GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

OPEN PROBLEMS AND RESEARCH DIRECTIONS

Jaroslav Křivánek
Charles University | Render Legion | ChaosGroup

ACADEMIA – INDUSTRY

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

Computer Graphics Charles University

corona

Jaroslav Křivánek – Rendering in architecture and product viz (1)

- In this last part of the course, I would like to take advantage of my academic and industry backgrounds to discuss the relation – and sometimes a disconnect – between the two, and to give some directions for future research.

- I will be specifically focusing on the underlying light transport algorithms, such as path tracing, Bidir, Metropolis light transport, VCM, etc…

- So what remains to be done in physically-based light transport? Isn't it a solved problem?
- I believe it is not. To motivate the need for further research in this area, let me start by proposing the ultimate light transport algorithm challenge.
- The challenge goes: develop a light transport simulation algorithm that renders a Cornell Box at least as fast as an unidirectional path tracer (if not much faster).
- But at the same time, the same algorithm should be able to render complex scenes like the ones above in a reasonable amount of time, with no objectionable artifacts, without setting a single user parameter, interactively, and progressively.
- The algorithm should be able to handle complex indirect lighting, caustics, multiple specular reflections/refractions, complex volumetric transport, rendering of hair, human skin etc. It should handle geometric complexity well and deal with a huge number of light sources.
- We are definitely not there yet as of today.

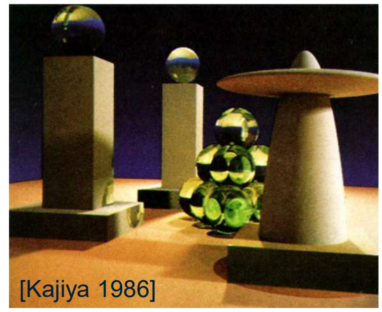
- In the rest of this talk, I will give my view of why it is that we do not yet have the desired `ultimate' solution, and I will propose some research directions that could help us approach this goal.

TODAYS' RENDERING IS OLD NEWS

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- From **Matt Pharr**'s editorial to ACM TOG special issue on production rendering [Pharr 2018]:

> *"Today … renderers are … based on … path tracing. Introduced … by Jim Kajiya (**1986**)."*
>
> *"Many advancements were made … including*
> *- more effective light sampling algorithms (Shirley et al. **1996**),*
> *- high-quality sampling patterns (Kollig and Keller **2002**), and*
> *- multiple importance sampling (Veach and Guibas **1995**),"*
>
> *"… the core ray tracing [got] more efficient (Wald et al. **2001**)."*



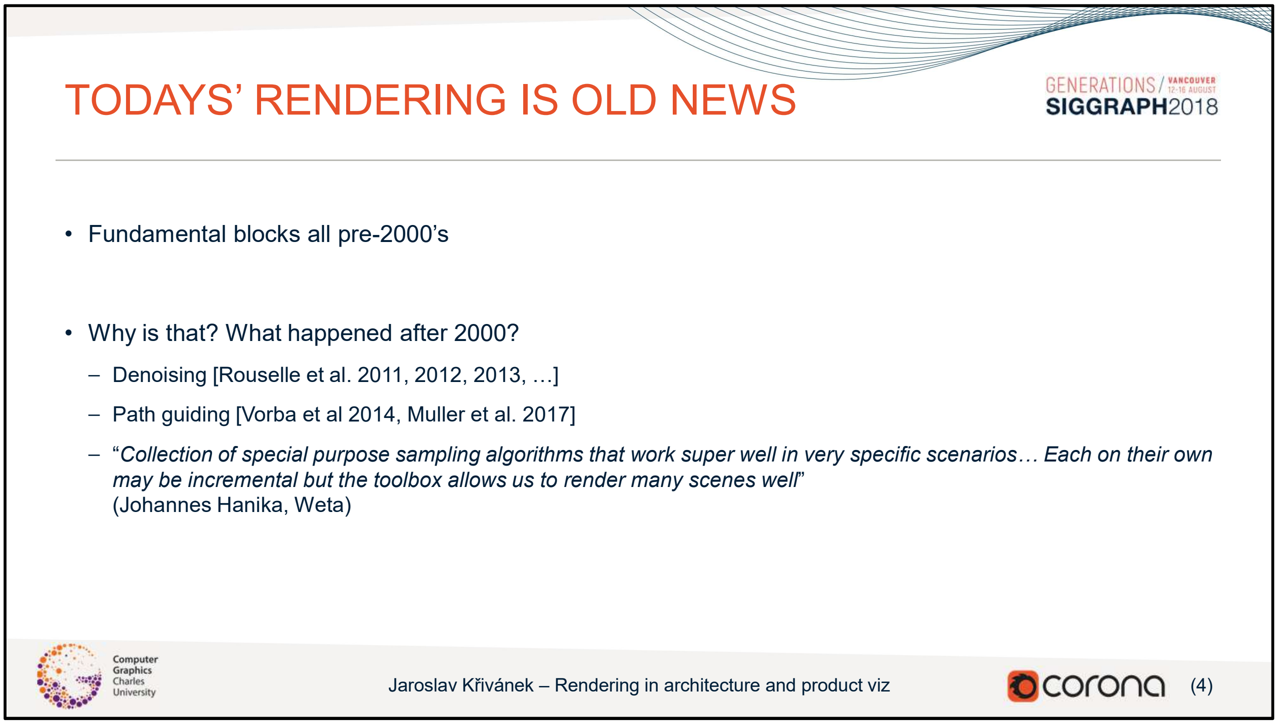
[Kajiya 1986]

Computer Graphics Charles University

Jaroslav Křivánek – Rendering in architecture and product viz

corona (3)

- So where exactly are we, then? Let's start by having a look at what today's typical physically-based renderer is made of.
- Essentially, most renderers are *unidirectional path tracers*, with some other, more recent ingredients from around 2000.
- This is nicely illustrated by looking at the references in this quote from Matt Pharr's editorial to the ACM TOG special issue on production rendering.

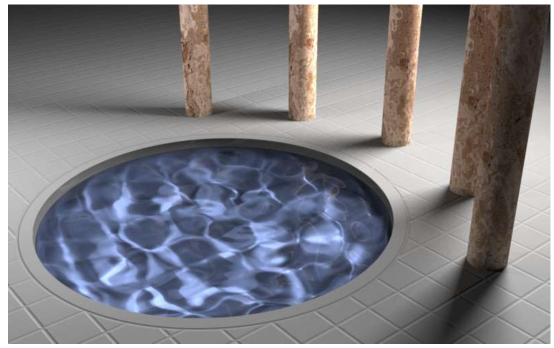


- So what happened after 2000 in research?
- Of course, there's been some important developments, notably denoising, possibly also path guiding, and, as Johannes Hanika from Weta puts it "*Collection of special purpose sampling algorithms that work super well in very specific scenarios… Each on their own may be incremental but the toolbox allows us to render many scenes well*"

ADVANCED LIGHT TRANSPORT

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- Why are advanced light transport algorithms not used in practice?



Metropolis Light Transport [Veach and Guibas 1997]

Computer Graphics Charles University

Jaroslav Křivánek – Rendering in architecture and product viz

corona (5)

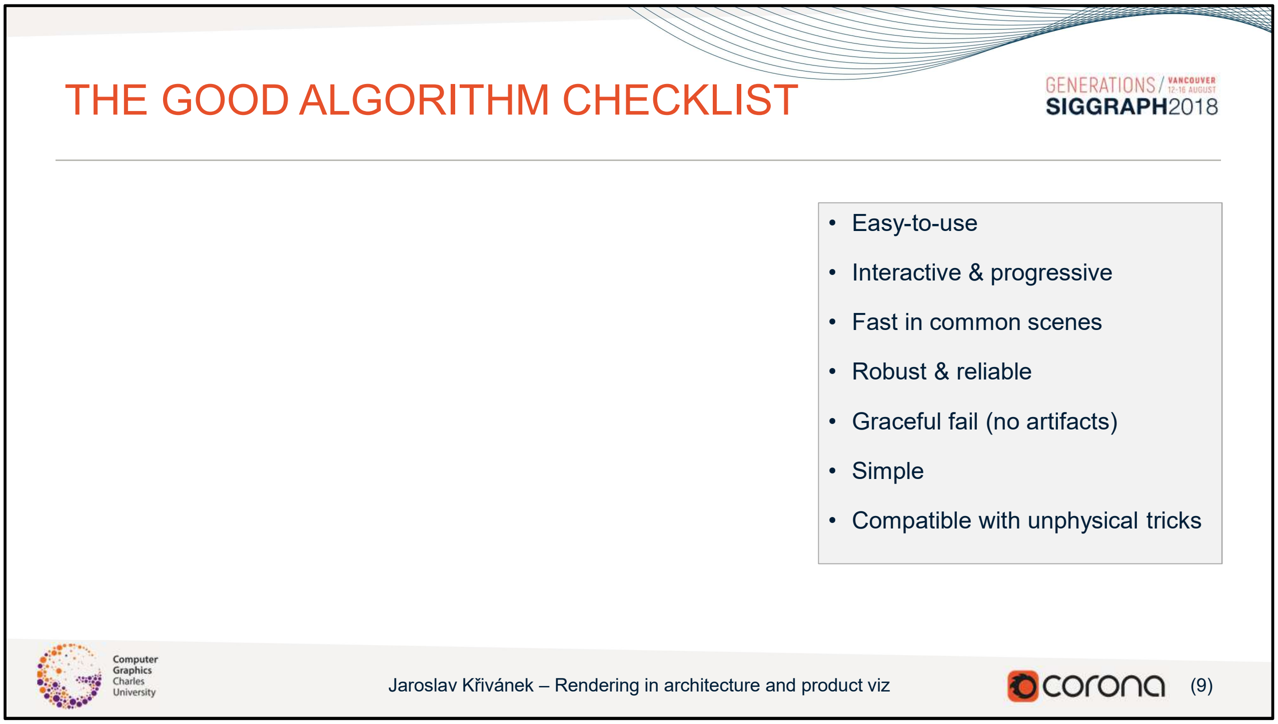
- Nonetheless, the fundamental light transport algorithm is still the simple, unidirectional path tracer – the underlying light transport technology is 30 years old!
- Why is it that none of the more advanced light transport methods, such as Metropolis Light Transport, Bidirectional path tracing or Vertex connection and merging are used in practice (at least not very often)?

WHY PATH TRACING…

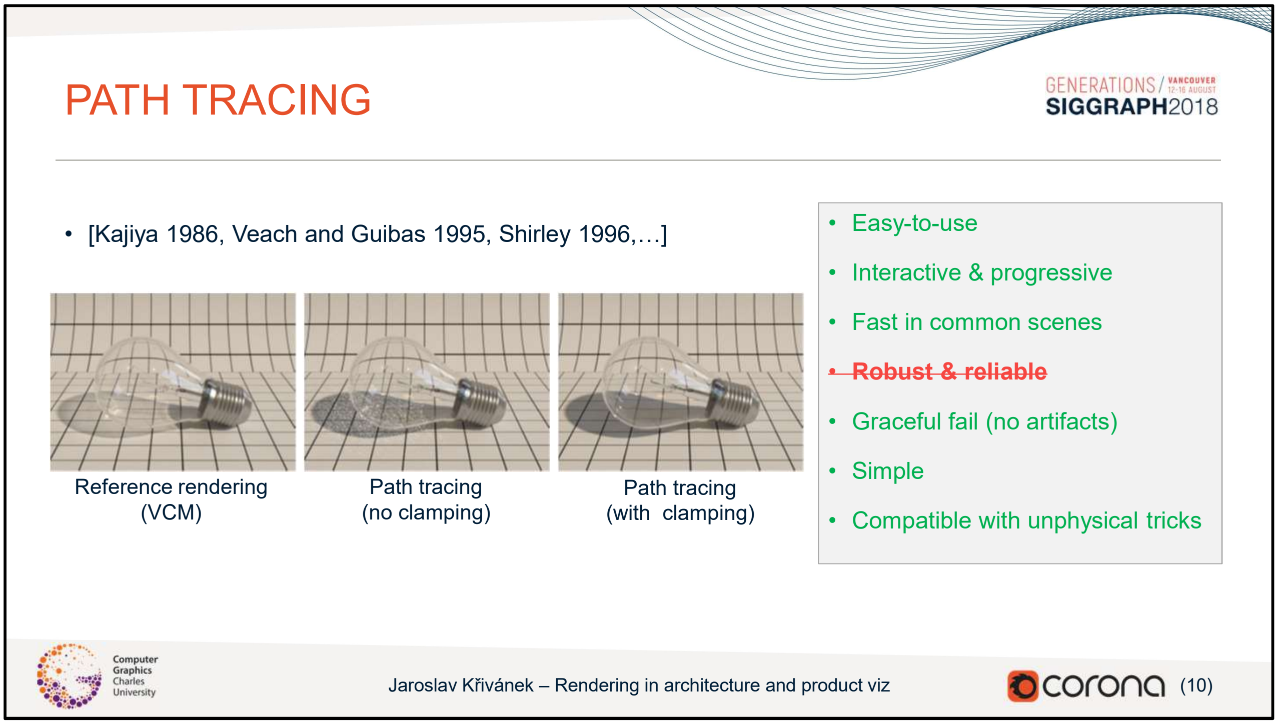
… when we have so many more advanced light transport algorithms?

- To answer that question, let's first assemble a checklist of properties that a "good" light transport algorithm should have, and let's match it to some of the existing algorithms.

- So here's the list.
- Many of the points have been discussed in the earlier parts of the course, so let me just list them without further justification.
- Let me also point out that this is not meant to be an exhaustive list. But it's a good start.
- For one thing, a good light transport algorithm does not necessarily need to be neither physically-correct nor unbiased, as long as the images are consistent and do not show artifacts (unless, of course, our aim is predictive rendering – but this is rarely the case in visualizations).
- On the other hand, some of the essential properties are:
    - Ease-of-use – no technical user parameters are allowed. At most, there can be one speed-quality slider.
    - Interactivity – time to first pixel on the screen should be minimized (at most a fraction of a second)
    - Progressivity – image quality steadily improves as the calculation progresses. One can inspect the image and resume rendering if needed at any point in time.
    - Speed – no overhead over the currently accepted solution (path tracing) is acceptable.
    - Robustness – the algorithm must handle scenes with complex lighting and geometry reasonably fast and with no artifacts.
    - Graceful fail – when the algorithm happens to fail, it must not show any conspicuous image artifacts.
    - Simple to implement and maintain
    - Flexible (compatible with non-physical tricks)

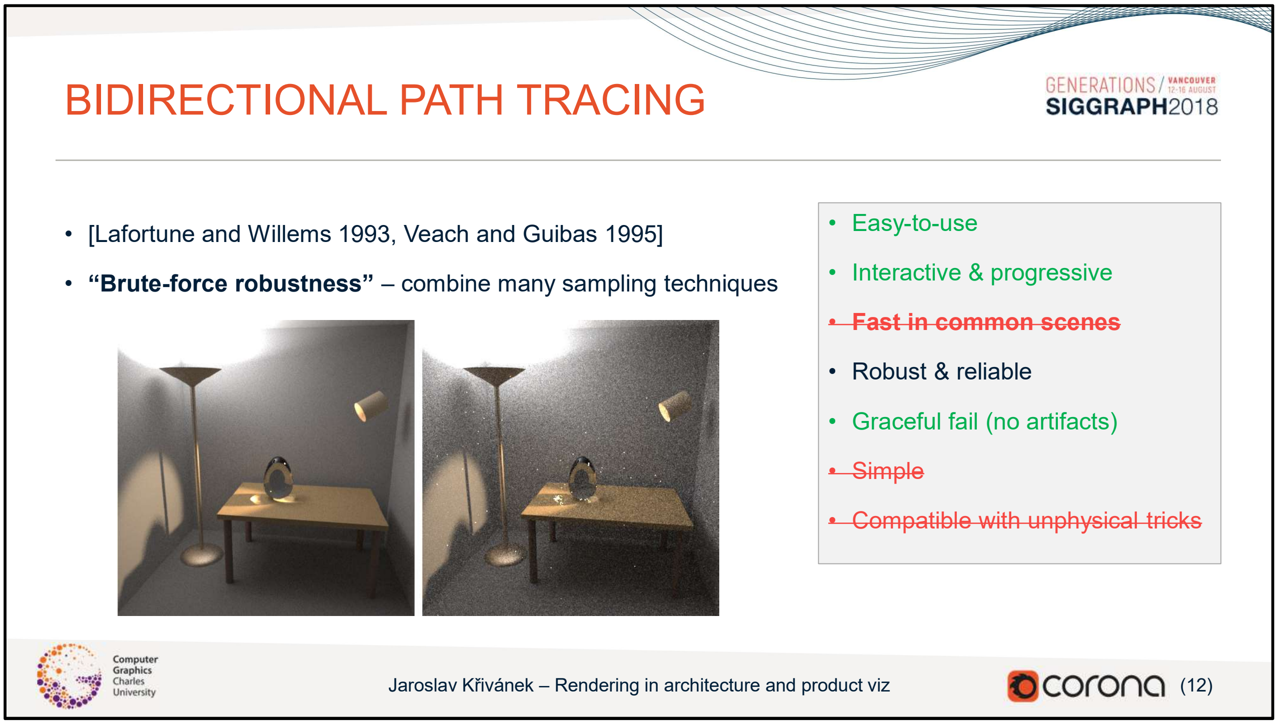


- So that's our checklist, let's see how the existing algorithms will compare.



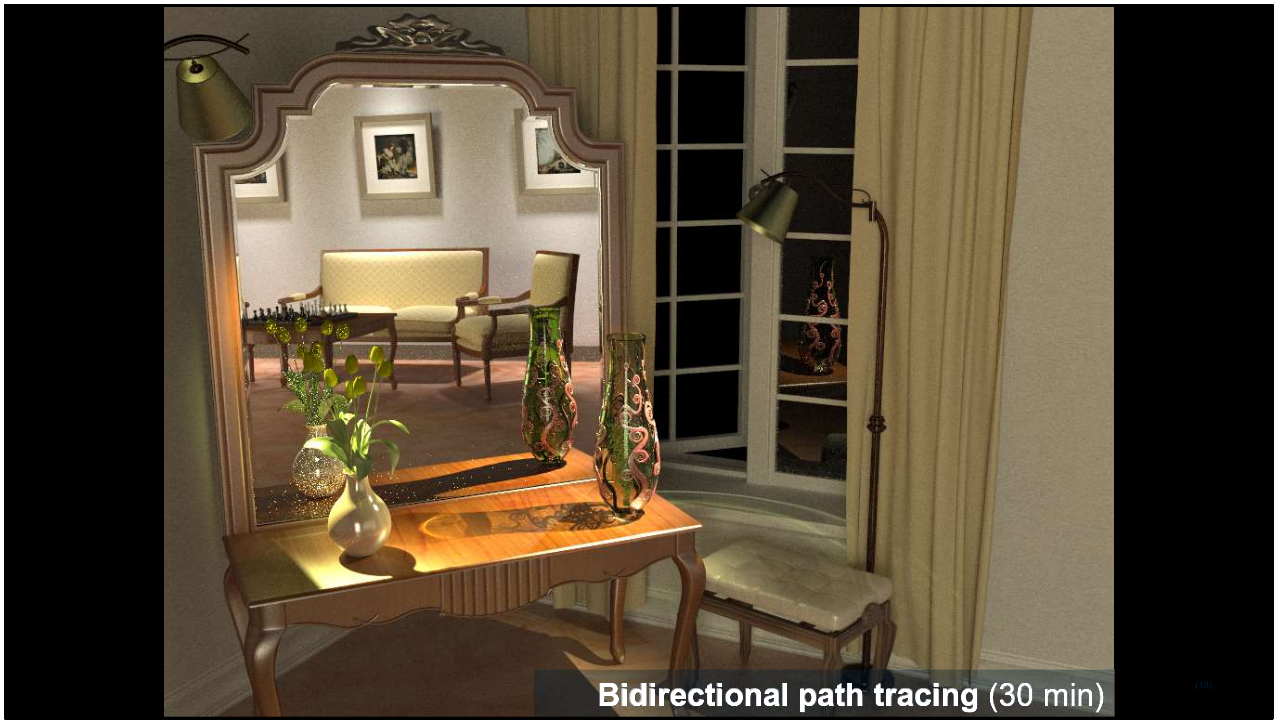
- Simple unidirectional path tracing – the technology used in most today's renderers – checks most of the boxes!
- The only issue is: The algorithm does not handle some complex light transport, notably caustics and strong, concentrated indirect illumination. In other words, PT is not robust in the presence of complex light transport.
- Instead of a caustic, path tracing will generate just a bunch of fireflies.
- These can fortunately be removed by selective energy clamping (which is what everyone does in practice), so the image in the end looks quite ok (that is, PT + energy clamping fails gracefully).
- And that's the whole story of why path tracing is so incredibly popular.

- Ok, path tracing is great, but sometimes, we just can't get away without caustics.
- The obvious choice for caustics rendering is of course photon mapping.
- The original photon mapping algorithm from 1996 was based on a set of heuristics, such as classifying photons into the global and caustics maps, the user had to tweak the number of photons in each map, and numerous other parameters.
- Finding the right parameter settings required lots of experience and trial-and-error.
- But when it worked, it worked really great. The algorithm could be very fast for the time.
- So in summary, photon mapping can be fast but fails most of the other boxes.



- Now, let's have a look at bidirectional path tracing. It's a complement of photon mapping in terms of pros and cons.
- It is a robust solution for caustics and other complex lighting effects. You can see the advantage over path tracing in this famous image due to Eric Veach.
- BPT is as easy-to-use as a simple path tracer, it is interactive and progressive, and it fails gracefully (after filtering out the noise).
- The problem is that the algorithm is generally rather slow: it's a bulldozer that is hard to stop, but it does not go very fast, at least compared to simple path tracing.

- Why is that? The reason is that BPT's robustness is achieved by combining many path sampling techniques, some generating paths from the camera, other from light sources.
- All these sampling techniques are used "just in case" – but many end up contributing very little to the final image in most cases.
- This is an example of the approach that I would call "**brute-force robustness**" – on the one hand, we make the algorithm more robust, but on the other hand, this comes at the cost of incurring significant overhead in simple cases.



- While BPT works great for caustics, it runs into troubles with reflected caustics as shown here.

- So we and other researchers have independently developed a more robust version, now commonly referred to as Vertex Connection and Merging, or VCM.

VCM

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- [Georgiev et al. 2012, Hachisuka et al. 2012]
- Photon mapping + bidir
- **"Super-brute-force robustness"**
  - combine even more sampling techniques than bidir
- Addresses robustness
- **Makes overhead even worse**

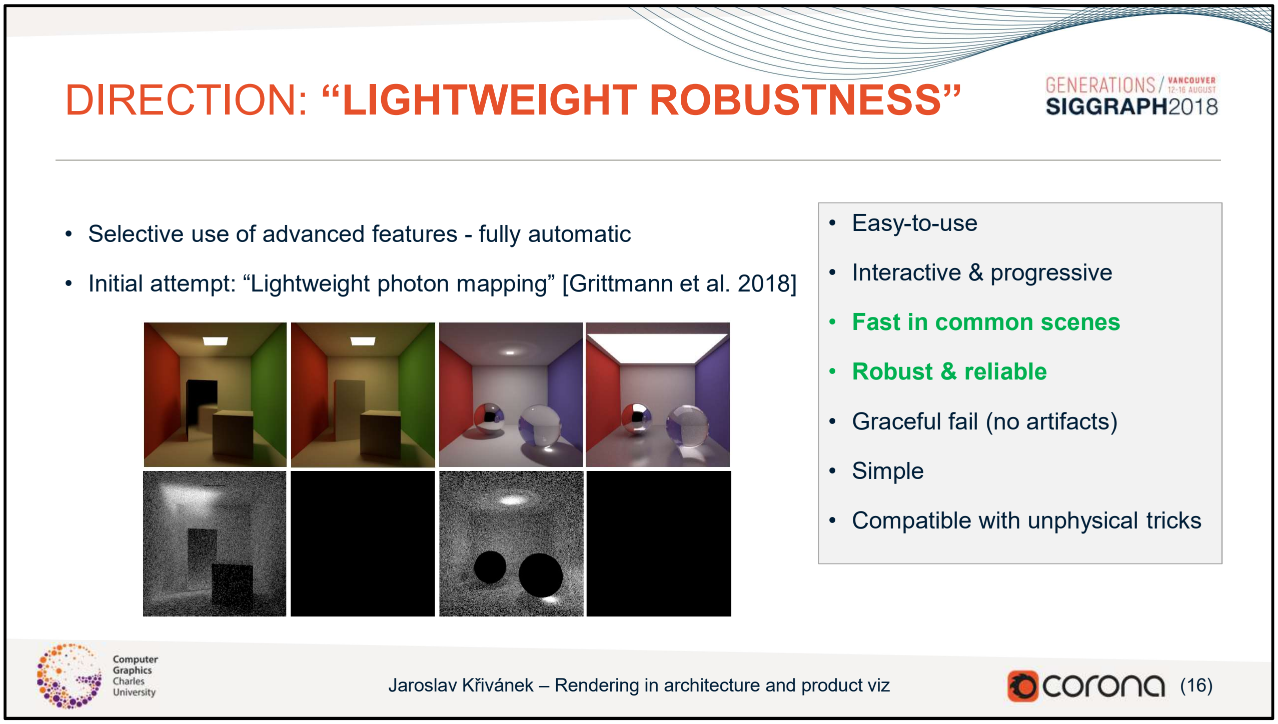
- Easy-to-use
- ~~**Interactive**~~ & progressive
- ~~**Fast in common scenes**~~
- **Robust & reliable**
- Graceful fail (no artifacts)
- ~~Simple~~
- ~~Compatible with unphysical tricks~~

Computer Graphics Charles University

Jaroslav Křivánek – Rendering in architecture and product viz

corona (15)

- VCM achieves robustness by combining BPT with photon mapping.
- By doing that, it can render reflected caustics well.
- But it does that at the cost of making the overhead in simple scenes even worse than in BPT.
- So VCM addresses one issue (lack of robustness) at the cost of making another issue (overhead in common scenes) much worse.
- I believe this is not the right direction if we hope to ever achieve a practical solution.



- This finally brings me to my first research direction, or rather research objective, which I've labelled `*lightweight robustness*'.
- We need algorithms that are robust, but without incurring overhead in common cases.
- Such algorithms should perhaps run the advanced sampling techniques selectively only when needed, and this should be fully automatic without the user having to worry about what is happening under the hood.

- An initial work in this direction is our this year's EGSR paper "lightweight photon mapping", where our goal was to design a "minimum intrusive caustics renderer on top of a unidirectional path tracer".
- The algorithm can decide how many photons to shoot, where they are need, if at all – as you can see on the illustration (top is the scene, bottom the photon density – white means many photons, black means no photons).

PATH GUIDING

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- [Jensen 1995, Vorba et al. 2014, Mueller et al. 2017]
- **Online adaptive sampling** in path space

- Easy-to-use
- ~~Interactive &~~ progressive
- ~~Fast in common scenes~~
- ~~Robust & reliable~~
- ~~Graceful fail (no artifacts)~~
- ~~Simple~~
- Compatible with unphysical tricks

Computer Graphics Charles University

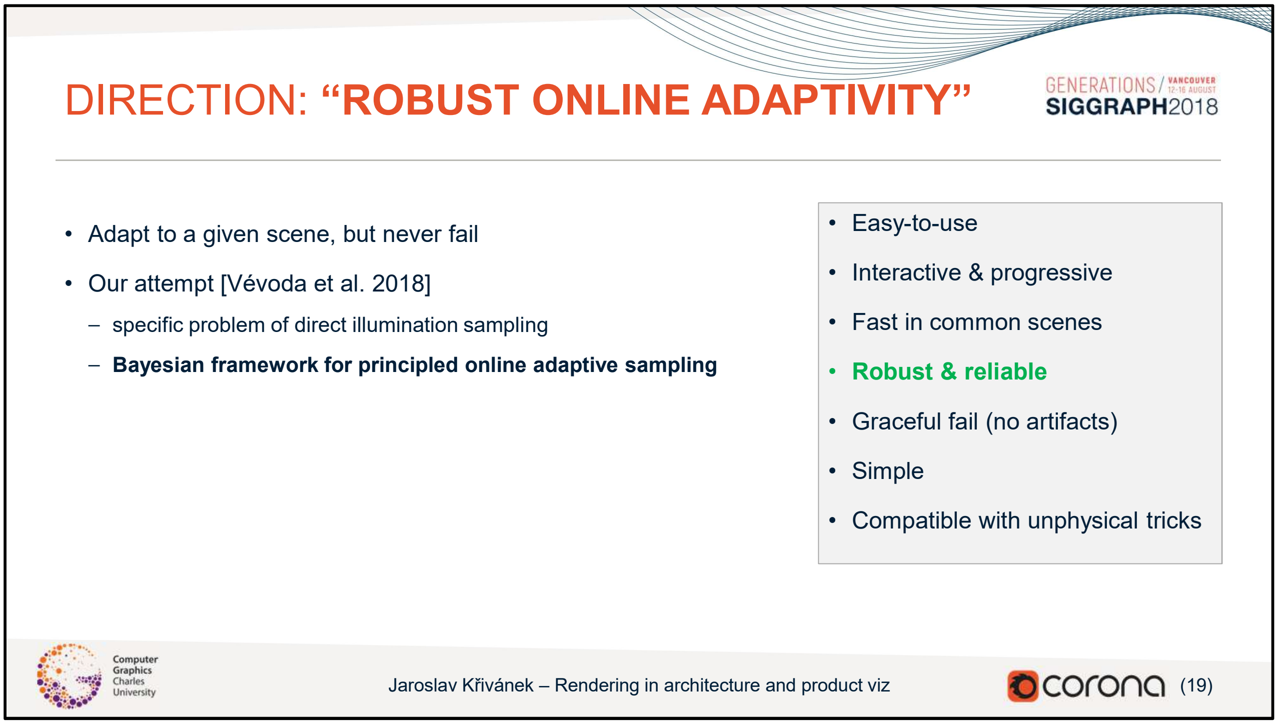
Jaroslav Křivánek – Rendering in architecture and product viz

corona (17)

- Let us continue the journey.
- One class of algorithms that is being talked about today is the so called `path guiding'.
- The idea is to guide paths in the path tracer towards locations where light actually comes from – so it is an adaptive sampling algorithm in the path space.
- The basic idea, again, is due to Henrik Jensen.
- In 2014, we have shown how some fundamental tools from machine learning can be used to refine the algorithm to achieve much better performance.

- The discussion of path guiding is currently ongoing and the idea has the potential to replace (or rather extend) path tracing as the next universal solution. More research work is, however, necessary before we get there.
- For the discussion on this slide, we address specifically the algorithm in our paper [Vorba et al. 2014].
- Why is it not yet the 'ultimate' solution?
- First, incurs significant overhead (precomputation, rendering).
- Second, and more importantly, it does not fail gracefully.

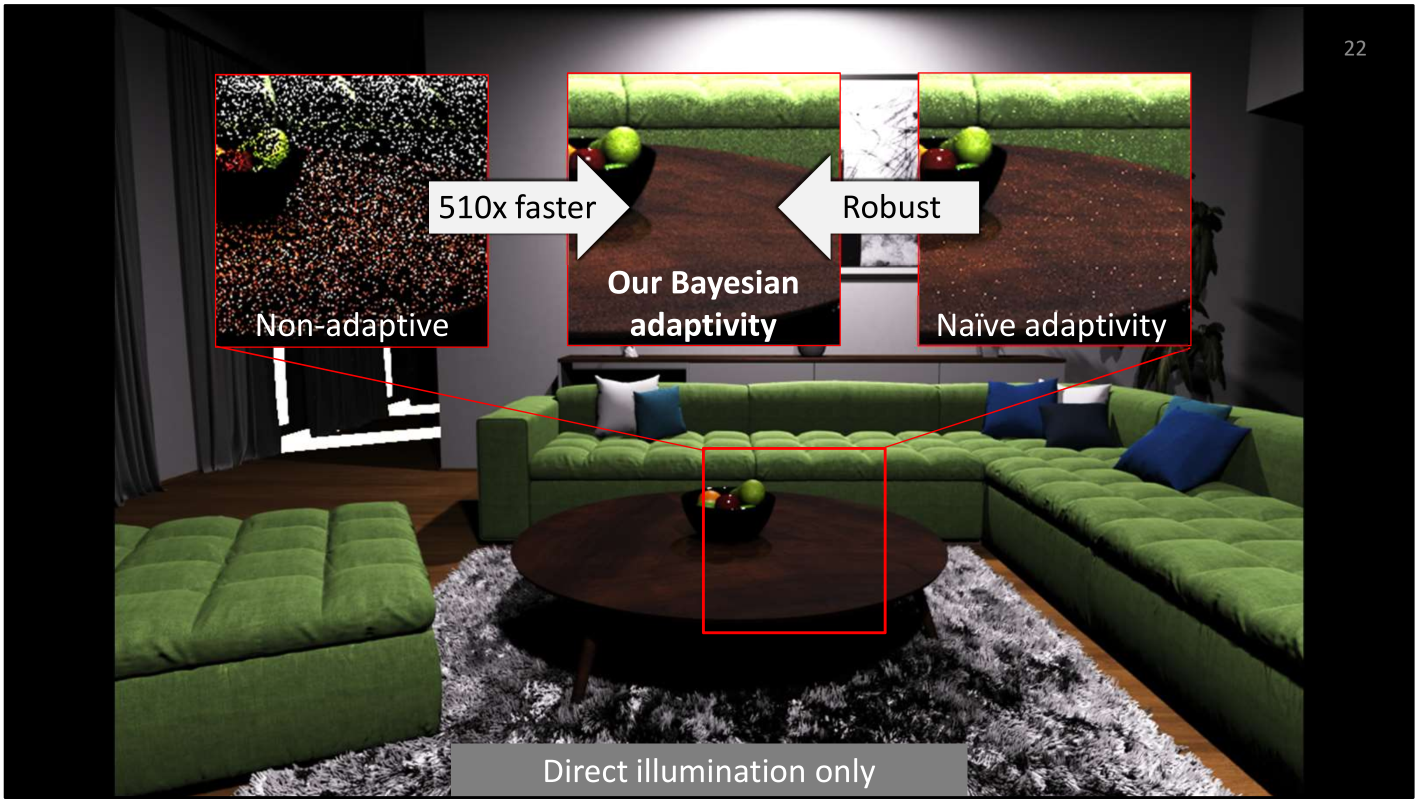
- In this example from the paper of Thomas Muller and colleagues, we see that our path guiding produces uneven noise – there are conspicuous islands of high noise which is difficult to remove by filtering.
- So despite the fact that the numerical error may be small, this flaw prevents the algorithms from being used in practice is the form described in the paper (rumors have it, though, that this could be fixed).



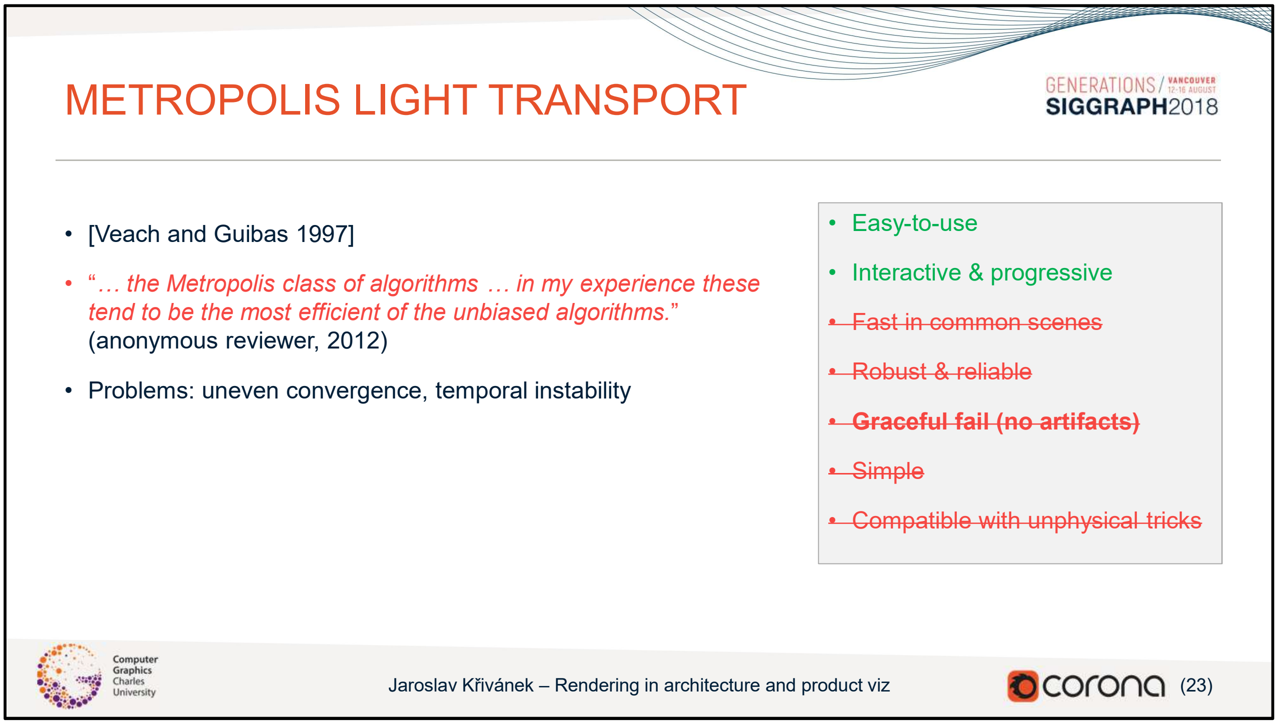
- What path guiding brought to our attention is that *adaptive sampling can be really powerful*.
- Adaptive sampling refers to the simple idea that information (samples) gathered during rendering the scene are used to make the subsequent decisions "smarter" in some way.
- Many adaptive sampling algorithms have been proposed in light transport in the past.
- But *adaptivity has traditionally suffered from the lack of robustness*.
- The reason is that early on in the rendering, we have only a few samples contaminated by noise - when should we start trusting them? How quickly should we adapt? etc.
- These are fundamental questions that, in my opinion, cannot be answered without a good theoretical basis.
- We need theoretical frameworks to underpin adaptive algorithms that will enable achieving adaptivity without giving up robustness.
- At this SIGGRAPH, we are presenting one attempt in this direction.
- Again, we turn to machine learning because extracting useful information from uncertain and noisy data in a robust way is a fundamental objective of this field.
- We propose to view of adaptive sampling as Bayesian inference.

- The specific problem that we apply our framework to is sampling of …

- … direct illumination. In fact, this algorithm will become the default direct illumination solver in the next version of Corona.



- We can see that our Bayesian adaptive sampling algorithm provides a substantial speedup over non-adaptive sampling, while being more robust than previous adaptive approaches.



- And of course, any discussion of advanced light transport wouldn't be complete without Metropolis Light Transport.
- For some time, this algorithm managed to keep the aura of being the 'ultimate light transport solution'.
- In fact, still around 2011/2012, we had some hard time publishing our light transport work because some reviewers believed that MLT had already solved the problem.
- Fortunately, Wenzel Jakob has released his MLT implementation in Mitsuba, which made it possible to see for oneself just how disappointing the performance of MLT really is in practice.

MLT WITH MANIFOLD EXPLORATION

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

MLT + Manifold exploration [Jakob and Marschner 2012]

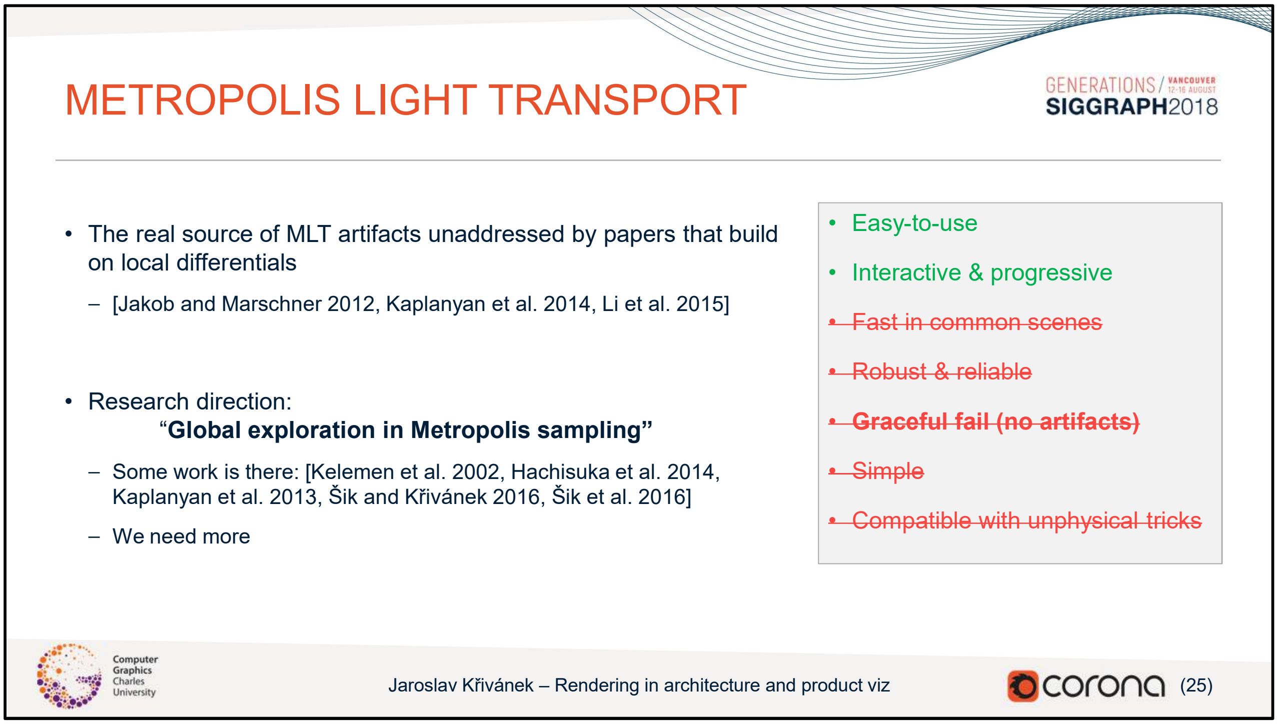
Reference

Computer Graphics Charles University

Jaroslav Křivánek – Rendering in architecture and product viz

corona (24)

- MLT's main problem is uneven convergence, lots of nasty image artifacts, and temporal instability as shown in the video.



- Working on MLT-style algorithms for some time with my student Martin Sik, I got entirely convinced that the real source of the problems is `insufficient global exploration of the path space'.
- Let me explain – what MLT does is that once finds some highlight, it will keep sampling it so it may end up rendering that one highlight really well.
- But what MLT cannot do well is finding all the various highlights in the first place – that is what I mean by bad global exploration.

- Despite this observation, there's been quite some work building on the idea of using local path space differentials for better *local exploration* in MLT.
- While these works provide some interesting insights into the structure of the path space, they fail to address what I believe is the real root of MLT's problem – bad global exploration.
- So here is a challenge for light transport enthusiasts that do not like to take easy paths: solve the global exploration issue in Metropolis sampling.
- This can be hard but the reward would be sweet: Metropolis could become the ultimate light transport solution that would finally send path tracing to a well-deserved retirement.

- Of course, there's been some works in this direction already…

PRIMARY SAMPLE SPACE MLT

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

PSS-MLT [Kelemen et al. 2002] | Reference

Computer Graphics Charles University

Jaroslav Křivánek – Rendering in architecture and product viz

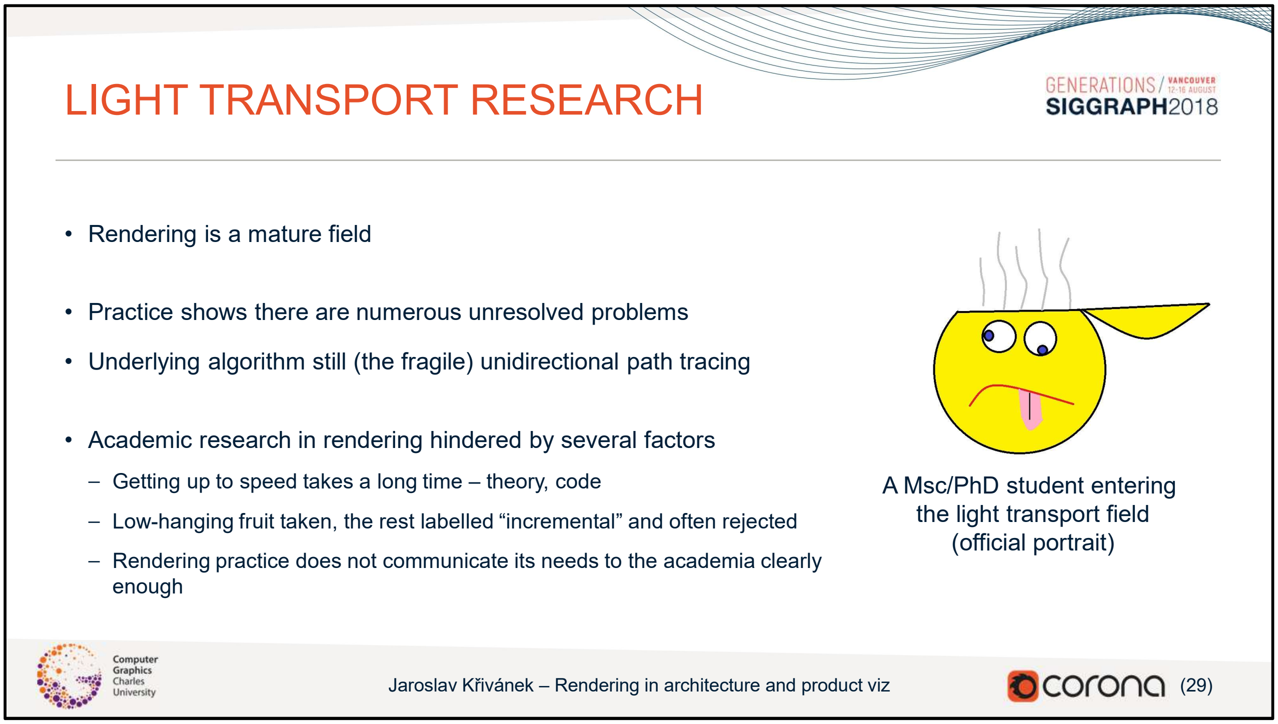
corona (26)

- Here, you see the results of Kelemen's Primary sample space MLT.
- The 'large step' mutation in the algorithm makes global exploration much better as you can see in this result.
- But the temporal instability is still significant.

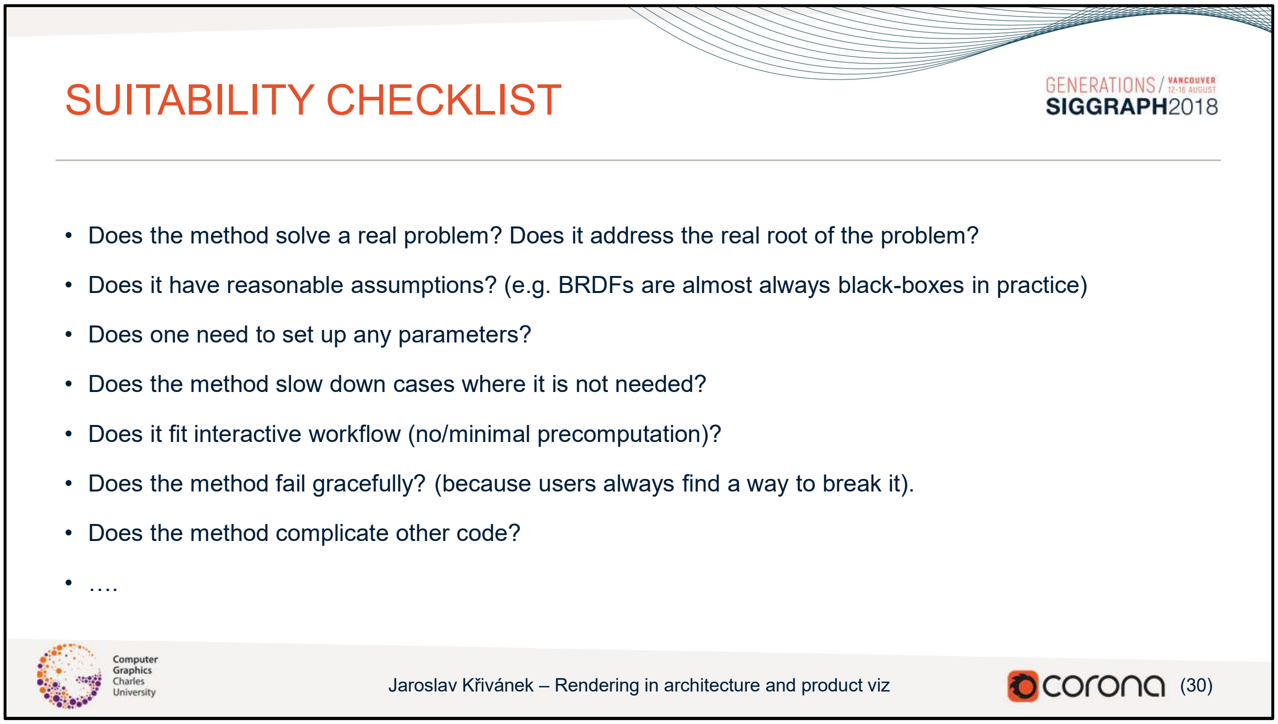
- Based on the realization that bad global exploration is at the root of MLT problems, we have proposed the Metropolized VCM algorithm [Šik et al. 2016].
- The results are much better, the algorithm converges predictably, produces noise that can be removed by filtering.
- But since the basic building block is VCM, the overhead over path tracing is still too large, so more work is needed here.

LIGHT TRANSPORT RESEARCH

What will it take to make more meaningful progress?

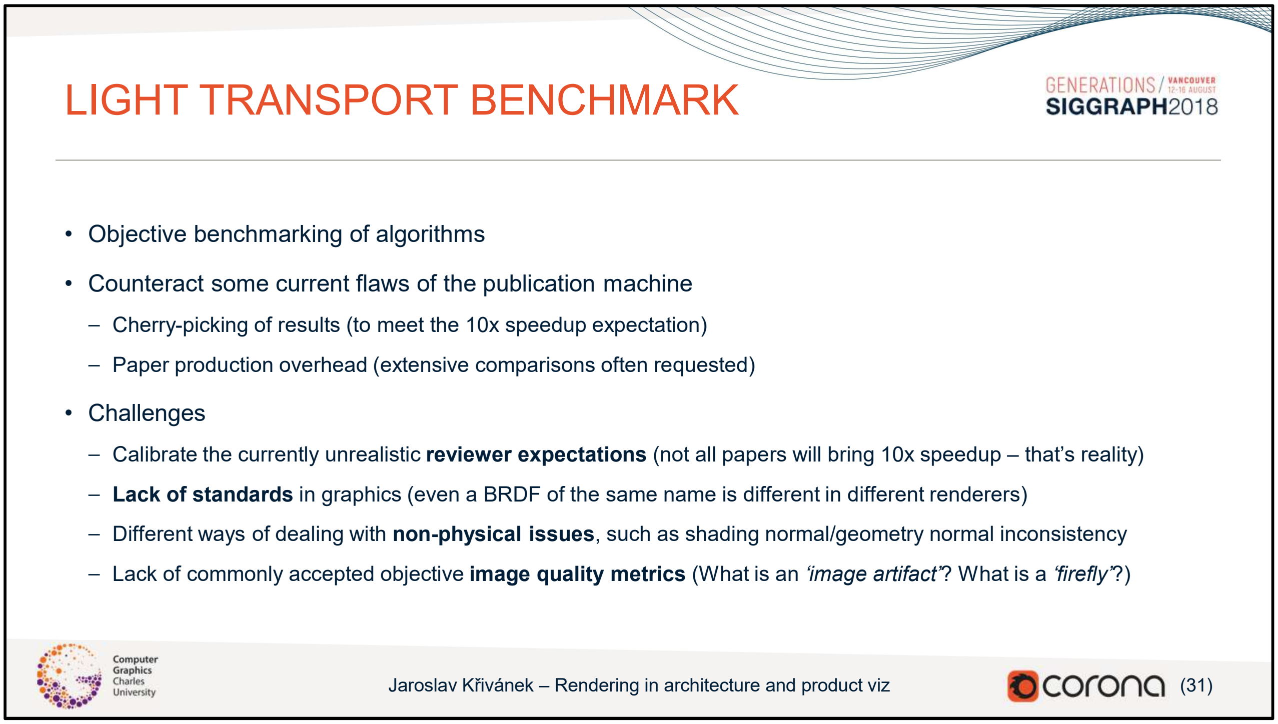


- Ok, so based on this discussion, let me give some direction for light transport research.
- First, rendering is a mature field but there are still unresolved problems. Performance of current solutions is underwhelming, we still rely on the fragile unidirectional path tracing. This should be enough to encourage us to pursue research in this field.
- A word of warning, though.
- Light transport has become fairly advanced, getting up to speed can be difficult both in terms of the theory that needs to be assimilated and in terms of the code that needs to be written (many thanks to Matt Phar and Wenzel Jakob for PBRT and Mitsuba).
- The low-hanging fruit has been mostly reaped. It is not uncommon that a method building on the large body of existing work ends up labelled ‘incremental’ and dismissed in the review process. (The rendering research community itself is to blame here.)
- And lastly, it is difficult for an academic researcher to know what the needs and constraints if the industry are – this course is one attempt to fix this.



- Concerning the industry needs – I've already given the 'good algorithm checklist'. I like checklists so here's another one here.

- Some extra information on the last point above: production renderers are already packed with complex code to the limit of feasibility. Asking to implement something on top of the existing infrastructure might be difficult since it needs to handle all the specifics we do (AOVs, blackboxes, fakes, DR, IR, shadowcatcher, adaptivity, lightmixing, …).

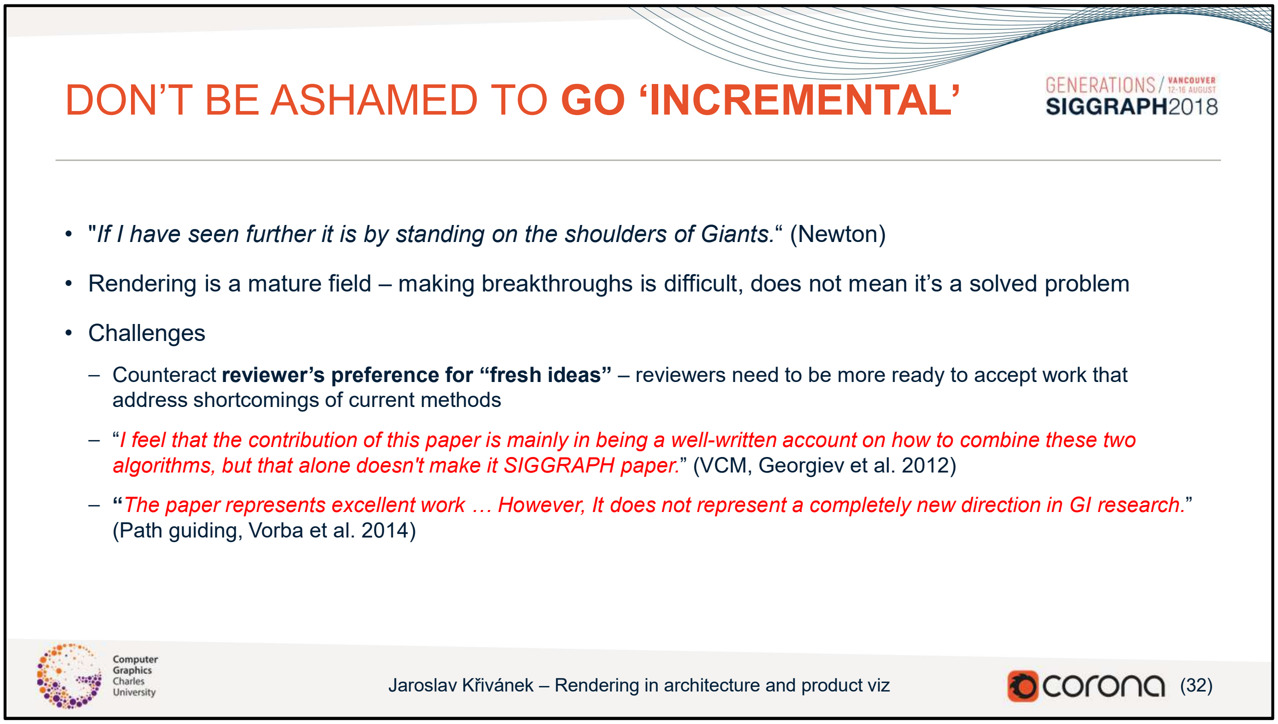


- One thing that we desperately need to be able to make serious progress in light transport, is an objective benchmark.
- I will not discuss how it should be implemented. Instead I will focus on why I believe it is needed and what challenges are in the way.

- Why do we need a benchmark?
- Mostly because the publication process itself cannot ensure objective evaluation. At some point in the past, inflation of the expected speedups started – an order of magnitude speedup is now the norm for acceptance to SIGGRAPH. Is this realistic? I don't believe so. But the review process gives a clear incentive to prefer such statements in the papers. So the only way around it if researchers wants to publish their work is to carefully cherry-pick the reported results. And by doing this, we make the speedup expectations even more unrealistic. It's a vicious circle, it's not helping progress in the field, students and advisors alike have (or at least should have) serious moral reservations to accepting this game. I strongly believe that an objective benchmark would be a very effective way out of this circle.
- An anectodical evidence of the degree of dishonesty in the result reporting in light transport papers is the difficulty that we, developers of a commercial renderer, have to explain to a naïve user the reasons why the algorithm in the latest EGSR or SIGGRAPH paper is not as great as the paper makes an unaware reader believe. And no, unfortunately, we will not implement it in the next release of Corona.

- Another point is that extensive evaluations are often asked by the reviewers, which makes the paper production cost often unreasonably high.
- Some good ideas – especially from the industry - may end up unpublished because it's just too much hassle to meet the expected production and polishing standards. Again, an objective and easy-to-use benchmark could help alleviate this problem by removing the burden of running all the comparisons again and again.
- Furthermore, we can draw inspiration from other fields such as computer vision – the CVPR papers are arguably not as polished as SIGGRAPH ones, and still, the field of computer vision has been making tremendous progress.

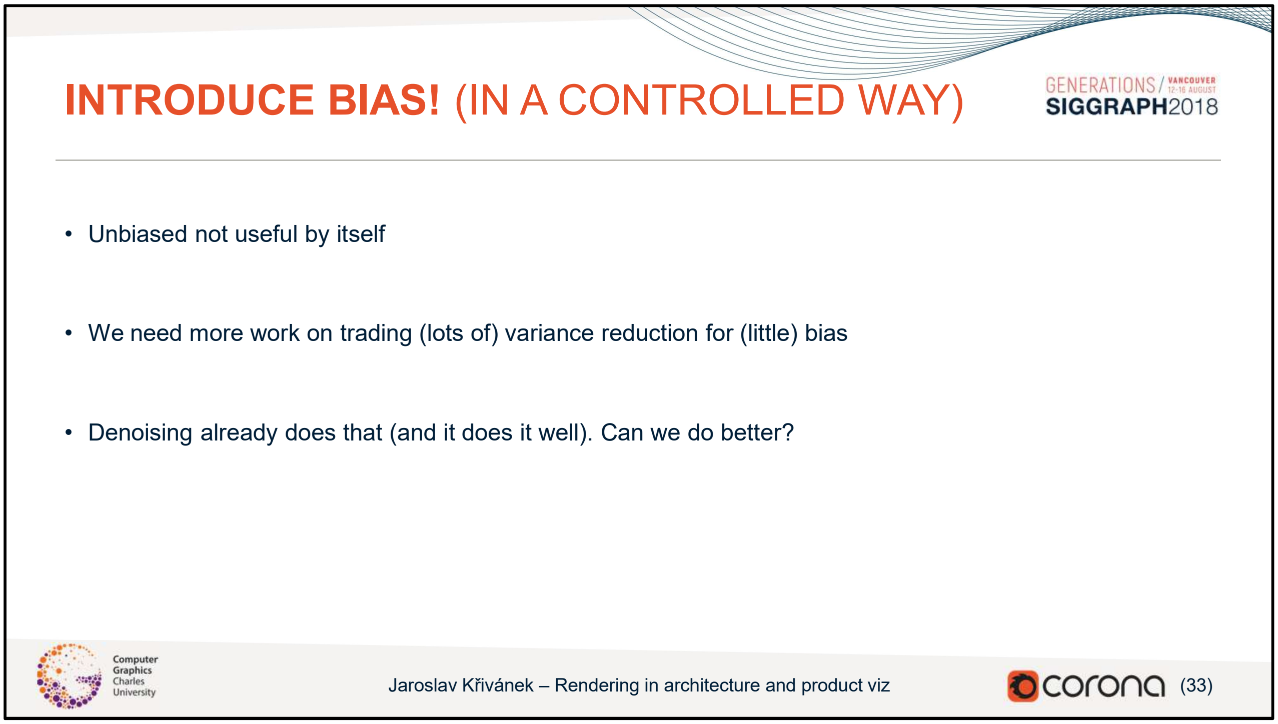
- What are the major challenges associated with building a benchmark?
- The first challenge again concerns the reviewing culture - we need to recalibrate the reviewer's expectations. 10x speedup in every paper is unrealistic. (If we could do 2x speedup every other year, we'd already be at Moore's law. I am quite certain that should be good enough to warrant publication.)
- The remaining challenges are technical – but also fundamental.
    - First, since we do not adhere to standards in graphics, we have no common ground for truly fair comparisons.
    - Second, practical renderers are full of non-physical constructs (think of shading normal and bump-mapping), and each renderer deals with these in its own way. This again complicates fair comparison.
    - Finally – and perhaps most importantly – we have no commonly accepted "rendered image quality metric". Define an "image artifact"? Define a "firefly"? We have lots of work to do here…



- My other recommendation would be to work on improving existing ideas rather than necessarily trying to propose new ones all the time.
- We already have a long list of advanced light transport algorithms, but we still rely on unidirectional path tracing in practice. There is no need to make the list of unused advanced algorithms even longer.

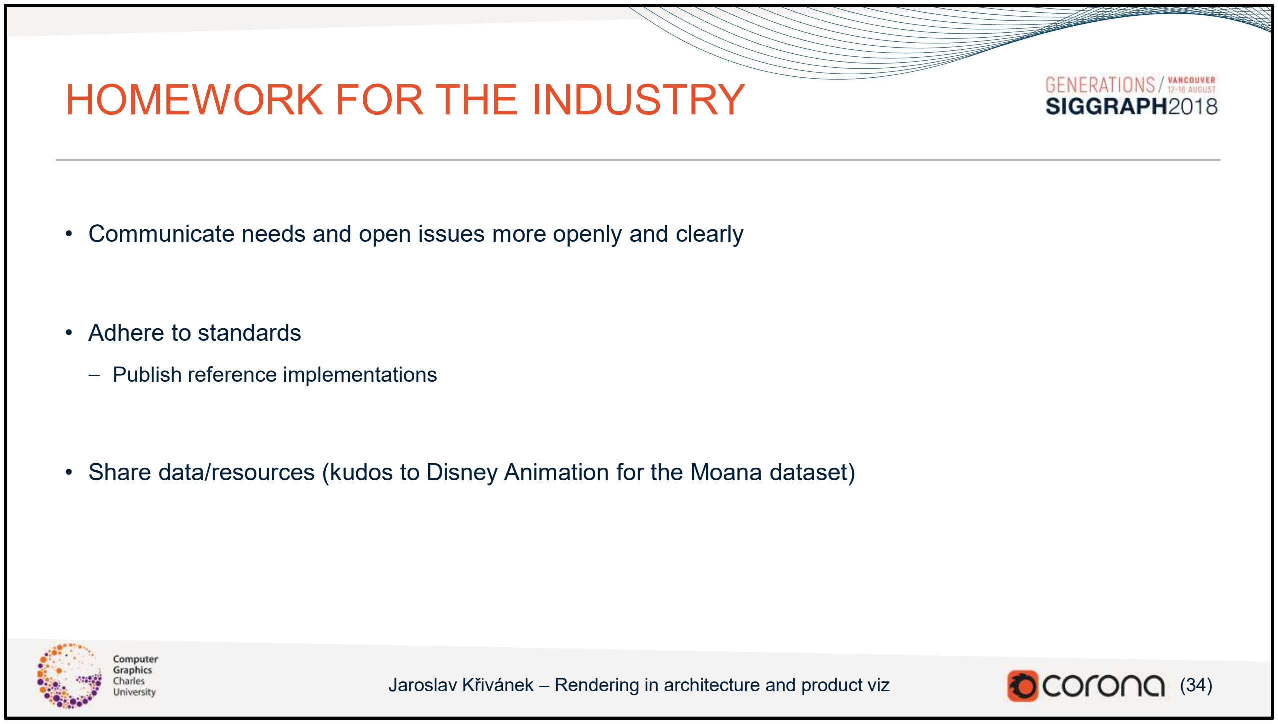
- For this to work, we again need to adjust the reviewing culture. Researchers must feel safe to do that – valuable improvements of existing techniques should not be rejected on the basis of 'having too small delta'.
- The slide provides related quotes from reviews of some of my papers. I will leave the reader to judge for herself.

- As an anecdote, we can say that deep learning is an utterly incremental idea – it's a rehash of what was invented some 30 years before. And Kajiya's rendering equation – that's an outrageous copy right from the transport theory literature!



- “Unbiased” has been for a long time considered kind of a certificate of quality of a light transport algorithm.
- But unbiasedness by itself is not useful.
- What we care about is low visual error after a short rendering time – this is something that unbiasedness does not address at all.

- Science has know for a long time that variance can be traded for bias.
- What we need is to find a way to trade lots of variance reduction, for little, controlled amount of bias.
- Denoising is a great example of this general idea (though it’s rarely though about in this way).
- But can we do better than a simple image postprocess by taking advantage of the tremendous amount of information we have about the scene?



- There's a homework for the industry, too.
- First, we need to communicate more clearly an openly the issues and the constraints that we deal with in practice.
- We should adhere to standards whenever possible. We should publish reference implementations. (This is complicated by IP issues.)
- We should share data – many thanks to Disney Animation for releasing the Moana dataset.

CONCLUSION

- Time to wrap up.
- We have seen that current practical renderers rely on venerable light transport technology – namely unidirectional path tracing. This brings difficulties when dealing with complex light transport such as caustics.
- We have shed some light on why none of the advanced algorithms is a good replacement for path tracing – the reason being mostly unreasonable overhead in common scenes, complex implementation, and sometimes bad image artifacts.
- To make a steady progress in light transport, I believe we should create and adhere to an objective light transport benchmark, and we should recalibrate the expectations in the reviewing process to better match reality.
- When developing a new algorithm, we should make sure that we understand the real root of the problem and that we are not addressing it at the cost of making other problems even worse (namely – minimum overhead for advanced algorithms).
- And eventually, once we're done with light transport, we should change gears to realistic content, because that's currently one of the main limiting factors of realism.

THANK YOU!

- Acknowledgments
  - Johannes Hanika
  - Funding: Czech Science Foundation (16-18964S)
  - Colleagues and students from Charles university & Corona developers
  - Course presents
  - You

EXTRA SLIDES

REVIEW QUOTES FOR "FRESH IDEAS"

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- "*I feel that the contribution of this paper is mainly in being a well-written account on how to combine these two algorithms, but that alone doesn't make it SIGGRAPH paper.*" (VCM, Georgiev et al. 2012)
- **"***The paper represents excellent work … However, It does not represent a completely new direction in GI research.*" (Path guiding, Vorba et al. 2014)
- "*… little algorithmic novelty is present: the idea of building adaptive distributions is almost 20 years old. This algorithm is a nicer implementation of the same basic idea.*" (Path guiding, Vorba et al. 2014)
- *"There wasn't … something surprising that I learned from this paper, which I expect from a SIGGRAPH-level submission.*" (Metropolised VCM, Šik et al. 2016)

REVIEWERS JUDGING PRACTICAL MERIT

- "*The authors argument that the presented approach, however, can also handle spatially varying BRDFs and is algorithmically simpler. These are correct, yet not very strong arguments, since* ***spatially … varying BRDFs are a very special scenario****.*" (Filtered importance sampling submission to EGSR 2007)
- "*I'm not convinced long-term, with the use of ultra-fast raytracers, that an approach like this will have small enough overheads to be useful.*" (Path guiding, Vorba et al. 2014)
- "*While many path tracers implement some form of Russian roulette (since we need to terminate path generation at some point), nearly none of these path tracers implement splitting even though Arvo and Kirk introduced them since 1990. There seems to be some big obstacles preventing people from implementing splitting in their path tracers.*" (ADDRS, Vorba and Křivánek 2016)
  - Corona, V-ray, Arnold, etc. all use splitting
- "*… in my opinion the work does not demonstrate enough practical value even for further qualitative theoretical analysis.*" (about an algorithm running in Corona for two years, improving performance 1.5x – 2x on average)