Path Integral Methods for Light Transport Simulation: Theory & Practice

Comparison of Advanced Light Transport Methods

Equal Time Comparison

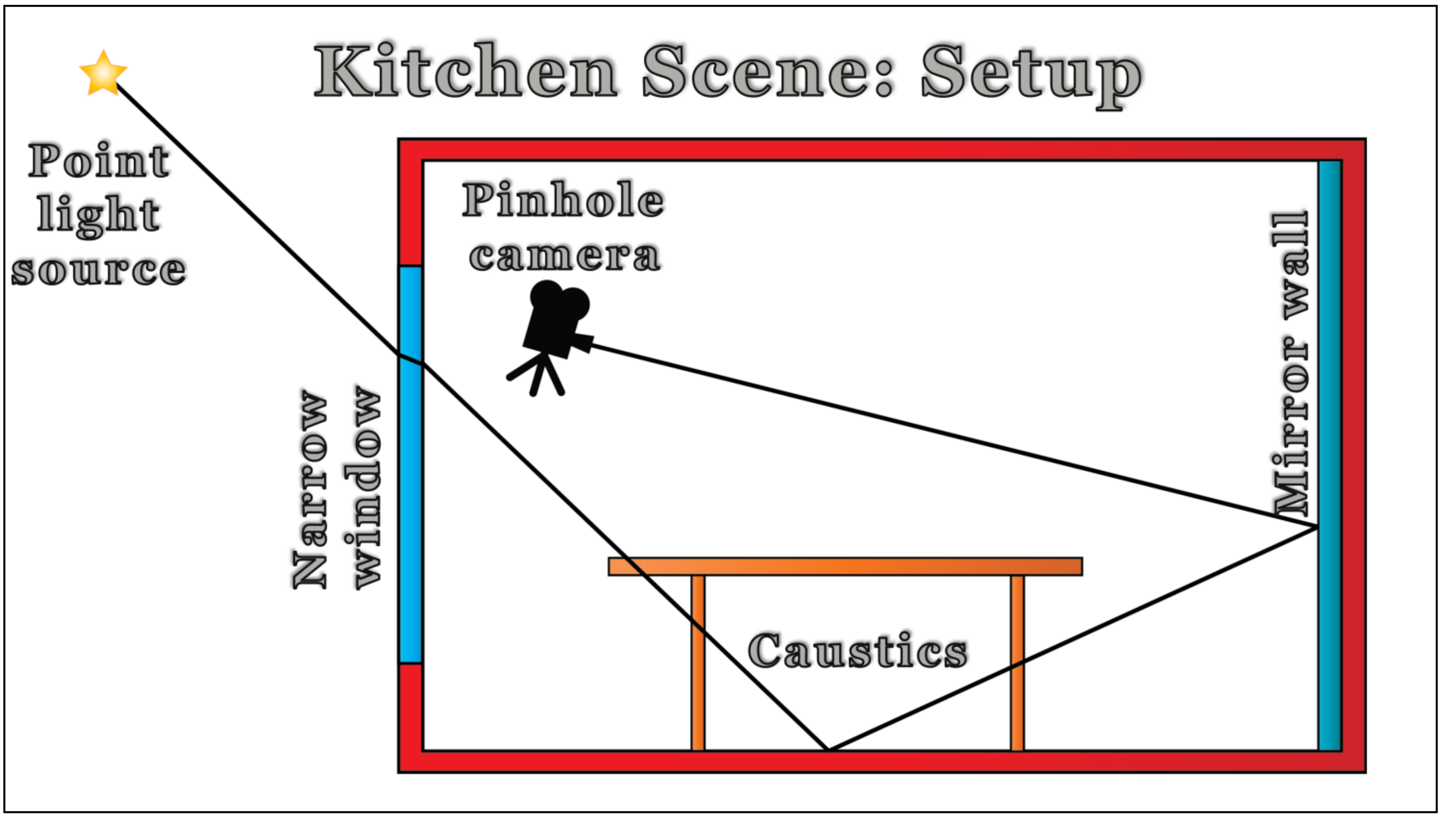
- Rendering time: 1 hour on GeForce 580
  - Some are rendered with Mitsuba [Jakob10]
- *Kitchen* – a hard-to-render scene
  - Light occluded by glass, caustics
  - Many mirrors, reflected caustics
  - Glossy materials
  - Curved glass with many specular interactions

In this part of the course I would like to provide a qualitative comparison of different existing methods.

All renderings are done in equal time (1 hour) on the Nvidia GeForce 580 GTX GPU.

Some images are rendered on CPU with Mitsuba renderer with a performance equivalent of 1 hour on GPU.

I have set up a scene with multiple difficult lighting features in order to evaluate state-of-the-art light transport methods.



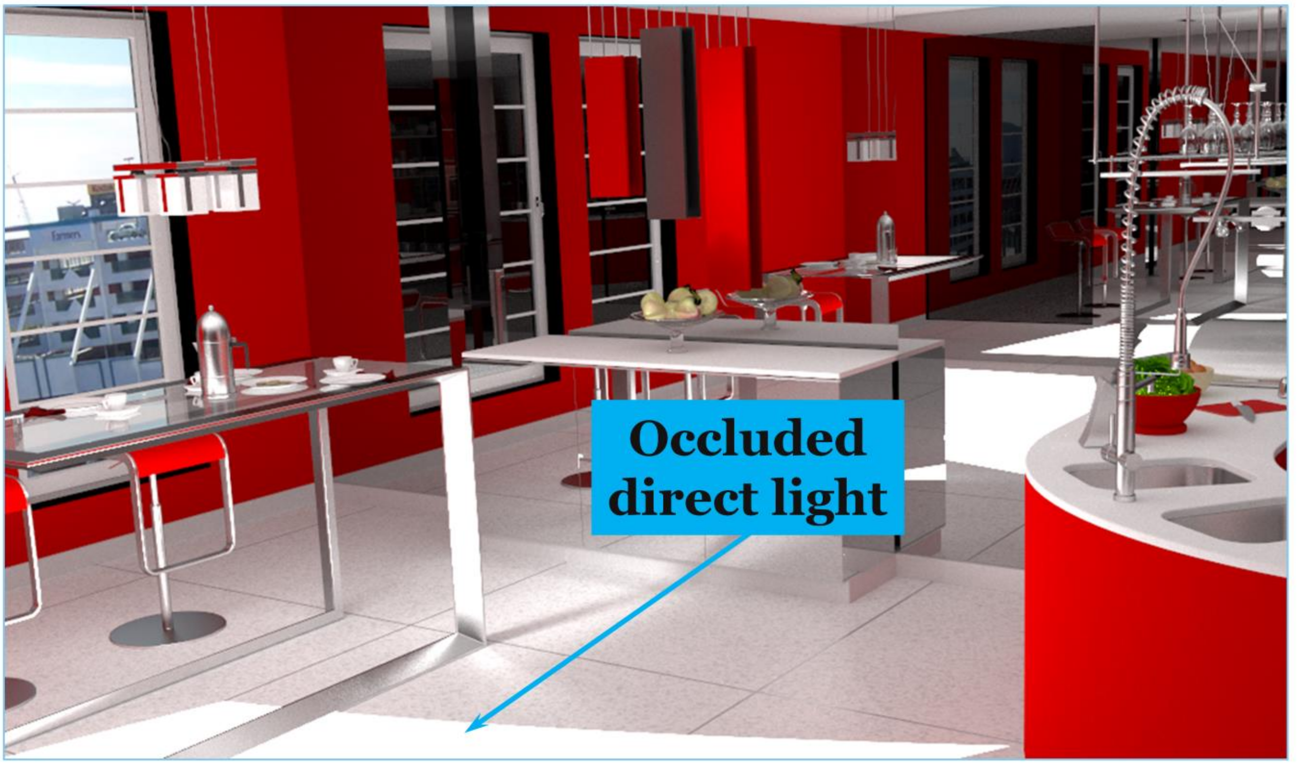
Here is the scene configuration.

Point light source is placed outside, illuminating the room through a glass window, causing a caustic on the floor.

The scene has also a mirror wall causing a lot of reflections.

This is the “Kitchen” scene rendered with diffuse materials to show the geometric topology.

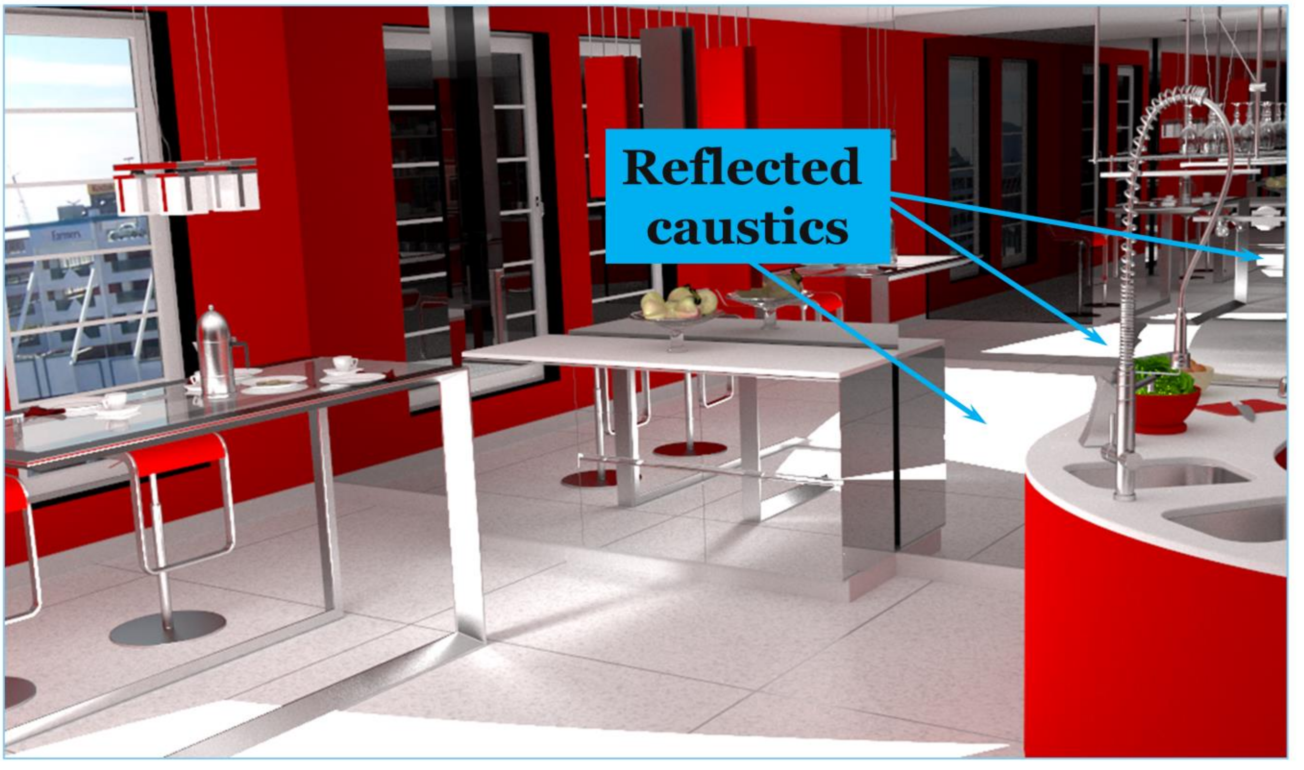
This is the reference image of the “Kitchen” scene with final materials. Rendered with Vertex Connection and Merging technique for 24 hours.



Here are some difficult features.

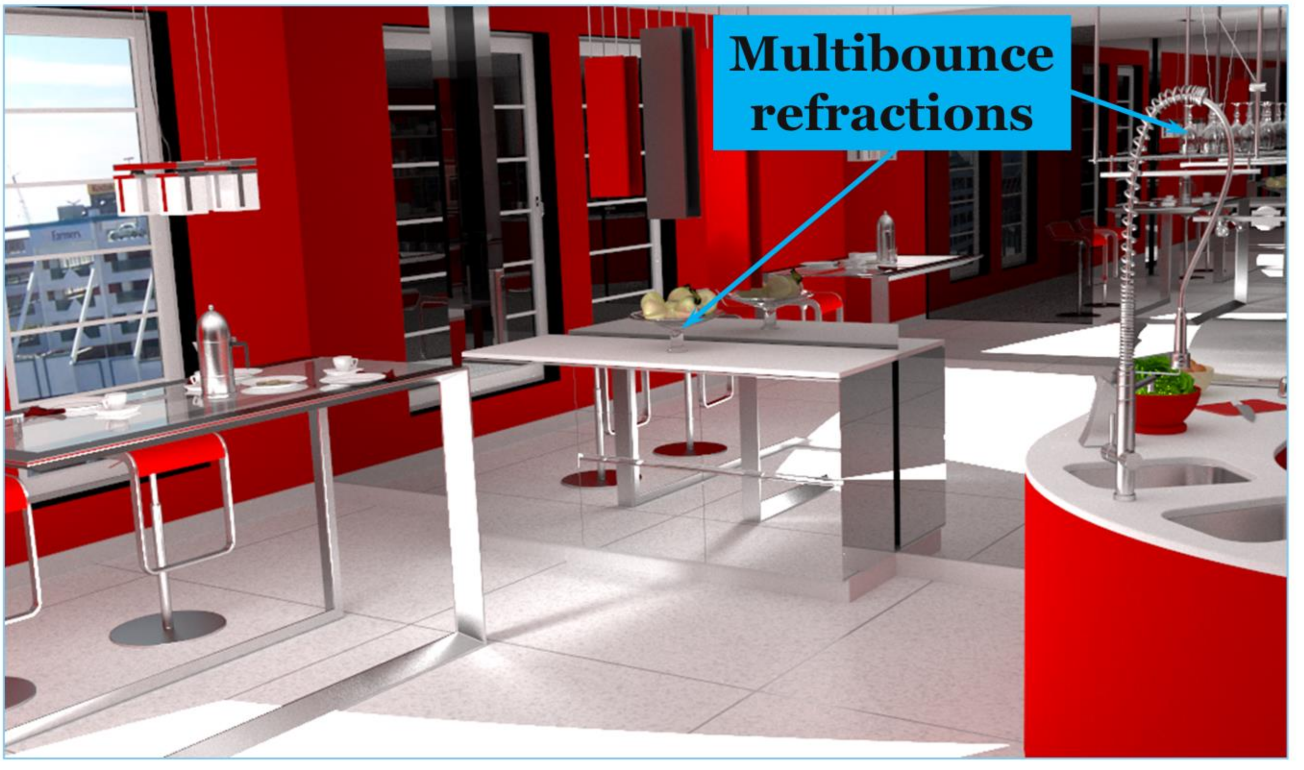
First of all, the direct lighting is highly occluded, since the window visible from the light source takes a small part of the emissive sphere of directions.

This situation is similar to the classical complex light transport scenarios, like an ajar door [Veach97].

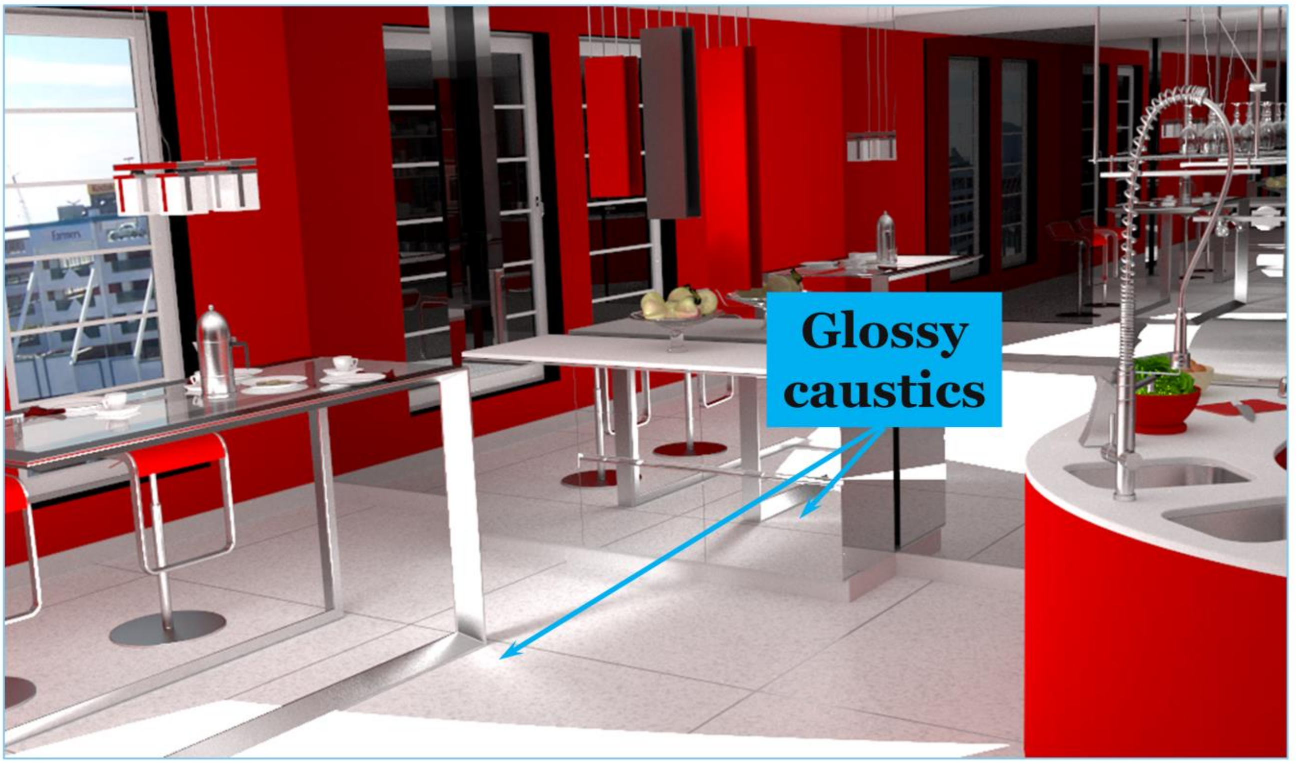


Then, the reflected caustics caused by the point light source are impossible to find with all unbiased methods.

Also caustics, experiencing multiple bounces (window->table->floor) pose a difficult sampling problem for some Monte Carlo methods.



Refractions with multiple bounces (specular chains) has been also known to be a hard sampling problem [Jakob12].



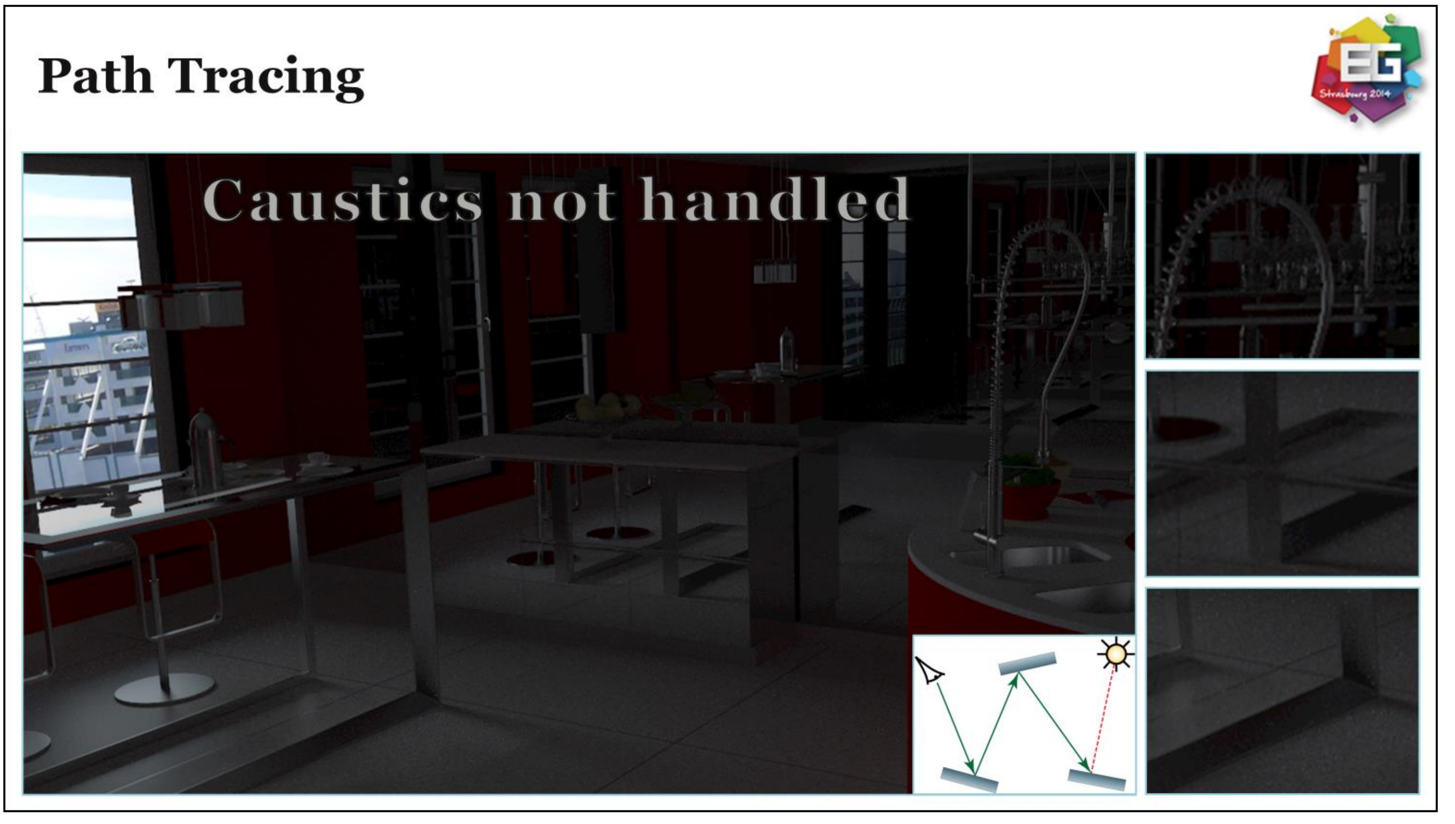
Finally, caustics from highly glossy objects are hard to find in path space (they have a small "mass"), thus might be difficult to discover even by advanced MCMC methods.

Ordinary
Monte Carlo Methods

Ordinary Monte Carlo Methods

- Path tracing [Kajiya86]
- Light tracing [Arvo86, Dutre93]
- Bidirectional path tracing [LaFortune93, Veach94]
- Vertex connection and merging [Hachisuka12, Georgiev12]

We start with ordinary Monte Carlo methods. They can usually handle majority of scenes.
We will show the following Monte Carlo methods.



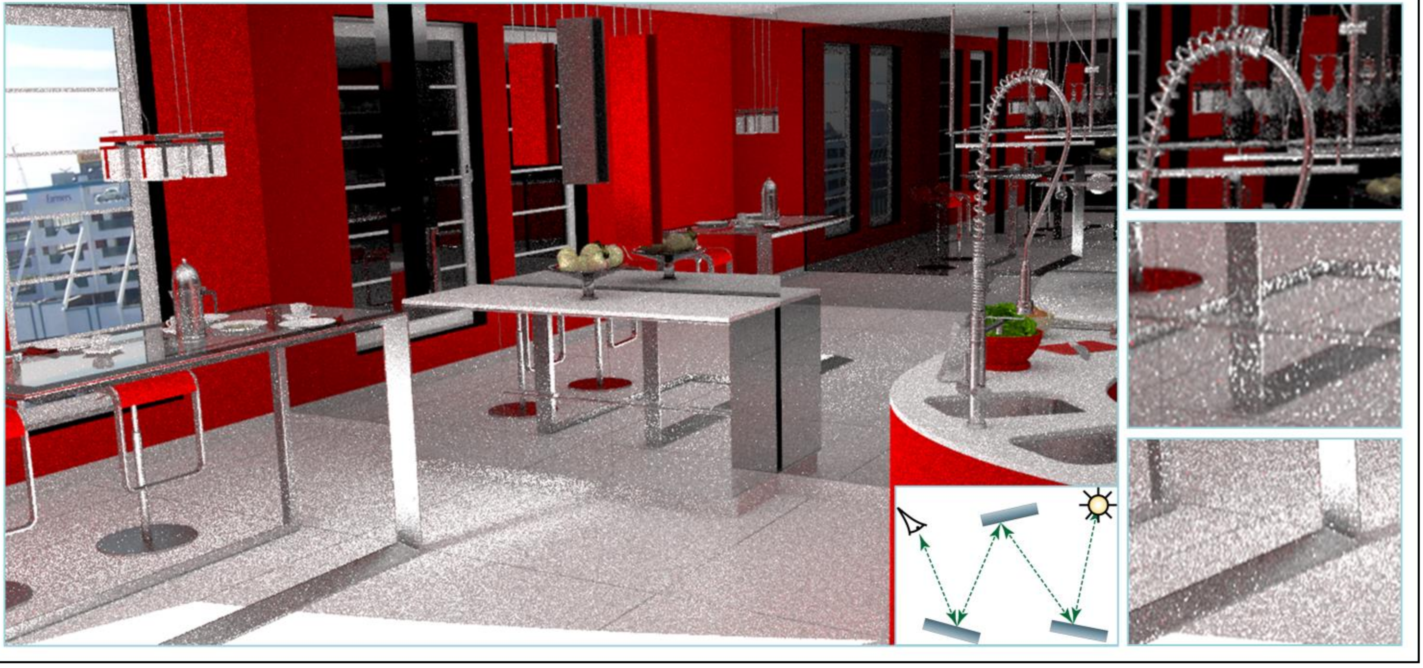
The first method: path tracing. It can handle only the small fraction of light coming from the environment.

This is due to the fact that al illumination coming from the point light source first comes through the glass. This makes it impossible to directly connect to the light source.

Light Tracing

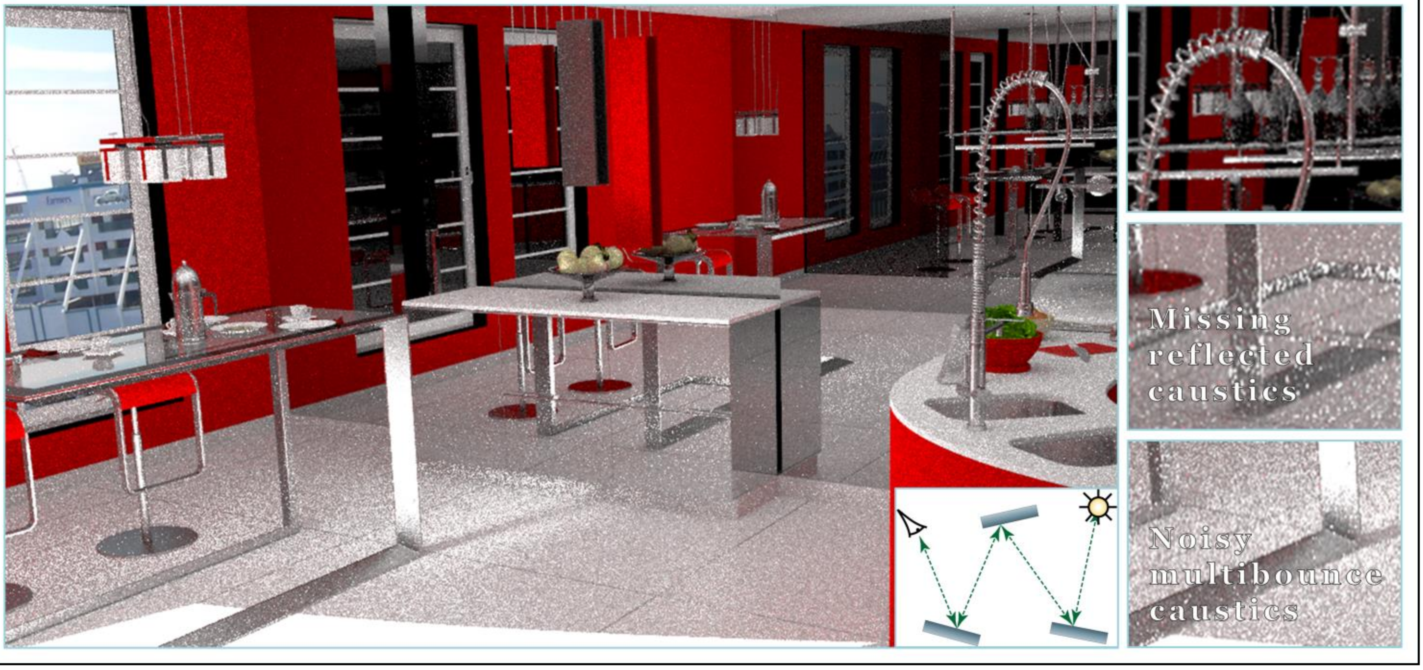
In opposite, the light tracing technique handles caustics coming through the window quite well, yet failing all specular reflections and refractions, because they cannot be constructed using connection to camera through specular surfaces.

Bidirectional Path Tracing



Bidirectional path tracing handles almost all possible paths, however some difficult paths are noisy or missing.

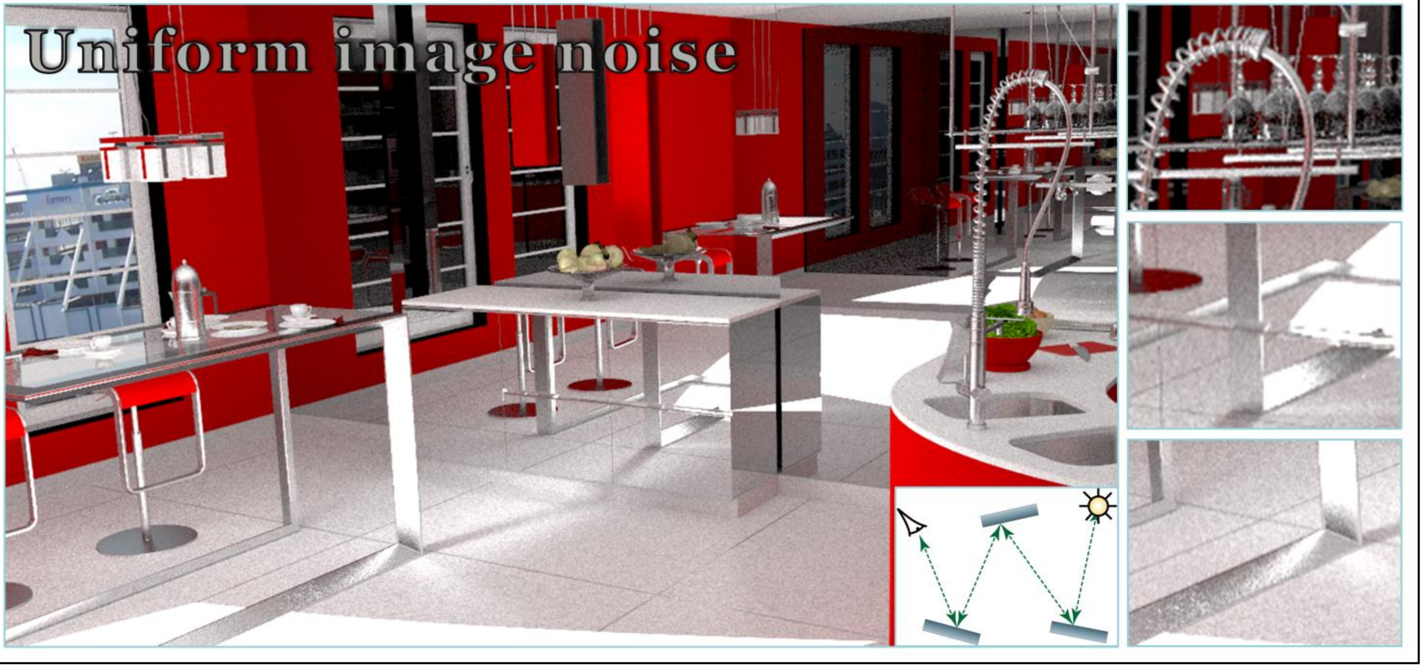
Bidirectional Path Tracing



BDPT has difficulties sampling caustics that go through multiple bounces. The reason is that the mass of such caustics in the path space is small, making it difficult to find them stochastically.

And the reflected caustics from point light cannot be sampled due to missing connection opportunity (that is, a path edge without singular materials).

Vertex Connection and Merging



Recent vertex connection and merging technique can handle caustics much better.

Moreover it can even sample reflected caustics. However, the method has uniform image noise due to the highly occluded light source (which is being sampled stochastically).

Why Monte Carlo methods work well in most of the cases?

You have probably seen the recent femtosecond photography of real world light propagation.

(0) I decided to do the same, but with the light transport simulation.

(1) This video demonstrates that the light transport turns into a diffusion process very quickly.

(2) The same happens when I propagate the virtual wavefront of “importance” from the camera. Note that it is not a physically valid value, yet it shows how the adjoint quantity is propagated, e.g., in bidirectional path tracing.

Both quantities spread around the scene very quickly, making it easy to establish the connection for unidirectional and bidirectional methods.

(3) However let’s take a look at the glossy example: the diffusion is not that prominent anymore. This makes such configuration difficult to sample by ordinary Monte Carlo methods, because occasional sampling and stochastic connections of two subpaths lead mostly to low or zero contribution of the path on the image plane.

Markov Chain Monte Carlo Methods

However as we will see, some difficult illumination features and visibility situations can be handled more efficiently with Markov chain Monte Carlo methods.

Markov Chain Monte Carlo Methods

- Metropolis light transport [Veach97]
- Different mutations
  - Primary sample space mutation [Kelemen02]
  - Path space mutations [Veach97]
  - + Manifold exploration mutation [Jakob12]
- Energy redistribution path tracing [Cline05]

I'll present the following techniques, mostly various techniques in MLT framework.

Markov Chain Monte Carlo Methods, part II

- PPM with MCMC photon tracing [Hachisuka11]
- Population Monte Carlo energy redistribution [Lai07]

I will also show these recent and advanced MCMC methods.

Metropolis Light Transport

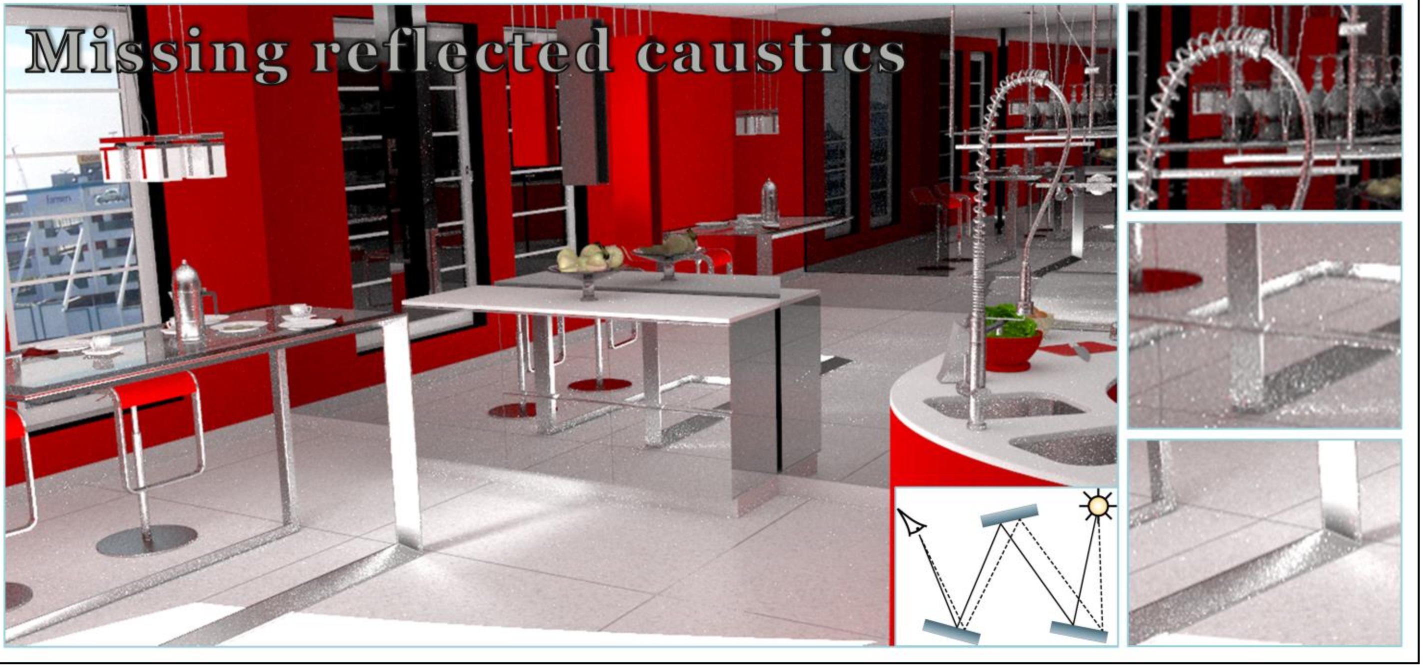
The first MCMC method (also chronologically) is the original Metropolis light transport algorithm [Veach97] with the original set of mutation strategies.

The image is handled well, without noticeable noise. Yet reflected caustics are missing, as expected from an unbiased method.

Note some under-sampling at geometric boundaries, especially around glossy surfaces (for example, the farther horizontal table leg on the floor has some bright splotches).

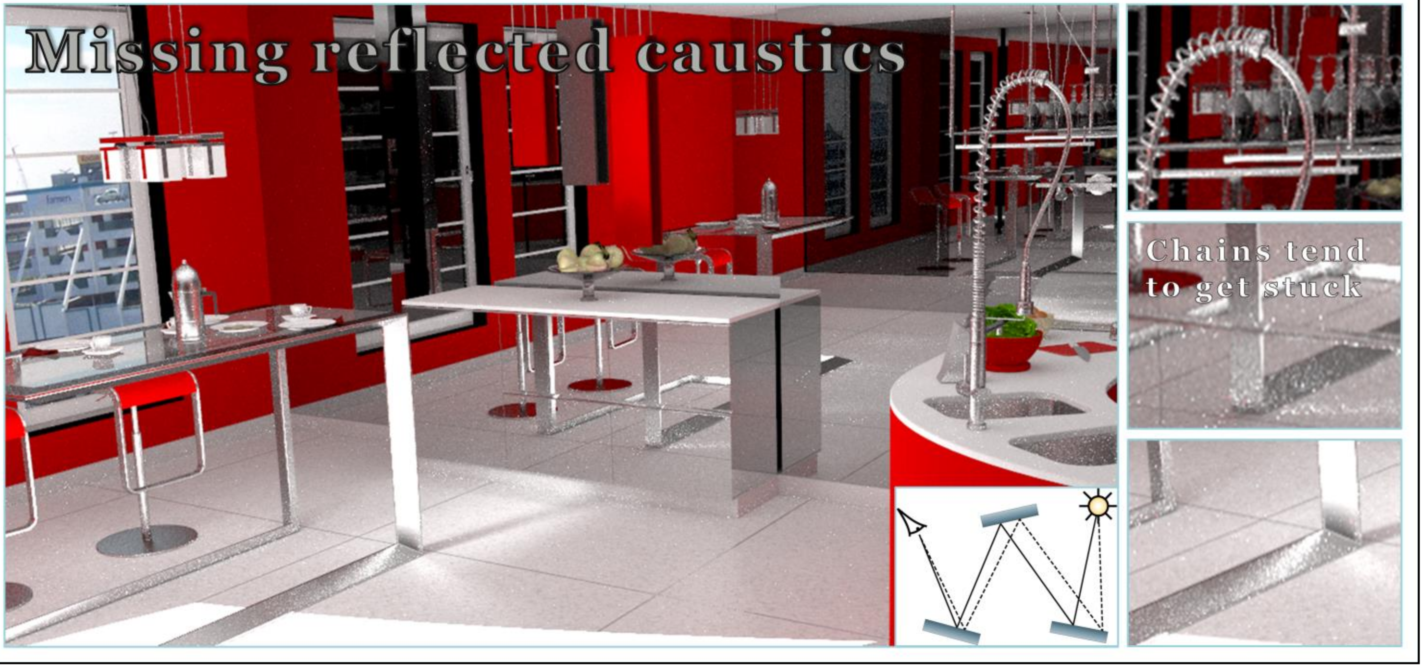
This is due to the sudden change in path measurement contribution function leading to high rejection rate of perturbations closer to the corners of the surfaces.

MLT in Primary Sample Space



Here is Metropolis light transport with mutations in primary sample space [Kelemen02].

MLT in Primary Sample Space



Note that some chains get stuck in complex caustics paths.

This happens because of many rejections since the whole path is usually perturbed, thus constantly jumping from its valid form.

Also the small step size is specified globally for random numbers and is not adapted for long or highly-occluded paths.

MLT with Manifold Exploration

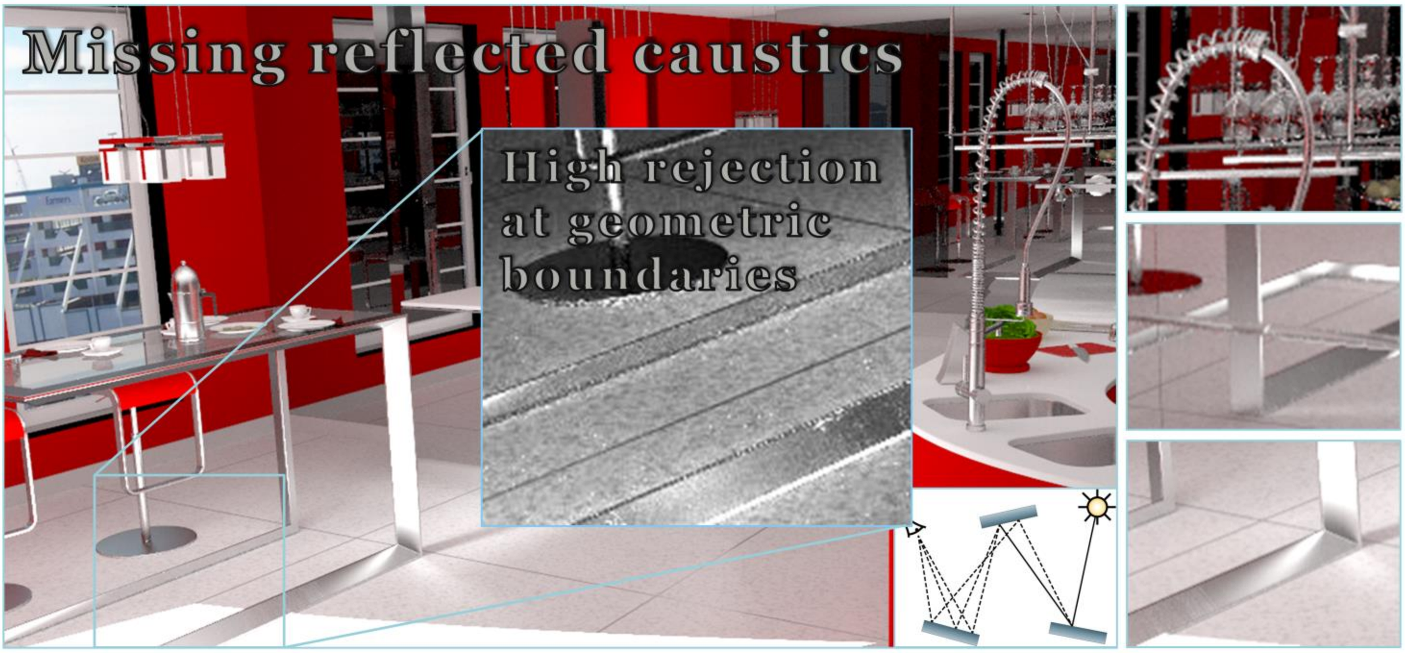
Here is the original MLT algorithm with the recently introduced manifold exploration mutation [Jakob12].

Note how much better the multibounce refractions are handled.

Also note that the reflection of the glossy caustics from the table leg are handled much better than with the original MLT.

This is due to the fact that this mutation can skip through specular interactions, such as mirror reflections and refractions.

MLT with Manifold Exploration



Also note that some subtle under-sampling is present along geometric edges, because of high rejection rate due to the invalidation of local differential geometry and sudden change of path throughput when jumping from one surface to another.

This issue can be addressed with the adaptive sampling proposed in the gradient-domain Metropolis light transport method [Lehtinen13] by Jaakko Lehtinen.

Markov Chain PPM

Here is MCPPM, another recent method by Toshiya Hachisuka.

It is an extension to stochastic progressive photon mapping [Hachisuka08] with Markov chain photon shooting [Hachisuka11].

The original Kelemen mutation strategy with recommended parameters was used to mutate photon paths.

In order to keep the target distribution constant, the camera subpaths should be updated rarely.

This leads to under-sampling at glossy reflective and refractive surfaces.

Energy Redistribution Path Tracing (ERPT)

Here is the original energy redistribution path tracing method [Cline05] with the original set of mutations [Veach97], seeded with BDPT.

We used one chain per pixel (on average) and 100 mutations per chain (being the original values recommended by the author).

As we can see, ERPT has difficulties efficiently redistributing energy from glossy paths, especially if they also experience one or more perfect reflections/refractions.

Moreover, some difficult parts of the image are under-sampled in the same rendering time comparing to MLT. This happens because Markov chains are re-seeded with BDPT much more often, yet BDPT doesn't provide a good quality of seeding, as we have seen from the BDPT rendering (due to the highly occluded light source and complex light transport).

ERPT + Manifold Exploration

Another combination is ERPT with manifold exploration.

As we can see, manifold exploration allows to redistribute the energy of complex specular and glossy paths much better, leaving just a few splotches on the image.

ERPT, Manifold Exploration Only

Another interesting experiment is to provide only manifold exploration as an available redistribution mutation, while relying on BDPT for the stratification and search of new features.

As we can see it also works relatively well, leading to the conclusion that this single mutation strategy is enough for achieving good results with ERPT.

However the equilibrium is not achieved with any of ERPT variations in a given time budget of 1 hour on this scene.

Population Monte Carlo ER + ME

In this rendering we use the Population Monte Carlo framework applied on top of ERPT [Lai et al. 2007] with multiple manifold exploration mutations.

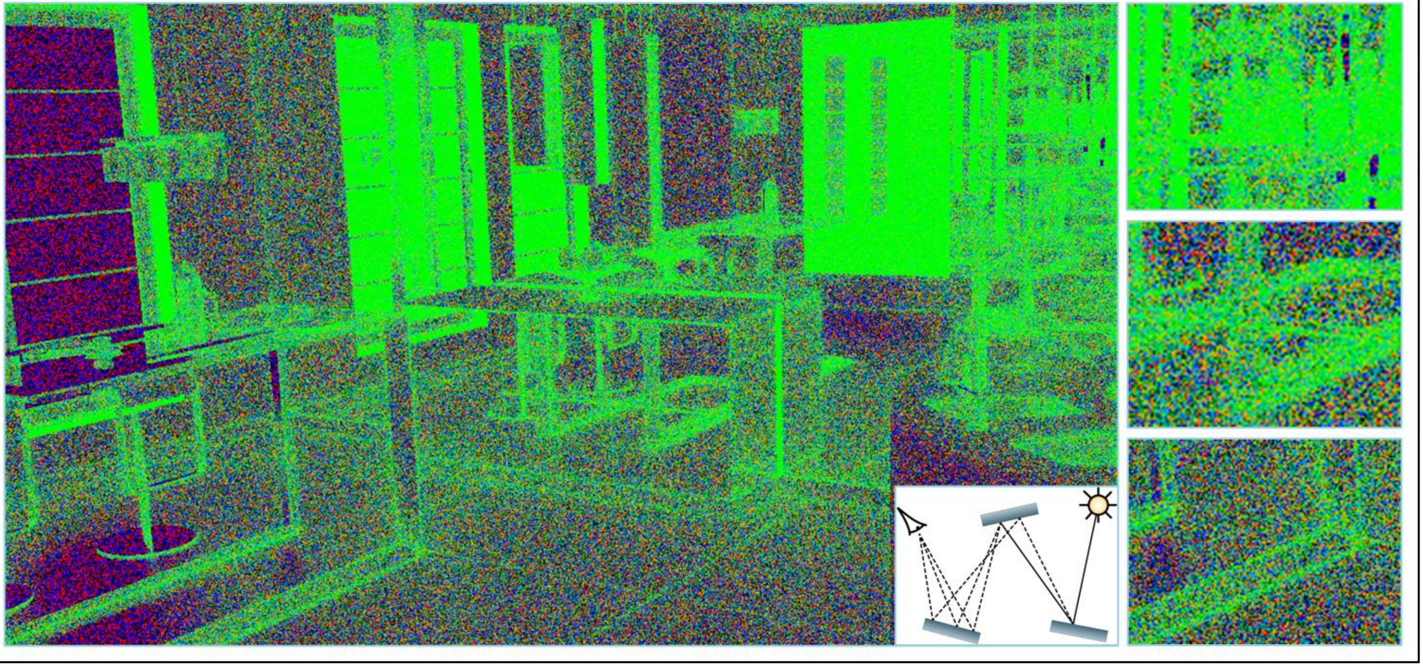
We use a population of 1024 chains and various values of step parameter *lambda* ranging from 20 to 200 for manifold exploration mutation.

Note that the geometric edges are sampled much better due to adaptive selection of the optimal mutation step size.

For example, edging of the glossy table is sampled much better, which is even more noticeable in the reflection of the table legs.

This is due to the fact that the population adapts and selects the mutation strategy with smaller perturbation step.

Population Monte Carlo ER + ME



This is a false-colored overlay of weights for different selected strategies.

Greener color corresponds to higher weight of a mutation with smaller step size.

These weights are reset every time the chain is re-seeded from BDPT, thus the weights are that noisy.

Note how the method prefers mutations with smaller step along geometric discontinuities.

Conclusion

- MCMC is more robust to complex lighting
  - \+ Better survives the *curse of dimensionality*
  - \+ Rule of thumb: for ≥~7-10 bounces → MCMC
  - \+ Helps with highly occluded and glossy scenes
  - – Non-uniform convergence: bad for animation and previews
- PMC adapts mutation parameters

I'd like to mention that modern MCMC method survive the course of dimensionality better than ordinary MC methods (like PT and BDPT).

That means that whenever the majority of light transport in the image is high-dimensional (that is, roughly more than 7-10 bounces), then MCMC behaves much better in such cases.

It also helps exploring narrow islands and peaks in path space caused by highly occluded lighting configuration and/or highly glossy materials.

However, MCMC is famous for its non-uniform convergence, that is, it might take a while before Markov chain finds some important feature on the image, like some distant reflected caustic.

In such case, this feature just suddenly appears on the image rendered with progressive rendering.

This is unwanted in rendering of animated sequences and in quick preview scenarios, when artists want to briefly assess the lighting.

Additionally, population Monte Carlo framework provides an easy and unbiased adoption of parameters for mutation strategies and also provides slightly better redistribution.

Mutation parameters' adaption can play a significant role in scenarios with high rejection rate, for example, caused by high-frequency geometry of tree leaves when rendering forest-like scene.

To conclude, so far there is no silver bullet method: MCMC methods are better standing the curse of dimensionality, but they are still not completely usable for production – Juan will talk about that in his part.

Thank You for Your attention.

Part two questions?