

# Volume Rendering 方程式から 大気散乱モデルを導く

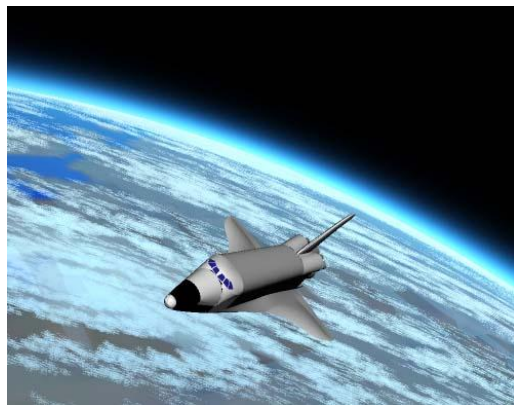
2018.9.1  
レイトレ合宿6  
@ishiyama\_

\* このスライドは, レイトレ合宿6アドベントカレンダーの内容に基づいています :  
[https://github.com/k-ishiyama/rt6\\_advent\\_calendar5/blob/master/doc/atmospheric\\_scattering\\_model.pdf](https://github.com/k-ishiyama/rt6_advent_calendar5/blob/master/doc/atmospheric_scattering_model.pdf)

# 大気散乱モデル

## 地球大気をモデル化 [NSTN93]

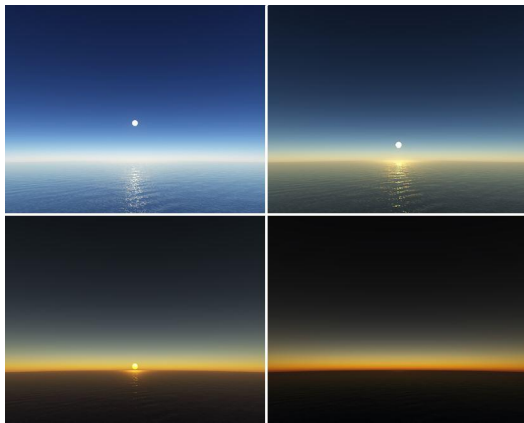
- 光源は太陽のみ
- レイリー散乱、ミー散乱
- 高さ方向に媒質濃度が減衰



[NSTN93]

## 事前計算大気散乱 [BN08, EK10]

- テクスチャに事前計算
- 多重散乱



[EK10]

# 動機

大気散乱モデルで作った天球を IBL に使う

→ レイトレーシングのときに散乱係数を流用しよう

→ Volume Rendering方程式との関係を確認しておく

# 内容

Volume Rendering 方程式から出発して,

[EK10] Oskar Elek and Petr Kmoch. Real-Time Spectral Scattering in Large-Scale Natural Participating Media.

## の内部散乱項

$$I_{S_{\{R,M\}}}^{(1)}(P_O, V, L, \lambda) = R(P_O, V, L, \lambda) + I_I(\lambda) F_{\{R,M\}}(\theta) \cdot \frac{\sigma_{\{R,M\}}(\lambda)}{4\pi} \int_{P_a}^{P_b} \rho_{\{R,M\}}(P) e^{-t(P P_c, \lambda) - t(P_a P, \lambda)} ds \quad (7)$$

$$I_{S_{\{R,M\}}}^{(k)}(P_O, V, L, \lambda) = A_{\{R,M\}}^{(k-1)}(P_O, V, L, \lambda) + \frac{\sigma_{\{R,M\}}(\lambda)}{4\pi} \cdot \int_{P_a}^{P_b} I_{G_{\{R,M\}}}^{(k-1)}(P, V, L, \lambda) \rho_{\{R,M\}}(P) e^{-t(P_a P, \lambda)} ds \quad (2)$$

の導出に焦点を当てる.

# 内容

- 大気散乱モデル
- Volume Rendering 方程式のおさらい
- 大気散乱モデルの内部散乱項の導出
  - 大気散乱モデル: 内部散乱項 (一般)
  - 大気散乱モデル: 内部散乱項 (地球)
- まとめ

# Volume Rendering 方程式のおさらい

**Radiative transfer 方程式**[Cha60] (Notation は [NGHJ18])

$$(\boldsymbol{\omega} \cdot \nabla)L(\boldsymbol{x}, \boldsymbol{\omega}) = -\mu_e(\boldsymbol{x})L(\boldsymbol{x}, \boldsymbol{\omega}) + \mu_a(\boldsymbol{x})L_e(\boldsymbol{x}, \boldsymbol{\omega}) + \mu_s(\boldsymbol{x})L_s(\boldsymbol{x}, \boldsymbol{\omega})$$

$$L_s(\boldsymbol{x}, \boldsymbol{\omega}) = \oint p(\boldsymbol{\omega}, \boldsymbol{\omega}')L_i(\boldsymbol{x}, \boldsymbol{\omega}')d\Omega(\boldsymbol{\omega}')$$

- $\boldsymbol{x}$  : 位置,  $\boldsymbol{\omega}$  : 方向
- $L$  : 光線の輝度、 $L_e$  : 自己発光輝度、 $L_i$  : 入射輝度
- $\mu_a$ : 吸収係数、 $\mu_s$ : 散乱係数、 $\mu_e$  : 消散係数
- $p$  : 散乱の位相関数

# Volume Rendering 方程式のおさらい

**Volume Rendering 方程式** (Radiative Transfer 方程式の積分形)

$$L(\mathbf{x}(s), \boldsymbol{\omega}) = T(\mathbf{x}(0) \rightarrow \mathbf{x}(s)) L(\mathbf{x}(0)) \\ + \int_0^s T(\mathbf{x}(s') \rightarrow \mathbf{x}(s)) [\mu_a(\mathbf{x}(s)) L_e(\mathbf{x}(s), \boldsymbol{\omega}) + \mu_s(\mathbf{x}(s')) L_s(\mathbf{x}(s'), \boldsymbol{\omega})] ds'$$

$$T(\mathbf{x}(a) \rightarrow \mathbf{x}(b)) = \exp \left( - \int_a^b \mu_e(\mathbf{x}(t)) dt \right)$$

ボリウムレンダリングの基礎となる方程式.

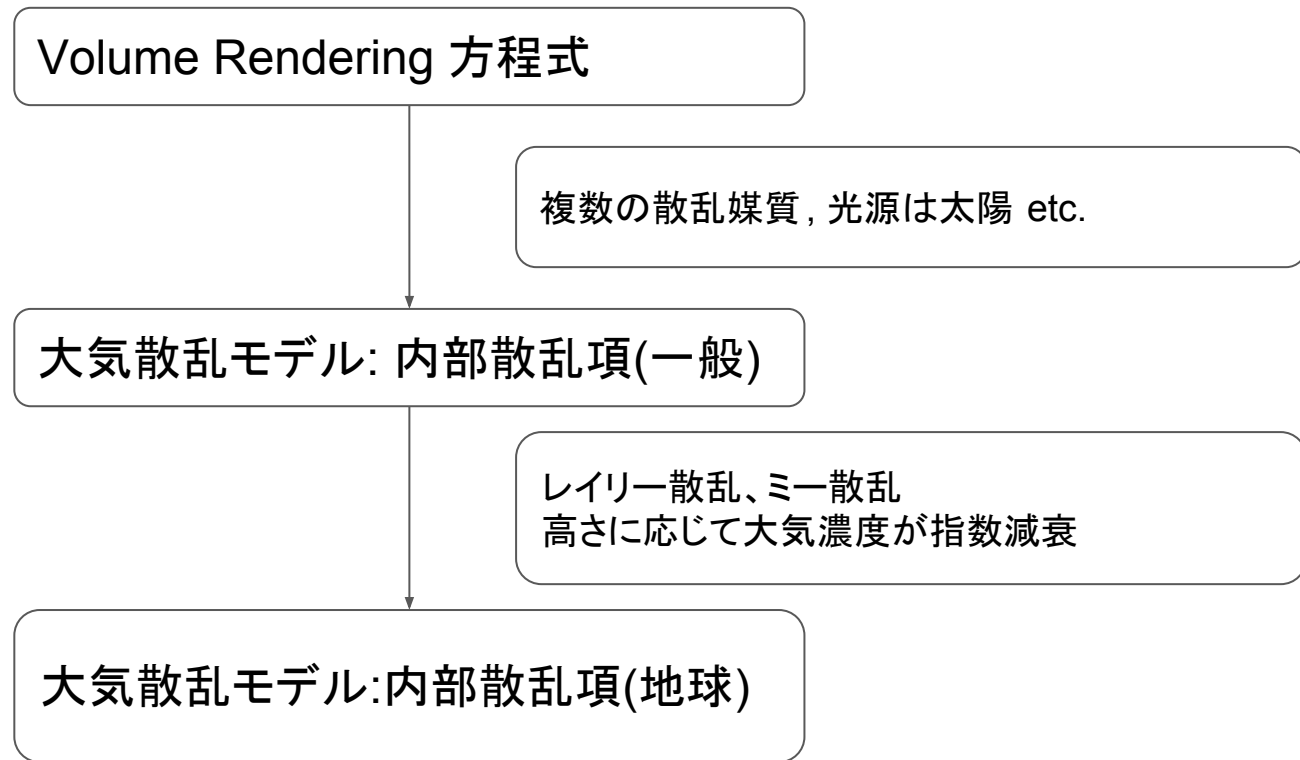
ここから[EK10]の内部散乱項を導く.

# 大気散乱モデルの導出

- 大気散乱モデル: 内部散乱項 (一般)
- 大気散乱モデル: 内部散乱項 (地球)



# 流れ



# Volume Rendering 方程式に対する条件

基本的に [NSTN93] を踏襲

- 自己発光なし

$$L_e(\boldsymbol{x}, \boldsymbol{\omega}) = 0$$

- 複数の散乱媒質

$$\mu_s(\boldsymbol{x}) = \sum_j \mu_{sj}(\boldsymbol{x}), \quad \mu_a(\boldsymbol{x}) = \sum_j \mu_{aj}(\boldsymbol{x})$$

# Volume Rendering 方程式に対する条件

- 光源は太陽光のみ. 太陽光は平行光として近似

$$L_i(\boldsymbol{x}, \boldsymbol{\omega}) = L_{\text{sun}} \delta(\boldsymbol{\omega} - \boldsymbol{\omega}_{\text{sun}})$$

- レイは直進
  - 積分を簡単にするため

- 宇宙と大気、大気と地表を隔てる境界がある
  - 積分範囲を決めるため

# Volume Rendering 方程式に対する条件

- 入射光は前回の散乱結果にのみ依存

$$L_s^{(n)}(\mathbf{x}(s')) = \oint p_j(\boldsymbol{\omega}, \boldsymbol{\omega}') L^{(n-1)}(\mathbf{x}(s'), \boldsymbol{\omega}') d\Omega(\boldsymbol{\omega}')$$

Volume Rendering 方程式を漸化式として書く

$$L^{(n)}(\mathbf{x}(s)) = T(\mathbf{x}(0) \rightarrow \mathbf{x}(s)) L^{(n)}(\mathbf{x}(0)) + \int_0^s T(\mathbf{x}(s') \rightarrow \mathbf{x}(s)) \mu_s(\mathbf{x}(s')) L_s^{(n)}(\mathbf{x}(s')) ds'$$

以降, 内部散乱項に注目

$$L_{\text{inscatter}}^{(n)}(\mathbf{x}(s)) = \int_0^s T(\mathbf{x}(s') \rightarrow \mathbf{x}(s)) \mu_s(\mathbf{x}(s')) L_s^{(n)}(\mathbf{x}(s')) ds'$$

# Volume Rendering 方程式に対する条件

- 散乱媒質間の相互作用なし[EK10]

$$L_{\text{inscatter}}^{(n)} = \sum_n \left( L_R^{(n)} + L_M^{(n)} \right)$$

求める内部散乱項

$$L_{\text{inscatter}}^{(n)}(\mathbf{x}(s)) = \sum_j \int_0^s T(\mathbf{x}(s') \rightarrow \mathbf{x}(s)) \mu_{sj}(\mathbf{x}(s')) L_{sj}^{(n)}(\mathbf{x}(s')) ds'$$
$$L_{sj}^{(n)}(\mathbf{x}(s')) = \oint p_j(\boldsymbol{\omega}, \boldsymbol{\omega}') L_j^{(n-1)}(\mathbf{x}(s'), \boldsymbol{\omega}') d\Omega(\boldsymbol{\omega}')$$

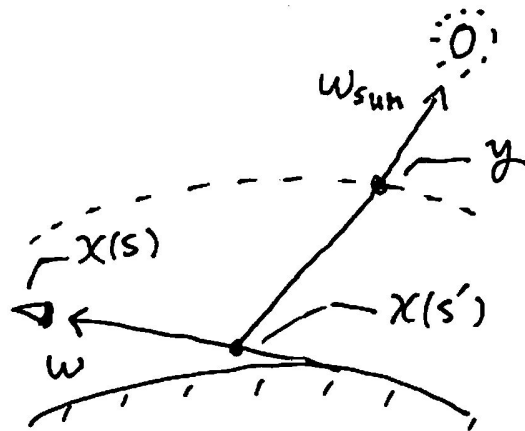
# 1次散乱

入射光: 太陽

$$L_j^{(0)}(\mathbf{x}(s'), \boldsymbol{\omega}') = T(\mathbf{y} \rightarrow \mathbf{x}(s')) L_{\text{sun}} \delta(\boldsymbol{\omega}' - \boldsymbol{\omega}_{\text{sun}})$$

散乱光

$$\begin{aligned} L_{sj}^{(1)}(\mathbf{x}(s')) &= \oint p_j(\boldsymbol{\omega}, \boldsymbol{\omega}') T(\mathbf{y} \rightarrow \mathbf{x}(s')) L_{\text{sun}} \delta(\boldsymbol{\omega}' - \boldsymbol{\omega}_{\text{sun}}) d\Omega(\boldsymbol{\omega}') \\ &= p_j(\boldsymbol{\omega}, \boldsymbol{\omega}_{\text{sun}}) L_{\text{sun}} T(\mathbf{y} \rightarrow \mathbf{x}(s')) \end{aligned}$$



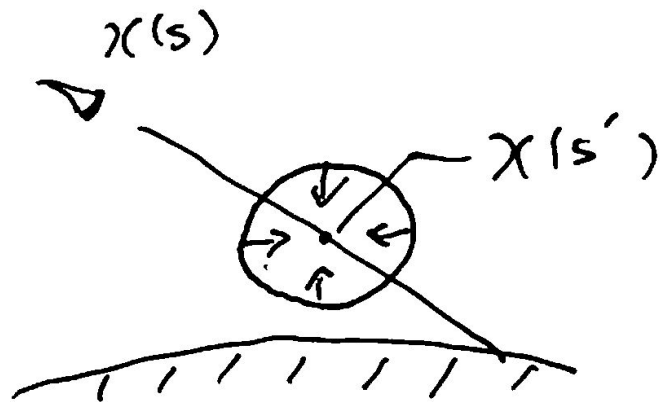
# 高次散乱( $n > 1$ )

入射光: 前回の散乱結果

$$L_j^{(n-1)}(\mathbf{x}(s'), \omega')$$

散乱光

$$L_{sj}^{(n)}(\mathbf{x}(s')) = \oint p_j(\omega, \omega') L_j^{(n-1)}(\mathbf{x}(s'), \omega') d\Omega(\omega')$$



# 大気散乱モデル: 内部散乱項 (一般)

$$L_{\text{inscatter}}^{(n)}(\mathbf{x}(s)) = \sum_j \int_0^s T(\mathbf{x}(s') \rightarrow \mathbf{x}(s)) \mu_{sj}(\mathbf{x}(s')) L_{sj}^{(n)}(\mathbf{x}(s')) ds'$$

Transmittance

$$T(\mathbf{x}(a) \rightarrow \mathbf{x}(b)) = \exp \left( - \int_a^b \sum_j \mu_{ej}(\mathbf{x}(t)) dt \right).$$

1次散乱

$$L_{sj}^{(1)}(\mathbf{x}(s')) = p_j(\boldsymbol{\omega}, \boldsymbol{\omega}_{\text{sun}}) L_{\text{sun}} T(\mathbf{y} \rightarrow \mathbf{x}(s'))$$

高次散乱

$$L_{sj}^{(n)}(\mathbf{x}(s')) = \oint p_j(\boldsymbol{\omega}, \boldsymbol{\omega}') L_j^{(n-1)}(\mathbf{x}(s'), \boldsymbol{\omega}') d\Omega(\boldsymbol{\omega}')$$



# 大気散乱モデル (地球)

あとは, 散乱媒質 を与えれば, 具体的な表現が得られる.

[NSTN93, BN08, EK10] と同様に

- 大気 : レイリー散乱とミー散乱、高度に従って指数関数で減衰

とする

# 散乱媒質に対する条件

## 散乱媒質

- レイリー散乱、ミー散乱

$$p_R = \frac{3}{16\pi}(1 + \cos^2 \theta) \quad p_M = \frac{3}{8\pi} \frac{1 - g^2}{2 + g^2} \frac{1 + \cos^2 \theta}{(1 + g^2 - 2g \cos \theta)^{3/2}}$$

$\times \theta = \cos^{-1}(\boldsymbol{\omega} \cdot \boldsymbol{\omega}')$

- 高さ  $h(\mathbf{x})$  に従って減衰

$$\sum_j^{R, M} \mu_{aj}(\mathbf{x}) = \sum_j^{R, M} \beta_{aj} e^{-\frac{h(\mathbf{x})}{H_j}}, \quad \sum_j^{R, M} \mu_{sj}(\mathbf{x}) = \sum_j^{R, M} \beta_{sj} e^{-\frac{h(\mathbf{x})}{H_j}}$$

# 大気散乱モデル(地球): 内部散乱項

$$L_{\text{inscatter}}^{(n)}(\mathbf{x}(s)) = \sum_j^{R, M} \int_0^s T(\mathbf{x}(s') \rightarrow \mathbf{x}(s)) \mu_{sj}(\mathbf{x}(s')) L_{sj}^{(n)}(\mathbf{x}(s')) ds'$$

Transmittance

$$T(\mathbf{x}(a) \rightarrow \mathbf{x}(b)) = \exp \left( - \int_a^b \sum_j^{R, M} \mu_{ej}(\mathbf{x}(t)) dt \right)$$

1次散乱

$$L_{sj}^{(1)}(\mathbf{x}(s')) = p_j(\boldsymbol{\omega}, \boldsymbol{\omega}_{\text{sun}}) L_{\text{sun}} T(\mathbf{y} \rightarrow \mathbf{x}(s'))$$

多重散乱

$$L_{sj}^{(n)}(\mathbf{x}(s')) = \oint p_j(\boldsymbol{\omega}, \boldsymbol{\omega}') L_j^{(n-1)}(\mathbf{x}(s'), \boldsymbol{\omega}') d\Omega(\boldsymbol{\omega}')$$

# まとめ

Volume Rendering方程式から出発して

- [NSTN93] の条件

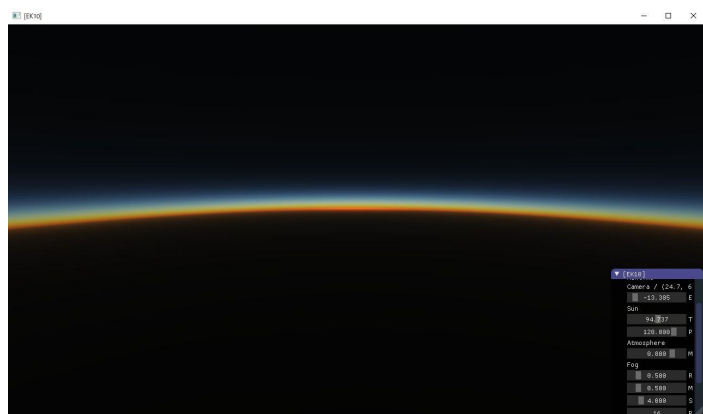
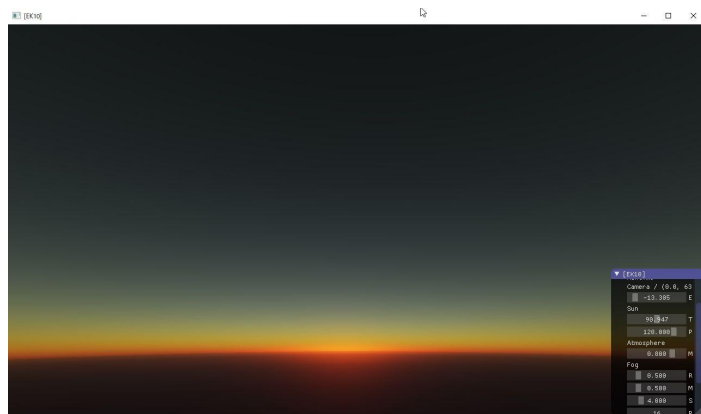
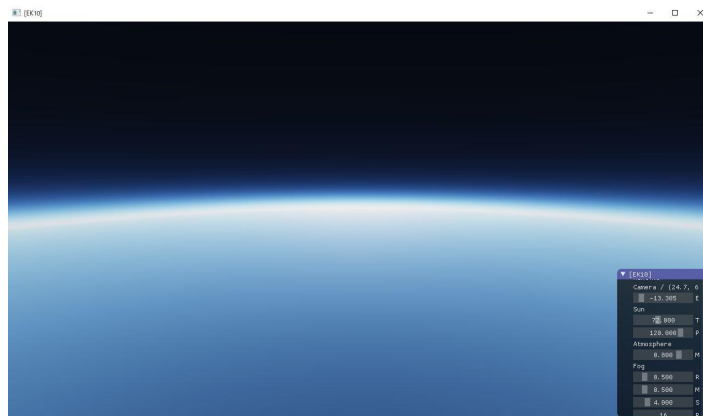
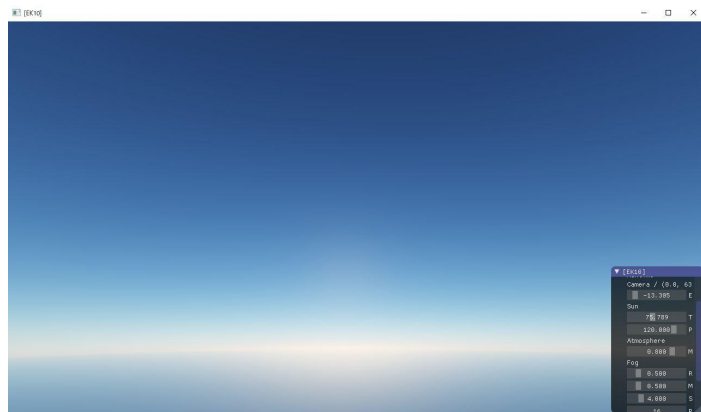
そして

- [EK10] の導出過程

を追うことで [EK10]と同様の内部散乱項を導き,

大気散乱モデルは Volume Rendering 方程式のひとつの形であることを確かめた

# 実装結果: 内部散乱項のみ



# 参考文献

[Cha60] Subrahmanyan Chandrasekhar. *Radiative Transfer*. Dover Publications Inc., 1960

[Kaj86] James T. Kajiya. The rendering equation. *SIGGRAPH Comput. Graph.*, 20(4):143--150, August 1986.

[NSTN93] Tomoyuki Nishita, Takao Sirai, Katsumi Tadamura, and Eihachiro Nakamae. Display of The Earth Taking into Account Atmospheric Scattering. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH'93, pages 175--182, New York, NY, USA, 1993. ACM

[BN08] Eric Bruneton and Fabrice Neyret. Precomputed Atmospheric Scattering. *Computer Graphics Forum*, 27(4):1079--1086, 2008.

[EK10] Oskar Elek and Petr Knoch. Real-Time Spectral Scattering in Large-Scale Natural Participating Media. In *Proceedings of the 26th Spring Conference on Computer Graphics*, SCCG'10, pages 77--84, New York, NY, USA, 2010. ACM

[NGHJ18] Jan Novák, Iliyan Georgiev, Johannes Hanika, and Wojciech Jarosz. Monte carlo methods for volumetric light transport simulation. *Computer Graphics Forum (Proceedings of Eurographics - State of the Art Reports)*, 37(2), may 2018.