ADVANCED THEORY OF FIELD CURVATURE

by

Yuhao Wang

A Dissertation Submitted to the Faculty of the

COLLEGE OF OPTICAL SCIENCES

In Partial Fulfillment of the Requirements

For the Degree of

DOCTOR OF PHILOSOPHY

In the Graduate College

THE UNIVERSITY OF ARIZONA

2014

THE UNIVERSITY OF ARIZONA
GRADUATE COLLEGE

As members of the Dissertation Committee, we certify that we have read the dissertation prepared by Yuhao Wang, titled Advanced Theory of Field Curvature and recommend that it be accepted as fulfilling the dissertation requirement for the Degree of Doctor of Philosophy.

_____ Date: (Enter Date)
Jose M. Sasian

_____ Date: (Enter Date)
Rongguang Liang

_____ Date: (Enter Date)
Jim Schwiegerling

Final approval and acceptance of this dissertation is contingent upon the candidate's submission of the final copies of the dissertation to the Graduate College.

I hereby certify that I have read this dissertation prepared under my direction and recommend that it be accepted as fulfilling the dissertation requirement.

_____ Date: (Enter Date)
Dissertation Director: Jose M. Sasian

STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of the requirements for an advanced degree at the University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this dissertation are allowable without special permission, provided that an accurate acknowledgement of the source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: Yuhao Wang

ACKNOWLEDGEMENTS

I am deeply grateful to my advisor, Dr. José Sasián, for supporting me in the study of my dissertation work. I appreciate his patient guidance which kept me on track throughout my research studies.

I am also deeply grateful to the other members of my dissertation committee, Dr. Rongguang Liang and Dr. Jim Schwiegerling, for reviewing my dissertation.

Additional thanks to Dmitry Reshidko and Chia-Ling Li for their help on my research study.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Classical field curvature theory emphasizes the Petzval theorem, which models field curvature aberration to the 4^{th} order. However, modern lens designs use aspheric surfaces. These surfaces strongly induce higher order field curvature aberration which is not accounted for Petzval field curvature. This dissertation focuses on developing higher order field curvature theories that are applied to highly aspheric designs. Three new theories to control field curvature aberration are discussed. Theory 1: an aspheric surface that is close to the image and has two aspheric terms sharply reduces field curvature by 85%. Theory 2: an aspheric surface that is farther from the image plane induces astigmatism to balance Petzval field curvature. Theory 3: oblique spherical aberration can be induced to balance Petzval field curvature. All three theories are applied to real design examples including the following lenses: cellular phone, wide angle, fast photographic, and zoom lenses. All of the analyses results are consistent with the theories. Moreover, two types of novel aspheric surfaces are proposed to control field curvature. Neither of the surfaces are polynomial-type surfaces. Examples show that the novel aspheric surfaces are equivalent to even aspheric surfaces with two aspheric coefficients in terms of field curvature correction. The study on field curvature correction using aspheric surfaces provides an alternative method to use when aspheres are accessible. Overall, this dissertation advances the theory of field curvature aberration, and it is particularly valuable to evaluate highly aspheric designs when Petzval theory is inapplicable.

CHAPTER 1

INTRODUCTION TO FIELD CURVATURE

1.1 Field Curvature

Field curvature is also known as “Petzval Curvature” which is the inherent problem of an optical system imaging a flat object onto a curved surface. When a flat detector is used to receive the image, only the center of the image is sharp but blurred more and more when looking at the image away from the center. Field curvature effect is more obvious in wide angle lenses.

1.2 Effect of Field Curvature on Image Quality

An image in Figure 1-1 was taken with a F/5 objective lens. The left image is the original; the right image is the original with 2 waves of field curvature. The right image only looks sharp in the center areas; the image comes out of focus in the off-centered areas. The more off-centered the image, the more blurry it looks.

Figure 1-1 Original image compared to the image with field curvature

Since field curvature is an inherent property of lenses, it is not limited to photographic cameras; it can be seen in microscopes, telescopes and many other optical instruments as well.

1.3 Classical Theory of Field Curvature

Classical thoery of field curvature is governed by "Petzval Theorem" which was found by Joseph Petzval. Petzval theorem states that the field curvature of an optical system is nothing but the sum of the lens power weighted by material index of refraction. The "Petzval Sum" is given below.

$$Petzval\ Sum = \frac{1}{R_p} = \sum \frac{n'-n}{nn'r} = \sum \frac{\emptyset_i}{n_i} \qquad (1\text{-}1)$$

R_p represents the Petzval radius, n the index on the object side of the surface, n' the index on the image side of the surface, and r the radius of the surface. Φ represents the surface power, and i the surface number. Basically, the Petzval Theorem claims that the perfect image falls onto a curved image plane (Petzval surface) with the radius of curvature R_p calculated from the Petzval Sum.

A single positive lens's field curvature is plotted in Figure 1-2. In this case, the Petzval surface is inward bending with a Petzval radius R_p equal to the lens focal length f multiplied by the glass index of refraction n. This is a useful rule of thumb to quickly estimate field curvature of a thin lens without complex computation.

$$R_p = -f * n \qquad (1\text{-}2)$$

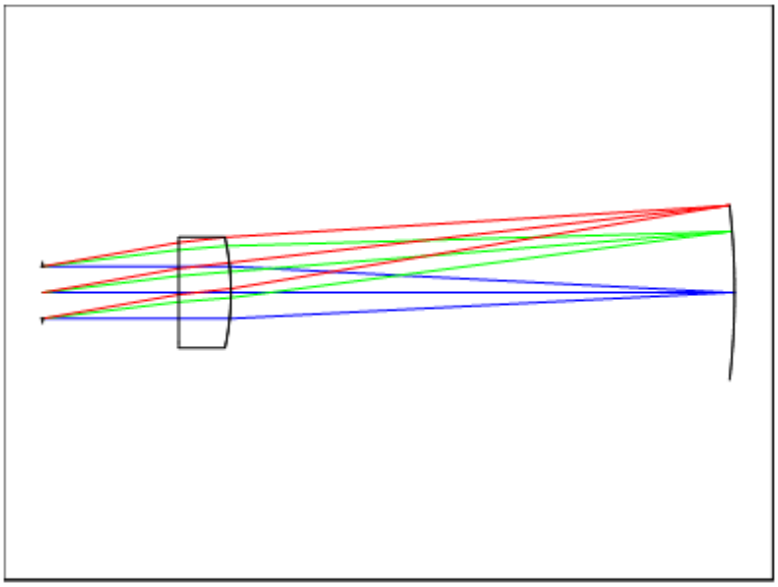


Figure 1-2 Schematic of field curvature of a positive single lens

The wave coefficient W_{220} describes field curvature. H is the field dependency and ρ is the pupil dependency. x_p and y_p are normalized pupil coordinates. In this way, field curvature can be treated as quadratic field dependent defocus.

$$W(H,\rho) = W_{220}H^2\rho^2 = W_{220}H^2({x_P}^2 + {y_P}^2) \qquad (1\text{-}3)$$

Here, the simple example below illustrates pure Petzval field curvature (W_{220} only). W_{220} = 2 waves, FOV (field of view) = ±15°, f/# = 5.

As Petzval predicted, the image falls onto a spherical surface (Petzval surface). The surface sag can be computed as follow:

$$\delta_z = 8(f/\#)^2 W_{220}H^2 \qquad (1\text{-}4)$$

Field curvature plot is in Figure 1-3. A perfect quadratic shape appears when only W_{220} exists.

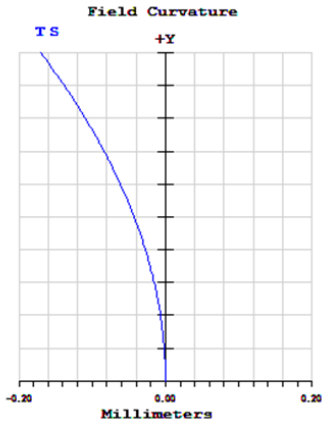


Figure 1-3 Field curvature plot of pure Petzval curvature

The wavefront error is shown in Figure 1-4. There is no aberration along the on-axis plot; the off-axis plot displays a quadratic shape represents field curvature. The plot is very similar to defocus; but the magnitude changes with the field.

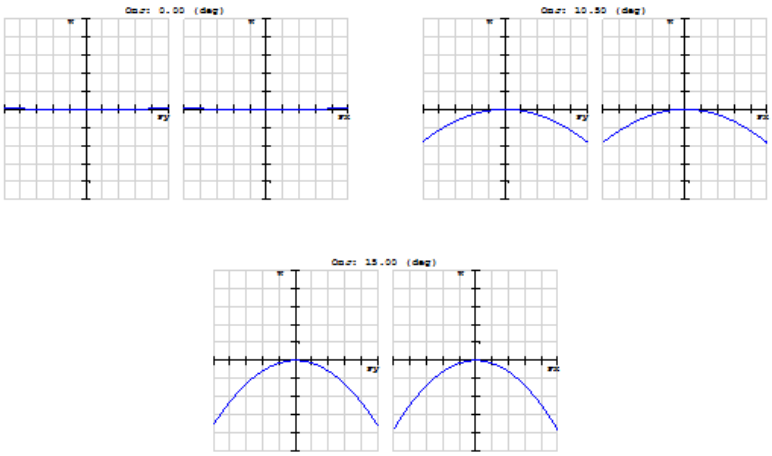


Figure 1-4 OPD (optical path difference) of field curvature

The transverse ray aberration plot is shown in Figure 1-5. There is no aberration on-axis; the off-axis plot displays a linear error as transverse ray aberration which is the derivative of wave aberration. ε_Y and ε_X represents transverse ray errors. R is the radius of the wavefront at the exit pupil; r_P is the exit pupil radius. $R/r_P \approx 2f/\#_w$

$$\varepsilon_Y = -2\frac{R}{r_P}W_{220}H^2y_P \qquad (1\text{-}5)$$

$$\varepsilon_X = -2\frac{R}{r_P}W_{220}H^2x_P \qquad (1\text{-}6)$$

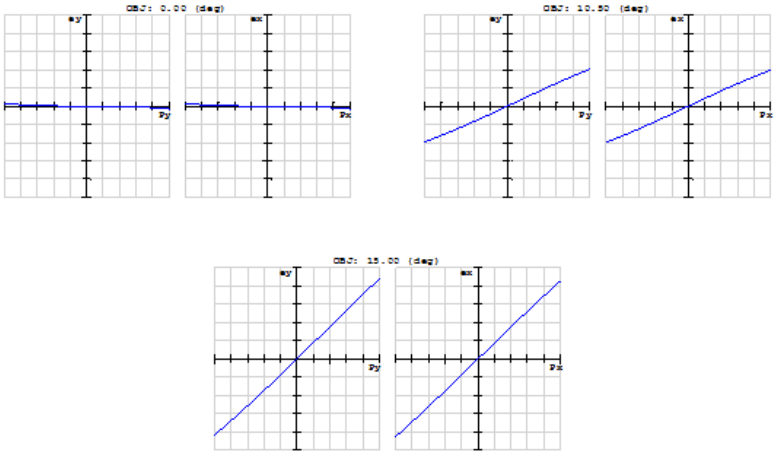


Figure 1-5 Transverse ray aberration plot of field curvature

The spot diagram is shown in Figure 1-6. There is a point focus along the on-axis plot and a blurred spot along the off-axis plot. The larger the field, the larger the spot.

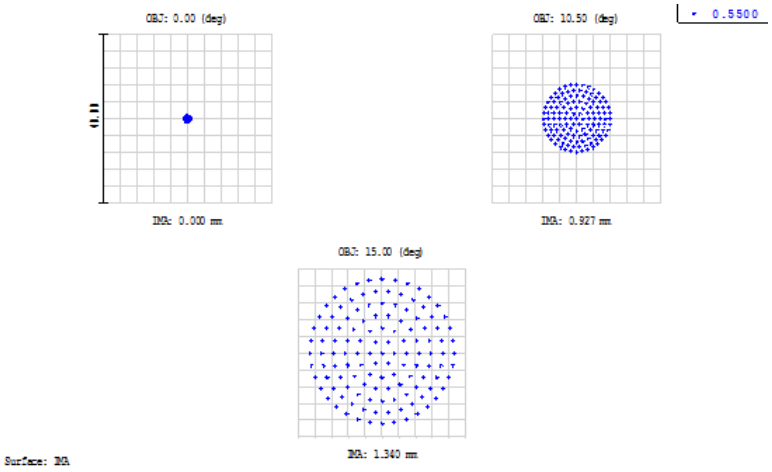


Figure 1-6 Spot diagram of field curvature

The Modulation Transfer Function (MTF) is shown in Figure 1-7. Since there is no aberration along the on-axis plot, the on-axis MTF curve (blue) is coincident with the diffraction limited MTF curve (black). The off-axis curve dramatically falls off when spatial frequency increases because of field curvature. Phase reversal occurs in this case.

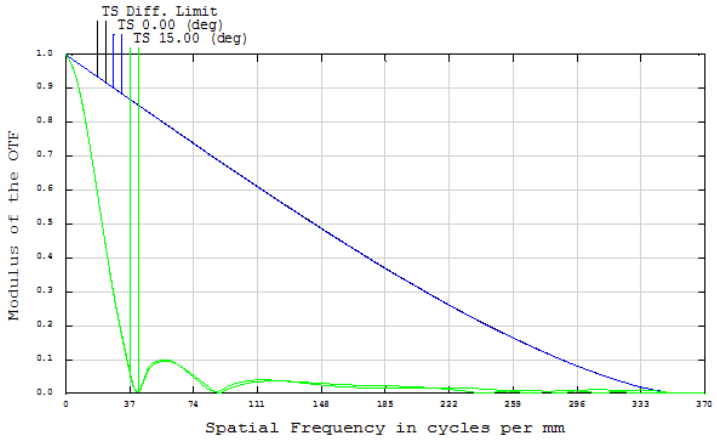


Figure1-7 MTF (modulation transfer function) of field curvature

The MTF curve through field of view is displayed in Figure 1-8. The MTF curve falls off when the field angle increases. The larger the field, the lower the contrast.

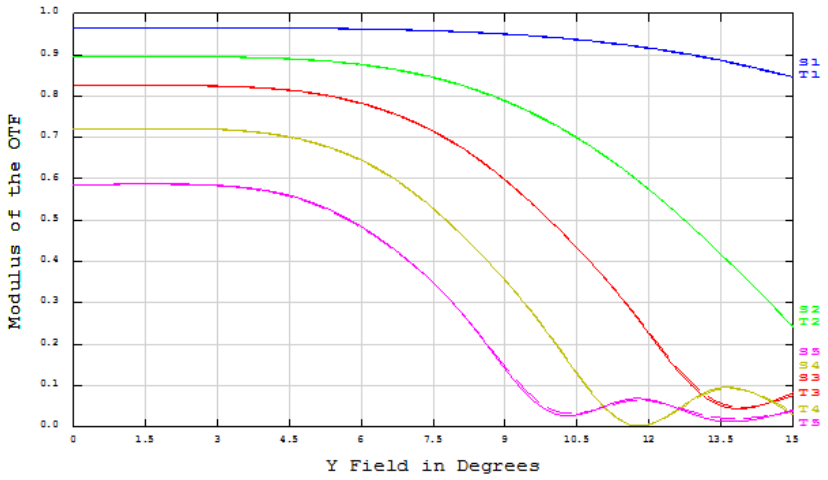


Figure 1-8 MTF through field of view plot of field curvature

After reviewing the basic properties of field curvature, it can be seen that field curvature has a significant effect on image quality. The classical field curvature correction methods in lens design are addressed in the next chapter.

CHAPTER 2

CLASSICAL METHODS TO CONTROL FIELD CURAVATURE

As discussed in the previous chapter, field curvature significantly degrades image quality so that it is very critical to control it in lens design. Based on the classical Petzval theorem, field curvature depends only on the lens power distribution and material index. In other words, field curvature should be considered at the first-order design because after that field curvature will not change during the lens optimization process. Therefore, field curvature is often considered to be one of the most difficult aberrations to correct in classical lens design. In this chapter, the classical methods to control field curvature will be illustrated.

2.1 Meniscus Lens

In order to minimize the Petzval sum, it is ideal to have a lens with two surfaces of the same radius of curvature in both magnitude and sign so that the net contribution to the Petzval sum is zero. The total meniscus lens power is proportional to its thickness. A thick meniscus lens contributes to optical power but not field curvature. An example is illustrated below to demonstrate how a meniscus lens helps control field curvature.

Double Gauss lens: F/3, focal length = 100mm, FOV= ±14°

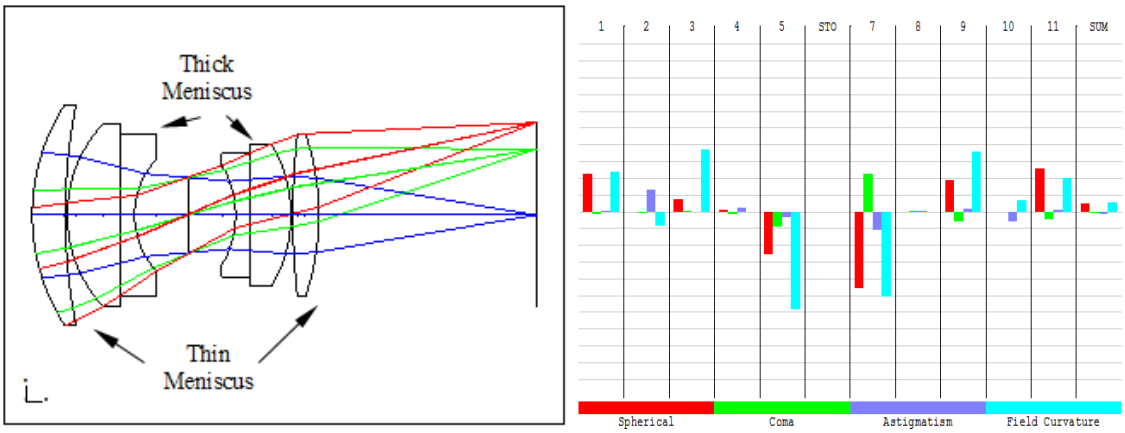


Figure 2-1 Double Gauss lens layout and Seidel aberration diagram

The two cemented thick meniscus lenses are symmetrical to the stop and closer to the stop. The two thick meniscus elements (surface 3, 5 and 7, 9) contribute to the total amount of negative field curvature. The cemented surfaces (4, 8) have very little effect on field curvature. The two outer thin meniscus elements (surface 1, 2 and 10, 11) contribute to positive field curvature. Therefore, the negative and positive field curvature will cancel each other out. The whole Double Gauss lens has little positive field curvature residue. The field curvature plot is shown in Figure 2-2.

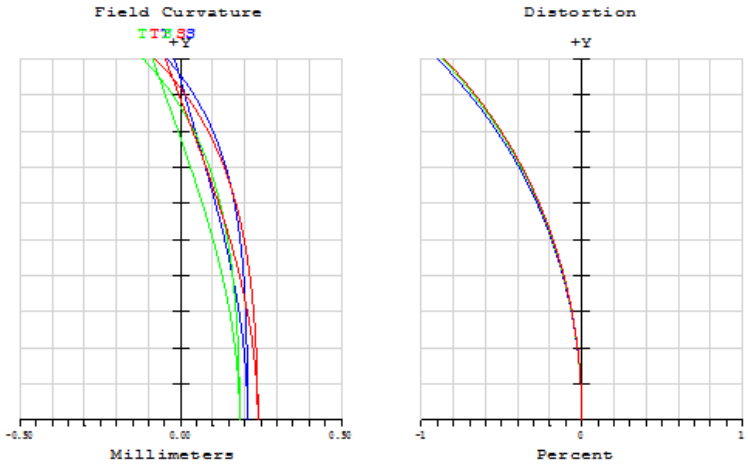


Figure 2-2 Field curvature and distortion plot

A meniscus lens is a powerful tool to control field curvature in classical photographic objective lenses. Due to the fabrication limitation (such as the center too thick for a thick meniscus lens and the edge too thin for a thin meniscus), it is difficult to achieve an ideal meniscus with two of same curvature surface that cancel out field curvature. The art of controlling field curvature is balancing the thin and thick meniscus lenses' curvature and thickness.

2.2 Field Flattener Lens.

Most optical imaging systems require positive power to focus light. The most important aberration is spherical aberration. For a single lens with an index of refraction 1.5, the optimal shape to minimize spherical aberration is bi-convex. However, a bi-convex shape represents the worst case for field curvature. Therefore, if a design consists of groups of positive bi-convex lenses, it will have excessive field curvature aberration. One way to control field curvature, this these cases, is locating a negative field flattener lens near the image plane. The negative field flattener lens will contribute to a strong negative field curvature that balances the excessive positive field curvature. The field flattener lens induces very little optical power, spherical aberration, coma, and astigmatism. An example is illustrated below to demonstrate how a field flattener lens corrects field curvature.

Petzval Lens: F/2, f = 100mm, FOV = ±4°

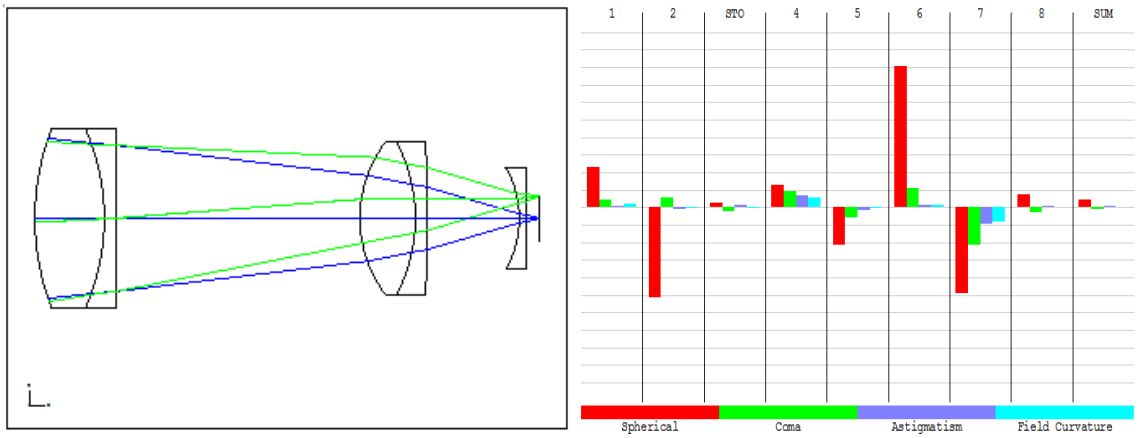


Figure 2-3 Petzval lens layout and Seidel aberration diagram

For this Petzval lens, two cemented doublet lenses correct spherical aberration and coma effectively but leave some positive field curvature. The negative field flattener lens (surface 7, 8) contributes negative field curvature that balances the positive field curvature. Since the amount of spherical aberration, coma, and astigmatism is a function of marginal ray height, there is very little contribution when the field flattener lens is close to the image plane. As the design is very fast (F/2), there is some excessive spherical aberration. The field curvature plot is shown in Figure 2-4.

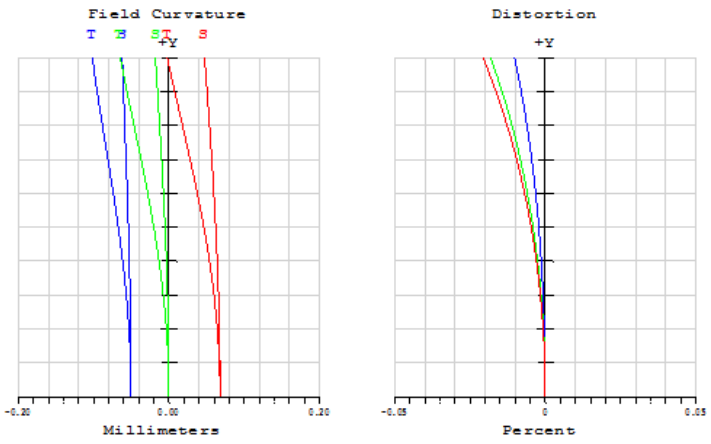


Figure 2-4 Field curvature and distortion plot

A field flattener lens is very useful for designs with excessive positive field curvature and no other degree of freedom to control field curvature. But it is not always feasible to locate a lens near the image plane due to a mechanical mounting and detector cover glass confliction concerns.

2.3 Separated Thin Lenses

For an optical system which contains several separated thin lenses, the Petzval sum is given by $\sum \frac{\emptyset_i}{n_i}$. When the positive and negative thin lenses are alternately used, the Petzval sum could be minimized and field curvature could be well controlled. An example is illustrated below to demonstrate how field curvature is controlled by using separated thin lenses.

Cooke Triplet Lens: F/5, f = 50mm, FOV = ±20°

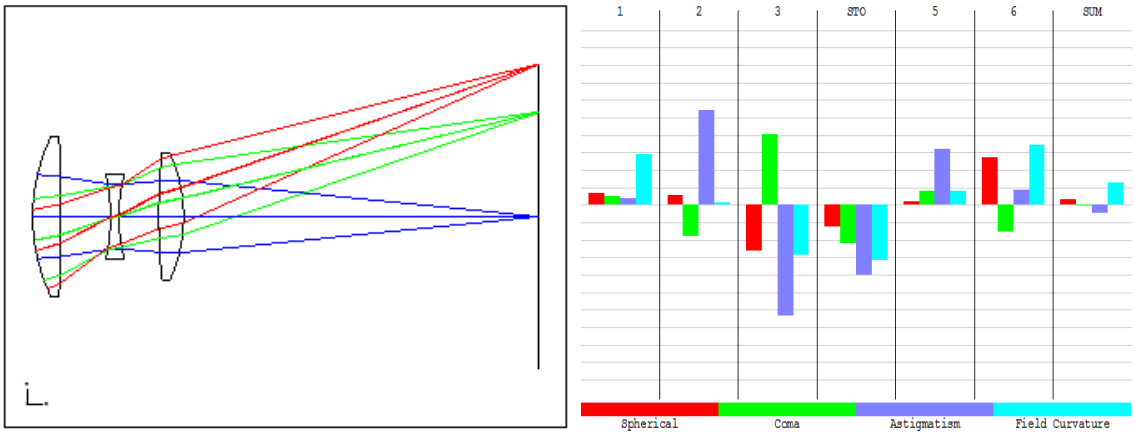


Figure 2-5 Cooke triplet lens layout and Seidel aberration diagram

The Cooke triplet lens consists of three thin lenses. The negative lens in the middle is the system stop. The two positive lenses are symmetric to the stop. The positive power contributed by the 1st and 6th surfaces is balanced with the negative power contributed by the 3rd and 4th (stop) surfaces. The 2nd and 5th surfaces contribute very little power since they are almost flat. The whole system has a little positive field curvature which depends on the focal length of the design. The field curvature plot is shown in Figure 2-6.

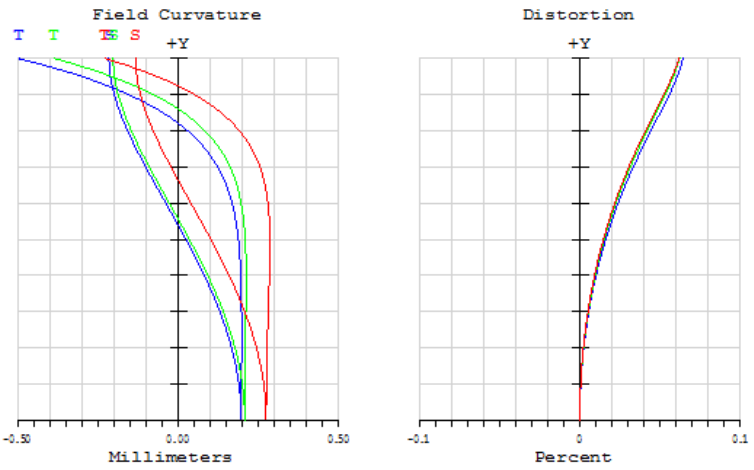


Figure 2-6 Field curvature and distortion plot

Separating thin lenses is an effective method to correct field curvature. It is particularly useful in the designs that have enough thin lenses with alternating positive and negative powers, such as a micro-lithographic lens design with a profile of bulges and constrictions. However, it is always a tradeoff between system focal length and the number of thin lenses in terms of field curvature correction. The shorter the focal length, the more elements needed to balance field curvature and maintain strong optical power at the same time.

2.4 New Glass Type

In order to minimize the Petzval sum, minimizing the net optical power is a effective tool. However, the choice of glass also plays a role. It is ideal if the glass used has a high index of refraction. The old high index glass usually has high dispersion. But the new type of glass makes it possible to have a high index and a low dispersion. The old and new achromat doublet designs are illustrated and analyzed below.

The old achromat doublet lens consists of a crown glass with low index and low dispersion followed by a flint glass with high index and high dispersion. The old achromat has been used in telescopes and many other system designs. By optimizing the three surfaces' curvature, spherical aberration and longitudinal color aberration could be controlled, but the Petzval sum (2-1) is always positive since $\Phi_1>\Phi_2$, $n_1<n_2$. Therefore, the old achromat is not effective for correcting field curvature.

$$Petzval\ Sum = \frac{\emptyset_1}{n_1} + \frac{\emptyset_2}{n_2} \qquad (2\text{-}1)$$

The new achromat design takes advantage of a new type of crown glass with high index and low dispersion followed by the new flint glass with low index and high dispersion. In the new achromat configuration ($\Phi_1>\Phi_2$, $n_1>n_2$), it is possible to cancel out the Petzval sum. Many classical photographic objective lenses such as the Protar lens have adopted the new achromat such.

Material selection is critical in lens design in term of color correction. But selecting glass type can help reduce field curvature.

2.5 Miscellaneous Methods

2.5.1 Gradient refraction index (GRIN) glass[1-3]

The normal glass used in lens design has a homogeneous index of refraction. Gradient index glass has a variation of index of refraction. The variation could be axial, radial, or spherical. The advantage of GRIN material is in controlling aberration. The most common example of a GRIN optical system is the human eye.

Since field curvature has a quadratic pupil dependency, the field curvature may be compensated by using a glass material with a radial gradient index. For a positive thin lens, when pupil size increases field curvature increases so that the beam focuses near. So, in the case of a glass with radially decreased index from lens center to edge, the ray path length could compensate so that the image falls on a flat image surface rather than a curved surface. The radial GRIN glass's index of refraction is a function of the pupil which could be written as,

$$n(r) = N_{00} + N_{10}r^2 + N_{20}r^4 + \cdots \qquad (2\text{-}2)$$

N_{00} represents the index on the optical axis. N_{10} and N_{20} are the corresponding order of radial gradient coefficients.

Equation (2-3) gives the transverse ray aberration of 3rd order field curvature coefficients for a single lens.

$$\sigma_4 = \frac{H^2}{n_k u'_{aK}} \left(\frac{-N_{10}t}{N_{00}^2}\right) \quad (2\text{-}3)$$

n_k and u'_{aK} are the refractive index and the paraxial marginal ray angle after refraction at the image plane; *H* is the Lagrange invariant. *t* is the lens thickness.

Integrating equation (2-3) gives the wave aberration coefficients W_{220P},

$$W_{220P} = \frac{DH^2}{2n_k N_{00}^2 \rho^2} * N_{10} t \qquad (2\text{-}4)$$

D represents the aperture size, ρ the pupil coordinate. Equation indicates that field curvature is a linear function of lens thickness *t* and radial gradient N_{10}.

GRIN materials have been used in some optical systems. The challenging part of using GRIN materials involves, for example, the issues of color aberrations, thermal variations of the gradient, and the need for fine alignment to the optical axis.

2.5.2 Curved Film and Curved Detector

Instead of correcting field curvature, a curved film or detector with the radius matching the Petzval radius can solve the field curvature problem. The most common optical system with a curved detector plane is the human eye. Field curvature is not a large problem for the human eye because the retina could auto-adjust its curvature biologically.

The Schmidt camera used a curve film or detector to compensate for field curvature. Sometimes the film is made curved; in other cases the flat film could be mechanically conformed to the shape matching the Petzval curvature [4]. The film can be sprung to quite a sharp curve by means of a ring which presses the film against a convex base. Thin film, such as the ordinary roll film, is not suitable, as it will wrinkle at the edges, but the heavier cut film can be sprung to the required curve, and when removed from the holder

will spring flat again[5]. If a flat film is used, a plano-convex field flattener lens is usually needed to correct field curvature.

Optical fibers could be used to make a curved detector [6]. The incident fiber surface consists of a rod of fiber which could be polished with a curvature. Each fiber acts like a "pixel" which samples the image and transmits it to a flat detector. The curved fiber guide is coupled to CCD as an input window.

Another example using a curved detector is on space telescope detector arrays[7]. The image-sensor array of the Kepler space observatory is curved to compensate for the telescope's Petzval curvature. Several pieces of CCDs have been assembled on a curved substrate. The individual flat CCDs are mounted on a curved substrate and fitted with individual field flattener optics.

There are some issues with these kinds of curved detectors. First, it is difficult to make the curvature too large. Second, the resolution could not be made too high. Third, the curved detector is not interchangeable. Finally, the cost is still too high compared to normal detectors. Therefore, it is not widely used in many optical imaging systems.

2.5.3 Powerless field flattener

Sasián[8-9] has invented a powerless field flattener which is a single element with a stepped profile across the aperture. The step is designed to have a curvature approximately opposite to the inherent field curvature. The superiority of this invention is that it does not significantly introduce extra optical power and other aberrations. The performance and fabrication ease have been studied but no mass produced utilizing this type of flattener so far.

CHAPTER 3

WAVE ABERRATION THEORY OF HIGH ORDER FIELD CURVATURE

In previous chapters, the classical field curvature theory up to the 4^{th} wave aberration order has been discussed. Petzval theorem can fully explain the behavior of field curvature in most classical designs and works well for designs using spherical surfaces. However, when the modern aspheric surfaces are widely used, field curvature cannot be explained by the Petzval theorem because of the higher field curvature aberration induced by aspheres. Indeed, higher order field curvature aberration can balance the 4^{th} order field curvature aberration. In this chapter, a discussion of how aspheric surfaces help control field curvature will be elaborated.

3.1 High Order Aberrations

For most classical optical systems, 4^{th} order wave aberration theory is enough to explain the aberration correction. But when the design becomes faster and the FOV becomes larger, high order aberrations occur and affect aberration correction. Moreover, when using aspheric surfaces in optical design, the high order aberration effect is more obvious. Therefore, studying high order aberrations is very critical to further understand aberration corrections in modern lens design. The most recognized 6^{th} order aberration coefficients are listed in the Table 3-1[10]. Column *A* shows the 6^{th} order aberration coefficients with an increasing field dependency of H^2 compared to the corresponding 4^{th} order

coefficients. Column *B* shows the special 6th order aberration coefficients with an increasing pupil dependency of ρ^2 compared to the corresponding 4th order coefficients. Even high order aberrations (8th, 10th, 12th order…) also exist in highly aspheric and fast designs.

Column *A*		Column *B*	
Oblique spherical	W_{240}	6th order spherical	W_{060}
6th order coma	W_{331}	unnamed	W_{151}
6th order astigmatism	W_{422}	unnamed	W_{242}
6th order field curvature	W_{420}	unnamed	W_{333}
6th order distortion	W_{511}		

Table 3-1 List of sixth order wave aberration coefficients

3.2 Aspheric Contribution to High Order Field Curvature

First, the most widely used even asphere surface sag equation is reviewed below.

$$z = \frac{Cr^2}{1+(1-(1+k)C^2r^2)^{1/2}} + A_4r^4 + A_6r^6 + \cdots \quad (3\text{-}1)$$

C represents the base sphere curvature, *r* the pupil radius, k the conic constant, A_4 the 4th order aspheric term, and A_6 the 6th order aspheric term.

The even asphere equation contains two parts. The first part represents the base sphere with a conic constant. The second part represents the high order aspheric part starts from A_4r^4. The surface base curvature *C* and conic k contribute to the surface power. But the high order aspheric terms A_4 and A_6 do not affect surface power at all. However, high order aspheric terms A_4 and A_6 do contribute to high order field curvature, namely

W_{420} and W_{620}. W_{420} represents the 6th order field curvature and W_{620} the 8th order field curvature. Based on Sasian's high order aberration theory [11], the aspheric terms contribution to high order field curvature could be derived and is shown below in equation (3-2;3-3).

$$W_{420as} = \frac{Ж^2}{2nn'} A_4 \Delta(n) \bar{y}^2 \qquad (3\text{-}2)$$

$$W_{620as} = \frac{Ж^2}{8nn'} A_6 \Delta(n) \bar{y}^4 \qquad (3\text{-}3)$$

Ж represents the Lagrange invariant, $\bar{y}$ the chief ray height, and $\Delta(n)$ the change of refraction index. With W_{420} and W_{620} calculated, the overall field curvature aberration up to 8th order can be computed.

3.3 Aberration Balancing between 4th and Higher Order Field Curvature Aberration

If an optical system has field curvature up to 8th order, the wave aberration function can be written as,

$$W = W_{220} H^2 \rho^2 + W_{420} H^4 \rho^2 + W_{620} H^6 \rho^2 \qquad (3\text{-}4)$$

In order to compute the sag of field curvature, defocus W_{020} is added to the wave expansion to make W = 0. Since the pupil dependent ρ^2 canceled out, the defocus can be written as,

$$W_{020} = -W_{220} H^2 - W_{420} H^4 - W_{620} H^6 \qquad (3\text{-}5)$$

Then bringing W_{020} into the sag equation:

$$\delta_z = 8(f/\#)^2 W_{020} = -8(f/\#)^2 (W_{220}H^2 + W_{420}H^4 + W_{620}H^6) \qquad (3\text{-}6)$$

With the sag equation (3-6) calculated, field curvature can be computed when the three coefficients are given. Theoretically, W_{420} and W_{620} can balance W_{220} to minimize the sag of the field curvature. The corresponding aspheric coefficients can be calculated by equation (3-2) and (3-3).

An example is given below to demonstrate how the field is flattened using three field curvature coefficients. Assuming the system works at *f/10* with 4th order field curvature $W_{220} = -1\lambda$ ($\lambda = 550$ nm). Figure 3-1 plots the Petzval field curvature.

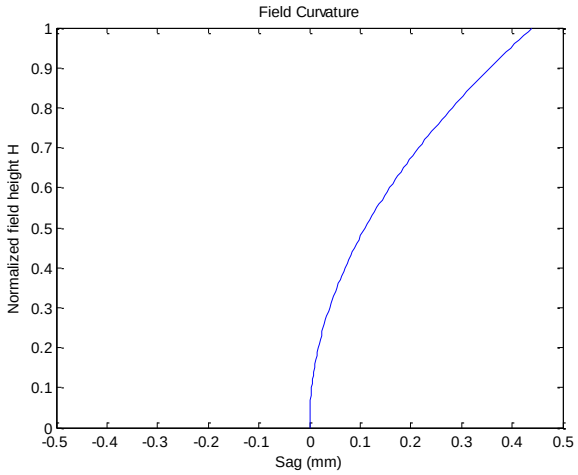


Figure 3-1 Field curvature with $W_{220} = -1\lambda$. (Sag = 0.44 mm P-V)

When W_{420} is added, the field curve can be flattened substantially. The optimization algorithm used minimizes peak to valley (P-V) error. Figure 3-2 shows the optimized field curve.

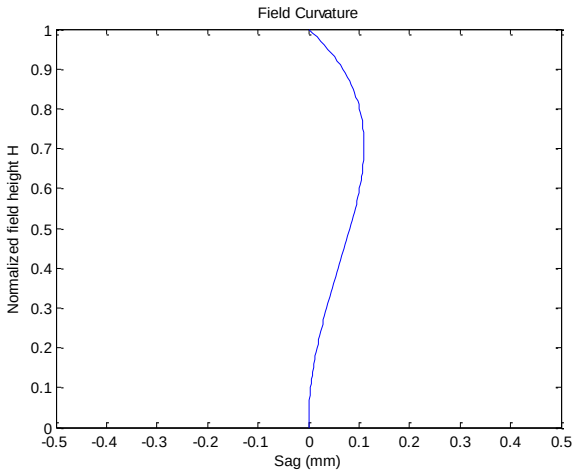


Figure 3-2 Field curvature with W_{220} = -1λ, W_{420} = 1λ. (Sag = 0.11 mm P-V; sag reduction in percentage 75.0%)

Then, when both W_{420} and W_{620} are added, field curvature can be further flattened. Figure 3-3 shows the further optimized field curve.

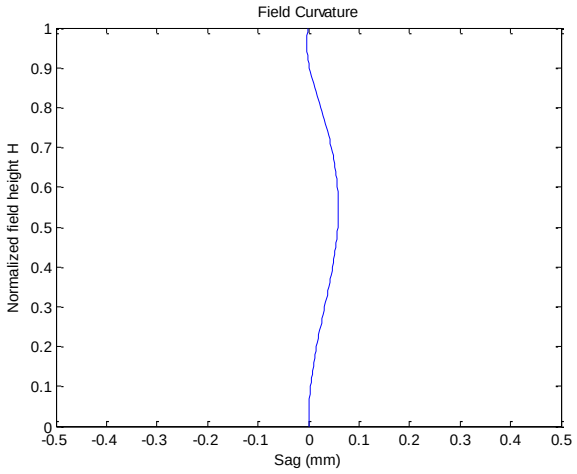


Figure 3-3 Field curvature with W_{220} = -1λ, W_{420} = 2.2λ, W_{620} = -1.2λ. (Sag = 0.0625 mm; sag reduction in percentage 85.8%)

From the optimization results, asphericity can significantly help correct field curvature. The mechanism is the 4^{th} and 6^{th} asphericities, namely A_4 and A_6 induce the 6^{th} and 8^{th} order field curvature, namely W_{420} and W_{620} respectively. Then W_{420} and W_{620} can balance W_{220} to minimize field curvature. With only two aspheric coefficients, A_4 and A_6, field curvature can be reduced more than 85%.

When high order field curvature occurs, the field curve is not quadratic anymore. Instead, the curve starts to "wiggle". The number of turning points indicates the order of asphericity. The location of the turning point depends on the optimization algorithm. This theory will be applied to some real designs in Chapter 4 to consolidate its validation.

CHAPTER 4

MODELING FIELD CURVATURE CORRECTION USING AN ASPHERIC SURFACE

The field curvature correction using high order wave aberration theory was elaborated in the previous chapter. The theory indicates that asphericity can significantly reduce field curvature. In this chapter, software simulation will demonstrate field curvature correction using aspheric surfaces. The location of the correcting asphere and the number of aspheric terms will be studied. The study starts with a design model that has pure 4^{th} order Petzval field curvature. Then, an aspheric surface with different number of aspheric terms at different locations is added. Software optimization is run to minimize the overall field curvature. Finally, the field curvature correction results are shown and conclusions are drawn.

4.1 Model of Petzval Field Curvature

A model with a very small pinhole located at the center of curvature of a concave mirror is shown in Figure 4-1. On the paraxial image plane, only 4^{th} order Petzval field curvature is displayed. The other aberrations are zero (or negligible) in this model. Spherical aberration is negligible because of the small pinhole size. Coma, astigmatism, and distortion are zero due to the symmetry. There is no color aberration for mirrors. Therefore, Petzval field curvature is the only aberration in this model.

The model adds a flat mirror behind the concave mirror that folds the image to the right just for display purposes. No additional aberration is induced due to the flat mirror. The only thing that changes is the sign of the Petzval radius which does not influence to the analysis.

The layout of the model is shown in Figure 4-1. Some design parameters are given in Table 4-1. Seidel aberration coefficients distribution in term of surfaces is shown in Figure 4-2.

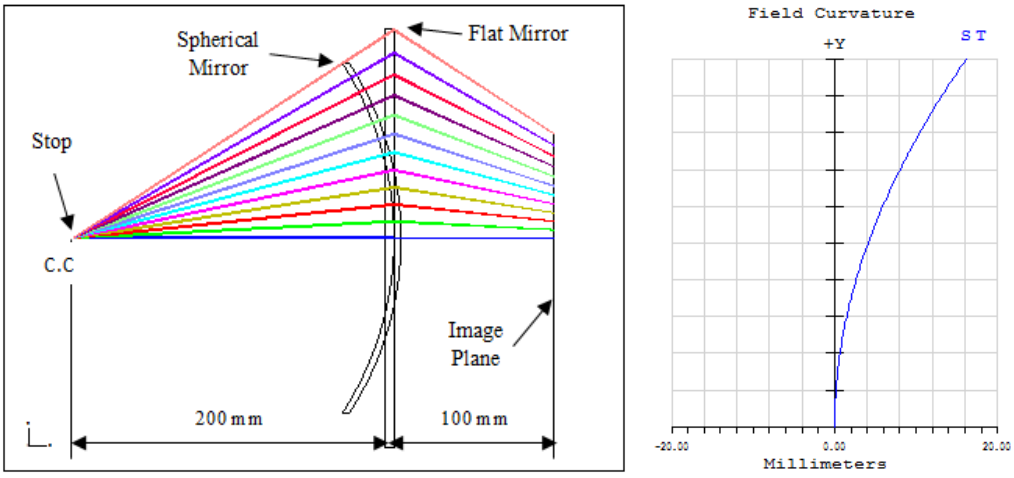


Figure 4-1 Layout of model and plot of field curvature

Stop size	1 mm
Radius of the mirror	-200 mm
Back focal distance	100 mm
FOV	± 33°
f/#	f/100
Petzval radius	100 mm
Wavelength	550 nm

Table 4-1 Specifications of field curvature model

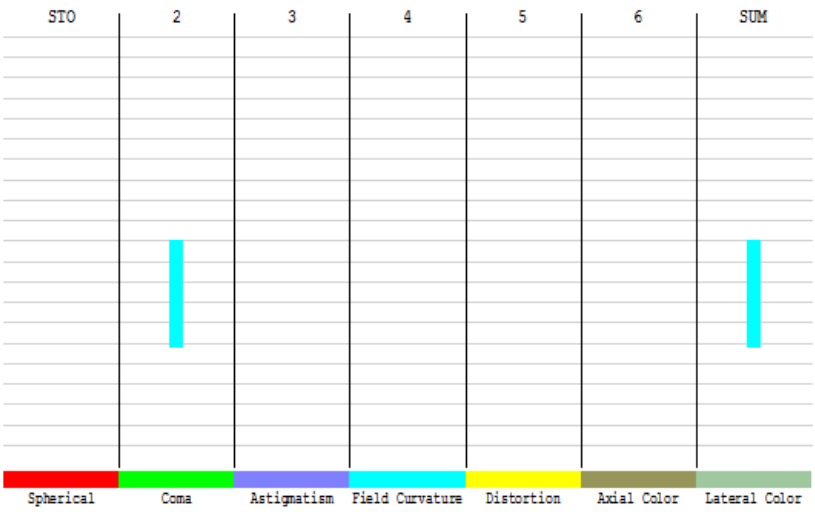


Figure 4-2 Seidel diagram of the field curvature model

In Figure 4-1, a perfect 4^{th} order Petzval field curvature is generated. Next, one aspheric surface with different number of aspheric terms will be added in the image space, and then the software optimization will be run to minimize field curvature.

4.2 Model of Field Curvature Correction using One Aspheric Surface

To balance the 4^{th} order Petzval field curvature, one asphere was added in the image space. First, the aspheric surface was located exactly at the image plane. The aspheric surface's base radius is infinity. The surface type is a mirror in order to avoid color aberration. Similarly, there is a flat mirror directly behind the asphere to fold the image to the right. Next, the 4^{th} order aspheric coefficient A_4 was set as a variable in order to optimize field curvature. In theory, A_4 will contribute to W_{420} and W_{420} will balance W_{220}. The aspheric surface is located at the image plane, so the 4^{th} order spherical aberration, coma, and astigmatism are strictly zero because of the zero marginal ray height. The

optimized aspheric surface and field curvature plots are shown in Figure 4-3. The image plane is overlapping with the aspheric surface vertex.

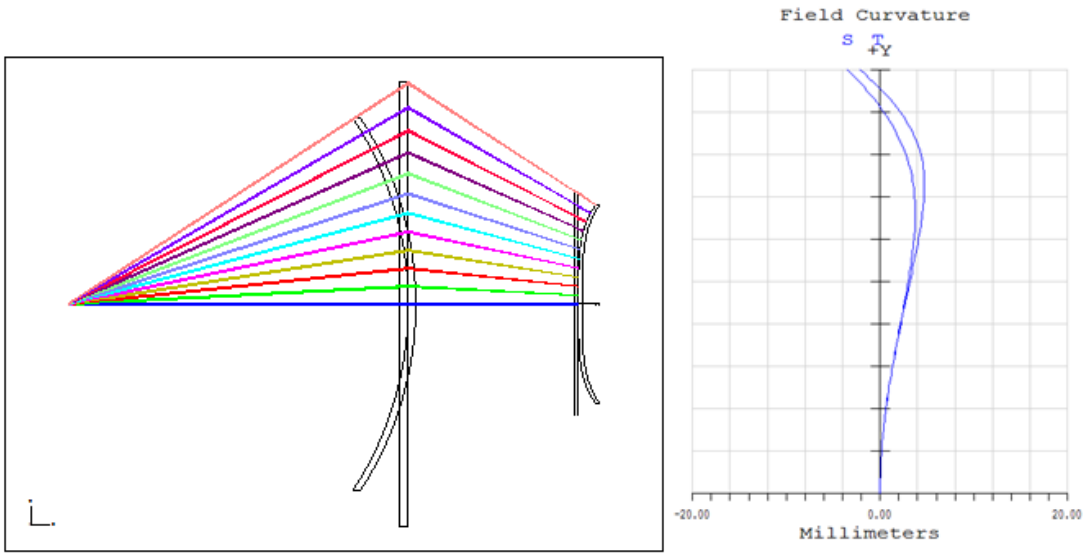


Figure 4-3 Design layout and field curvature plot with one aspheric term A_4. (P-V sag reduction is 74.4%)

Next, both A_4 and A_6 are set as variables and the optimization was run again. In this case, A_4 and A_6 will introduce W_{420} and W_{620} respectively, and W_{420} and W_{620} will balance W_{220}. The optimization results are shown in Figure 4-4.

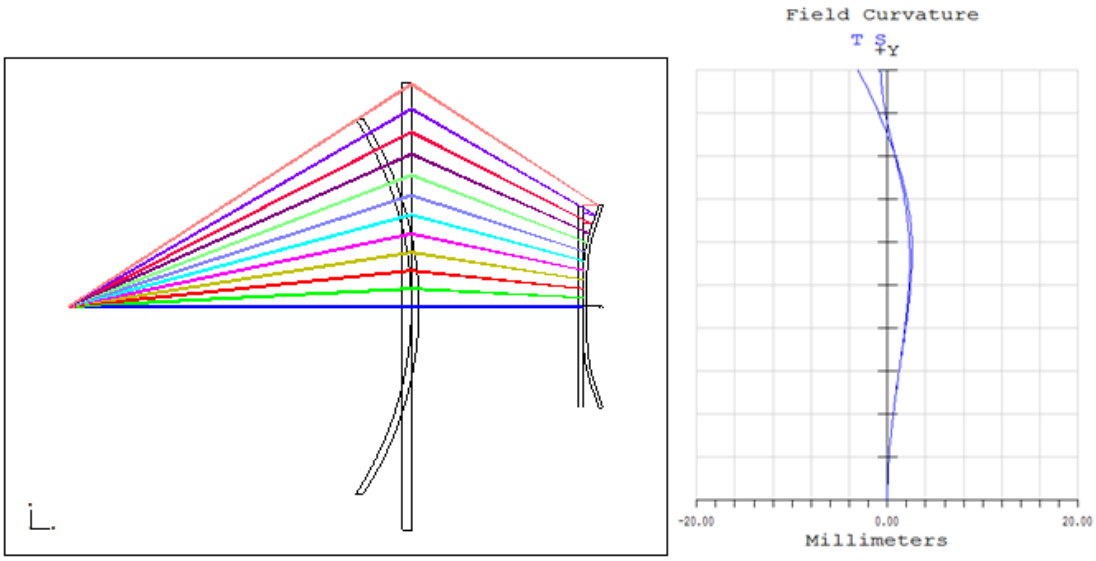


Figure 4-4 Design layout and Field curvature plot with two aspheric terms A_4 and A_6. (P-V sag reduction is 86.1%)

The software optimization results showed above agrees with the theoretical calculation showed in the previous chapter, in terms of the shape of the field curve and the amount of sag reduction. The comparison figures and table are shown in Figure 4-5, 4-6 and Table 4-2.

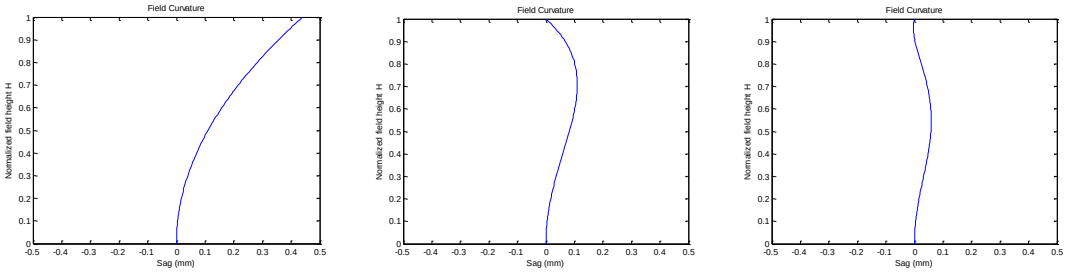


Figure 4-5 Theoretical calculation results of (a) W_{220} only, (b) W_{220} balanced with W_{420}, (c) W_{220} balanced with W_{420} and W_{620}.

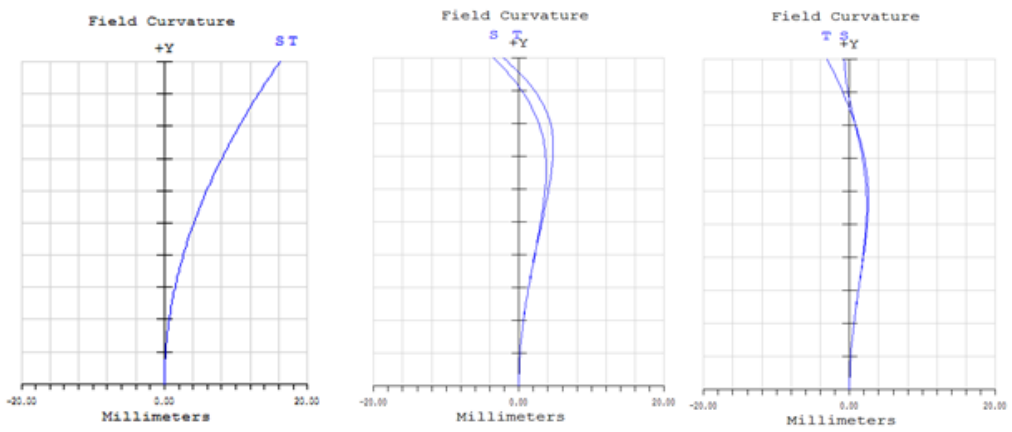


Figure 4-6 Software optimization results for (a) W_{220} only, (b) W_{220} balanced with W_{420}, (c) W_{220} balanced with W_{420} and W_{620}.

	With A_4 only	With A_4 and A_6
Theoretical Calculation	70.0%	85.8%
Software Optimization	74.4%	86.1%

Table 4-2 The amount of field curvature reduction comparison between mathematical prediction and software optimization.

From the comparison above, first, the field curve shape for both theoretical calculation and software optimization are similar. The number of turning points of the field curve depends on the number of high order aspheric terms used. Second, the amount of P-V field curvature reduction is fairly close except there is a little high order astigmatism shown in software optimization result.

Next, the aspheric surface will be moved away from the image plane. This situation is more practical because it is not always practical to locate a lens at the image plane.

When the aspheric surface moves away from the image plane, more astigmatism will occur due to the high order asphericities. In order to evaluate field curvature, the "Generalized Petzval Surface" will be introduced. When field curvature is biased with astigmatism, the tangential, medial, sagittal, and Petzval surfaces are equally spaced as shown in Figure 4-7[12]. The sag of the "Generalized Petzval Surface" can be computed when having the sag of the tangential and sagittal surfaces. Equation 4-1 computes the sag of the "Generalized Petzval Surface".

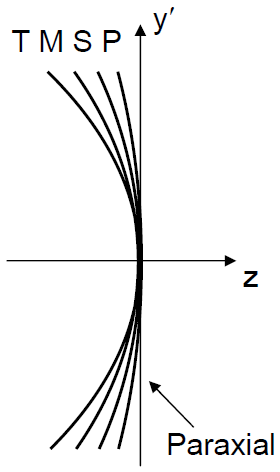
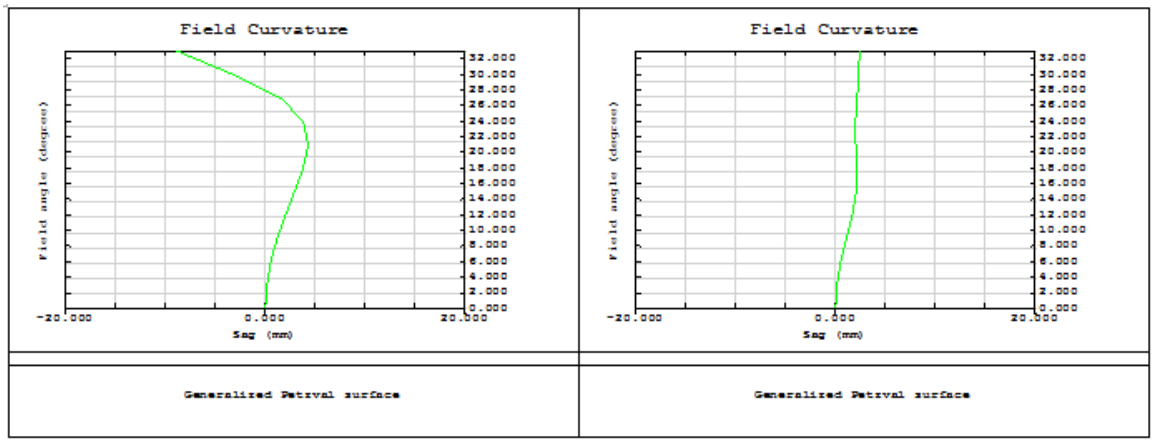


Figure 4-7 Four image surfaces (T-tangential, M-medial, S-sagittal, P- Petzval) when astigmatism is biased on field curvature.

$$Sag_{GP} = sag_S - \frac{(sag_T - sag_S)}{2} = 1.5 * sag_S - 0.5 * sag_T \qquad (4\text{-}1)$$

With equation (4-1) calculated, field curvature can be computed when the aspheric surface is away from the image plane. The same optimization procedure is followed. Figure 4-8 shows the field curvature plots of the "Generalized Petzval Surface" when the asphere is 10mm away from the image plane. The shape of the field curve is similar but the amount of field curvature reduction is different.



(a) (b)

Figure 4-8 Plots of the Generalized Petzval Surface when asphere is 10 mm away from image (a) with one aspheric term A_4 (b) with two aspheric terms A_4 and A_6

Similarly, by moving the asphere to a different location and adding different number of aspheric terms respectively, the optimization results can be obtained and shown in Table 4-3. The aspheric terms are used up to A_{16}.

Distance from image plane / Aspheric Term #	0 mm	10 mm	20 mm	30 mm	40 mm	50 mm	60mm	70 mm	80mm	90 mm	100 mm
1 (A4)	73%	45%	23%	16%	12%	9%	7%	5%	4%	3%	2%
2 (A4 ~ A6)	86%	85%	50%	46%	44%	42%	41%	39%	38%	37%	36%
3 (A4 ~ A8)	92%	81%	46%	48%	43%	40%	38%	36%	35%	34%	33%
4 (A4 ~ A10)	95%	88%	37%	50%	46%	43%	41%	39%	38%	37%	36%
5 (A4 ~ A12)	96%	89%	43%	50%	46%	43%	41%	39%	38%	37%	36%
6 (A4 ~ A14)	97%	89%	44%	50%	49%	48%	41%	45%	38%	37%	35%
7 (A4 ~ A16)	98%	90%	41%	50%	49%	48%	46%	46%	45%	45%	43%

Table 4-3 Field curvature reduction for different aspheric surface location using different aspheric terms.

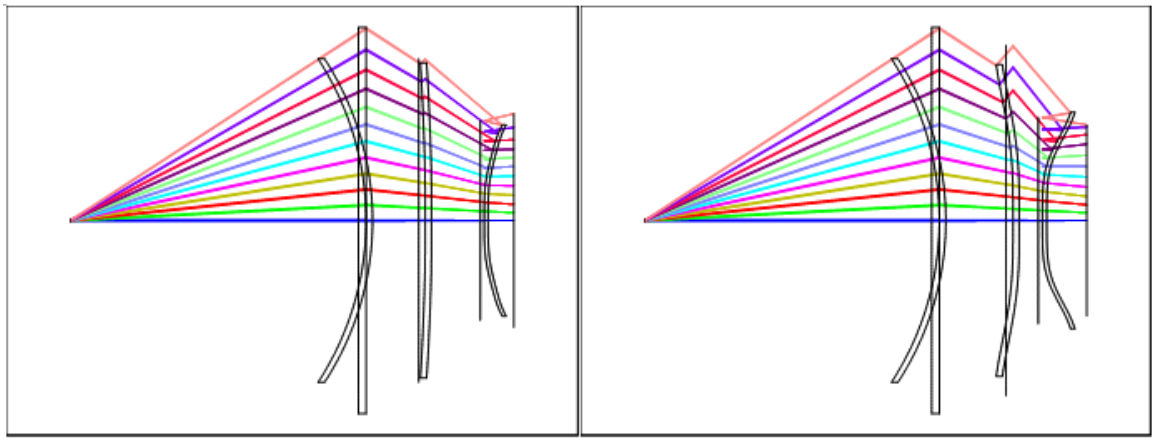
The data indicate field curvature can be substantially reduced by only two aspheric terms. The amount of field curvature reduction depends on the distant between the asphere and the image plane. The further away from the image of the asphere, the less amount the reduction of field curvature. Adding more aspheric terms general does not help much.

4.3 Model of Field Curvature Correction using Two Aspheric Surface

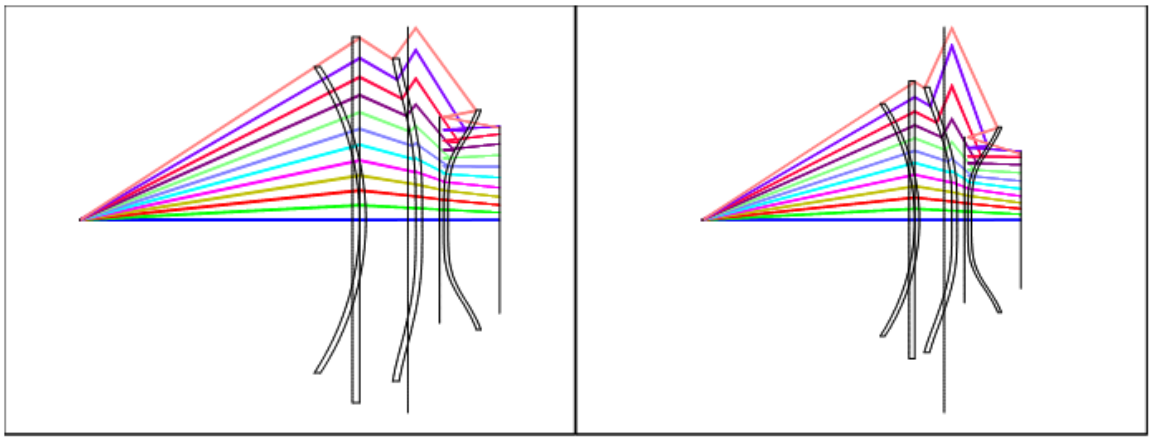
The study just showed one aspheric surface close to image plane with two aspheric terms can reduce field curvature more than 85%. When the aspheric surface moves away from the image plane, the field curvature correction is not as much no matter how many aspheric terms used. Also, more astigmatism occurs when asphere moves away from the image. So, more degree of freedom is needed, namely more aspheric surfaces, if the aspheric surface is located farther from the image.

The example below illustrates how the field curvature correction is done when we have two aspheric surfaces. The same procedures are repeated but with two aspheric surfaces away from the image (say more than 20 mm away from image out of a 100mm back focal distance) and see how the correction is. The aspheric terms used on both asphere are limited to 5 (up to the 12^{th} order even aspheric coefficient A_{12}) for the fabrication and testing consideration.

Figure 4-9 and Table 4-4 show four optimization results when using two aspheric surfaces correct field curvature.



(a) (b)



(c) (d)

Figure 4-9 Optimization results for two aspheric surfaces model.

Figure.8	Distance between two aspheres (mm)	Distance from the second asphere to the image (mm)	Aspheric terms on the first asphere	Aspheric terms on the second asphere	Field curvature reduction in percentage %
(a)	40	20	5	5	90
(b)	20	30	5	5	84
(c)	20	40	5	5	78
(d)	15	50	5	5	70

Table 4-4 Optimization data of two aspheres

From the optimization results, basically, more aspheric surfaces with more aspheric terms are needed to achieve similar amount of field curvature reduction. This makes sense because extrinsic (induced) aberrations exist when two aspheric surfaces used. Therefore, more aspheric terms are needed to correct not only intrinsic aberration but also extrinsic aberrations, namely, the total 10 aspheric terms in this example not only used to correct intrinsic field curvature, but also control other induced aberrations.

When one of the asphere is closer to the image, the correction is better. In other words, fewer terms are needed to achieve similar amount of reduction when one of the asphere is closer to the image. Also, if one of the asphere is closer to the image, the less curved it looks. So it is easier for the aspheric surface fits into the system without edge interference with other elements. On the other hand, the aspheres that are further from the image need bend more to correct field curvature so they appear curved more.

In conclusion, one aspheric surface with two aspheric terms can sharply reduce field curvature by 85% when the asphere is close to image plane. The closer the asphere to the image, the better the correction is. If even better correction needed, more aspheric surfaces are needed. Meanwhile, more total aspheric terms are needed because of the induced aberration. The exact location of the aspheric surfaces and the number of aspheric terms can be optimized by design software.

4.4 Induced Aberration

Induced aberration is also known as extrinsic aberration. Induced aberration is the aberration transferred from a previous surface or system, not exists intrinsically in the current surface itself. An example is given below to understand induced aberration.

Figure 4-10 shows an example of an on-axis point object collimated by a single positive lens and followed by a concave mirror to bring the beam to a focus. The spot diagram and the wavefront aberration plots are shown in Figure 4-11.

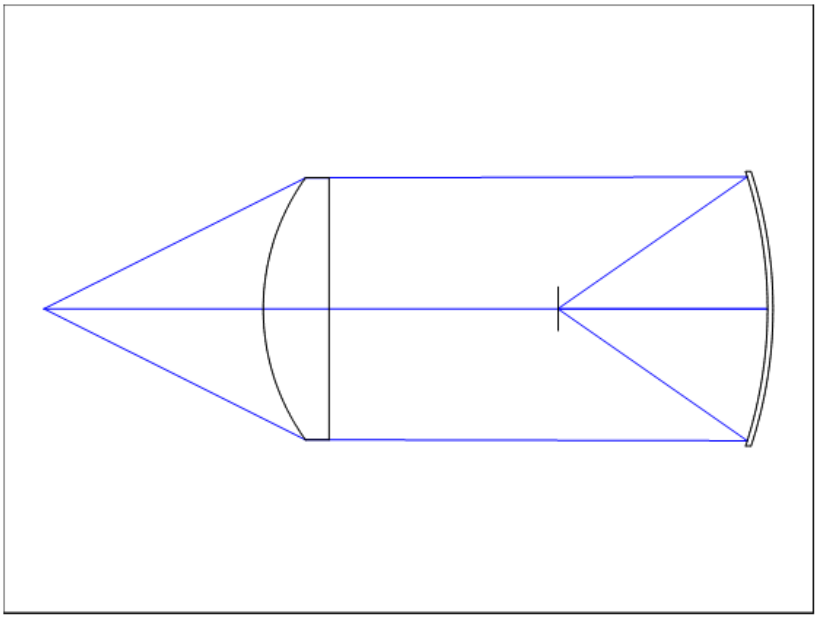


Figure 4-10 Model of induced aberration layout

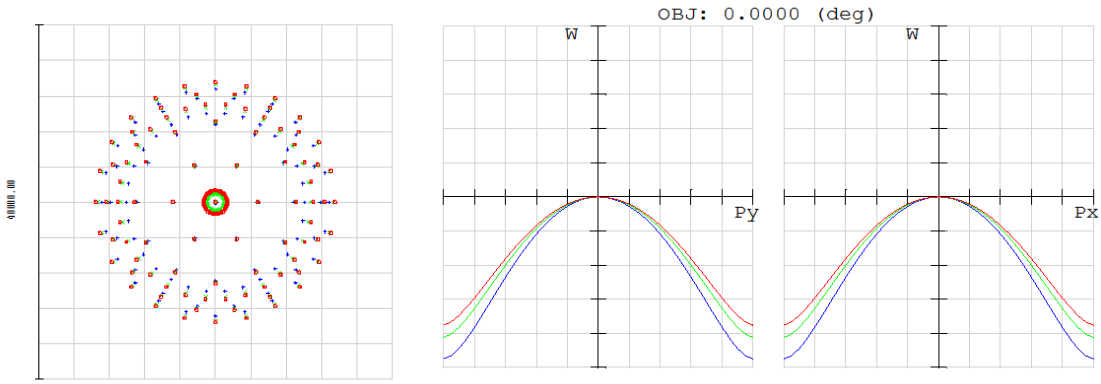


Figure 4-11 Spot diagram and OPD

Generally, mirror does not generate chromatic aberration. But the longitudinal chromatic aberration does shown in Figure 4-11. The reason for that is because the collimating lens is dispersive. Chromatic aberration is induced to the mirror and displayed in the image. It is true that mirror itself does not generate any chromatic aberration intrinsically. But the chromatic aberration generated by the dispersive lens in this case is treated as induced aberration to the mirror.

Back to the two aspheres example discussed in the last section, before the light passes through the first asphere, only the 4^{th} order Petzval field curvature exists. When the light passes through the first asphere, the high order aspheric terms generates higher order field curvature that balanced the 4^{th} order Petzval field curvature. But, at the same time, the high order aspheric coefficients also generate some unwanted high order aberration namely high order astigmatism intrinsically. These unwanted higher order astigmatism are treated as "induced aberration" to the second asphere. The second asphere not only need to generate high order field curvature to further balance field curvature, but also need to generate the opposite amount of high order astigmatism. This explains why more aspheric terms are needed for both aspheres.

4.5 Aspheric field flattener

The study shows an aspheric surface near image plane with two high order aspheric terms can significantly reduce field curvature, which agrees with the theoretical prediction. In

this section, real classical design examples will be demonstrated using aspheric field flattener lens to control field curvature.

EXAMPLE 1 Schmidt Telescope with an Aspheric Field Flattener

Schmidt telescope (also known as Schmidt camera) uses a spherical mirror with the stop at its center of curvature. An aspheric corrector is located at the stop to control spherical aberration. Since the corrector plate is located at the stop, only high order spherical aberration is induced to balance the 4^{th} order spherical aberration. The limiting aberration for the Schmidt telescope is field curvature. For a small field of view, defocus could possibly minimize the wavefront aberration. But when the system becomes faster and the FOV becomes larger, field curvature will degrade image quality significantly. Figure 4-12 shows a classical Schmidt telescope design. The telescope works at f/5 with FOV ±2°.

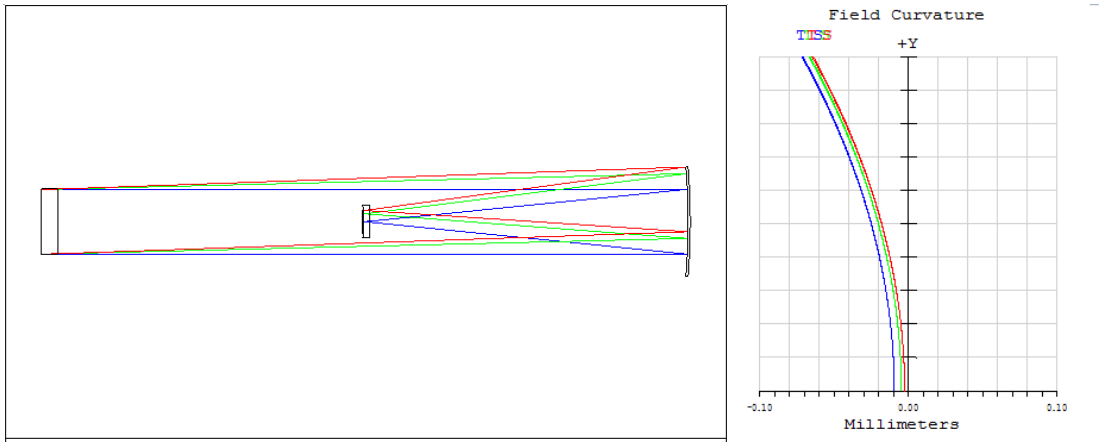


Figure 4-12 Schmidt telescope design layout and its field curvature

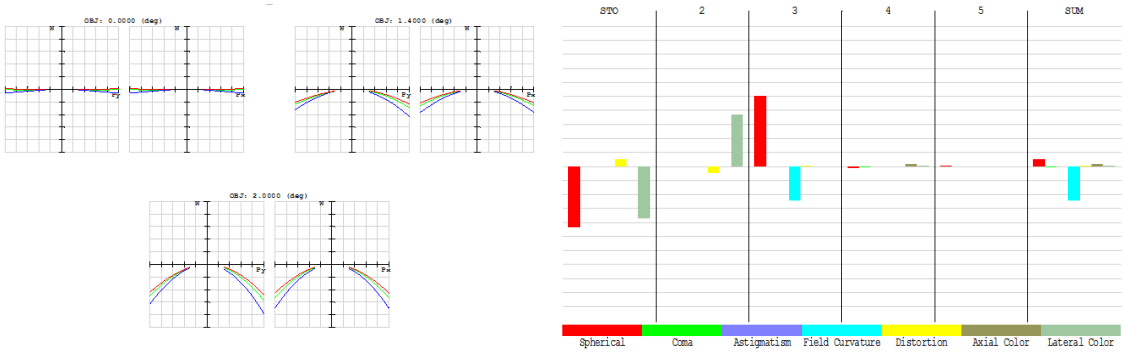


Figure 4-13 OPD (1 waves scale) and Seidel diagram

In Figure 4-12, there is about 1 wave of OPD mainly field curvature. Since there is a flat window pre-located at the image, the back surface could be set as even asphere and the high order aspheric coefficients A_4 and A_6 could be added to optimize the OPD. Figure 4-14 shows the optimization results.

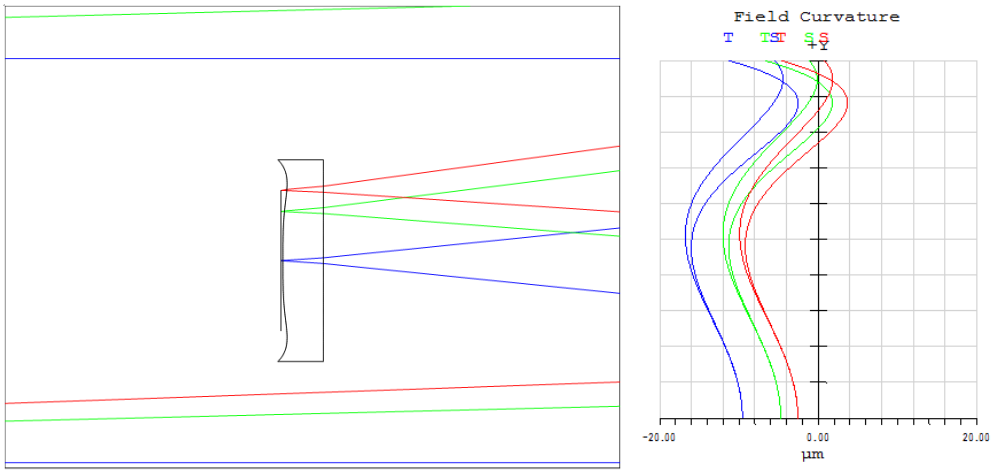


Figure 4-14 Aspheric flattener lens zoomed layout and field curvature plot when flattener lens applied.

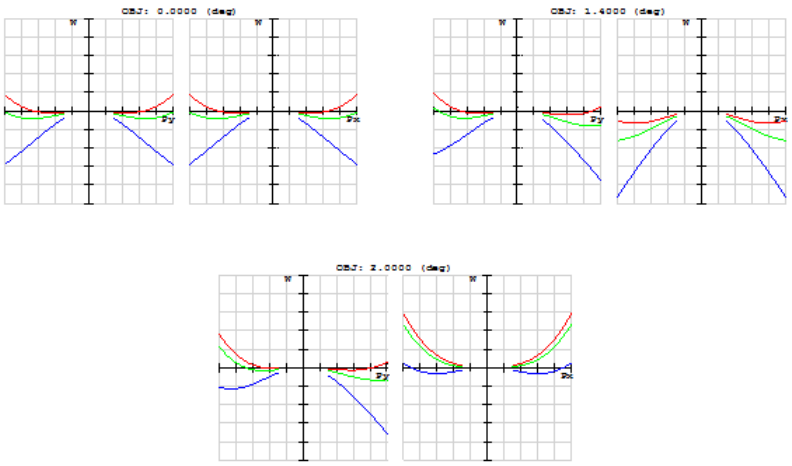


Figure 4-15 OPD plot (0.1 waves scale) when aspheric flattener lens applied

The optimization results show one aspheric surface near image plane (100μm away) with two high order aspheric coefficients (A_4 and A_6) reduces field curvature from 100μm scale to 20μm scale, and the OPD reduced from 1 wave to 0.1wave. Moreover, the field curve shape exactly agrees with the ideal model demonstrated in the previous chapter. Very small high order astigmatism induced because the aspheric flattener is not exactly at image plane.

The classical Schmidt telescope uses a positive lens to give a flat field design solution. The comparison is given below to show how the positive lens solution competes to the aspheric flattener lens solution. Figure 4-16 shows the positive lens zoomed layout and the field curvature plot, and Figure 4-17 shows the OPD. Table 4-5 compares the field curvature correction and the OPD for the two different solutions.

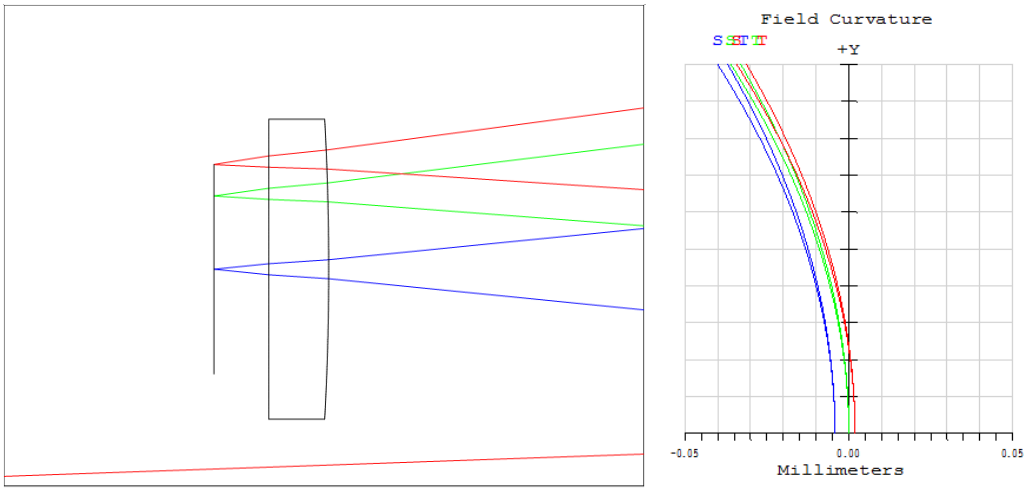


Figure 4-16 Positive lens zoomed layout and field curvature plot

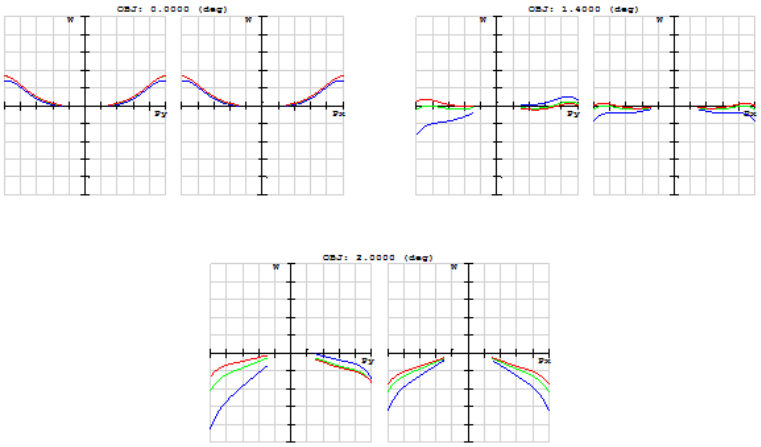


Figure 4-17 OPD plot (0.4 waves scale) when positive flattener lens applied

Solution	Field curvature	OPD
Aspheric flattener	20 µm	0.1 wave
Positive flattener	50 µm	0.4 wave

Table 4-5 Comparison of aspheric flattener solution and positive flattener solution

As shown in Table 4-5, the aspheric flattener solution has better results in terms of controlling field curvature and the OPD.

EXAMPLE 2 Petzval lens with an Aspheric Field Flattener

As shown in Chapter 2, the Petzval lens design uses a negative flattener lens near the image control field curvature. Here, an aspheric field flattener solution is discussed. First, the pure 4th order Petzval field curvature is generated. The image plane is set with a curvature which equals to the Petzval radius R_p = -100mm, and the system focal length EFL = 100mm. The design is optimized onto the Petzval surface and is shown in Figure 4-18. The OPD error is 0.01 wave. Then the image is set back to flat so that a pure quadratic Petzval field curvature is displayed and showed in Figure 4-19. The OPD error showed in Figure 4-20 is about 1 wave, mainly field curvature.

EFL	100 mm
Petzval Radius R_p	-100 mm
F/#	f/10
FOV	±4°

Table 4-6 Lens specifications

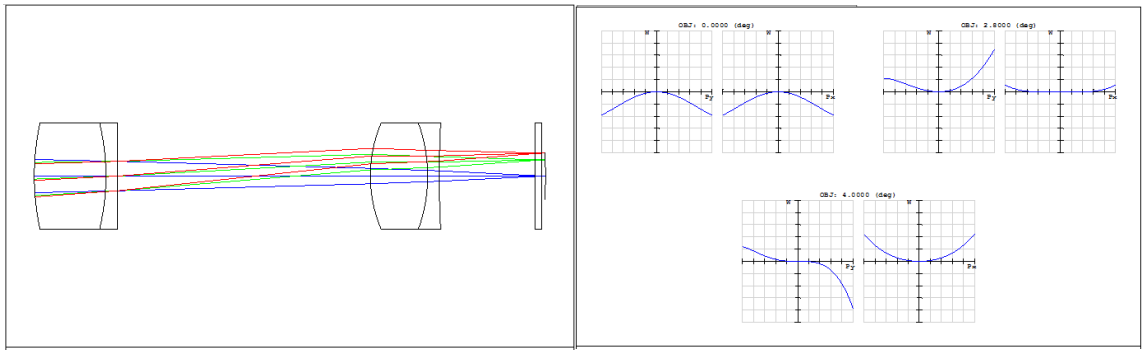


Figure 4-18 Optimized Petzval lens on its Petzval surface layout and its OPD plot (0.01 waves scale)

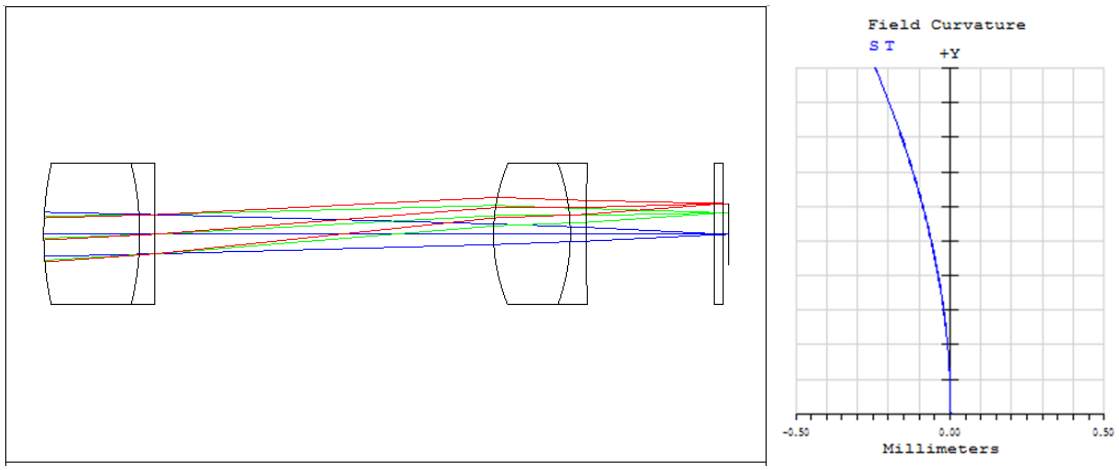


Figure 4-19 Optimized Petzval lens on a flat image surface and its field curvature plot

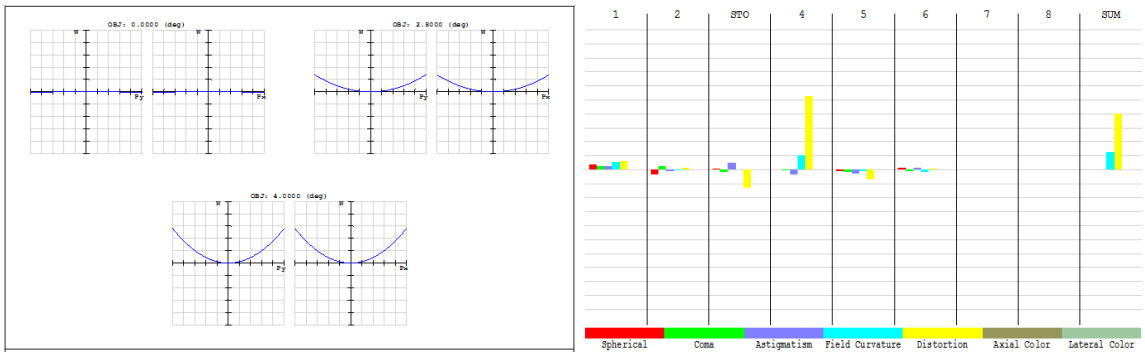


Figure 4-20 OPD (1 waves scale) and Seidel diagram

Next, the back surface of the flat window near the image is set to even asphere and optimize the design performance. Figure 4-21 shows the optimized design layout and field curvature plot for three cases: (a) original; (b) even asphere with A_4; (c) even asphere with A_4 and A_6.

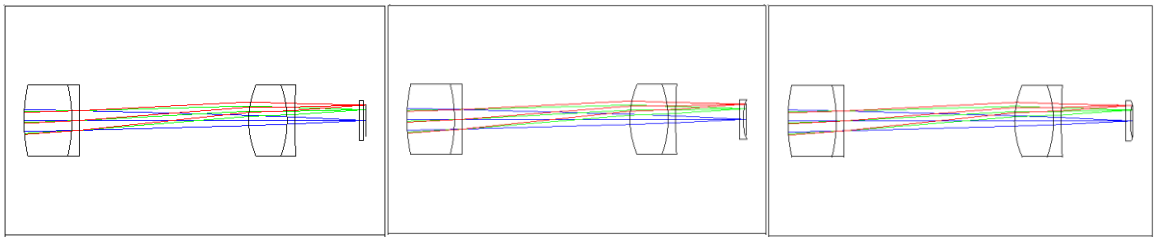


Figure 4-21 Layout of (a) original (b) asphere with one term A_4 (c) asphere with two terms A_4 and A_6

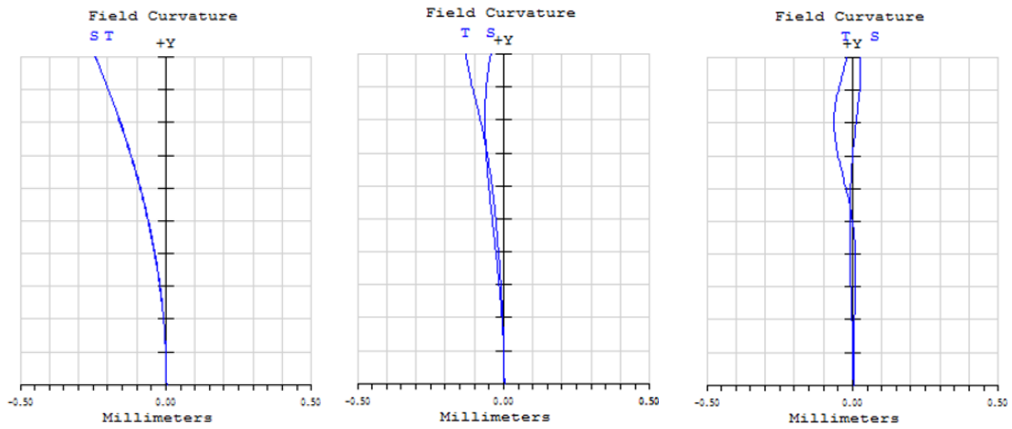


Figure 4-22 Field curvature plot of (a) original (b) asphere with one term A_4 (c) asphere with two terms A_4 and A_6

There are some astigmatism showed in the field curvature plot because of the aspheric flattener is not strictly at the image. In order to clearly show the field curvature correction, the "Generalized Petzval Surface" is plotted in Figure 4-23.

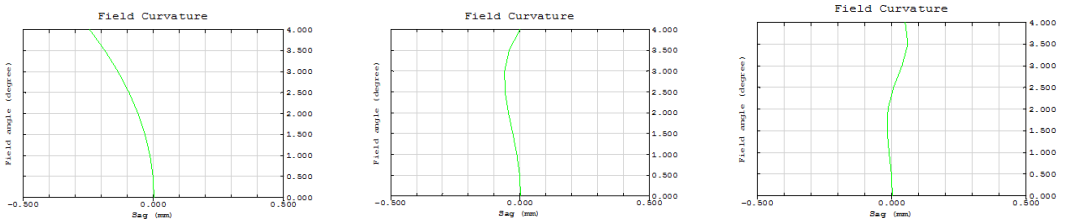


Figure 4-23 Generalized Petzval surface plot of (a) original (b) asphere with one term A_4 (c) asphere with two terms A_4 and A_6

The optimization results agree with the simulation results showed in Chapter 3. Field curvature is significantly reduced when using one aspheric surface with two high order aspheric terms.

In order to further control the induced astigmatism, both the front and the back surfaces are set as even asphere. The optimization results are showed in Figure 4-24 & 4-25 when A_4 and A_6 are used for both surfaces.

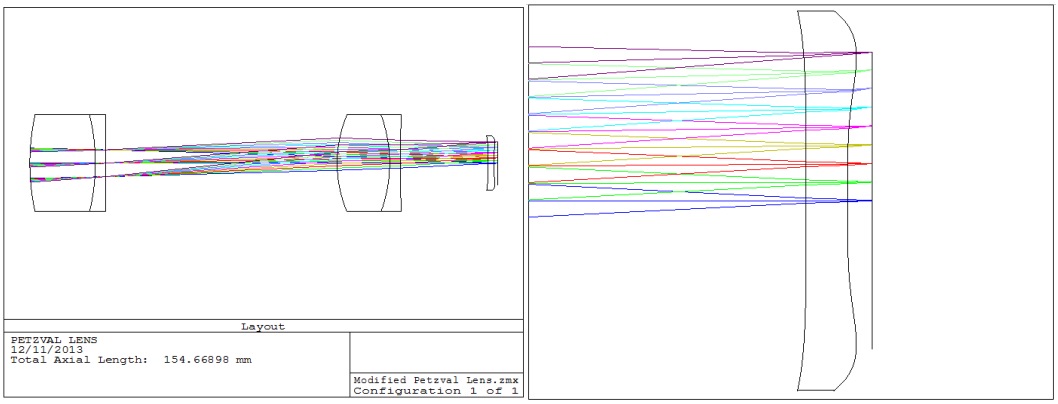


Figure 4-24 Petzval lens with a flattener lens use two aspheric surfaces

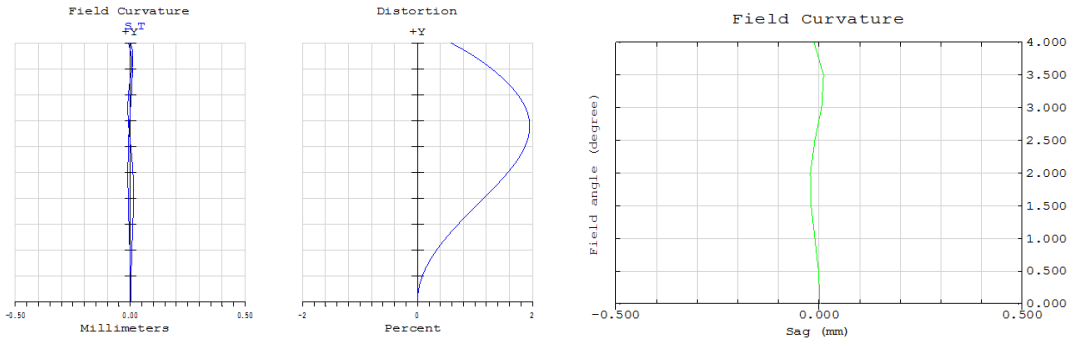


Figure 4-25 Field curvature & distortion and Generalized Petzval surface

When the aspheric field flattener is close to image plane, two aspheric surfaces with 4 high order aspheric terms could control both field curvature and astigmatism.

When the aspheric field flattener is farther away from the image, induced aberrations will be problematic so that more aspheric terms are needed. The example given below shows when the aspheric flattener is located halfway between the second doublet and the image. Figure 4-26 shows the optimization results when A_4 and A_6 are used on both the front and back surfaces.

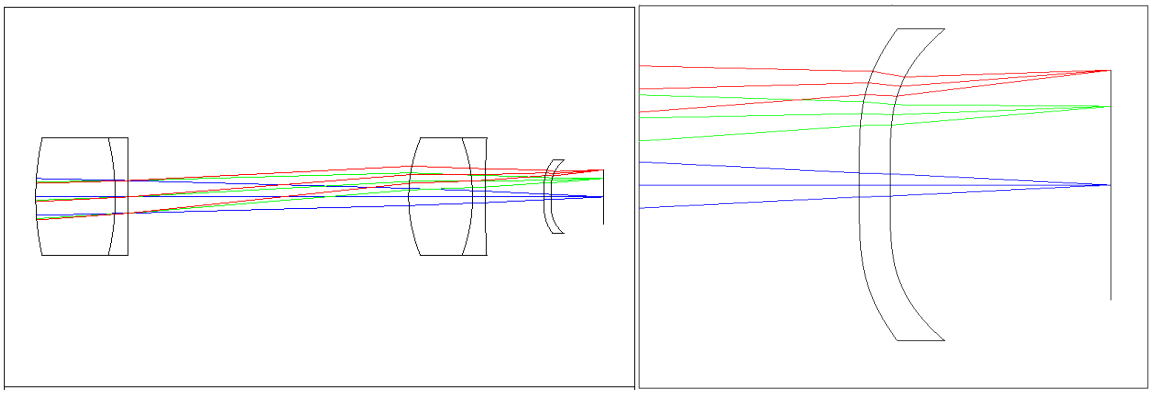


Figure 4-26 Petzval lens with an aspheric flattener lens that are away from image

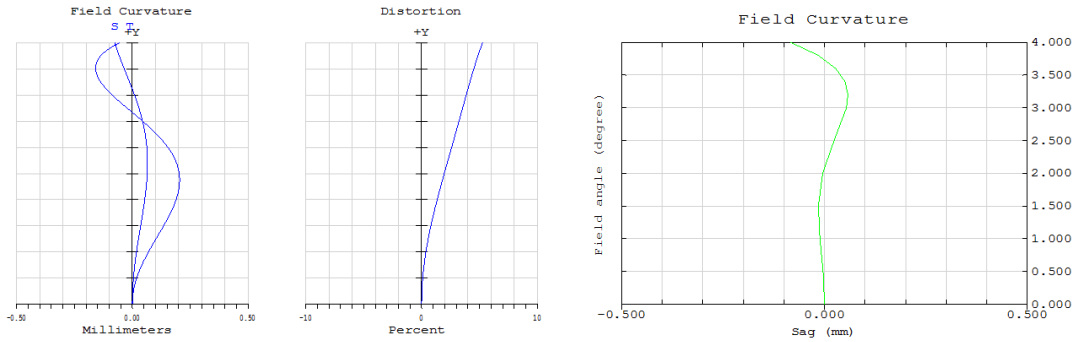


Figure 4-27 Field curvature & distortion and Generalized Petzval surface

In Figure 4-27, field curvature is well controlled but astigmatism is too large. In order to better control astigmatism, either more coefficients are needed and/or more aspheric elements are needed.

The analysis above indicates an aspheric flattener close to image plane could substantially correct field curvature for the Petzval lens. The classical Petzval lens use negative lens flattener control field curvature. The comparison between the negative flattener solution and the aspheric flattener solution is illustrated below. Figure 4-28 shows the negative lens solution zoomed layout the field curvature plot.

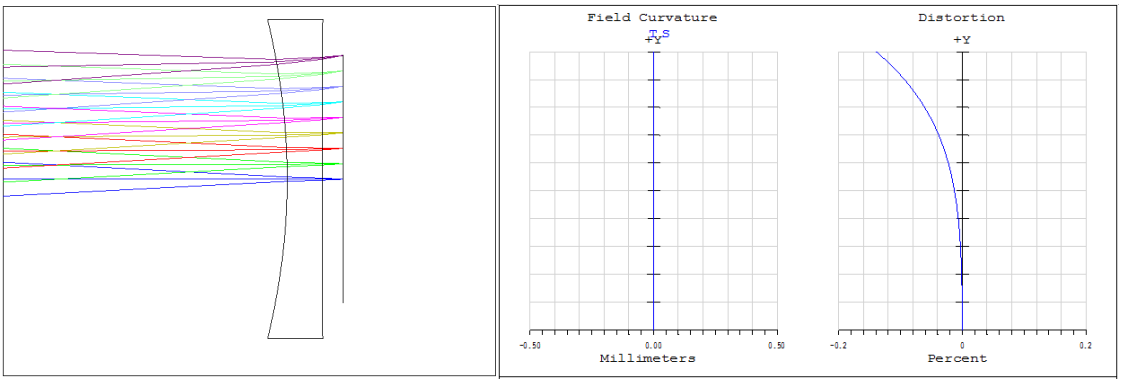


Figure 4-28 Negative flattener lens zoomed layout and field curvature & distortion plot

Compare Figure 4-28 and Figure 4-25, negative flattener solution gives an even flatter field compared to the aspheric flattener solution. However, the front two doublet lenses have to be slightly changed because of the weak negative power of the negative flattener. The superiority of the aspheric flattener lens is that it could be applied to any designs with large Petzval field curvature without affecting the first order properties of the original design.

4.6 Field curvature evaluation criteria

From the two examples shown above, when aspheric flattener is used, the actual field curve could be relatively flat even though the Petzval sum is relatively large. In other

words, the classical Petzval theorem cannot evaluate field curvature when higher order field curvature are strong. Because the high order asphericity does not contribute to optical power but substantially flattens field curvature. So, next question is how to evaluate field curvature when aspheric field flattener lens exists. Here, the thickness variation instead of optical power is used to evaluate field curvature. In this section, a new criterion will be given to evaluate field curvature and some real aspheric design examples will be demonstrated and shown how the criterion applies.

Our derivation starts from the Petzval sum in Equation (4-2).

$$Petzval\ Sum = \frac{1}{R_p} = \sum \frac{n'-n}{nn'r} = \sum \frac{\emptyset_i}{n_i} \qquad (4\text{-}2)$$

If $y^2/2$ is multiplied on both side of equation (4-2), equation (4-3) is given as:

$$\frac{y^2}{2R_p} = \sum \frac{n'-n}{nn'} * \frac{y^2}{2r} \qquad (4\text{-}3)$$

y represents the marginal ray height.

Assume the system uses the same glass for all elements, when the elements have constant thickness across the pupil for a specific field, the Petzval sum will be zero. In other word, the Petzval field curvature increases because of the element thickness variation across the pupil. The index of refraction acts as a weighing factor.

Since the constant thickness variation across pupil is the key to control field curvature, the thickness versus FOV needs to be computed in order to evaluate the aspheric contribution. Equation (4-4) computes the aspheric contribution to thickness sum.

$$Aspheric\ Contribution\ to\ Thickness\ \ Sum\ (ACTS) = \sum \frac{n'-n}{nn'} * \left(l_{asp} - \ l_{no\ asp}\right) \quad (4\text{-}4)$$

l_{asp} represents the ray path length across pupil with all aspheric coefficients on, $l_{no\ asp}$ the ray path length across pupil with all aspheric terms removed. So, for a spherical lens design, the ACTS vs FOV curve should be completely flat. When the aspheric terms are added, the ACTS will change when FOV increases.

To test the criteria of ACTS, two examples of both the aspheric surface is near and far from the image is illustrated below.

EXAMPLE 1 Petzval lens with aspheric field flattener

Figure 4-29 &4-30 shows the Petzval lens example and plots ACTS with the flattener lens near the image. Figure 4-31 &4-32 shows the Petzval lens example and plots ACTS with the flattener lens far from the image.

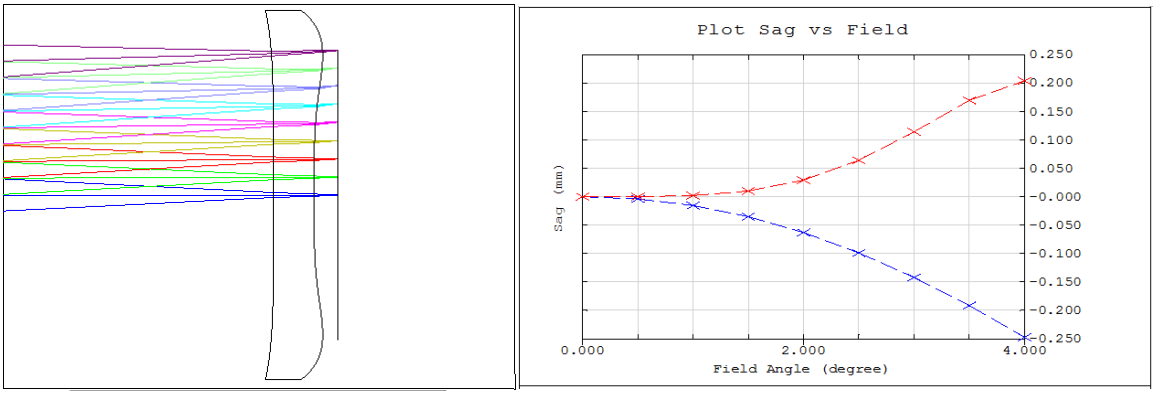


Figure 4-29 Aspheric field flattener lens that is close to image and ACTS plot

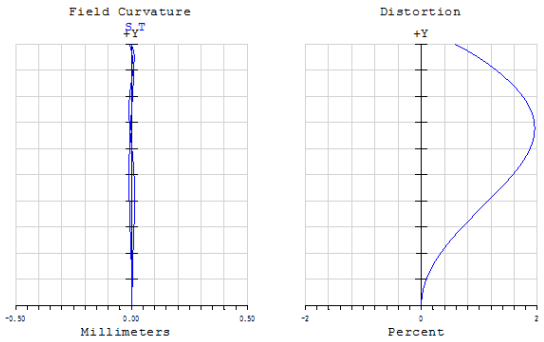


Figure 4-30 Field curvature and distortion

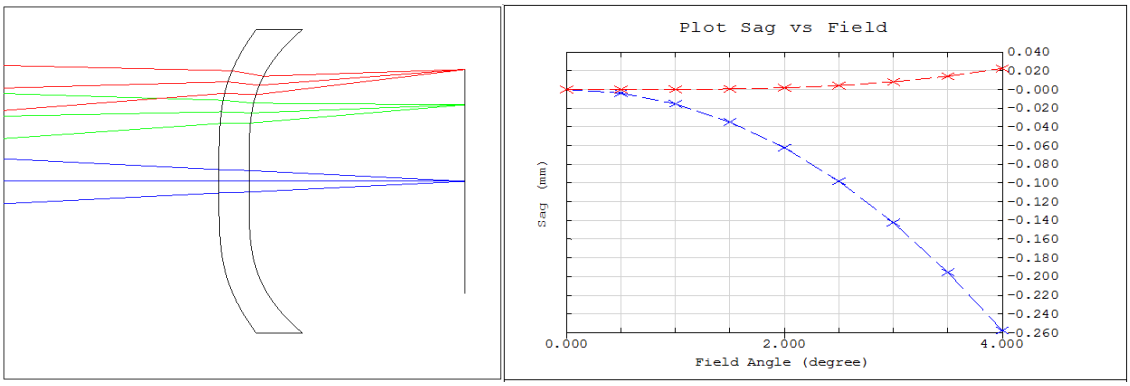


Figure 4-31 Aspheric field flattener lens that is away from image and ACTS plot

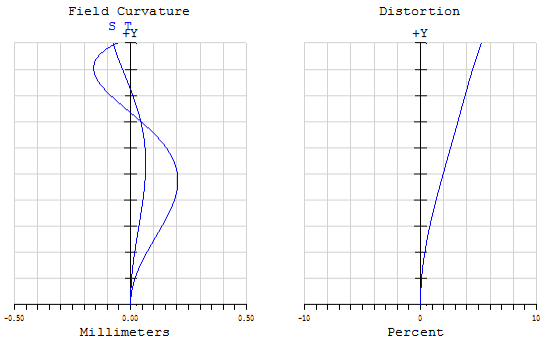


Figure 4-32 Field curvature and distortion

In the ACTS plots, the blue curve represents the Petzval curve and the red curve represents the ACTS curve. When the flattener lens is close to the image, the aspheric

terms contribute to the opposite amount of thickness that offsets the Petzval sag, therefore, the design has a flat field. When the flattener lens is away from the image, the aspheric terms contribute very little thickness so that the field is not flat.

EXAMPLE 2 Zoom lens

There are three aspheric surfaces in this zoom lens example. When switching zoom configurations, the distance from the aspheric surfaces to the image plane varies. The three different zoom configurations are plotted in Figure 4-33to show how field curvature changes when the aspheres move.

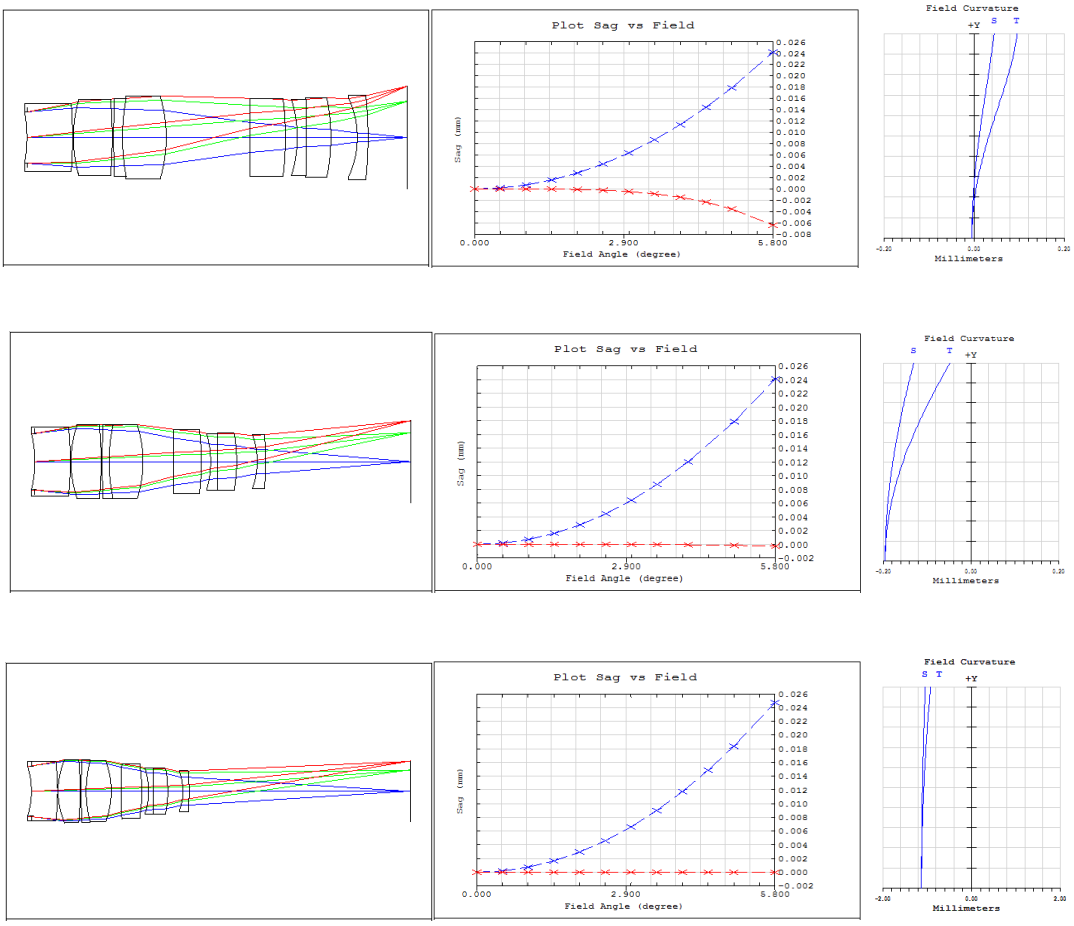


Figure 4-33 Three zoom lens configurations with their ACTS and field curvature plots

When the aspheres move away from the image, the ACTS curve becomes flatter and the field curvature becomes larger. This observation agrees with the conclusion drawn in the previous chapter.

CHAPTER 5

BALANCING ASTIGMATISM AND PETZVAL FIELD CURVATURE

5.1 Petzval and Astigmatism

Field curvature is equivalent to Petzval curvature when there is no astigmatism. When astigmatism exists, field curvature will be biased. Classical aberration theory defines field curvature as Petzval curvature plus half of the astigmatism as shown in equation (5-1).

$$W_{220} = W_{220P} + \frac{1}{2} W_{222} \qquad (5\text{-}1)$$

W_{220} represents field curvature, W_{220P} the Petzval curvature, and W_{222} the astigmatism.

In order to control field curvature, either the Petzval and astigmatism are both minimized or they balance each other. For example, if there is 1 wave of Petzval, 2 waves of astigmatism can be generated to make field curvature zero. However, the induced -2 wave astigmatism will be problematic. In this chapter, the study will be shown how to use astigmatism to balance Petzval field curvature.

5.2 High order astigmatism

As discussed in previous chapter, when the aspheric surface is away from the image, higher order astigmatisms will be induced. Higher order astigmatism has the same quadratic pupil dependency but higher order in field. The wave aberration coefficient for

6^{th} order astigmatism is W_{420}, 8^{th} order W_{620}, 10^{th} order W_{820} etc. Equation (5-1) could be written as Equation (5-2) when high order astigmatisms exist.

$$W_{220} = W_{220P} + \frac{1}{2}(W_{222} + W_{422} + W_{622} + W_{822} + \cdots) \qquad (5\text{-}2)$$

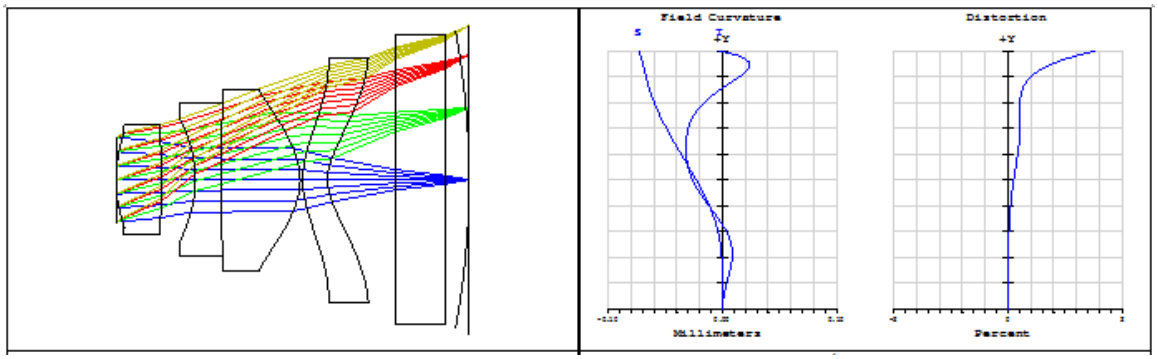
Similar as field curvature, astigmatism could also be balanced among different orders. So, the induced excessive 4^{th} order astigmatism could be used to balance Petzval and high order astigmatism will balance 4^{th} order astigmatism. In the end, the field curvature is minimized. The art of control field curvature in such cases is balancing among Petzval curvature, 4^{th} order astigmatism, and higher order astigmatism. Next, some real design examples are given and shown how the theory applies.

5.3 Examples of cellular phone lenses

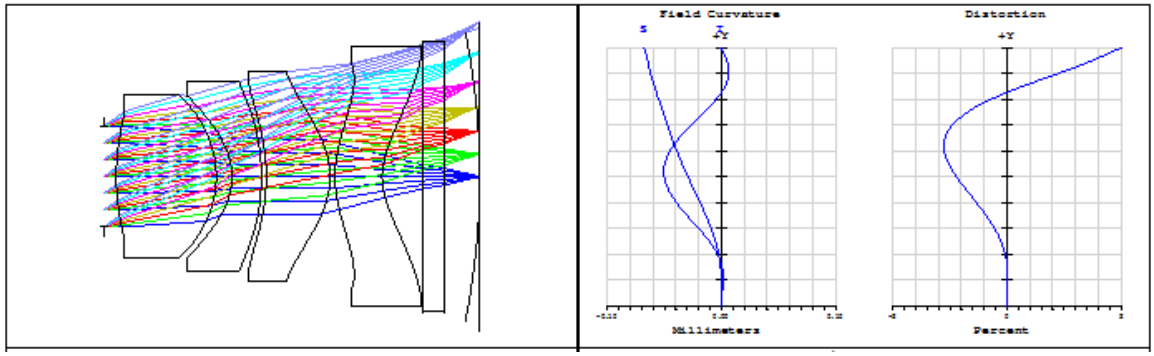
In recent years, with the rapid mobile technology revolution, a new type of cellular phone camera design came to our vision. The earliest design patent was filed on year 2004[13]. This type of design is very different from the classical lens because it corrects aberration in a different way, namely the high order asphericity plays an important role in controlling aberrations particularly field curvature.

The novel cellular phone design usually consists of 3~5 aspheric elements. The materials used are almost always plastics due to the ease of manufacturing and low cost. The total track of the design is usually within 7 mm. The design is relatively fast. The full field of view is usually over 60° therefore field curvature correction is very critical.

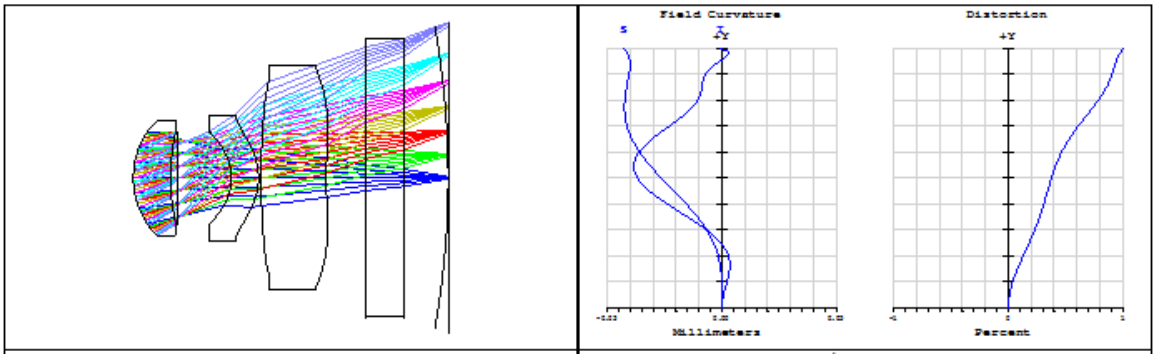
Four cellular phone design examples [13-15] are analyzed. The lens layout and field curvature & distortion plots are shown in Figure 5-1. The curved dummy surface before the image is the Petzval surface.



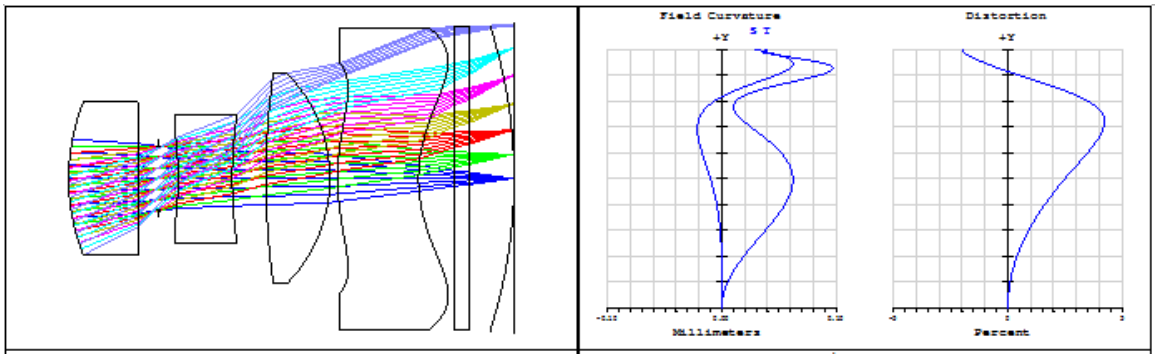
(a)



(b)



(c)



(d)

Figure 5-1 Four cellular phone lens examples and their field curvature & distortion plot

The lens specifications are listed in Table 5-1.

Design	Focal length	F/#	FOV	Petzval radius	W_{222}	W_{220P}	W_{220}
(a)	4.60 mm	2.78	± 33°	-16.2 mm	-16.15 λ	8.08 λ	0.005 λ
(b)	2.20 mm	2.14	± 35°	-7.4 mm	-15.82 λ	7.91 λ	0
(c)	3.40 mm	2.95	± 30°	-10.9 mm	-9.66 λ	4.83 λ	0
(d)	3.48 mm	3.45	± 30°	-6.3 mm	-12.03 λ	6.01 λ	-0.005 λ

Table 5-1 Lens specifications

From the analysis data, the field curves are relatively flat although the Petzval sags are relatively large. By computing the wave aberration coefficients, the induced 4^{th} order astigmatism is used to balance the excessive Petzval. By observing the shape of the field curve, high order astigmatisms are used to balance the 4^{th} order astigmatism.

After reviewing these designs example's prescriptions, it indicates most designs used many aspheric terms on the field flattener lens, namely, up to the 16^{th} order A_{16}. Based on

the theory, when the aspheric flattener lens is far from the image, two aspheric terms are enough in terms of field curvature correction. So, it indicates these designs overused aspheric terms and it is possible to simplify them, namely decreases the aspheric terms but keeps similar imaging performance.

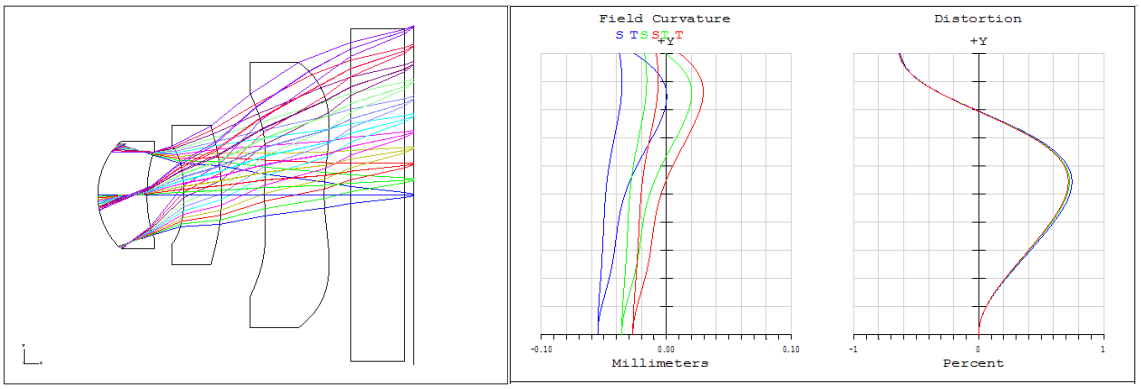
An example is given below to show how the asphericity is reduced from the 16th order to the 6th order. Figure 5-2 shows the original lens prescription data; Figure 5-3 shows the design layout, field curvature & distortion curves, and the OPD error.

Lens Data Editor

Edit Solves View Help

Surf:Type		4th Order Term	6th Order Term	8th Order Term	10th Order Term	12th Order Term	14th Order Term	16th Order Term	Par 9
OBJ	Standard								
STO	Standard								
2	Even Asph..	-0.03087130	2.576580000	-14.9944000	68.59740000	0.000000000	0.000000000	0.000000000	
3	Even Asph..	-0.01396230	-0.46101800	-22.0517000	81.73830000	0.000000000	0.000000000	0.000000000	
4	Even Asph..	-1.06380000	-3.68074000	40.83450000	-155.050000	-1534.35000	0.000000000	0.000000000	
5	Even Asph..	-0.72420400	2.548190000	-0.39187200	-9.78736000	24.34210000	0.000000000	0.000000000	
6	Even Asph..	-0.81727700	0.677648000	-0.12249400	0.040812300	0.083987900	6.1826E-003	-0.13217900	
7	Even Asph..	-0.41909000	0.247890000	-0.33376400	0.174269000	7.2481E-003	-0.04158750	-9.318E-004	
8	Standard								
9	Standard								
IMA	Standard								

Figure 5-2 Lens prescription data



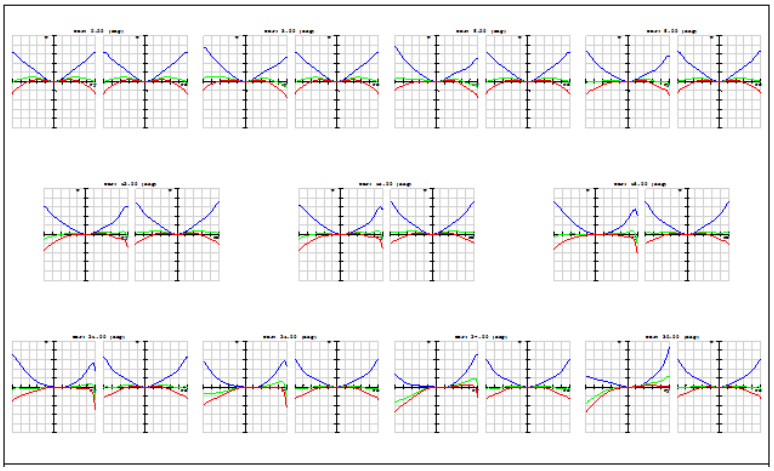


Figure 5-3 Original design and its field curvature & distortion and OPD plots (1 waves scale)

The field flattener (surface 6, 7) has aspheric coefficients up to the 16th order. The goal is to reduce the aspheric order but keep the similar OPD error and field curvature. All the first order properties such as focal length, f/#, FOV, lens base curvature, thickness, and spacing are fixed as original during the optimization process. In order words, only high order aspheric coefficients are allowed to change. Figure 5-4 and 5-5 show the simplified design and its performance.

Lens Data Editor

Edit Solves View Help

Surf:Type		4th Order Term		6th Order Term		8th Order Term		10th Order Term		12th Order Term		14th Order Term		16th Order Term		Par 9
OBJ	Standard															
STO	Standard															
2	Even Asph..	-0.23278569	V	-4.59302587	V	54.59198450	V	-241.663852	V	0.000000000		0.000000000		0.000000000		
3	Even Asph..	1.182698072	V	-13.8641320	V	107.8341140	V	65.37757549	V	0.000000000		0.000000000		0.000000000		
4	Even Asph..	-1.06380000		-3.68074000		40.83450000		-155.050000		-1534.35000		0.000000000		0.000000000		
5	Even Asph..	-0.72420400		2.548190000		-0.39187200		-9.78736000		24.34210000		0.000000000		0.000000000		
6	Even Asph..	-1.12424417	V	0.839064677	V	0.000000000		0.000000000		0.000000000		0.000000000		0.000000000		
7	Even Asph..	-0.37058062	V	-0.01685965	V	0.000000000		0.000000000		0.000000000		0.000000000		0.000000000		
8	Standard															
9	Standard															
IMA	Standard															

Figure 5-4 Lens prescription data after optimization

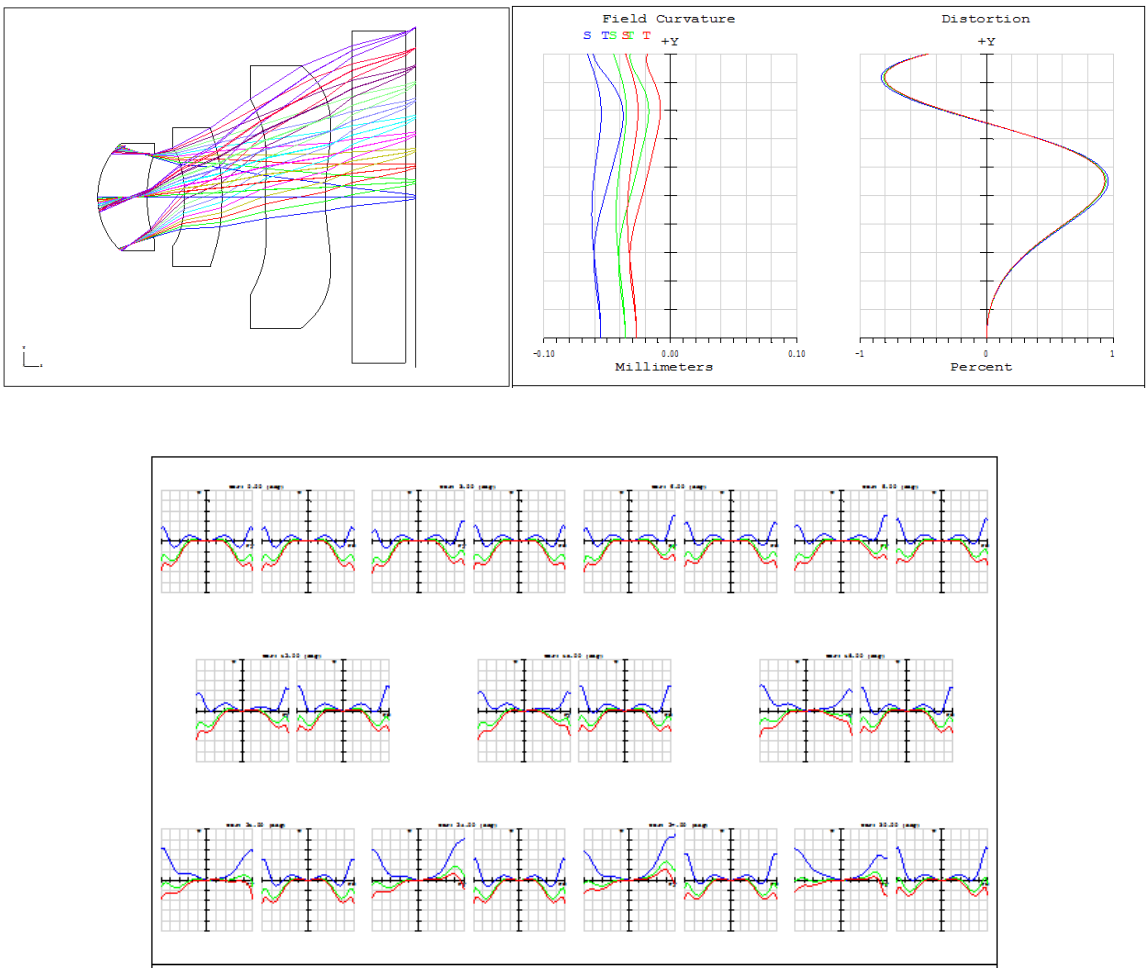


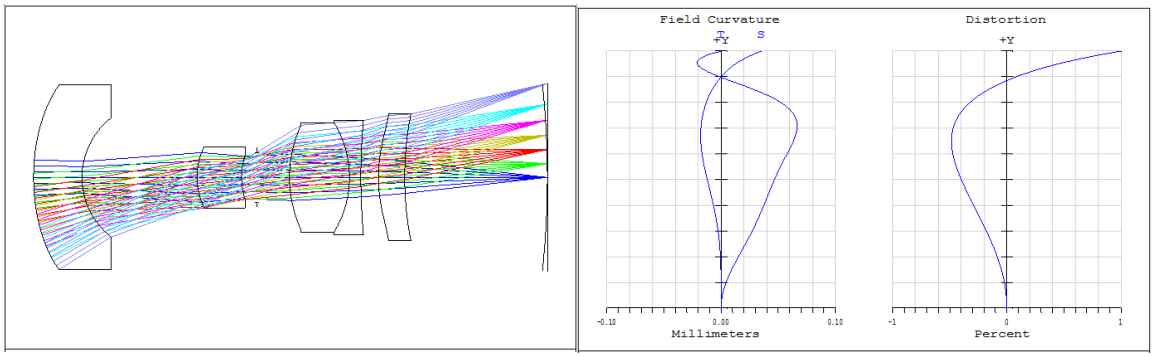
Figure 5-5 Re-optimized design and its field curvature & distortion and OPD plots

Compare the re-optimized design to the original design, the aspheric terms used on the field flattener are reduced from the 16th order down to the 6th order without affecting the imaging performance namely OPD error, field curvature, and distortion. The success of simplifying the aspheric design could tremendously ease the design tolerancing, fabrication, and testing of the aspheric elements. At the same time, it validates the previous conclusion drawn in the previous chapter.

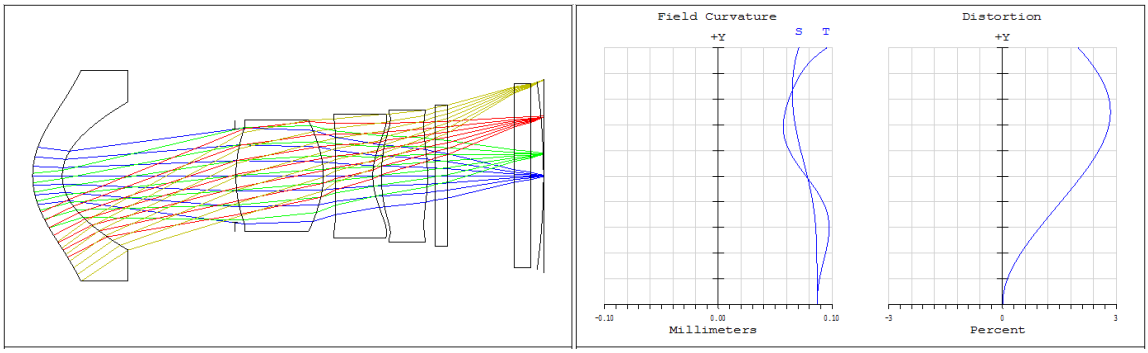
5.4 Examples of wide angle lenses

Field curvature aberration is a function of FOV. The Petzval theorem indicates field curvature increases quadratically with FOV. Therefore, it is critical to control field curvature in wide angle lenses. In this section, four wide angle lenses design [16-17] will be analyzed and shown how astigmatisms help control field curvature.

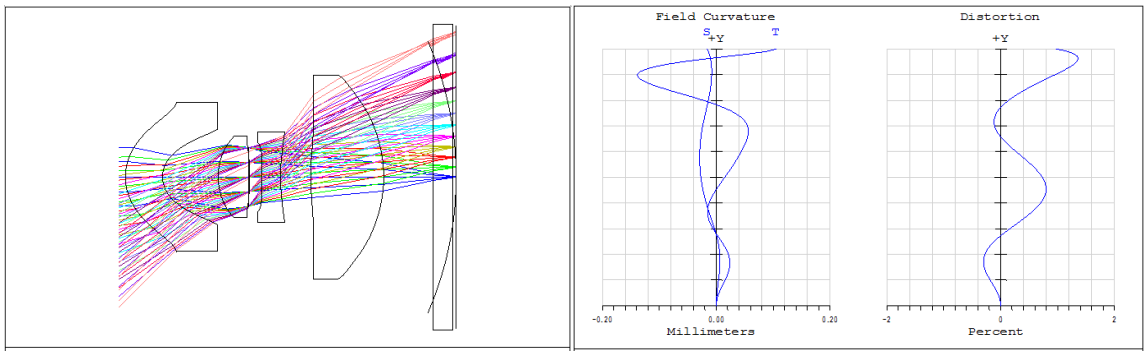
The lens layout and field curvature & distortion plots are shown in Figure 5-6. The curved dummy surface before image plane is the Petzval surface. Table 5-2 gives the lens specification data.



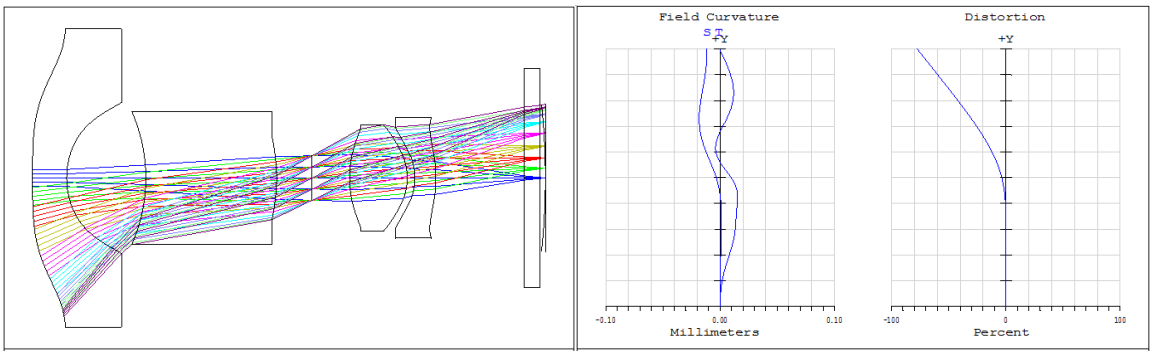
(a)



(b)



(c)



(d)

Figure 5-6 Four wide angle lens examples and their field curvature & distortion plot

Design	Focal length	F/#	FOV	Petzval Radius	W_{222}	W_{220P}	W_{220}
(a)	9.00 mm	4.5	± 30.5°	-49.7 mm	-6.34 λ	3.17 λ	0
(b)	3.01 mm	2.15	± 37.5°	-16.4 mm	-15.87 λ	7.89 λ	-0.045 λ
(c)	2.25 mm	2.42	± 45°	-5.33 mm	-41.35 λ	20.70 λ	0.025 λ
(d)	1.06 mm	3.15	± 80°	-13.1 mm	-63.71 λ	31.85 λ	-0.005 λ

Table 5-2 Lens specifications

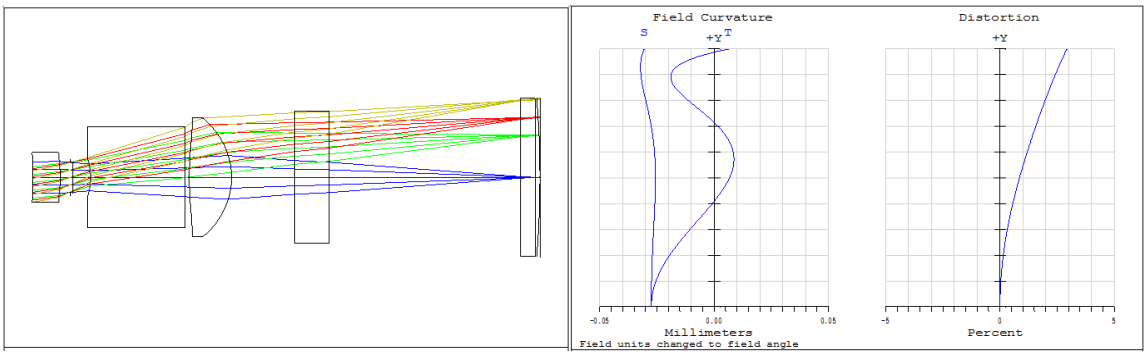
Aspheric surfaces are commonly used in these 4 wide-angle lenses. High order astigmatisms are obvious by observing the field curvature plots. Similarly, the field

curvature correction mechanism applied to wide angle lenses as well as cellular phone lenses. The aspheric surfaces induced high order astigmatisms and they were used to balance the 4th order astigmatism and Petzval.

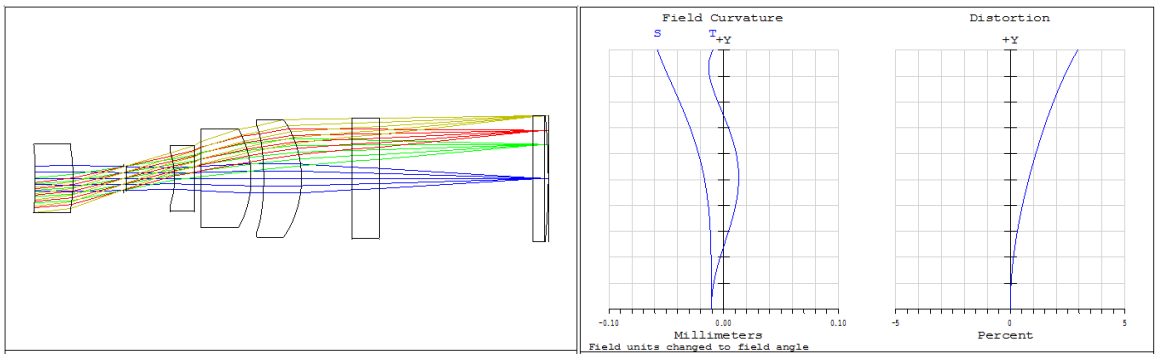
5.5 Miscellaneous examples

Examples of modern cellular phone lenses and wide angle lenses are given to demonstrate the balancing between astigmatism and Petzval curvature. In this section, more examples that use less aspheric terms are given.

The lens layout and field curvature & distortion plots are shown in Figure 5-7. The curved dummy surface before image plane is the Petzval surface. Lens specification data are shown in Table 5-3.



(a)



(b)

Figure 5-7 Two barcode scanner lens examples and their field curvature & distortion plot

Design	Focal length	F/#	FOV	Petzval Radius	W_{222}	W_{220P}	W_{220}
(a)	6.70 mm	6.67	± 19.8°	-30.8 mm	-1.16 λ	0.51 λ	-0.07λ
(b)	7.98 mm	8.00	± 16.8°	-25.2 mm	-0.88 λ	0.44 λ	0

Table 5-3 Lens specifications

For both lenses, the last element before the flat window is the aspheric corrector lens and the aspheric terms used are up to the 6^{th} order. As only A_4 and A_6 terms are used, the astigmatisms are more obvious to observe. The aspheric terms A_4 and A_6 could induce 6^{th} and 8^{th} order astigmatism, namely W_{422} and W_{622}. In field curvature plot, the 6^{th} and 8^{th} order astigmatisms help flatten the tangential curve in order to minimize the 4^{th} order astigmatism.

CHAPTER 6

CONTROL PETZVAL FIELD CURVATURE BY OBLIQUE SPHERICAL ABERRATION

6.1 Oblique spherical aberration

Oblique spherical aberration is one of the new 6th order aberrations. It is similar to the 4th order spherical aberration but with a quadratic dependency on field. The wave aberration coefficient for oblique spherical aberration is W_{240}. Oblique spherical aberration is common in fast photographic lenses such as double Gauss lens. In this chapter, a study will show how to use oblique spherical aberration balance Petzval field curvature.

6.2 Balancing Petzval field curvature with Oblique spherical aberration

Assume a system with all 4th and 6th order aberrations well controlled except oblique spherical aberration and Petzval field curvature, the wave aberration could be written as

$$W = W_{220P}H^2\rho^2 + W_{240}H^2\rho^4 = H^2\,(W_{220P}\,\rho^2 + W_{240}\,\rho^4) \qquad (6\text{-}1)$$

As the two aberration coefficients $W_{220P}$and W_{240} have the same field dependency, the balancing is happened between the quadratic dependent pupil and 4th order pupil. The plots in Figure 6-1 show the oblique spherical with the opposite amount of Petzval could reduce wavefront aberration more than 70%.

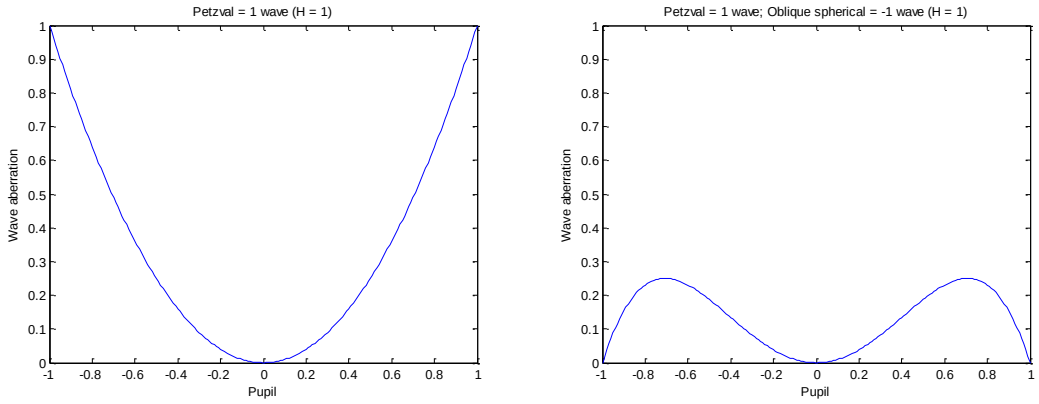


Figure 6-1 Wavefront aberration of (a) Petzval only (b) Petzval and oblique spherical

6.3 Double Gauss lens example

Oblique spherical aberration exists in fast Double Gauss type of photographic lenses. An Double Gauss lens works at F/3 with field of view ±14° is given below. Figure 6-2 shows the lens layout and OPD. Figure 6-3 shows the field curvature plot and Seidel coefficients.

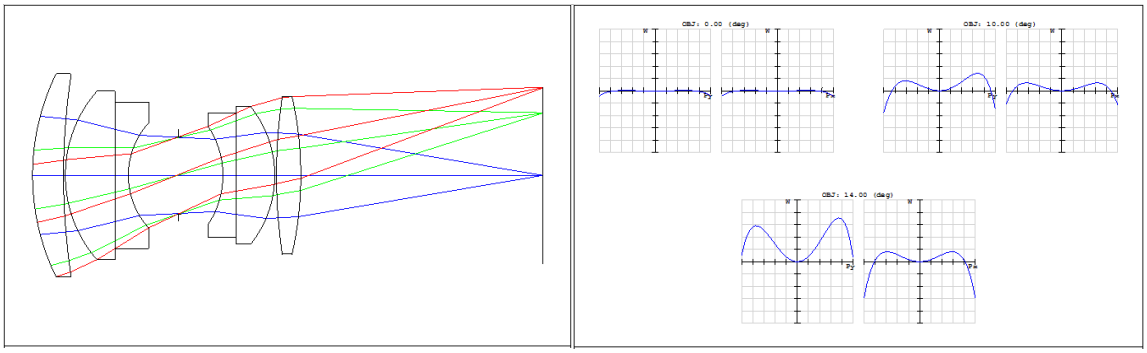


Figure 6-2 Double Gauss lens layout and OPD (5 waves scale)

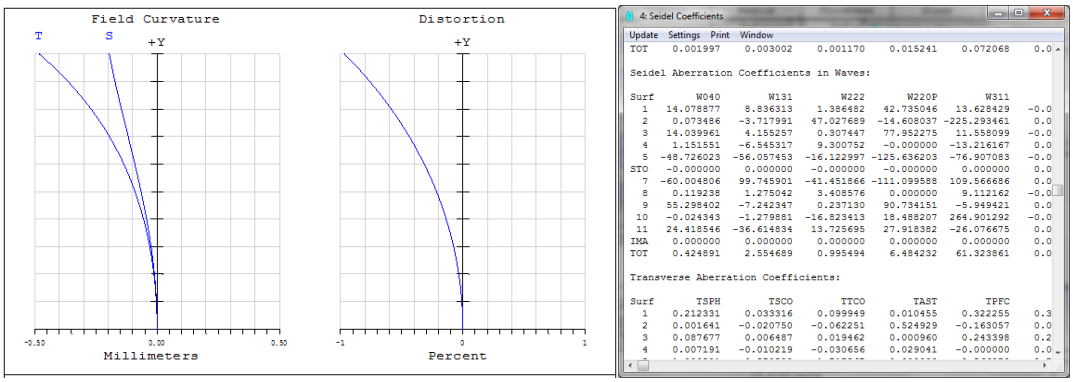


Figure 6-3 Field curvature & distortion plot and Seidel coefficients charts

In the OPD plot, almost no on-axis aberration appears. Off-axis Petzval curvature and oblique spherical balanced each other at different field. To verify the observation, the wave aberration coefficients are computed and listed in Table 6-1.

Aberration	Coefficient	Amount in waves
Spherical	W_{040}	0.4 wave
Oblique spherical	W_{240}	-6.3 wave
Petzval	W_{220P}	6.5 wave

Table 6-1 Wave aberration coefficients

Table 6-1 shows the oblique spherical and Petzval has almost equal amount of aberration but different sign. Therefore, the off-axis OPD plot shows the quadratic and 4^{th} order balancing as the simulation shown in section 6.2.

Another example is given below. This lens works at F/2.3 with a FOV ±25°. It is a double Gauss type of lens as well but even faster and has a larger FOV compared to the previous lens. By separating the front meniscus lens and shifting the stop, the overall aberration is better controlled.

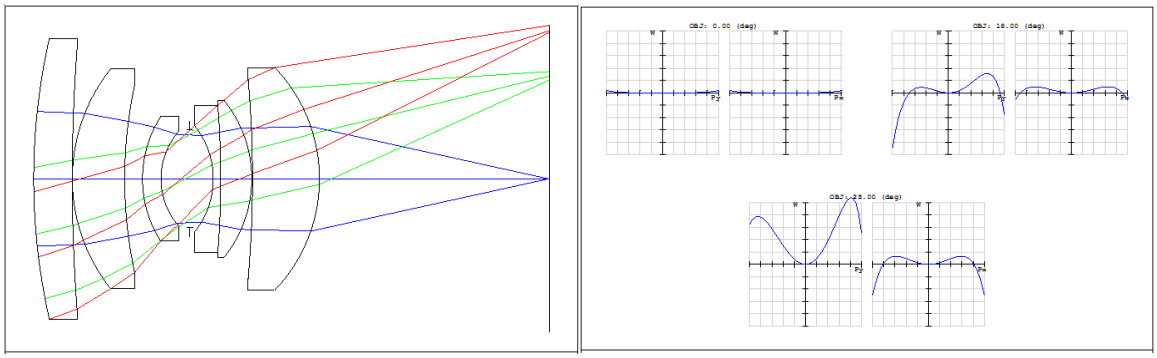


Figure 6-4 Lens layout and OPD (1 waves scale)

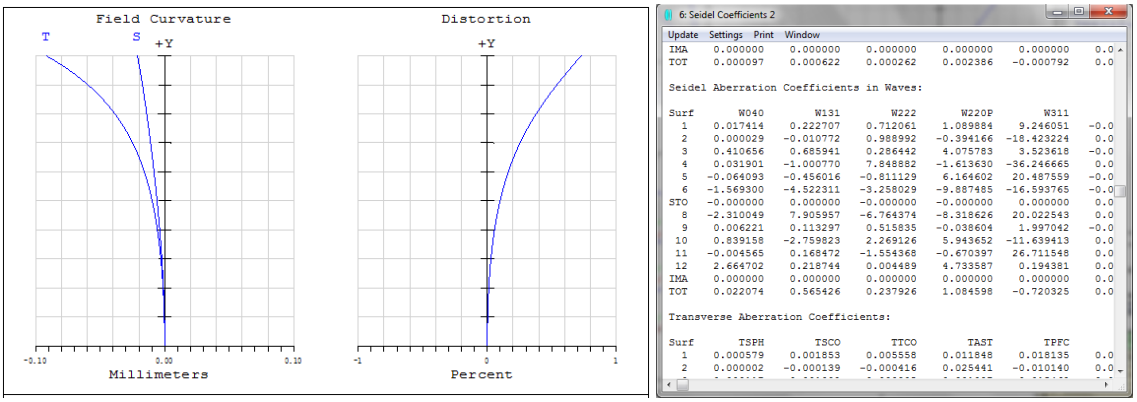


Figure 6-5 Field curvature & distortion plot and Seidel coefficients charts

Aberration	Coefficient	Amount in waves
Spherical	W_{040}	0.02 wave
Oblique spherical	W_{240}	-1.01 wave
Petzval	W_{220P}	1.08 wave

Table 6-2 Wave aberration coefficients

By inspecting the OPD plot, there are very little defocus and spherical aberration terms on axis. The off-axis plot indicates the balancing between oblique spherical aberration and Petzval which is verified by computing the wave aberration coefficients which shown in Table 6-2.

CHAPTER 7

NOVEL ASPHERIC SURFACES

Asphere, by definition, means any surfaces that depart from a perfect sphere. Aspheric surface is not new in lens design. The first high-quality aspheric lens was made and used on telescopes back to 1660s [18]. Nowadays, more and more aspheres are used because of the boost precision fabrication and testing technology. The examples of the optical systems that use aspheres include the aspheric eyeglasses, modern photographic zoom lenses, the nano-scaled precision lithographic objective lenses etc.

The superiority of aspheric surface in optical design is the additional degrees of freedom that can control aberrations. Moreover, asphere can reduce the complicity of a design, namely using fewer elements, making the total track shorter and smaller, and sometimes even cheaper.

In this chapter, some commonly used aspheric surfaces are reviewed. And two novel types of aspheres are proposed.

7.1 Review of the Existing Aspheric Surfaces

Basically, two types of aspheric surfaces are commonly used in lens design. The first type is the conic surface. A conic constant k is added to the base sphere equation (7-1).

$$z = \frac{Cr^2}{1+\left(1-(1+k)C^2r^2\right)^{1/2}} \qquad (7\text{-}1)$$

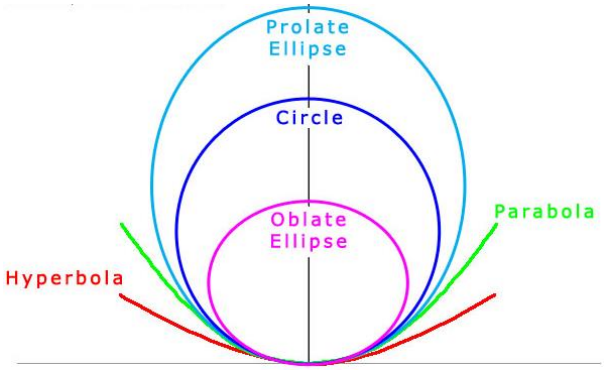


Figure 7-1 Conic surfaces with the same base radius of curvature

When the conic constant *k* has different values, the surface can have different shape. The commonly used conic surfaces are plotted together with the same base radius of curvature in Figure 7-1. The conic surface has two conjugate foci. Light from one focus reflects off the surface and comes back to the other focus without any aberrations. Therefore, most astronomical telescope mirrors use porabolas and hyperbolas because this unique property.

Biconic surface [19] is the conic surface that allow different base radius and conic constant in X and Y directions. One common application of biconic surface is the model of the human eye.

The second type of surface adds polynomials on the conic surface. Rotationally symmetric polynomial aspheric surface is a combinition of a base sphere (or conic) and a polynomial expansion. Different types of polynomials were invented.

Zernike polynomials [20] are well-known in the field of intereferometic testing. It uses a set of orthogonal polynomials that represent diferent aberrations to fit to the wavefront error. Zernike polynomials could also be used to define the aspheric departure from a

base sphere. Also, Zernike polynomials could be added on biconic surface to make it the biconic Zernike surface. For example, biconic Zernike surface is used in ophthalmology describes aberrations of the cornea.

Even asphere surface is the most commonly used aspheric surface in lens design. It uses the even powers of the radial coordinate to describe the asphericity. The even asphere's sag equation is described as:

$$z = \frac{Cr^2}{1+\sqrt{(1-(1+k)C^2r^2)}} + A_4r^4 + A_6r^6 + A_8r^8 \ldots \qquad (7\text{-}2)$$

Note the even aspheric coefficients have units and they are not orthgonal. The sign of the different coefficents are usually alternating in order to give the aspheric departure from the base sphere.

Q type polynomial is invented by Greg Forbes in recent years[21-22]. Q type asphere uses a set of orthogonal polynomials represent rotation symmetric asphere. Two specific types are defined. (a) strong aspheres departing from a conic, and (b) mild aspheres departing from a best-fit sphere. The superiority of the Q type asphere is facilataing the manufaturing constraning and design tolerancing. Because each polynomial term has a meaning in term of surface error. By now, this new type asphere is already included in most conmmerial optical design softwares. Also, it graduately be adopted in optical specification, fabrication, and testing in optics industrial.

7.2 Novel Aspheric Surfaces

Polynomial type surface is powerful to define aspheres with a wide range of aspheric departure. However, when the aspheric orders used are too high, the fabrication and testing become very challenging. Therefore, the idea here is to see if a non-polynomial type asphere can be invented and applied to lens design. In this chapter, two novel non-polynomial type surfaces which define by a few first order parameters are introduced. Furthermore, the application of these novel aspheric surfaces is studied.

The idea came from the insight on the cellular phone designs demonstrated in Chapter 5. The three design examples are shown in Figure 7-2.

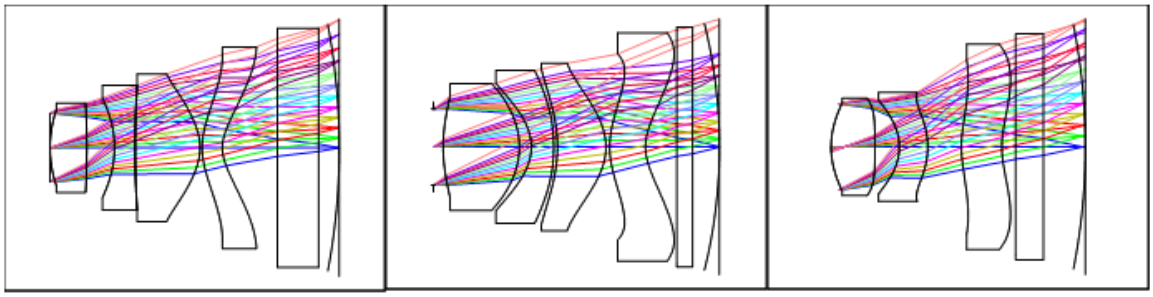


Figure 7-2 Three cellular phone lens examples

The aspheric flattener lens in front of the flat window is mainly used to reduce field curvature. All these aspheric flatteners have very little optical power because the front and back surface usually has similar base radius of curvature. But the flatteners have wave shape due to the aspheric terms. The exactly shape depends on the order of aspheric terms used. Based on the "wavy" shape observed from the cellular phone field flattener lens, a novel surface that has two radii of curvature but different signs is invented. The slope at the turning point is continuous. In this way, the new surface could have the "wavy" shape as the field flattener in cellular phone lens.

7.2.1 Type 1 Basic Dual Radii Surface

The mathematical definition of the basic dual radii surface is described as

$$R(r) = \begin{cases} infinity & (r = 0) \\ R_1 & (0 < r \le r_1) \\ R_2 & (r_1 < r \le r_{max}) \end{cases} \qquad (7\text{-}3)$$

The equation (7-3) defines the radii of curvature *R(r)* at different portions of the pupil *r*.

The surface drawing is shown in Figure 7-3. On optical axis, the surface slope is infinity. When the pupil size increases from 0 to r_1, the surface radius of curvature is R_1. R_1 could be positive or negative. When the pupil size increases from r_1 to the edge of the surface r_{max}, the surface radius of curvature is R_2 which has an opposite sign to R_1. At $r = r_1$, there is a turning point connects the two parts of the surface. The slope at the turning points is continuous. Two center of curvatures C.C1 and C.C2 are alone the surface normal.

There are 3 variables in this type of surface, namely R_1, R_2 and r_1. The surface is rotationally symmetric to the optical axis. The base radius of curvature is R_1. The difference between this surface and the normal spherical surface is the R_2 part when r is larger than r_1.

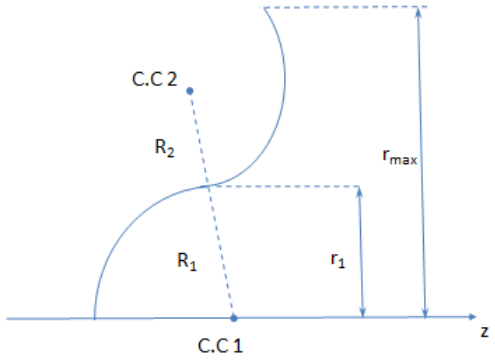


Figure 7-3 Type 1 surface layout

Type 1 surface could be written in optical design software. The surface layout in design software is shown in Figure 7-4. The surface data are in Table 7-1.

R_1	50 mm
R_2	25 mm
r_1	5 mm
r_{max}	12.5 mm

Table 7-1 Surface specifications

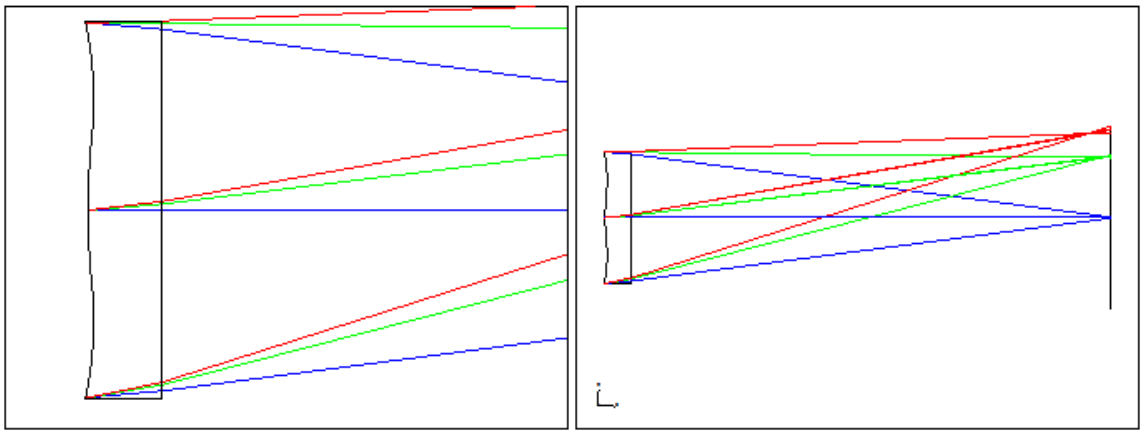


Figure 7-4 Type 1 surface simulation in design software

To compare Type 1 surface and even asphere, an example is given below to see how the two types of surface fit to each other. Figure 7-5 and 7-6 show the fitting results. Two even asphere coefficients A_4 and A_6 are used. The P-V surface sag difference is about 120 μm over a 1 inch aperture. Using more aspheric terms does not help minimize the sag difference. The red color surface is the even asphere.

	Surf:Type	Thickness		Glass		Semi-Diameter	Conic		Par 0(unused)		2nd Order Term	4th Order Term		6th Order Term		8th Order
OBJ	Standard	Infinity				0.000000000	0.000000000									
STO	Dual_radius	0.000000000		1.00,0.0		12.70000000	0.000000000				5.000000000	25.00000000				
2	Even Asph..	5.000000000				12.70000000	0.000000000				0.000000000	-7.352E-005	V	2.0778E-009	V	0.00000
3	Paraxial	50.00000000	M			12.70000000					50.00000000	1				
IMA	Standard	-				0.000000000	0.000000000									

Figure 7-5 Lens prescription data

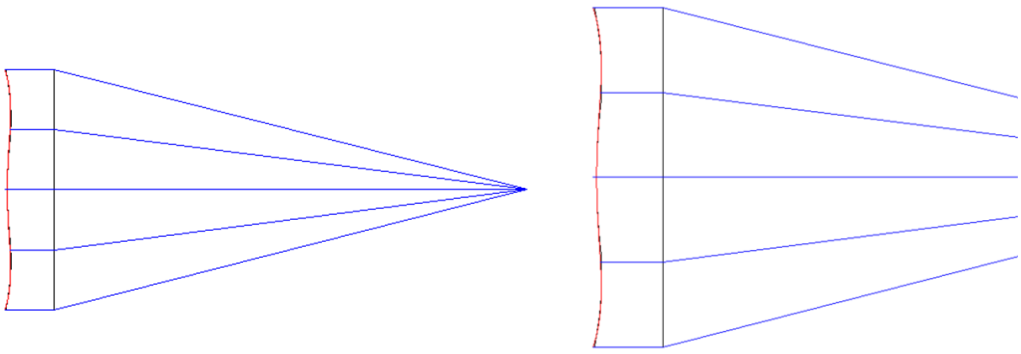


Figure 7-6 Fitting Type 1 surface to even asphere has coefficients A_4 and A_6

For a lens has one inch aperture size, the novel Type 1 surface with three first-order parameters can fit to the even asphere with two high order coefficients with very small fitting residue. As discussed in Chapter4, an even asphere near image plane with two high order terms could significantly reduce field curvature. Therefore, the novel Type 1 surface could be used as an aspheric field flattener.

7.2.2 Type 2 Powerless Dual Radii Surface

The Type 1 dual radii surface has a base radius of curvature R_1. This Type 2 powerless dual radii surface adds a flat piston part on-axis to make the lens powerless. The mathematical definition of this Type 2 surface is described as,

$$R(r) = \begin{cases} infinity & (0 \le r \le r_1) \\ R_1 & (r_1 < r \le r_2) \\ R_2 & (r_2 < r \le r_{max}) \end{cases} \qquad (7\text{-}4)$$

The surface drawing is shown in Figure 7-7. When r is between 0 and r_1, the surface is flat. When r is larger than r_1, the surface shape is exactly the same as Type 1 surface. As this Type 2 surface adds a flat piston part, the surface power is zero on-axis.

There are four variables consist of this type of surface, namely R_1, R_2, r_1 and r_2. The surface is rotationally symmetric to the optical axis.

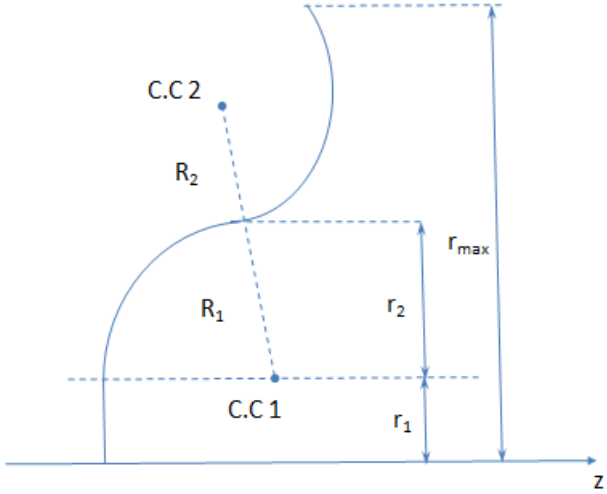


Figure 7-7 Type 2 surface layout

An example of Type 2 surface is demonstrated in design software. Figure 7-8 gives the layout of the Type 2 surface. Table 7-2 gives the surface data.

R_1	25 mm
r_1	3 mm
R_2	5 mm
r_2	8 mm
r_{max}	12.5 mm

Table 7-2 Surface specifications

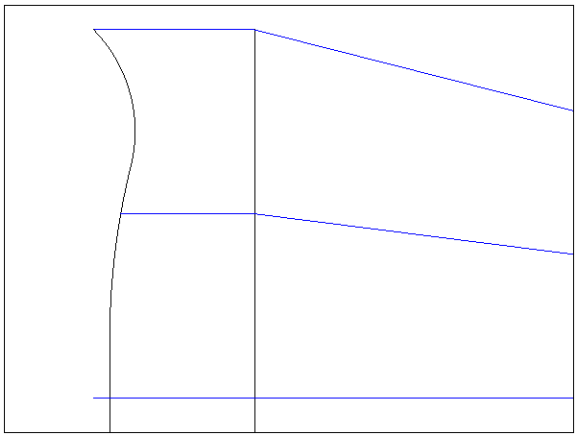


Figure 7-8 Type 2 surface simulation in design software

Next, the even asphere is used to fit to this Type 2 surface example. The fitting figures and results are shown in Figure 7-9 and 7-10. The red color surface is the even asphere.

Surf:Type		Glass	Semi-Diameter	Conic	Par 0(unused)	2nd Order Term	4th Order Term	6th Order Term	8th Order Term
OBJ	Standard		0.000000000	0.000000000					
STO	flat+Dual..	1.00,0.0	12.70000000	0.000000000		3.000000000	25.00000000	8.000000000	5.000000000
2	Even Asph..		12.70000000	0.000000000		0.000000000	1.9914E-004 V	-1.507E-006 V	0.000000000
3	Paraxial		12.70000000			50.00000000	1		
IMA	Standard		0.000000000	0.000000000					

Figure 7-9 Lens prescription data

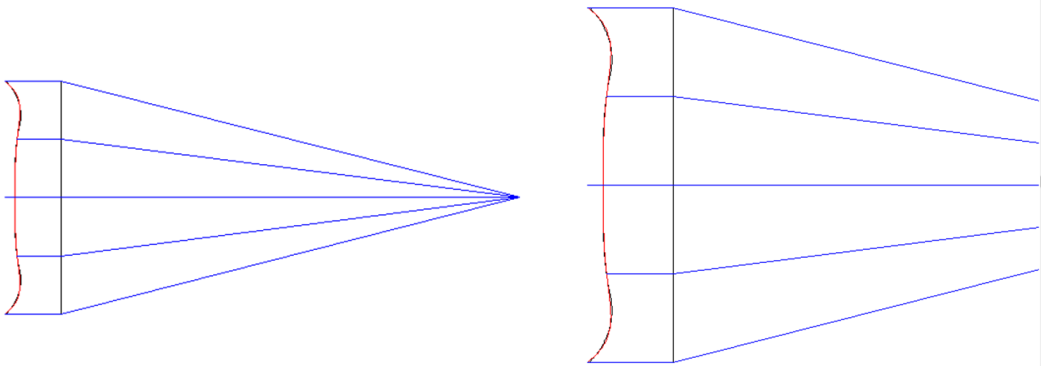


Figure 7-10 Fitting Type 2 surface to even asphere has coefficients A_4 and A_6

Two even asphere coefficients A_4 and A_6 are used. The P-V surface sag difference is about 180 μm over one inch aperture. Using more aspheric terms does not minimize the sag difference.

With a powerless flat portion added on Type 1, the Type 2 surface could still replace an even asphere with two high order aspheric terms. This Type 2 surface could be used as a powerless field flattener.

7.3 Application of novel aspheric surfaces

In Chapter 4, we showed an aspheric field flattener near the image with two aspheric terms could significantly reduce field curvature. In this chapter, two novel types of asphere are proposed and demonstrate that they could fit to the even asphere with two coefficients. Thus, there is a potential to use the Type 1 or Type 2 surface replace the traditional even asphere which serves as an aspheric field flattener.

In Chapter 4, we demonstrated how an aspheric field flattener successfully reduces field curvature in the Petzval Lens. Here, an example is given to demonstrate how the novel Type 2 powerless dual radii asphere replaces the even asphere field flattener.

The Petzval Lens design example is recalled in Figure 7-11. A pure 4^{th} order Petzval field curvature is generated when the negative spherical field flattener is set to flat.

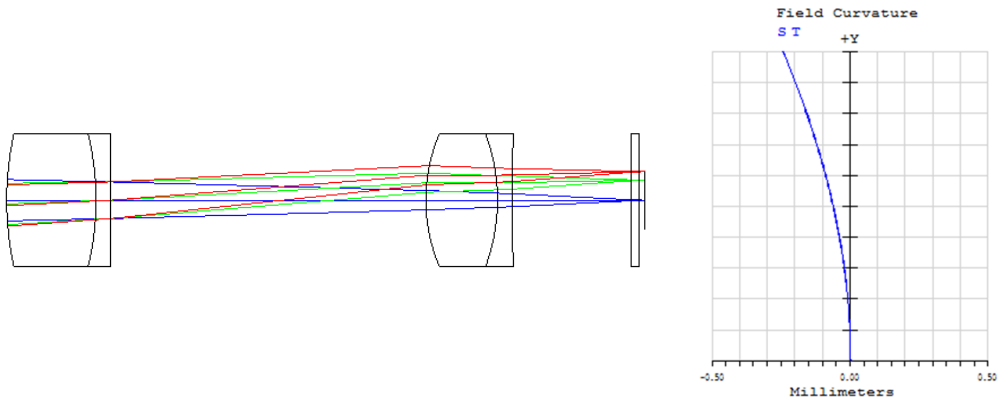


Figure 7-11 Petzval lens example and its field curvature plot

Then, the back surface of the flat window is set as novel Type 2 asphere and allowed r_1, R_1, r_2, and R_2 vary. The optimization results are shown in Figure 7-12 and Table 7-3.

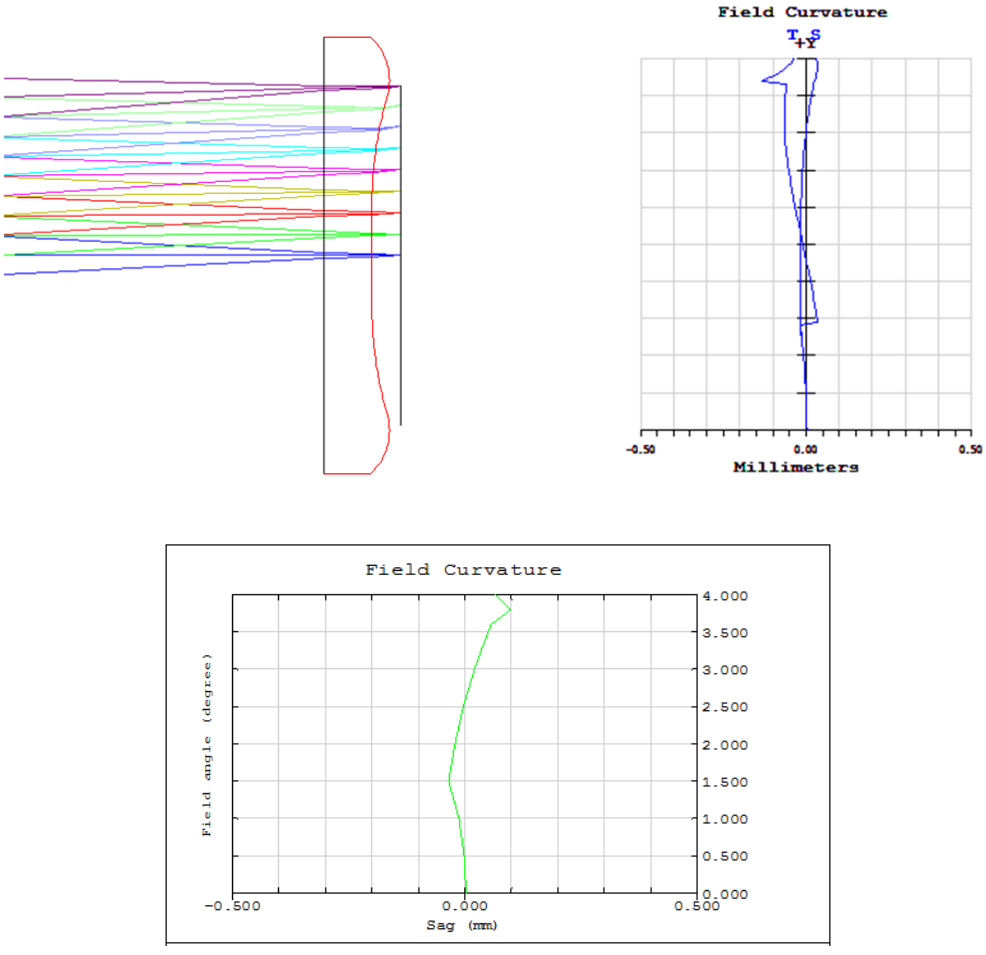


Figure 7-12 Petzval lens with a novel aspheric (Type 2) flattener lens and its field curvature and generalized Petzval curvature plots.

Type 2 Asphere Specifications	
R_1	16.37 mm
r_1	2.02 mm
R_2	2.51 mm
r_2	6.51 mm
r_{max}	9.0 mm

Table 7-3 Surface specifications

The optimization results of using even asphere with coefficients A_4 and A_6 are recalled and shown in Figure 7-13 and Table 7-4.

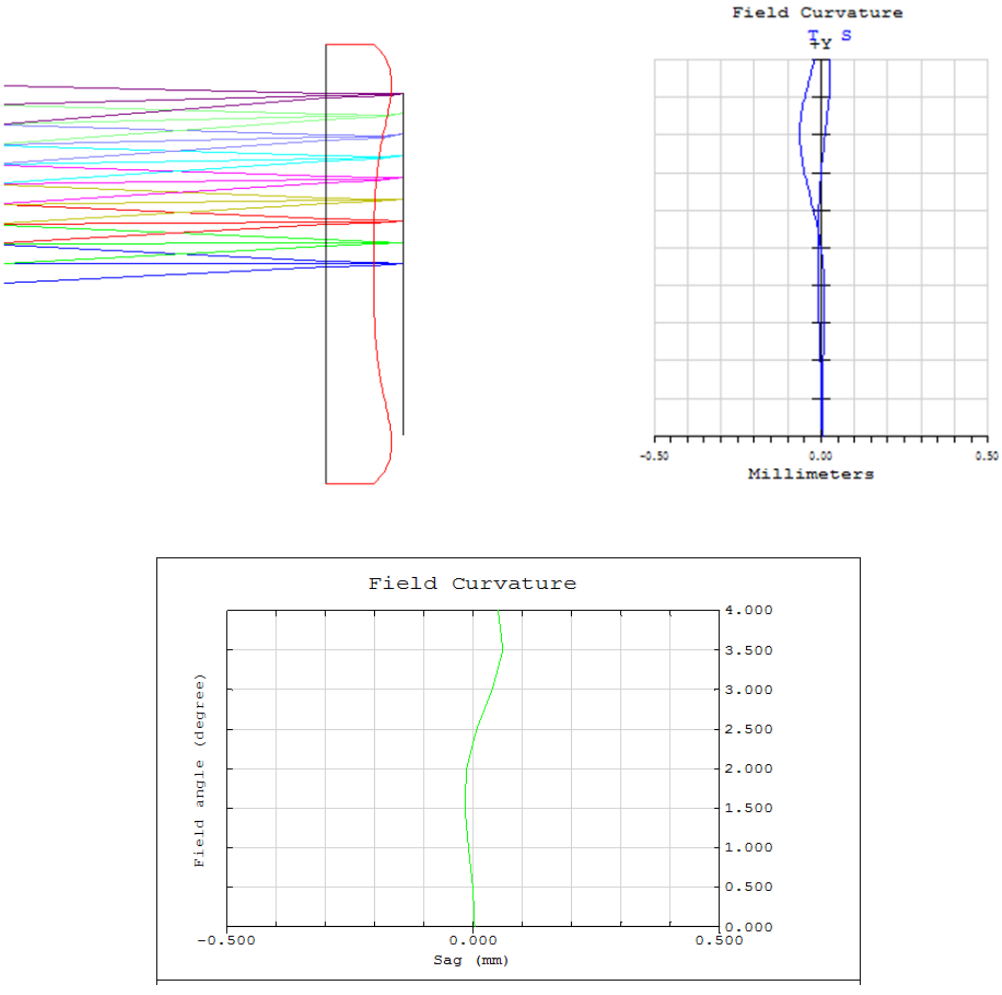


Figure 7-13 Petzval lens with an aspheric (even asphere) flattener lens and its field curvature and generalized Petzval curvature plots

Even Asphere Specifications	
A_4	$7.4078e^{-4}$
A_6	$-9.1810e^{-6}$

Table 7-4 Surface specifications

Next, the two types of surface are plotted together and shown in Figure 7-14. The red color surface is Type 2 asphere. The P-V difference is 67 μm.

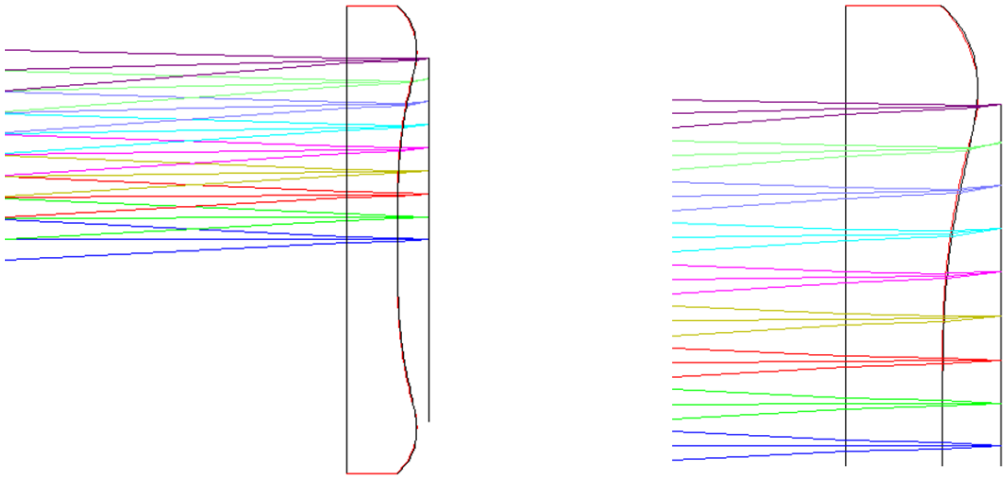


Figure 7-14 Fitting Type 1 asphere to Even Asphere

By comparison, both surfaces could significantly reduce field curvature. The field curve generated by Type 2 asphere is not as smooth as field curvature generated by even asphere. But the amount of field curvature reduction and the shape of the curve are similar. Also, the sag difference between the two types of asphere is small.

In summary, the two types of novel aspheric surfaces are proposed. With 3~4 first order parameters, these novel aspheres agree with the even asphere with two high order terms.

A design example is given to show both even asphere and Type 2 novel asphere can serve as a field flattener near the image.

CHAPTER 8

SUMMARY & FUTURE WORK

8.1 Summary

This dissertation gives an in-depth study of one of the hardest-to-fix aberrations in lens design, namely field curvature.

In Chapter 1, an introduction is given to field curvature aberration and its effect to image quality. The classical Petzval theorem is introduced.

In Chapter 2, the classical methods to control field curvature are discussed. Thick meniscus lens, separating positive and negative elements, using field flattener, and using new achromat glasses are the four usual ways to reduce field curvature. Also, some miscellaneous methods such as gradient index glass and curved detector in some special applications are discussed as well.

In Chapter 3, high order field curvature theory is developed to elaborate the balancing between 4^{th} order Petzval field curvature and higher order field curvature. Higher order field curvature coefficients were given when high order aspheric terms are used. Examples are given to show the field curvature balancing mechanism between different orders.

In Chapter 4, software simulation is given to demonstrate how aspheric surfaces help correct field curvature. The best location for the aspheric field flattener, the number of aspheric surfaces, and the number of aspheric terms are studied. The simulation results

indicate one aspheric surface close to the image with two aspheric terms could reduce field curvature more than 85% which agrees to the theory developed in Chapter 3. Examples are given to show how aspheric flattener reduces field curvature in real designs.

In Chapter 5, astigmatism aberration is discussed. The 4^{th} order and higher order astigmatism can be induced to balance Petzval curvature. Design examples are analyzed and shown the balancing between astigmatisms and Petzval field curvature.

In Chapter 6, a new 6^{th} order aberration - oblique spherical aberration is introduced. Oblique spherical aberration can be used to balance Petzval field curvature. Simulation results indicate the reduction is more than 70% in terms of OPD error. Two Double Gauss type of lenses demonstrate how oblique spherical aberration is induced and used to balance Petzval field curvature.

In Chapter 7, classical types of aspheric surface are reviewed and two novel aspheric surfaces are proposed. Without using polynomials, only three to four first order parameters are used to define the novel aspheric surfaces which can be used to correct field curvature, as efficient as an even asphere.

In conclusion, this dissertation gives three alternate methods to correct field curvature for the systems that have large Petzval sum. Instead of minimizing the Petzval sum, the three methods use high order aberration to balance the Petzval curvature. The three methods are using (1) high order field curvature; (2) 4^{th} order and higher order astigmatism; (3) oblique spherical aberration. Method (1) One aspheric surface with two aspheric terms near the image can induce high order field curvature which balances the Petzval curvature. The reduction of field curvature, in term of sag, is more than 85%. Method (2)

An aspheric surface that is farther from the image could induce high order astigmatism which helps control 4^{th} order astigmatism. And then the 4^{th} order astigmatism is used to compensate the Petzval curvature. Method (3) Oblique spherical aberration could be used to correct the Petzval curvature. When the two aberration coefficients have the same quantity but opposite sign, the OPD reduction is more than 70%. Many real design examples are analyzed and given to illustrate each method. The design examples include modern cellular phone lenses, wide angle lenses, photographic lenses, and zoom lenses, etc. All the data agree to the theories and the simulations. Two non-polynomial types of novel aspheric surfaces are proposed. They could be used as the aspheric flattener to control field curvature. The simulation indicates they are as effective as the even asphere flattener with two aspheric terms.

8.2 Future Work

An aspheric surface is widely used in modern lens design. In this dissertation, we demonstrate high order aspheric coefficients can introduce high order aberrations which balance the Petzval field curvature. However, we were aware of that the high order asphericities also induce other high order aberrations, such as high order astigmatism and high order distortion which degrade the image quality. Therefore, it is worthy to explore the theories of aspheric contribution to high order aberrations and how to use asphericity to control other high order aberration besides field curvature.

APPENDIX

Zeamx Macro used to compute ACTS in Chapter 4

```
! This macro computes the aspheric contribution to optical ray path length across pupil weighted by the index factor,
specifically ACTS (Aspheric Contribution to Thickness  Sum )


Savelens "current.zmx"


nfield = NFLD() # number of field

maxfield = MAXF() # maximum field angle

n = NSUR() # number of surfaces


DECLARE psx, DOUBLE, 1, nfield

DECLARE psy, DOUBLE, 1, nfield

DECLARE ppx, DOUBLE, 1, nfield

DECLARE ppy, DOUBLE, 1, nfield

DECLARE pox, DOUBLE, 1, nfield

DECLARE poy, DOUBLE, 1, nfield


FOR i, 1 , nfield, 1


psx(i)=FLDY(i)

ppx(i)=FLDY(i)

pox(i)=FLDY(i)


hy = FLDY(i)/maxfield


FORMAT 1.0

PRINT "Field number ", i

FORMAT 1.1

PRINT "  Y-field angle : ", FLDY(i)

RAYTRACE 0,hy,0,0
```

```
Q=0

P=0


 FOR j,1,n-1,1

 FORMAT 1.0

!PRINT "   surface number ", j


IF (INDX(j)==1)

FORMAT 1.5

!PRINT "Petzval power is ", 0


ELSE

  PetzvalPower= (INDX(j)-1)*ABSO(RAYT(j+1))/INDX(j)


Q=Q+PetzvalPower


Petzvaloblique= (INDX(j)*COSI(ATAN((RAGY(j+1)-RAGY(j))/(RAGZ(j+1)-RAGZ(j))))-INDX(j-
1)*COSI(ATAN((RAGY(j)-RAGY(j-1))/(RAGZ(j)-RAGZ(j-1)))))*(RAYT(j+1)-THIC(j))/INDX(j)/INDX(j-1)

P=P+Petzvaloblique


 FORMAT 1.5


ENDIF


 NEXT

 FORMAT 1.5

PRINT "      Petzval power is ", Q

PRINT "      Oblique power is ", P


psy(i)=SAGG(RAYX(n-1),RAYY(n-1),n-1)

ppy(i)=Q
```

```
poy(i)=ABSO(P)

!PRINT "field points ", ppx(i)
!PRINT "Petzval power ", ppy(i)

NEXT

title$ = "Plot Sag vs Field"                    # Plot Title
xtitle$ = " Field Angle (degree) "      # X-axis label
ytitle$ = " Sag (mm) "
x_min = 0.0                                     # X-axis minimum value
!y_min = -2.0                                   # Y-axis minimum value
x_max = maxfield                                # X-axis maximum value
!y_max = 0.0

comment1$ = "       Blue curve is the Petzval surface"
comment2$ = "       Red curve is the Thickness variation sum"

!GRAPHICS
PLOT NEW
PLOT TITLE, title$
PLOT TITLEX, xtitle$
PLOT TITLEY, ytitle$
PLOT COMM1, comment1$
PLOT COMM2, comment2$
PLOT RANGEX, x_min, x_max
!PLOT RANGEY, miny, maxy
!PLOT CHECK, x_increment, y_increment
!PLOT TICK, x_increment, y_increment
!PLOT FORMATX, format_string
!PLOT FORMATY, format_string
!PLOT LINE, x1, y1, x2, y2
```

```
!PLOT LABEL, x, y, angle, size, string
!PLOT DATA, psx, psy, nfield, 1 ,1, 1
!PLOT DATA, ppx, ppy, nfield, 3 ,1, 1
!PLOT DATA, pox, poy, nfield, 2 ,1, 1
!PLOT GO
!PRINT "All Done!"

! Make ALL aspheric coefficients zero

for i= 1, NSUR(), 1

SURP i, 10, 0, 1
SURP i, 10, 0, 2
SURP i, 10, 0, 3
SURP i, 10, 0, 4
SURP i, 10, 0, 5
SURP i, 10, 0, 6
SURP i, 10, 0, 7
SURP i, 10, 0, 8
update

Next

! Re-run the same calculation as below for the lens with no aspheric coefficiences

DECLARE ppyn, DOUBLE, 1, nfield
DECLARE diff, DOUBLE, 1, nfield

FOR i, 1 , nfield, 1

psx(i)=FLDY(i)
ppx(i)=FLDY(i)
```

```
pox(i)=FLDY(i)

hy = FLDY(i)/maxfield

FORMAT 1.0
PRINT "Field number ", i
FORMAT 1.1
PRINT "  Y-field angle : ", FLDY(i)

RAYTRACE 0,hy,0,0

AQ=0
AP=0

 FOR j,1,n-1,1
 FORMAT 1.0
!PRINT "   surface number ", j

IF (INDX(j)==1)
FORMAT 1.5
!PRINT "Petzval power is ", 0

ELSE
   PetzvalPower= (INDX(j)-1)*ABSO(RAYT(j+1))/INDX(j)
AQ=AQ+PetzvalPower

Petzvaloblique= (INDX(j)*COSI(ATAN((RAGY(j+1)-RAGY(j))/(RAGZ(j+1)-RAGZ(j))))-INDX(j-
1)*COSI(ATAN((RAGY(j)-RAGY(j-1))/(RAGZ(j)-RAGZ(j-1)))))*(RAYT(j+1)-THIC(j))/INDX(j)/INDX(j-1)
AP=AP+Petzvaloblique

 FORMAT 1.5
 !PRINT "index is ", INDX(j)
```

```
 !PRINT "ray path length to current surface is ", RAYT(j)
 !PRINT "ray path length to next surface is ", RAYT(j+1)
 !PRINT "ray path length difference is ", RAYT(j+1)-RAYT(j)
 !PRINT "thickness is ", THIC(j)
!PRINT "Petzval power is ", PetzvalPower

ENDIF

 NEXT
 FORMAT 1.5
PRINT "      Petzval power is ", AQ
PRINT "      Oblique power is ", AP

!psy(i)=ABSO(SAGG(RAYX(n-1),RAYY(n-1),n-1))
ppyn(i)=AQ
!poy(i)=ABSO(AP)

!diff(i)=ABSO(ppy(i)-ppyn(i))
diff(i)=ppy(i)-ppyn(i)

!PRINT "field points ", ppx(i)
!PRINT "Petzval power ", ppy(i)

NEXT

title$ = "Plot Sag vs Field"                    # Plot Title
xtitle$ = " Field Angle (degree) "              # X-axis label
ytitle$ = " Sag (mm) "
x_min = 0.0                                     # X-axis minimum value
!y_min = -2.0                                   # Y-axis minimum value
x_max = maxfield                                # X-axis maximum value
!y_max = 0.0
```

```
comment1$ = "          Blue curve is the Petzval surface"
comment2$ = "          Red curve is the difference with and without asphere"

!GRAPHICS
PLOT NEW
PLOT TITLE, title$
PLOT TITLEX, xtitle$
PLOT TITLEY, ytitle$
PLOT COMM1, comment1$
PLOT COMM2, comment2$
PLOT RANGEX, x_min, x_max
!PLOT RANGEY, miny, maxy
!PLOT CHECK, x_increment, y_increment
!PLOT TICK, x_increment, y_increment
!PLOT FORMATX, format_string
!PLOT FORMATY, format_string
!PLOT LINE, x1, y1, x2, y2
!PLOT LABEL, x, y, angle, size, string
PLOT DATA, psx, psy, nfield, 1 ,1, 1
PLOT DATA, ppx, diff, nfield, 3 ,1, 1
!PLOT DATA, pox, poy, nfield, 2 ,1, 1
PLOT GO
PRINT "All Done!
Loadlens current.zmx
```

User Defined Surface code for TYPE 1 surface in Chapter 7

```
#include <windows.h>
#include <math.h>
#include <string.h>
#include <stdio.h>
#include "usersurf.h"


int __declspec(dllexport) APIENTRY UserDefinedSurface(USER_DATA *UD, FIXED_DATA *FD);

/* a generic Snells law refraction routine */
int Refract(double thisn, double nextn, double *l, double *m, double *n, double ln, double
mn, double nn);

BOOL WINAPI DllMain (HANDLE hInst, ULONG ul_reason_for_call, LPVOID lpReserved)
        {
   return TRUE;
   }

/*

This DLL models a novel aspheric surface described in:

"Advanced Theory of Field Curvature"
By Yuhao Wang
Ph.D. dissertation Chapter 7: TYPE 1 Basic Dual Radii Surface

This surface is a non-polynomial type asphere.

*/

int __declspec(dllexport) APIENTRY UserDefinedSurface(USER_DATA *UD, FIXED_DATA *FD)
        {
   int i, loop, pmax;
   double P1,R2, r, rs,R1, tp, alpha, dr, power, rad, t, x, y, z, dz, sag, mm;

   switch(FD->type)
        {
      case 0:
       /* ZEMAX is requesting general information about the surface */
         switch(FD->numb)
               {
            case 0:
               /* ZEMAX wants to know the name of the surface */
                        /* do not exceed 12 characters */
                        strcpy(UD->string,"Dual_radius");
               break;
            case 1:
               /* ZEMAX wants to know if this surface is rotationally symmetric */
```

```
            /* it is, so return any character in the string; otherwise, return a null
string */
            strcpy(UD->string, "1");
            break;
         case 2:
            /* ZEMAX wants to know if this surface is a gradient index media */
            /* it is not, so return a null string */
            UD->string[0] = '\0';
            break;
         }
       break;
     case 1:
      /* ZEMAX is requesting the names of the parameter columns */
       /* the value FD->numb will indicate which value ZEMAX wants. */
       /* Only "q" in parameter 1 is used for this surface type */
       /* returning a null string indicates that the parameter is unused. */
       switch(FD->numb)
             {
         case 1:
            strcpy(UD->string, "P1");
            break;
                  case 2:
            strcpy(UD->string, "R2");
            break;
         default:
            UD->string[0] = '\0';
            break;
         }
      break;
     case 2:
      /* ZEMAX is requesting the names of the extra data columns */
       /* the value FD->numb will indicate which value ZEMAX wants. */
       /* returning a null string indicates that the extradata value is unused. */
       switch(FD->numb)
             {
        case 1:
            strcpy(UD->string, "# Terms");
            break;
          default:
            if (FD->numb <= FD->xdata[1] + 1)
            {
               sprintf(UD->string, "Term %i", FD->numb - 1);
               }
            else
            {
                  UD->string[0] = '\0';
               }
            break;
         }
      break;
     case 3:
      /* ZEMAX wants to know the sag of the surface */
```

```
        UD->sag1 = 0.0;
        UD->sag2 = 0.0;
        P1 = FD->param[1];
               R2 = FD->param[2];
               if (R2 < 0) return(-1);

        rs = UD->x * UD->x + UD->y * UD->y;
        r = sqrt(rs);
               R1=1/(FD->cv);

        alpha = 1 - (1+FD->k)*FD->cv*FD->cv*rs;
        if (alpha < 0) return(-1);
 if (R1>0)
       {       if(abs(r)<=P1)
         { UD->sag1 = (FD->cv*rs)/(1 + sqrt(alpha)); UD->sag2 = UD->sag1;}

               else
                      { if(r>P1)
                       {UD->sag1 = R1 - ((R1+R2)*sqrt(R1*R1-P1*P1))/R1 + sqrt(R2*R2-(r-
(R1+R2)*P1/R1)*(r-(R1+R2)*P1/R1)); UD->sag2 = UD->sag1;}

               else
                       {UD->sag1 = R1 - ((R1+R2)*sqrt(R1*R1-P1*P1))/R1 + sqrt(R2*R2-
(r+(R1+R2)*P1/R1)*(r+(R1+R2)*P1/R1)); UD->sag2 = UD->sag1;}
               }
   }

 else
       {       if(abs(r)<=P1)
         { UD->sag1 = -(((-FD->cv)*rs)/(1 + sqrt(1 - (1+FD->k)*(-FD->cv)*(-FD->cv)*rs)));
UD->sag2 = UD->sag1;}

               else
                      { if(r>P1)
                       {UD->sag1 = -((-R1) - (((-R1)+R2)*sqrt((-R1)*(-R1)-P1*P1))/(-R1) +
sqrt(R2*R2-(r-((-R1)+R2)*P1/(-R1))*(r-((-R1)+R2)*P1/(-R1)))); UD->sag2 = UD->sag1;}

               else
                       {UD->sag1 = -((-R1) - (((-R1)+R2)*sqrt((-R1)*(-R1)-P1*P1))/(-R1) +
sqrt(R2*R2-(r+((-R1)+R2)*P1/(-R1))*(r+((-R1)+R2)*P1/(-R1)))); UD->sag2 = UD->sag1;}
               }
   }


   break;

      case 4:
       /* ZEMAX wants a paraxial ray trace to this surface */
        /* x, y, z, and the optical path are unaffected, at least for this surface type
*/
        /* for paraxial ray tracing, the return z coordinate should always be zero. */
```

```
   /* paraxial surfaces are always planes with the following normals */
   /* we will ignore the aspheric terms, even the quadratic one, since it has a */
   /* meaning that is hard to interpret if q != 0.0 */

   UD->ln =  0.0;
   UD->mn =  0.0;
   UD->nn = -1.0;
   power = (FD->n2 - FD->n1)*FD->cv;
   if ((UD->n) != 0.0)
         {
      (UD->l) = (UD->l)/(UD->n);
      (UD->m) = (UD->m)/(UD->n);

      (UD->l) = (FD->n1*(UD->l) - (UD->x)*power)/(FD->n2);
      (UD->m) = (FD->n1*(UD->m) - (UD->y)*power)/(FD->n2);

      /* normalize */
      (UD->n) = sqrt(1/(1 + (UD->l)*(UD->l) + (UD->m)*(UD->m) ) );
      /* de-paraxialize */
      (UD->l) = (UD->l)*(UD->n);
      (UD->m) = (UD->m)*(UD->n);
      }
   break;
case 5:
  /* ZEMAX wants a real ray trace to this surface */
   /* okay, not a plane. */
   /* do not allow n == 0 */
   if (UD->n == 0.0) return -1;

   /* Now, we illustrate an iterative method of finding
      the intercept for a general surface. */

                /* make sure we do at least 1 loop */

                t = 100.0;
   tp = 0.0;
   x = UD->x;
   y = UD->y;
   z = UD->z;
   loop = 0;
   P1 = FD->param[1];
          R2 = FD->param[2];
 /*  pmax = FD->xdata[1];*/

   while (fabs(t) > 1e-10)
         {
                      /*
      First, compute the sag using whatever the surface sag expression is.
      This is given the x and y starting points. The following block of code
      will change depending upon the surface shape, the rest of this iteration
      is typically common to all surface shapes.
      */
```

```
                rs = x * x + y * y;
            r = sqrt(rs);

                alpha = 1.0 - (1.0 + FD->k)*FD->cv*FD->cv*rs;

                  if (alpha < 0.0) return(-1);

                      R1=1/(FD->cv);
if (R1>0)
    {     { if(abs(r)<=P1)
                      sag = (FD->cv*rs)/(1 + sqrt(alpha));

               else
                     { if(r>P1)
                       sag = R1 - ((R1+R2)*sqrt(R1*R1-P1*P1))/R1 + sqrt(R2*R2-(r-
(R1+R2)*P1/R1)*(r-(R1+R2)*P1/R1));

                    else
                       sag = R1 - ((R1+R2)*sqrt(R1*R1-P1*P1))/R1 + sqrt(R2*R2-
(r+(R1+R2)*P1/R1)*(r+(R1+R2)*P1/R1));
                    }
                }
    }
else
    {     { if(abs(r)<=P1)
                      sag = -(((-FD->cv)*rs)/(1 + sqrt(1.0 - (1.0 + FD->k)*(-FD->cv)*(-
FD->cv)*rs)));

               else
                     { if(r>P1)
                       sag = -((-R1) - (((-R1)+R2)*sqrt((-R1)*(-R1)-P1*P1))/(-R1) +
sqrt(R2*R2-(r-((-R1)+R2)*P1/(-R1))*(r-((-R1)+R2)*P1/(-R1))));

                    else
                       sag = -((-R1) - (((-R1)+R2)*sqrt((-R1)*(-R1)-P1*P1))/(-R1) +
sqrt(R2*R2-(r+((-R1)+R2)*P1/(-R1))*(r+((-R1)+R2)*P1/(-R1))));
                    }
                }
    }
            /* okay, now with sag in hand, how far are we away in z? */
                      // modified 11/2012 to support larger angles of incidence
                      dz = (sag - z)*fabs(UD->n);

            /* now compute how far along the z axis this is */
            /* note this will crash if n == 0!! */
            t = dz / (UD->n);

            /* propagate the additional "t" distance */
            x += UD->l*t;
            y += UD->m*t;
            z += UD->n*t;
```

```
            /* add in the optical path */
                              tp += t;

            /* prevent infinte loop if no convergence */
            loop++;
            if (loop > 1000) return(-1);
            }

          /* okay, we should be a the intercept coordinates now */
          UD->x = x;
          UD->y = y;
          UD->z = z;

         /* don't forget the path! */
                          UD->path = tp;

          rs = x * x + y * y;
          r = sqrt(rs);
          P1 = FD->param[1];
                 R2 = FD->param[2];
                 R1=1/(FD->cv);
          alpha = 1.0 - (1.0+FD->k)*FD->cv*FD->cv*rs;
          if (alpha < 0) return(-1); /* ray misses */
          alpha = sqrt(alpha);
                 mm = 0.0;

                          /* now do the normals */
if (R1>0)
   {   if (r==0)
                 {
            UD->ln =  0;
            UD->mn =  0;
            UD->nn = -1;
            }
          else
                { if(0<abs(r)<P1)
                            {mm += (FD->cv*r/(1.0+alpha))*(2.0 + (FD->cv*FD->cv*rs*(1.0+FD-
>k))/(alpha*(1.0+alpha)));
                    rad = 1.0 + (mm*mm);
                rad = mm / sqrt(rad);
                UD->ln = (x/r)*rad;
                UD->mn = (y/r)*rad;
                UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
                if (UD->nn >= 1.0) return(-1);
                           else UD->nn = -sqrt(1.0 - UD->nn);}

                    else { if(r>P1)
                             { mm += ((R1+R2)*P1/R1-r)/(sqrt(R2*R2-(r-(R1+R2)*P1/R1)*(r-
(R1+R2)*P1/R1)));
                                    rad = 1.0 + (mm*mm);
                        rad = mm / sqrt(rad);
```

```
                        UD->ln = (x/r)*rad;
                        UD->mn = (y/r)*rad;
                        UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
                        if (UD->nn >= 1.0) return(-1);
                                  else UD->nn = -sqrt(1.0 - UD->nn);}

                                    else
                                        { mm += (-(R1+R2)*P1/R1-r)/(sqrt(R2*R2-
(r+(R1+R2)*P1/R1)*(r+(R1+R2)*P1/R1)));
                                                        rad = 1.0 + (mm*mm);
                                rad = mm / sqrt(rad);
                                UD->ln = (x/r)*rad;
                                UD->mn = (y/r)*rad;
                                UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
                                if (UD->nn >= 1.0) return(-1);
                                          else UD->nn = -sqrt(1.0 - UD->nn);}
                                   }
                        }
    }
else
   {   if (r==0)
                {
            UD->ln =  0;
            UD->mn =  0;
            UD->nn = -1;
            }
         else
               { if(0<abs(r)<P1)
                           {mm += -(((-FD->cv)*r/(1.0+(1.0 - (1.0+FD->k)*(-FD->cv)*(-FD-
>cv)*rs)))*(2.0 + ((-FD->cv)*(-FD->cv)*rs*(1.0+FD->k))/((1.0 - (1.0+FD->k)*(-FD->cv)*(-FD-
>cv)*rs)*(1.0+(1.0 - (1.0+FD->k)*(-FD->cv)*(-FD->cv)*rs)))));
                    rad = 1.0 + (mm*mm);
                rad = mm / sqrt(rad);
                UD->ln = (x/r)*rad;
                UD->mn = (y/r)*rad;
                UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
                if (UD->nn >= 1.0) return(-1);
                          else UD->nn = -sqrt(1.0 - UD->nn);}

                    else { if(r>P1)
                             { mm += -((((-R1)+R2)*P1/(-R1)-r)/(sqrt(R2*R2-(r-((-
R1)+R2)*P1/(-R1))*(r-((-R1)+R2)*P1/(-R1)))));
                                        rad = 1.0 + (mm*mm);
                        rad = mm / sqrt(rad);
                        UD->ln = (x/r)*rad;
                        UD->mn = (y/r)*rad;
                        UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
                        if (UD->nn >= 1.0) return(-1);
                                  else UD->nn = -sqrt(1.0 - UD->nn);}

                                    else
```

```
                                        { mm += -((-((-R1)+R2)*P1/(-R1)-r)/(sqrt(R2*R2-
(r+((-R1)+R2)*P1/(-R1))*(r+((-R1)+R2)*P1/(-R1)))));
                                                rad = 1.0 + (mm*mm);
                            rad = mm / sqrt(rad);
                            UD->ln = (x/r)*rad;
                            UD->mn = (y/r)*rad;
                            UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
                            if (UD->nn >= 1.0) return(-1);
                                       else UD->nn = -sqrt(1.0 - UD->nn);}
                               }
                     }
    }
         if (Refract(FD->n1, FD->n2, &UD->l, &UD->m, &UD->n, UD->ln, UD->mn, UD->nn))
return(-FD->surf);
         break;

      case 6:
        /* ZEMAX wants the index, dn/dx, dn/dy, and dn/dz at the given x, y, z. */

         /* This is only required for gradient index surfaces, so return dummy values */
         UD->index = FD->n2;
         UD->dndx = 0.0;
         UD->dndy = 0.0;
         UD->dndz = 0.0;
        break;
      case 7:
        /* ZEMAX wants the "safe" data. */
         /* this is used by ZEMAX to set the initial values for all parameters and extra
data */
         /* when the user first changes to this surface type. */
         /* this is the only time the DLL should modify the data in the FIXED_DATA FD
structure */
         /*for (i = 1; i <= 8; i++) FD->param[i] = 0.0;*/
                 FD->param[1] = (FD->sdia)/2;
                 FD->param[2] = 100000000000;
         for (i = 1; i <= 200; i++) FD->xdata[i] = 0.0;
         break;
      }
   return 0;
   }

int Refract(double thisn, double nextn, double *l, double *m, double *n, double ln, double
mn, double nn)
{
double nr, cosi, cosi2, rad, cosr, gamma;
if (thisn != nextn)
        {
        nr = thisn / nextn;
        cosi = fabs((*l) * ln + (*m) * mn + (*n) * nn);
        cosi2 = cosi * cosi;
        if (cosi2 > 1) cosi2 = 1;
        rad = 1 - ((1 - cosi2) * (nr * nr));
```

```
        if (rad < 0) return(-1);
        cosr = sqrt(rad);
        gamma = nr * cosi - cosr;
        (*l) = (nr * (*l)) + (gamma * ln);
        (*m) = (nr * (*m)) + (gamma * mn);
        (*n) = (nr * (*n)) + (gamma * nn);
        }
return 0;
}
```

User Defined Surface code for TYPE 2 surface in Chapter 7

```
#include <windows.h>
#include <math.h>
#include <string.h>
#include <stdio.h>
#include "usersurf.h"


int __declspec(dllexport) APIENTRY UserDefinedSurface(USER_DATA *UD, FIXED_DATA *FD);

/* a generic Snells law refraction routine */
int Refract(double thisn, double nextn, double *l, double *m, double *n, double ln, double
mn, double nn);

BOOL WINAPI DllMain (HANDLE hInst, ULONG ul_reason_for_call, LPVOID lpReserved)
        {
   return TRUE;
   }

/*

This DLL models a novel aspheric surface described in:

"Advanced Theory of Field Curvature"
By Yuhao Wang
Ph.D. dissertation Chapter 7: TYPE 2 Powerless Dual Radii Surface

This surface is a non-polynomial type asphere.

*/

int __declspec(dllexport) APIENTRY UserDefinedSurface(USER_DATA *UD, FIXED_DATA *FD)
        {
   int i, loop, pmax;
   double P1,P2,R2, r, rs,R1, tp, alpha, dr, power, rad, t, x, y, z, dz, sag, mm;

   switch(FD->type)
        {
      case 0:
       /* ZEMAX is requesting general information about the surface */
         switch(FD->numb)
               {
            case 0:
               /* ZEMAX wants to know the name of the surface */
                        /* do not exceed 12 characters */
                        strcpy(UD->string,"flat+Dual_radius");
               break;
            case 1:
               /* ZEMAX wants to know if this surface is rotationally symmetric */
```

```
            /* it is, so return any character in the string; otherwise, return a null
string */
            strcpy(UD->string, "1");
            break;
         case 2:
            /* ZEMAX wants to know if this surface is a gradient index media */
            /* it is not, so return a null string */
            UD->string[0] = '\0';
            break;
         }
      break;
   case 1:
     /* ZEMAX is requesting the names of the parameter columns */
      /* the value FD->numb will indicate which value ZEMAX wants. */
      /* Only "q" in parameter 1 is used for this surface type */
      /* returning a null string indicates that the parameter is unused. */
      switch(FD->numb)
            {
         case 1:
            strcpy(UD->string, "P1");
            break;
                  case 2:
            strcpy(UD->string, "R1");
            break;
                  case 3:
            strcpy(UD->string, "P2");
            break;
                  case 4:
            strcpy(UD->string, "R2");
            break;
         default:
            UD->string[0] = '\0';
            break;
         }
     break;
   case 2:
     /* ZEMAX is requesting the names of the extra data columns */
      /* the value FD->numb will indicate which value ZEMAX wants. */
      /* returning a null string indicates that the extradata value is unused. */
      switch(FD->numb)
            {
       case 1:
            strcpy(UD->string, "# Terms");
            break;
        default:
            if (FD->numb <= FD->xdata[1] + 1)
            {
               sprintf(UD->string, "Term %i", FD->numb - 1);
               }
            else
            {
                  UD->string[0] = '\0';
```

```
                }
              break;
            }
        break;
      case 3:
        /* ZEMAX wants to know the sag of the surface */

         UD->sag1 = 0.0;
         UD->sag2 = 0.0;
         P1 = FD->param[1];
                R1 = FD->param[2];
         P2 = FD->param[3];
                R2 = FD->param[4];
                if (R2 < 0) return(-1);

         rs = UD->x * UD->x + UD->y * UD->y;
         r = sqrt(rs);
               /* R1=1/(FD->cv);*/


        /* alpha = 1 - (1+FD->k)*FD->cv*FD->cv*rs;
         if (alpha < 0) return(-1);*/
 if (R1>0)
        {        if(fabs(r)<=P1)
          { UD->sag1 = 0; UD->sag2 = UD->sag1;}

                else
                   { if(fabs(r)>P1&&fabs(r)<=P2)
                        {UD->sag1 = R1 - sqrt(R1*R1-(fabs(r)-P1)*(fabs(r)-P1)); UD->sag2 =
UD->sag1;}


                         else
               {UD->sag1 = R1-(R1+R2)*sqrt(R1*R1-(P2-P1)*(P2-P1))/R1+sqrt(R2*R2-(fabs(r)-
R2*(P2-P1)/R1-P2)*(fabs(r)-R2*(P2-P1)/R1-P2)); UD->sag2 = UD->sag1;}
            }

   }

 else
        {        if(fabs(r)<=P1)
          { UD->sag1 = 0; UD->sag2 = UD->sag1;}

                else
                    { if(fabs(r)>P1&&fabs(r)<=P2)
                        {UD->sag1 = -(fabs(R1) - sqrt(R1*R1-(fabs(r)-P1)*(fabs(r)-P1)));
UD->sag2 = UD->sag1;}


                         else
```

```
                        {UD->sag1 = -(fabs(R1)-(fabs(R1)+R2)*sqrt(R1*R1-(P2-P1)*(P2-
P1))/fabs(R1)+sqrt(R2*R2-(fabs(r)-R2*(P2-P1)/fabs(R1)-P2)*(fabs(r)-R2*(P2-P1)/fabs(R1)-
P2))); UD->sag2 = UD->sag1;}
            }

   }


   break;

      case 4:
        /* ZEMAX wants a paraxial ray trace to this surface */
         /* x, y, z, and the optical path are unaffected, at least for this surface type
*/
         /* for paraxial ray tracing, the return z coordinate should always be zero. */
         /* paraxial surfaces are always planes with the following normals */
         /* we will ignore the aspheric terms, even the quadratic one, since it has a */
         /* meaning that is hard to interpret if q != 0.0 */

         UD->ln =  0.0;
         UD->mn =  0.0;
         UD->nn = -1.0;
         power = (FD->n2 - FD->n1)*FD->cv;
         if ((UD->n) != 0.0)
               {
            (UD->l) = (UD->l)/(UD->n);
            (UD->m) = (UD->m)/(UD->n);

            (UD->l) = (FD->n1*(UD->l) - (UD->x)*power)/(FD->n2);
            (UD->m) = (FD->n1*(UD->m) - (UD->y)*power)/(FD->n2);

            /* normalize */
            (UD->n) = sqrt(1/(1 + (UD->l)*(UD->l) + (UD->m)*(UD->m) ) );
            /* de-paraxialize */
            (UD->l) = (UD->l)*(UD->n);
            (UD->m) = (UD->m)*(UD->n);
            }
         break;
      case 5:
        /* ZEMAX wants a real ray trace to this surface */
         /* okay, not a plane. */
         /* do not allow n == 0 */
         if (UD->n == 0.0) return -1;

         /* Now, we illustrate an iterative method of finding
            the intercept for a general surface. */

                     /* make sure we do at least 1 loop */

                     t = 100.0;
         tp = 0.0;
         x = UD->x;
```

```
        y = UD->y;
        z = UD->z;
        loop = 0;
        P1 = FD->param[1];
               R1 = FD->param[2];
        P2 = FD->param[3];
               R2 = FD->param[4];
      /*  pmax = FD->xdata[1];*/

        while (fabs(t) > 1e-10)
              {
                              /*
           First, compute the sag using whatever the surface sag expression is.
           This is given the x and y starting points. The following block of code
           will change depending upon the surface shape, the rest of this iteration
           is typically common to all surface shapes.
           */

               rs = x * x + y * y;
           r = sqrt(rs);
        P1 = FD->param[1];
               R1 = FD->param[2];
        P2 = FD->param[3];
               R2 = FD->param[4];

             /*  alpha = 1.0 - (1.0 + FD->k)*FD->cv*FD->cv*rs;

                 if (alpha < 0.0) return(-1);

                     R1=1/(FD->cv);*/

if (R1>0)
       {       if(fabs(r)<=P1)
          sag = 0;

               else
                      { if(fabs(r)>P1&&fabs(r)<=P2)
                        sag = R1 - sqrt(R1*R1-(fabs(r)-P1)*(fabs(r)-P1));


                       else
                            sag = R1-(R1+R2)*sqrt(R1*R1-(P2-P1)*(P2-P1))/R1+sqrt(R2*R2-
(fabs(r)-R2*(P2-P1)/R1-P2)*(fabs(r)-R2*(P2-P1)/R1-P2));
          }

   }
else
       {       if(fabs(r)<=P1)
          sag = 0;

               else
                      { if(fabs(r)>P1&&fabs(r)<=P2)
```

```
                        sag = -(fabs(R1) - sqrt(R1*R1-(fabs(r)-P1)*(fabs(r)-P1)));

                    else
                        sag = -(fabs(R1)-(fabs(R1)+R2)*sqrt(R1*R1-(P2-P1)*(P2-
P1))/fabs(R1)+sqrt(R2*R2-(fabs(r)-R2*(P2-P1)/fabs(R1)-P2)*(fabs(r)-R2*(P2-P1)/fabs(R1)-
P2)));
            }

    }
            /* okay, now with sag in hand, how far are we away in z? */
                    // modified 11/2012 to support larger angles of incidence
                    dz = (sag - z)*fabs(UD->n);

            /* now compute how far along the z axis this is */
            /* note this will crash if n == 0!! */
            t = dz / (UD->n);

            /* propagate the additional "t" distance */
            x += UD->l*t;
            y += UD->m*t;
            z += UD->n*t;

            /* add in the optical path */
                            tp += t;

            /* prevent infinte loop if no convergence */
            loop++;
            if (loop > 1000) return(-1);
            }

          /* okay, we should be a the intercept coordinates now */
          UD->x = x;
          UD->y = y;
          UD->z = z;

        /* don't forget the path! */
                        UD->path = tp;

          rs = x * x + y * y;
          r = sqrt(rs);
          P1 = FD->param[1];
                 R1 = FD->param[2];
          P2 = FD->param[3];
                 R2 = FD->param[4];

                /* R1=1/(FD->cv);
          alpha = 1.0 - (1.0+FD->k)*FD->cv*FD->cv*rs;*/
         /* if (alpha < 0) return(-1); /* ray misses */
         /* alpha = sqrt(alpha);*/
                 mm = 0.0;

                        /* now do the normals */
```

```
if (R1>0)
   {   if (r==0)
               {
           UD->ln =  0;
           UD->mn =  0;
           UD->nn = -1;
           }
        else
             { if(fabs(r)>0&&fabs(r)<=P1)
                       {mm += 0;
                  rad = 1.0 + (mm*mm);
               rad = mm / sqrt(rad);
               UD->ln = (x/r)*rad;
               UD->mn = (y/r)*rad;
               UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
               if (UD->nn >= 1.0) return(-1);
                         else UD->nn = -sqrt(1.0 - UD->nn);}

                      else
                {  if(fabs(r)>P1&&fabs(r)<=P2)
                          {mm += (fabs(r)-P1)/sqrt(R1*R1-(fabs(r)-P1)*(fabs(r)-P1));
                  rad = 1.0 + (mm*mm);
               rad = mm / sqrt(rad);
               UD->ln = (x/r)*rad;
               UD->mn = (y/r)*rad;
               UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
               if (UD->nn >= 1.0) return(-1);
                         else UD->nn = -sqrt(1.0 - UD->nn);}


                                  else
                                { mm += (-fabs(r)+R2*(P2-P1)/R1+P2)/sqrt(R2*R2-(fabs(r)-R2*(P2-
P1)/R1-P2)*(fabs(r)-R2*(P2-P1)/R1-P2));
                                      rad = 1.0 + (mm*mm);
                    rad = mm / sqrt(rad);
                    UD->ln = (x/r)*rad;
                    UD->mn = (y/r)*rad;
                    UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
                    if (UD->nn >= 1.0) return(-1);
                              else UD->nn = -sqrt(1.0 - UD->nn);}
                      }

                }

    }
else
   {   if (r==0)
               {
           UD->ln =  0;
           UD->mn =  0;
           UD->nn = -1;
           }
```

```
        else
              { if(fabs(r)>0&&fabs(r)<=P1)
                        {mm += 0;
                   rad = 1.0 + (mm*mm);
                rad = mm / sqrt(rad);
                UD->ln = (x/r)*rad;
                UD->mn = (y/r)*rad;
                UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
                if (UD->nn >= 1.0) return(-1);
                          else UD->nn = -sqrt(1.0 - UD->nn);}

                  else
                 {   if(fabs(r)>P1&&fabs(r)<=P2)
                          {mm += -((fabs(r)-P1)/sqrt(R1*R1-(fabs(r)-P1)*(fabs(r)-P1)));
                   rad = 1.0 + (mm*mm);
                rad = mm / sqrt(rad);
                UD->ln = (x/r)*rad;
                UD->mn = (y/r)*rad;
                UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
                if (UD->nn >= 1.0) return(-1);
                          else UD->nn = -sqrt(1.0 - UD->nn);}


                                  else
                               { mm += -((-fabs(r)+R2*(P2-P1)/fabs(R1)+P2)/sqrt(R2*R2-
(fabs(r)-R2*(P2-P1)/fabs(R1)-P2)*(fabs(r)-R2*(P2-P1)/fabs(R1)-P2)));
                                  rad = 1.0 + (mm*mm);
                        rad = mm / sqrt(rad);
                        UD->ln = (x/r)*rad;
                        UD->mn = (y/r)*rad;
                        UD->nn = UD->ln*UD->ln + UD->mn*UD->mn;
                        if (UD->nn >= 1.0) return(-1);
                                  else UD->nn = -sqrt(1.0 - UD->nn);}
                          }

                }
    }
        if (Refract(FD->n1, FD->n2, &UD->l, &UD->m, &UD->n, UD->ln, UD->mn, UD->nn))
return(-FD->surf);
        break;

      case 6:
        /* ZEMAX wants the index, dn/dx, dn/dy, and dn/dz at the given x, y, z. */

        /* This is only required for gradient index surfaces, so return dummy values */
        UD->index = FD->n2;
        UD->dndx = 0.0;
        UD->dndy = 0.0;
        UD->dndz = 0.0;
       break;
      case 7:
        /* ZEMAX wants the "safe" data. */
```

```
        /* this is used by ZEMAX to set the initial values for all parameters and extra
data */
        /* when the user first changes to this surface type. */
        /* this is the only time the DLL should modify the data in the FIXED_DATA FD
structure */
        /*for (i = 1; i <= 8; i++) FD->param[i] = 0.0;*/
                FD->param[1] = (FD->sdia)/5;
                FD->param[2] = 100000000000;
                FD->param[3] = FD->param[1]*2;
                FD->param[4] = 100000000000;
        for (i = 1; i <= 200; i++) FD->xdata[i] = 0.0;
        break;
      }
   return 0;
   }

int Refract(double thisn, double nextn, double *l, double *m, double *n, double ln, double
mn, double nn)
{
double nr, cosi, cosi2, rad, cosr, gamma;
if (thisn != nextn)
        {
        nr = thisn / nextn;
        cosi = fabs((*l) * ln + (*m) * mn + (*n) * nn);
        cosi2 = cosi * cosi;
        if (cosi2 > 1) cosi2 = 1;
        rad = 1 - ((1 - cosi2) * (nr * nr));
        if (rad < 0) return(-1);
        cosr = sqrt(rad);
        gamma = nr * cosi - cosr;
        (*l) = (nr * (*l)) + (gamma * ln);
        (*m) = (nr * (*m)) + (gamma * mn);
        (*n) = (nr * (*n)) + (gamma * nn);
        }
return 0;
}
```

REFERENCES

[1] Duncan T. Moore, "Gradient-index optics: a review" Applied Optics, Vol. 19, Issue 7, pp. 1035-1038 (1980)

[2] C. Gary Blough, John P. Bowen, Niels Haun, Douglas S. Kindred, R. John Koshel, Daniel M. Krill, Duncan T. Moore, Christopher E. Saxer, and David Y. H. Wang, "Effects of axial and radial gradients on Cooke triplets" Applied Optics, Vol. 29, Issue 28, pp. 4008-4015 (1990)

[3] Florian Bociort, "Aberration correction with thin radial gradient-index lenses" Proc. SPIE 2774, Design and Engineering of Optical Systems, (23 August 1996)

[4] http://en.wikipedia.org/wiki/Schmidt_camera

[5] Albert G. Ingalls (1952) Amateur Telescope Making Advanced (Book Two).

[6] http://www.hamamatsu.com/resources/pdf/etd/FOP_TMCP1040E01.pdf

[7] http://en.wikipedia.org/wiki/Petzval_field_curvature

[8] Jose M. Sasian-Alvarado (1991) U.S. Patent No. 5153778 A "Powerless field-corrective lens"

[9] Jose M. Sasian and Russell A. Chipman, "Staircase lens: a binary and diffractive field curvature corrector" Applied Optics, Vol. 32, Issue 1, pp. 60-66 (1993)

[10] José Sasián (2013) Introduction to Aberrations in Optical Imaging Systems, Cambridge University Press

[11] José Sasián “Theory of sixth-order wave aberrations” Applied Optics, Vol. 49, Issue 16, pp. D69-D95 (2010)

[12] http://fp.optics.arizona.edu/OT/Opti503/503%20-%20Aberrations.pdf Page86

[13] Takahiro Amanai (2004) U.S. Patent No. 20040218285 A1 “Image forming optical system and electronic instrument using the same”

[14] Chun-Cheng Ko (2012) U.S. Patent No. 8320061 B2 “Lens system having wide-angle, high resolution, and large aperture”

[15] Sayuri Noda (2007) U.S. Patent No. 7184225 B1 “Aspherical type imaging lens array”

[16]Takayuki Noda (2004) U.S. Patent No. 6825993 B2 “Wide-angle lens having aspheric surface”

[17] Fei-Hsin Tsai (2011) U.S. Patent No. 7933077 B1 “Wide-angle imaging lens module”

[18] http://en.wikipedia.org/wiki/Aspheric_lens

[19] http://www.visualopticslab.com/opti535/Lectures/Class03_08.pdf

[20] http://en.wikipedia.org/wiki/Zernike_polynomials

[21] G. W. Forbes “Shape specification for axially symmetric optical surfaces” Optics Express, Vol. 15, Issue 8, pp. 5218-5226 (2007)

[22] Greg Forbes “Better Ways to Specify Aspheric Shapes Can Facilitate Design, Fabrication and Testing Alike” International Optical Design Conference Jackson Hole, Wyoming United States June 13-17, 2010