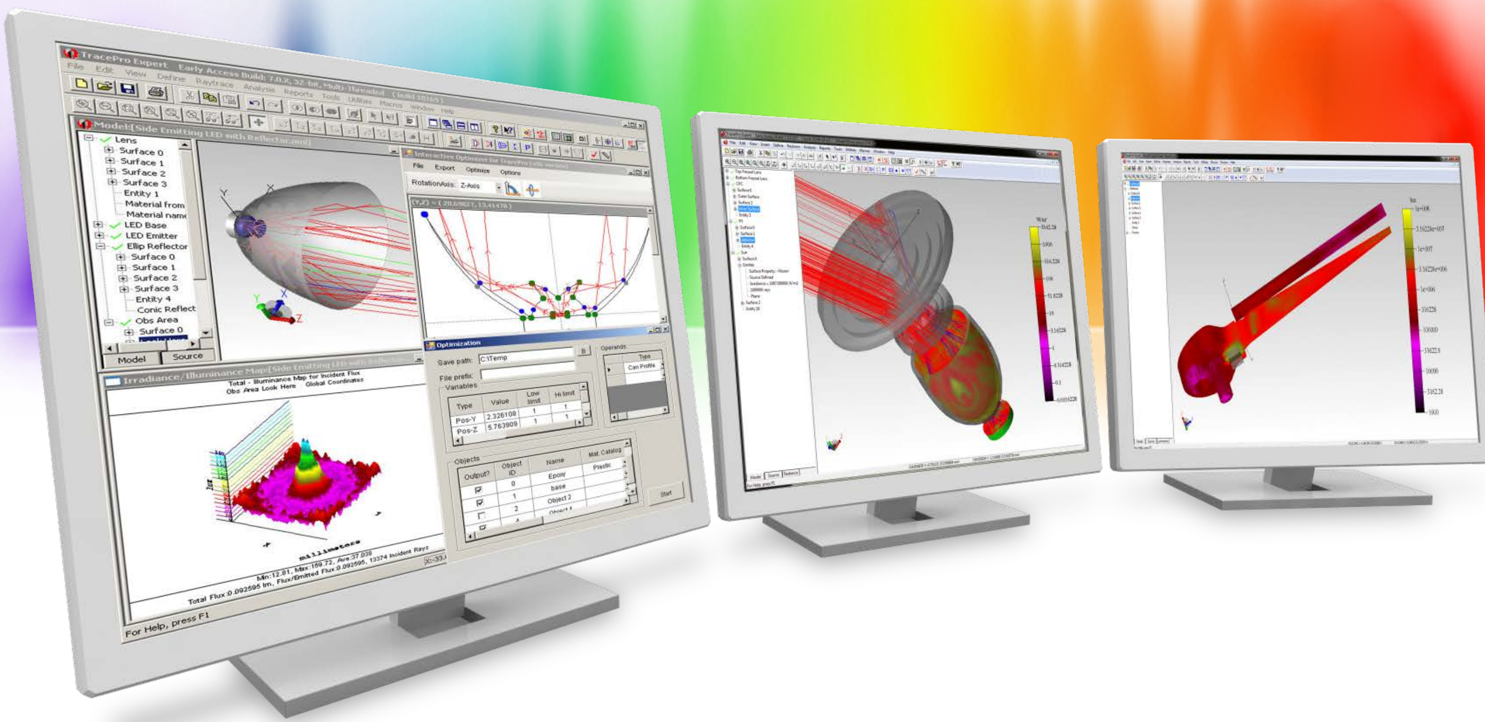
TracePro



Optimizing the TracePro Optimization Process

A TracePro Webinar

December 17, 2014

LAMBDA RESEARCH CORPORATION

Presenter

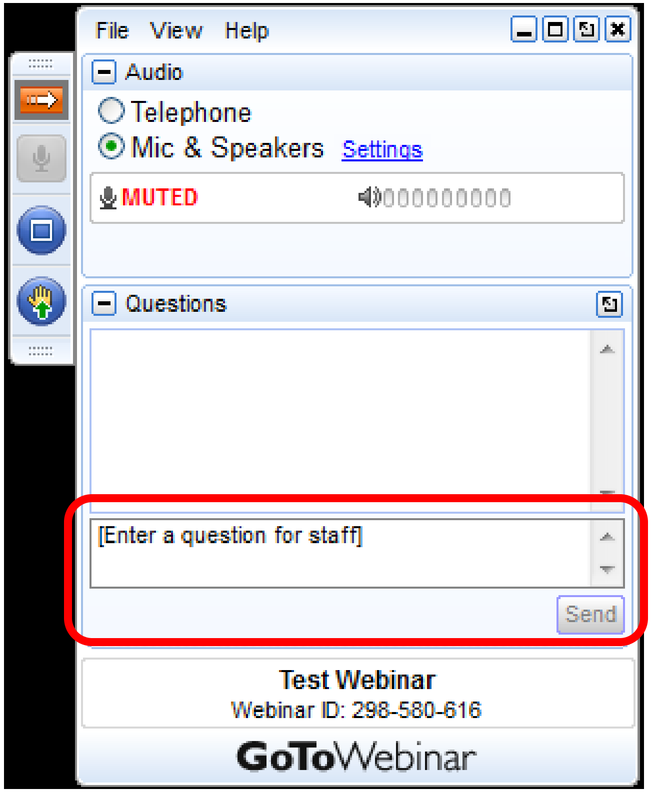
- **Presenter**
  Dave Jacobsen
  Sr. Application Engineer
  Lambda Research Corporation

- **Moderator**
  Mike Gauvin
  Vice President of Sales and Marketing
  Lambda Research Corporation

TracePro

Format

- A 25-30 minute presentation followed by a question and answer session
- Please submit your questions anytime using Question box in the GoToWebinar control panel



TracePro

Additional Resources

- Past TracePro Webinars

  http://www.lambdares.com/webinars

- TracePro Tutorial Videos

  http://www.lambdares.com/videos

- TracePro Tutorials

  http://www.lambdares.com/features/tracepro-tutorials

- Information on upcoming TracePro Training Classes

  http://www.lambdares.com/training/software-training

TracePro

Upcoming TracePro Training

- **University of Applied Sciences – Jena, Germany**
  - Introduction to TracePro – Mar. 10 -11, 2014
  - Optimization with TracePro– Mar. 12, 2014
  - Stray Light Analysis Using TracePro – Mar. 13, 2015

- **Littleton, MA USA**
  - Introduction to TracePro – Mar. 23 – Mar. 24, 2015
  - Optimization with TracePro – Mar. 25, 2015
  - Stray Light Analysis Using TracePro – Mar. 26, 2015
  - Scheme Macro Programming – Mar. 27, 2015

Latest TracePro Release

TracePro 7.5.1

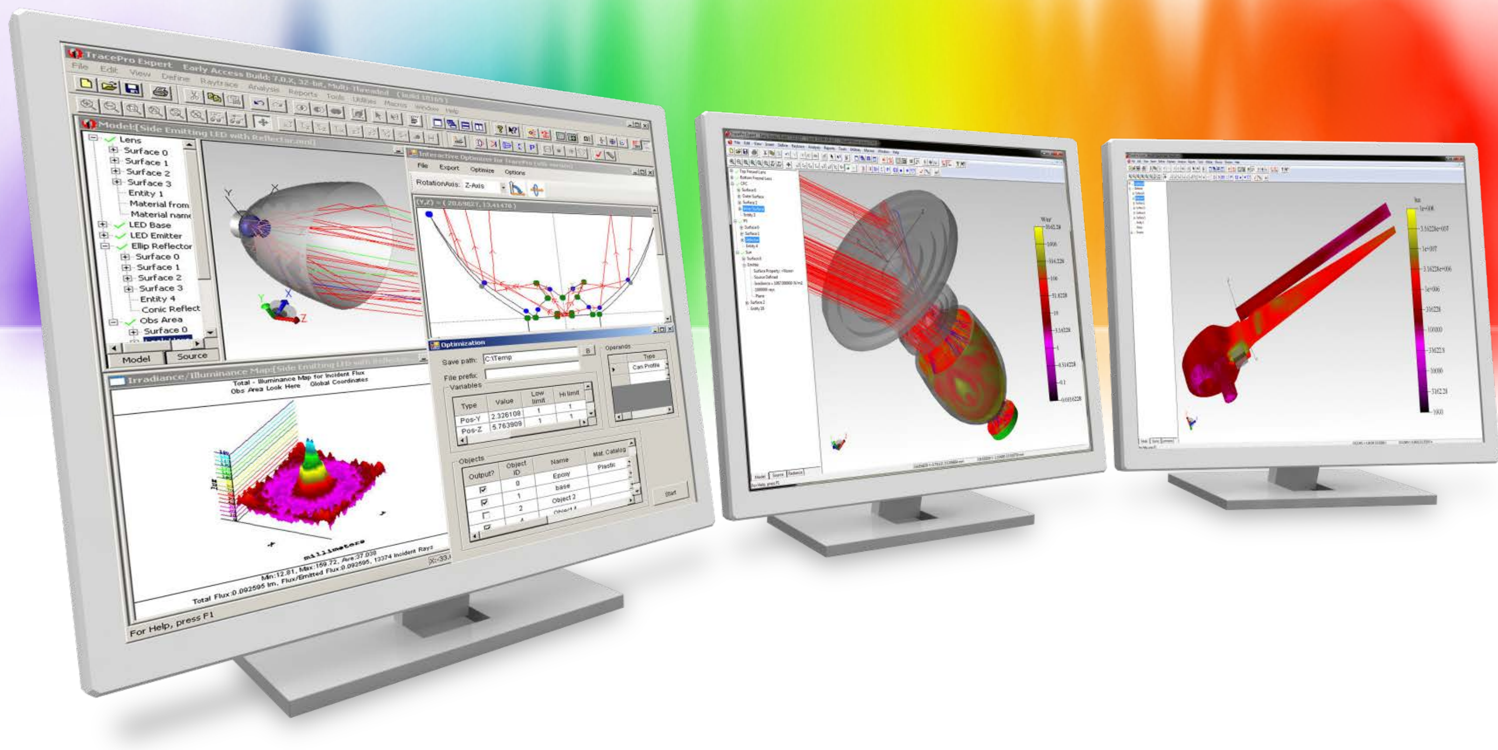
Released December 4, 2014

Customers with current maintenance and support agreements can download this new release at:

http://www.lambdares.com/CustomerSupportCenter/index.php/trace-pro/current-release

Agenda

- Introduction
- The need for an optimization process
- Optimization theory and methods
- Optimization parameters and settings
- Hybrid system optimization example
- Optimization tips
- Review and questions and answers



Introduction

Introduction

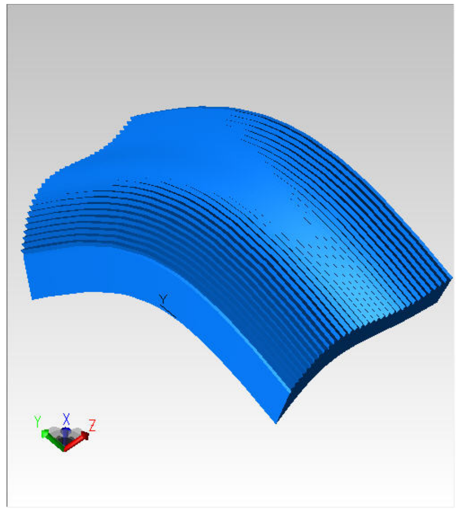
What is optimization?

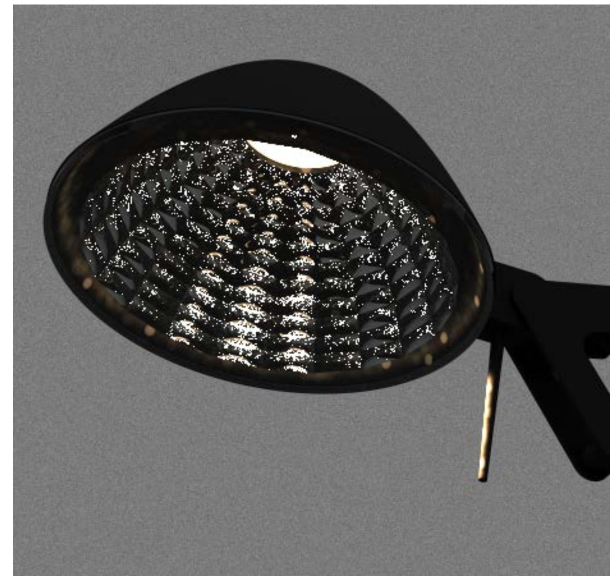
- An act, process, or methodology of making something (as a design, system, or decision) as fully perfect, functional, or effective as possible. *(Source: Merriam-Webster online dictionary)*

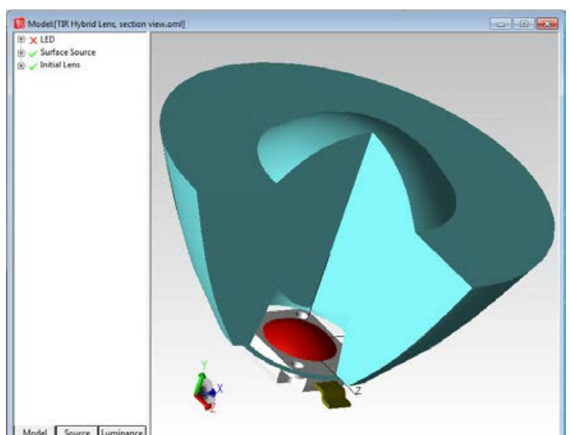
Introduction

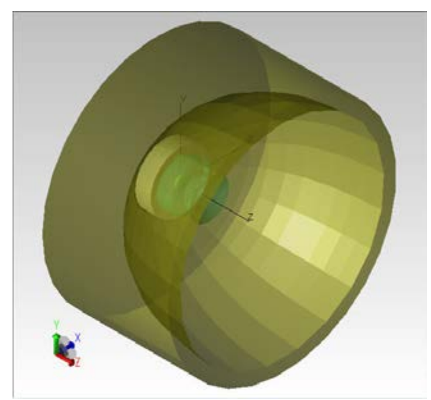
What are some of the parameters that can optimized?

- Geometry
- Curvature
- Facets
- Position
- Angle
- Spacing
- Thickness
- Properties









TracePro

Introduction

What many people would like to see

Introduction

Or

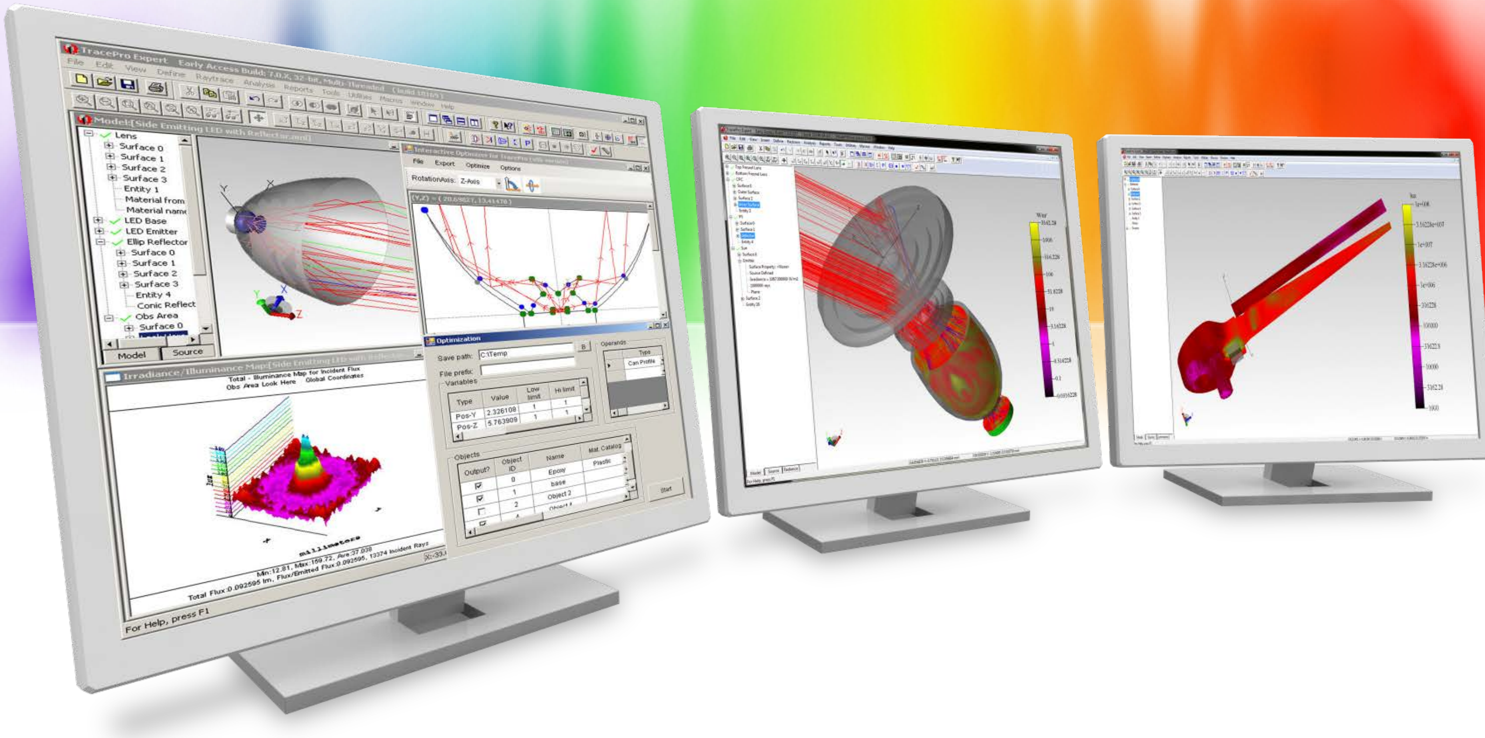
TracePro

Introduction

What we can try to do

TracePro

TracePro

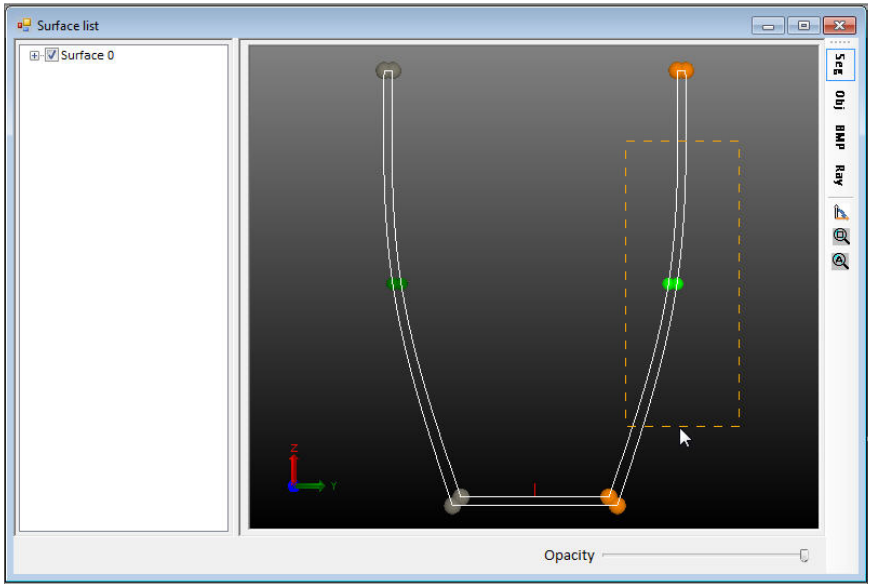


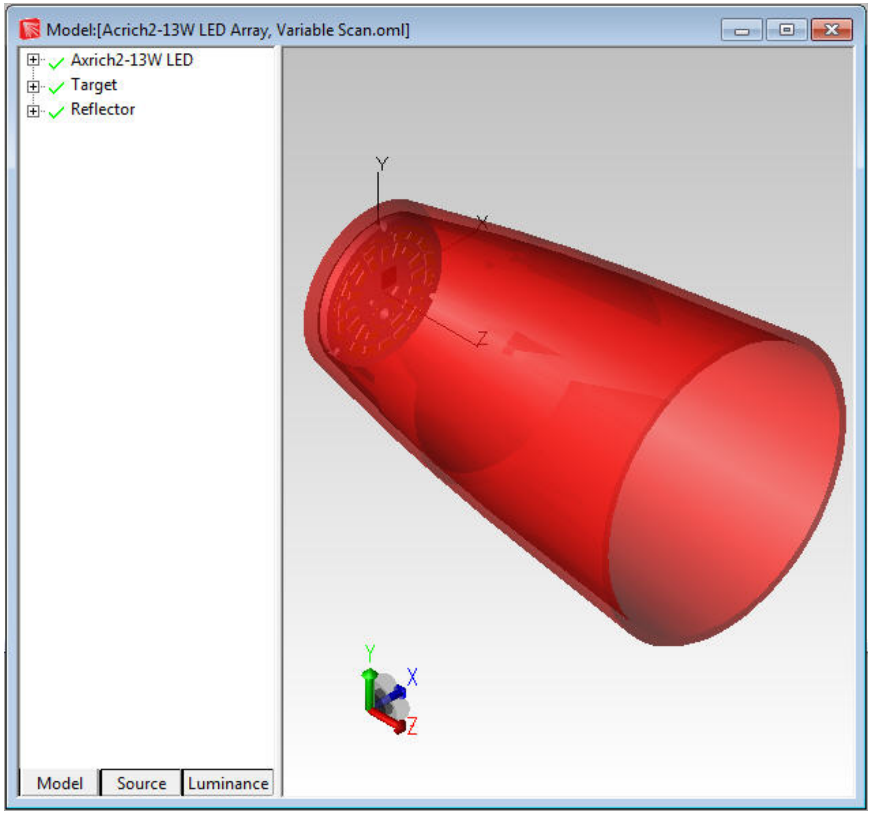
Why do we need an optimization process?

LAMBDA
RESEARCH CORPORATION

Why do we need an optimization process?

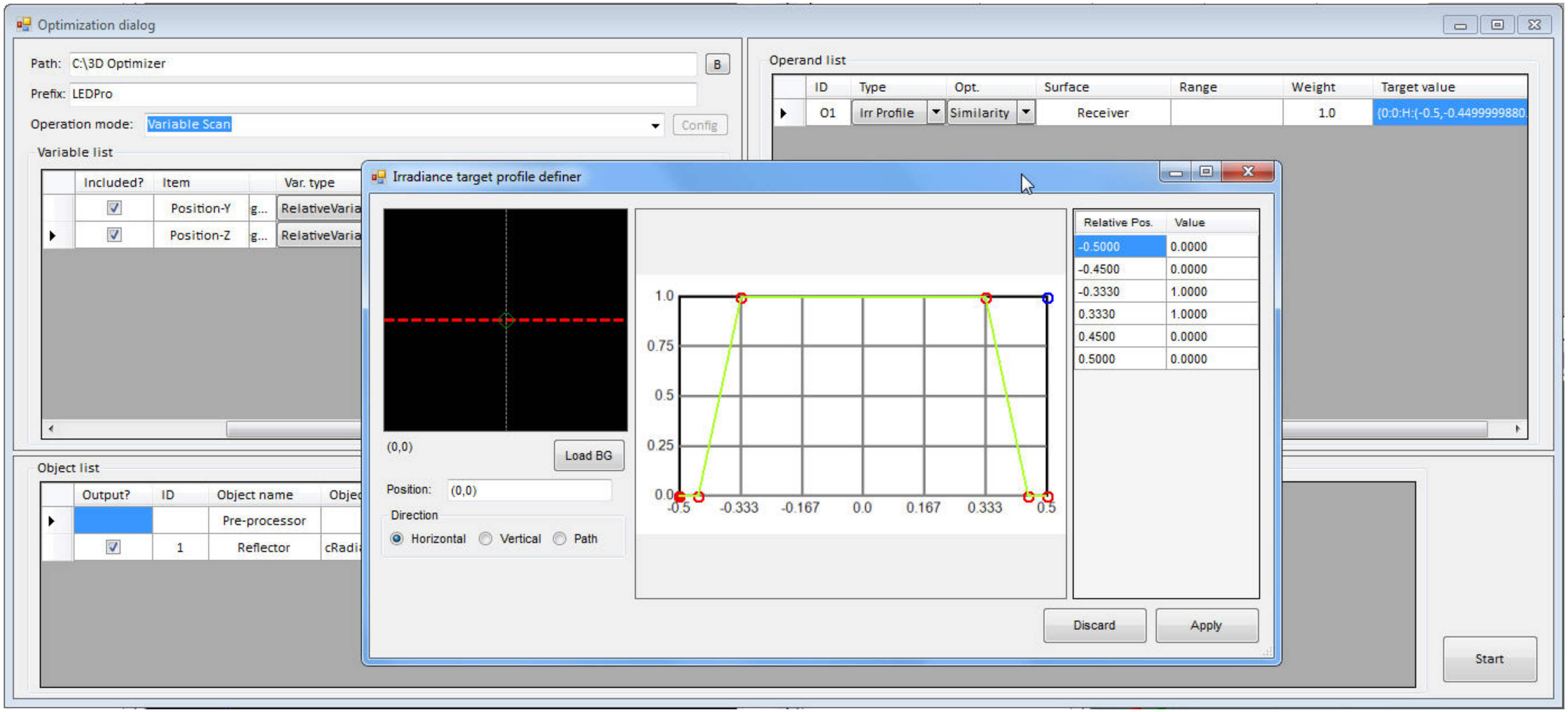
Why do we need an optimizer? - Brute force vs. Optimization algorithm – The goal is to optimize the reflector shown below





Why do we need an optimization process?

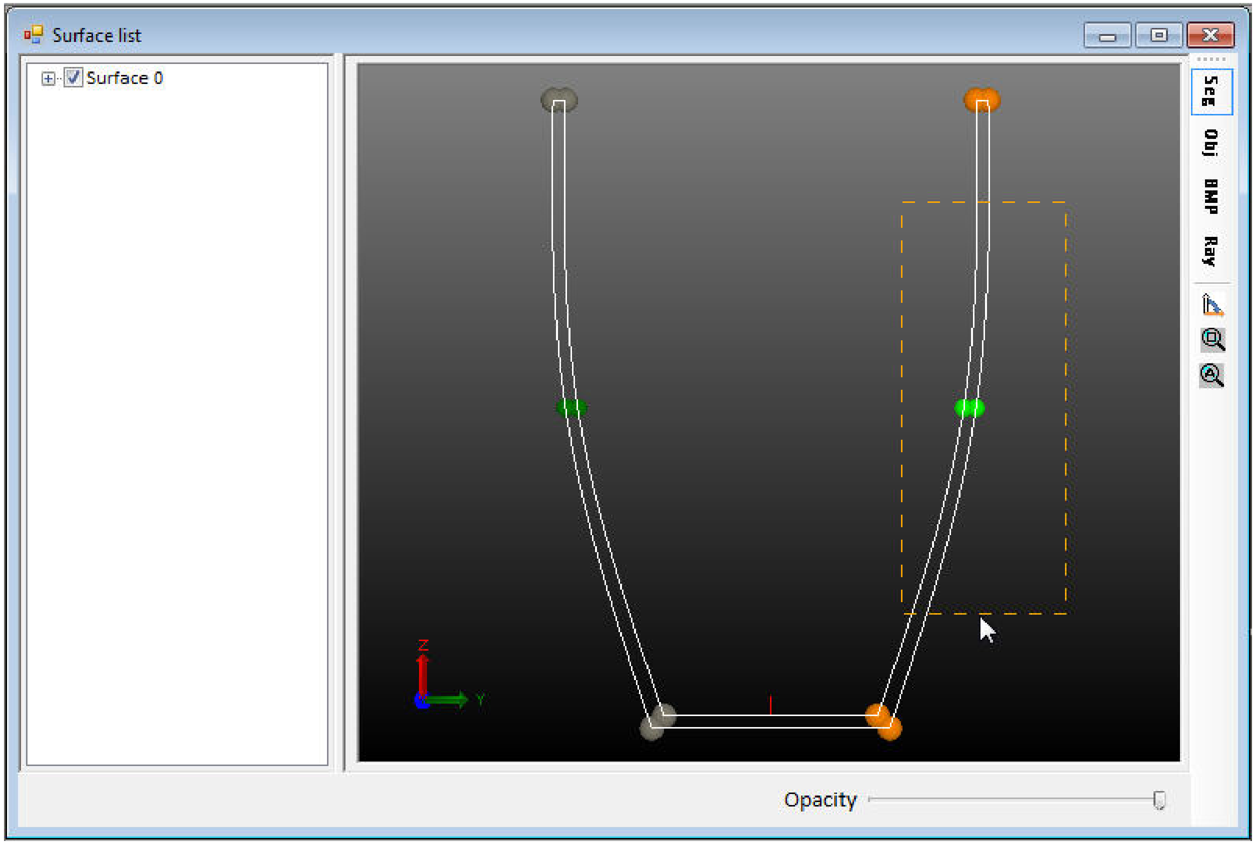
Optimization Goal



TracePro

Why do we need an optimization process?

Variable range – 40mm in Y-axis and 100mm in Z-axis



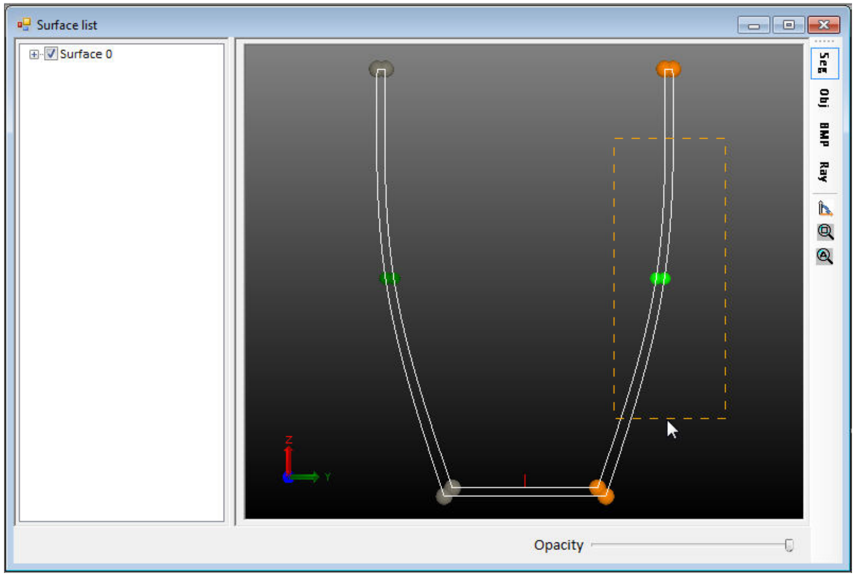
TracePro

Why do we need an optimization process?

Variable range – 40mm in Y-axis and 100mm in Z-axis

Scanning the entire variable range in 0.1mm increments would take 41 x 101 = 4141 increments.

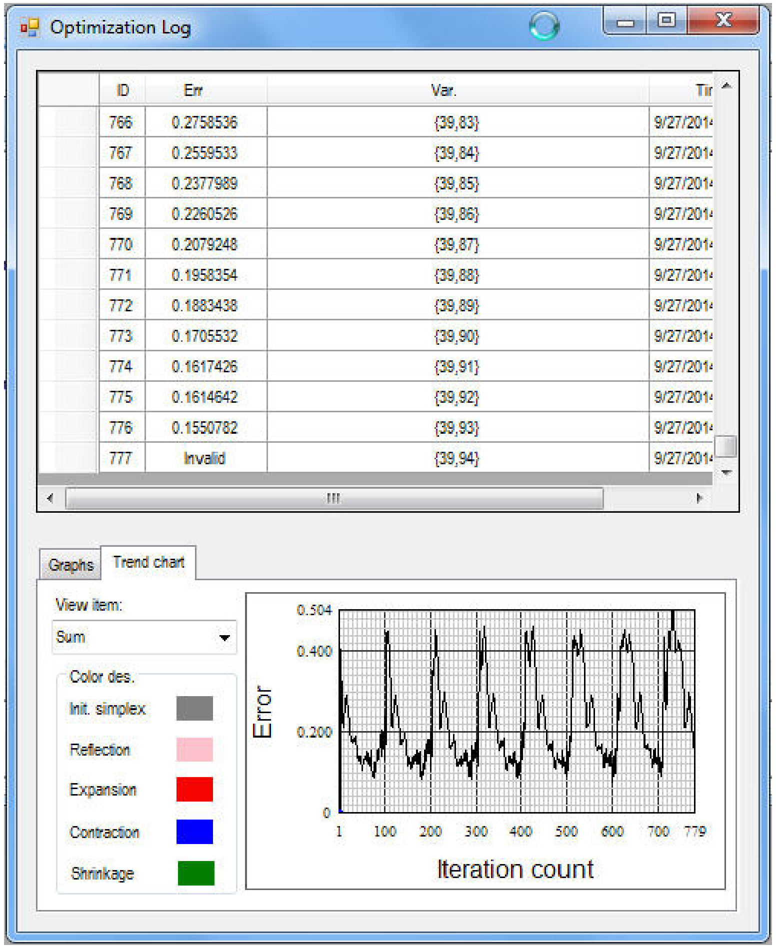
If the raytrace time is 1-minute per iteration, this would take around 70-hours to complete.



TracePro

Why do we need an optimization process?

Brute force – Optimization Log after 14-hours of raytracing

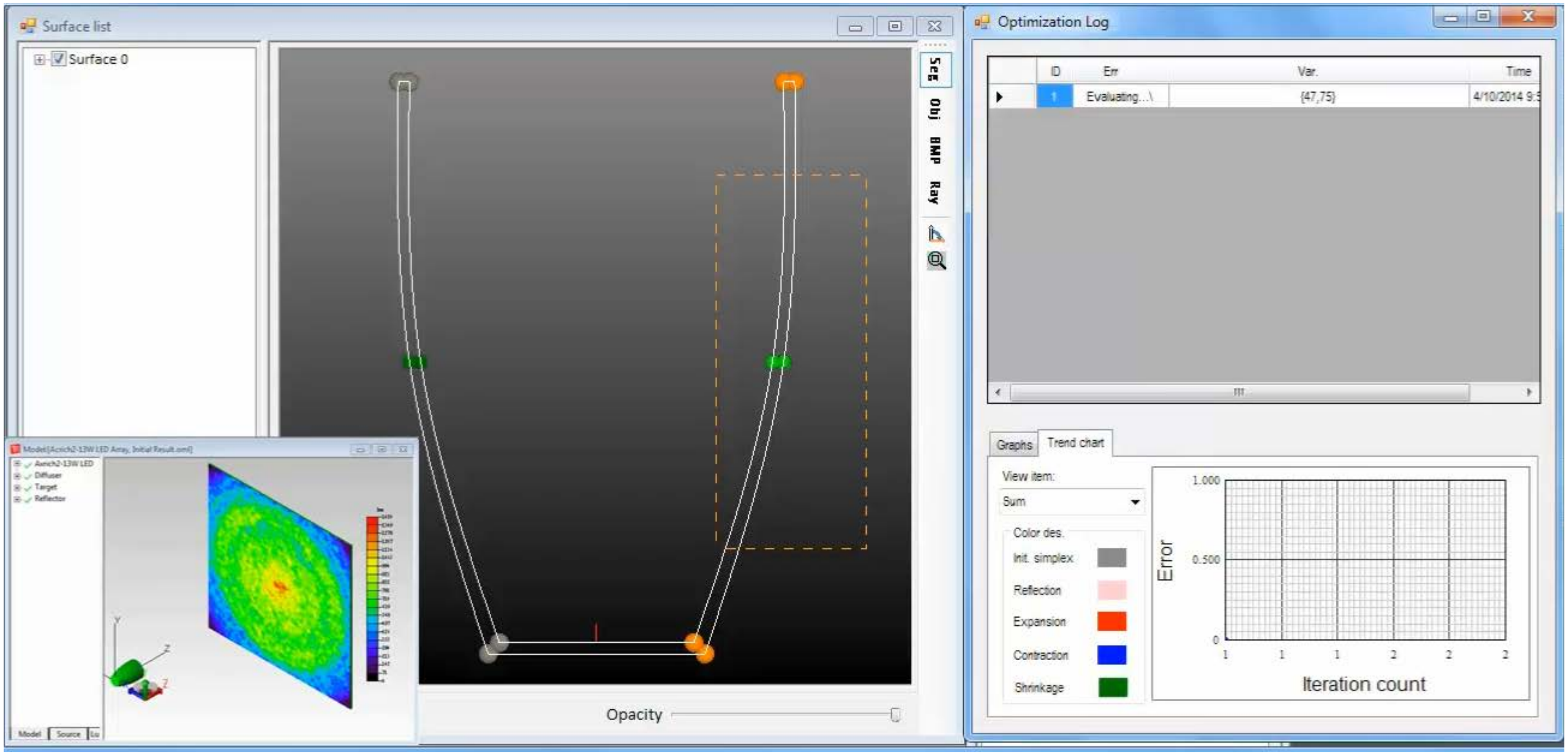


ID	Err	Var.	Ti
766	0.2758536	{39,83}	9/27/201
767	0.2559533	{39,84}	9/27/201
768	0.2377989	{39,85}	9/27/201
769	0.2260526	{39,86}	9/27/201
770	0.2079248	{39,87}	9/27/201
771	0.1958354	{39,88}	9/27/201
772	0.1883438	{39,89}	9/27/201
773	0.1705532	{39,90}	9/27/201
774	0.1617426	{39,91}	9/27/201
775	0.1614642	{39,92}	9/27/201
776	0.1550782	{39,93}	9/27/201
777	Invalid	{39,94}	9/27/201

TracePro

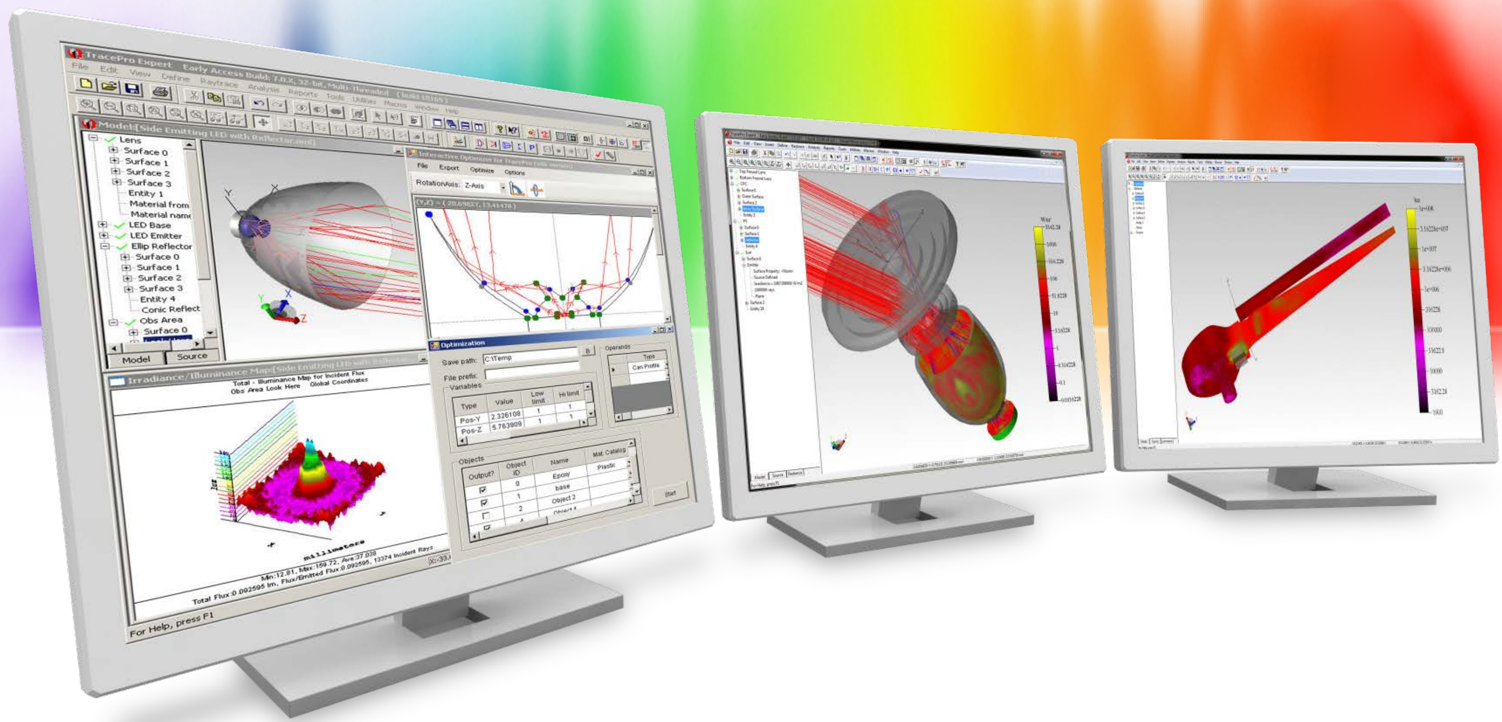
Why do we need an optimization process?

Optimization algorithm – total time of about 2 hours 20 minutes with more rays traced for each iteration - Video



TracePro

TracePro



Optimization Theory and Methods

LAMBDA
RESEARCH CORPORATION

Optimization theory and methods

Generally there are 2 types of optimizers: Global and Local.

Global optimizers will search the entire solution space to find the best solution based on the optimization goal or merit function.

Local optimizers will find the solution closest to the starting point of the optimization process. Changing the starting conditions can change the results of the optimization process.

Optimization theory and methods

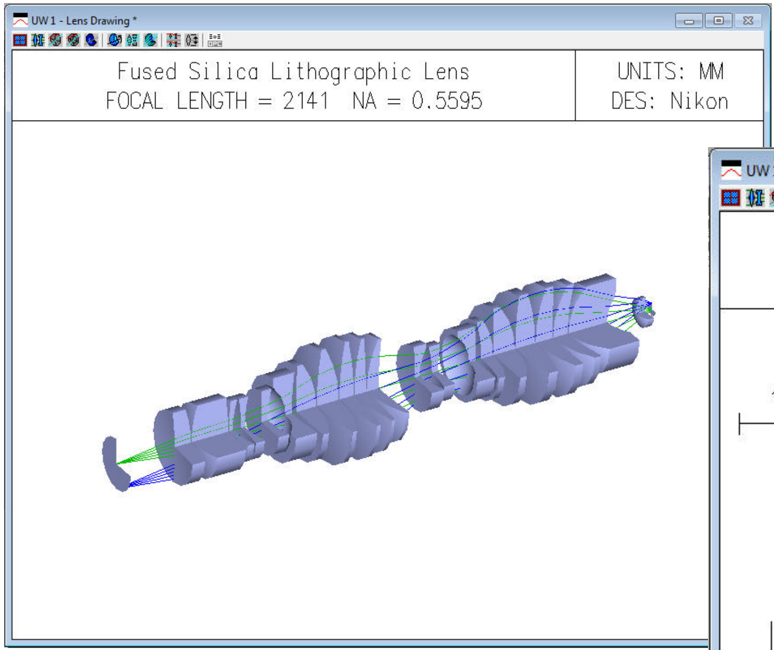
Examples of Global optimization methods include:

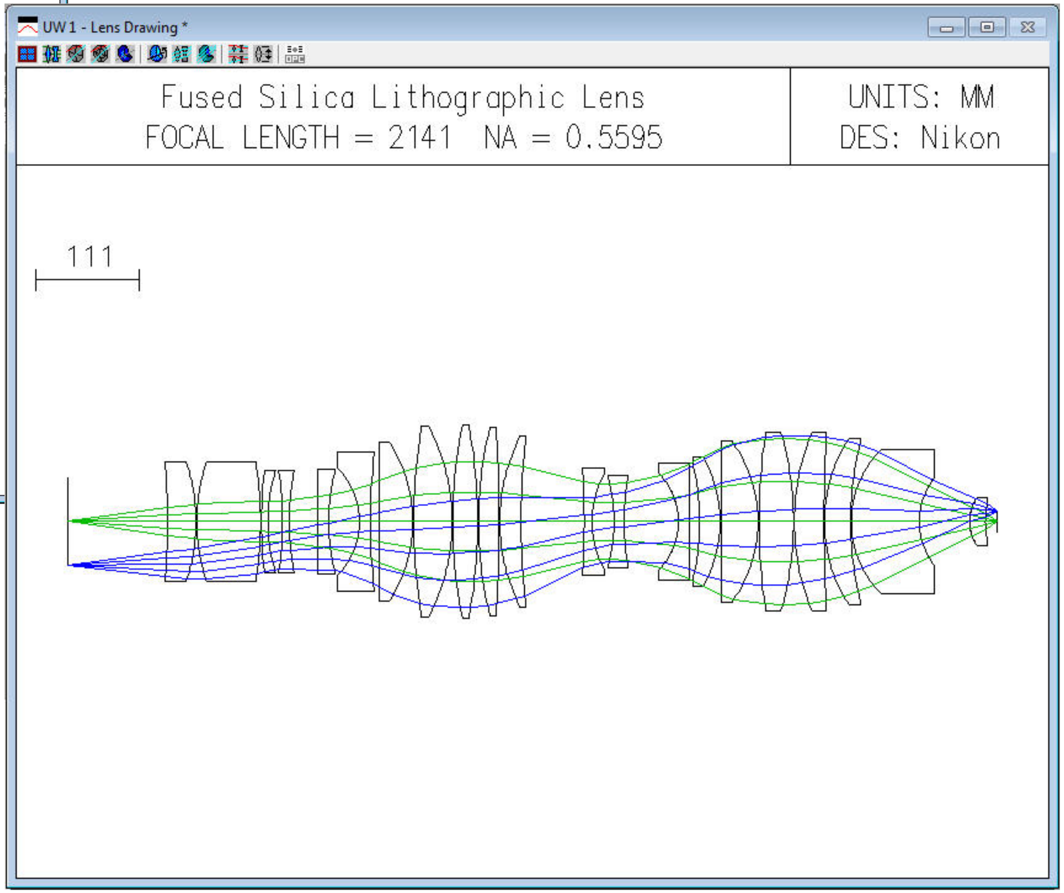
- Global Explorer
- Adaptive Simulated Annealing
- Global Synthesis
- Hammer optimization

Global optimization routines will generally have a function to allow them to escape from local solutions and sample more of the solution space in an attempt to find the best overall solution. Lens design programs such as OSLO will typically have global optimization options.

TracePro

Optimization theory and methods





TracePro

Optimization theory and methods

Examples of Local optimization methods include:

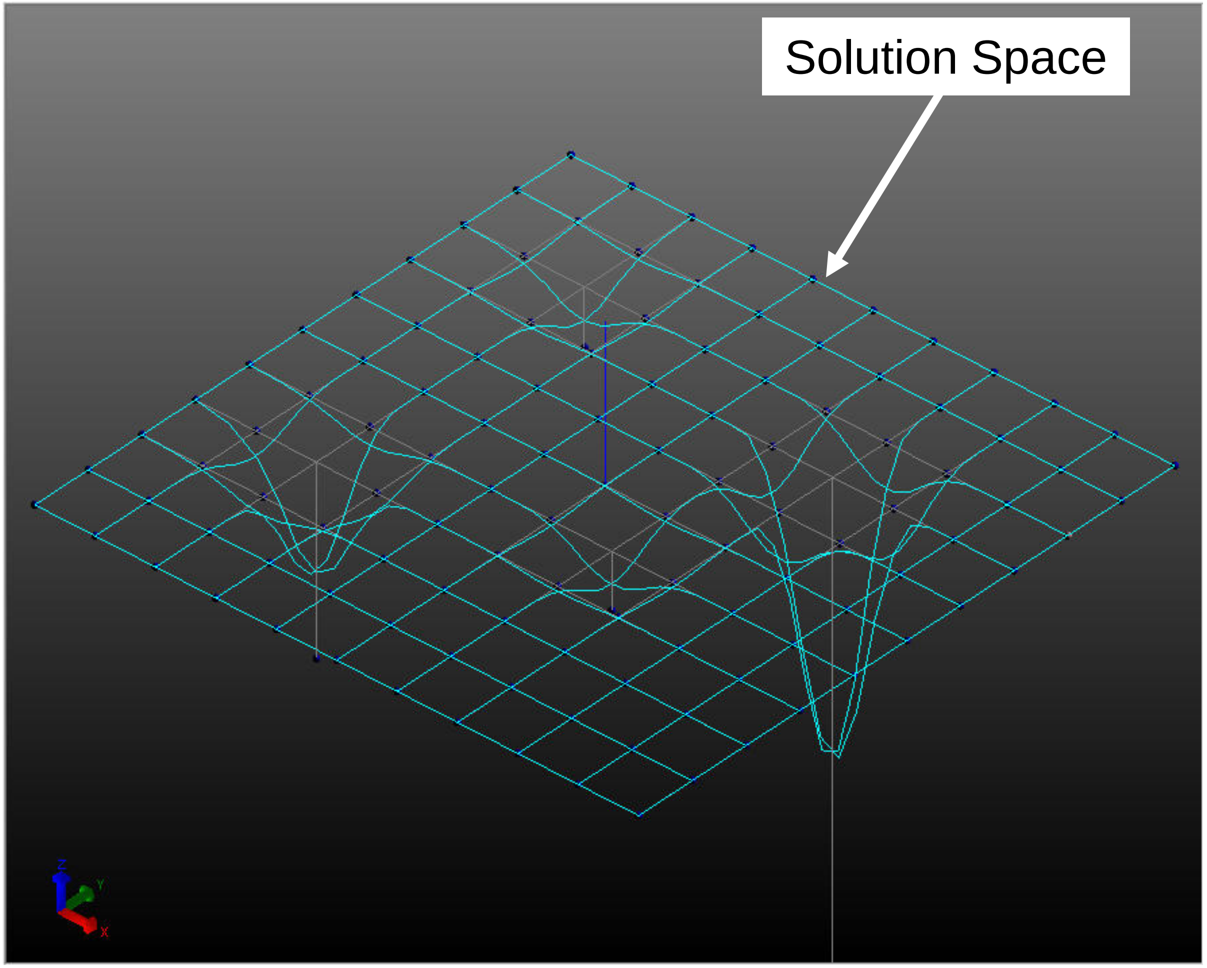
- Damped Least Squares
- Powell's Method
- **Nelder-Mead or Downhill Simplex Method**
- **Variable Scanning**

Local optimization routines do not have an escape function and will tend to converge on the solution closest to the starting condition. Changing the starting conditions will allow the optimization routine to sample more of the solution space and see if better solutions are available. Illumination design programs such as TracePro will typically feature local optimizers.

TracePro

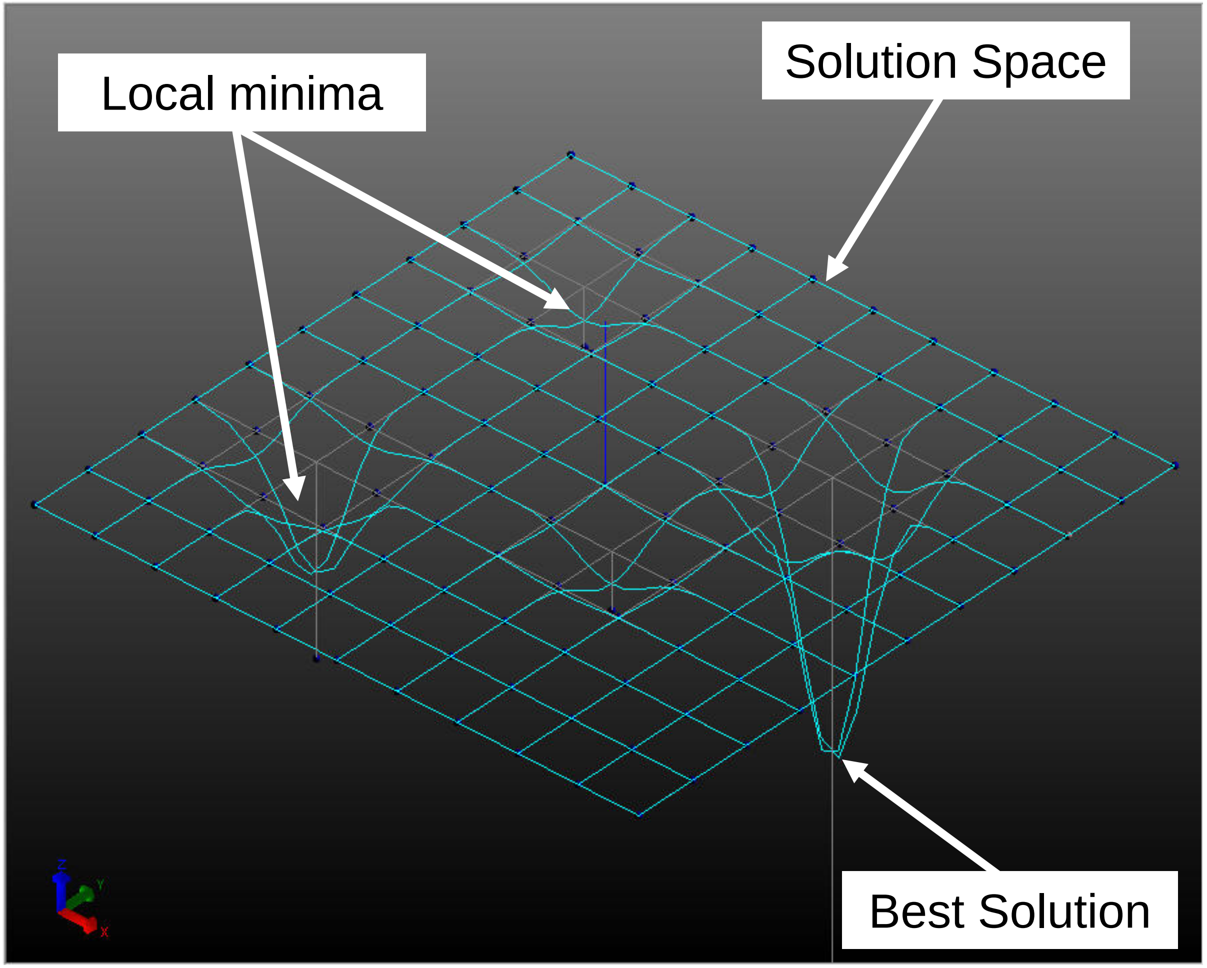
Optimization theory and methods

Solution Space for optimization problem



Optimization theory and methods

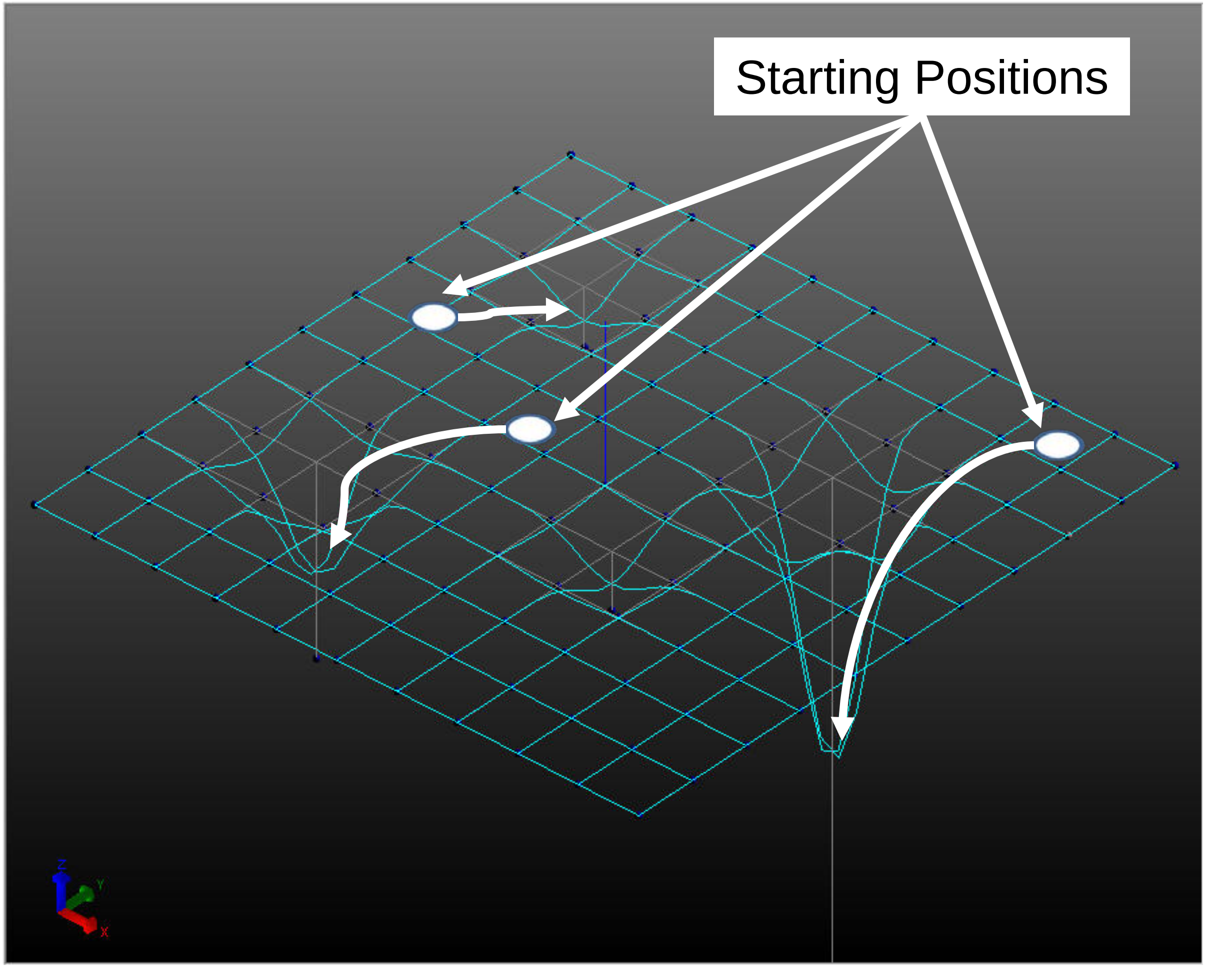
Solution Space for optimization problem with possible solutions



TracePro

Optimization theory and methods

Solution Space for optimization problem with possible solutions



Optimization theory and methods

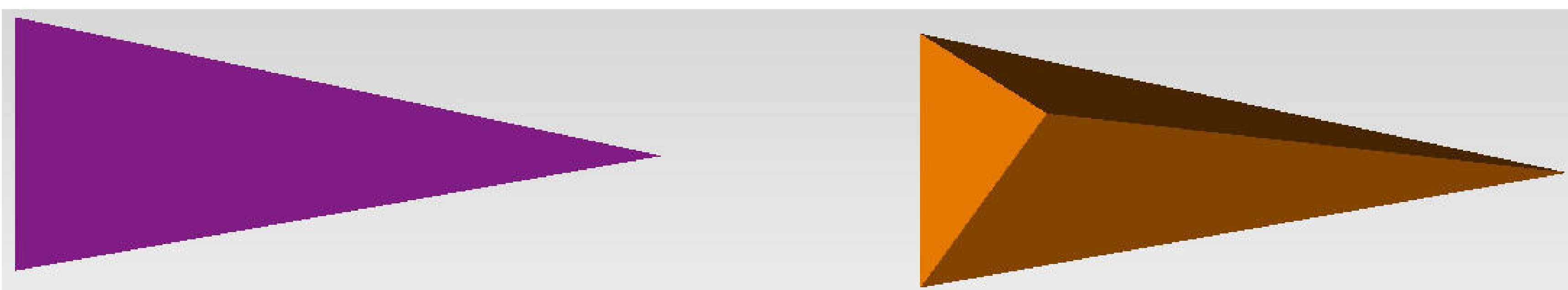
The Downhill Simplex, or Nelder-Mead, method for optimization was proposed by John Nelder and Roger Mead in 1965.

The Downhill Simplex method is a local optimization method, meaning it will converge to the solution closest to the starting point. It's possible that a better solution is available. Changing the initial starting conditions can be used as a test to see if a better solution is available. This is a good choice when optimizing geometry, position, and rotation where it is desirable to "jump" around the solution space to find and then refine the best choices for variable values.

Optimization theory and methods

The Nelder-Mead method uses the concept of a simplex, which is a special polytope of N+1 vertices in N dimensions. Examples of simplicies include a line segment on a line, a triangle on a plane, and a tetrahedron in 3-dimesional space.

A polytope is a geometric object with flat sides, which exists in any general number of dimensions. A polygon is a polytope in two dimensions, a polyhedron in three dimensions, and so on in higher dimensions (such as a polychoron in four dimensions).



2 Variables = 2 Dimensions & 3 Vertices

3 Variables = 3 Dimensions & 4 Vertices

Source: Wikipedia

Optimization theory and methods

A simple example for 2 variables:

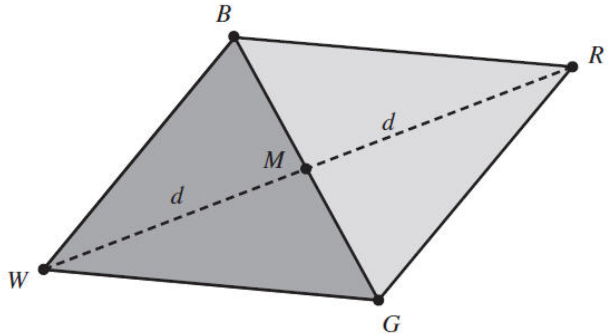
For two variables, the simplex is a triangle. The algorithm compares the error function at each vertex of the triangle, rejects the vertex where the error function is highest, and replaces it with a new vertex. This forms a new triangle and the process is repeated.

The process generates a sequence of triangles where the error function at the vertices gets smaller and smaller. The size of the triangles is reduced and the local minimum is found.

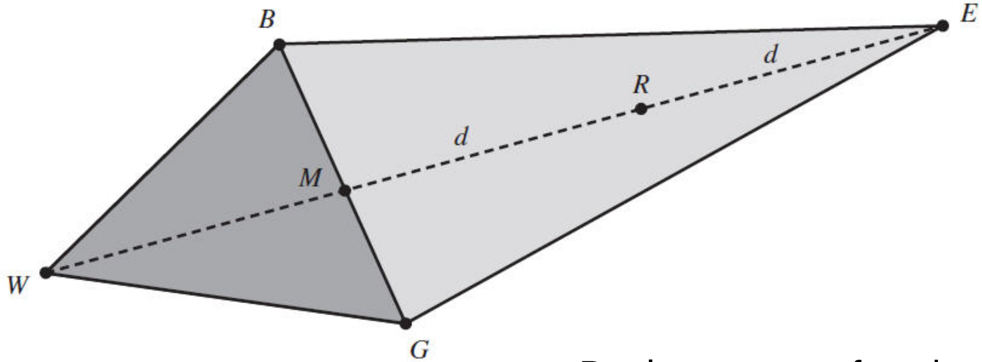
The method uses reflection, expansion, contraction, and shrinkage to generate the new vertices.

Optimization theory and methods

Methods for calculating new vertices

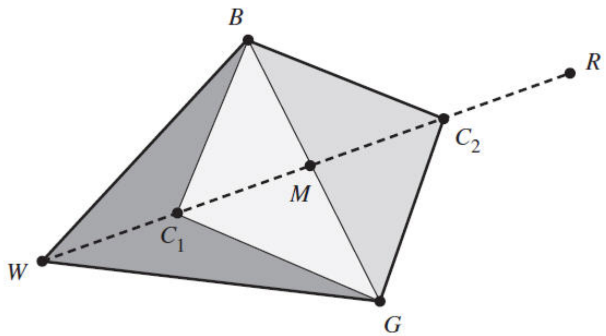


Reflection

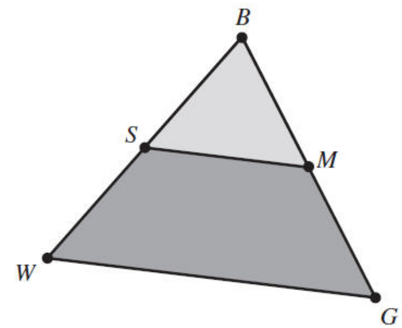


Expansion

B = lowest error function
G = middle error function
W = highest error function



Contraction

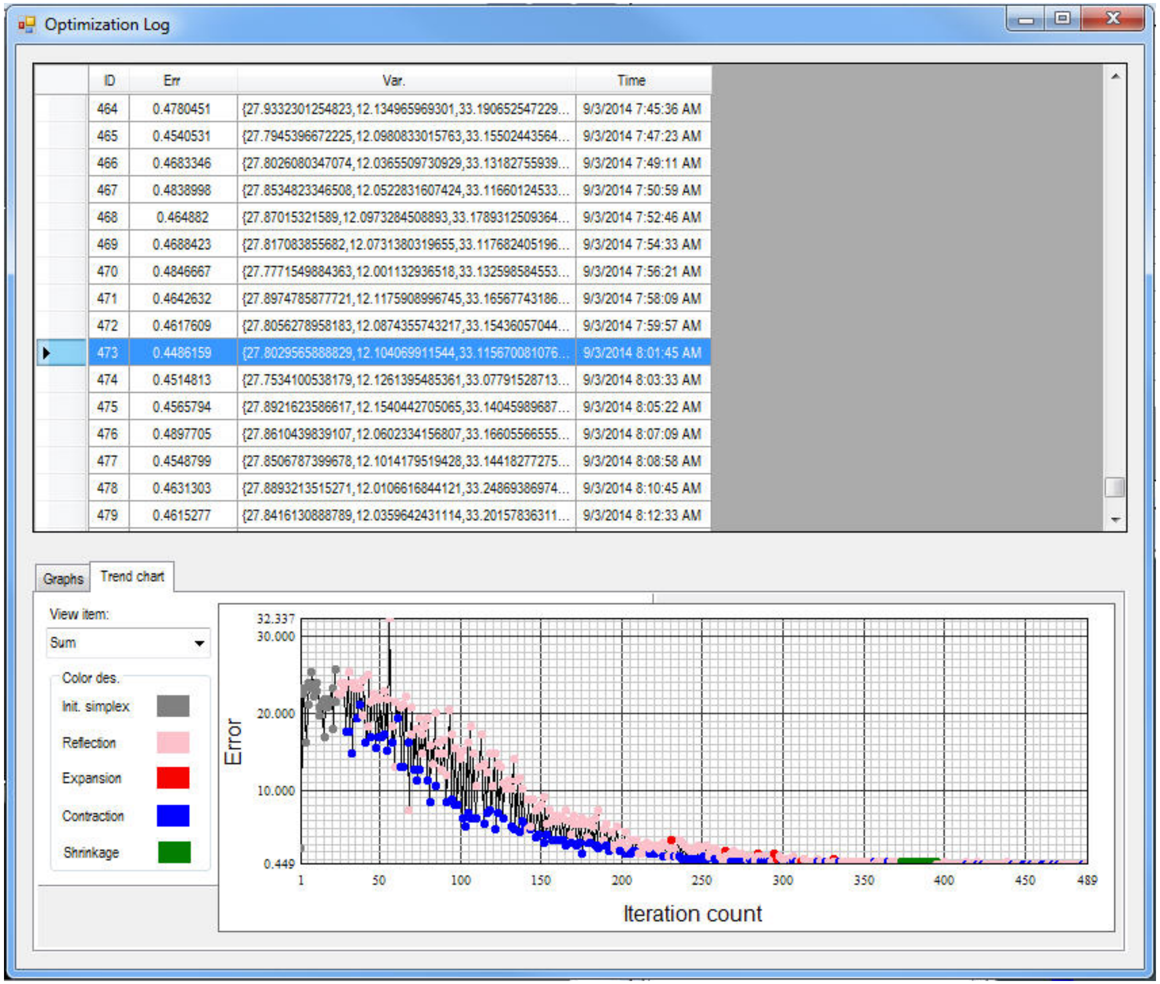


Shrinkage

Source: Numerical Methods Using Matlab 4th Edition

Optimization theory and methods

Optimization Log showing Downhill-Simplex operations – 11 variables



Optimization theory and methods

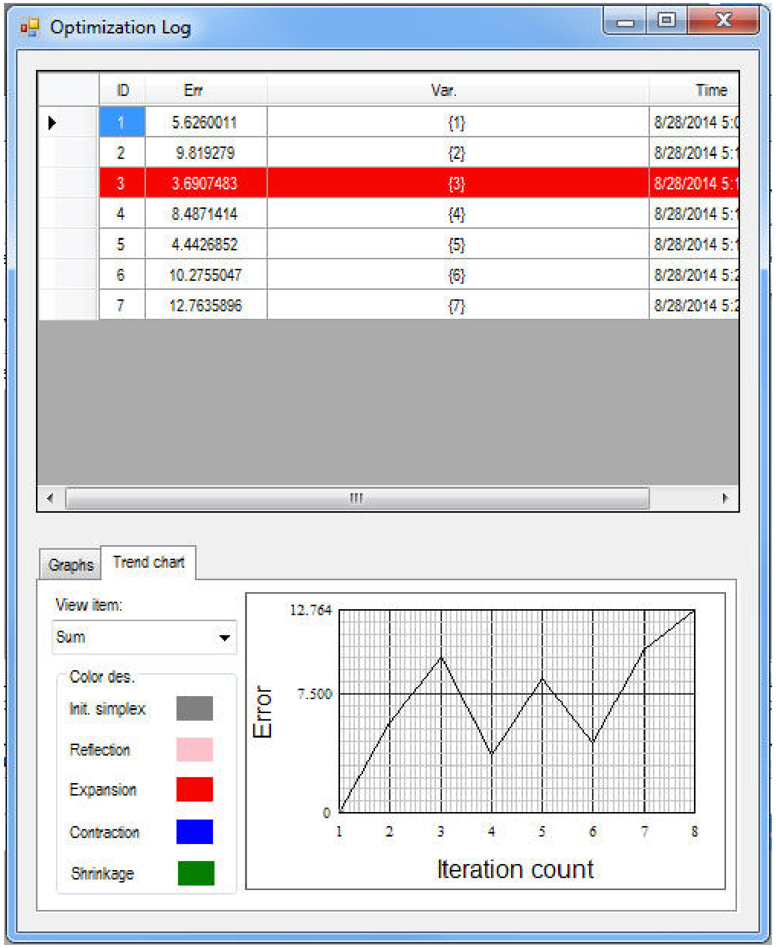
Variable Scanning method

- The Variable Scanning method is used to scan or step through all possible variable combinations
- Scanning the range of a variable to find a suitable starting condition for the Downhill-Simplex optimization method
- Moving a variable in fixed interval steps to monitor results
- Tolerancing
- Finding the best surface or material property for a given application by automatically scanning through all properties in a catalog and showing the simulation results for each

TracePro

Optimization theory and methods

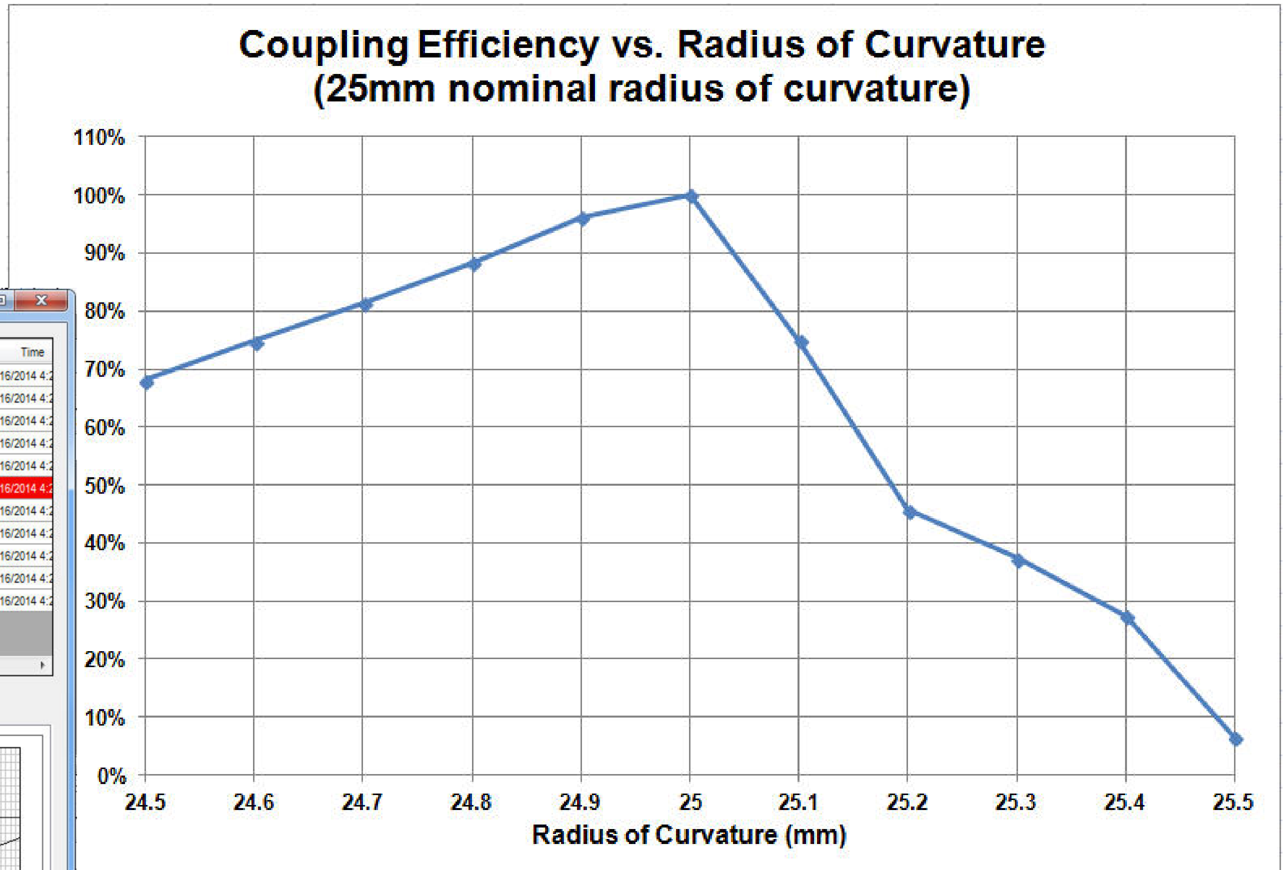
Variable Scanning results examples – Selecting the best result by scanning though a catalog of diffuser properties

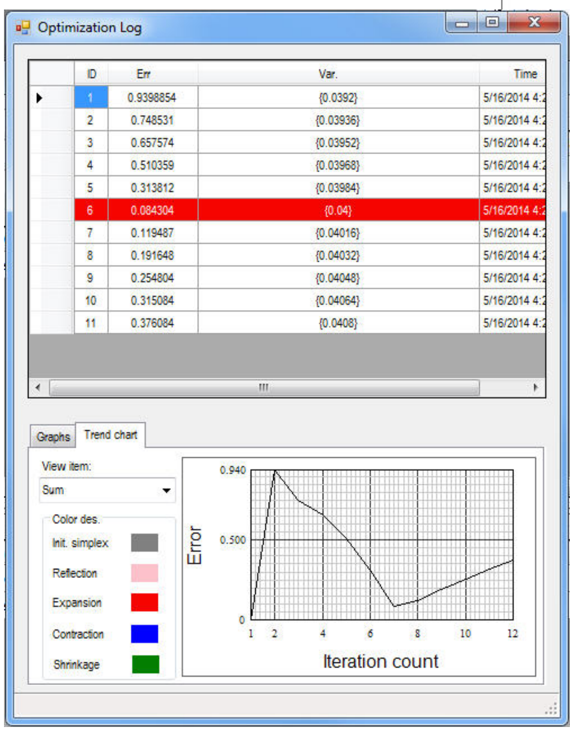


TracePro

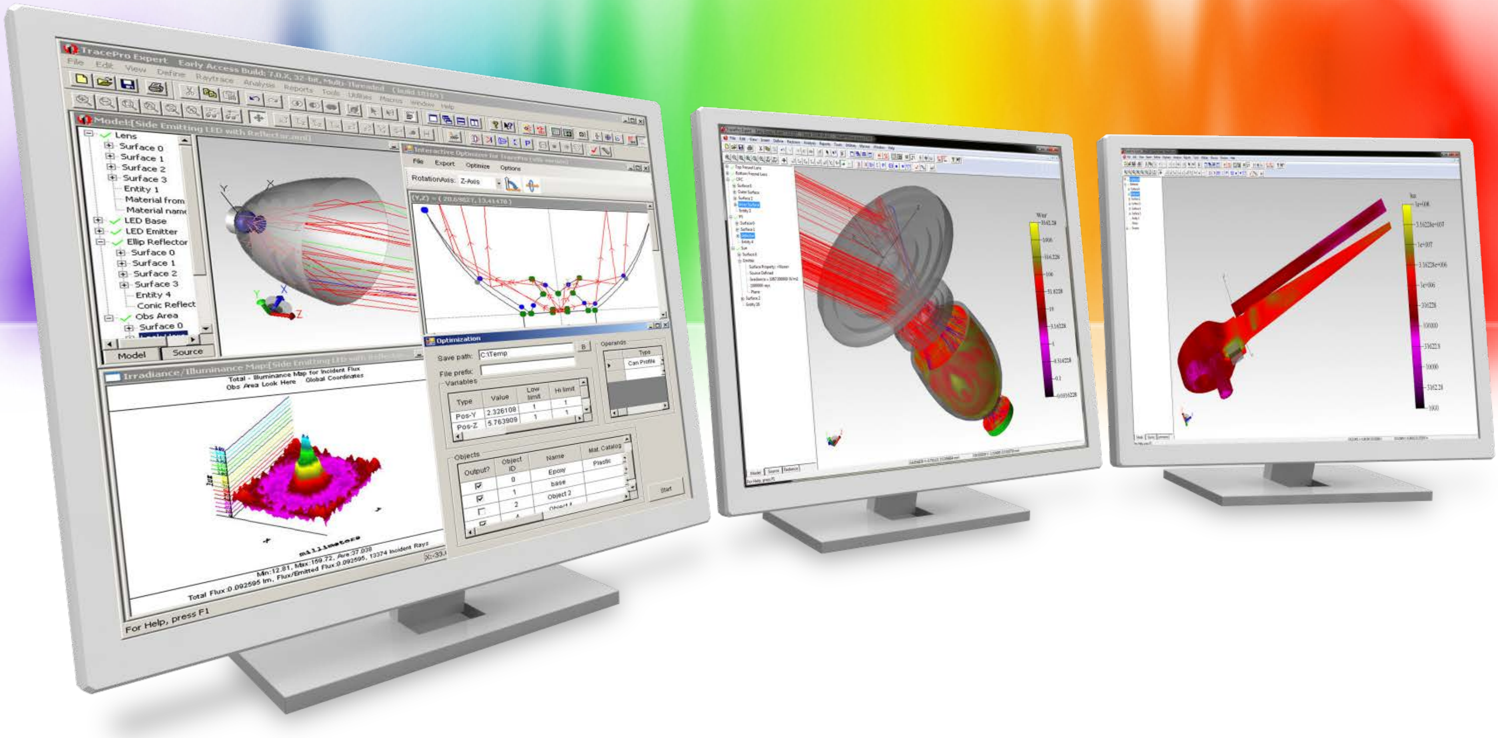
Optimization theory and methods

Variable Scanning results examples – Tolerancing example





TracePro

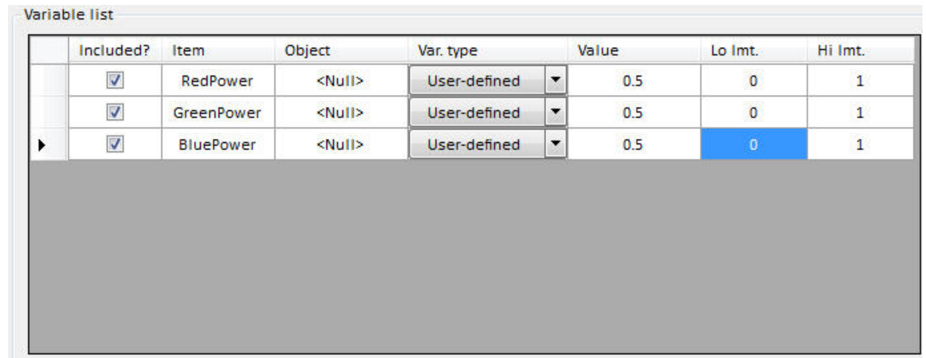


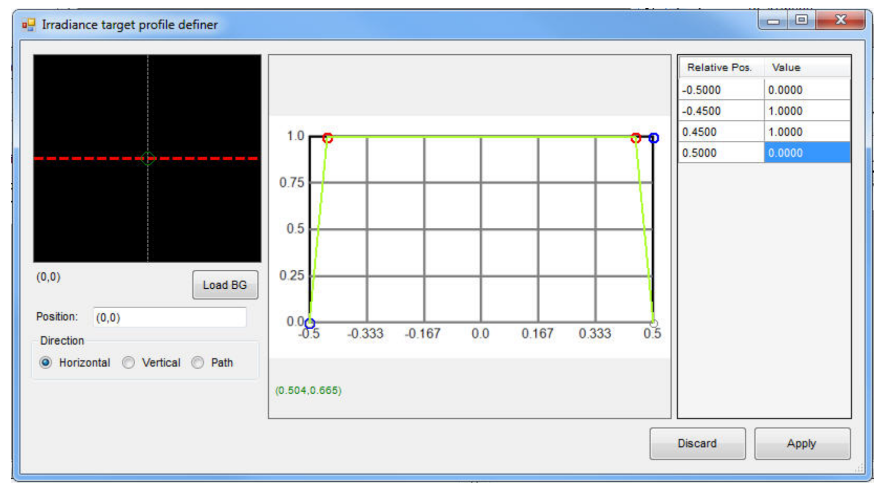
Optimization Parameters and Settings

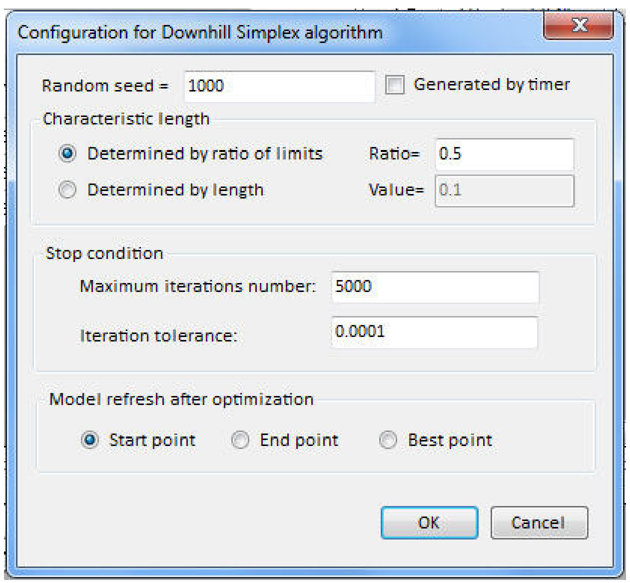
Optimization parameters and settings

Optimization parameters and settings

- Variables
- Optimization operands
- Optimization settings







TracePro

Optimization parameters and settings

Variables are the parameters that are allowed to change during the optimization process. These can include:

- Control point position in 1, 2, or 3 dimensions
- Curvature
- Conic Constant
- Rotational Angle
- Distance
- Separation
- Pick-ups
- Custom or User Defined

When the variable is defined the range of the variable is specified. The range is how much the variable will be allowed to “move” during the optimization process. The range of the variable can be set to limit or control the size of the optical element.

TracePro

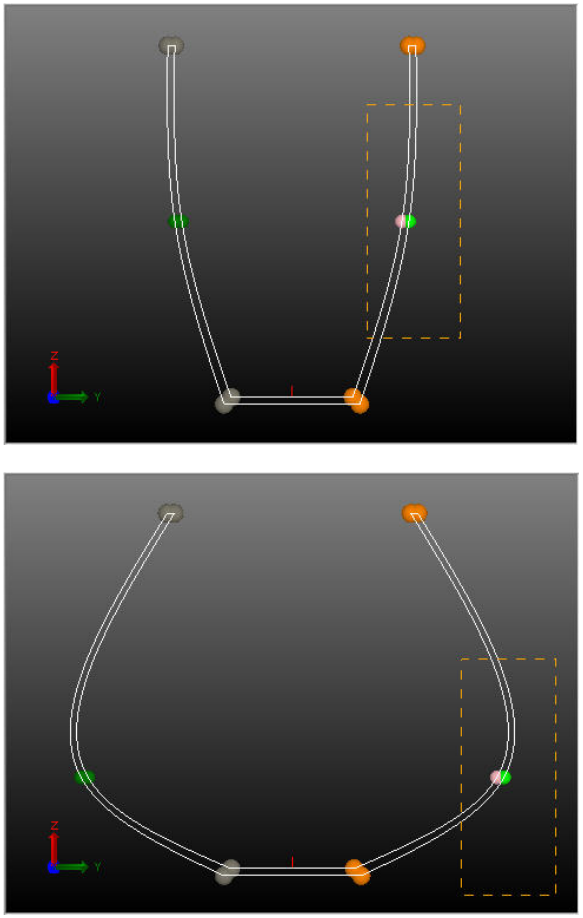
Optimization parameters and settings

Variables can be Absolute, Relative, or Pick-ups

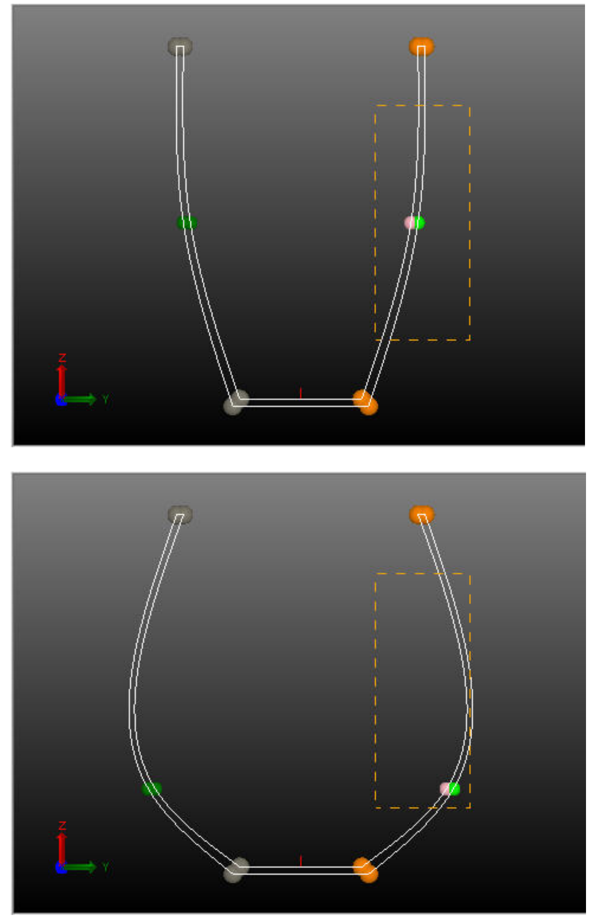
- Absolute variables are defined using absolute or global coordinates of the range of variables motion. If the original variable's location is changed, the range will remain fixed.
- Relative varies are defined relative to current variable's location, so if the variable is moved, the variable range will move with the variable.
- Pick-ups define the position and movement of a variable based on the value of another variable. For example, a variable can be defined as a Pick-up to maintain a constant thickness in a material, or a specific separation between 2 components.

Optimization parameters and settings

Absolute vs. Relative variable examples



Relative Variable

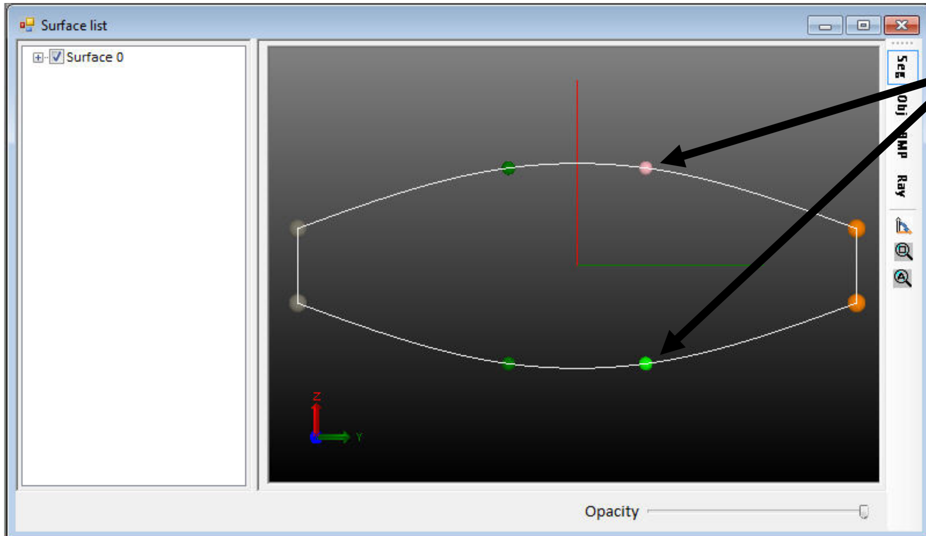


Absolute Variable

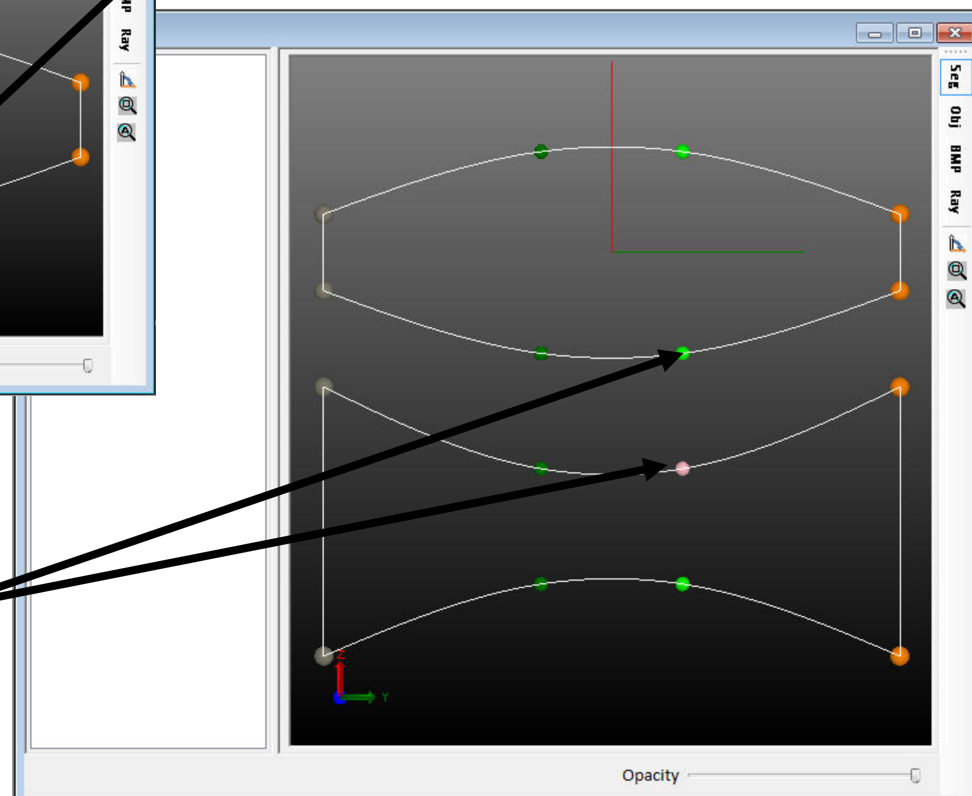
TracePro

Optimization parameters and settings

Pick-up variable examples



Use to make both sides of the lens the same radius of curvature



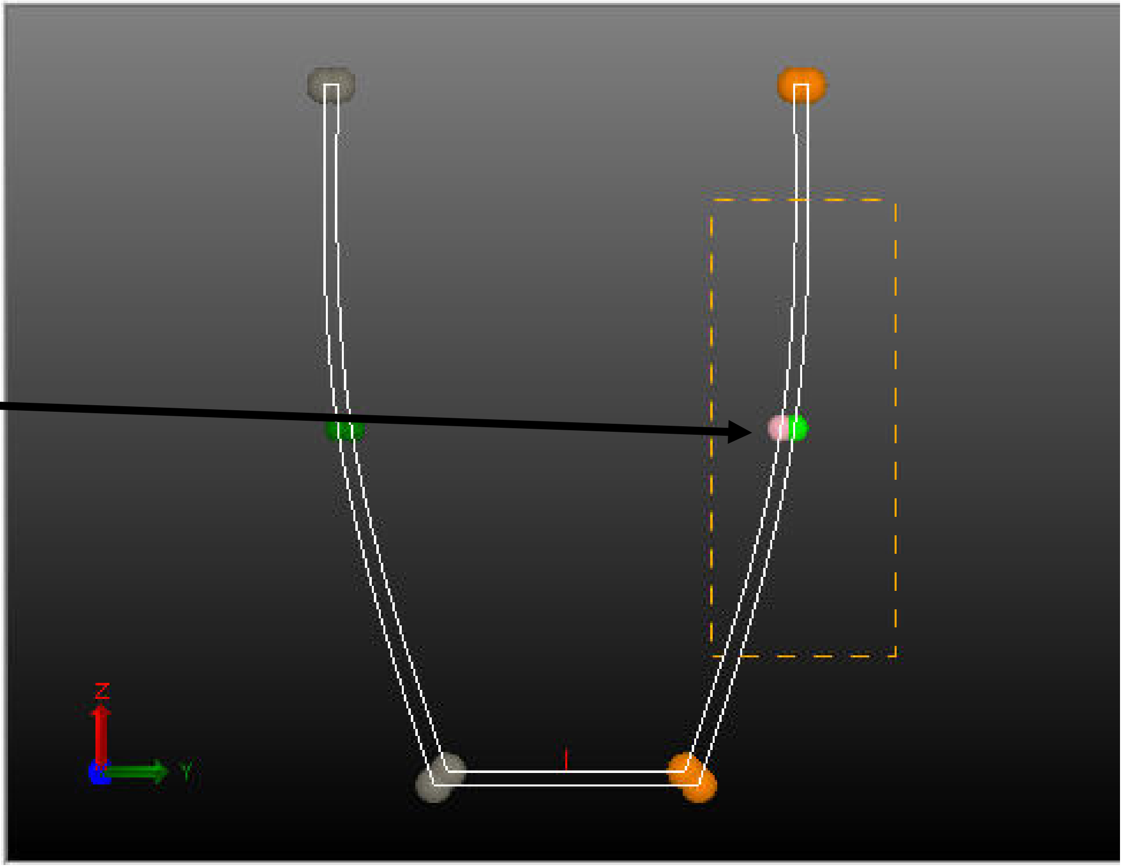
Use to maintain a constant spacing between 2 surfaces/components

TracePro

Optimization parameters and settings

Pick-up variable examples

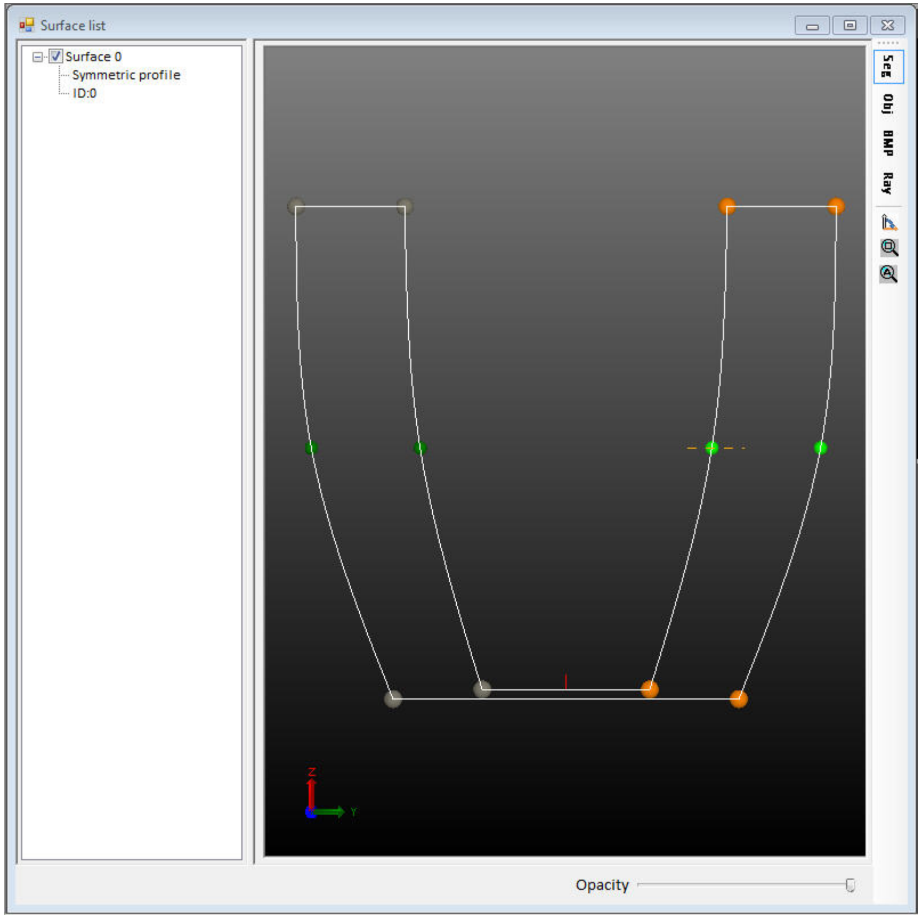
Use to maintain a constant wall thickness in a reflector

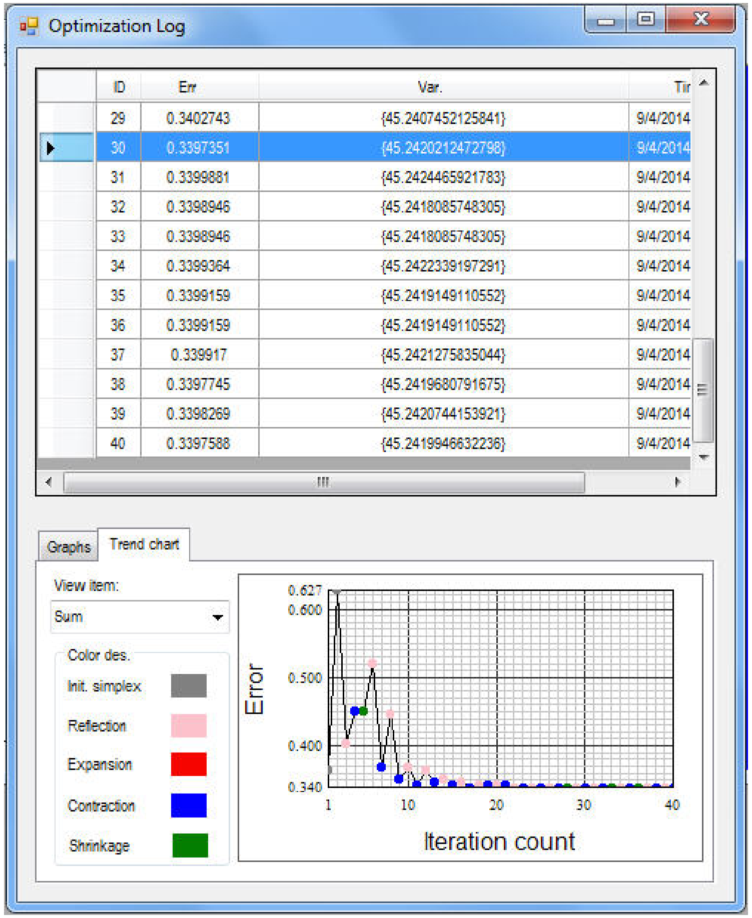


TracePro

Optimization parameters and settings

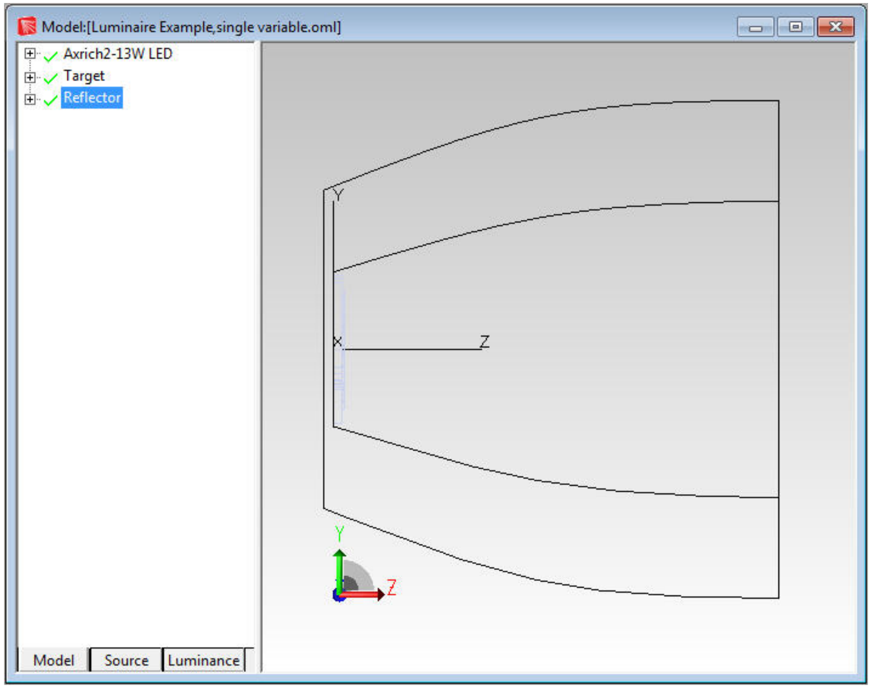
Number of variables to use: Not enough variables example

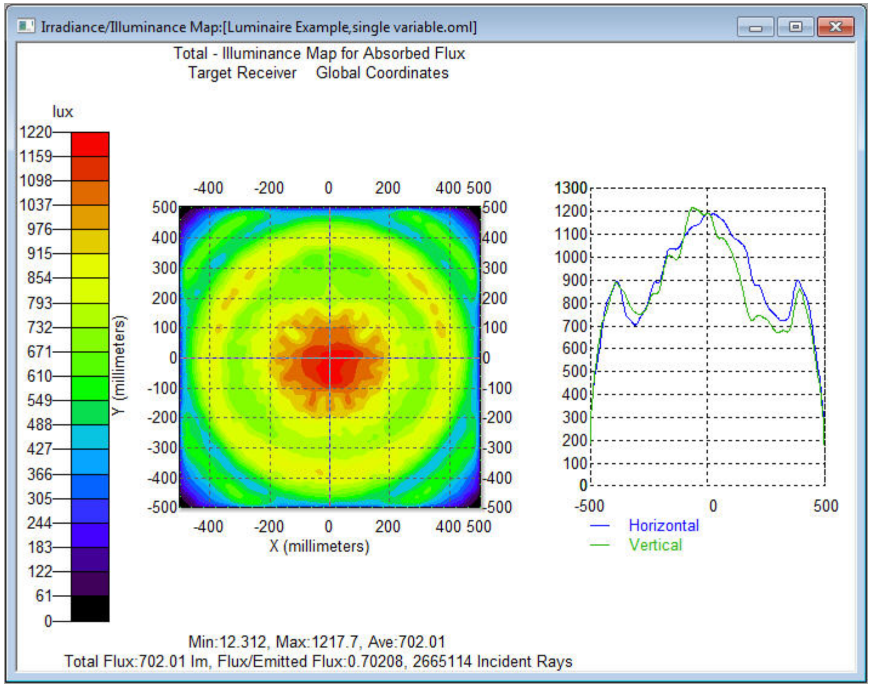




Optimization parameters and settings

Number of variables to use: Not enough variables example

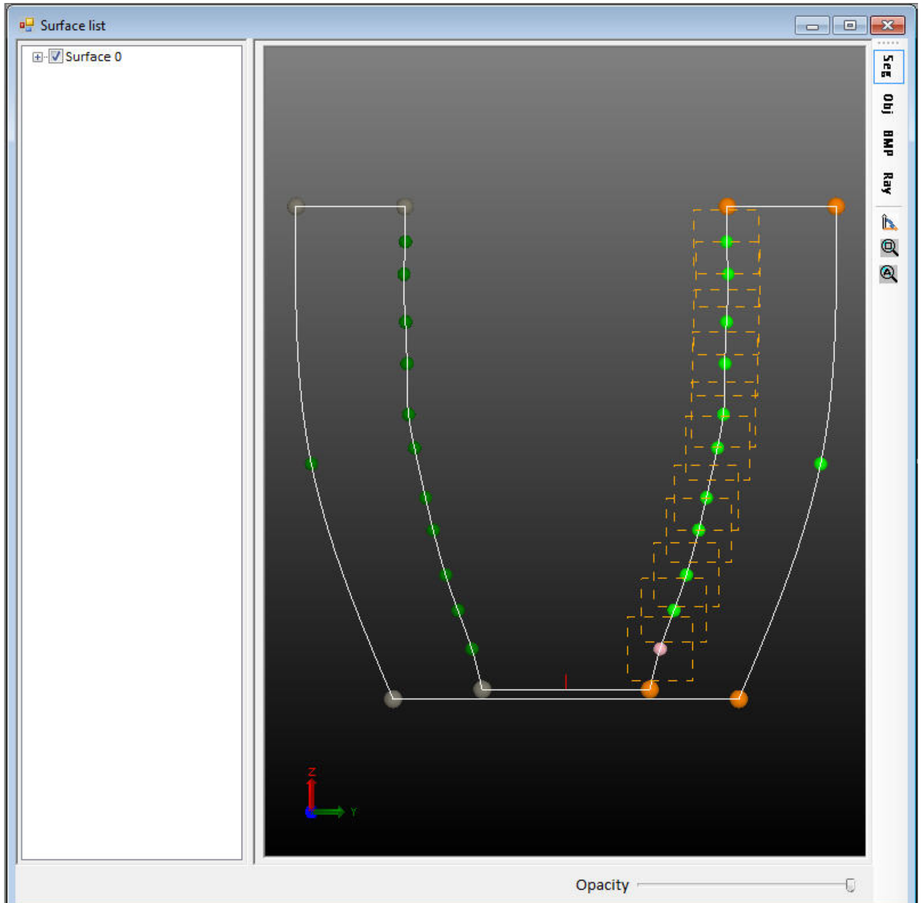


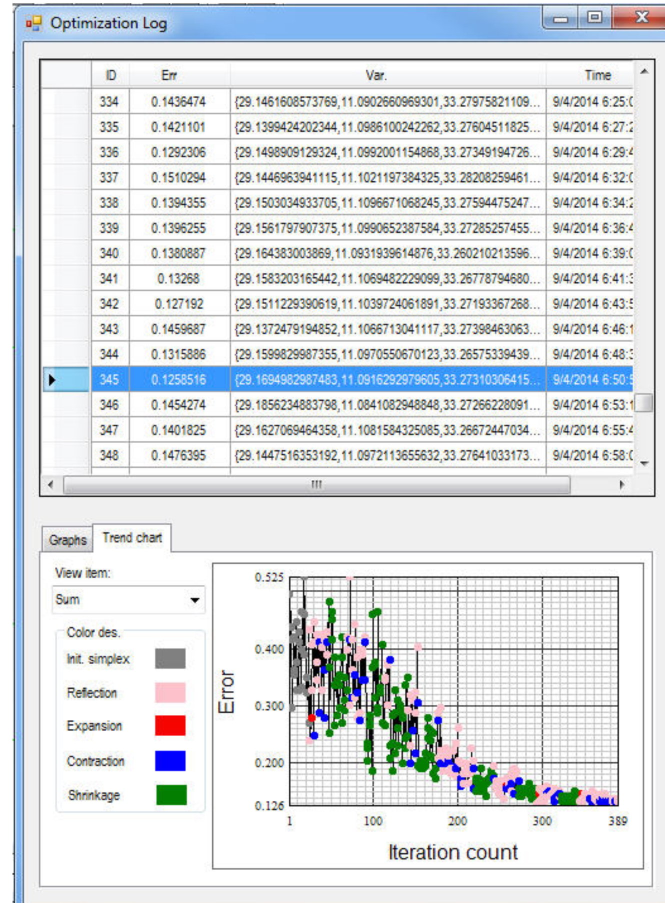


TracePro

Optimization parameters and settings

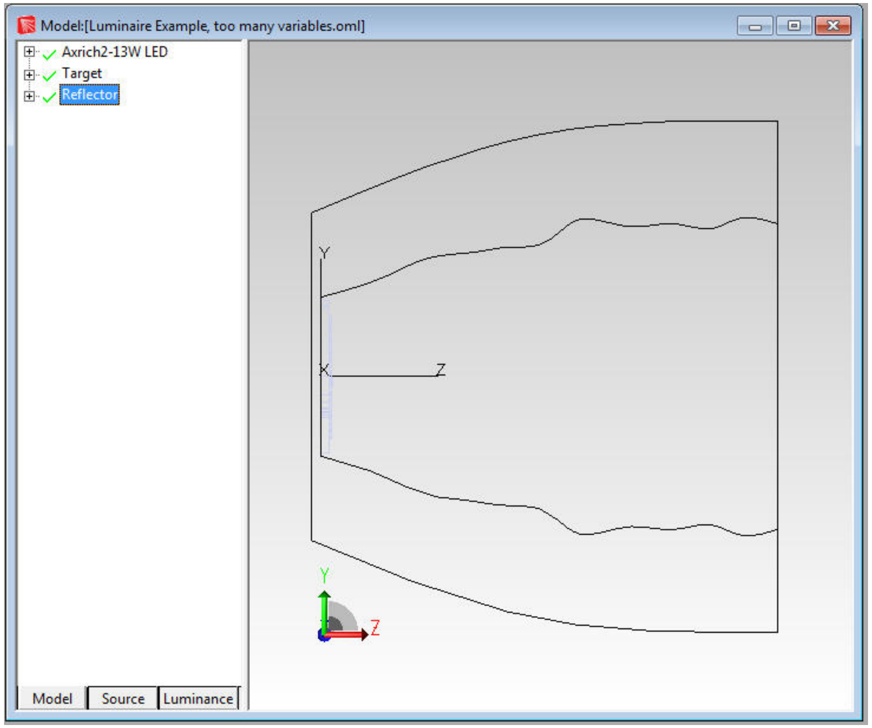
Number of variables to use: Too many variables example

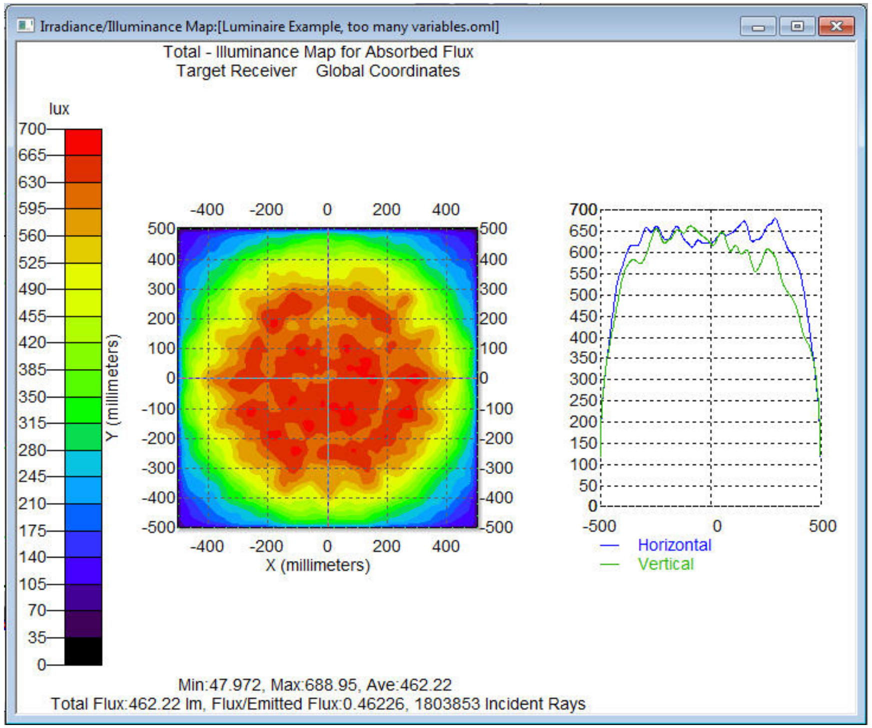




Optimization parameters and settings

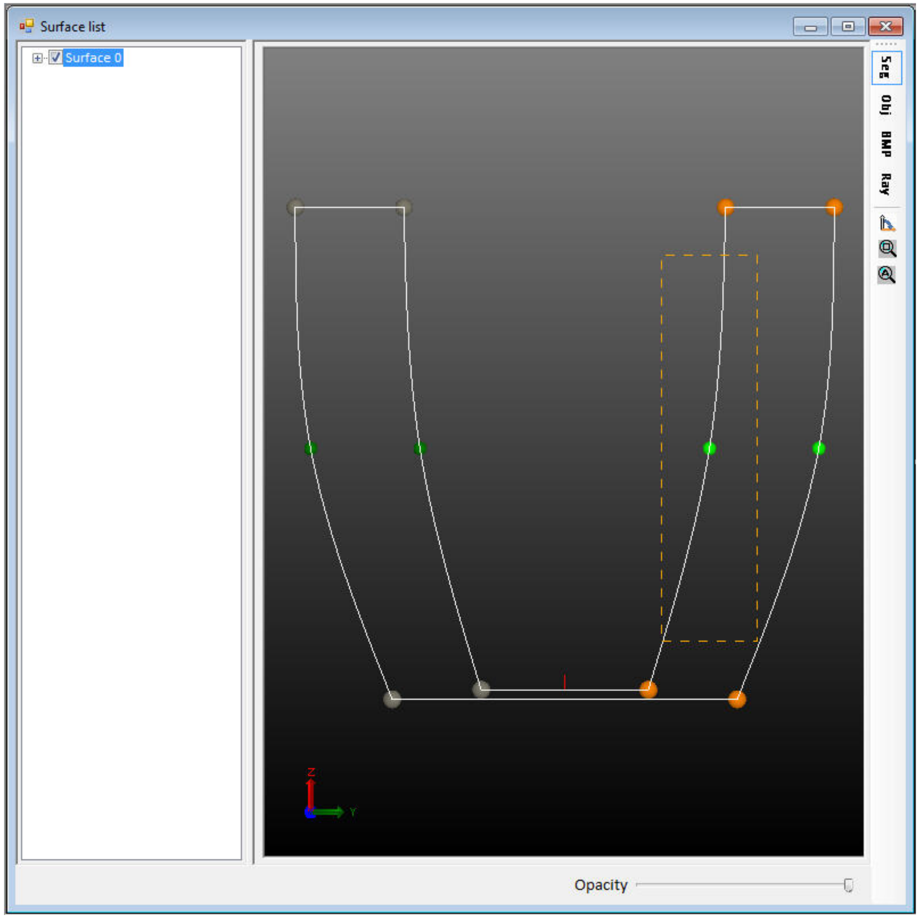
Number of variables to use: Too many variables example

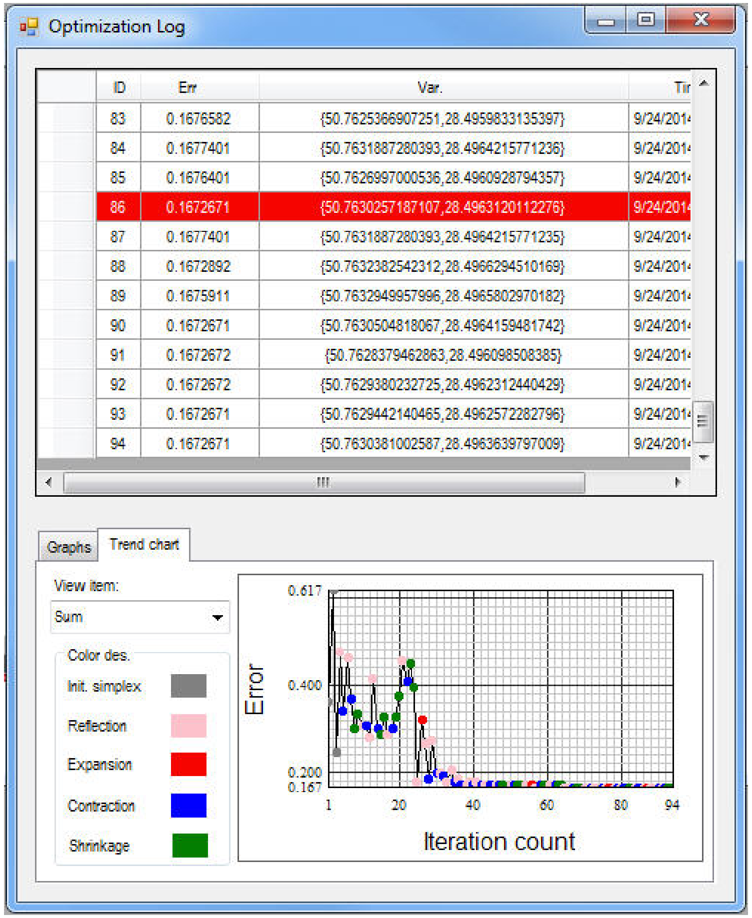




Optimization parameters and settings

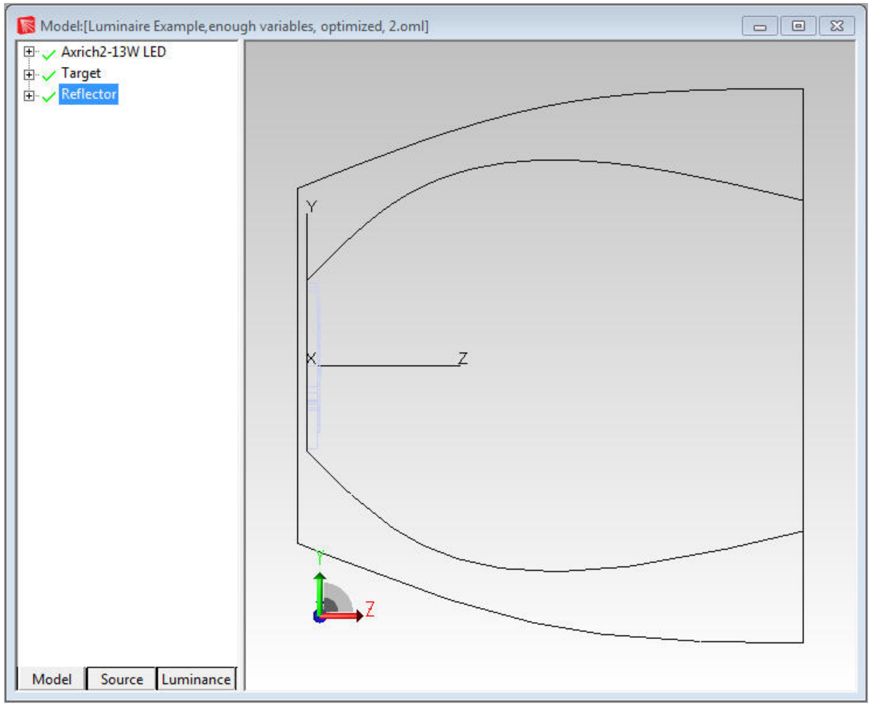
Number of variables to use: Adequate number of variables example

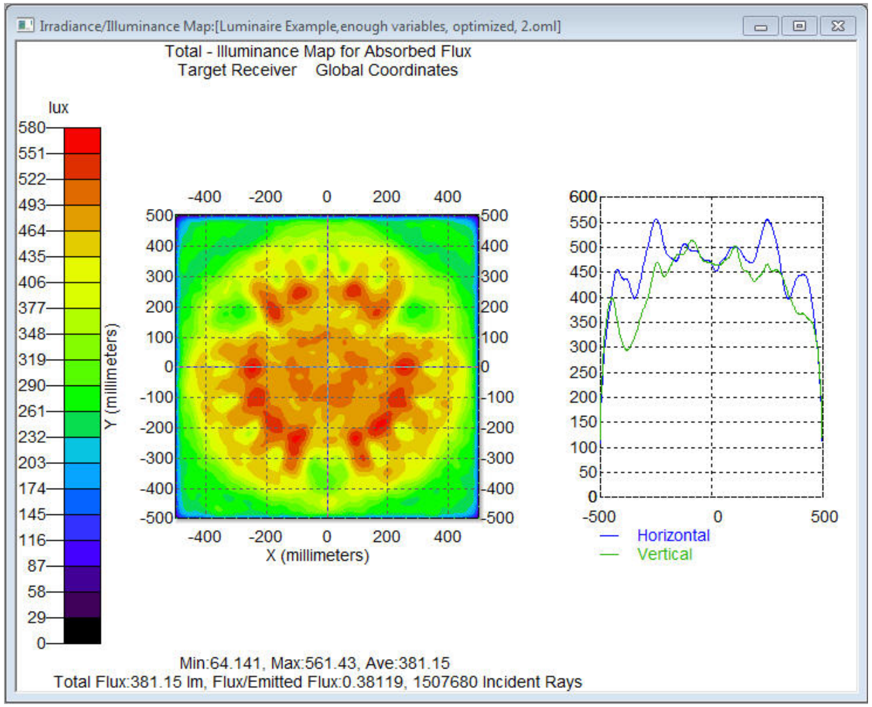




Optimization parameters and settings

Number of variables to use: Adequate number of variables example





TracePro

Optimization parameters and settings

Optimization Operands

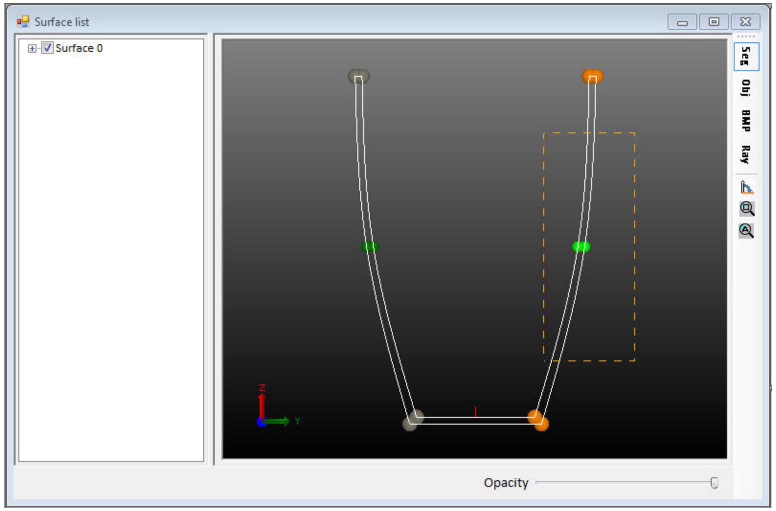
Optimization operands are used to define the target or goal of the optimization process. Some examples include:

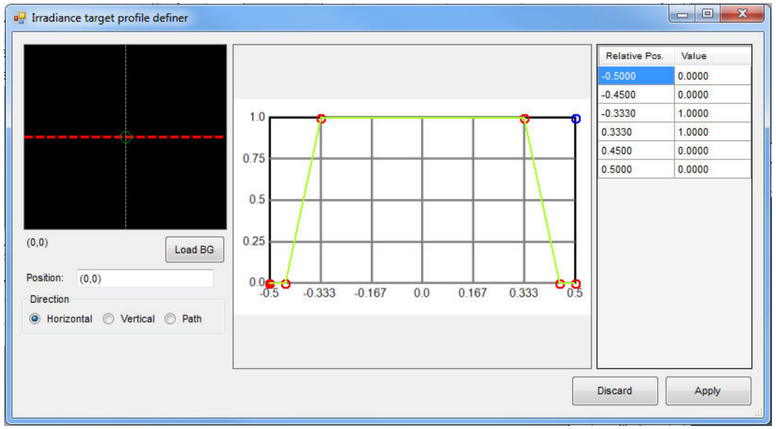
- Flux
- CIE color coordinates
- Irradiance
- Irradiance Profiles
- Intensity
- Candela or Intensity Profiles
- Uniformity
- Beam Width
- User Defined or Custom

TracePro

Optimization parameters and settings

Varying the starting point of the optimization process – Initial design and optimization goal

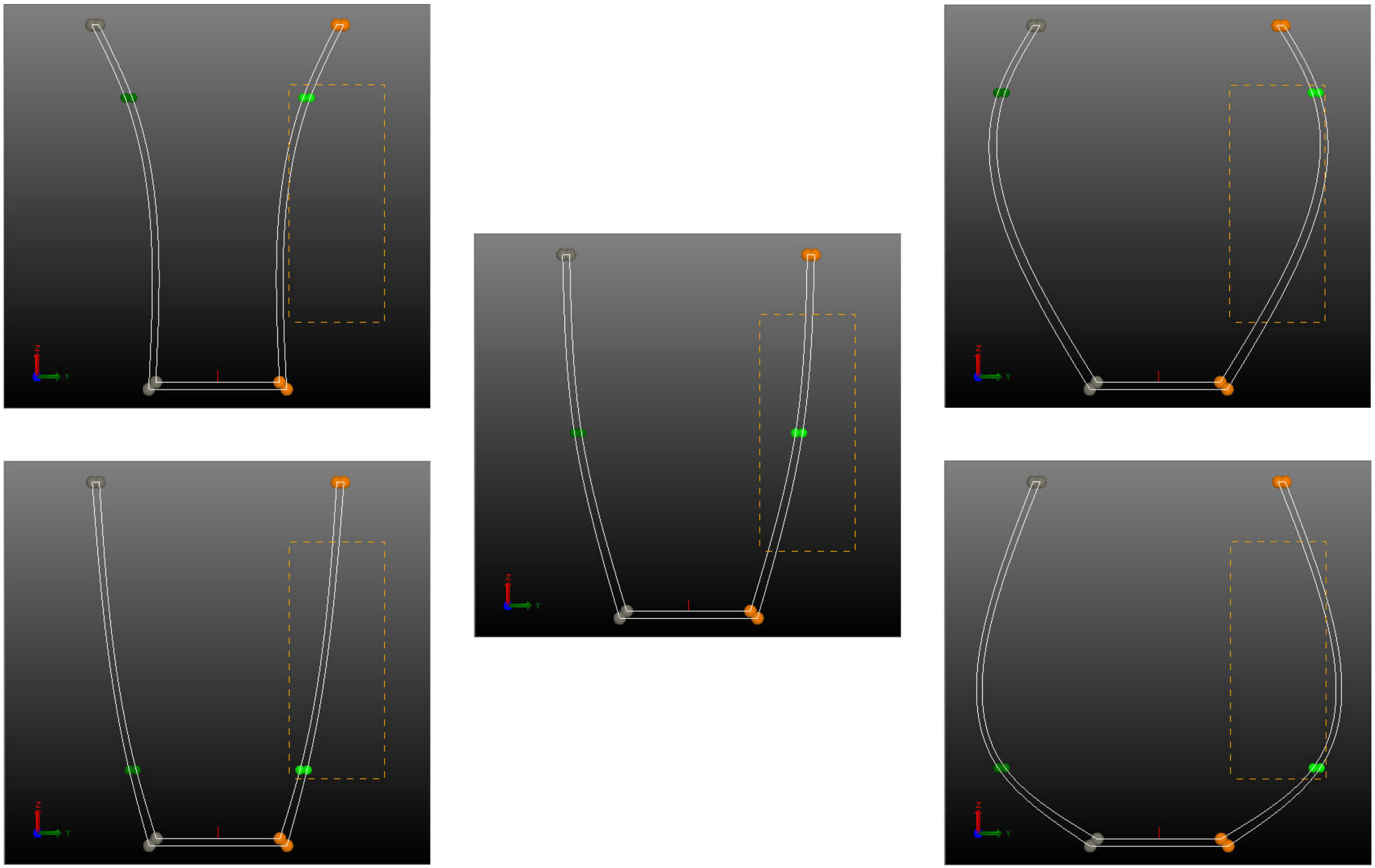




TracePro

Optimization parameters and settings

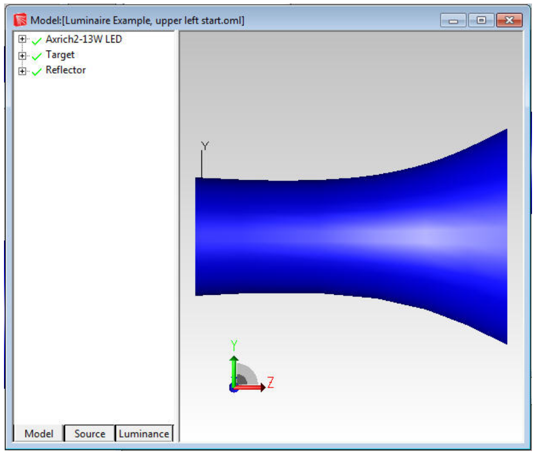
Varying the starting point of the optimization process

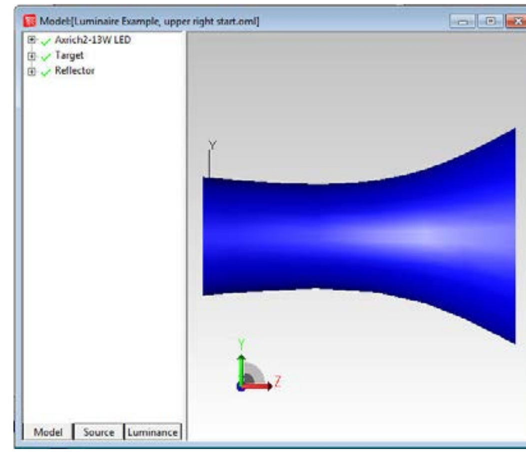


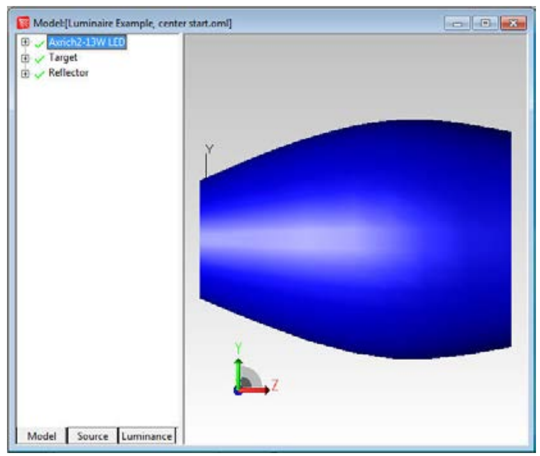
TracePro

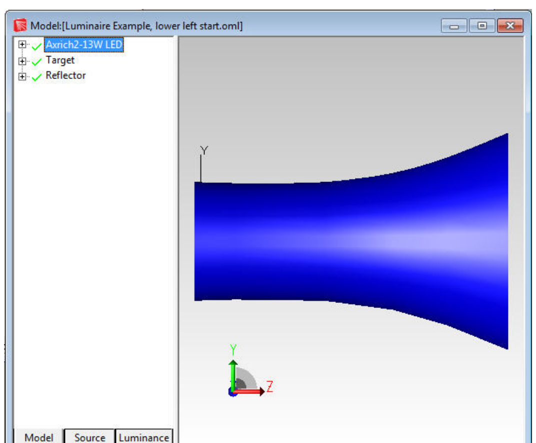
Optimization parameters and settings

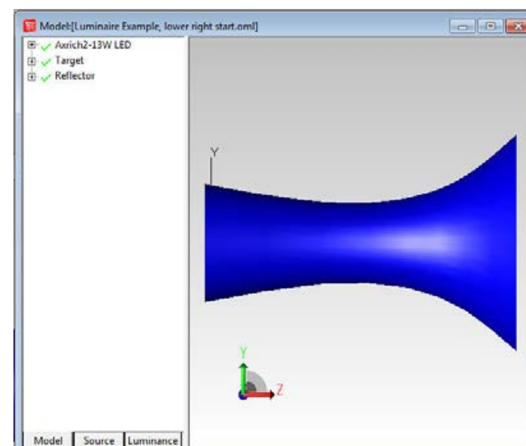
Varying the starting point of the optimization process







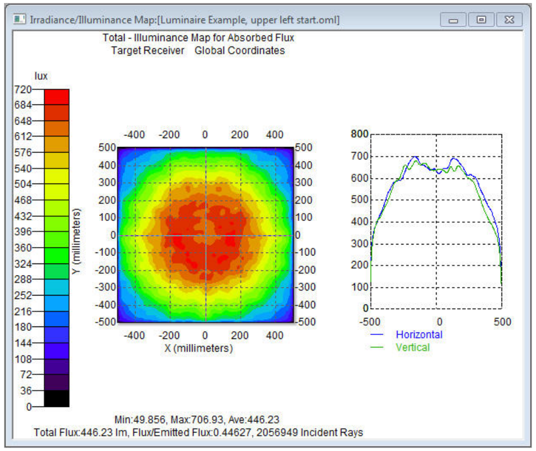




TracePro

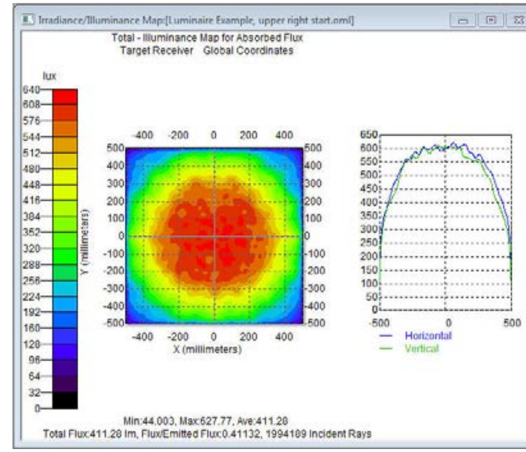
Optimization parameters and settings

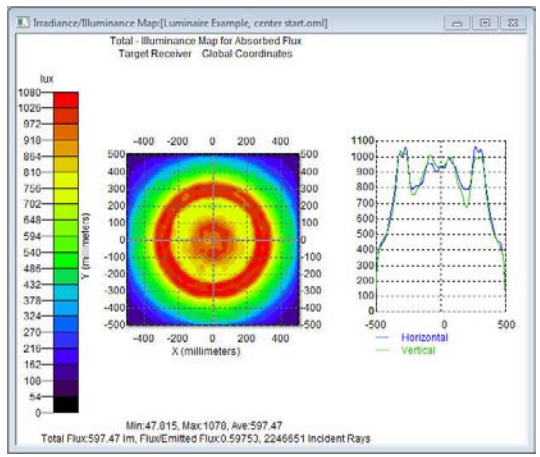
Varying the starting point of the optimization process



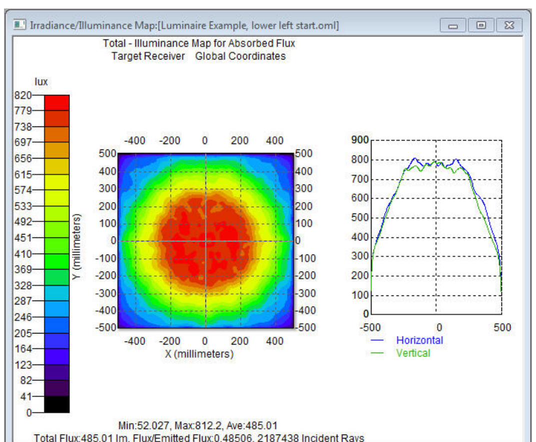
44.6%

41.1%



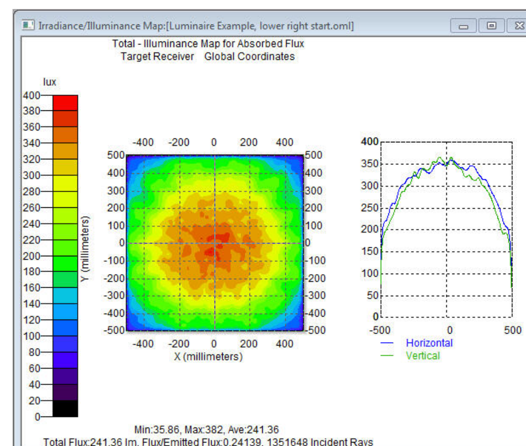


59.7%



48.5%

24.1%



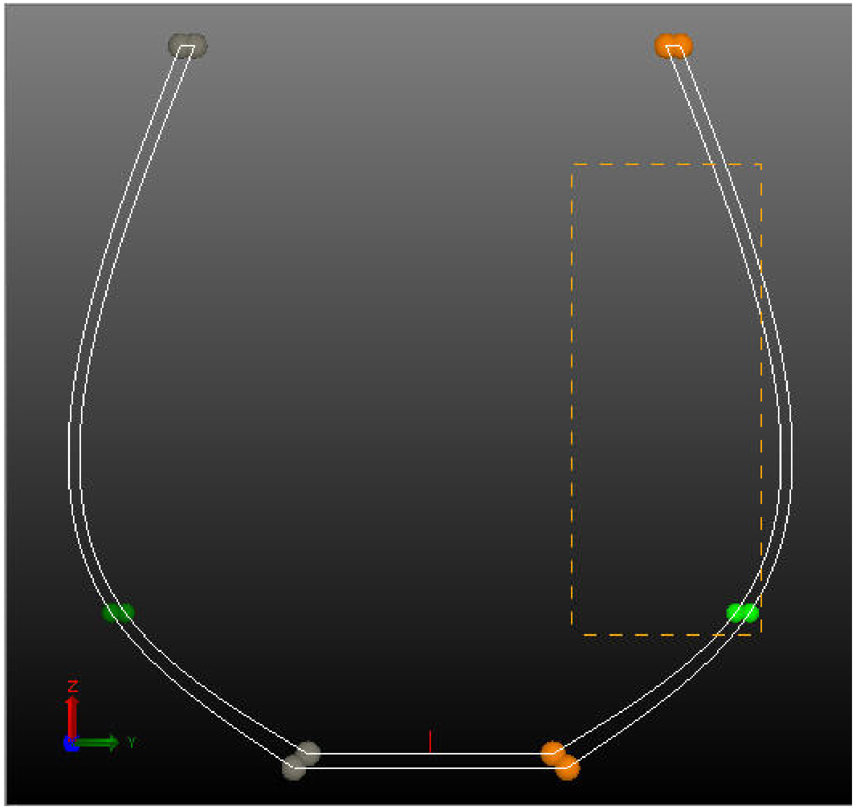
TracePro

Optimization parameters and settings

Improve the results by adding a second optimization target – use the lower right corner starting condition from the previous example

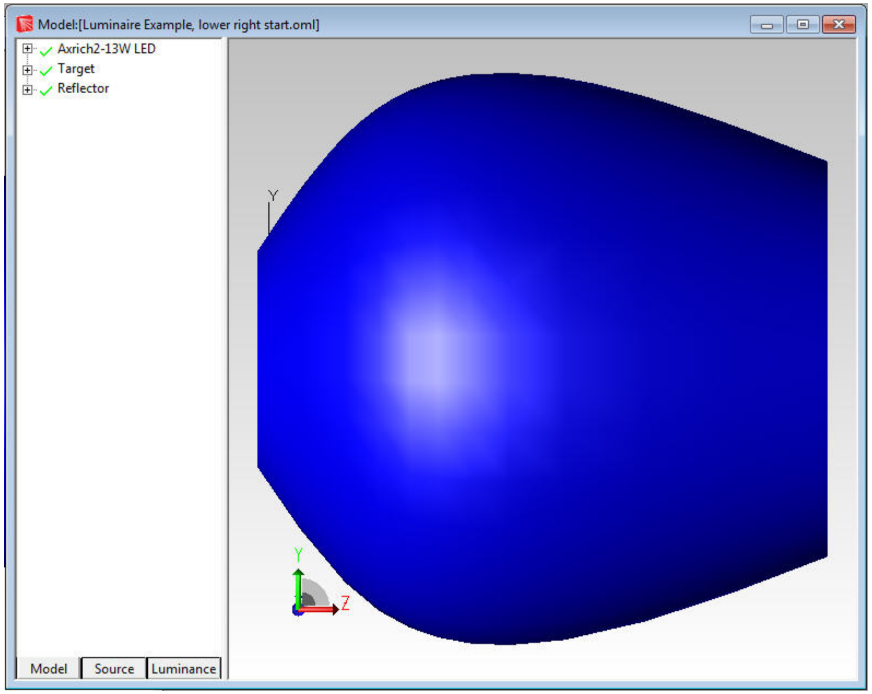
Keep the Irradiance Profile target from the previous examples, but add an additional Flux operand with a target goal of 750 lumens.

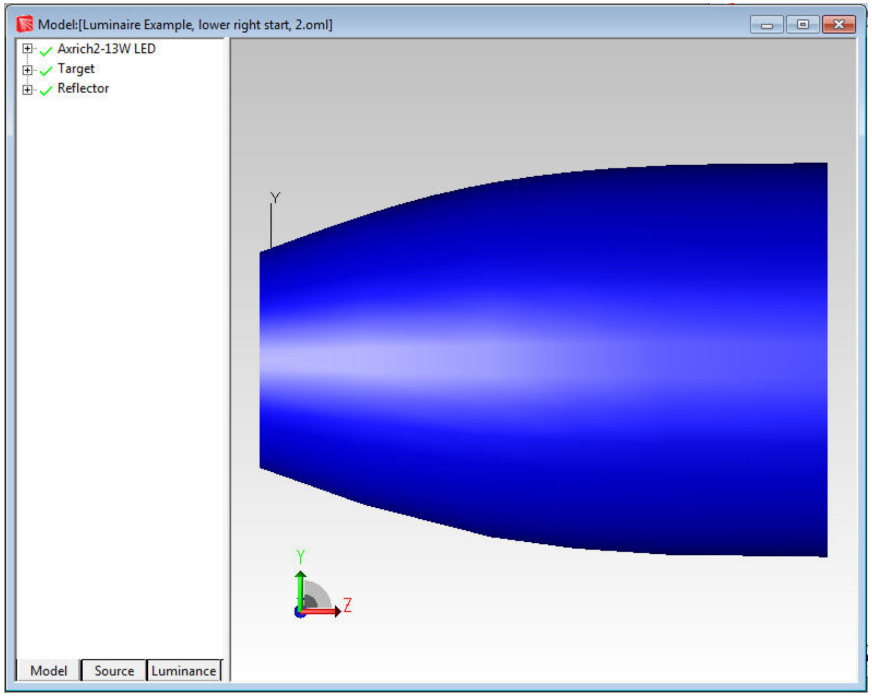
The two operands can be weighted so that contribution of each can be varied. In this case the were set to have similar contributions to the overall error function.



Optimization parameters and settings

Improve the results by adding a second optimization target





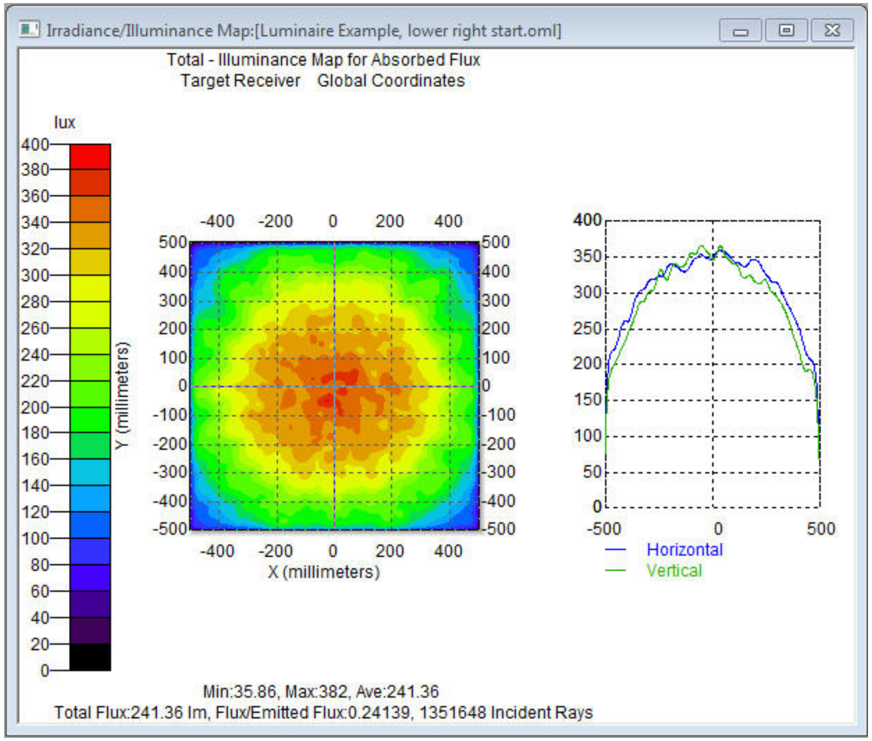
Initial Optimization

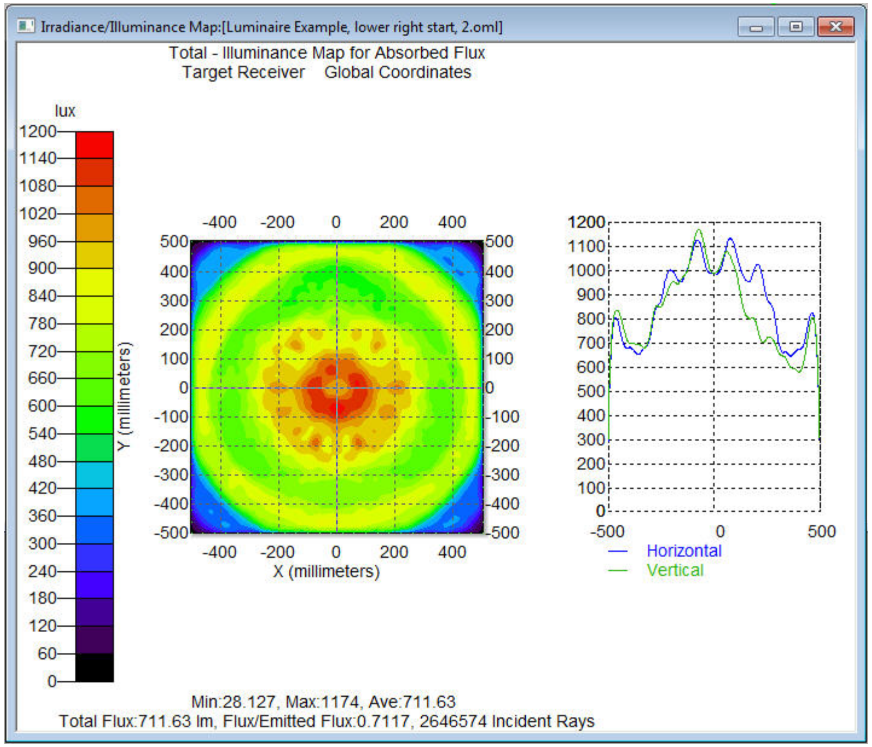
After adding second optimization operand

TracePro

Optimization parameters and settings

Improve the results by adding a second optimization target





Initial Optimization

After adding second optimization operand

TracePro

Optimization parameters and settings

Optimization Settings

The optimization settings can be used to control how the optimization process runs. Changes in these settings can sometimes result in improvements in the final design. Wrong choices can lead to poor results.

Examples of optimization settings that can be varied include:

- Optimization type
- Characteristic Length – Ratio of Limits and Length
- Stopping conditions
- Number of rays traced
- Accurate source model – geometric or rayfile

Optimization parameters and settings

Optimization Method

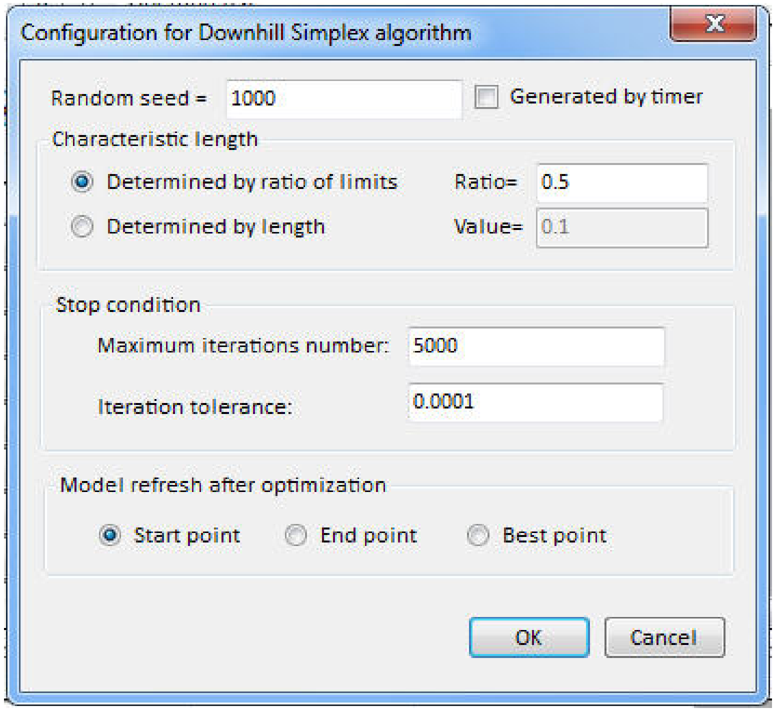
Choose the optimization method that best suits the application.

•Optimizing geometry or position – choose the Downhill-Simple (Nelder-Mead) method and allow the optimizer to search through a range of variable.

Optimization parameters and settings

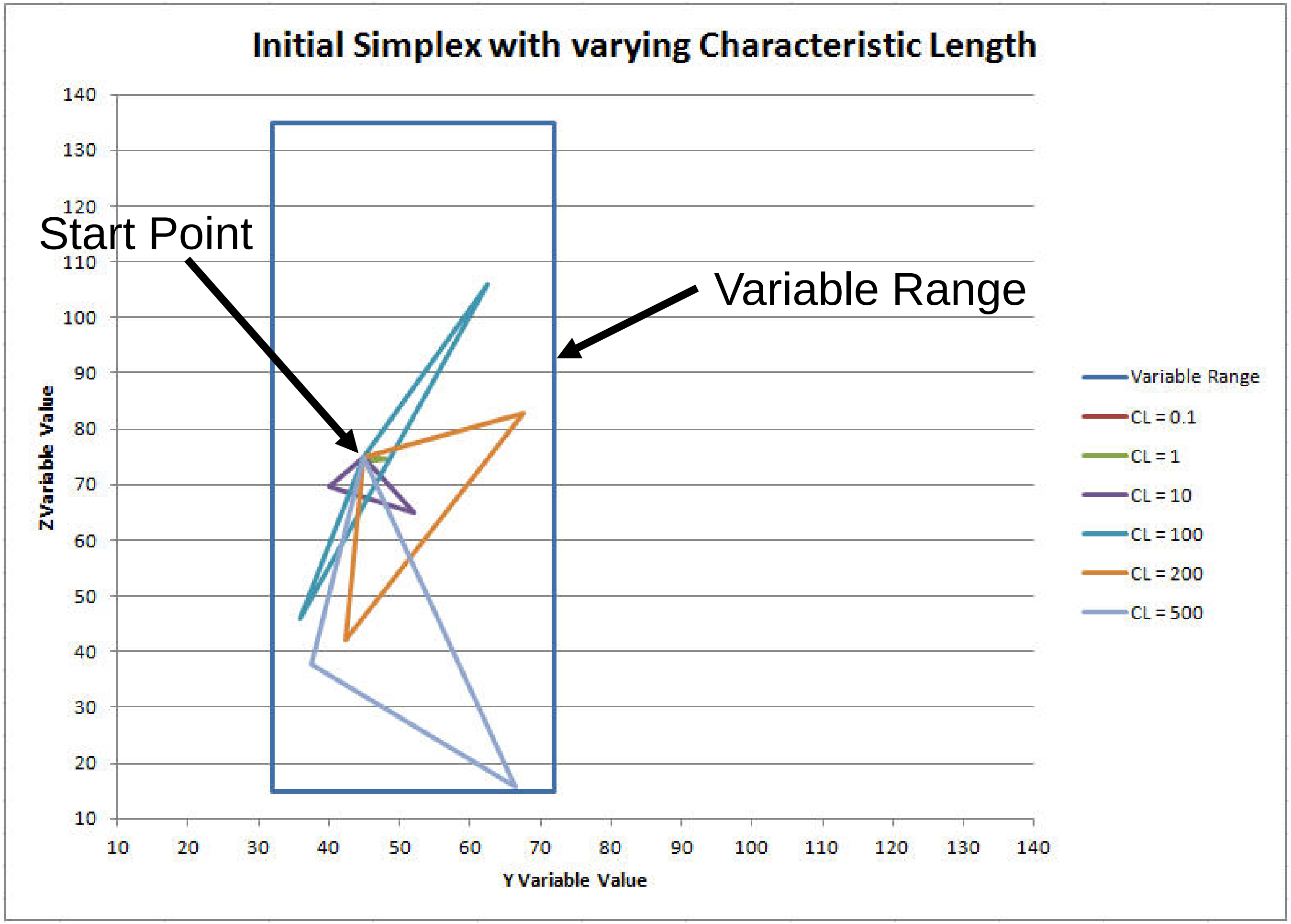
Characteristic Length

The Characteristic Length is an estimate of the size of the solution space for an optimization process. It is used when defining the initial simplex. Each vertex of the initial simplex is a variable set that is a function of the Characteristic Length and a random number.



Optimization parameters and settings

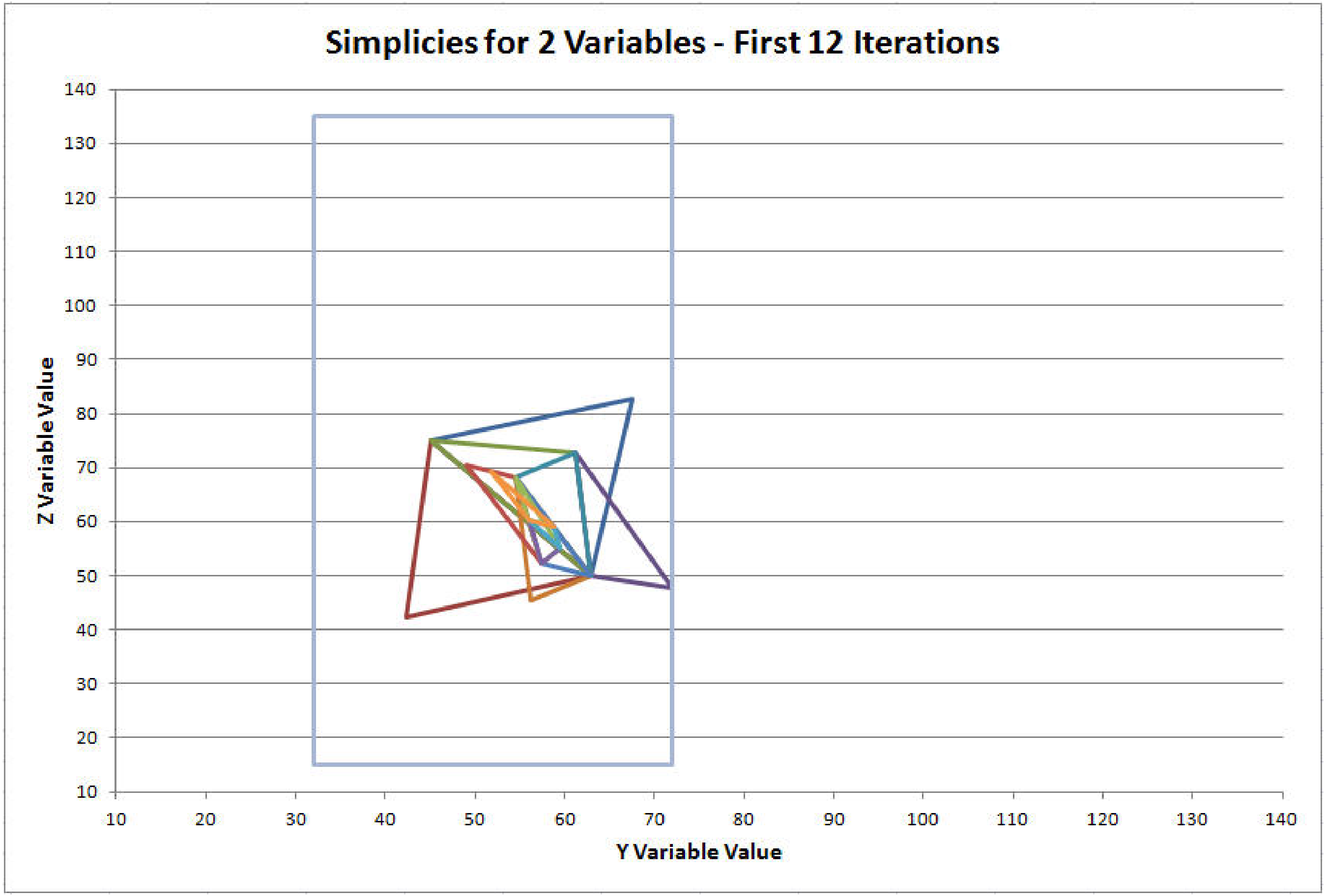
Different Characteristic Length Examples



TracePro

Optimization parameters and settings

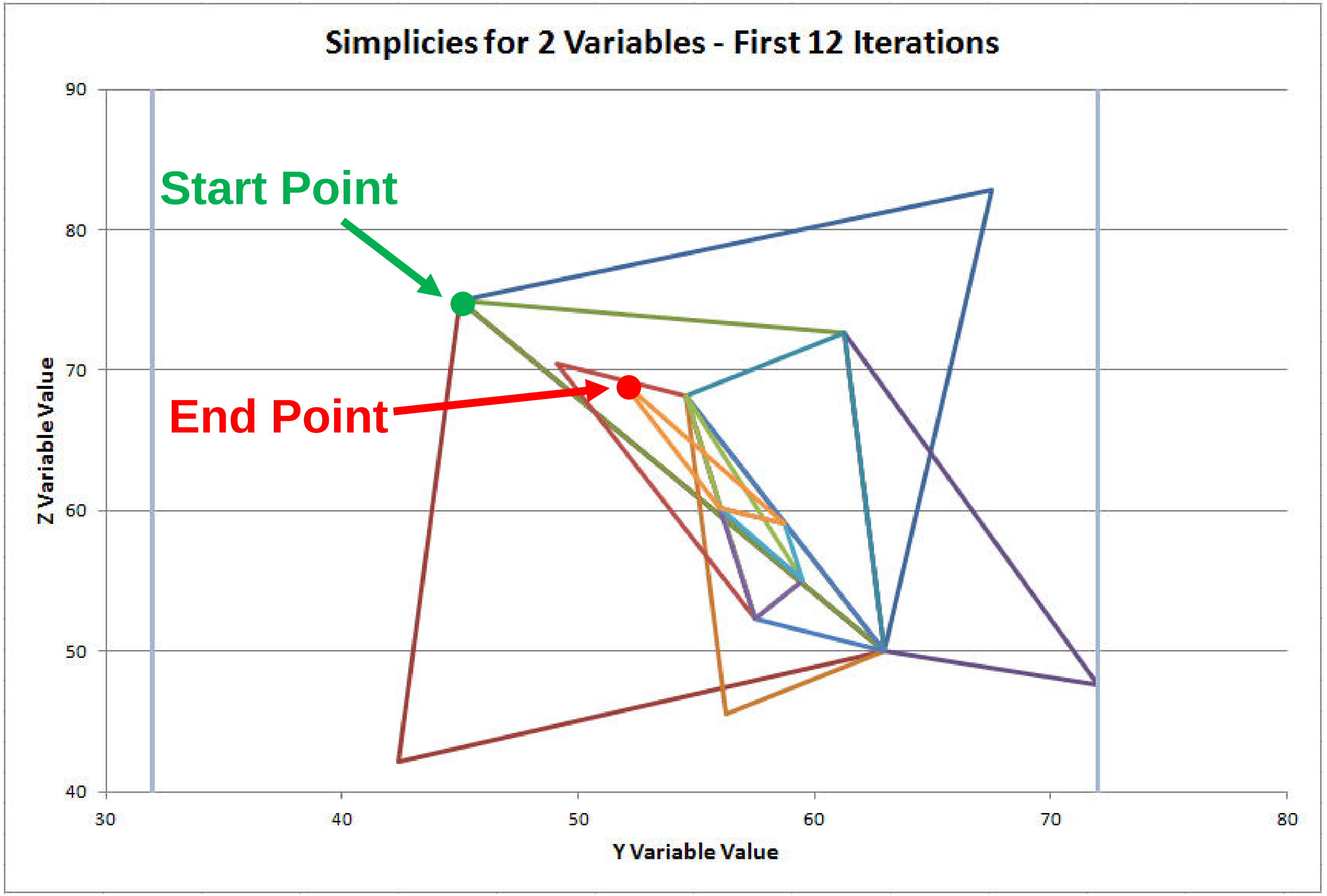
2 Variable Simplex – Iterations



TracePro

Optimization parameters and settings

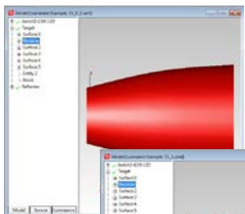
2 Variable Simplex – Iterations



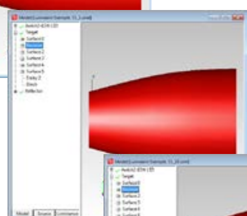
TracePro

Optimization parameters and settings

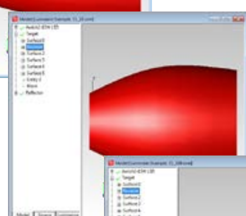
Different Characteristic Length Examples



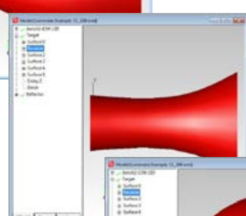
Characteristic Length = 0.1



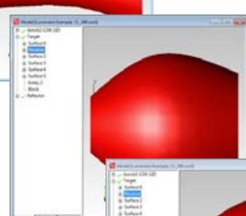
Characteristic Length = 1



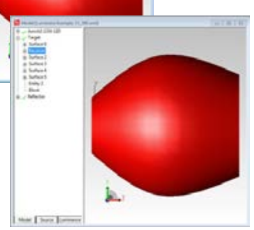
Characteristic Length = 10



Characteristic Length = 100



Characteristic Length = 200

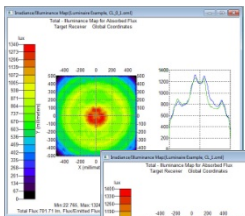


Characteristic Length = 500

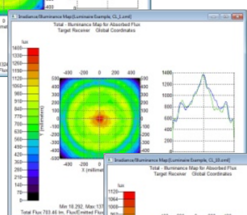
TracePro

Optimization parameters and settings

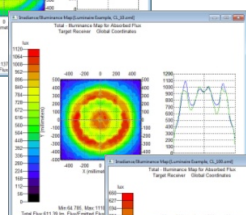
Different Characteristic Length Examples



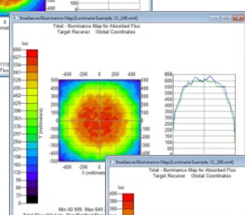
Characteristic Length = 0.1



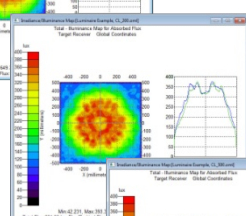
Characteristic Length = 1



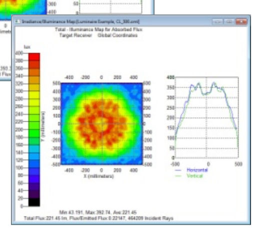
Characteristic Length = 10



Characteristic Length = 100



Characteristic Length = 200



Characteristic Length = 500

Optimization parameters and settings

Stopping Conditions

The stopping conditions determine when the optimization process will be considered finished or complete. Possible stopping conditions include:

•Goal is reached – the process stops when the goal is reached

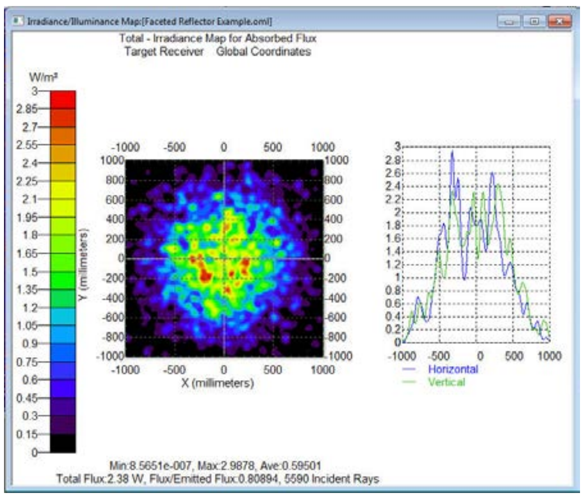
•Number of iterations – the process will stop after a user defined number of iterations

•Iteration tolerance – the process stops when the variation in results from one iteration to the next falls below a certain level

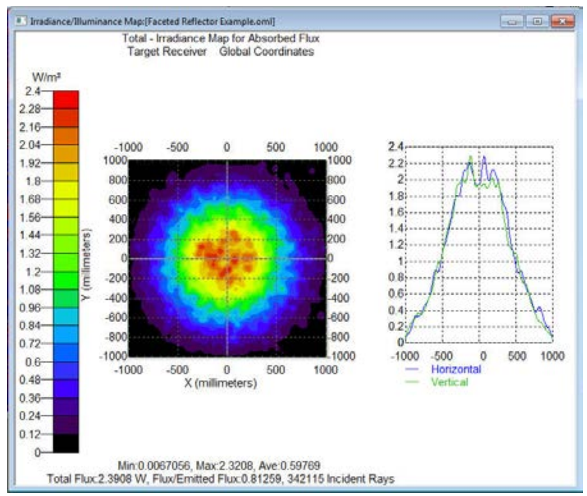
Optimization parameters and settings

Number of rays to trace

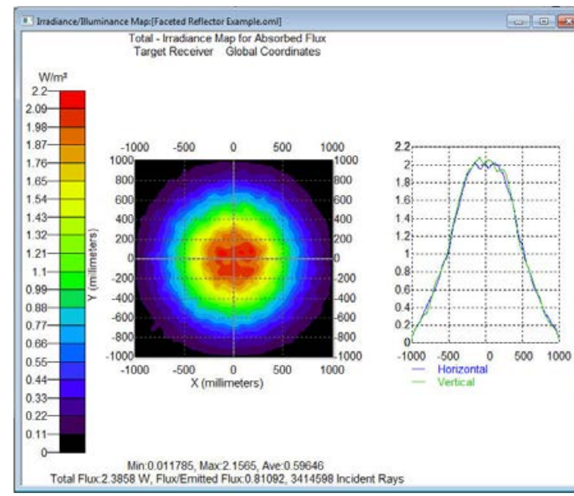
Trace enough rays to get an accurate result in the analysis tools. If too few rays are traced the graphs can be "noisy" and the results will be difficult for the optimizer to interpret.



3000 rays traced



300000 rays traced



3000000 rays traced

TracePro

Optimization parameters and settings

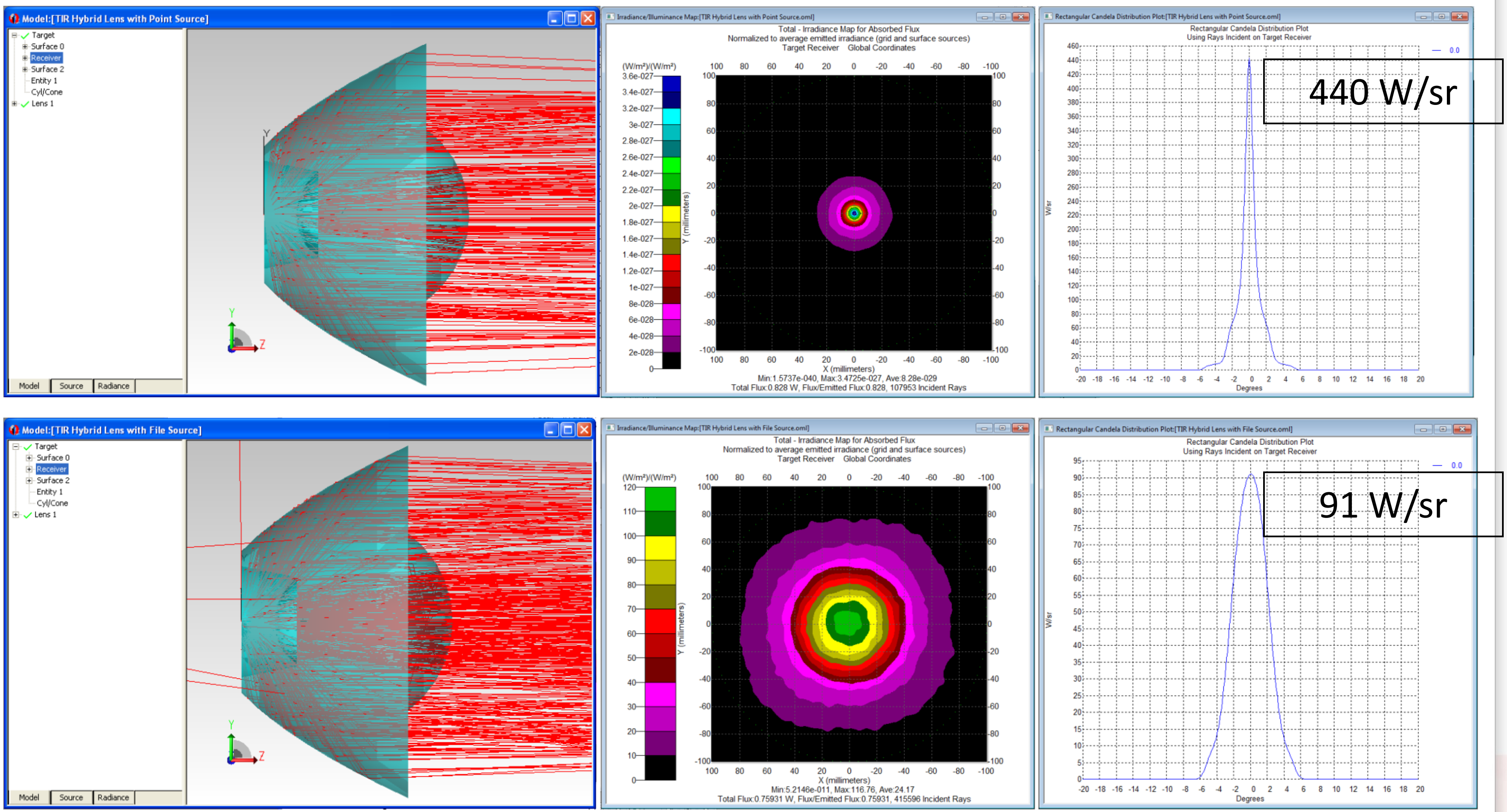
Accurate Source Model

It is very important to have a source model that is as accurate as possible. Source models can include rayfiles, source property files, and full 3D solid models of the source. A bad source model will lead to poor results.

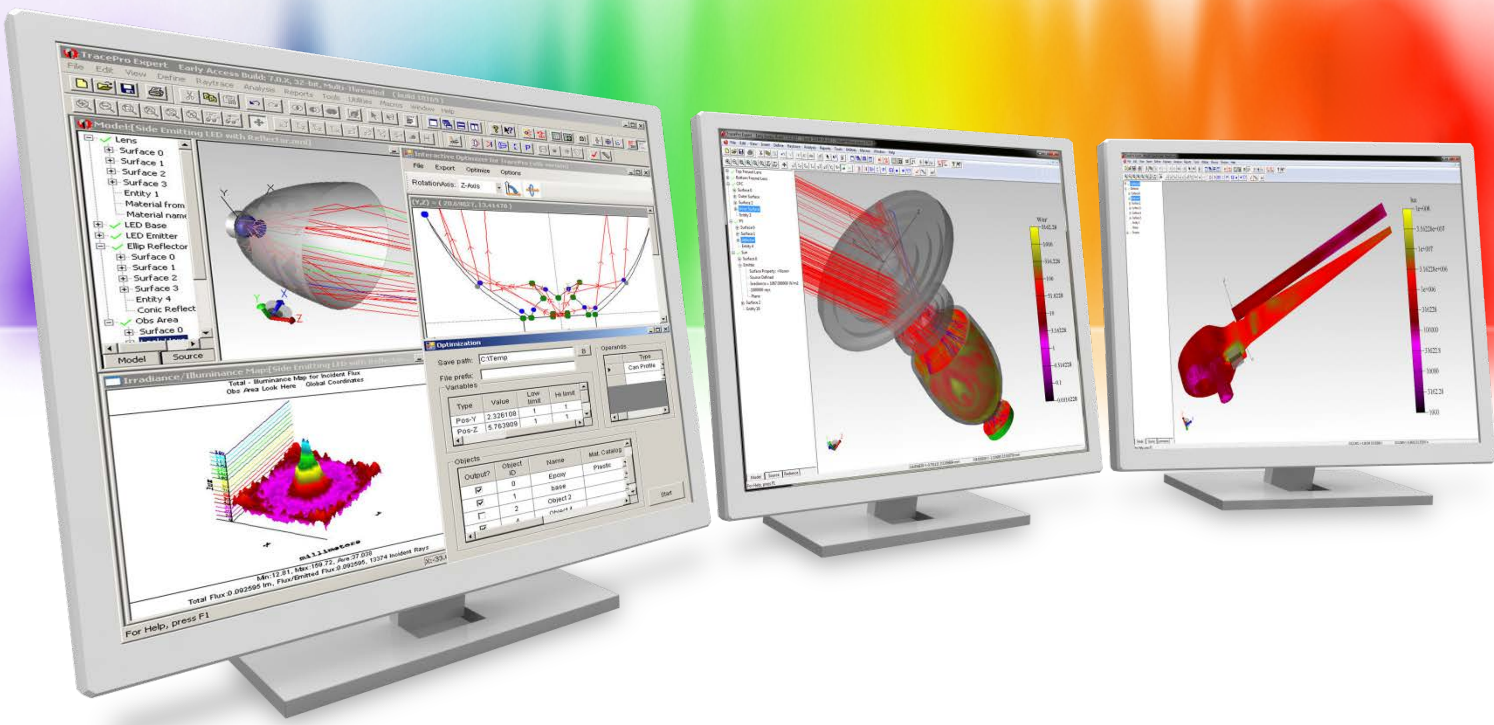
Some factors to consider in a source model include: size, shape, angular distribution, spatial distribution, spectrum/color, and number of rays.

Optimization parameters and settings

Accurate Source Model



TracePro

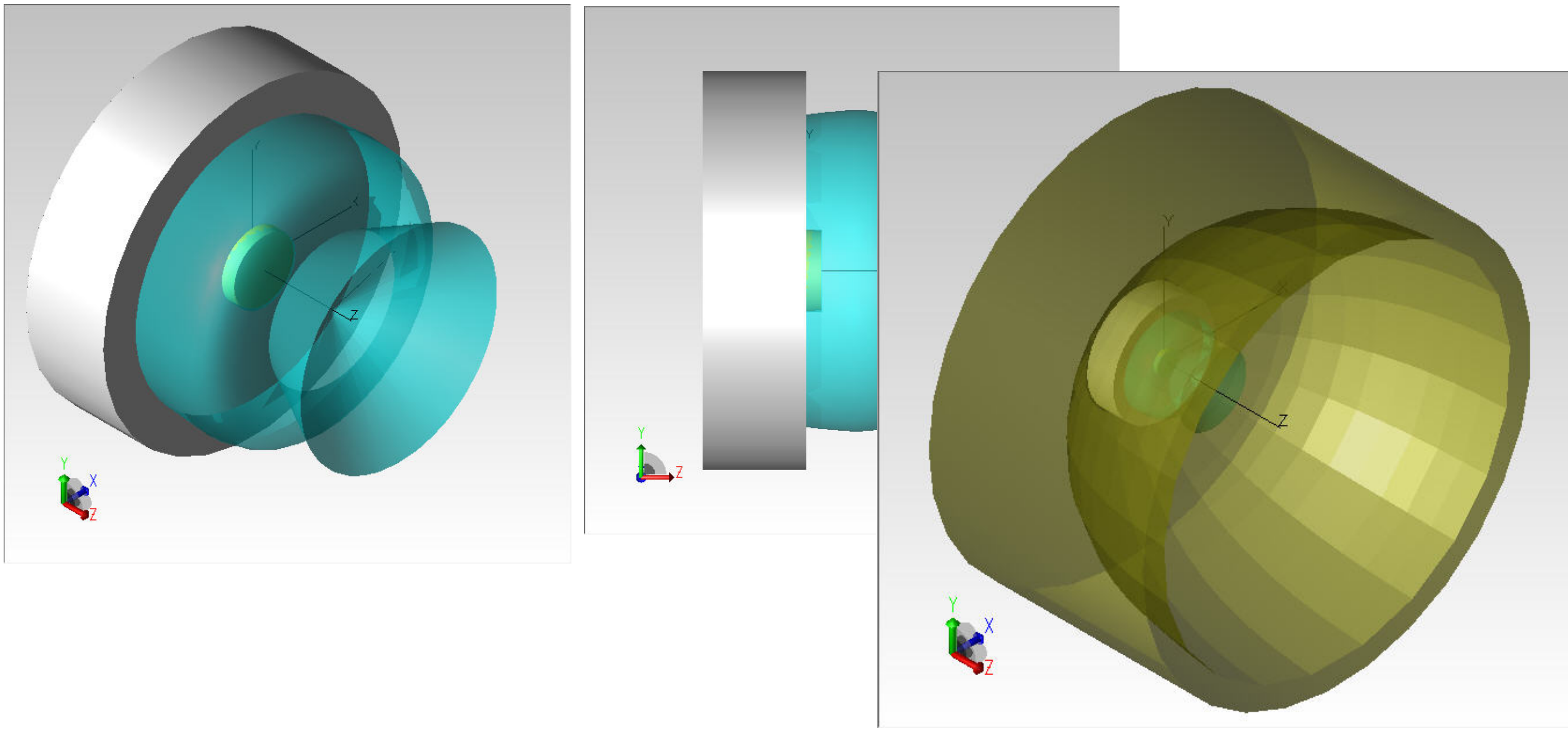


Hybrid System Optimization Example

LAMBDA RESEARCH CORPORATION

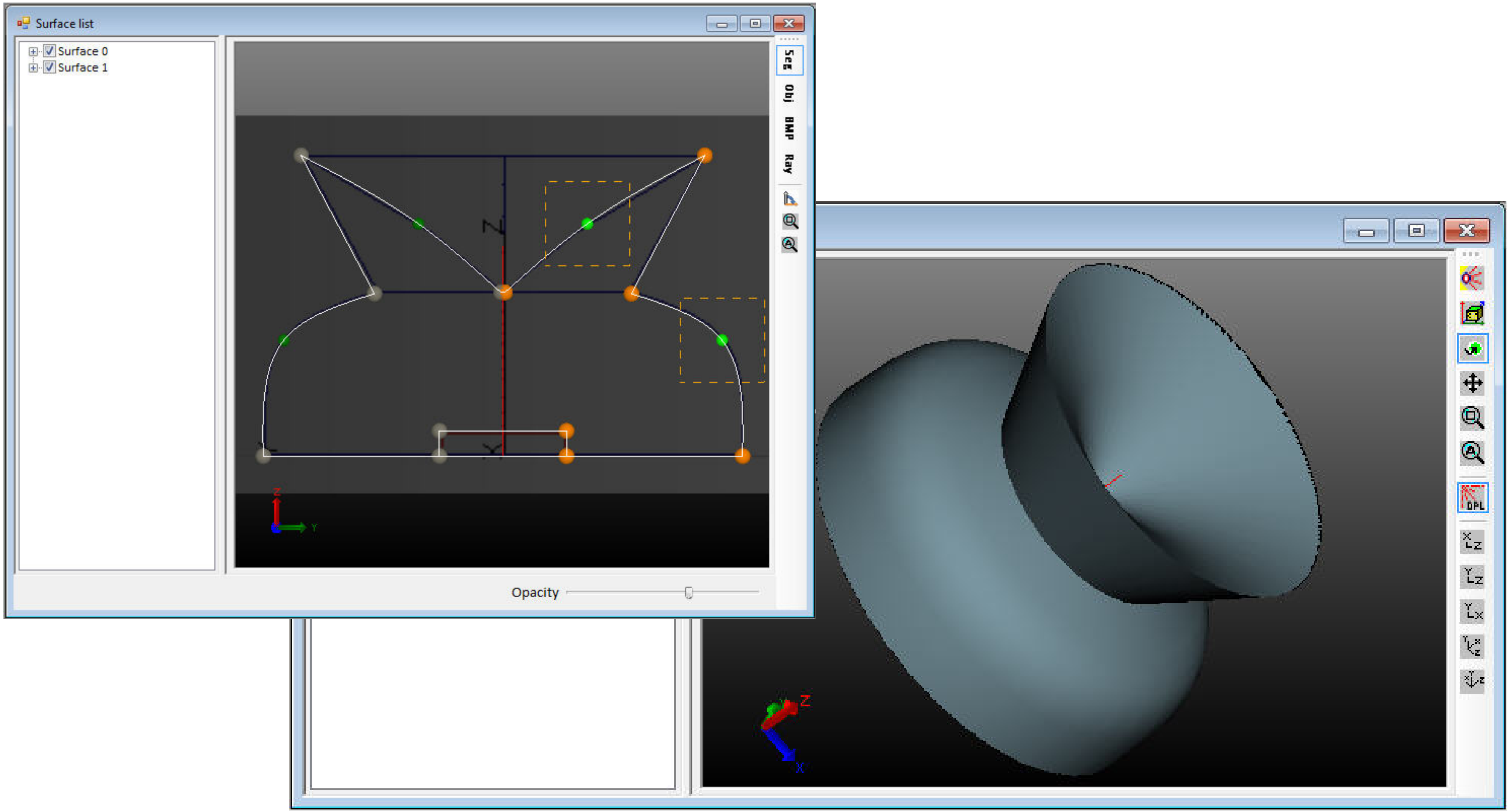
Example: Hybrid System – Lens and Reflector

The Goal: Optimize the shape of a side emitting LED lens and reflector combination



Example: Hybrid System – Lens and Reflector

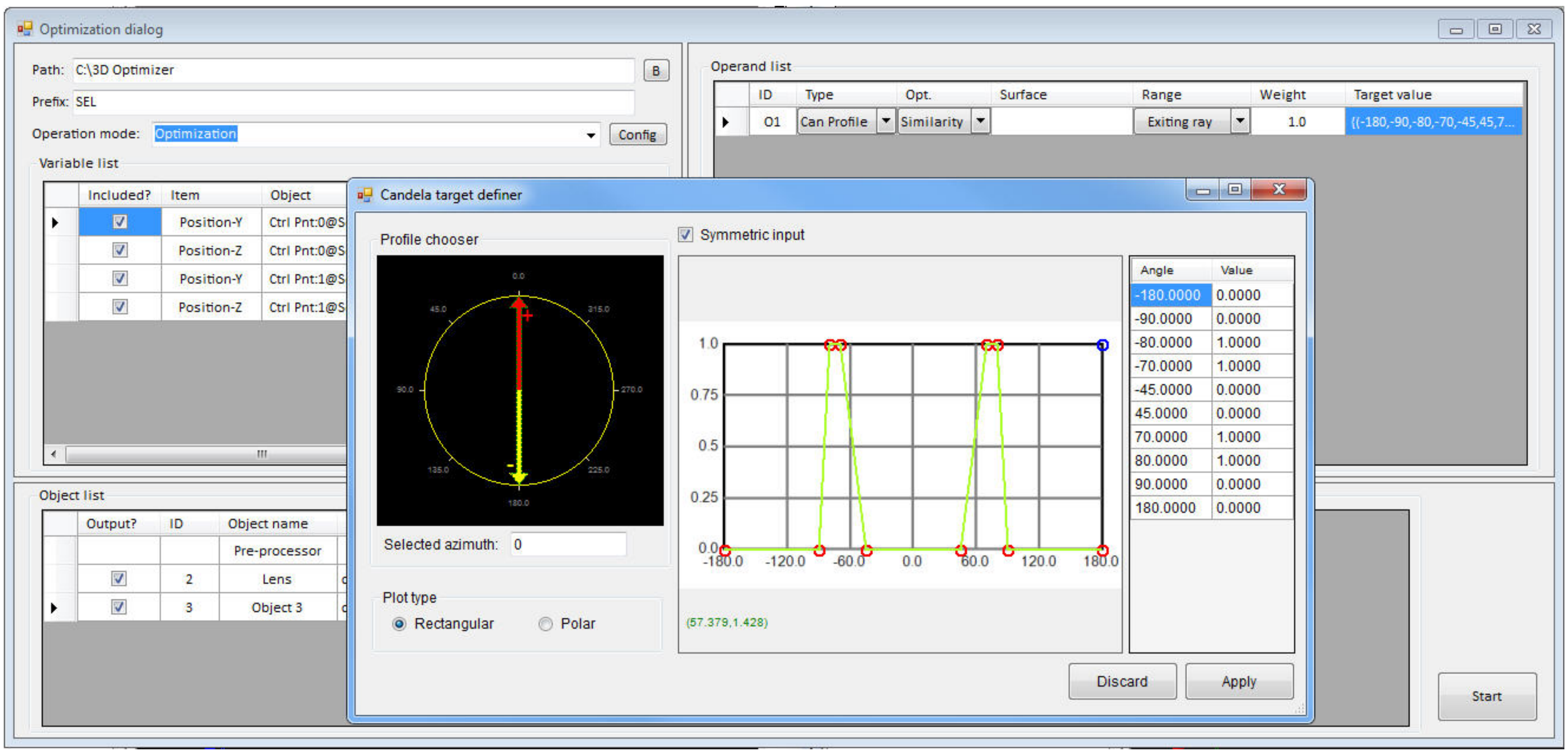
Set-up the side emitting lens



TracePro

Example: Hybrid System – Lens and Reflector

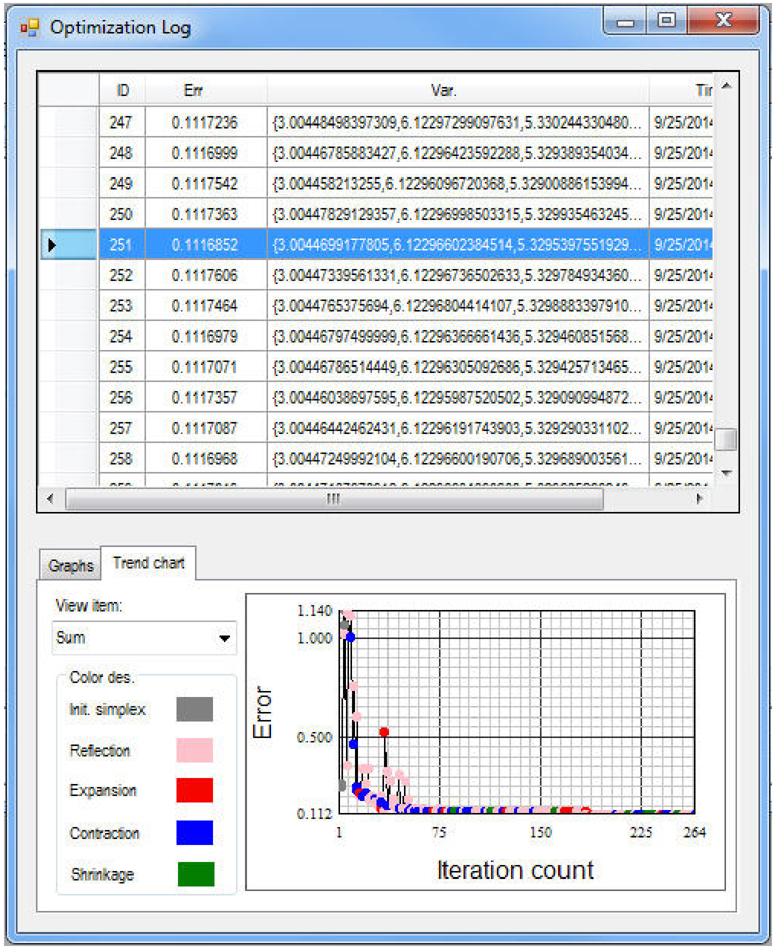
Optimization Goal – Candela profile from 45 to 80 degrees and from -45 to -80 degrees with as little output between those lobes as possible



TracePro

Example: Hybrid System – Lens and Reflector

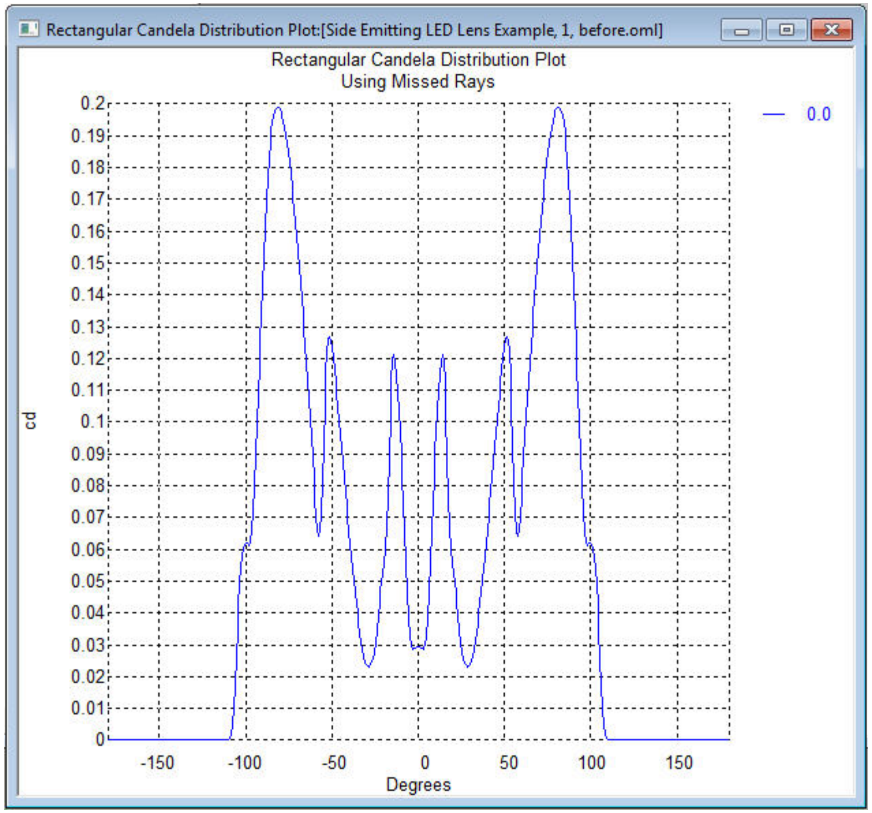
Optimization Log



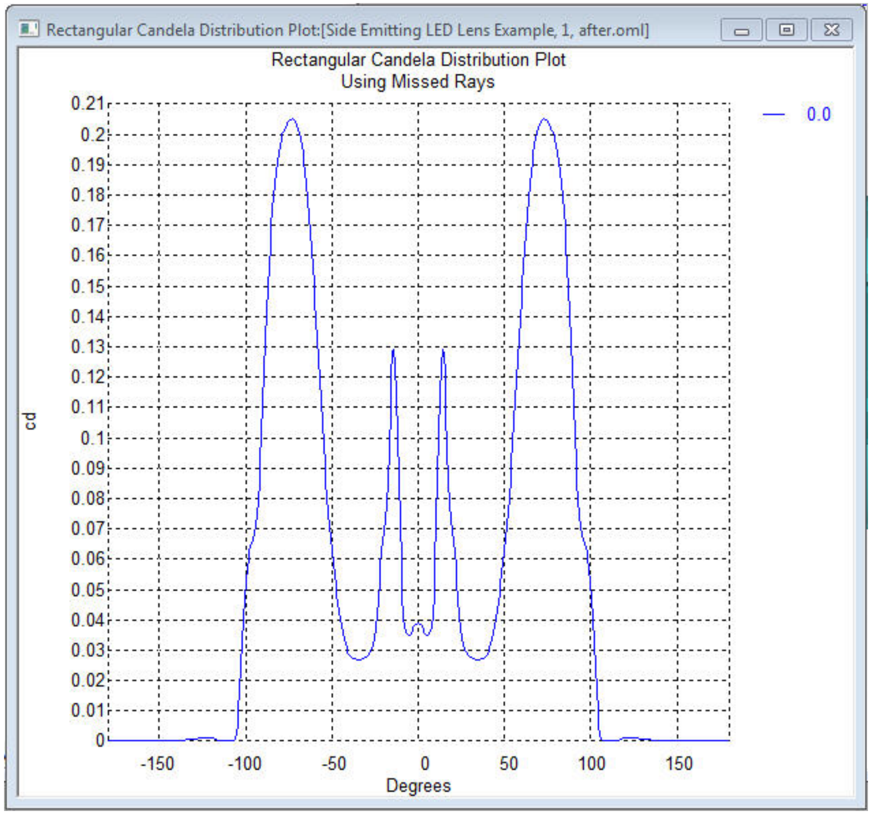
ID	Err	Var.	Ti
247	0.1117236	{3.00448498397309,6.12297299097631,5.330244330480...	9/25/201
248	0.1116999	{3.00446785883427,6.12296423592288,5.329389354034...	9/25/201
249	0.1117542	{3.004458213255,6.12296096720368,5.32900886153994...	9/25/201
250	0.1117363	{3.00447829129357,6.12296998503315,5.329935463245...	9/25/201
251	0.1116852	{3.0044699177805,6.12296602384514,5.3295397551929...	9/25/201
252	0.1117606	{3.00447339561331,6.12296736502633,5.329784934360...	9/25/201
253	0.1117464	{3.0044765375694,6.12296804414107,5.3298883397910...	9/25/201
254	0.1116979	{3.00446797499999,6.12296366661436,5.329460851568...	9/25/201
255	0.1117071	{3.00446786514449,6.12296305092686,5.329425713465...	9/25/201
256	0.1117357	{3.00446038697595,6.12295987520502,5.329090994872...	9/25/201
257	0.1117087	{3.00446442462431,6.12296191743903,5.329290331102...	9/25/201
258	0.1116968	{3.00447249992104,6.12296600190706,5.329689003561...	9/25/201

Example: Hybrid System – Lens and Reflector

Candela Profile– Before and after optimization



Before optimization

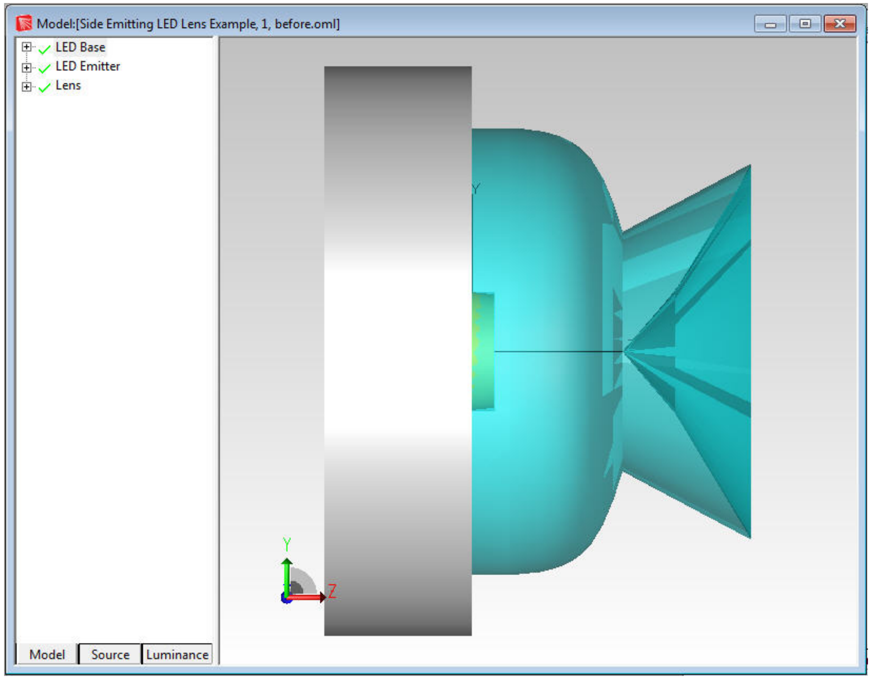


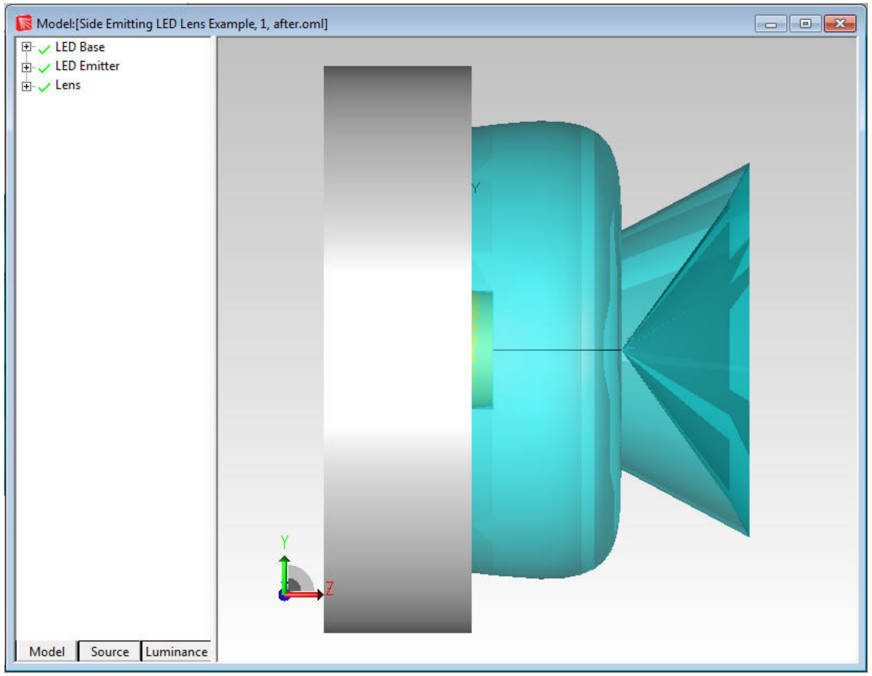
After optimization

TracePro

Example: Hybrid System – Lens and Reflector

Lens Profile– Before and after optimization





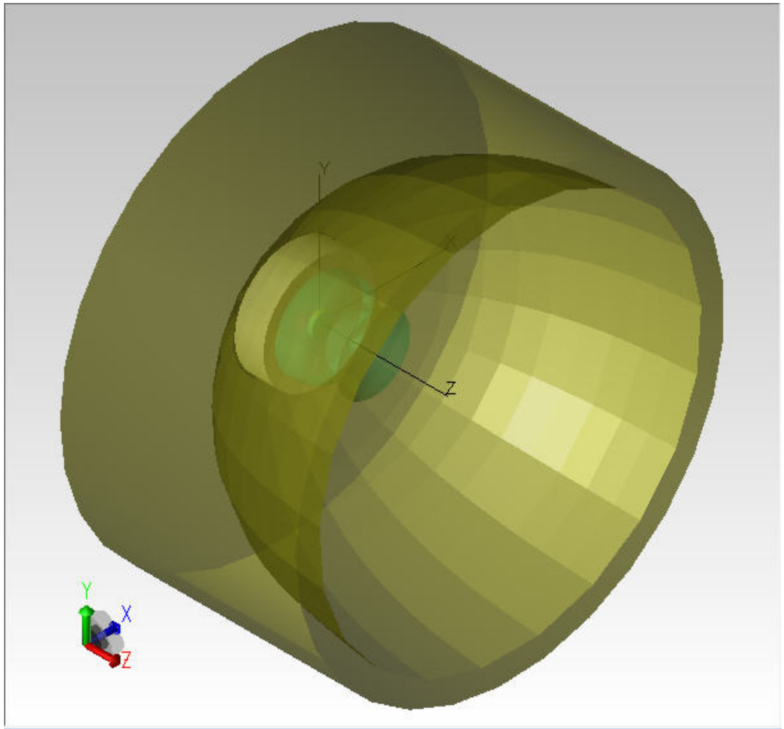
Before optimization

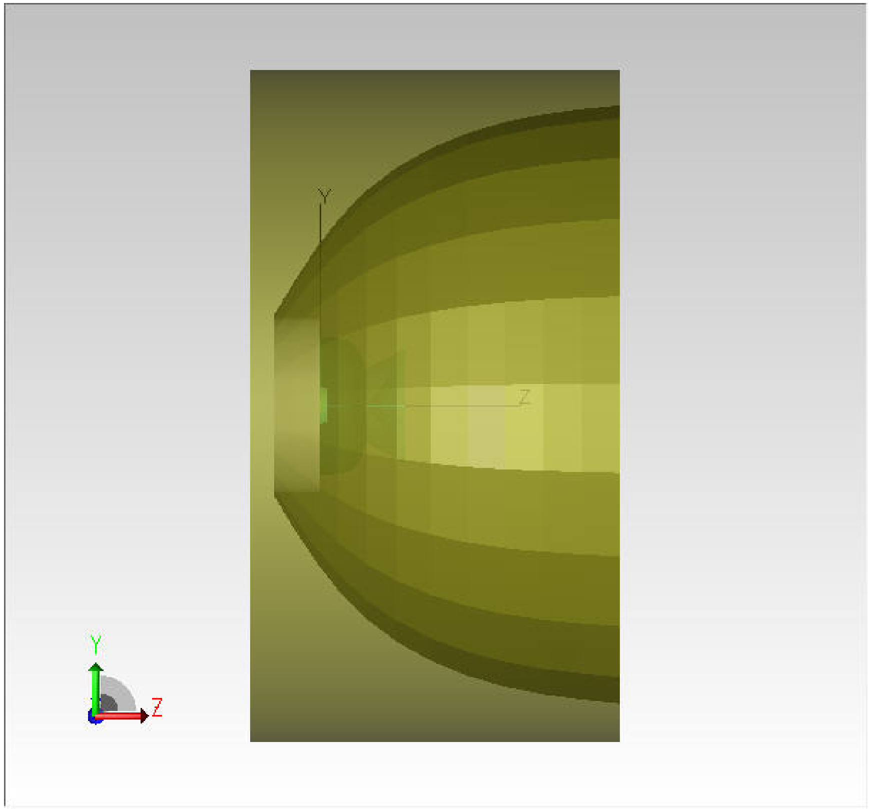
After optimization

TracePro

Example: Hybrid System – Lens and Reflector

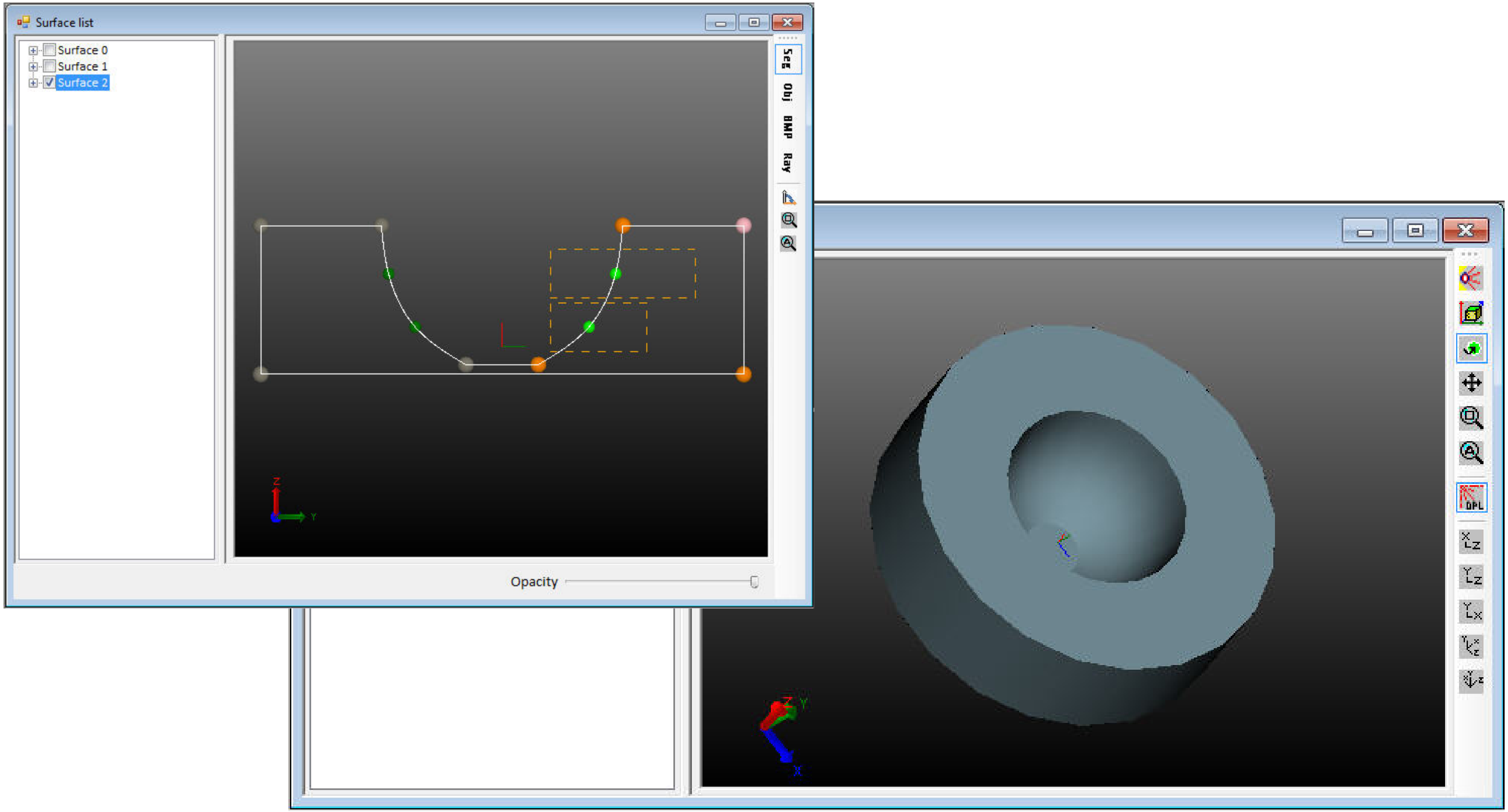
Add a reflector to the lens assembly





Example: Hybrid System – Lens and Reflector

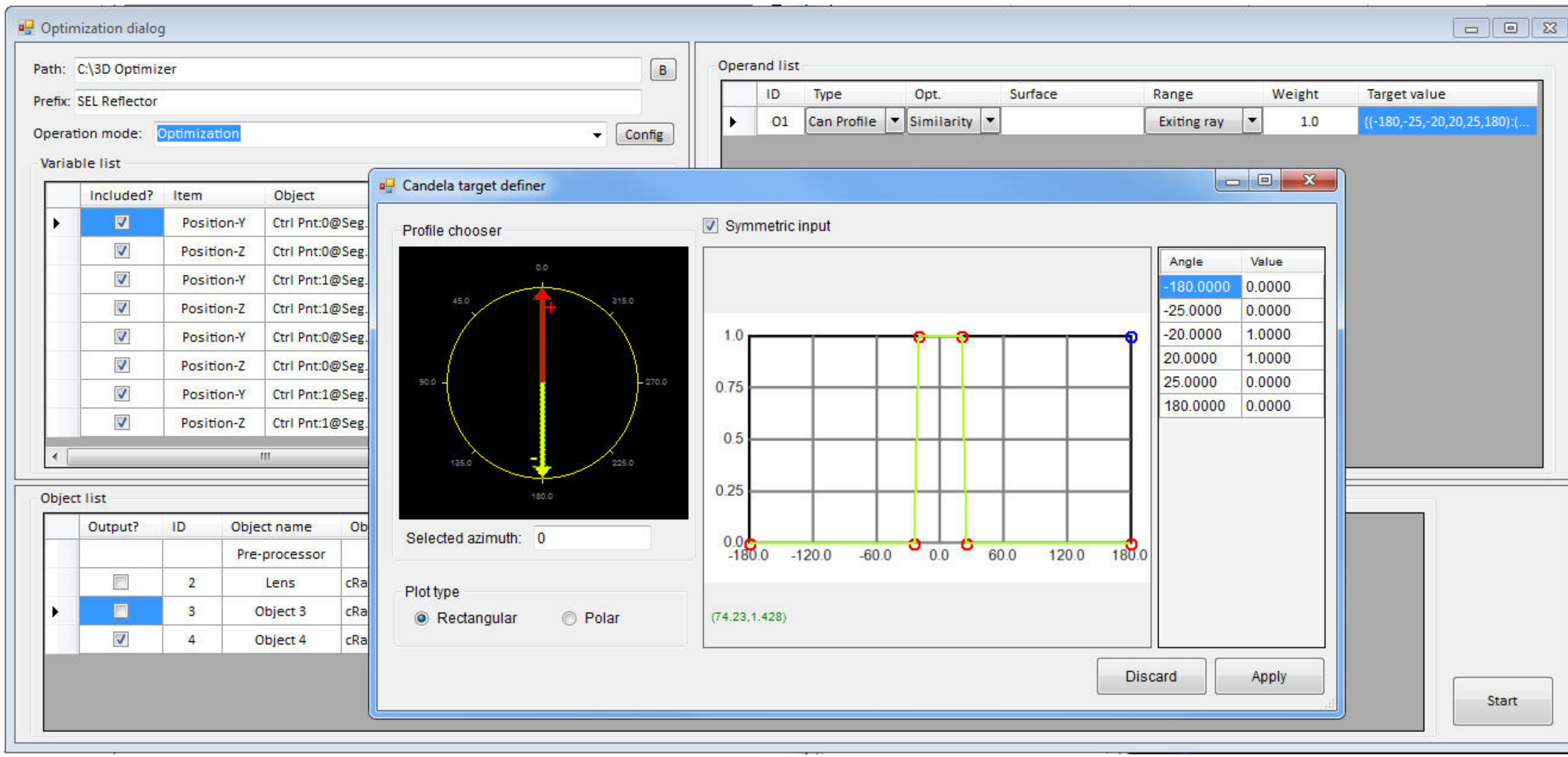
Set-up



TracePro

Example: Hybrid System – Lens and Reflector

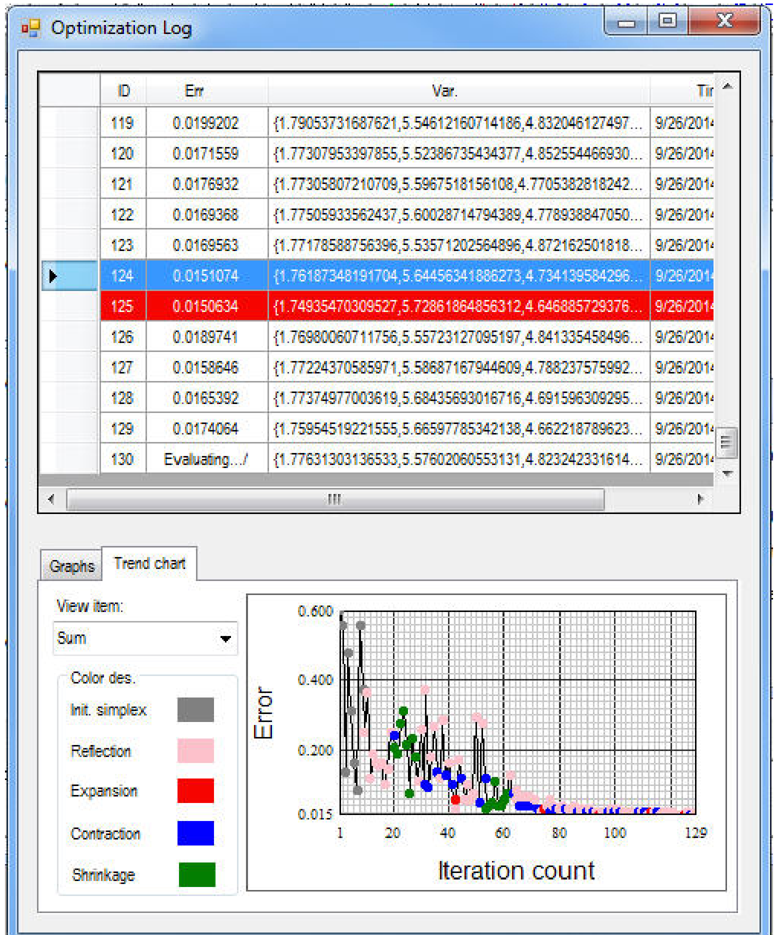
Optimization Goal – Uniform Candela Profile from +/- 20-degrees falling to zero at +/- 25-degrees



TracePro

Example: Hybrid System – Lens and Reflector

Optimization Log

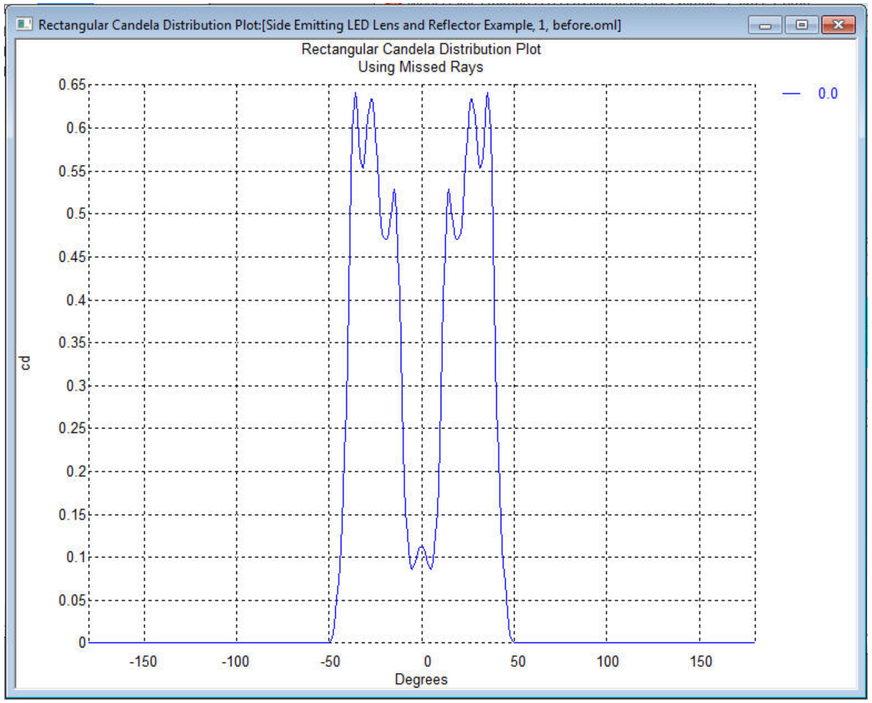


ID	Err	Var.	Tir
119	0.0199202	{1.79053731687621,5.54612160714186,4.832046127497...	9/26/201
120	0.0171559	{1.77307953397855,5.52386735434377,4.852554466930...	9/26/201
121	0.0176932	{1.77305807210709,5.5967518156108,4.7705382818242...	9/26/201
122	0.0169368	{1.77505933562437,5.60028714794389,4.778938847050...	9/26/201
123	0.0169563	{1.77178588756396,5.53571202564896,4.872162501818...	9/26/201
124	0.0151074	{1.76187348191704,5.64456341886273,4.734139584296...	9/26/201
125	0.0150634	{1.74935470309527,5.72861864856312,4.646885729376...	9/26/201
126	0.0189741	{1.76980060711756,5.55723127095197,4.841335458496...	9/26/201
127	0.0158646	{1.77224370585971,5.58687167944609,4.788237575992...	9/26/201
128	0.0165392	{1.77374977003619,5.68435693016716,4.691596309295...	9/26/201
129	0.0174064	{1.75954519221555,5.66597785342138,4.662218789623...	9/26/201
130	Evaluating.../	{1.77631303136533,5.57602060553131,4.823242331614...	9/26/201

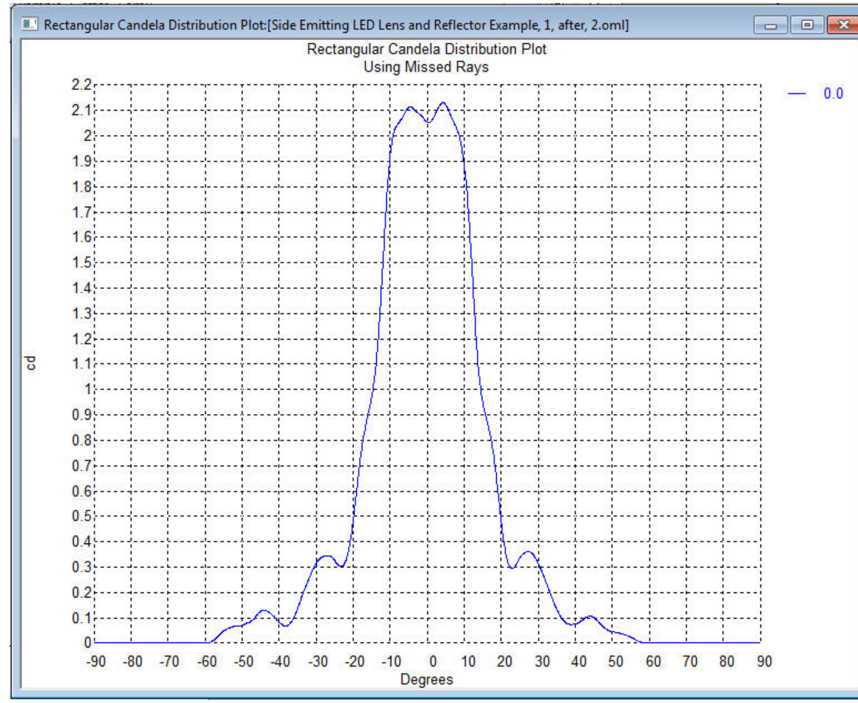
TracePro

Example: Hybrid System – Lens and Reflector

Candela Profile– Before and after optimization



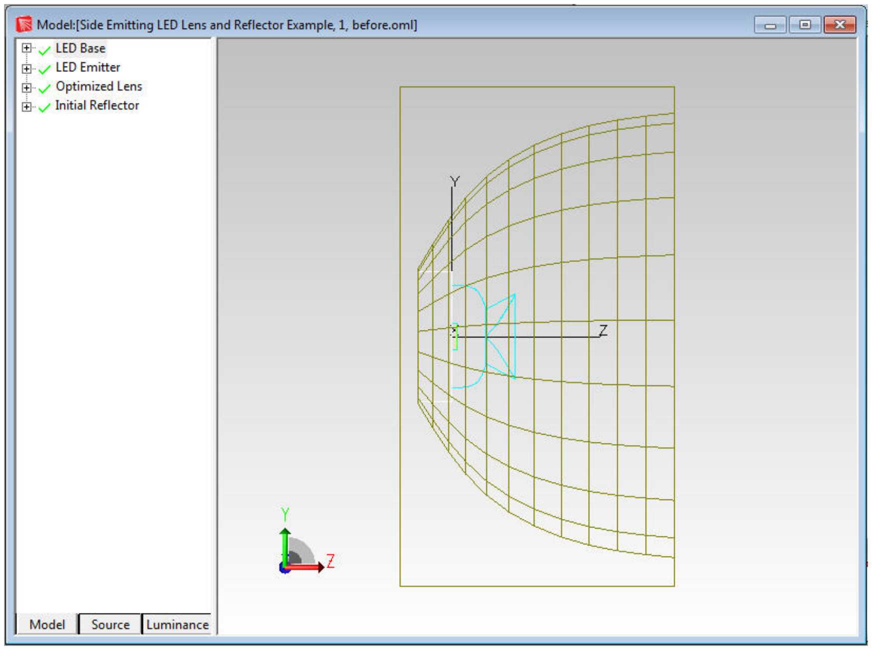
Before optimization

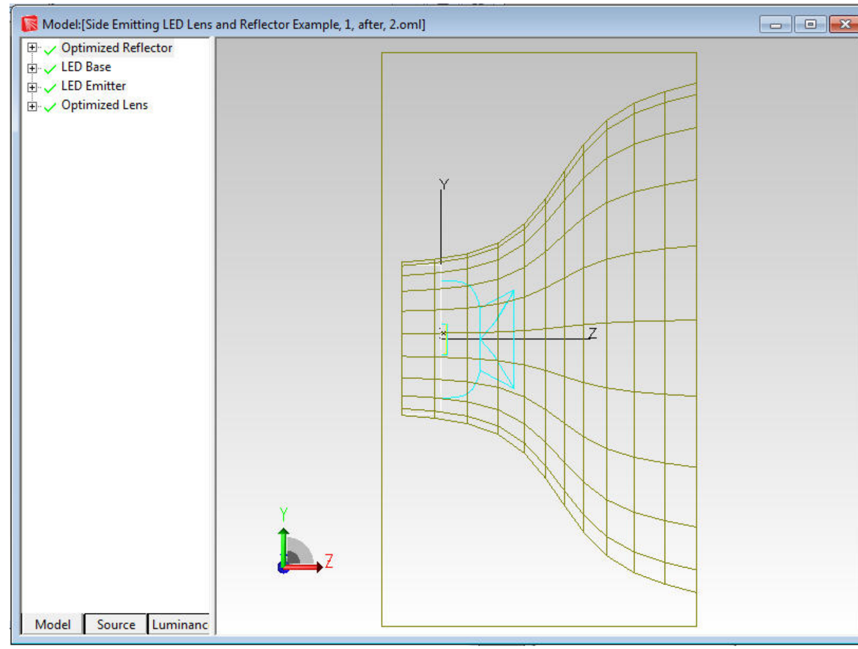


After optimization

Example: Hybrid System – Lens and Reflector

Lens Profile– Before and after optimization





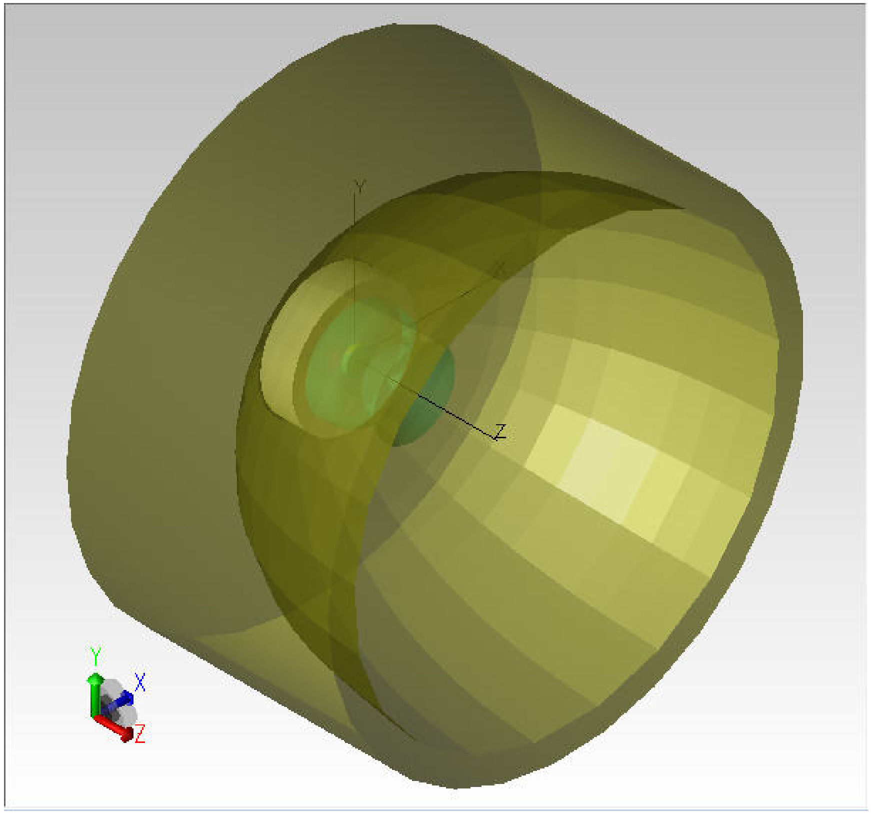
Before optimization

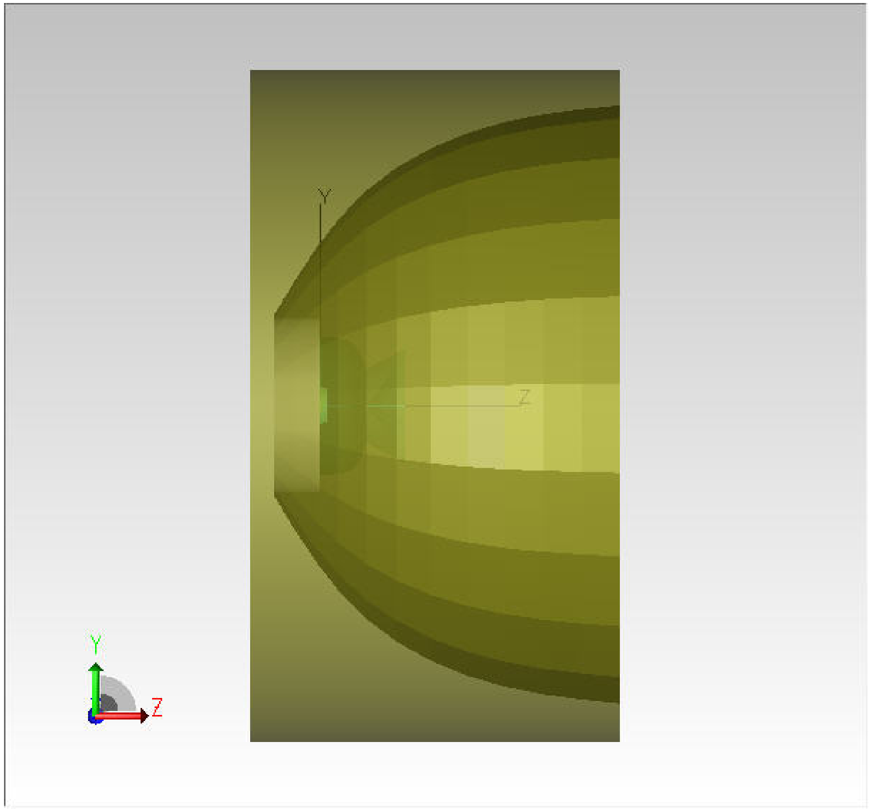
After optimization

TracePro

Example: Hybrid System – Lens and Reflector

The Goal: Optimize Lens and Reflector as a system

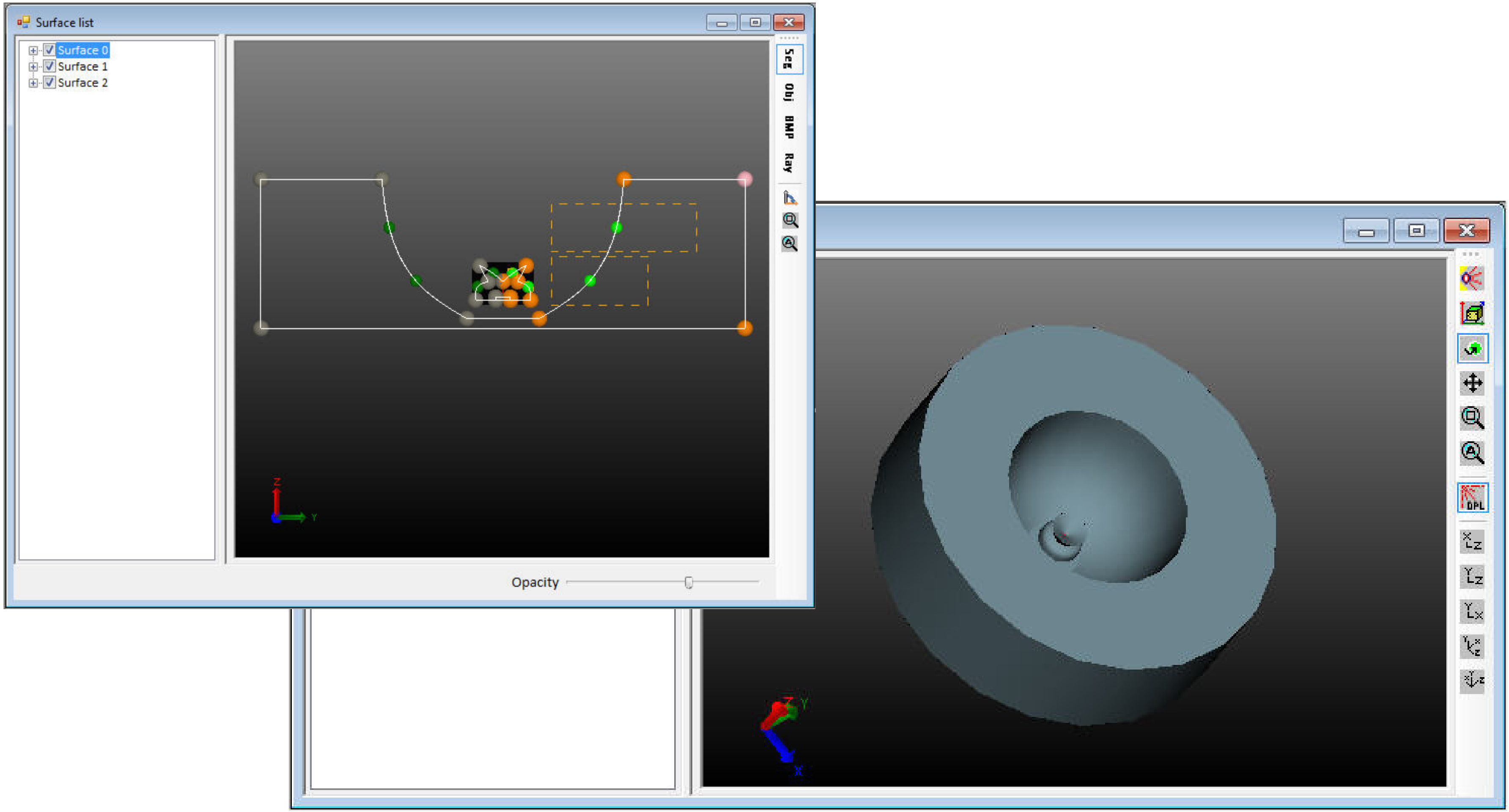




TracePro

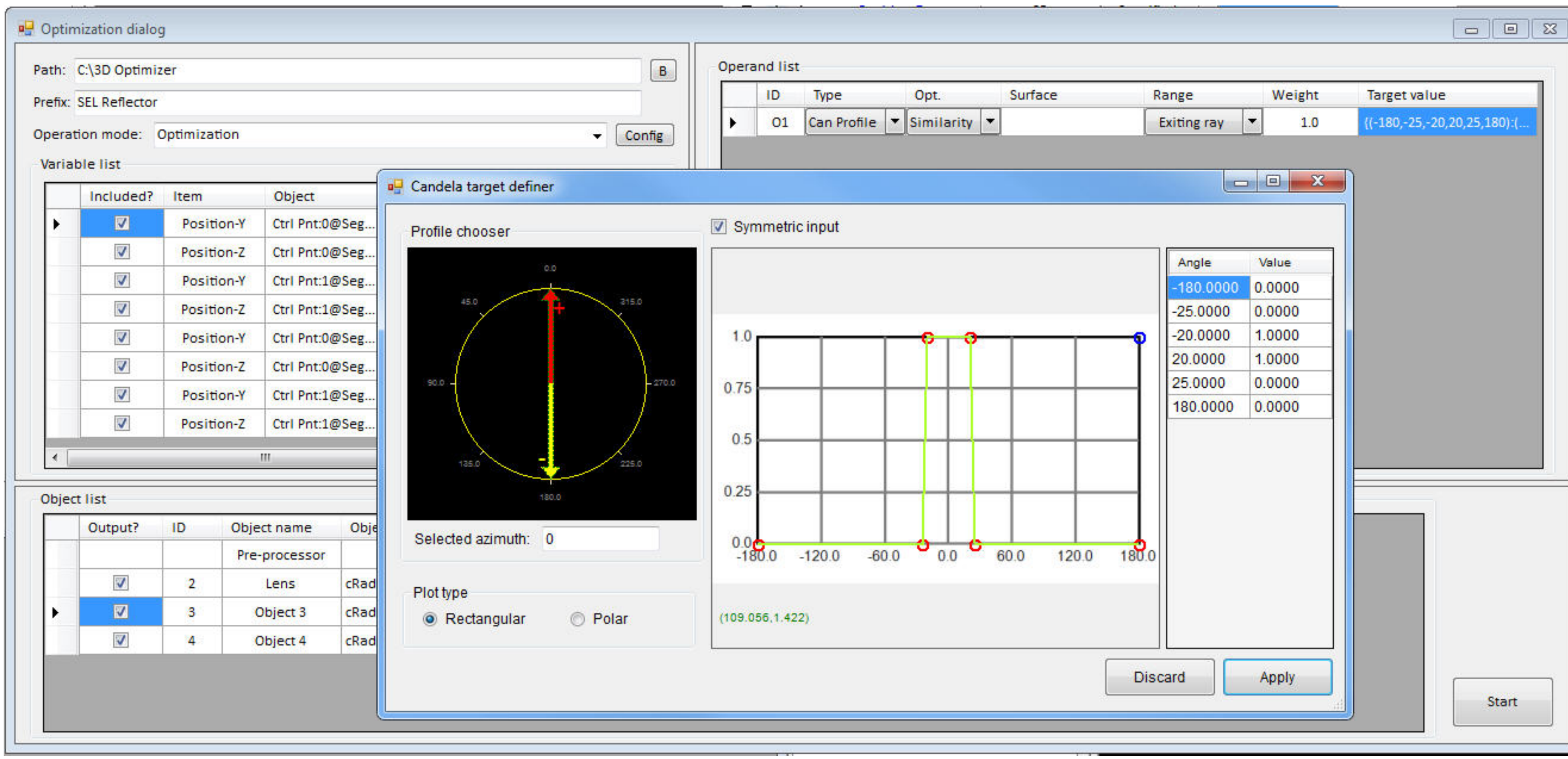
Example: Hybrid System – Lens and Reflector

Set-up



Example: Hybrid System – Lens and Reflector

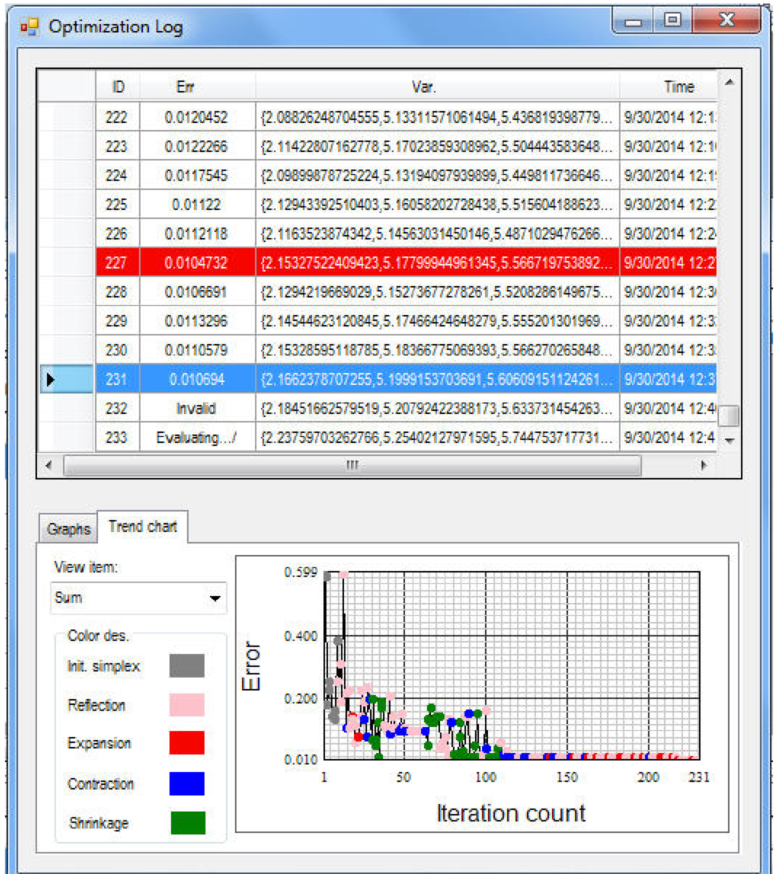
Optimization Goal – Uniform Candela Profile from +/- 20-degrees falling to zero at +/- 25-degrees



TracePro

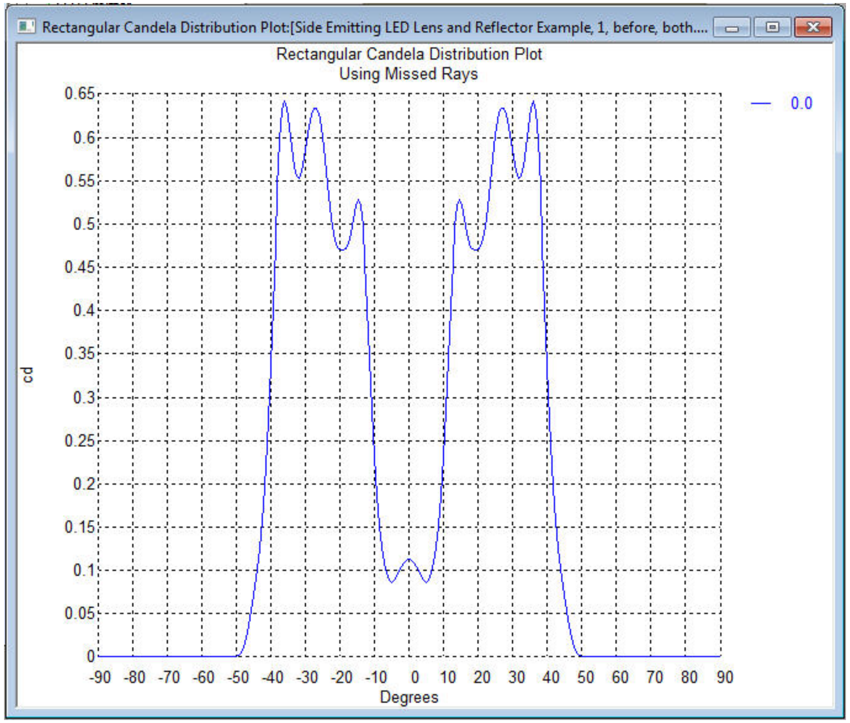
Example: Hybrid System – Lens and Reflector

Optimization Log – Combined optimization

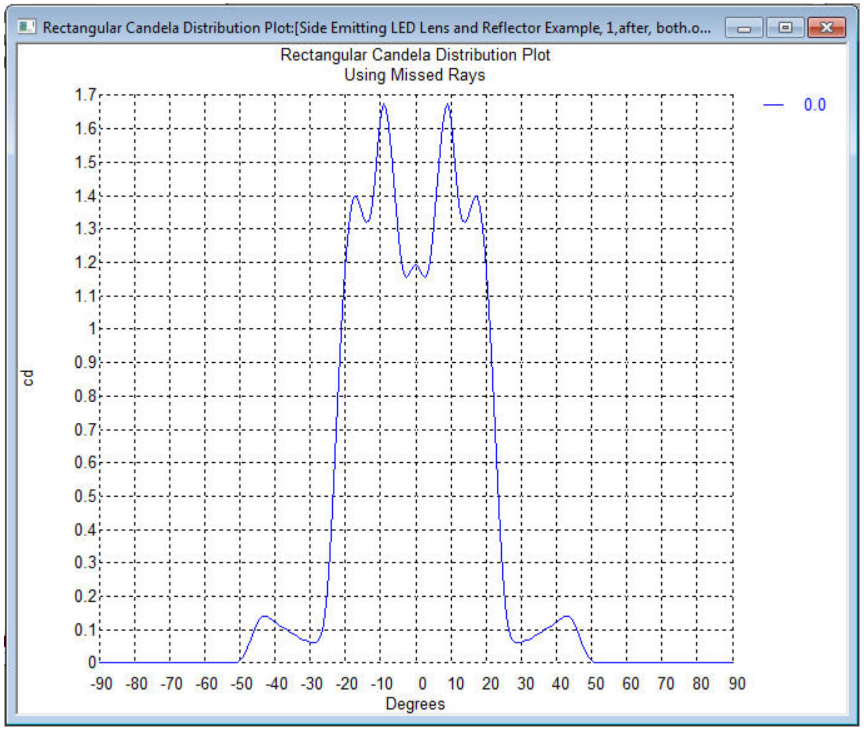


Example: Hybrid System – Lens and Reflector

Candela Profile– Before and after combined optimization



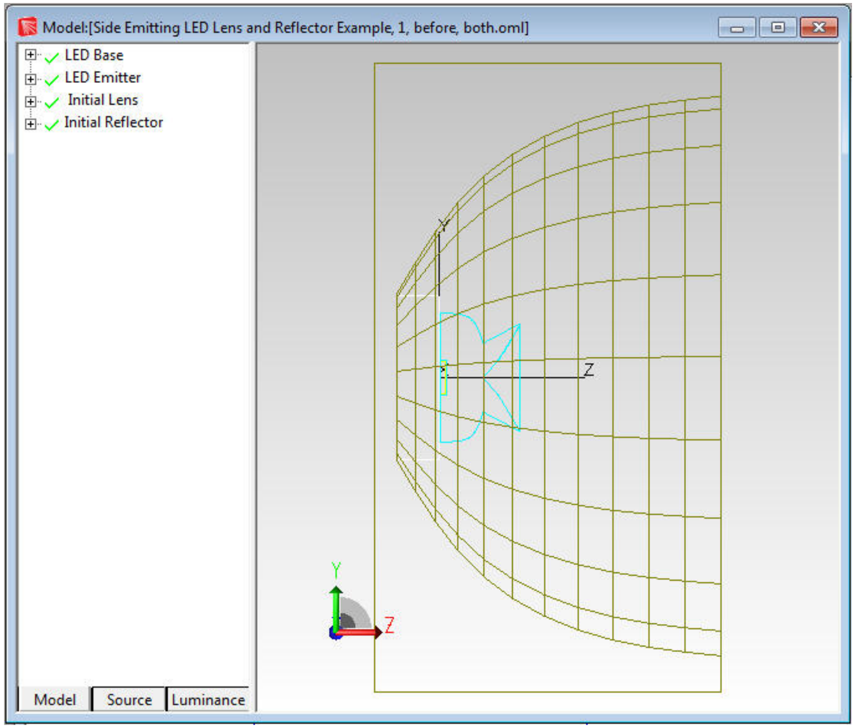
Before optimization



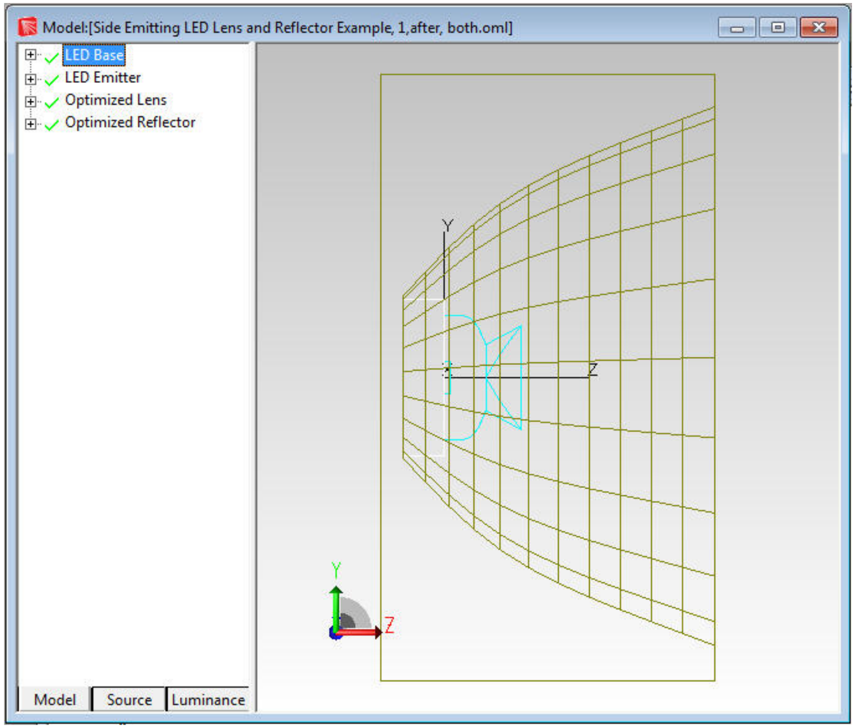
After optimization

Example: Hybrid System – Lens and Reflector

Lens and Reflector Profile – Before and after combined optimization



Before optimization

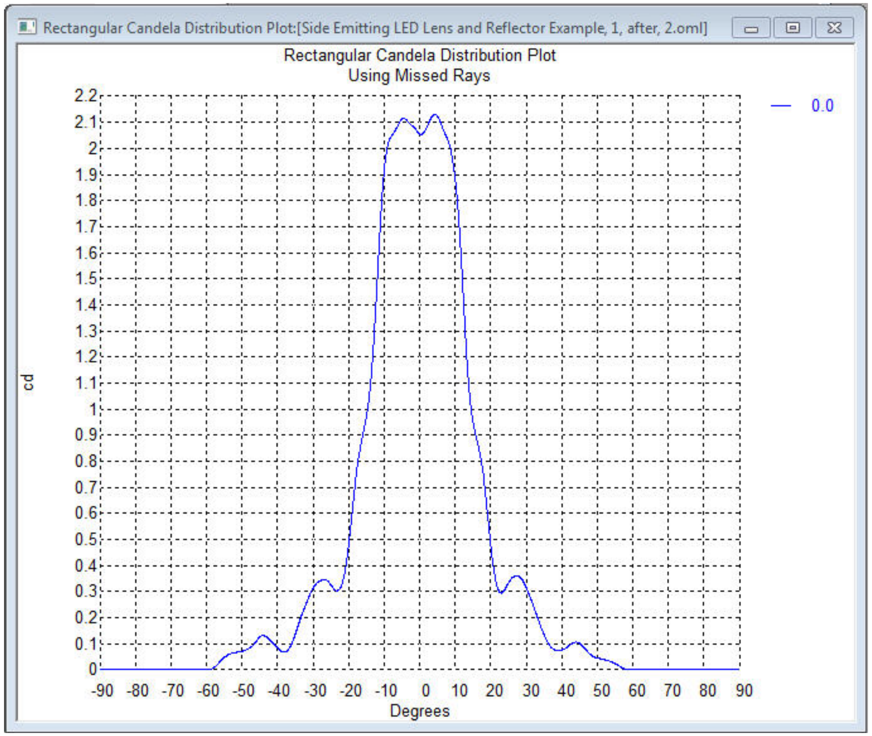


After optimization

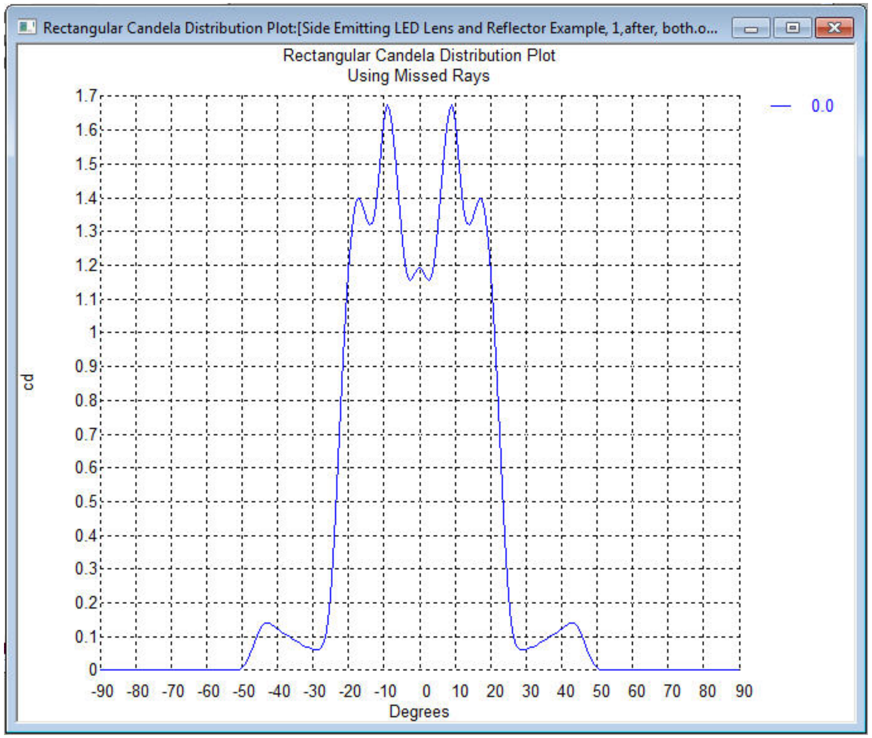
TracePro

Example: Hybrid System – Lens and Reflector

Candela Profile – 2 different optimization procedures



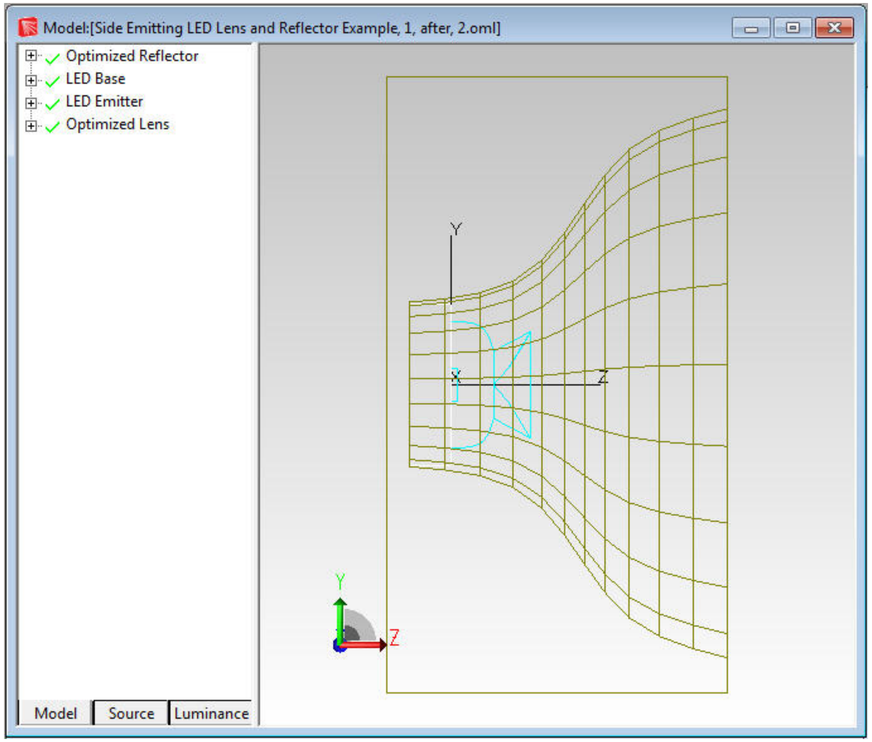
Separate optimization



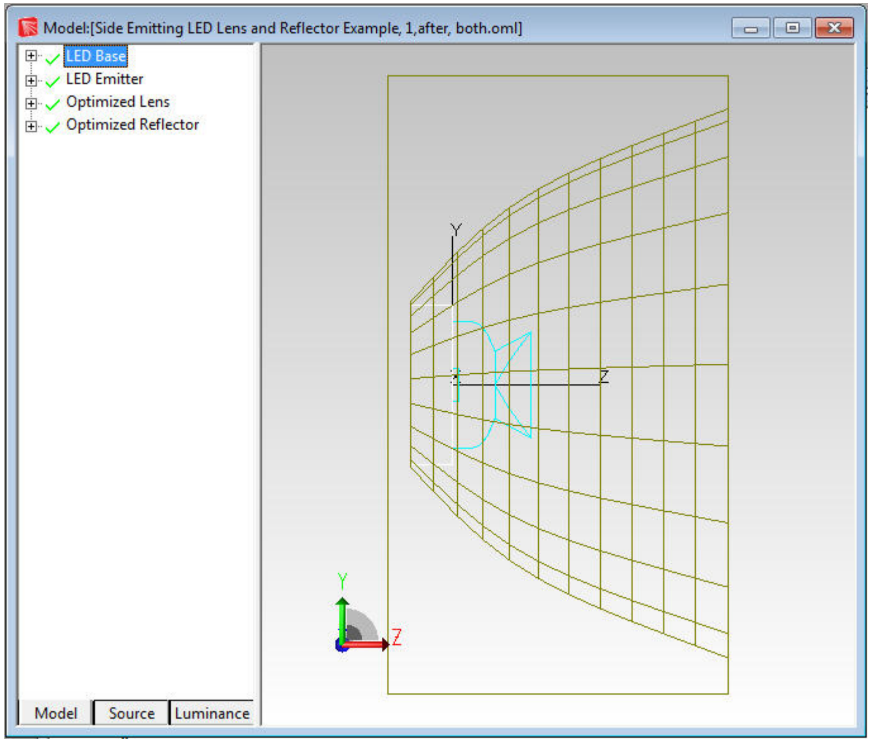
Combined optimization

Example: Hybrid System – Lens and Reflector

Lens and Reflector Profile – 2 different optimization procedures



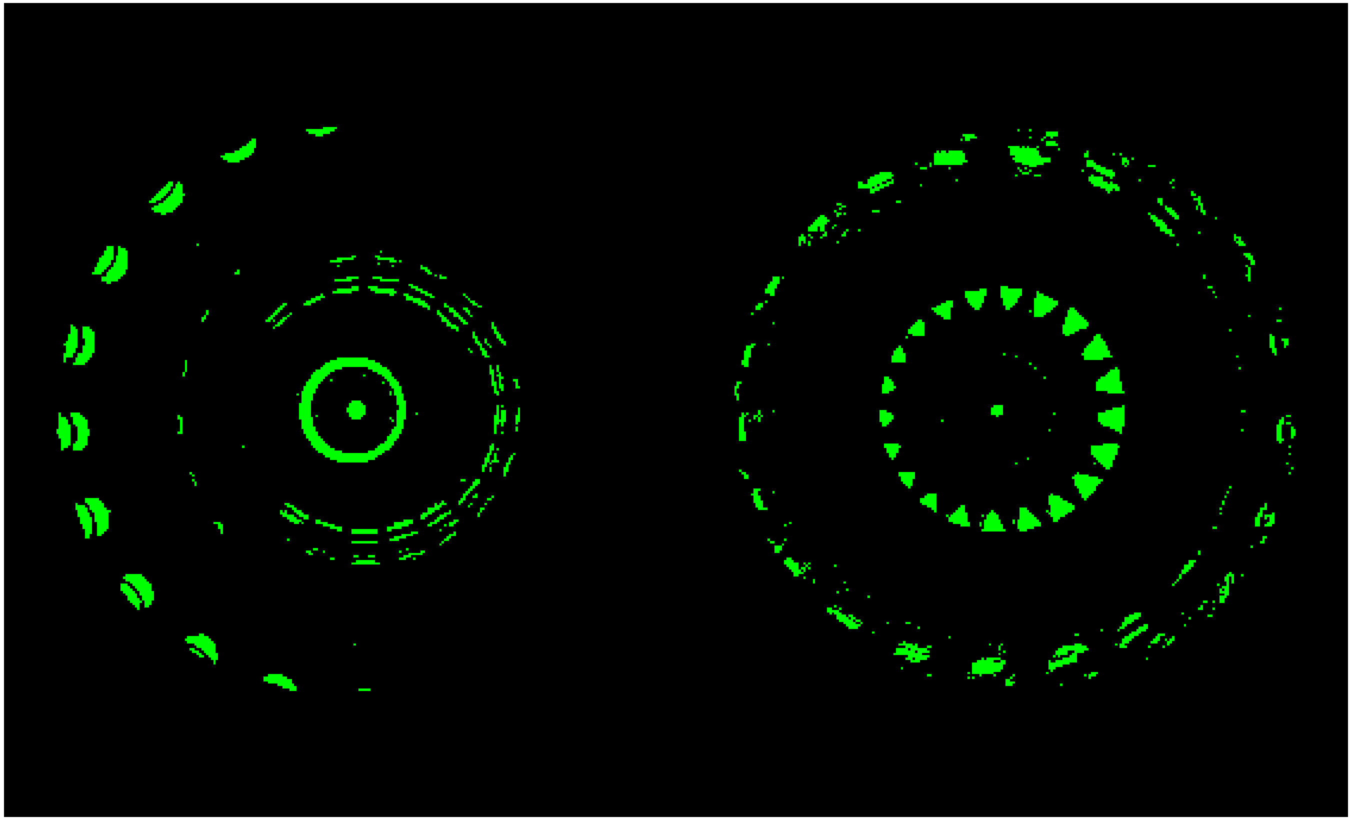
Separate optimization



Combined optimization

Example: Hybrid System – Lens and Reflector

Photorealistic Rendering – 2 different optimization procedures

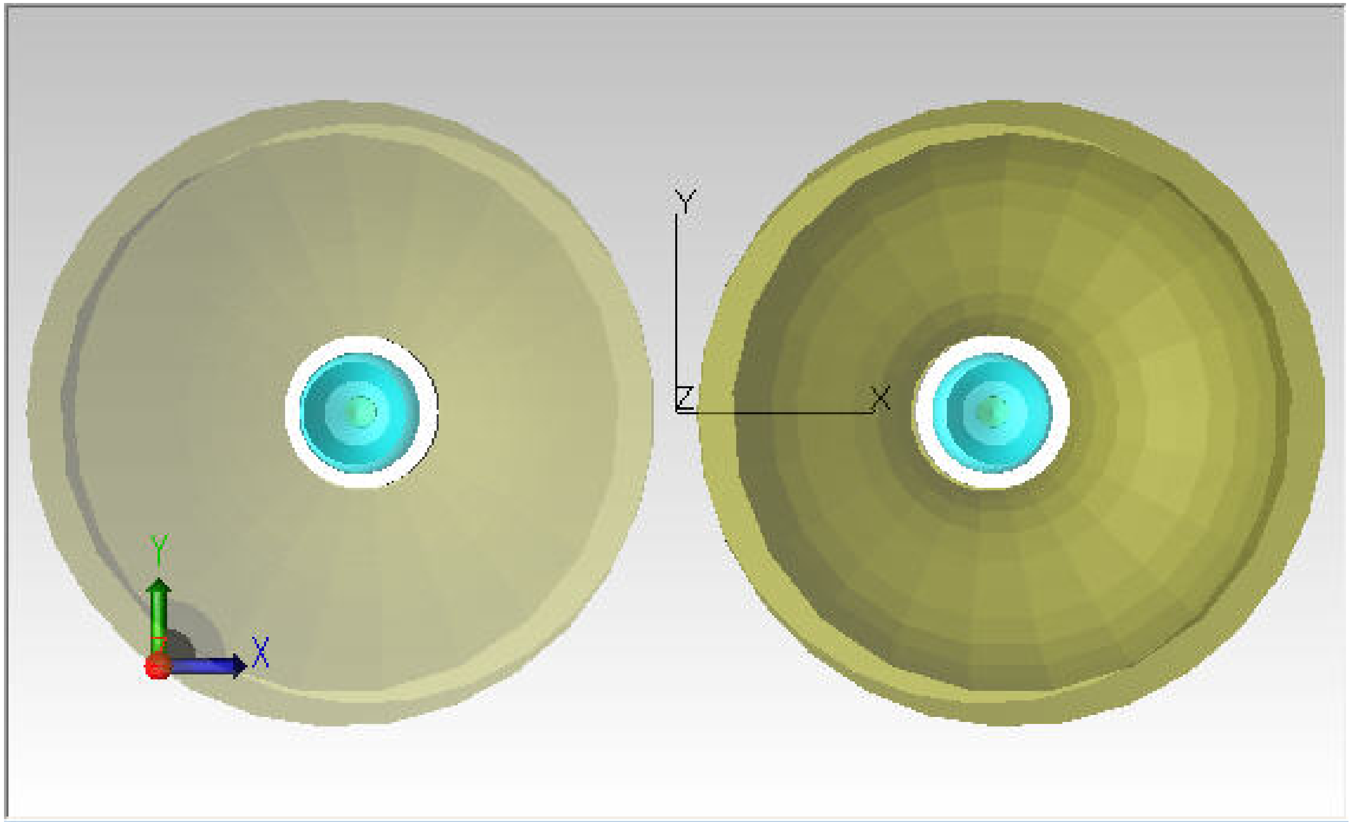


Combined optimization	Separate optimization

TracePro

Example: Hybrid System – Lens and Reflector

Photorealistic Rendering – 2 different optimization procedures



Combined optimization

Separate optimization

Optimization Tips

- Start with a good initial design if possible
- Use accurate models including geometry and properties
- Use accurate source models
- Define enough variables so that the model is not over or under constrained
- Set the Characteristic Length to adequately sample the solution space
- Define achievable optimization operands or goals

Optimization Tips

- Trace enough rays so that the analysis maps are not noisy and the optimizer can make accurate decisions
- Change optimization parameters to check for better solutions
- Know the capabilities of your optical analysis and optimization software

Summary and Questions

Software based optimization allows the user to easily search a large range of solutions to find the best result for a given problem:

- ✓ Luminaire design process time can be shortened considerably
- ✓ Designs can be tested “virtually”, cutting down on the need for physical prototypes
- ✓ A large number of solutions can be searched in a short period of time
- ✓ In addition to geometric shape optimization can also include position, rotational angle, and properties
- ✓ Tolerancing can also be accomplished

For more information or to sign up for our free 30-day trial please visit us at:

www.lambdares.com

Phone: +1 978-486-0766 E-mail: sales@lambdares.com

TracePro