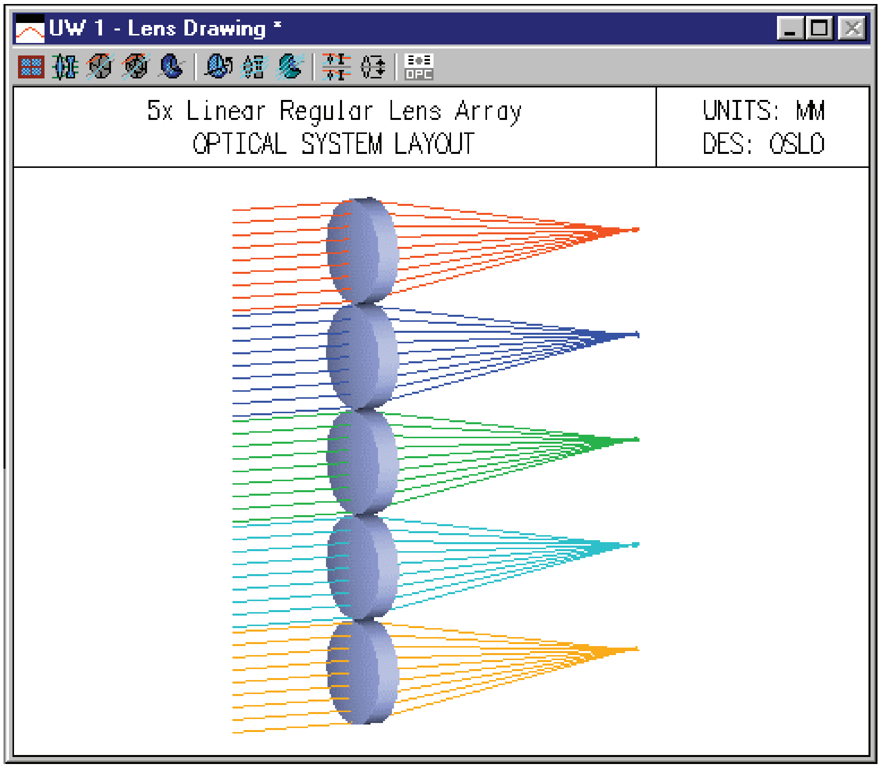
Array ray tracing

Regular array

There are many applications of lens arrays, ranging from micro-optics switching systems to multiple mirror telescopes. The following example shows a simple system comprising a 5-element linear array of lenses, set up as a regular array.



Gen	Setup	Wavelength	Field Points	Variables	Draw Off	Surfs	Notes
Lens: 5x Linear Regular Lens Array		Zoom 1 of 1		Efl 29.527797			
Ent beam radius 5.000000		Field angle 5.7296e-05		Primary wavln 0.587562			

SRF	RADIUS	THICKNESS	APERTURE RADIUS	GLASS	SPECIAL
OBJ	0.000000	1.0000e+20	1.0000e+14	AIR	
AST	0.000000	0.000000	25.000000 A	AIR	F
2	30.000000	3.000000	20.000000 X	BK7 C	
3	-30.000000	28.521734 S	20.000000 X	AIR	C
IMS	0.000000	3.000000	20.000000		

The array data is entered using SPECIAL>>Surface Control>>Regular Lens Array. Since there is a single row of lenses, the *x* spacing is 0. The number of lenses is controlled by the aperture of the channel surface (surface 1). Only the vertex of each channel needs to be within the aperture of the channel surface to be included in the array, although here the aperture has been set to enclose the entire array surface.

```
*LENS ARRAY DATA
 SRF 1:
 TYPE Regular              END SURF 3                DRAW ALL CHANNELS: Yes
 X SPACING      --         Y SPACING   10.000000     Y OFFSET      --
```

The aperture of the elements themselves are determined by rectangular special apertures on surface 2 and 3:

```
*APERTURES
 SRF   TYPE APERTURE RADIUS
  2     SPC    20.000000
     Special Aperture Group 0:
     A  ATP    Rectangle  AAC     Transmit  AAN       --
        AX1    -5.000000  AX2     5.000000  AY1    -5.000000  AY2     5.000000

  3     SPC    20.000000
     Special Aperture Group 0:
     A  ATP    Rectangle  AAC     Transmit  AAN       --
        AX1    -5.000000  AX2     5.000000  AY1    -5.000000  AY2     5.000000
```

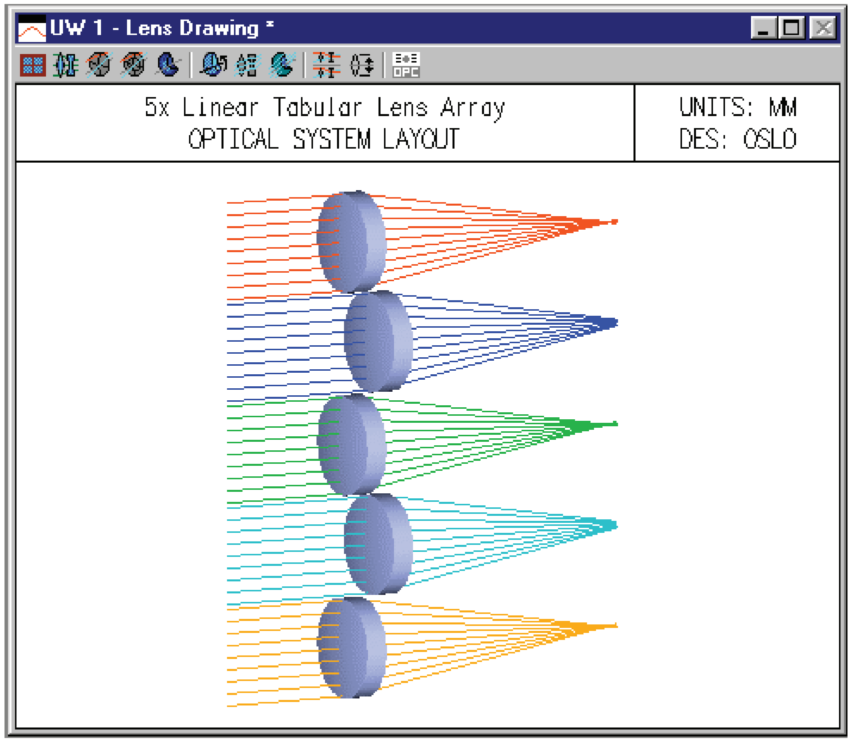
The system shown here has five light sources. In OSLO, these are modeled as separate field points. The required lens drawing conditions (non-default only) are shown below.

```
*CONDITIONS: LENS DRAWING
   Drawn apertures (solid):       Full    Image space rays:        Image srf
   Number of field points (rays):    5    DXF/IGES file view:     Unconverted
   Fpt Frac Y Obj Frac X Obj Rays Min Pupil  Max Pupil    Offset  Fan Wvn Cfg
     1     1.00000      --      9   -0.95000    0.95000      --     Y   1   0
     2     1.00000      --      9    1.05000    2.95000      --     Y   1   0
     3     1.00000      --      9    3.05000    4.95000      --     Y   1   0
     4     1.00000      --      9   -2.95000   -1.05000      --     Y   1   0
     5     1.00000      --      9   -4.95000   -3.05000      --     Y   1   0
```

Since a spot diagram pertains to a single field point, the data obtained for an array of the type shown here may not be what is desired, and it may be preferable to construct custom CCL commands to carry out evaluation that is tailored to the system at hand. Please note that since lens arrays use **rco** (return coordinates) surfaces, paraxial analysis will not be correct. In the system here, a 3mm image focus shift has been added to the paraxial solve value, to make up for the thickness of the array elements.

Tabular array

This example shows a modification the preceding regular array, to make a tabular array. Two of the elements have been offset to illustrate the difference between the two types.



The main surface data spreadsheet is identical to the one for the regular array. The difference is in the array data spreadsheet (SPECIAL>>Surface Control>>Tabular Lens Array), which enumerates the coordinates of the vertices of each element (channel) in the array. Note that a *z* displacement has been added to elements 2 and 3. This is not accounted for in the above drawing, which shows rays traced to the nominal image surface, from a field point 10 degrees off axis.

Surface 1 — Delete Lens Array

Array type: Tabular   Number of channels: 5   Draw all channels: (•) Yes ( ) No

End surface: 3

CH NBR	X CTR	Y CTR	Z CTR	TLA	TLB	TLC
1	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
2	0.000000	10.000000	3.000000	0.000000	0.000000	0.000000
3	0.000000	-10.000000	3.000000	0.000000	0.000000	0.000000
4	0.000000	20.000000	0.000000	0.000000	0.000000	0.000000
5	0.000000	-20.000000	0.000000	0.000000	0.000000	0.000000

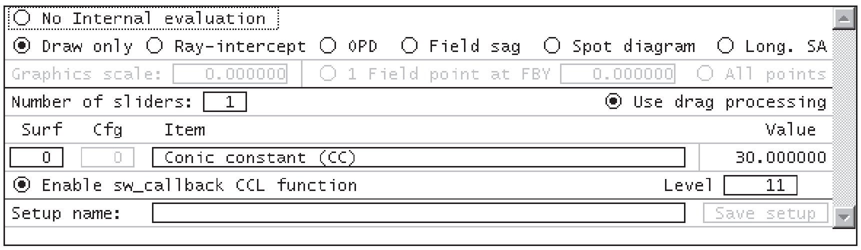
Array ray tracing is comparatively fast to non-sequential ray tracing, because surfaces are selected according to the nearest channel vertex rather than the actual surface. For many situations, this is a good model, but for this tabular array, it is not adequate for large field angles. To see this, it is worth attaching the field angle to a graphic slider so that it can be adjusted by dragging while the ray trajectories are observed.

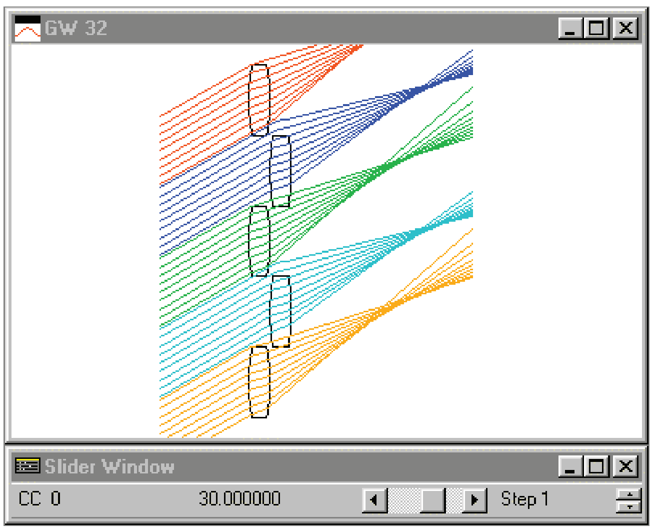
In order to attach the field angle to a slider, we use the same technique used elsewhere in these examples, making use of the fact that the conic constant of the object surface has no optical function when the surface is flat. We make a slider-wheel callback function as shown below, and put it in the private CCL directory.

```
cmd Sw_callback(int cblevel, int item, int srf)
```

```
{
        if (cblevel == 11)
        {
                stp outp off;
                ang cc[0];
                stp outp on;
        }
        else
                ite cblevel;
}
```

After recompiling the private CCL, we setup a slider-wheel window as follows.





When the setup window is closed, the slider-wheel window appears, and you can see that at wide angles, rays do not follow their actual trajectories, because of the way that channels are selected. This is not a problem for narrow fields or when surfaces are not displaced from the channel surface, as you can verify by manipulating the slider.

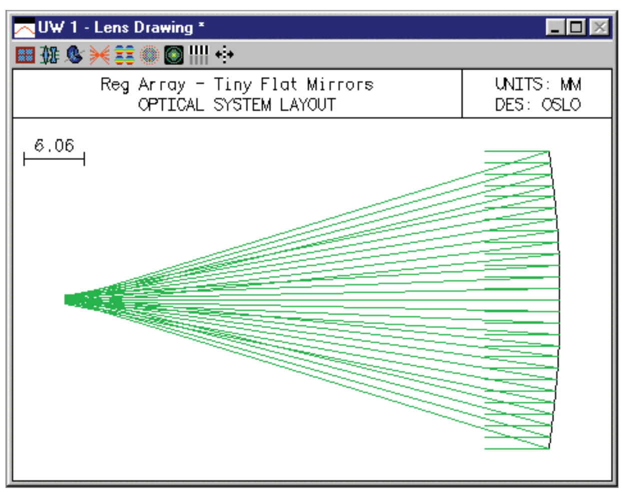
Note that for the slider to work properly in this example, the Fractional Y object height for all the field points must be set to 1, as shown in the table below. You may also note that it is not possible to set the field angle to zero using the slider. This is a feature of OSLO, which automatically converts field angles of 0.0 to 1 micro-degree, since 0.0 is not an allowed value for the paraxial field angle.

```
*CONDITIONS: LENS DRAWING
   Drawn apertures (solid):        Full    Image space rays:        Image srf
   Number of field points (rays):    5     DXF/IGES file view:     Unconverted
   Fpt Frac Y Obj Frac X Obj Rays Min Pupil   Max Pupil     Offset   Fan Wvn Cfg
    1       1.00000     --       9   -0.95000    0.95000      --      Y   1   0
    2       1.00000     --       9    1.05000    2.95000      --      Y   1   0
    3       1.00000     --       9    3.05000    4.95000      --      Y   1   0
    4       1.00000     --       9   -2.95000   -1.05000      --      Y   1   0
    5       1.00000     --       9   -4.95000   -3.05000      --      Y   1   0
```

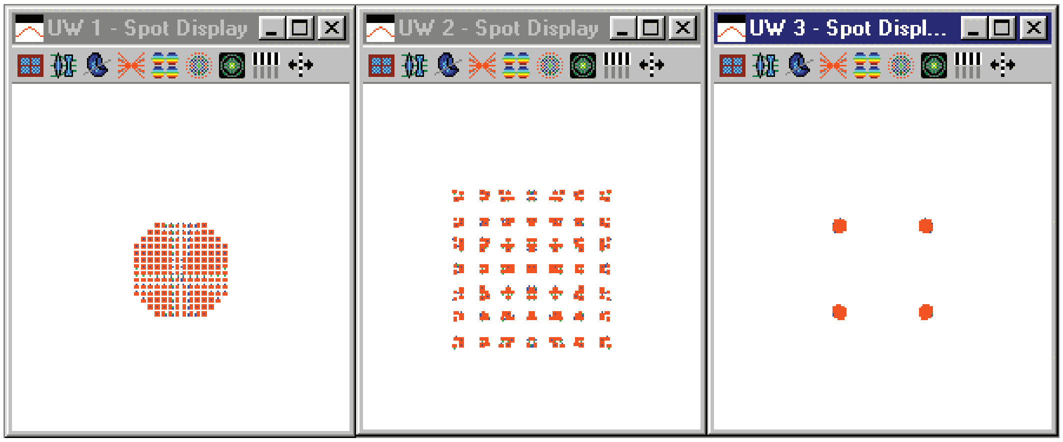
2D Array

As an example of a 2D array, we show a system comprising a large number of small flat mirrors mounted on a parabolic substrate with a focal length of 50mm and a diameter of 30mm (f/1.67). The mirrors have 1 mm width, and a center spacing of 1 mm. The data for the system (mirorary.len) are shown below.

```
*LENS DATA
Reg Array - Tiny Flat Mirrors
 SRF      RADIUS       THICKNESS    APERTURE RADIUS       GLASS  SPE  NOTE
 OBJ        --        1.0000e+20     8.7489e+18             AIR

 AST   -100.000000        --          15.000000 A           AIR   *

  2         --        -50.000000       0.707100 KX      REFLECT   *

 IMS        --            --          25.000000

*CONIC AND POLYNOMIAL ASPHERIC DATA
 SRF         CC           AD          AE          AF          AG
  1      -1.000000        --          --          --          --

*TILT/DECENTER DATA
  2      RCO    1

*LENS ARRAY DATA
 SRF 1:
 TYPE Regular              END SURF 2                 DRAW ALL CHANNELS:  No
 X SPACING     1.000000      Y SPACING    1.000000     Y OFFSET       --

*APERTURES
 SRF    TYPE APERTURE RADIUS
  0      SPC   8.7489e+18
  1      SPC     15.000000
  2      SPC      0.707100   CHK

     Special Aperture Group 0:
     A   ATP     Rectangle  AAC      Transmit  AAN       --
         AX1     -0.500000  AX2      0.500000  AY1    -0.500000   AY2     0.500000

  3      SPC     25.000000
```



Evaluating the system using a spot diagram produces results that depend strongly on the aperture divisions used, and the focus shift from the focal point of the parabolic substrate. (Since the system has only flat mirrors, it actually has an infinite focal length.) The figure below shows spot diagrams for various aperture divisions (15, 17.5, and 20), with a focal shift of 0.1 mm. The command used was **pls cen sym 0.1 1.0.**



The explanation for these curious results is that there is aliasing between the ray grid and the mirror grid. The overall diameter of the paraboloid is 30 mm, so when APDIV = 15, there is one ray that strikes the center of every other mirror. When APDIV = 17.5, the mirror spacing and the ray spacing are not coupled, so rays hit in nearly random points on the mirrors, and we see a (reversed) shadow of the 1 mm square mirrors. When APDIV = 20, no rays strike the center of a mirror, but all rays strike one of four possible locations on a mirror. This leads to the four-dot pattern shown above, which of course bears no similarity to the real light distribution. (The center pattern above gives the closest approximation to the real light distribution.)

When using spot diagrams (or any type of evaluation routine) with lens arrays, it is well to be aware of the possibility of aliasing effects between the ray grids used for evaluation, and the lens array grid itself. Often the best solution to these types of problems is to use random ray tracing. The figure below, for example, shows the image distribution computed using the xsource routine (Source>>Pixelated Object), using a small disc object that subtends a field angle of 0.01 degrees.