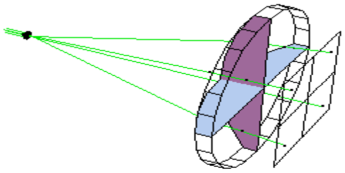
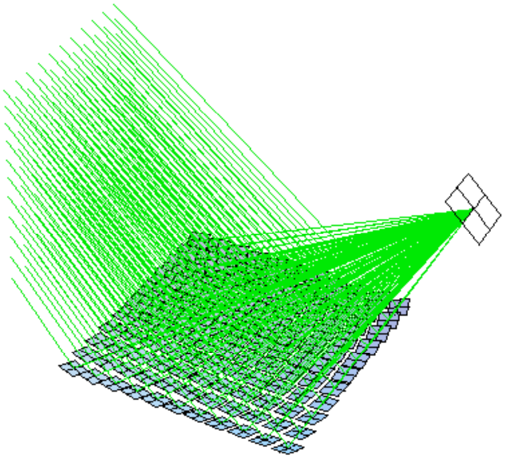
Optica Software ™

Issue No. 19 June 2008

ADDRESSING THE OPTICAL DESIGN AND ANALYSIS NEEDS OF CORPORATE, BUSINESS, EDUCATIONAL, GOVERNMENTAL AND INDIVIDUAL USERS.





Solar Arrays

Optica News

Optica Software has recently hired Andrew Wesly. Andrew will be working on documentation improvement, *Mathematica* demonstrations, optical component database updates and other tasks. If you use components from vendors not in our database, please let us know so that we can be sure to include them in the update.

Wolfram research has been creating video portraits of select *Mathematica* users. If you have something interesting to show and would like to be a part of this, let me know. Check out http://www.wolfram.com/portraits to view some of the interesting videos made so far. Along he same lines, we will some be making video tutorials on the use of *Optica* and its applications.

Optica3 Annual Support Plan for Rayica-Wavica users

Rayica-Wavica users up-to-date with their *Rayica-Wavica* annual support plan can be transferred free of charge to the Optica3 support plan, and download the latest version of *Optica3*. Users needing to renew their support plans should contact us. The latest build date was May 22, 2008. ***Optica3*** has all the feature set of ***Rayica/Wavica*** plus the new **ManipulateSystem** function and the ability to import CAD models into your optical systems. The **ManipulateSystem** function allows users to create wonderful custom interactive interfaces that can run as standalones under ***Mathematica 6*** and MM6 Player.

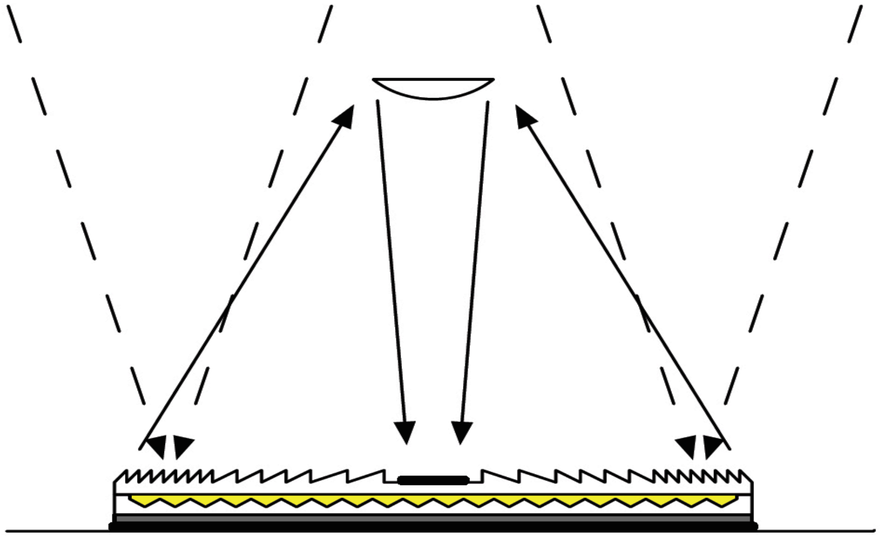
Optica Software is always striving towards improvement. Any of your comments regarding our services will be greatly appreciated. Feel free to send us an email at support@opticasoftware.com with “Optica Software comments” in the subject.

User Research Section: The following is an excerpt from Goldstream Solar's website. Please go there for more details, explanations and excellent animations. If you are a user with some research or development you would like to share with others please let us know.

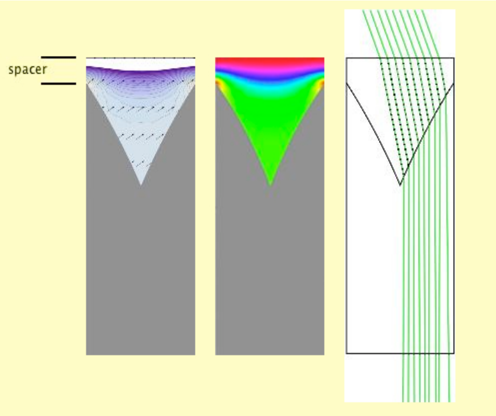
Designing a Motion Free Tracking Solar Concentrator with Rayica:

Contributed by: Dr. John Pender, Goldstream Solar, Fairbanks, ALASKA, USA
http://www.goldstreamsolar.com/SolarConcentrator/SCHome.htm

Motion Free Tracking Solar Concentrator



Goldstream Solar has designed, prototyped and patented a motion free solar concentrator that tracks the Sun and focuses the light onto a target using the physical properties of liquid crystals. We have developed a comprehensive computer model from first principles using *Mathematica* and *Rayica*. We use *Mathematica* to calculate the director as a function of position in the liquid crystal volume, and *Rayica* to trace rays through the liquid crystal/film substrate combination. The liquid crystal is a graded-index (GRIN) medium. The model correctly describes the amount of beam steering and divergence imposed on an incident columnated beam by a steering panel under a wide range of conditions.

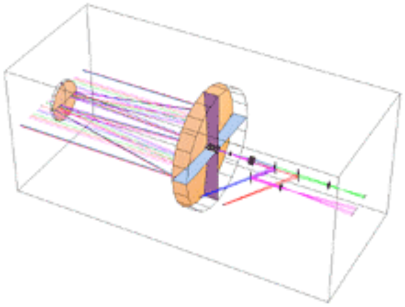


From: Prototype 4-6 pages
http://www.goldstreamsolar.com/SolarConcentrator/SCHome.htm

The figures to the left show the director distribution in the liquid crystal volume as a contour plot and the extraordinary refractive index as a density plot for the case of a narrow spacer between the top of the film and the upper electrode. A modest curvature was placed on the prisms optimized to compensate for lensing. See the website above for more pictures and details.

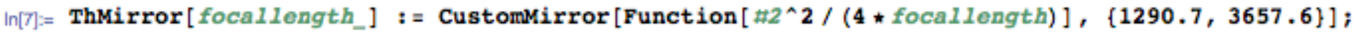
Notes from the Developer

Donald Barnhart, Ph.D. – Optica Software Lead Developer
donald@opticasoftware.com



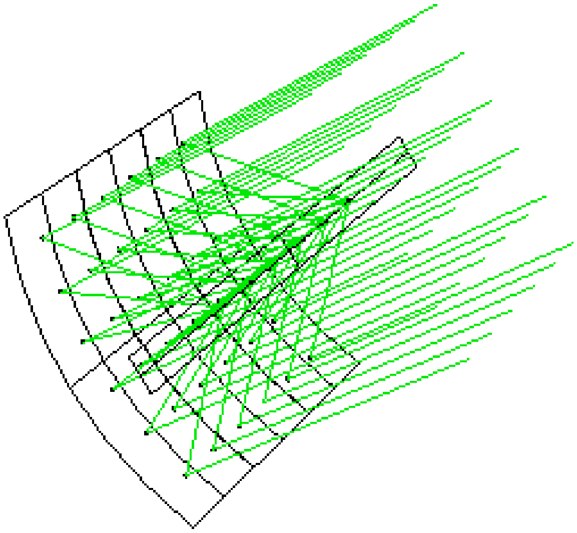
Q/A:
How do I model a parabolic mirror that is parabolic only along one-axis, as in a solar trough?

You would use a *CustomMirror* and define the surface function in the first variable as in this example:



```
In[7]:= ThMirror[focallength_] := CustomMirror[Function[#2^2 / (4 * focallength)], {1290.7, 3657.6}];
```

Here the aperture is {1290.7, 3657.6} The *ParabolicMirror* function assumes the mirror is parabolic in two axes.



Solar Trough

How do I create a model of the sun with Rayica or Optica?

We have now put a couple of approaches to this on our website at http://www.opticasoftware.com/support One approach sends a *GridOfRays* through a *CustomDiffuser*. This is described in detail at http://www.opticasoftware.com/documentation/tutorials/SunModelDiscussionWeb.pdf We also have a function contributed by Ben Jacobson, at Illumitech, http://www.illumitech.com His function is useful with tracking mirrors. See http://www.opticasoftware.com/documentation/tutorials/SunSourceIllumitech.pdf

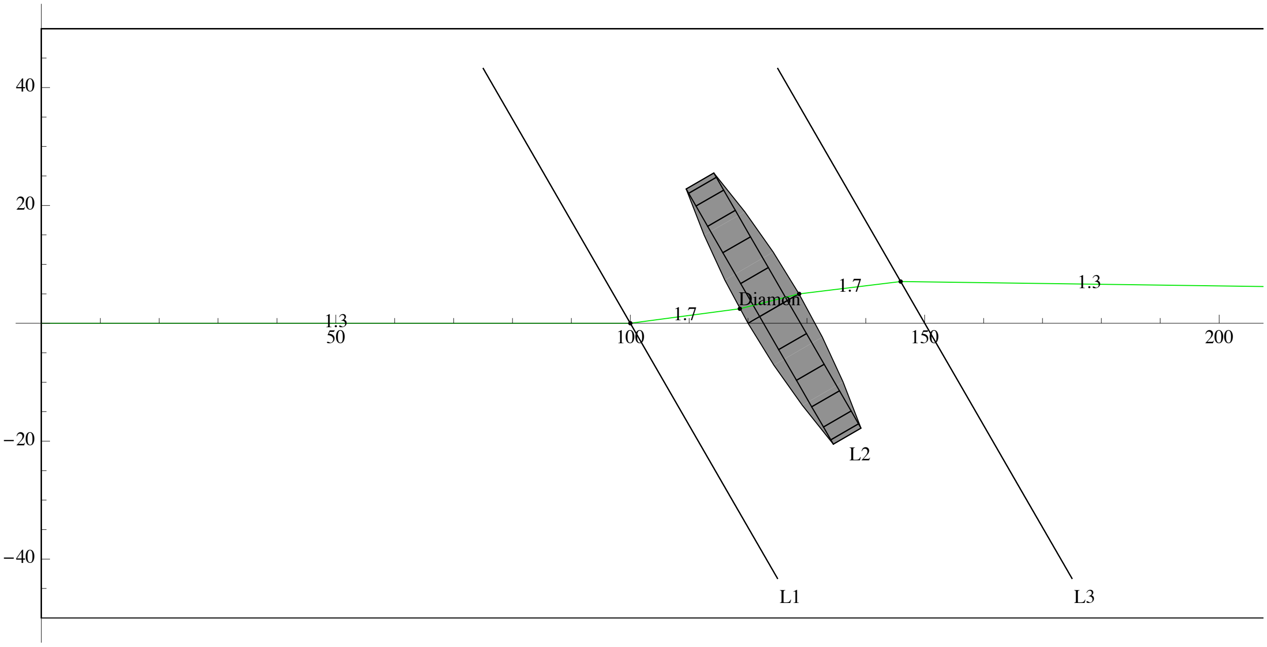
How does one change optical mediums ?

The user may modify the *IntrinsicMedium* , *OpticalMedium* or *ComponentMedium* . Here is an example :

```
DrawSystem[ {

   SingleRay[
    OpticalMedium → 1.3],

   Move[LensSurface[{100, 100}, ComponentMedium→ { 1.3, 1.7}],
    {100, 0, 0}, 30],
   Move[BiConvexLens[100, 50, 10, ComponentMedium→ {1.7, Diamond}], {120, 0, 0}, 30],
   Move[LensSurface[{100, 100}, ComponentMedium→ {1.7, 1.3}],
    {150, 0, 0}, 30],
   Boundary[{0, -50, -50}, {210, 50, 50}]},
  PlotType → TopView, Axes → True, ShowArrows → False, ShowText → OpticalMedium];
```

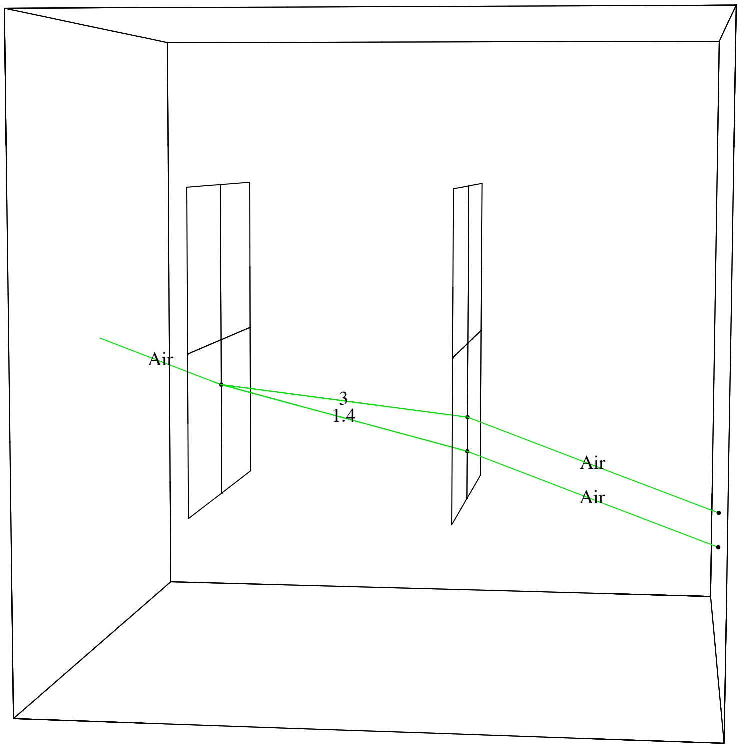


How can one do Birefringence ?

The user may use *BirefringentLensSurface* in combination with one or more additional BirefringentLensSurface component types to create a complete optical element such as a prism. *BirefringentLensSurface* uses the option *ComponentMedium -> {mediumO, mediumE}* to designate the type of birefringent optical material with the alignment axis of materialO given by axisangle. Otherwise, *ComponentMedium -> {{mediumO, mediumE}, isotropicmedium}* specifies the interface with an isotropic material other than the default IntrinsicMedium setting.

The default *ComponentMedium* is *ComponentMedium->{CalciteO,CalciteE}*

```
DrawSystem[{Move[SingleRay[], {0, 0, 0}, 20],
  Move[BirefringentLensSurface[40, {50, 50}, ComponentMedium→ {1.4, 3}], 20],
   Move[BirefringentLensSurface[40, {50, 50}, ComponentMedium→ {1.4, 3}], 60],
  Boundary[100]}, ShowText → OpticalMedium]
```


Air
3
1.4
Air
Air