

## Lesson 40. An Aspheric Camera Lens from Scratch

When developing a modern cell-phone camera lens or a pinhole spy camera, designers are resorting more and more to using multiple aspheric surfaces. These are typically embodied as small plastic elements, and even though the molds are expensive to machine, the lenses can be produced in quantity at very low cost. It is even possible to mold mounting flanges directly onto the elements, making assembly simpler and enabling some dimensions to be held to very tight tolerances.

To help in designing such systems, DSEARCH™ can do a global search for systems with aspherics. Users are encouraged to read about this powerful feature in the SYNOPSIS™ User's Manual. We give here an example of how to use DSEARCH for a typical system.

This will be our input to DSEARCH:

```
PROJ          ! start project timer
CCW           ! clear command window
CORE 16      ! use 16 cores for speed

DSEARCH 1 QUIET      ! start DSEARCH; put best lens in library location 1
SYSTEM           ! define the system specs
ID DSEARCH ASPHERIC CAMERA LENS ! identification
OBB 0 41.3 .285    ! infinite object, semi field 41.3 degrees, semi ap. 0.285
UNI MM          ! lens will be in millimeters
WAVL CDF        ! use visual wavelengths at C, d, and F lines
END             ! end of system section

GOALS          ! define the goals here
ELEMENTS 5      ! we want a four-element lens with a cover glass
BACK 0.4 SET    ! ask for 0.4 mm back focus distance
FNUM 2.7 10     ! ask for F/2.7, weight of 10
THSTART 1      ! global search use thicknesses 1 mm
RSTART 30      ! and starting radius of 30 mm
ASPH Q         ! use conic constant even in quick mode
ASPH 3         ! allow three aspheric terms: CC, 4th, 6th power of aperture

DELAY OFF      ! these runs are fast, so don't ask to continue on timeout
ANNEAL 10 1 Q  ! anneal each case, temp 10 degrees, cool 1, including quick
SNAP 5         ! redraw PAD screen every five passes
STOP FIRST     ! put the stop in front
STOP FIXED     ! and keep it there
QUICK 50 50    ! run quick mode 50 passes, then real mode 50
NGRID 6        ! 6x6 grid of rays in pupil
NPASS 50       ! 50 passes in the MACro when finished
TOPD           ! correct both transverse aberrations and OPDs
FOV 0 .2 .4 .6 .8 1 ! correct six field points
FWT 5 4 3 3 3 3 ! with these weights
COVER .3 1.51872 64 ! the cover glass will be 0.3 mm thick with this GLM
PLASTIC 1 3 5 7 ! the four elements will be plastic
END            ! end of goals section

SPECIAL PANT   ! special PANT section starts here
RDR .001       ! these are tiny lenses, reduce derivative increments
TLIMIT 3 .1    ! limits on thicknesses and spaces
SLIMIT 5 .1
END            ! end of PANT section

SPECIAL AANT   ! start of special AANT section; these go into the merit fn.
ACC 1.0        ! center thickness no more than 1.0 mm
ACM .2 .1 .2   ! and no thinner than 0.2 mm
ACA 60         ! avoid critical angle; 60 degrees from surface normal
AEC .2 .1 .2   ! keep edges over 0.2 mm
M 1.35 10 A P YA 1 ! target the chief ray at three field points
M .945 10 A P YA .7 ! to control distortion
M .54 10 A P YA .4
```

```

END           ! end of AANT section
GO           ! DSEARCH runs
PROJ        ! when it is finished, see how long the run took.

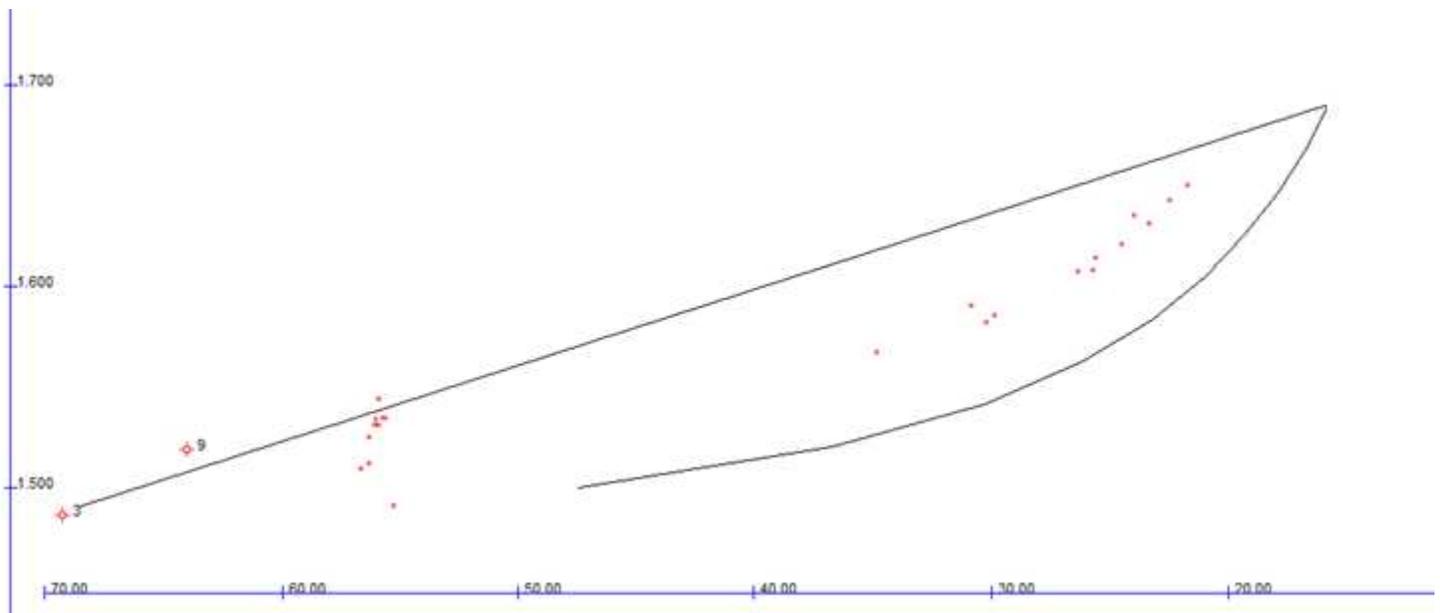
```

We should mention some subtle considerations here. First, these will be very tiny lenses, and the default edge-control target (1 mm) that DSEARCH puts in its optimization MACro is too thick. So we override that with our own AEC monitor. Also, the default minimum airspace and thickness monitor of 1 mm, also too thick, is overridden with our own ACM of 0.2 mm. Our added ACC monitor will not let thickness grow to more than 1.0 mm, overriding the default of 25.4 mm.

Those monitors are very weak as entered. This is done on purpose: if you strongly control those items, DSEARCH will tend to favor designs that do not offend them – but we want the program to favor designs with small *image errors*, and don't really care much about the mechanical properties at first. When we get a good design, we can easily modify those monitors, increasing the weight to make the design practical later on.

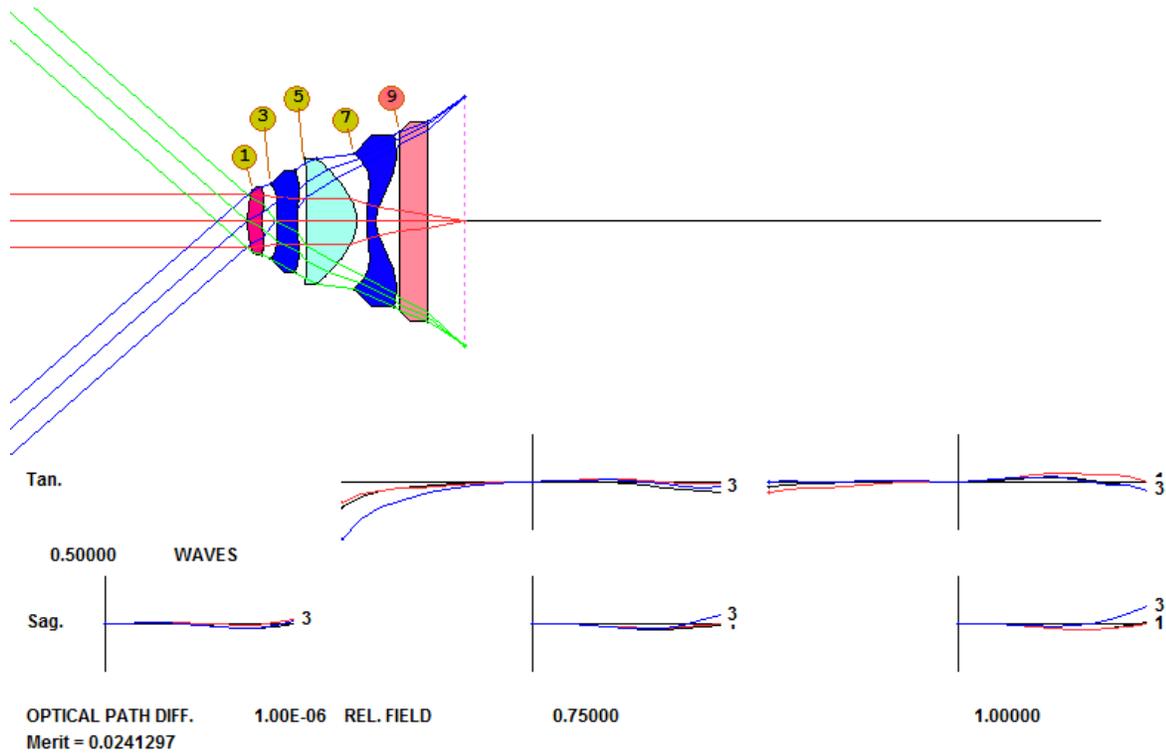
Since we are enabling use of aspherics, we have to be careful to give a grid higher than the default NGRID of 4, and to correct at six field points. Otherwise there will likely be intermediate pupil and field zones that fly away out of control.

The bounds on the glass variables also need some attention. We will replace the model glasses with plastics from the U catalog (Unusual materials) when we get a good design, and we want the model glass to fall in the area where plastics are to be found. This is the purpose of the **PLASTIC** declaration in the input file. Any surface so designated is restricted to the area shown below on the glass map.

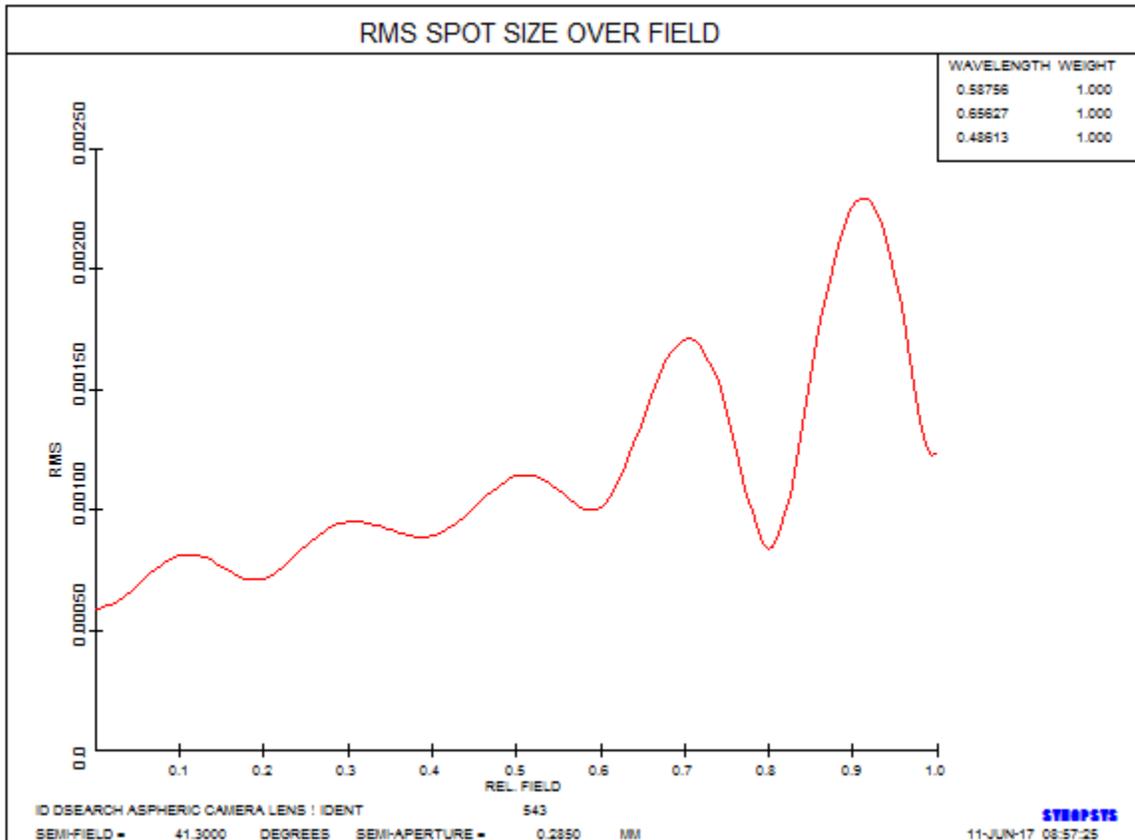


The red dots are the plastics currently in the unusual material catalog (U). The program will keep glass model variables within the area shown. Those glasses that reach the boundary (which is all of them since the area is so small) will slide up and down along those boundaries.

Okay, we run the DSEARCH MACro listed above, and after about eight minutes we see the best design the program found, shown below.



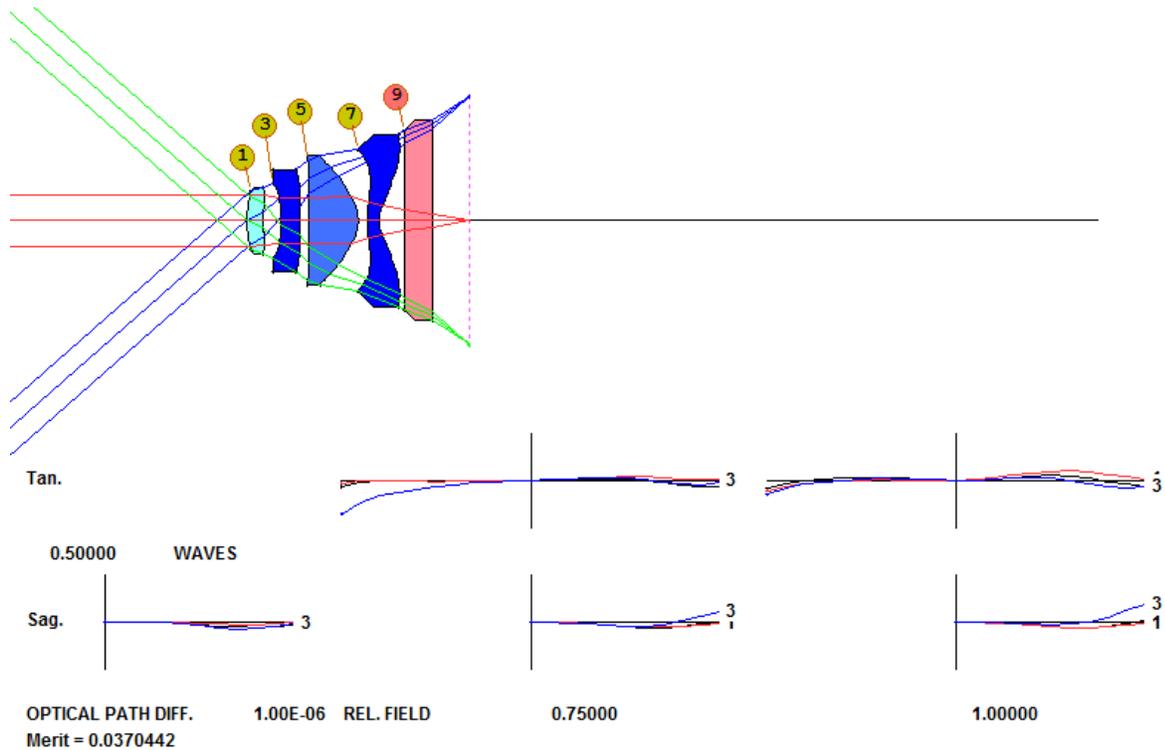
This is amazing. The lens is almost diffraction limited, straight out of DSEARCH. The wavefront errors are less than  $\frac{1}{2}$  wave everywhere. The TFAN at 0.75 field is questionable, however. We have to look at the performance over the field, to see where the correction needs attention. We open the geometrical image menu (MGI) select Over field on the RMS section, Multicolor, and then click the RMS button. The spot size is largest at 0.7 and 0.9 field.



The program has created an optimization MACro, and we add two more lines to the AANT file:

```
GNO      0.000000      0.094362      6 M      0.700000
GNO      0.000000      0.094362      6 M      0.900000
```

Then we run the MACro and anneal. The MF comes out at 0.037, and we have a great design!

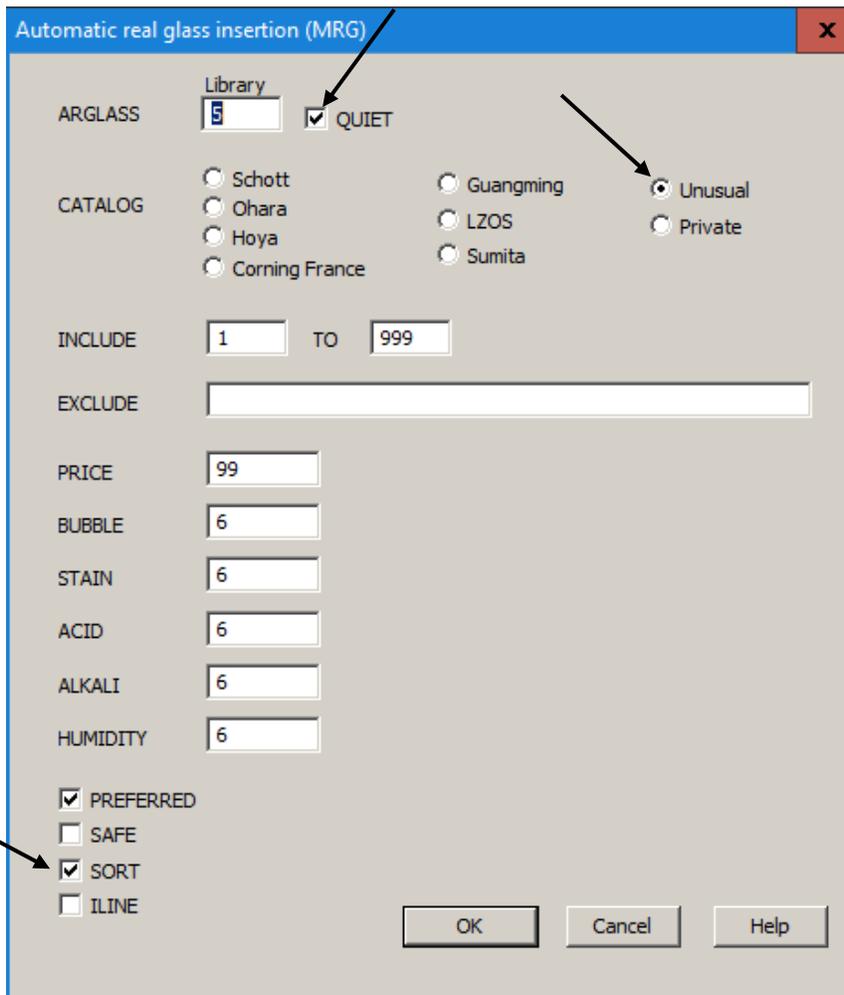


Now it's time to switch to real glasses – but first we change the material on surface 9 to the real glass that the customer wants to use: Hoya type BSC7. To do so, we open the WorkSheet (WS) and type in the edit pane

**9 GTB H  
BSC7**

click Update, and save a checkpoint. The model is replaced. Now we open the real-glass menu (**MRG**) and select the U catalog. That catalog does not have ordinary optical glasses -- but it does have the plastic materials. When you specify the U catalog, the ARGLASS program (which is run from the **MRG** dialog) automatically selects only plastics and only replaces GLMs designated PLASTIC in the RLE file. It has two modes; it can replace the lenses in numerical order, or it can sort them so it replaces the ones furthest from a real material first. The second option is sometimes better, so we select Sort in the MRG dialog, select the Quiet option and then OK.

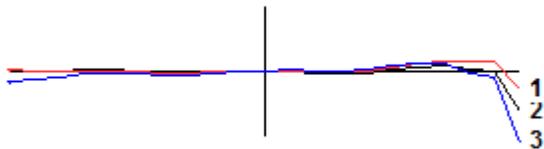
Sometimes changing to a real glass causes ray failures. The program adjusts the curvatures to maintain element power, but if aspheric terms are present, some rays can still fail. If this happens, run ARGLASS again after the other materials are changed. This usually works.



Now there are real materials everywhere. Just to be sure we have an optimum design, we delete the GLM variables in the PANT file (or change them to a single `VLIST GLM ALL`, which only varies GLMs that are already in the lens), and optimize some more. We are almost there. Now it's time to try some more aspheric terms. We add to the PANT file the lines

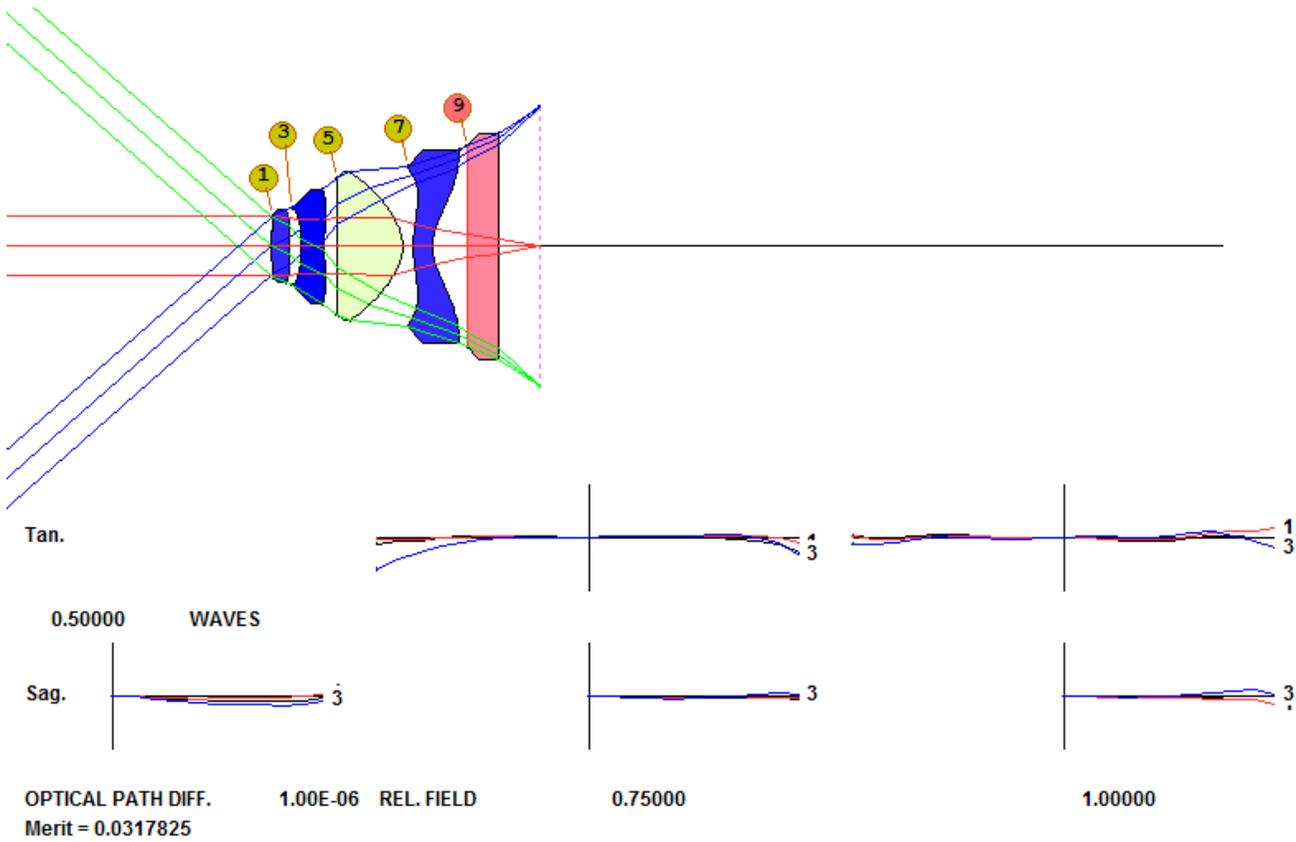
```
VY 1 G 10
VY 2 G 10
VY 3 G 10
VY 4 G 10
```

And optimize again. But now the OPD TFAN at full field is starting to fly away at the upper rim ray.

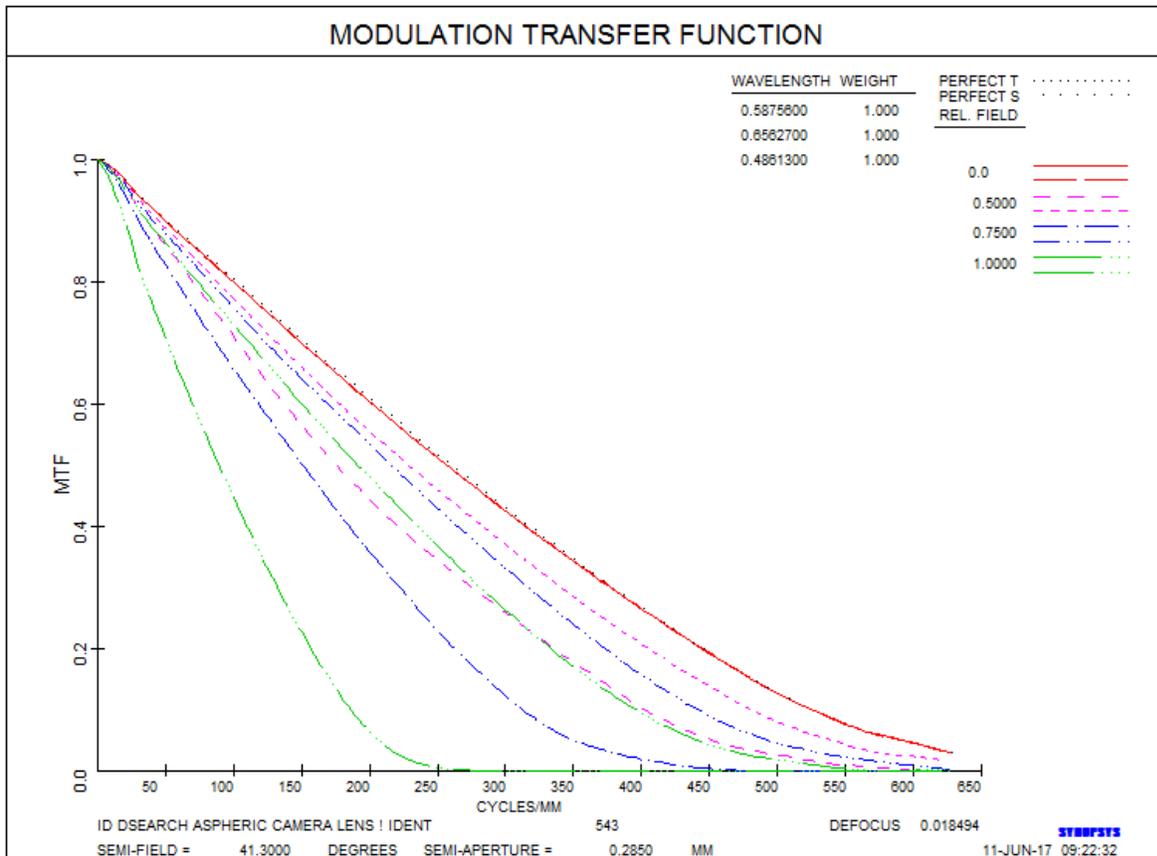


So we add a new line to the AANT file and optimize some more. Much better!

```
M 0 .5 A P OPD 1 0 1
```



Here are the MTF curves for this design. It is close to perfect.



## Encore

Well, that's a start, and now you understand how to use the program, but what could we have done differently? This design is at the diffraction limit, but the MTF at full field is much lower than on axis. Why is that?

Well, since the lens has the stop in front and we are correcting distortion, the image necessarily shows  $\cos^4$  darkening. In fact, at a field angle of 41.3 degrees, that means the edge is just under 32% as bright as the center. How does nature manage to do that? By changing the effective F/number! We type the commands

```
FN 0
FN 1
```

and see that while the on-axis F/number is indeed about 2.7, at the edge it is 6.2 in the tangential direction and 3.5 in the sagittal. The higher F/number increases the size of the Airy diffraction disk, which lowers the cutoff frequency in the Y direction. That is what the MTF curves tell us.

If that situation is satisfactory, we are done. But let's assume you really want uniform illumination over the field. You can't get that unless you let the distortion grow. That may not be a problem if you plan to compensate electronically afterwards. Here's what to do:

1. Delete (or comment out) the lines in the SPECIAL AANT section of the DSEARCH input that gave targets to the chief-ray YA at three field points.

```
SKIP
M 1.35 10 A P YA 1
M .945 10 A P YA .7
M .54 10 A P YA .4
EOS
```

2. Add some new requirements. These will control the relative illumination at five field points.

```
M 1 1 A P ILLUM .2
M 1 1 A P ILLUM .4
M 1 1 A P ILLUM .6
M 1 1 A P ILLUM .8
M 1 1 A P ILLUM 1
```

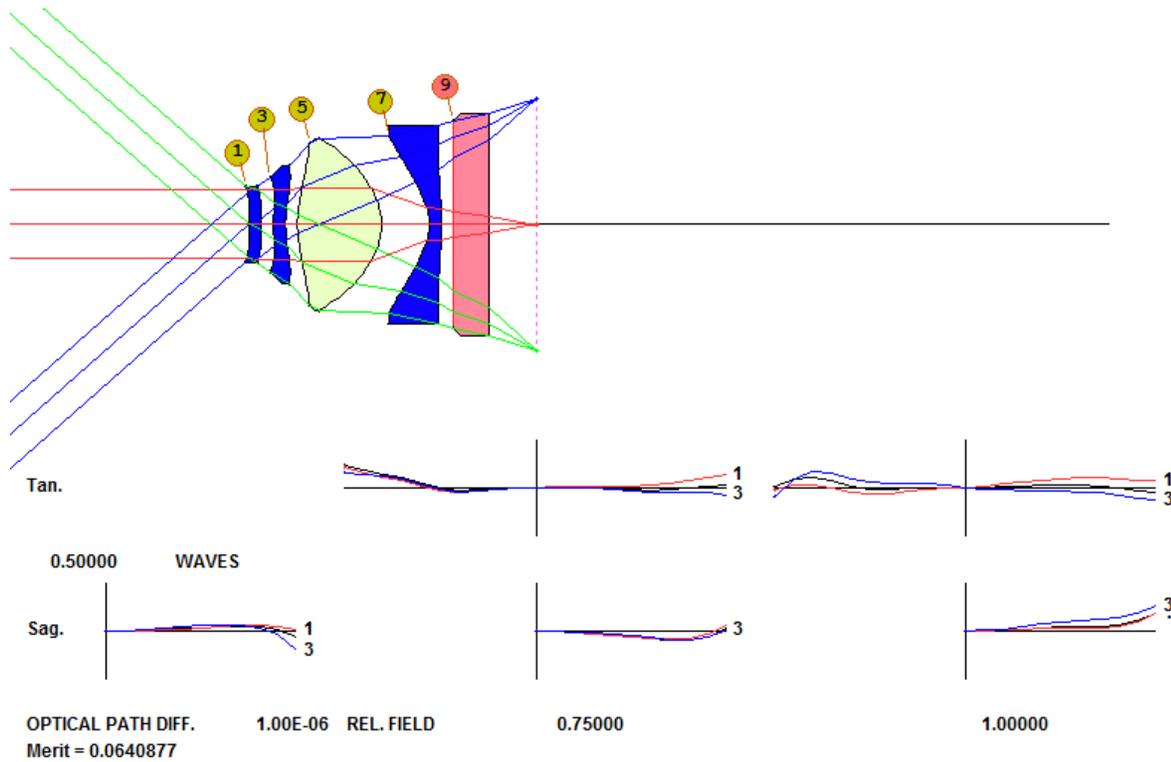
3. Since the F/number at the edge of the field will now be smaller – which is harder to correct, we increase the weights on the outer two fields from 3.0 to 4.0.

Now run this version on DSEARCH, and the lens construction is very different. We optimize some more and notice that the lower rim ray at full field is rapidly flying away, so we add to the MF the line

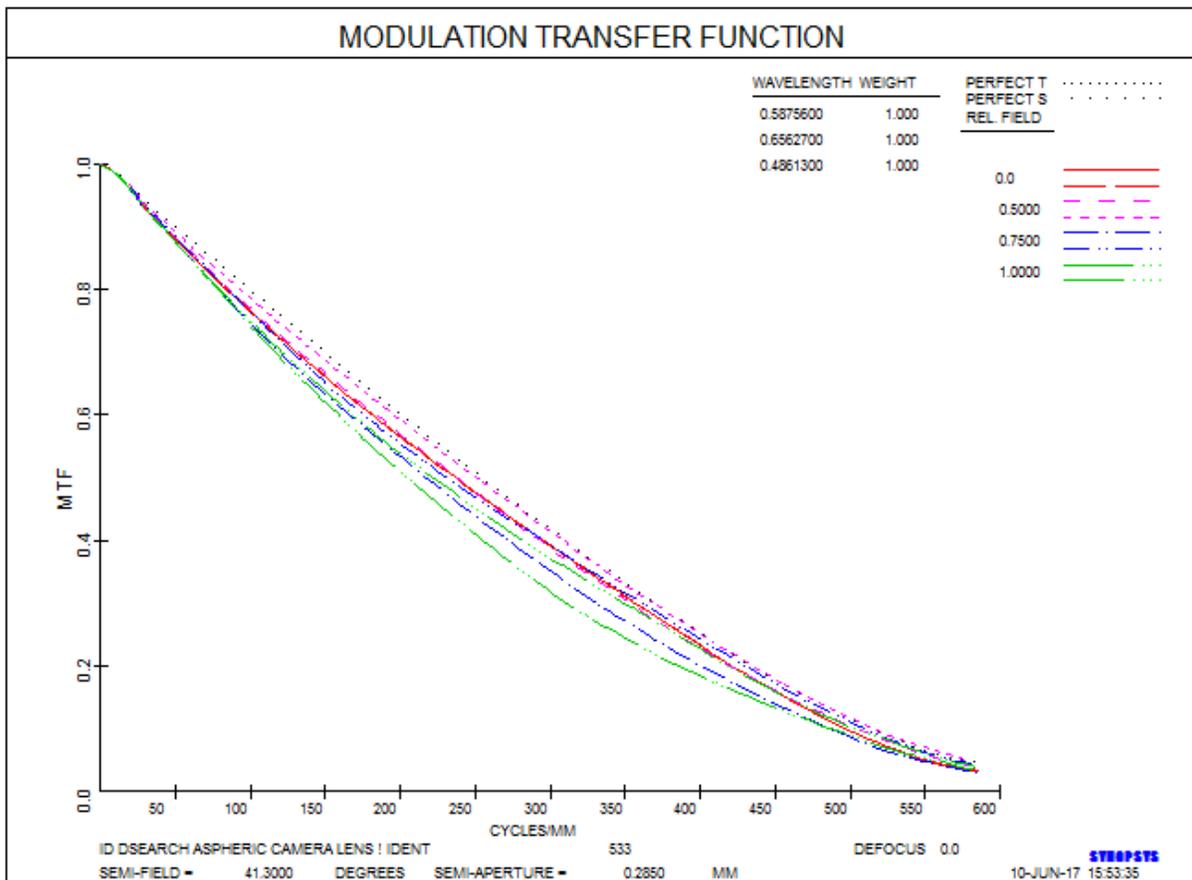
```
M 0 1 A P OPD 1 0 -1
```

and optimize again. The lens is better.

Here is that lens, after optimization and inserting real plastics with the **MRG** dialog:

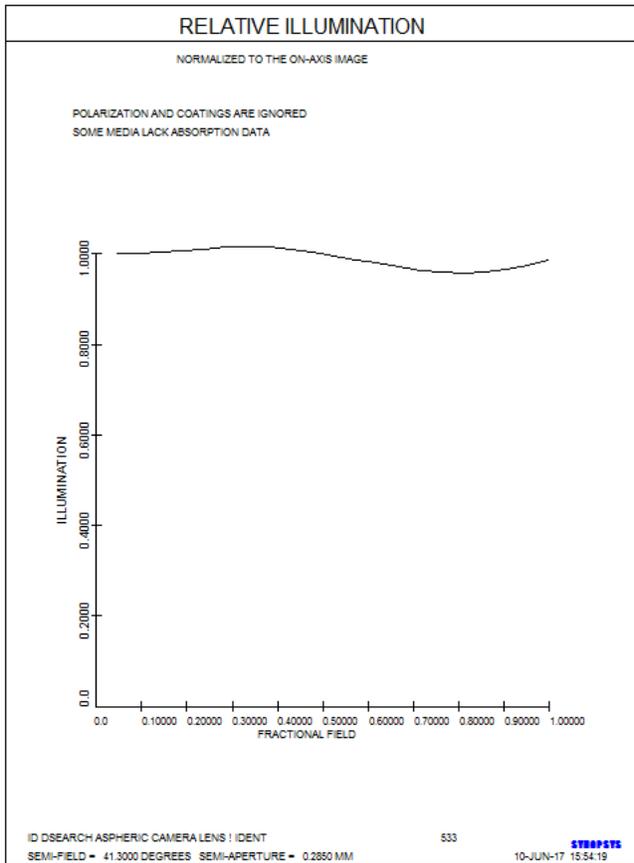


The MTF is quite good, as shown below.



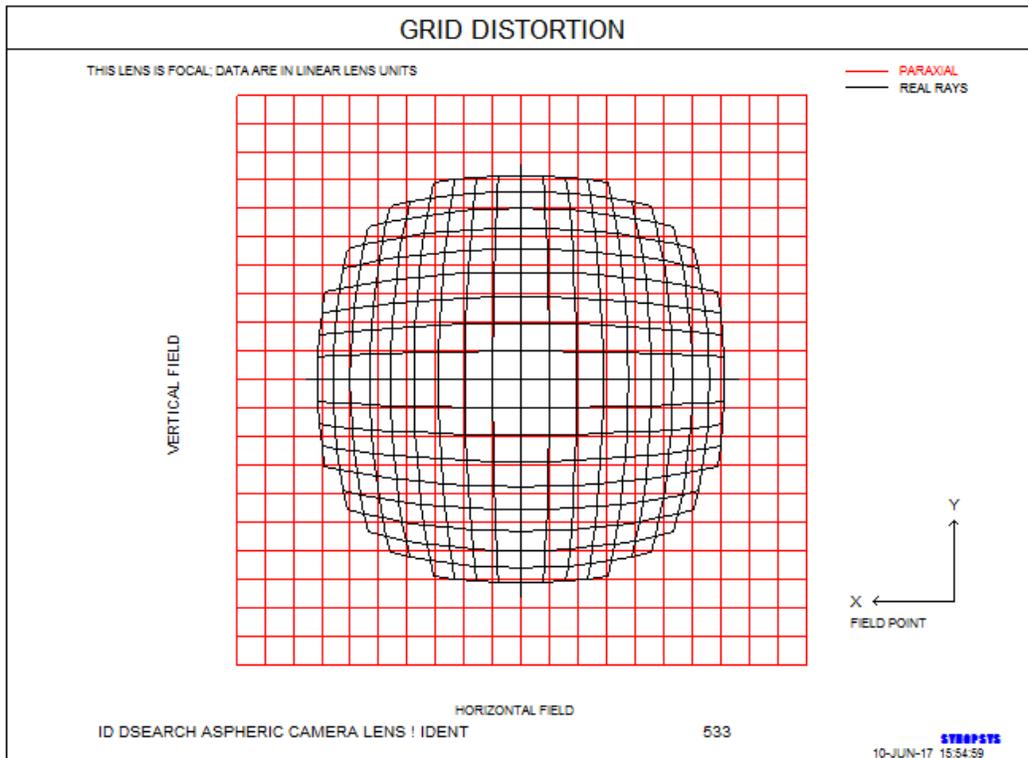
And the illumination is fairly uniform, as plotted with the command

# ILLUM 500 P



The program has indeed introduced significant distortion. Here is the plot produced by the command

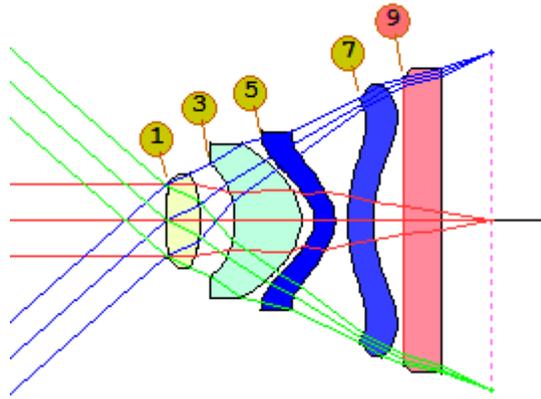
# GDIS 21 G



## Coda

We made it look easy, and it is if you follow the steps above. But of course, lens design has pitfalls all over the place, and things do not always work perfectly the first time. Here are some of the problems that you may encounter, and how to deal with them:

1. We specified an aspheric count of 3 in this example; that assigns terms up to  $R^{**6}$  to the surfaces. What happens if you use fewer or more terms than this? As a rule, it is better to start with a smaller number and then add more after you have optimized the result as well as possible, as we did above. Too many terms right at the start can send the design to a region where the terms are fighting each other and become too large. Also, raytracing can prove a problem with many high-order terms, since the beam can exhibit caustics or large ray angles where you don't want them. We have obtained excellent results by starting with only two terms, and then adding more as we optimized the result.
2. Note that the **FNUM** request in the DSEARCH input file specified a weight of 10; this is more important than meets the eye. If we left the weighting factor off, the program would control the F/number exactly, with a paraxial solve – which can lead to ray failures if the radius that results is too steep. So, for fast lenses such as this one, we usually add a weight. Then the program adds a requirement to the merit function to control the F/number, and the radius is given by the RSTART value. In the second example, where we did not target the image height, the F/number would probably have grown larger than the target value if we had assigned a lower weight. The program will do absolutely anything to reduce the merit function, and giving up a little on that score would likely bring down the other aberrations significantly, resulting in a great image at a higher F/number. So we specified a weight of 10 so that solution would not look so attractive.
3. We chose to give the back focus distance a fixed value in this example. If we instead input a weighting factor on the **BACK** line, the program would assign a YMT solve to the last surface, so the image would always be at the paraxial focus, and then add a target to the AANT file to drive it to the requested value. Both methods work, but when you are targeting the YA of selected rays in order to control the image height, it is best to set the value yourself. Otherwise the program may fail to correct a virtual image since the image height has to change in the process.
4. Remember that DSEARCH uses the annealing feature (if you request it, which is nearly always a good idea), and that feature makes small random changes to the lens, over and over. This greatly improves the optimization of each case, but the results are not repeatable from run to run. For that reason, it is often a good idea to run DSEARCH more than once, and look at some of the other configurations it returns each time. We ran it several times for this lesson, and the results shown above were the best of the lot.
5. These designs met our goals very nicely. But suppose you don't want the cost of a four-element lens. What can you do with three elements? Try it and find out! It will likely not be as good, but then, maybe you don't need that level of resolution with your sensor.
6. Remember that DSEARCH is searching a very bushy design tree and it can't examine every branch every time. If you change almost anything in the DSEARCH input, such as the RT parameter, field weights, monitor targets, etc. – the program will search a different set of branches and return different results. The power of this method is that it can search a huge number of branches simultaneously, and we find that most runs return at least one lens that is at or close to our requirements. By all means experiment with the input and keep the better results in the library so you can examine them at your leisure. Here is another lens, distortion corrected, that was returned during this process. It is much different from our first one but has identical performance:



This lesson uses plastics for all the lens elements except the cover glass. What if you want some elements to be made of glass and the others of plastic? Simple. Just declare which elements are of plastic in the DSEARCH input file, and the program will restrict them to the smaller range where plastics are to be found. Glass elements, on the other hand, will still be free to move over the usual range of the glass map. When the design is satisfactory and you run ARG, the program will match only the plastic elements if the “U” catalog is selected – and will not match them with any other catalog. Simple indeed.