samples per pixel (step = 0.1). Each image has taken 32 seconds to render.* **Top row:** *uniform sampling.* **Bottom row:** *distance sampling.*

media rendering with different solvers: the two reflectors in the back of the scene project a cone of light with a clear discontinuity that affects the whole image, additionally to the shadows casted by the bunnies. Standard ray marching requires the step to be greatly reduced for an accurate solution. Shadows hinder the performance of the Gauss-Kronrod rule and Dormand-Prince method: as they are both adap-

tive, they attempt to locate the samples where they are most needed (close to the shadows). Monte Carlo quadrature requires more samples in order to reduce the variance.
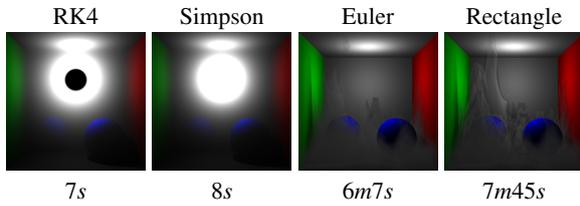

RK4	Simpson	Euler	Rectangle
$7s$	$8s$	$6m7s$	$7m45s$

Figure 7: *Comparison of the performance of IVP solvers and quadrature rules for both homogeneous and a heterogeneous media. Equiangular sampling with 10 samples per path has been applied on all four cases. Notice that, in this case, quadrature rules are more suited for homogeneous media and IVP solvers for heterogeneous media.*
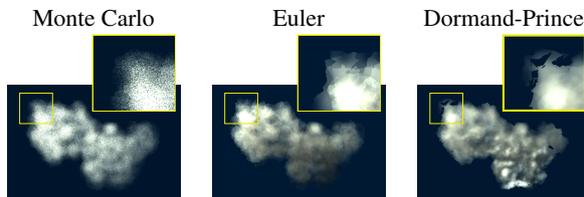

Monte Carlo	Euler	Dormand-Prince

Figure 8: *Different types of noise are generated by the application of different kinds of methods using low number of samples.* **Left:** *Monte Carlo integration leads to unstructured high frequency noise.* **Middle:** *Constant step numerical methods lead to structured banding noise.* **Right:** *Adaptive methods lead to structured incoherent noise.*

Homogeneous vs. heterogeneous media. The performance of the different numerical methods greatly varies depending on whether the medium is homogeneous or heterogeneous. The perceivable optical features of homogeneous media are related to shadows (discussed above) and proximity to light sources. Sharp shadows are often more efficiently simulated using adaptive quadrature rules (see Figure 1) as the adaptation strategy involves splitting the domain of integration. Proximity of point light sources leads to numerical singularities (that can be alleviated by equiangular importance sampling [KF12]) and in general to stiff equations, that are a worst case scenario for IVP solvers. Figure 7 (top row) shows that a RK4 method fails to simulate light transport in a simple homogeneous medium due to such stiffness (although the artifact would be solved by using more samples), while Simpson result does not yield any rendering artifact. On the other hand, heterogeneous media pose two challenges for quadrature rules. First, the transmittance T_r needs to be approximated per-sample (for instance using the same quadrature rule), while the differential form of the RTE does not present this requirement. Therefore, the efficiency of quadrature rules is hindered (see Figure 7, bottom row and
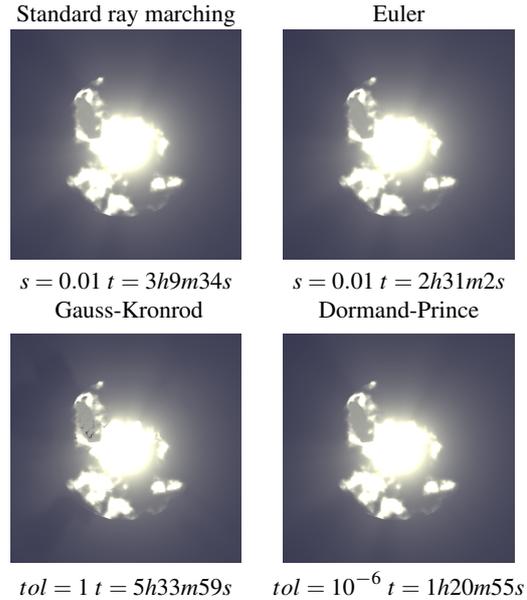

Standard ray marching	Euler
$s = 0.01\ t = 3h9m34s$	$s = 0.01\ t = 2h31m2s$
Gauss-Kronrod	Dormand-Prince
$tol = 1\ t = 5h33m59s$	$tol = 10^{-6}\ t = 1h20m55s$

Figure 9: *Test of the performance of different solvers across a high frequency procedural heterogeneous medium.* **Left column:** *Quadrature rules.* **Right column:** *IVP solvers.* **Top row:** *Constant step order 0 methods (rectangle quadrature and Euler's method).* **Bottom row:** *Higher-order adaptive methods (Dormand-Prince and Gauss-Kronrod).*

Figure 9, top row), although this issue could be solved by decoupling the computations related to transmittance from the scattering simulation [KF12]. Also, the accuracy of the result depends on the accuracy of the T_r approximation, which may lead to perceivable errors compared to equivalent IVP solvers (or even introduce discontinuities), as illustrated in the bottom row of Figure 7. In the case of heterogeneous media, sub-sampling leads to different kinds of noise, depending on the numerical method (see Figure 8).

Regarding adaptive rendering of heterogeneous media, Figure 9 (bottom row) compares the results from two high order adaptive methods: Dormand-Prince and Gauss-Kronrod. Dormand-Prince greatly improves efficiency (without accuracy loss) compared to standard ray marching and Euler. However, Gauss-Kronrod shows a much worse performance: it does not converge to the accurate solution even after almost double the time than standard ray marching. Apart from the transmittance approximation of quadrature rules discussed above, the splitting strategy for nested quadrature rules is suboptimal for high-frequency heterogeneous media. At every rejected integration, all samples are discarded (15 samples per rejected step). High frequency media properties lead to many rejected integration attempts and therefore a great performance loss. Dormand-Prince, on the other hand, at each step, adapts the step size to the RTE no matter if the step is finally accepted or rejected.

This flexibility leads to a more accurate error prediction and therefore less rejected steps.
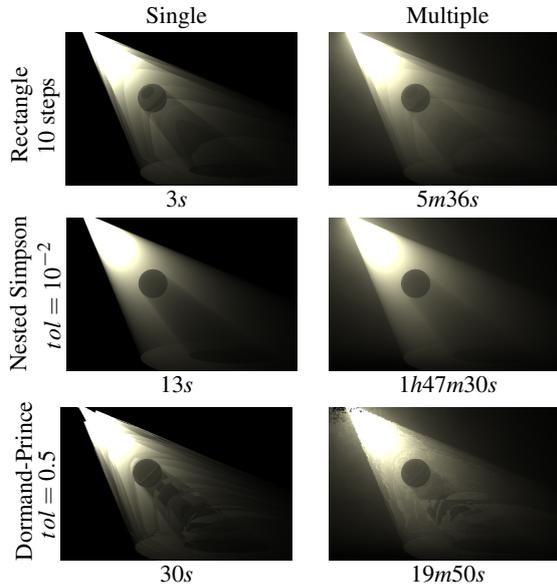

	Single	Multiple
Rectangle 10 steps	3s	5m36s
Nested Simpson tol = 10^{-2}	13s	1h47m30s
Dormand-Prince tol = 0.5	30s	19m50s

Figure 10: *Multiple scattering simulation using photon mapping and several numerical methods.*

Multiple scattering. Previous results account just for single scattering. Multiple scattering calculations can easily be included on the in-scattering term $L_i(\mathbf{x_s}, \omega)$ based on any standard algorithm. However, applying Monte Carlo for gathering indirect bounces would lead to high frequency noise, which is toxic for the presented numerical methods. We use photon mapping instead [Jen01] for the indirect bounces of light, which yields a smooth (but biased) representation of multiple scattering that favors the solvers used in this article. Figure 10 shows the result of several simulations for several methods. An interesting outcome is that, for adaptive solvers, a specific tolerance parameter that leads to efficient accurate results for single scattering may be too restrictive for multiple scattering (Nested Simpson method, Figure 10, middle row), while adaptive methods that lead to visible artifacts for single scattering may improve their results when including multiple scattering (Dormand-Prince method, Figure 10, bottom row).

7. Conclusions and future work

We have revisited the concept of ray marching as the use of a numerical solver for the Radiative Transfer Equation, both on its differential form (initial value problem solvers) and on its integral form (quadrature rules). While there are definitely many more numerical solvers in the literature, we have analyzed a significant representative subset of them, that enables us to extract conclusions in side-by-side comparisons between different order methods and between fixed

and adaptive techniques. Apart from Monte Carlo integration, we have analyzed two constant step IVP solvers (Euler and order four Runge-Kutta) and two constant step quadrature rules (rectangle rule, which is the standard ray marching technique, and order two Simpson rule). We have shown that, while on ideal scenarios higher order translates into faster convergence, in the presence of strong shadows the performance of all the methods is similar to Monte Carlo for the same number of samples, no matter the order of the numerical solver.

We have also taken into account several adaptive techniques: Bogacki-Shampine (order three) and Dormand-Prince (order five) as IVP solvers and Nested Simpson rule (order two) and Gauss-Kronrod quadrature rules. Given a tolerance parameter, those techniques adapt the location of the samples according to a local estimation of the error. The tolerance parameter of such methods is unintuitive to set: First, error control is local (per-step) and therefore there is no direct control over global error. Furthermore, each of the methods estimates the error at a different order, so there is no optimal tolerance parameter for all of them. However, given a specific adaptive method, its optimal tolerance parameter does not vary greatly between different scenes. . We have shown how this adaptiveness maximizes accuracy for a number of scenarios, including sharp shadows and heterogeneous media. Adaptive techniques show a better convergence rate than constant step methods and Monte Carlo quadrature.

We have also included in our formulation two widely used participating media importance sampling techniques: distance sampling [RSK08] and equiangular sampling [KF12], which are no longer Monte Carlo specific but can be applied with any numerical method. All constant step methods distribute their samples according to the importance, and adaptive methods use the importance information as a prior for their adaptation strategy. Interesting paths for future research could involve including other importance sampling strategies, such as strategies dealing with heterogeneous media (such as Woodcock tracking [WMHL65]) or combined strategies (multiple importance [Vea97, KF12] or joint importance sampling [GKH*13]) for any generic solver.

Our tests show that IVP solvers present a better performance at heterogeneous media, while quadrature rules are better suited to deal with equation stiffness (singularities caused by point lights or optically thick media). Furthermore, the convergence of adaptive methods is tied to the error adaptation strategy. Specifically, IVP solvers adapt step size to local error estimations at every single step, which is an adequate strategy for heterogeneous media, while quadrature rules split the integration range recursively, which is suited to deal with sharp discontinuities such as shadows.

As future research, it would be interesting to test how the use of these numerical solvers can be applied to other techniques. Any technique that is based on ray marching (or on sampling points along the path of light using Monte Carlo

integration) would be suitable for these numerical solvers. We have already shown that it can be used for volumetric photon mapping [Jen01]. For instance, it could be applied to integrate light transport between camera rays and virtual ray lights [NNDJ12b] or to sample light and extinction distributions along photon beams or ray beams [JNSJ11]. We hope that our work inspires further research on efficient participating media rendering, which is still an enormous and difficult task.

Acknowledgements

References

[BS89] BOGACKI P., SHAMPINE L.: A 3(2) pair of runge - kutta formulas. *Applied Mathematics Letters 2*, 4 (1989), 321 – 325. 5

[Cha60] CHANDRASEKHAR S.: *Radiative Transfer*. Dover Publications, Inc., 1960. 1, 2

[CL85] CODDINGTON E. A., LEVINSON N.: *Theory of Ordinary Differential Equations*, 9 ed. Tata Mcgraw-Hill, 1985. 2, 4

[Col68] COLEMAN W.: Mathematical verification of a certain monte carlo sampling technique and applications of the technique to radiation transport problems. *Nucl. Sci. Eng. 32* (1968), 76–81. 2

[DP80] DORMAND J., PRINCE P.: A family of embeded runge-kutta formulae. *Journal of Computational and Applied Mathematics 6(1)* (1980), 19–26. 5

[EJNT88] ENRIGHT W., JACKSON K., NORSETT S., THOMSEN P.: Effective solution of discontinuous IVPs using a Runge-Kutta formula pair with interpolants. *Applied Mathematics and Computing 27* (1988), 313–335. 7

[Gea71] GEAR C.: *Numerical initial value problems in ordinary differential equations*. Prentice-Hall series in automatic computation. Prentice-Hall, 1971. 2, 4

[GJJD09] GUTIERREZ D., JENSEN H. W., JAROSZ W., DONNER C.: Scattering. In *ACM SIGGRAPH ASIA 2009 Courses* (New York, NY, USA, 2009), SIGGRAPH ASIA '09, ACM, pp. 15:1–15:620. 2

[GKH*13] GEORGIEV I., KŘIVÁNEK J., HACHISUKA T., NOWROUZEZAHRAI D., JAROSZ W.: Joint importance sampling of low-order volumetric scattering. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2013) 32*, 6 (Nov. 2013). 2, 9

[GSMA06] GUTIERREZ D., SERON F. J., MUNOZ A., ANSON O.: Simulation of atmospheric phenomena. *Comput. Graph. 30*, 6 (Dec. 2006), 994–1010. 2

[Jen01] JENSEN H. W.: *Realistic image synthesis using photon mapping*. A.K. Peters, Natick, Massachussets, 2001. 1, 2, 3, 9, 10

[JLSJ11] JOHNSON J. M., LACEWELL D., SELLE A., JAROSZ W.: Gaussian quadrature for photon beams in tangled. In *ACM SIGGRAPH 2011 Talks* (New York, NY, USA, Aug. 2011), SIGGRAPH 2011, ACM. 2

[JNSJ11] JAROSZ W., NOWROUZEZAHRAI D., SADEGHI I., JENSEN H. W.: A comprehensive theory of volumetric radiance estimation using photon points and beams. *ACM Transactions on Graphics (Presented at ACM SIGGRAPH 2011) 30*, 1 (Jan. 2011), 5:1–5:19. 2, 10

[JZJ08] JAROSZ W., ZWICKER M., JENSEN H. W.: The beam radiance estimate for volumetric photon mapping. *Computer Graphics Forum (Proceedings of Eurographics 2008) 27*, 2 (Apr. 2008), 557–566. 2

[KF12] KULLA C., FAJARDO M.: Importance sampling techniques for path tracing in participating media. In *Computer Graphics Forum* (2012), vol. 31, Wiley Online Library, pp. 1519–1528. 2, 5, 8, 9

[KvH84] KAJIYA J. T., VON HERTZEN B. P.: Ray tracing volume densities. *Computer Graphics 18*, 3 (1984), 165–174. 2

[Muñ12] MUÑOZ A.: Differential ray marching. In *Proceedings of the XXII Spanish Computer Graphics Conference* (2012), pp. 91–98. 2, 4

[NNDJ12a] NOVÁK J., NOWROUZEZAHRAI D., DACHSBACHER C., JAROSZ W.: Progressive virtual beam lights. *Computer Graphics Forum (Proceedings of EGSR 2012) 31*, 4 (June 2012). 2

[NNDJ12b] NOVÁK J., NOWROUZEZAHRAI D., DACHSBACHER C., JAROSZ W.: Virtual ray lights for rendering scenes with participating media. *ACM Trans. Graph. 31*, 4 (July 2012), 60:1–60:11. 2, 10

[Pat68] PATTERSON T. N. L.: The optimum addition of points to quadrature formulae. *Mathematics of Computation 22*, 104 (1968), 847–856+s21–s31. 4

[PH89] PERLIN K., HOFFERT E. M.: Hypertexture. *SIGGRAPH Comput. Graph. 23*, 3 (July 1989), 253–262. 1, 2

[PM93] PATTANAIK S. N., MUDUR S. P.: Computation of global illumination in a participating medium by monte carlo simulation. *The Journal of Visualization and Computer Animation 4*, 3 (1993), 133–152. 2

[PSS11] PEGORARO V., SCHOTT M., SLUSALLEK P.: A mathematical framework for efficient closed-form single scattering. In *Graphics Interface* (2011), Brooks S., Irani P., (Eds.), Canadian Human-Computer Communications Society, pp. 151–158. 2

[PTVF07] PRESS W. H., TEUKOLSKY S. A., VETTERLING W. T., FLANNERY B. P.: *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 2007. 2, 4, 5

[RSK08] RAAB M., SEIBERT D., KELLER A.: Unbiased global illumination with participating media. In *Monte Carlo and Quasi-Monte Carlo Methods*. Springer, 2008, pp. 591–606. 2, 5, 9

[RT87] RUSHMEIER H. E., TORRANCE K. E.: The zonal method for calculating light intensities in the presence of a participating medium. *j-COMP-GRAPHICS 21*, 4 (July 1987), 293–302. 2

[SS66] STROUD A. H., SECREST D.: *Gaussian Quadrature Formulas*. Prentice-Hall series in automatic computation. Prentice-Hall, 1966. 2

[Vea97] VEACH E.: *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis, Stanford University, 1997. 2, 9

[WMHL65] WOODCOCK E., MURPHY T., HEMMINGS P., LONGHWORTH S.: Techniques used in the gem code for monte carlo neutronics calculations in reactors and other systems of complex geometry. In *Applications of Computing Methods to Reactor Problems*. (1965), Argonne National Laboratories Report. 2, 9