粒径分布を考慮した大気と雲の写実的なレンダリング

Photorealistic Rendering of Atmosphere and Clouds Taking into Account Particle Size Distribution

Yonghao Yue[1]　岩崎 慶[2]　陳 炳宇[3]　土橋 宜典[4]　西田 友是[1]
[1]東京大学　[2]和歌山大学　[3]國立臺灣大學　[4]北海道大学

Yonghao Yue[1]　Kei Iwasaki[2]　Bing-Yu Chen[3]　Yoshinori Dobashi[4]　Tomoyuki Nishita[1]
[1]The University of Tokyo　[2]Wakayama University　[3]National Taiwan University　[4]Hokkaido University

概要

屋外シーンのレンダリングにおいて，写実的な空の描写は重要であり，空気分子，エアロゾル，水滴による光の散乱効果を計算することが重要である．本研究では，粒径分布を考慮して雲やエアロゾルの散乱パラメータを計算し，確率論的に unbiased な方法によってレンダリング方程式を解く．本稿で提案する区間分割に基づくサンプリング法により，従来の unbiased な方法よりも 2~3 桁程度の高速化を実現できる．比較実験により，粒径分布の考慮による効果，またサンプリング効率の向上を示す．レンダリング例として，多重散乱まで考慮した大気と雲のレンダリング結果を示す．

1. はじめに

屋外シーンのレンダリングにおいて，写実的な空の描写は重要である．時刻とともに空の見え方は変化し，光跡や夕焼けのように，しばしば印象的な視覚的効果をもたらす．空は窒素，酸素，水蒸気，二酸化炭素などの気体からなる空気と，すす，塵，微生物，海洋性の塩，火山性の硫安などの微粒子(エアロゾル)，及び雲や霧などに含まれる水滴から構成される．光が空を通過するとき，光は空中の粒子によって吸収されたり散乱されたりする．このように散乱効果をもたらす媒質は関与媒質(以下単に媒質と呼ぶ)と呼ばれ，散乱による照明効果を計算するには，各構成物質の散乱を既述する散乱パラメータを求め，散乱過程を既述した複雑なレンダリング方程式を解く必要がある．

本研究では，粒径分布を考慮して雲やエアロゾルの散乱パラメータを計算する．また，確率論的に unbiased な方法によってレンダリング方程式を解く．本稿で提案する区間分割に基づくサンプリングを行うことにより，従来の unbiased な方法よりも 2~3 桁程度の高速化を実現できる．比較実験により，粒径分布の考慮による効果，またサンプリング効率の向上を示す．レンダリング例として，多重散乱まで考慮した大気と雲のレンダリング結果を示す．

2. 関連研究

媒質のためのレンダリング手法はさまざま提案されており，詳細なサーベイは文献[2]に記述があるので，本稿では概略のみ述べる．また，前計算を行ってリアルタイムにレンダリングを行う手法もあるが，本研究とは目的が異なるので，ここでは触れない．

大気の表示法として，大気の色分布を時刻や大気の汚染度によってパラメータ化した手法[15]があり，また夜空を表示する手法[24]が開発されている．物理則に忠実な計算法として，光による散乱を考慮した手法[13]や，ミー散乱をより正確に考慮した手法[8]がある．雲の多重散乱を計算する手法としては文献[14]がある．

媒質のレンダリング手法として，ボクセルベースのラジオシティ法[14]やサンプリングベースの手法があり，後者はさらにレイマーチングに基づく手法[4]とランダムサンプリングに基づく手法[3,17]に分けられる．ボクセルベースのラジオシティ法では，計算に膨大なメモリが必要であるため，ボクセル数の多いシーンのレンダリングには向いていない．レイマーチングに基づく手法は，biased な手法[1]であり，サンプリング間隔に応じて結果が変化し，適切な調節が必要である．確率論に基づくランダムサンプリングの手法は，unbiased であり，正しい結果に収束することが理論的に保証されている．しかし大気や雲のように，媒質が非一様で，空間が広大なシーンでは，従来法[3,17]のサンプリング効率は悪く，実用的ではない．本研究では，文献[3,17]の手法を改良し，2~3 桁の計算時間の短縮を実現する．

最後に補足として，大気や雲のレンダリングに必要な散乱パラメータ等に関して以下に概観する．大気や雲をレンダリングするときに，媒質中の構成物質の種類や大きさ，また散乱パラメータの計算に必要な屈折率を知ることが重要である．気象学の分野では，空気やエアロゾル，雲に関する観測が盛んに行われており，

---

1 厳密解とは異なる値を返す可能性がある手法

文献［1,6,7,10,11,19,20］が挙げられる．これらの構成物質の大きさや屈折率に応じて，散乱のパラメータを計算する方法はLorenz-Mie理論[12,21]によって確立されており，Lorenz-Mie 理論に基づく効率的な計算法[5,22]や小さな粒子のためのレイリー近似[1]が提案されている．地面の反射率の測定については，たとえば文献[16]が挙げられる．

3. 散乱計算

光が媒質中を通過するとき，光は媒質によって吸収されたり，散乱したりする．これらの効果は次式の方程式によって記述される．

$$L(\lambda,\mathbf{x},\vec{\omega})=e^{-\tau(\lambda,\mathbf{y},\mathbf{x})}L(\lambda,\mathbf{y},\vec{\omega})+ \int_{\mathbf{y}}^{\mathbf{x}} e^{-\tau(\lambda,\mathbf{x}',\mathbf{x})}\frac{k_s(\lambda,\mathbf{x}')}{4\pi}\int_{S^2} p(\lambda,\mathbf{x}',\vec{\omega}'\cdot\vec{\omega})L(\lambda,\mathbf{x}',\vec{\omega}')d\omega' dx' \tag{1}$$

$$\tau(\lambda,\mathbf{x}',\mathbf{x})=\int_{\mathbf{x}'}^{\mathbf{x}} k_t(\lambda,\mathbf{x}'')dx''$$

ここで，$L(\lambda,\mathbf{x},\vec{\omega})$は，$\vec{\omega}$方向から位置$\mathbf{x}$に向かう波長$\lambda$の入射光の輝度を表し，$\mathbf{y}$は，位置$\mathbf{x}$，射出方向$\vec{\omega}$を起点とする光線と媒質の境界(地表や大気の境界面)との交点を表す．k_sは媒質の散乱係数(scattering coefficient)，pは媒質の位相関数，k_tは媒質の消滅係数(extinction coefficient)を表し，τは光学深度を表す．

式(1)によって各ピクセルの輝度Lについて計算を行うには，散乱効果を記述するパラメータである消滅係数k_t，散乱係数k_s，位相関数pの空間分布が与えられている必要がある．これらは，媒質の粒径分布や密度分布がわかると，Lorenz-Mie 理論[5,12,21]によって計算することができ，4 節にて詳述する．次に，式(1)の積分方程式を解くにあたり，本研究では確率論に基づく効率的なインポータンスサンプリング法を提案し，unbiased で正確な計算を行う．これについては 5 節で詳説する．提案法のサンプリング法は，モンテカルロ光路追跡法，メトロポリスサンプリング[18]，フォトンマッピング法[9]など，さまざまな手法と組み合わせて利用することができる．本稿では，モンテカルロ光路追跡法への応用に焦点をあてる．

4. 散乱パラメータ計算

文献[6]によれば，空気の消滅係数は，エアロゾルと水滴を含む大気全体の消滅係数の約 1 割程度であり，残りはエアロゾルと水滴が占める．このため，本研究では媒質として，空気，エアロゾル，水滴を考慮する．水滴は雲や霧にだけ存在するのではなく，エアロゾルにも吸着している．空気は気体であり，その散乱特性はレイリー散乱によって近似できる．エアロゾルと水滴は空中に粒子として存在し，それらの散乱特性はLorenz-Mie 理論[5,12,21]をもとに計算される．以下では，まず空気，エアロゾル，水滴の各構成要素についてそれぞれ散乱パラメータを計算する方法を説明し，最後にこれらの構成要素が混合した場合(すなわち実際の大気と雲)についての扱いを述べる．

4.1. 空気の散乱パラメータ計算

文献[1]により，標準空気の scattering cross section $\sigma_{s,air}$，extinction cross section $\sigma_{t,air}$，及び位相関数p_{air}は下記のように与えられる．

$$\sigma_{t,air}(\lambda)=\sigma_{s,air}(\lambda)=\frac{24\pi^3\left(\eta^2(\lambda)-1\right)^2}{\lambda^4 N_s^{\ 2}\left(\eta^2(\lambda)+2\right)^2}\left(\frac{6+3p_n(\lambda)}{6-7p_n(\lambda)}\right)$$

$$p_{air}(\lambda,\theta)=\frac{3}{4(1+2\gamma(\lambda))}\left[(1+3\gamma(\lambda))+(1-\gamma(\lambda))\cos^2\theta\right]$$

ここで，ηは空気の屈折率，p_nはdepolarization factorと呼ばれ，文献[1]にデータが記載されている．$N_s=2.54743\times10^{25}$は 1m^{3}中の空気の粒子数であり，$\gamma(\lambda)=p_n(\lambda)/(2-p_n(\lambda))$である．大気は高度に応じて密度が変化し，高度$h$での粒子数は$N_{air}(h)=N_s\exp(-h/H_{air})$のように指数関数的に減少する．ここで, H_{air}は大気のスケールハイトである．これらをもとにして，

$$k_{t,air}(\lambda,h)=\sigma_{t,air}(\lambda)N_{air}(h),\quad k_{s,air}(\lambda,h)=\sigma_{s,air}(\lambda)N_{air}(h)$$

と計算される．

4.2. エアロゾルと水滴の散乱パラメータ計算

エアロゾルと水滴については，まず，光の波長と単一の粒子が与えられた場合の散乱効果を表すextinction efficiency Q_t と, scattering efficiency Q_s，位相関数p' をLoerenz-Mie理論によって計算する.具体的な計算法は文献[22]を参照されたい．図 1 に水滴の場合のQ_tのグラフを示す．

次にこれらのパラメータと任意の場所での粒径分布(半径が r から $r+dr$ 間にある粒子の個数が $N(r)dr$ で与えられる) が与えられた場合，レンダリングに必要なパラメータである消滅係数，散乱係数，位相関数はそれぞれ下記のように計算できる．

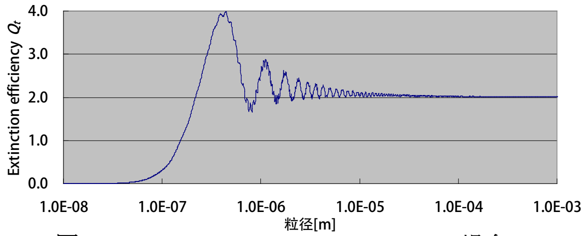


図 1　Extinction efficiency (λ=600nm の場合)

$$k_t = \int_0^\infty Q_t(r/\lambda)\pi r^2 N(r)dr \qquad (2)$$

$$k_s = \int_0^\infty Q_s(r/\lambda)\pi r^2 N(r)dr \qquad (3)$$

$$p(\lambda,\theta) = \frac{\int_0^\infty p'(\lambda,r,\theta)Q_s(\lambda,r)\pi r^2 N(r)dr}{\int_0^\infty Q_s(\lambda,r)\pi r^2 N(r)dr} \qquad (4)$$

まず，エアロゾルの粒径分布の計算法について説明し，その後水滴の粒径分布の計算法を述べる．

気象学の分野では，エアロゾルの粒径分布はLog-normal 分布で近似できることが知られており[10]，次式で与えられる．

$$NL(r;\sigma,r_{eff}) = \frac{1}{\sqrt{2\pi}\sigma r}\exp\left(-\frac{\ln^2(r/r_{eff})}{2\sigma^2}\right)$$

ここで，σ, r_{eff}は分布を表すパラメータであり，それぞれ，分布の広がり，有効半径(代表半径)を表す．エアロゾルは複数の物質から構成されており，その粒径分布はそれぞれの分布の和として，

$$N_a(\mathbf{x},r) = \sum_i N_{a,i}(\mathbf{x})NL_i(r;\sigma_i,r_{eff,i}(\mathbf{x}))$$

と表される．

本研究では，文献[20]にならって，エアロゾルについては，すす，ダスト状エアロゾル，水溶性エアロゾルを考慮する．これらの物質について，Lorenz-Mie理論によってQ_t, Q_s, p'を計算するのに必要なパラメータは文献[20]に記載がある．それぞれの物質についてLog-normal分布を仮定し，下記のパラメータを使用した．

	$N_{a,i}$	σ_i	$r_{eff,i}$
すす	$N_b \times (1\text{-}a)$	0.4	0.5μm
ダスト	$N_b \times a \times (1\text{-}b)$	0.4	0.5μm
水溶性	$N_b \times a \times b$	0.35	0.03μm

a, b は混合比をあらわし，a=b=0.999875 とした[20]．また，N_bは粒子数を表し，高度の関数として，

$$N_b(h) = N_0\exp(-h/H_{aerosol})$$

と計算される．ここで，$H_{aerosol}$はエアロゾルのスケールハイトで 2,500m, N_0は天候条件や場所等により変化し，$2.0\times10^{10}\sim5.0\times10^{11}$程度の値をとる．

水滴の粒径分布についてはガンマ分布によって近似できることが知られており[11]，次式で与えられる．

$$G(r;\alpha,r_{eff}) = \frac{1}{\Gamma(\alpha)r_{eff}}\left(\frac{r}{r_{eff}}\right)^{\alpha-1}\exp\left(-\frac{r}{r_{eff}}\right)$$

αは分布の広がりを表すパラメータである．水滴の粒径分布はガンマ分布を用いて，

$$N_w(\mathbf{x},r) = N_w(\mathbf{x})G(r;2,r_{eff}(\mathbf{x}))$$

と表される．

水滴(雲)の分布は，観測データや水滴半径を考慮したシミュレーションによって得られる，密度と有効半径の空間分布を用いることができる．なお，従来のCG 分野での雲のシミュレーション手法では，水滴半径の変化はシミュレーションされていない．この場合は，雲の実測データをもとに，密度をパラメータとする関数によって有効半径を近似する手法[19]を利用して，

$$r_{eff}(\mathbf{x}) = [3.0+3.1\cosh^{-1}(1+1.8LWC(\mathbf{x}))]\times10^{-6}\ \ [\mathrm{m}] \qquad (5)$$

と計算する．LWC (Liquid Water Content)は空間中に存在する水滴の密度(g/m^3)である．また，$N_w(\mathbf{x})$は，

$$N_w(\mathbf{x}) = 3.0LWC(\mathbf{x})/(\rho_w\cdot4.0\pi r_{eff}^3)$$

と計算され，ρ_wは水の密度(10^{6}g/m^3)である．

なお，Lorenz-Mie理論によってQ_t, Q_s, p'を計算したり，式(2)~(4)の積分計算を実行したりすることは時間がかかるので，各分布関数について散乱パラメータを事前計算しておき，テーブル化する．

4.3. 空間中の散乱パラメータ計算

前節で述べた方法を用いて，空気，エアロゾル，水滴の各構成要素の散乱パラメータを計算することができる．任意の場所での散乱パラメータはこれらの混合として，下記のように求める．

$$k_t = k_{t,air} + k_{t,aerosol} + k_{t,water}$$

$$k_s = k_{s,air} + k_{s,aerosol} + k_{s,water}$$

$$p = \frac{k_{s,air}p_{air} + k_{s,aerosol}p_{aerosol} + k_{s,water}p_{water}}{k_{s,air} + k_{s,aerosol} + k_{s,water}}$$

5. インポータンスサンプリング

本節では，式(1)に対するモンテカルロ Estimator を示し，次に効率よく計算するためのインポータンスサ

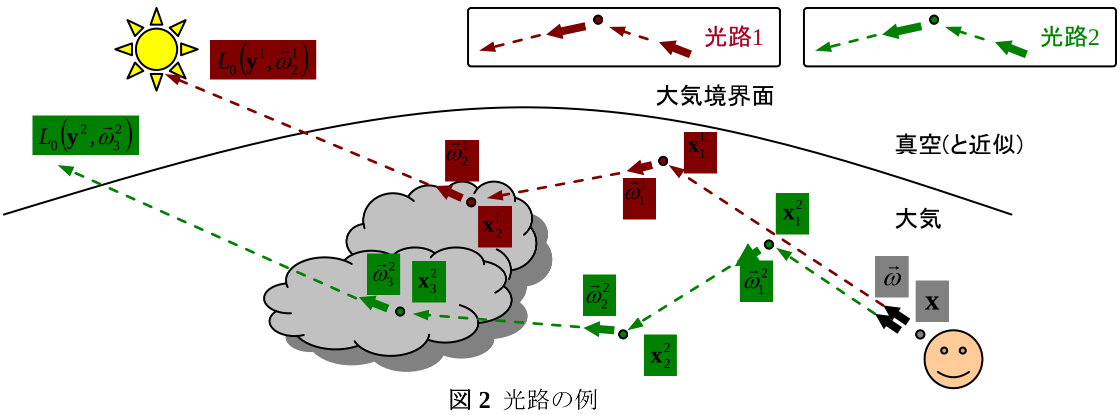


図 2 光路の例

ンプリング法について議論する．

まず，式(1)を下記のように変形する(記号λは省いた)．

$$L(\mathbf{x},\vec{\omega}) = e^{-\tau(\mathbf{y},\mathbf{x})}L(\mathbf{y},\vec{\omega}) + \int_{\mathbf{y}}^{\mathbf{x}}\int_{S^2} e^{-\tau(\mathbf{x}',\mathbf{x})}\frac{k_s(\mathbf{x}')}{4\pi}p(\mathbf{x}',\vec{\omega}'\cdot\vec{\omega})L(\mathbf{x}',\vec{\omega}')d\omega' dx'$$

ここで，

$$L_y(\mathbf{x},\vec{\omega}) = e^{-\tau(\mathbf{y},\mathbf{x})}L(\mathbf{y},\vec{\omega})$$

$$K(\mathbf{x},\vec{\omega},\mathbf{x}',\vec{\omega}') = e^{-\tau(\mathbf{x}',\mathbf{x})}\frac{k_s(\mathbf{x}')}{4\pi}p(\mathbf{x}',\vec{\omega}'\cdot\vec{\omega})$$

$$\Omega = [\mathbf{y}\to\mathbf{x}]\times S^2$$

とおくと，式(1)は，

$$L(\mathbf{x},\vec{\omega}) = L_y(\mathbf{x},\vec{\omega}) + \int_{\Omega} K(\mathbf{x},\vec{\omega},\mathbf{x}',\vec{\omega}')L(\mathbf{x}',\vec{\omega}')d\omega' dx' \quad (6)$$

と変形できる．式(6)は，第二型フレドホルム積分方程式であり，そのモンテカルロ Estimator $\hat{L}(\mathbf{x},\vec{\omega})$は次式のように書ける(図 2 参照)．

$$\hat{L}(\mathbf{x},\vec{\omega}) \approx \frac{1}{N}\sum_{i=1}^{N} L_0^i(\mathbf{x},\vec{\omega}) \quad (7)$$

$$L_d^i(\mathbf{x}_d^i,\vec{\omega}_d^i) = \begin{cases} \dfrac{L_y(\mathbf{y}^i,\vec{\omega}_d^i)}{pdf_{dist}(\mathbf{y}^i)} & \cdots\text{境界に到達した場合} \\ \dfrac{K(\mathbf{x}_d^i,\vec{\omega}_d^i,\mathbf{x}_{d+1}^i,\vec{\omega}_{d+1}^i)}{pdf_{dist}(\mathbf{x}_{d+1}^i)\cdot pdf_{dir}(\vec{\omega}_{d+1}^i)} L_{d+1}^i(\mathbf{x}_{d+1}^i,\vec{\omega}_{d+1}^i) & \cdots\text{散乱が起こる場合} \end{cases} \quad (8)$$

ここで，iは光路番号，dはd番目の散乱を表し，$\mathbf{x}_0^i = \mathbf{x}$，$\vec{\omega}_0^i = \vec{\omega}$であり，$pdf_{dist}$と$pdf_{dir}$はそれぞれ距離と散乱角のサンプリングの確率分布関数である．式(8)の条件分岐は，サンプリングされた距離が，境界までの距離を越えるかどうかで判定される．また，

$$\frac{K(\mathbf{x}_d^i,\vec{\omega}_d^i,\mathbf{x}_{d+1}^i,\vec{\omega}_{d+1}^i)}{pdf_{dist}(\mathbf{x}_{d+1}^i)\cdot pdf_{dir}(\vec{\omega}_{d+1}^i)} = \frac{e^{-\tau(\mathbf{x}_d^i,\mathbf{x}_{d+1}^i)}k_s(\mathbf{x}_{d+1}^i)}{pdf_{dist}(\mathbf{x}_{d+1}^i)}\cdot\frac{p(\mathbf{x}_{d+1}^i,\vec{\omega}_d^i\cdot\vec{\omega}_{d+1}^i)}{4\pi\cdot pdf_{dir}(\vec{\omega}_{d+1}^i)}$$

のように，次の散乱が起こる位置$\mathbf{x}_{d+1}^i$(までの距離)のサンプリング，及び散乱角$\vec{\omega}_{d+1}^i$のサンプリングに分解できる．以下では，式(7,8)による推定が収束するためには，位相関数についてインポータンスサンプリングしなければならないことを示し，そのための方法を示す．また，計算効率向上のための方法として，区間分割を用いた距離についてのインポータンスサンプリング法を提案する．

5.1. 散乱角のサンプリング

式(7,8)のモンテカルロ Estimatorには，$K/[pdf_{dist}\cdot pdf_{dir}]$の乗算があり，散乱の回数だけ掛け合わされる．もし$K/[pdf_{dist}\cdot pdf_{dir}]$が 1 より大きな値をとると，複数回の散乱を経た後の光路のコントリビューションは，いくらでも大きな値を取る可能性がある．その場合，有限回のサンプリングでは，結果が収束するとは限らない．この問題点は，図 3(b)に示すようなスパイク状のノイズとして表れる．もし一次散乱についてのみ計算する場合は，$K/[pdf_{dist}\cdot pdf_{dir}]$が 1 を超える値をとりえても，$K/[pdf_{dist}\cdot pdf_{dir}]$に上限が存在するならば，計算は収束する(図 3(a)参照)．従って，この問題は多重散乱計算特有の問題である．

$K/[pdf_{dist}\cdot pdf_{dir}]$のうち，$e^{-\tau(\mathbf{x}_d^i,\mathbf{x}_{d+1}^i)}k_s(\mathbf{x}_{d+1}^i)/pdf_{dist}(\mathbf{x}_{d+1}^i)$は 5.2 節の方法によって 1 以下であることを保証できる．$p(\mathbf{x}_{d+1}^i,\vec{\omega}_d^i\cdot\vec{\omega}_{d+1}^i)/[4\pi\cdot pdf_{dir}(\vec{\omega}_{d+1}^i)]$については，水滴の場合，位相関数は強い前方散乱を示し，水滴の大きさや散乱角によっては，100,000 程度の大きな値となる．したがって，計算が収束するためには，任意の散乱角について$p(\mathbf{x}_{d+1}^i,\vec{\omega}_d^i\cdot\vec{\omega}_{d+1}^i)/[4\pi\cdot pdf_{dir}(\vec{\omega}_{d+1}^i)]$が 1 以下となることを保証しなければならない．位相関数の定義から，

$$\int_{S^2}\frac{1}{4\pi}p(\mathbf{x},\vec{\omega}'\cdot\vec{\omega})d\vec{\omega}' = 1$$

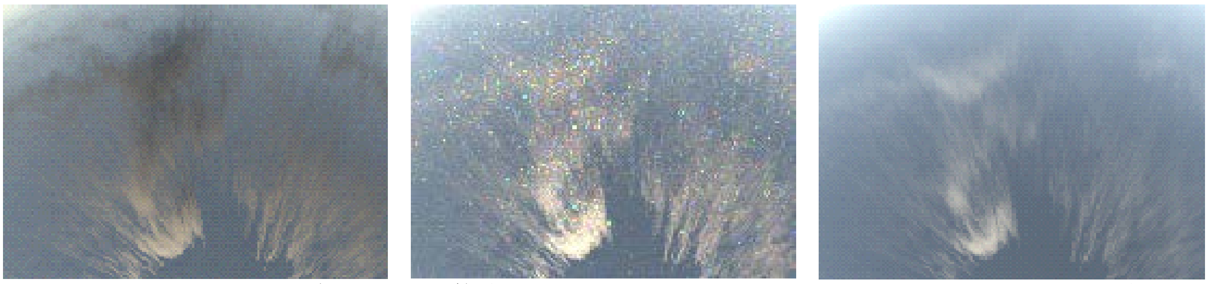


図 3 左，中：一次散乱と多重散乱の計算結果 (位相関数のインポータンスサンプリングは行わない場合)
右：多重散乱の計算結果(位相関数のインポータンスサンプリングを行った場合)

であるから，$pdf_{dir}\left(\vec{\omega}_{d+1}^{i}\right)=p\left(\mathbf{x}_{d+1}^{i},\vec{\omega}_{d}^{i}\cdot\vec{\omega}_{d+1}^{i}\right)/4\pi$ でなければならない．位相関数は複雑な形をしているため，なんらかの解析的な関数で確率分布を近似しても完全に一致させることはできない．これに対して，本研究では，Wavelet Importance Sampling (WIS) [23] を用いる．WIS は，wavelet 展開できる任意の形状の関数をインポータンスサンプリングできる非常に強力な手法である．

以下では，散乱角のサンプリングの概略を述べる．事前に位相関数(とsinθの乗算)のテーブルをHaar waveletによって展開しておく．サンプリング時は，まず初期値 ξ_0 を[0,1)区間内の一様な乱数とする．次に，Hierarchical Warpingによって，各階層のWaveletの重みに応じて ξ_i を修正していく．階層 0 (ルートの階層)，区間[0,1]から始めて，階層iにおいて，ξ_i が属する区間が[s,t]であるときを考える．まず，左右の子ノードの重みをそれぞれW_L, W_Rとすると，この区間[s,t]を$W_L : W_R$ に内分する点cを求める．次に，ξ_i が属する子ノードの区間を図 4 のように区間[s,t]の半分になるようにワープし，ξ_{i+1} を求める．すなわち，

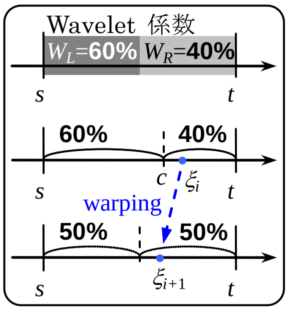


図 4 Hierarchical Warping

$$\xi_{i+1}=\begin{cases}\dfrac{W_L+W_R}{2W_L}\xi_i+\dfrac{W_L-W_R}{2W_L}s & \xi_i<c\\[2ex]\dfrac{W_L+W_R}{2W_L}\xi_i+\dfrac{W_R-W_L}{2W_L}t & \xi_i\geq c\end{cases}$$

とする．この手順を葉ノード(階層Mとする)に行き着くまで再帰的に繰り返していくと，最終的にξ_M の分布は，入力の関数に比例する[23]．このサンプリング法によって，$K/[pdf_{dist}\cdot pdf_{dir}]$は常に 1 以下となることが保証される．レンダリングにおいては，十分なサンプリングを行えば，図 3(c)のように，スパイクノイズはなくなる．

5.2 距離に関するサンプリング

距離に関するサンプリングでは，次の散乱が起こる位置までの距離を，反復的にサンプリングしていく．このとき，次の散乱が起こる位置までに光路が進む区間を区間光路と呼ぶことにする．

従来，確率論的に距離をサンプリングする方法 [3,17] では，Algorithm 1 に示す手順で距離に関するサンプリングを行い，次の散乱が起こるまでの距離 d を求める．

Algorithm 1

```
double sampleDist1(double k_{t,M}, Point x_0, Direction ω){
      double d = - log(rand()) / k_{t,M};
      while(k_t(x_0+d・ω)/ k_{t,M} < rand ())
            d -= log(rand()) / k_{t,M};
      return d;
}
```

ここで，rand()は[0,1)の乱数を返す．このとき，

$$pdf_{dist}\left(\mathbf{x}'=\mathbf{x}+d\cdot\omega\right)=e^{-\tau\left(\mathbf{x}',\mathbf{x}\right)}k_t\left(\mathbf{x}'\right) \tag{9}$$

を満たす[3]．なお，媒質の周り(無限遠まで)が真空であったり，媒質が壁に囲まれていたりする場合は，媒質の外のどこで散乱が起こるのかを考えることは意味がなく，単に媒質の外にでたかどうか，壁にぶつかったかどうかだけが重要である．媒質の境界までの距離をd_M，その外について便宜的に$k_t\equiv c$と定義する．cの値によらずに，

$$\begin{aligned}&P\left(\mathbf{x}'=\mathbf{x}+d\cdot\omega,d<d_M\right)=e^{-\tau\left(\mathbf{x}',\mathbf{x}\right)}k_t\left(\mathbf{x}'\right)\\&P\left(d\geq d_M\right)=e^{-\tau\left(\mathbf{x}+d_M\cdot\omega,\mathbf{x}\right)}\end{aligned} \tag{10}$$

となる．ここで，$P(\cdot)$は，()内の事象の確率を表す．レンダリングでは，d_Mは大気境界面または地面までの距離に設定しておく．5.1 節の内容とあわせて，式(8)は下記のように計算される．

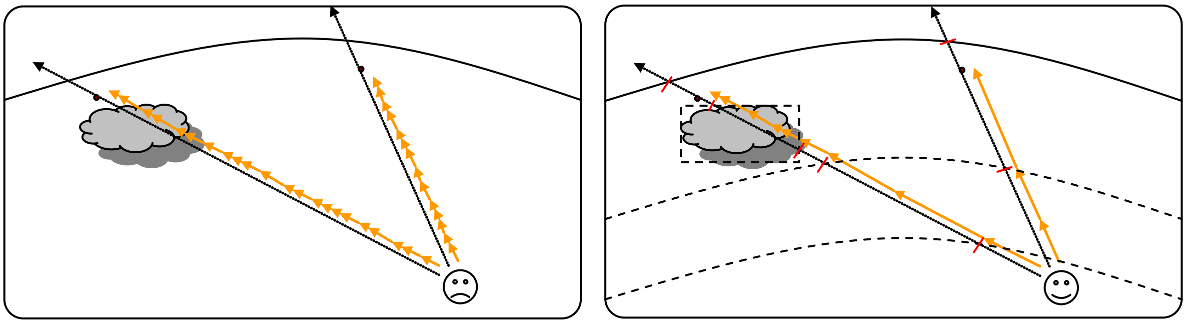


図 5 左: 従来法では，任意の区間光路のサンプリング距離は，領域全体の平均自由行程の最小値に依存するため，サンプリング効率が悪い．右: 提案法では，区間を分割し，区間ごとにサンプリング距離の平均値がかわり，効率的である．

$$L_d^i\left(\mathbf{x}_d^i,\vec{\omega}_d^i\right)=\begin{cases} L\left(\mathbf{y}^i,\vec{\omega}_d^i\right) & \text{境界に到達した場合} \\ \dfrac{k_s\left(\mathbf{x}_{d+1}^i\right)}{k_t\left(\mathbf{x}_{d+1}^i\right)} L_{d+1}^i\left(\mathbf{x}_{d+1}^i,\vec{\omega}_{d+1}^i\right) & \text{散乱が起こる場合} \end{cases}$$

ここで，$k_s\left(\mathbf{x}_{d+1}^i\right)/k_t\left(\mathbf{x}_{d+1}^i\right)$は single scattering albedo であり，物理的な媒質では，必ず 1 以下である．

Algorithm 1 によって，1 回の反復で進む距離は平均 $1/k_{t,M}$ ($1/k_t$は平均自由行程)であり，$k_{t,M}$が大きければ，一定距離を進むのに必要な反復回数が多くなる．確率的に正しくサンプリングするためには，$k_{t\,M}$は区間光路が通過する領域中におけるk_tの最大値以上の値を選ぶ必要がある．従来法[3,17]では，シーン全体におけるk_tの最大値を$k_{t,M}$としている．大気と雲では雲のほうが平均自由行程が短く，通常 2 桁から 3 桁の違いがあるため，従来法では光路が大気を通過するときも，短いサンプリング距離で反復することになり，非効率的である(図 5 左参照)．

改良の一法として，例えば事前に区間光路が通過しうる領域のk_tの最大値を調べてから，距離のサンプリングを行うことが考えられる．これによって区間光路が大気しか通過しないときのサンプリングを高速化できる．しかし，区間光路が大気と雲の領域の双方を通過しうるとき(例えば視点から雲を眺める区間光路)，その全区間を雲の平均自由行程にあわせてサンプリングしなければならず，やはり効率が悪い(図 5 左参照)．

提案法の基本的なアイデアは，サンプリング区間を分割して各区間内でその区間内のk_tの最大値を使用することである(図 5 右参照)．これによって，単一の区間光路中でもサンプリング距離が変化し，雲の領域を通過しているときだけ，サンプリング距離が短くなる．このため，全体的に反復回数が減り，計算効率が向上する．このときに解決すべき課題は下記二点である，一点目は，確率論的に正しいことを保証しながら，区間ごとにサンプリング距離を変える方法である．二点目は，サンプリング効率がなるべく最適になるような区間の分割とそのような区間分割の下での光路の追跡法である．

5.2.1. 区間分割サンプリング

本節では，提案法の核となる区間分割定理を示す．その前に，Algorithm 1 に最短距離d_{min}，最長距離d_{max}を拡張したAlgorithm 2 を示す．

Algorithm 2

```
double sampleDist2(double d_min, d_max, k_t,M, Point x_0, Direction ω){
    double d = d_min - log(rand()) / k_t,M;
    while(d < d_max ∧ k_t(x_0+d・ω)/ k_t,M < rand ())
        d -= log(rand()) / k_t,M;
    return d;
}
```

Algorithm 1 は，Algorithm 2 で，d_{min} = 0, d_{max} = ∞の特殊な場合に相当する．次に区間分割定理を示す．

区間分割定理

区間[*s*,*t*]に関する次の二つのサンプリング手法は，ともに同じ確率で距離 *d* をサンプリングでき，このときの確率は

$P\left(d, s\le d<t\right)=e^{-\tau(d,s)}k_t\left(d\right)$

$P\left(d\ge t\right)=e^{-\tau(s,t)}$

となる(簡便のため，位置を表示するかわりに，距離によって表記した)．

(a) d = sampleDist2(s, t, $k_{t,M}$, $\boldsymbol{x}_0$, ω)
によって求めた場合．ただし，$k_{t,M}=\max_{s\le z\le t} k_t\left(z\right)$.

(b) $s<q<t$ なる q を選び，
  (i) d_1 = sampleDist2(s, q, $k_{t,ML}$, $\boldsymbol{x}_0$, ω)
  によって d_1 を求め ($k_{t,ML}=\max_{s\le z\le q} k_t\left(z\right)$)，$d_1\le q$ であれば，d_1 を返す．
  (ii) もし，(i) で $d_1>q$ ならば，
  d_2 = sampleDist2(q, t, $k_{t,MR}$, x_0, w)
  によって d_2 を求め ($k_{t,MR}=\max_{q\le z\le t} k_t\left(z\right)$)，$d_2$ を返す．

(b)の戦略は，まず区間[*s*,*q*)についてサンプリングを行い，このとき得られた距離d_1がqを超えているなら，開始位置をqに戻してから，区間[*q*,*t*)に対してサンプリングを行う．区間分割定理の証明は付録に示した．

もし，区間を分割した場合に，$k_{t,ML}$と$k_{t,MR}$の少なくとも一方が$k_{t,M}$よりも十分に小さければ，サンプリングの反復回数が減り，サンプリング効率が向上する.もし区間を分割しても，$k_{t,ML}$と$k_{t,MR}$が$k_{t,M}$とほとんど変わ

図 6 空間の分割

らなければ, (b)の手法を用いると, (ii)が成立するときに，かえって反復回数が一回ふえてしまう．したがって，媒質の存在する空間をうまく分割して，サンプリング効率をあげることが重要である．

5.2.2 区間分割データ構造の構築

提案法では区間分割を記述するデータ構造として，kd-Tree を採用する．通常 kd-Tree は xyz 空間上で定義され，座標軸に垂直な平面で区間を分割するが，地球の表面は球面であり，また大気の密度は高度と共に指数関数的に変化することを考慮して，次のような空間で分割する．まず，地球全体を図 6 のように六分割し，次に座標変換を行う(式は Y+の場合のみ).

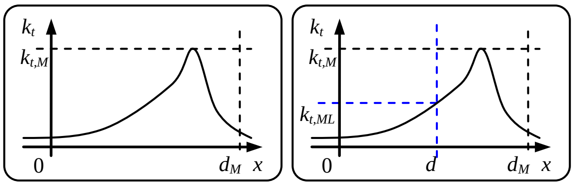
$$\begin{cases} \tilde{x} = x/y \\ h = \left(\sqrt{x^2 + y^2 + z^2} - R_{EARTH}\right)/H \\ \tilde{z} = z/y \end{cases}$$

ここで，$\tilde{x}$，h，$\tilde{z}$ は新しい座標系(以下では $\tilde{x}h\tilde{z}$ 座標系と呼ぶ), R_{EARTH}は地球の半径, Hはスケーリング係数であり，ここでは, 100kmとした(大気の厚さ)．なお, 逆変換は,

$$\begin{cases} x = \tilde{x}\dfrac{hH + R_{EARTH}}{\sqrt{\tilde{x}^2 + 1 + \tilde{z}^2}} \\ y = \dfrac{hH + R_{EARTH}}{\sqrt{\tilde{x}^2 + 1 + \tilde{z}^2}} \\ z = \tilde{z}\dfrac{hH + R_{EARTH}}{\sqrt{\tilde{x}^2 + 1 + \tilde{z}^2}} \end{cases}$$

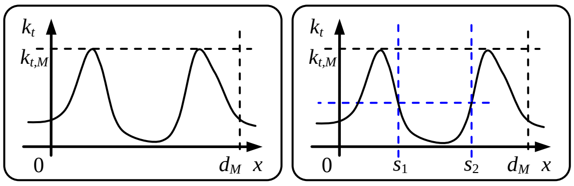
である．上記の座標変換は次の対応関係を示している．h が一定である領域は, xyz 空間内では高度が一定の球面に対応し，$\tilde{x}$ 及び $\tilde{z}$ は, h 一定の球面上の点と原点を結ぶ直線と, y=1 の平面との交点に対応する．また，$\tilde{x}$ が一定である領域は, xyz 空間内では z 軸を含む平面に対応し，$\tilde{z}$ が一定である領域は, xyz 空間内では x 軸を含む平面に対応する．

xyz 空間内の関与媒質は，まず微小な要素に分割される(雲データがボクセルで与えられているときは, 各ボクセルを考える．同様に粒子のデータで与えられている場合は，各粒子の有効半径に応じた球を考える．大気については，高度に応じて層状に分割しておく)．次に各要素の $\tilde{x}h\tilde{z}$ 空間内での Axis Aligned Bounding Box を計算する．こうしておくと，木の分割はあたかも xyz 空間内で分割を行うのと同じように扱うことができる．



(a) (b)

図 7 $k_t(x)$の例 1



(a) (b)

図 8 $k_t(x)$の例 2

木の分割では，最適な分割位置の決定法と，分割を継続するかどうかの判定法を定義する必要があるが, 簡単のため，まず一次元の場合について考える．

x軸上に媒質が分布していると仮定し，任意の位置xでの消滅係数が$k_t(x)$で与えられているとする(図 7(a)参照)．いま，区間[0,d_M]について考える．この区間内のk_tの最大値を$k_{t,M}$ とおくと, 1 反復で進む平均距離は 1/$k_{t,M}$であるから，この区間を通過するのにかかる反復回数の平均は, $d_M \cdot k_{t,M}$で与えられる．仮に，図 7(b)のように位置dで分割すると，区間[0,d]間のk_tの最大値は$k_{t,ML}$であり，区間[0, d_M]を通過した場合の反復回数の平均は$d \cdot k_{t,ML}$+(d_M−d)$\cdot k_{t,M}$+1 になる．最後の項+1 は, 区間を分割したために, 1 回分反復回数が増えるためである．すると，分割後の反復回数が分割前の反復回数より少なければ，すなわち

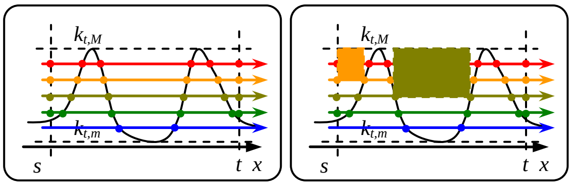
$$\begin{aligned} & d \cdot k_{t,ML} + (d_M - d) \cdot k_{t,M} + 1 < d_M \cdot k_{t,M} \\ \Leftrightarrow\ & d \cdot (k_{t,M} - k_{t,ML}) > 1 \end{aligned}$$

ならば，分割による利得があるといえ，またその利得が最大になるような分割位置 d を見つければよい．

ただし，この戦略には一つ問題がある．例えば, $k_t(x)$のグラフが図 8(a)のように，極小値をとる区間があると, その区間内のどこで分割しても，反復回数を減らすことができない．しかし，仮に図 8(b)のように二回分割すれば，反復回数を減らすことができる．このようなケースにも対応するため，反復のコストを次のように考える．図 7(a)の例では，区間[0, d_M]を通過するときの反復回数は$d_M \cdot k_{t,M}$であったが，これはちょうど，このグラフを囲む矩形の面積となっている．そして，図 7(b)のケースでは，左側を通過する場合の反復回数が$d \cdot (k_{t,M} - k_{t,ML})$だけ減っているが，これは面積が$d \cdot (k_{t,M} - k_{t,ML})$だけ減っていることと対応している．つまり分

Algorithm 3

1. 区間[s,t]内の $k_t(x)$の最大値 $k_{t,M}$ を求める.
2. グラフ $k_t(x)$と $y=k_{t,M}$, $x=s$, $s=t$ で囲まれる領域内に入る矩形の中で，面積が最大のものを探す.
3. 得られた矩形の面積を S とし，$S \le 1$ なら分割を行わず，終了する.
4. $S > 1$ なら，矩形の両端の x 座標を d_1, d_2 とし，区間の中心に近いほうを選択し，d とおく.
5. $x=d$ で区間を二分し，それぞれについてこのアルゴリズムを再帰的に適用する.



(a) (b)

図 9 分割位置の決定過程における矩形候補の列挙

割前のサンプリングでは，グラフ上部の領域が無駄になっており，分割後は，複数の矩形によってグラフをカバーすることで，グラフ上部の無駄な領域が削減されたため，サンプリング効率が向上している．提案法では，グラフ上部の領域を探索し，面積が最大の領域を探し出して，分割位置の候補を決定する．一次元の場合の分割のアルゴリズムをAlgorithm 3 に示す．このうち，面積が最大の矩形を探し出すことは簡単ではないが，次のようにして近似的に解く．

まず，区間内の$k_t(x)$の最大値$k_{t,M}$と最小値$k_{t,m}$を調べて，その間をM等分し，

$$y_i = \frac{i}{M}\left(k_{t,M} - k_{t,m}\right) + k_{t,m}$$

とおき，$y=y_i$とグラフ$k_t(x)$，及び$x=s$, $x=t$の交点を見つける(図 9 参照)．それらの交点を先頭から二つずつ組にすると，その上部には$k_t(x)$のグラフが重なることがないから,この時の上部の領域は取り除く矩形の候補となる．こうして，候補となる矩形を列挙していき，面積が最も大きいもの選択する．

Algorithm 3 による分割の計算コストは，ノード数をNとすると, $O(MN\log N)$であり，ノード数が増えても，計算コストが爆発的に増大することはない．また，分割が進むほど$k_{t,M}$と$k_{t,m}$は近づくので, M等分による近似計算の近似精度は，子ノードほどよくなる．

二次元や三次元の場合も基本的な考え方は同じであるが，グラフの上部の領域を探す操作は一次元の場合よりもはるかに複雑になるので，次のように近似的に求める．各軸について, k_tの最大値をプロットすると，各軸ついては一次元の問題に帰着される．三軸それぞれについて Algorithm 3 を適用して，もっとも分割利得の大きい軸で分割を行う．ただし，これだけだと，図 10 のような場合分割ができないので，次のようにヒューリスティックに対応する．最小値もプロットし，最大値の最小値の差にスケーリング係数をかけたものをk_tとみなして分割候補を探し，最大値による分割候補と合わせて，そのなかから最も利得の大きいものを考える．

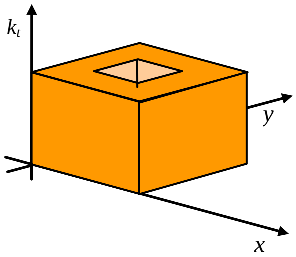


図 10 二次元の例．領域の中央付近のみ kt が小さな値をとる場合．

5.2.3 光路のトラバース

前節で構築した kd-Tree を用いた距離のサンプリング法の擬似コードを Algorithm 4 に示す．

kd-Tree を用いて，光路のトラバースを行う手順は，通常のレイトレーシングの場合に似ているが, h での分割面は球面なので扱いが異なる．各葉ノードにたどり着いたら, Algorithm 2 を実行して距離をサンプリングする．

6. 結果

提案法による計算結果及び比較実験について示す．なお，すべての実験は, Intel Core2 Extreme QX9650 3.00GHz の CPU，及び主記憶 2.0GB を搭載した PC で行った．

まず，粒径分布に関する実験を以下に示す．

4 節では，雲の粒径分布として，ガンマ分布を考慮し，有効半径及び LWC によってパラメータ化できることを示した．ガンマ分布を無視して，単一の粒径の水滴のみから構成される雲を考えることもできるが，写実的なレンダリングではこのガンマ分布を考慮することが重要である．図 11 左は，単一の粒径の水滴のみを考慮した場合で，太陽を中心とする同心円状の模様が見られる．これらは，いわゆる主虹，副虹，三次虹，及びその他の虹からなる系列であり，もし雲が本当に単一の粒径の水滴のみから構成されるときは，このように見えるのが物理的に正しい．しかし，一般的な雲では，単位体積内の水滴の半径にばらつきがある．虹の系列が現れる角度は，粒子の半径に応じて変化するため，粒径の分布に広がりがあると，平均化されて虹が見えにくくなる．実際の雲に含まれる水滴の粒径分布をよく近似できるガンマ分布を考慮することによっ

Algorithm 4

```
double sampleDistkdTree(double t_far, Point x_0, Direction ω){
    double t_near = 0; Stack s; initStack(s);
    kdTreeNode* p = rootNode();
    while(1){
        while(!isLeaf(p)){
            if(splitAt x̃ (p)){
                double d = findIntersection x̃ (p->coord);
                if(d <= t_near) p = p->backNode;
                else if(d >= t_far) p = p->frontNode;
                else{
                    pushStack(stack, p->backNode, d, t_far);
                    p = p->frontNode;
                    t_far = d;
                }}
            else if(splitAt z̃ (p)){
                double d = findIntersection z̃ (p->coord);
                if(d <= t_near) p = p->backNode;
                else if(d >= t_far) p = p->frontNode;
                else{
                    pushStack(s, p->backNode, d, t_far);
                    p = p->frontNode;
                    t_far = d;
                }}
            else{
                double s_m, s_M;
                findIntersectionh(p->coord, &s_m, &s_M);
                if(#solution <= 1) p = p->topNode;
                else{
                    if((t_near < s_m) && (s_M < t_far)) {
                        pushStack(s, p->topNode, s_M, t_far);
                        pushStack(s, p->bottomNode, s_m, s_M);
                        p = p->topNode;
                        t_far = s_m;
                    }
                    else if((t_near < s_m) && (s_m < t_far) && (t_far <= s_M)){
                        pushStack(s, p->bottomNode, s_m, t_far);
                        p = p->topNode;
                        t_far = s_m;
                    }
                    else if((s_m <= t_near) && (t_near < s_M) && (s_M < t_far)){
                        pushStack(s, p->topNode, s_M, t_far);
                        p = p->bottomNode;
                        t_far = s_M;
                    }
                    else if((s_m <= t_near) && (t_far <= s_M))
                        p = p->bottomNode;
                    else p = p->topNode;
                }
            }
        }

        double t = sampleDist2(t_near, t_far, p->maxValue, x_0, ω);
        if(t < t_far) return t;
        if(isEmpty(s)) return INFINITY;
        popStack(s, & p , &t_near, &t_far);
    }
}
```

て，図 11 右のように，同心円状の模様は見えなくなる．

雲の粒径分布は，有効半径及び *LWC* によってパラメータ化できるが，これらが変化した場合に，雲の見え方がどのように影響されるかを以下に示す．まず，これら二つのパラメータの影響を独立に考えるために，式(5)によるパラメータ化を考慮しない場合の結果を図 12 に示す．この実験では，大気とエアロゾルの影響は無視し，光の散乱は多重散乱まで考慮した．図 12 から分かるように，雲の見え方は有効半径と *LWC* の双方に応じて変化する．有効半径が小さく，*LWC* が大きいときは，多重散乱光が支配的となり，雲内部の詳細な形状が見えなくなる．

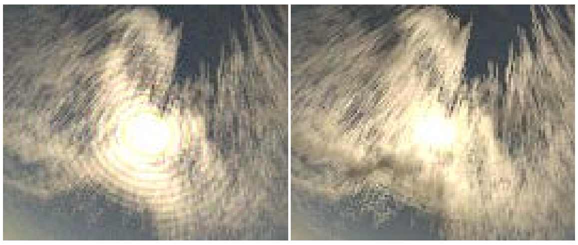


図 11 ガンマ分布を考慮しない場合(左)とガンマ分布を考慮した場合(右)

図 14 従来法と提案法の，区間光路 1,000,000 本あたりの距離のサンプリングにかかる時間

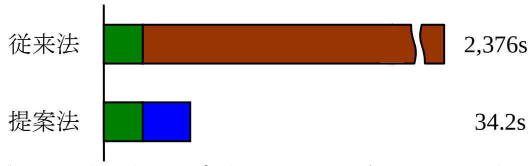


図 15 従来法と提案法の，区間光路 1,000,000 本あたりのレンダリングにかかる時間．緑色はサンプリング以外の計算時間であり，ほぼ共通である．

次に，式(5)による近似を仮定した場合，すなわち,粒径分布が *LWC* のみによってパラメータかされる場合の結果を図 13 に示す. *LWC* の増大に伴って，有効半径も大きくなるため，雲の見え方の違いは，図 13 で有効半径を固定した場合よりも緩やかになり, *LWC* が大きいときも，雲内部の詳細形状が認識できる．

次に，提案法のインポータンスサンプリングによる効果を示す．まず，区間光路のサンプリングの所要時間について示す．図 14 に示したように, 1,000,000 本の区間光路あたりの距離のサンプリングにかかる時間は, 従来法が 2,362 秒であり，提案法は 19.5 秒であった．また，レンダリング時間全体に占める距離のサンプリングの所要時間の変化を調べるために，図 15 に, 1,000,000 本の区間光路あたりのレンダリング時間を示した. Algorithm 1 に基づく従来法[3,17]では，距離のサンプリングが全体の 99.4%を占めており，ボトルネックであったが，提案法では, 57%に軽減されている．

これらの結果から，提案法により計算時間が 2~3 桁程度改善されたといえ，その有効性が示された．なお，

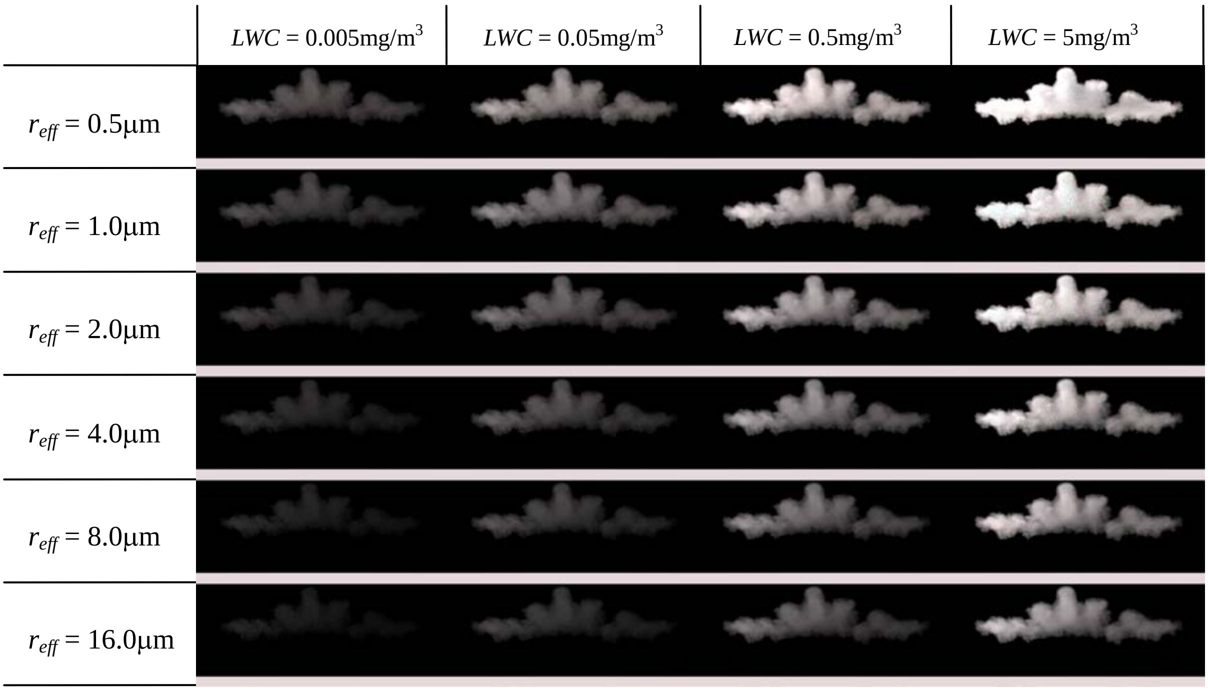


図 12 有効半径と *LWC* を変えた場合の雲の見え方の違い(式(5)による近似は行わない)

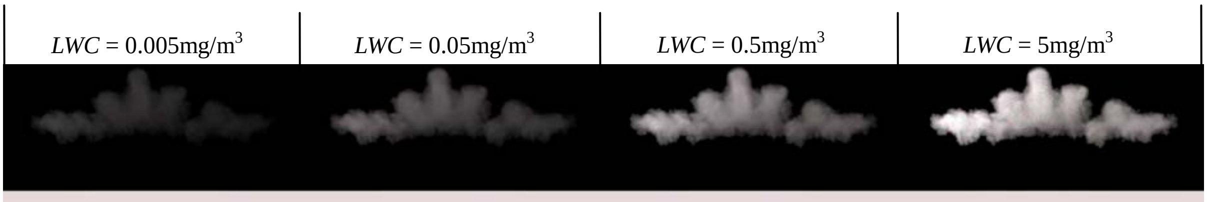


図 13 *LWC* を変えた場合の雲の見え方の違い(式(5)によって有効半径を近似)

kd-Tree の構築時間は数秒から十数秒であった.

最後に，提案法によって，空気とエアロゾル，雲を含む空のレンダリング結果を図16に示す．実験に用いた例では, 5:50am が日の出の時刻であり，その時刻から始めて, 80 分おきの結果を 4 枚示した．画像の解像度は 560×240であり，レンダリングには一枚当たり5時間要した.

7. まとめと今後の課題

本稿では，粒径分布に基づいた，大気と雲の写実的なレンダリング手法について述べた．レンダリングにおいて重要と思われる構成物質の種類やパラメータ，また，散乱計算に必要なパラメータの計算法について述べた．提案法は，確率論に基づく unbiased な手法であり，区間分割に基づくサンプリングによって計算効率が 2~3 桁向上した．また，計算が収束するために，散乱角のインポータンスサンプリングを行う必要があることを述べ，そのための方法を示した.

今後の課題として, GPU を利用した実装が考えられる．また，レンダリング時間短縮のために，メトロポリス法を利用することが考えられる.

参考文献

[1] A. Bucholtz, “Rayleigh-Scattering Calculations for the Terrestrial Atmosphere,” Applied Optics, Vol. 34, No. 15, pp.2765-2773, 1995

[2] E. Cerezo, F. Perez, X. Pueyo, F.J. Seron, F.X. Sillion, “A Survey on Participating Media Rendering Techniques,” Visual Computer, 21: 303-328, 2005

[3] W. Coleman, “Mathematical Verification of a Certain Monte Carlo Sampling Technique and Applications of the Technique to Radiation Transport Problems,” Nuclear Science and Engineering, 32: 76-81, 1968

[4] Y. Dobashi, K. Kaneda, H. Yamashita, T. Okita, T. Nishita, “A simple, efficient method for realistic animation of clouds,” SIGGRAPH 2000, pp. 19-29, 2000

[5] J.R. Frisvad, N.J. Christensen, H.W. Jensen, “Computing the Scattering Properties of Participating Media using Lorenz-Mie Theory,” TOG(Proc.

SIGGRAPH 07), Vol. 26, No. 3, 60, 2007

[6] P.J. Groblicki, G.T. Wolff, R.J. Countess, "Visibility-Reducing Species in the 'Denver Brown Cloud'—I. Relationships between Extinction and Chemical Composition," Atmospheric Environment, Vol. 15, No. 12, pp. 2473-2484, 1981

[7] H. Horvath, "Atmospheric Light Absorption—A Review," Atmospheric Environment, Vol. 27A, No. 3, pp. 293-317, 1993

[8] D. Jackel, B. Walter, "Modeling and Rendering of the Atmosphere Using Mie Scattering," Computer Graphics Forum, Vol. 16, No. 4, 201-210, 1997

[9] H.W. Jensen, P.H. Christensen, "Efficient Simulation of Light Transport in Scenes with Participating Media using Photon Maps," SIGGRAPH '98, pp. 311-320, 1998

[10] A.A. Kokhanovsky, "Aerosol Optics: Light Absorption and Scattering by Particles in the Atmosphere," Springer, 2008

[11] A.A. Kokhanovsky, "Cloud Optics (Atmospheric and Oceanographic Sciences Library)," Springer, 2006

[12] G. Mie, "Optics of Turbid Media," Ann. Physik, Vol. 25, No. 3, pp.37-45, 1908

[13] T. Nishita, Y. Dobashi, K. Kaneda, H. Yamashita, "Display Method of the Sky Color Taking into Acount Multiple Scattering," Proc. Pacific Graphics '96, pp.117-132, 1996

[14] T. Nishita, Y. Dobashi, E. Nakamae, "Display of Clouds Taking into Account Multiple Anisotropic Scattering and Sky Light," SIGGRAPH '96, pp. 379-386, 1996

[15] A.J. Preetham, P. Shirley, B. Smits, "A Practical Analytic Model for Daylight," SIGGRAPH 99, pp. 91-100, 1999

[16] B.E. Psiloglou, H.D. Kambezidis, "Estimation of the Ground Albedo for the Athens Area, Greece," J. Atmospheric and Solar-Terrestrial Physics, Vol. 71, pp.943-954, 2009

[17] M. Raab, D. Seibert, A. Keller, "Unbiased Global Illumination with Participating Media," Monte Carlo and Quasi-Monte Carlo Methods 2006, pp. 591-605, 2006

[18] M. Pauly, T. Kollig, A. Keller, "Metropolis Light Transport for Participating Media," Proc. Eurographics Workshop on Rendering Techniques 2000, pp. 11-22, 2000

[19] J.S. Reid, P.V. Hobbs, A.L. Rangno, D.A. Hegg, "Relationships between Cloud Droplet Effective Radius, Liquid Water Content, and Droplet Concentration for Warm Clouds in Brazil Embedded in Biomass Smoke," J. Geophysical Research, Vol. 104, No. D6, pp. 6145-6153, 1999

[20] E.P. Shettle, R.W. Fenn, "Models for the Aerosols of the Lower Atmosphere and the Effects of Humidity Variations on Their Optical Properties," AFGL-TR-79-0214, 1979

[21] H.C. van de Hulst, "Light Scattering by Small Particles (Structure of Matter Series)," Dover Publications, 1981

[22] W.J. Wiscombe, "Improved Mie Scattering Algorithms," Applied Optics, Vol. 19, No. 9, pp. 1505-1509, 1980

[23] P. Clarberg, W. Jarosz, T. Akenine-Moller, H.W. Jensen, "Wavelet Importance Sampling: Efficiently Evaluating Products of Complex Functions," TOG (Proc. SIGGRAPH 05), Vol. 24, No. 3, pp. 1166-1175, 2005

[24] H.W. Jensen, F. Durand, M.M. Stark, S. Premoze, J. Dorsey, P. Shirley, "A Physically-Based Night Sky Model," SIGGRAPH 2001, pp. 399-408, 2001

付録 区間分割定理の証明

証明 定理における(a)が成り立つのは式(10)より明らかなので, (b)が条件をみたすことを示す. (a)の確率分布から, (i)の手順では,

$$P_{(\mathrm{i})}(d, s \le d < q) = e^{-\tau(d,s)} k_t(d), \quad P_{(\mathrm{i})}(d \ge q) = e^{-\tau(s,q)}$$

なる確率で, d_1がサンプリングされることが分かる.
同じように, (ii)では,

$$P_{(\mathrm{ii})}(d, q \le d < t) = e^{-\tau(d,q)} k_t(d), \quad P_{(\mathrm{ii})}(d \ge t) = e^{-\tau(q,t)}$$

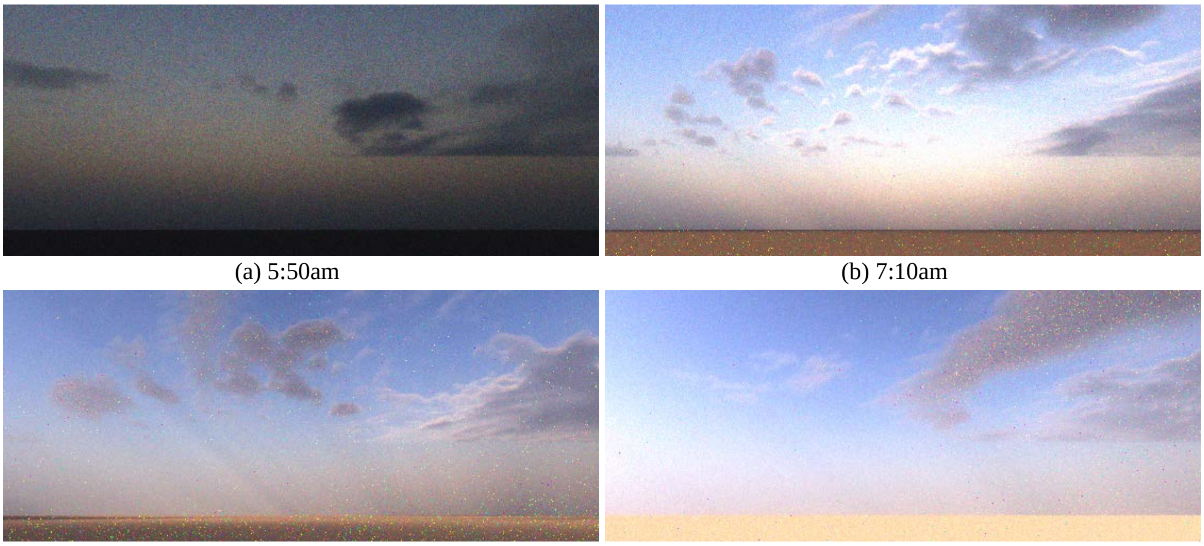
なる確率でd_2がサンプリングされる. (ii)の操作は, (i)の手順で, d_1がqを超えたときしか実行されないので, 結局,

$$P(d, s \le d < q) = P_{(\mathrm{i})}(d, s \le d < q) = e^{-\tau(d,s)} k_t(d)$$

$$P(d, q \le d < t) = P_{(\mathrm{i})}(d \ge q) \cdot P_{(\mathrm{ii})}(d, q \le d < t)$$
$$= e^{-\tau(q,s)} e^{-\tau(d,q)} k_t(d) = e^{-\tau(d,s)} k_t(d)$$

$$P(d \ge t) = P_{(\mathrm{i})}(d \ge t) \cdot P_{(\mathrm{ii})}(d \ge t) = e^{-\tau(q,s)} e^{-\tau(t,q)} = e^{-\tau(t,s)}$$

となる. ■



(a) 5:50am　(b) 7:10am

(c) 8:30am　(d) 9:50am

図 16　大気と雲のレンダリング例