

Introductory Exercise - Landscape Lens

This section shows how to enter lens data in OSLO from first principles. The overall goal is not only to learn the mechanics of entering data, but also to learn how to apply some of the standard tools in OSLO to gain an understanding of the optical performance of a simple landscape lens similar to that used on low-cost cameras. You should follow the steps exactly as shown here. When you have successfully completed the formal steps, you can return to the beginning and repeat this exercise with some side trips to satisfy your curiosity.

In a cookbook example of this type, it is important that you start with the same configuration as the machine used to make the example. If you have changed many preferences before starting this exercise, you can restore your preferences to *factory-default* condition by deleting the file *oslo.ini* in your private/bin directory. Normally this will not be required.

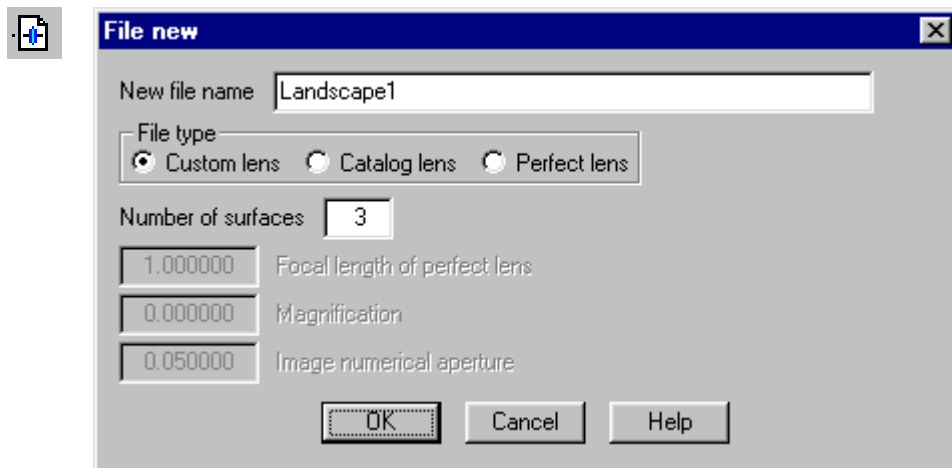
The steps in the exercise are as follows:

- **Lens entry** - Enter a convex-plano lens with a displaced aperture stop behind the lens.
- **Lens Drawing** - Set up the lens drawing conditions to show desired ray trajectories.
- **Optimization** - Optimize the lens so it has no coma, a focal length of 100, and covers a field of ± 20 degrees at an aperture of $f/10$.
- **Slider-wheel design** - Attach sliders to parameters so you can analyze trade-offs.

You will see during the exercise that design with OSLO is quite different from most software. You don't set up the system and then press an *Auto* key that lets the computer optimize it for you. Instead, you work in a highly interactive mode, taking small steps so you can understand what is happening as you progress towards a solution.

Lens entry

Click the New Lens tool in the main toolbar to open the following dialog box.



Enter the file name "Landscape1", select Custom lens and enter 3 surfaces, then click OK. A new spreadsheet will appear. Fill out the spreadsheet with the following data.

In the fixed area of the spreadsheet, set the Lens: cell to "Landscape Lens Exercise", the entrance beam radius to 5, and the field angle to 20. Leave the other cells with their default values.

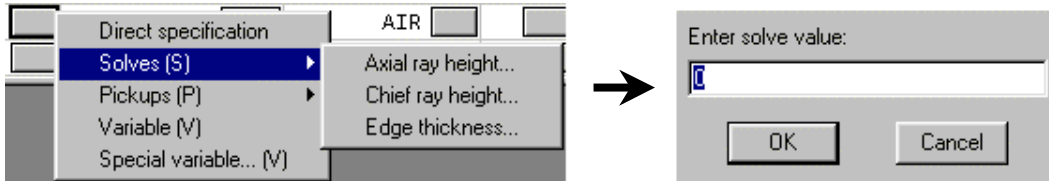
In the scrolled area of the spreadsheet, enter "BK7" for the glass on surface 1.

Introductory Exercise - Landscape Lens

In row 3, click the button in the Aperture Radius cell, then click Aperture Stop from the list that pops-up. Note that the row button for row 3 will now say AST, and A will be added to the button in the Aperture Radius column.

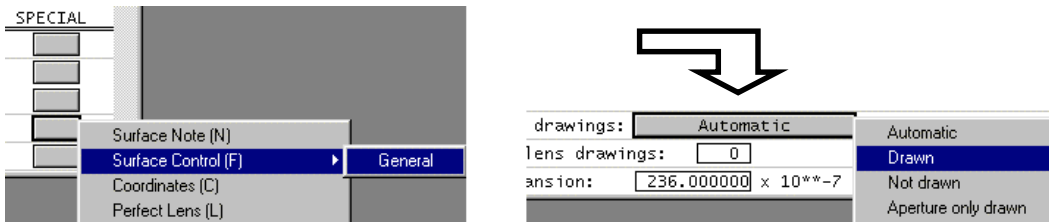
Now click in the Radius cell of row 1, enter 50 for the radius of curvature, then click on the thickness cell and enter 4 for the thickness. Click on the thickness cell for row 2 and enter 10.

In row 3, instead of entering a thickness directly, click on the button in the thickness cell, and select Axial ray height solve from the pop-up menu. This will cause a data-entry box to pop-up prompting for the value of the solve. Accept the default value (0) by clicking OK.



Click the button in the Special cell for row 3. From the list that pops up, select Surface control, and from the flyout menu, select General. This will cause a whole new spreadsheet to cover up the current spreadsheet. Click the button that says Automatic in the 5th row, and from the list that pops up, select Drawn. Then click the Green checkmark to close the spreadsheet and return to the surface data spreadsheet. You will see an F on the special button for row 3.

Repeat the above sequence for the special button in row 4.



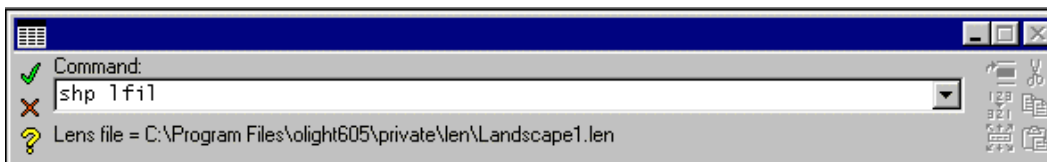
This completes the basic data entry for your lens. At this point, your spreadsheet should look like the following.

Surface Data						
Gen	Setup	Wavelength	Field Points	Variables	Draw Off	Group
Lens: Landscape 1		Zoom	1 of 1	Efl	96.749205	
Ent beam radius	5.000000	Field angle	20.000000	Primary wavln	0.587560	
SRF	RADIUS	THICKNESS	APERTURE RADIUS	GLASS	SPECIAL	
OBJ	0.000000	1.0000e+20	3.6397e+19	AIR		
1	50.000000	4.000000	10.290581	BK7	C	
2	0.000000	10.000000	9.050250	AIR		
AST	0.000000	84.112075	4.346913	AIR	F	
IMS	0.000000	0.000000	35.213831		F	

Check to make sure you have the same Efl (focal length), and that the buttons have the same labels. The S on the thickness button means that the value was determined by a solve (the axial ray height solve). The S on the aperture buttons also means that the apertures were determined by solves; this is the default action for apertures and is based on the entrance beam radius and the field angle.

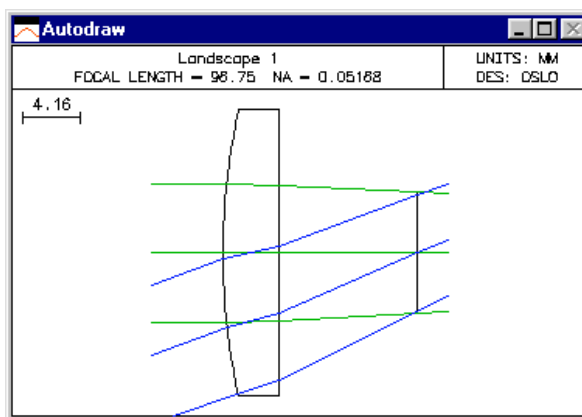
Click the Green check to close the spreadsheet and save the lens by clicking the Save lens toolbar button. You have already indicated the file name on the opening dialog box. The lens will be

saved in the current lens directory, which is normally your private directory. If you want to confirm where the lens is saved, you can enter the command *shp lfil* (*show_preference lensfile*) in the command line. The file name will appear in the message area.



Lens Drawing

Click the DRAW OFF button in the fixed area of the lens spreadsheet. A window labeled Autodraw will appear showing the lens that you have entered. If you click on a cell in the spreadsheet, you will see that the surface corresponding to the cell becomes dotted. If you were to change any data for the lens, the picture would be updated automatically.



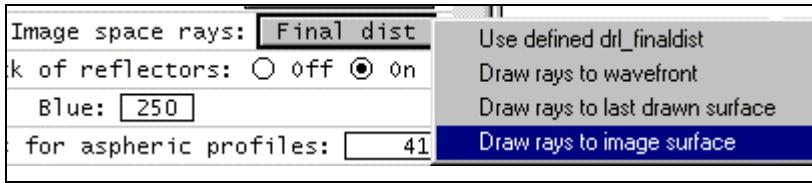
The Autodraw window is a special window that does not have the full functionality of a regular graphics window, but is updated automatically, which is not the case for ordinary graphics windows. The lens picture that is shown is drawn according to the standard lens conditions used for all lens drawings. In order to illustrate this exercise better, we are going to change one of the operating conditions so that the ray trajectories are drawn all the way to the image surface.

From the Lens menu on the main tool bar, select the bottom item, Lens Drawing Conditions. This will cause the following spreadsheet to cover the lens spreadsheet:

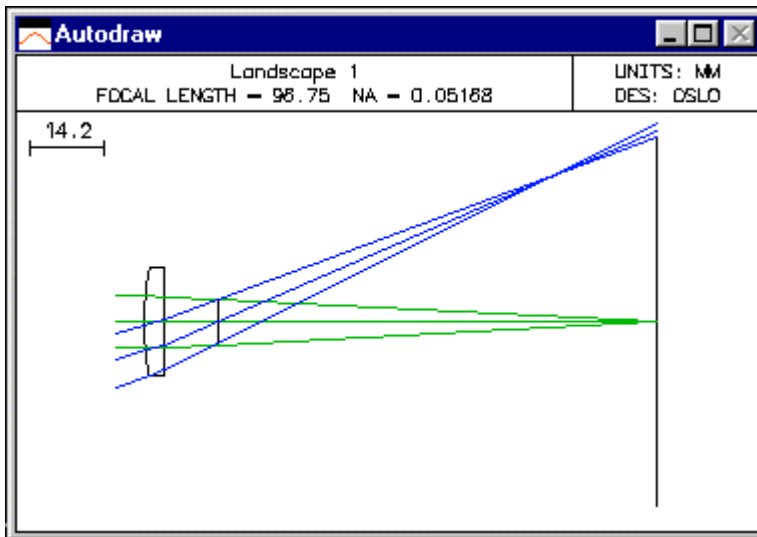
Initial distance:	<input type="text" value="0.000000"/>	Final distance:	<input type="text" value="0.000000"/>						
Horizontal view angle:	<input type="text" value="240"/>	Vertical view angle:	<input type="text" value="30"/>						
First surface to draw:	<input type="text" value="0"/>	Last surface to draw:	<input type="text" value="0"/>						
X shift:	<input type="text" value="0.000000"/>	Y shift:	<input type="text" value="0.000000"/>						
DXF/IGES view:		<input type="button" value="Unconverted"/>							
Apertures:	<input type="button" value="Quadrant"/>	Rings:	<input type="text" value="3"/>						
Spokes:	<input type="text" value="4"/>	Image space rays:	<input type="button" value="Final dist"/>						
Draw aperture stop:	<input checked="" type="radio"/> off <input type="radio"/> on	Hatch back of reflectors:	<input type="radio"/> off <input checked="" type="radio"/> on						
Shaded solid color - Red:	<input type="text" value="175"/>	Green:	<input type="text" value="185"/>						
Blue:	<input type="text" value="250"/>								
Number of field points for ray fans:	<input type="text" value="3"/>	Points for aspheric profiles:	<input type="text" value="41"/>						
Frac Y Obj	Frac X Obj	Rays	Min Pupil	Max Pupil	Offset	FY	FX	Wvn	Cfg
<input type="text" value="0.000000"/>	<input type="text" value="0.000000"/>	<input type="text" value="3"/>	<input type="text" value="-1.000000"/>	<input type="text" value="1.000000"/>	<input type="text" value="0.000000"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="text" value="1"/>	<input type="text" value="0"/>
<input type="text" value="0.700000"/>	<input type="text" value="0.000000"/>	<input type="text" value="0"/>	<input type="text" value="0.000000"/>	<input type="text" value="0.000000"/>	<input type="text" value="0.000000"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="text" value="1"/>	<input type="text" value="0"/>
<input type="text" value="1.000000"/>	<input type="text" value="0.000000"/>	<input type="text" value="3"/>	<input type="text" value="-1.000000"/>	<input type="text" value="1.000000"/>	<input type="text" value="0.000000"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="text" value="1"/>	<input type="text" value="0"/>

Introductory Exercise - Landscape Lens

The item to change on this spreadsheet is called Image space rays: You should change it from Final dist to Draw to image surface.



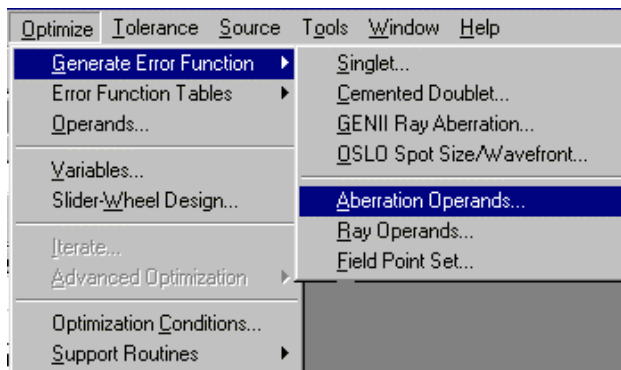
After making the modification, click the Green check to close the spreadsheet. The lens spreadsheet will reappear, and the Autodraw window will be updated as follows.



Now you can see the details of image formation by the lens. Obviously the off-axis image is not so good. The main aberration that you see is field curvature; the off-axis image is not formed on a plane surface perpendicular to the axis, but rather on a curved surface. Unfortunately, there is not too much that can be done about this.

Optimization

In order to study the imaging of this lens in more detail, we will set up a small optimization task. We will define an error function that makes the focal length exactly 100mm, and also eliminates the Seidel coma from the image. To do this, click the optimize button on the main menu, then navigate to the generate error function >> Aberration Operands item. This will pop up the spreadsheet shown on the following page.



OP	MODE	WGT	NAME	DEFINITION
1	Min	1.000000	PY	OCM1
2	Min	1.000000	PU	OCM2
3	Min	1.000000	PYC	OCM3
4	Min	1.000000	PUC	OCM4
5	Min	1.000000	PAC	OCM5
6	Min	1.000000	PLC	OCM6
7	Min	1.000000	SAC	OCM7
8	Min	1.000000	SLC	OCM8
9	Min	1.000000	SA3	OCM9
10	Min	1.000000	CMA3	OCM10
11	Min	1.000000	AST3	OCM11
12	Min	1.000000	PTZ3	OCM12
13	Min	1.000000	DIS3	OCM13
14	Min	1.000000	SA5	OCM14
15	Min	1.000000	CMA5	OCM15
16	Min	1.000000	AST5	OCM16
17	Min	1.000000	PTZ5	OCM17
18	Min	1.000000	DIS5	OCM18
19	Min	1.000000	SA7	OCM19
20	Min	1.000000	TOTAL_SPH	OCM20
21	Min	1.000000	EFL	OCM21

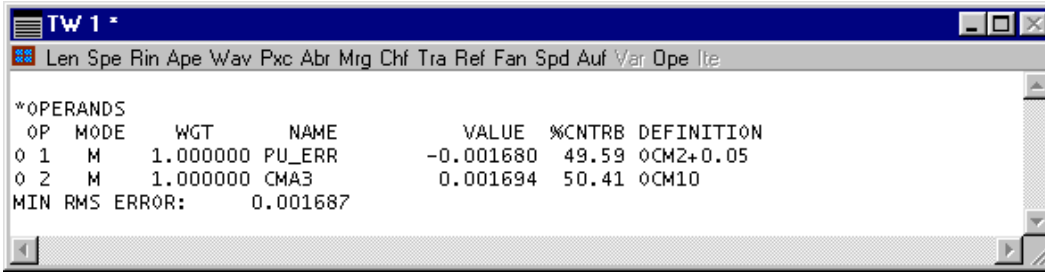
At this point, we need to decide which of the terms will be used for the error function. The initial spreadsheet lists all of the possibilities, and you must delete the ones you don't want. In this case, the operands to keep are PU and CMA3. All the other should be deleted. To do this, click and drag from row buttons 11-21, then press the delete key. Next click and drag from row buttons 3-9, then press the delete key. Finally, click row button 1 and press the delete key. The spreadsheet should now appear as follows.

OP	MODE	WGT	NAME	DEFINITION
1	Min	1.000000	PU	OCM2
2	Min	1.000000	CMA3	OCM10

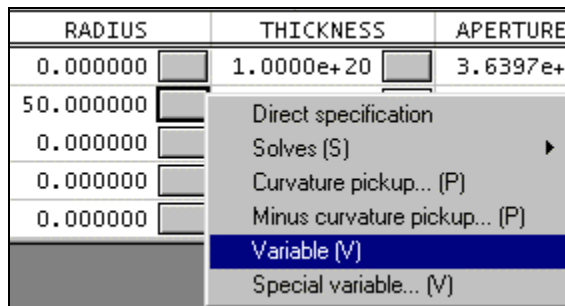
Next, we need to modify the definition of the first operand, PU. In OSLO, all operands are targeted to zero. That is, the optimization algorithm will minimize all of its individual terms. The PU operand is the axial ray slope leaving the lens. We don't want this to be zero, but rather the value that will make the lens have an f-number of 10, which was prescribed in the initial data. Since the f-number is $-1/2*PU$ (when the object is at infinity) the desired value for PU is $-.05$. We accommodate this by modifying the definition of the first operand to be $OCM2+0.05$, and we will also change the name to PU_ERR. Click on the NAME and DEFINITION cells and enter the new data.

OP	MODE	WGT	NAME	DEFINITION
1	Min	1.000000	PU_ERR	OCM2+0.05
2	Min	1.000000	CMA3	OCM10

Now click the Green check to close the spreadsheet, and click the Ope button in the text window. The current operand values, together with the current value of the error function, will be shown.



Now the operands and the error functions are defined. If the error function is minimized to zero, we will have a $f/10$ lens with a focal length of 100, since the focal length is $-PU*EBR$ (EBR is the entrance beam radius), and the coma will be zero. We need to specify the variables that are to be used to achieve this. The ones that we will use are the curvature of the first surface ($CV 1$), and the distance between the lens and the aperture stop ($TH 2$). In the lens spreadsheet, click on the button in the Radius of Curvature cell for row 1, and select Variable from the options list.

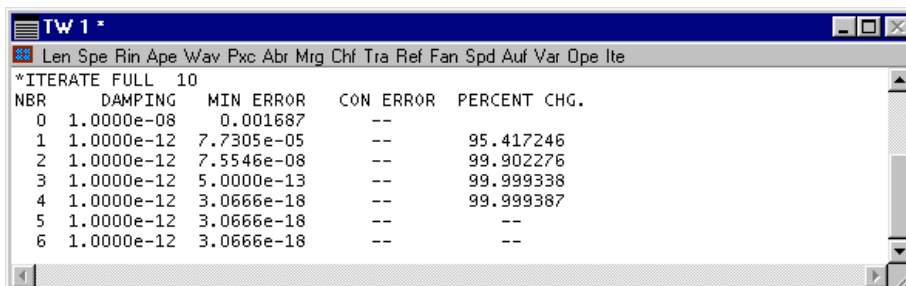


Repeat the same procedure for using the button in the Thickness cell for row 2. The buttons should now have V (variable) on them.

RADIUS	THICKNESS
0.000000	1.0000e+20
50.000000	4.000000
0.000000	10.000000
0.000000	84.112075

Assuming that they do, you should now be ready to optimize the lens. Before doing this, close the lens spreadsheet (Green check) and immediately re-open it. The reason for doing this is that OSLO gives you the capability, by canceling (red X) from a spreadsheet, to revert to the system that existed when the spreadsheet was opened. By closing and opening the spreadsheet in this way, you establish a base system to revert to.

To optimize, click the Ite button in the text window (you may have noticed that this button was disabled until you entered both operands and variables). The text window should show the following output. You see that the error function has been reduced to essentially zero, indicating that the variables were appropriate for solving the problem. The lens spreadsheet shows their current values.



Gen	Setup	Wavelength	Field Points	Variables	Draw Off	Group	Notes
Lens: Landscape 1				Zoom	1 of 1	Efl	100.000000
Ent beam radius		5.000000	Field angle	20.000000	Primary wavln		0.587560
SRF	RADIUS	THICKNESS	APERTURE RADIUS	GLASS	SPECIAL		
OBJ	0.000000	1.0000e+20	3.6397e+19	AIR			
1	51.680011	V	4.000000	S	BK7	C	
2	0.000000	V	9.135802	S	AIR		
AST	0.000000	S	4.411353	AS	AIR		F
IMS	0.000000		36.397023	S			F

You can see in the lens spreadsheet that the Efl is exactly 100, as it must be in order to correct the operands. Click the Abr button in the text window, and you see all the aberration data for the lens, showing that PU = -0.05, and that the Seidel coma CMA3 is zero, as requested.


*PARAXIAL TRACE						
SRF	PY	PU	PI	PYC	PUC	PIC
4	--	-0.050000	-0.050000	36.397023	0.412538	0.412538
*CHROMATIC ABERRATIONS						
SRF	PAC	SAC	PLC	SLC		
SUM	-0.077222	-0.053466	0.069914	0.048406		
*SEIDEL ABERRATIONS						
SRF	SA3	CMA3	AST3	PTZ3	DIS3	
SUM	-0.013612	4.3368e-18	-0.313405	-0.218345	1.101908	
*FIFTH-ORDER ABERRATIONS						
SRF	SA5	CMA5	AST5	PTZ5	DIS5	SA7
SUM	-9.6466e-05	-5.9253e-05	-0.020638	0.017274	0.070480	-7.7081e-07

At this point, the initial task has been completed. Close the spreadsheet (Green check) and save the lens. This will become the base system for the next phase of the exercise, which shows how to use OSLO's slider window to study the effects of changing other lens parameters. You might want to save another copy of this system under a different file name (e.g. landscape_bkp.len) so you can always return to a defined starting point, even if the current lens data is somehow corrupted.

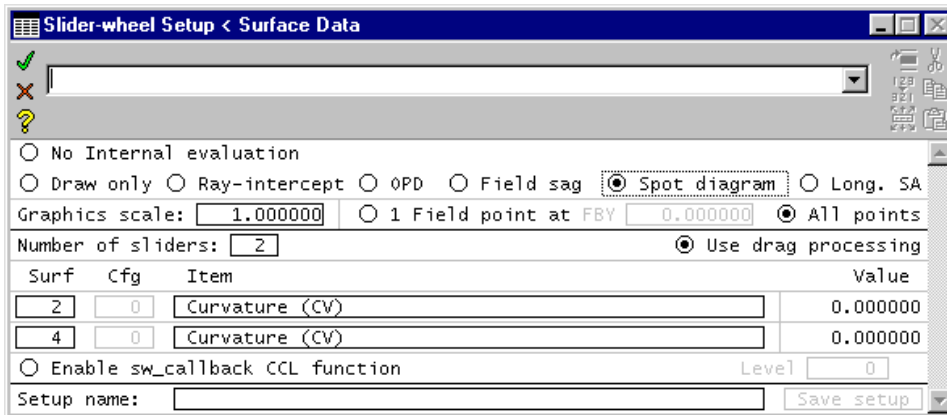
Slider-wheel design

A singlet such as the landscape lens doesn't have very many degrees of freedom, but in this exercise, we have constrained the back surface of the lens to be plane. Actually, this should not be called a landscape lens; so far all we have done is to find the position of the aperture stop that removes the coma for a convex-plano lens. A landscape lens is normally a meniscus form. Next we show how to use OSLO's slider-wheel window to find the optimum form. We also show the effects of using a curved image surface.

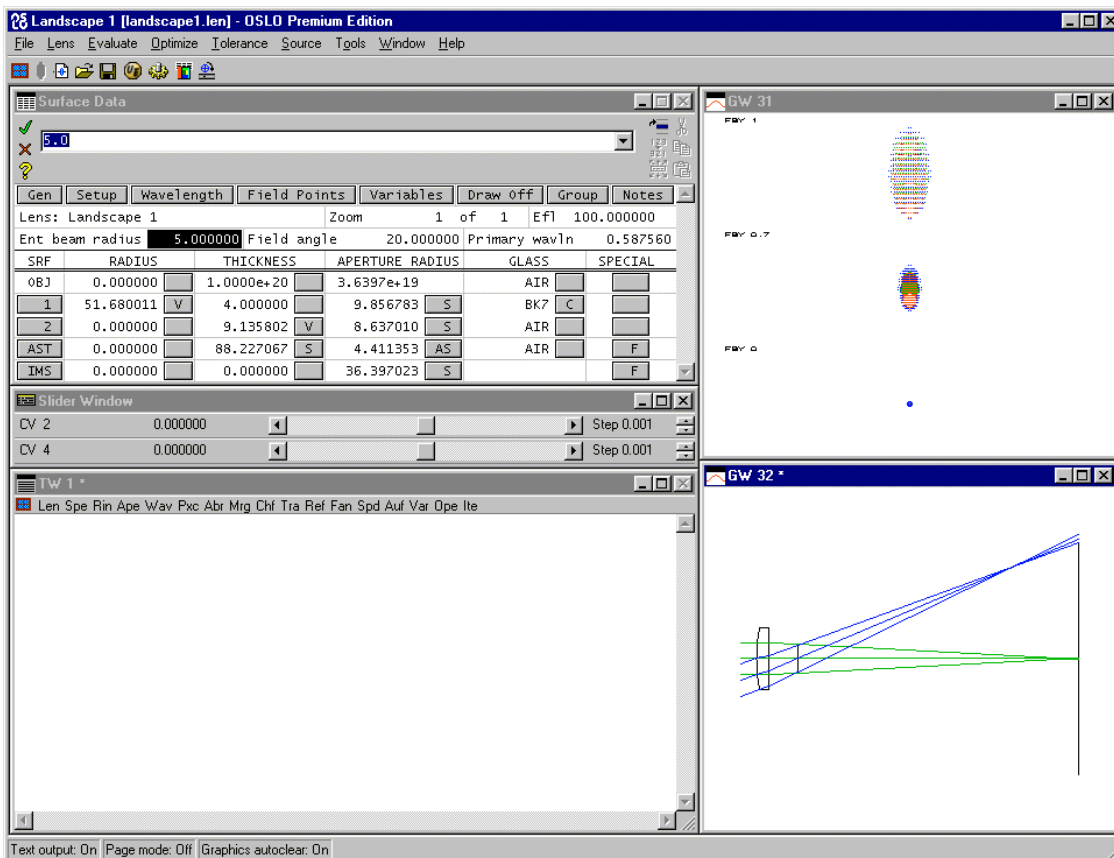
First, we set up a basic slider-wheel window, attaching the curvatures of surfaces 2 and 4 to sliders. Later, we show how to use the full power of OSLO's optimization callback to continuously optimize the system while dragging a slider or rotating the mouse wheel.

 With the landscape lens as set up in the preceding portion of this exercise, click the slider-wheel tool in the main tool bar to open the slider-wheel spreadsheet. This spreadsheet is used to set up slider-wheel windows.

Initially, set the options shown in the figure below. That is, click Spot diagram, set the Graphics scale to 1.0, and select All points. Leave the number of sliders at the default (2), enter 2 and 4 in the Surfs column, and set the parameters to CV for both sliders.

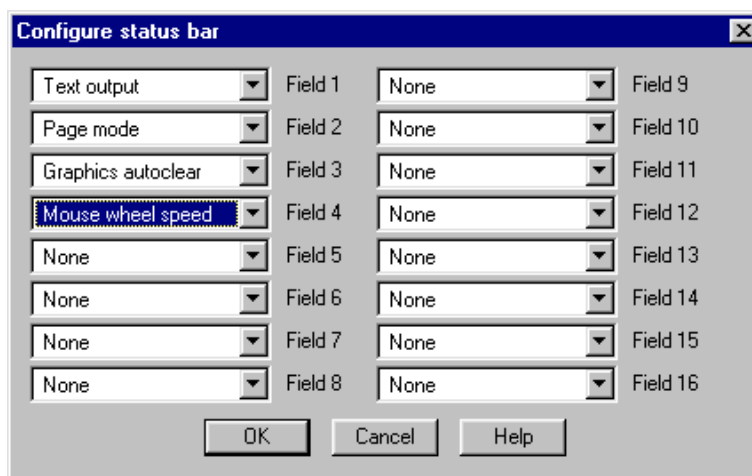


After you have entered the data in the slider-wheel spreadsheet, close the spreadsheet. You should see a slider-wheel window and two additional graphics windows (31 & 32) on your screen. If these windows have toolbars, remove them by left-clicking to select a window, then right-clicking in the window and choosing the Remove Toolbar item. Since the slider-wheel maintains its own graphics windows, you should minimize the normal graphics output window (GW1). Next, use the Tile windows command on the main window toolbar to tile the windows. Your overall screen should look similar to the one below.



The slider-wheel window appears below the spreadsheet, with the two sliders that we set up. You can manipulate sliders by dragging or clicking as you would with a normal scroll bar (the sliders are in fact scroll bars), or by using the mouse wheel. The mouse wheel is usually preferred (if it is installed and properly supported on your computer) but you may need to change its settings. To use the mouse wheel with a slider, place the pointer *anywhere* in a slider and rotate the wheel.

To change the settings of your mouse wheel, you need to first find out what they are at present. To do this, double click on the status bar in the main OSLO window. This will pop up a dialog box as shown below. Navigate to the first empty field, click on the combo box, and select Mouse wheel speed, as shown.



Click OK to dismiss the box. You should see an additional field in the status bar, as shown. OSLO supports two mouse-wheel speeds: fast and slow. Slow means one event per wheel notch; fast means several events per wheel notch (however many you set up in the mouse dialog in your computer's control panel). You change between fast and slow by clicking the mouse button itself. Generally in OSLO, you will prefer Slow.

If you experiment, you will see that the slider-wheel window can be dragged anywhere on the display, even outside the OSLO window. Also, you will see that you can drag the window width down to the point where the slider itself completely disappears. If you use the mouse wheel, this setting may be preferable because it minimizes the use of screen space while still providing full functionality. You will see that the step size in the parameter taken per click can be adjusted using the Step control at the right-hand end of the window.

For now, it is best to leave the slider window where the Tile windows command puts it, with its default settings. To reset these, enter the original values (0) of CV[2] and CV[4] in the lens spreadsheet, then click the slider-window tool in the main toolbar and close the slider-wheel setup spreadsheet immediately to update the slider-wheel window (note that the settings for the analysis options are not changed).

Returning to optics, you see that the spot diagrams show what the image looks like at the center, zone, and edge of the field. The elliptical shape is characteristic of a system with astigmatism, but no coma. Now you can drag the sliders or rotate your mouse wheel to see what happens. Actually, if you do not use the mouse wheel, you would be better advised to click on a button at the end of a slider rather than drag, so you can move in small enough steps.

What you see is pretty much what you would expect. If you make CV[2] negative, the lens becomes positive and the focal length gets smaller. The image quality changes, since the system is no longer free of coma. If you change the curvature too much (< -0.033) the ray trace will fail. If this happens, click OK to dismiss the error box, then move the slider back towards the center. If you make CV[2] positive, the lens becomes first a meniscus with a long focal length, but eventually the lens becomes negative; the beam diverges and the display blows up. After you have experimented with CV[2], set it back to zero.

When you change CV[4] (the curvature of the image surface) you see that by making it negative, you can improve the size of the image off-axis, and in fact you can find a position where there is a line focus in the horizontal direction (tangential focus), or the vertical direction (sagittal focus). This is indicative of a system with no coma. If you set CV[2] to a value where the system has substantial coma (try -.020) you will not be able to achieve this. This shows what designers mean when they say that you can't focus out coma.

The slider-wheel analysis you have observed so far is perhaps interesting, but the real power of sliders in OSLO comes when you allow the program to re-optimize the system as you drag a slider. To do this, reset the curvatures to zero, then re-open the slider spreadsheet, set *Enable sw_callback CCL function* to On, and set the level to 2.

<input type="radio"/> No Internal evaluation			
<input type="radio"/> Draw only <input type="radio"/> Ray-intercept <input type="radio"/> OPD <input type="radio"/> Field sag <input checked="" type="radio"/> Spot diagram <input type="radio"/> Long. SA			
Graphics scale:	<input type="text" value="1.000000"/>	<input type="radio"/> 1 Field point at FBY	<input type="text" value="0.000000"/> <input checked="" type="radio"/> All points
Number of sliders:	<input type="text" value="2"/>	<input checked="" type="radio"/> Use drag processing	
Surf	Cfg	Item	Value
<input type="text" value="2"/>	<input type="text" value="0"/>	Curvature (CV)	0.000000
<input type="text" value="4"/>	<input type="text" value="0"/>	Curvature (CV)	0.000000
<input checked="" type="radio"/> Enable sw_callback CCL function			Level <input type="text" value="2"/>
Setup name:	<input type="text"/>		<input type="button" value="Save setup"/>

A callback function is a command that gets executed whenever some event happens in the program. Callback functions are the essence of a windows program, and OSLO has several of them, many written in CCL (such as *sw_callback*). Generally, these are not complicated routines written by computer programmers, but simple actions set up by users. The default *sw_callback* routine is as follows.

```
Cmd Sw_callback(int cblevel, int item, int srf)
{
    if (cblevel)
        ite cblevel;
}
```

When *Enable sw_callback CCL function* is turned on, this routine is executed every time a graphic slider is moved, or the mouse-wheel is rotated when the pointer is in the slider-wheel window. The arguments of this function are *cblevel*, the value entered in the Level field in the slider-wheel spreadsheet; *item*, the type of variable (CV = 1, TH = 3, etc.) that was changed, and *srf*, the number of the surface associated with that variable. These arguments can be used to produce whatever action you want. In the present case, the function only looks to see if *cblevel* is non-zero, and if it is, the function carries out that many design iterations.

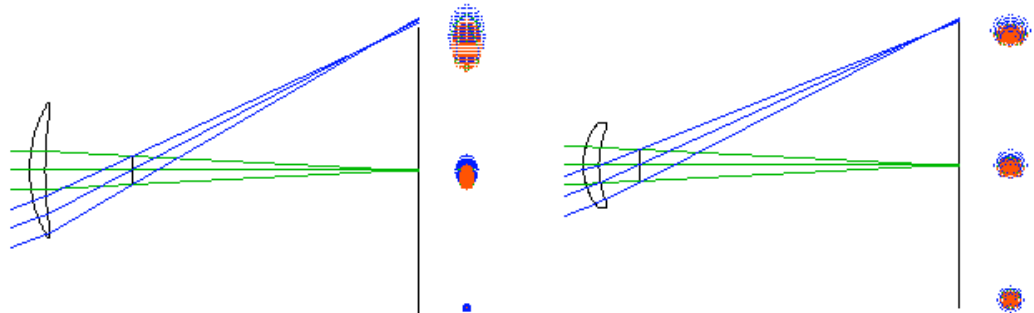
To see what this does, close the slider window. You should not see any change in the system, because it was already optimized, but you will see text out similar to the following. As you move a slider, you will see the numbers flash in the text output window as the system is re-optimized.

TW 1 *																
Len	Spe	Rin	Ape	Wav	Pxc	Abr	Mrg	Chf	Tra	Ref	Fan	Spd	Auf	Var	Ope	Ite
*ITERATE FULL					2											
NBR	DAMPING	MIN ERROR	CON ERROR	PERCENT CHG.												
0	1.0000e-08	3.6409e-17	--													
1	1.0000e-08	3.0666e-18	--	91.577286												
2	1.0000e-08	--	--	100.000000												

Introductory Exercise - Landscape Lens

47

In the graphics windows, you will see a quite different scenario from before. To begin, drag the CV[2] slider to make it more positive. This makes the lens into a meniscus, but now the focal length is held constant by virtue of the first curvature's being continually re-optimized. Moreover, the aperture stop surface moves as the coma is continually re-optimized to zero. The image surface is held in the paraxial image plane by the height solve on surface 3, and the diameter of the lens is adjusted by the aperture solves so that the beam from the edge of the field of view can pass through the system. The effect of all of this is that the stop initially moves away from the lens as it becomes a meniscus, but as the bending becomes larger, the stop shift reverses and the aperture stop moves back towards the lens. The following shows what you should observe when the surface 2 curvature has values 0.01, and 0.04 (n.b. the images pertain to center-to-edge, not to bottom-to-top; the bottom spot is the on-axis image).



Of course, the spherical aberration of the system is not constant, and the improvement in image quality at the edge of the field comes at the expense of a poorer on-axis image. This leads to a consideration of what might be accomplished by using an aspheric lens, but that is beyond the scope of this introductory exercise, whose purpose is to introduce OSLO data entry and the general steps for using the slider-wheel window.