Eurographics Symposium on Rendering 2019
T. Boubekeur and P. Sen
(Guest Editors)

Volume 38 (2019), Number 4

Quantifying the Error of Light Transport Algorithms, Supplemental Material

A. Celarek[1,2], W. Jakob[3], M. Wimmer[2] and J. Lehtinen[1]

[1]Aalto University, Finland [2]TU Wien, Austria [3]École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

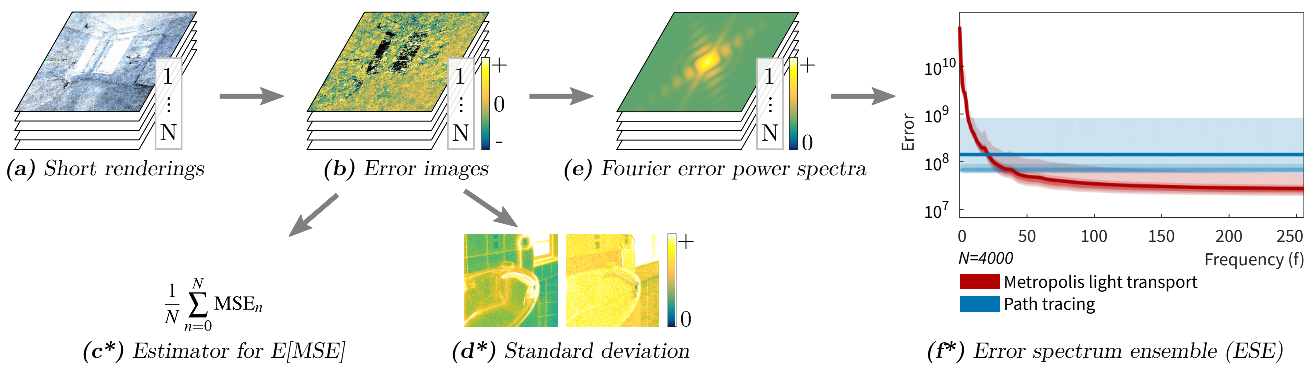


Figure 1: *Flow chart of the proposed method with outputs (*): A long rendering process is partitioned into many short runs (a) which are used to estimate error images (b). These are used to calculate a reliable estimate of the expected mean square error (MSE, c*), that could e.g. be used to rank a set of different rendering algorithms. The error images (b) are also used to generate a standard-deviation-per-pixel visualization (d*), which shows which of several competing algorithms is best for a specific lighting situation. Finally, Fourier power spectra (e) are computed and combined into the error spectrum ensemble (ESE, f*) that plots the expected error and outliers with respect to frequency, visualizing for instance correlation between pixels.*

Abstract

This paper proposes a new methodology for measuring the error of unbiased physically based rendering algorithms. The current state of the art includes mean squared error (MSE) based metrics and visual comparisons of equal-time renderings of competing algorithms. Neither is satisfying as MSE does not describe behavior and can exhibit significant variance, and visual comparisons are inherently subjective. Our contribution is two-fold: First, we propose to compute many short renderings instead of a single long run and use the short renderings to estimate MSE expectation and variance as well as per-pixel standard deviation. An algorithm that achieves good results in most runs, but with occasional outliers is essentially unreliable, which we wish to quantify numerically. We use per-pixel standard deviation to identify problematic lighting effects of rendering algorithms. The second contribution is the error spectrum ensemble (ESE), a tool for measuring the distribution of error over frequencies. The ESE serves two purposes: It reveals correlation between pixels and can be used to detect outliers, which offset the amount of error substantially.

CCS Concepts

• ***Computing methodologies*** → ***Ray tracing;***

Contents

1 Introduction

2 Results for Complex Scenes

2.1 Overview, solutions without using a reference

2.2 Torus with N=16000

2.3 Comparison with ESE / SDpp using a reference

2.4 Influence of N, without a reference

2.5 Influence of N, while using a reference

2.6 Relative Error

3 Parameterized Box Scene

4 Adaptive Sampling, Consistent and Sub-$\Theta(1/\sqrt{N})$ Convergence Algorithms

5 MLT configurations

5.1 Seed-pool size (number of luminance samples)

5.2 Number of seeds per pool

5.3 Chain length

5.4 Comparison with Energy Redistribution Path Tracing

5.5 PSSMLT Parameters

1. Introduction

This supplemental material was partly auto-generated and is therefore quite extensive. Of particular interest might be the sections about MLT seeding at the relative end. Otherwise we recommend to open the PDF twice for easier comparisons.

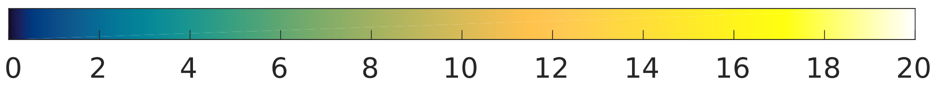
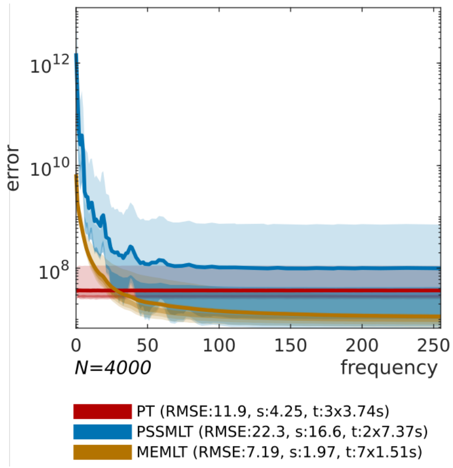


Figure 2: *Colour mapping used throughout the publication for SDpp.*

2. Results for Complex Scenes

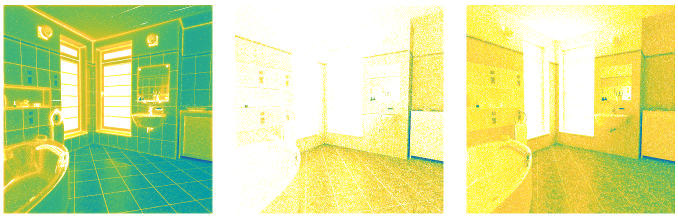
2.1. Overview, solutions without using a reference



(a) *Ensemble power spectrum*

(b) *Reference*

MLT *PSSMLT* *PT*

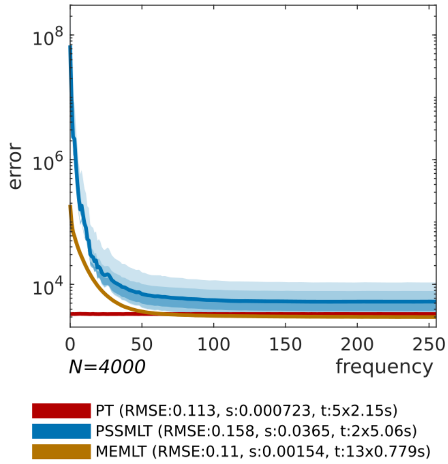


(c) *Standard deviation per pixel (SDpp)*

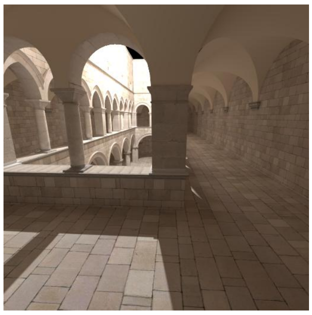
(d) *Example short rendering*

(e) *Proxy result*

Figure 3: *ESE, SDpp and other data for bathroom scene (N=4000, solution using no reference).*

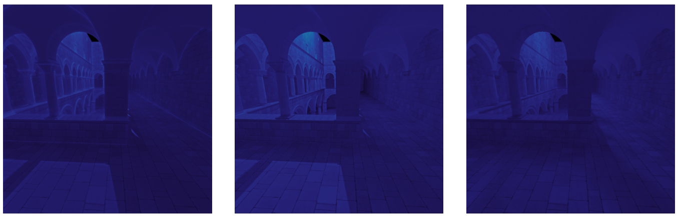


(a) *Ensemble power spectrum*



(b) *Reference*

MLT *PSSMLT* *PT*

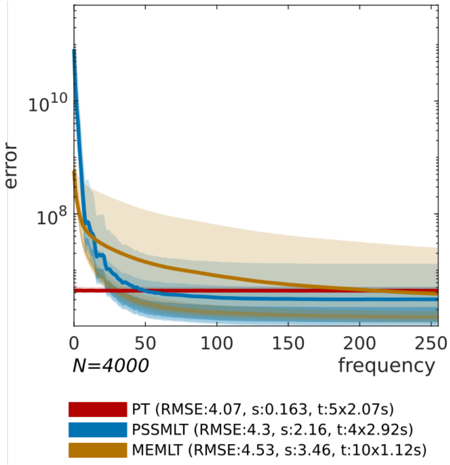


(c) *Standard deviation per pixel (SDpp)*

(d) *Example short rendering*

(e) *Proxy result*

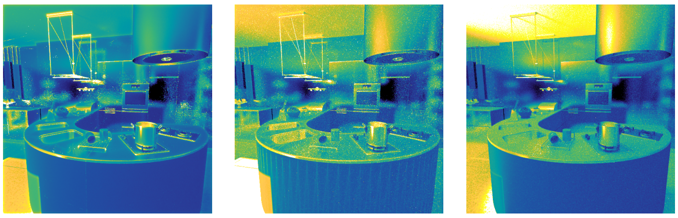
Figure 4: *ESE, SDpp and other data for sponza scene (N=4000, solution using no reference).*



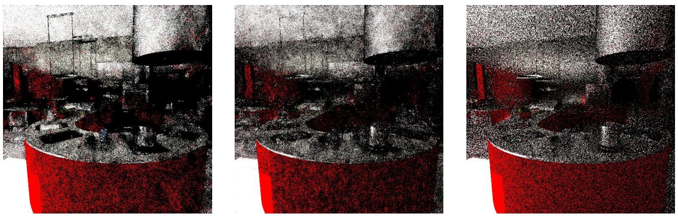
(a) *Ensemble power spectrum*

(b) *Reference*

MLT *PSSMLT* *PT*



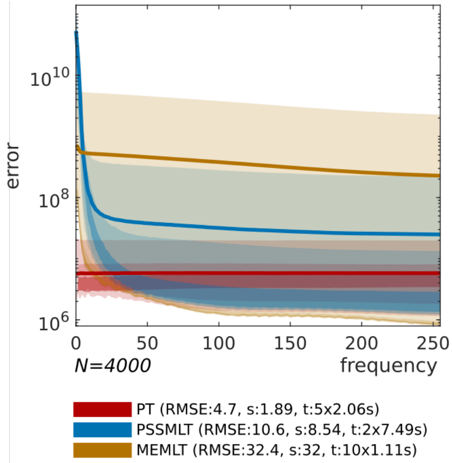
(c) *Standard deviation per pixel (SDpp)*



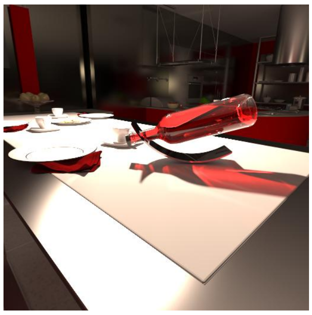
(d) *Example short rendering*

(e) *Proxy result*

Figure 5: *ESE, SDpp and other data for kitchen scene (N=4000, solution using no reference).*

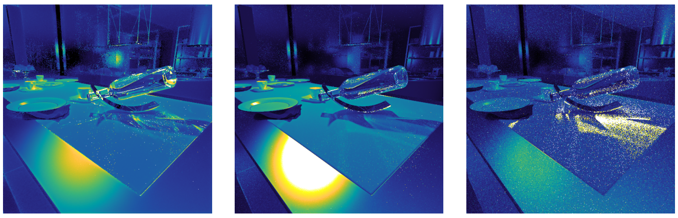


(a) *Ensemble power spectrum*

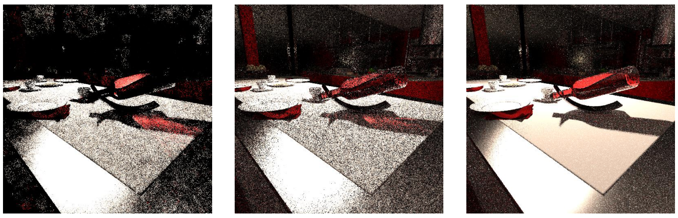


(b) *Reference*

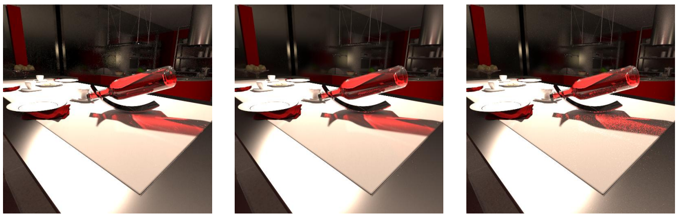
MLT *PSSMLT* *PT*



(c) *Standard deviation per pixel (SDpp)*

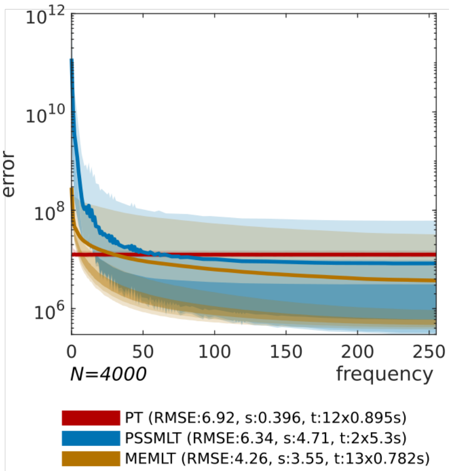


(d) *Example short rendering*



(e) *Proxy result*

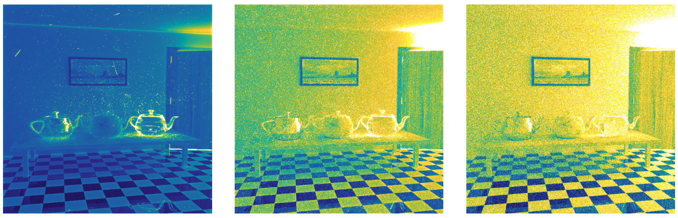
Figure 6: *ESE, SDpp and other data for bottle scene (N=4000, solution using no reference).*



(a) *Ensemble power spectrum*

(b) *Reference*

MLT *PSSMLT* *PT*

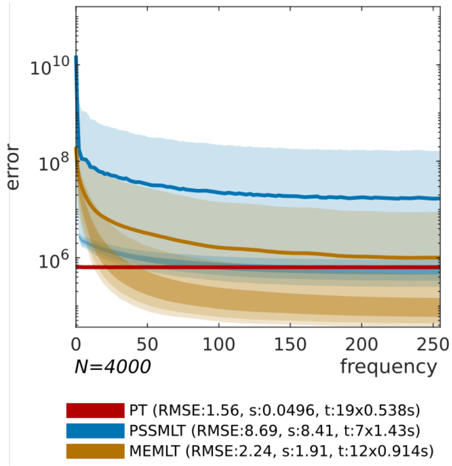


(c) *Standard deviation per pixel (SDpp)*

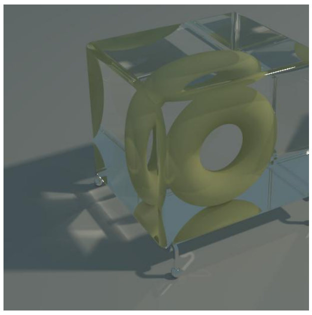
(d) *Example short rendering*

(e) *Proxy result*

Figure 7: *ESE, SDpp and other data for door scene (N=4000, solution using no reference).*

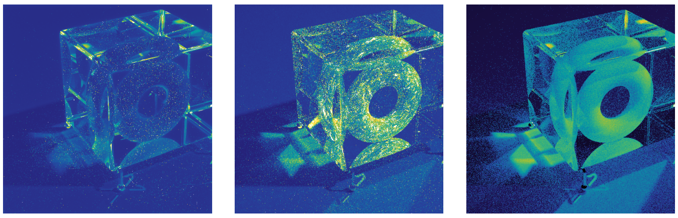


(a) *Ensemble power spectrum*

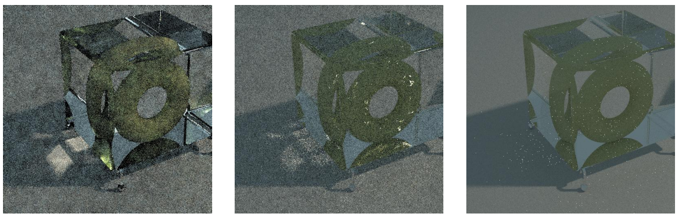


(b) *Reference*

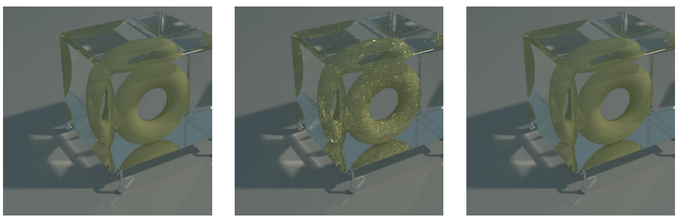
MLT *PSSMLT* *PT*



(c) *Standard deviation per pixel (SDpp)*



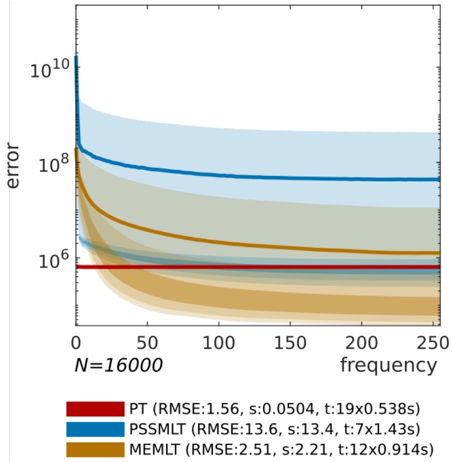
(d) *Example short rendering*



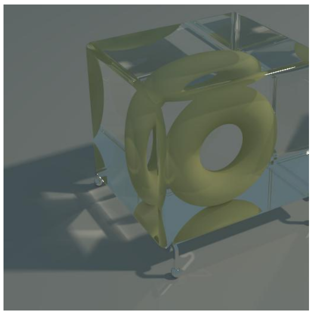
(e) *Proxy result*

Figure 8: *ESE, SDpp and other data for torus scene (N=4000, solution using no reference).*

2.2. Torus with N=16000

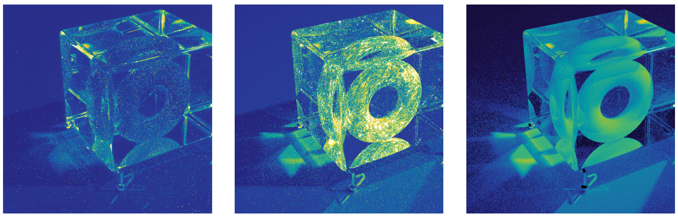


(a) *Ensemble power spectrum*

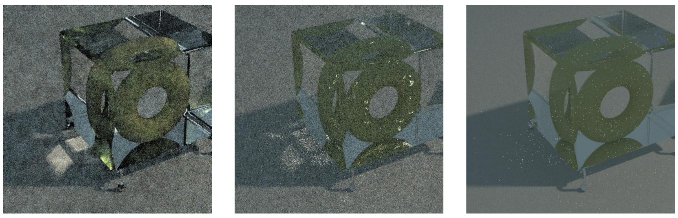


(b) *Reference*

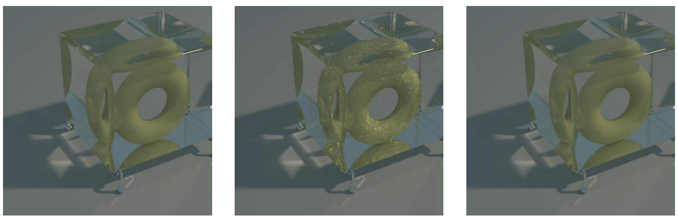
MLT *PSSMLT* *PT*



(c) *Standard deviation per pixel (SDpp)*



(d) *Example short rendering*

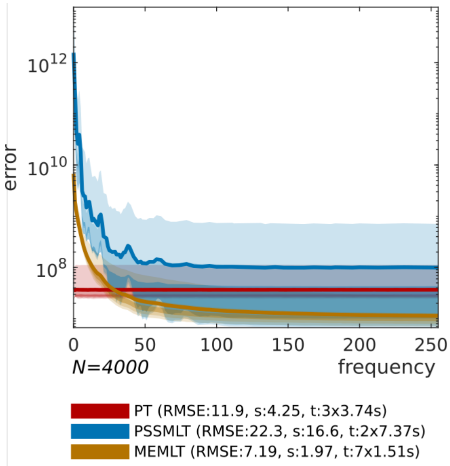


(e) *Proxy result*

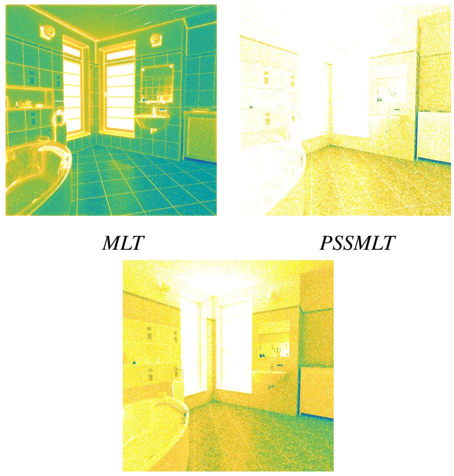
Figure 9: *ESE, SDpp and other data for torus scene (N=16000, solution using no reference).*

2.3. Comparison with ESE / SDpp using a reference

Compare with Section 2.1.

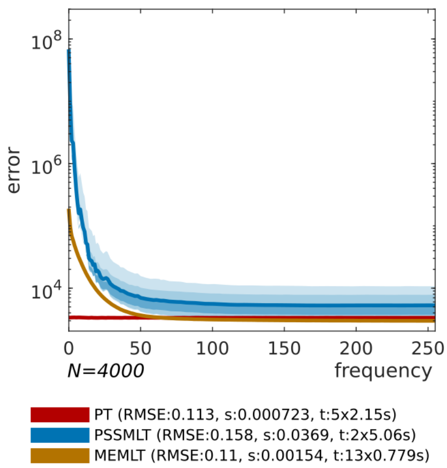


(a) *Ensemble power spectrum*

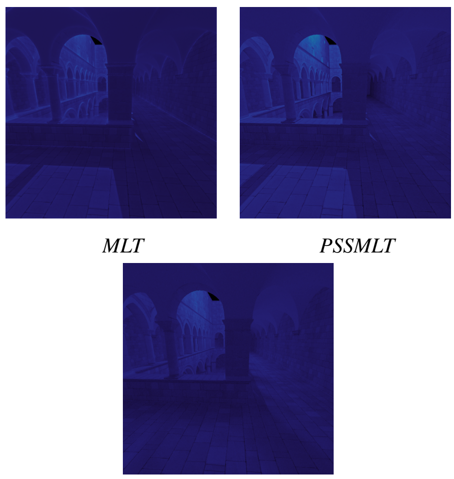


(b) *Standard deviation per pixel (SDpp)*

Figure 10: *ESE, SDpp and other data for bathroom scene (N=4000, solution using a reference).*

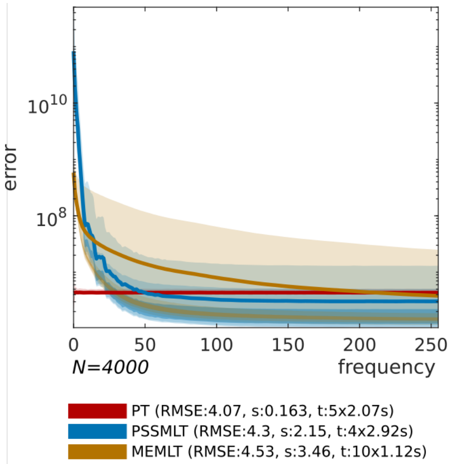


(a) *Ensemble power spectrum*

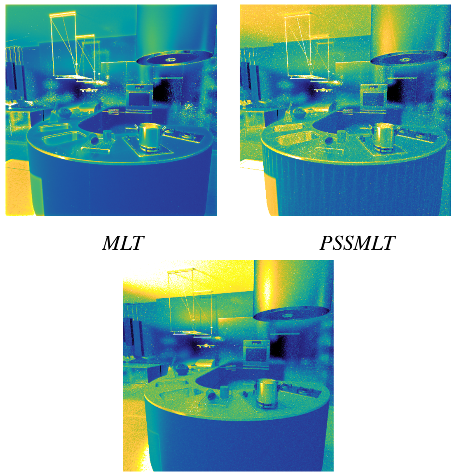


(b) *Standard deviation per pixel (SDpp)*

Figure 11: *ESE, SDpp and other data for sponza scene (N=4000, solution using a reference).*



(a) *Ensemble power spectrum*

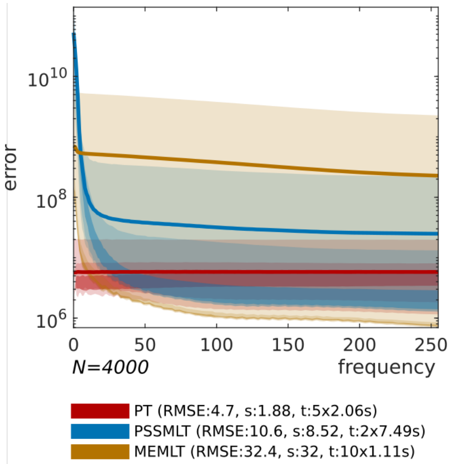


MLT *PSSMLT*

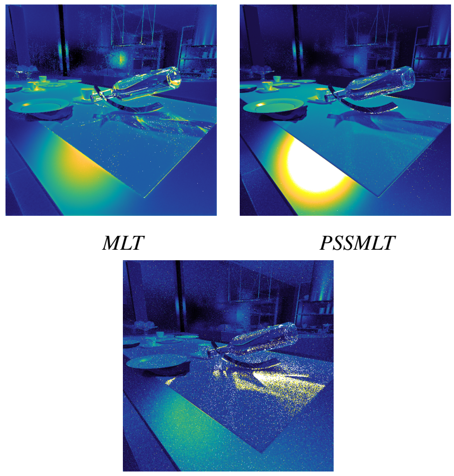
PT

(b) *Standard deviation per pixel (SDpp)*

Figure 12: *ESE, SDpp and other data for kitchen scene (N=4000, solution using a reference).*



(a) *Ensemble power spectrum*

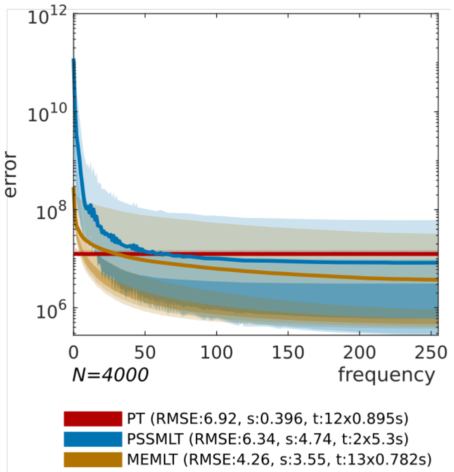


MLT *PSSMLT*

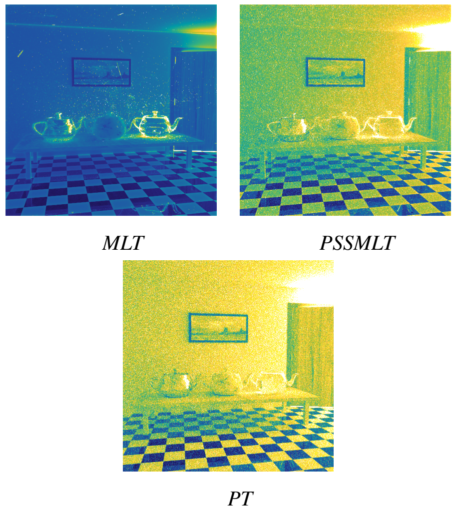
PT

(b) *Standard deviation per pixel (SDpp)*

Figure 13: *ESE, SDpp and other data for bottle scene (N=4000, solution using a reference).*

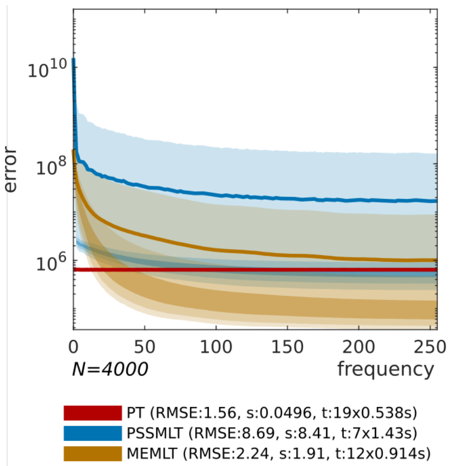


(a) *Ensemble power spectrum*

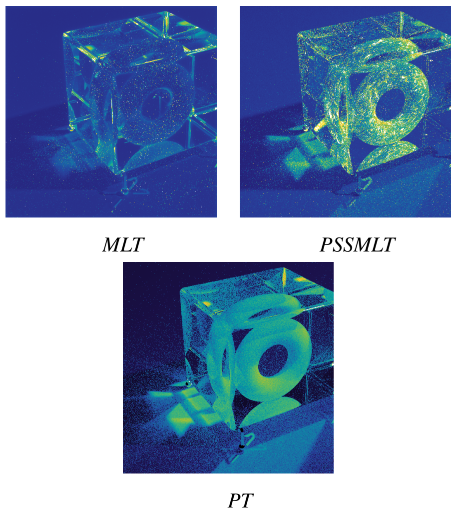


(b) *Standard deviation per pixel (SDpp)*

Figure 14: *ESE, SDpp and other data for door scene (N=4000, solution using a reference).*



(a) *Ensemble power spectrum*

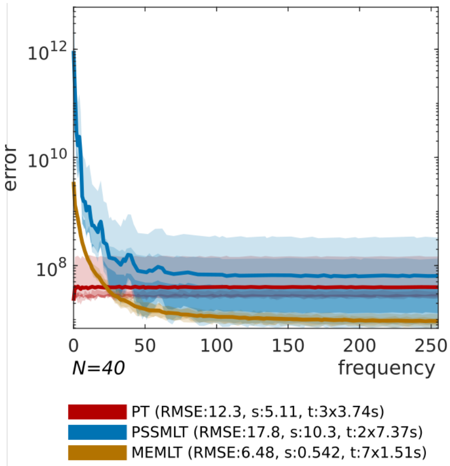


(b) *Standard deviation per pixel (SDpp)*

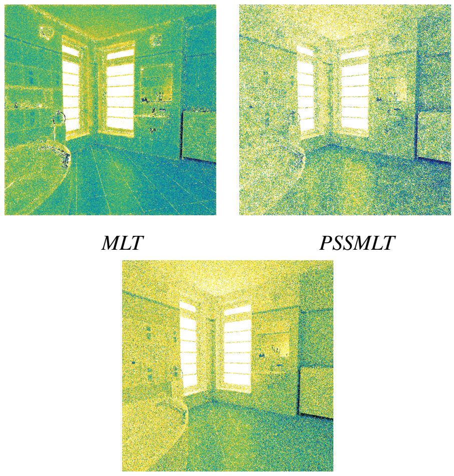
Figure 15: *ESE, SDpp and other data for torus scene (N=4000, solution using a reference).*

2.4. Influence of N, without a reference

See Section 2.1 for $N = 4000$.



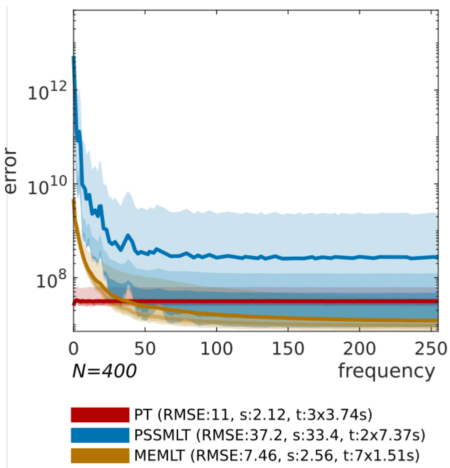
(a) *Ensemble power spectrum*



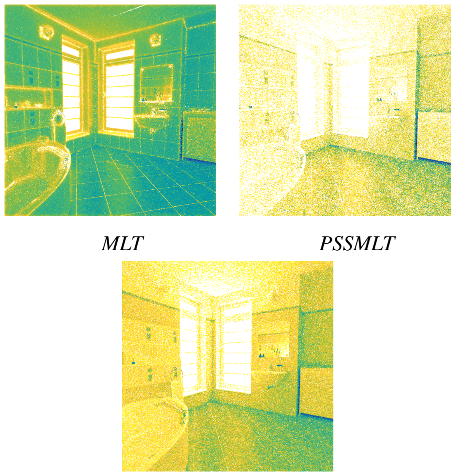
PT

(b) *Standard deviation per pixel (SDpp)*

Figure 16: *ESE, SDpp and other data for bathroom scene (N=40, solution using no reference).*



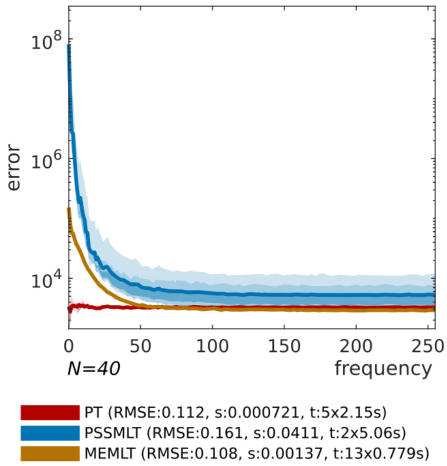
(a) *Ensemble power spectrum*



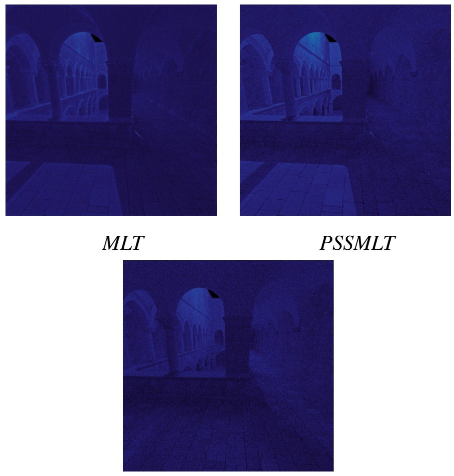
PT

(b) *Standard deviation per pixel (SDpp)*

Figure 17: *ESE, SDpp and other data for bathroom scene (N=400, solution using no reference).*

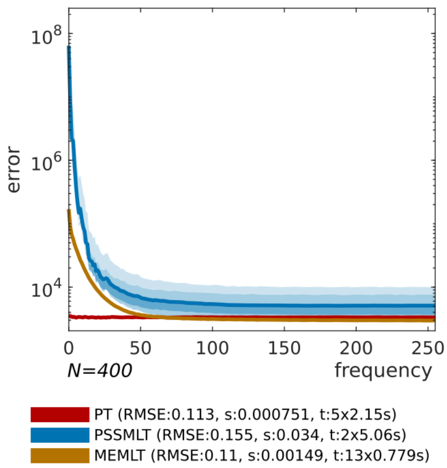


(a) *Ensemble power spectrum*

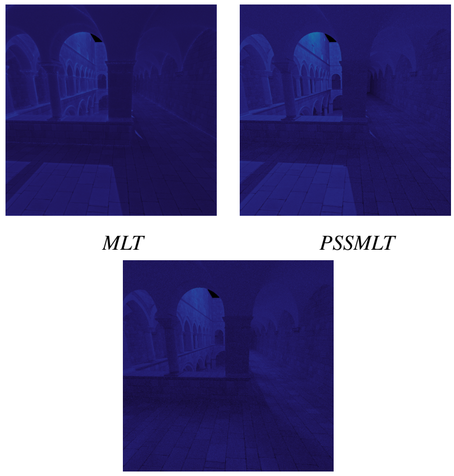


(b) *Standard deviation per pixel (SDpp)*

Figure 18: *ESE, SDpp and other data for sponza scene (N=40, solution using no reference).*

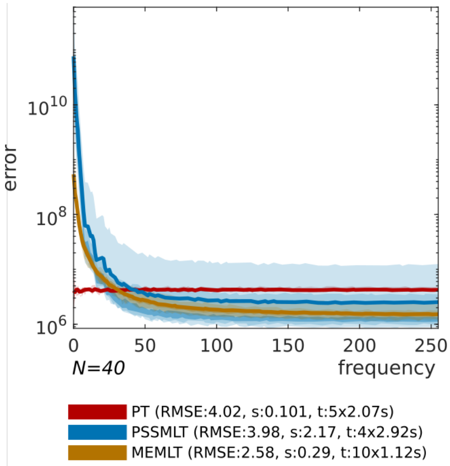


(a) *Ensemble power spectrum*

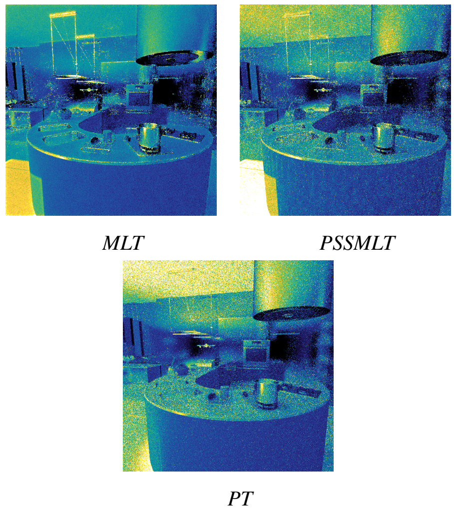


(b) *Standard deviation per pixel (SDpp)*

Figure 19: *ESE, SDpp and other data for sponza scene (N=400, solution using no reference).*

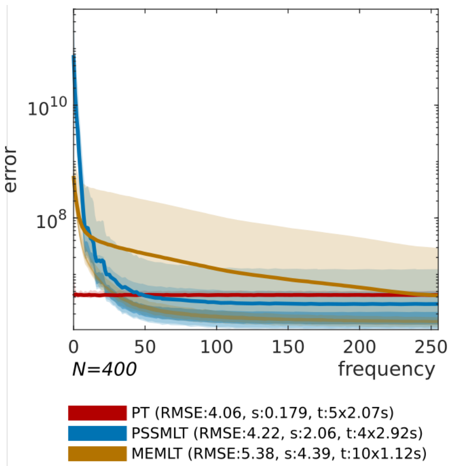


(a) *Ensemble power spectrum*

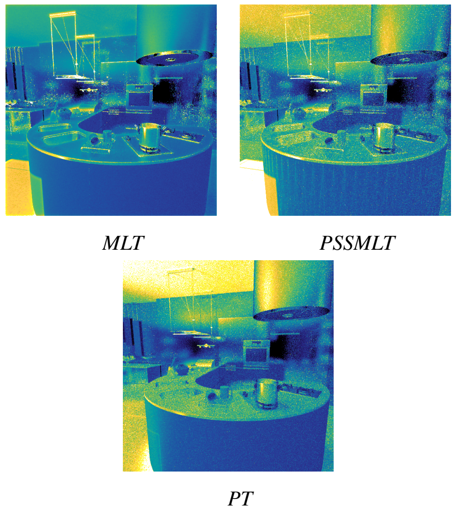


(b) *Standard deviation per pixel (SDpp)*

Figure 20: *ESE, SDpp and other data for kitchen scene (N=40, solution using no reference).*

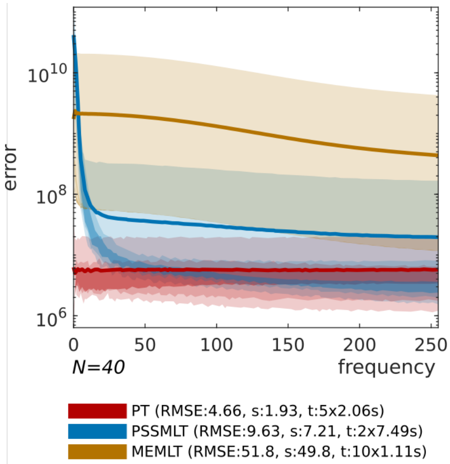


(a) *Ensemble power spectrum*

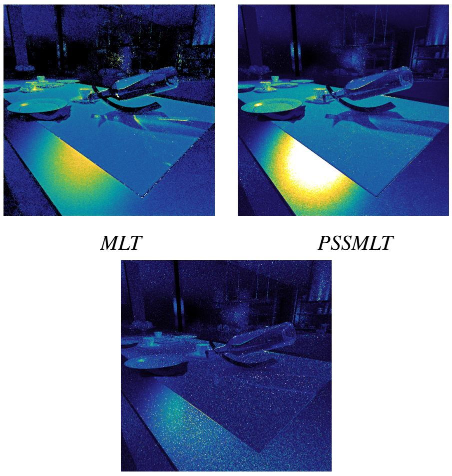


(b) *Standard deviation per pixel (SDpp)*

Figure 21: *ESE, SDpp and other data for kitchen scene (N=400, solution using no reference).*



(a) *Ensemble power spectrum*

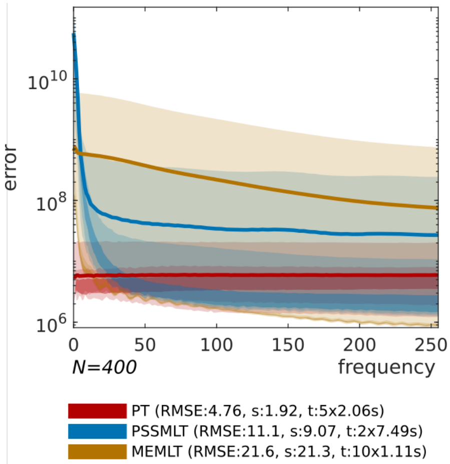


MLT *PSSMLT*

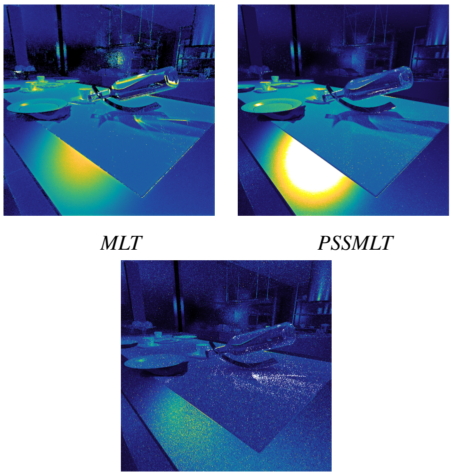
PT

(b) *Standard deviation per pixel (SDpp)*

Figure 22: *ESE, SDpp and other data for bottle scene (N=40, solution using no reference).*



(a) *Ensemble power spectrum*

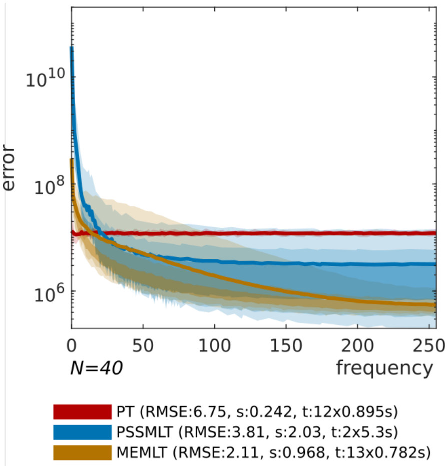


MLT *PSSMLT*

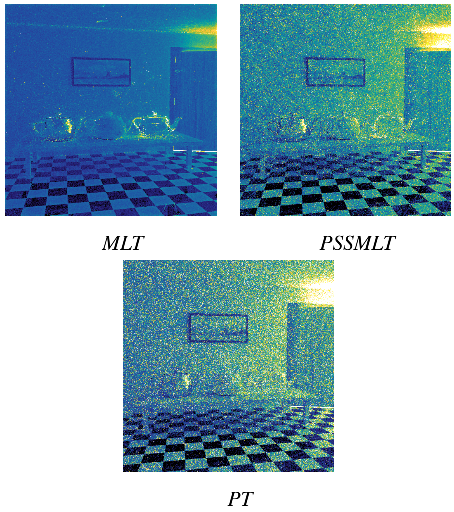
PT

(b) *Standard deviation per pixel (SDpp)*

Figure 23: *ESE, SDpp and other data for bottle scene (N=400, solution using no reference).*

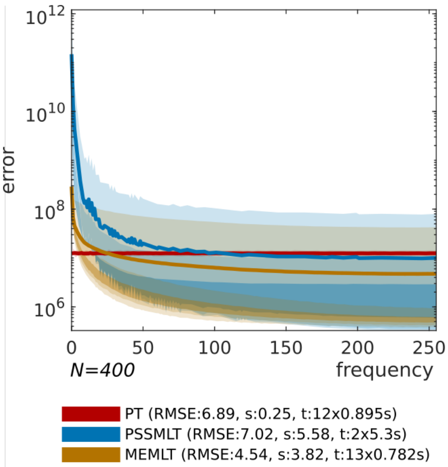


(a) *Ensemble power spectrum*

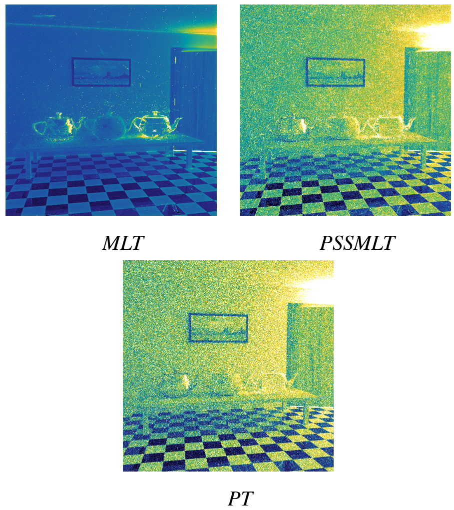


(b) *Standard deviation per pixel (SDpp)*

Figure 24: *ESE, SDpp and other data for door scene (N=40, solution using no reference).*

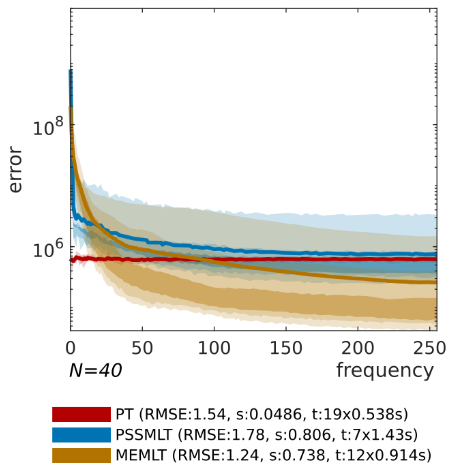


(a) *Ensemble power spectrum*

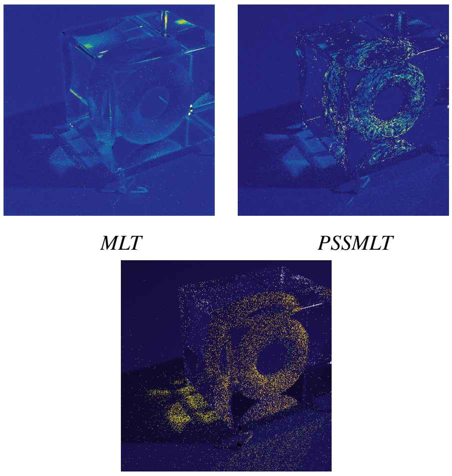


(b) *Standard deviation per pixel (SDpp)*

Figure 25: *ESE, SDpp and other data for door scene (N=400, solution using no reference).*

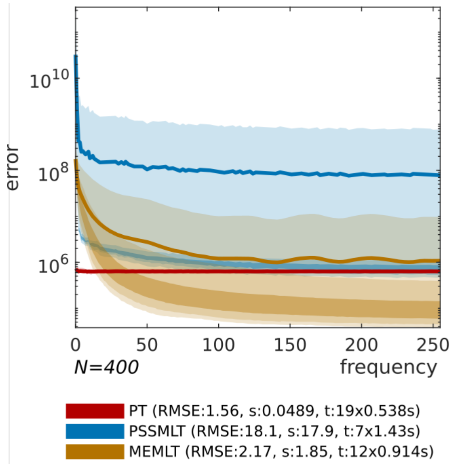


(a) *Ensemble power spectrum*

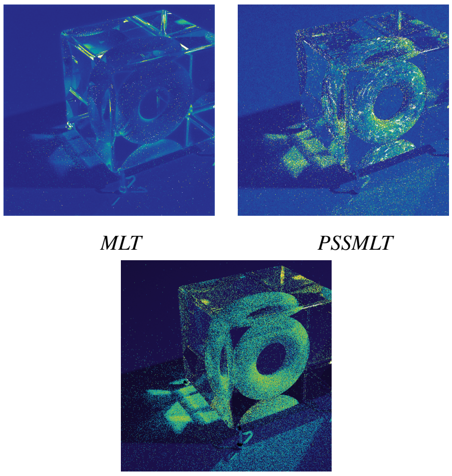


(b) *Standard deviation per pixel (SDpp)*

Figure 26: *ESE, SDpp and other data for torus scene (N=40, solution using no reference).*



(a) *Ensemble power spectrum*

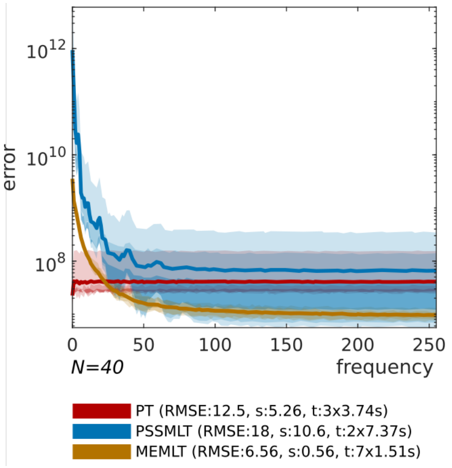


(b) *Standard deviation per pixel (SDpp)*

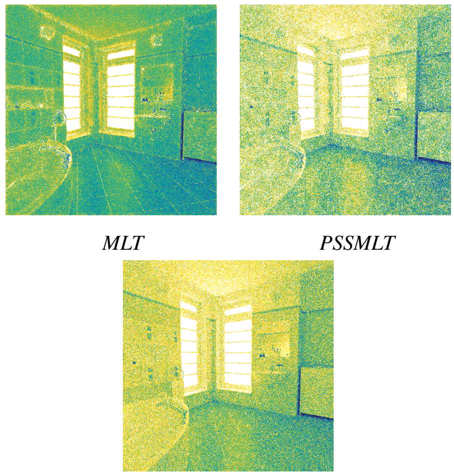
Figure 27: *ESE, SDpp and other data for torus scene (N=400, solution using no reference).*

2.5. Influence of N, while using a reference

See Section 2.3 for $N = 4000$.

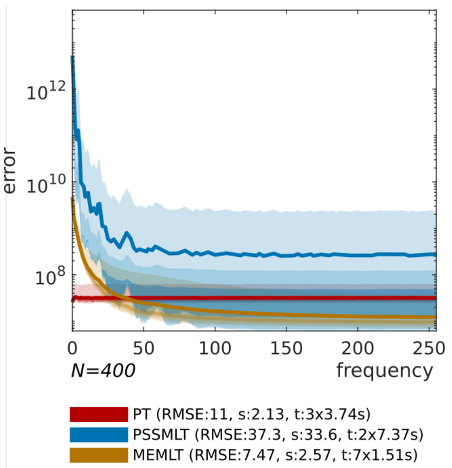


(a) *Ensemble power spectrum*

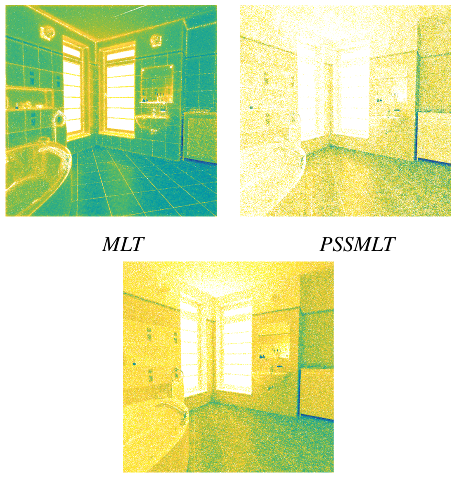


(b) *Standard deviation per pixel (SDpp)*

Figure 28: *ESE, SDpp and other data for bathroom scene (N=40, solution using a reference).*

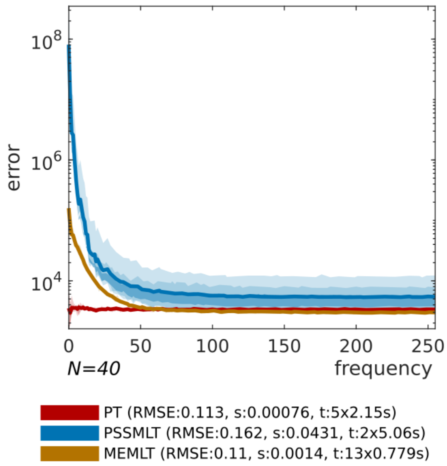


(a) *Ensemble power spectrum*

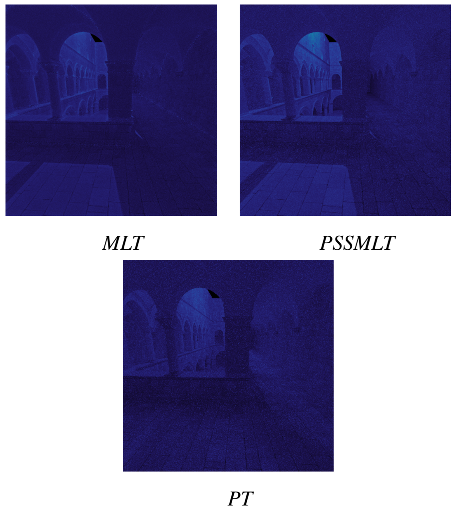


(b) *Standard deviation per pixel (SDpp)*

Figure 29: *ESE, SDpp and other data for bathroom scene (N=400, solution using a reference).*

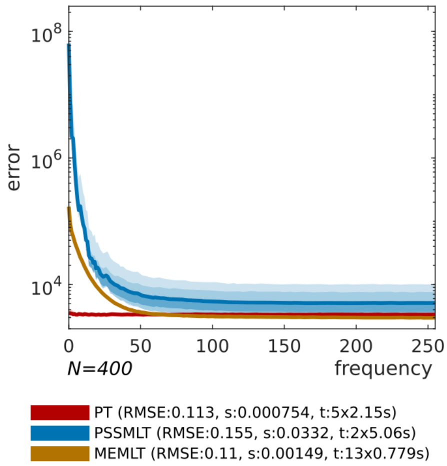


(a) *Ensemble power spectrum*

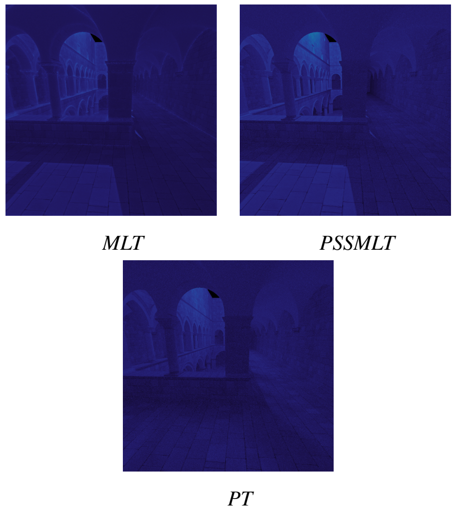


(b) *Standard deviation per pixel (SDpp)*

Figure 30: *ESE, SDpp and other data for sponza scene (N=40, solution using a reference).*

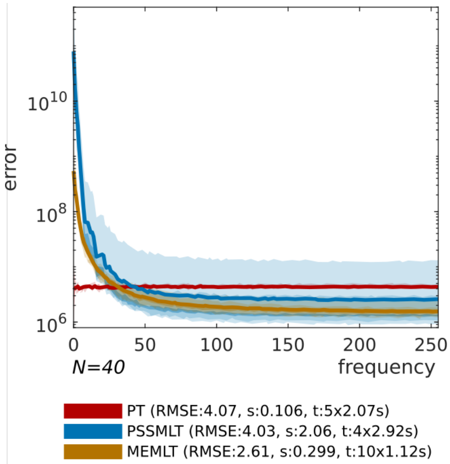


(a) *Ensemble power spectrum*

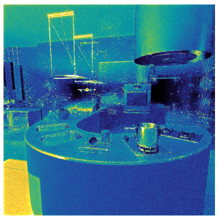


(b) *Standard deviation per pixel (SDpp)*

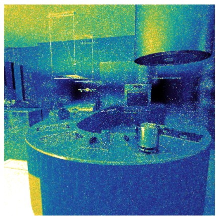
Figure 31: *ESE, SDpp and other data for sponza scene (N=400, solution using a reference).*



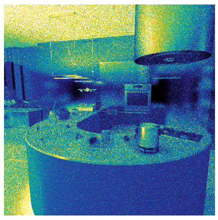
(a) *Ensemble power spectrum*



MLT



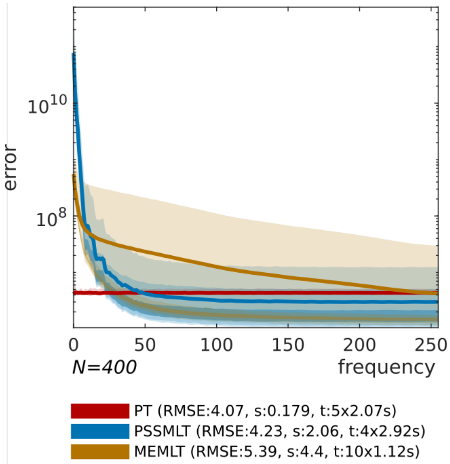
PSSMLT



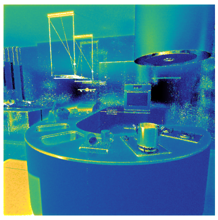
PT

(b) *Standard deviation per pixel (SDpp)*

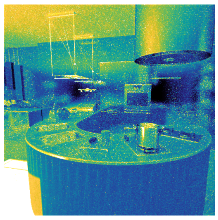
Figure 32: *ESE, SDpp and other data for kitchen scene (N=40, solution using a reference).*



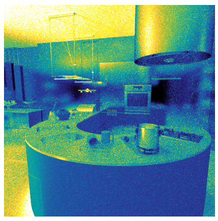
(a) *Ensemble power spectrum*



MLT



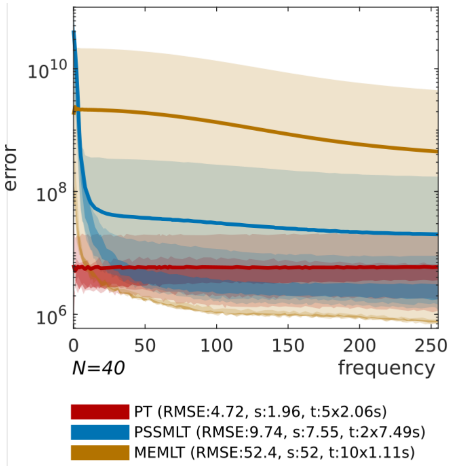
PSSMLT



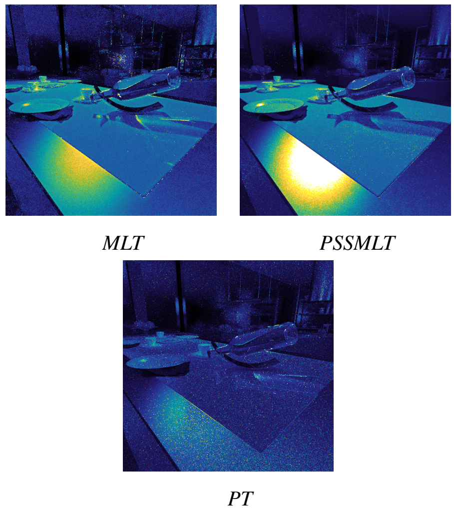
PT

(b) *Standard deviation per pixel (SDpp)*

Figure 33: *ESE, SDpp and other data for kitchen scene (N=400, solution using a reference).*

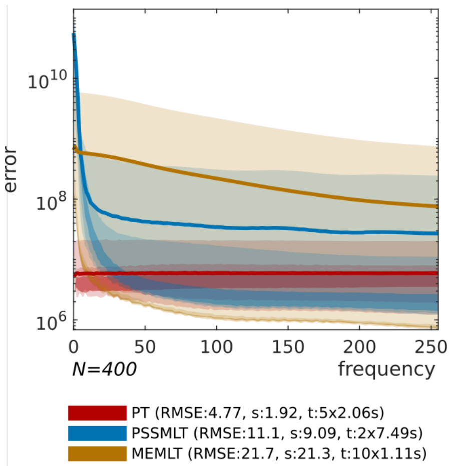


(a) *Ensemble power spectrum*

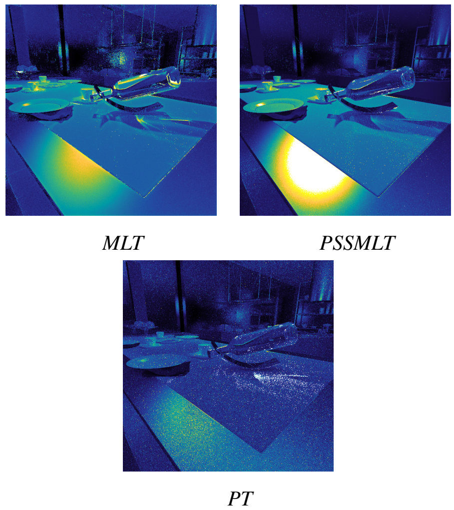


(b) *Standard deviation per pixel (SDpp)*

Figure 34: *ESE, SDpp and other data for bottle scene (N=40, solution using a reference).*

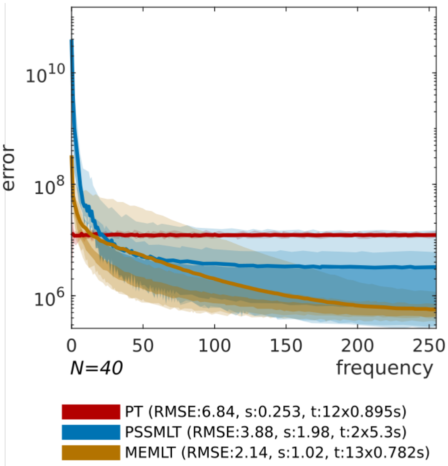


(a) *Ensemble power spectrum*

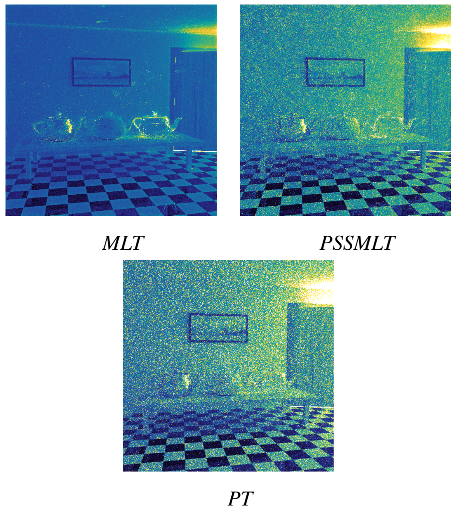


(b) *Standard deviation per pixel (SDpp)*

Figure 35: *ESE, SDpp and other data for bottle scene (N=400, solution using a reference).*

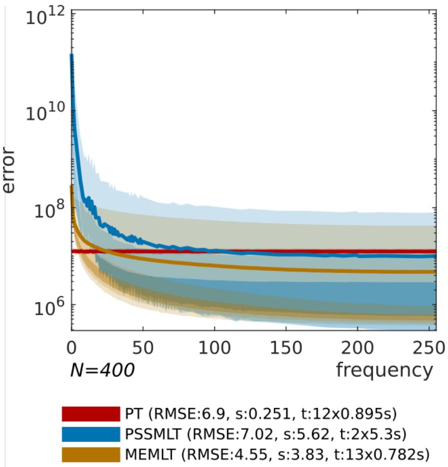


(a) *Ensemble power spectrum*

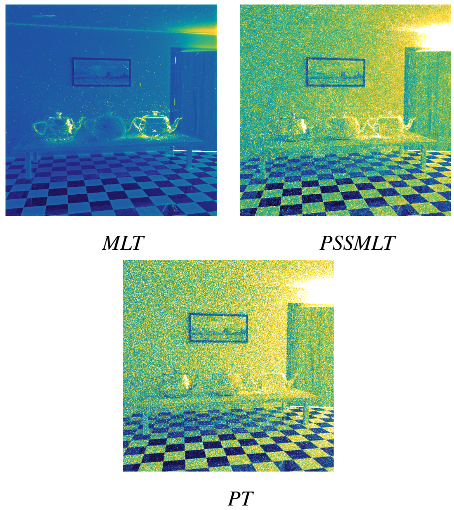


(b) *Standard deviation per pixel (SDpp)*

Figure 36: *ESE, SDpp and other data for door scene (N=40, solution using a reference).*

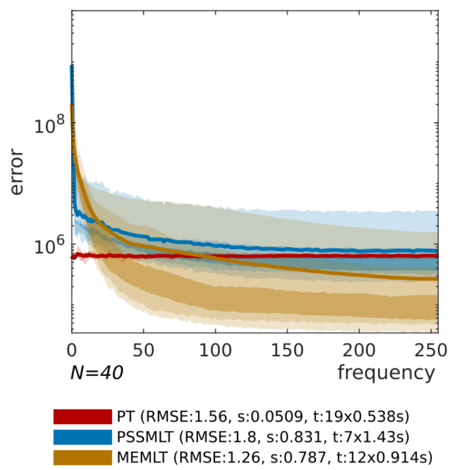


(a) *Ensemble power spectrum*

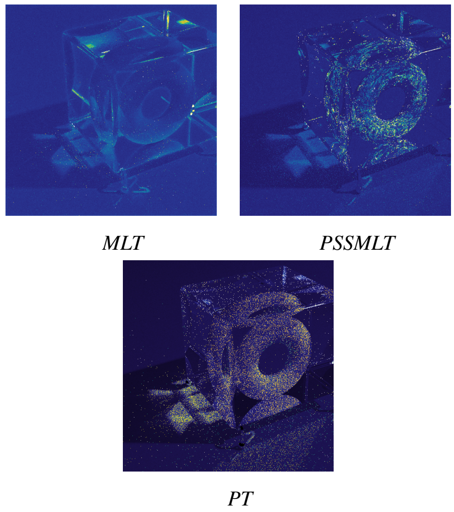


(b) *Standard deviation per pixel (SDpp)*

Figure 37: *ESE, SDpp and other data for door scene (N=400, solution using a reference).*

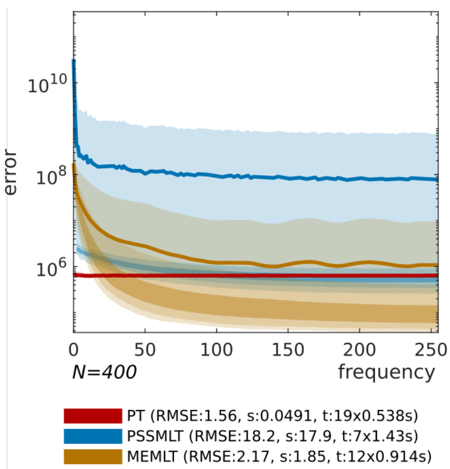


(a) *Ensemble power spectrum*

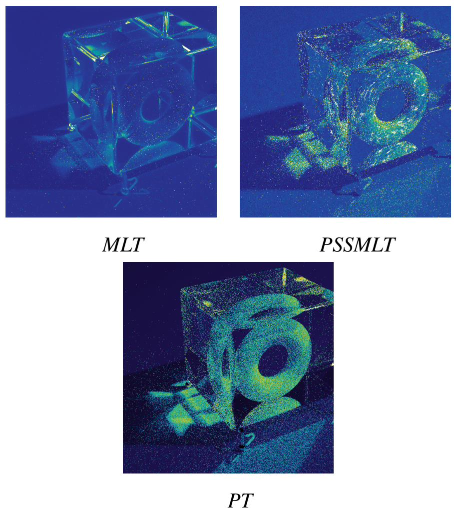


(b) *Standard deviation per pixel (SDpp)*

Figure 38: *ESE, SDpp and other data for torus scene (N=40, solution using a reference).*



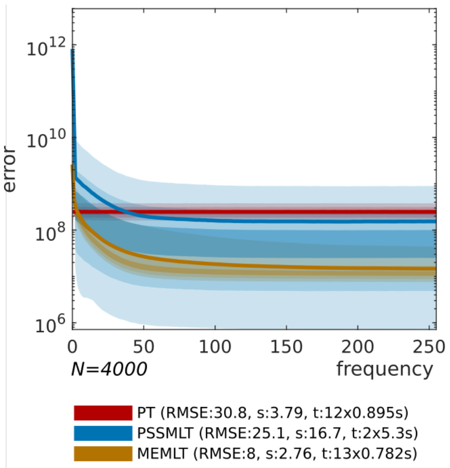
(a) *Ensemble power spectrum*



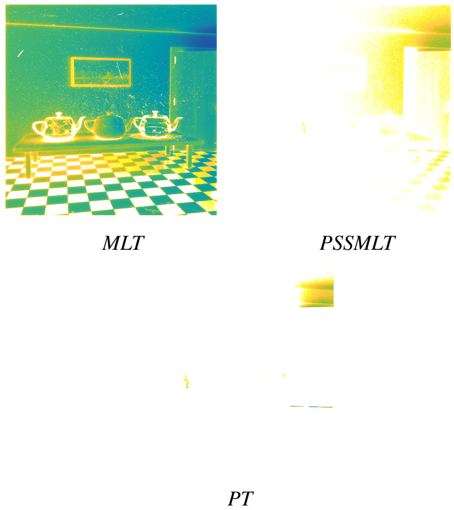
(b) *Standard deviation per pixel (SDpp)*

Figure 39: *ESE, SDpp and other data for torus scene (N=400, solution using a reference).*

2.6. Relative Error

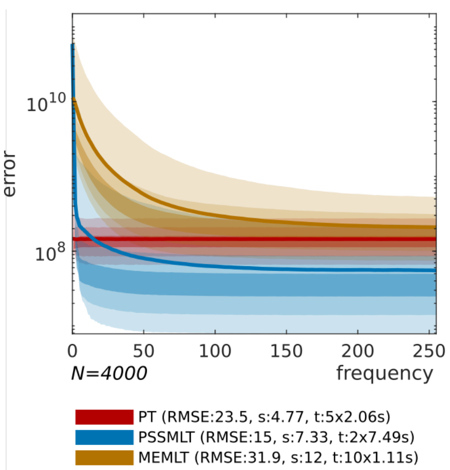


(a) *Ensemble power spectrum*

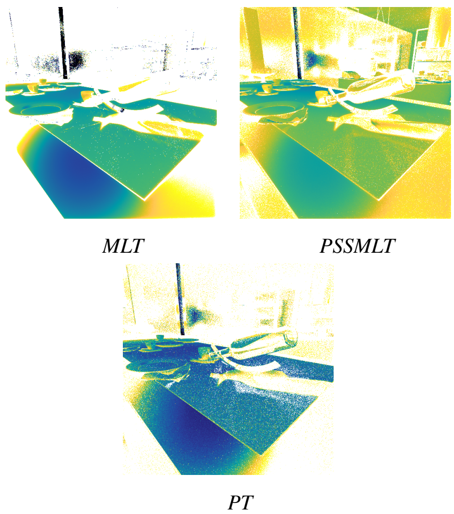


(b) *Standard deviation per pixel (SDpp)*

Figure 40: *ESE, SDpp and other data for door scene (N=4000, relative error, solution using no reference).*

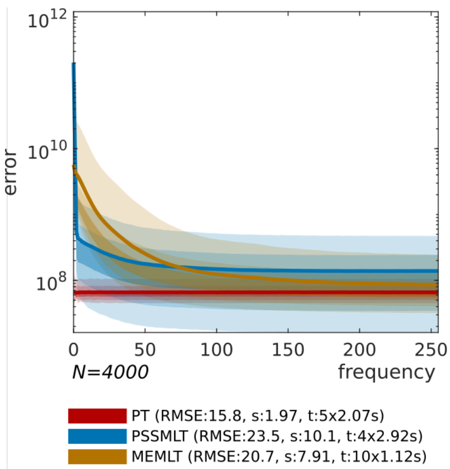


(a) *Ensemble power spectrum*

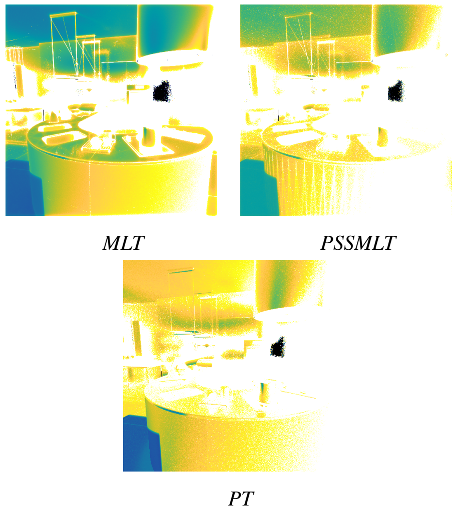


(b) *Standard deviation per pixel (SDpp)*

Figure 41: *ESE, SDpp and other data for bottle scene (N=4000, relative error, solution using no reference).*

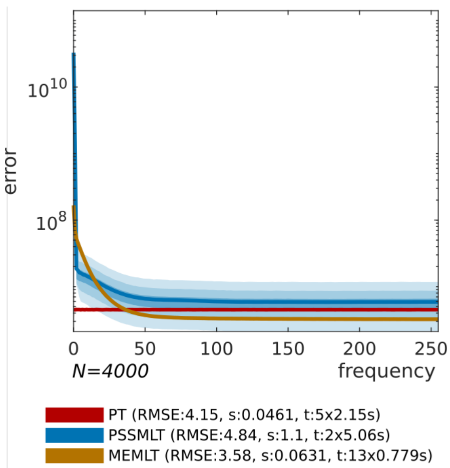


(a) *Ensemble power spectrum*

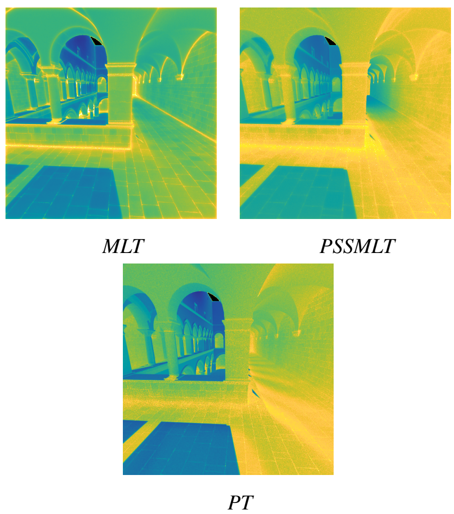


(b) *Standard deviation per pixel (SDpp)*

Figure 42: *ESE, SDpp and other data for kitchen scene (N=4000, relative error, solution using no reference).*

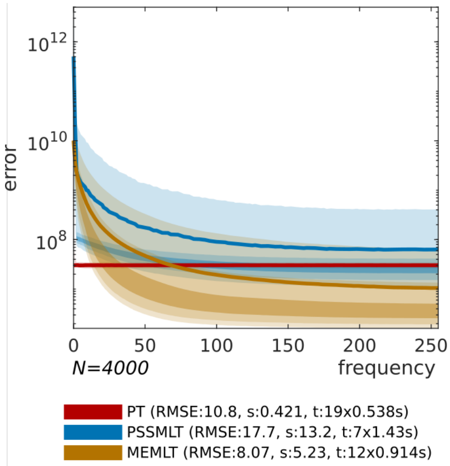


(a) *Ensemble power spectrum*

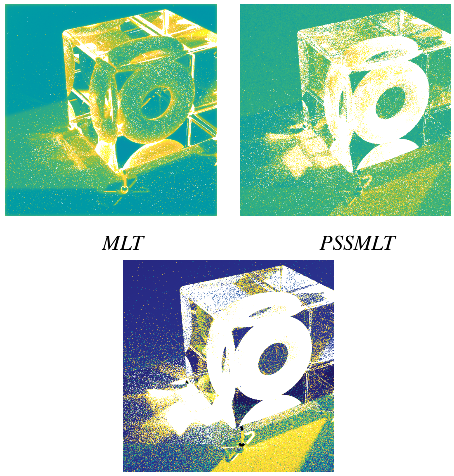


(b) *Standard deviation per pixel (SDpp)*

Figure 43: *ESE, SDpp and other data for sponza scene (N=4000, relative error, solution using no reference).*

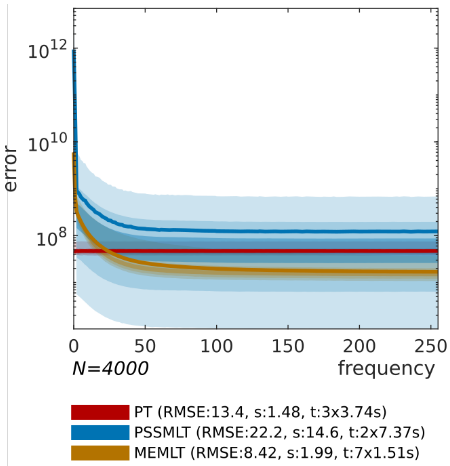


(a) *Ensemble power spectrum*

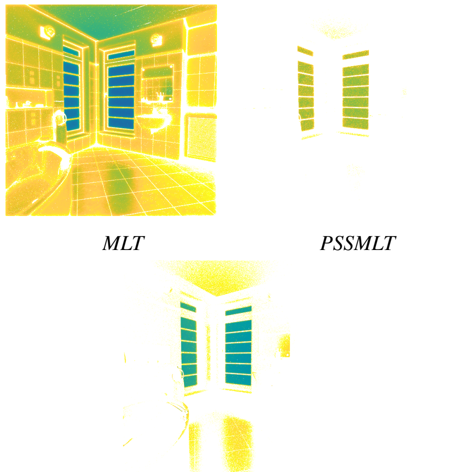


(b) *Standard deviation per pixel (SDpp)*

Figure 44: *ESE, SDpp and other data for torus scene (N=4000, relative error, solution using no reference).*



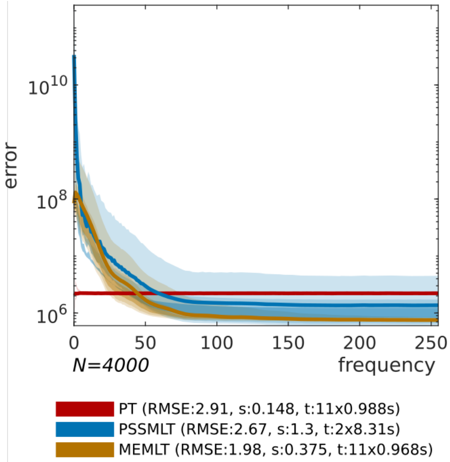
(a) *Ensemble power spectrum*



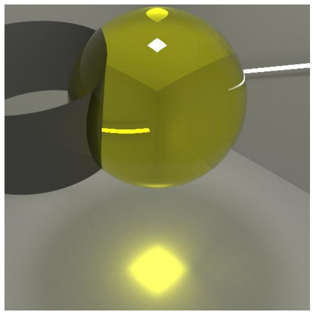
(b) *Standard deviation per pixel (SDpp)*

Figure 45: *ESE, SDpp and other data for bathroom scene (N=4000, relative error, solution using no reference).*

3. Parameterized Box Scene

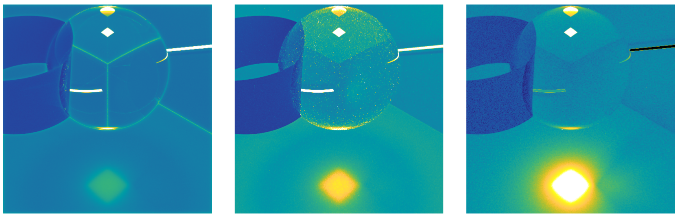


(a) *Ensemble power spectrum*

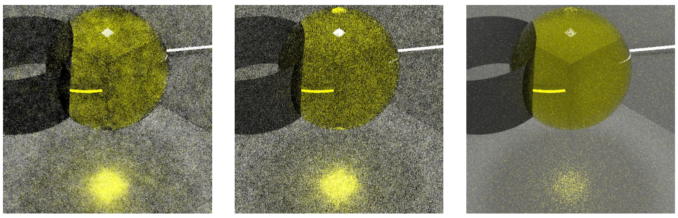


(b) *Reference*

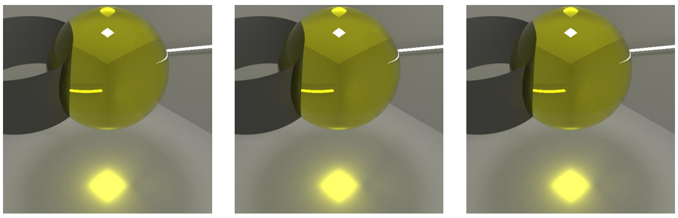
MLT *PSSMLT* *PT*



(c) *Standard deviation per pixel (SDpp)*

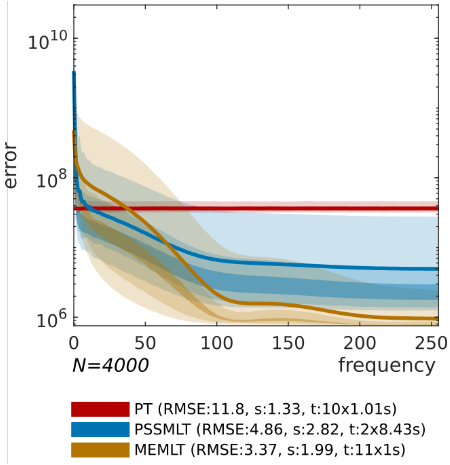


(d) *Example short rendering*

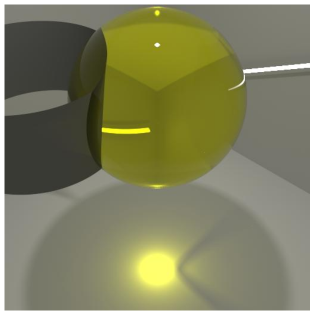


(e) *Proxy result*

Figure 46: *ESE, SDpp and other data for boxRoughn1O2048Light100 scene (N=4000, solution using no reference).*

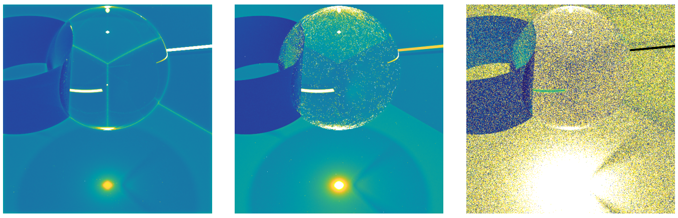


(a) *Ensemble power spectrum*

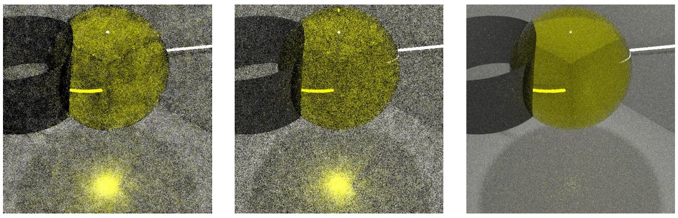


(b) *Reference*

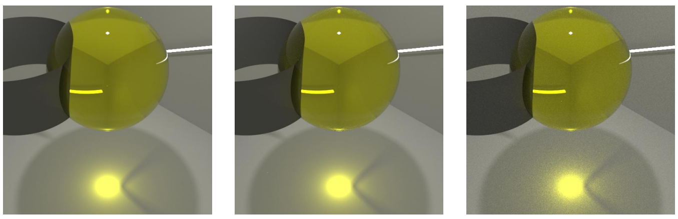
MLT *PSSMLT* *PT*



(c) *Standard deviation per pixel (SDpp)*

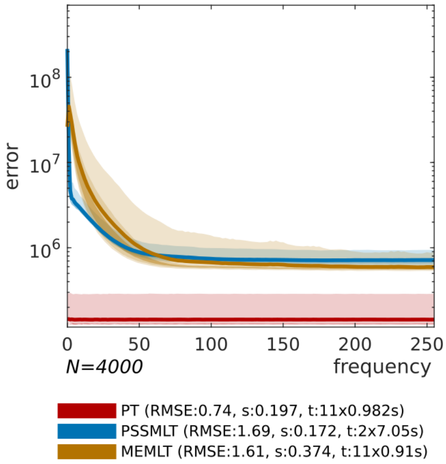


(d) *Example short rendering*

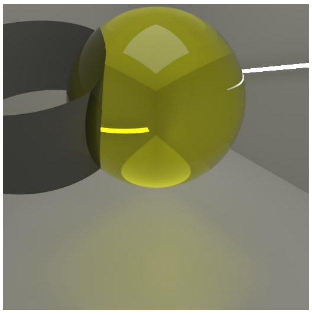


(e) *Proxy result*

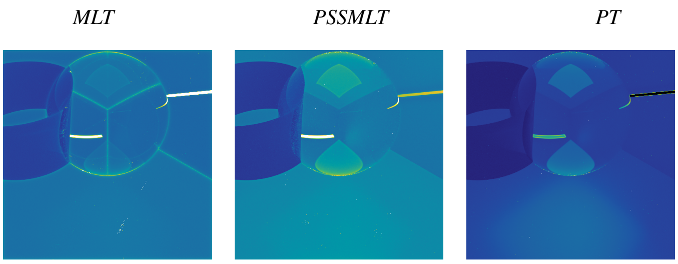
Figure 47: *ESE, SDpp and other data for boxRoughn1O256Light25 scene (N=4000, solution using no reference).*



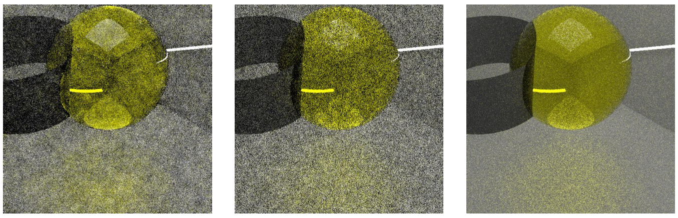
(a) *Ensemble power spectrum*



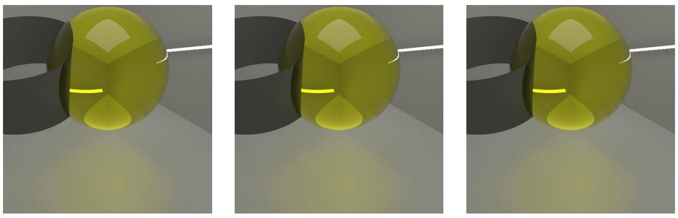
(b) *Reference*



(c) *Standard deviation per pixel (SDpp)*

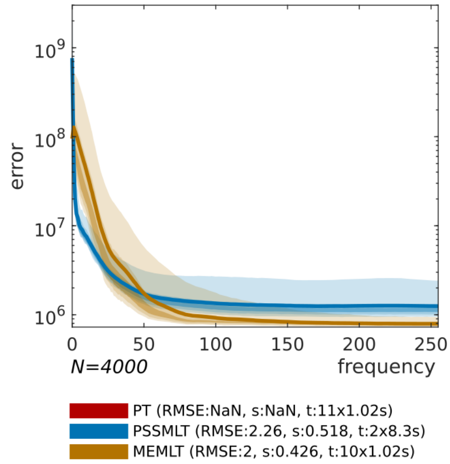


(d) *Example short rendering*

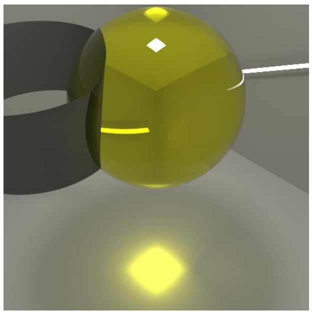


(e) *Proxy result*

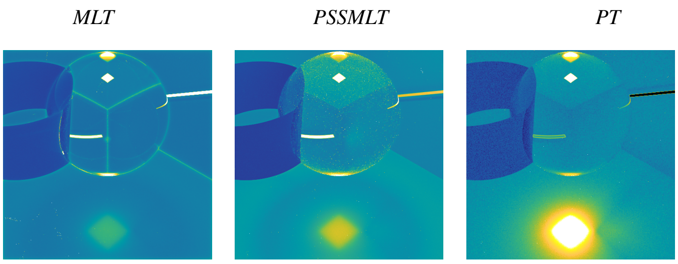
Figure 48: *ESE, SDpp and other data for boxRoughn1O256Light400 scene (N=4000, solution using no reference).*



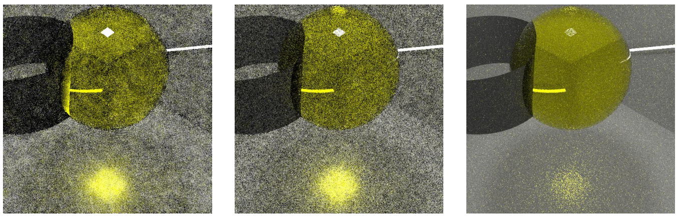
(a) *Ensemble power spectrum*



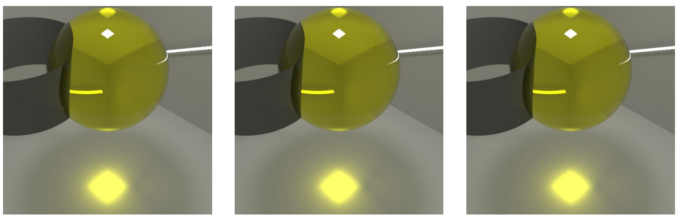
(b) *Reference*



(c) *Standard deviation per pixel (SDpp)*

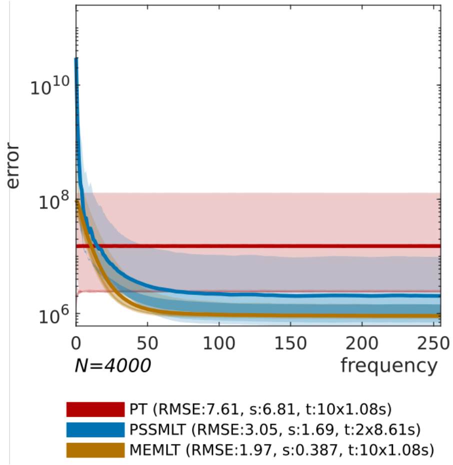


(d) *Example short rendering*

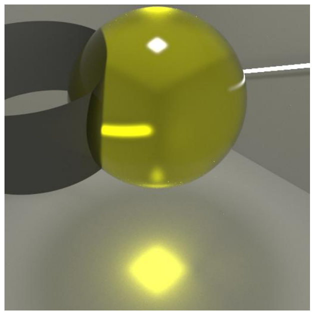


(e) *Proxy result*

Figure 49: *ESE, SDpp and other data for boxRoughn1O256Light100 scene (N=4000, solution using no reference).*

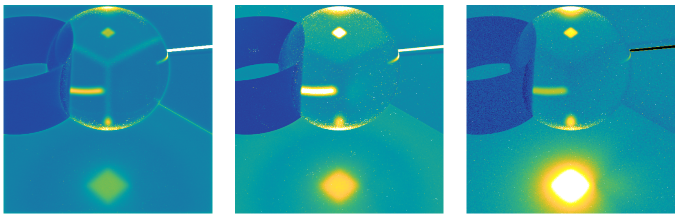


(a) *Ensemble power spectrum*

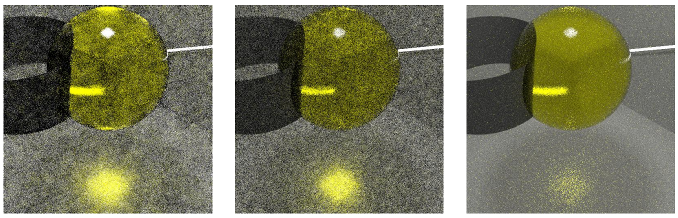


(b) *Reference*

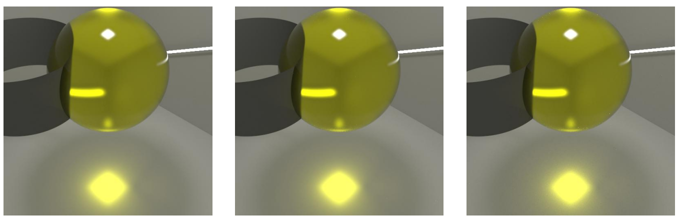
MLT *PSSMLT* *PT*



(c) *Standard deviation per pixel (SDpp)*



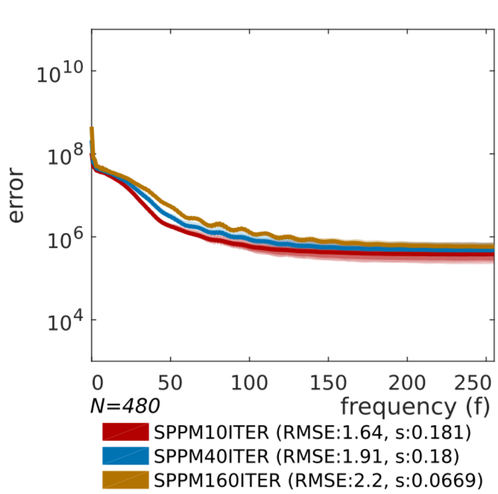
(d) *Example short rendering*



(e) *Proxy result*

Figure 50: *ESE, SDpp and other data for boxRoughn1O32Light100 scene (N=4000, solution using no reference).*

4. Adaptive Sampling, Consistent and Sub-$\Theta(1/\sqrt{N})$ Convergence Algorithms

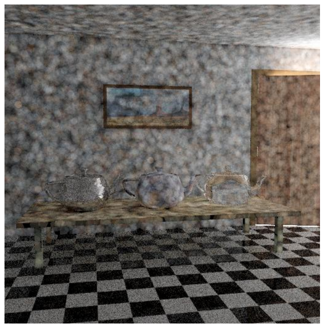


(a) *Error Spectrum Ensemble*

(b) *SDpp 10 iter.*

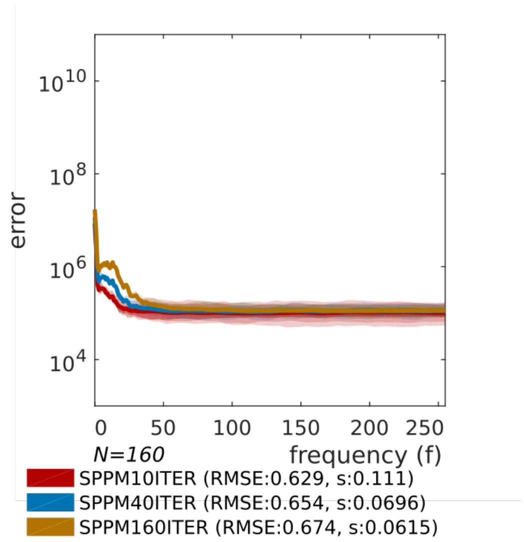
(c) *SDpp 160 iter.*

(d) *Example 10 iter.*

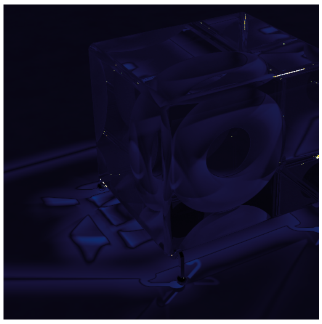


(e) *Example 160 iter.*

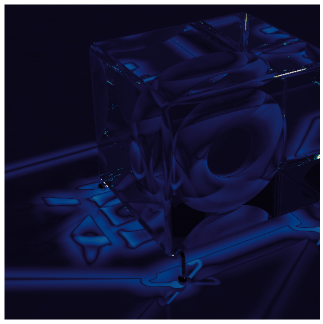
Figure 51: *Error descriptors for SPPM on door scene. Error scaled with $\sqrt{iter}$. ESE shows that convergence is slower than $\Theta(1/iter)$ for mid and high frequencies.*



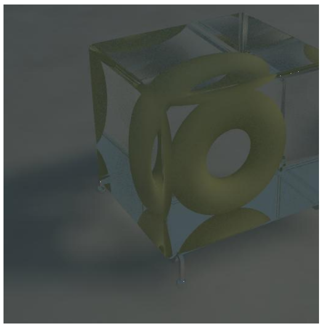
(a) *Error Spectrum Ensemble*



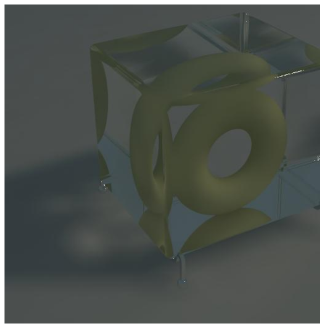
(b) *SDpp 10 iter.*



(c) *SDpp 160 iter.*

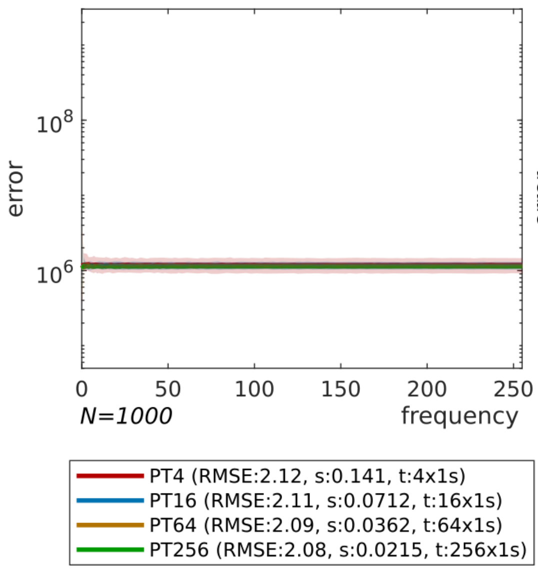


(d) *Example 10 iter.*

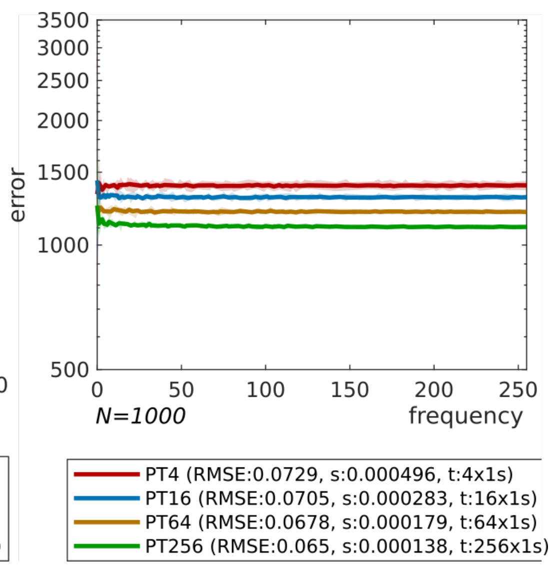


(e) *Example 160 iter.*

Figure 52: *Error descriptors for SPPM on torus scene. Error scaled with $\sqrt{iter}$. ESE shows that convergence is slower than $\Theta(1/iter)$ for low frequencies.*



(a) *Torus rendered with path tracing / QMC*



(b) *Sponza rendered with path tracing / QMC*

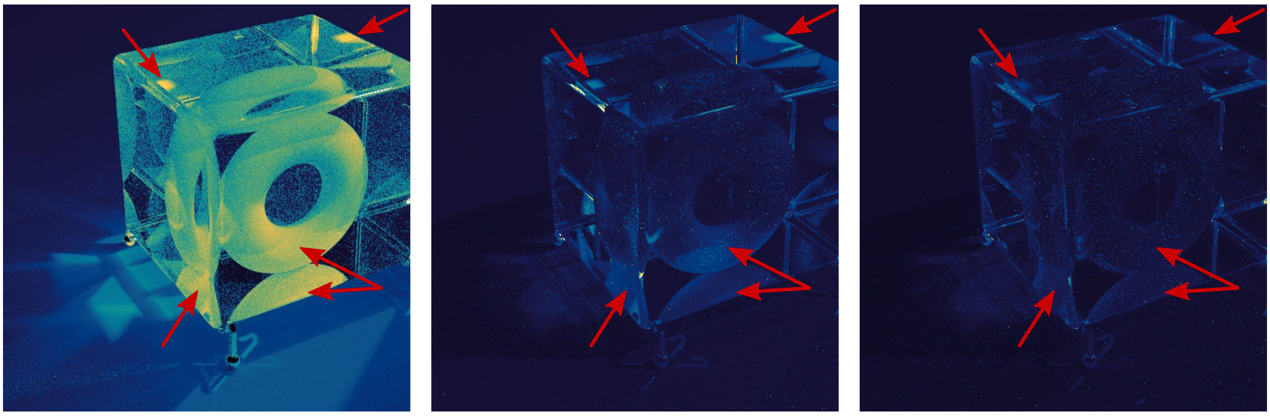
5. MLT configurations

The glass roughness parameter of the box scene is set to $1/256$ and light size to 50.

5.1. Seed-pool size (number of luminance samples)

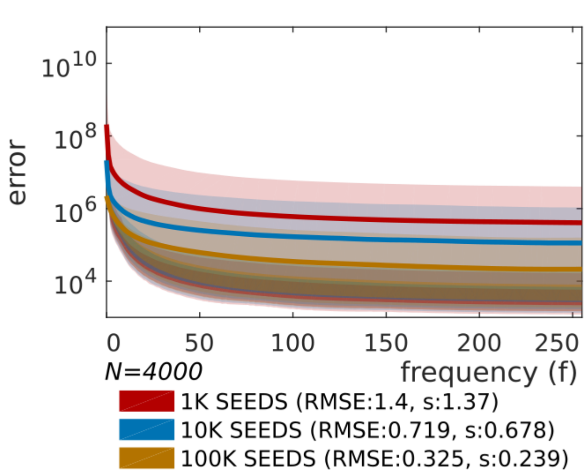
This parameter has two implications. The first one is the accuracy of the brightness scaling factor W. It is possible to factor out the influence in short renders by rescaling with the luminance of the reference (labelled W correction). The resulting algorithm becomes biased. When comparing the two variants in Figure 55, it is visible that with few luminance samples, the variance of W significantly increases the error.

The second implication is the distribution of starting paths sampled from the pool. With a small pool the probability of starting several chains from the same seed path is relatively high, especially when the pool contains MC outliers. Additionally, with a larger seed pool the starting points are distributed closer to the equilibrium distribution. The effect is visible in Figures 55(b and d): error in low frequencies is strictly decreasing with rising pool sizes. In the standard deviation image (Figure 55e) it is visible that with more seeds there is less error in the two hard-to-sample light effects. Figure 54 shows that light effects that are problematic for the seeding method (a) also show more error in the MLT image if the seed pool is too small (compare b and c).

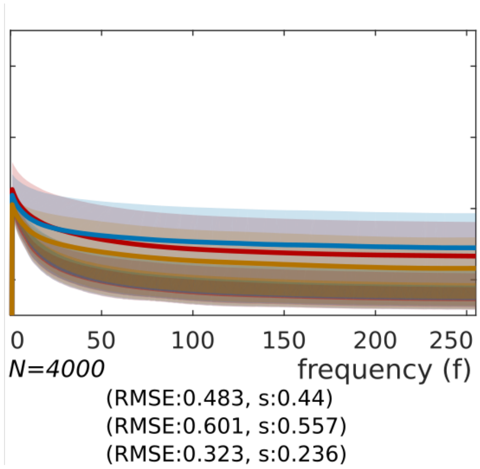


(a) *BDPT* **(b)** *1k seeds* **(c)** *100k seeds*

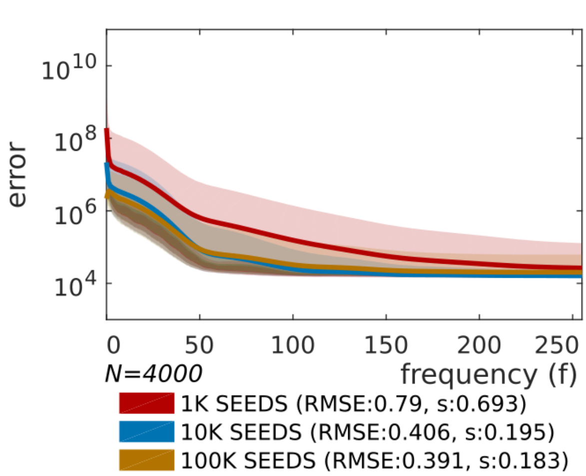
Figure 54: *Torus standard deviation: Light effects that are hard to sample for the algorithm populating the seeding pool (BDPT) are also problematic if the pool size is small (indicated by the arrows).*



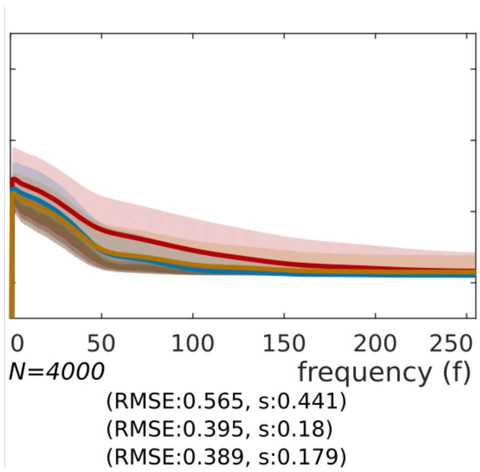
(a) *Torus*



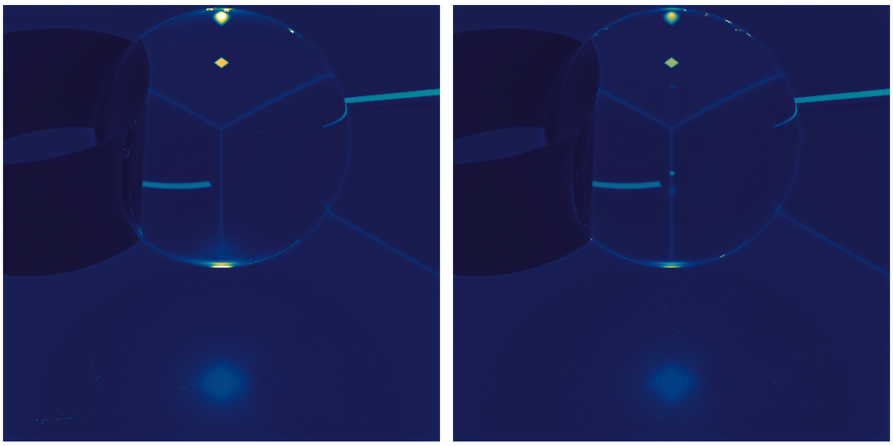
(b) *Torus (W correction)*



(c) *Box*

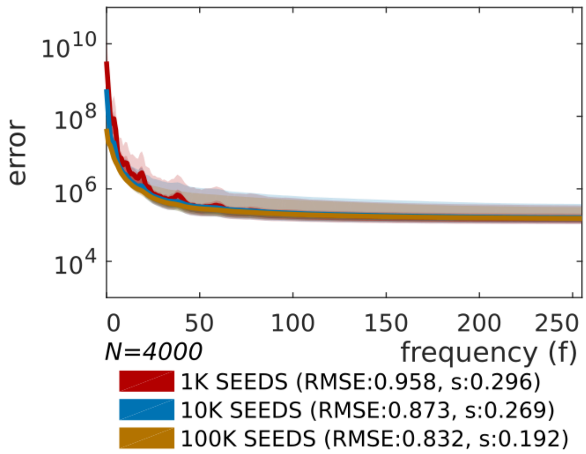


(d) *Box (W correction)*

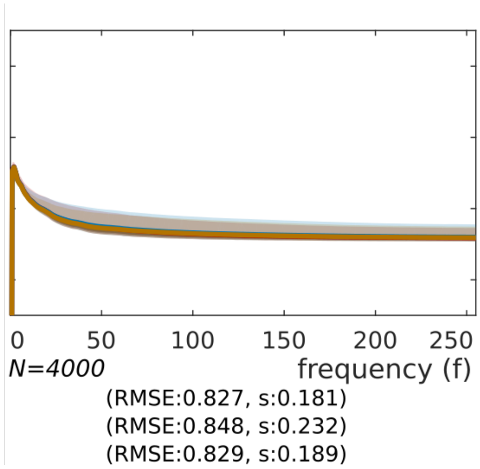


(e) *Box std. dev. detail (1k and 100k)*

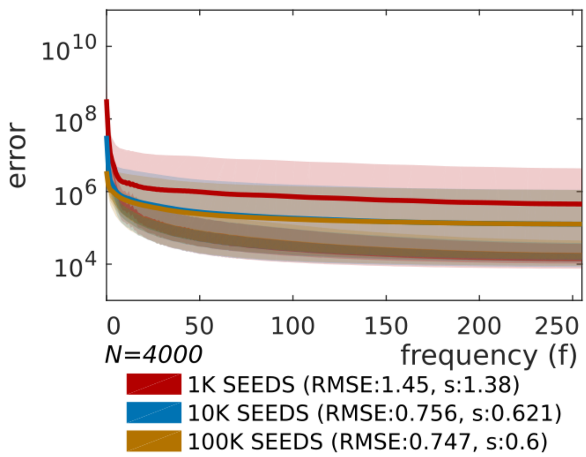
Figure 55: *Varying seed pool-sizes (number of luminance samples): Reducing the size of the seed pool increases variance of the chain weight W, which in turn increases overall error (a and c). But this is not the only effect as can be seen in (b) and (d), where W was replaced by the reference's W: An increased pool size benefits low frequencies (while high frequencies are inconclusive). (e) shows two bright and hard-to-sample light effects: If the seed pool is small, chances are no sample will cover those effects and W is relatively small, reducing the contribution of the whole chain. Once luminance samples do cover the light effects, W and therefore the contribution will be larger. Additionally the probability of starting several chains from the same path is high. This means that in case of small seed pools, light effects that are hard to sample for the seeding method, cause outliers. More scenes in Figure 56.*



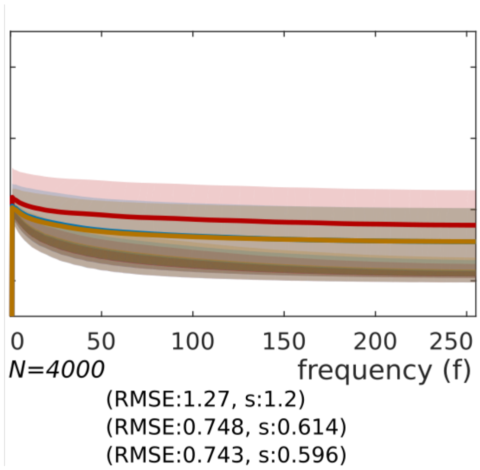
(a) *Bathroom*



(b) *with equalised brightness*



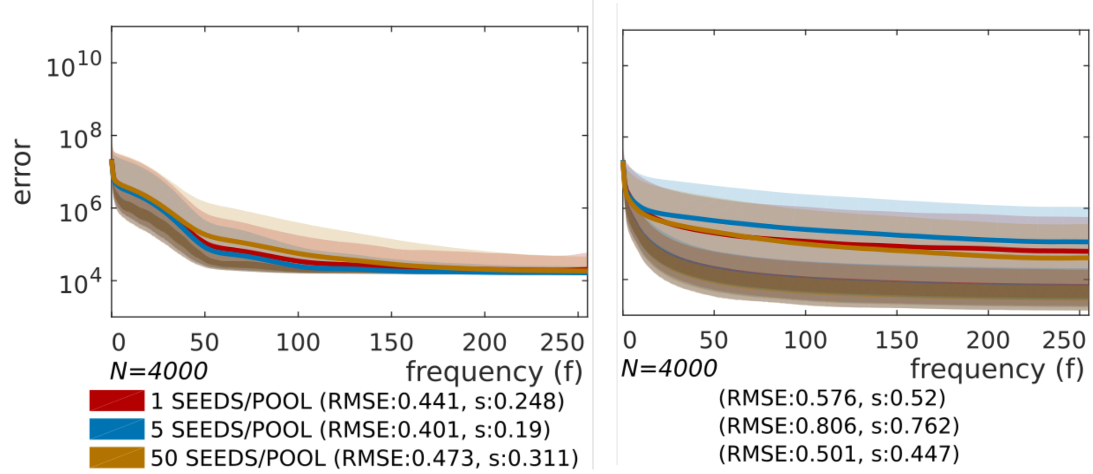
(c) *Door*



(d) *with equalised brightness*

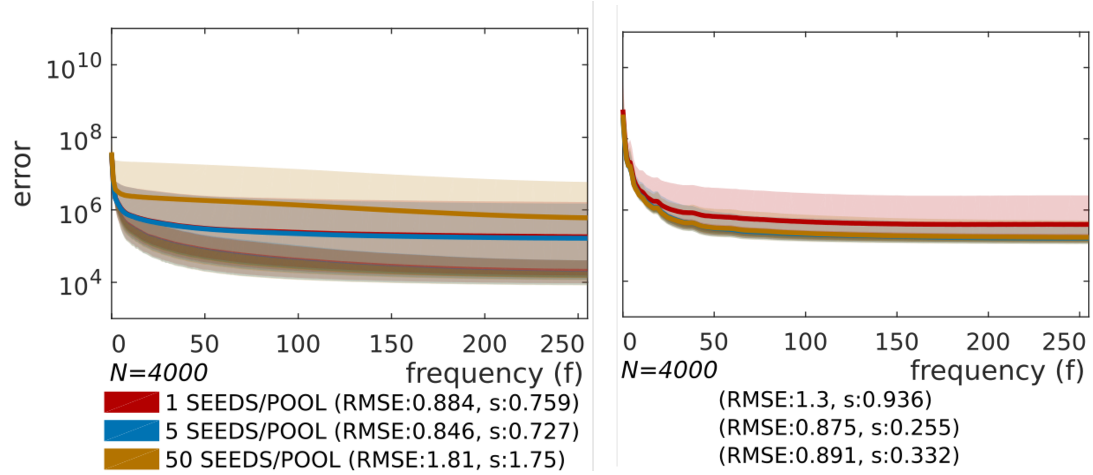
Figure 56: *More ESEs for varying size of the seed pool / number of luminance samples. The advantage that can be gained by increasing the pool size depends on the scene. Note however, that RMSE is strictly monotone falling in (a) and (c). See also Figure 55*

5.2. Number of seeds per pool



(a) *Box* **(b)** *Torus*

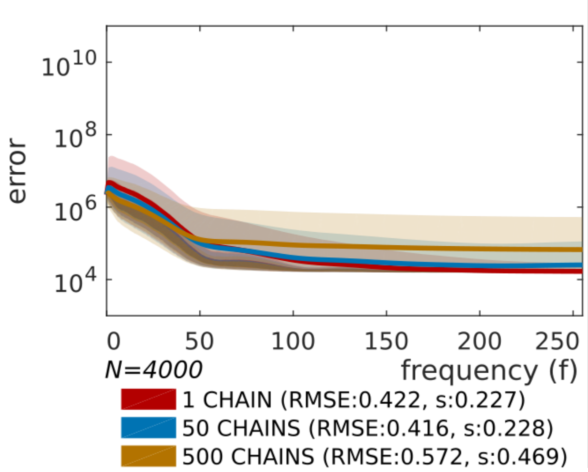
Figure 57: *ESEs for various pooling strategies of the luminance samples into seed pools. The total number of luminance samples is 10k, which is partitioned randomly into 50, 10 and 1 seed pools with 1, 5 and 50 chains, each.*



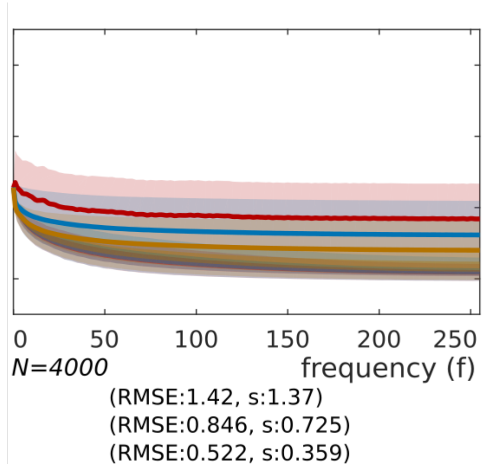
(a) *Door* **(b)** *Bathroom*

Figure 58: *More ESE for various partion variants of the luminance samples into seed pools.*

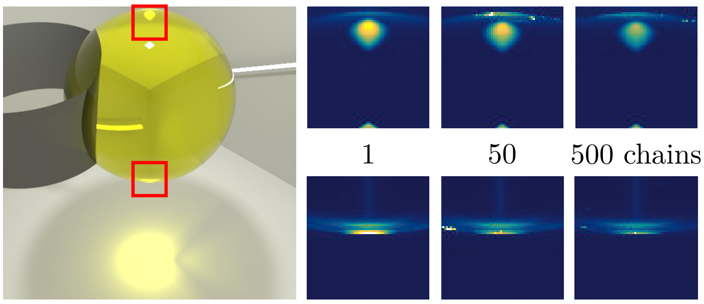
5.3. Chain length



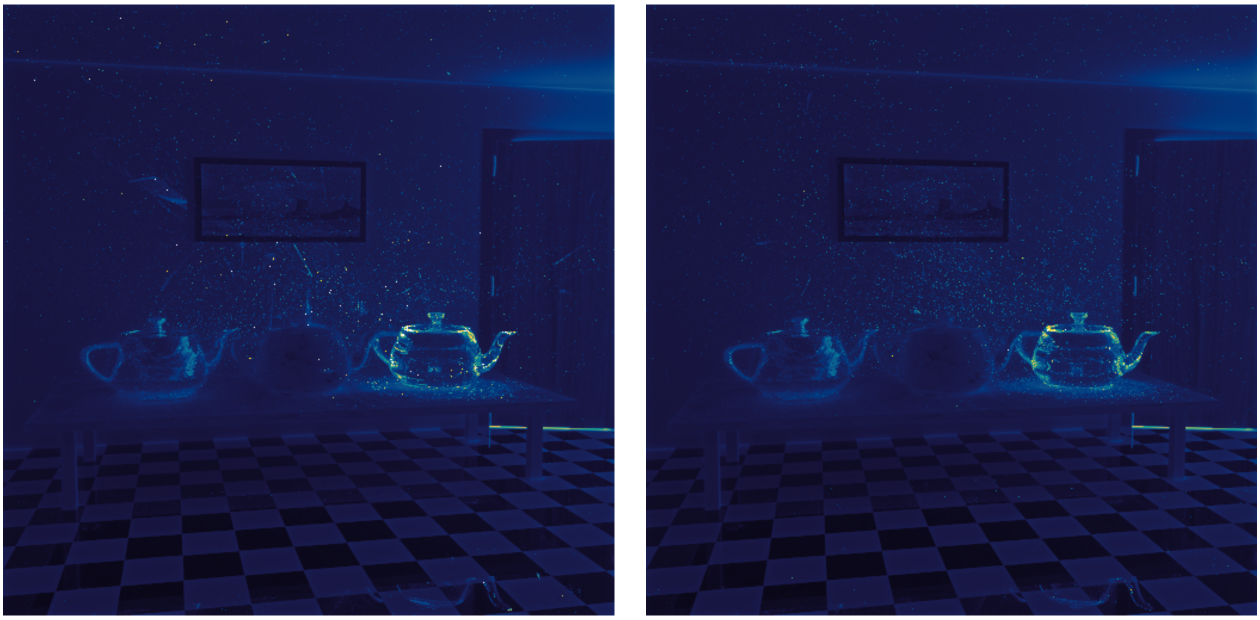
(a) *Box ESE*



(b) *Door ESE*

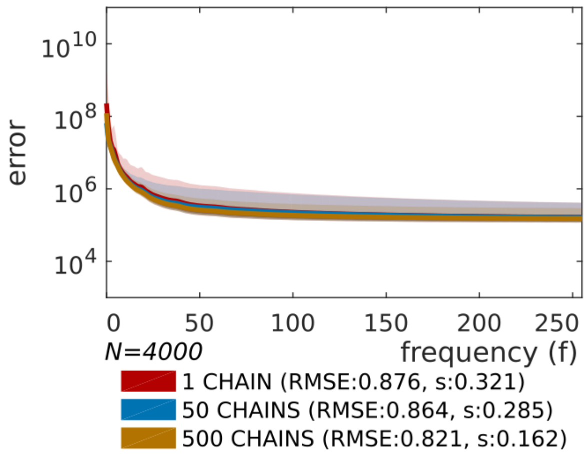


(c) *Box standard deviation*

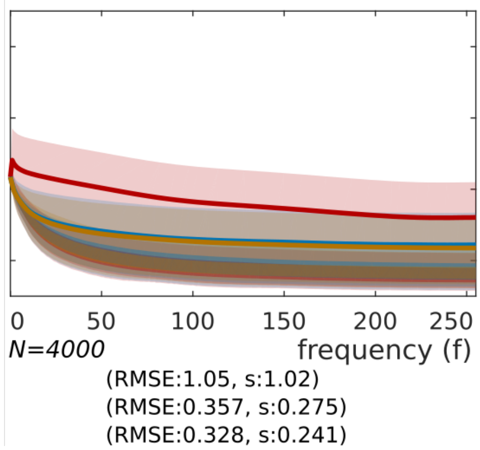


(d) *Door std. dev. (50 and 500 chains)*

Figure 59: *ESEs for varying chain lengths (100k seed pool, 50 samples per pixel). Reducing chain length shifts error in the box scene from low to high frequencies (a), i.e., less error in refractions (c) and more in pixel outliers. In the door scene, short chains reduce the amplitude of pixel outliers (d), cutting RMSE error to 36% (b). More scenes in Figure 60.*



(a) *Bathroom*



(b) *Torus*

Figure 60: *More ESEs for varying chain lengths and 100k seed pool size.*

5.4. Comparison with Energy Redistribution Path Tracing

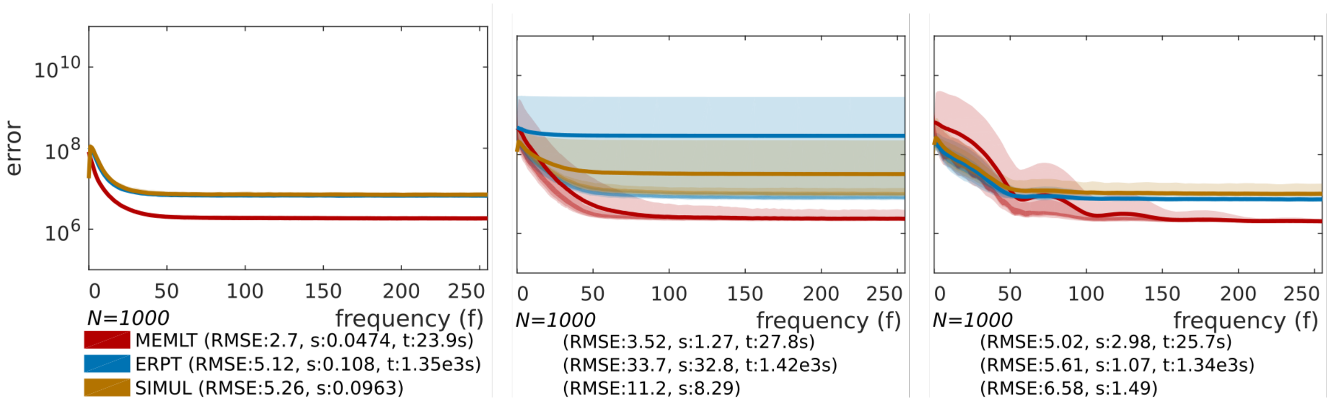
In this experiment we want to test how well energy redistributing path tracing (ERPT) can be simulated by MLT configured for short chains and a large seed pool (labelled SIMUL).

We used the following settings:

- 10 samples per pixel (those are seeding paths) for ERPT and accordingly $512 \times 512 \times 10 \approx 2.6e6$ luminance samples for MLT.
- on average 1 chain per pixel (they are started probabilistically) for ERPT and accordingly $512 \times 512 \approx 2.6e5$ chains for MLT
- chain length of 200 for ERPT and accordingly 200 samples per pixel for MLT
- ERPT and MLT use only the manifold mutation.

The rendering time for SIMUL is very long because mitsuba is not designed for this type of workload. Since the total number of samples is equal in both cases, we scale the error of the inefficient SIMUL using ERPT's time.

The results are in Figure 61, along with a normal MLT algorithm for comparison. The performance of ERPT and SIMUL are similar: The head of ERPT is slightly below SIMUL. ERPT has more outliers with roughness 1/64 (b), but that could be a coincidence. It is interesting, that SIMUL and ERPT in Figure 61c seem like a continuation of further increasing the number of chains in Figure 59a: Error in low frequencies is below default MEMLT and in high frequencies it is above.



(a) *Roughness* 1/4 **(b)** *Roughness* 1/64 **(c)** *Roughness* 1/1024

Figure 61: *ERPT test with box scene (varying roughness, light size is 50). SIMUL is MEMLT with parameters tuned to simulate ERPT (see text). MEMLT was not scaled to match ERPT/SIMUL, therefore only the shape can be compared. The figure shows that ERPT can be approximated pretty closely by tuning MLT parameters. (c) shows in addition that ERPT and SIMUL follow the head of MEMLT, avoiding the outliers in the tail of MEMLT. (b) on the other hand demonstrates that ERPT and SIMUL can also have a long tail. SIMUL is very slow because the architecture of the used rendering system was not designed for the tuned parameters (see text).*

5.5. PSSMLT Parameters

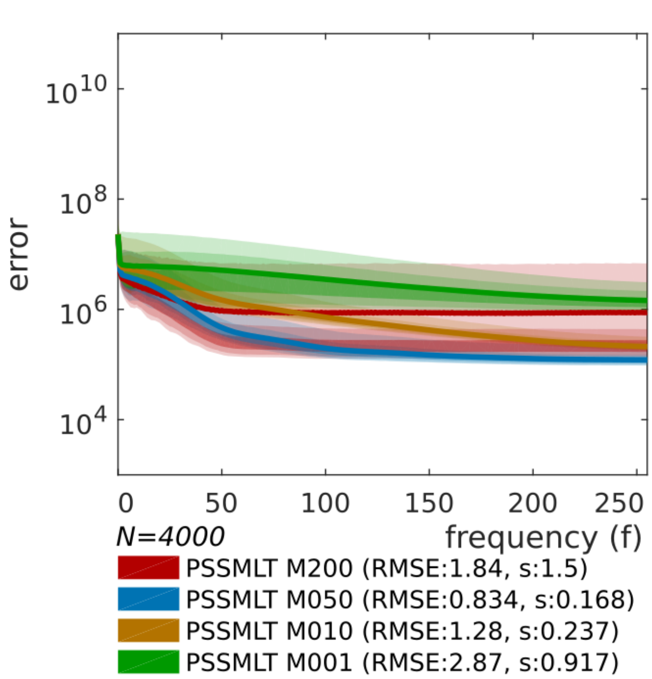


Figure 62: *ESEs for 200, 50, 10 and 1% of default small mutation size, 30% percentage large mutations. Box scene with roughness* $1/256$ *and light size* 50.

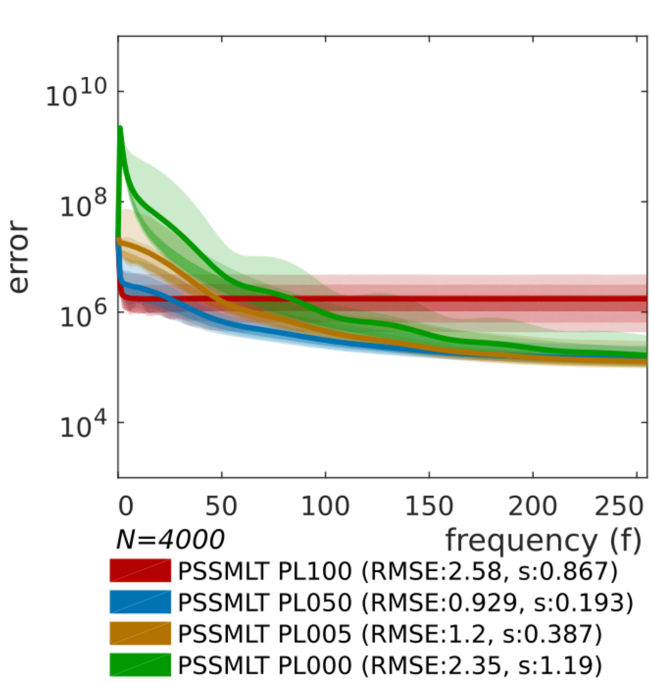


Figure 63: *ESEs for changing percentage of large mutations,* ***small mutations25%*** *of default value. Box scene with roughness* $1/256$ *and light size* 50.

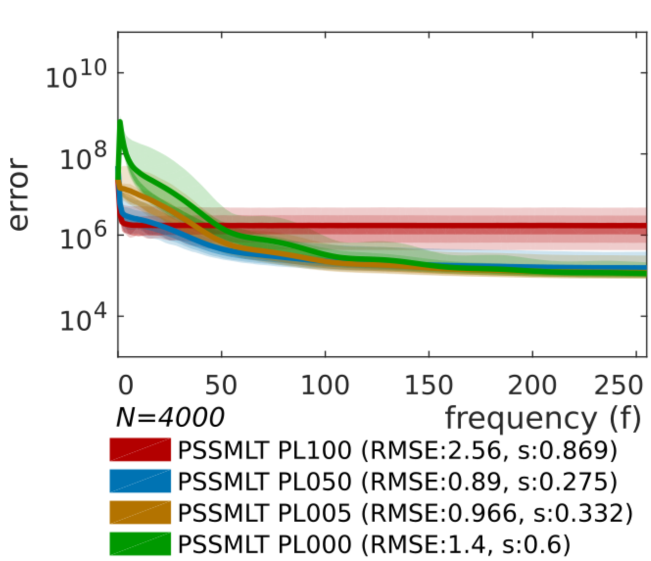


Figure 64: *ESE for changing percentage of large mutations,* ***small mutations 50%*** *of default value. Box scene with roughness* $1/256$ *and light size* 50*.*

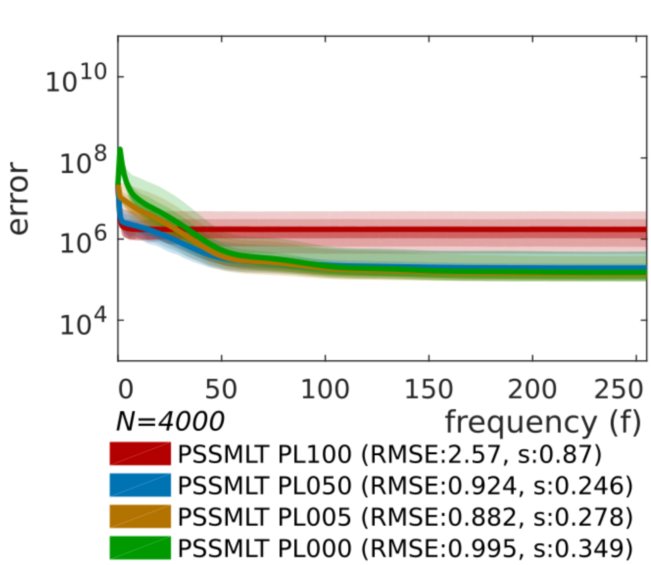


Figure 65: *ESE for changing percentage of large mutations,* ***default sized*** *small mutations. Box scene with roughness* $1/256$ *and light size* 50*.*