Computer graphics III – Bidirectional path tracing

Jaroslav Křivánek, MFF UK

Jaroslav.Krivanek@mff.cuni.cz

“Science, it works …

(bitches!)”

Quote from Richard Dawkins

http://www.youtube.com/watch?v=n6hxo1sC-dU

… and so does path tracing!

corona

Jerome White

Yes, it does work!

Yes, it does work!


corona
F509682A
Ondra Karlík

Light transport – Global illumination

Archviz

Movies

Light transport – Global illumination

- **More information**
  - "The State of Rendering"

Measurement equation

Measurement equation

- Rendering equation enables evaluating radiance at isolated points in the scene
- But in fact, we are interested in **average radiance** over a pixel: an **integral**, again?!
- Yes, it's called the **Measurement equation**

Measurement equation

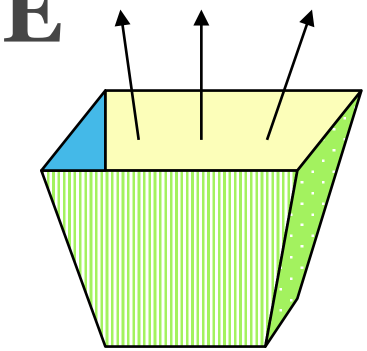
Response of a virtual linear sensor to light (most commonly the **pixel color**).

Relative response (weight). Each sensor (pixel) has a different W_e function.

$$I = \int_M \int_{H(\mathbf{x})} W_e(\mathbf{x}, \omega) \cdot L_i(\mathbf{x}, \omega) \cdot \cos\theta \, d\omega \, dA$$

Integrate over the entire scene surface.
(We assume that the virtual sensor is a part of the scene. The response is non-zero only on the sensor area because W_e is zero elsewhere.)

Example measurement: Radiant flux over a region formulated as a ME



- Given a region S in ray space

$$S \subset M \times H$$

(a subset of the Cartesian product of the scene surfaces and directions)

- For W_e defined as

$$W_e(x,\omega) = \begin{cases} 1 & \text{for } (x,\omega) \in S \\ 0 & \text{otherwise} \end{cases}$$

the result of the measurement equation is the **radiant flux Φ(S)**.

Measurement equation as a scalar product of functions

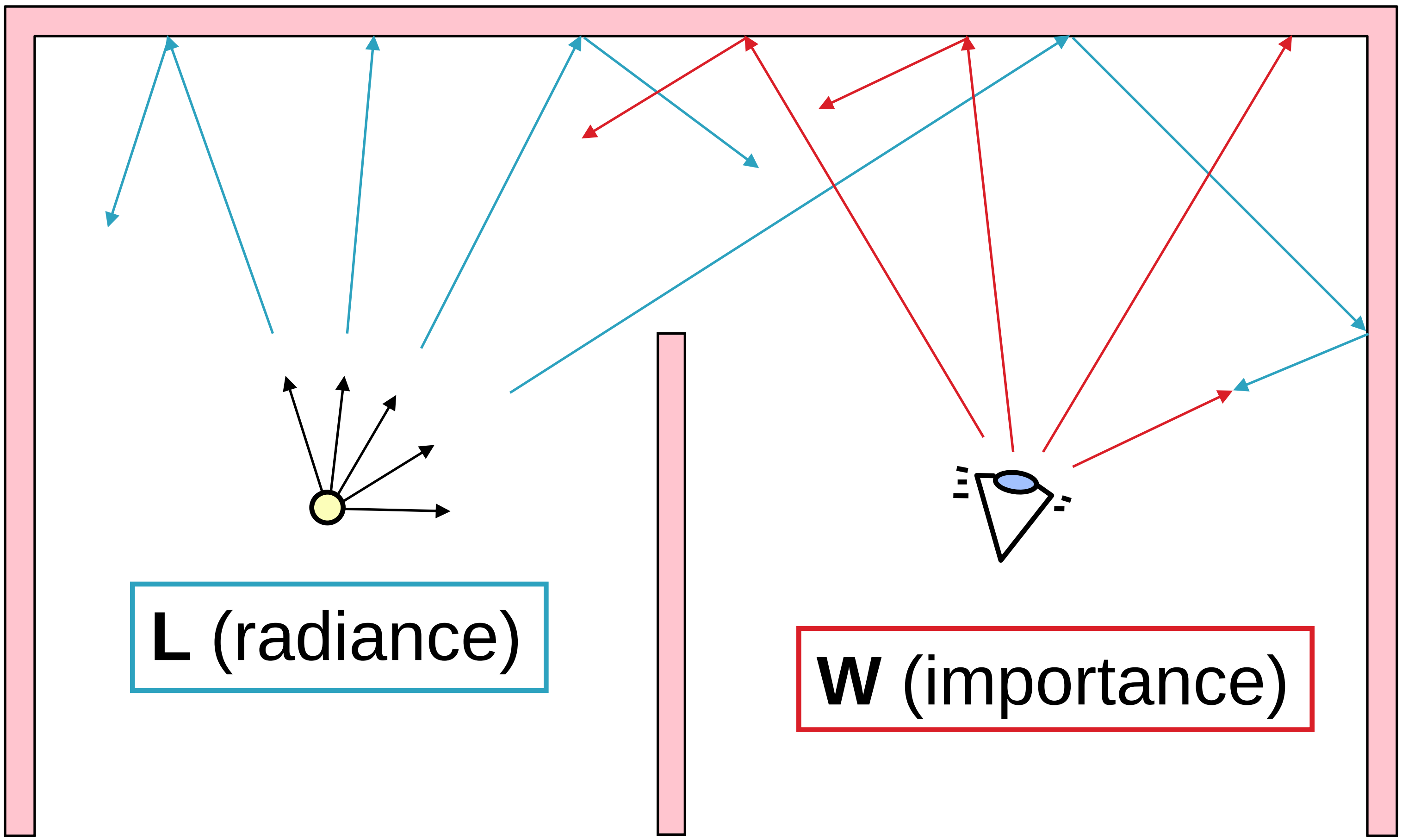
- Let us define a **scalar product** of function ***f*** **and *g*** as:

$$\langle f, g \rangle = \int_M \int_{H(\mathbf{x})} f(\mathbf{x}, \omega)\, g(\mathbf{x}, \omega) \cos\theta \, \mathrm{d}\omega \, \mathrm{d}A$$

- The **Measurement equation** can now be written as

$$I = \langle W_\mathrm{e}, L_\mathrm{i} \rangle$$

Transport of radiance and visual importance



Visual importance

- W_e describes how important is the incident radiance to the sensor response
- One step into the scene: Incident radiance on the sensor = outgoing radiance from other scene points
- And we can go on to 2, 3, … steps into the scene…

- As a result, W_e can be interpreted as an (imaginary) transport quantity emitted from the sensor (similarly to how radiance L_e is emitted from light sources)
- In this interpretation, we call W_e the **emitted importance function**

Transport of visual importance

- The importance function is transported by the similar rules to radiance and settles down on an **equilibrium (steady state)** given by the **equilibrium visual importance function *W***:

$$W(\mathbf{x},\omega_{\mathrm{o}}) = W_{\mathrm{e}}(\mathbf{x},\omega_{\mathrm{o}}) + \int_{H(\mathbf{x})} W(\mathrm{r}(\mathbf{x},\omega_{\mathrm{i}}),-\omega_{\mathrm{i}}) \cdot \underline{f_r(\mathbf{x},\omega_{\mathrm{o}} \to \omega_{\mathrm{i}})} \cdot \cos\theta_{\mathrm{i}} \, \mathrm{d}\omega_{\mathrm{i}}$$

As in the rendering equation except that the BRDF arguments are exchanged (No difference for reflection because the BRDF is symmetrical, but it makes difference for transmission, which is in general not symmetrical.)

Duality of importance and radiance

Emitted importance

Equilibrium incident radiance

$$I = \langle W_e , L_i \rangle = \langle W_i , L_e \rangle$$

Equilibrium incident importance

Emitted radiance

Duality of importance and radiance – proof

r stands for (**x**,ω)

The proof of Eq. (9), i.e., $I = \langle W^{\mathrm{e}}, L\rangle = \langle L^{\mathrm{e}}, W\rangle$ given here follows [Kalos and Whitlock 2008]. We can write $Q = \int_\Omega L(\mathbf{r})\, W(\mathbf{r})\, d\mathbf{r}$ in two possible ways, either by expanding $L(\mathbf{r})$ using the radiation transport equation (1) or by expanding $W(\mathbf{r})$ using the importance transport equation (8):

$$Q = \int_\Omega L^{\mathrm{e}}(\mathbf{r})\, W(\mathbf{r})\, d\mathbf{r} + \int_\Omega \int_\Omega L(\mathbf{r}')\, T(\mathbf{r}' \to \mathbf{r})\, W(\mathbf{r})\, d\mathbf{r}'\, d\mathbf{r},$$

$$Q = \int_\Omega L(\mathbf{r})\, W^{\mathrm{e}}(\mathbf{r})\, d\mathbf{r} + \int_\Omega \int_\Omega L(\mathbf{r})\, T(\mathbf{r} \to \mathbf{r}')\, W(\mathbf{r}')\, d\mathbf{r}'\, d\mathbf{r}.$$

We can now swap $\mathbf{r}$ and $\mathbf{r}'$ in one of the double integrals on the r.h.s. to see that they are in fact equal. This immediately yields the desired result.

Duality of importance and radiance

- In a given scene, there is only one emitted and equilibrium radiance function
- But **each pixel has its own emitted and equilibrium visual importance function**

Duality in practice: Light tracing

- Path tracing recursively solves the rendering equation

- Similarly, **light tracing** recursively solves the importance transport equation

  - Light paths start at the light sources and are traced into the scene using exactly the same rules as photons in photon mapping

  - They may either hit the sensor by chance (for a finite aperture camera) or we can explicitly connect vertices to the sensor (as in explicit light source sampling in PT)

Light tracing

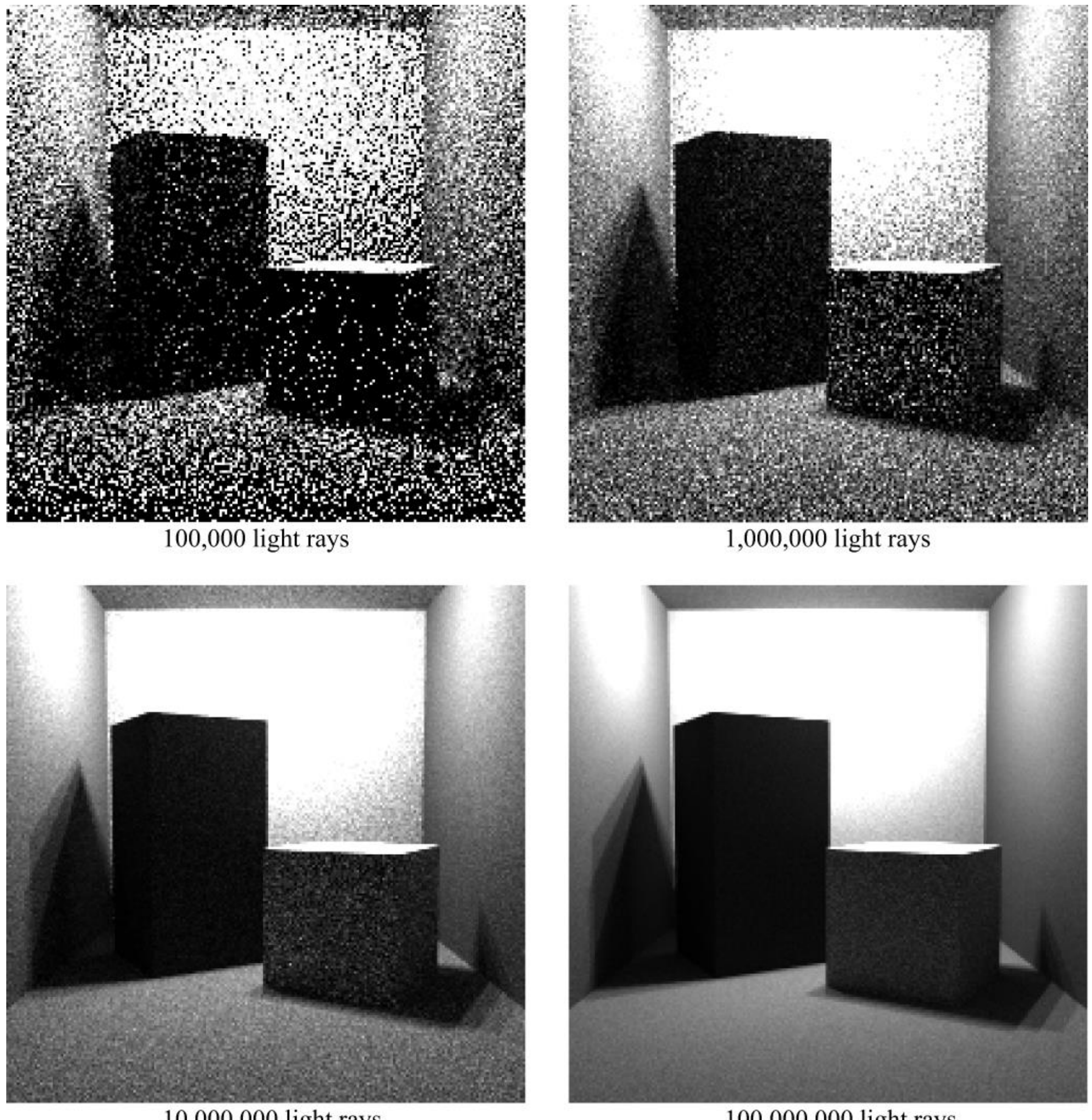
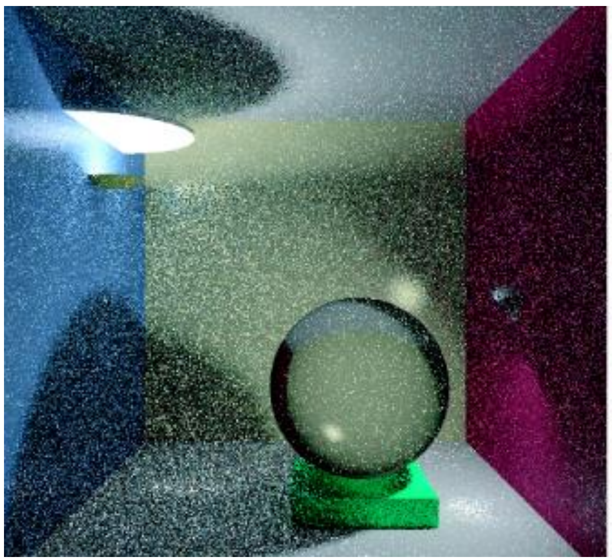


Image: Dutre et al. Advanced Global Illumination

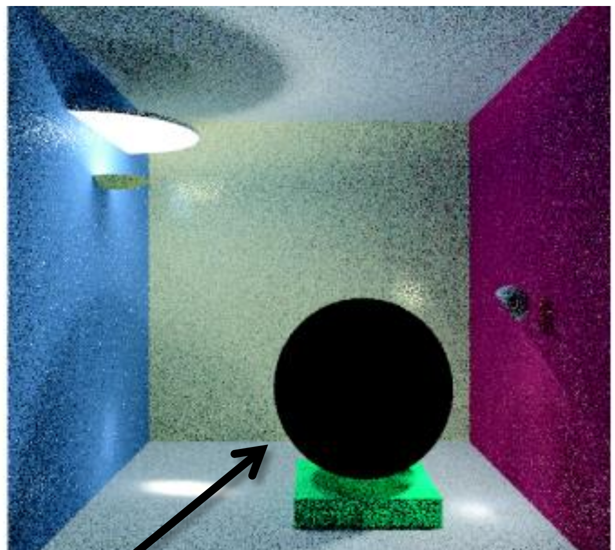
Light tracing in practice

- Generally less efficient than PT
- But it certain case, it may be much better. One example are **caustics.**
- Light tracing and path tracing are the basis of bidirectional methods, such as
  - Bidirectional path tracing, BPT
  - Photon mapping, etc.

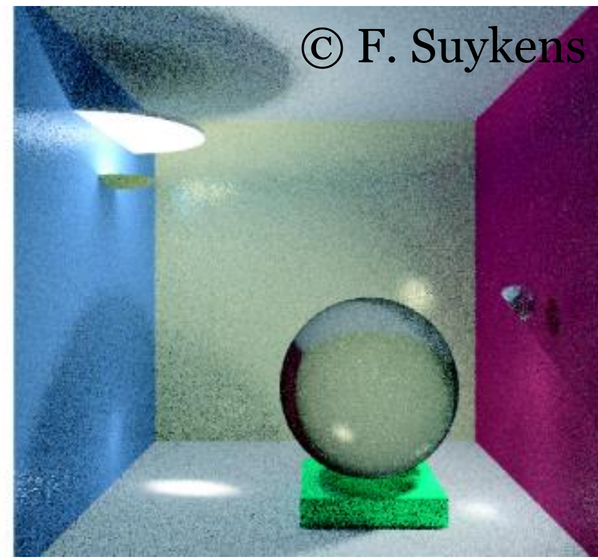
Comparison



Path tracing



Light tracing



Bidirectional path tracing

Q: Why is the glass sphere entirely black?

Advanced light transport simulation methods

Main issue in light transport simulation

- **Robustness**
  - None of the existing algorithms works for all scenes
  - Robust estimation

    "An estimation technique which is insensitive to small departures from the idealized assumptions which have been used to optimize the algorithm."

Wolfram MathWorld
the web's most extensive mathematics resource

Bidirectional path tracing (BPT) vs. (unidirectional) path tracing (PT)

BPT, 25 path per pixel

PT, 56 path per pixel

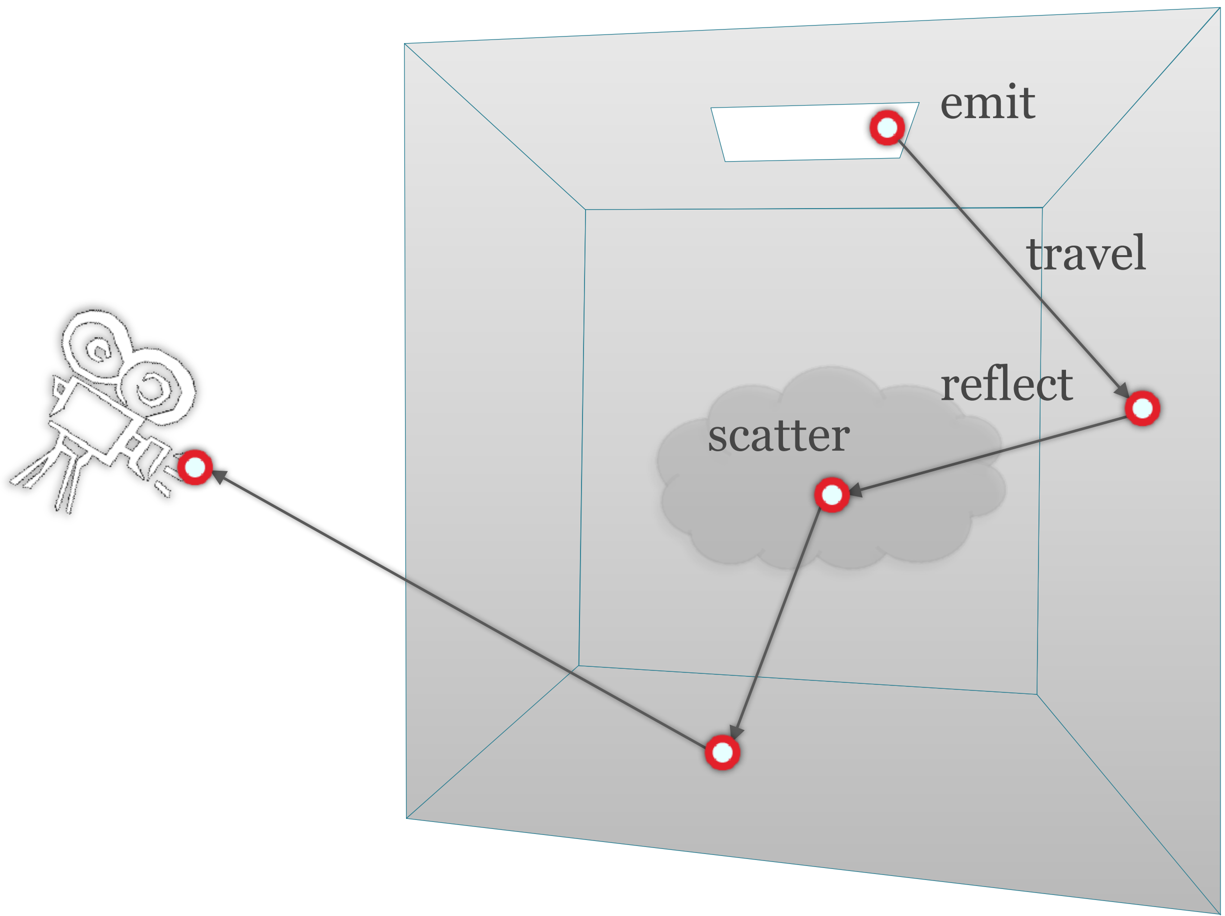
Image: Eric Veach

Path integral formulation of light transport

Light transport expressed as an integral over the space o light transport paths

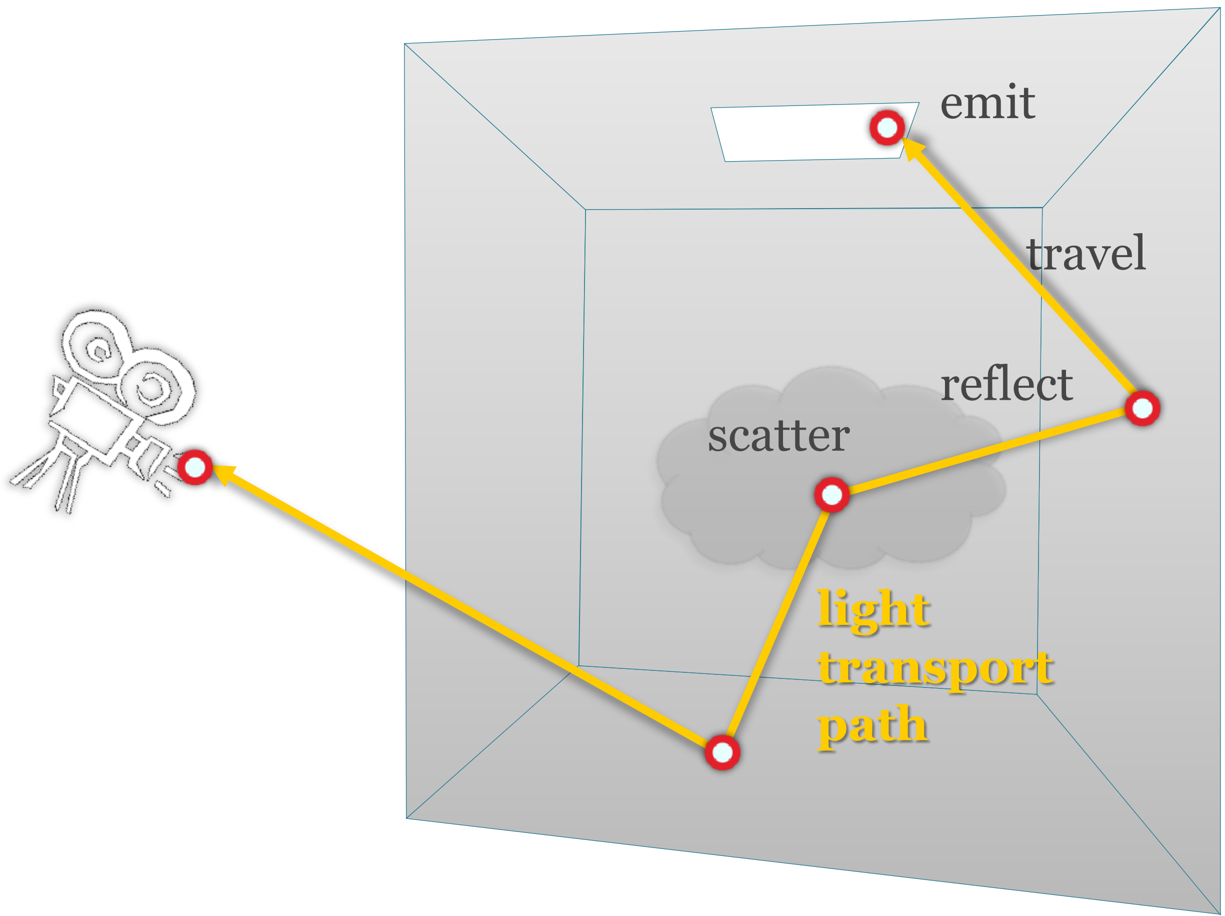
Light transport

- Geometric optics



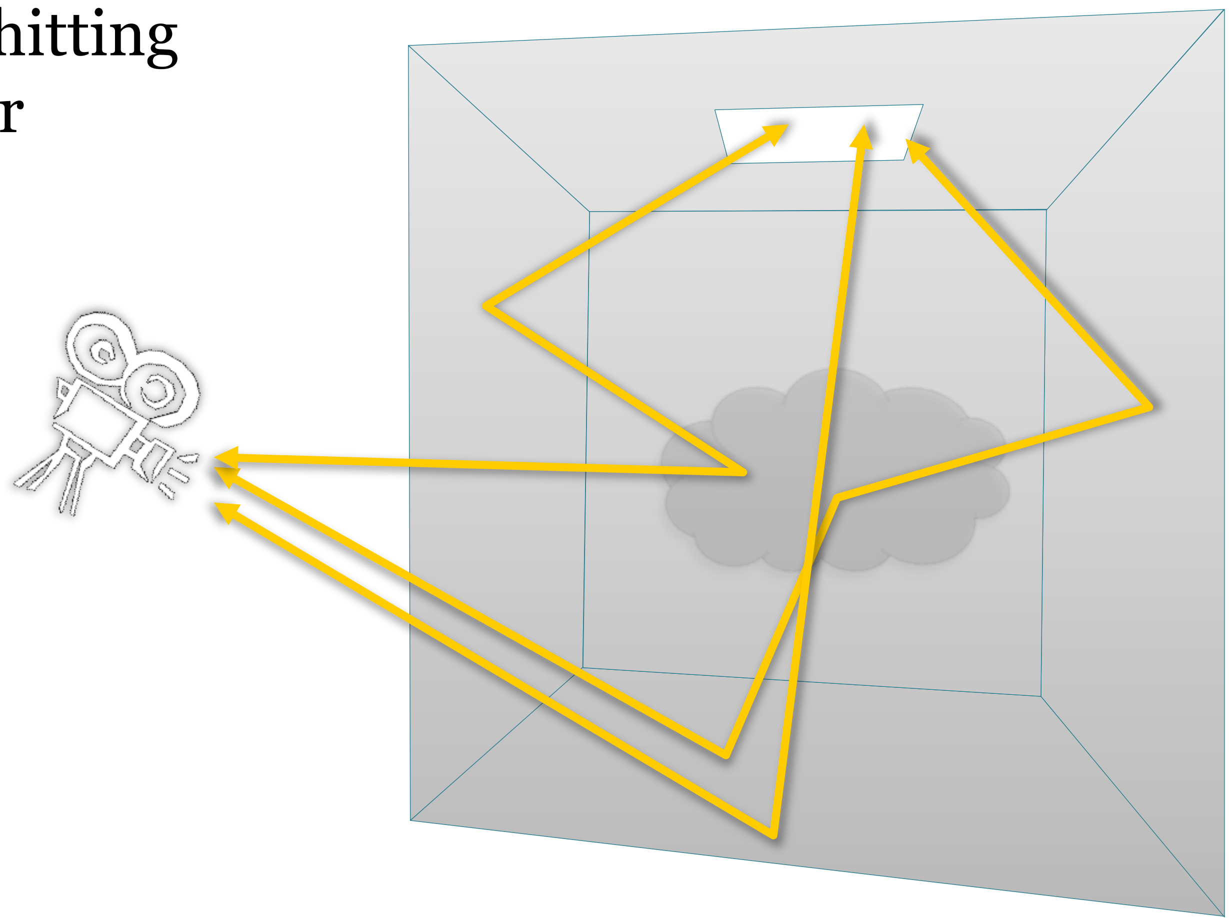
Light transport

- Geometric optics



Light transport

- **Camera response**
  - all paths hitting the sensor



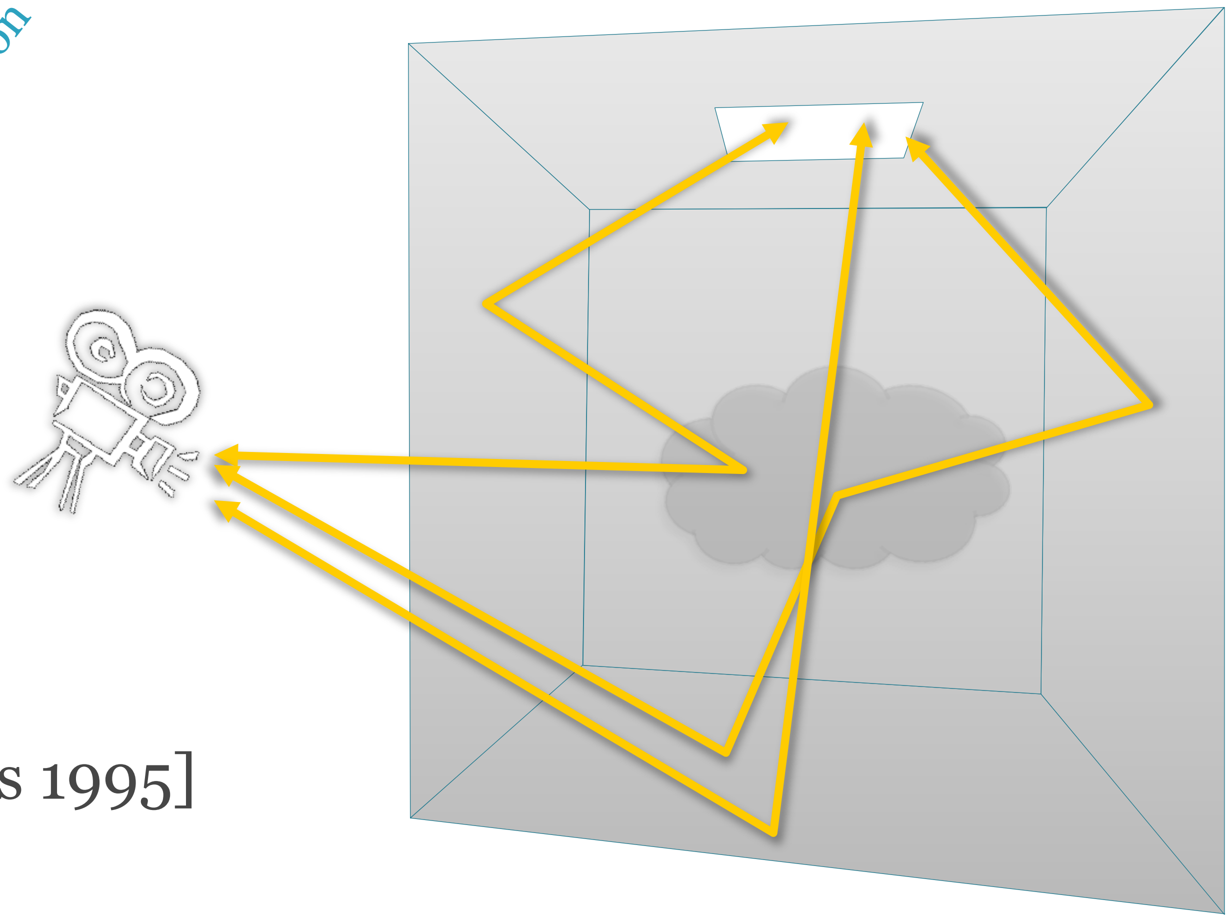
Path integral formulation

$$I_j = \int_\Omega f_j(\bar{x}) \, \mathrm{d}\mu(\bar{x})$$

camera resp. (j-th pixel value)

all paths

measurement contribution function



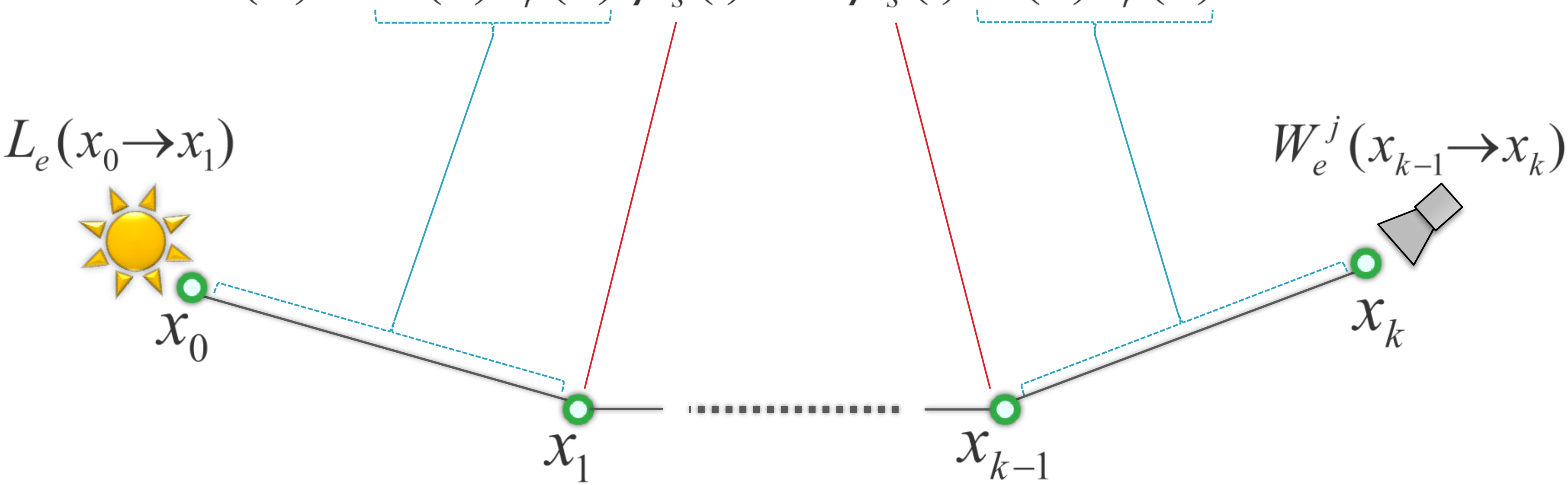
[Veach and Guibas 1995]
[Veach 1997]

Measurement contribution function

$$\bar{x} = x_0 x_1 \ldots x_k$$

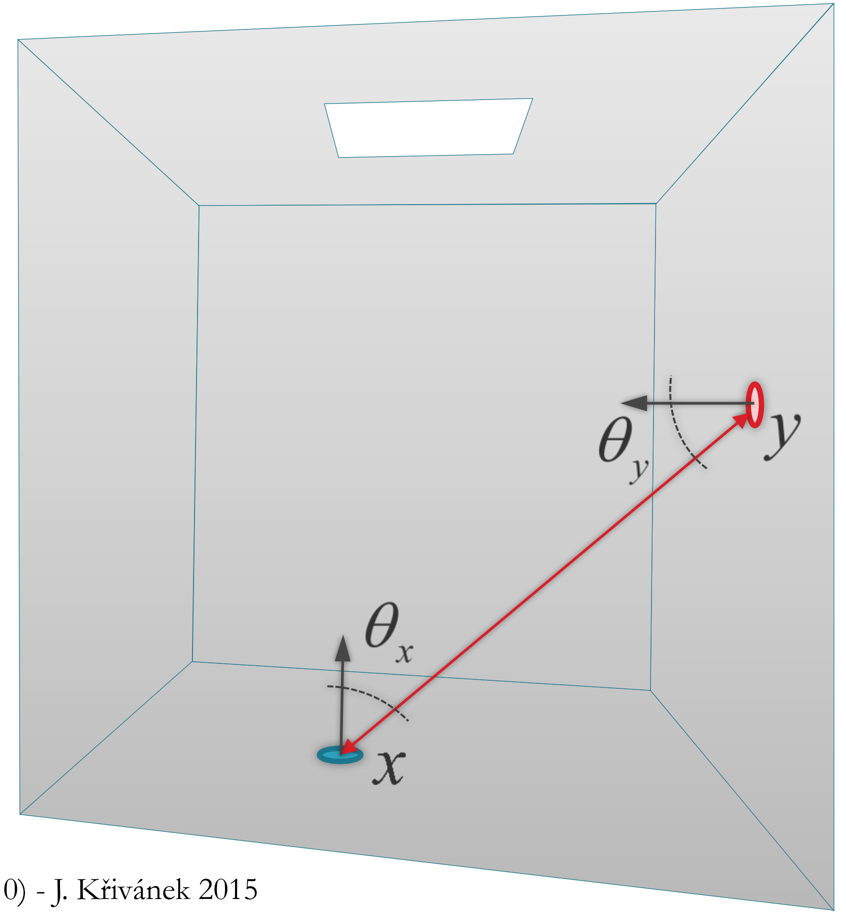
$$f_j(\bar{x}) = \underbrace{L_e(x_0 \to x_1)}_{\text{emitted radiance}} \; \underbrace{T(\bar{x})}_{\text{path throughput}} \; \underbrace{W_e^j(x_{k-1} \to x_k)}_{\text{sensor sensitivity (“emitted importance”)}}$$

$$T(\bar{x}) = G(-)T_r(-)\;\rho_s(\cdot)\;\ldots\;\rho_s(\cdot)\;G(-)T_r(-)$$



Geometry term

$$G(x \leftrightarrow y) = \frac{|\cos\theta_x||\cos\theta_y|}{\left\|x-y\right\|^2} V(x \leftrightarrow y)$$



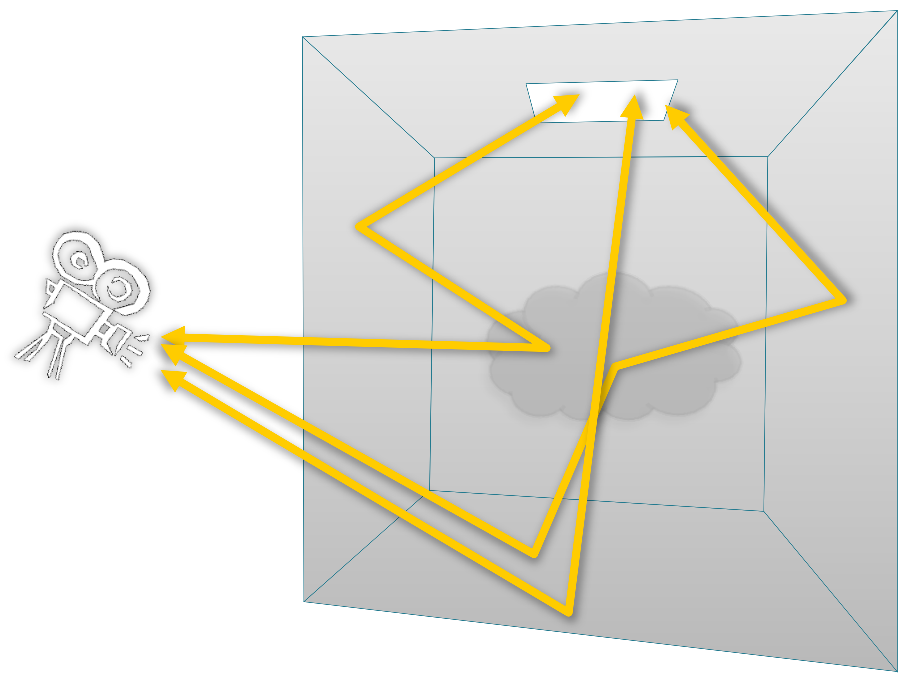
Path integral formulation

$$I_j = \int_\Omega f_j(\bar{x}) \, \mathrm{d}\mu(\bar{x})$$

camera resp. (j-th pixel value) ?

all paths

measurement contribution function ✓



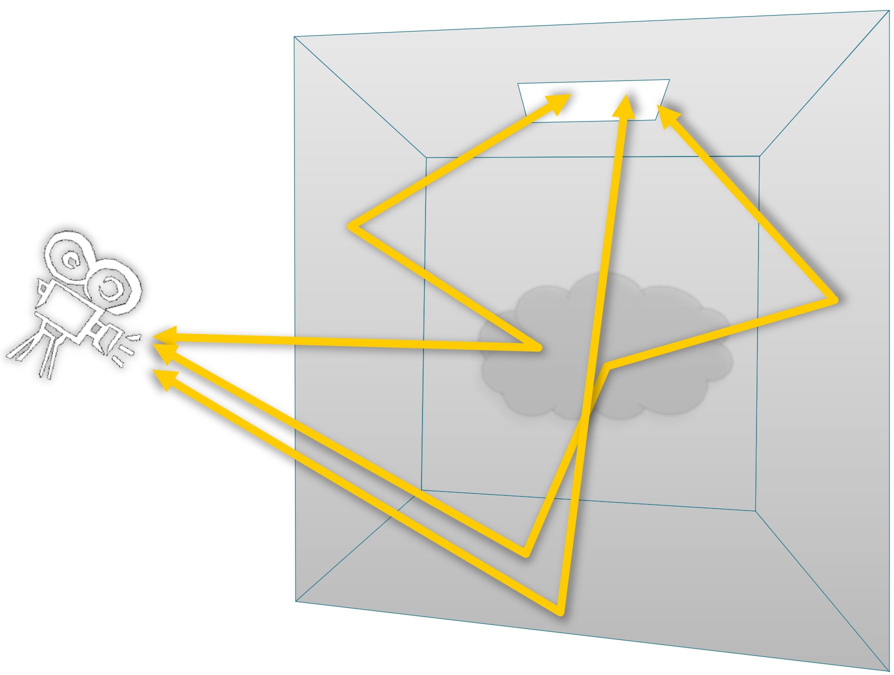
Path integral formulation

$$I_j = \int_\Omega f_j(\bar{x}) \; \mathrm{d}\mu(\bar{x})$$

$$= \sum_{k=1}^{\infty} \int_{M^{k+1}} f_j(x_0 \ldots x_k) \; \mathrm{d}A(x_0) \ldots \mathrm{d}A(x_k)$$

all path lengths

all possible vertex positions



Path integral

$$I_j = \int_\Omega f_j(\bar{x})\, \mathrm{d}\mu(\bar{x})$$

pixel value

all paths

contribution function

Rendering :

Evaluating the path integral

Path integral

$$I_j = \int_{\Omega} f_j(\bar{x})\, \mathrm{d}\mu(\bar{x})$$

pixel value

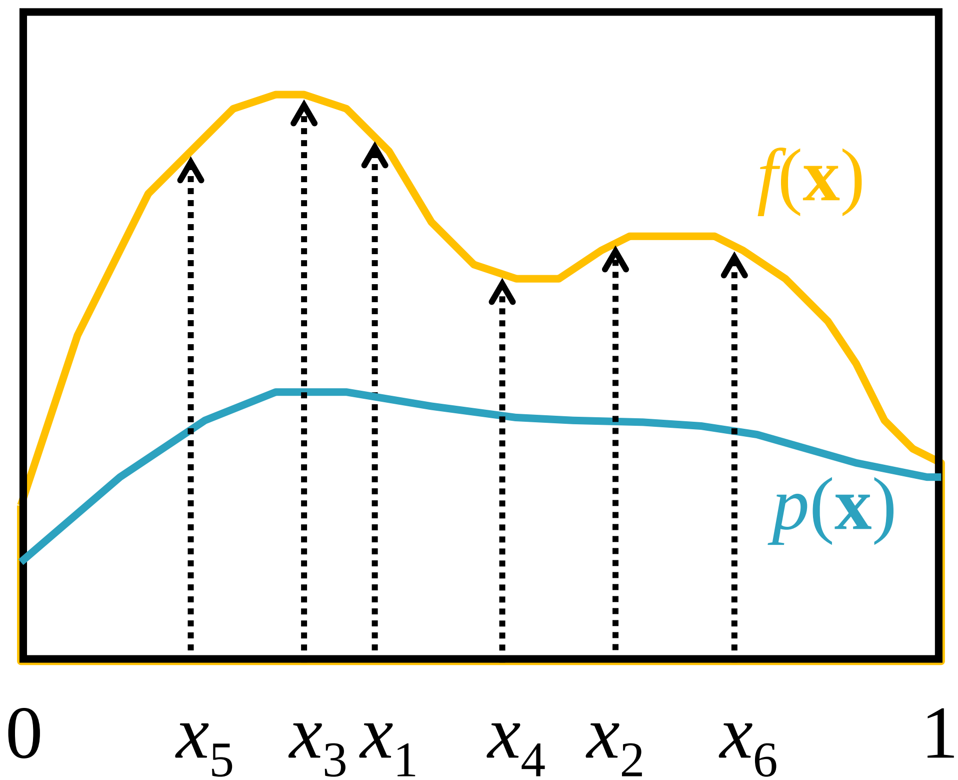
all paths

contribution function

- **Monte Carlo integration**

Monte Carlo integration

- General approach to numerical evaluation of integrals



Integral:

$$I = \int f(x)\,\mathrm{d}x$$

Monte Carlo estimate of I:

$$\langle I \rangle = \frac{1}{N} \sum_{i=1}^{N} \frac{f(x_i)}{p(x_i)}; \quad x_i \propto p(x)$$

Correct „on average“:

$$E[\langle I \rangle] = I$$

MC evaluation of the path integral

Path integral

$$I_j = \int_\Omega f_j(\bar{x})\,\mathrm{d}\mu(\bar{x})$$

MC estimator

$$\langle I_j \rangle = \frac{f_j(\bar{x})}{p(\bar{x})}$$

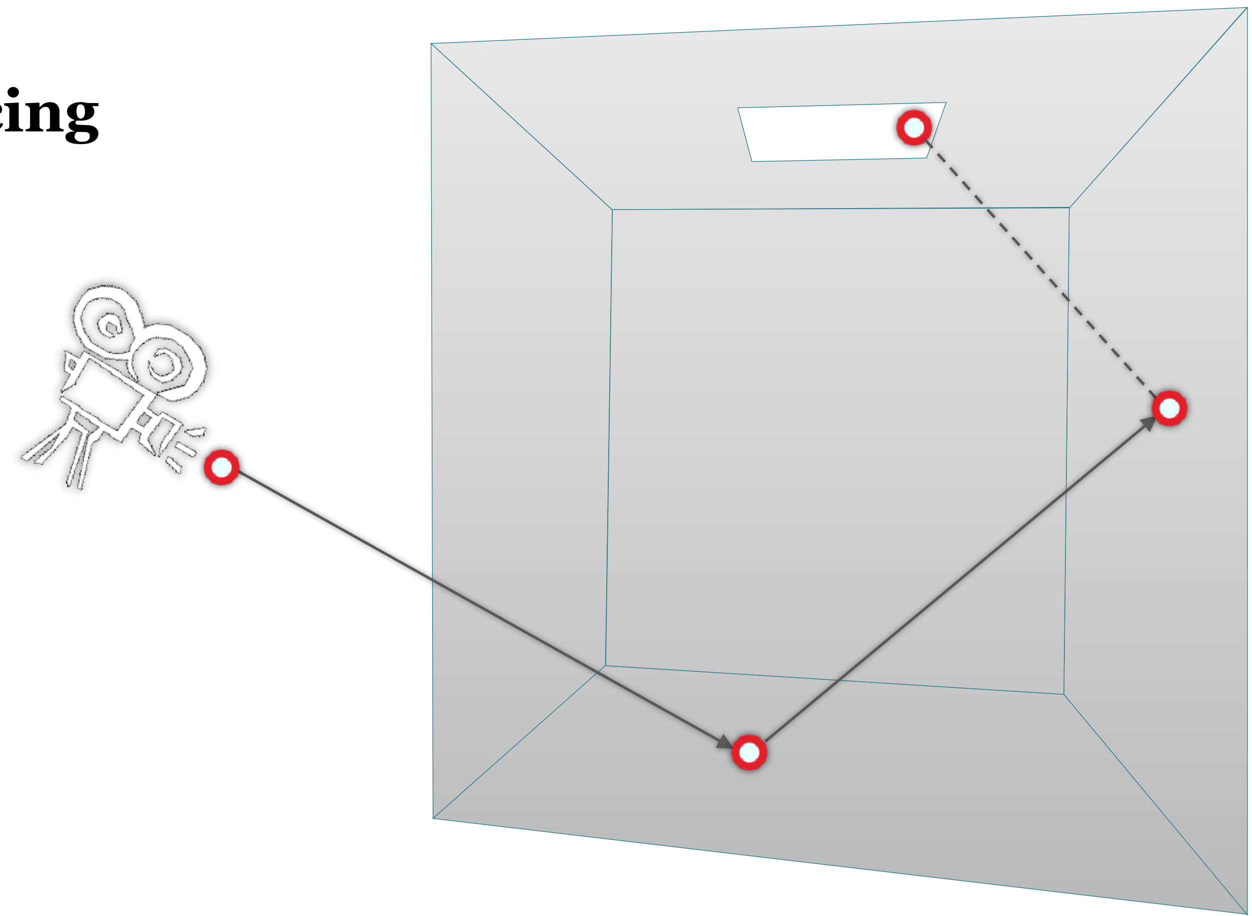
- Sample path $\bar{x}$ from some distribution with PDF $p(\bar{x})$?
- Evaluate the probability density $p(\bar{x})$?
- Evaluate the integrand $f_j(\bar{x})$ ✓

Path sampling

- Algorithms = different path sampling techniques

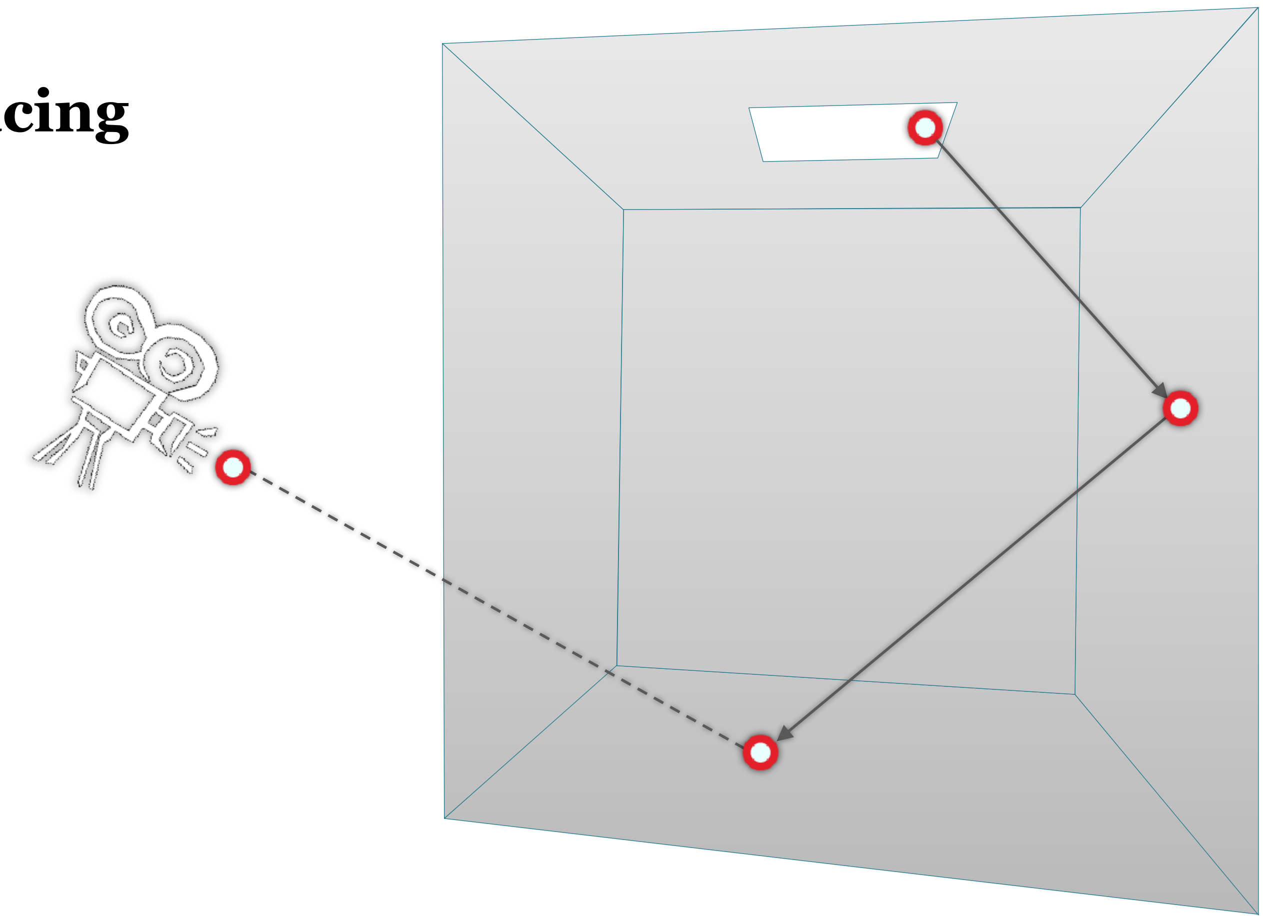
Path sampling

- Algorithms = different path sampling techniques
  - **Path tracing**



Path sampling

- Algorithms = different path sampling techniques
  - **Light tracing**



Path sampling

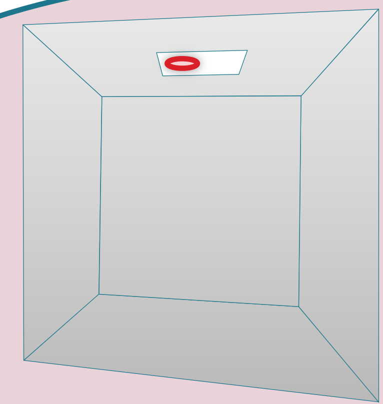
- Algorithms = different path sampling techniques

- **Same** general form of **estimator**

$$\langle I_j \rangle = \frac{f_j(\bar{x})}{p(\bar{x})}$$

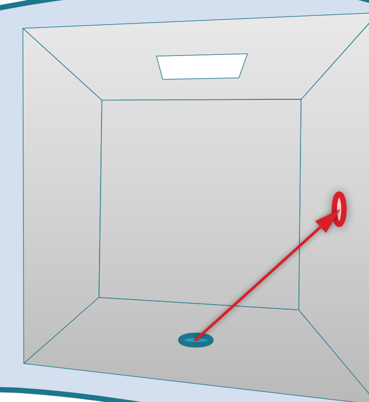
Path sampling
&
Path PDF

Local path sampling

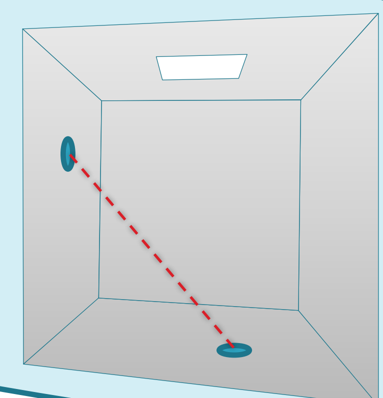


- Sample one path vertex at a time

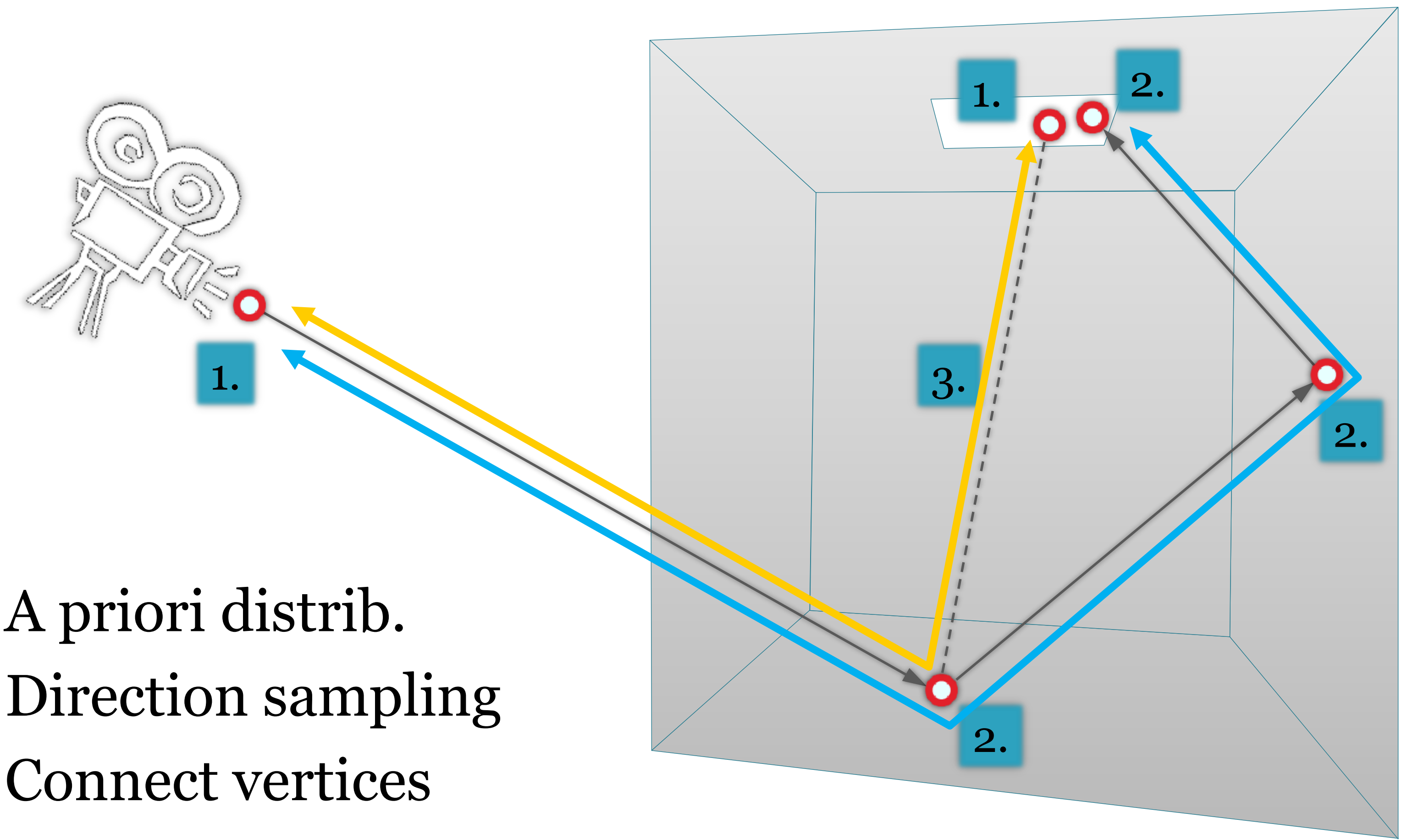
1. From an a priori distribution
   - lights, camera sensors



2. Sample direction from an existing vertex
3. Connect sub-paths
   - test visibility between vertices



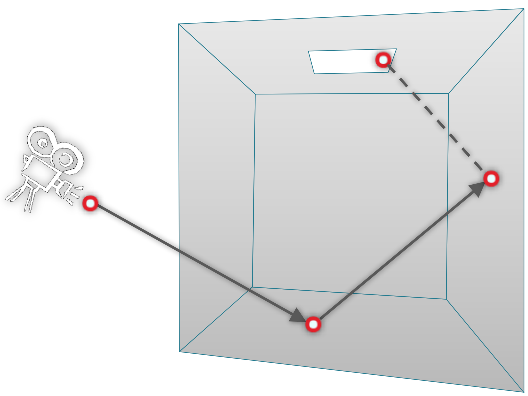
Example – Path tracing



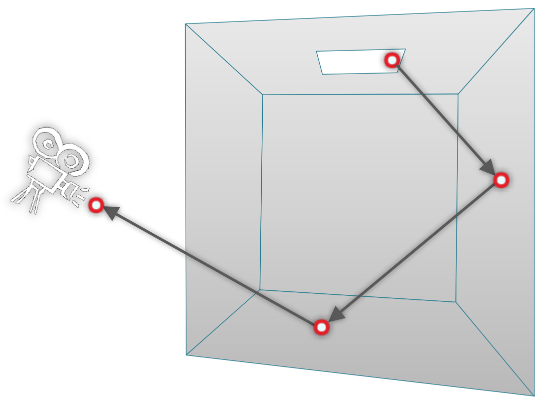
1. A priori distrib.
2. Direction sampling
3. Connect vertices

Use of local path sampling

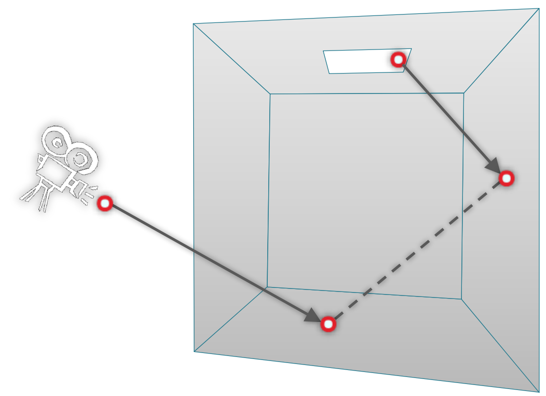
Path tracing



Light tracing



Bidirectional path tracing

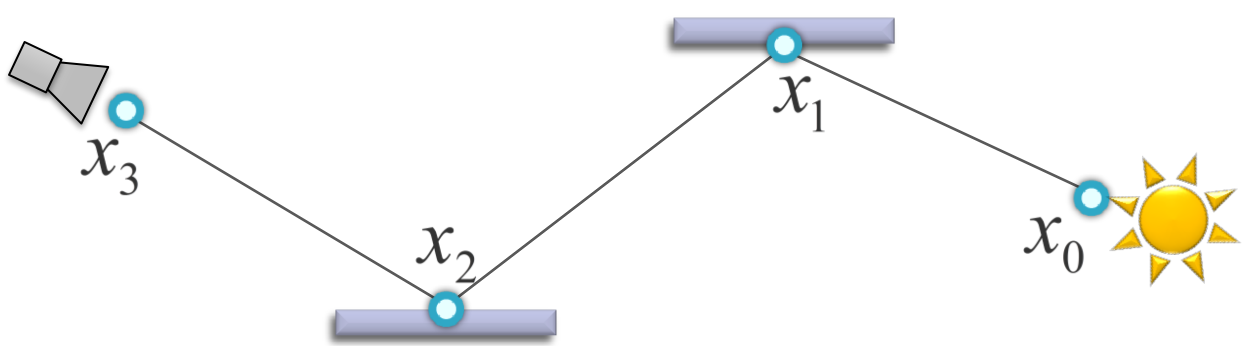


Probability density function (PDF)

path PDF

$$p(\bar{x}) = p(x_0, \ldots, x_k)$$

joint PDF of path vertices

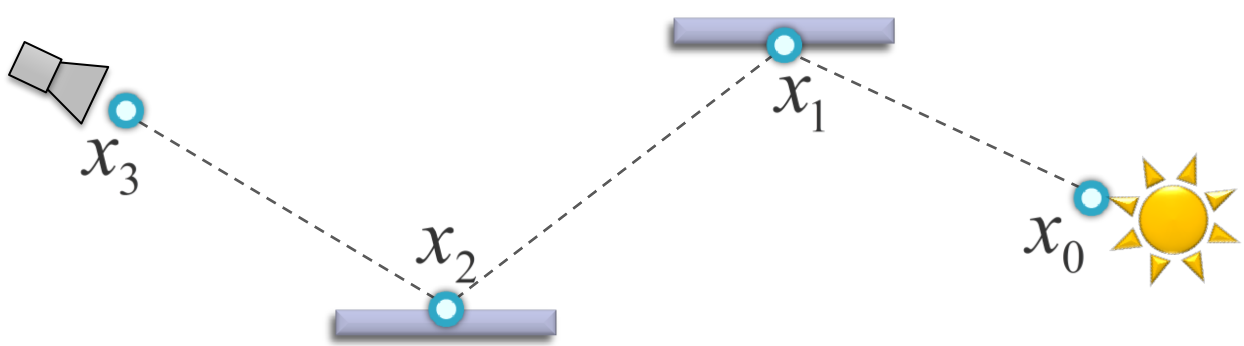


Probability density function (PDF)

path PDF

$$\underline{p(\bar{x})} = \underline{p(x_0, \ldots, x_k)}$$

joint PDF of path vertices



Probability density function (PDF)

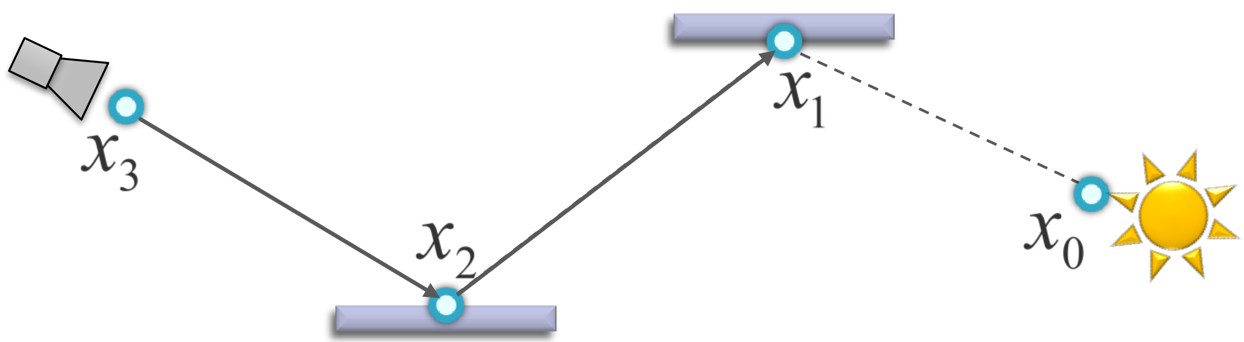
path PDF

$$p(\bar{x}) \quad = \quad p(x_0, \ldots, x_k) \quad = \quad p(x_3)\, p(x_2 \mid x_3)\, p(x_1 \mid x_2)\, p(x_0)$$

joint PDF of path vertices

product of (conditional) vertex PDFs

Path tracing example:



Probability density function (PDF)

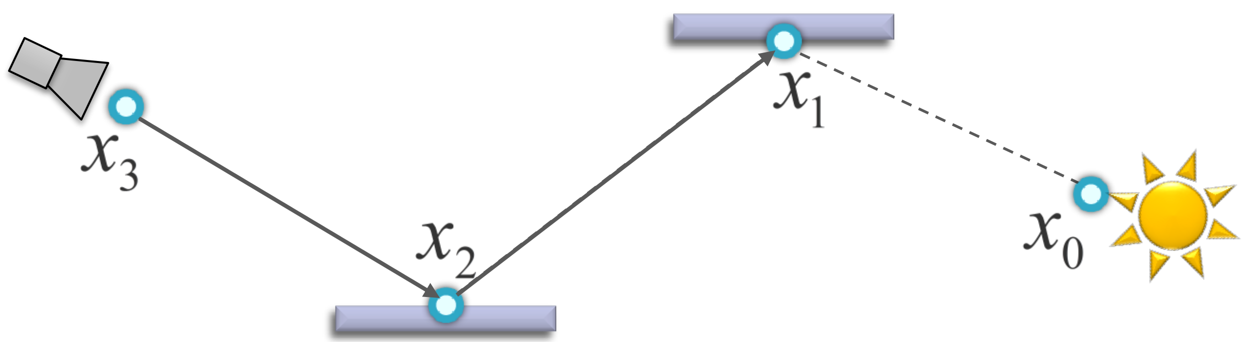
path PDF

$$p(\bar{x}) = p(x_0, \ldots, x_k) = p(x_3)\, p(x_2)\, p(x_1)\, p(x_0)$$

joint PDF of path vertices

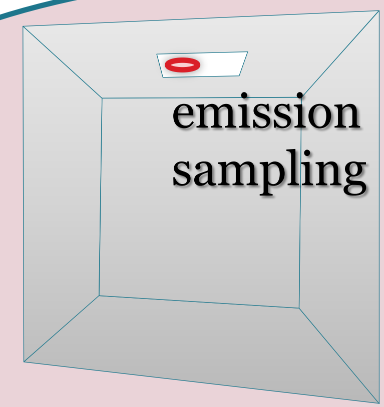
product of (conditional) vertex PDFs

Path tracing example:

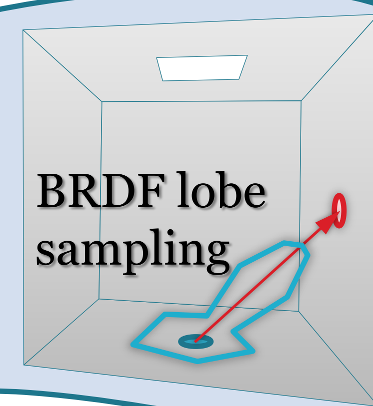


Vertex sampling

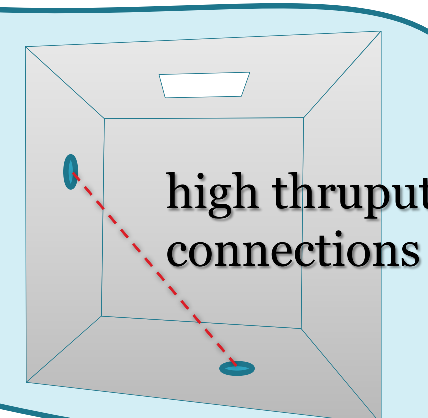
- **Importance sampling** principle



1. Sample from an a priori distrib.

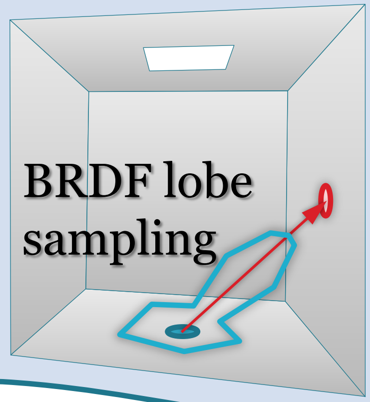


2. Sample direction from an existing vertex
3. Connect sub-paths



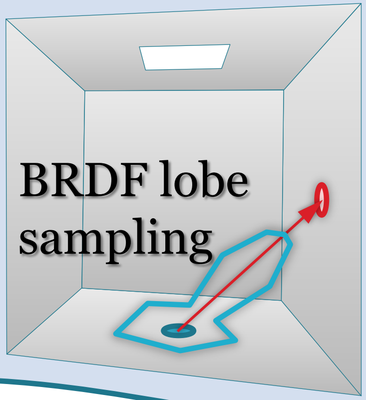
Vertex sampling

- Sample direction from an existing vertex



Measure conversion

- Sample direction from an existing vertex

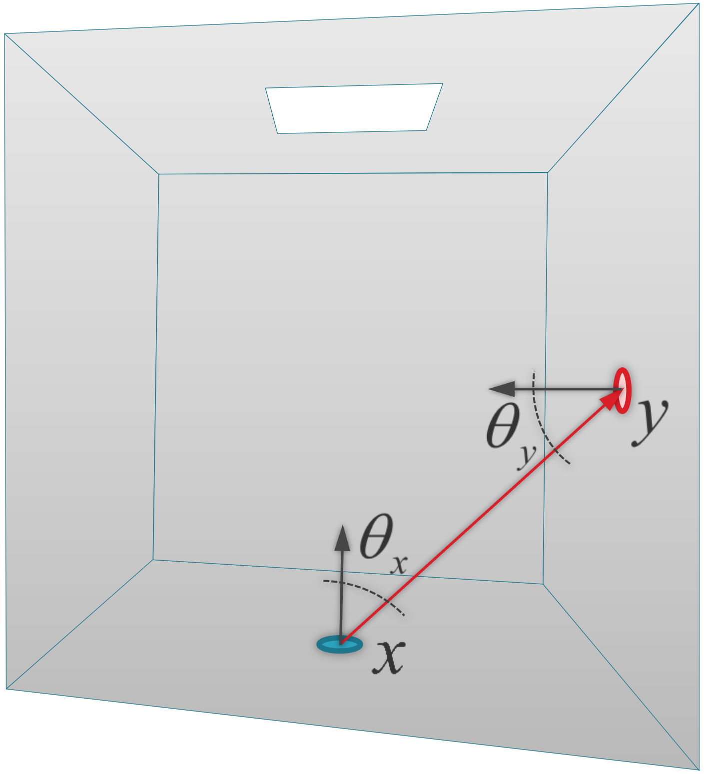


$$p(y) = p^{\perp}(x \rightarrow y)\, G(x \leftrightarrow y)$$

w.r.t. area

w.r.t. proj. solid angle

$$\langle I_j \rangle = \frac{f_j(\bar{x})}{p(\bar{x})} = \frac{\cdots \rho_s(x \rightarrow y) G(x \leftrightarrow y) \cdots}{\cdots p^{\perp}(x \rightarrow y) G(x \leftrightarrow y) \cdots}$$



Summary

Path integral

$$I_j = \int_{\Omega} f_j(\bar{x})\, \mathrm{d}\mu(\bar{x})$$

pixel value — all paths — contribution function

MC estimator

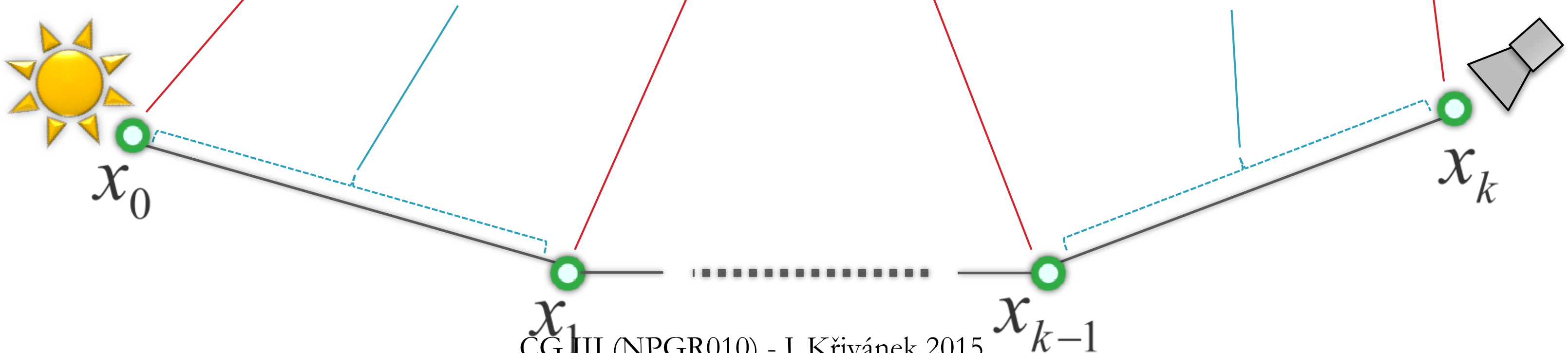
$$\langle I_j \rangle = \frac{f_j(\bar{x})}{p(\bar{x})}$$

path pdf — sampled path

$$\bar{x} = x_0 \ldots x_k$$

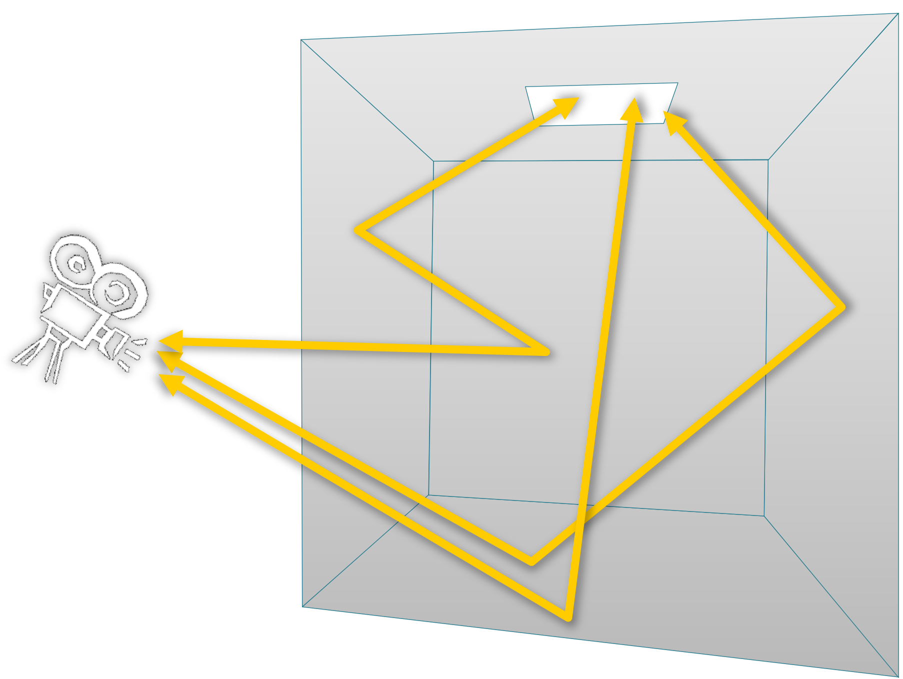
$$p(\bar{x}) = p(x_0) \ldots p(x_k)$$

$$f_j(\bar{x}) = L_e\, G(x_0 \leftrightarrow x_1)\, \rho_s(x_1) \ldots \rho_s(x_{k-1})\, G(x_{k-1} \leftrightarrow x_k)\, W_e^j$$



Summary

- **Algorithms**
  - different path sampling techniques
  - different path PDF



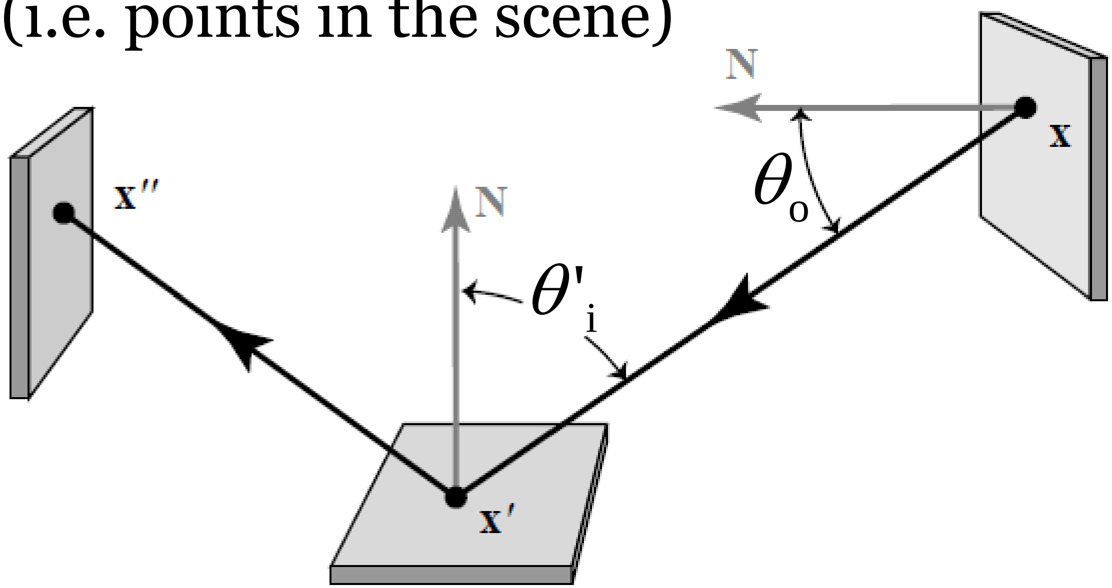
Why is the path integral view so useful?

- Identify source of problems
  - **High contribution paths** sampled with **low probability**
- Develop solutions
  - Advanced, global **path sampling techniques**
  - **Combined** path sampling techniques (MIS)

Derivation of the path integral from the rendering and measurement equations

Three-point formulation of light transport

- Let's eliminate all directions and only talk about path vertices (i.e. points in the scene)

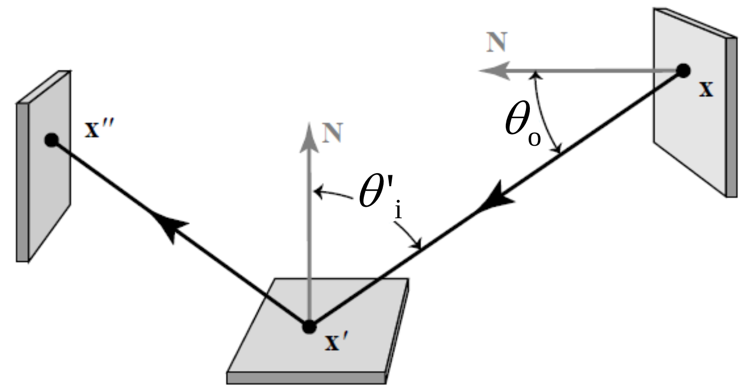


- We introduce notation: $L(\mathbf{x} \to \mathbf{x}') \equiv L(\mathbf{x}, \omega)$

$$f_r(\mathbf{x} \to \mathbf{x}' \to \mathbf{x}'') \equiv f_r(\mathbf{x}', \omega_\mathrm{i} \to \omega_\mathrm{o})$$

Rendering equation in the 3-pt formulation

- Let's use the above notation to rewrite the RE



$$L(\mathbf{x}' \to \mathbf{x}'') = L_e(\mathbf{x}' \to \mathbf{x}'') + \\ + \int_M L(\mathbf{x} \to \mathbf{x}') \cdot f_r(\mathbf{x} \to \mathbf{x}' \to \mathbf{x}'') \cdot G(\mathbf{x} \leftrightarrow \mathbf{x}')\, dA_{\mathbf{x}}$$

$$G(\mathbf{x} \leftrightarrow \mathbf{x}') = V(\mathbf{x} \leftrightarrow \mathbf{x}') \frac{\left|\cos\theta_o \cos\theta'_i\right|}{\left\|\mathbf{x} - \mathbf{x}'\right\|^2}$$

Measurement equation in the 3-pt formulation

$$I_j = \int_{M \times M} W_e^{(j)}(\mathbf{x} \to \mathbf{x}') \cdot L(\mathbf{x} \to \mathbf{x}') \cdot G(\mathbf{x} \leftrightarrow \mathbf{x}') \, \mathrm{d}A_{\mathbf{x}} \, \mathrm{d}A_{\mathbf{x}'}$$

Visual importance emitted from **x**' to **x**
(Notation: arrow = direction of the flow of light, not importance)

x' … on the sensor
x … on the scene surface

Derivation of the path integral: A sketch

- Plug the Neumann expansion of the RE into the measurement equation, you get a sum of integrals.

- The integrand of this sum is the path contribution function.

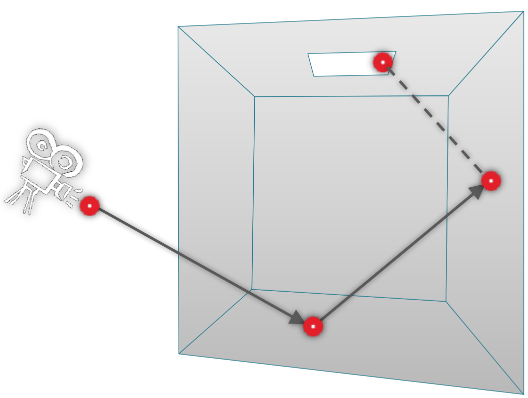
"Path integral" – A historical remark

- This course [Veach and Guibas 1995], [Veach 1997]
  - Easily derived form the rendering equation [Veach 1997]
- Feynman path integral formulation of quantum mechanics [Feynman and Hibbs 65]
- Homogeneous materials [Tessendorf 89, 91, 92]
- Rendering [Premože et al. 03, 04]

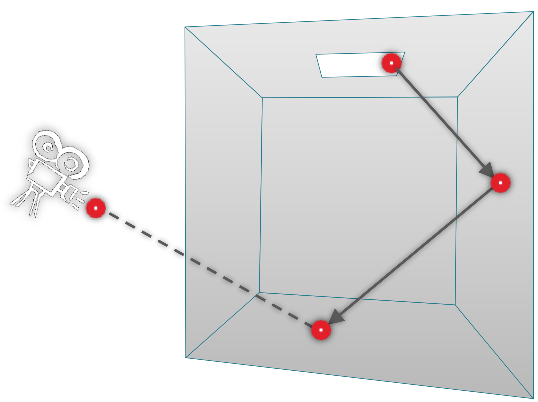
Bidirectional path tracing

Bidirectional path tracing

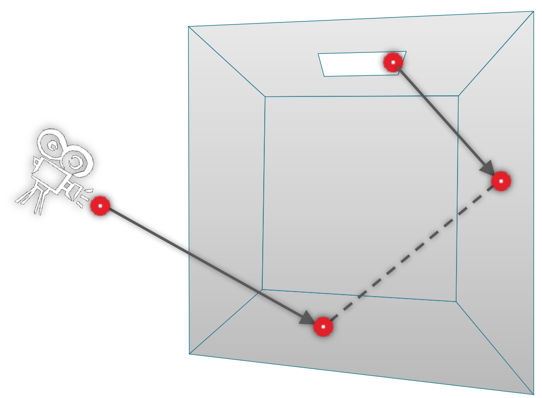
Path tracing



Light tracing

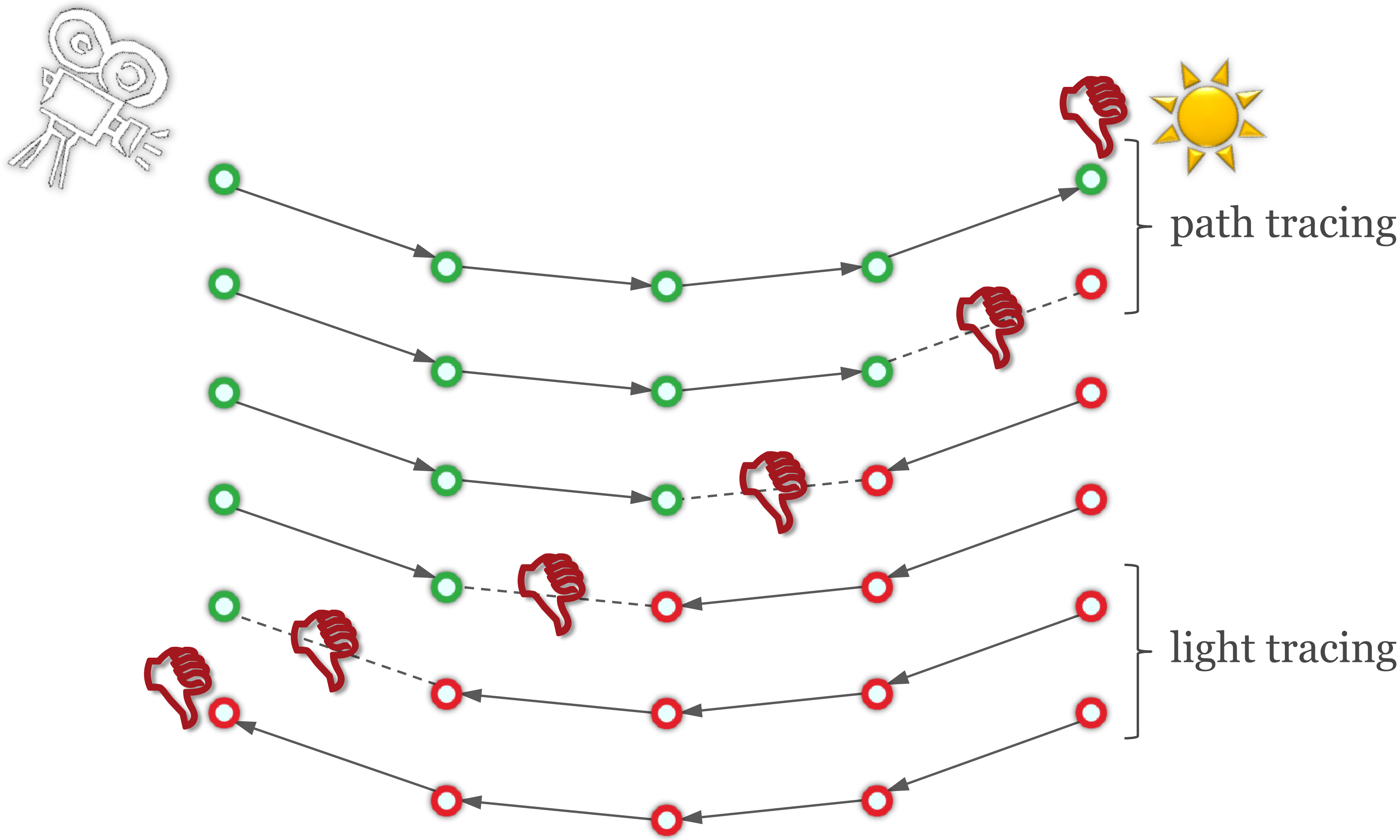


Bidirectional path sampling



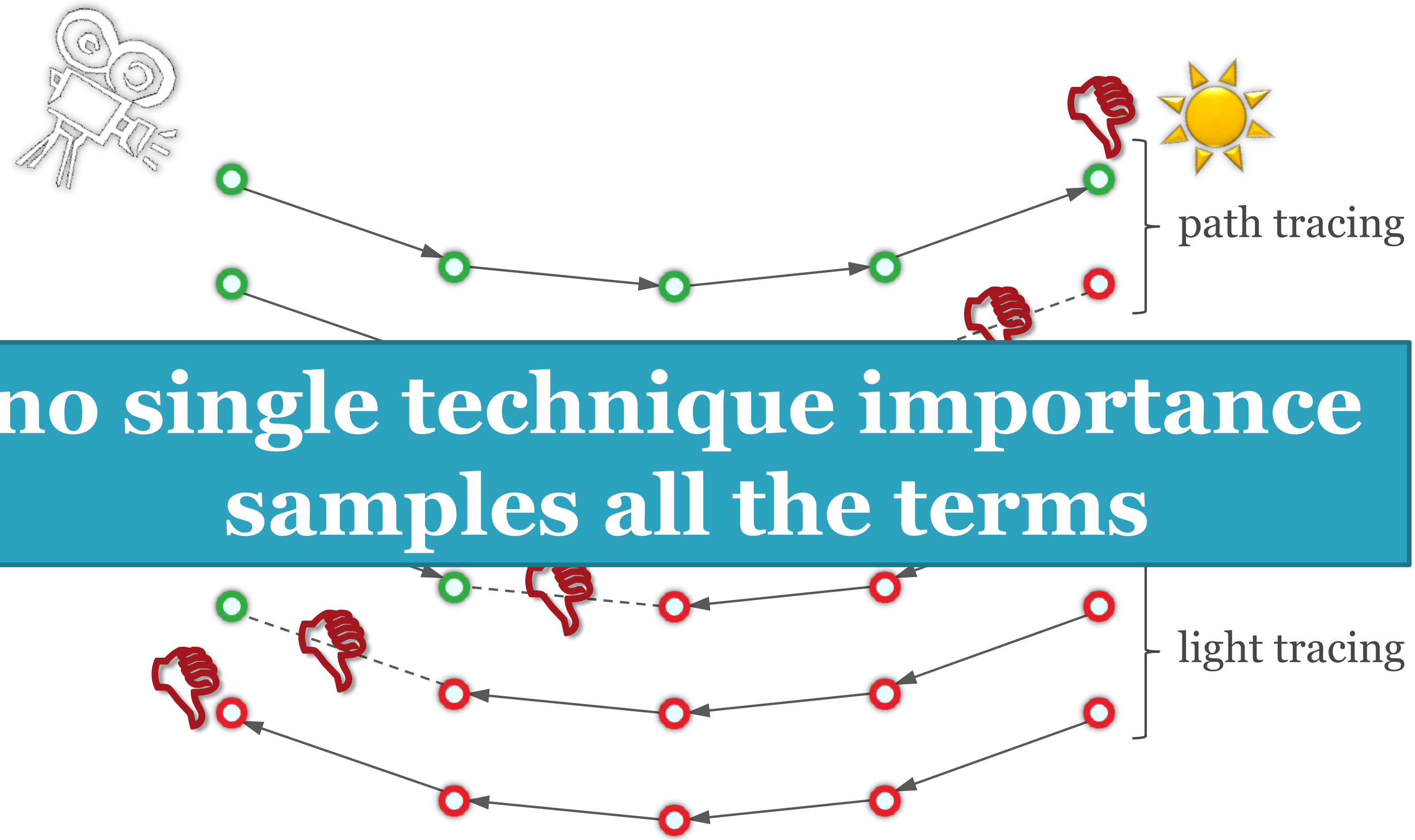
All possible bidirectional techniques

- vertex on a **light sub-path**
- vertex on en **eye sub-path**



All possible bidirectional techniques

- vertex on a **light sub-path**
- vertex on en **eye sub-path**



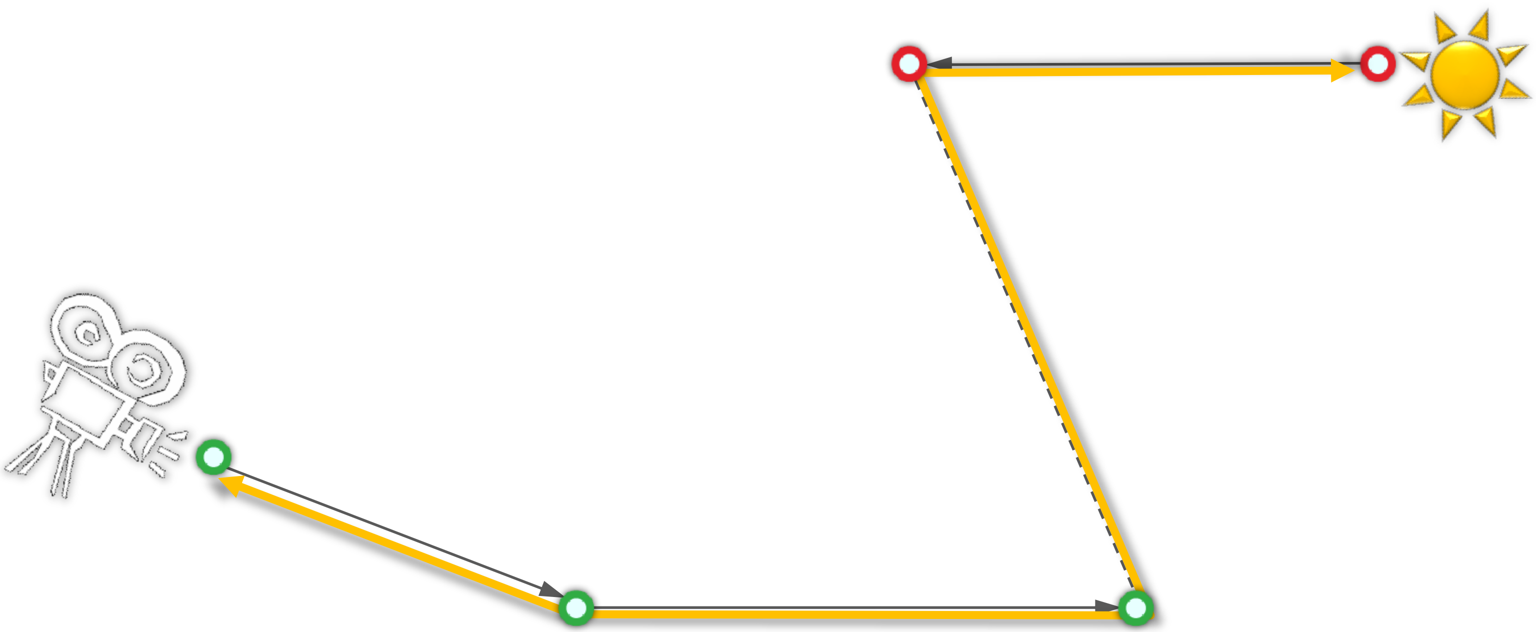
no single technique importance samples all the terms

Bidirectional path tracing

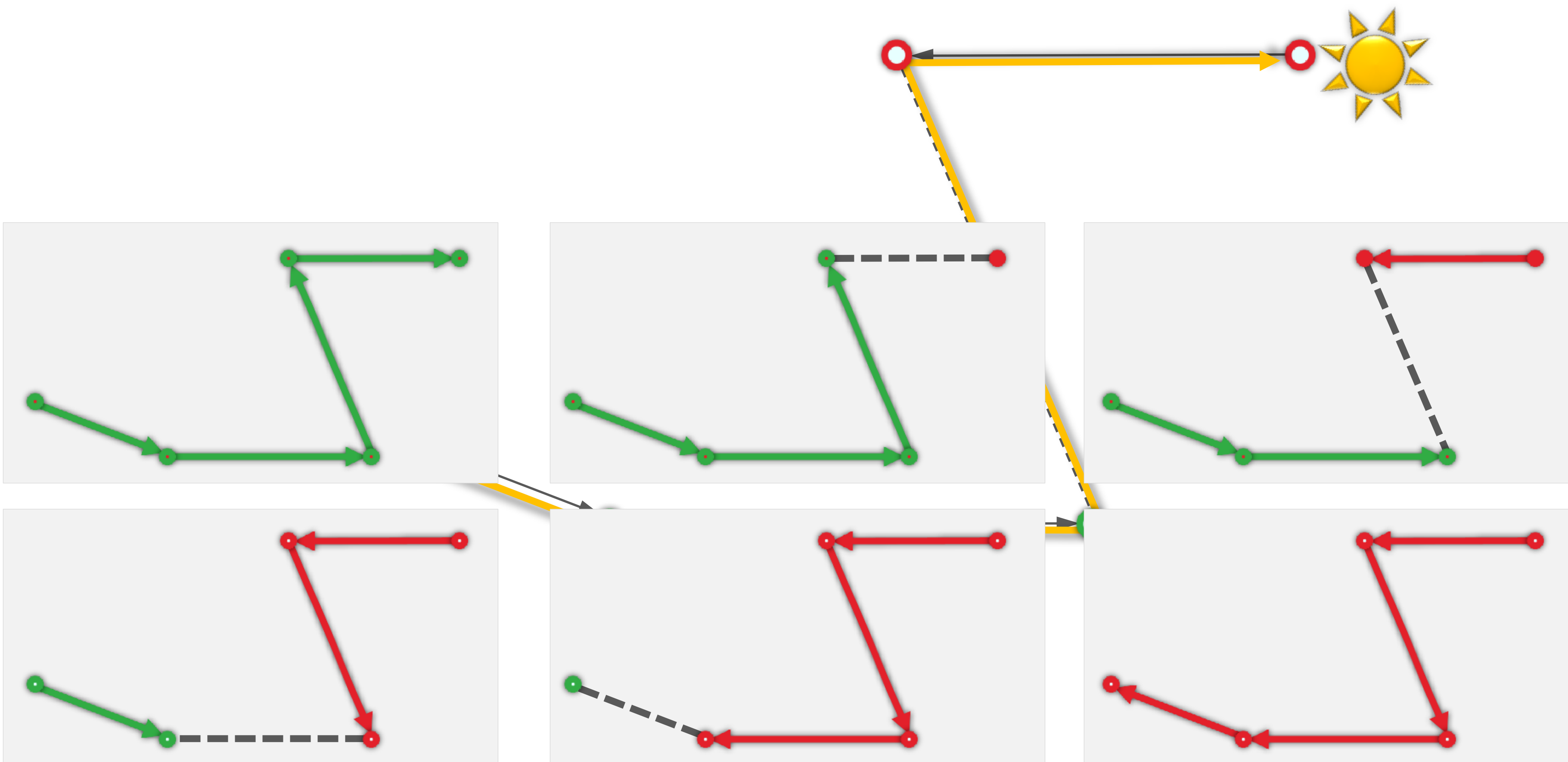
- Use **all** of the above sampling techniques
- Combine using **Multiple Importance Sampling**

- Generalizes the combined strategy for calculating direct illumination in a path tracer
  - **PT**: Different strategies for sampling a direction toward a light source
  - **BPT:** Different strategies for sampling **entire light transport paths**

Naive BPT

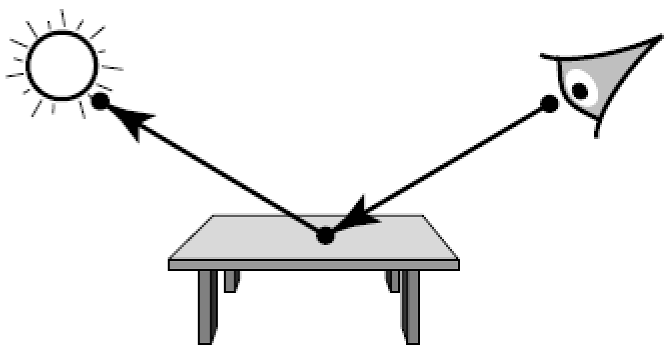


MIS weight calculation

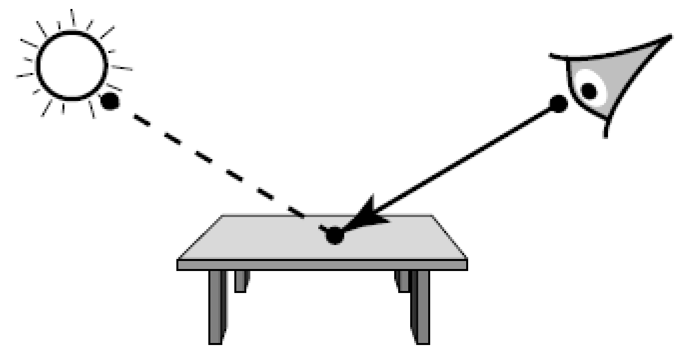


Sampling techniques in BPT

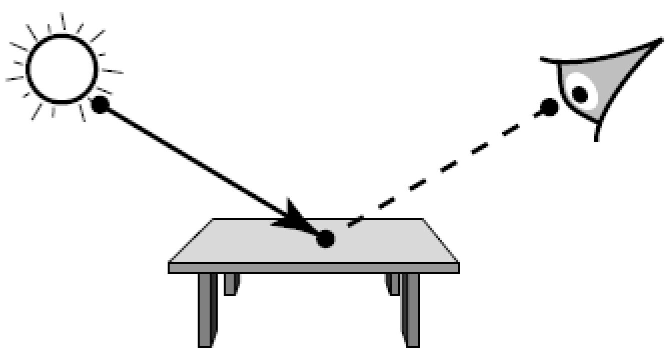
Example: Four techniques for $k = 2$



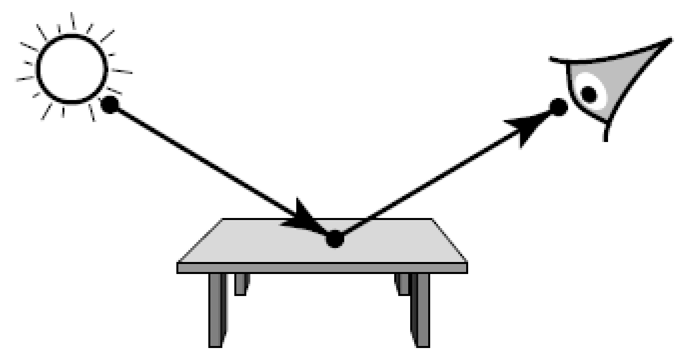
(a) $s = 0, t = 3$



(b) $s = 1, t = 2$



(c) $s = 2, t = 1$



(d) $s = 3, t = 0$

Image: Eric Veach

Sampling techniques in BPT

- Sub-path with t vertices sampled from the camera
- Sub-path with s vertices sampled from the light sources
- Connection segment of length 1
- Total path length : $k = s + t - 1$ (number of **segments**)

- In BPT, there are $k+2$ way to generate a path of length k

Sampling techniques in BPT

- Each path sampling technique has a different **probability density** $\boldsymbol{p_{s,t}}$
- Each techniques is efficient at sampling different kinds of lighting effects
- All of them estimate the **same integral**

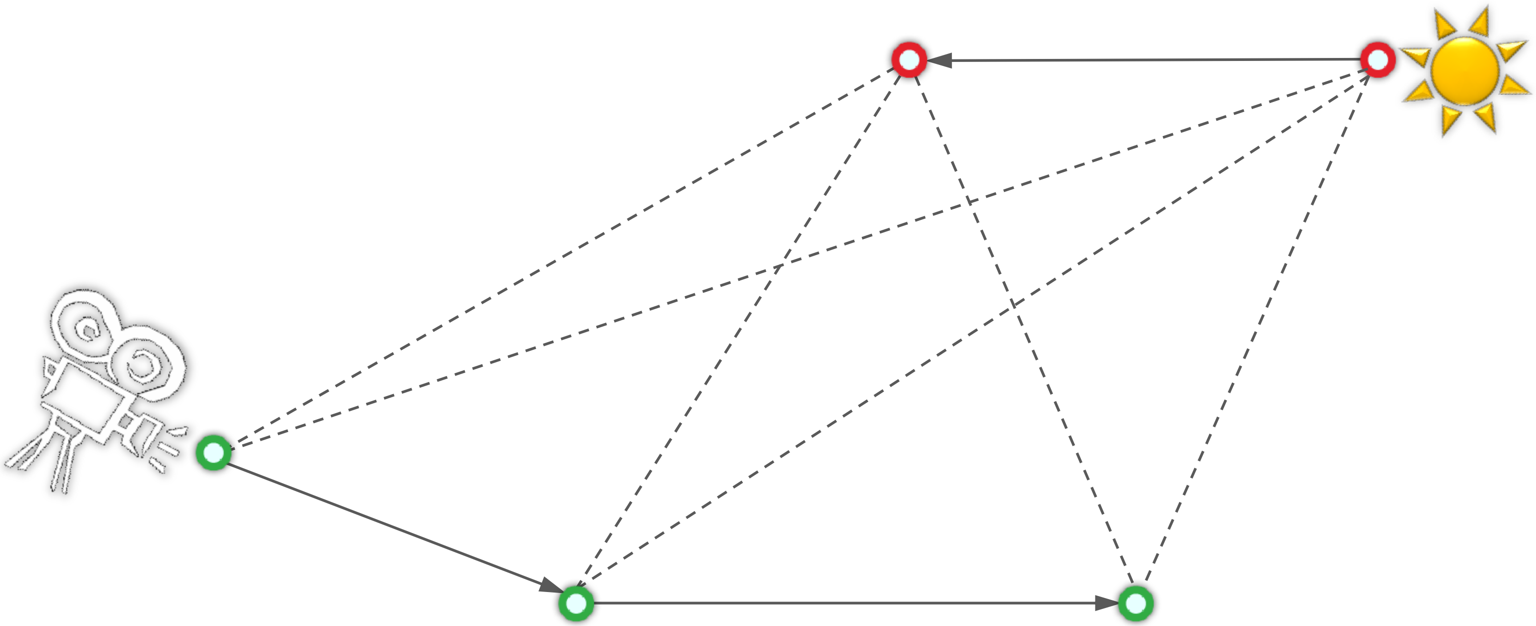
Combination of path sampling techniques

- Combined estimator (MIS)

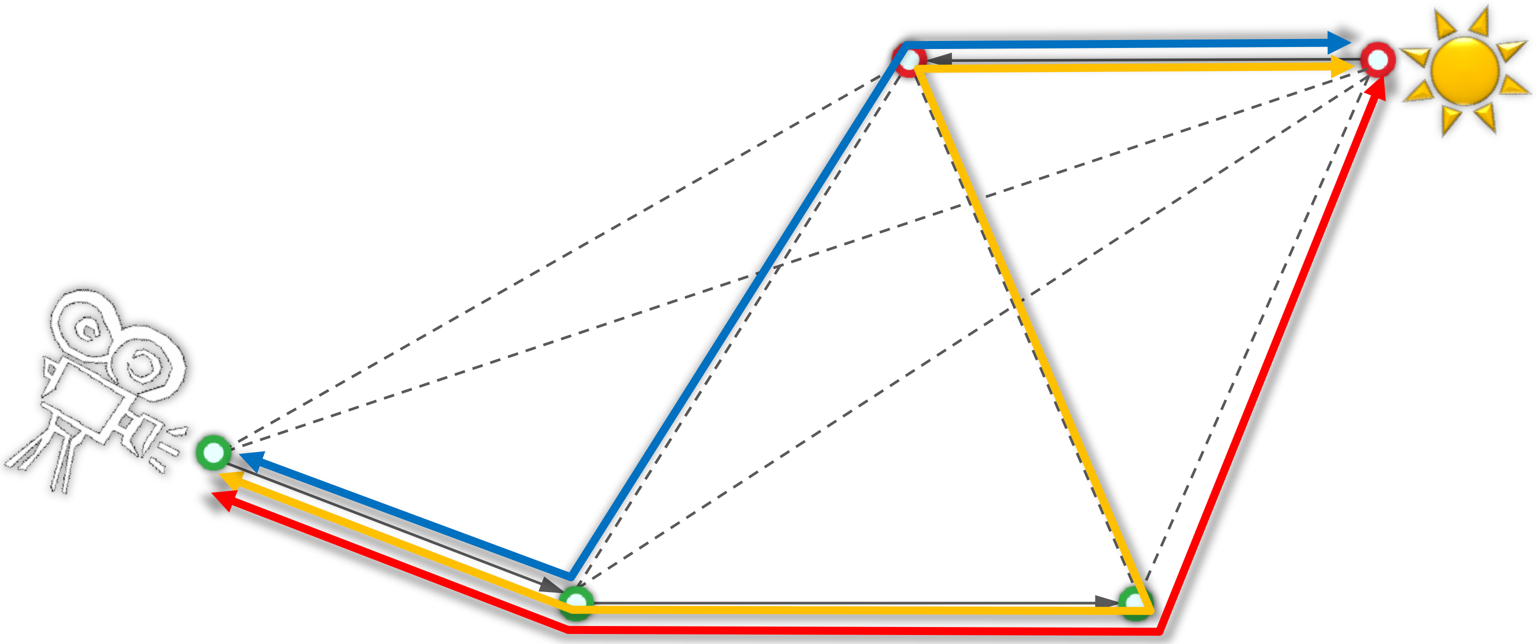
$$F = \sum_{s \geq 0} \sum_{t \geq 0} w_{s,t}(\bar{x}_{s,t}) \frac{f_j(\bar{x}_{s,t})}{p_{s,t}(\bar{x}_{s,t})}$$

MIS weights
(e.g. the balance heuristic)

BPT implementation in practice



BPT implementation in practice



BPT implementation in practice

- Sample a sub-path of a random length starting **from light sources**

$$\mathbf{y}_0 \dots \mathbf{y}_{n_L - 1}$$

- Sample a sub-path of random length starting **from the camera**

$$\mathbf{z}_{n_E - 1} \dots \mathbf{z}_0$$

- Connect each **prefix of a sub-path from light** with each **suffix of a sub-path from the camera**

$$\bar{x}_{s,t} = \mathbf{y}_0 \dots \mathbf{y}_{s-1} \mathbf{z}_{t-1} \dots \mathbf{z}_0$$

Results

BPT, 25 samples per pixel

PT, 56 samples per pixel

Images: Eric Veach

k = 2 (2x)

k = 3 (4x)

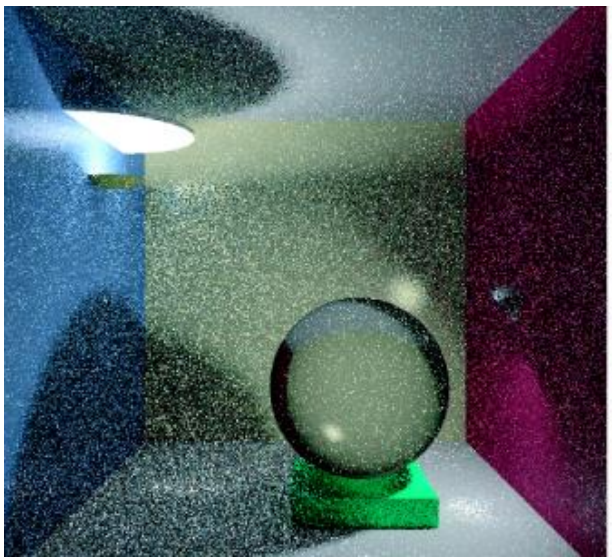
k = 4 (8x)

k = 5 (16x)

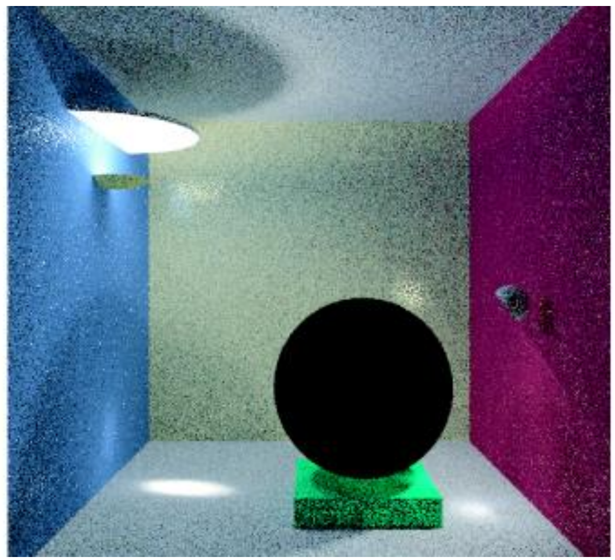
s = 1 *s* = 2 … *t* = 2 *t* = 1

s / t = number of vertices on the sub-path from light / camera

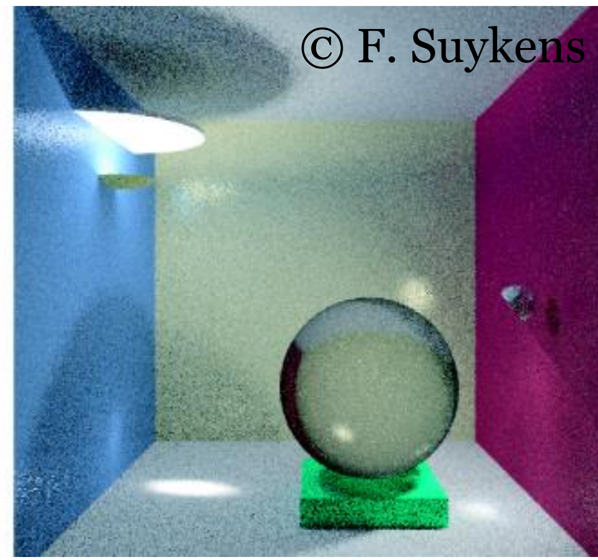
Algorithm comparison again



Path tracing



Light tracing



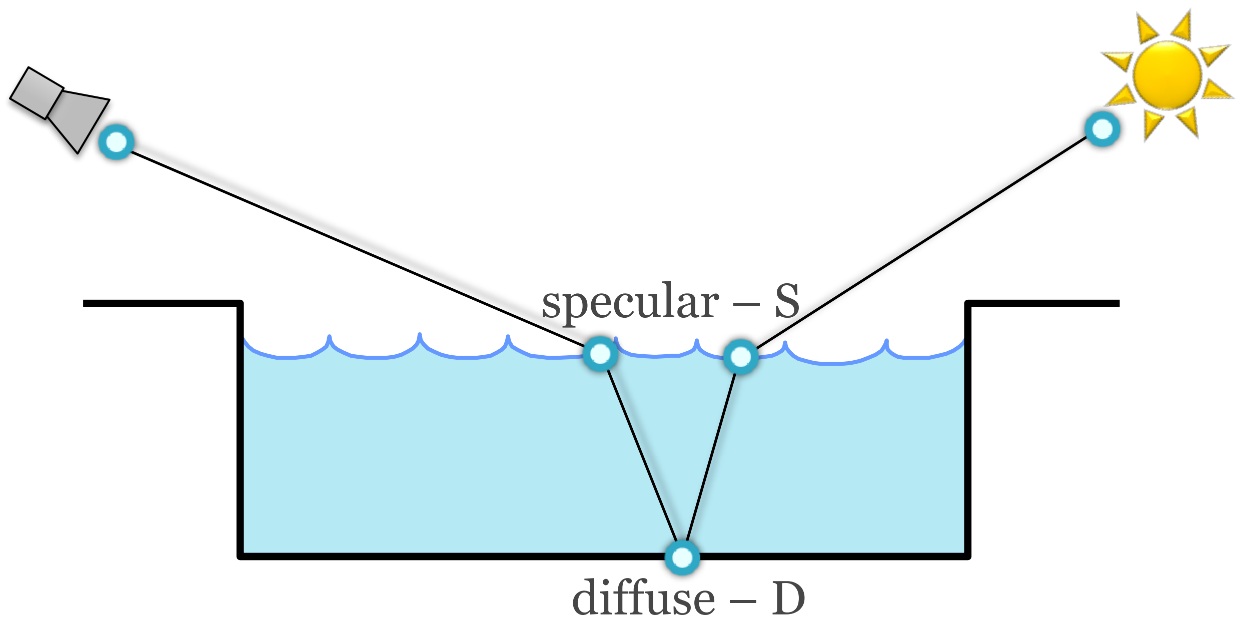
Bidirectional path tracing

LIMITATIONS OF LOCAL PATH SAMPLING


Reference solution
Bidirectional path tracing

Insufficient path sampling techniques

- Some paths sampled with zero (or very small) probability



Alternatives to local path sampling

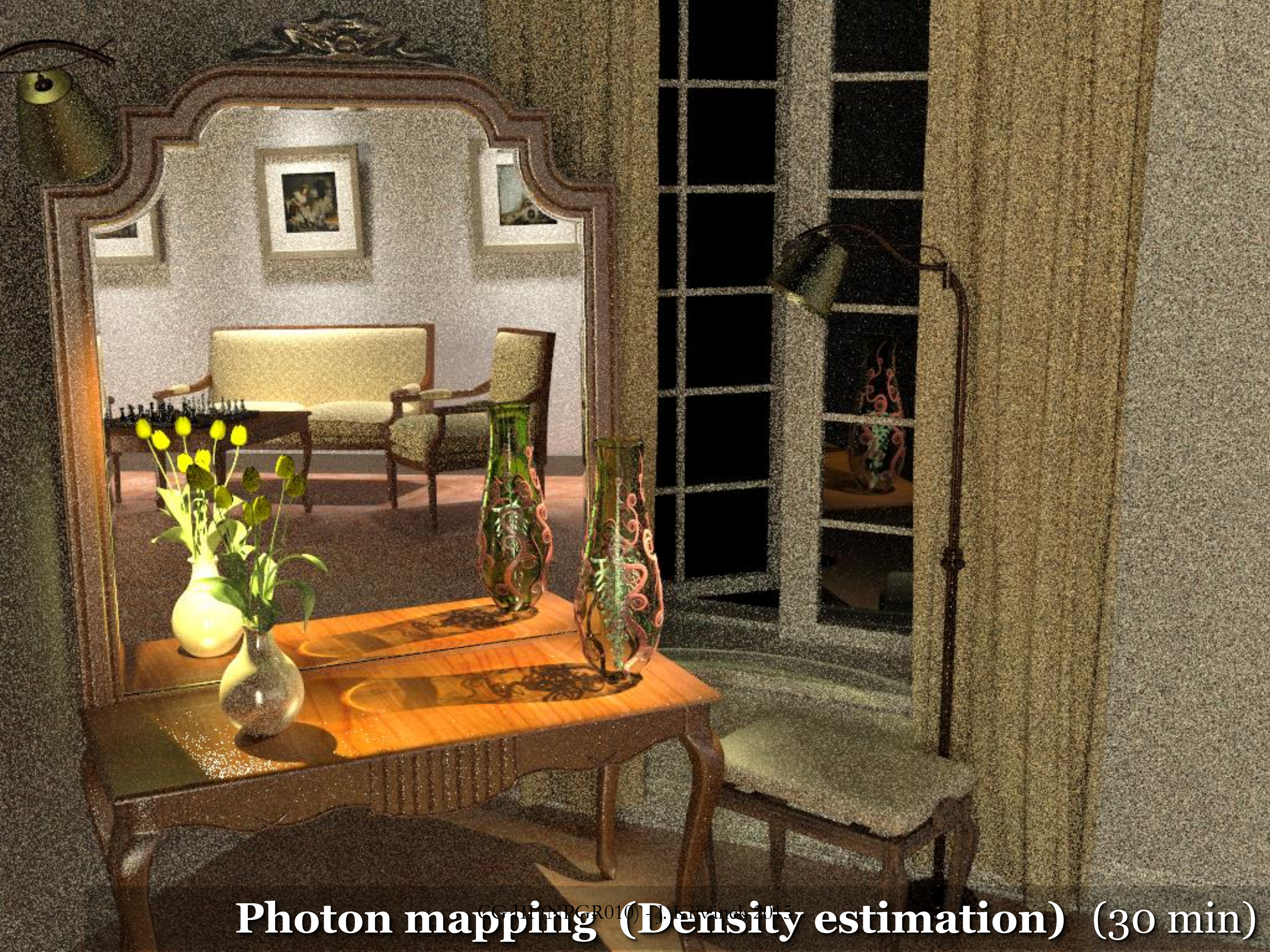
- **Global** path sampling – **Metropolis light transport**
  - Initial proposal still relies on local sampling
- Leave path integral framework
  - Density estimation – **photon mapping**
- **Unify** path integral framework and density estimation
  - **Vertex Connection & Merging**

Our work:
Vertex Connection and Merging

Robust photon mapping

- Where exactly on the camera sub-path should we look-up the photons?
- Commonly solved via a **heuristic**:
  - Diffuse surface … make the look-up right away
  - Specular surface … continue tracing and make the look-up later
- But what exactly should be classified as "diffuse" and "specular"?
  - We need a more **universal** and **robust** solution
  - Solution:
    - **Bidirectional photon mapping** [Vorba 2011]
    - **Vertex Connection and Merging** [Georgiev et al., 2012]

Bidirectional path tracing (30 min)



Photon mapping (Density estimation) (30 min)

Vertex connection and merging (30 min)

Overview

- ☹ **Problem**: different mathematical frameworks
  - ❑ **BPT**: Monte Carlo estimator of a path integral
  - ❑ **PM**: Density estimation

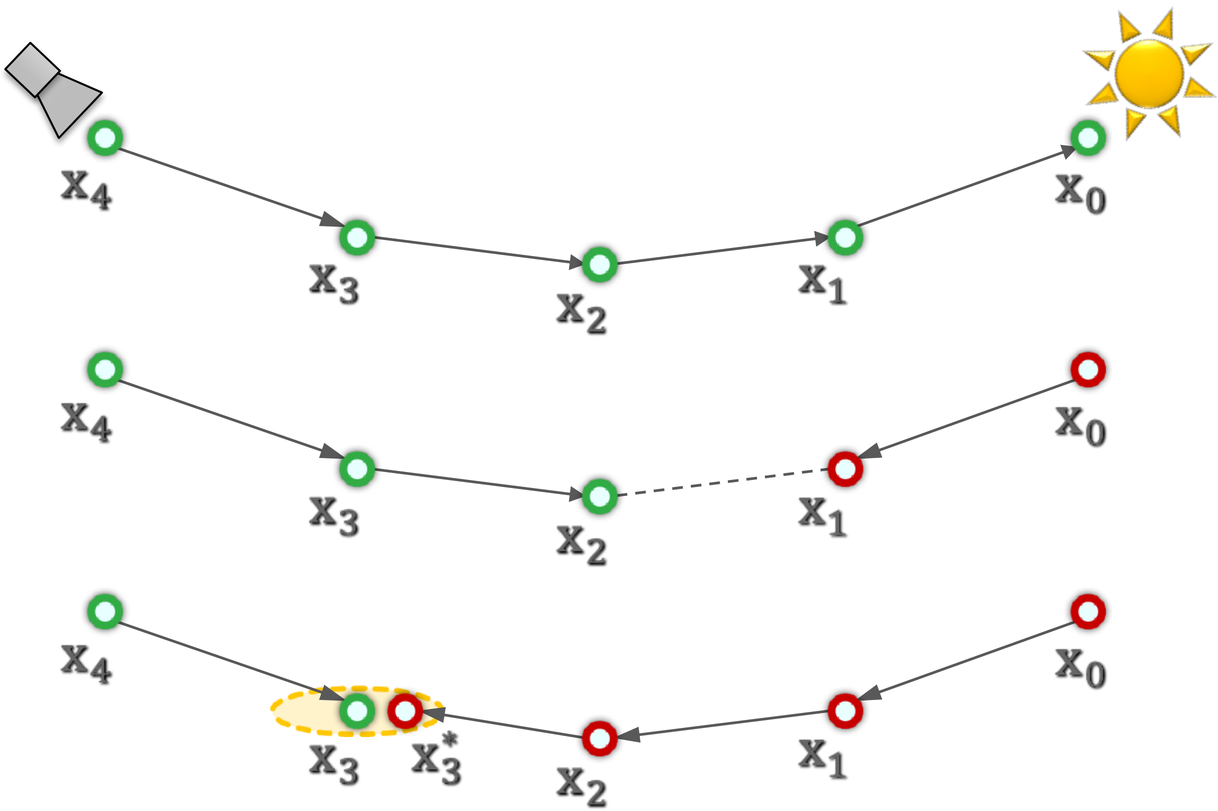
- ☝ **Key contribution:** Reformulate photon mapping in Veach's path integral framework

  1) Formalize as path sampling technique
  2) Derive path probability density

- ✓ Combination of BPT and PM into a **robust** algorithm

Sampling techniques

- Light vertex
- Camera vertex

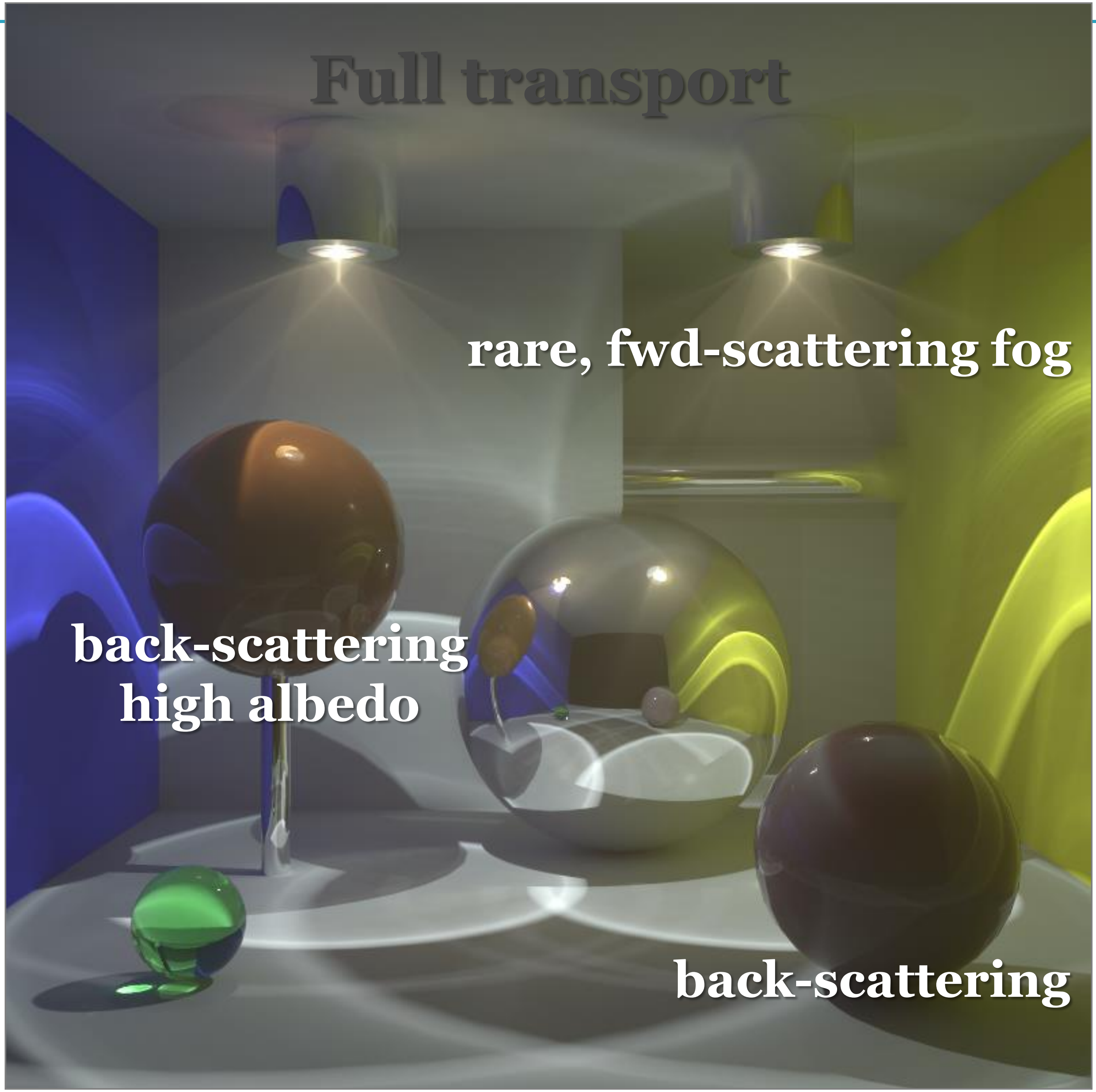


Unidirectional	2 ways
Vertex connection	4 ways
Vertex merging	**5 ways**
Total	**11 ways**

Combining path sampling techniques for volumetric light transport

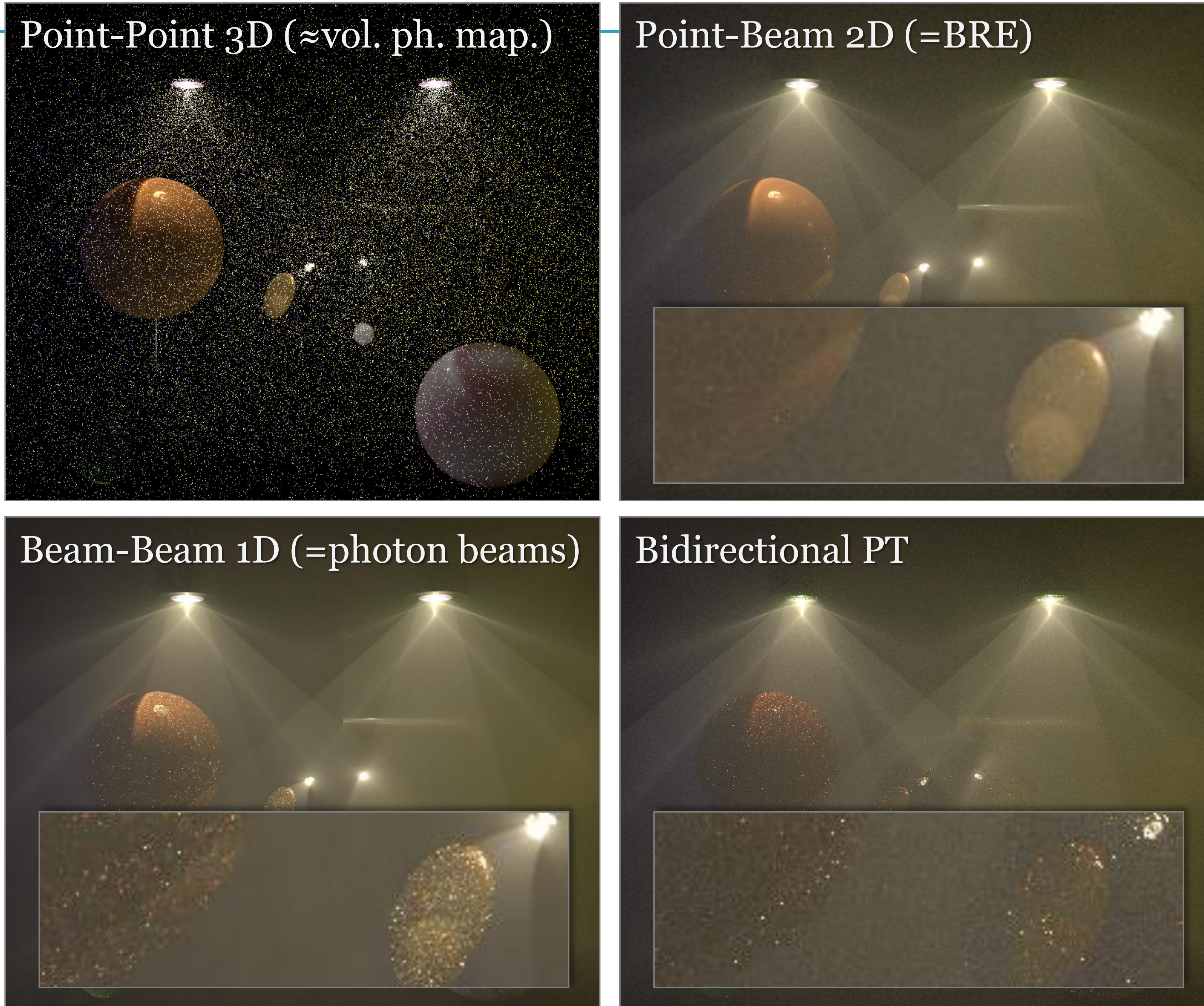
In the following we apply MIS to combine full path sampling techniques for calculating light transport in participating media.

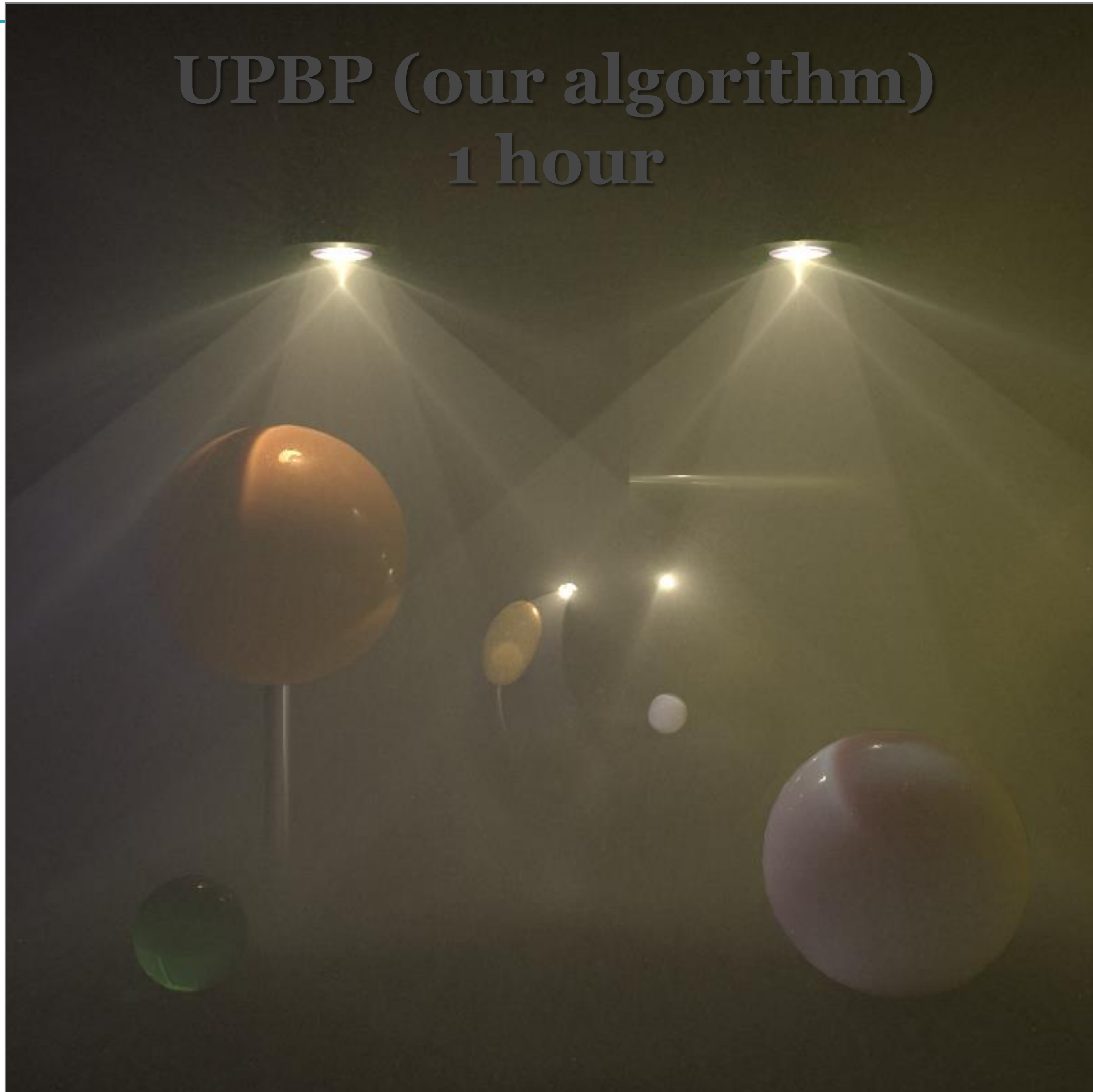
The results come from the SIGGRAPH 2014: Křivánek et al. Unifying points, beams and paths in volumetric light transport simulation.


Full transport
rare, fwd-scattering fog
back-scattering
high albedo
back-scattering

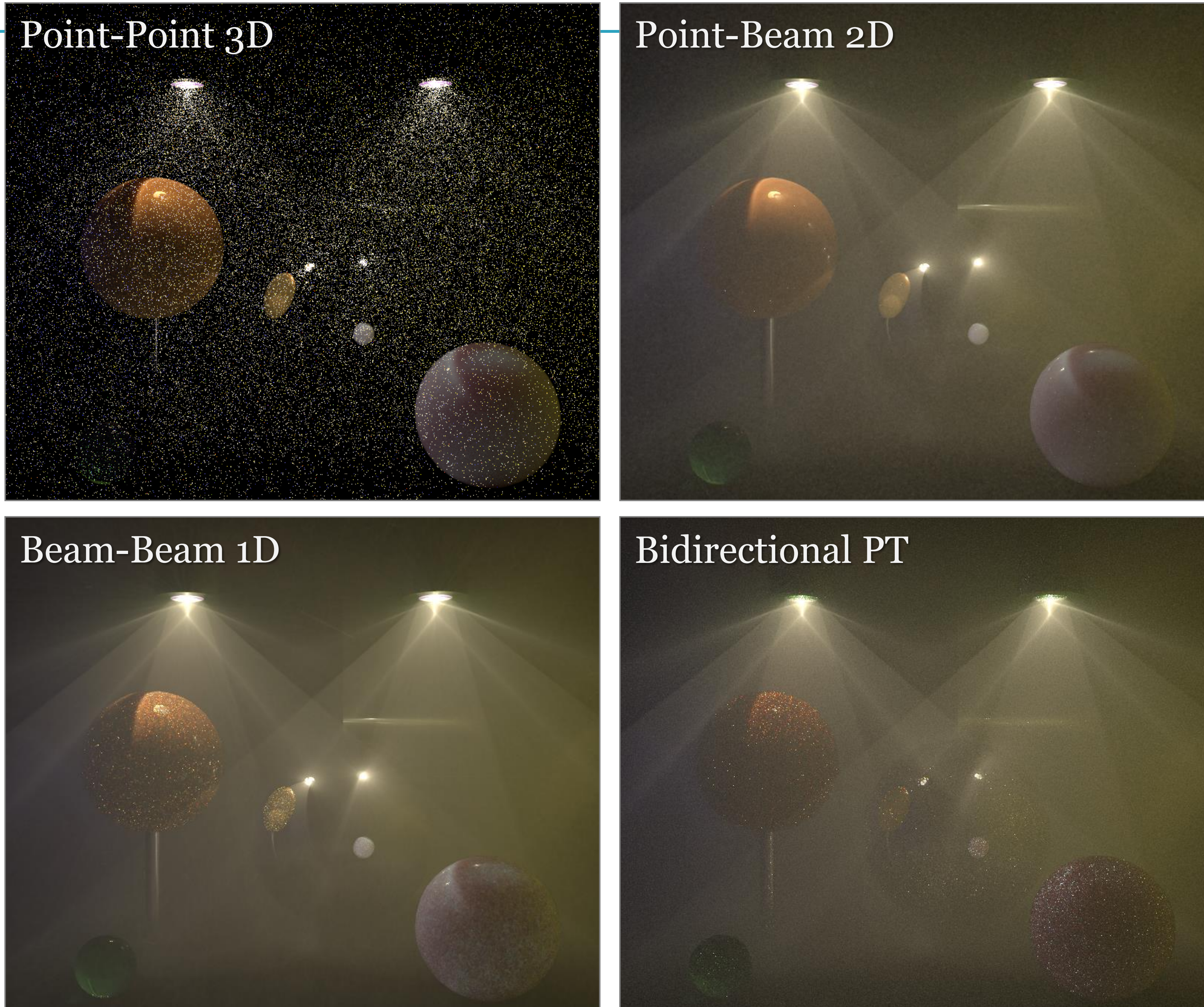

Medium transport only

Previous work comparison, 1 hr




UPBP (our algorithm)
1 hour

Previous work comparison, 1 hr




Weighted contributions
Point-Point 3D
Point-Beam 2D
Beam-Beam 1D
Bidirectional PT

Literature

E. Veach: Robust Monte Carlo methods for light transport simulation, PhD thesis, Stanford University, 1997, pp. 219-230, 297-317
http://www.graphics.stanford.edu/papers/veach_thesis/