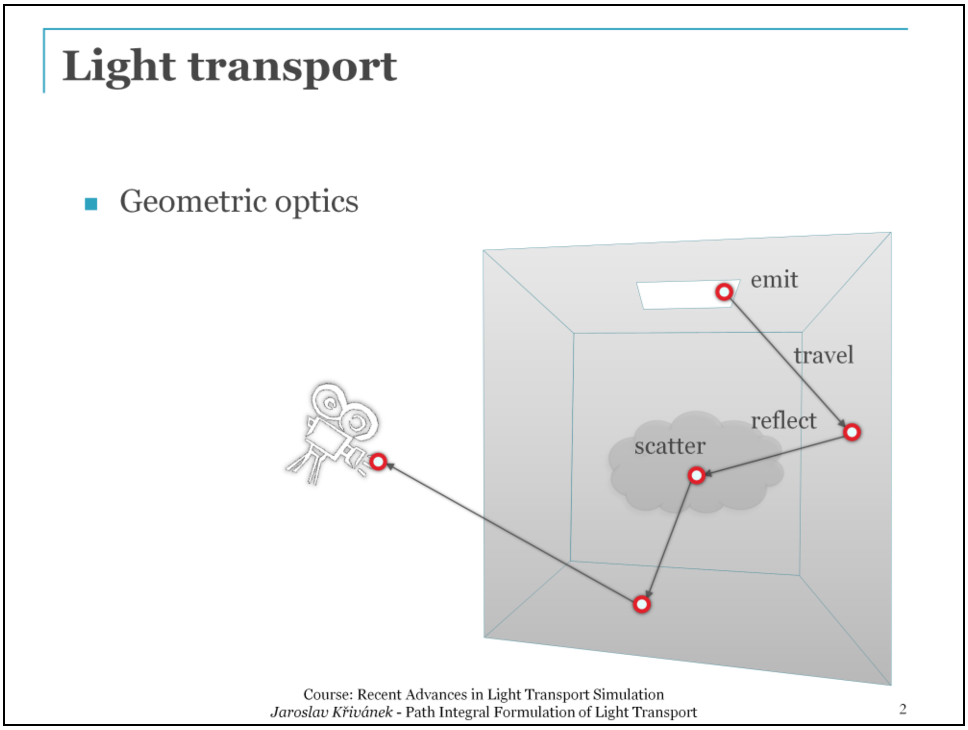
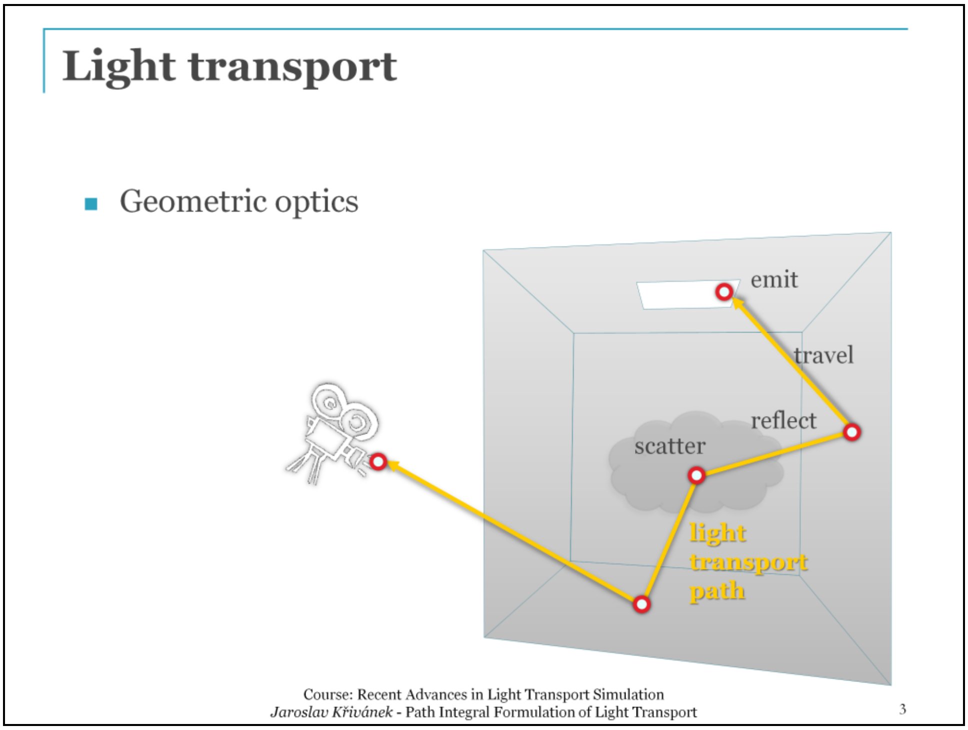
PATH INTEGRAL FORMULATION OF LIGHT TRANSPORT

Jaroslav Křivánek
Charles University in Prague
http://cgg.mff.cuni.cz/~jaroslav/



- In the real world, light sources emit light particles, which travel in space, reflect at objects or scatter in volumetric media (potentially multiple times) until they are absorbed.
- On their way, they might hit the sensor of the camera which will record the light contribution.

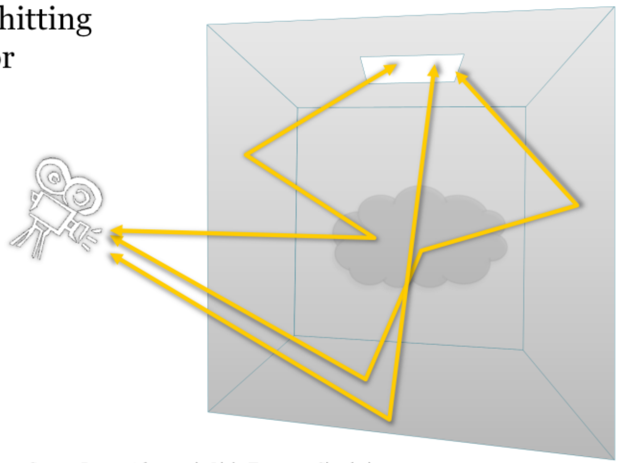


• The light particles travel along trajectories that we call “light transport paths”.

• In an environment consisting of surfaces and media with constant index of refraction, these paths are polylines whose vertices correspond to reflection at surfaces or scattering events in media, and the straight edges correspond to light travelling the in the space.

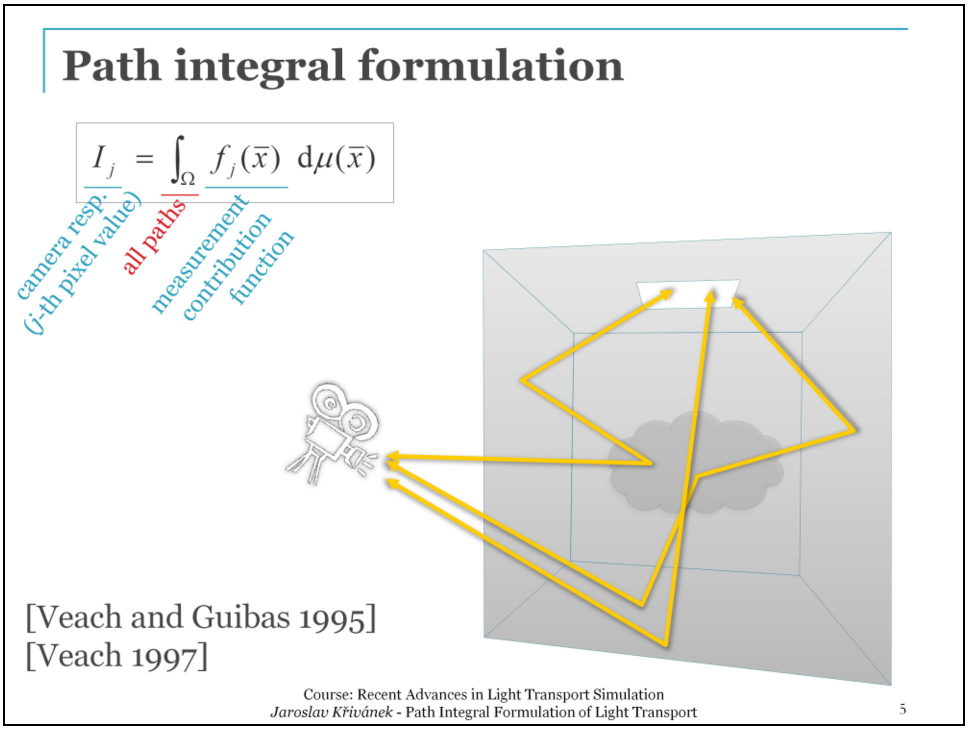
Light transport

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



Course: Recent Advances in Light Transport Simulation
Jaroslav Křivánek - Path Integral Formulation of Light Transport 4

• The final response of the camera is due to all the light particles – travelling over all possible paths – that hit the camera sensor during the shutter open period.

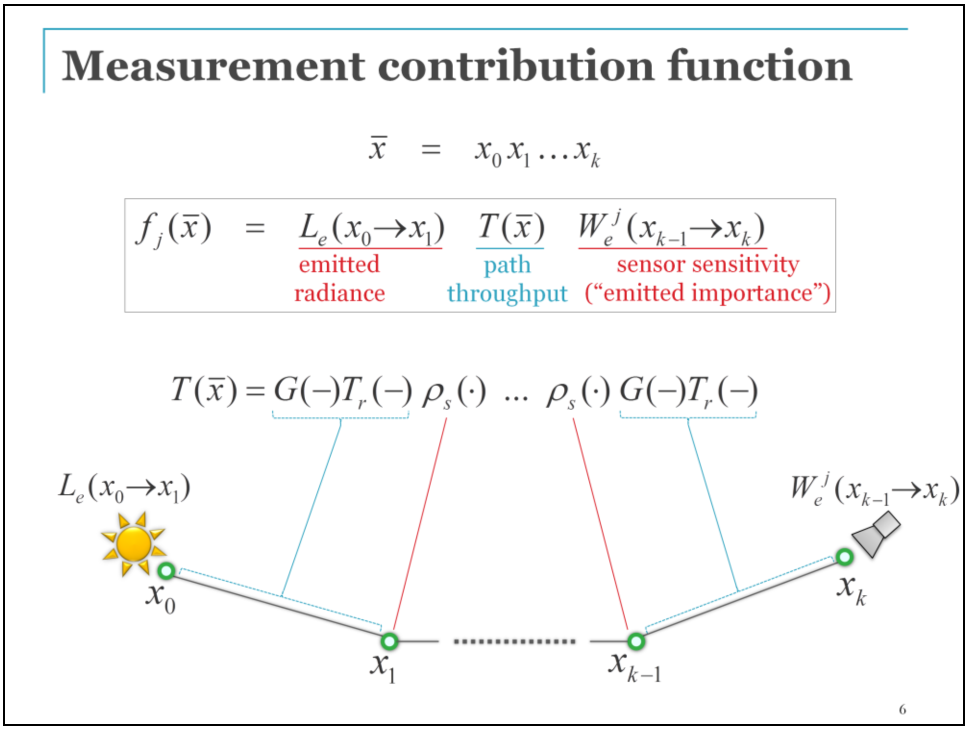


• The path integral formulation of light transport is nothing but a mathematical formulation of this simple physical idea.

• The camera response (which, in image synthesis will be the value of a pixel) is written as an integral over all light transport paths of all lengths in the scene.

• The integrand of this integral is the so called “measurement contribution function”.

• The measurement contribution function of a given path encompasses the “amount” of light emitted along the path, the light carrying capacity of the path, and the sensitivity of the sensor to light brought along the path.



• Let us now look at a more formal definition of the measurement contribution for a light path.

• As I already mentioned, a light transport path is a polyline with vertices corresponding to light reflection on surfaces or scattering in media.

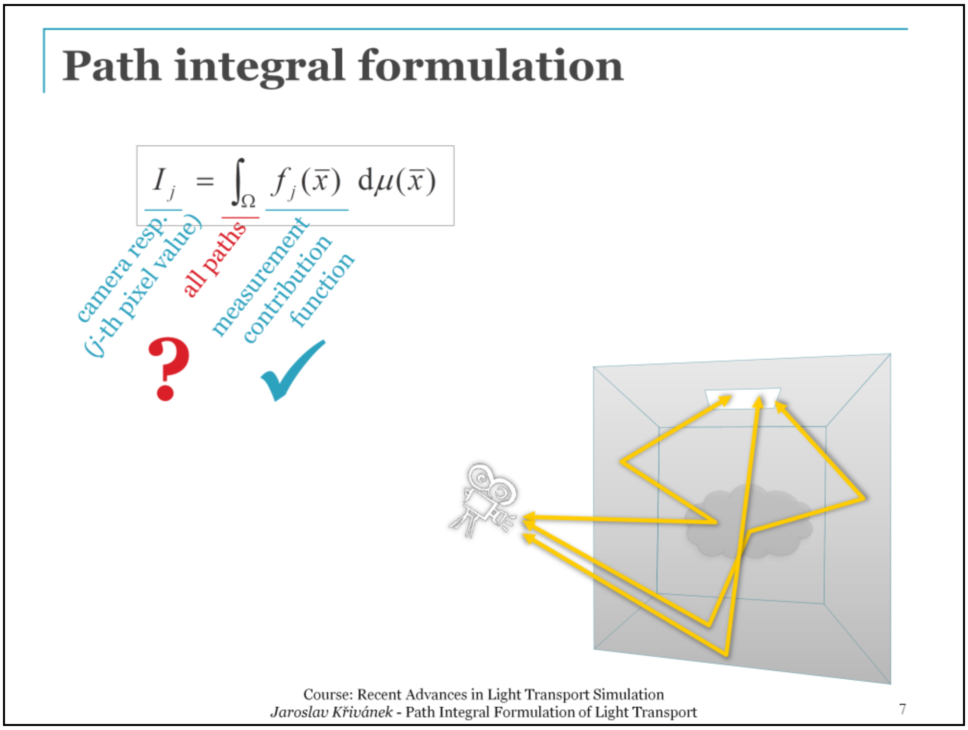
• We write the path (of length *k*) simply as a sequence of vertices, in the order of light flow. So the first vertex, x_0, corresponds to light emission at the light source and the last vertex, x_k, to light measurement at the sensor.

• The measurement contribution function *f*(*x*) for a path *x* is defined as the emitted radiance L_e at the first vertex, times the sensitivity (or “emitted importance”) at the last vertex, times the path throughput *T*(*x*).

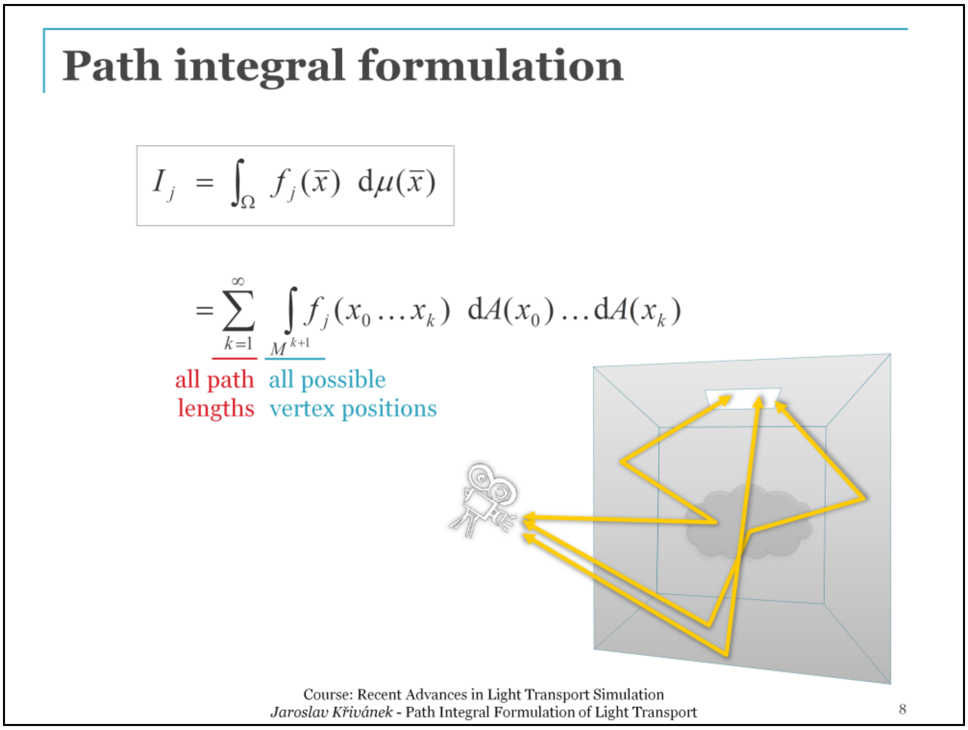
• The throughput is a product of the geometry term *G* and volume transmittance T_r for each path edge, and reflection/scattering terms for each interior path vertex.

• The geometry term measures the differential throughput along an edge and is essentially a point-to-point form factor.

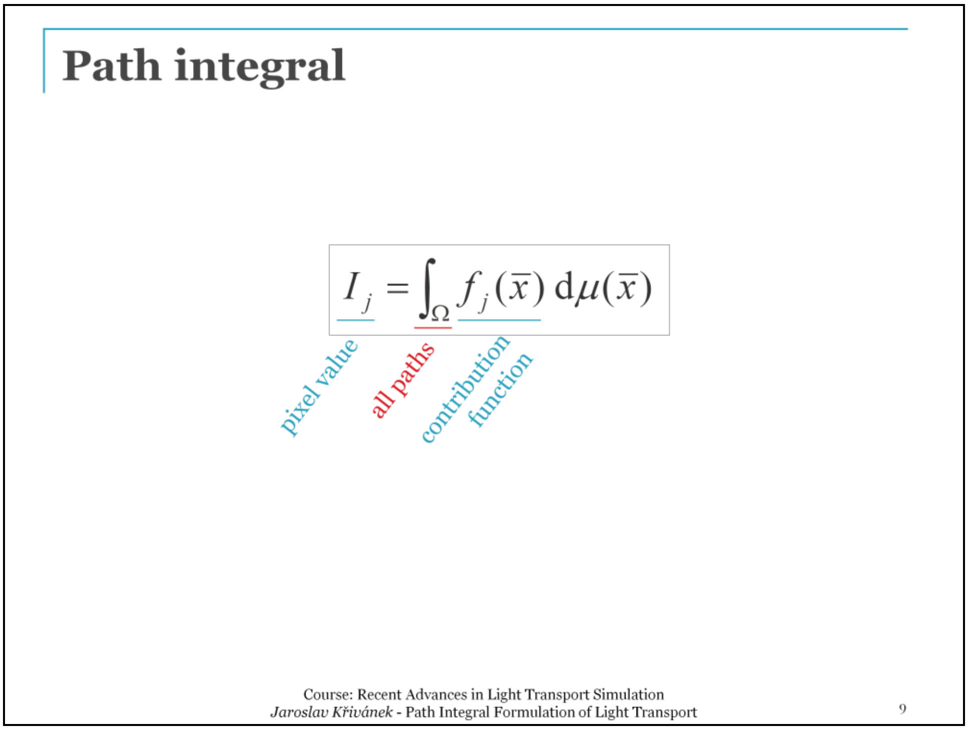
• The transmittance T_r tells us how much light is *not* attenuated due to medium absorption and out-scattering.



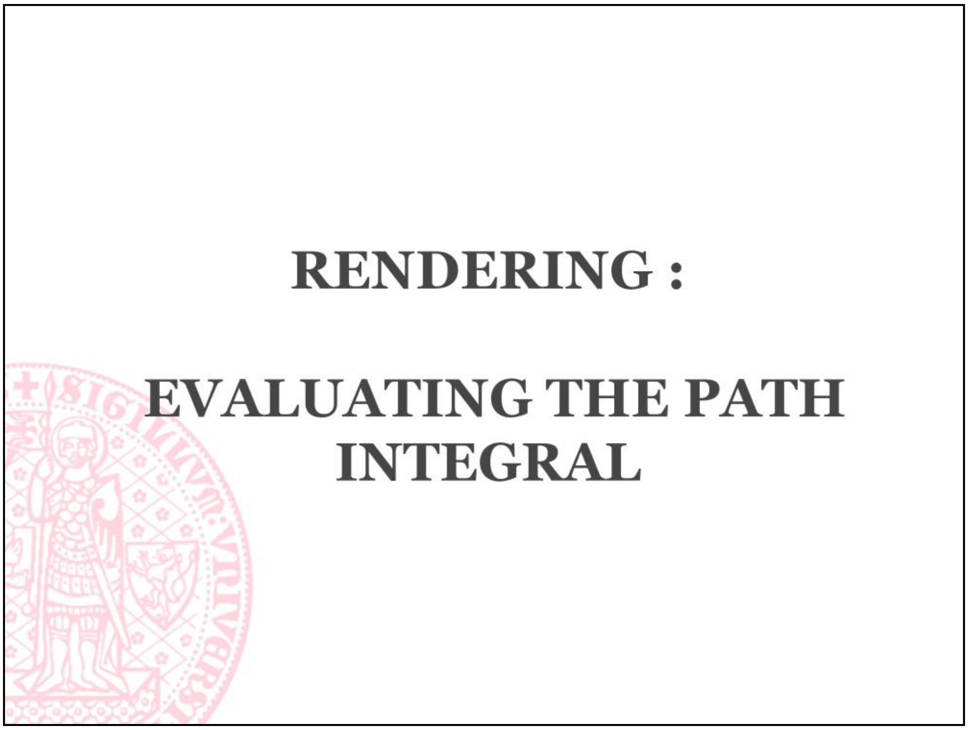
- Back to the path integral…
- We now know the meaning of the integrand – the path contribution function – but the integration domain of “all path” deserves more clarification.



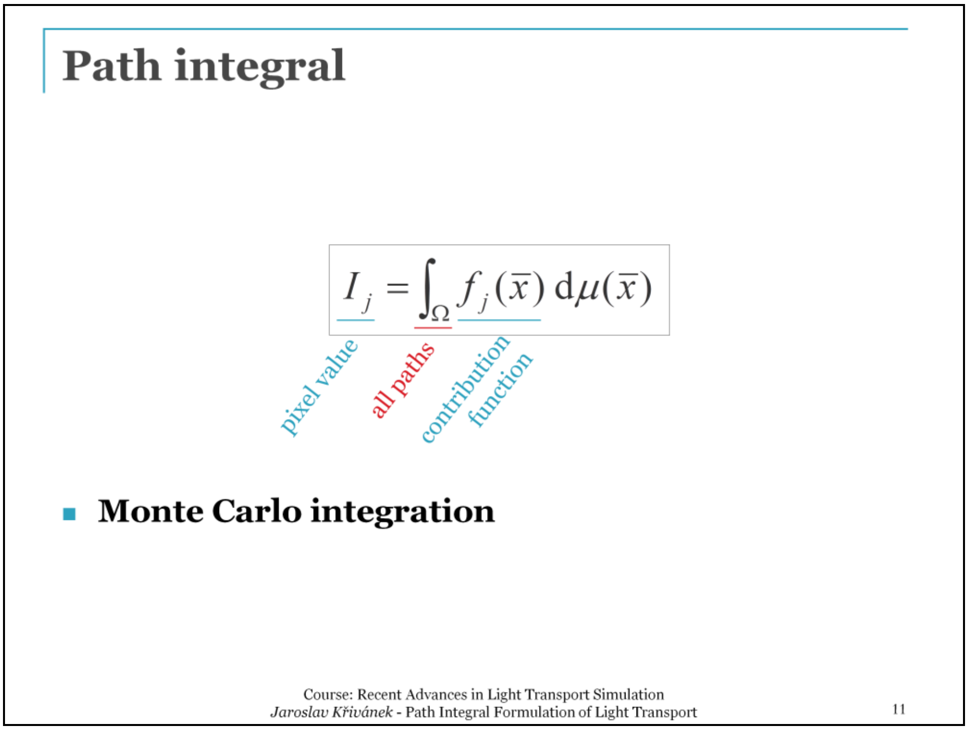
- The path integral actually hides an infinite sum over all possible path lengths.
- Each summand of this sum is a multiple integral, where we integrate the path contribution over all possible positions of the path vertices.
- So each of the integrals is taken over the Cartesian product of the surface of the scene with itself, taken k+1 times (=number of vertices for a length-k path.)
- The integration measure is the product of the measures for each vertex: an area measure, i.e. the natural surface area, for vertices on surfaces, and the natural volume measure for vertices in media.



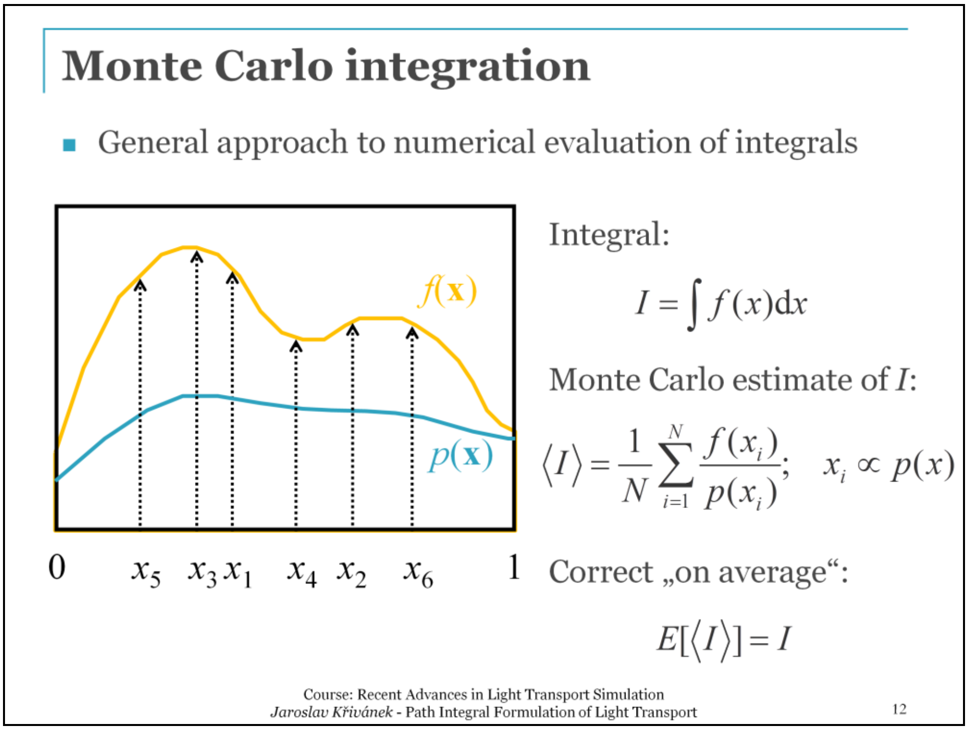
- We now have a formula for pixel values that has a form of a simple (though infinite-dimensional) integral.
- Rendering and images is nothing but a numerical evaluation of this integral for all image pixels.



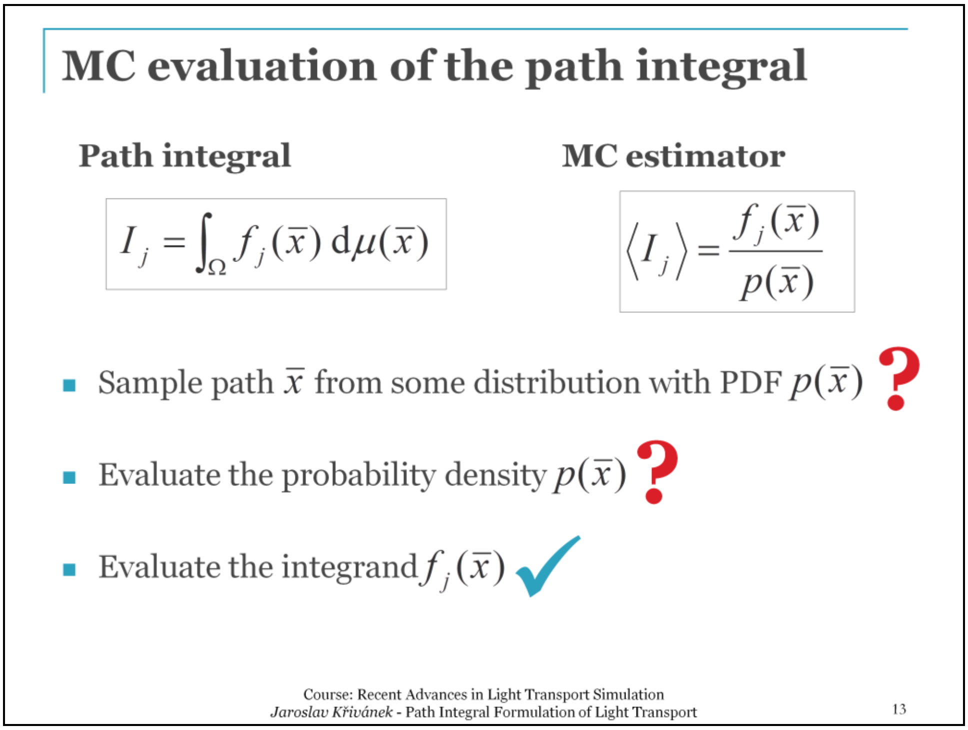
• The next section of this presentation well be devoted to numerical evaluation of the path integral.



- We will use Monte Carlo integration for this purpose because that is the most flexible and general numerical integrations scheme.



- Let me briefly review the basic elements of MC integration.
- Suppose we are given a real function *f*(*x*) and we want to compute the integral Int *f*(*x*) d*x* over some domain (in this example we use the interval [0,1] for simplicity, but the domain can be almost arbitrary.)
- The Monte Carlo integration procedure consists in generating a ‘sample’, that is, a random *x*-value from the integration domain, drawn from some probability distribution with probability density *p*(*x*). In the case of the path integral, the *x*-value is an entire light transport path.
- For this sample x_i, we evaluate the integrand $f(x_i)$, and the probability density $p(x_i)$.
- The ratio $f(x_i)$ / $p(x_i)$ is an estimator of the integrand. To make the estimator more accurate (i.e. to reduce its variance) we repeat the procedure for a number of random samples x_1, x_2, …, x_N , and average the result as shown in the formula on the slide.
- This procedure provides an unbiased estimator of the integrand, which means that “on average”, it produces the correct result i.e. the integral that we want to compute.



- Thanks to the formal simplicity of the path integral formulation, applying Monte Carlo integration is really a more-or-less mechanical process.
- For each pixel, we need to repeatedly evaluate the estimator shown at the top right of the slide and average the estimates.
- This involves the following three steps:
    - First, we need to draw (or sample, or generate – all are synonyms) a random light transport path *x* in the scene (connecting a light source to the camera).
    - Then we need to evaluate the probability density of this path, and the contribution function.
    - Finally, we simply evaluate the formula at the top of the slide.
- Evaluating the path contribution function is simple – we have an analytic formula for doing this that takes a path and returns a number (or an RGB value) – the path contribution.
- However, we have not discussed so far how paths can be sampled and how the PDF of the resulting path can be evaluated.

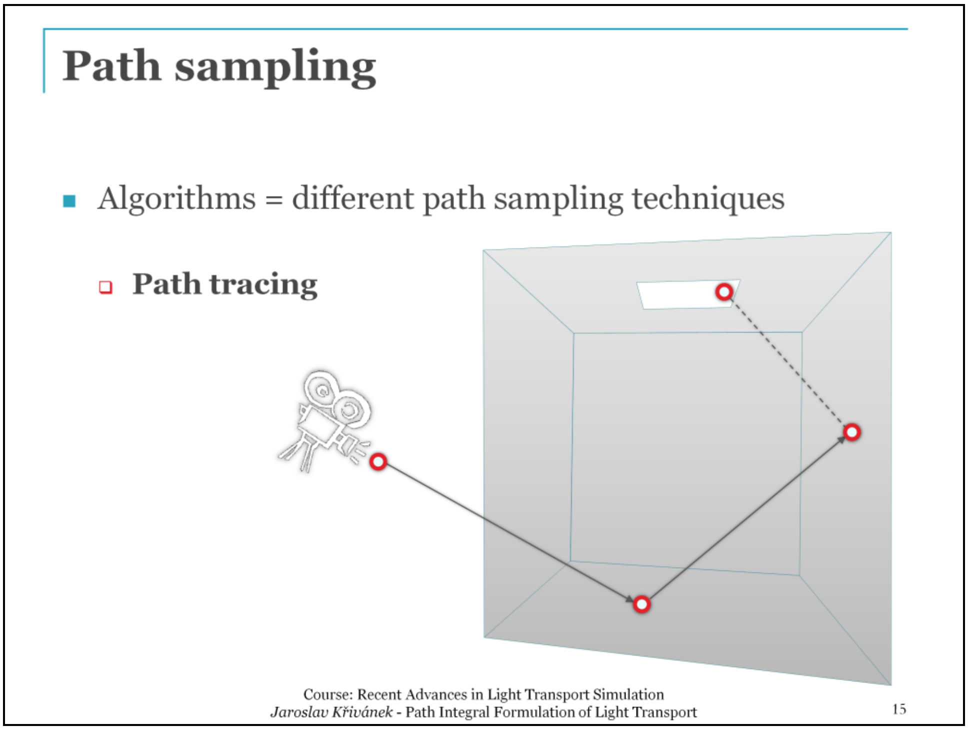
Path sampling

- Algorithms = different path sampling techniques

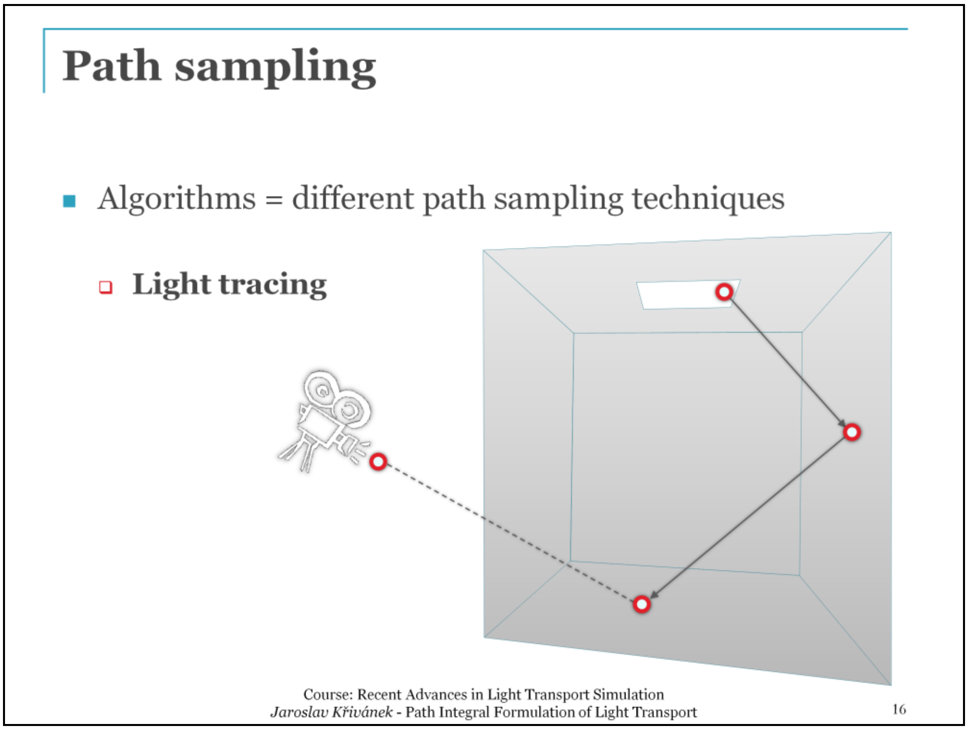
Course: Recent Advances in Light Transport Simulation
Jaroslav Křivánek - Path Integral Formulation of Light Transport 14

• Path sampling techniques and the induced path PDF are an essential aspect of the path integral framework.

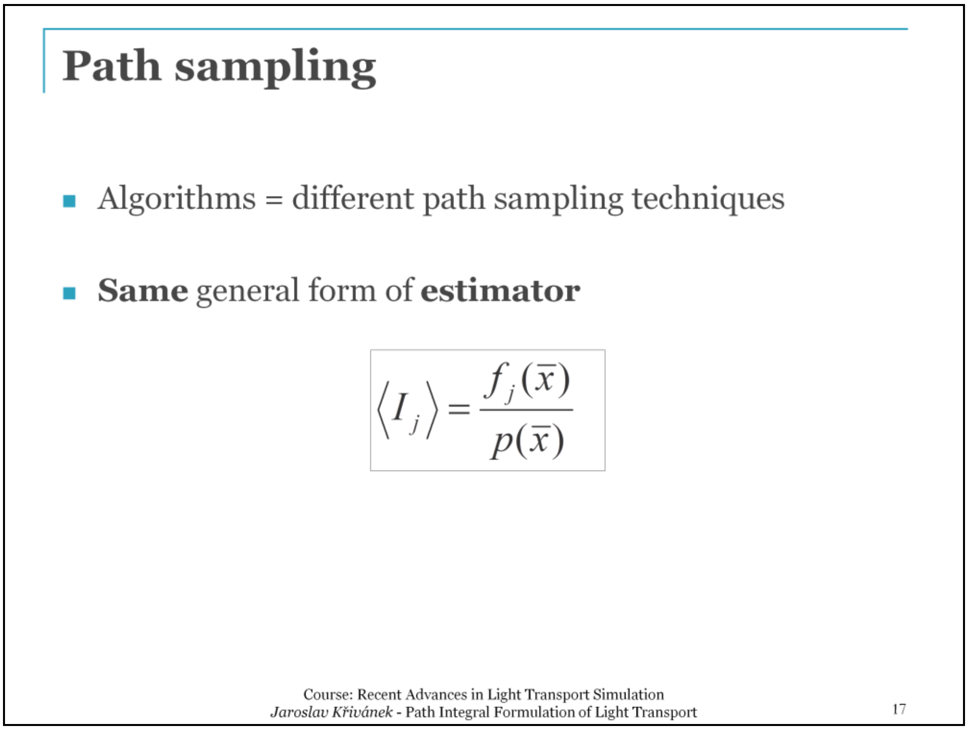
• In fact, from the point of view of the path integral formulation, the only difference between many light transport simulation algorithms are the employed path sampling techniques and their associated PDFs.



• For example, path tracing samples paths by starting at the camera sensor, and extending the path by BRDF importance sampling, and possibly explicitly connecting a to a vertex sampled on the light source.

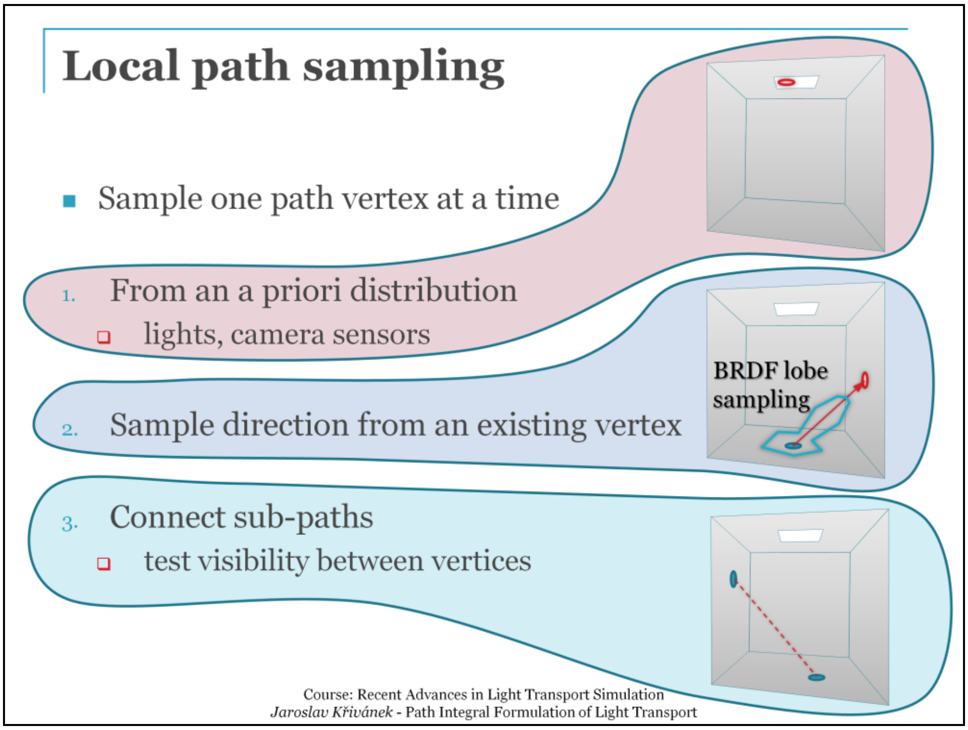


• Light tracing, on the other hand, starts paths at the light sources.

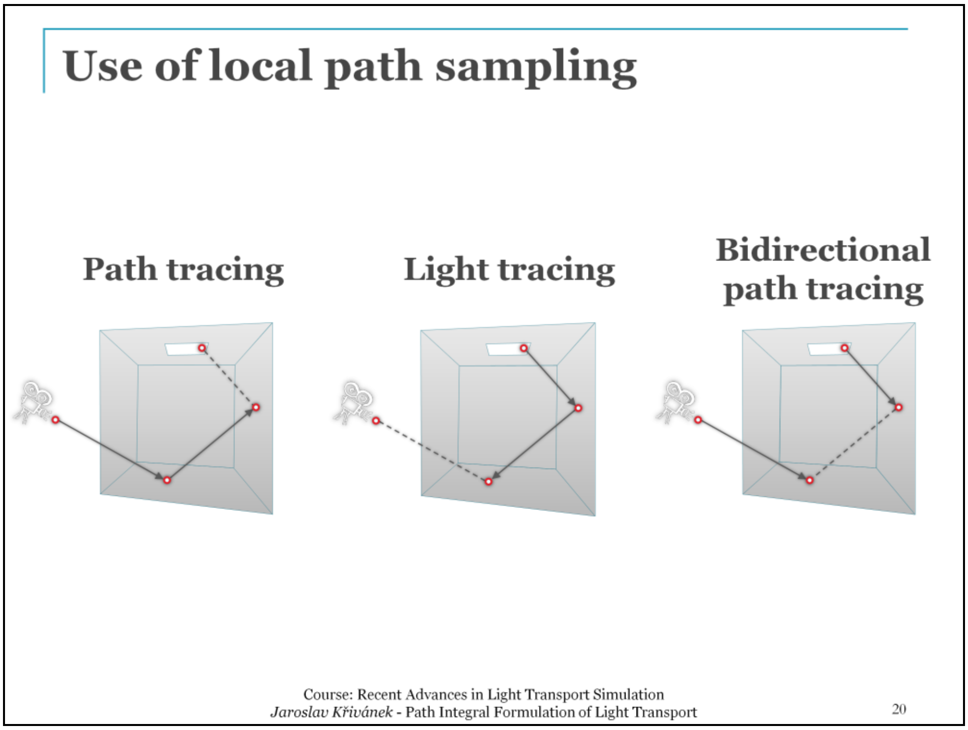


- Though path tracing and light tracing may seem to be very different algorithms, from the path integral point of view they are essentially the same.
- The only difference is the path sampling procedure and the associated path PDF.
- But the general form of the Monte Carlo estimator is exactly the same – only the PDF formula changes.
- Without the path integral framework, we would need equations of importance transport to formulate light tracing, which can get messy.

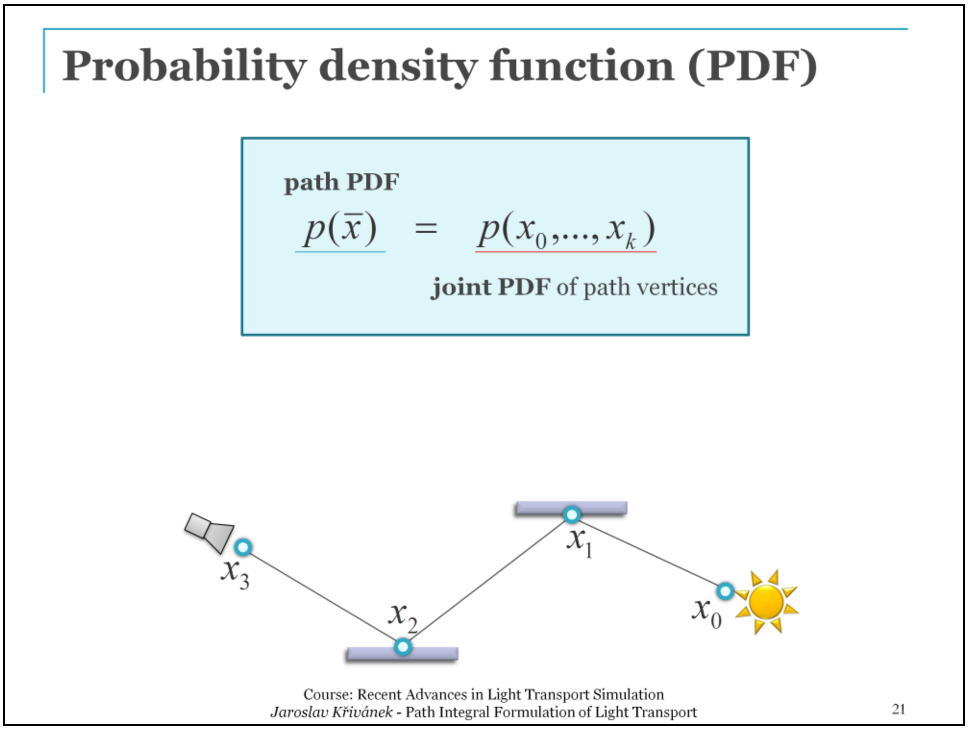
• So how exactly do we sample the paths and how do we compute the path PDF?



- Many practical algorithms rely on local path sampling, where paths are build by adding one vertex at a time until a complete path is built.
- There are three common basic operations.
  - First, we can sample a path vertex from an a priori given distribution over scene surfaces. We usually employ this technique to start a path either on a light source or on the camera sensor.
  - Second, given a sub-path that we've already sampled with a vertex at its end, we may sample a direction from that vertex, and shoot a ray in this direction to obtain the next path vertex. We usually use BRDF-importance sampling (on surfaces) or phase function importance sampling (in media) to choose the next direction.
  - Finally, given two sub-paths, we may connect their end-vertices to form a full light transport path. This technique actually does not add any vertex to the path. It is more or less a simple visibility check to see if the contribution function of the path is non-zero.



• These basic operations are used to construct paths in path tracing, light tracing or bidirectional path tracing.



• Now that we know how to construct a path, we need to evaluate its PDF so that we can plug it into the MC estimator.

• In general the PDF of a light path is simply the **joint** PDF of the path vertices.

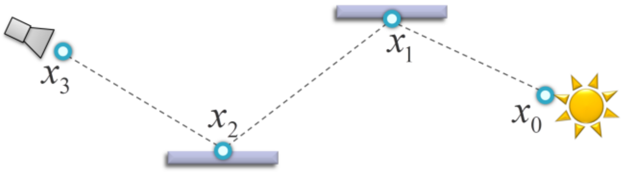
• That is to say, the PDF that the first vertex is where it is *and* the second vertex is where it is, etc.

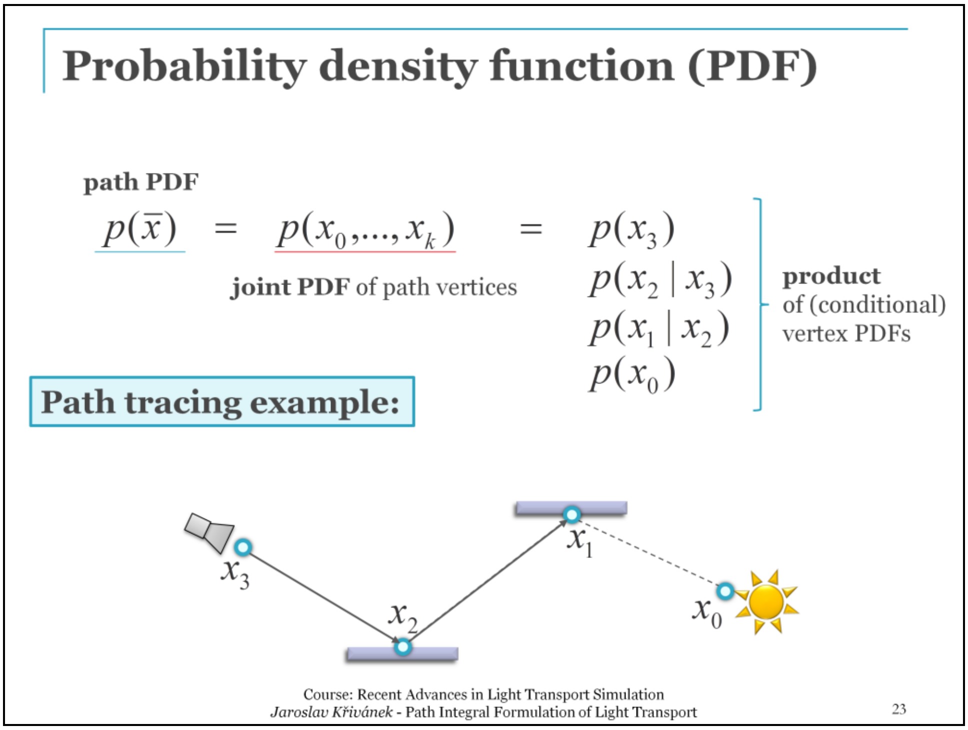
Probability density function (PDF)

path PDF

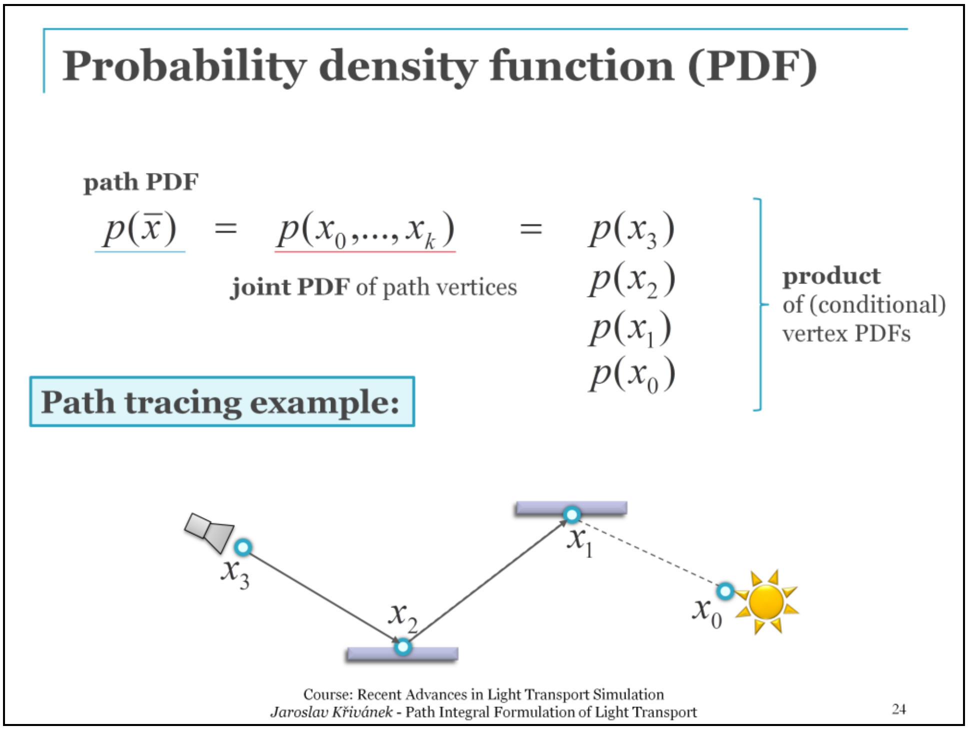
$$p(\bar{x}) = p(x_0, \ldots, x_k)$$

joint PDF of path vertices





- The joint path PDF (as any other joint PDF) can be factorized into the product of the conditional vertex PDF.
- To see what this means, let us take the example of path tracing, where we build a path starting from the camera.
- Vertex x_3 comes from an a priori distribution $p(x_3)$ over the camera lens (usually uniform; or the delta distribution for a pinhole camera).
- Vertex x_2 is sampled by generating a random direction from x_3 and shooting a ray. This induces a PDF for x_2, $p(x_2 \mid x_3)$, which is in fact conditional on vertex x_3.
- The same thing holds for vertex x_1, which is sampled by shooting a ray in a random direction from x_2.
- Finally, vertex x_0 on the light source might be sampled from an uniform distribution over the light source area with pdf $p(x_0)$, independently of the other path vertices.
- The full joint PDF is given by the product of all these individual terms.



• It is customary to simplify this somewhat pedantic notation and leave out the conditional signs. Nonetheless, it is important to keep in mind that the path vertex PDFs for vertices that are not sampled independently are indeed conditional PDFs.

MC evaluation of the path integral

Path integral

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

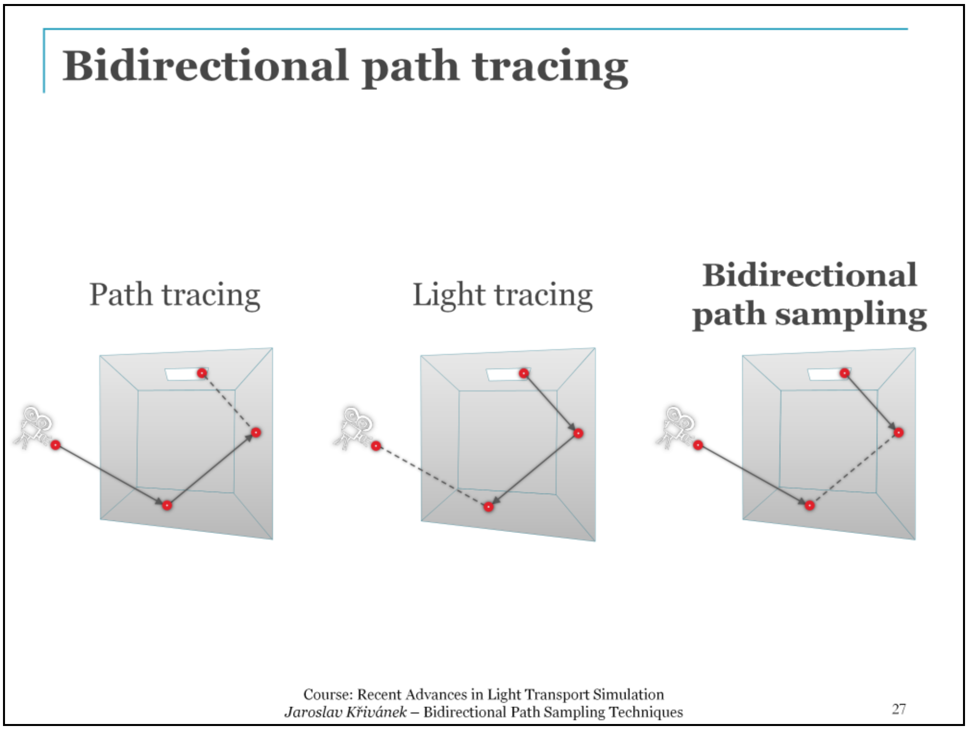
MC estimator

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

- Sample path $\bar{x}$
- Evaluate the probability density $p(\bar{x})$
- Evaluate the integrand $f_j(\bar{x})$

• Going back to this slide, we see that we now have all the elements to evaluate the MC estimator on the top right.

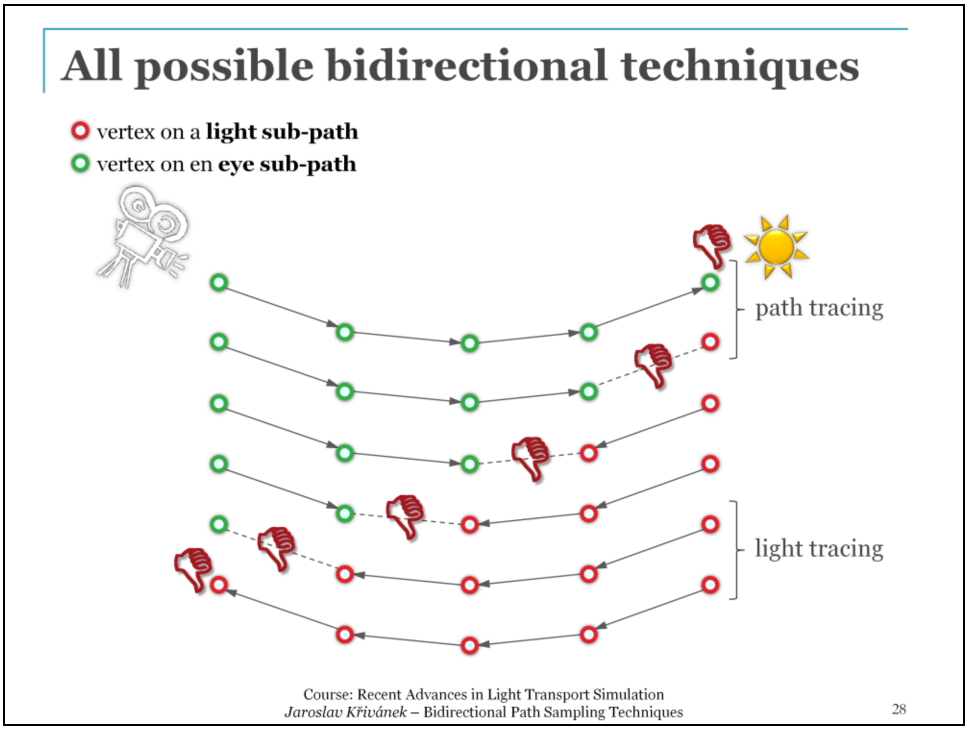
- Now that we understand the path sampling techniques and their associated probability densities, we have all the ingredients necessary to construct a bidirectional path tracer.
- We will see that with a proper understand of the path integral formulation, this less complicated than it might seem.



• Bidirectional path tracing is based on combining many different path sampling techniques.

• It uses the path sampling from a path tracer and a light tracer.

• And it adds the bidirectional path sampling techniques, where a full path is built by connecting a sub-path from the camera and from the light source.



• In fact, one single path can be sampled with the local sampling techniques in many different ways.

• This slide schematically shows all the possible bidirectional techniques that we can obtain by starting a path either on a light source or on the camera for an example path of length 4.

• The first two cases correspond to what a regular path tracer usually does (randomly hitting the light sources and explicit light source connections.)

• The last two are complementary to the first two and therefore correspond to light tracing.

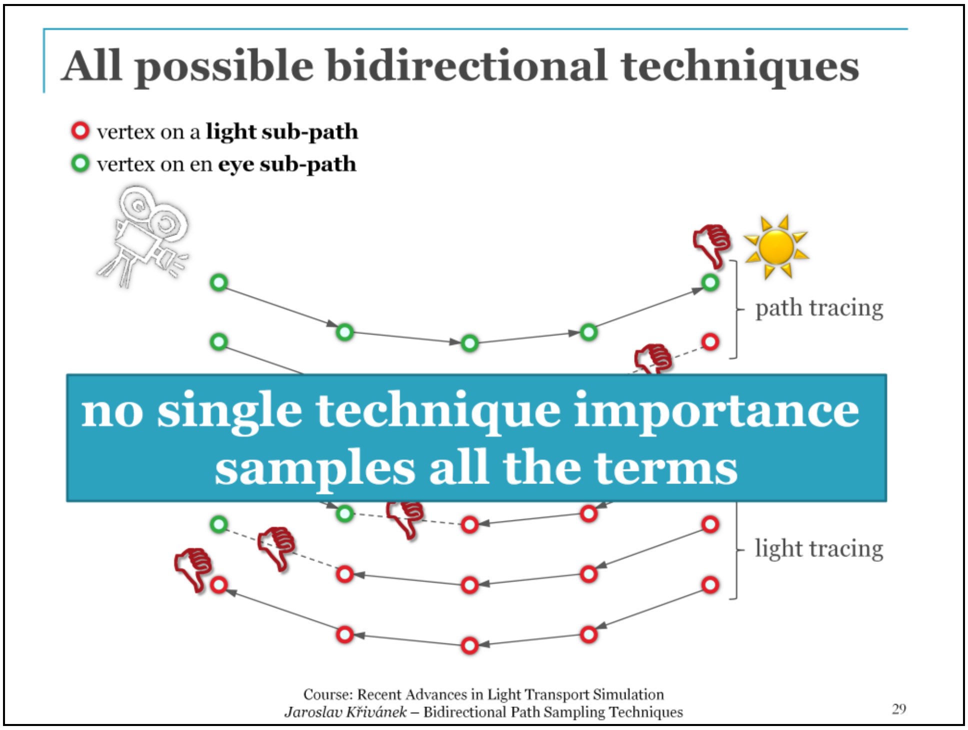
• The techniques in between are the bidirectional techniques where a sub-path sampled from the camera is connected to a sub-path sampled from a light source. The different bidirectional techniques correspond to the different number of vertices at the camera and light sub-paths.

• Each sampling technique importance samples a different subset of terms of the measurement contribution function.

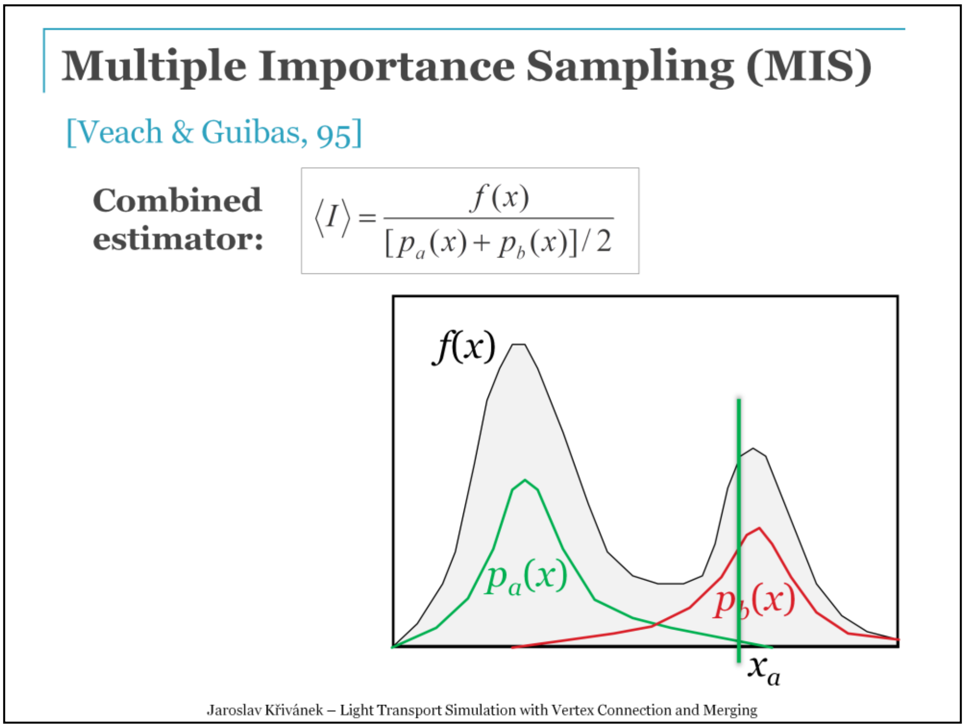
• However, in each of these techniques, there are some terms of the measurement contribution function that are not importance sampled.

• The purely unidirectional techniques (top and bottom) do not importance sample the light emission and sensor sensitivity, respectively. Indeed, for example the technique at the top relies on randomly hitting a light source, without incorporating any information about the location of light sources in the scene.

• All the bidirectional techniques, that is, those that involve connection of two sub-paths, are usually unable to importance sample the terms associated with the connection edge.



- So we have many different techniques, but none of them is able to importance sample all of the terms of the measurement contribution function.
- But because each technique fails at importance sampling a different term of the contribution function, they have complementary strong points and weaknesses.
- It is exactly for this reason that bidirectional path tracing is built around the idea of taking the best of each of the techniques by combining them together using Multiple Importance Sampling.



• This slide shows this situation (multiple sampling techniques, each failing in a different way) in a simpler setting.

• We have a multimodal integrand $f(x)$ that we want to numerically integrate using a MC method with importance sampling.

• Unfortunately, we do not have a PDF that would mimic the integrand in the entire domain.

• Instead, we can draw the sample from two different PDFs, p_a and p_b, each of which is a good match for the integrand under different conditions – i.e. in different part of the domain.

• However, the estimators corresponding to these two PDFs have extremely high variance.

• To see why the variance can be so high, take the example of the sample x_a taken from the tail of PDF p_a, where $f(x)$ might still be large.

• The value of the estimator, $f(x) / p_a(x)$, for this sample divides the large $f(x)$ by the small $p_a(x)$, producing an outlier.

• We can use Multiple Importance Sampling (MIS) to combine the sampling techniques corresponding to the two PDFs into a single, robust, combined technique.

• The MIS procedure is extremely simple: it randomly picks one distribution to sample from (p_a or p_b , say with a fifty-fifty chance) and then takes the sample from the selected distribution.

• This essentially corresponds to sampling from a weighted average of the two distributions, which is reflected in the form of the estimator, shown on the slide.

• This combined estimator is really powerful at suppressing outlier samples.

• Going back the “green” sample x_a , we see that the combined technique has a much higher chance of producing this particular x (because it can sample it also from p_b), so the combined estimator divides $f(x)$ by $[p_a(x) + p_b(x)] / 2$, which yields a much more reasonable sample value.

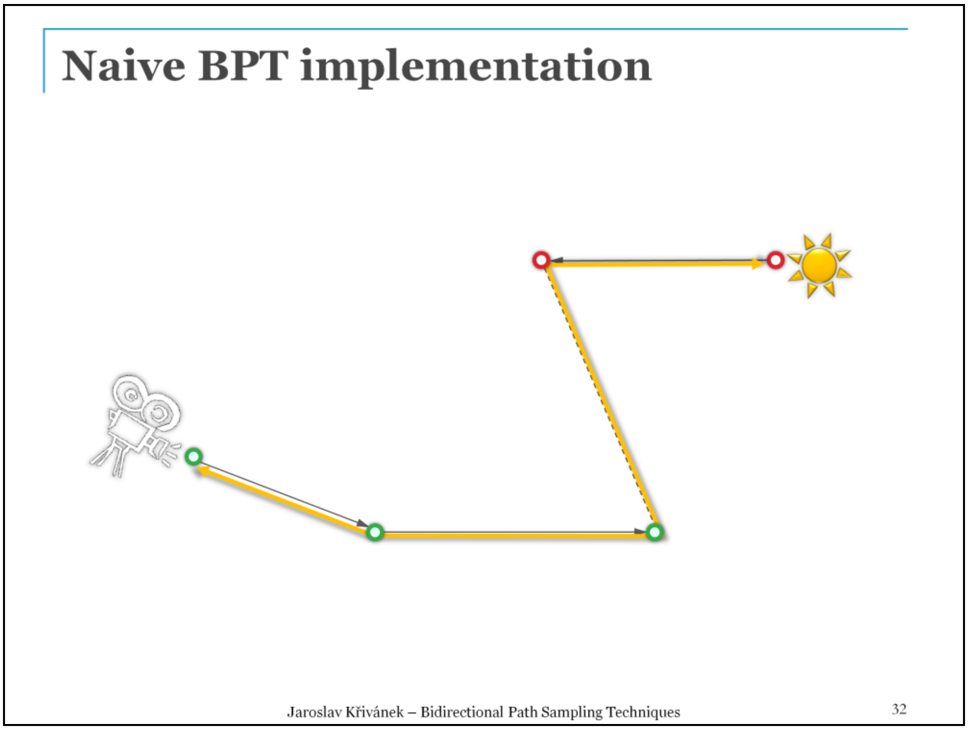
• I want to note that what I’m showing here is called the “balance heuristic” and is a part of a wider theory of Multiple Importance Sampling on weighted combinations of estimators proposed by Veach and Guibas [1995].

• This year, Eric Veach was awarded an Academy Award for this an other works on robust light transport simulation. (19 years after the publication – this is how far ahead of time his contribution was!).

Bidirectional path tracing

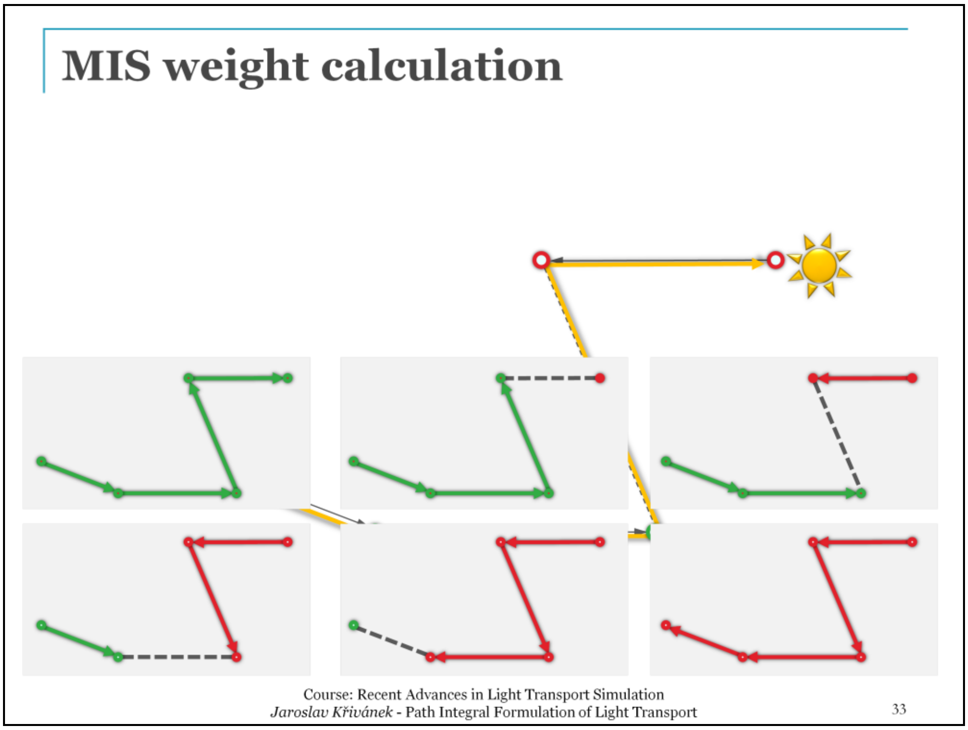
- Use **all** of the above sampling techniques
- Combine using **Multiple Importance Sampling**

• The main idea of bidirectional path tracing is to use all of the sampling techniques above and combine them using multiple importance sampling.

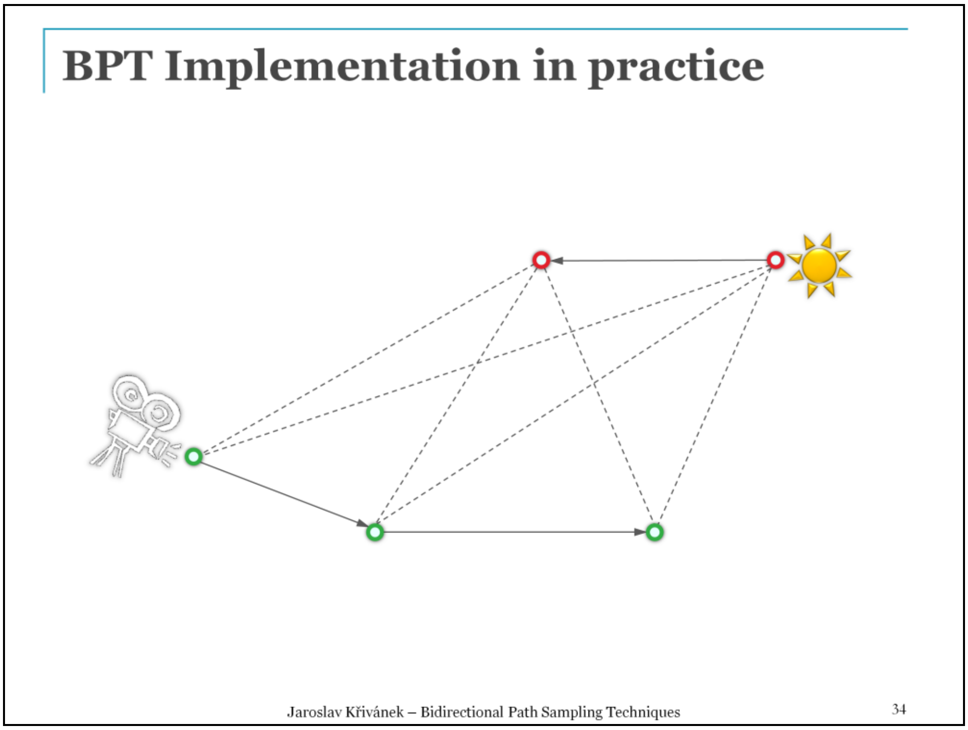


• A basic implementation of BPT samples two independent sub-path, one from the light source and another form the camera.

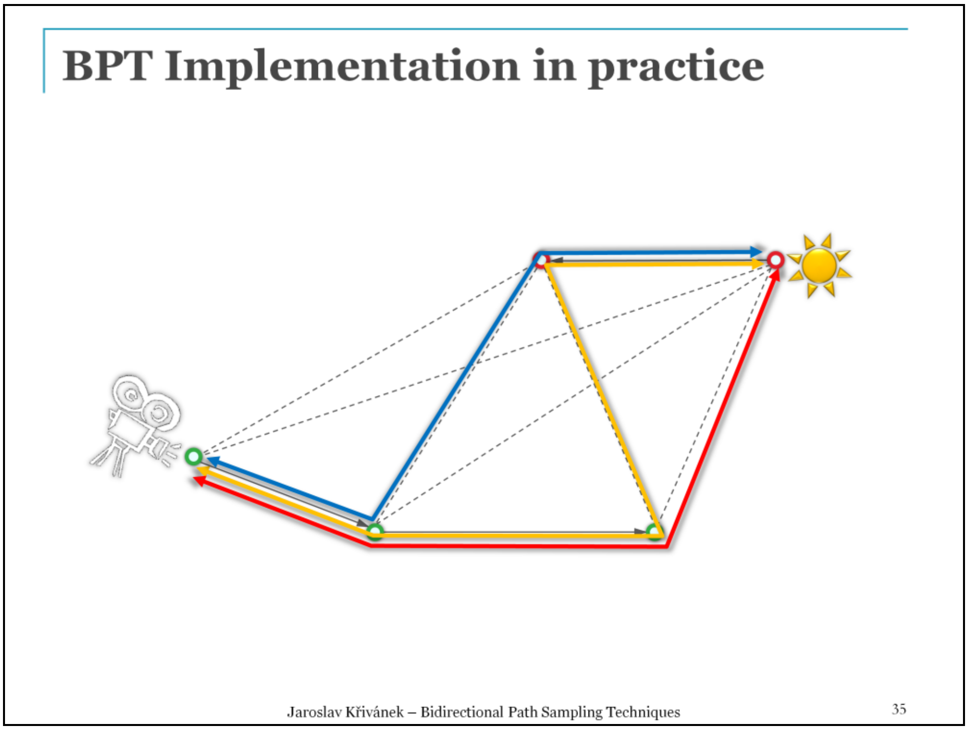
• After that, the end points of these sub-paths are connected to generate a full path.



- Because the length of the camera and light sub-path is selected randomly, this procedure has a non-zero probability for generating this exact same path in many different ways, all shown in the boxes at the bottom.
- To evaluate the MIS weight of the generated path, we need to calculate the PDF of all these alternative sampling techniques, and plug them into the MIS formula.
- This is fairly straightforward with the PDF formulas given earlier.
- That’s really all for BPT – the basic idea is very simple.



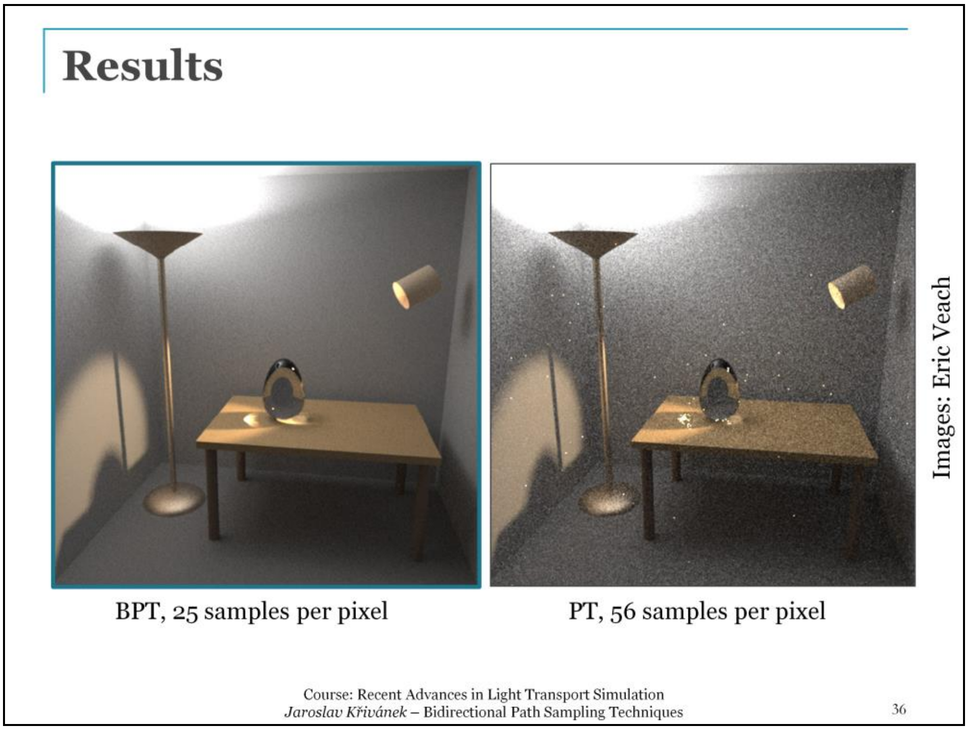
- To speed up the calculation, in practice each vertex from the first sub-path is connected to each vertex of the second sub-path.



• This generates a full family of paths, some of which are shown above.

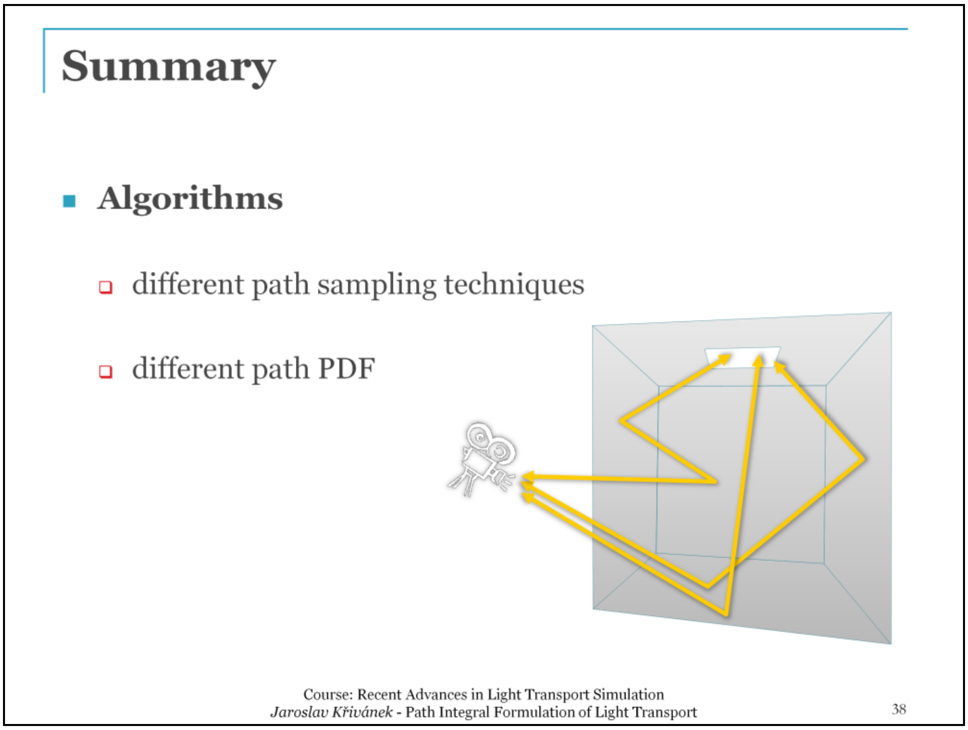
• Nonetheless, on a conceptual level, each of these paths is treated completely independently of all the other ones.

• I really want to stress that this path reuse scheme is just an implementation detail. It does improve the efficiency thanks to the reuse of the sub-paths, but does not contribute to the robustness of BPT in any way.

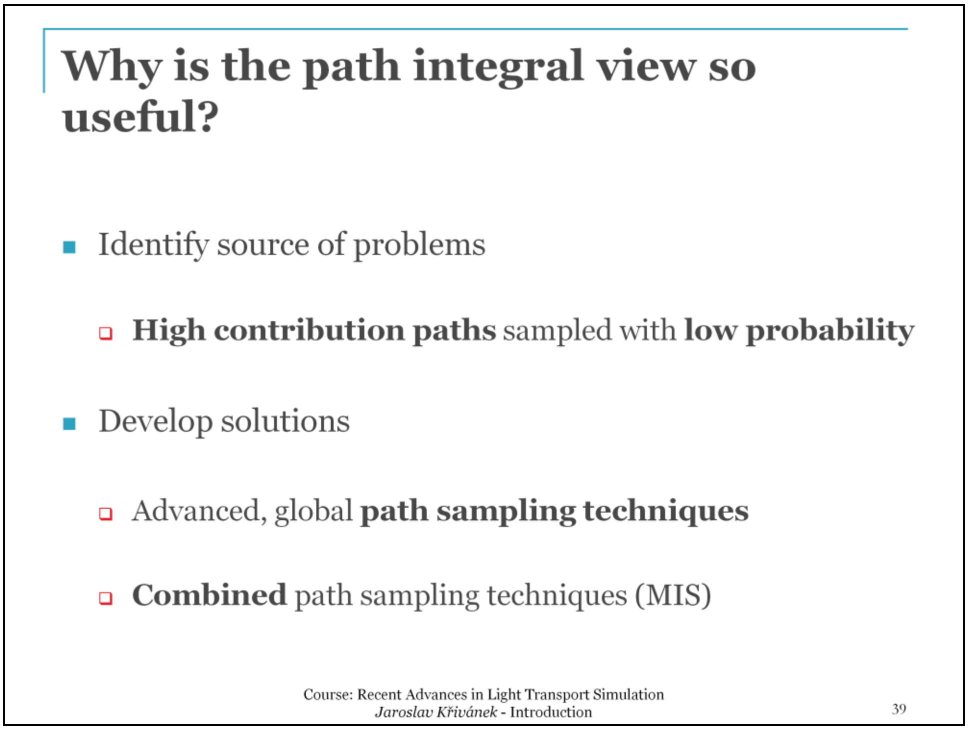


This slide show the famous result of bidirectional path tracing generated by Eric Veach.

NEARLY THERE…



- At this point, we should summarize the development so far.
- From the point of view of the path integral framework, different light transport algorithms (based on Monte Carlo sampling) only differ by the path sampling techniques employed.
- The different sampling techniques imply different path PDF and therefore different relative efficiency for specific lighting effects.



• The path integral framework is extremely useful for several reasons.

• First, it allows us to identify the weaknesses of existing algorithms.

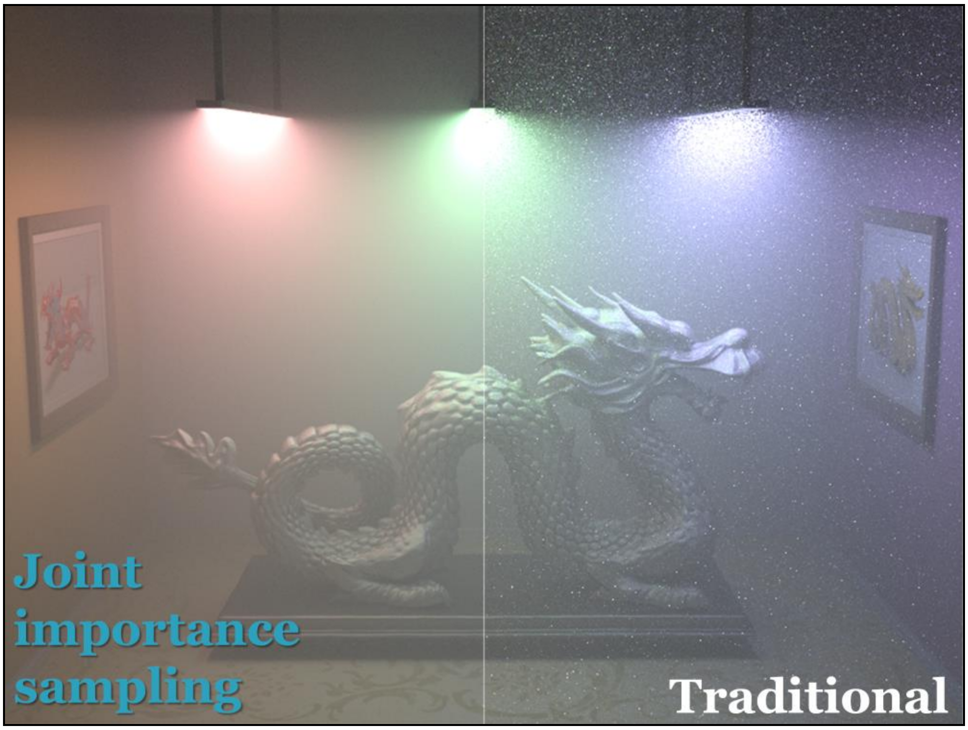
• With a little bit of simplification, we could say that all problems of current light transport solutions boil down to poor path sampling.

• Specifically to the fact that some light transport paths that bring significant amount of energy from the light sources to the camera are not sampled with appropriately high probability.

• This means high estimator variance that produces noise and fireflies in the renderings.

• Second, the path integral framework allows us to develop new light transport algorithms based on advanced, global path sampling techniques, such as Metropolis Light Transport.

• It also provides us with a means of combining different path sampling techniques in a provably good way using Multiple importance sampling.



- As an example, I'd like to mention our work from last year's SIGGRAPH Asia on path sampling in participating media.
- Just by designing path sampling techniques that more closely follows the path contribution function, we were able to reduce the variance more than thousand-fold in certain cases.

THANK YOU!

Time for questions…

Course: Recent Advances in Light Transport Simulation

Jaroslav Křivánek - Path Integral Formulation of Light Transport

Acknowledgements

- **Czech Science Foundation**
  - grant no. P202-13-26189S
- Images
  - **Eric Tabellion**
  - **Marcos Fajardo**