Operator Formulation of Light Transport II

CS295, Spring 2017

Shuang Zhao

Computer Science Department
University of California, Irvine
Announcement

• The deadline for PA1 has been extended to this Thursday (May 4)
Last Lecture

- Monte Carlo Path Tracing III
  - MIS for direct illumination

- Operator formulation of light transport
  - Ray space, throughput measure
  - Functions and operators
  - Operator formulation of the RE
Today’s Lecture

• Operator formulation of light transport II
  • Sensors and measurements
  • Adjoint operators and adjoint particle tracing
Recap: Ray Space

• Let $\mathcal{M}$ be the set of surfaces in the scene, then $\mathcal{R} := \mathcal{M} \times S^2$ is the ray space consisting of all light rays $(x, \omega)$ originating from all surface points $x \in \mathcal{M}$.

• Let $r := (x, \omega)$, then the throughput measure is given by

$$d\mu(r) = d\sigma_{\text{area}}(\omega) = |\langle n_x, \omega \rangle| d\omega$$

$$= d\sigma_{\text{projected solid angle}}(\omega)$$

$$= d\sigma_{\text{solid angle}}.$$
Recap: Useful Operators

• For any function \( h : \mathcal{R} \mapsto \mathbb{R} \)

  • The local scattering operator \( K \):
    \[
    (Kh)(x, \omega) := \int_{\Omega_x} h(x, \omega_i) f_r(x, \omega_i \to \omega) \langle n_x, \omega_i \rangle \, d\omega_i
    \]

  • The propagation operator \( G \):
    \[
    (Gh)(x, \omega) := \begin{cases} 
    h(y, -\omega) & \text{ray } (x, \omega) \text{ hits a surface at } y, \\
    0 & \text{ray } (x, \omega) \text{ goes to infinity}
    \end{cases}
    \]
Recap: Operator Formulation of the RE

• Given the two operators \( K \) and \( G \), we can rewrite the rendering equation (RE) as

\[
L = L_e + KG L = L_e + \underbrace{T}_{:=KG} L
\]

• Solving the RE is effectively inverting \((I - T)\)
where \( I \) denotes the identity operator:

\[
L - TL = (I - T)L = L_e
\]

\[
L = (I - T)^{-1}L_e = L_e + TL_e + T^2L_e + \ldots
\]
Operator Formulation of the RE

\[ L = (I - T)^{-1} L_e = L_e + TL_e + T^2L_e + \ldots \]
Operator Formulation of the RE

\[ L = (I - T)^{-1} L_e = L_e + TL_e + T^2 L_e + \ldots \]

Direct illumination

\[
\begin{align*}
& L \\
\Rightarrow & L_e \\
\Rightarrow & TL_e \\
\Rightarrow & \ldots
\end{align*}
\]
Operator Formulation of the RE

\[ L = (I - T)^{-1} L_e = L_e + T L_e + T^2 L_e + \ldots \]

2-bounce indirect illumination
Sensors and Measurements

• Up to this point, we have been considering estimating individual radiance values $L(x, \omega)$ or $L_i(x, \omega)$

• In practice, we often need to estimate the response of some sensor described by the measurement function $W_e : \mathcal{R} \rightarrow \mathbb{R}$

$$I = \langle W_e, L_i \rangle := \int_{\mathcal{R}} W_e(r) L_i(r) \, d\mu(r)$$

$$= \int_{\mathcal{M}} \int_{\Omega_\omega} W_e(x, \omega) L_i(x, \omega) \langle n_x, \omega \rangle \, d\omega \, dx$$
Sensors and Measurements

- \( I = \langle W_e, L_i \rangle \) can be estimated using MC integration:
  \[
  \langle I \rangle = \frac{W_e(x, \omega) L_i(x, \omega) \langle n_x, \omega \rangle}{p_0(x, \omega)},
  \]
  where \( p_0 \) is a probability density over all rays.

- In practice, it is desirable to have \( p_0 \propto W_e \) or
  \[
  p_0(x, \omega) \propto W_e(x, \omega) \langle n_x, \omega \rangle
  \]

- After drawing \( r \sim p_0 \), \( L_i(r) \) can in turn be estimated using path tracing.
Example: Irradiance Meter

\[ I = \int_A \int_{\Omega_x} L_i(x, \omega) \langle n_x, \omega \rangle \, d\omega \, dx \]

\[ = \int_M \int_{\Omega_x} \underbrace{1[x \in A]}_{W_e(x, \omega)} L_i(x, \omega) \langle n_x, \omega \rangle \, d\omega \, dx \]

\[ p_0(x, \omega) = \frac{1[x \in A] \langle n_x, \omega \rangle}{|A|\pi}, \quad \langle I \rangle = |A| \pi L_i(x, \omega) \]
Example: Irradiance Meter

\[ p_0(\mathbf{x}, \omega) = \frac{\mathbf{1}[\mathbf{x} \in A] \langle n_x, \omega \rangle}{|A| \pi}, \quad \langle I \rangle = |A| \pi L_i(\mathbf{x}, \omega) \]

**computel Irradiance()**:

\[
\begin{align*}
\mathbf{x} &= \text{uniformSampleSensor()} \\
\omega &= \text{uniformRandomPSA}(n_x) \\
\text{return } |A| * \pi * \text{receivedRadiance}(\mathbf{x}, \omega)
\end{align*}
\]
Example: Pinhole Camera

\[ I_i = \frac{1}{|A_i|} \int_{A_i} L_i(\mathbf{x}, \mathbf{o} \rightarrow \mathbf{x}) \text{d}\mathbf{x} \]  

(\text{where} \ \mathbf{o} \rightarrow \mathbf{x} := \frac{\mathbf{x} - \mathbf{o}}{||\mathbf{x} - \mathbf{o}||})

\[ = \int_{\mathcal{M}} \int_{\Omega_\omega} \frac{\mathbb{1}[\mathbf{x} \in A_i] \delta_{\mathbf{o} \rightarrow \mathbf{x}}(\omega)}{|A_i| \langle n_x, \omega \rangle} L_i(\mathbf{x}, \omega) \langle n_x, \omega \rangle \text{d}\omega \text{d}\mathbf{x} \]

\[ = W_e(x, \omega) \]

\[ p_0(x, \omega) = \frac{\mathbb{1}[\mathbf{x} \in A_i] \delta_{\mathbf{o} \rightarrow \mathbf{x}}(\omega)}{|A_i|}, \quad \langle I \rangle = L_i(\mathbf{x}, \mathbf{o} \rightarrow \mathbf{x}) \]
Example: Pinhole Camera

\[ p_0(x, \omega) = \frac{\mathbb{1}[x \in A_i] \delta_{o \rightarrow x}(\omega)}{|A_i|}, \quad \langle I \rangle = L_i(x, o \rightarrow x) \]

**computePixelIntensity\((i)\):**
\[
\begin{align*}
    x &= \text{uniformSamplePixel}(i) \\
    \omega &= \text{normalize}(x - o) \\
    \text{return receivedRadiance}(x, \omega)
\end{align*}
\]
Adjoint Operators

• The adjoint of an operator $H$ is denoted $H^*$, and is defined by the property that

$$\langle H^* f, g \rangle = \langle f, H g \rangle \quad \text{for all } f, g$$

• If $H = H^*$, then $H$ is called self-adjoint

• The propagation operator $G$ is self-adjoint
Adjoint Operators

• The local scattering operator $K$ satisfies that

\[
(Kh)(x, \omega) = \int_{\Omega_x} h(x, \omega_i) f_r(x, \omega_i \rightarrow \omega) \langle n_x, \omega_i \rangle \, d\omega_i
\]

\[
(K^*h)(x, \omega) = \int_{\Omega_x} h(x, \omega_o) f_r(x, \omega \rightarrow \omega_o) \langle n_x, \omega_o \rangle \, d\omega_o
\]

• $K$ is self-adjoint if $f_r(x, \omega_i \rightarrow \omega_o) = f_r(x, \omega_o \rightarrow \omega_i)$ for all $\omega_i, \omega_o \in \Omega_x$

• This is the case for all BRDFs with reciprocity (which holds for all BRDFs we have seen up to this point)
Adjoint Operators

• For any operators $A$, $B$ and $C$:
  - $A + B = B + A$
  - $A(BC) = (AB)C$
  - $(A + B)^* = A^* + B^*$, $(A B)^* = B^* A^*$, $(A^{-1})^* = (A^*)^{-1}$

• Based on these properties and the fact that $G$ is self-adjoint, we have

$$
(G(I - KG)^{-1})^* = (I - GK^*)^{-1}G = \left( \sum_{i=0}^{\infty} (GK^*)^i \right) G = \sum_{i=0}^{\infty} (GK^*)^i G
$$

$$
= \sum_{i=0}^{\infty} G(K^*G)^i = G \left( \sum_{i=0}^{\infty} (K^*G)^i \right) = G(I - K^*G)^{-1}
$$
Adjoint Operators

- $G(I - KG)^{-1}$ is self-adjoint when $K$ is self-adjoint

Therefore,

\[
\langle W_e, L_i \rangle = \langle W_e, GL \rangle = \langle W_e, G(I - KG)^{-1}L_e \rangle = \langle G(I - K^*G)^{-1}W_e, L_e \rangle
\]

- $(I - KG)^{-1} L_e$ gives the radiance $L$ (as a function on the ray space) satisfying the RE: $L = KGL + L_e$

- $(I - K^*G)^{-1} W_e$ gives another function $W : \mathcal{R} \rightarrow \mathbb{R}$ satisfying $W = K^*GW + W_e$
The Importance Transport

• $W$ is called the **importance** function
  
  • Computing $W$ requires solving the *important transport* problem given by $W = K^*GW + W_e$
  
  • The sensor acts as the light source in the light transport problem by *emitting* importance $W_e$
  
  • Any algorithm that estimates $L$ for the original light transport problem (e.g., path tracing) can be adapted to estimate $W$
Light vs. Importance Transport

Light transport

• Transport equation
  \[ L = KGL + L_e \]

• Central quantity
  \( L \) (radiance)

• Source term
  \( L_e \) (radiance emitted by light sources)

• Measurements
  \[ I = \langle W_e, GL \rangle = \langle W_e, L_i \rangle \]

Importance transport

• Transport equation
  \[ W = K*GW + W_e \]

• Central quantity
  \( W \) (importance)

• Source term
  \( W_e \) (importance “emitted” by sensors)

• Measurements
  \[ I = \langle GW, L_e \rangle = \langle W_i, L_e \rangle \]

Same results!
Adjoint Particle Tracing

• **Goal:** estimating

\[ I = \langle W_i, L_e \rangle = \int_M \int_{\Omega_x} W_i(x, \omega) L_e(x, \omega) \langle n_x, \omega \rangle \, d\omega \, dx \]

:= GW

• **Initial step:**
  • Draw \( x \) and \( \omega \) from some probability density \( p_0^* \propto L_e \)
  or \( p_0^*(x, \omega) \propto L_e(x, \omega) \langle n_x, \omega \rangle \)
  • Then,

\[ \langle I \rangle = \frac{L_e(x, \omega) \langle n_x, \omega \rangle}{p_0^*(x, \omega)} W_i(x, \omega) \]

Known after drawing \( x \) and \( \omega \)
Example: Sampling Area Lights

Area light emitting radiance $L_0$ into all directions

\[
I = \int_A \int_{\Omega_x} L_0 W_i(x, \omega) \langle n_x, \omega \rangle \, d\omega \, dx
\]

\[
= \int_M \int_{\Omega_x} L_0 \mathbb{1}[x \in A] W_i(x, \omega) \langle n_x, \omega \rangle \, d\omega \, dx
\]

\[
= L_e(x, \omega)
\]

\[
p^*_0(x, \omega) = \frac{\mathbb{1}[x \in A] \langle n_x, \omega \rangle}{|A| \pi}, \quad \langle I \rangle = |A| L_0 \pi W_i(x, \omega)
\]
Example: Sampling Area Lights

Area light emitting radiance $L_0$ into all directions

$$p_0^* (x, \omega) = \frac{\mathbb{1} [x \in A] \langle n_x, \omega \rangle}{|A| \pi},$$

$$\langle I \rangle = |A| \ L_0 \ \pi \ W_i(x, \omega)$$

$computeMeasurement()$: 
- $x = uniformSampleLight()$
- $\omega = uniformRandomPSA(n_x)$

return $|A| * L_0 * \pi * receivedImportance(x, \omega)$

Similar to irradiance meters in the light transport problem
Adjoint Particle Tracing

• **Main step**: estimating $W(x, \omega)$
  
  • This can be done using an algorithm equivalent to path tracing since

  $$W(x, \omega) = W_e(x, \omega) + \int_{\Omega_x} W_i(x, \omega_o) f_r(x, \omega \rightarrow \omega_o) \langle n_x, \omega_o \rangle \, d\omega_o$$

  $$=: W_r(x, \omega)$$

• We call this algorithm *adjoint particle tracing* to emphasize that it solves the adjoint (importance transfer) problem
Path vs. Adjoint Particle Tracing

# Path tracing (version 0.1)
```python
radiance(x, \omega):
    rad = emittedRadiance(x, \omega)
    \omega_i = uniformRandomPSA(n_x)
    y = RayTrace(x, \omega_i)
    rad += \pi * radiance(y, -\omega_i) * brdf(x, \omega_i, \omega)
return rad
```

# Adjoint particle tracing (version 0.1)
```python
importance(x, \omega):
    imp = emittedImportance(x, \omega)
    \omega_o = uniformRandomPSA(n_x)
    y = RayTrace(x, \omega_o)
    imp += \pi * importance(y, -\omega_o) * brdf(x, \omega, \omega_o)
return imp
```
Next-Event Estimation

• Similar to path tracing, *next-event estimation* can drastically improve the convergence rate

• Same idea: separating direct & indirect

Path tracing:

\[ L_r(x, \omega) = L_r^{\text{direct}}(x, \omega) + L_r^{\text{indirect}}(x, \omega) \]

Adjoint particle tracing:

\[ W_r(x, \omega) = W_r^{\text{direct}}(x, \omega) + W_r^{\text{indirect}}(x, \omega) \]
Direct and Indirect Importance

\[ W_r^{\text{direct}}(x, \omega) = \int_{\Omega_x} W_e(y, -\omega_o) f_r(x, \omega \rightarrow \omega_o) \langle n_x, \omega_o \rangle \, d\omega_o \]

\[ W_r^{\text{indirect}}(x, \omega) = \int_{\Omega_x} W_r(y, -\omega_o) f_r(x, \omega \rightarrow \omega_o) \langle n_x, \omega_o \rangle \, d\omega_o \]
Evaluating Direct Importance

- Sensor sampling

\[ W^\text{direct}_r(x, \omega) = \int_A W^\text{direct}_i(x, \omega_o) f_r(x, \omega \rightarrow \omega_o) G(x \leftrightarrow y) \, dy \]

\[ \omega_o = x \rightarrow y \]

\[ \langle W^\text{direct}_r(x, \omega) \rangle_{\text{sensor}} = \frac{W^\text{direct}_i(x, \omega_o) f_r(x, \omega \rightarrow \omega_o) G(x \leftrightarrow y)}{p_{\text{sensor}}(y)} \]

- \( p_{\text{sensor}} \) is usually picked as the uniform distribution over the surface \( A \) of the sensor. Namely,

\[ p_{\text{sensor}}(y) \equiv \frac{1}{|A|} \quad \text{for all } y \in A \]
Evaluating Direct Importance

• BRDF sampling

\[ W_{r}^{\text{direct}}(x, \omega) = \int_{\Omega_x} W_{i}^{\text{direct}}(x, \omega_o) f_r(x, \omega \rightarrow \omega_o) \langle n_x, \omega_o \rangle \, d\omega_o \]

\[ \langle W_{r}^{\text{direct}}(x, \omega) \rangle_{\text{BRDF}} = \frac{W_{i}^{\text{direct}}(x, \omega_o) f_r(x, \omega \rightarrow \omega_o) \langle n_x, \omega_o \rangle}{p_{\text{BRDF}}(\omega_o)} \]

• Desirable to have

\[ p_{\text{BRDF}}(\omega_o) \propto f_r(x, \omega \rightarrow \omega_o) \langle n_x, \omega_o \rangle \]

or

\[ p_{\text{BRDF}}(\omega_o) \propto f_r(x, \omega \rightarrow \omega_o) \]
Evaluating Direct Importance

• Similar to evaluating direct illumination, the two sampling methods can be combined using multiple importance sampling (MIS)

\[
\langle W_r^{\text{direct}}(x, \omega) \rangle_{\text{balance}} = \frac{W_i^{\text{direct}}(x, \omega_{o1}) f_r(x, \omega \rightarrow \omega_{o1}) \langle n_x, \omega_{o1} \rangle}{p_{\text{sensor}}(\omega_{o1}) + p_{\text{BRDF}}(\omega_{o1})} + \frac{W_i^{\text{direct}}(x, \omega_{o2}) f_r(x, \omega \rightarrow \omega_{o2}) \langle n_x, \omega_{o2} \rangle}{p_{\text{sensor}}(\omega_{o2}) + p_{\text{BRDF}}(\omega_{o2})}
\]
Path vs. Adjoint Particle Tracing

• Since $I = \langle W_e, L_i \rangle = \langle W_i, L_e \rangle$, both algorithms lead to the same answer

• Which one to use depends on the light source and measurement functions (namely, $L_e$ and $W_e$)
Challenging Light Sources

• Collimated beam

\[ L_e(x, \omega) = \mathbb{1}[x \in A] \delta_{\omega_e}(\omega) \]

• Problem:
  • Neither light source nor BRDF sampling can give a point \( y \) on \( A \) with \((y \rightarrow x)\) precisely aligned with \( \omega_e \)
  • Can never hit \( A \) precisely from direction \(-\omega_e\)
Challenging Light Sources

- Collimated beam

\[ L_e(x, \omega) = \mathbb{1}[x \in A] \delta_{\omega_e}(\omega) \]

- In this case, we should solve the adjoint problem instead

\[ p_0^*(x, \omega) = \frac{\mathbb{1}[x \in A] \delta_{\omega_e}(\omega)}{|A|}, \]

\[ \langle I \rangle = |A| W_i(x, \omega_e) \langle n_x, \omega_e \rangle \]
Next Lecture

• Path integral formulation