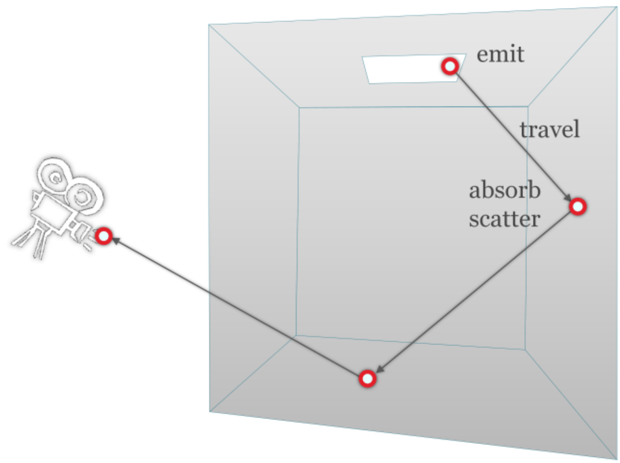
PATH INTEGRAL FORMULATION OF LIGHT TRANSPORT

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

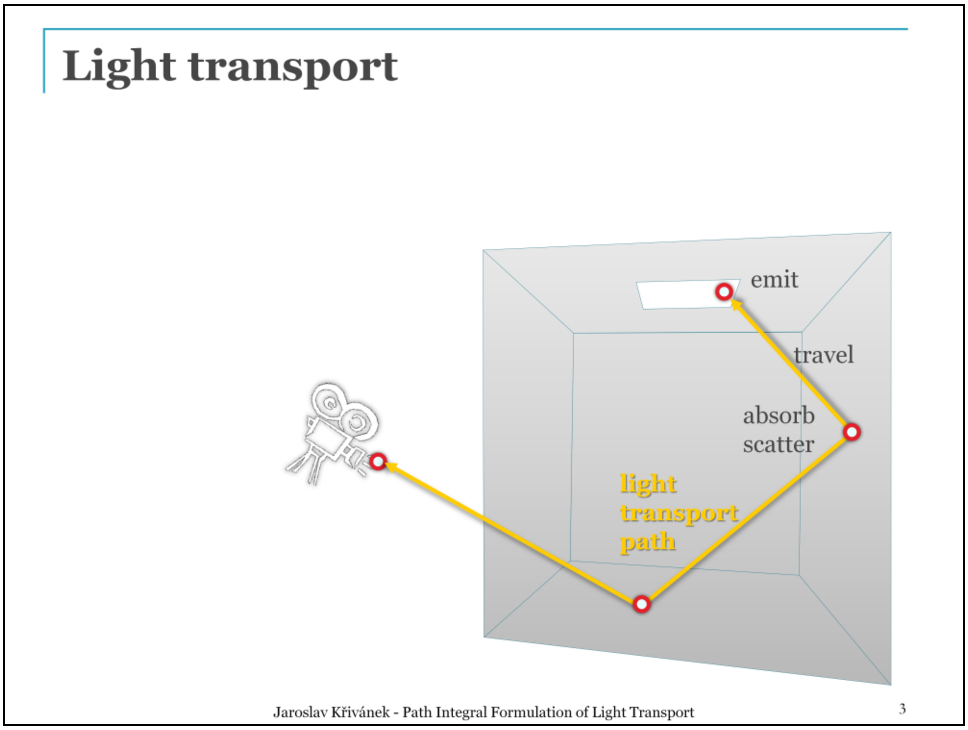
Light transport

- Geometric optics



Jaroslav Křivánek - Path Integral Formulation of Light Transport 2

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

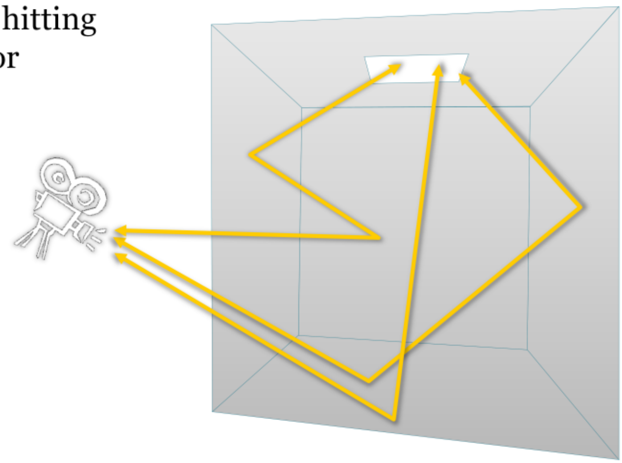


• The light particles travel along trajectories that we call “light transport paths”.

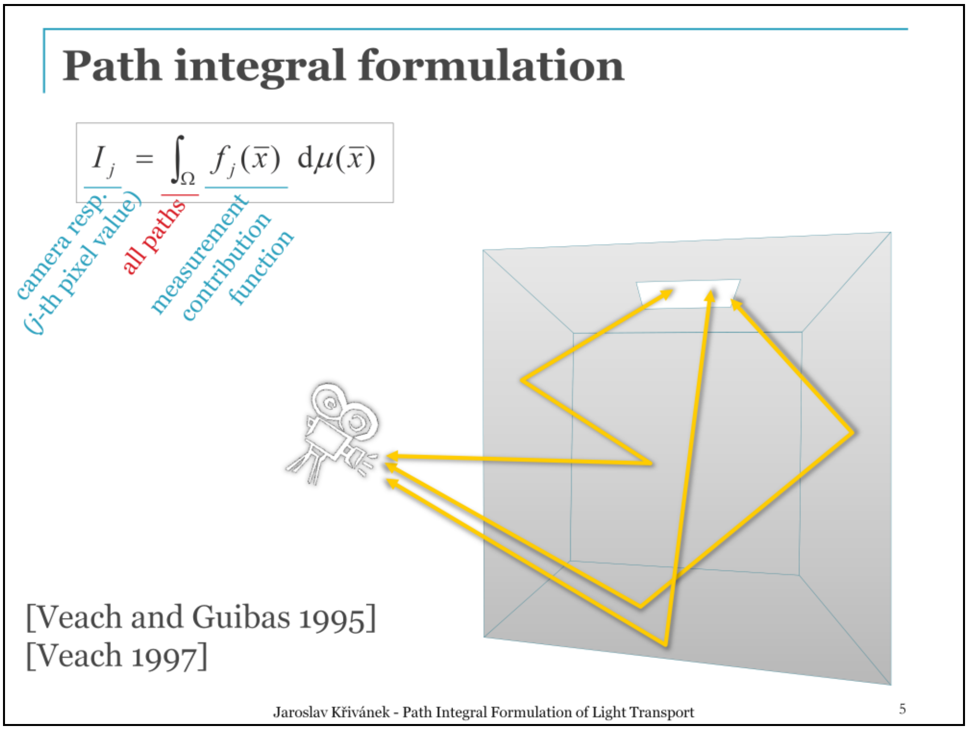
• In an environment consisting of opaque surfaces in vacuum, these paths are polylines whose vertices correspond to scattering (reflection) at surfaces, and the straight edges correspond to light travelling the in free space.

Light transport

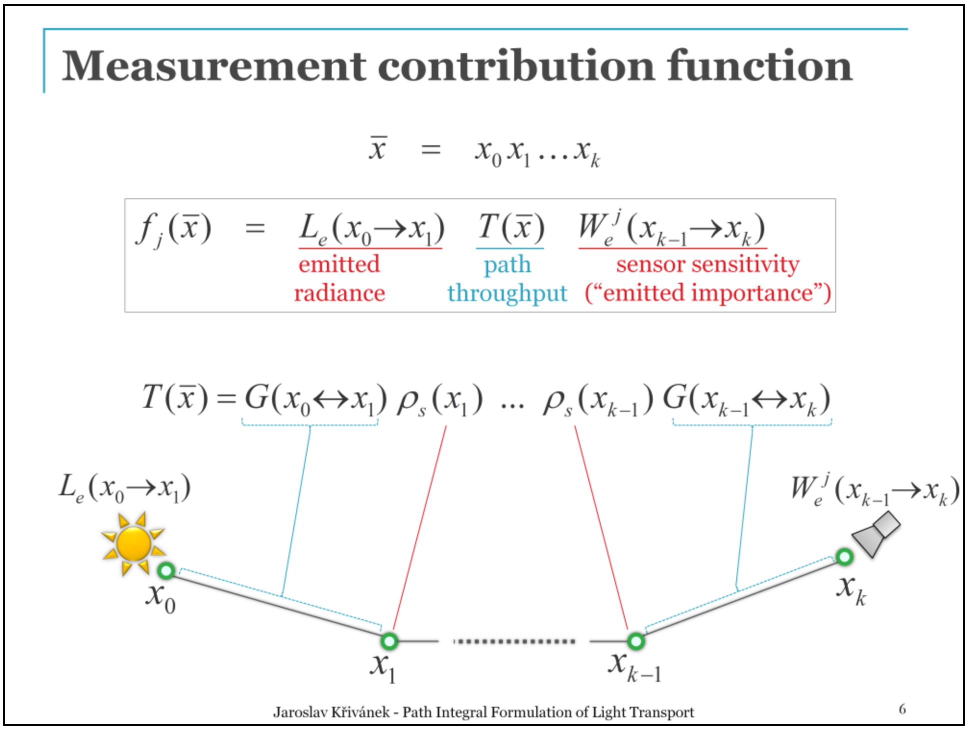
- **Camera response**
  - all paths hitting the sensor



• 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 formalizes this idea by writing the camera response (which, in image synthesis will be the value of a pixel) 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.

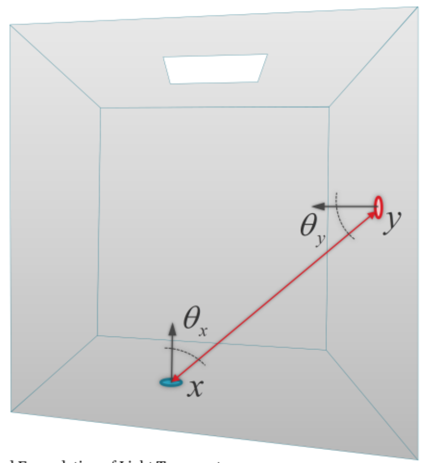
• As I already mentioned, a light transport path is a polyline with vertices corresponding to light scattering on surfaces.

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

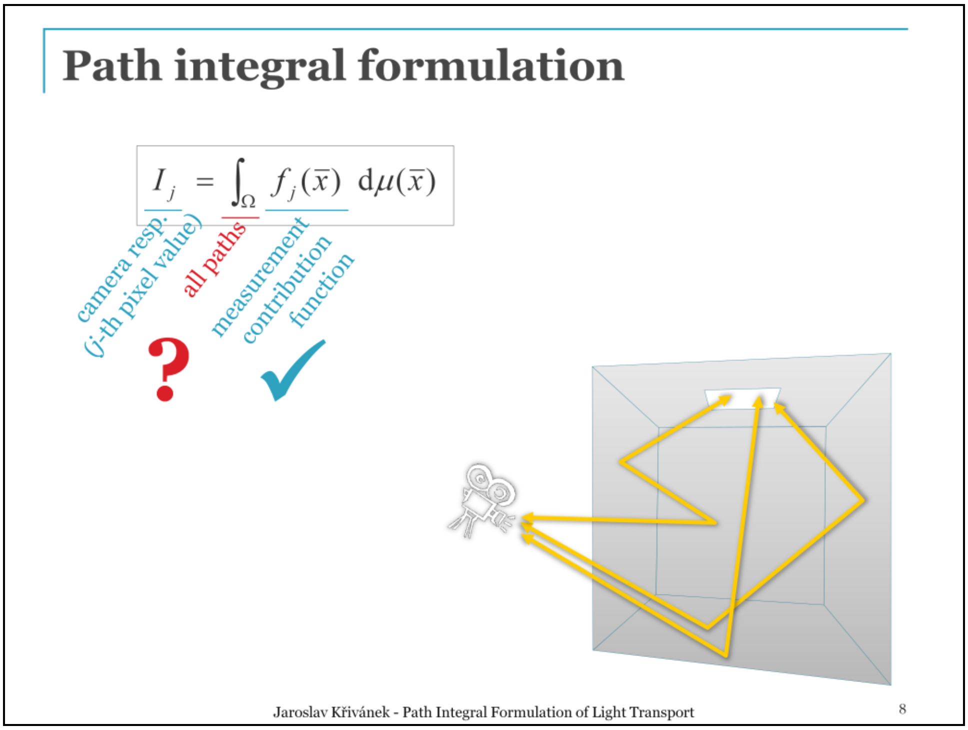
• The measurement contribution function *f*(*x*) for a path *x* of length *k* is defined as the emitted radiance L_e at the first vertex, times the sensitivity (or “emitted importance”) at the last vertex, times the path throughput *T*(*x*). The throughput is defined as the product of the geometric *G* and scattering terms associated with the path edges and interior vertices, respectively.

Geometry term

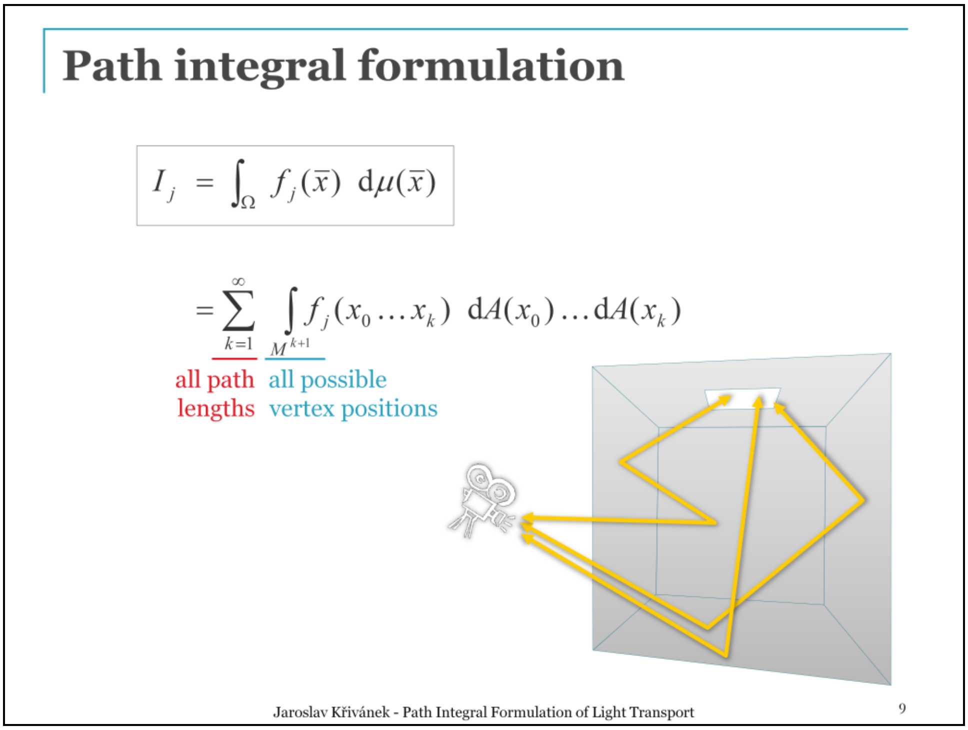
$$G(x \leftrightarrow y) = \frac{|\cos\theta_x||\cos\theta_y|}{\|x-y\|^2} V(x \leftrightarrow y)$$



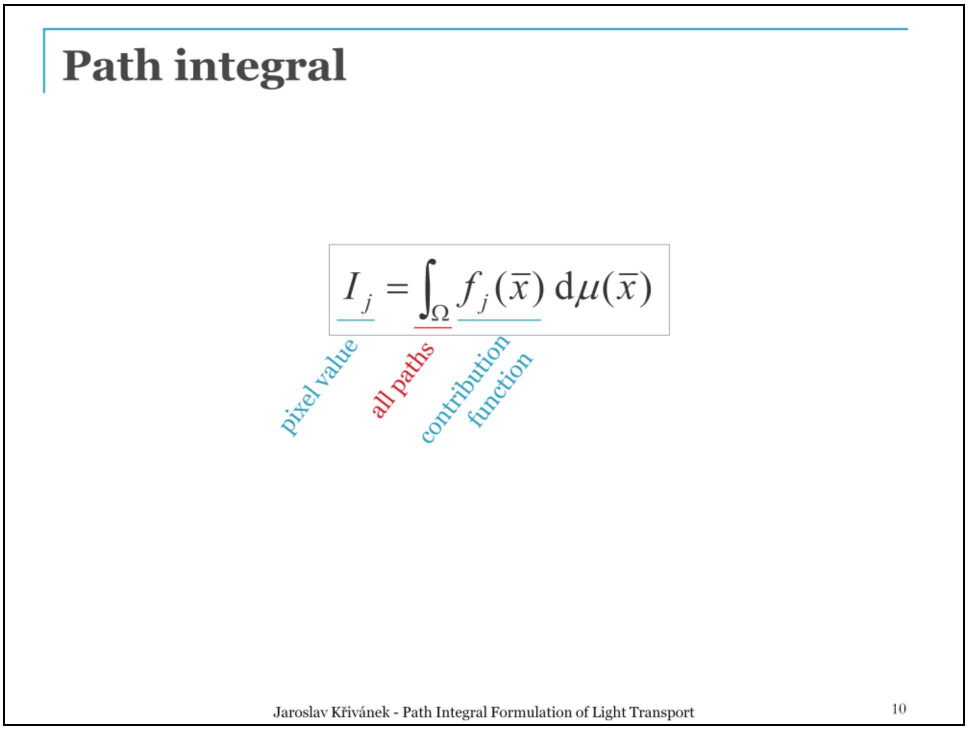
• The geometry term, or point-to-point form factor, of a path edge expresses the throughput of the edge due to geometry configuration of the scene, and is given by the product of the inverse-squared-distance, cosine factor at the vertices and the visibility term.



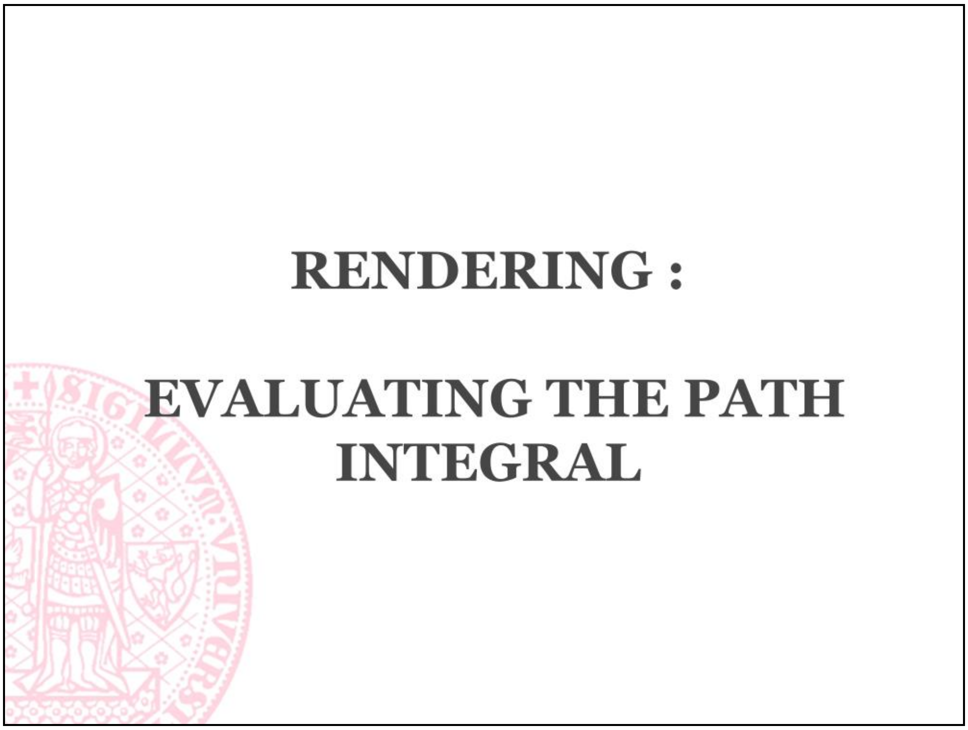
- Back to the path integral…
- We now know the meaning of the integrand – the path contribution function – but the integration domain “all path” needs more clarification.



- 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 area-product measure, i.e. the natural surface area, raised to the power of k+1.



- We now have a formula for pixel values that has a form of a simple (though infinite-dimensional) integral.
- To render images, we need to numerically evaluate this integral for all image pixels.



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

Path integral

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

pixel value

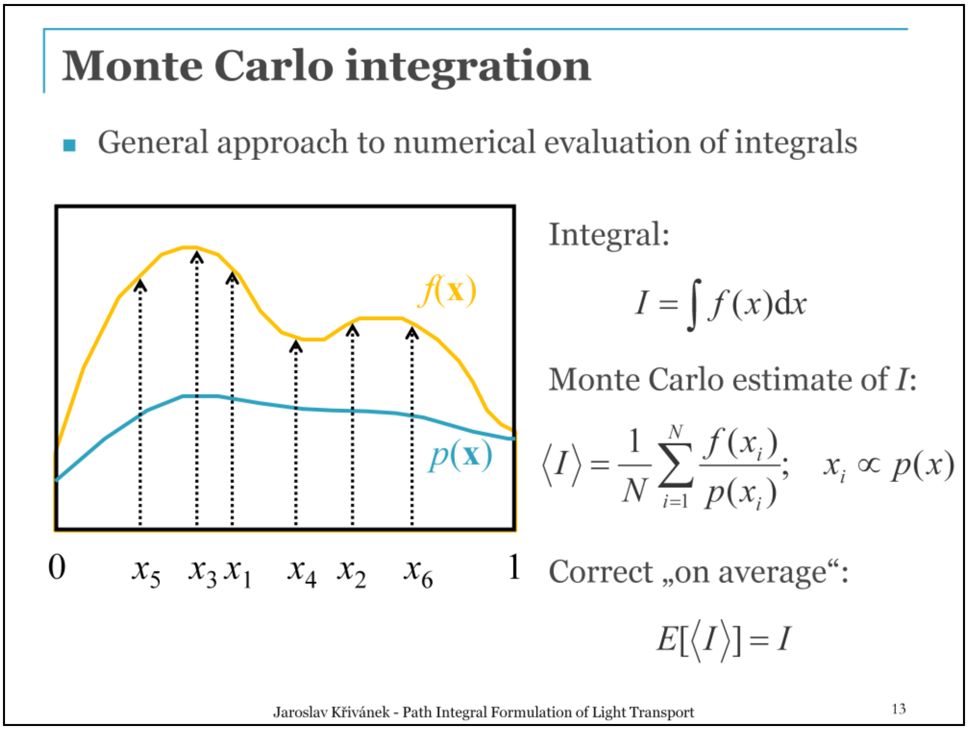
all paths

contribution function

- **Monte Carlo integration**

Jaroslav Křivánek - Path Integral Formulation of Light Transport 12

- We will use Monte Carlo integration for this purpose.



• Let me briefly review the basic elements of MC integration.

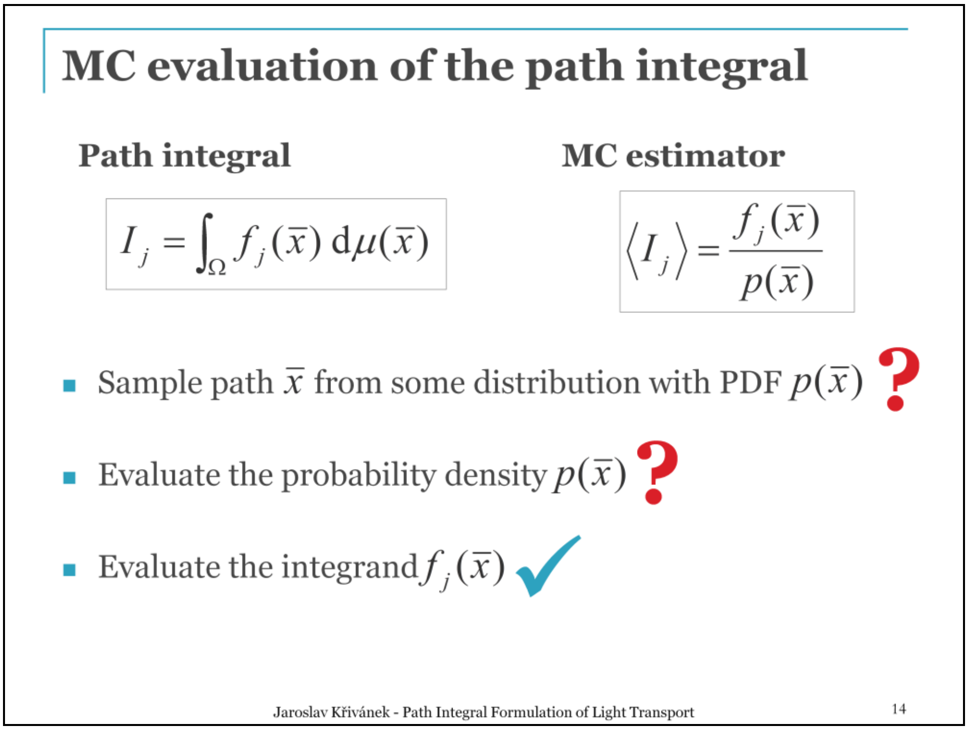
• Suppose we are given a real function *f*(*x*) and we want to compute the integral Int *f*(*x*) d*x* over some domain (in this example we use the interval [0,1] for simplicity, but the domain can be almost arbitrary.)

• 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 path integral, the *x*-value is an entire light transport path.

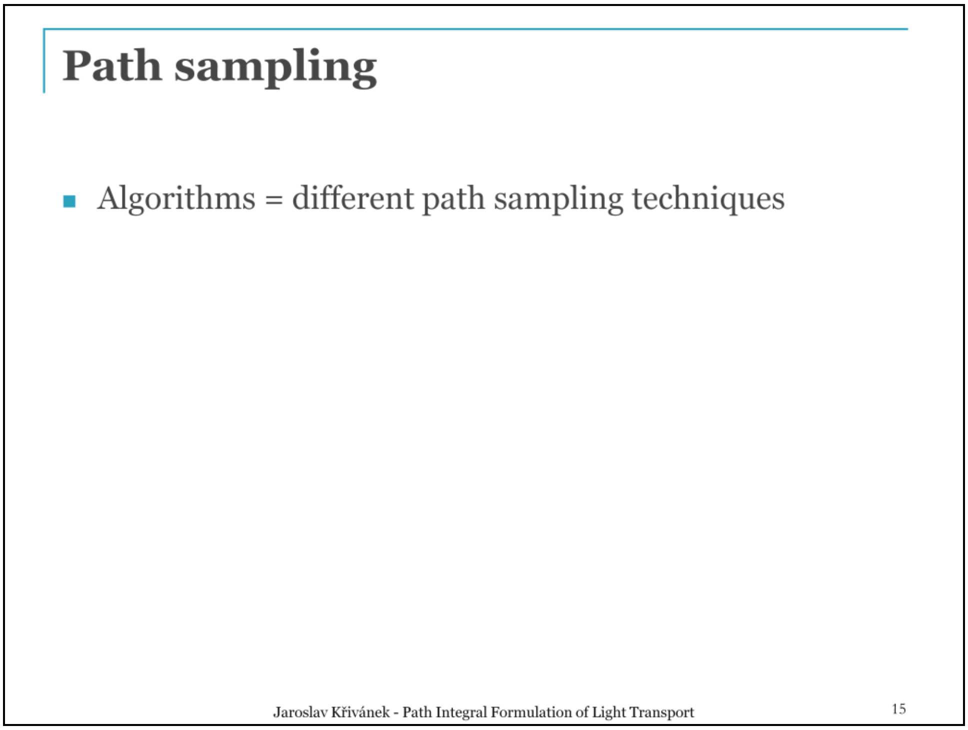
• For this sample x_i, we evaluate the integrand $f(x_i)$, and the probability density $p(x_i)$.

• 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.

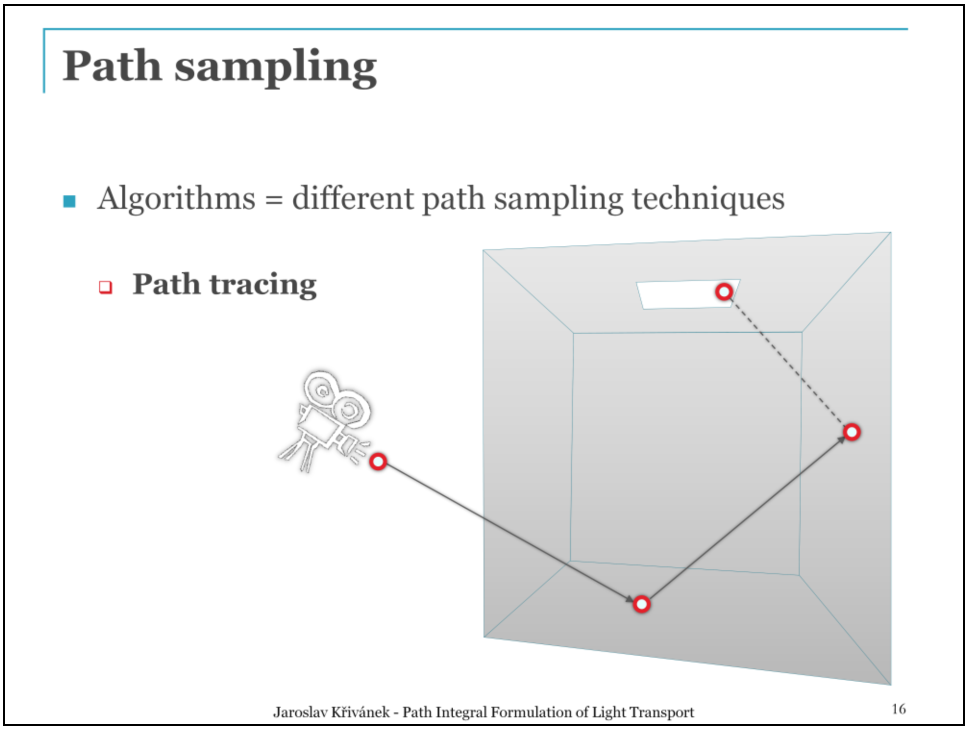
• This procedure provides an unbiased estimator of the integrand, which means that “on average”, it produces the correct result i.e. the integral that we want to compute.



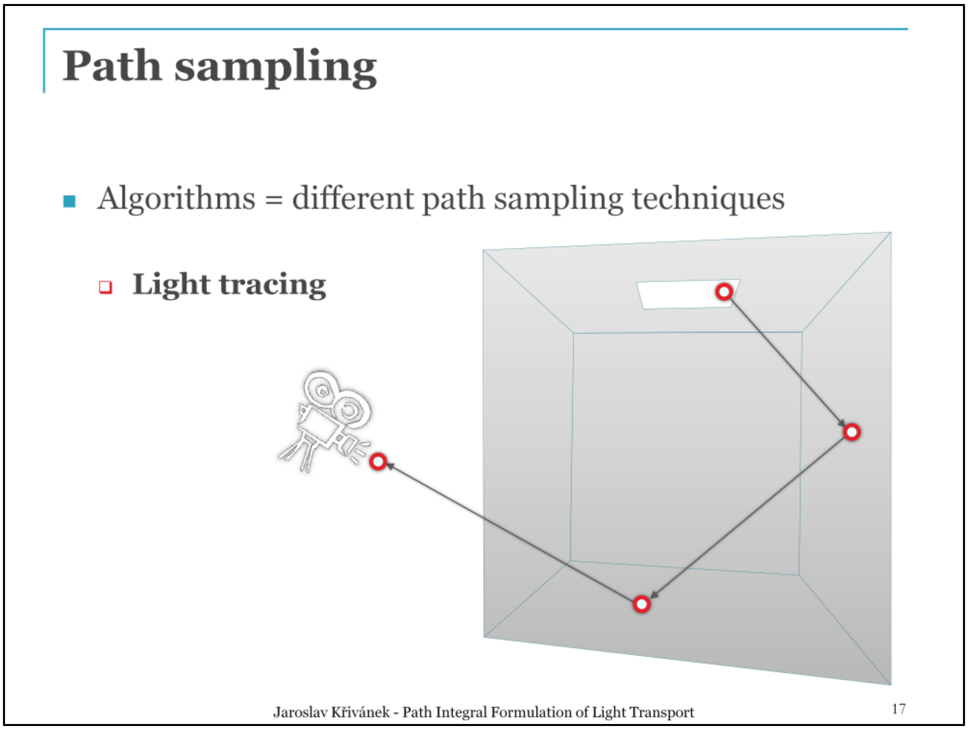
- Thanks to the formal simplicity of the path integral formulation, applying Monte Carlo integration is really a more-or-less mechanical process.
- For each pixel, we need to repeatedly evaluate the estimator shown at the top right of the slide and average the estimates.
- This involves the following 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 this that takes a path and returns a number – 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.



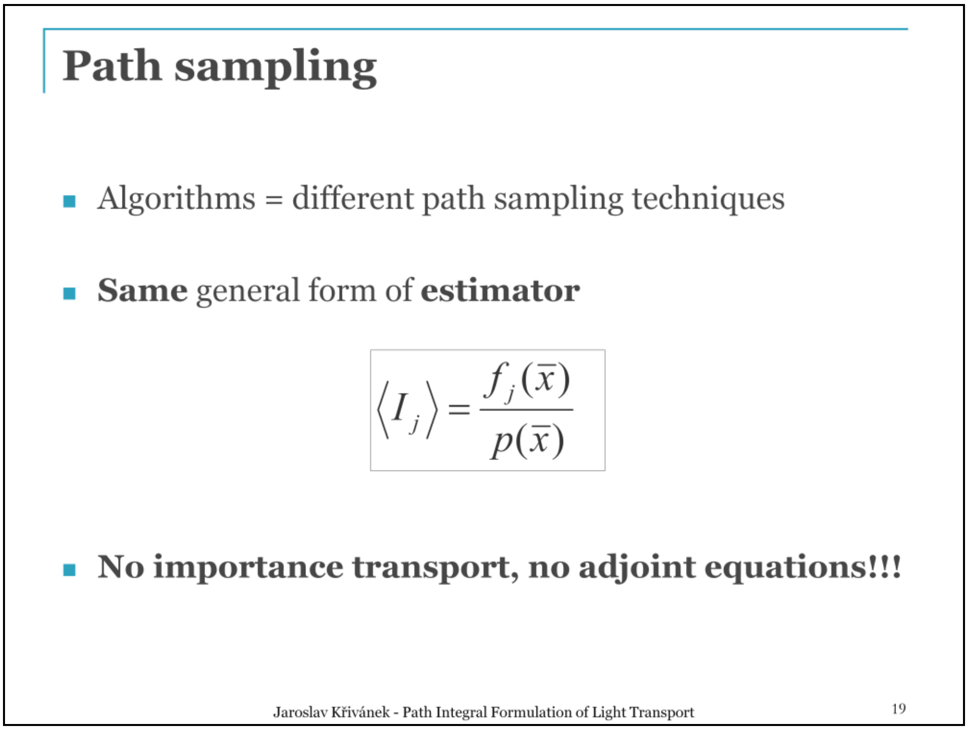
- Path sampling techniques and the induced path PDF are an essential aspect of the path integral framework.
- In fact, from the point of view of the path integral formulation, the only difference between many light transport simulation algorithms are the employed path sampling techniques and their PDFs.



• 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.

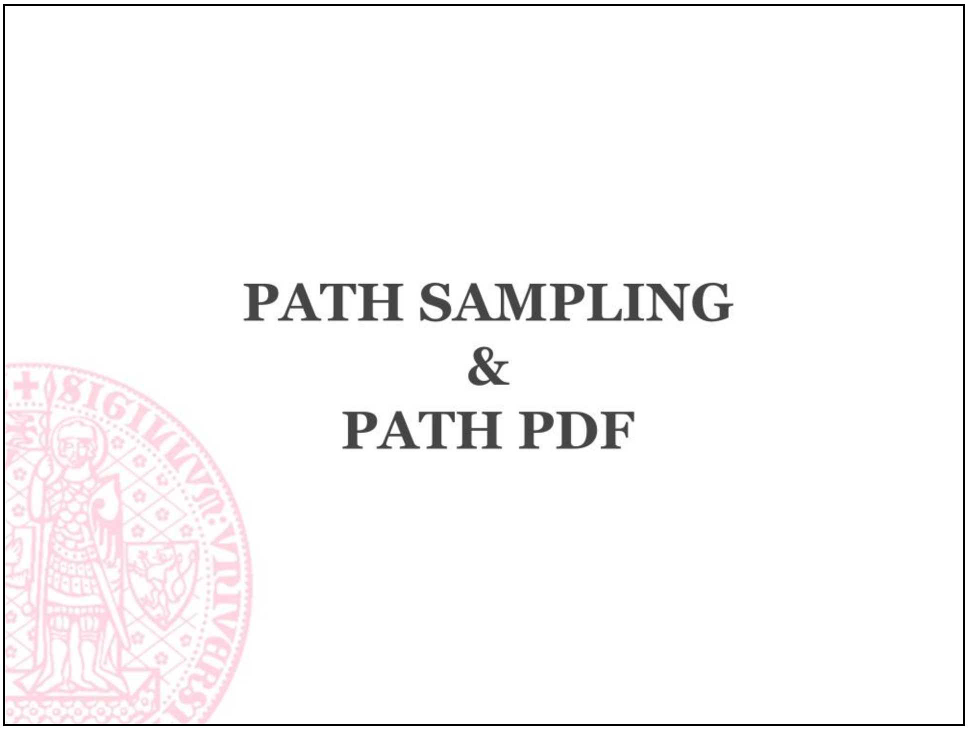


• Though path tracing and light tracing may seem like very different algorithms, from the path integral point of view they are essentially the same.

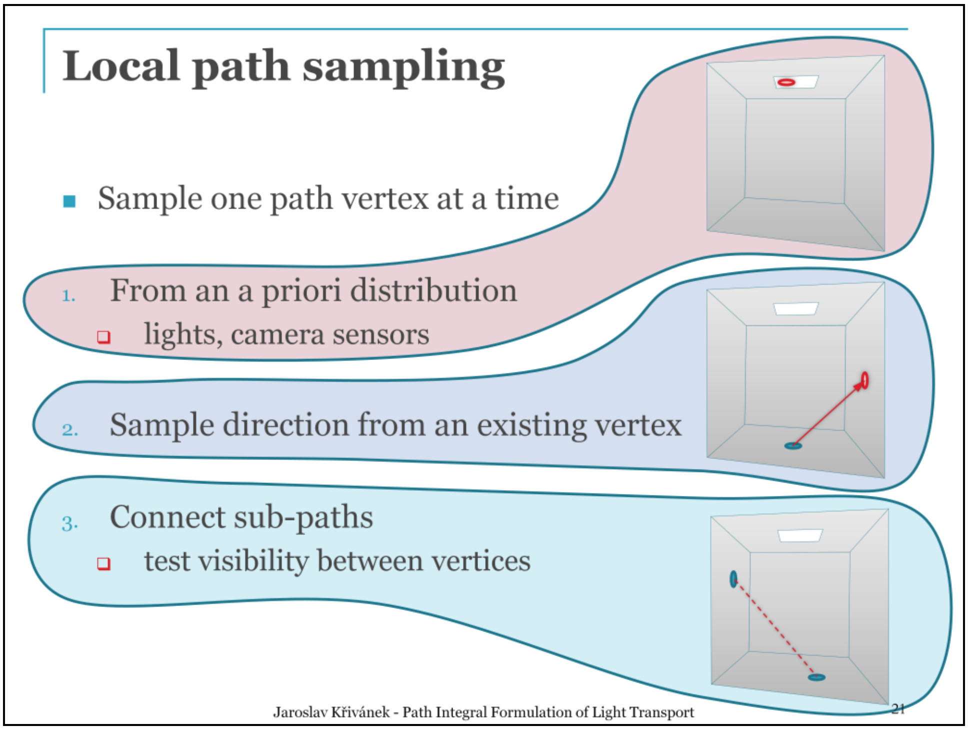
• The only difference is the path sampling procedure, and the associated path PDF.

• But the general 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?



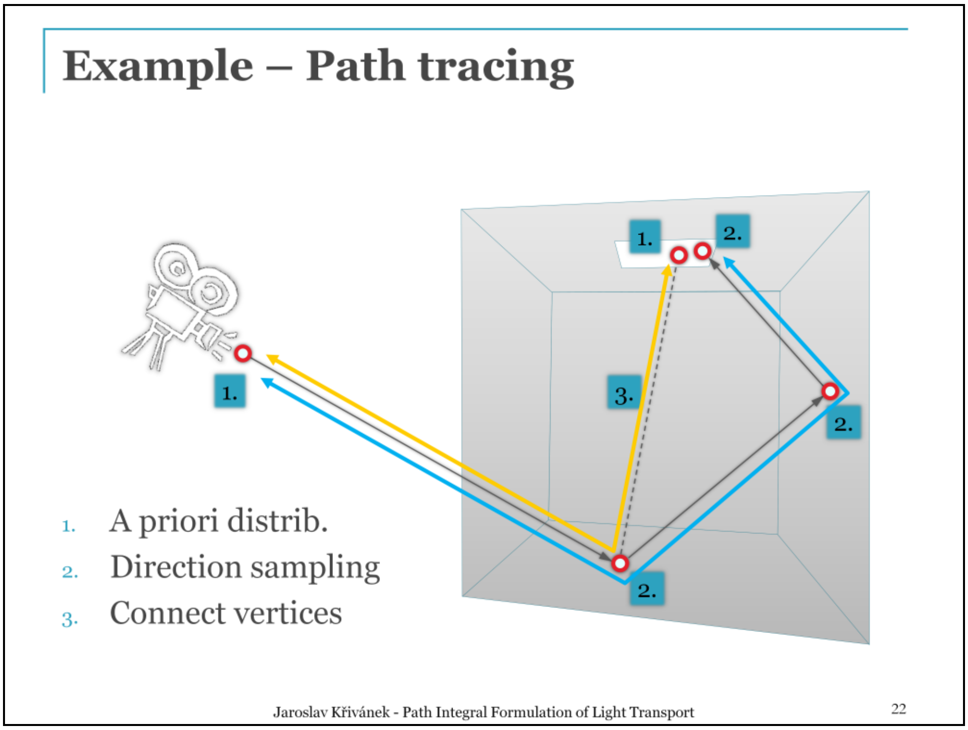
• Most of the practical algorithms rely on local path sampling, where paths are build by adding one vertex at a time until a complete path is built.

• 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.

  • 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.



• Let us see how these three basic operations are used in a simple path tracer.

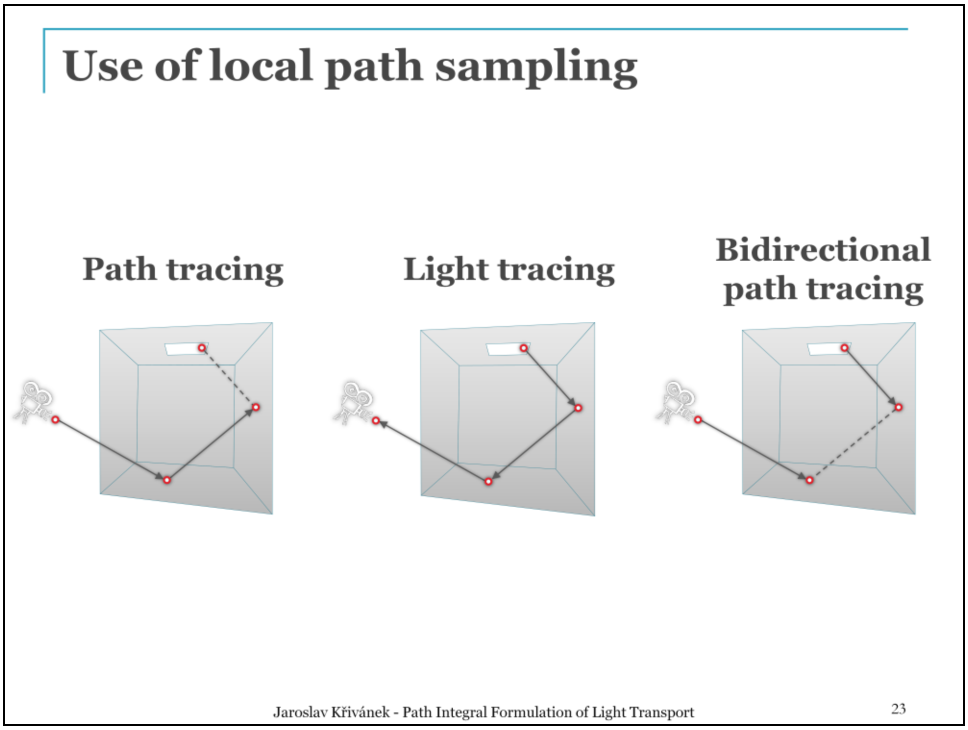
• First, we generate a vertex on the camera lens, usually from a uniform distribution over the lens surface – so this corresponds to operation 1.

• Second, we pick a random direction form this point such that it passes through the image plane, and shoot a ray to extend the path. This is operation 2.

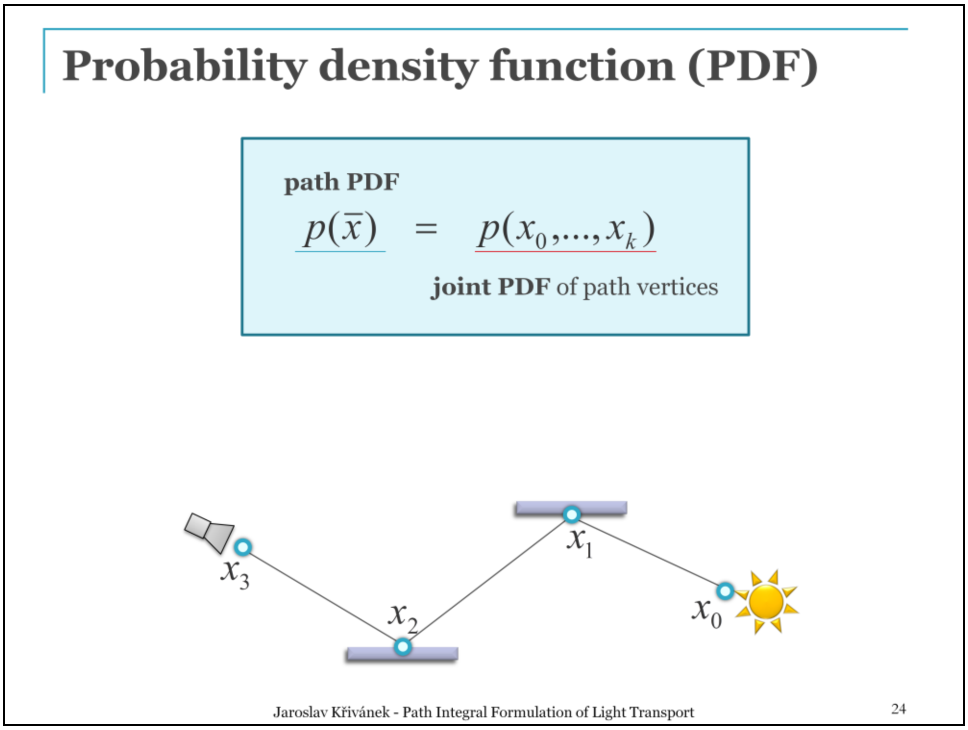
• Third, we may generate an independent point on the light source (operation 1) and test visibility (operation 3) to form a complete light transport path.

• We could also continue the path by sampling a random direction and shooting a ray (operation 2), and eventually hit the light source to complete the path.

• An important thing to notice is that one single primary ray from the camera actually creates a full family of light transport paths. These path are correlated, because they share some of the path vertices, but they are distinct entities in the path space.



• These same basic operations are used to construct paths in light tracing and bidirectional path tracing.



• Not 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.

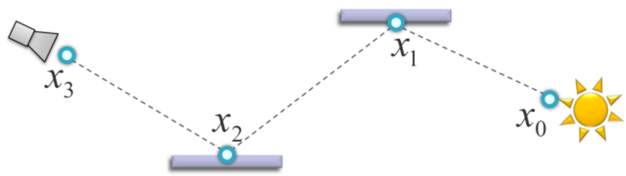
• That is to say, the PDF that the first vertex is where it is *and* the second vertex is where it is, etc.

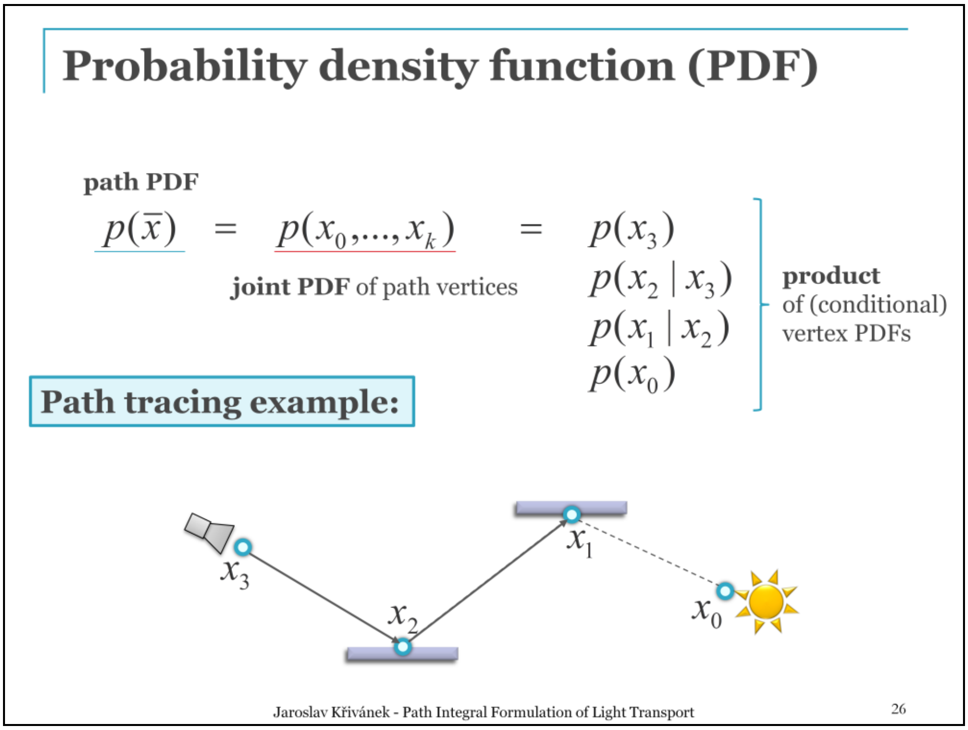
Probability density function (PDF)

path PDF

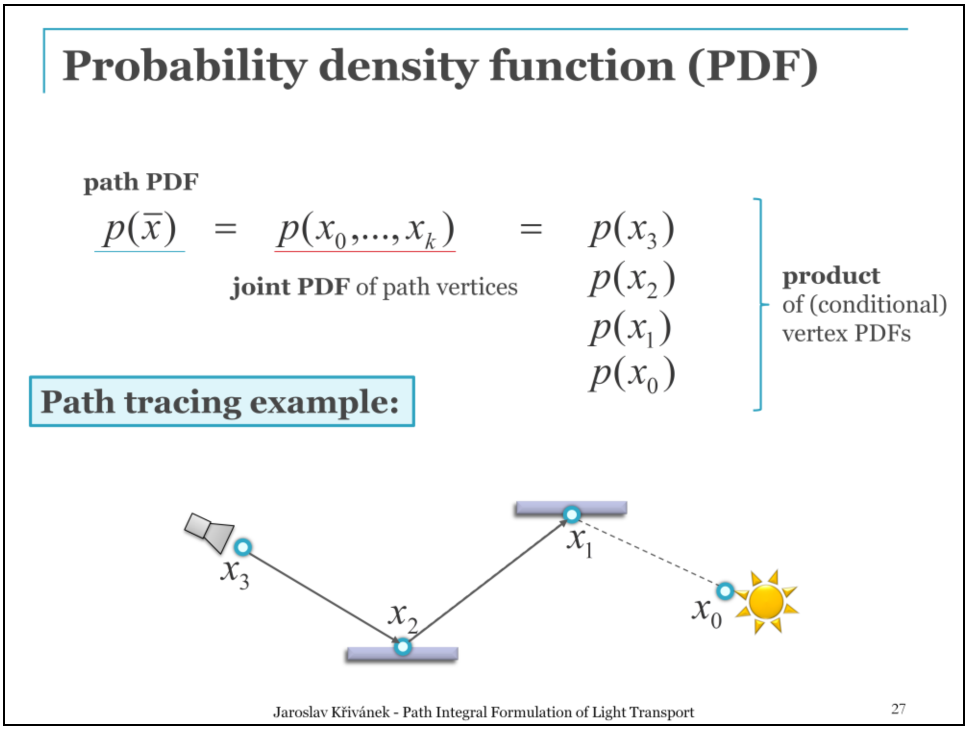
$$\underline{p(\bar{x})} = \underline{p(x_0, \ldots, x_k)}$$

joint PDF of path vertices

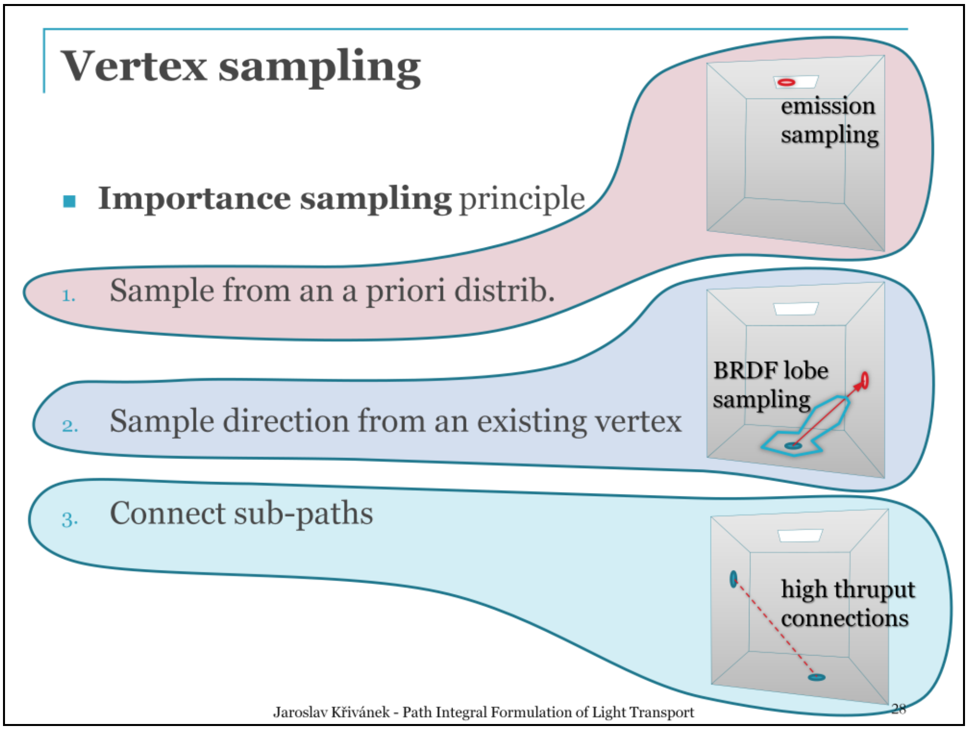




- The joint path PDF is given by the product of the conditional vertex PDF.
- To see what this means, let us again take the example of path tracing, where we build a path starting from the camera.
- Vertex x_3 comes from an a priori distribution $p(x_3)$ over the camera lens (usually uniform; or the 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.



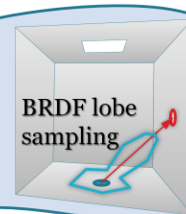
- It is customary to simplify this somewhat pedantic notation and leave out the conditional signs. Nonetheless, it is important to keep in mind that the path vertex PDFs for vertices that are not sampled independently are indeed conditional PDFs.

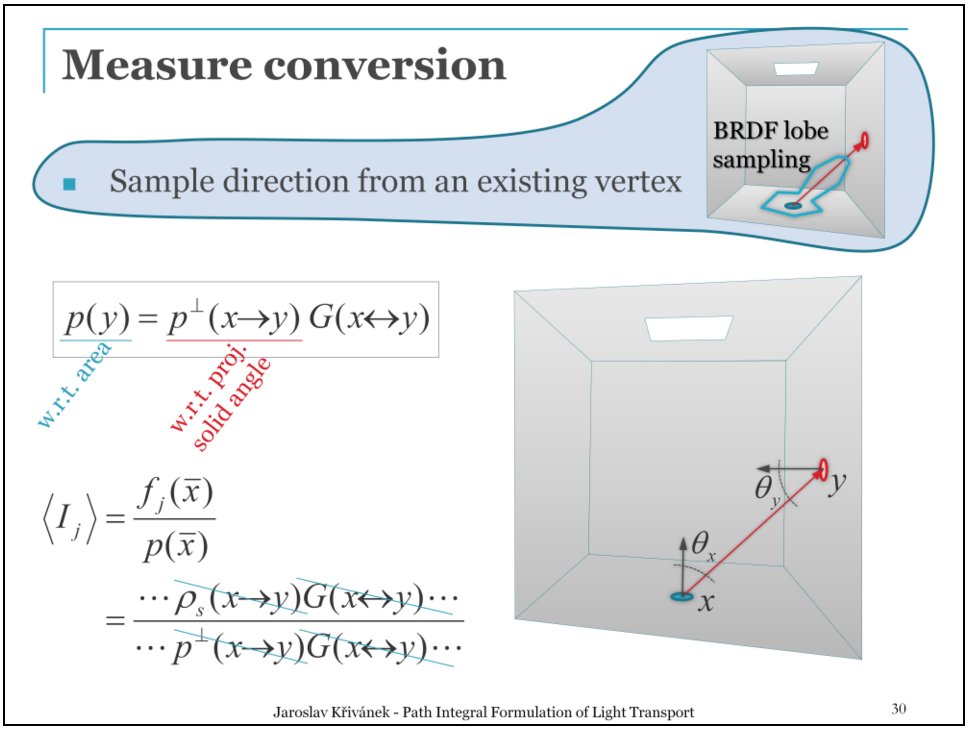


- In accordance with the principle of importance sampling, we want to generate full paths paths from a distribution with probability density proportional to the measurement contribution function. That is, high-contribution paths should have proportionally high probability of being sampled.
- Local path sampling takes an approximate approach, where each local sampling operation tries to importance sample the terms of the contribution function associated with the vertex being sampled.
- For example, when starting a path on the light source, we usually sample the initial vertex from a distribution proportional to the emitted power.
- When extending the path from an existing vertex, we usually sample the random direction proportionally to the BRDF at the vertex.
- Similarly, when connecting two sub-paths with an edge, we may want to prefer connections with high throughput (though this is rarely done in practice).

Vertex sampling

- Sample direction from an existing vertex



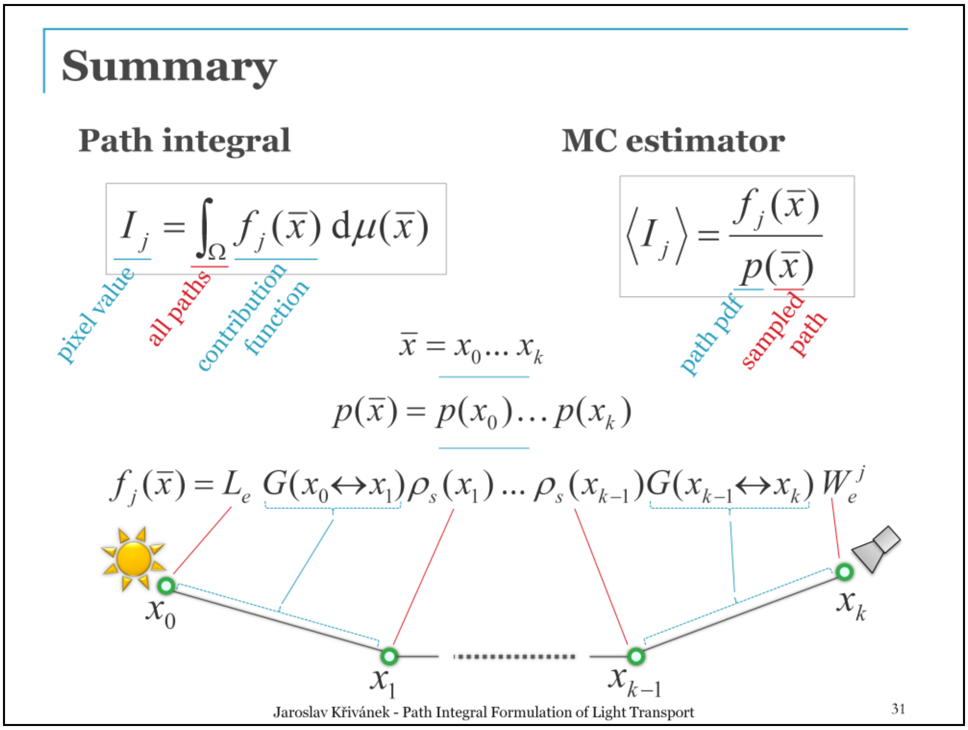


• There is one important technical detail associated with computing the path PDF for vertices created by direction sampling.

• The path integral is expressed with respect to the surface area measure – we are integrating over the surface of the scene – but the direction sampling usually gives the PDF with respect to the (projected) solid angle.

• The conversion factor from the projected solid angle measure to the area measure is the geometry factor.

• This means that any vertex generated by first picking a direction and then shooting a ray has the geometry factor of the generated edge importance sampled – the only geometry factor that are not importance sampled actually correspond to the connecting edges (operation 3 in local path sampling).



• At this point, we should summarize the mathematical development so far.

• The path integral formulation gives the pixel value as an integral over all light transport paths of all lengths.

• A light transport path is the trajectory of a light carrying particle. It is usually a polyline specified by its vertices.

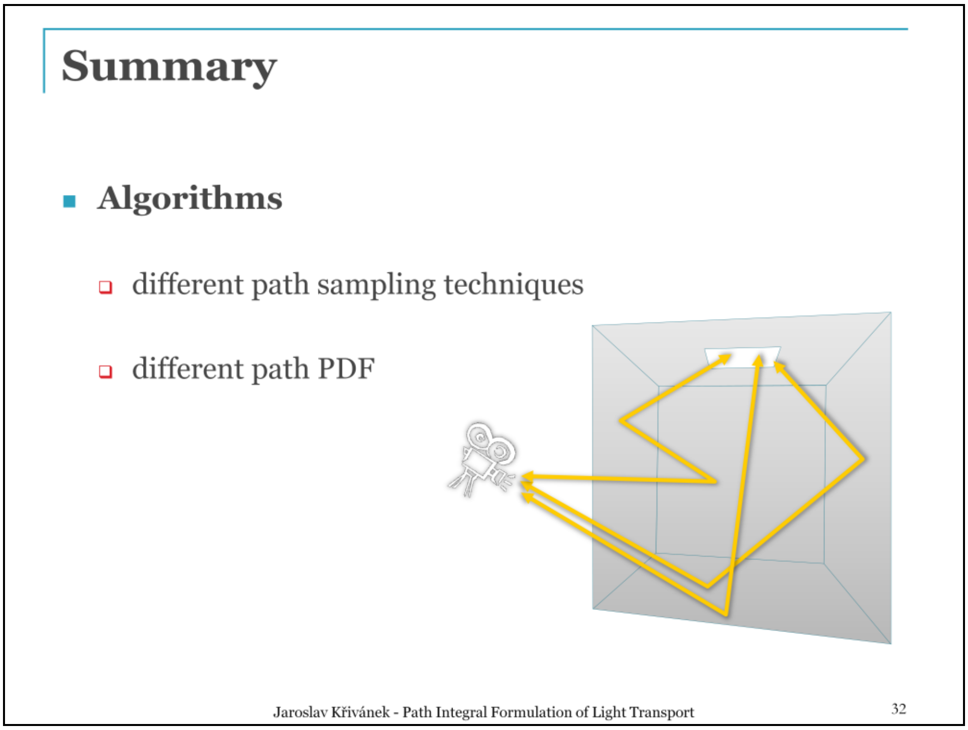
• The integrand is the measurement contribution function, which is the product of emitted radiance, sensor sensitivity, and path throughput.

• Path throughput is the product of BRDFs at the inner path vertices and geometry factors at the path edges.

• To evaluate the integral, we use Monte Carlo estimator in the form shown at the top right of the slide.

• To evaluate the estimator, we need to sample a random path from a suitable distribution with PDF $p(x)$.

• We usually use local path sampling to construct the paths, where we add one vertex at a time. In this case, the full path PDF is given by the product of the (conditional) PDFs for sampling the individual path vertices.



- 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. This will be detailed in the next part of the course.

Time for questions…

Tutorial: Path Integral Methods for Light Transport Simulation

Jaroslav Křivánek – Path Integral Formulation of Light Transport