

Monte Carlo Method and advanced topics to understand statistical noise in TracePro

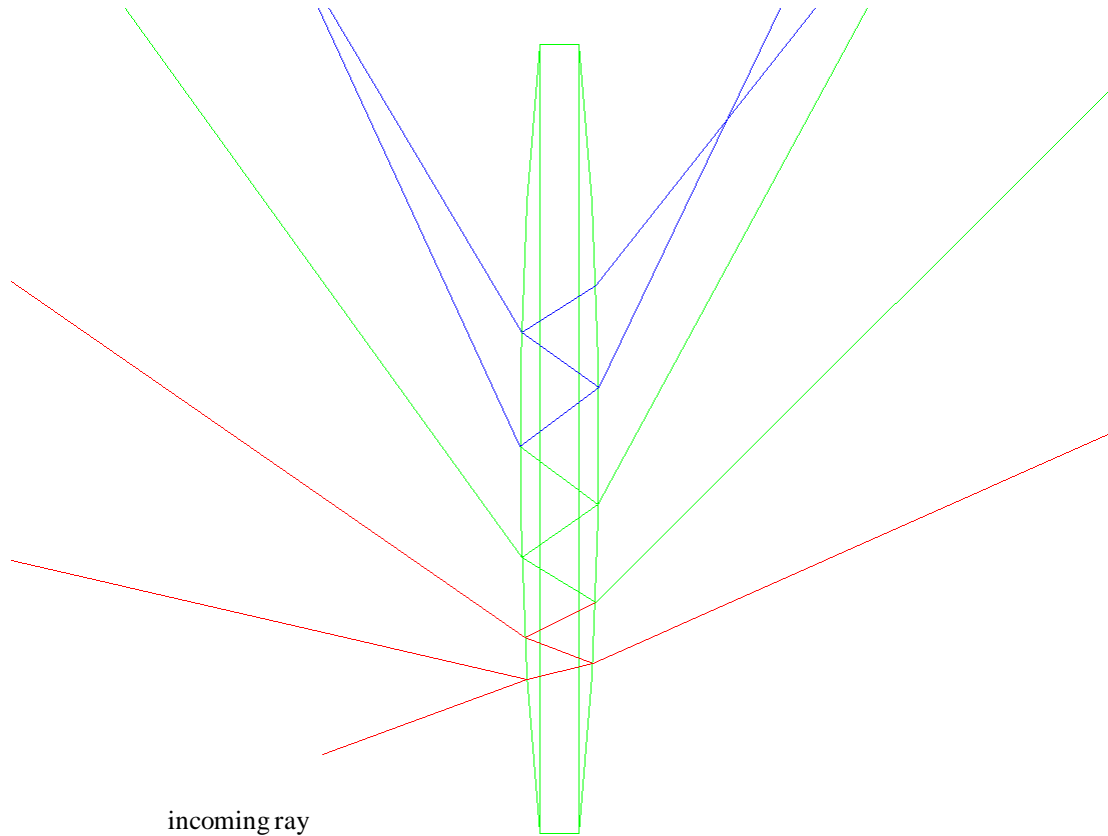
Monte Carlo Ray-tracing and Sampling used in TracePro

- A crude Monte Carlo calculation is the simplest form of a probability experiment
 - Perform an experiment N times, count the number of times n that the event occurs
 - An estimate of the probability is: $p_e = n / N$
 - We can never get an exact value of p_e , but we can make the uncertainty in p_e arbitrarily small by increasing N.
 - The absolute uncertainty in p_e is: $\sigma_{ab} = \sqrt{\frac{p(1-p)}{N}}$
 - The relative uncertainty in p_e is: $\sigma_{rel} = \sqrt{\frac{(1-p)}{pN}}$ (where p denotes the true probability)
 - *Hence, the accuracy of the result is inversely proportional to the square root of the number of trials*
- On a higher level, Monte Carlo is a technique of numerical integration for complicated multiple integrals that cannot be done by more conventional numerical methods
 - An integral such as $\int \dots \int g(x_1, x_2, \dots, x_L) dx_1 dx_2 \dots dx_L$ can be estimated by sampling the variables x_i , computing g for this set of samples, and repeating this process N times, summing the terms to obtain the estimate.

TracePro Variance Reduction Methods

- Variance reduction techniques are used to reduce the variance or uncertainty in the result of a Monte Carlo calculation after a given number of trials. Conversely, the number of trials needed to obtain a given uncertainty can be reduced.
- Splitting is a variance reduction technique used in Monte Carlo simulation. Ray splitting is used in TracePro.
- Importance sampling is a commonly used method for variance reduction.

Ray Splitting to reduce sampling issues



Importance Sampling

- As a simple example, consider a probability experiment to determine the chance of obtaining 3 on the roll of a pair of dice. Using crude Monte Carlo, we would roll the dice N times and count the number of times n that we get 3. We can also calculate the exact probability of obtaining 3 using the knowledge that each die face is equally likely to come up. The probability is

$$p = 2 \cdot \frac{1}{6} \cdot \frac{1}{6} = \frac{1}{18}$$

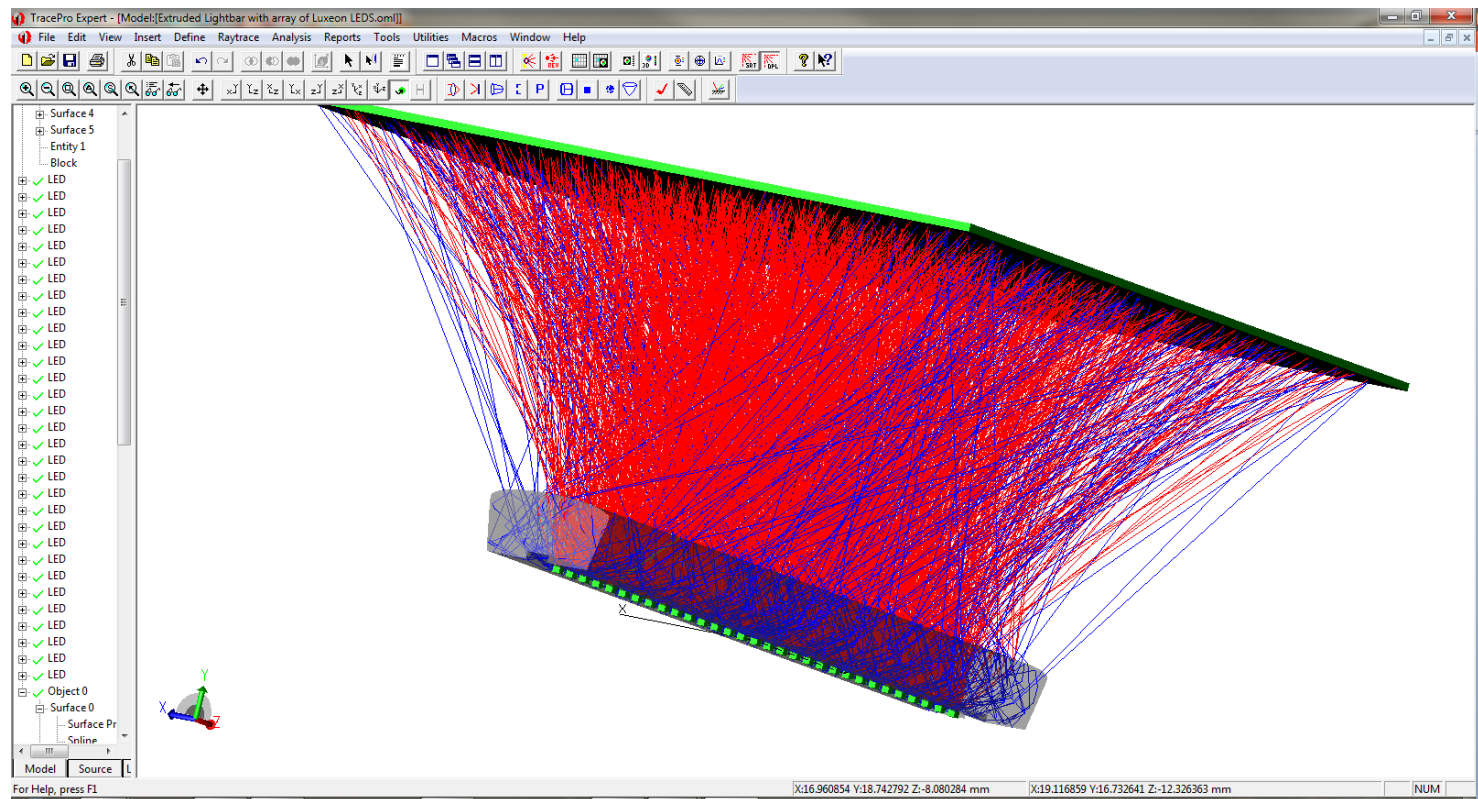
- Now suppose we load the dice so that ones and twos occur with probability $1/3$ instead of $1/6$. Then 3 would occur 4 times as often as before. The estimated probability of obtaining 3 is now

$$\hat{p} = \frac{1}{4} \cdot \frac{n}{N}$$

- That is, we increase the probability of threes occurring by a factor of 4, then divide the result of our experiment by 4 to obtain the true probability. This is the basis of importance sampling.
- This technique as applied in TracePro causes scattered rays to go in certain directions with higher probability than they would if scattered at random.

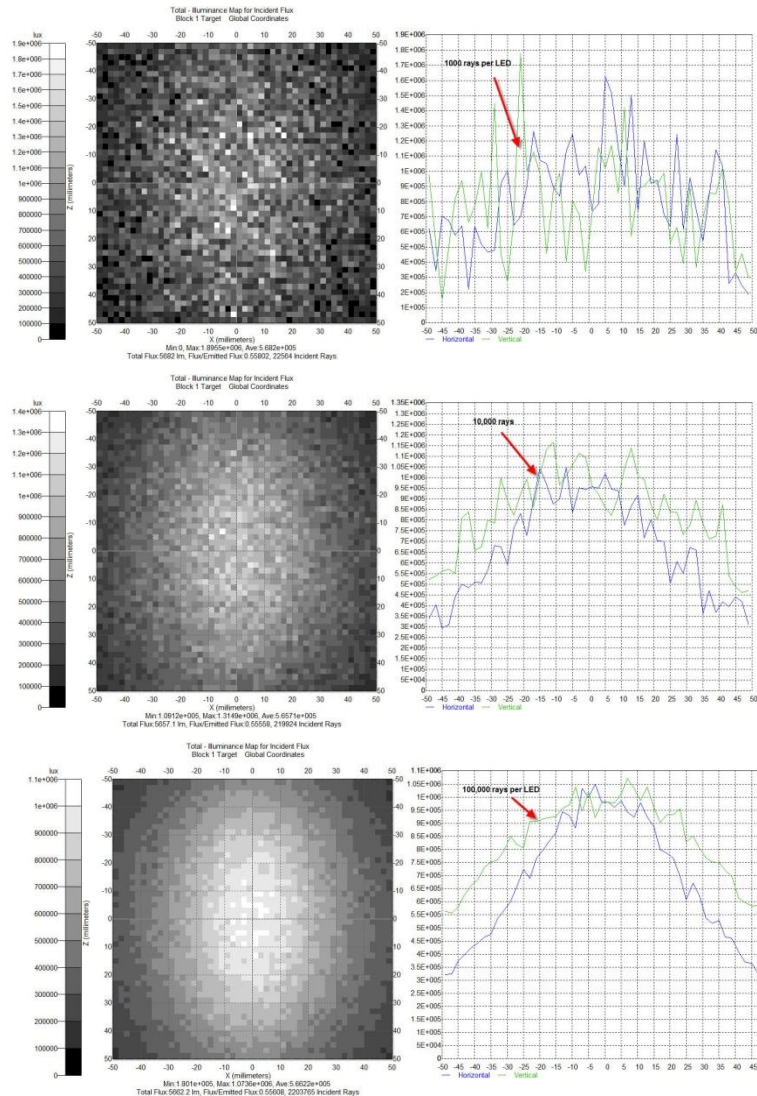
Methods to Reduce Noise

- Trace more rays, TracePro is now 64 bit compliant and has multi-threading to trace more rays and in much less time. For example accurate raytracing of LEDs should be over 100,000 rays to reduce sampling noise when trying to find artifacts in the light pattern.. We will use the following system shown below with 31 LEDs and a light pipe to discuss the number of rays that should be traced for a simulation.



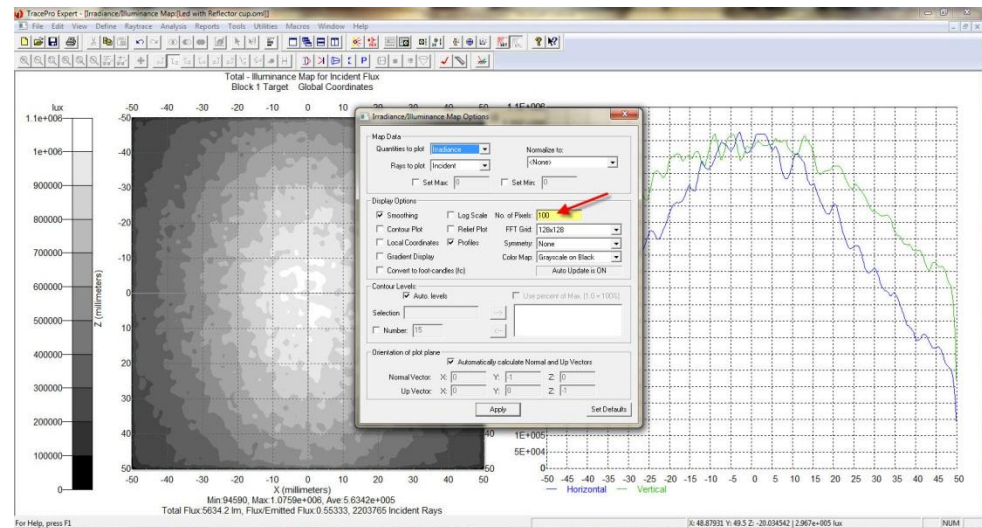
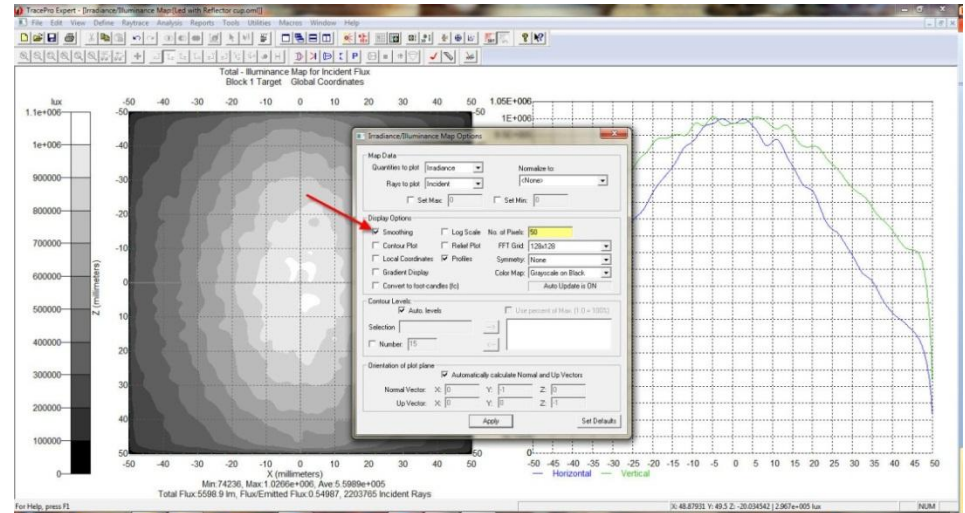
Methods to Reduce Noise

- Lets look at the raw data for 3 cases where we trace 1000, 10000 and 100000 rays from each LED without smoothing and a pixel count of 50.
- If we look at the flux for each case, we see very good agreement, 1000 rays - 5682 lumens, 10,000 rays - 5657 lumens and for 100,000 rays - 5598 lumens.
- But the illuminance plots are quite a bit different and we can easily see the noise in each plot, especially in the profiles on the right of each plot.
- We can see by tracing more and more rays that the pattern fills in and that in the end we will expect to see a smooth line if we trace even more rays.



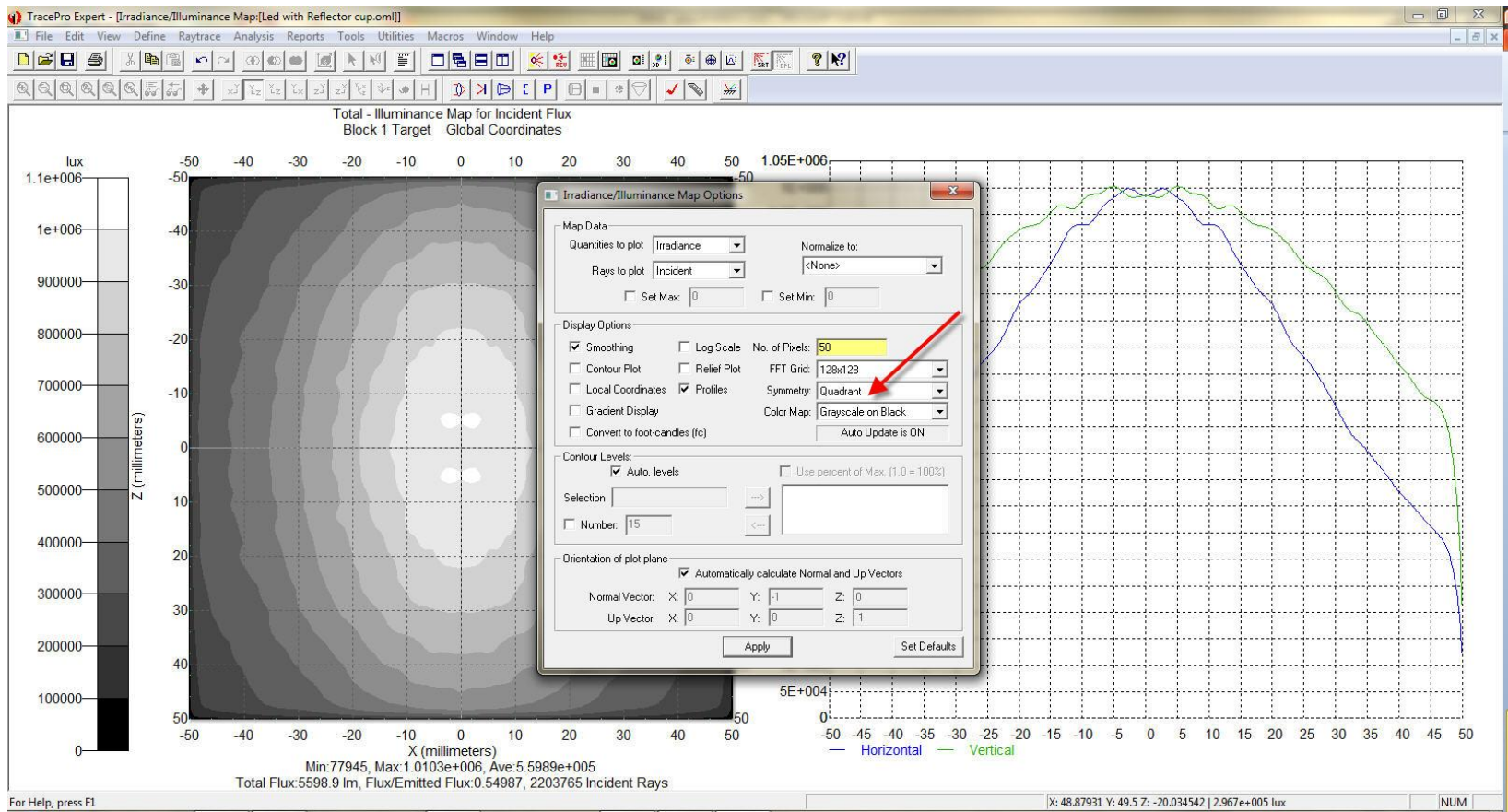
Methods to Reduce Noise

- If we use smoothing we start to approach a smooth line curve. If we look at max flux for the smooth and unsmoothed curves the maximum changes from $1.026\text{E}+6$ for the smooth case to $1.07\text{E}+6$ for the non-smoothed case. Not much of a change.
- If we change the number of pixels to 100, 4 times as many bins to catch the rays the maximum flux on the plot is $1.07\text{E}+6$.
- So we must run a fine line between smoothing, number of pixels and number of rays to trace to get the best answer



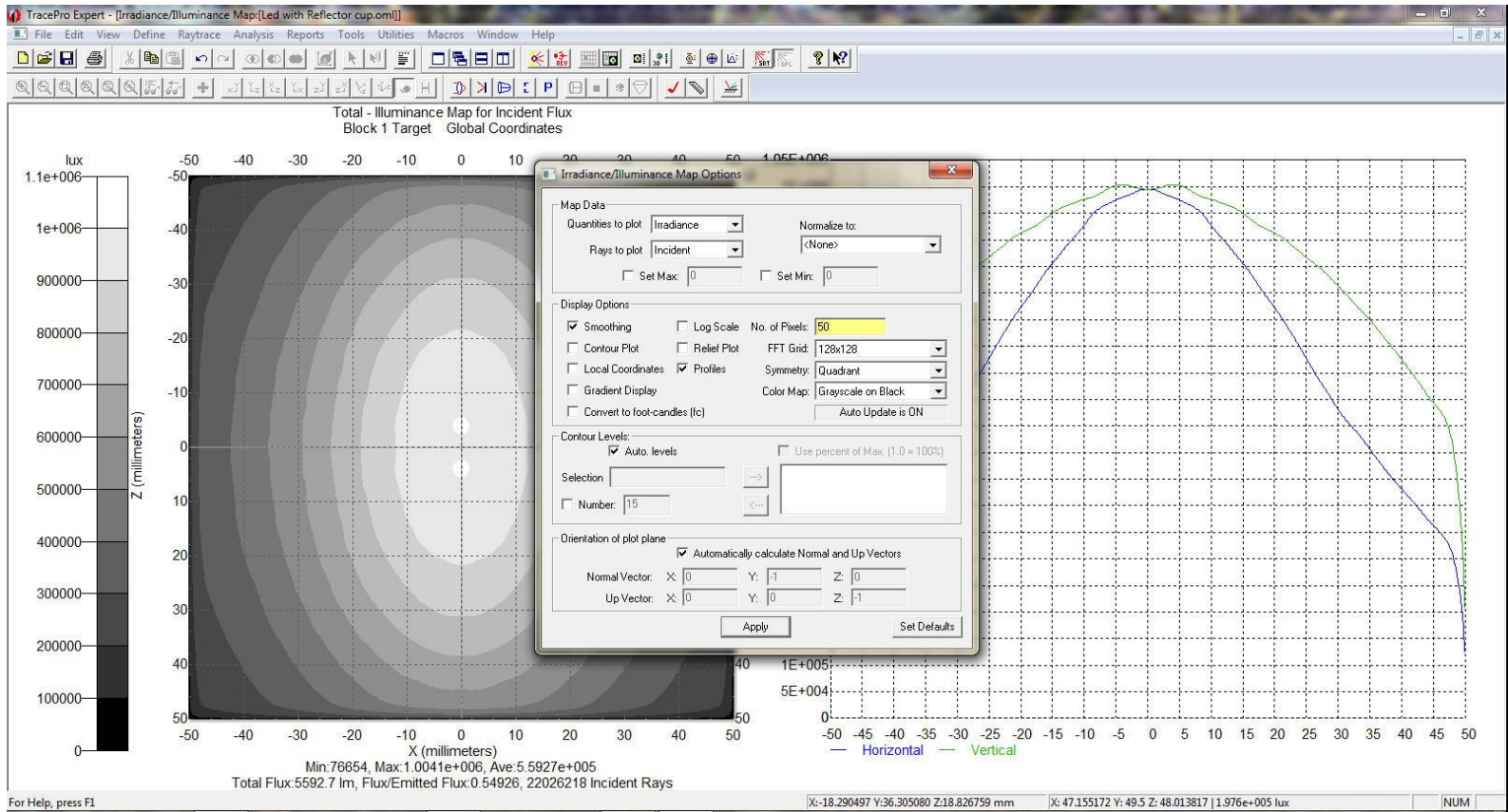
Methods to Reduce Noise

Use symmetry conditions when possible in illuminance maps and candela plots to get better answers. We know that in our LED/lightpipe system that we have quadrant symmetry in the system. The lightpipe is symmetric about both axes. This allows the user to add the illuminance from all four quadrants together and then divide the result by 4 to get a more accurate answer.



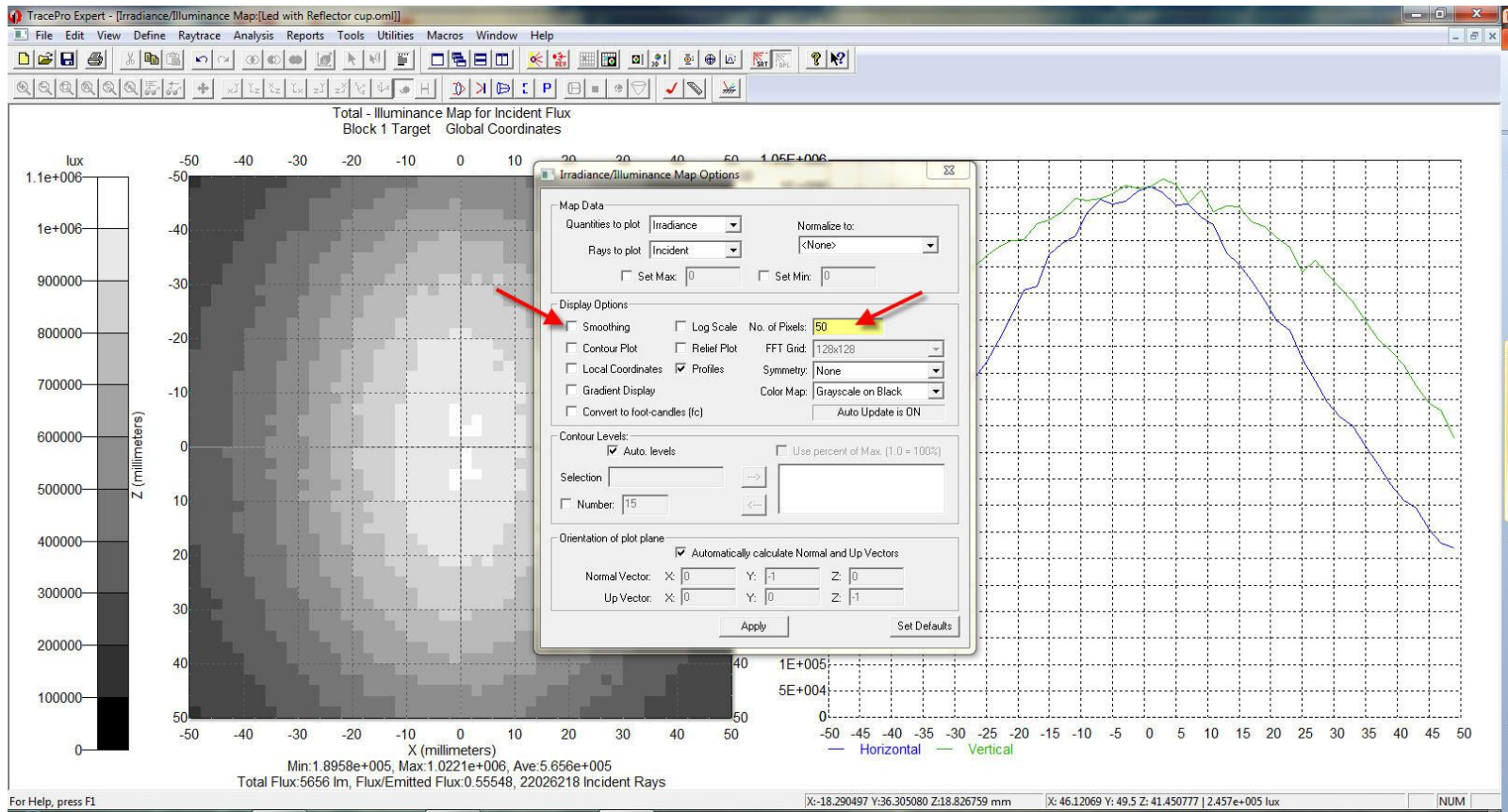
Methods to Reduce Noise

A final raytrace with 1 million rays per LED for a total of 31 million rays is shown below, over 22 million hit the target. The final answer with smoothing, 50 pixels and quadrant symmetry is shown below. As you can see the lines are very smooth and there is very little noise in the plot.



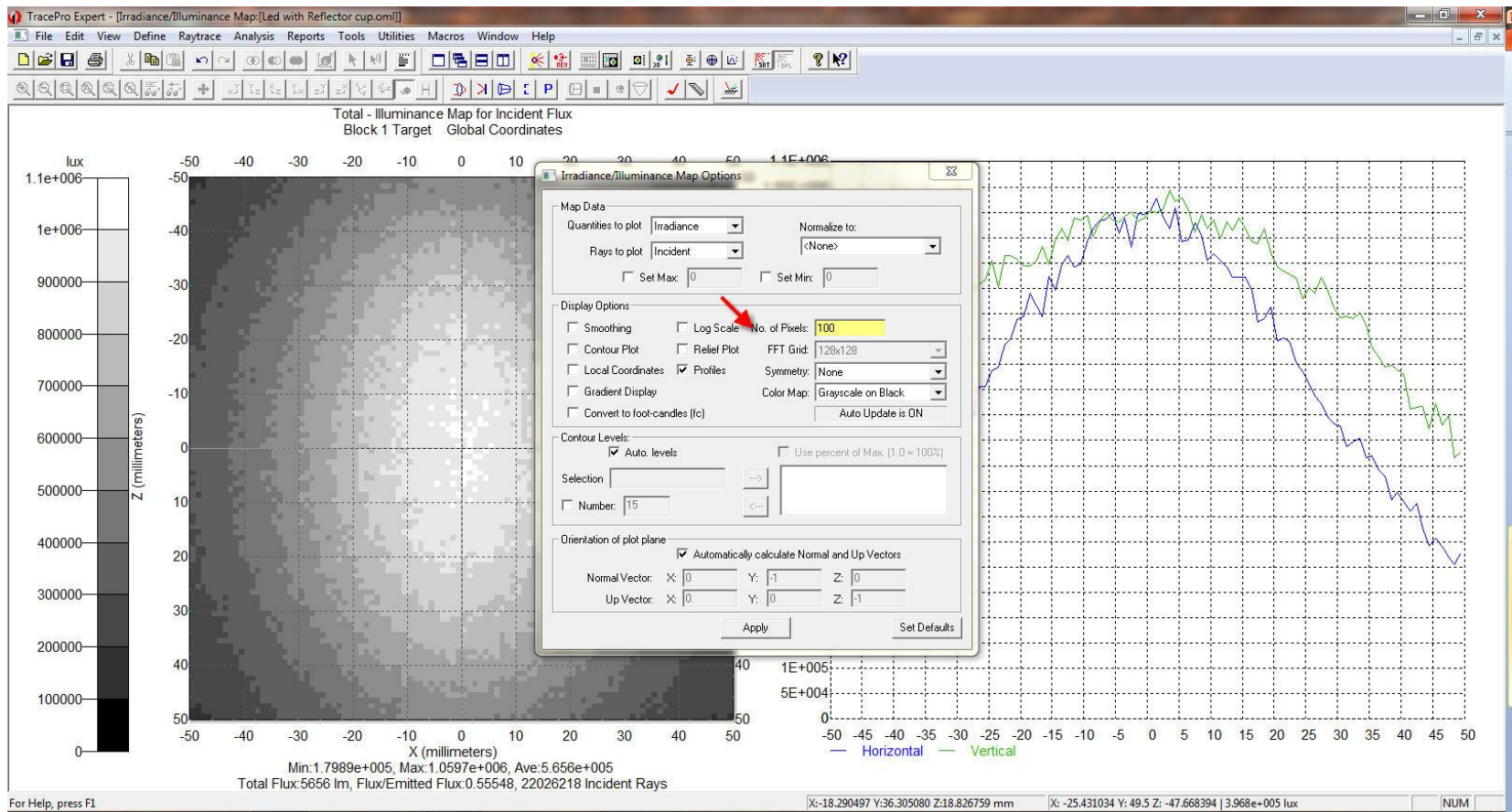
Methods to Find Artifacts

The illuminance map for the same 1 million per LED raytrace is shown below. This map has no smoothing or quadrant symmetry. If you are looking for artifacts, this is the plot you should use. Each one of the pixels or bins is 2 mm in size, 100 mm square target with 50 bins. If you are looking for artifacts smaller than 2 mm in size you will need to increase the number of pixels or bins.



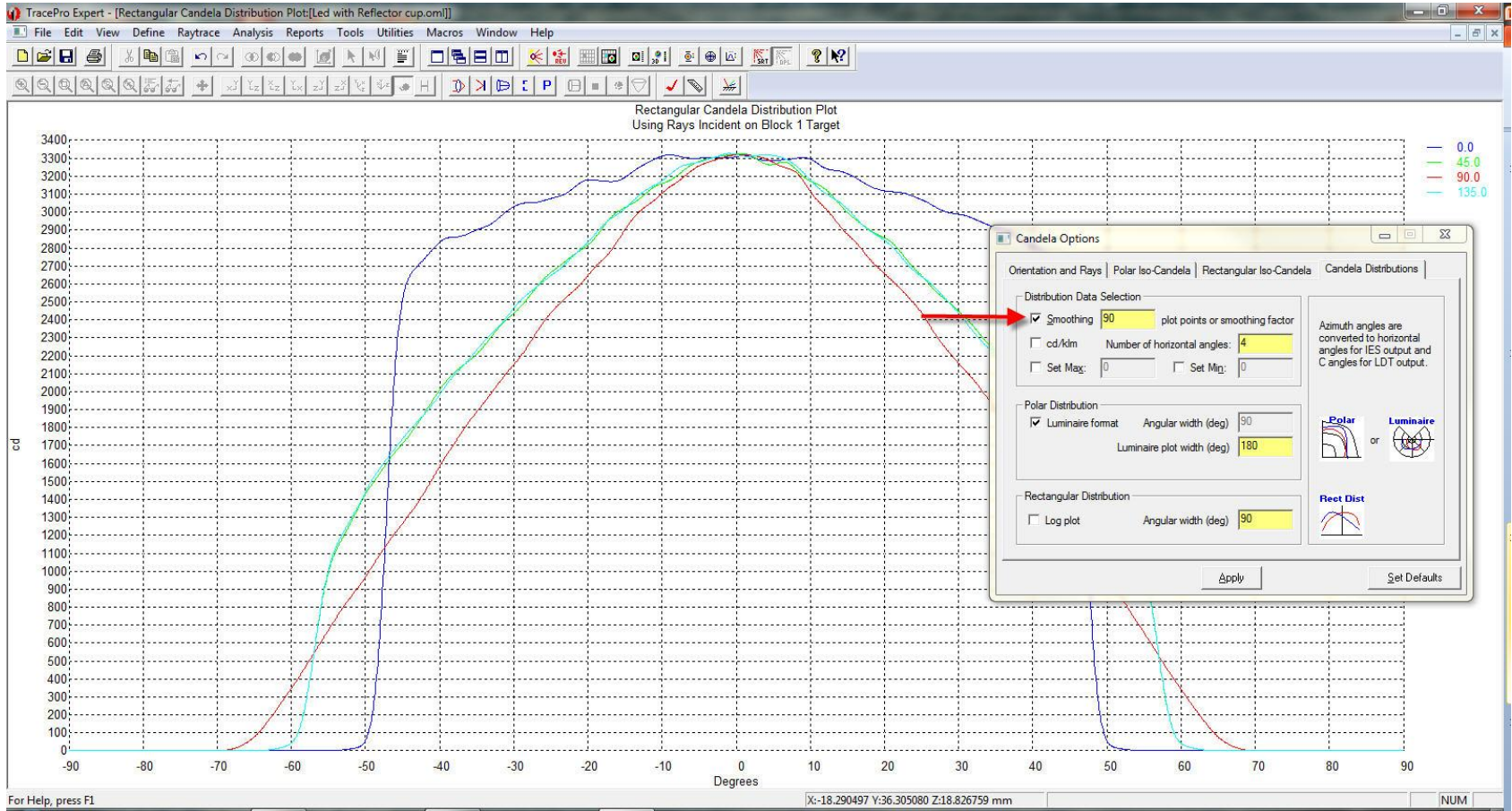
Methods to Find Artifacts

If we want to find artifacts that are 1 mm in size, all we need to do is set the pixels in the illuminance options to 100 and now the pixel or bin size will be set to 1 mm. The illuminance map for the 100 pixel setting is shown below. So judicious use of the illuminance map options and an understanding of how many rays to trace is important to output a correct answer.



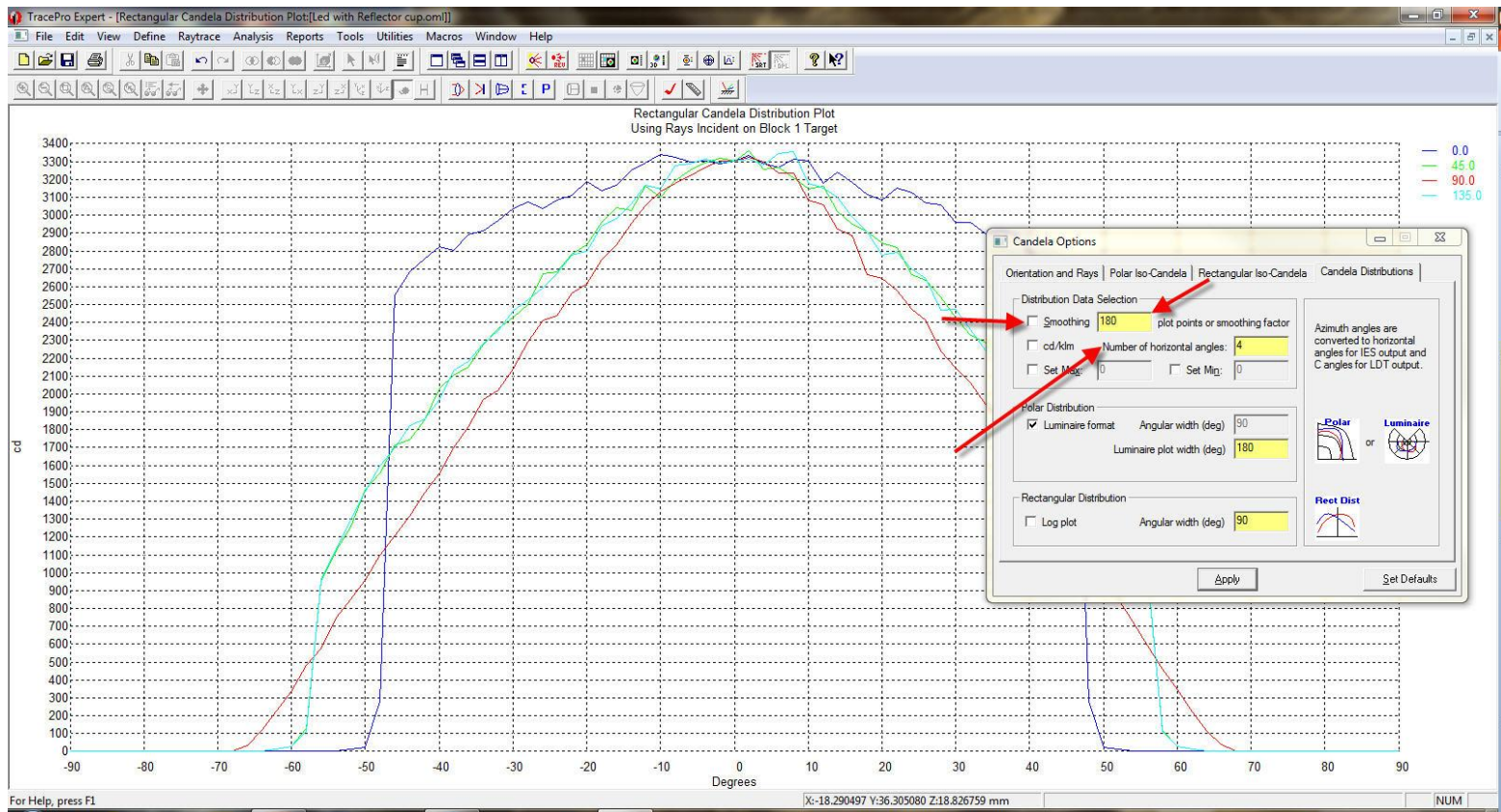
Methods to Find Angular Artifacts

For candela plots the number of bins is in angular space for the rectangular candela distribution plot. The plot points refer to the angular bins for the entire hemispherical plot which is set to 90 bins which specifies an angular bin resolution of 360 degrees for the extent of the plot divided by 90 bins or 4 degrees for each bin.



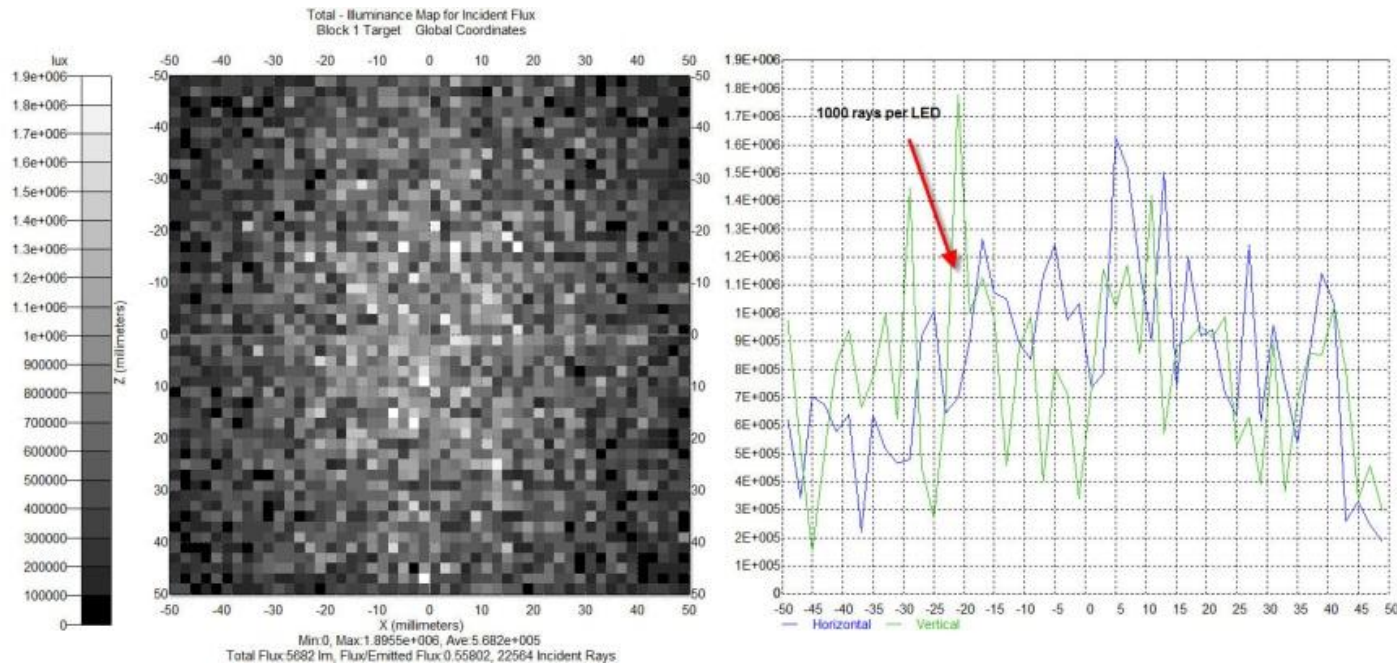
Methods to Find Angular Artifacts

If we set the options to no smoothing and 180 bins then we are plotting raw data at every 2 degree increment. Here we can see the plot is getting noisy even though we have a high number of rays, over 22 million hitting the target. When you think about plotting the entire angular space, this is 64,800 bins which is 360 degree horizontal by 180 degrees vertical. A good number of rays in each bin is 8 for good data collection, especially at the outer angles. This means we need to trace at least 518,400 rays if the rays are evenly distributed across the bins.



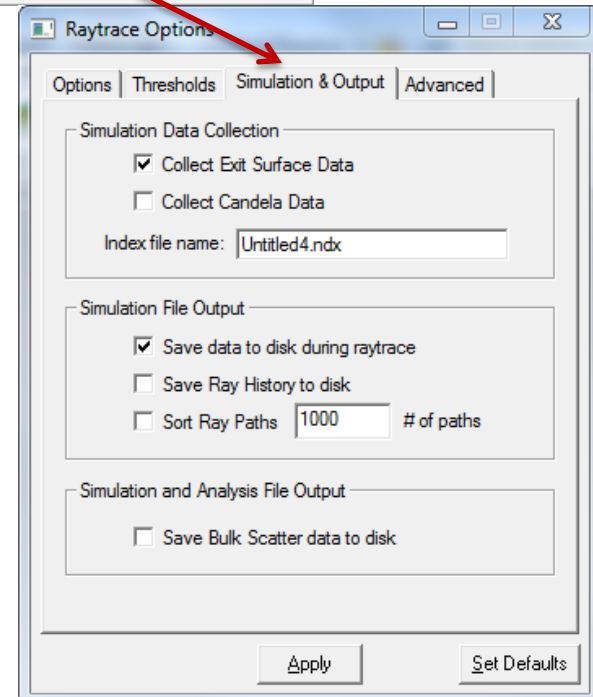
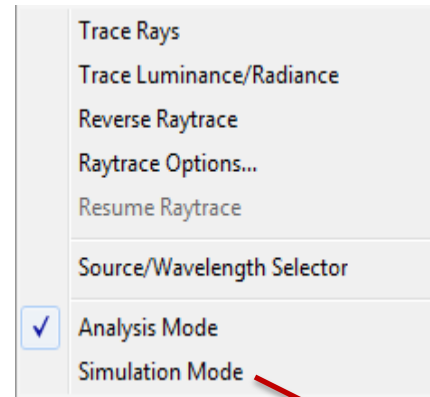
Methods to Validate Data

- Look at the data without smoothing to determine if you have enough rays to make create a good figure. In the first case when we traced 1000 rays, we just don't have enough data to determine a result. But if we look at the 1000, then 10000 and finally the 100,000 ray trace we can see the answer starting to emerge. If you have the time to trace 1,000,000 rays and you are looking for artifacts in a design, this is the only way to do it. 100,000 LEDs is just not enough to determine these as you can see from these illuminance plots.



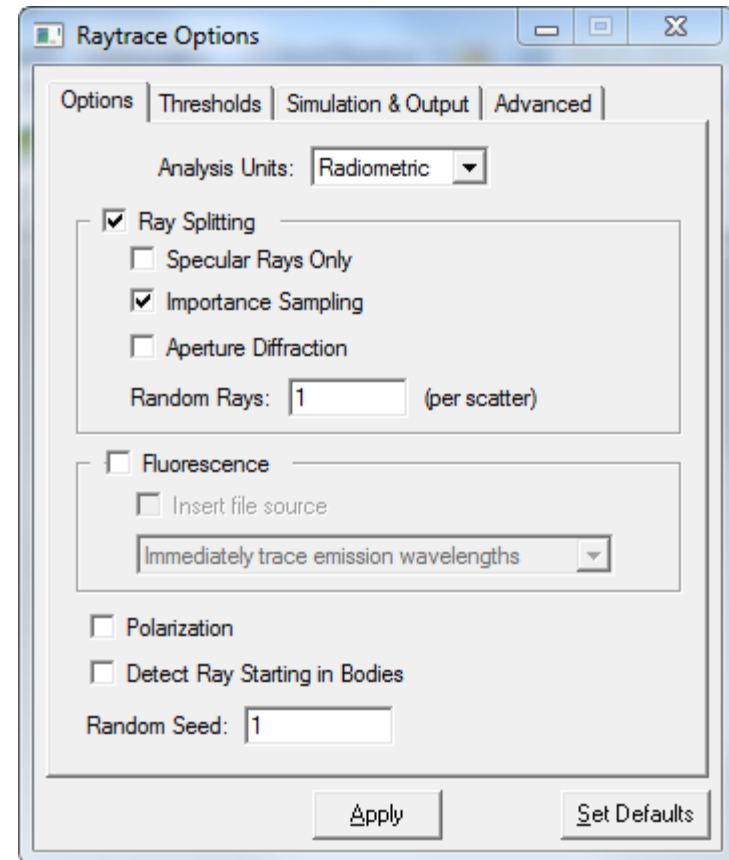
Analysis and Simulation Modes

- Analysis Mode
 - Stores all ray tree nodes
 - Very memory intensive
 - Allows plots and tables for all model surfaces
 - Permits Ray History
 - Display rays
 - Path Sorting Table
 - 3D irradiance/illuminance
- Simulation Mode
 - Stores ray data incident on a single surface
 - Stores ray data for missed rays
 - Much less memory used
 - No Ray Histories
 - No Ray display
 - Path Sorting available saved in file
- You can run out of memory in both modes
 - Be judicious about flux threshold
 - Use the task manager to monitor memory use
 - Use TracePro's flux report to do smaller raytraces and see how much memory is used and then extrapolate to make sure there is enough memory before starting the raytrace
 - Raytraces of 50 million rays in simulation mode are possible with 8MB on a medium system
 - 1 Billion rays have been raytraced with 16GB RAM systems with large virtual page sizes



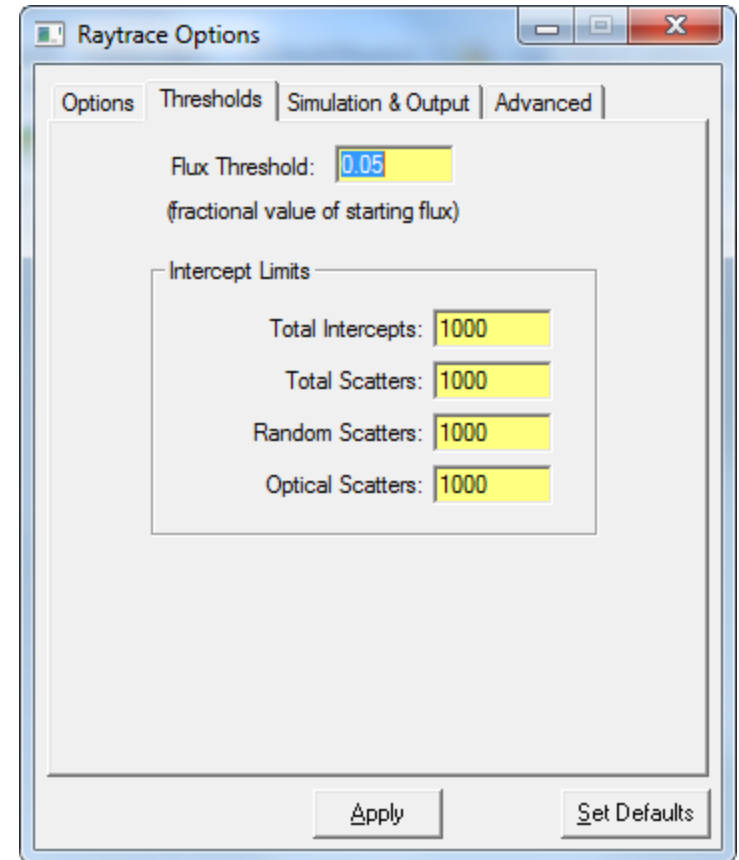
Methods to speed up Raytracing and reduce memory requirements

- Ray-trace features may be turned off to skip some processing during the ray trace to save memory
- Multiple random rays may be used to increase sampling but requires lower flux threshold and more processing time and memory to trace the rays.
- Fluorescence ray tracing can be controlled and set to run as a single or two-stage operation.
- Changing the random seed provides a “new set” of random numbers for a raytrace.



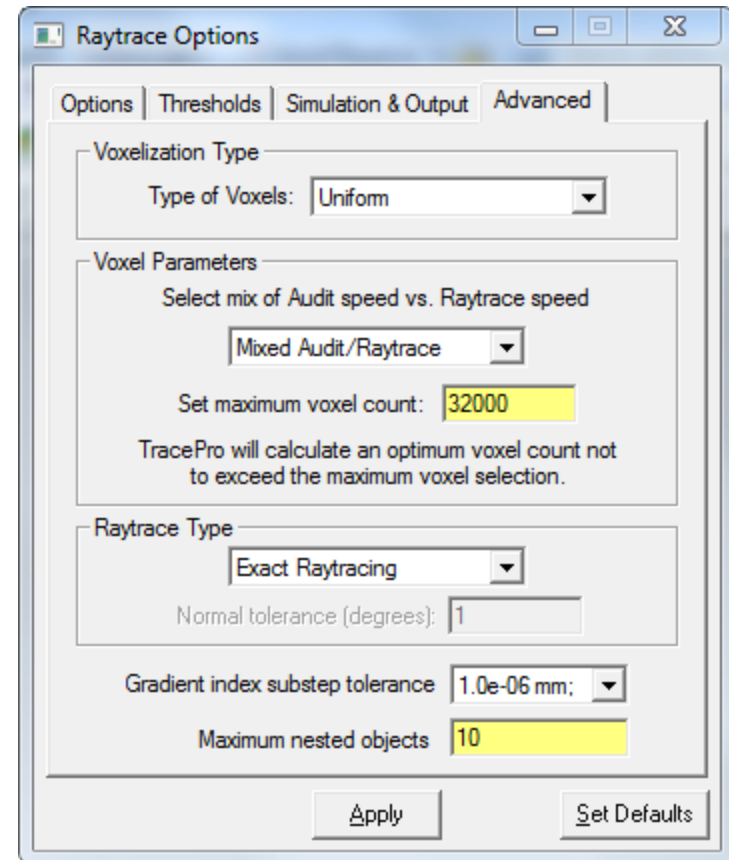
Methods to speed up Raytracing and reduce memory requirements

- Thresholds specify how rays are terminated.
- Five thresholds can be set by the user
 - Flux Threshold
 - fractional value of starting flux
 - Good value for illumination system is .05
 - Ghost analysis should be set to .001
 - For stray light and importance sampling $1E-6$
 - Total Intercepts
 - Total Scatters
 - Random Scatters
 - Optical Scatters

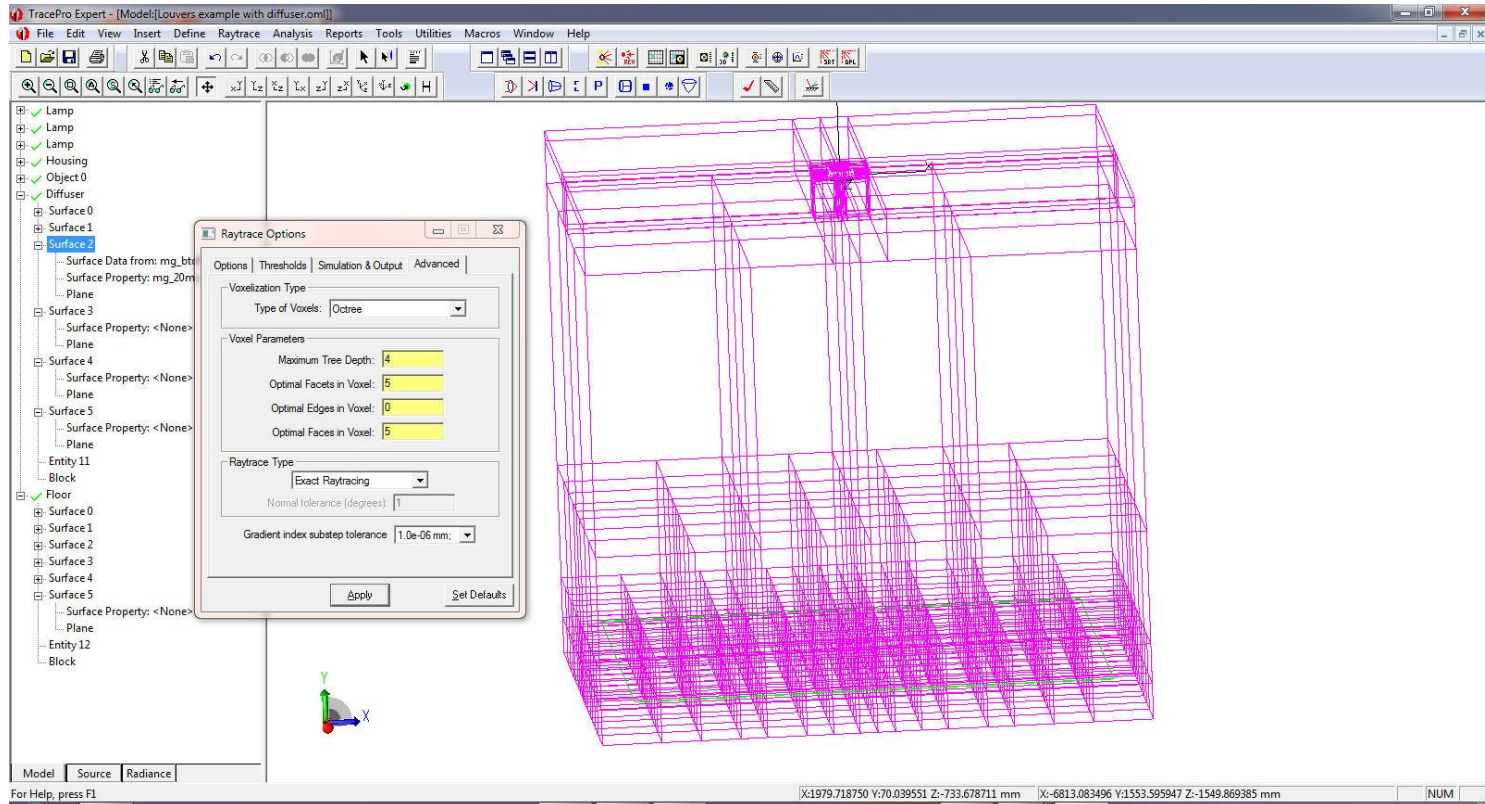


Methods to speed up Raytracing and reduce memory requirements

- TracePro uses Space Partitioning to improve raytrace speed via two types of Voxels.
 - Uniform
 - You can select voxel count:
 - Raytrace speed increases with more voxels.
 - Audit speed decreases with more voxels.
 - Use Fastest Audit during initial model development.
 - Use Fastest Raytrace during simulation.
 - Octree
 - Improves raytrace efficiency for models with large distances between objects which is especially good for systems with far away targets
 - Use View|Display Voxels after an Audit to view voxelization



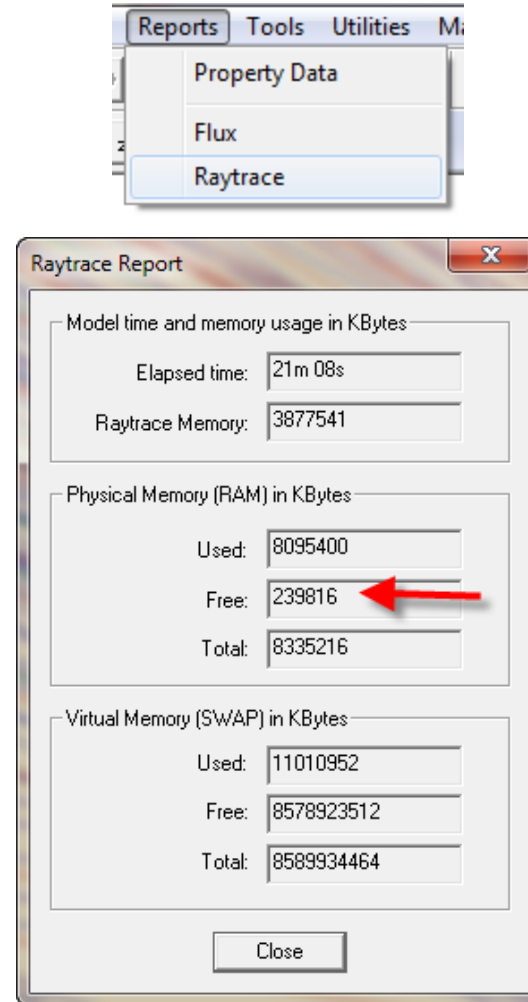
Methods to speed up Raytracing and reduce memory requirements



Use Octree Voxels if you have large amounts of unused space in your model to speed up raytracing. Use the View|Display Voxel option to show the voxels in your model after you audit the model.

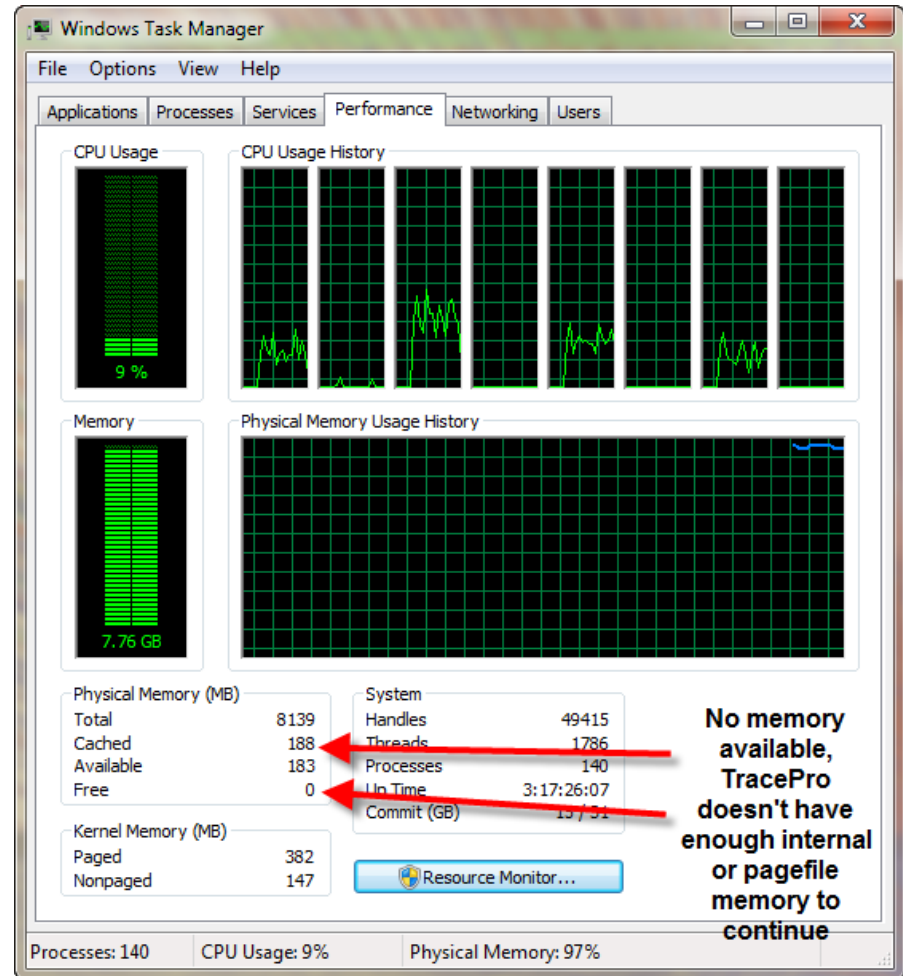
Methods to speed up Raytracing and reduce memory requirements

Use the Reports|Raytrace menu option to see how much memory was used in a raytrace. For our sample system with 100,000 rays traced for a total of 3.1 million rays, we only have 239816 Kbytes free. We are not going to be able to trace many more rays before running out of memory.



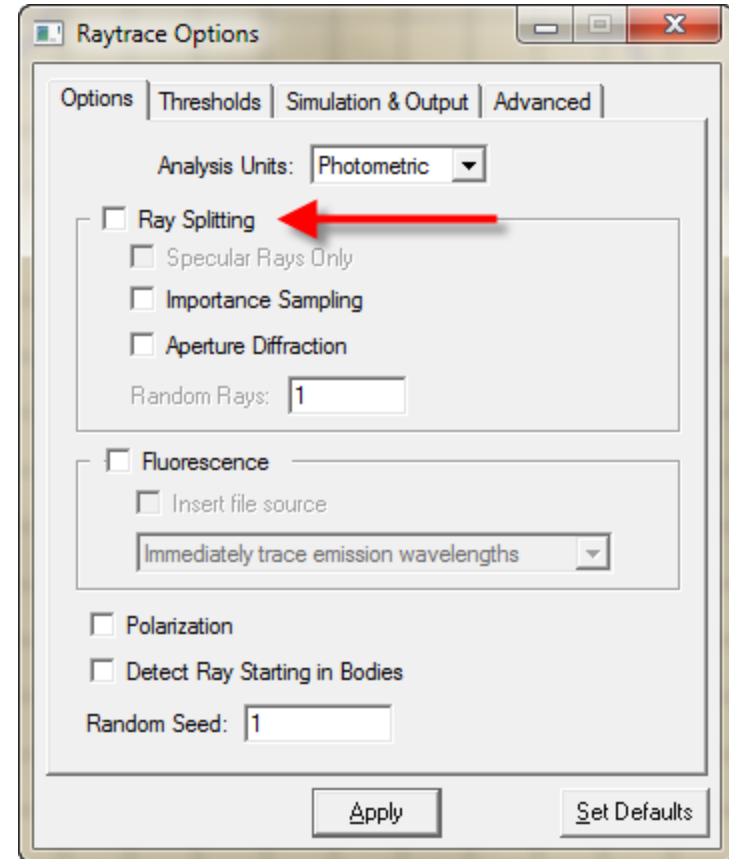
Methods to speed up Raytracing and reduce memory requirements

Use the Windows Task Manager to keep tabs on your memory and how well the multi-core raytrace is taking advantage of all the cores in your computer. If you run into a memory situation as shown at right, (3.1 million rays for our sample system) you will need to switch from analysis mode to simulation mode or change your threshold or scattering settings to be able to raytrace further. Adding RAM, or increasing Virtual Page Size will also help this memory problem.

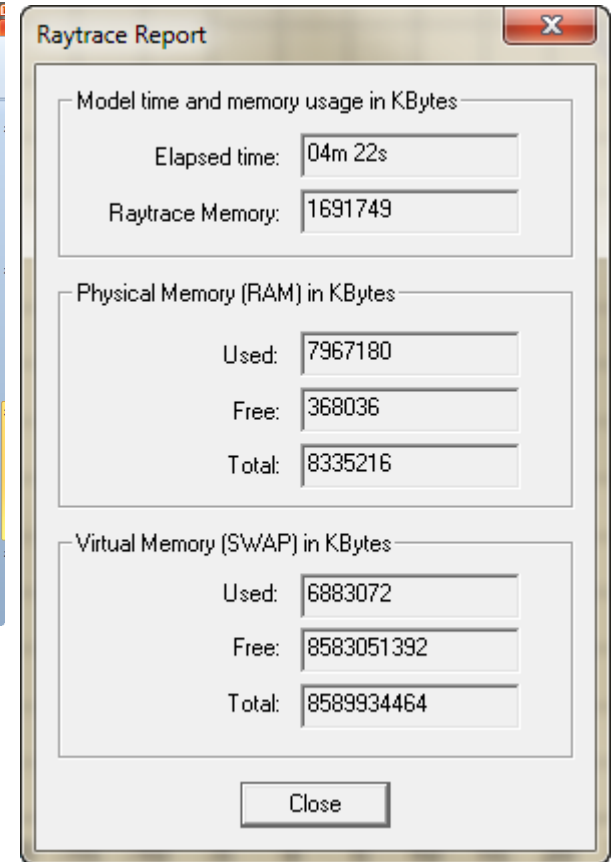
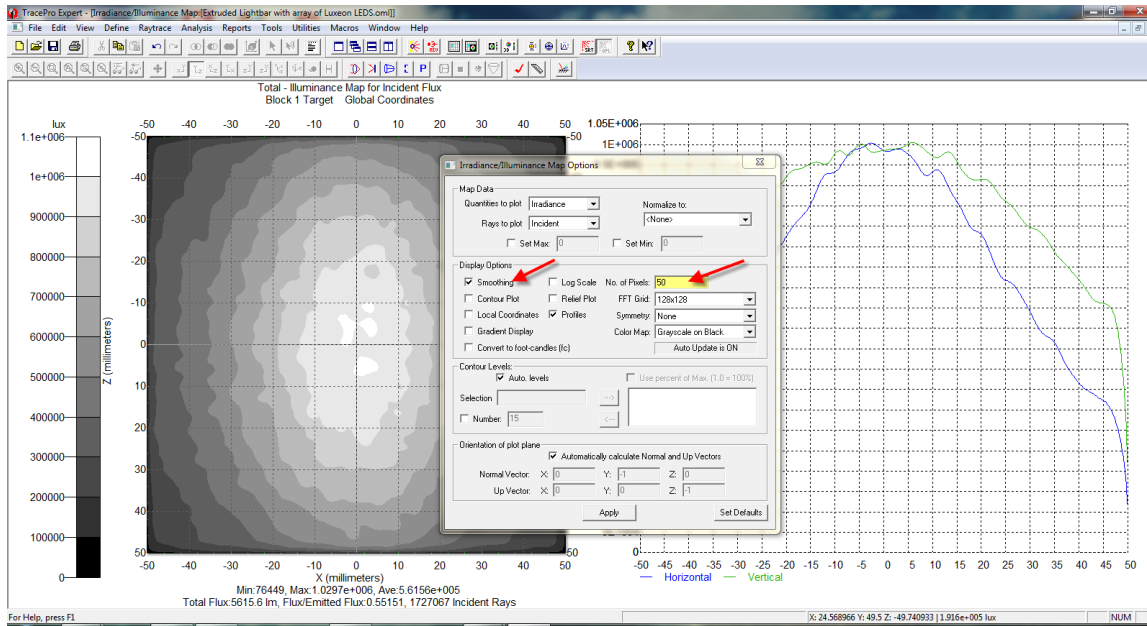


Methods to speed up Raytracing and reduce memory requirements

There is a second method of ray splitting in TracePro. This option is activated when we turn off ray splitting. Instead of splitting the ray at each surface, the ray is not split but the percentage of the ray interaction with the surface is taken into account. So for this method if we trace 100 rays that intersect a lens surface that is 5 percent transmitting, 4% fresnel reflecting and 1 percent reflective scattering here is how TracePro would handle this case. 95 rays would transmit through the surface, 4 would reflect off and 1 ray would be scattered using random number generation on average. This would use less memory, speed up raytracing and provide excellent answers if enough rays were traced.



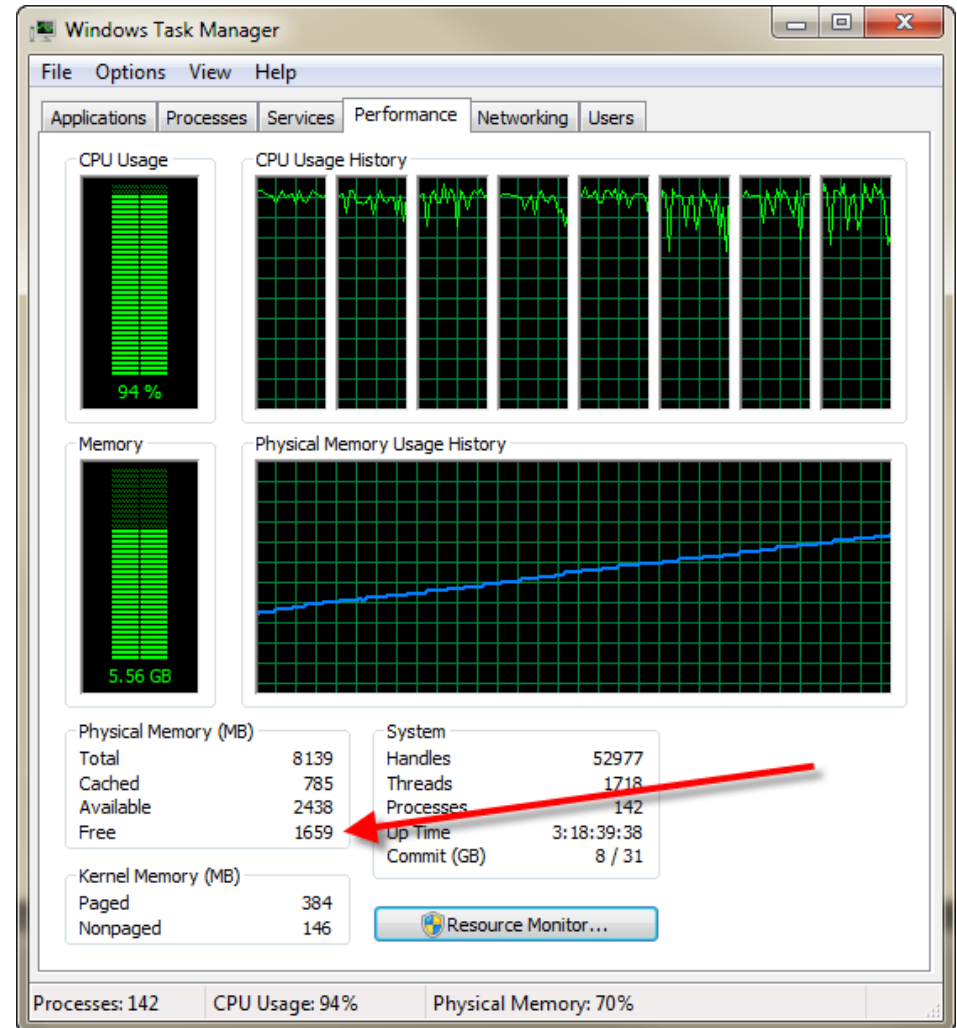
Methods to speed up Raytracing and reduce memory requirements



Using this second raytracing method, we get very similar results for the illuminance map for 10,000 rays per LED for our sample system. But it only takes 4 minutes to raytrace instead of 21 minutes and only uses 1691749 Kbytes of memory instead of 3.8 million for the ray splitting raytrace. **But for this method to be accurate you need to trace large quantities of rays since ray splitting is a variance reduction technique that provides excellent answers in low ray count simulations.**

Methods to speed up Raytracing and reduce memory requirements

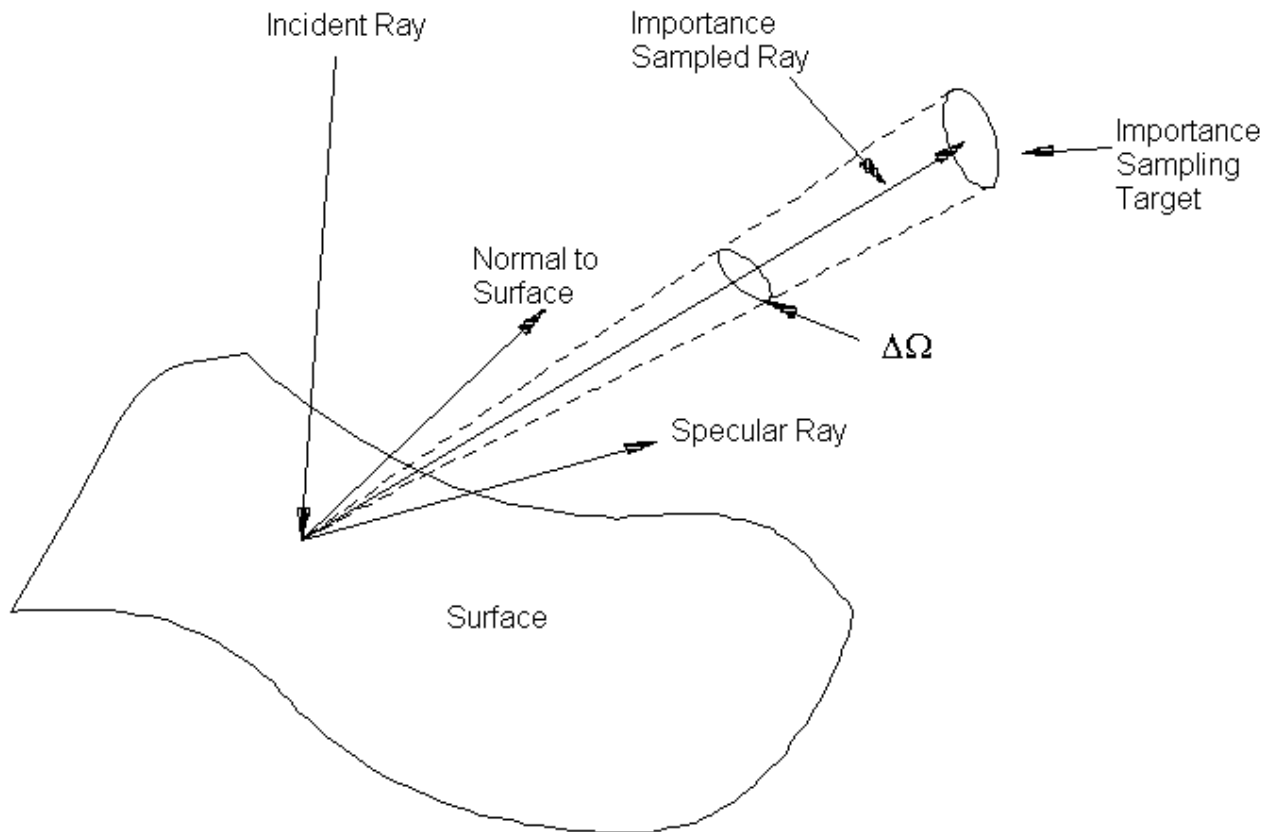
The task manager at right shows the sample system with 100,000 rays per LED with only a few seconds left in the non-splitting raytrace. You can see that there is still quite a bit of memory left and that the non-splitting raytrace mode is also multi-threaded. You can also see the Physical Memory Usage creep up over the simulation.



Scatter and Source Methods to Reduce Noise

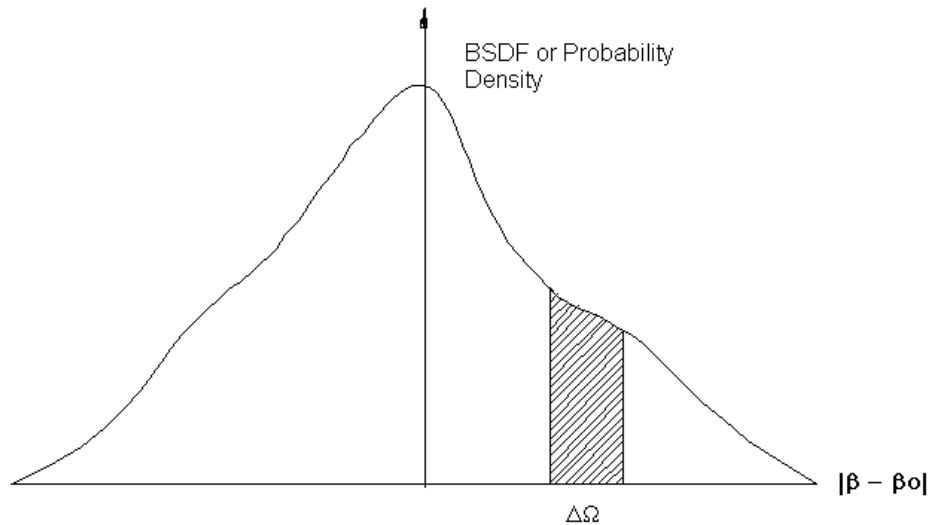
- Importance Sampling is used to improve the sampling of random events without dramatically increasing the number of rays started.
- Uses the scattering distribution function as a probability density to apportion a fraction of the scattered ray flux into a desired direction.
- May be used for emitted, scattered and diffracted rays only, on surface sources, scattering surfaces, diffracting surfaces and bulk scattering objects.
- Apply to object(s) for Bulk Scatter.
- Apply to surface(s) for all others.

Scatter and Source Methods to Reduce Noise - Importance Sampling



Scatter and Source Methods to Reduce Noise

- Importance Sampling for Flux

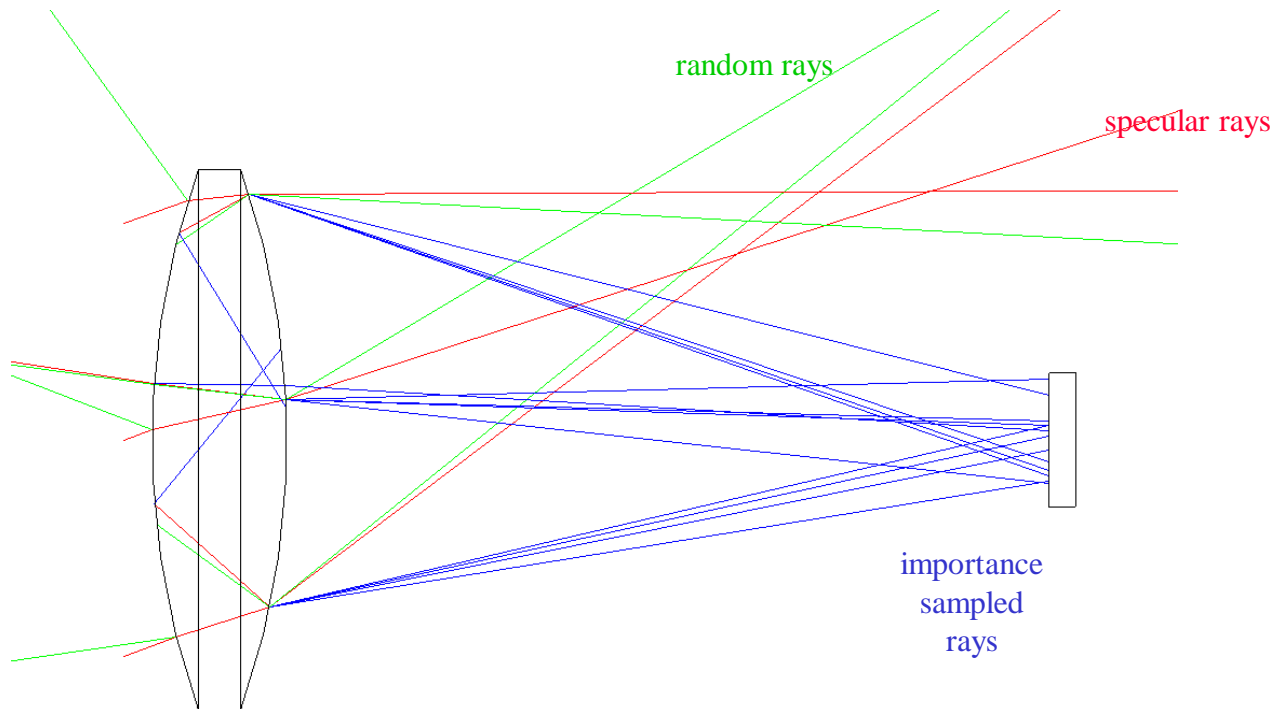


$$\Phi_{imp.samp.} = \Phi_{inc.} \int_{\Delta\Omega} BSDF \cos \theta d\Omega$$

$$\Phi_{random} = \Phi_{inc.} \cdot TS - \Phi_{imp.samp.}$$

$$TS = \int_{hemisphere} BSDF \cos \theta d\Omega$$

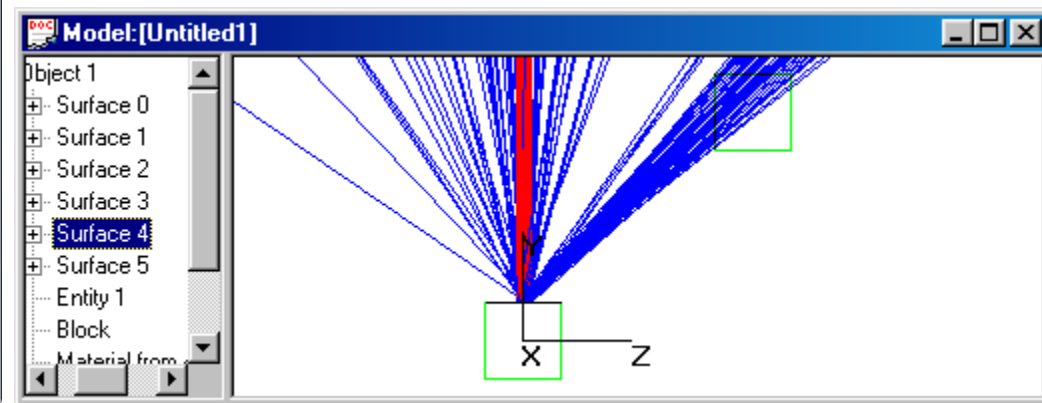
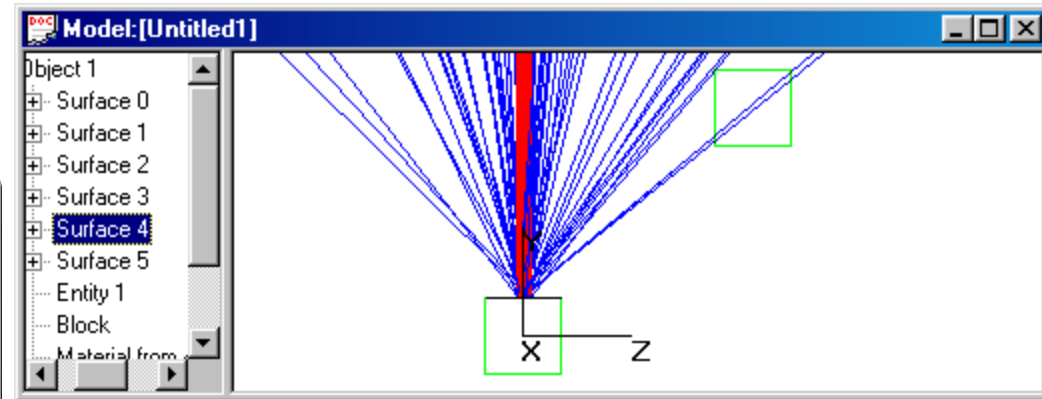
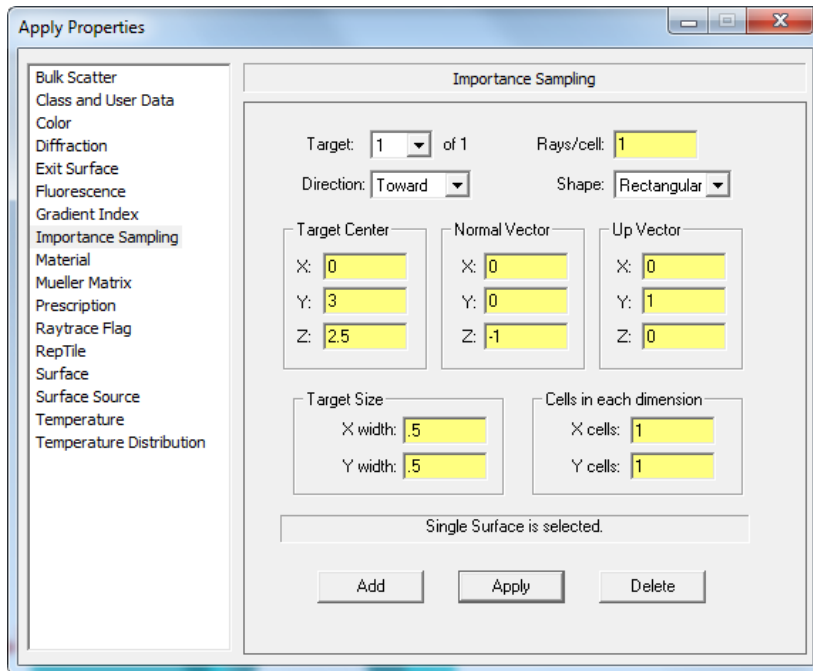
Scatter and Source Methods to Reduce Noise - Importance Sampling Scatter Example



Scatter and Source Methods to Reduce Noise - Importance Sampling Scatter Example

Surface Example

- Flux threshold typically set very low



Scatter and Source Methods to Reduce Noise - Importance Sampling Source Example

