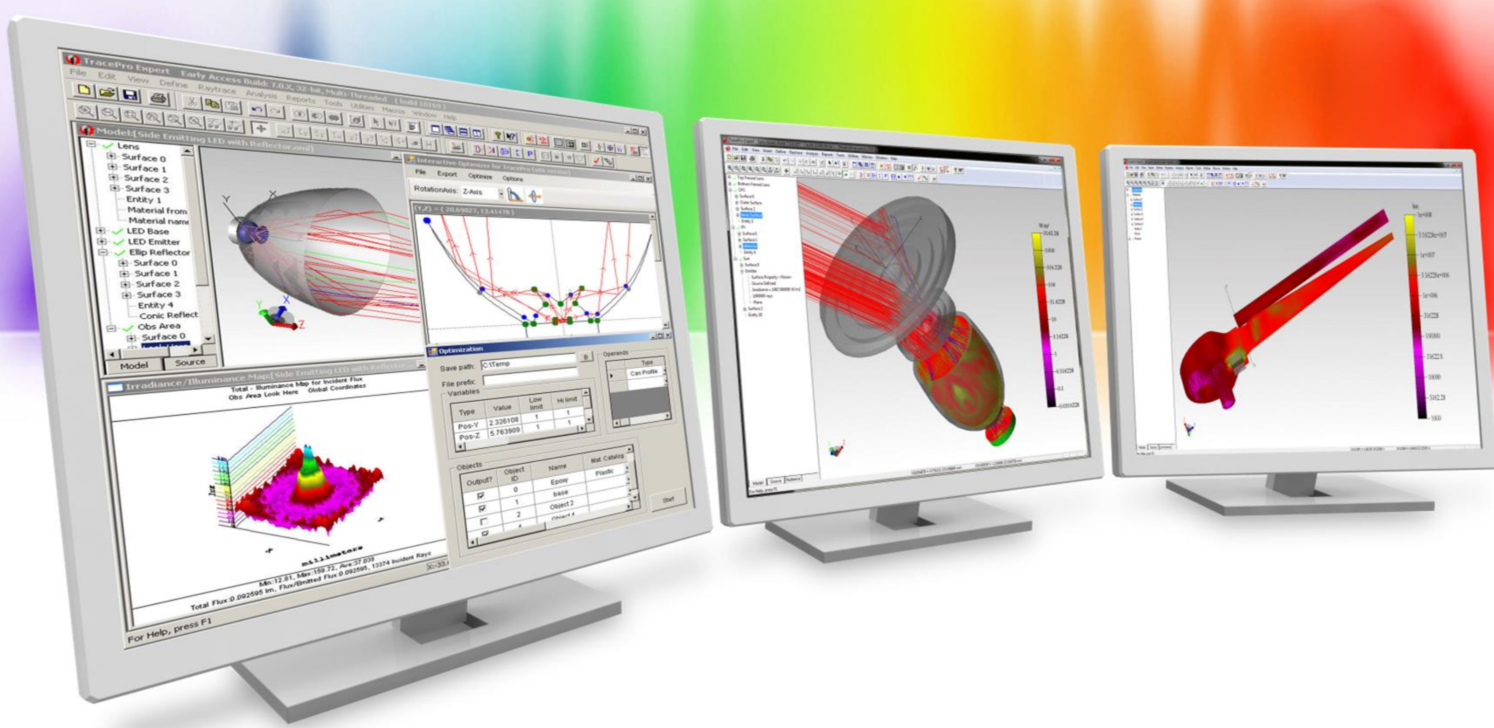
TracePro



HOW TO DO MONTE CARLO TOLERANCING IN TRACEPRO'S 3D INTERACTIVE OPTIMIZER

Presented by :
Lambda Research Corporation
25 Porter Rd.
Littleton, MA 01460
www.lambdares.com

LAMBDA RESEARCH CORPORATION

Agenda

Introduction on how to do Monte Carlo Tolerancing in TracePro

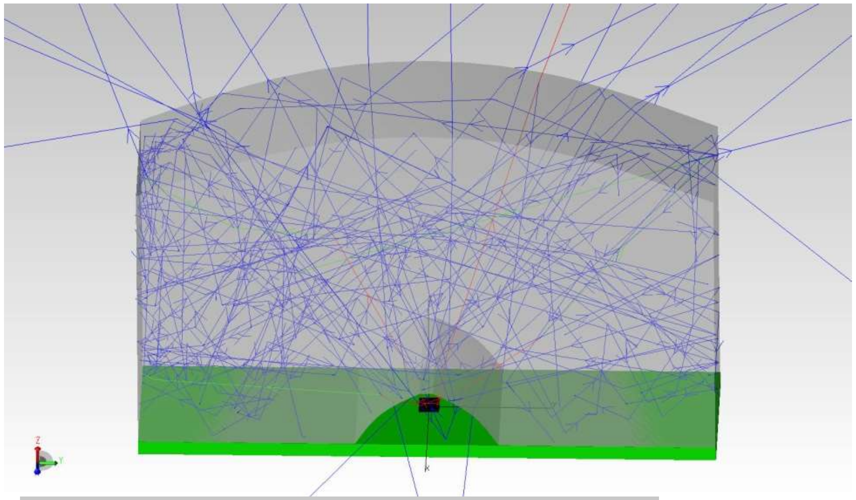
- Introduction to the 3D Interactive Optimizer
- Algorithm behind the Monte Carlo Tolerancing
- How to setup a tolerancing analysis in the 3D interactive optimizer
- How to run a tolerancing analysis in the 3D interactive optimizer
- Three Examples
  - Simple Switch, Simple Switch with LED placement, Luminaire PAR 38 example

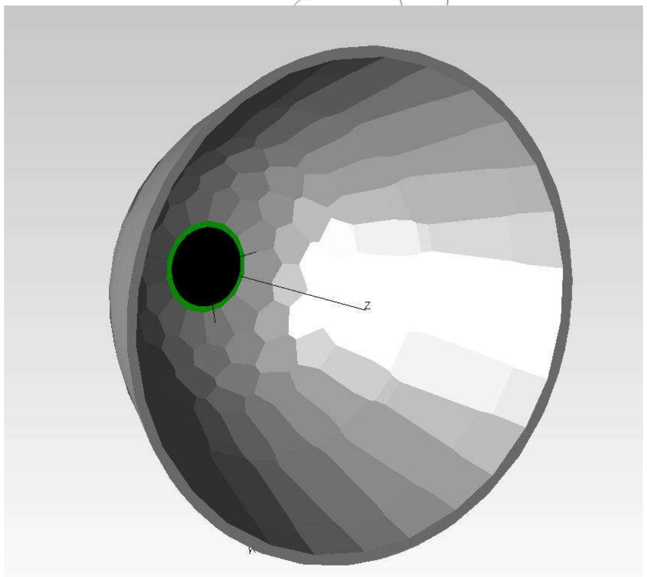
TracePro

Goals for a Monte Carlo Tolerance Analysis

What can be accomplished by doing a Monte Carlo tolerancing analysis?

- Check on changes due to manufacturing processes
- To look at possible installation errors and how they will affect the end result
- To get an idea of how the system will perform after manufacture

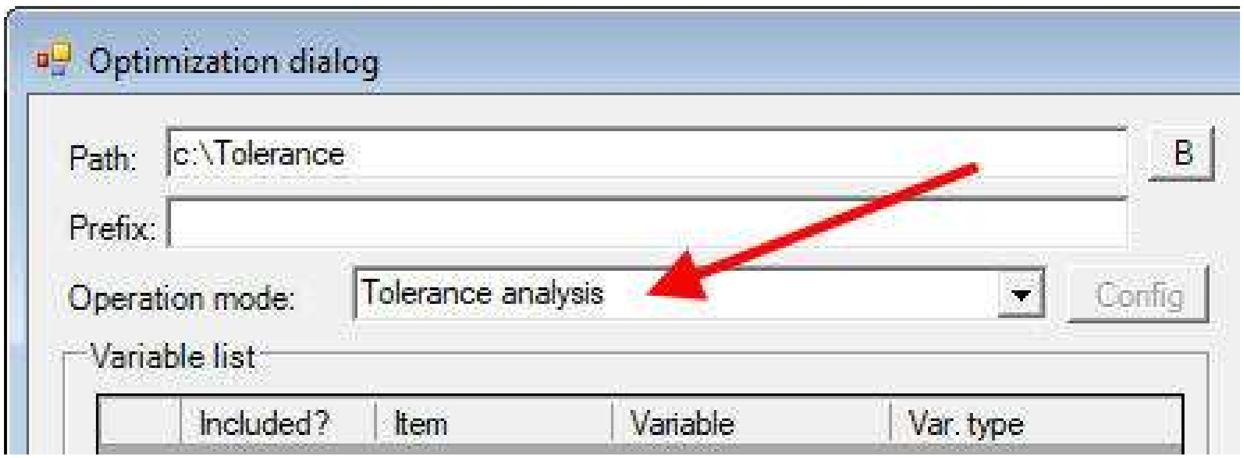




TracePro

Introduction to the Monte Carlo Tolerancing Capability

- The Monte Carlo Tolerancing capability is available in the 3D Interactive Optimizer in the Optimization Dialog by selecting the Operation Mode →Tolerance Analysis



TracePro

Algorithm behind the Monte Carlo Tolerancing Capability

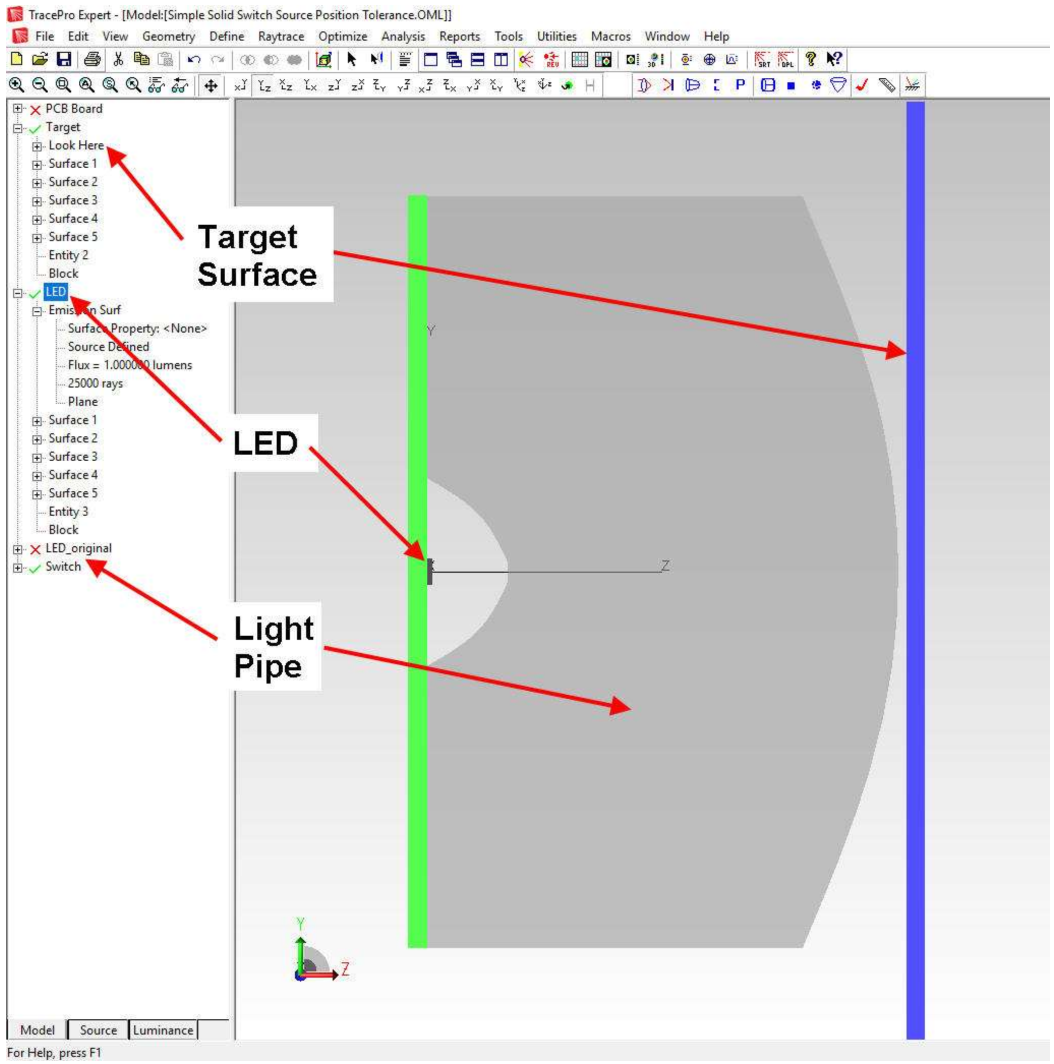
- The code behind the Monte Carlo Tolerancing in TracePro comes from OSLO. It has been reliably used for over 30 years for Monte Carlo tolerancing on all types of systems.

- Monte Carlo analysis uses random numbers to generate a sequence of lenses, light pipes or mirrors, where the maximum magnitude of the perturbations is determined by the current values of the tolerances (lower and higher limits of the user-defined variables). Each random realization constructs the lens, light pipe or optical component by generating random numbers having a prescribed probability density function and then using these random numbers along with the tolerances to perturb the construction parameters of the system. An advantage of Monte Carlo analysis is that all of the construction parameters may be perturbed simultaneously. Analysis of the performance of the resulting systems provides a statistical prediction of the distribution of the final fabricated optical components. Because of the stochastic nature of the process, depending upon the optical component and its sensitivity to its construction parameters, the Monte Carlo analysis may converge slowly to the true value of the performance statistics. Also, since all of the parameters are varied simultaneously, it can be difficult to locate which parameters are the most sensitive. Which is why all iterations are save for post-processing. Monte Carlo analysis can be quite useful in evaluating both optical and illumination systems.

Light Pipe Switch Example

Example 1 –System View and Description of the Light Pipe Switch Assembly

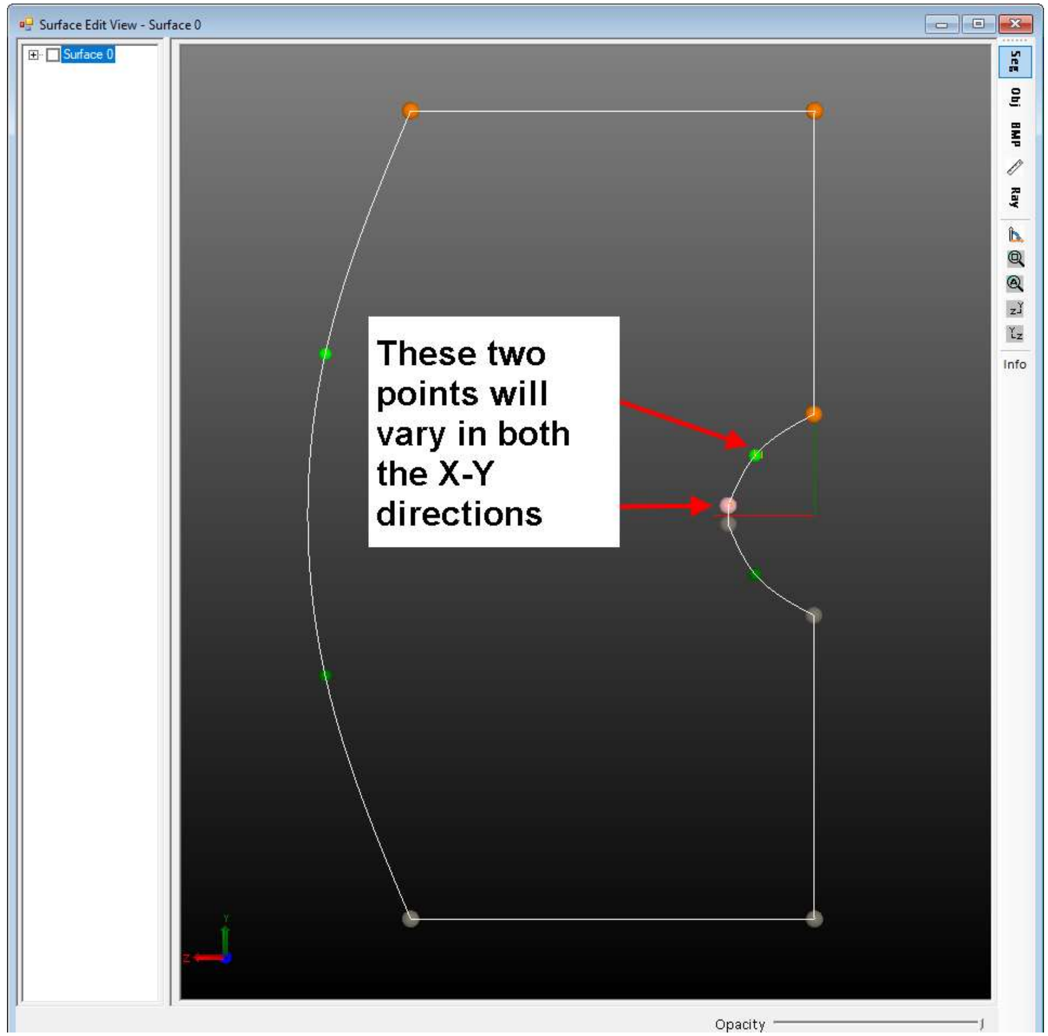
This example uses a simple light pipe and LED. The LED emits 1 lumen of power in a Lambertian manner. The light pipe uses a curved entrance aperture to collimate the light to the target. The function of this light pipe is a simple switch used in a car interior with mask to provide information to the driver of the vehicle.



TracePro

Example 1 –Tolerancing a Light Pipe Switch Assembly

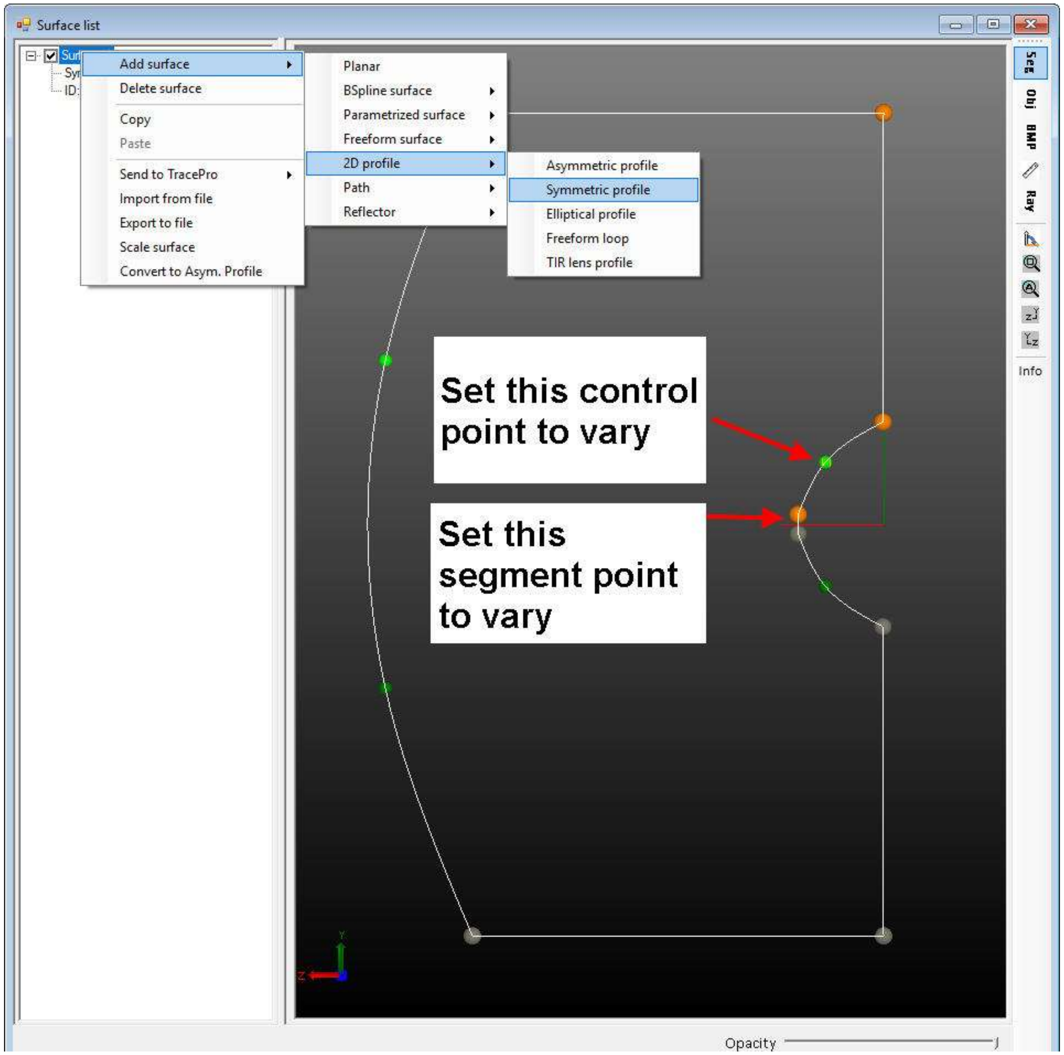
This example walks through the setup process for a Monte Carlo Tolerance on a light pipe entrance aperture to ascertain molding tolerances on the curvature of the light entry area. The changes in the aperture directly affect the amount of flux to the target which we will use as the error function to see how far off the manufacturing process changes the intended system output.



TracePro

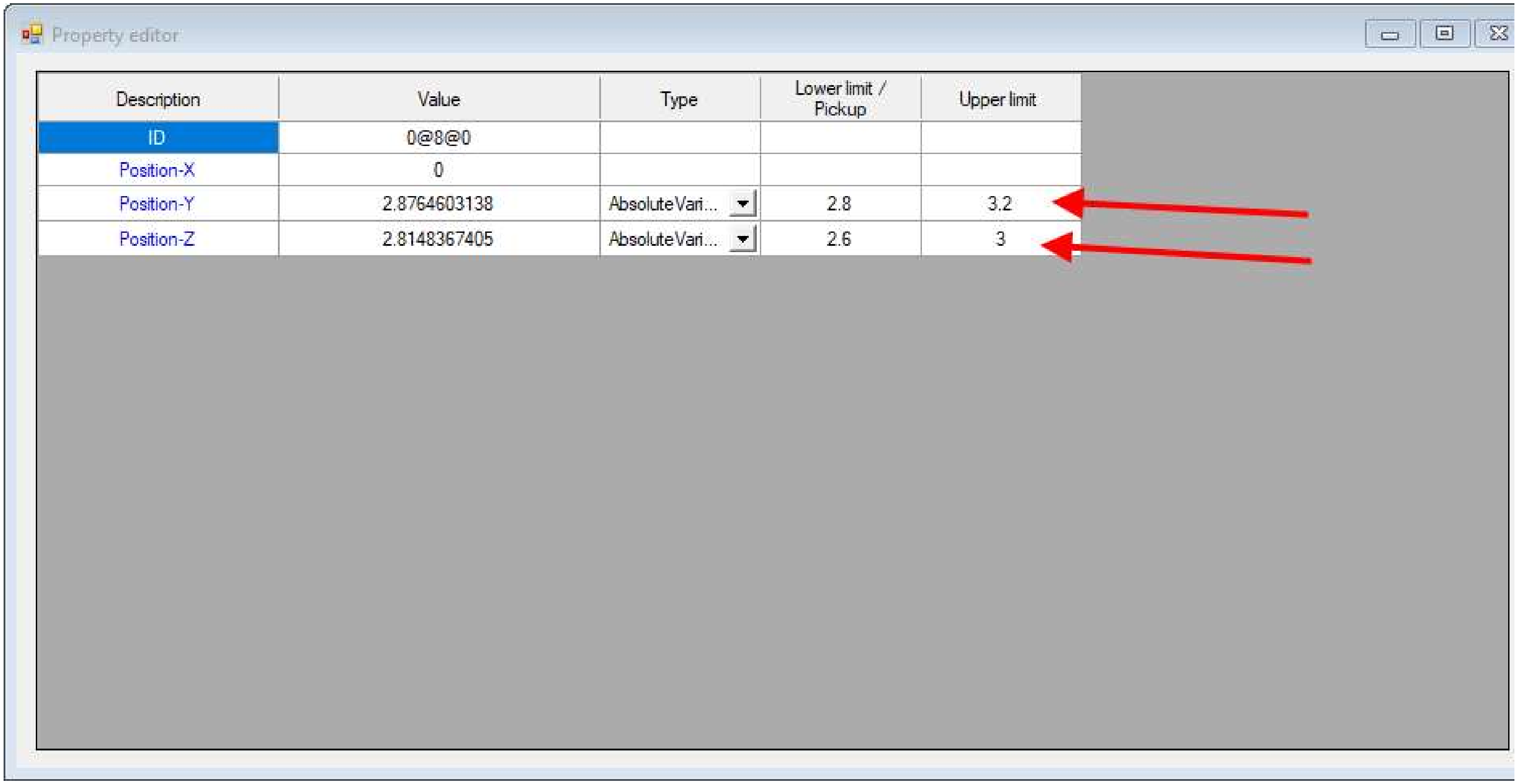
Example 1 – Create the 2D profile of the Light Pipe

Create the 2D profile directly in the 3D interactive optimizer's Surface Edit Viewer. For this example we are using an 2D Profile→Symmetric Profile surface. Make sure that you create multiple control and segment points so that you can set these as variables to pertubate the model to correlate to the changes in the model due to manufacturing processes.



TracePro

Example 1 – Setting the two points as variables

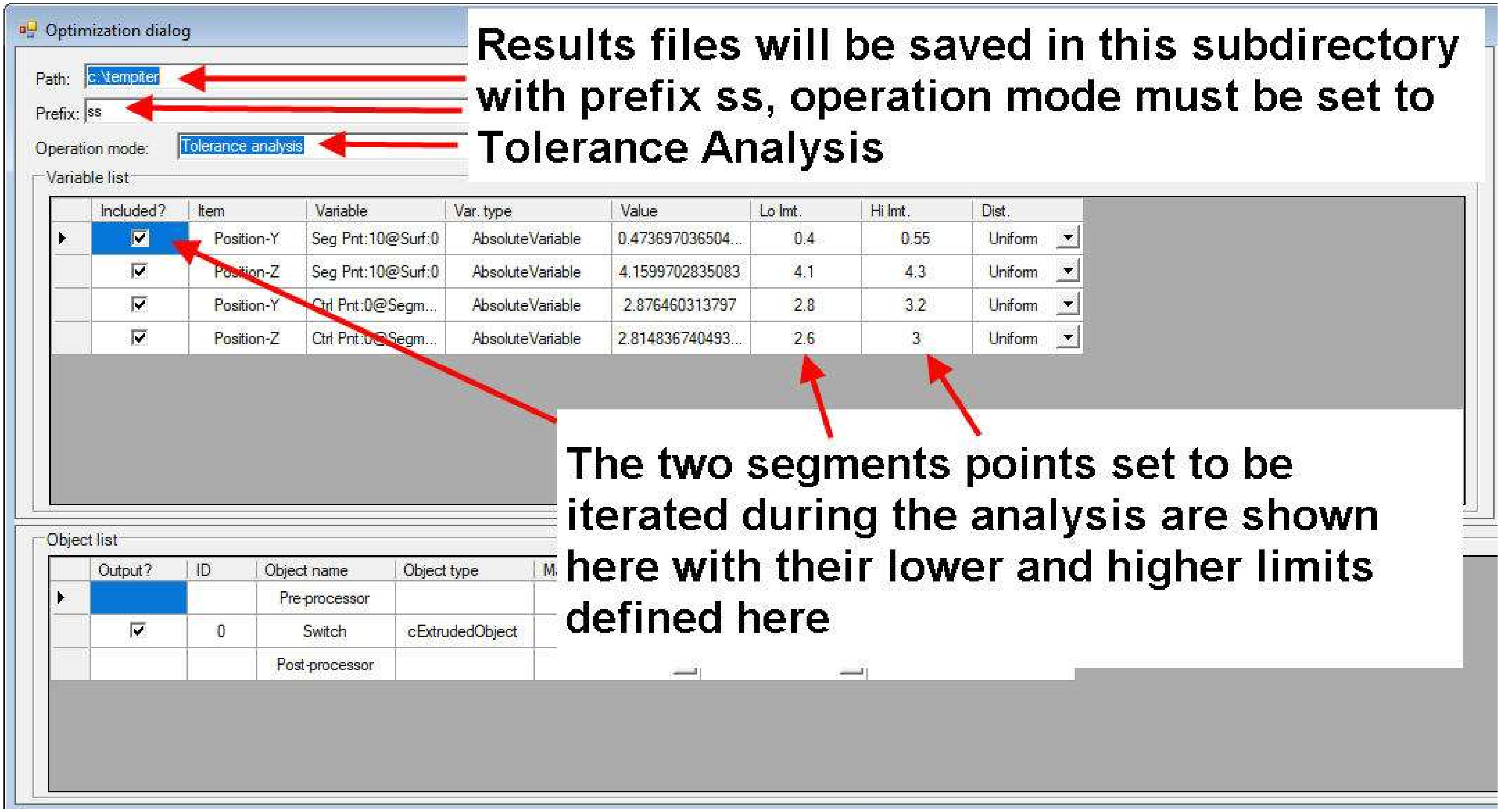


Property editor

Description	Value	Type	Lower limit / Pickup	Upper limit
ID	0@8@0			
Position-X	0			
Position-Y	2.8764603138	AbsoluteVari...	2.8	3.2
Position-Z	2.8148367405	AbsoluteVari...	2.6	3

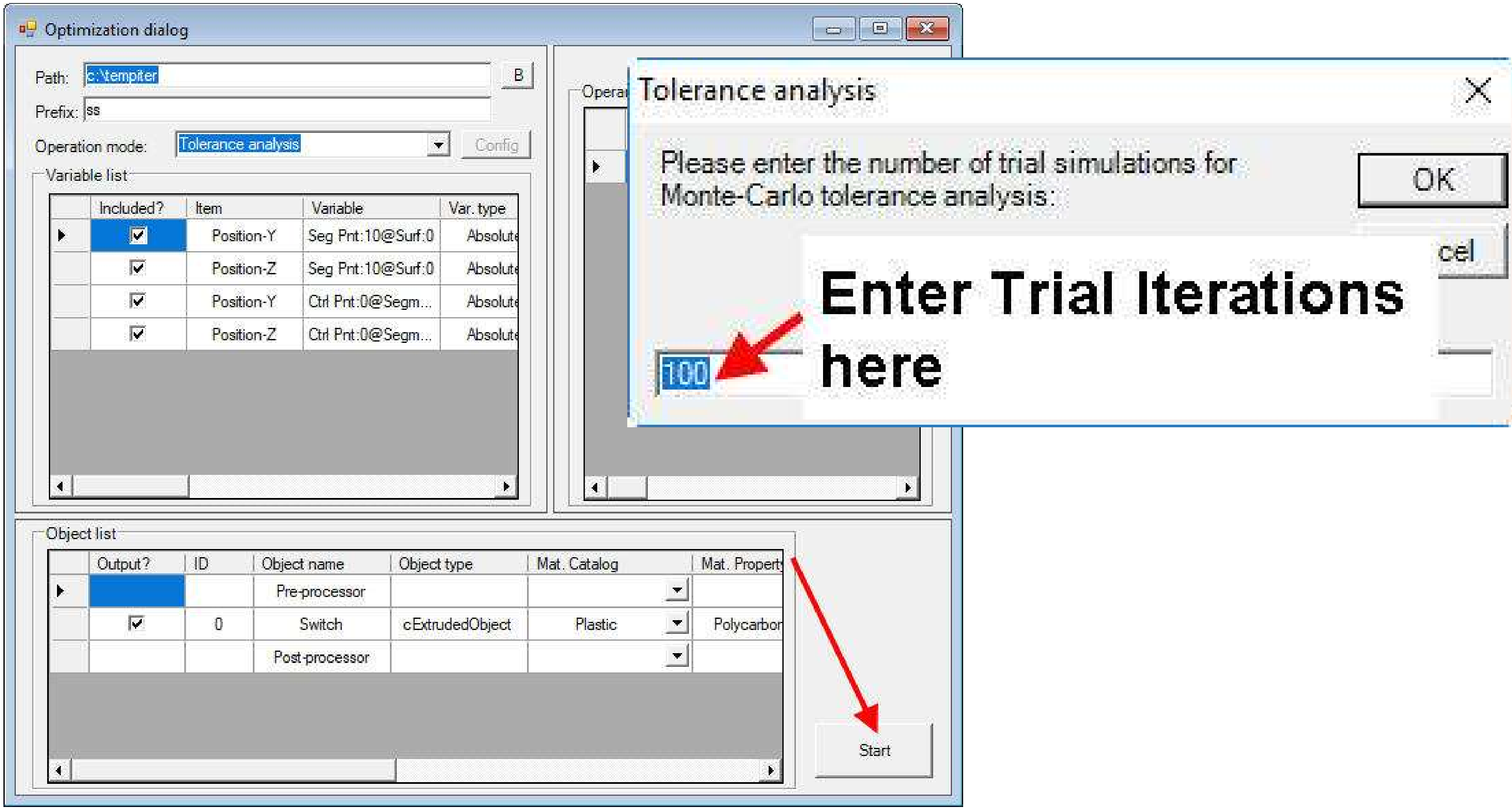
After selecting the segment or control point in the Surface List view, the Property Editor shows the exact value of the point . Changing the type from Specified to Variable allows the user to select either an Absolute or Relative variable type and to put in lower and upper limits for the variable.

Example 1 – Setting parameters in the optimization dialog



Now specify where you want the interim files to be placed, the operation mode must be changed to Tolerance Analysis and your variables should be shown in the Variable list as per the setup in the Property Editor.

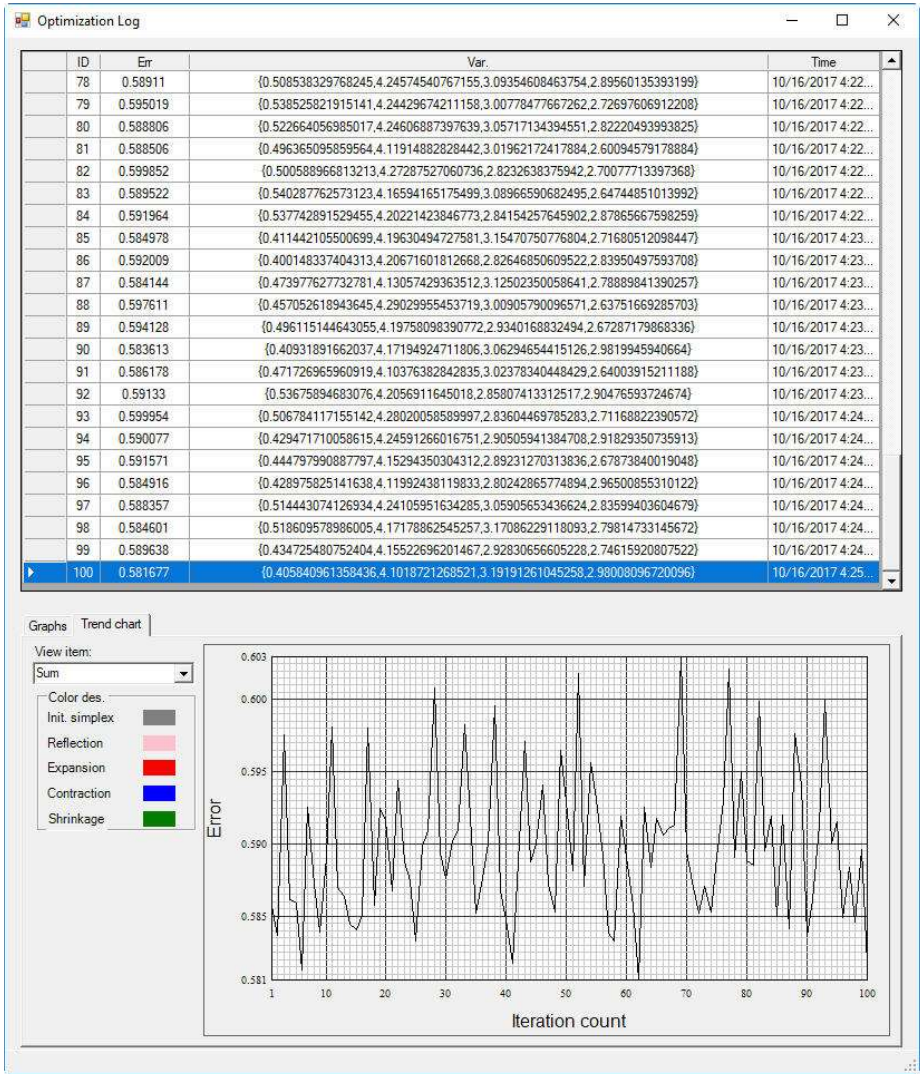
Example 1 – Starting the analysis and setting the number of trials



To start the analysis, click on the Start button in the lower right-hand corner of the optimization dialog. After a few seconds, the Tolerance analysis dialog should appear, you can now enter the number of iterations, 100 for this case.

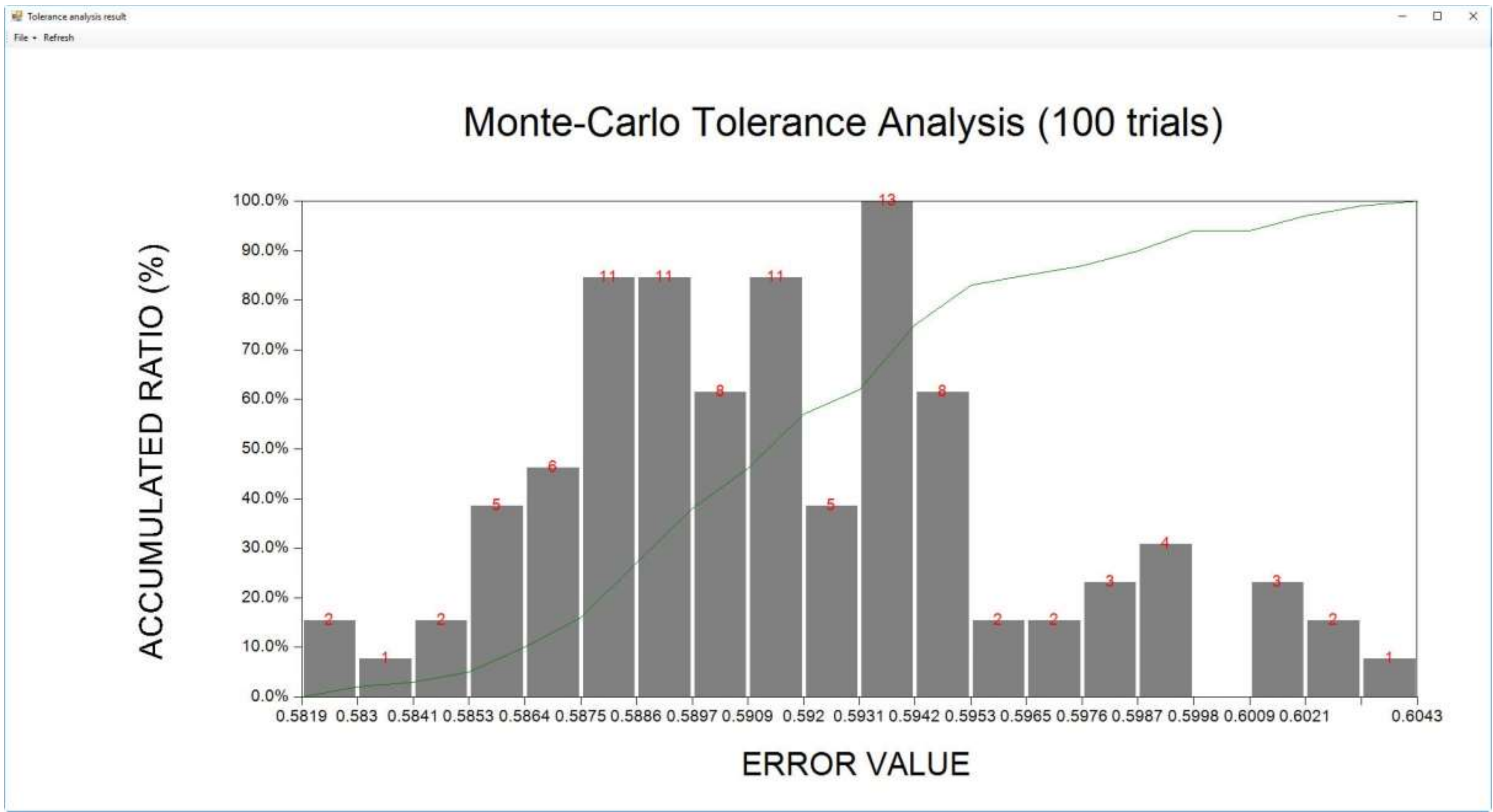
Example 1 -Tolerancing Log of the 100 Monte-Carlo trials

After the completion of the 1st iteration, the Optimization log will appear with the results for each iteration. The log shows the Err function which is 1- the flux on the target for this example, the position of each of the four variables and the amount of time it took for each iteration. It also has a trend chart of the error function. The analysis took a little over 15 minutes.



ID	Err	Var.	Time
78	0.58911	{0.508538329768245,4.24574540767155,3.09354608463754,2.89560135393199}	10/16/2017 4:22...
79	0.595019	{0.538525821915141,4.24429674211158,3.00778477667262,2.72697606912208}	10/16/2017 4:22...
80	0.588806	{0.522664056985017,4.24606887397639,3.05717134394551,2.82220493993825}	10/16/2017 4:22...
81	0.588506	{0.496365095859564,4.11914882828442,3.01962172417884,2.60094579178884}	10/16/2017 4:22...
82	0.599852	{0.500588966813213,4.27287527060736,2.8232638375942,2.70077713397368}	10/16/2017 4:22...
83	0.589522	{0.540287762573123,4.16594165175499,3.08966590682495,2.64744851013992}	10/16/2017 4:22...
84	0.591964	{0.537742891529455,4.20221423846773,2.84154257645902,2.87865667598259}	10/16/2017 4:22...
85	0.584978	{0.411442105500699,4.19630494727581,3.15470750776804,2.71680512098447}	10/16/2017 4:23...
86	0.592009	{0.400148337404313,4.20671601812668,2.82646850609522,2.83950497593708}	10/16/2017 4:23...
87	0.584144	{0.473977627732781,4.13057429363512,3.12502350058641,2.78889841390257}	10/16/2017 4:23...
88	0.597611	{0.457052618943645,4.29029955453719,3.00905790096571,2.63751669285703}	10/16/2017 4:23...
89	0.594128	{0.496115144643055,4.19758098390772,2.9340168832494,2.67287179868336}	10/16/2017 4:23...
90	0.583613	{0.40931891662037,4.17194924711806,3.06294654415126,2.9819945940664}	10/16/2017 4:23...
91	0.586178	{0.471726965960919,4.10376382842835,3.02378340448429,2.64003915211188}	10/16/2017 4:23...
92	0.59133	{0.53675894683076,4.2056911645018,2.85807413312517,2.90476593724674}	10/16/2017 4:23...
93	0.599954	{0.506784117155142,4.28020058589997,2.83604469785283,2.71168822390572}	10/16/2017 4:24...
94	0.590077	{0.429471710058615,4.24591266016751,2.90505941384708,2.91829350735913}	10/16/2017 4:24...
95	0.591571	{0.444797990887797,4.15294350304312,2.89231270313836,2.67873840019048}	10/16/2017 4:24...
96	0.584916	{0.428975825141638,4.11992438119833,2.80242865774894,2.96500855310122}	10/16/2017 4:24...
97	0.588357	{0.514443074126934,4.24105951634285,3.05905653436624,2.83599403604679}	10/16/2017 4:24...
98	0.584601	{0.518609578986005,4.17178862545257,3.17086229118093,2.79814733145672}	10/16/2017 4:24...
99	0.589638	{0.434725480752404,4.15522696201467,2.92830656605228,2.74615920807522}	10/16/2017 4:24...
100	0.581677	{0.405840961358436,4.1018721268521,3.19191261045258,2.98008096720096}	10/16/2017 4:25...

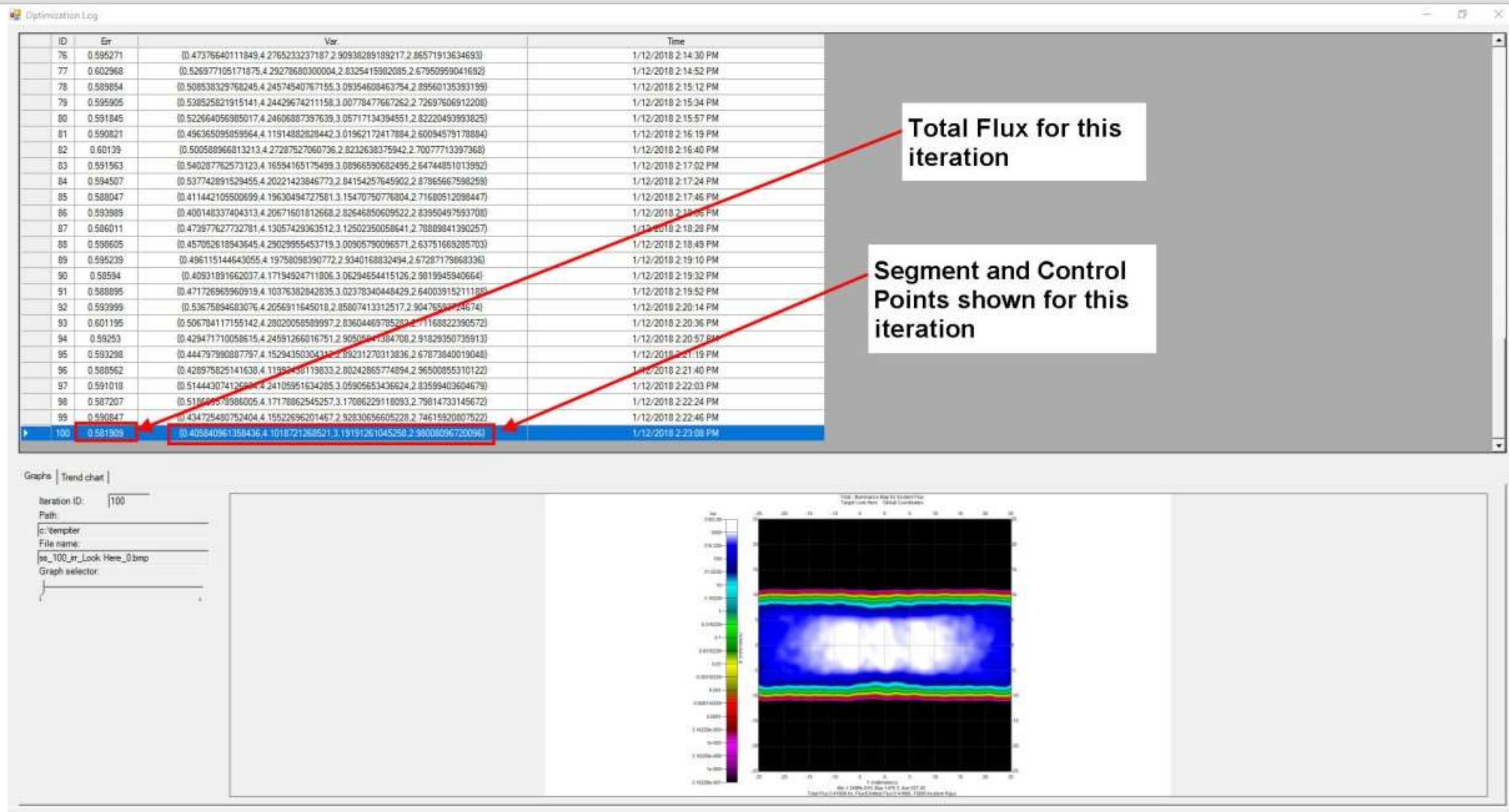
Results of Monte Carlo Tolerancing on Switch 100 trials



After the analysis finishes the 100 trials, a graphical result will be shown. This figure shows the error value versus accumulated ratio with the number of trials matching this result shown at the top in red. The majority of results for this analysis are in the .5853 to .5953 range with 75 results.

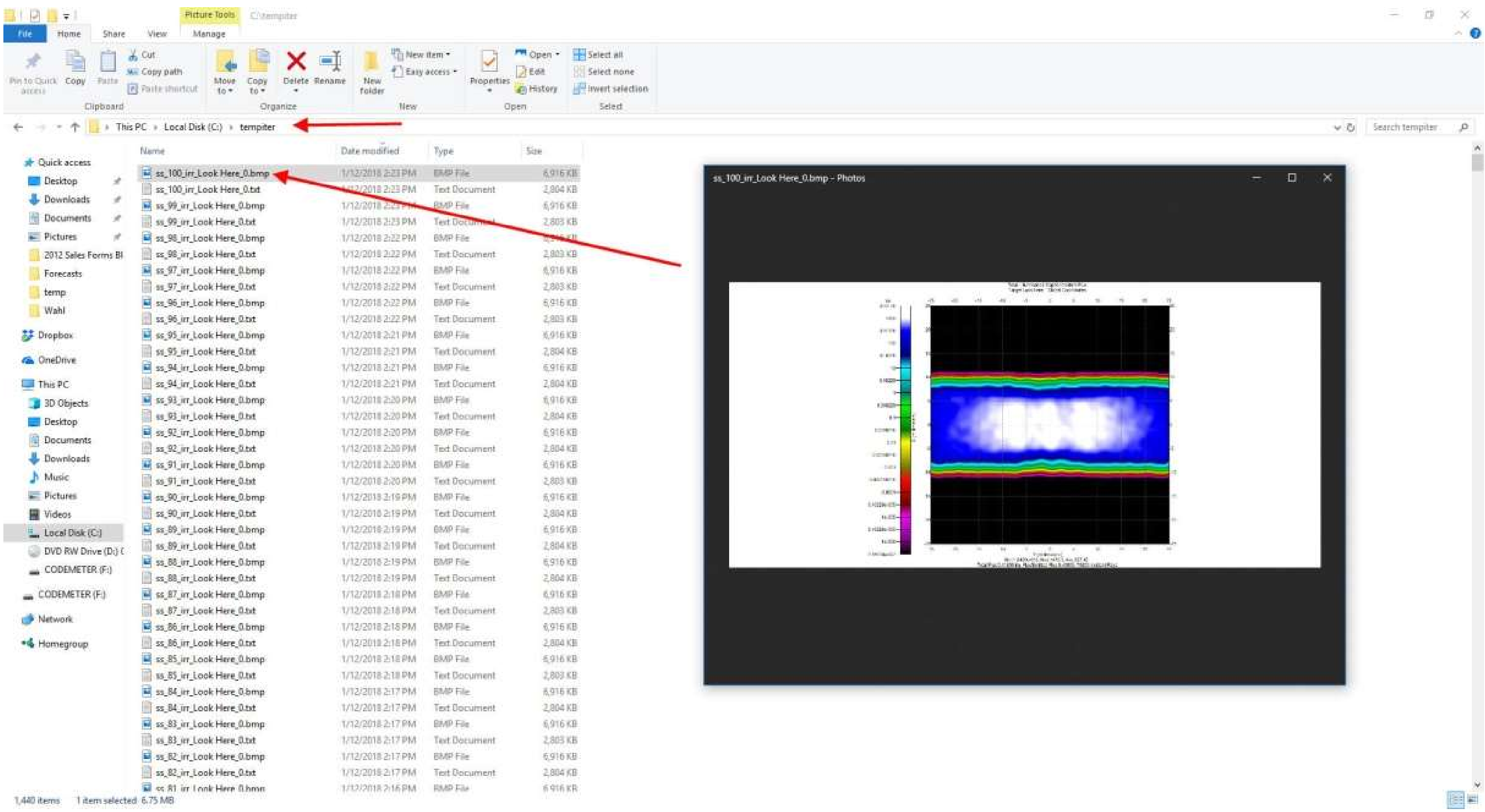
TracePro

Example 1 - Results for each Monte-Carlo iteration



The results for iteration #100 is shown above in the optimization log. The error function result is shown in the 2nd column and the Variable positions in the 3rd column. The Graph option is selected for the 100th iteration 1 and the corresponding illuminance map in pseudo-color output is shown at bottom right.

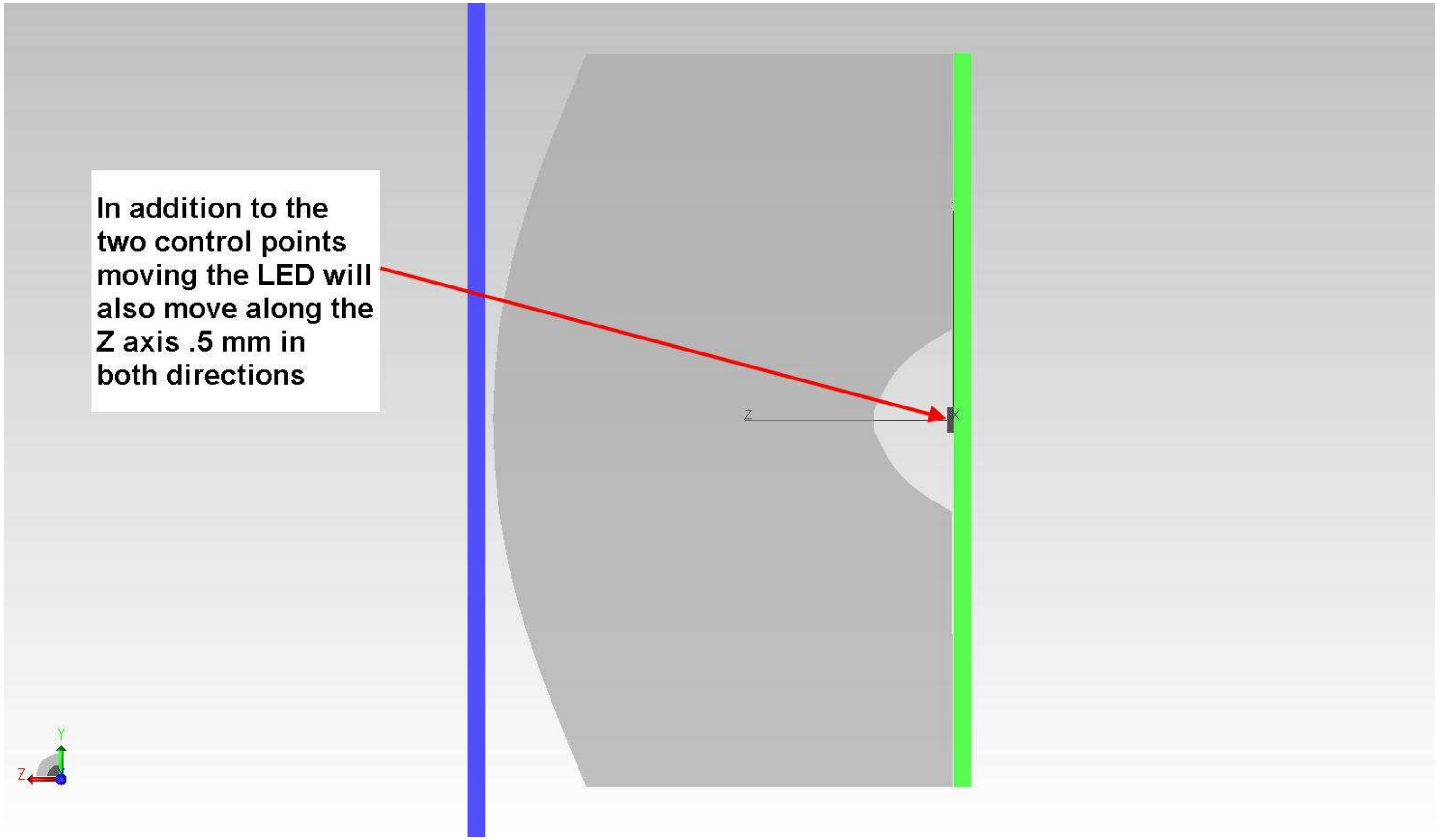
Examples 1 - Results for all 100 iterations are kept



All the results for each of the 100 iterations are saved as designated by the Path field in the optimization dialog for post-process viewing.

Light Pipe Switch LED Position Example

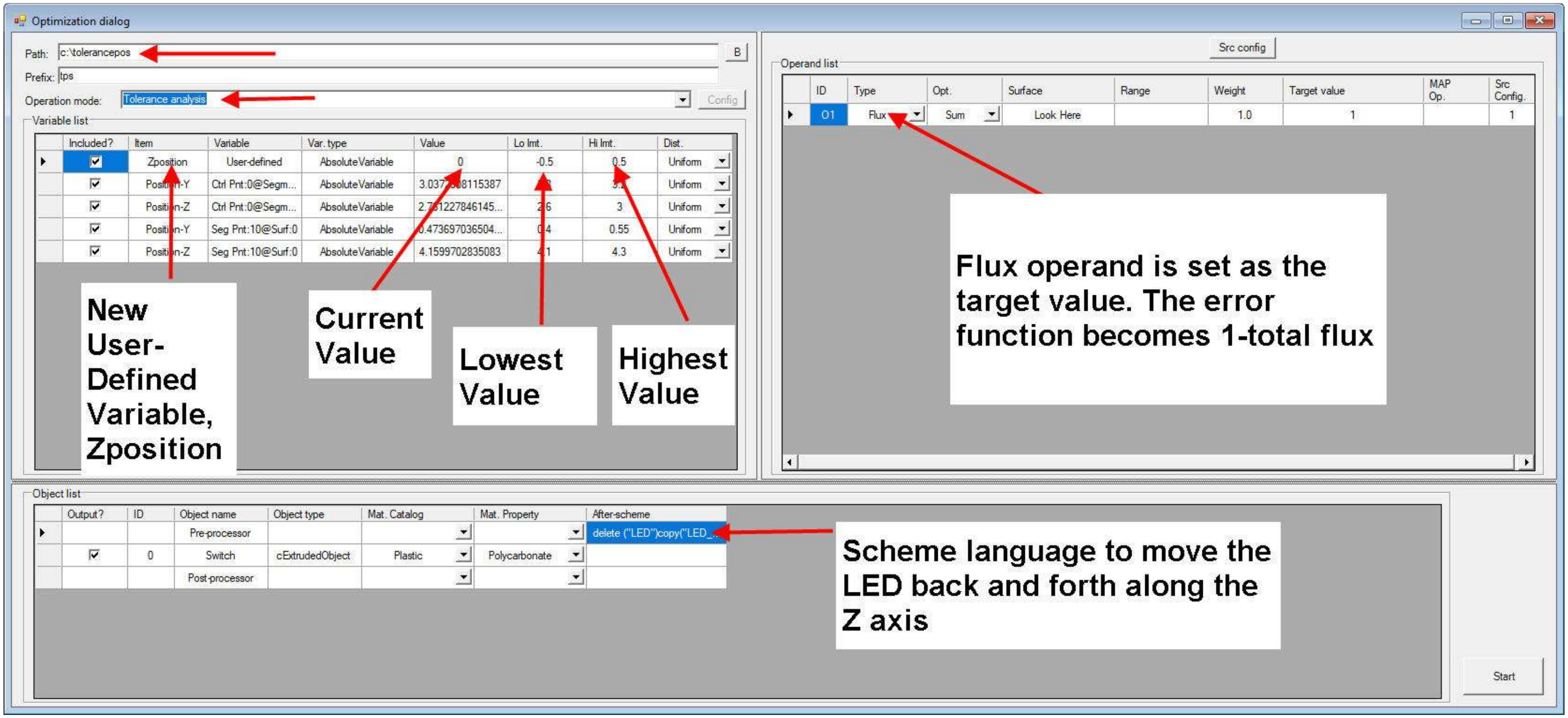
Example 2 – Light Pipe Tolerancing with LED Position



Will we add the LED position to the list of variables for the switch and allow it to move back and forth .5mm in Z position for this tolerancing analysis. This will mean that we now have 5 variables for the tolerance analysis.

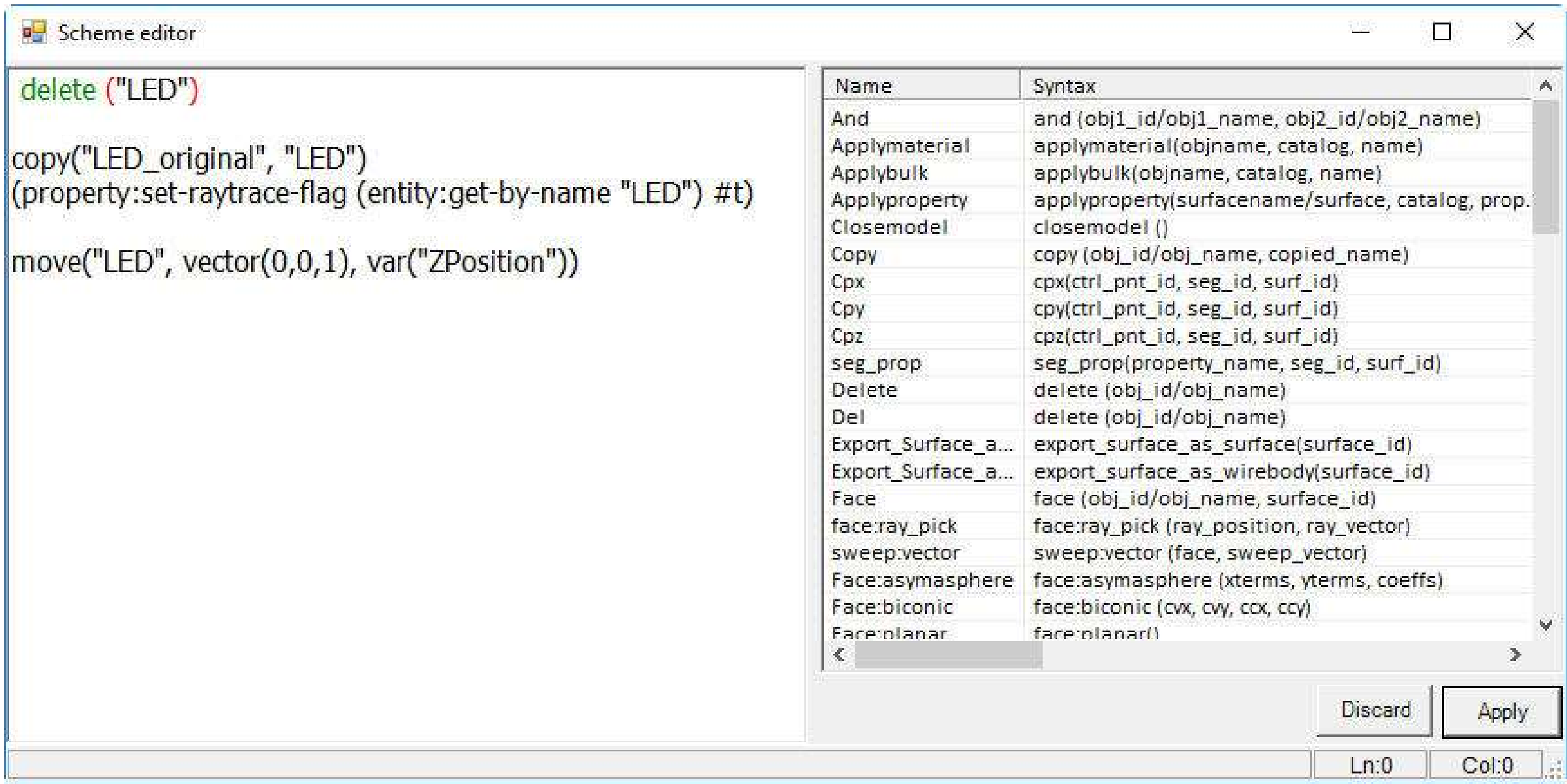
TracePro

Example 2 – Add User Defined Variable and Pre-Processor Scheme macro to position the LED



In addition to the 4 variables we used in Example 1, we will add a user-defined variable, Zposition. Zposition has a lower limit of -.5mm to a high limit of .5mm in correlation to the front of the PCB board. We will set a flux operand on the target surface and write a scheme macro to be pre-processed before execution of the system analysis.

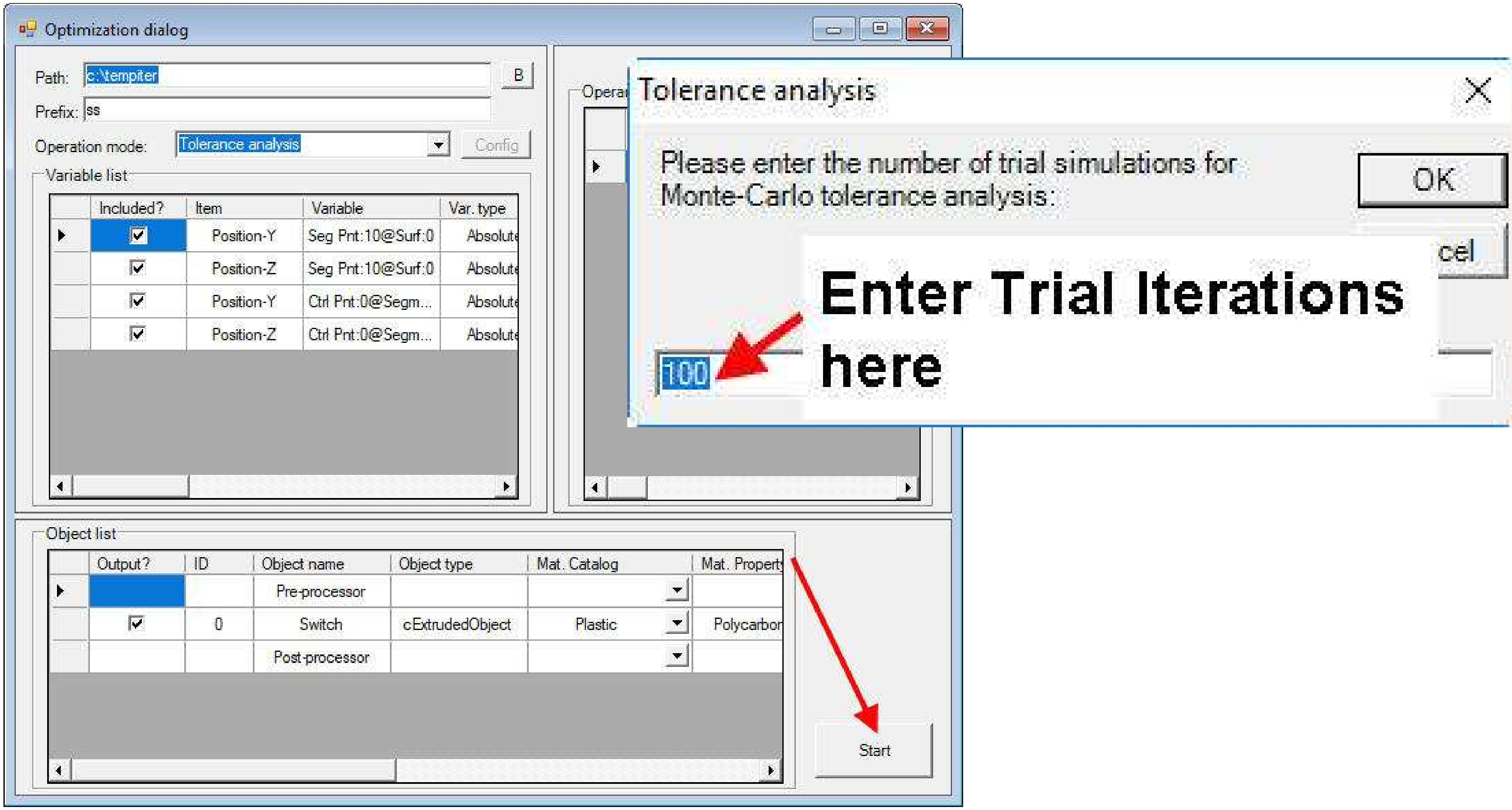
Example 2 – Pre-processed Scheme macro



The scheme macro for this analysis is simple. First, we delete any existing LED object created during any subsequent iteration. Next, we copy the original LED named LED_original and to the object named LED. Next, we turn the raytrace flag on since the LED_original object is set to the condition don't raytrace. Finally we move the LED by the user-defined variable Zposition.

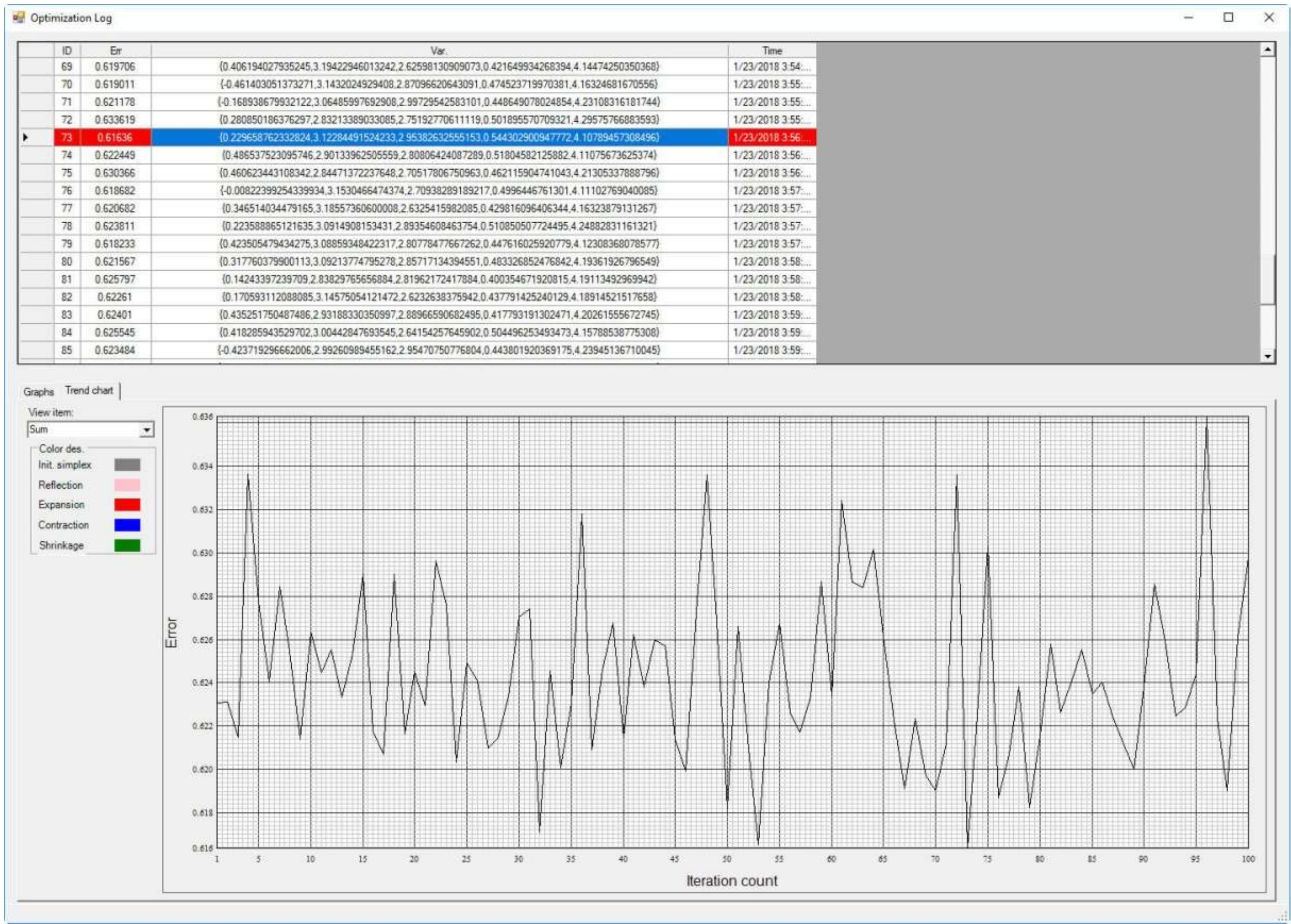
TracePro

Example 2 – Starting the analysis and setting the number of trials



To start the analysis, click on the Start button in the lower right-hand corner of the optimization dialog. After a few seconds, the Tolerance analysis dialog should appear, you can now enter the number of iterations, 100 for this case. TracePro

Example 2 -Tolerancing Log of the 100 Monte-Carlo trials



Optimization Log

ID	Err	Var.	Time
69	0.619706	{0.406194027935245,3.19422946013242,2.62598130909073,0.421649934268394,4.14474250350368}	1/23/2018 3:54:...
70	0.619011	{-0.461403051373271,3.1432024929408,2.87096620643091,0.474523719970381,4.16324681670556}	1/23/2018 3:55:...
71	0.621178	{-0.168938679932122,3.06485997692908,2.99729542583101,0.448649078024854,4.23108316181744}	1/23/2018 3:55:...
72	0.633619	{0.280850186376297,2.83213389033085,2.75192770611119,0.501895570709321,4.29575766883593}	1/23/2018 3:55:...
73	0.61636	{0.229658762332824,3.12284491524233,2.95382632555153,0.544302900947772,4.10789457308496}	1/23/2018 3:56:...
74	0.622449	{0.486537523095746,2.90133962505559,2.80806424087289,0.51804582125882,4.11075673625374}	1/23/2018 3:56:...
75	0.630366	{0.460623443108342,2.84471372237648,2.70517806750963,0.462115904741043,4.21305337888796}	1/23/2018 3:56:...
76	0.618682	{-0.00822399254339934,3.1530466474374,2.70938289189217,0.4996446761301,4.11102769040085}	1/23/2018 3:57:...
77	0.620682	{0.346514034479165,3.18557360600008,2.6325415982085,0.429816096406344,4.16323879131267}	1/23/2018 3:57:...
78	0.623811	{0.223588865121635,3.0914908153431,2.89354608463754,0.510850507724495,4.24882831161321}	1/23/2018 3:57:...
79	0.618233	{0.423505479434275,3.08859348422317,2.80778477667262,0.447616025920779,4.12308368078577}	1/23/2018 3:57:...
80	0.621567	{0.317760379900113,3.09213774795278,2.85717134394551,0.483326852476842,4.19361926796549}	1/23/2018 3:58:...
81	0.625797	{0.14243397239709,2.83829765656884,2.81962172417884,0.400354671920815,4.19113492969942}	1/23/2018 3:58:...
82	0.62261	{0.170593112088085,3.14575054121472,2.6232638375942,0.437791425240129,4.18914521517658}	1/23/2018 3:58:...
83	0.62401	{0.435251750487486,2.93188330350997,2.88966590682495,0.417793191302471,4.20261555672745}	1/23/2018 3:59:...
84	0.625545	{0.418285943529702,3.00442847693545,2.64154257645902,0.504496253493473,4.15788538775308}	1/23/2018 3:59:...
85	0.623484	{-0.423719296662006,2.99260989455162,2.95470750776804,0.443801920369175,4.23945136710045}	1/23/2018 3:59:...

During the 1st iteration, the Optimization log will appear with the results for each iteration as they occur. The log above shows the Err function, the position of the five variables in order of the entry in the variable definition and the amount of time it took for each iteration.

TracePro

General Understanding of how to calculate the Error Function

The calculation of the error function in the Interactive Optimizer is different for each operand type and system:

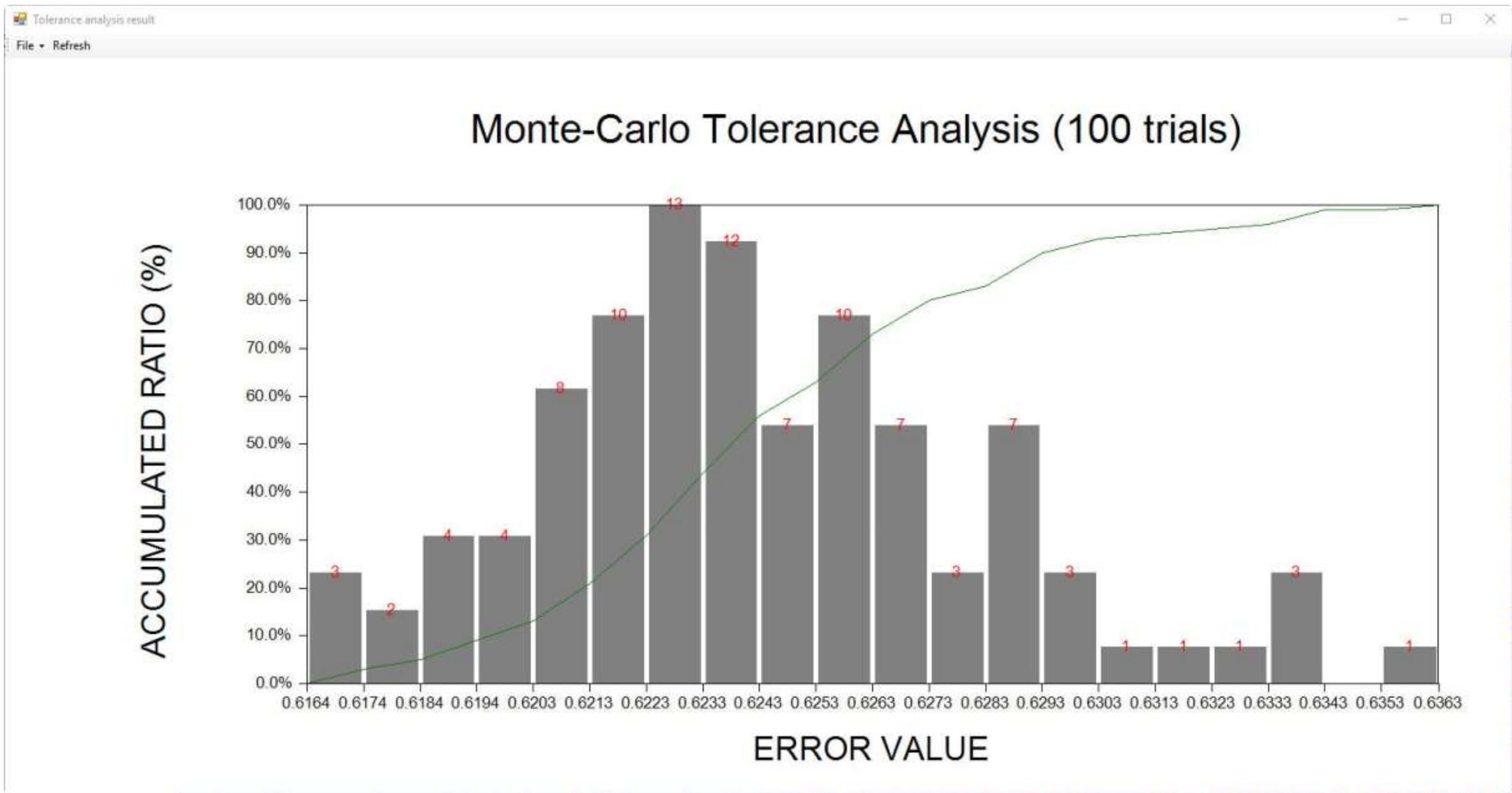
- For the flux operand, which we are using in this example, the error value will vary across a range less than .02. But this is not always the case. For example, if the target is 1 lumen and the simulated flux is .4 lumens, then the error for this iteration will be the absolute value of (1-.4) or .6 If you combine the flux error with the profile similarity error, it is possible for the profile similarity error to be nearly invisible due to the comparatively small value. The solutions to this problem are:
  - Set the target value more reasonable. For example, if the simulated flux is around the range of 2W-5W, the proper value for the target is 5W or 6W, not 10W or 100W.
  - Adjusting the weights for each item. Before starting optimization, we suggest using right clicking and choosing “Create model & run cmd & Raytrace” in the after-scheme cell to perform the simulation once. The value of each item will be shown in the optimization log. This is a good way for you to check the quantities of each error and adjust the weights of each error item to make them “comparable”.

TracePro

Understanding the Error Function

- For the Irradiance and Candela profiles, the calculated error will fall into a range between 0.0 and 2.0 depending on how "similar" the simulated profile is to the operand profile. The method used is the Pearson correlation coefficient.
- For colorimetric operands, such as CIE xy and CIE u'v', the error is the distance between the target and the simulated color on the color gamut.
- For the uniformity operand, the error will be determined by the formula defined in the uniformity target definer.

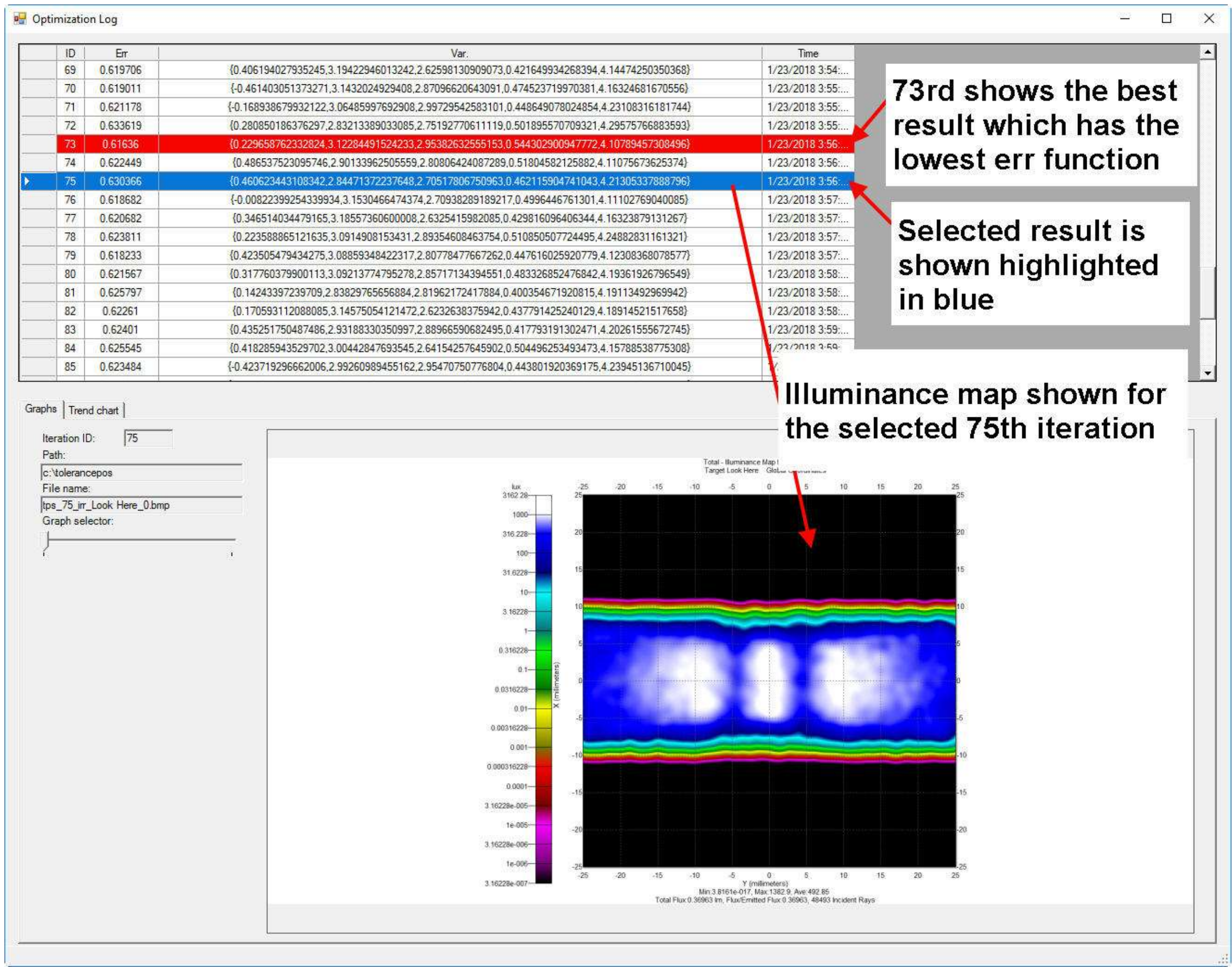
Example 2 - Results of Monte Carlo Tolerancing on Switch 100 trials



After the tolerance analysis is finished the 100 trials, a graphical result will be shown. This figure shows the error value versus accumulated ratio with the number of trials matching this error value result shown at the top of the bar in red. The majority of results for this analysis have an error value in the .62 to .63 range, the lowest error value denotes highest flux on the target.

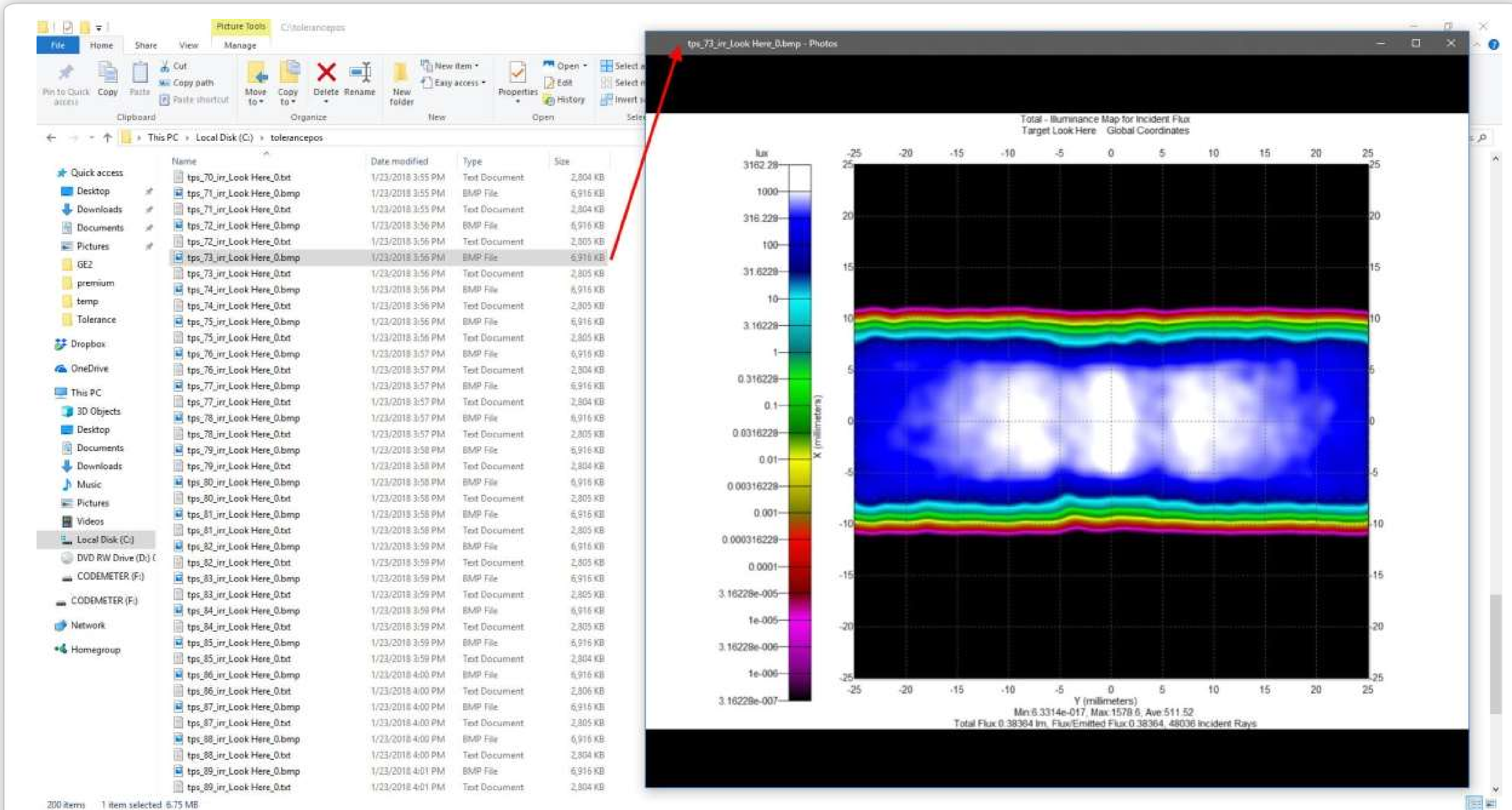
Example 2 - Results for each Monte-Carlo iteration

The results for the selected 75th iteration is shown in the optimization log. The Error Function is shown in the 2nd column, variable positions in the 3rd column. The Graph option is selected so that the illuminance map is available to be shown during the tolerancing operation. In the illuminance map, the total flux for this iteration was .36953 lumens.



TracePro

Example 2 - Results for iteration 73 is shown



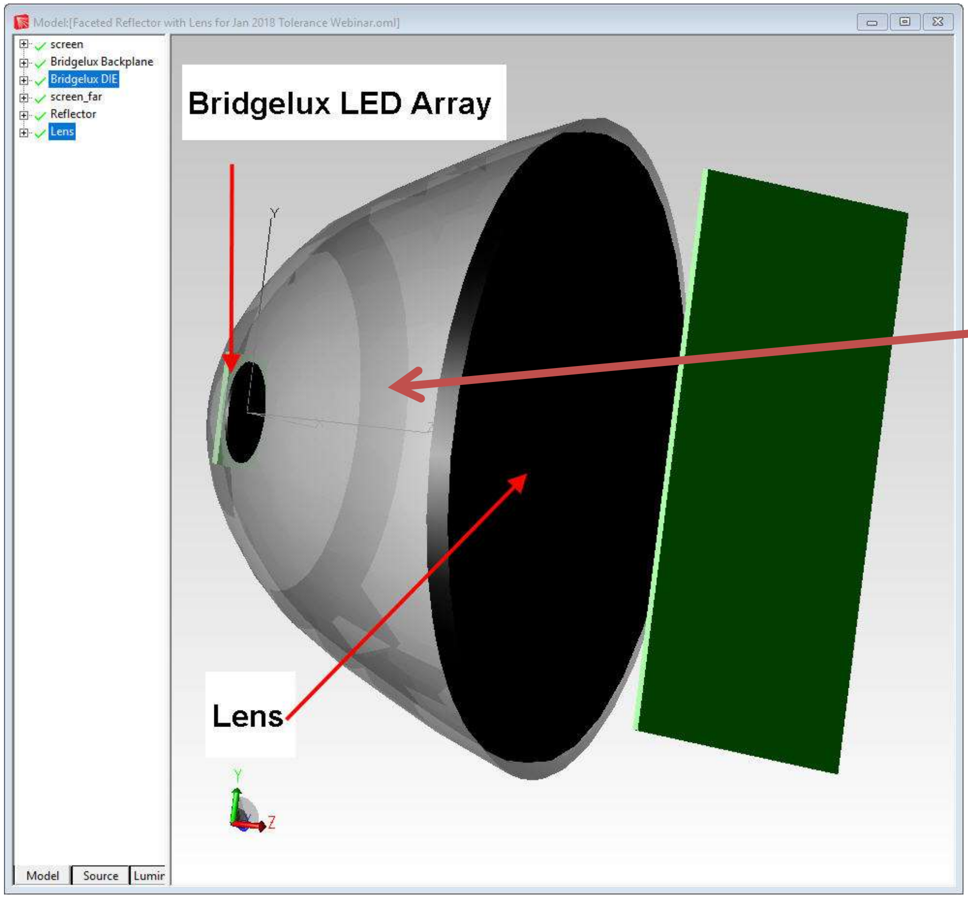
The results for the best iteration, which is the 73rd, is shown which has the largest flux .38353 Lumens. This has a Z position of .229mm above the PCB board shown as the first variable in the variable list, column 3.

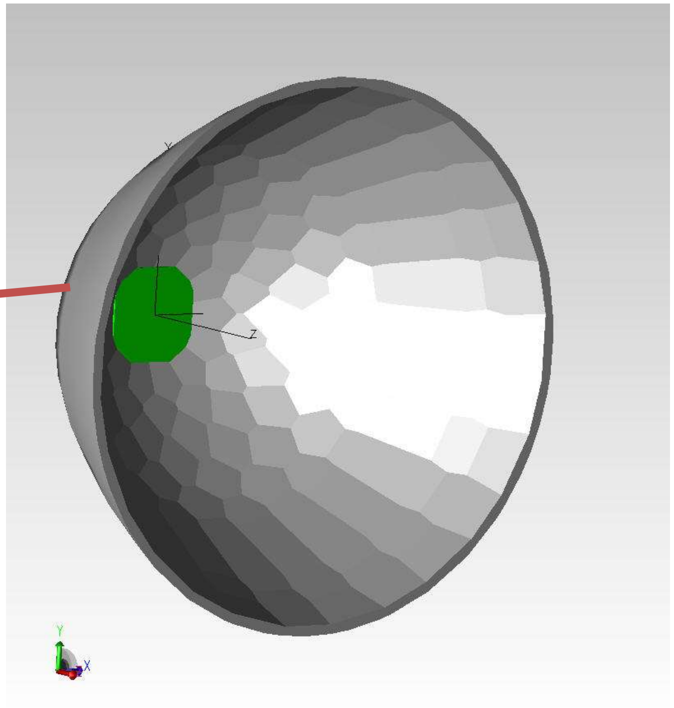
TracePro

Example 3 - Luminaire and Lens Shape Example

TracePro

Example 3 – Faceted Luminaire Reflector and Lens



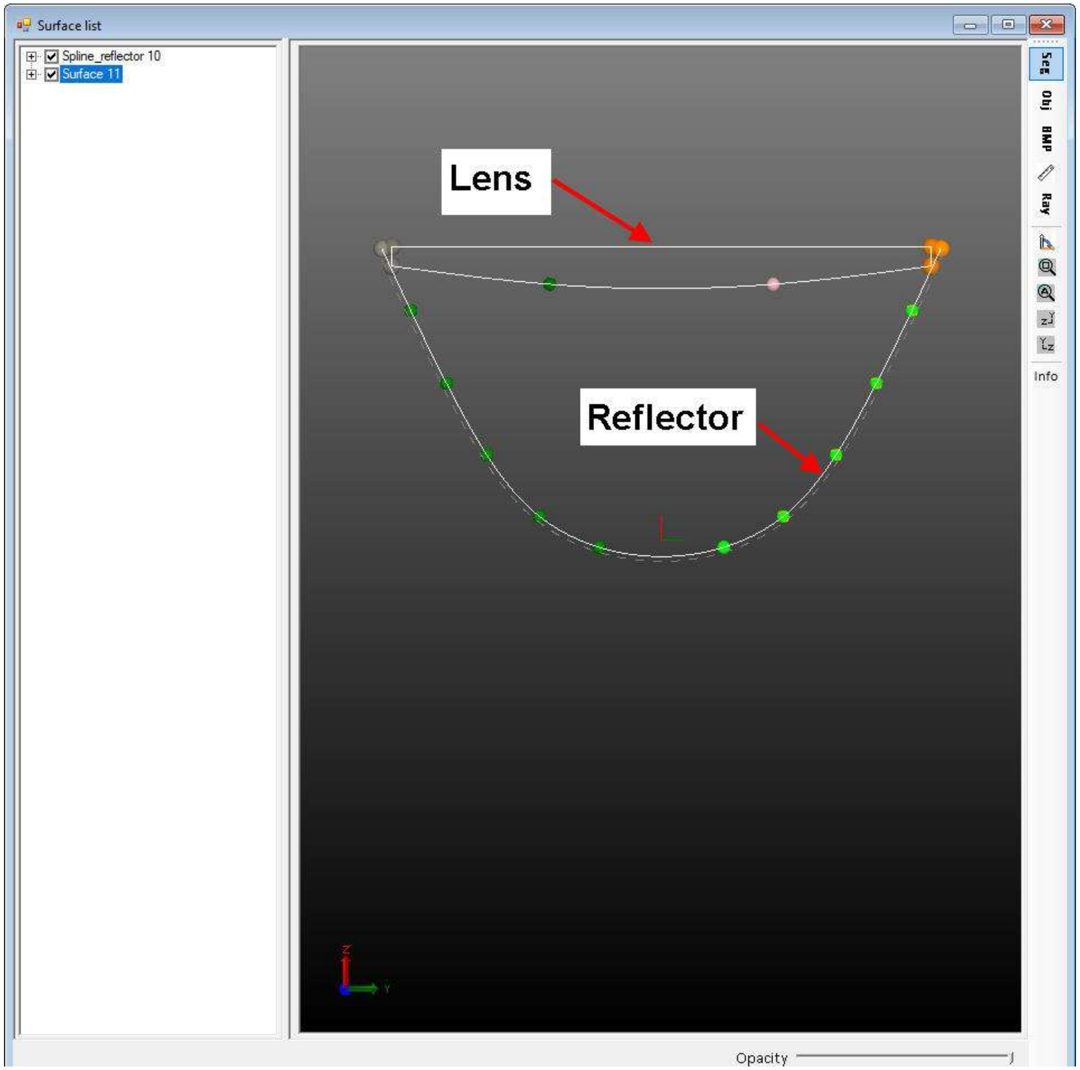


This example demonstrates how to setup a Monte Carlo Tolerance on a lens and reflector duo for a luminaire. The reflector is hexagonally faceted with 15 steps and 15 rings with an overall diameter of two inches. A plano-convex lens is placed at the exit aperture for the reflector.

TracePro

Example 3 – Faceted Luminaire Reflector and Lens

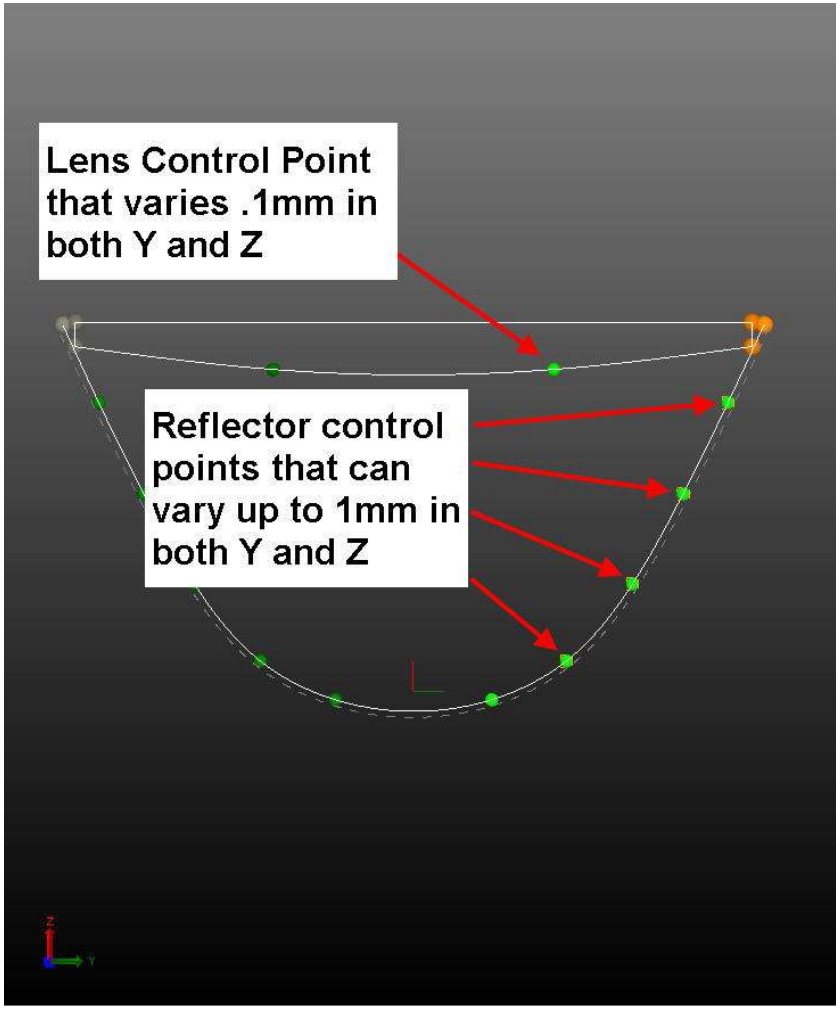
This example demonstrates how to setup a Monte Carlo Tolerance on a lens and reflector duo for a luminaire. The reflector is hexagonally faceted with 15 steps and 15 rings with an overall diameter of two inches. A plano-convex lens is placed at the exit aperture for the reflector.



TracePro

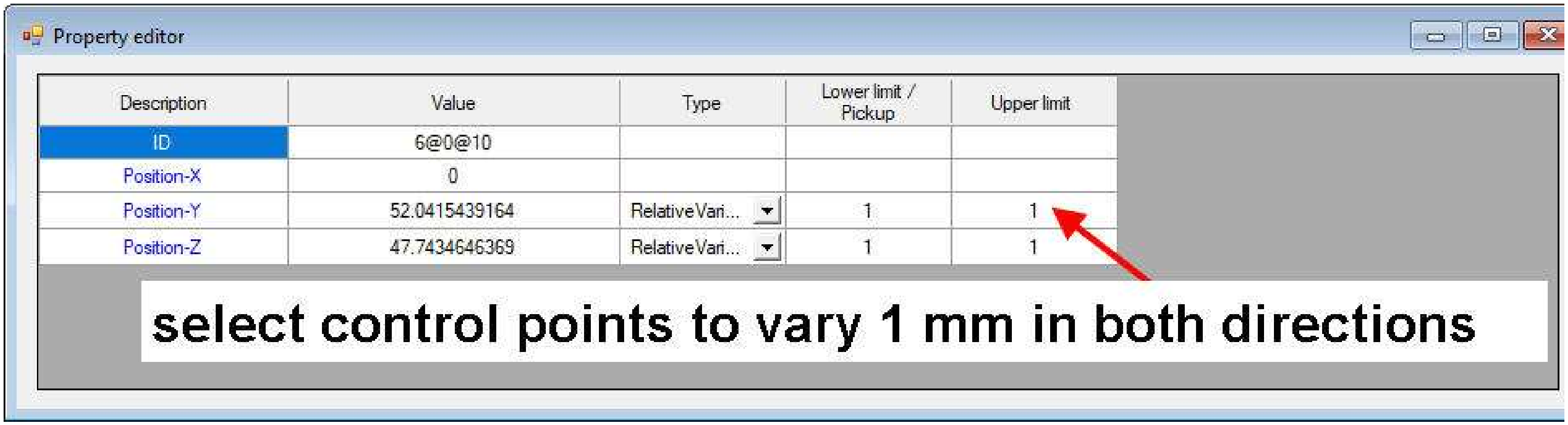
Example 3 – Luminaire Reflector and Lens Shape

The reflector has multiple control points and the lens has 1 that we want to tolerance for this example. We specify each one of the reflector control points to vary by 1 mm in both the Y an Z directions due to possible changes in the manufacturing process and to allow the lens control point to vary .1 mm in both the Y and Z directions as well. This gives us a total of 10 variables for this tolerance analysis.



TracePro

Example 3 – Setting two points as variables

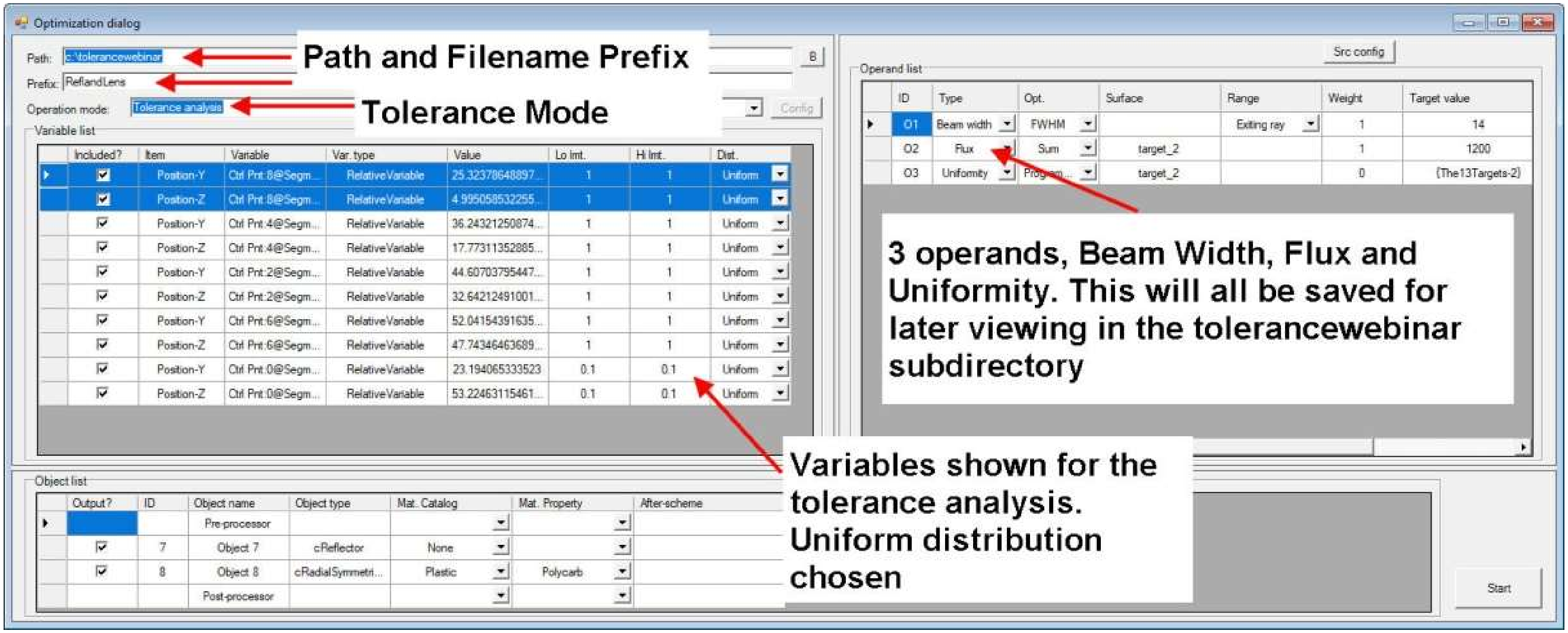


Description	Value	Type	Lower limit / Pickup	Upper limit
ID	6@0@10			
Position-X	0			
Position-Y	52.0415439164	RelativeVari...	1	1
Position-Z	47.7434646369	RelativeVari...	1	1

After selecting one segment or control point in the Surface List view, the Property Editor shows the exact value. Changing the type from Specified to Variable allows the user to select either an Absolute or Relative variable type and allows the user to input lower and upper limits for the variable.

TracePro

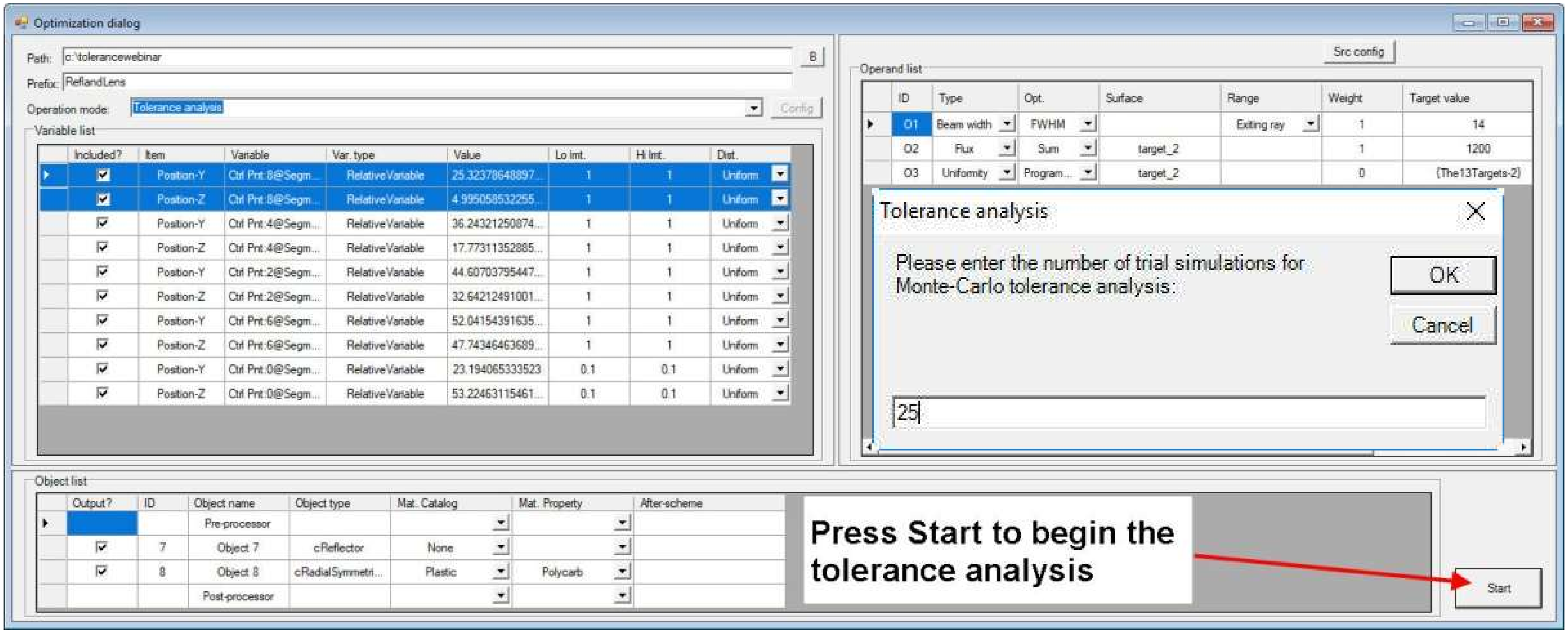
Example 3 – Setting parameters in the optimization dialog



Once again the optimization dialog is where the interim files location is set, the operation mode should be changed to Tolerance Analysis and the variables shown were previously set in the Property Editor. There are three operands used in this analysis to create the error function, Flux, Beam Width and Uniformity.

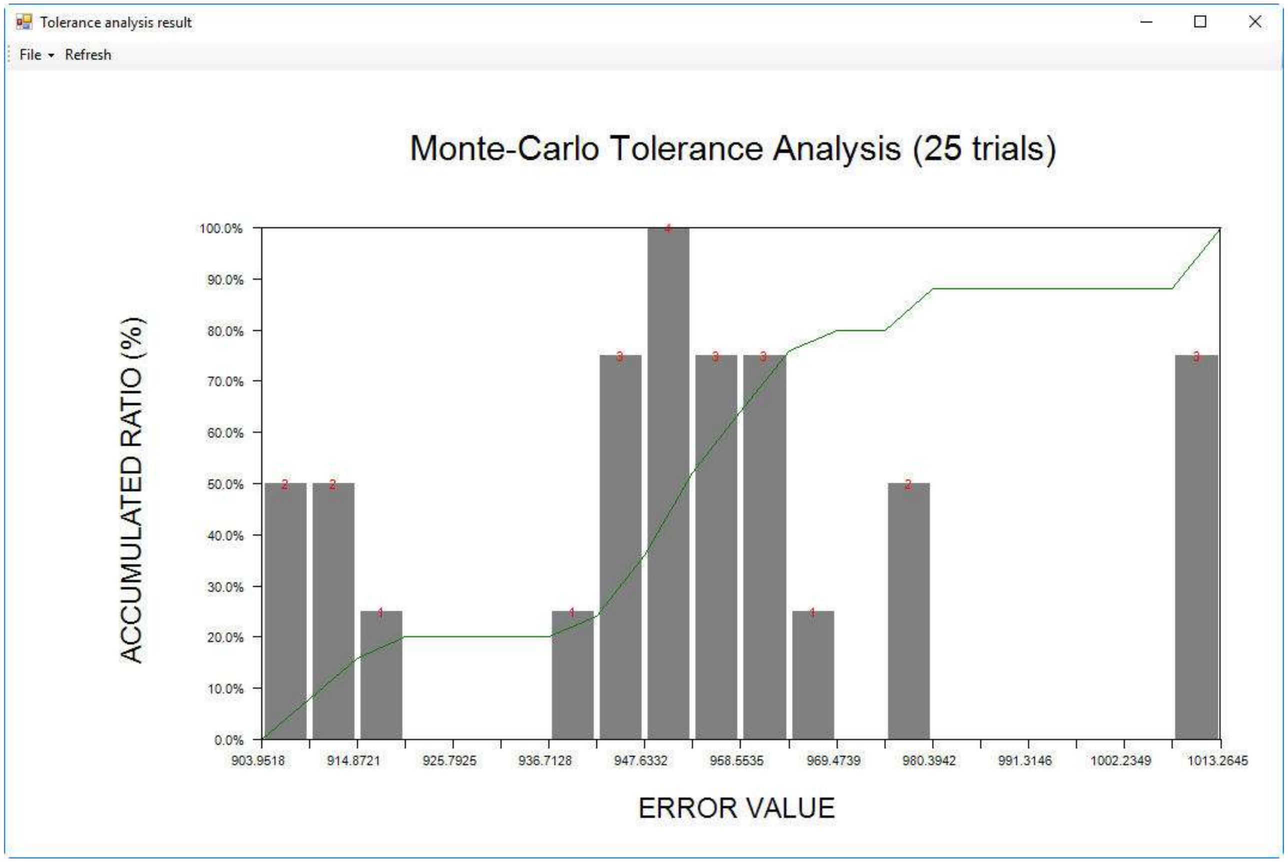
TracePro

Example 3 – Starting the analysis and setting the number of trials



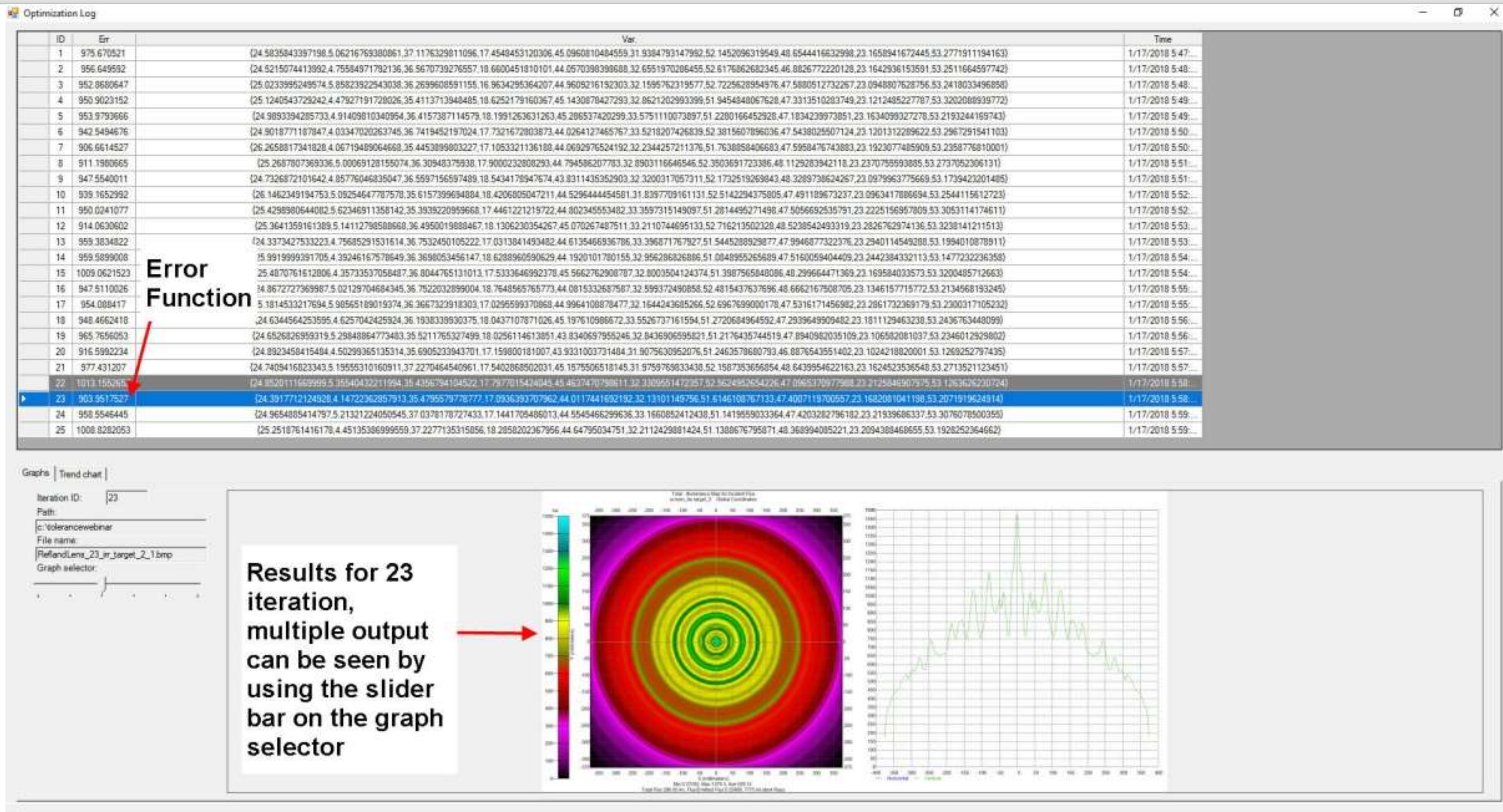
To start the analysis, click on the Start button in the lower right-hand corner of the optimization dialog. After a few seconds, the Tolerance analysis dialog should appear, you can now enter the number of iterations, 25 for this case.

Example 3 - Results of Monte Carlo Tolerancing 25 trials



After the analysis finishes the 25 trials, a graphical result will be shown. This figure shows the amount of error value versus accumulated ratio with the number of trials matching this result shown at the top in red. The majority of results for this analysis are in the 936 to .969 error value range .

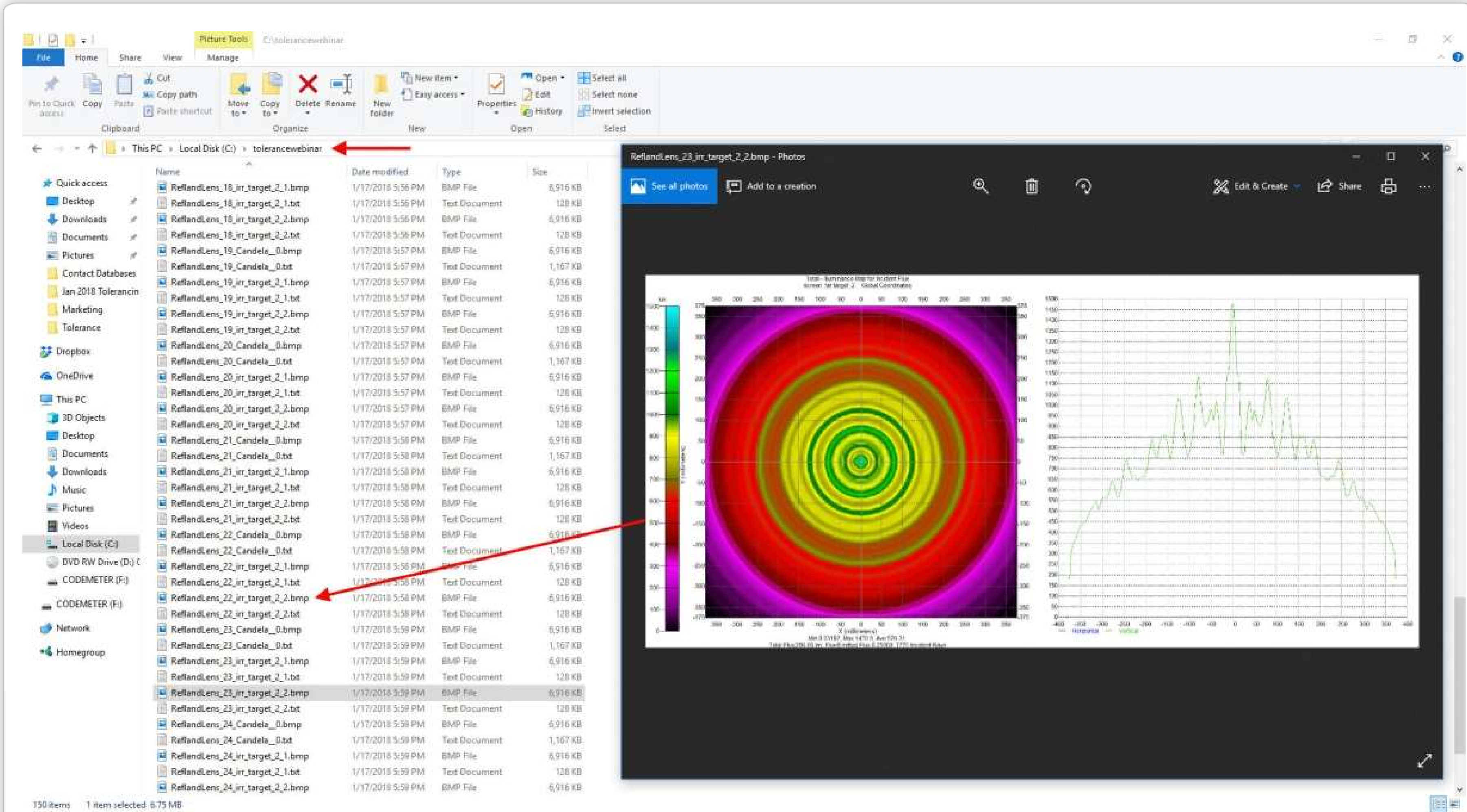
Example 3 - Results for each Monte-Carlo iteration



The results for iteration #23 is shown above in the optimization log. The Error Function is shown in the 2nd column and the Variable positions in the 3rd column. The Graph option is selected for the 23rd iteration and the corresponding illuminance map in pseudo-color output is shown bottom right.

TracePro

Examples 3 - Results for all 25 iterations are kept in the designated path set in the optimization dialog



View any result afterward by interrogating the files in the temporary subdirectory selected in the optimization dialog.

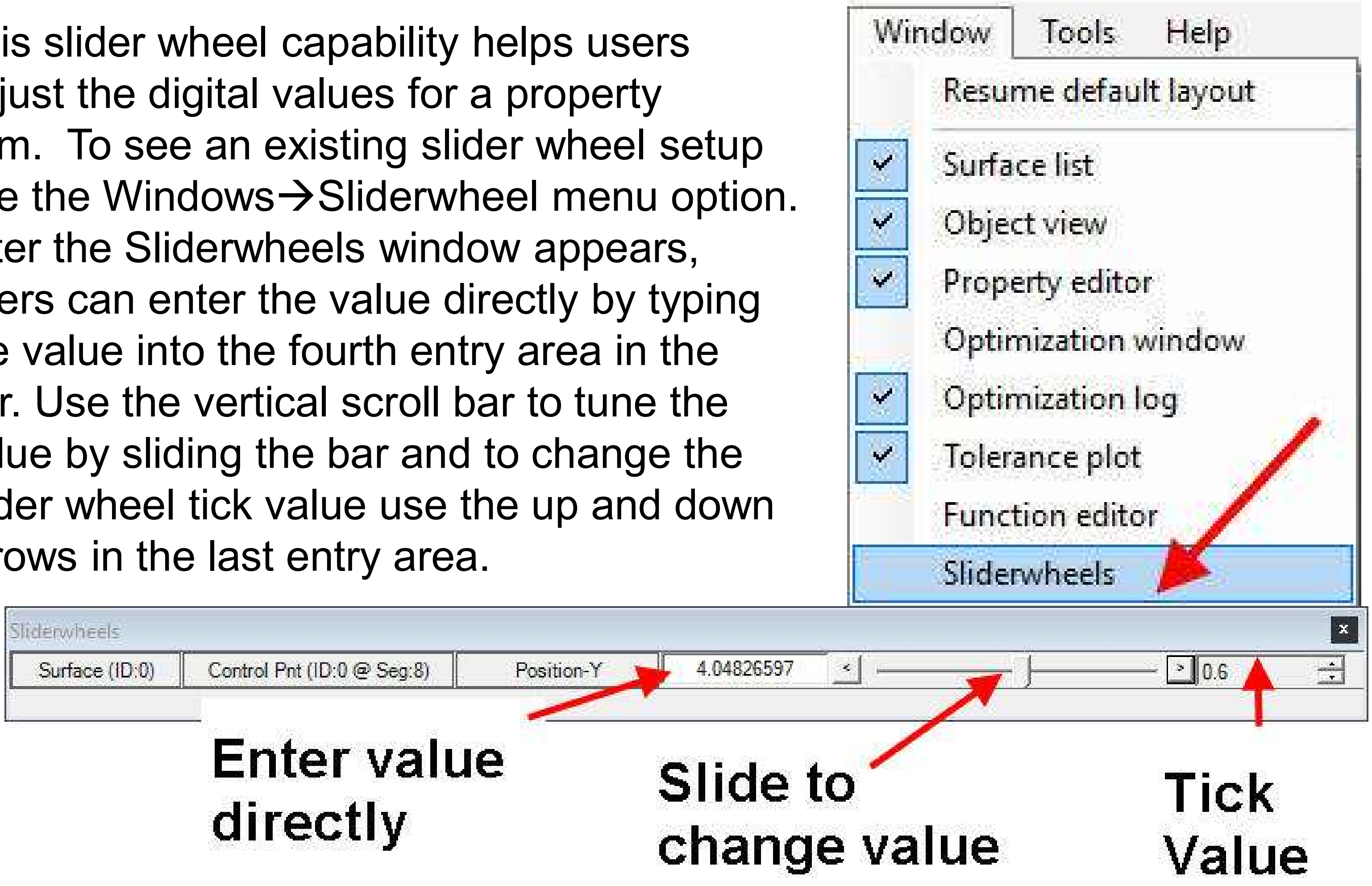
TracePro

Easter Egg - Slider Wheel Capability added to the 3D interactive optimizer

TracePro

Slider Wheels

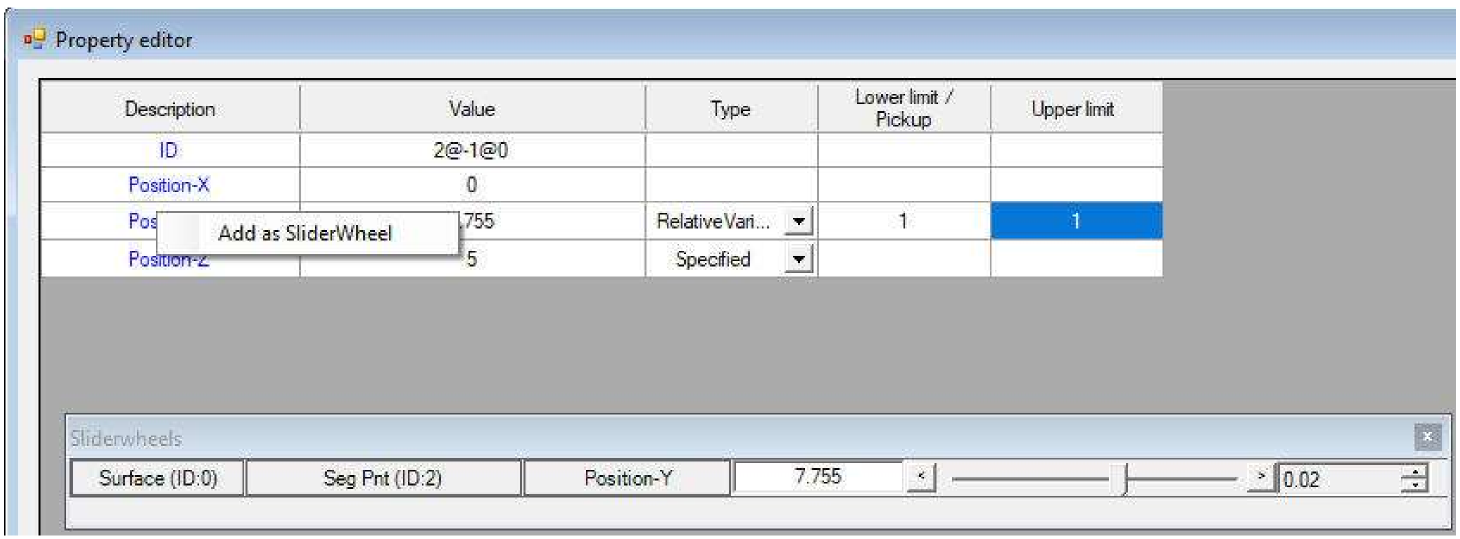
This slider wheel capability helps users adjust the digital values for a property item. To see an existing slider wheel setup use the Windows→Sliderwheel menu option. After the Sliderwheels window appears, users can enter the value directly by typing the value into the fourth entry area in the bar. Use the vertical scroll bar to tune the value by sliding the bar and to change the slider wheel tick value use the up and down arrows in the last entry area.



TracePro

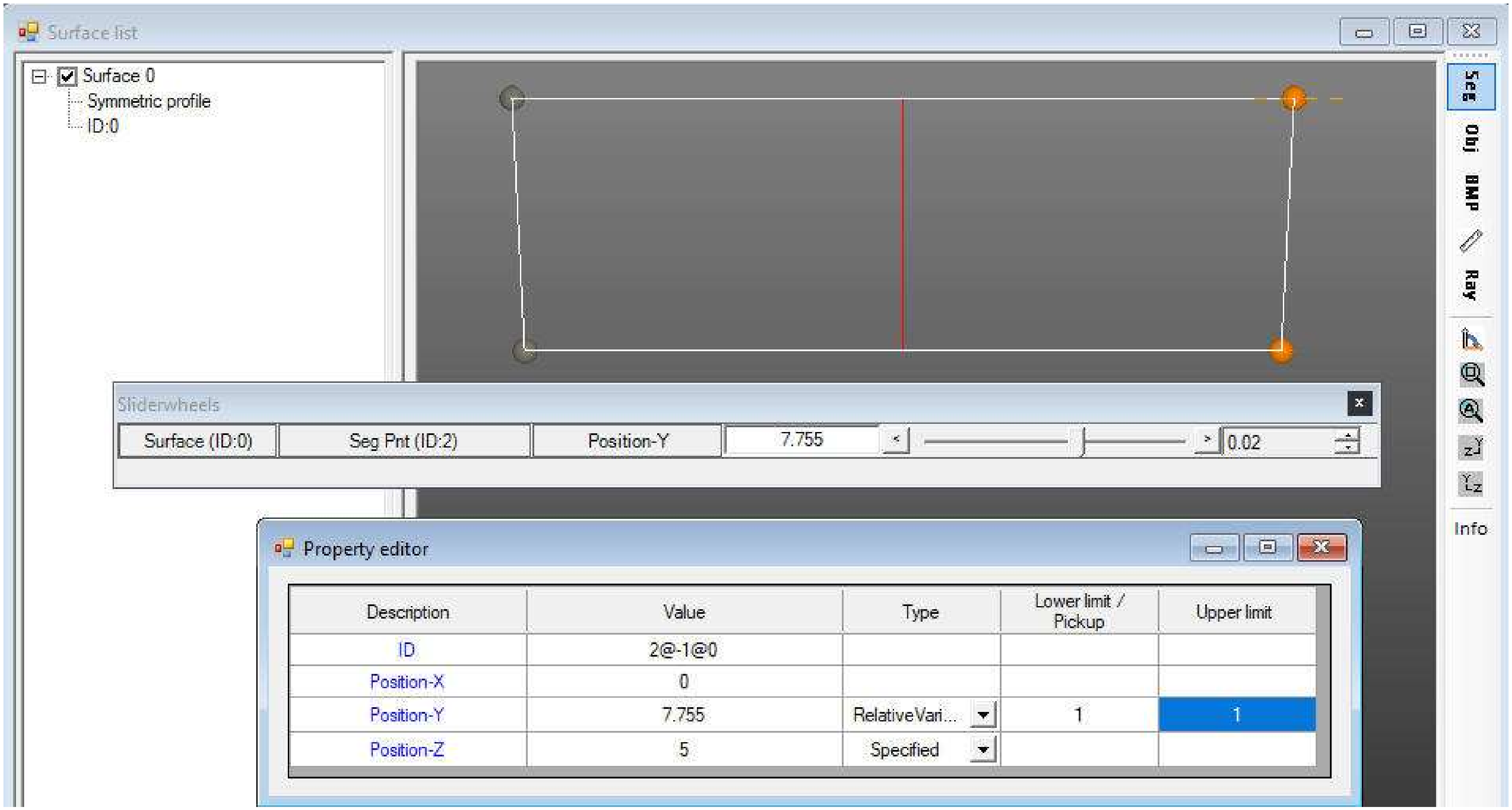
How to add a Slider Wheel

- Step 1 – Create any geometry
- Step 2 – Select any control or segment point on an objects
- Step 3 – in the Property Editor modify the type from Specified to Relative or Absolute Variable
- Step 4 - Define a lower and upper limit
- Step 5 - Right Click on the Description column either Position Y or Position Z, the slider wheel option will then be displayed
- Step 6 – Click on the pop-up - Add as a Sliderwheel
- Step 7 – Drag the slider-wheel to change the geometry in the Surface list window



TracePro

Slider wheel control example



The defined Slider wheel varies the two Y Positions of the upper right segment point since this is a symmetric profile.

TracePro

Conclusions

- A Monte Carlo Tolerancing analysis is quite beneficial to see how a system will perform due to errors in the manufacturing process.

- Investigate the post-process results to see how the system results performed individually to get a feeling on how well your system performs for both the best and worst result. This will let you know your manufacturing failure rate before production.

- Make sure you specify enough trial iterations. For the first 2 examples,100 trials was sufficient to get a good statistical result. In the last example, 25 trials were not enough and there were gaps in the error function results.

Questions & Answers

Thank You!!

Interested in Learning More?

Sign up for a **free** 30-day trial of TracePro at:
http://lambdares.com/trials

Michael Gauvin
Vice President Sales & Marketing
mgauvin@lambdares.com
+1 978-486-0766 x0037

TracePro