

(2)

NPS-67-84-016

NAVAL POSTGRADUATE SCHOOL

Monterey, California

AD-A151 476



THESIS

SUPERSONIC MISSILE SEEKER LENS SYSTEM
USING GRADIENT INDEX MATERIAL
FOR FIRST ELEMENT LENS

by

David S. Davidson

September 1984

Thesis Advisor:

Allen E. Fuhs

DTIC
ELECT
MAR 22 1985

D

Approved for public release; distribution unlimited.

Prepared for: Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, VA 22209

85 03 11 113

DTIC FILE COPY

NAVAL POSTGRADUATE SCHOOL
Monterey, California

Commodore R. H. Schumaker
Superintendent

David A. Schradý
Provost

This thesis prepared in conjunction with research supported in part by Defense Advanced Research Projects Agency under DARPA Order 4035.

Reproduction of all or part of this report is authorized.

Released as a
Technical Report by:



J. N. DYER
Dean of Science and Engineering

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM	
1. REPORT NUMBER NPS-67-84-016	2. GOVT ACCESSION NO. A157476	3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) Supersonic Missile Seeker Lens System Using Gradient Index Material for First Element Lens		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; September 1984	
7. AUTHOR(s) David S. Davidson		6. PERFORMING ORG. REPORT NUMBER NPS-67-84-016	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		8. CONTRACT OR GRANT NUMBER(s) N622718WE40007	
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS	
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) COL Rene Larriva, USMC Defense Advanced Research Projects Agency 1400 Wilson Boulevard Arlington, Virginia 22209		12. REPORT DATE September 1984	
		13. NUMBER OF PAGES 119	
		15. SECURITY CLASS. (of this report) Unclassified	
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE	
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		Accession For NTIS GRA&I ☒ DTIC TAB ☐ Unannounced ☐ Justification	
18. SUPPLEMENTARY NOTES <i>(cont. to p 4)</i> <i>include:</i>		By Distribution/ Availability Codes Avail and/or Dist Special	
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Missile Sensors, Missile Aerodynamics, Projectile Aerodynamics, Gradient Refractive Index, Lens Design, Conical Optics, Conical Lens, Guided Projectile, and Ray tracing. <i>✓</i>		A-1	
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The design of a supersonic missile seeker lens system using an aerodynamically efficient lens, a scanning mirror and detector was accomplished through the use of ray-tracing routines. The design of the first element lens uses a gradient in refractive index to overcome the severe handicap that the pointed shape imposes. This problem has been previously solved for positive orientations of the center of symmetry of the Gradient Refractive Index (GRIN) material with respect to the lens. In this study the negative			

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE
S N 0102-LF-014-6601UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract continued

orientations are resolved and the lens studies for a range of orientations and strengths of gradient. The best lens is then used in conjunction with the mirror and detector in a ray-tracing scheme. Such a system was found to be feasible and worthy of further study.

S N 0102- LF- 014- 6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Approved for public release; distribution unlimited.

SUPERSONIC MISSILE SEEKER LENS SYSTEM
USING GRADIENT INDEX MATERIAL
FOR FIRST ELEMENT LENS

by

David S. Davidson
Captain, Canadian Forces
B. Eng (Engineering Physics), Royal Military College, 1976

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN PHYSICS

and

MASTER OF SCIENCE IN ENGINEERING SCIENCE

from the

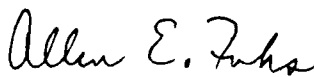
NAVAL POSTGRADUATE SCHOOL
September 1984

Author: .



David S. Davidson

Approved by:



A. E. Fuhs, Thesis Advisor



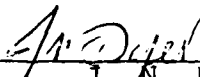
E. A. Milne, Second Reader



G. E. Schacher, Chairman
Department of Physics



M. F. Platzer, Chairman
Department of Aeronautics



J. N. Dyer
Dean of Science and Engineering

ABSTRACT

The design of a supersonic missile seeker lens system using an aerodynamically efficient lens, a scanning mirror and detector was accomplished through the use of ray-tracing routines. The design of the first element lens uses a gradient in refractive index to overcome the severe handicap that the pointed shape imposes. This problem has been previously solved for positive orientations of the center of symmetry of the Gradient Refractive Index (GRIN) material with respect to the lens. In this study the negative orientations are resolved and the lens studies for a range of orientations and strengths of gradient. The best lens is then used in conjunction with the mirror and detector in a ray-tracing scheme. Such a system was found to be feasible and worthy of further study. /

TABLE OF CONTENTS

I.	INTRODUCTION	.15
	A. OUTLINE OF LENS DESIGN TO DATE	.15
	B. GRADIENT REFRACTIVE INDEX (GRIN)	.15
	C. AIM AND INTENT OF THIS STUDY	.16
II.	GRIN LENS DESIGN	.18
	A. GRIN LENS DESIGN METHODOLOGY	.18
	B. GRIN RAY PATH DESCRIPTION	.19
	C. GRIN LENS GEOMETRY	.22
	D. RAY MATCHING	.24
III.	NEAR-RADIAL LINES	.30
	A. DESCRIPTION OF DIFFICULTY	.30
	B. PRELIMINARY WORK - IMPROVING THE ITERATION SCHEME	.31
	C. PRELIMINARY WORK - DOUBLE PRECISION	.36
	D. PRELIMINARY WORK - CHECKING ACCURACY	.36
	E. FIRST PROPOSED SOLUTION - STRAIGHT LINES	.38
	F. SECOND PROPOSED SOLUTION - REDUCED EQUATION	.38
	G. THIRD PROPOSED SOLUTION - MARCHAND METHOD	.45
	H. FOURTH PROPOSED SOLUTION - ANGLE IN TERMS OF RADIUS	.46
IV.	GRADIENT INDEX SEEKER LENS RAY TRACE RESULTS	.53
	A. FRONT SURFACE ACCURACY	.53
	B. SPOT SIZE RESULTS	.53

C.	SELECTION OF A BEST LENS	.53
V.	MIRROR AND DETECTOR SYSTEM	.57
A.	DESCRIPTION OF SYSTEM	.57
B.	LENS TO DETECTOR RAY TRACE	.58
1.	Ray Direction Exiting Lens	.58
2.	Point of Intersection on Mirror	.58
3.	Ray Direction Exiting Mirror	.61
4.	Ray Count on Detector	.64
C.	MIRROR AND DETECTOR SYSTEM RESULTS	.65
VI.	CONCLUSIONS AND RECOMMENDATIONS	.68
A.	CONCLUSIONS	.68
B.	RECOMMENDATIONS	.69
APPENDIX A:	PROGRAM GISL LISTING	.70
APPENDIX B:	PROGRAM CHECK LISTING	.93
APPENDIX C:	PROGRAM DETPLOT LISTING	.103
APPENDIX D:	DETECTOR SIGNAL	.107
LIST OF REFERENCES		.117
INITIAL DISTRIBUTION LIST		.118

LIST OF TABLES

I. Front Surface Accuracy	.55
-------------------------------------	-----

LIST OF FIGURES

2.1	GRIN Lens Coordinate System and Parameters (from Carr)	.20
2.2	GRIN Lens and Ray Geometry	.23
2.3	Ray Path Showing Improper Matching and Minimum Radius	.26
2.4	Ray Path Radius and Angle	.27
2.5	Ray Matching Regions	.29
3.1	Ray Trace Showing Region of Radial Lines . . .	.32
3.2	Newton-Raphson Iteration Geometry	.34
3.3	Newton-Raphson Logic Flowchart	.35
3.4	Geometry of Accuracy Check - Forward Ray Trace	.39
3.5	Expanded View of Front Surface - Straight Line Method	.41
3.6	Reduced Equation Implementation	.44
3.7	Radius and Angle Relation	.48
3.8	Angle in Terms of Radius Iteration Logic Flowchart	.50
3.9	Angle in Terms of Radius Iteration Geometry . .	.51
3.10	Diverging Iteration - Forward Ray Trace	.52
4.1	Spot Size Results - ALFAP = 0.3 Radians	.56
5.1	Lens, Mirror and Detector System	.59
5.2	Detector Signal	.60
5.3	Reflection of Ray on Mirror	.63

5.4	Mirror Tilt for On-Axis Spot and Angle of	
	Incidence	.67

LIST OF SYMBOLS

SYMBOL	FORTTRAN EQUIVALENT	DEFINITION	UNITS
a	A	index function parameter	nondimensional
arg	ARG	argument of arcsine function in GRIN ray equation	nondimensional
arg ₀	-	argument of initial arcsine function in GRIN ray equation	nondimensional
b	B	index function parameter	nondimensional
c	C	ray matching constant	radians
D _{1m}	D1	ray geometrical distance from lens to mirror	nondimensional
D _{nd}	D2	ray geometrical distance from mirror to detector	nondimensional
DT	DETP	position of detector along x-axis	nondimensional
e	E	GRIN ray trace constant	nondimensional
K	CK	x-direction cosine of ray from lens to mirror	nondimensional
K'	CKK	x-direction cosine of ray from mirror to detector	nondimensional
θ_0	-	z-direction cosine of ray at initial point	nondimensional
L	CL	y-direction cosine of ray from lens to mirror	nondimensional
L'	CLL	y-direction cosine of ray from mirror to detector	nondimensional

M	CM	z-direction cosine of ray from lens to mirror	nondimensional
M'	CMM	z-direction cosine of ray from mirror to detector	nondimensional
M_0	-	Marchand reduced integral expression	-
MP	MIRP	position of mirror pivot point on x-axis	nondimensional
n	-	index of refraction	nondimensional
n_0	-	index of refraction at initial point of GRIN ray	nondimensional
$\vec{n}$		vector normal to mirror plane	-
n_x	-	x-component of vector normal to mirror plane	nondimensional
n_y	-	y-component of vector normal to mirror plane	nondimensional
p_0	-	x-direction cosine of GRIN ray at initial point	nondimensional
q_0	-	y-direction cosine of GRIN ray at initial point	nondimensional
r	RAD	radius to ray from center of symmetry	nondimensional
r_0	RO	radius to ray at initial point	nondimensional
r'	RP	first guess radius in iteration scheme	nondimensional
r''	-	second guess radius in iteration scheme	nondimensional
r'''	-	third guess radius in iteration scheme	nondimensional
r_{fs}	-	radius to front surface	nondimensional

$\dot{r}_{fs}$	-	derivative with respect to angle of radius to front surface	nondimensional
r_{gr}	-	radius to GRIN ray	nondimensional
$\dot{r}_{gr}$	-	derivative with respect to angle of radius to GRIN ray	nondimensional
$\vec{r}_i$	-	ray vector incident to mirror	nondimensional
$r_{xi},$ r_{yi}, r_{zi}	-	components of ray vector incident to mirror	nondimensional
$\vec{r}_r$	-	ray vector reflected off mirror	-
$r_{xr},$ r_{yr}, r_{zr}	-	components of ray reflected off mirror	nondimensional
R	R	radius from x-axis to outermost point of inner surface	nondimensional
R_z	Rzero	radius to edge of GRIN material	nondimensional
U	U	angle of incident radiation for lens construction	radians
x_d	XD	x-component of ray at detector	nondimensional
x_m	XM	x-component of ray at mirror	nondimensional
x_0	XO	x-component of ray at lens inner surface	nondimensional
y_d	YD	y-component of ray at detector	nondimensional
y_m	YM	y-component of ray at mirror	nondimensional

y_0	YO	y-component of ray at lens inner surface	nondimensional
z_d	ZD	z-component of ray at detector	nondimensional
z_m	ZM	z-component of ray at mirror	nondimensional
z_0	ZO	z-component of ray lens inner surface	nondimensional
ϵ	EPSLON	sign function of GRIN ray equation	-
ψ	PSI	angle between GRIN ray and radial arm from center of symmetry	radians
ψ_0	PSIO	angle between GRIN ray and radial arm at initial point	radians
θ	THETA	angle from x-axis to radius	radians
θ_0	THETO	angle from x-axis to initial point	radians
θ_L	-	angle for GRIN ray part to left of part of minimum radius	radians
θ_R	-	angle for GRIN ray part to right of point of minimum radius	radians
θ'	THETAP	angle of first guess in iteration scheme	radians
θ''	-	angle to second guess in iteration scheme	radians
θ'''	-	angle of third guess in iteration scheme	radians
θ_m	THETM	angle from x-axis to plane of mirror	radians

ACKNOWLEDGEMENTS

The author would like to thank Distinguished Professor Allen E. Fuhs for his guidance and perserverance during the writing of this thesis.

The author would also like to express his gratitude to his wife Johanne for her support during this period.

I. INTRODUCTION

A. OUTLINE OF LENS DESIGN TO DATE

The design of a 'pointed' aerodynamically efficient missile seeker lens has been the topic of a number of studies. Frazier [Ref. 1], and Terrell [Ref. 2], first studied the idea of using a pointed lens as a means of obtaining some degree of aerodynamic efficiency. They also introduced the idea of using Gradient Refractive Index (GRIN) material in order to overcome the difficulties in imaging imposed by the odd shape. Amichai [Ref. 3], and Carr [Ref. 4], studied the GRIN concept in more detail, Amichai by defining the outer surface as a cone and solving for the inner surface and Carr by defining the inner surface as a cone and solving for the outer. The main reference for this author's study was drawn from Carr's work and a review of that thesis would be helpful if the reader seeks a more detailed background from that given in the next few pages.

B. GRADIENT REFRACTIVE INDEX (GRIN)

Although the theory for GRIN has been developed and known for some time it was not until fairly recently that materials have become available that allow the theory to be put into practice. A gradient in index will cause light rays to bend in an otherwise homogeneous material. Thus,

the lens designer now has a new dimension of flexibility which will allow him to solve problems that could not be formerly solved. A proper selection of the manner in which the gradient varies is also important. The designer would normally select certain degrees of symmetry. Three are most common; axial, radial or spherical gradients.

GRIN material in the shape of rods now commonly use axial and radial gradients in such devices as fiber-optic cables and photocopying machines. Spherical gradients are not available because spherical gradient material has not yet been produced. Spherical gradients do offer an advantage in that they lend themselves more easily to mathematical description since closed form solutions for the ray paths can be obtained.

A good introduction to gradients and the materials currently available for production of large scale GRIN lens systems is provided by Moore [Ref. 6], and Light [Ref. 7]. These also give the reader an idea of the advantages that may be realized along with the problems associated with the design and production of such items.

C. AIM AND INTENT OF THIS STUDY

The aim of this study is to complete the design of the Gradient Index Seeker Lens (GISL) initiated by Carr. This design uses a spherical gradient and 'pointed' lens shape. This study does not determine the aerodynamic efficiency of

the design but rather ensures that the lens has a generally pointed character that will also allow some degree of imaging for off-axis objects. The spherical gradient is chosen since it seems to fit the geometrical shape of the lens and also since the three-dimensional analysis of rays is much simpler.

This design is accomplished through the use of ray tracing techniques. Thus, no actual study of the lens material nor the manner in which such a lens could be manufactured is intended. It is the hope of the author that studies of this nature will provide the impetus for the development of suitable GRIN material with spherical gradients.

Once the lens design was completed a 'best' lens was chosen and a simple detector system applied to it to see if precise definition of off-axis targets is possible.

II. GRIN LENS DESIGN

A. GRIN LENS DESIGN METHODOLOGY

The method by which the GRIN Lens was designed in Carr's thesis [Ref. 4], is the subject of this section. A cross-section view of the top half of the lens is shown in Figure 2.1.

As can be seen, the lens is designed by defining the inner surface as a cone and then constructing the outer surface so that a backwards ray trace from a point source will result in parallel lines emerging. This is done by tracing rays from the back edge of the lens towards the front surface and finding the point of intersection of the ray and front surface line. The front surface line is found by using the previous ray solution as described below. The intersection of the ray and the line is determined using an iteration procedure. Once a solution is found the front surface line that will cause the ray to emerge parallel with all other rays is determined. This front surface line is then used to find a point solution for the next ray. About 1000 rays are traced starting at the outside edge of the inner cone and working inwards towards the center. Details of the iteration process is included in later sections.

Once the lens outside surface has been constructed the lens is then studied to see how well it will image three dimensional skew rays that enter the lens at varying degrees of incidence. To accomplish this, a grid is placed in front of the lens and then rays are traced from the intersection points of the grid through the lens and onto the image plane. The grid is then tilted at successive angles and the resulting 'spots' on the image plane recorded for each case. The manner in which these spots move on the image plane and their size are the major performance characteristics of the lens.

B. GRIN RAY PATH DESCRIPTION

Following Marchand [Ref. 5], GRIN ray paths in spherical gradients are found through the solution of the appropriate line integral:

$$r = r_0 + e \int_{r_0}^r \frac{dr}{r(n^2 r^2 - e^2)^{1/2}} \quad (2.1)$$

where

$$e = n r \sin \theta = n_0 r_0 \sin \theta \quad (2.2)$$

and

$$\sigma = \pm 1$$

It should be noted that the value of σ is -1.0 if the radius is decreasing with angle and +1.0 if the radius is increasing with angle.

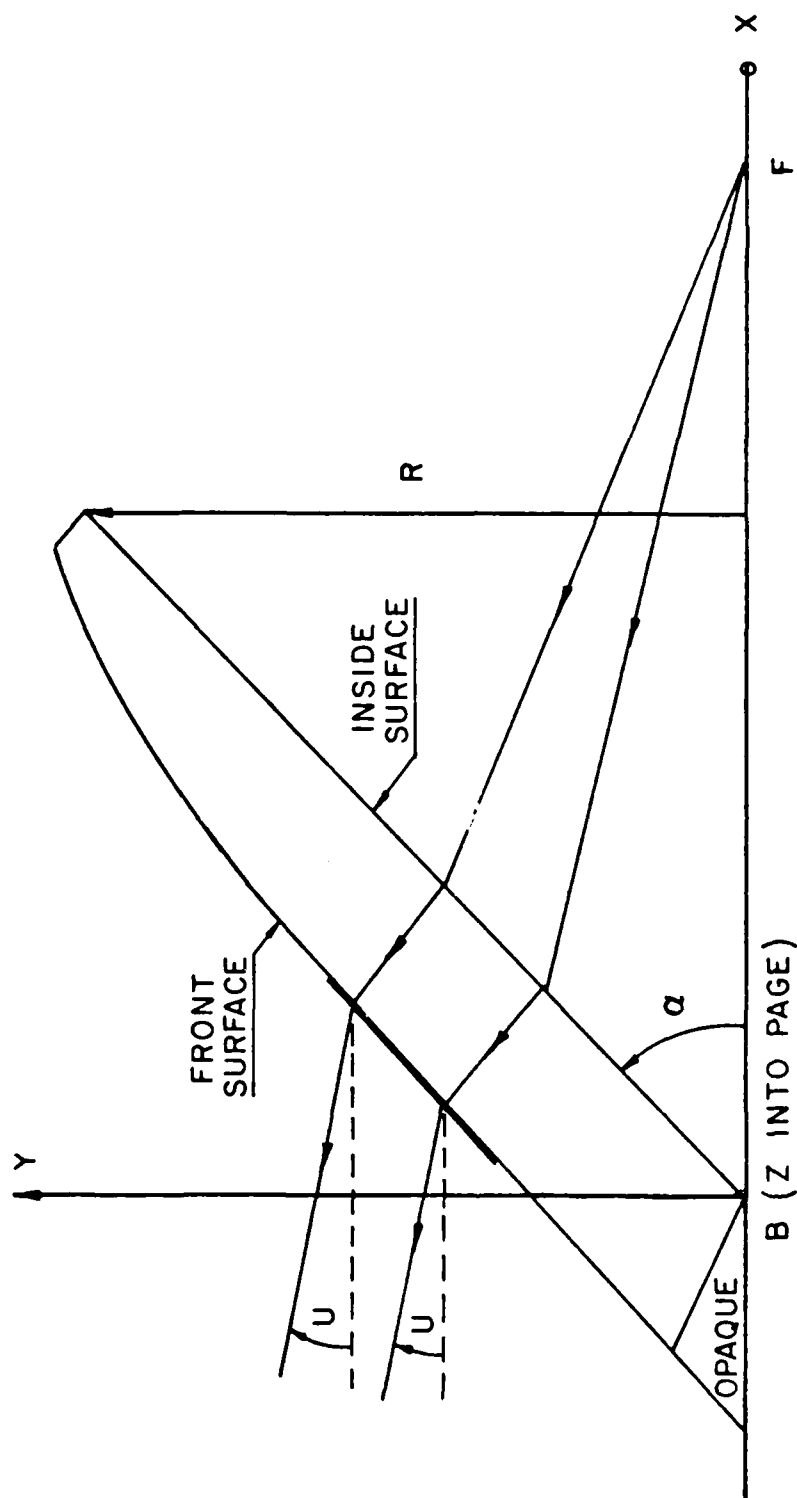


Figure 2.1 GRIN Lens Coordinate System and Parameters (from Carr).

The choice of index gradient is instrumental in allowing the integral to be solved in closed form. In this study the index is defined as:

$$n^2 = (a + br^2/R_z^2)$$

The constant a defines the strength of the index at the origin and the constant b defines the strength of the gradient and whether it is negative or positive.

As can be seen in Figure 2.2, the ray path is described by the constant e in conjunction with the parameter θ , the angle between the radial arm of the coordinate system and the direction of the ray at any point.

The solution of the integral in equation 2.1 results in the following expression:

$$\theta = \theta_0 - \frac{\varepsilon}{2} \{ \sin^{-1}(\arg_0) - \sin^{-1}(\arg) \} \quad (2.3)$$

where

$$\arg_0 = \frac{a - 2e/r_0^2}{(a^2 + 4be^2/R_z^2)^{1/2}}$$

and

$$\arg = \frac{a - 2e/r^2}{(a^2 + 4be^2/R_z^2)^{1/2}}$$

or, expressed in terms of the radius,

$$r = \frac{\sqrt{2}|e|}{\{a + (a^2 + 4be^2/R_z^2) \sin [-2\epsilon(\alpha - \alpha_0) - \sin^{-1}(\arg_0)]\}^{1/2}} \quad (2.4)$$

C. GRIN LENS GEOMETRY

A factor which must be considered is the manner in which the lens is cut out of the GRIN material. Figure 2.2 shows an orientation of the material with respect to the lens such that the center of symmetry of the material is inside the lens. Carr has defined this as a negative value of OB , the distance from the point O to the point B . Carr was able to work out the lens problem for positive values of this parameter only, thus the major portion of this study is to develop methods that enable solutions for negative values of OB . In so doing, two factors have to be overcome:

a) ray matching - When the ray path radius initially decreases and then increases with change in angle the ray path equation must be matched at the point of minimum radius.

b) near-radial lines - Rays that show very little curvature are difficult to describe in terms of radius and angle and these cause problems in the iterative solution scheme.

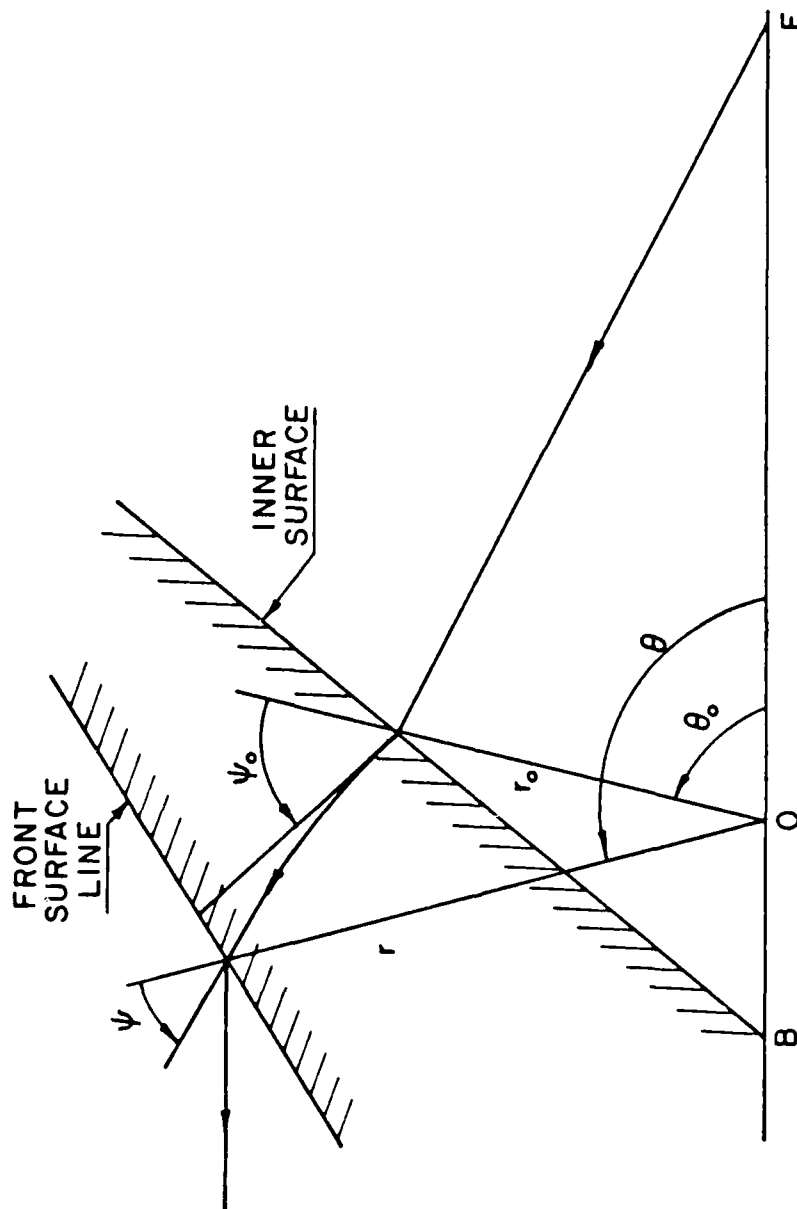


Figure 2.2 GRIN Lens and Ray Geometry.

The solution to the ray matching problem is relatively simple and is discussed in the next section. The resolution of near-radial lines is a more complex task and is covered in the following chapter.

D. RAY MATCHING

As stated previously the ray path equation changes signs as the radius of the path decreases and then increases. Such a path is shown in Figure 2.3. Simply changing sign in the equation as the point of minimum radius is passed does not produce the desired result, but rather has the effect shown in Figures 2.3 and 2.4.

The resolution of this problem is fairly simple. At the point of minimum radius the arcsine function in equation 2.3 containing this radius (arg) reduces to $-\pi/2$. This is due to the fact that at this point the value of e is $-1/2$. Thus:

$$\sin \phi = 1$$

and

$$e^2 = n^2 r^2$$

Thus:

$$\arg = \frac{a - 2n^2}{(a^2 + 4bn^2 r^2 / R_z^2)^{1/2}} = \frac{a - 2(a + br^2 / R_z^2)}{(a^2 + 4b(a + br^2 / R_z^2) r^2 / R_z^2)} = 1$$

and

$$\sin^{-1}(\arg) = -\pi/2$$

If the radius is initially decreasing as shown in Figure 2.3, we can find the angle of the minimum radius. We call this the critical angle or θ_L :

$$\theta_L = \theta_0 - \frac{\epsilon}{2} \{ \sin^{-1}(\arg_0) + \pi/2 \}$$

The positive and the negative forms of the equation should both contain the point of minimum radius, in other words, by approaching the point from the left or the right the same result should be obtained. The fact that this does not happen is shown in Figure 2.3 and reveals the need to match the rays at the point of minimum radius.

This matching is achieved by simply adding a constant to either the positive or the negative form of equation 2.3 and then matching at the point of minimum radius in the following manner:

$$\theta_R = \theta_L + c$$

thus

$$- \frac{\epsilon}{2} \{ \pi/2 + \sin^{-1}(\arg_0) \} = + \frac{\epsilon}{2} \{ \pi/2 + \sin^{-1}(\arg_0) \} + c$$

The constant c can then be found:

$$c = - \epsilon \{ \pi/2 + \sin^{-1}(\arg_0) \}$$

This constant must now be applied for all points past the point of minimum radius; in our example all points in

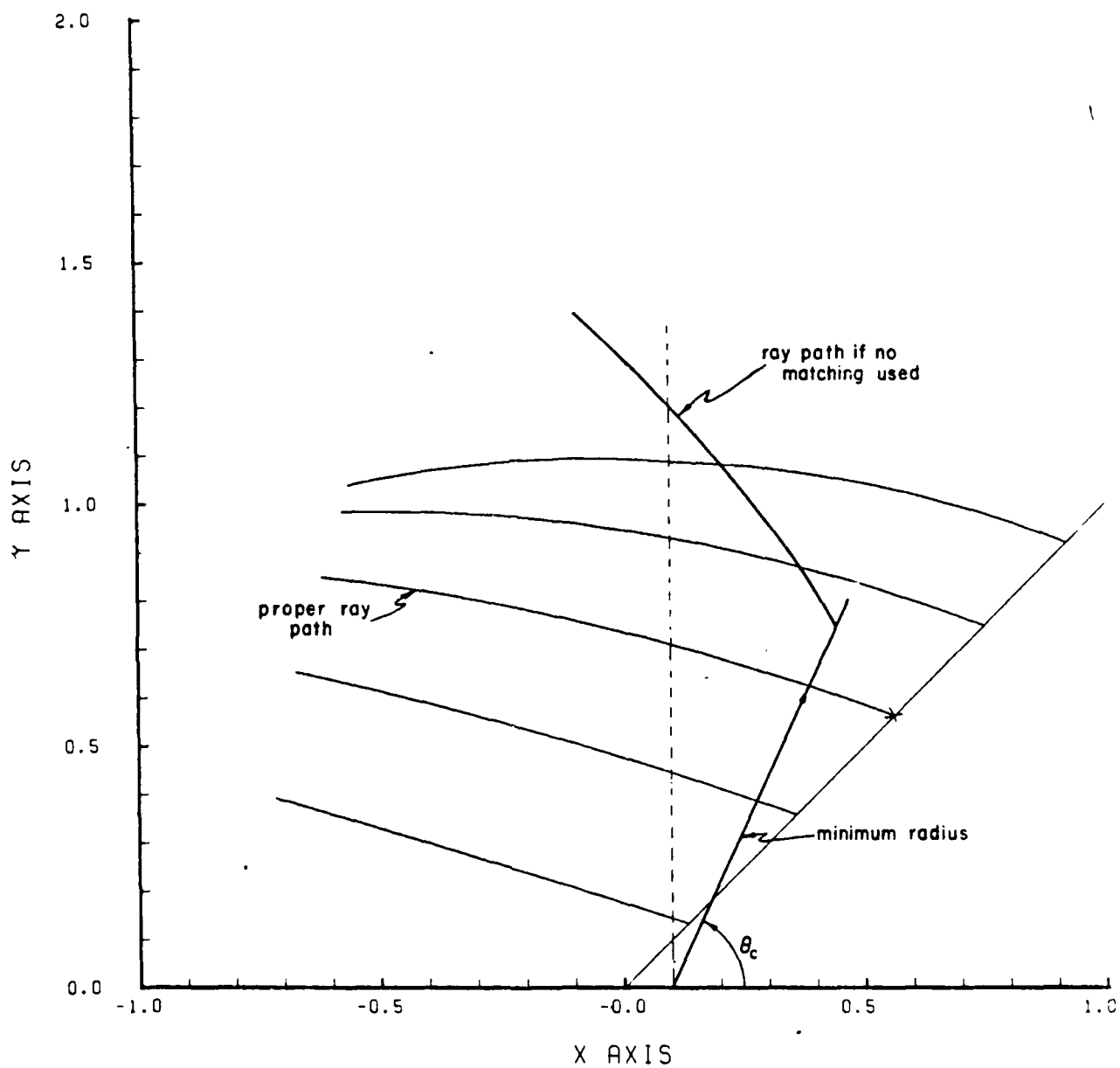


Figure 2.3 Ray Path Showing Improper Matching and Minimum Radius.

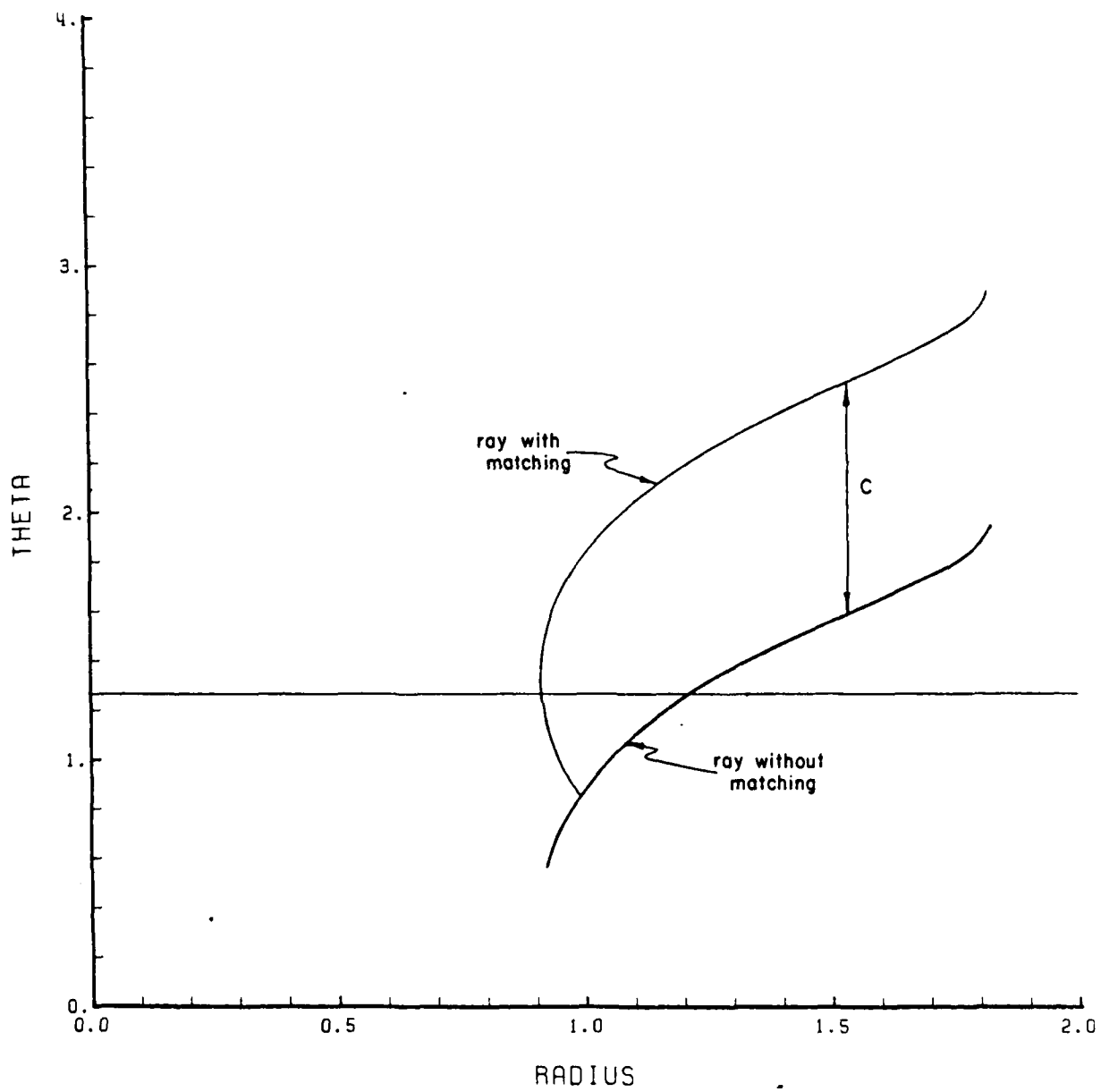


Figure 2.4 Ray Path Radius and Angle.

the increasing radius portion of the ray must be found using the form:

$$\theta = \theta_0 + \frac{\epsilon}{2} [\sin^{-1}(\arg_0) = \sin^{-1}(\arg)] + c$$

In the iteration scheme the ray path equation must be more carefully handled. First, the critical angle and the constant c must be found. Second, the angle of the ray must be checked to see if it has passed the critical angle. If it has then the constant c is added to equation 2.3 and the point recalculated. The details of the iteration methods will be described in the following chapter.

For further clarification Figure 2.5 shows the regions in which ray matching is required for a certain position of the center of symmetry. Note also that were the value of OB positive then all rays in the lens would never reach a minimum radius point. Thus ray matching is a problem unique to the negative OB case.

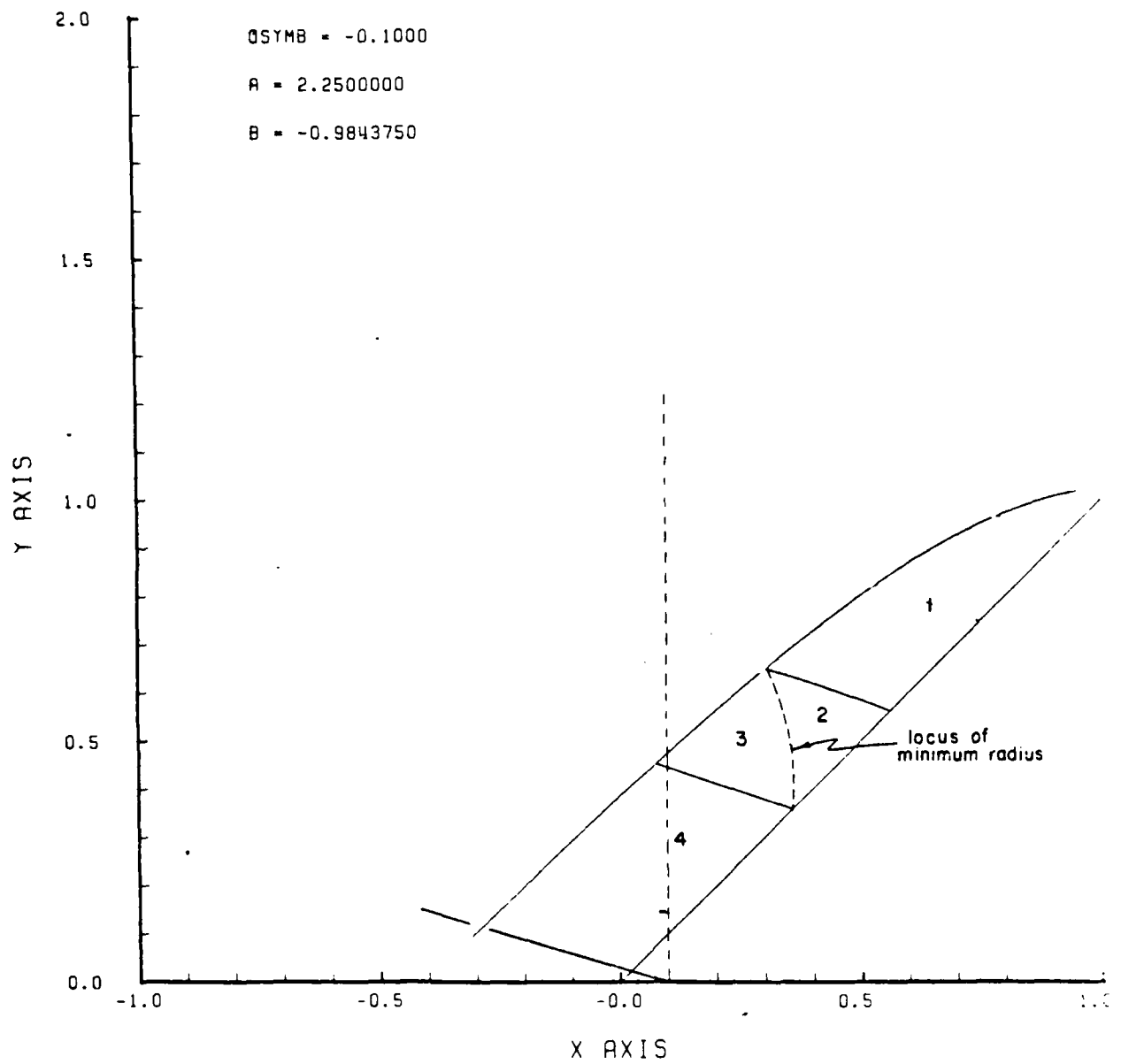


Figure 2.5 Ray Matching Regions.

III. NEAR-RADIAL LINES

A. DESCRIPTION OF DIFFICULTY

As mentioned previously, the ray paths that are nearly radial are difficult to describe mathematically in terms of radius and angle. This is due to the simple reason that a straight line originating from the center of the coordinate system has no change in angle and an undefinable change in radius.

The region in which this problem occurs in the lens design is shown in Figure 3.1. As can be seen, near-radial lines mean that the angle ψ and thus the constant e will be very small, so small as to cause equation 2.4 to approach a singularity and thus break down. In the iterative scheme, as the front surface solutions are found from the outside edge of the lens inwards, the values of ψ and e become increasingly smaller if the center of symmetry is inside the lens (OB negative). When this region is reached, equation 2.4 reduces the r_0 for all angles and further construction of the front surface is rendered impossible. This effect is shown in Figure 3.1.

Of note is the fact that if the radial line region is passed successfully, the rays begin to curve in the opposite sense, that is they are always convex to the

radial lines in the case of a negative gradient and concave to the radial lines if a positive gradient is used.

The resolution of the near-radial lines proved to be a difficult task and became a major portion of this study. The manner in which it was overcome is the subject of the following sections.

B. PRELIMINARY WORK - IMPROVING THE ITERATION SCHEME

The first step in the resolution of the problem was to simplify the iteration scheme. In his work Carr developed a scheme based on the geometry of the rays and coordinate system [Ref. 4], which worked well for all positive values of OB (that is, for all orientations of the center of symmetry to the left of the inside surface cone. See Figure 2.2). It was decided instead to use the Newton-Raphson approach, which is simpler in concept and not dependent on geometry.

Carr used the Newton-Raphson method in the second portion of the design involving the skew ray trace, but he did not use it for the lens construction.

The manner in which this scheme works is now described so that the reader will be able to better understand the concept of how all iterative methods work for this design, as well as how the ray matching previously described fits in.

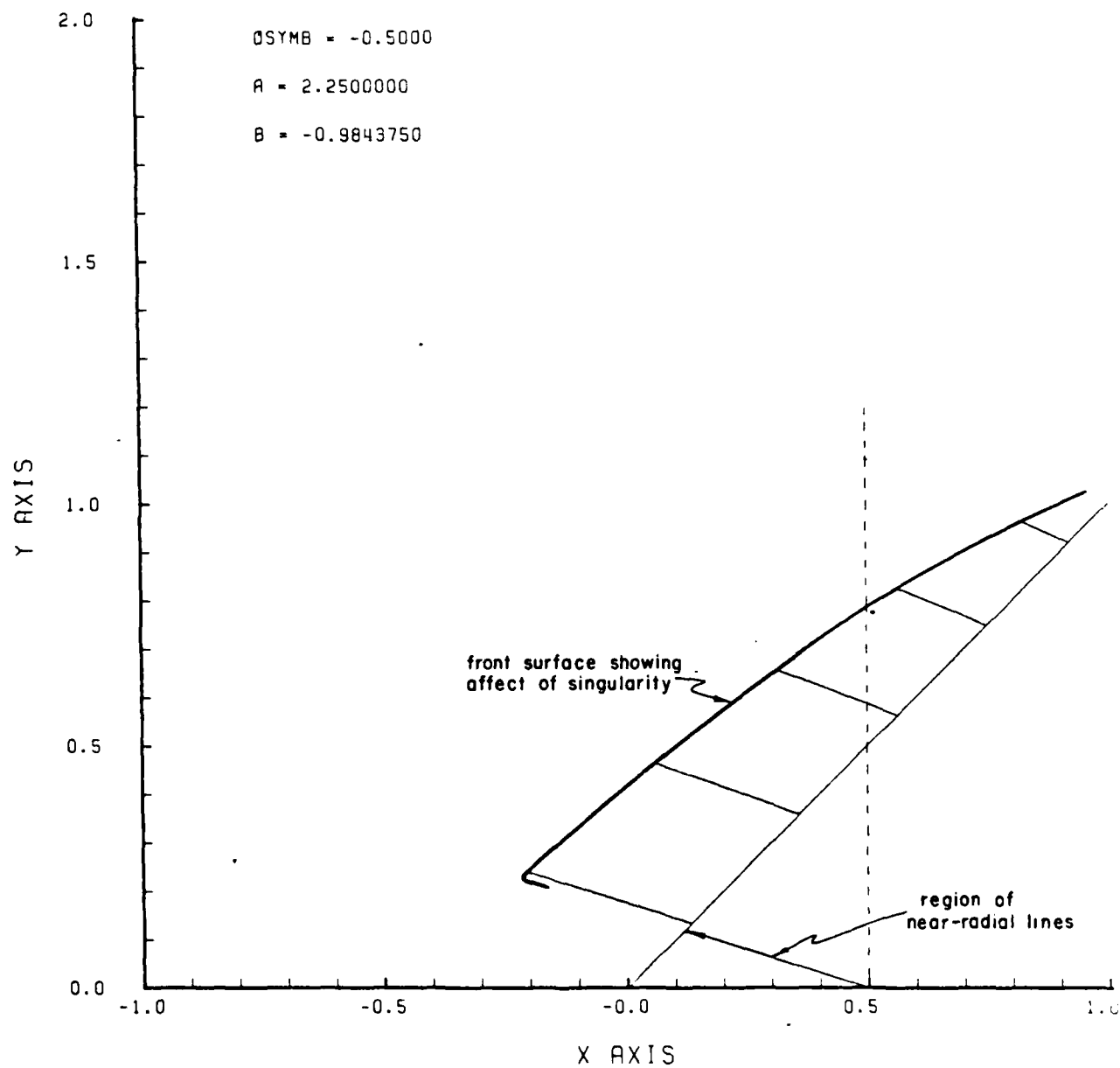


Figure 3.1 Ray Trace Showing Region of Radial Lines and
Effect of Singularity.

The geometry associated with this plan is shown in Figure 3.2 and a flowchart outlining the iteration logic is shown as Figure 3.3. As can be seen with reference to the two figures, an initial point is first chosen by drawing a straight line from the inside surface intercept to the slope of the front surface obtained from a previous ray path solution. Newton-Raphson seeks to find the intercept of this front surface line with the curved ray path. This is done by using the initial angle to find the radius and derivative with respect to angle for each line and then calculating a new angle according to the relation:

$$\theta'' = \theta' - \frac{r_{fs} - r_{gr}}{\dot{r}_{fs} - \dot{r}_{gr}}$$

This new angle is then used to recalculate the radii and derivatives and thus an iterative loop is formed. Once the difference in angle is small enough the iteration is stopped and the solution considered to be found.

Ray matching is also employed in this scheme. As each new angle is calculated, it is checked to see if the critical angle has been passed. If it has, the constant c of equation 2.5 is added to the angle and thus the proper radius and derivative will be then found.

Once a solution has been found it is used to find a new front surface slope by determining that slope which will

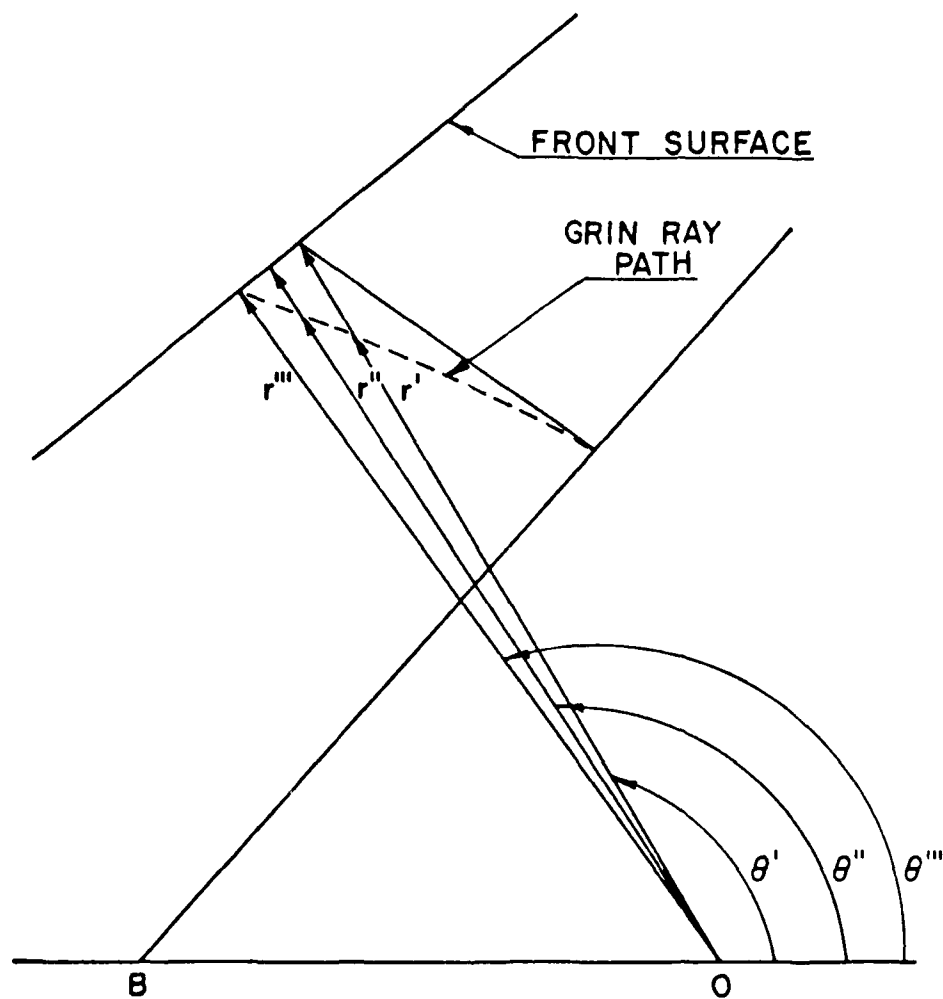


Figure 3.2 Newton-Raphson Iteration Geometry.

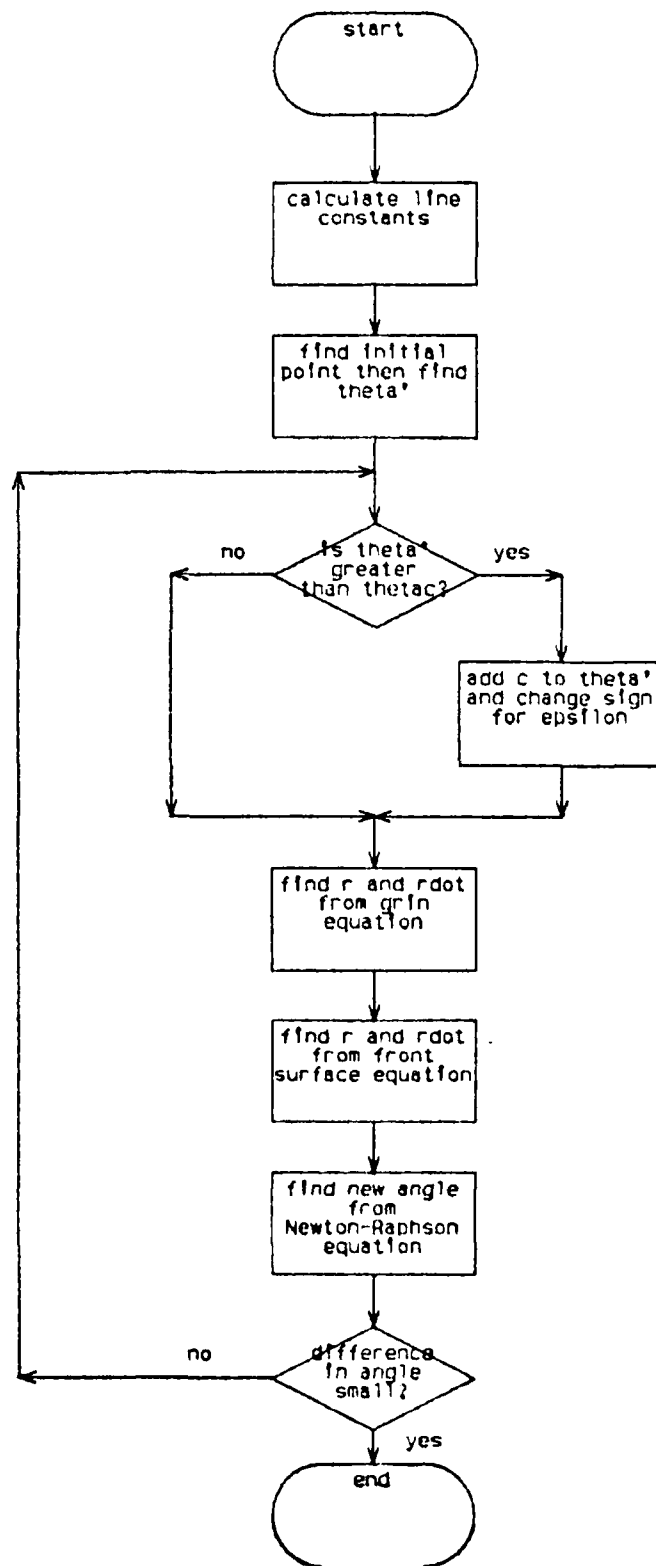


Figure 3.3 Newton-Raphson Logic flowchart.

cause the refracted ray to emerge parallel to the axis. It is this slope that is used to find the solution for the next ray in the scheme.

When this new iterative procedure was implemented it was found to work efficiently for both the positive and negative gradients, but it did not resolve the problem of near-radial lines.

The reader should refer to Appendix A for a listing of program GISL, which is the ray tracing program originally written by Carr and modified by the author.

C. PRELIMINARY WORK - DOUBLE PRECISION

The next avenue that was explored was to place the ray tracing scheme in double precision. This was considered necessary since very small differences in angle were required in the region of the radial lines.

Once this was installed in the routine it was found to improve significantly the accuracy of the calculations and succeeded in delaying the point at which the iterations broke down. It did not solve the crux of the difficulty however.

D. PRELIMINARY WORK - CHECKING ACCURACY

At this point it was decided to develop a scheme to check the accuracy of the front surface solution. The manner in which this was done is shown in Figure 3.4

The ray trace was completed in a forward direction from the front surface to the focal point in a two-dimensional fashion. These rays were caused to originate from the mid point of each set of points that define the front surface. The slope of the front surface for each of these rays was taken to be the line drawn between the two adjacent points on the front surface. In this fashion the worst case was chosen since the skew ray trace used to determine the performance of the lens uses incident rays that can hit the front surface at any point.

These rays were then traced through the lens and back towards the focal point. The x-coordinate of the focal point was defined and the y-coordinate calculated from the ray trace to see how close it would come to the y-coordinate of the focal point originally used to start construction of the front surface. A listing of this routine is shown as Appendix B.

Another benefit of this two-dimensional ray trace was that it allowed the author to confirm the methods by which a forwards ray trace is done. In this manner ray matching and sign convention in angles was checked. Also this ray trace is different in that a solution between the ray path in the lens and the back surface, rather than the front surface, was now required in the iterative solution scheme and this could now be verified to ensure that it was properly done.

Another method of checking the accuracy of the solution would be to study the front surface smoothness in the region of near-radial lines. Although not as precise as the above method this idea allowed the author to confirm that the front surface was sufficiently smooth. By this means it was also possible to see a physical representation of the lens in the region of the singularity.

E. FIRST PROPOSED SOLUTION - STRAIGHT LINES

The first proposed method of solving the mathematical difficulty was to simply replace the near-radial lines with completely straight lines just before the singularity was reached. This could be done by following the decrease of the parameter ϕ_0 until it became small enough to ensure that the lines were nearly straight.

This idea was implemented but it was found not to be sufficiently accurate. It must be emphasized that an inaccuracy in even a single point could not be tolerated since this would mean that all succeeding points in the iterative solution scheme would carry that error as well.

For these reasons the straight line solution was rejected and other means of solving the difficulty were sought.

F. SECOND PROPOSED SOLUTION - REDUCED EQUATIONS

The second method that was used to resolve the problem of near-radial lines was to reduce equation 2.4 by

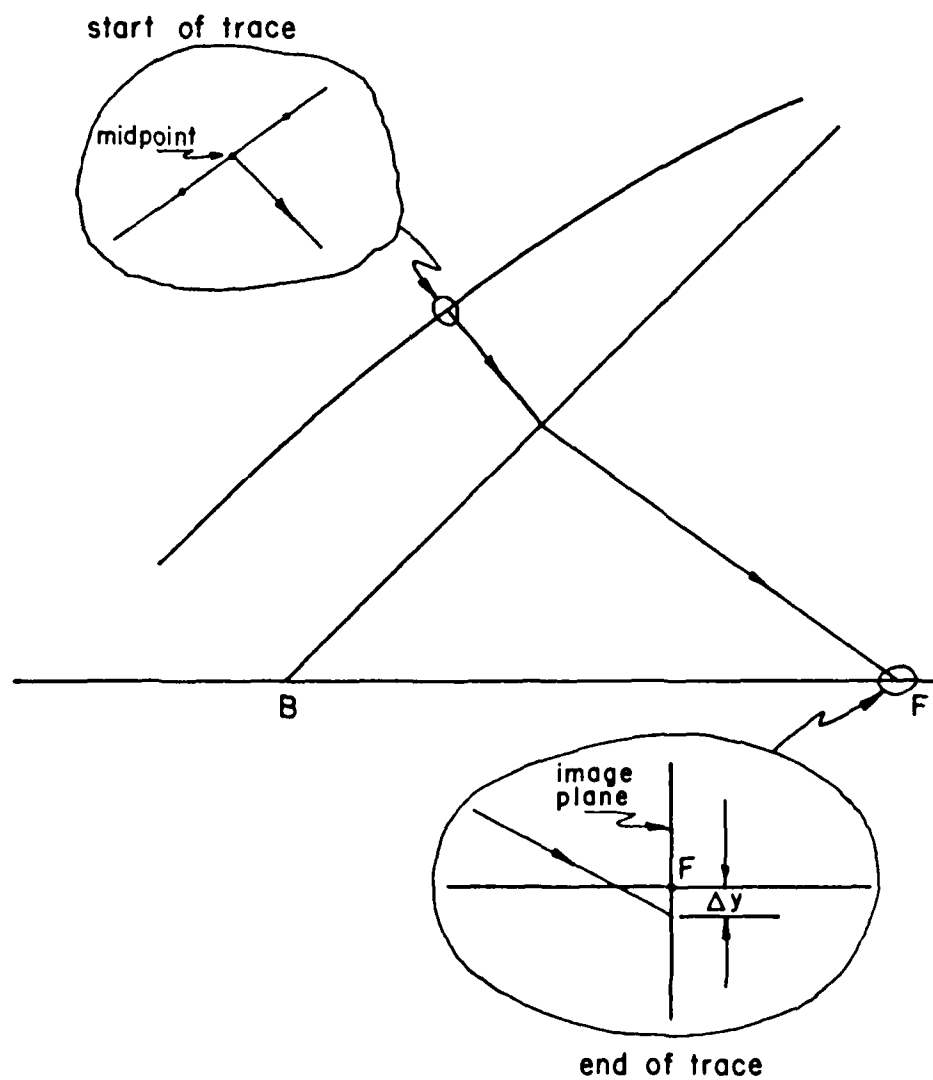


Figure 3.4 Geometry of Accuracy Check - Forwards Ray Trace.

employing small angle approximations, binomial expansions and curve fitting. The details of how this was done follows.

Recalling equation 2.3:

$$A = \theta_0 - \frac{r}{2} \{ \sin^{-1}(\arg_0) - \sin^{-1}(\arg) \}$$

The arguments of these arcsine functions were reworked as follows:

$$\arg_0 = \frac{a - 2(a+b')\psi_0^2}{(a^2 + 4b'(a+b')\psi_0^2)^{1/2}}$$

and

$$\arg = \frac{a - 2(a+b')\psi_0^2 r_0^2 / r^2}{(a^2 + 4b'(a+b')\psi_0^2)^{1/2}}$$

where

$$\psi_0 \approx \sin \theta_0$$

$$b' = br_0^2 / R_z^2$$

Note the angle ψ_0 is very small.

These arguments were then reduced using a binomial expansion:

$$\arg_0 = 1 - \frac{2(a+b')^2 \psi_0^2}{a^2}$$

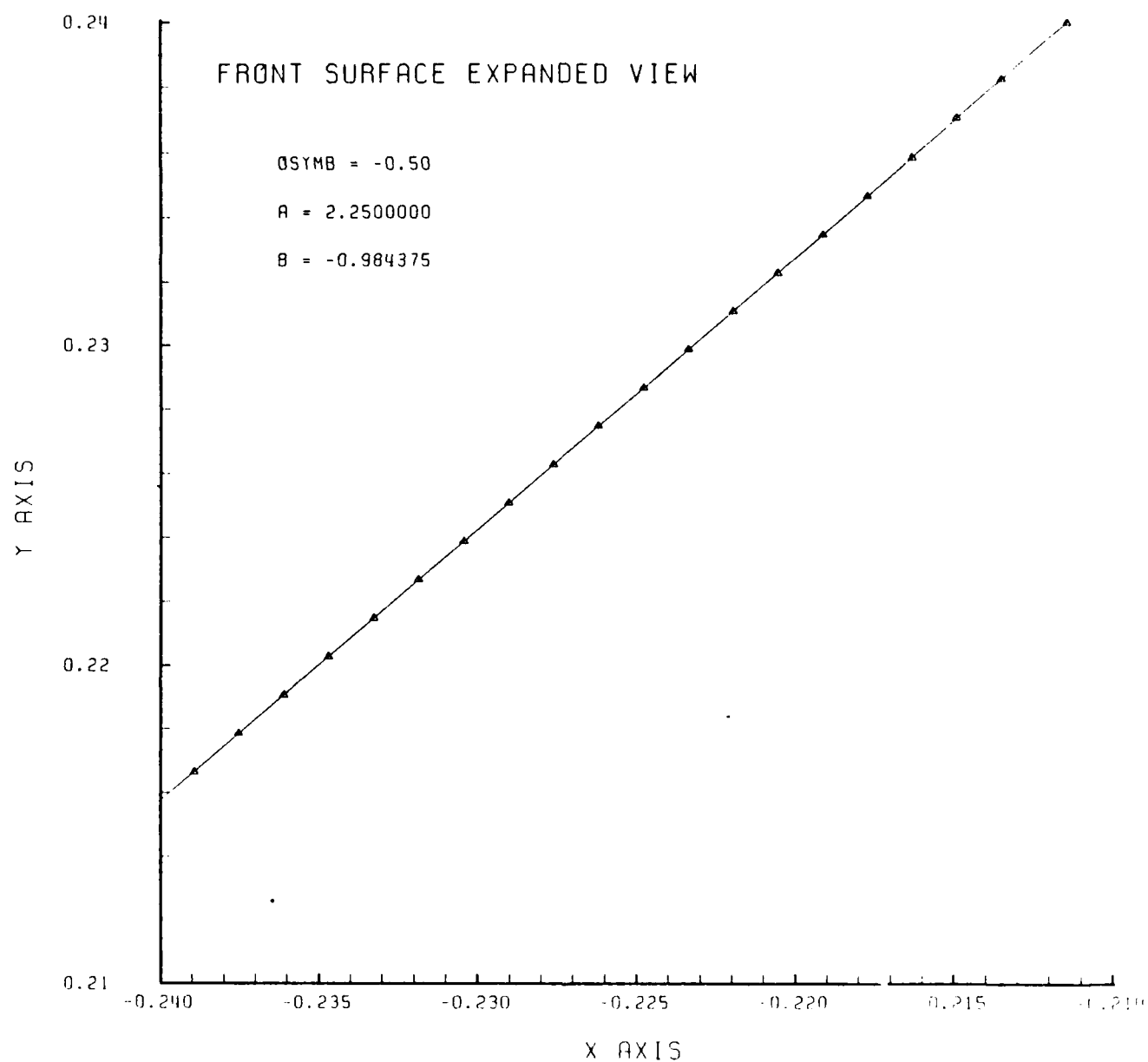


Figure 3.5 Expanded View of Front Surface -- Straight Line Method.

and

$$\arg = 1 - \frac{2(a+b')\psi_0^2 \left(\frac{r_0^2}{r^2} a+b'\right)}{a^2}$$

Again the ψ_0^2 term ensures that the expansion is valid.

A function of the type $\arcsin(x)$ where x is known to be close to the value 1 can be curve fitted into the form:

$$\sin^{-1}(x) = \pi/2 - [2(1-x)]^{1/2}$$

This equation applies to both arcsine functions as the ψ_0^2 term ensures that the argument is close to 1.

When placed in the equation the arguments result in the following final expression:

$$\theta = \theta_0 + \frac{\epsilon\psi_0}{2} \left\{ (a+b') - [(a+b')\left(\frac{r_0^2}{r^2} a+b'\right)]^{1/2} \right\}$$

In terms of radius:

$$r = r_0 \frac{(a(a+b'))^{1/2}}{\left\{ \left[\frac{a}{\epsilon\psi_0} (\theta - \theta_0) - (a+b') \right]^2 - b'(a+b') \right\}^{1/2}}$$

The reduced equation was then implemented in the iterative front surface solution scheme by constantly checking it against the complete equation. When the

difference between the two became small, the reduced equation was subsequently used in solving for the front surface. The reduced equation was then turned off when the rays had progressed an equal distance on the other side of the singularity. See Figure 3.2 for clarification.

The equation worked well in resolving the front surface for the lens construction and did produce a smooth surface similar to that shown in Figure 3.5.

It did not do well when the accuracy program check was applied. One reason is that although it was easy to implement the equation for the lens construction it was not so easy to complete the forwards trace since these rays hit the front surface randomly making it difficult to know when to use the reduced equation. In practice the reduced equation could be implemented when the absolute value of the parameter ψ_0 became smaller than that used in the lens construction phase during the time that the reduced equation was used. There was then no way of telling how accurate the front surface was, in other words, there was no method to show how the rays would physically react in the region of the singularity. Thus the reduced equation method would work but it is not an entirely satisfactory solution.

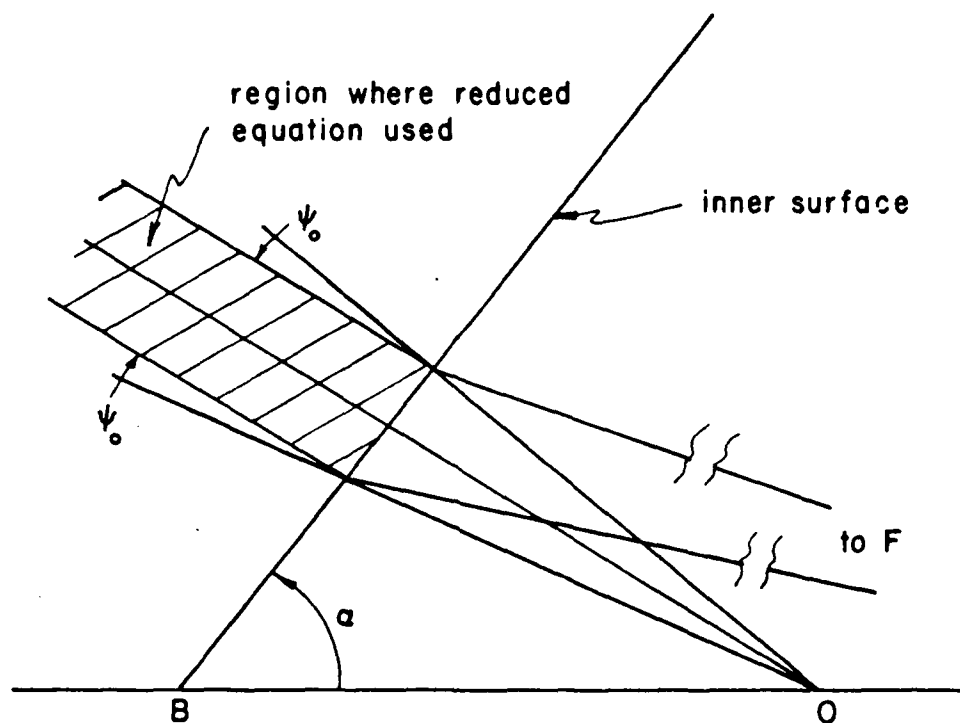


Figure 3.6 Reduced Equation Implementation.

Program CHECK was tried without using the reduced equation but disastrous results were obtained for those points in the region of the singularity.

For these reasons it was decided that another means of resolving the problem was required.

G. THIRD PROPOSED SOLUTION - MARCHAND METHOD

The third proposed solution is drawn from Marchand [Ref. 5] p. 27, in which he proposes a means of improving the ray trace formula so that nearly radial lines will be more easily described.

The method is based on redefining variables as follows:

$$\theta = eM_0 \quad (3.1)$$

where

$$M_0 = \int_{r_0}^r \frac{dr}{r(n^2 r^2 - e^2)^{1/2}}$$

Marchand describes the ray path using the parameters α and β such that:

$$\beta = \sin \theta / \sin \theta_0 \quad (3.2)$$

$$\alpha = \cos \theta - p \cos \theta_0$$

This enables the coordinates of the ray to be described as:

$$x = r \left(\alpha \frac{x_0}{r_0} + \beta \frac{p_0}{n_0} \right)$$

$$y = r \left(\frac{y_0}{r_0} + \frac{q_0}{n_0} \right)$$

$$z = r \left(\frac{z_0}{r_0} + \frac{0}{n_0} \right)$$

Upon substitution of equations 2.2 and 3.2, equation 3.1 becomes:

$$\theta = -n_0 r_0 M_0 \operatorname{sinc} \theta$$

The parameter θ can now be more accurately solved for small angles of θ using a Maclaurin series for the sinc function.

This method was then implemented in program GISL using an iteration scheme based on Cartesian rather than polar coordinates.

This method was not found to improve the accuracy of the ray trace in the region of near radial lines and was not adopted.

One reason for this may lie in the fact that in our solution to the integral in equation 2.1 the value e no longer appears at the front of the expression. Thus there is no advantage to redefining the variable as shown in equation 3.1.

H. FOURTH PROPOSED SOLUTION - ANGLE IN TERMS OF RADIUS

The fourth method and the one that proved successful involved the solution of the front surface slope and the

GRIN ray using the form of the ray trace equation shown in equation 2.3 rather than that of equation 2.4.

The reason for this choice is shown in Figure 3.7. This figure shows that when the rays are nearly radial a small change in angle will result in a huge change in radius. Thus the form of the ray trace equation using $r(\theta)$ is inherently unstable. Previously we have used this form in order to make use of the Newton-Raphson technique. It was thought that if the iteration scheme could be reworked so that the form $\theta(r)$ of the equation could be used then better results might be obtained.

A suitable iteration method was found and is illustrated in Figures 3.8 and 3.9.

The figures show that an initial point was chosen in the same manner as before, that is by drawing a straight line from the ray refracted at the inner surface out to the front surface slope. This point provides the radius for equation 2.4 so that a new angle can be calculated. This angle is then used to find the corresponding radius out to the front surface slope. This new radius is then used in equation 2.4 to find the next angle and thus an iterative loop is formed.

Thus a simple scheme was found. It should be noted however that this method will only converge when the initial angle is greater than 90 degrees. The reason for this lies in the geometry and is shown in Figure 3.10.

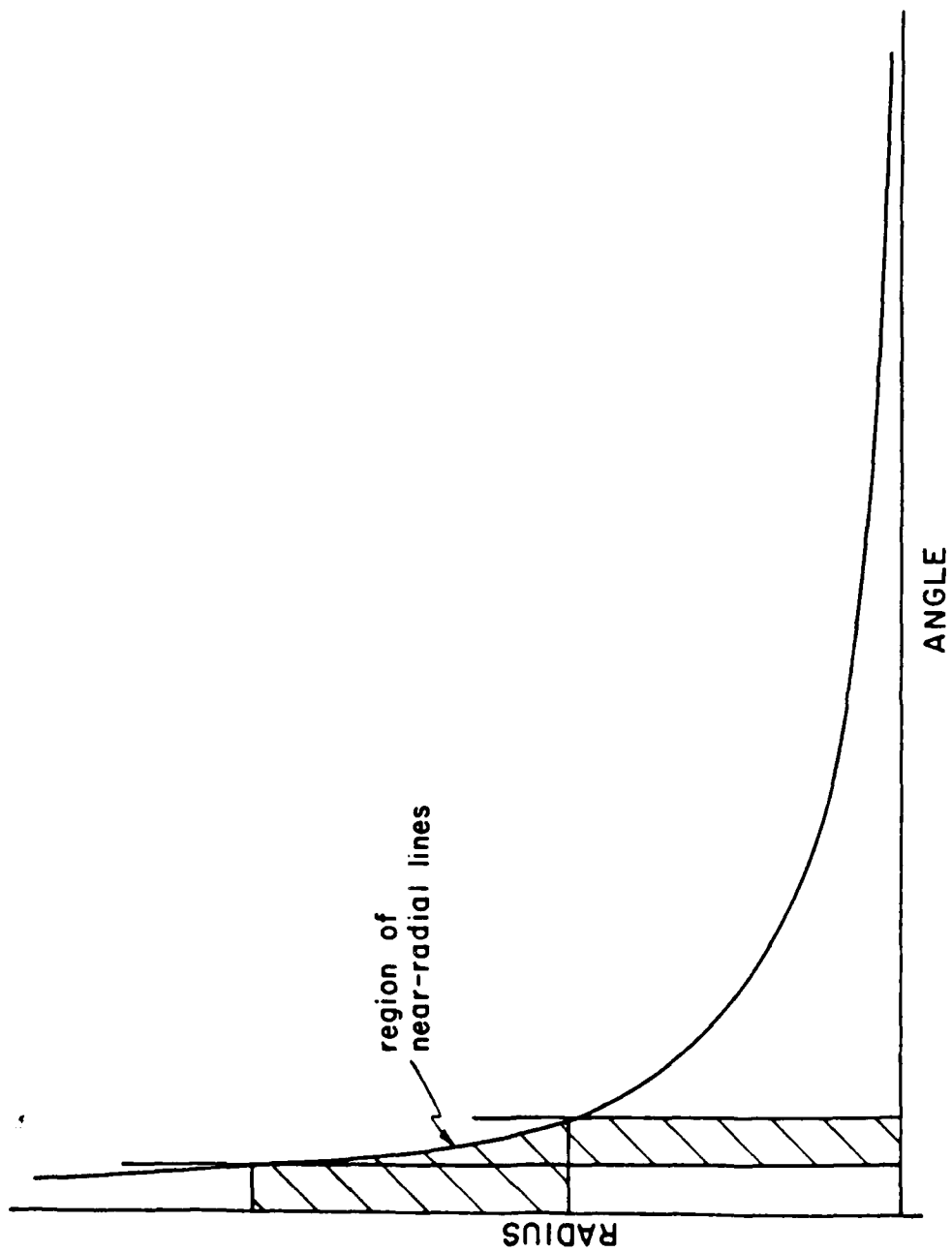


Figure 3.7 Radius and Angle Relation.

Thus the iteration will only converge when the rays are in the region of the nearly radial lines. This does not cause a difficulty since the Newton-Raphson method, modified to include ray matching, can find solutions efficiently in the region where the initial angle is less than 90 degrees.

Figure 3.10 is also drawn to show the reader an example of a forward ray trace.

This simple scheme was implemented and found to work successfully for both the lens construction and the skew ray trace; see [Ref. 8].

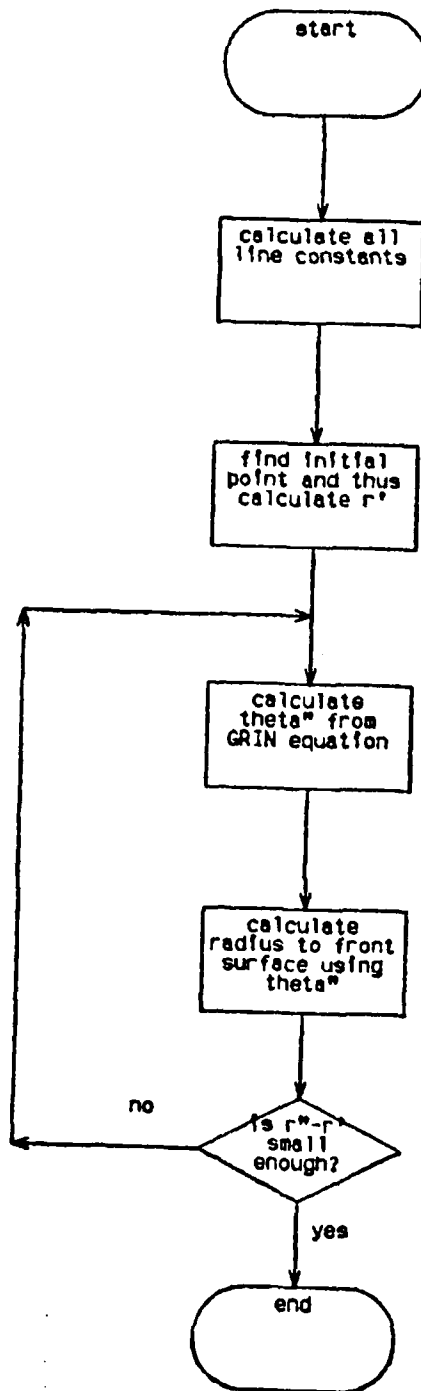


Figure 3.8 Angle in Terms of Radius Iteration Logic Flowchart.

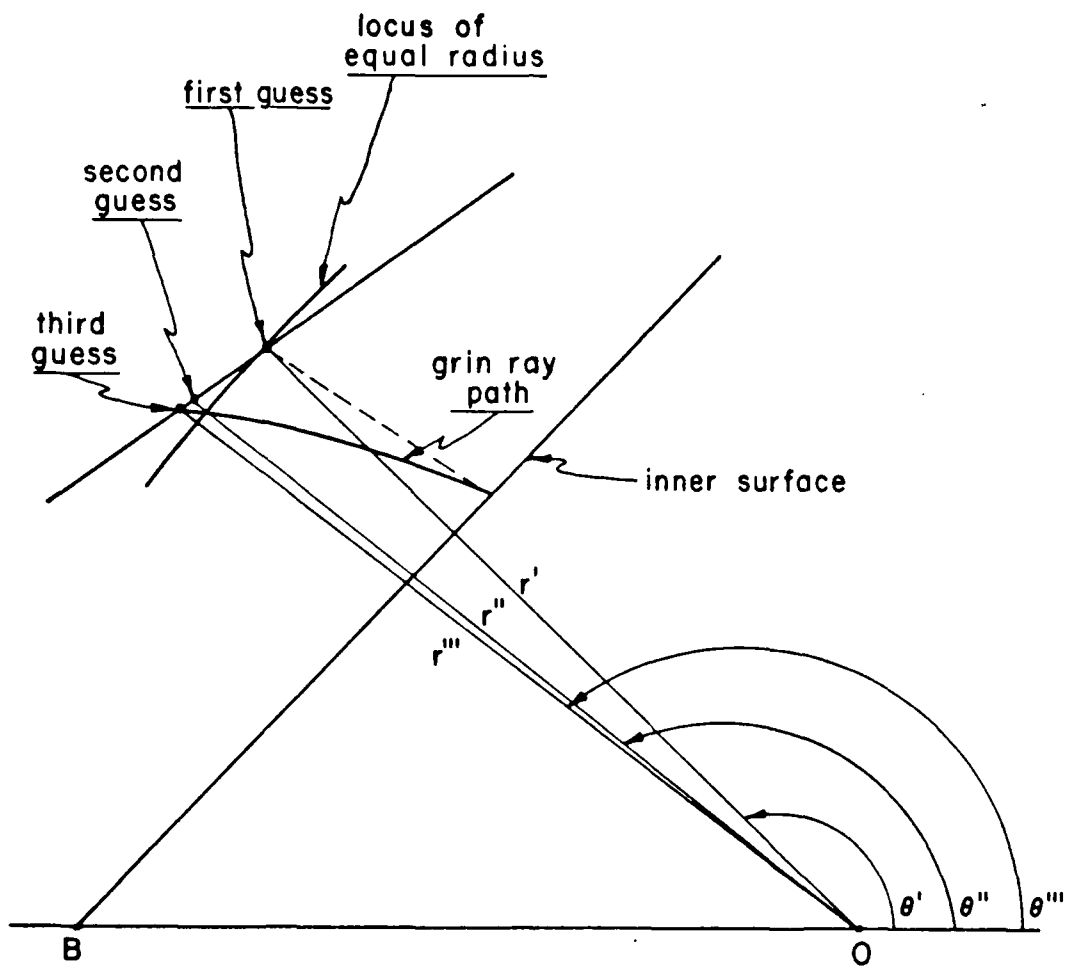


Figure 3.9 Angle in Terms of Radius Iteration Geometry.

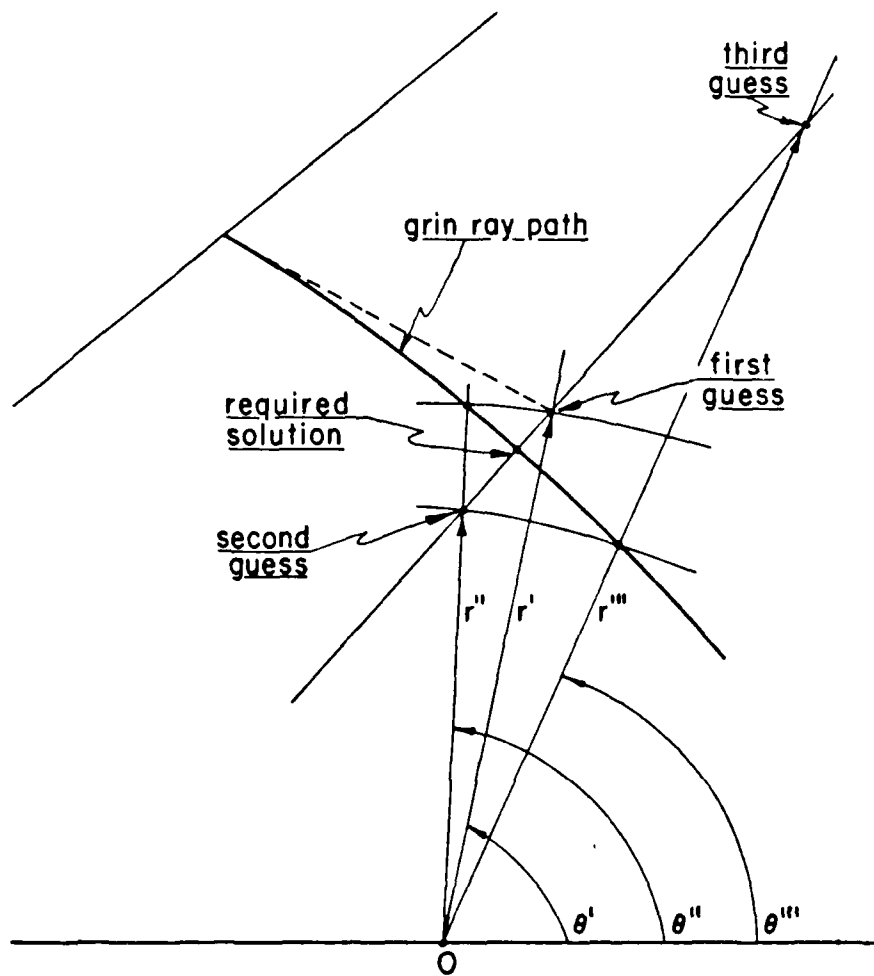


Figure 3.10 Diverging Iteration - Forward Ray Trace.

IV. GRADIENT INDEX SEEKER LENS RAY TRACE RESULTS

A. FRONT SURFACE ACCURACY

As mentioned previously, the front surface of the Gradient Index Seeker Lens was checked for accuracy by completing a ray trace from the midpoint of each set of solutions. A sample of the output from program Check is shown in Table I so that the reader will have an indication of the accuracy of the ray trace in the region of near-radial lines. Note that the most radial lines occur when the absolute value of the angle θ_0 is smallest.

B. SPOT SIZE RESULTS

The results of program GISL showing the variation of spot size are shown in Figure 4.1. This figure shows how the spot size varies as gradient strengths and orientations of the center of symmetry are changed for a median angle of incidence of 0.3 radians.

Note that all orientations of the center of symmetry are now included. The studious reader should refer to Carr's work [Ref. 4], to compare results.

C. SELECTION OF A BEST LENS

The best lens was selected by determining those parameters that would result in a small spot size and also cause the least number of rays to be reflected or refracted

outside the missile diameter. Figure 4.1 shows that these criterion were not easily discernable and a judgemental decision was required.

The reader should note that a value of 2.25 was chosen for the parameter a . This was done since the resulting value of 1.5 for the index of refraction at the center of symmetry is common to a large number of possible lens materials. Also of note is the fact that the focal point was chosen to be 2.0 units along the x-axis from the inner surface cone. This was done since this value resulted in smaller spot sizes generally.

With these criterion in mind a value of -0.2 for OB and a 5% negative gradient were chosen as parameters for the best lens design.

A more detailed study of the lens parameters could be accomplished with use of an optimization program in which a large number of parameters could be varied in order to find the best lens. This would achieve a more satisfactory result than the one shown.

TABLE I

FRONT SURFACE ACCURACY

Line Number	Angle θ_0 (Radians)	Error in y- coordinate	number of iterations
895	0.0411	0.0008	4
896	0.0390	0.0007	4
897	0.0369	0.0008	4
898	0.0349	0.0009	4
899	0.0328	0.0007	4
900	0.0308	0.0009	4
901	0.0288	0.0008	4
902	0.0268	0.0008	4
903	0.0247	0.0009	4
904	0.0227	0.0009	3
905	0.0207	0.0008	3
906	0.0187	0.0009	3
907	0.0167	0.0008	3
908	0.0147	0.0008	3
909	0.0127	0.0010	3
910	0.0107	0.0009	3
911	0.0087	0.0008	3
912	0.0067	0.0010	3
913	0.0048	0.0009	3
914	0.0028	0.0010	2
915	0.0008	0.0010	2
916	-0.0010	0.0010	2
917	-0.0030	0.0010	3
918	-0.0050	0.0010	3
919	-0.0069	0.0009	3
920	-0.0089	0.0011	3
921	-0.0108	0.0010	3
922	-0.0127	0.0010	3
923	-0.0147	0.0009	3
924	-0.0166	0.0011	3
925	-0.0185	0.0010	3
926	-0.0204	0.0011	4
927	-0.0223	0.0011	4
928	-0.0243	0.0010	4
929	-0.0261	0.0012	4
930	-0.0281	0.0011	4
931	-0.0300	0.0012	4
932	-0.0318	0.0012	4
933	-0.0337	0.0012	4
934	-0.0356	0.0011	4
935	-0.0375	0.0013	4

$$\frac{\Delta N}{N}$$

.234	.225	.206	.212	.217	.219	.221	.223	.230	.233	.243	.216	.222	.210	.295	.202	.283	.295	.274	.309	.408
+40 8.107	12.105	16.101	17.100	14.100	14.100	14.100	12.98	12.100	12.101	18.82	14.97	110.100	18.91	19.98	124.92	22.95	124.83	34.73	126.77	134.69
.220	.217	.205	.212	.216	.219	.221	.224	.226	.233	.243	.217	.224	.212	.299	.250	.288	.308	.284	.324	.422
+35 12.103	14.103	16.101	10.100	15.100	15.100	14.100	12.98	13.98	12.100	18.86	14.97	110.100	18.92	19.95	126.95	22.93	128.81	34.73	130.71	136.67
.214	.217	.219	.211	.216	.222	.222	.224	.240	.233	.234	.218	.220	.215	.219	.218	.294	.230	.316	.342	.436
+30 14.101	14.101	14.103	16.101	14.101	14.101	15.98	13.98	11.102	12.101	10.77	10.99	110.98	20.94	23.91	119.96	20.93	130.83	32.73	132.68	136.67
.220	.221	.222	.224	.215	.219	.222	.230	.238	.233	.230	.220	.217	.218	.221	.220	.333	.338	.345	.379	.455
+25 14.101	14.101	14.101	14.103	14.101	14.99	14.99	14.101	10.103	12.101	16.82	10.99	110.95	18.94	21.92	123.93	22.91	130.81	30.79	136.72	138.67
.241	.230	.226	.227	.229	.234	.235	.236	.236	.249	.231	.217	.221	.218	.234	.253	.231	.310	.268	.432	.377
+20 12.105	14.103	14.101	14.101	12.103	12.103	12.103	12.103	12.103	10.105	16.78	12.99	110.93	18.92	18.98	123.89	29.83	133.74	33.72	137.72	142.65
.245	.234	.234	.235	.235	.235	.235	.245	.245	.247	.235	.220	.215	.227	.235	.249	.242	.330	.282	.466	.365
+15 12.105	14.103	14.103	14.103	12.103	12.103	12.103	12.103	10.105	10.105	14.79	12.98	112.93	16.93	16.94	121.94	27.85	133.76	35.71	143.65	147.61
.250	.250	.250	.249	.249	.239	.248	.247	.246	.246	.233	.239	.233	.239	.235	.253	.264	.279	.301	.350	.425
+10 12.105	12.005	12.005	12.105	12.105	12.105	10.105	10.105	10.105	10.105	10.81	8.100	110.99	10.97	12.91	120.84	27.86	137.78	39.74	147.66	147.64
.256	.255	.255	.254	.254	.253	.251	.250	.247	.244	.248	.238	.234	.233	.272	.268	.300	.309	.378	.395	.499
+ 5 12.105	12.105	12.105	10.105	12.105	12.105	12.105	12.105	12.105	12.105	18.91	10.10	110.99	12.97	8.95	26.89	30.83	135.73	39.65	147.63	145.59
.262	.262	.261	.261	.260	.259	.257	.255	.251	.241	.246	.237	.230	.240	.247	.306	.340	.333	.426	.425	.588
0 12.105	12.105	12.105	12.105	12.105	12.105	12.105	12.105	12.105	12.105	12.104	9.100	19.100	11.98	12.91	120.83	22.79	130.73	39.64	139.64	137.60
.240	.239	.238	.237	.237	.236	.236	.265	.256	.238	.228	.238	.228	.230	.312	.250	.264	.385	.356	.399	.505
- 5 14.107	14.107	14.107	14.107	14.107	14.107	14.107	14.107	14.107	14.105	14.83	9.99	11.98	13.94	15.88	121.84	29.76	139.71	41.71	147.59	147.51
.240	.239	.237	.236	.240	.239	.238	.238	.238	.247	.212	.245	.229	.226	.243	.232	.266	.401	.461	.474	.579
-10 15.109	16.109	16.104	16.109	14.111	14.111	14.111	14.111	14.111	14.111	10.83	8.100	18.98	18.92	12.87	125.78	20.73	136.53	46.53	141.45	150.49
.263	.262	.261	.260	.256	.256	.256	.253	.250	.246	.243	.256	.231	.270	.328	.242	.322	.287	.342	.389	.462
-15 17.111	17.111	16.111	16.111	16.111	16.111	10.111	16.111	14.111	14.111	10.87	6.102	18.98	16.97	17.84	110.70	8.65	22.55	20.56	36.42	138.40
.297	.298	.296	.292	.289	.289	.289	.285	.280	.287	.267	.271	.236	.237	.338	.324	.371	.390	.462	.498	.515
-20 18.110	18.110	19.109	18.109	18.109	18.109	18.109	18.109	18.109	18.109	10.83	4.103	18.103	16.86	6.88	18.80	10.65	10.52	14.46	16.40	13.57
.294	.293	.291	.291	.290	.294	.294	.290	.280	.298	.266	.277	.289	.274	.334	.351	.424	.418	.463	.502	.556
-25 16.111	16.111	16.111	15.111	12.114	12.114	12.114	12.114	12.114	12.114	10.92	7.103	16.96	16.92	2.89	12.75	10.67	16.58	14.45	10.39	14.34
1.0	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	0	-.1	-.2	-.3	-.4	-.5	-.6	-.7	-.8	-.9	-.10

OSYMB

Figure 4.1 Spot Size Results - ALFAP = 0.3 Radians. Each Box Contains Spot Size, Number of Rays

Outside Missile Diameter, Number of Rays on Image Plane. Total Number of Incident Rays is 135.

Number of Rays Reflected Inside Lens Not Shown.

V. MIRROR AND DETECTOR SYSTEM

A. DESCRIPTION OF SYSTEM

A scanning mirror and a detector were added to the GRIN best lens in order to determine if precise definition of off-axis targets would be possible. The design of these two new elements is shown in Figure 5.1.

The mirror is made to 'nod' at a high rate and thus scans the spot across the detector. The detector consists of two sections, as shown in Figure 5.1. Thus, when the spot is scanned across the two plates a signal from each detector will be generated and will result in the form shown in Figure 5.2. This arrangement will allow an accurate measurement of the mirror angle for which the spot crosses the axis. The electronics of the detector signal processor could then be arranged to relate the mirror angle to the angle of incident radiation so that the precise line-of-sight angle of an off-axis target will be known.

Note that this is a simple detector system that will only resolve angles in one plane. A more sophisticated system along the same lines could be designed with a four-element detector and a more intricate method of scanning the spot across the detector. It was decided to keep the mirror and detector system as simple as possible since the

aim of this work is only to show that a precise angle measuring device is feasible.

B. LENS TO DETECTOR RAY TRACE

. This section describes the mathematics of the ray trace from the lens to the mirror and then to the detector. A listing of program DETPLOT which accomplishes this task is included as Appendix C.

The ray trace consists of the following parts:

- a. ray direction exiting lens
- b. point of intersection on mirror
- c. ray direction exiting mirror
- d. detector ray count

1. Ray Direction Exiting Lens

The ray direction exiting the lens is derived from the direction cosines of the ray as calculated by program GISL. This data is used as an input to program DETPLOT along with the coordinates of the point at which the ray left the lens.

2. Point of Intersection on Mirror

The point of intersection of the ray and mirror may be described by the following expressions:

$$x_m = D_m K + x_0 \quad (5.1)$$

$$y_m = D_m L + y_0 \quad (5.2)$$

$$z_m = D_m M + z_0 \quad (5.3)$$

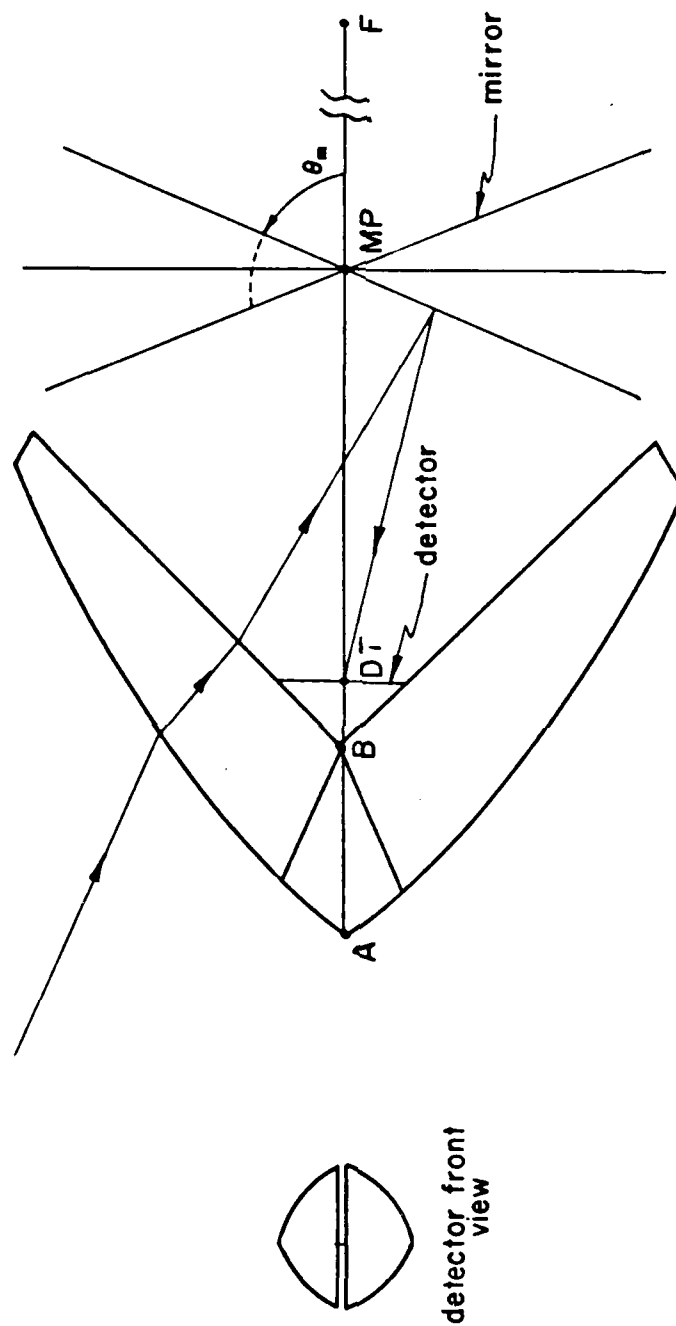


Figure 5.1 Lens, Mirror and Detector System.

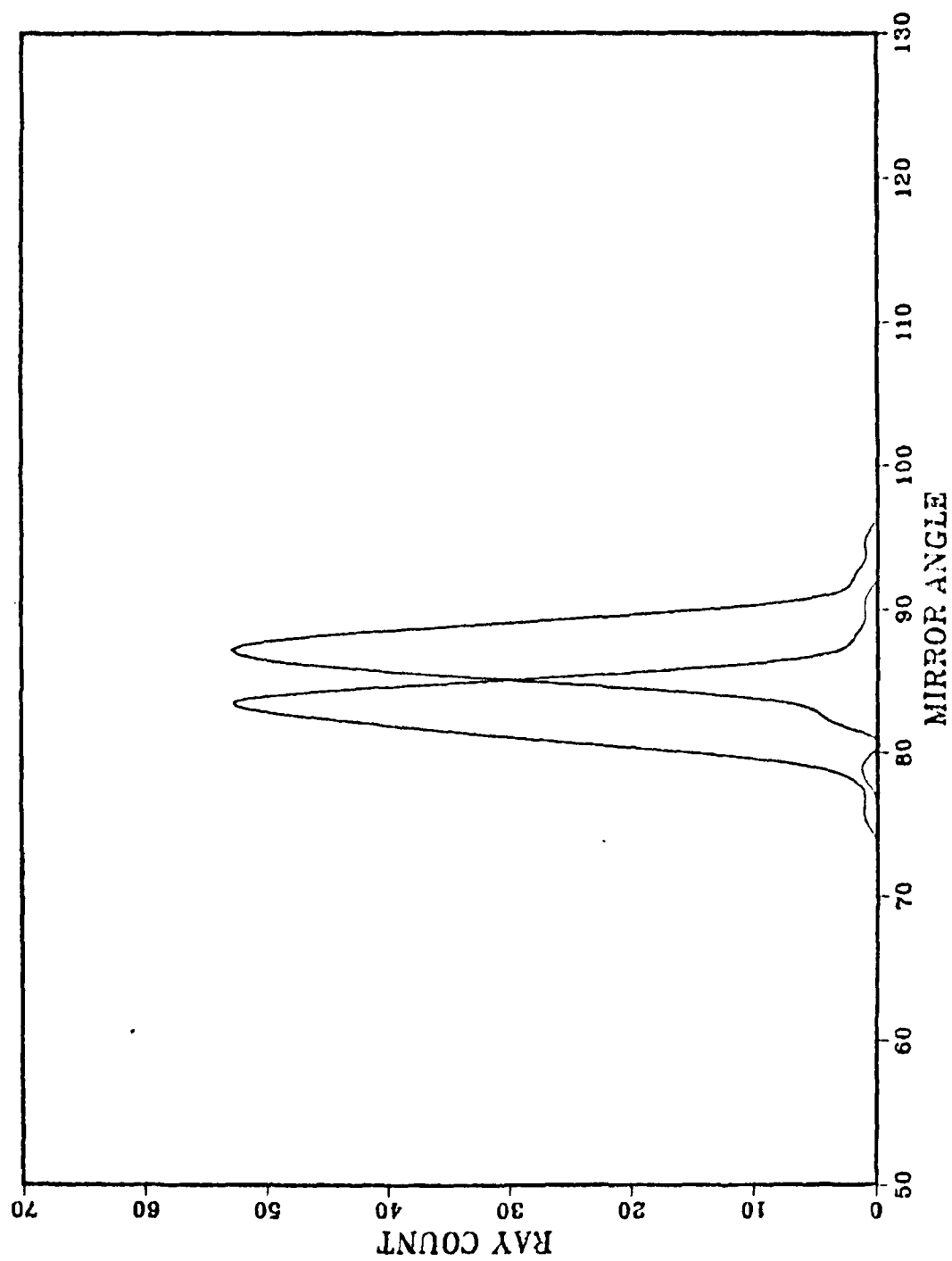


Figure 5.2 Detector Signal.

Since the surface of the mirror is tilted along the z-axis it can be described by the equations:

$$y_m = \tan \theta_m (x_m - MP) \quad (5.4)$$

$$y_m^2 \cos^2 \theta_m + z_m^2 \leq R \quad (5.5)$$

Where equation 5.4 describes a tilted plane and equation 5.5 limits the plane to the mirror diameter.

Neglecting equation 5.5 for a moment, equations 5.1, 5.2 and 5.4 may be combined to solve for D_{lm} . The result is:

$$D_{lm} = \frac{\tan \theta_m (x_0 - MP) - y_0}{L - K \tan \theta_m}$$

Now that D_{lm} is known, equations 5.1, 5.2 and 5.3 may be used to find the coordinates of the mirror intersection point. These are then checked with equation 5.5 to make sure they are not outside the mirror diameter.

3. Ray Direction Exiting Mirror

The solution of the ray direction leaving the mirror is best understood vectorially. See Figure 5.3.

Reflection off a mirror is governed by two factors. Firstly, the angles of incidence and reflection must equal. Using vectors this is expressed as:

$$\vec{n} \cdot \vec{r}_i = - \vec{n} \cdot \vec{r}_r$$

In component form:

$$n_x r_{xi} + n_y r_{yi} = - n_x r_{xr} - n_y r_{yr} \quad (5.6)$$

Secondly, the ray is reflected in the same plane as the incident ray. This may be expressed vectorially by using the cross product to generate the vector $\vec{n}_p$ as shown in Figure 5.3. Thus:

$$\vec{n}_p = \vec{r}_i \times \vec{n} = \vec{r}_r \times \vec{n}$$

In component form:

$$\vec{e}_x(-n_y r_{zi}) + \vec{e}_y(-n_x r_{zi}) + \vec{e}_z(n_y r_{xi} - n_x r_{yi}) =$$

$$\vec{e}_x(-n_y r_{zr}) + \vec{e}_y(-n_x r_{zr}) + \vec{e}_z(n_y r_{xr} - n_x r_{yr})$$

Three equations result:

$$-n_y r_{zi} = -n_y r_{zr}$$

$$-n_x r_{zi} = -n_x r_{zr}$$

$$n_y r_{xi} - n_x r_{yi} = n_y r_{xr} - n_x r_{yr} \quad (5.7)$$

The first two results are not unexpected as the mirror should not effect the z-component of the ray.

Equations 5.6 and 5.7 may now be combined to find the x- and y-components of the ray. When this is done the two following expressions result:

$$r_{xr} = \frac{(n_y^2 - n_x^2)}{(n_x^2 + n_y^2)} r_{xi} - \frac{2n_x n_y}{(n_x^2 + n_y^2)} r_{yi}$$

$$r_{yr} = \frac{(n_x^2 - n_y^2)}{(n_x^2 + n_y^2)} r_{yi} - \frac{2n_x n_y}{(n_x^2 + n_y^2)} r_{xi}$$

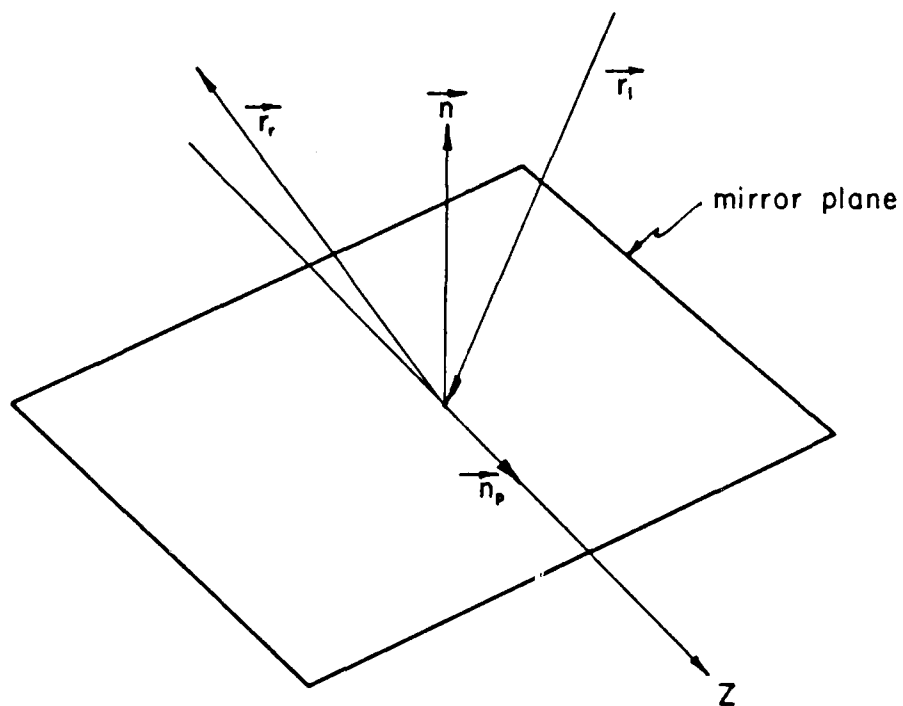


Figure 5.3 Reflection of Ray on Mirror.

Since the vector $\hat{n}$ is entirely in the x-y plane, its components may be related to mirror angle as follows:

$$n_x = -\sin \theta_m$$

$$n_y = \cos \theta_m$$

When the above equations, along with well known trigonometric identities are used the following expressions will be found.

$$r_{xr} = \cos(2\theta_m)r_{xi} + \sin(2\theta_m)r_{yi}$$

$$r_{yr} = -\cos(2\theta_m)r_{yi} + \sin(2\theta_m)r_{xi}$$

Thus these equations represent the change in direction for the exiting ray and form two of its direction cosines, the third being r_{zr} .

4. Ray Count on Detector

The coordinates of the ray at the detector are described by the expressions:

$$x_d = D_{nd} K' + x_m$$

$$y_d = D_{nd} L' + y_m$$

$$z_d = D_{nd} M' + z_m$$

Since x_d is used to define the position of the detector, it is known. Thus the value D_{nd} may be solved and the corresponding y and z -components may be found.

The ray is then checked to make sure it is within the detector diameter. If it is the y -coordinate is checked to find out if it is negative or positive, and thus

whether the ray has struck the upper or lower half of the detector. The ray count is then incremented accordingly.

Program DETPLOT traces all rays for each increment of mirror angle and thus generates plots such as that shown in Figure 5.2.

C. MIRROR AND DETECTOR SYSTEM RESULTS

Appendix D shows the results of the detector signals as a function of mirror angle at angles of incidence from 0 to 40 degrees. For these the mirror pivot point was placed at 1.4 units along the x-axis from the inner surface cone since this was the smallest distance that would still allow full movement of the scanning mirror. The focal length was adjusted to 2.8 units for the lens design calculations so that the spot would be imaged as small as possible on the detector. Figure 5.4 summarizes the results shown in Appendix D. As can be seen, the smaller angles of incidence (those in the range 0 to 16 degrees) result in a linear relation between angle of incidence and mirror tilt angle, corresponding to an on-axis spot. The larger values of the angle of incidence resulted in a more random relation between these two quantities.

Appendix D also shows that the error in determining mirror angle increases as the angle of incidence increases. This is due to the fact that each successive plot is less distinct and thus a precise measure of the angle becomes

more difficult. Although this effect shows a deterioration in the lens system performance, it is not harmful since, as the missile corrects itself, the measurement of angle will become more accurate.

For these reasons this simple system is considered to be a viable means of target detection and shows that the GISL can be a successful missile seeker lens.

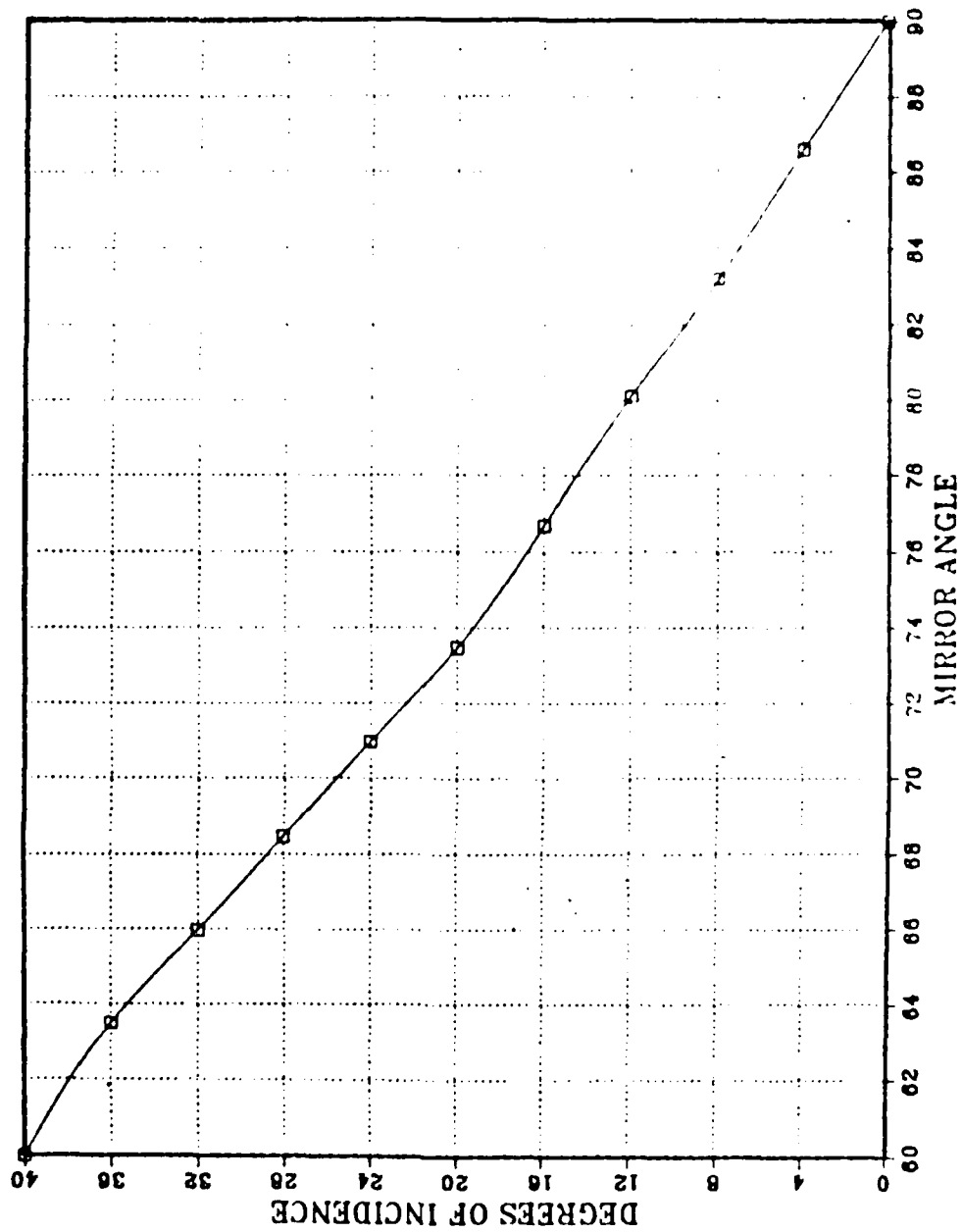


Figure 5.4 Mirror Tilt for On-Axis Spot and Angle of Incidence.

VI. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

Two objectives were accomplished in the course of this study. Firstly, the Gradient Index Seeker Lens design, as initiated by Carr [Ref. 4], was completed in that all orientations of the center of symmetry with respect to the lens were made possible to describe mathematically. The results obtained for the spot size on the image plane showed trends that indicated the parameters that would produce a 'best' lens, but precise definition of these values was not possible.

Secondly, it was shown that a scanning mirror and detector system applied to the lens could be described using ray-trace techniques and also that such a system could be used to provide precise angle measurements of off-axis targets at angles less than 16 degrees. For larger values of angle, error in definition became increasingly greater.

Specifically, then, the following conclusions are made:

a) The Gradient Index Seeker Lens can be described using ray-trace techniques and can be shown to provide adequate imaging of off-axis objects.

b) A scanning mirror and detector system can be used to provide precise angle measurements.

B. RECOMMENDATIONS

The following recommendations for further study are made:

- a) An optimization program should be applied to program GISL to study the variations of all important parameters.
- b) A detector system capable of detection in all directions should be designed.
- c) The manner in which signal processing of the detector signals can be carried out should be studied.
- d) A study of Gradient Index material and the methods by which a lens could be manufactured should be undertaken.

APPENDIX A

```

*****
**          GRADIENT INDEX SEEKER LENS
**          /
*****
THIS PROGRAM DESIGNS THE GRADIENT INDEX SEEKER LENS
IN TWO STAGES. FIRST THE FRONT SURFACE IS CONSTRUCTED
BY TRACING RAYS FROM THE FOCAL POINT BACK THROUGH THE
LENS DEFINING THE FRONT SURFACE SUCH THAT THE RAYS EMERGE
PARALLEL. SECOND A SKEW RAY TRACE IS COMPLETED IN A
FORWARD DIRECTION FOR ANY ANGLE OF INCIDENCE. THE SIZE
OF THE RESULTING SPOT PATTERN IS RECORDED. THAT
PERFORMANCE MEASUREMENTS MAY BE MADE.
THIS PROGRAM WAS INITIALLY WRITTEN BY H. CARR AND
THEN MODIFIED BY D. DAVIDSON
*****
SPECIFICATION STATEMENTS
*****
INTEGER I,J,K,L,G,F,M,N,P,FLAG(1000),W,RAYS,S,DRAYS,ELLNUM,
INUMBER,CCORD,ENUM,NUMB,IANGLE,THICK,CG,HF,COUNT,SHAPE,GGT,
ICCCR(800),NUMBT,ELLIPS,SQUARAY,TRIP,FILTER,CCORT,
IFACE,FLUG,COUNT2
REAL DVLCN(1010),DYDXT(1010),IP(1010),GAMMA,AB,ONP,
IT,ALFAF,GRIC,RAYZ(1000),RAYZ(1000),CENI,TORBI,BOTI,
IRADIUS,CEN,TOP,BCT,TURB,DELTA,DELTA,ST,RATIO,DYDXNP,LK,LL
REAL NPI,NPJ,NPK,APK,BO,CO,DU,EC,PAR1,PAR2,NINCTY(1000),
LELL,ELY(17,1000),ELZ(17,1000),ELZ2,LM,ZOYC,
IPAR3,XCIAPT(1000),YCIAPT(1000),SUM1,
ISUM2,SUM3,SIGMAZ(10,800),SIGMAV(10,800),RMSRAD(10,800)
REAL RCC(500),SUM4(500),FRACTN(500),SDRAD(1000),YCENTK(20,800),X1PP
1THETA,E,EPSCN,P,SI,XIP,YIP,XIHP,YIHP,ERRCR,XF,YF,ZP,RH,D2T,ABS,X1PP
REAL Y1FPP,ZETAP,BASE,BAS1,ROD0,N2X1K,N2X2K,
1PRCNTE,FRCNTT,PRCNTC,SURFL,TSURFL,SLINCR,ARKLOS,CHEKRO,CALIBK,
1SIGN,DIFFOR,XMAVE,XMAVET
DOUBLE PRECISION A,E,U,F,R,ALPHA,BETA,PI,PI2,
IX1(1010),Y1(1010),I2(1010),CSYMB,X2(1010),Y2(1010),X1H,
IY1H,I2P(1010),UP(1010),N1,N2,N20,N3,E,BF,RO,RZERO,
IDLUDP,ARG,RSINO,C,DIFF,PSI0,PSIP,ZETA,EXIT,THEIAO,THEIP,THEIAH,
IRAC,RACC,RDGG,RDGC,RCF,RDT,EUNR,THEIAP,THEIAT,RSIN,
IBX,SLI,RPF,CHECK1,CHECK2,KADH,RODCTS,RF,THIAP,BEF,F,SQKE,N2X21,
1XX,AR
DOUBLE PRECISION XC,YO,ZO,RO,PHI,PHIP,CK,CL,CKP,CLP,CMP,LKKF,
ICLLP,CMP,ROX,ROY,ROZ,NP0,NP0Y,NP0Z,XX,CC,CCM,THETA1,
1THETA2,NUM,NUM1,NUM2,NUM3,NUM4,NUM5,NUM6,NUM7,NUM8,LSYMPH,

```

```

1CPHX,OFFY,CPHZ,XINT,XINTD,BETA0,ALPHA0,XI,YI,ZI,AL,BL,CL,DP,
1LKP,LLP,LMP,KKLLMM,THETA1,PHI1,PHI2,TIRA,CKPP,CLPP,CMPP
1D1,C2,C3,XIF,YIH,ZIF,OPL(1000),YIM(1000),ZIM(1000),DIM(1000)

```

MAJOR VARIABLE DEFINITIONS

```

A      : STRENGTH OF INDEX CCNSTANT
B      : STRENGTH OF INDEX GRADIENT
F      : FOCAL LENGTH
ALPHA  : INSIDE SURFACE CONE HALF-ANGLE
U      : INCIDENT RAY OFFSET ANGLE
Y      : NUMBER OF ITERATIONS IN LENS CONSTRUCTION
R      : MAXIMUM INSIDE SURFACE RADIALS FROM X-AXIS
       : NOTE THAT ALL DIMENSIONS REFERENCED TO THIS VALUE
T      : THICKNESS OF LENS AT EDGE
N1,N3  : INDICES OF REFRACTION CLT SIDE LENS
CSYMB  : DISTANCE FROM CENTRE OF SYMMETRY TO INSIDE SURFACE
       : ALONG X-AXIS. POSITIVE IF 0 TO LEFT OF B.
ALFAP  : ANGLE OF INCIDENT RADIATION FOR SKEW RAY TRACE
GRID   : GRID INCREMENT, DEFINES NUMBER OF SKEW RAYS
ENUM    : NUMBER OF ELLIPSES. MUST BE INTEGER AND ODD
ELL     : Z-COORDINATE INCREMENT FOR ELLIPSE PLUTS

```

OUTPUT OPTIONS:

```

TO HAVE LENS SHAPE DATA PRINTED SET "SHAPE" TO 1. OTHERWISE
SET "SHAPE"=0.
SHAPE=C

```

```

TO HAVE SKEW RAY AND MIRROR IMAGE SKEW RAY DATA PRINTED SET
"SQLRAY" TO 1. ELSE SET "SQLRAY" TO 0. (INTEGER)
SQLRAY=C

```

```

TO HAVE ELLIPSE SHAPE DATA PRINTED SET "ELLIPS" TO 1. OTHERWISE
SET "ELLIPS"=0.
ELLIPS=C

```

SET CONSTANTS

```

F=4.0
ALPHA=C.7853982
U=0.0000000
I=1.008
R=1.0
T=0.05
N1=1.0
N3=1.0

```

```

GIS0C490
GIS0C500
GIS0C510
GIS0C520
GIS0C530
GIS0C540
GIS0C550
GIS0C560
GIS0C570
GIS0C580
GIS0C590
GIS0C600
GIS0C610
GIS0C620
GIS0C630
GIS0C640
GIS0C650
GIS0C660
GIS0C670
GIS0C680
GIS0C690
GIS0C700
GIS0C710
GIS0C720
GIS0C730
GIS0C740
GIS0C750
GIS0C760
GIS0C770
GIS0C780
GIS0C790
GIS0C800
GIS0C810
GIS0C820
GIS0C830
GIS0C840
GIS0C850
GIS0C860
GIS0C870
GIS0C880
GIS0C890
GIS0C900
GIS0C910
GIS0C920
GIS0C930
GIS0C940
GIS0C950
GIS0C960

```



```

THETAO=LATAN(Y2(J)/BASE)
IF (THETAO .LT. 0.0) THETAO=PI+THETAO
ED=T/R
X1H=X2(J)-ED*DCOS(LP(J))
Y1H=Y2(J)+ED*DSIN(UP(J))
BAS1=X1H+OSYMB
IF (BAS1 .EQ. 0.0) OSYMB=1.0000001*OSYMB
BAS1=X1H+CSYMB
THETAE=LATAN(Y1H/BAS1)
IF (THETAE .LT. 0.0) THEAE=PI+THEAE
PSIO=PI-LP(J)-THEAO
EPSLON=+1.0
IF (PSIO .GT. PI2-OR-PSIO .LT. 0.0) EPSLON=-1.0
E=N2*RC*DSIN(PSIO)
ARG=CSQRT(A**2+4*B*(E**2)/(RZERO**2))
RSINC=DARSIN(2*(E**2)/(RO**2)-A)/ARG)
C=PI2-RSINO
IF (PSIO .LT. PI2) C=0.0
TCRIT=THEAO-(EPSLON/2.)*(PI2-RSINO)
IF (THEAE .GE. TCRIT) THETE=THEAE-C
IF (THEAE .LT. TCRIT) THETE=THEAE
IF (THEAE .GE. TCRIT) EPSLON=+1.0
THETE=THETE-THEAO

FIND RADIUS CORRESPONDING TO ANGLE SPECIFIED BY EDGE THICKNESS
NOTE THAT NO ITERATION REQUIRED FOR FIRST RAY

CALL RADIUS(A, E, ARG, RSINO, THETE, RAD, RDCIG, FILTER, EPSLON)
IF (FILTER .EQ. 1) WRITE(6,1800)
IF (FILTER .EQ. 1) GC TO 82

CALCULATE COORDINATES, ANGLE AND SLCFE FOR FRONT SURFACE SOLUTION

X1(J)=RAD*CS(THEAE)-CSYMB
Y1(J)=RAD*DSIN(THEAE)
CALL INCEX(RAD, N2, A, B, RZERO)
PSI=DARSIN(E/(N2*RAD))
IF (THEAE .LT. TCRIT) PSI=PI-PSI
ZETA=PI-PSI-THEAE
I1(J)=DARSIN(DSQRT(((CSIN(ZETA-U))**2)/((DCCS(ZETA-U)-(N1/N2))**2+
1/((CSIN(ZETA-U))**2))))
I1P(J)=LARSIN((N1/N2)*CSIN(I1(J)))
DYDXN(J)=-DIAN(I1(J)+U)
DYDXT(J)=DCCTAN(I1(J)+U)
IF (SHAPE .EQ. 0) GC TO 5
WRITE(6,200) J, X1(J), Y1(J), X2(J), Y2(J), WOP(J), I2P(J), I2(J), UP(J),
I1P(J), I1(J), DYDXT(J)
5 CONTINUE

```

C C C

C C


```

7      X1(J)=RAD*DCOS(THETAP)-OSYMB
      Y1(J)=RAD*DSIN(THETAP)
      CALL INDEX(RAD,N2,A,B,RZERO)
      EONR=E/(N2*RAD)
      IF(ECNR.GT.1.0) WRITE(6,9978)ECNR
      FORMAT(1CX,'EONR =',F19.16)
      IF(ECNR.GT.1.0)ECNR=1.0
      PSIP=DARSIN(EONR)
      IF(THETAP.LT.ICRIT.AND.PSIO.GT.0.0) PSIP=PI-PSIP
      ZETA=PI-PSIP-THETAP
      GO TO 5
C
5      CONTINUE
      X1(J)=X1H
      Y1(J)=Y1H
      ZETA=LF(J)
C
5      CONTINUE
      LE=0.0 GO TO 11
      IF(Y1(J).LE.0.0) GO TO 11
      I1(J)=DARSIN(DSIN(ZETA-U))**2/((DCOS(ZETA-U)-(N1/N2))**2)
      I1(J)=DARSIN(ZETA-U)**2)
      IIP(J)=DARSIN((N1/N2)*DSIN(I1(J)))
      CYDXN(J)=(-DIAN(I1(J)+U))
      DYDXI(J)=DCOTAN(I1(J)+U)
      SLINCF=CSQRT((X1(J)-X1(L))**2+(Y1(L)-Y1(J))**2)
      SURFL=CSQRT(SLINCF)
      IF(SHAPE.EQ.0) GO TO 10
      WRITE(6,200)J,X1(J),Y1(J),UDF(J),I2P(J),I2(J),
      UP(J),IIP(J),I1(J),DYDXI(J)
      GO TO 12
10 CONTINUE
      GO TO 12
      END CF TRACE FOR FRONT SURFACE CONSTRUCION
      CPAQUE NCSE SECTION CALCULATIONS
C
11 CONTINUE
      K=J-1
12 CONTINUE
      AB=(-X1(K))+Y1(K))/DYDXI(K)
      STATNA=-AB
      GAMMA=ATAN(CYDXI(K))
      CNP=(Y1(K))/SIN(GAMMA)
      TSURFL=CNP+SURFL
      WRITE(6,300) GAMMA,STATNA,CNP,SURFL,TSURFL
C
      WRITE OUTPUT FILE FOR PROGRAM CHECK
C

```



```

      TOP1=CEN1+TORBI
      IF (RAYY(G) .GE. TOP1) GO TO 30
      RAYZ(G)=RAYZ(G)+GRID
      GC TO 20
30 CONTINUE
C SEARCH FOR Y1(J) JUST GT RAYZ(G); RETRIEVE X1(J)
  N=K/2
  IF (Y1(M) .LT. RAYZ(G)) GO TO 32
  IF (RAYY(G) .GT. 1.15) GO TO 32
  N=N-1
31 CONTINUE
  IF (Y1(N) .GT. RAYZ(G)) GO TO 36
  N=N-1
  IF (N .EQ. 0) GC TC 34
  GC TC 31
32 CONTINUE
  N=N
33 CONTINUE
  IF (Y1(N) .GT. RAYZ(G)) GO TO 36
  N=N-1
  IF (N .EQ. 0) GO TO 34
  GC TC 33
34 CONTINUE
  RAYZ(G)=0.0
  IF (RAYY(G) .LT. 0.0) GC TO 35
  IF RAYY(G)=RAYY(G)+GRID
  GC TC 20
35 CONTINUE
  RAYY(G)=RAYY(G)-GRID
  GC TO 20
36 CONTINUE
  RADIUS=LSQRT(RAYZ(G)**2+((RAYY(G)/CCS(ALFAP))-(X1(N)+AB))*
  1TAN(ALFAP))**2)
  IF (Y1(N) .GT. RADIUS) GO TC 40
  IF (RADIUS .GT. Y1(1)) GO TO 37
  N=N-1
  GC TC 36
37 CONTINUE
  N=N-1
  IF (N .GT. 0) GO TO 36
  CEN=(X1(1)+AB)*SIN(ALFAP)
  TURB=Y1(1)*COS(ALFAP)
  RAYZ(G)=0.0
  IF (RAYY(G) .GE. 0.0) GC TO 38
  IF RAYY(G)=RAYY(G)-GRID
  ECT=CEN-TURB
  CHECK FOR BOTTOM EDGE OF LENS, GO TO MIRROR IMAGE RAYS
  IF (RAYY(G) .LT. BCT) GO TO 60

```

G1S03650
 G1S03860
 G1S03870
 G1S03880
 G1S03890
 G1S03900
 G1S03910
 G1S03920
 G1S03930
 G1S03940
 G1S03950
 G1S03960
 G1S03970
 G1S03980
 G1S03990
 G1S04000
 G1S04010
 G1S04020
 G1S04030
 G1S04040
 G1S04050
 G1S04060
 G1S04070
 G1S04080
 G1S04090
 G1S04100
 G1S04110
 G1S04120
 G1S04130
 G1S04140
 G1S04150
 G1S04160
 G1S04170
 G1S04180
 G1S04190
 G1S04200
 G1S04210
 G1S04220
 G1S04230
 G1S04240
 G1S04250
 G1S04260
 G1S04270
 G1S04280
 G1S04290
 G1S04300
 G1S04310
 G1S04320

```

38 GO TO 20
CONTINUE
RAYY(G)=RAYY(G)+GRID
TUP=CEN+IGRE
IF (RAYY(G).GT.TOP) GO TO 39
GO TO 20
39 CONTINUE
RAYY(G)=-GRID
GO TO 20
40 CONTINUE
P=N+1
C
C
CALCULATE INTERCEPT PCINT ON OUTER SURFACE
AU=RAYY(G)/COS(ALFAP)
BU=TAN(ALFAP)
CO=(Y1(N)-Y1(P))/(X1(N)-X1(P))
DO=X1(P)
EU=Y1(P)
PAR1=(AC*CO*EQ-(CO**2)*DU-AB*(BC**2))
PAR2=(AC**2-CO**2)
PAR3=(AC**2+AB*(BO**2)*AB-2.0*AC*BC*AB+
1RAYZ(G)**2-(EU-CO*CO)**2)
XO=(PAR1/PAR2)+SQRT((PAR1/PAR2)**2-PAR3/PAR2)
IF ((PAR1/PAR2).GT.X1(1))XO=XO-2.0*SQRT((PAR1/PAR2)
YO=(RAYY(G)/COS(ALFAP))-(XO+AB)*TAN(ALFAP)
ZO=RAYZ(G)
C
C
CALCULATE THE DIRECTION COSINES OF OUTSIDE SURFACE NORMAL
DELTAX=X1(N)-X1(P)
DELTAY=Y1(N)-Y1(P)
SP=DSQRT((RADIUS-Y1(P))**2+(XO-X1(P))**2)
ST=SQRT(DELTAX**2+DELTAY**2)
RATIO=SP/ST
DYDXNP=RATIO*(DYDXN(N)-DYDXN(P))+DYDXN(P)
NP1=1.0/LYDXNP
IF (YO.EQ.C.O) GO TO 41
ZCYG=(ZO/YO)
IF (ZCYG.GE.8235000) GO TO 41
IF (ZCYG.LE.-8235000) GO TO 42
NPJ=CCS(DATAN(ZO/YO))
NPK=DSIN(DATAN(ZO/YC))
GO TO 43
41 CONTINUE
NPJ=0.0
NPK=1.0
GO TO 43

```

GISO4330
 GISO4340
 GISO4350
 GISO4360
 GISO4370
 GISO4380
 GISO4390
 GISO4400
 GISO4410
 GISO4420
 GISO4430
 GISO4440
 GISO4450
 GISO4460
 GISO4470
 GISO4480
 GISO4490
 GISO4500
 GISO4510
 GISO4520
 GISO4530
 GISO4540
 GISO4550
 GISO4560
 GISO4570
 GISO4580
 GISO4590
 GISO4600
 GISO4610
 GISO4620
 GISO4630
 GISO4640
 GISO4650
 GISO4660
 GISO4670
 GISO4680
 GISO4690
 GISO4700
 GISO4710
 GISO4720
 GISO4730
 GISO4740
 GISO4750
 GISO4760
 GISO4770
 GISO4780
 GISO4790
 GISO4800


```

C C
INTERNAL RAY LENGTH AND INSIDE SURFACE INTERCEPT
ROX=(XC+OSYMB)/RO
ROY=YU/FQ
ROZ=ZO/FC
NPCX=RCY*CMP-ROZ*CLP
NPCY=RCZ*CKP-ROY*CMP
NPCZ=RCX*CLP-ROX*CKP
NPO=DSQRT(NFOX**2+NPOY**2+NPOZ**2)
NPCX=NFCX/NPO
NPCY=NFCY/NPO
NPOZ=NFCZ/NPO

C C
CALCULATE PSIO AND THEN DETERMINE ITS SIGN
PSIO=DARCGS(ROX*CKP+ROY*CLP+ROZ*CMP)
IF (NPOZ.EQ.0.0) XX=CSYMB*RCZ/(ROX*NPOY)*(NPCY*ROY/ROZ+NPOZ)
IF (NPOZ.EQ.0.0) GO TO 121
XX=OSYMB*ROY/(ROX*NPOZ)*(NPOZ*ROZ/RCY+NFOY)
121 CONTINUE
CCCK=OSYMB/DSQRT(OSYMB**2+XX**2)
CCL=0.0
CCM=XX/DSQRT(OSYMB**2+XX**2)
IF (NPOZ.EQ.0.0) CCL=XX/DSQRT(OSYMB**2+XX**2)
IF (NPOZ.EQ.0.0) CCM=0.0
THETA1=PI-DARCGS(CCK*CKP+CCL*CLP+CCM*CMP)
THETA2=LARGCS(CCK*RCX+CCL*ROY+CCM*ROZ)
FLUG=0
IF (THETA1.GT.THETA2.AND.OSYMB.LT.0.0) FLUG=1
IF (FLUG.EQ.1) PSIO=-PSIO

C C
CALCULATE LINE CONSTANTS
EPSLON=+1.0
IF (PSIG.LT.PI.AND.PSIO.GT.PI2) EPSLON=-1.0
E=N20*RC*CSIN(PSIO)
ARG=DSQRT(A**2+B**2/RZERO**2)
RSINO=CAR SIN(2*E**2/RO**2-ARG)
C=PI2-RSINO
IF (PSIG.LT.PI2) C=0.0
TCRIT=-EPSLON/2*(PI2-RSINO)
NUM1=(CCLP*YC+CMP**2-C*CKP**2)*(DTAN(ALPHA)**2)
NUM2=(CCLP**2+CMP**2-(CKP**2)*(DTAN(ALPHA)**2))
NUM3=(YC**2+ZO**2-(XO**2)*(DTAN(ALPHA)**2))
NUM4=NCLP1**2-NUM2*NUM3

C C
CHECK FOR INTERCEPT WITH INSIDE SURFACE

```

```

GIS051290
GIS051300
GIS051310
GIS051320
GIS051330
GIS051340
GIS051350
GIS051360
GIS051370
GIS051380
GIS051390
GIS051400
GIS051410
GIS051420
GIS051430
GIS051440
GIS051450
GIS051460
GIS051470
GIS051480
GIS051490
GIS051500
GIS051510
GIS051520
GIS051530
GIS051540
GIS051550
GIS051560
GIS051570
GIS051580
GIS051590
GIS051600
GIS051610
GIS051620
GIS051630
GIS051640
GIS051650
GIS051660
GIS051670
GIS051680
GIS051690
GIS051700
GIS051710
GIS051720
GIS051730
GIS051740
GIS051750
GIS051760

```

```
C
C
C
C
IF(NUM4 .LT. 0.0) GC TC 50
CALCULATE FIRST GUESS POINT
D2=-NUM1/NUM2-DSQRT((NUM4)/NUM2
XIH=C2*CKP+XD
YIH=C2*CLP+YC
ZIH=C2*CMP+ZO
IF(ZIH .LT. 0.0) GC TO 50
OSYMFH=DSQRT((CSYMB*XIH)**2+YIH**2+ZIH**2)
OPHX=(CSYMB+XIHF)/CSYMFH
OPHY=YIH/OSYMFH
OPFZ=ZIH/O SYMFH
THETAP=LAR COS(ROX*CPHX+ROY*OPHY+ROZ*OPHZ)
RP=CSYMFH
IF(E.EQ.O.0) GC TO 124
SIGN=-1.0
IF(CSYMB.GT. 0.0) SIGN=+1.0
IF(FSIO.LT.0.0) THETAP=-THETAP
INITIALIZE SURF ROUTINE TO FIND CORRECT VALLE GF SIGN
CALL SURF(THETAP,RACD,RDOTS,SIGN,FILTER,NPOX,NPOY,NPOZ,PSIO,
1XO,YO,ZC,OSYMB,RO,CKKP,CLLP,CMP,ALPHA)
IF(RACD .LT. 0.0) SIGN=-SIGN
THETP=PI-THETA+THETAP
CALL NEWTON-RAPHSON ITERATION IF ANGLE LARGE ENOUGH
IF(LABS(PSEO).GT.3.0.AND.OSYMB.LT.0.0)
1 CALL INTER2(RP,THETAP,RAU,THETAP,A,E,ARG,RSIND,EPSLON,SIGN,
1 NPOX,NPOY,NPOZ,PSIO,XO,YO,ZC,OSYMB,RO,CKKP,CLLP,CMP,
1 ALPHA,THEFA,FILTER,FLUG)
IF(LABS(PSEO).GT.3.0.AND.OSYMB.LT.0.0) GO TO 118
CALL ANGLE IN TERMS GF RADIUS ROUTINE IN ANGLE SMALL ENOUGH
CALL INTER1(SIGN,THETAP,RAC,THETAP,FILTER,A,E,ARG,RSINU,
1EPSLON,C,ICRIT,NPOX,NPOY,NPOZ,PSIO,XO,YC,ZO,RO,CKKP,CLLP,CMP,
1ALPHA,CSYMB,FLUG)
118 CCONTINUE
CHECK FOR FAILED RAYS
IF(FILTER .EQ. 2.CR.FILTER .EQ. 1) GO TO 51
IF(FILTER .EQ. 3) GO TO 50
IF(FILTER .EQ. 4) GC TC 50
C
C
C
```

```

C          GU TC 123
          THTAP=THETAP
          RAC=RP
C          CALCULATE ALL REQUIRED PARAMETERS AT SOLUTION POINT
C          123
          BETAO=D SIN(THTAP)/D SIN(PSIO)
          ALFAO=D COS(THTAP)-BETAO*D COS(PSIO)
          XI=RAO*(ALFAO*((XG+USYMB)/RO)+BETAO*CKKP)-USYMB
          YI=RAO*(ALFAO*(YO/RO)+BETAO*CLLF)
          ZI=RAO*(ALFAO*(ZO/RO)+BETAO*CMMP)
          CALL INDEX(RAD,N2,A,B,RZERC)
          EONR=DAES(E/(N2*RAO))
          IF(ECNR.GT.1.0) WRITE(6,9978)EONR
          IF(ECNR.GT.1.0) ECNR=1.0
          PSIP=DARSIN(EUNR)
          IF(CSYMB.LT.0.0.AND.TCRIT.GT.THTAP) PSIP=PI-PSIP
C          CALCULATE REFRACTION CF RAY AT INNER SURFACE
C          457
          A1=(1+CSYMB)/RAD
          B1=YI/RAO
          C1=Z1/RAO
          CALL DIRECT(CKP,CLP,CMP,DIR,AL,B1,CL,NPCX,NPOY,NPOZ,PSIP)
          IF(XI.LT.0.0.CR.XI.GT.X2(1)) GO TO 51
          NUM5=DSQRT((XI**2)*(DTAN(ALPHA)**4)+YI**2+ZI**2)
C          DIRECTION COSINES CF INSIDE NORMAL
C          LKF=-XI*(DTAN(ALPHA)**2)/NUM5
C          LLF=YI/NUM5
C          LMF=Z1/NUM5
C          KKLLPM=CKP*LKP+CLP*LLP+CMP*LMP
C          IF(KKLLPM.GE.1.0) THETA1=0.0
C          IF(KKLLPM.LE.-1.0) THETA1=PI
C          IF((CABS(KKLLPM)).GE.1.0) GO TO 457
C          THETA1=CARCOS((CKP*LKP)+(CLP*LLP)+(CMP*LMP))
C          CCNTINCE
C          457
C          ANGLES CF INCIDENCE AND TOTAL INTERNAL REFLECTION
C          AT THE INSIDE SURFACE:
C          PH1=PI-THETA1
C          IF(FLUG.EQ.1) PH11=THETA1
C          TIFA=DARSIN(N3/N2)
C          CHECK FOR TOTAL INTERNAL REFLECTION:

```

```
C
C
C
C
IF(PHII.GE.TIRA) GO TO 52
PHIIP=DARSIN((N2/N3)*DSIN(PHII))
C
C
CALCULATION OF TRANSMITTED INTENSITY AT INSIDE SURFACE
C
FACE=2
CALL XMI1(PHII,N2,XMINCP,FACE,XM,N1,N3)
NTNCTY(G)=XMINC*XMINCP
XMAVE=XMAVET+NTNCTY(G)
XMAVE=XMAVET/(G*.1.O)
NUM8=CCOS(PHIIP)-(N2/N3)*DCOS(PHII)
C
C
CALCULATE DIRECTION COSINES OF INSIDE EXTERNAL REFRACTED RAY
CORRECTING FOR RAYS PAST REGION OF NEAR-RADIAL LINES
C
C
IF(FLUG.EQ.1) CKP=-CKP
IF(FLUG.EQ.1) CLP=-CLP
IF(FLUG.EQ.1) CMP=-CMP
CKPP=(N2/N3)*CKP-NUM8*LKP
CLPP=(N2/N3)*CLP-NUM8*LLF
CMPP=(N2/N3)*CMP-NUM8*LMP
C
C
WRITE OUTFLT FOR PROGRAM DETPLOT
C
C
126
WRITE(4,126) XI,YI,ZI,CKPP,CLFP,CMPP
FORMAT(10,F10.7)
C
LENGTH OF INSIDE EXTERNAL REFRACTED RAY
C
L2=(BF-XI)/CKPP
NUM6=((XO+AB)*CK+YO*CL)
NUM7=((XO+AB)*(-CL)+YO*CK)
C
LENGTH OF OUTSIDE INCIDENT RAY
C
C1=DSQRT((XO-NUM6)**2+(YC-NUM7)**2)
C
TOTAL OPTICAL PATHLENGTH:
C
CFL(G)=N1*D1+N2*D2+N3*D3
C
INTERSECTION WITH THE IMAGE PLANE:
C
YIM(G)=(BF-XI)/CKPP)*CLPP+YI
ZIM(G)=(BF-XI)/CKPP)*CMPP+ZI
CIM(G)=DSQRT(YIM(G)**2+ZIM(G)**2)
IF(DABS(YIM(G)).GT.1.O.OR.DABS(ZIM(G)).GT.1.O.UK.CIM(G).GT.
C
C
C
```

```

1 1.0) GC TO 55
1 IF(CMPP .LT. 0.000000000001 .AND. CMPP .GT. -0.000000000001)
GO TO 46
XDIAPT(G)=(-ZIM(G)/CMPP)*CKFP+AE
YDIAPT(G)=(-ZIM(G)/CMPP)*CLFP+YIM(G)
GC TO 47
46 CONTINUE
XDIAPT(G)=S9559.
YDIAPT(G)=S9999.
47 CONTINUE
1 IF(SQRAY .EQ. 1) WRITE(6,500)G,RAYY(G),RAYZ(G),XG,YG,XI,
YI,ZI,OPL(G),YIM(G),ZIM(G),NTNCTY(G),XDIAPT(G),YDIAPT(G),
FLAG(G)=0
45 CONTINUE
H=G
G=F+1
NTNCTY(G)=1.0
RAYY(G)=RAYY(H)
RAYZ(G)=RAYZ(H)+GRID
IF(FLAG(H) .EQ. 0) COUNT=CCOUNT+1
GO TO 50
END OF FAY TRACE
FLAG ALL FAILED RAYS
NO INTERCEPT WITH 2ND SURFACE
NTNCTY(G)=0.0
FLAG(G)=1
IF(SQRAY .EQ. 1) WRITE(6,600)G,RAYY(G),RAYZ(G)
GO TO 45
INTERCEPT OUTSIDE BOUNDARY OF SECOND SURFACE
NTNCTY(G)=0.0
FLAG(G)=2
IF(SQRAY .EQ. 1) WRITE(6,700)G,RAYY(G),RAYZ(G)
GO TO 45
TOTAL INTERNAL REFLECTION AT 2ND SURFACE
NTNCTY(G)=0.0
FLAG(G)=3
IF(SQRAY .EQ. 1) WRITE(6,800)G,RAYY(G),RAYZ(G)
GO TO 45
TOTAL EXTERNAL REFLECTION
NTNCTY(G)=0.0
FLAG(G)=4
IF(SQRAY .EQ. 1) WRITE(6,802) G,RAYY(G),RAYZ(G)
GO TO 45
RAY OUTSIDE LENS DIAMETER
NTNCTY(G)=0.0

```


G1S08170
 G1S08180
 G1S08190
 G1S08200
 G1S08210
 G1S08220
 G1S08230
 G1S08240
 G1S08250
 G1S08260
 G1S08270
 G1S08280
 G1S08290
 G1S08300
 G1S08310
 G1S08320
 G1S08330
 G1S08340
 G1S08350
 G1S08360
 G1S08370
 G1S08380
 G1S08390
 G1S08400
 G1S08410
 G1S08420
 G1S08430
 G1S08440
 G1S08450
 G1S08460
 G1S08470
 G1S08480
 G1S08490
 G1S08500
 G1S08510
 G1S08520
 G1S08530
 G1S08540
 G1S08550
 G1S08560
 G1S08570
 G1S08580
 G1S08590
 G1S08600
 G1S08610
 G1S08620
 G1S08630
 G1S08640

```

IF (SQURAY .EQ. 1) WRITE(6,801) G,S,RAYY(G),RAYZ(G)
GO TO 62
CONTINUE
C 66 IF (FLAG(S) .GT. 4) GO TO 67
IF (SQURAY .EQ. 1) WRITE(6,803) G,S,RAYY(G),RAYZ(G)
GO TO 62
C 67 RAY OUTSIDE LENS DIAMETER
IF (SQURAY .EQ. 1) WRITE(6,805) G,S,RAYY(G),RAYZ(G)
COUNT2=COUNT2+1
GO TO 62
70 CONTINUE
DRAYS=G-1
DRAYS=RAYS-C
WRITE(6,135C) QSYMB,RZERO
WRITE(6,900) RAYS,DRAYS,COUNT2,COUNT
XI=1.0
YI=1.0
ZI=1.0
WRITE(4,127) XI,YI,ZI,CKPP,CLPP,CMPP
127 FORMAT(10F10.7)
128 WRITE(4,128)
FORMAT(10F10.7)
C
C GENERATE OBJECT PLANE ELLIPSES:
IF (ELLIPS .EQ. 1) WRITE(6,1000) ENUM
NUMBER=1
NUME=1
71 CONTINUE
COCRD=1
ELZ(NUMB,COCRD)=-Y1(NUMBR)
72 CONTINUE
COCRD)=(X1(NUMBR)+AB)*SIN(ALFAP)+DSQRT((Y1(NUMBR))**2-
1(ELZ(NUMB,COCRD))**2)*COS(ALFAP)
ELZZ=ELZ(NUMB,COORD)
IF (ELLIPS .EQ. 1) WRITE(6,1100) NUMBR,COCRD,ELY(NUMB,COORD),
1ELZ(NUMB,COCRD)
COCRD=COORD+1
ELZ(NUMB,COCRD)=ELZZ+ELL
IF (ELZ(NUMB,COCRD) .GE. Y1(NUMBR)) GO TO 73
GO TO 72
73 CONTINUE
ELZZ=ELZ(NUMB,COORD)
74 CONTINUE
ELZ(NUMB,COCRD)=ELZZ-ELL
IF (ELZ(NUMB,COCRD) .LE. -Y1(NUMBR)) GO TO 75
ELY(NUMB,COCRD)=(X1(NUMBR)+AB)*SIN(ALFAP)-DSQRT((Y1(NUMBR))**2-
1(ELZ(NUMB,COCRD))**2)*COS(ALFAP)
  
```

```

ELZ2=EL2(NUMB,CCCRD)
IF(ELLIFS.EQ.1) WRITE(6,1100)NUMBR,CCCRD,ELY(NUMB,CUORD),
1EL2(NUMB,CUORD)
CUORD=CCCRD+1
GO TO 74
75 CONTINUE
NUMBER=NUMBR+(K-1)/ELLNUM
NUMB=NUMB+1
IF(NUMBR.GT.K) GC TC 76
GO TO 71
76 CONTINUE
IF(ELLIFS.EQ.1) WRITE(6,1200)
NUMB=NUMB-1
IMAGE PLANE SPOT DIAGRAM STATISTICAL ANALYSIS:
SUM1=0.0
DO 77 G=1,RAYS
IF(FLAG(G).GT.0) GC TO 77
SUM1=YIM(G)+SUM1
77 CONTINUE
REFERENCE EACH ITERATION TC INCIDENT ANGLE & THICKNESS BY:
IANGLE=ALFAP#10+1
THICK=(1/R)*100
CENTRIC OF SPOT:
YCENCTR(IANGLE,THICK)=SUM1/CCOUNT
STANDARD DEVIATIONS:
SUM2=0.0
SUM3=0.0
DO 78 G=1,RAYS
IF(FLAG(G).GT.0) GO TO 78
SUM2=ZIM(G)**2+SUM2
SUM3=(YCENCTR(IANGLE,THICK)-YIM(G))**2+SUM3
78 CONTINUE
SIGMAZ(IANGLE,THICK)=SUM2/CCOUNT
SIGMAY(IANGLE,THICK)=SUM3/CCOUNT
ROOT MEAN SQUARE SPCT SIZE:
RMSRAD(IANGLE,THICK)=SQRT(SIGMAZ(IANGLE,THICK)+SIGMAY(IANGLE,
1THICK))
WRITE(6,1300) I,U,ALFAP,R,YCENCTR(IANGLE,THICK),
1SIGMAY(IANGLE,THICK),SIGMAZ(IANGLE,THICK),RMSRAD(IANGLE,THICK)
SPCT DIAGRAM ENERGY DENSITY VS. RADIUS FROM CENTROID:
DO 79 G=1,RAYS
IF(FLAG(G).GT.0) GC TO 79
SURAD(G)=DSQRT((YCENCTR(IANGLE,THICK)-YIM(G))**2+ZIM(G)**2)

```



```

DOUBLE FREGISIGN R,N,A,B,RZERO
N=DSQRT(A+B*(R/RZERC)**2)
RETURN
END

```

C
C
C
C
C

SUBROUTINE TRANSMIT CALCULATES THE TRANSMITTED INTENSITY OF EACH RAY AT BOTH THE OUTSIDE AND INSIDE SURFACES USING THE FRESNEL EQUATIONS.

```

SUBROUTINE XMIT(PHI,N2,XM1,FACE,XM,N1,N3)
REAL PHI,N3,XM1,NTI,N1,N2,SQRT,COS
INTEGER FACE
NTI=N2/N1
IF (FACE.EQ. 2) NTI=N3/N2
VALUE=NTI**2-(SIN(PHI)**2)
IF (VALUE.LT. 0.0) GO TO 10
SQ=SQRT(VALUE)
CUSP=CCS(PHI)
RPER=(CCSP-SQ)/(COS*SQ)
RPAR=((NTI**2)*(COS**2)*CCSP+SQ)
XM1=1.0-0.5*(RPER**2+RPAR**2)
RETURN
10 CONTINUE
FACE=3
RETURN
END

```

10

92

SUBROUTINE RADYUS CALCULATES THE RADIAL DIMENSION FROM THE CENTER OF SYMMETRY TO THE RAY AT THE ANGLE SPECIFIED. RADYUS ALSO FINDS C(R)/D(THETA) FOR SUPER ITRATE.

```

SUBROUTINE RADYUS(A,E,ARG,RSING,ANGLE,R,DR,FILTER,EPSLON)
REAL EPSLON
DOUBLE PRECISION ANGLE,R,DR,ARG,RSINO,FRACTN,A,E
INTEGER J,FILTER
FRACTN=CSIN(-2.0*EPSLON*ANGLE+RSINO)
IF ((A+AFG*FRACTN).LE. 0.0) GO TO 10
R=SQRT((2.0)*DABS(E)/DSQRT(A+ARG*FRACTN))
DR=((EPSLON*R**3)/(2.0*E**2))*DCOS(-2.0*EPSLON*ANGLE+R*SINO)*ARG
10 CONTINUE
FILTER=0
IF ((A+AFG*FRACTN).LT. 0.0) FILTER=1
RETURN
END

```

10

SUBROUTINE RADUS CALCULATES THE RADIUS TO THE FRONT SURFACE AT

C
C

GIS1C570
GIS10580
GIS10590
GIS10600
GIS10610
GIS10620
GIS10630
GIS10640
GIS10650
GIS10660
GIS10670
GIS10680
GIS10690
GIS10700
GIS10710
GIS10720
GIS10730
GIS10740
GIS10750
GIS10760
GIS10770
GIS10780
GIS10790
GIS10800
GIS10810
GIS10820
GIS10830
GIS10840
GIS10850
GIS10860
GIS10870
GIS10880
GIS10890
GIS10900
GIS10910
GIS10920
GIS10930
GIS10940
GIS10950
GIS10960
GIS10970
GIS10980
GIS10990
GIS11000
GIS11010
GIS11020
GIS11030
GIS11040

```

C
C
C THE SPECIFIED ANGLE AND ALSO FINDS ITS DERIVATIVE WITH RESPECT
C TC ANGLE.
SUBROUTINE RADUS(P1,X1,Y1,I1,U,CSYMB,THETAP,RADG,RDCTG)
DOUBLE PRECISION X1,Y1,I1,U,THETAP,RADD,RDG,P1,P12,CSYMB
P12=PI/2.
S=DCOTAN(I1+U)
B=Y1-S*(X1+CSYMB)
RADU=B/(CSIN(THETAP))-S*DCOS(THETAP))
RDG=-B*(CCOS(THETAP))+S*DSIN(THETAP))/
1((CSIN(THETAP))-S*DCOS(THETAP))*2)
RETURN
END

C
C
C SUBROUTINE INTERCEPT PERFORMS ITERATION TO FIND R AND THETAP
C OF THE INTERCEPT OF THE GRIN SKEW-RAY AND THE INSIDE CONICAL
C SURFACE USING THE NEWTON-RAPHSON ITERATION PROCEDURE.
SUBROUTINE INTERI(SIGN,THETA,R,THETAP,FILTER,A,E,ARG,K,SINO,
1EPSLON,C,TCRIT,NPOX,NPOY,NPOZ,PSIO,XO,YC,ZO,RO,CKKP,CLLP,
1CMMP,ALPHA,CSYMB,FLUG)
INTEGER LCCP,FILTER,FLUG
REAL SIGN,EPSLON,THETA,R,THETAP,A,E,ARG,RSINO,RO,DSYMB,
1ALPHA,RADG,RDCTG,FX,FXDCT,RDCTG,RDCTS,XNEW,DIF,C,TCRIT,PSIO,
1FACTOR,EXIT,THET,P1,XO,YU,ZO,NPOX,NPOY,NPOZ,CKKP,CLLP,CMMP
LOCP=0
FILTER=C
PI=3.141592653589793
1C CONTINUE
THET=THETA
IF(THETA.GT.TCRIT) THET=THETA-C
IF(THETA.GT.TCRIT) EPSLON=+1.0
CALL RADUS(A,E,ARG,RSINO,THET,RADG,RDCTG,FILTER,EPSLON)
IF(FILTER.EQ. 1)RETURN
IF(FILTER.EQ. 2)RETURN
CALL SURF(THETA,RADG,RDCTS,SIGN,FILTER,NPOX,NPOY,NPOZ,PSIO,XU,
1YO,ZO,CSYMB,RO,CKKP,CLLP,CMMP,ALPHA)
IF(FILTER.EQ. 3)RETURN
FX=RADG-RADH
FXDCT=RCLCTG-RDCTS
FACTOR=1.7/(DABS(P1-PSIO))*-.5
IF(CSYMB.GT.0.0) FACTOR=1.3
XNEW=THETA-(FX/FACTOR)
DIF=DABS(THETA-XNEW)
EXIT=0.CC00C01*DABS(P1-PSIO)
IF(CSYMB.GT.0.0) EXIT=0.0000001

```

G1S11C50
 G1S11C60
 G1S11C70
 G1S11C80
 G1S11C90
 G1S11100
 G1S11110
 G1S11120
 G1S11130
 G1S11140
 G1S11150
 G1S11160
 G1S11170
 G1S11180
 G1S11190
 G1S11200
 G1S11210
 G1S11220
 G1S11230
 G1S11240
 G1S11250
 G1S11260
 G1S11270
 G1S11280
 G1S11290
 G1S11300
 G1S11310
 G1S11320
 G1S11330
 G1S11340
 G1S11350
 G1S11360
 G1S11370
 G1S11380
 G1S11390
 G1S11400
 G1S11410
 G1S11420
 G1S11430
 G1S11440
 G1S11450
 G1S11460
 G1S11470
 G1S11480
 G1S11490
 G1S11500
 G1S11510
 G1S11520

```

IF(CIF -GT. EXIT) GC TC 20
THETAP=THETA
R=RADG
RETURN
2C CONTINUE
IF(LOCF -GE. 90) GC TC 30
LULP=LCCP+1
THETA=XNEW
GO TO 1C
30 CONTINUE
IF(LOCF -GE. 90) FILTER=3
WRITE(6,641C)
FORMAT(1X,SOLUTION NOT FOUND*)
641C RETURN
ENC

SUBROUTINE INTERCEPT2 PERFCRMS ITERATION TO FIND THE INTERSECTION
OF THE GRIN RAY AND THE INSIDE SURFACE USING THE ANGLE IN
TERMS OF RADIUS PROCEDURE.

SUBROUTINE INTER2(RP,THETAP,RPP,A,E,ARG,RSINQ,EPSLON,
1SIGN,NPCX,NPOY,NPOZ,PSIO,XO,YO,ZO,CSYMB,RO,CKKP,CLLP,CMMP,
1ALPHA,THETA,FILTER,FLUG)
INTEGER TRIP,FILTER,FLUG
REAL SIGN,EPSLON
DOUBLE PRECISION RP,THETAP,RPP,A,E,ARG,RSINC,OSYMB,
1PSIO,RCC,ALPHA,RSIN,CHECK1,CHECK2,THETA,THATP,RDOT,ARGH,
1XU,YU,ZU,NPCX,NPOY,NPOZ,CKKP,CLLP,CMMP
FILTER=C
TRIP=0
TRIP=TRIP+1
ARGH=(2*E**2/RP**2-A)/ARG
IF(DABS(ARGH).GT.1.0) GC TC 50
RSIN=DAFSIN(ARGH)
THETPP=-EPSLON/2*(RSIN-RSINQ)
CHECK1=THETPP-THETA
CALL SURF(THETPP,RPP,RDOT,SIGN,FILTER,NPOX,NPOY,NPOZ,PSIO,XU,
1YU,ZU,CSYMB,RO,CKKP,CLLP,CMMP,ALPHA)
CHECK2=(ABS(RP-RPP)
RP=RPP
THETAP=THETPP
IF(TRIP.GT.100) GO TO 20
IF(CHECK2.LT.0.000001) GO TC 20
GO TO 1C
THATP=THETAP
RETURN
ENC
20
50
C

```

GIS11530
 GIS11540
 GIS11550
 GIS11560
 GIS11570
 GIS11580
 GIS11590
 GIS11600
 GIS11610
 GIS11620
 GIS11630
 GIS11640
 GIS11650
 GIS11660
 GIS11670
 GIS11680
 GIS11690
 GIS11700
 GIS11710
 GIS11720
 GIS11730
 GIS11740
 GIS11750
 GIS11760
 GIS11770
 GIS11780
 GIS11790
 GIS11800
 GIS11810
 GIS11820
 GIS11830
 GIS11840
 GIS11850
 GIS11860
 GIS11870
 GIS11880
 GIS11890
 GIS11900
 GIS11910
 GIS11920
 GIS11930
 GIS11940
 GIS11950
 GIS11960
 GIS11970
 GIS11980
 GIS11990
 GIS12000

```

SUBROUTINE SURF CALCULATES BOTH THE RADIUS TO THE
LOCUS OF THE INTERCEPT OF THE RAY PLANE AND THE IN-
SIDE SURFACE AND THE DERIVATIVE OF THE RADIUS WRT THETA
GIVEN THE ANGLE THETA. SURF IS DESIGNED PRIMARILY
FOR USE WITH SUBROUTINES INTERCEPT1 AND INTERCEPT2.

SUBROUTINE SURF(THETA,R,RDOT,SIGN,FILTER,NPOX,NPOY,NPOZ,PSIO,XO,
1YU,20,C*YMB,R0,CKKP,CLLP,C*MM*ALPHA)
INTEGER FILTER
REAL XC,YC,ZO,NPOX,NPOY,NPOZ,CKKP,CLLP,C*MM*ALPHA,SIGN
DOUBLE PRECISION THETA,R,RDOT,PSIO,C*SYME,
1RO,ALPHA,ALFA,BETA,A,B,C,A2,B2,C2,SQUARE,
ICAL,DBE,CA,CB,CC,DA,DB2,DC,DA2,DB3,C3
ALFA=DC*CS(THETA)/DSIN(THETA)*DCCTAN(PSIO)
BETA=DSIN(THETA)/DSIN(PSIO)
A=ALFA*(XC+C*SYMB)/RO+BETA*CKKP
B=ALFA*YC/RO+BETA*CLLP
C=ALFA*ZC/RO+BETA*C*MM
A2=B*2+C*2-(A*2)*((DTAN(ALPHA)**2)
B2=B*NP*CV+C*NP*OZ+A*NP*OX+2.0*A*C*SYMB*(DTAN(ALPHA)**2)
C2=C*SYME*NP*OX+(C*SYME*DTAN(ALPHA))**2+NPOX*XC+NPOY*YO+NPOZ*ZO
SQUARE=C*25*(B2/A2)**2+C2/A2
IF(SQUARE<.1) FILTER=3
IF(FILTER<.EQ.3) RETURN
R=-B2/(2.0*A2)+SIGN*DSQRT(SQUARE)
CALCULATE CR/DTHETA
DAL=-DSIN(THETA)-DCS(THETA)*DCCTAN(PSIO)
DBE=DCS(THETA)/DSIN(PSIO)
CA=CAL*(XC+C*SYMB)/RO+CEE*CKKP
CB=CAL*YC/RO+DBE*CLLP
CC=CAL*ZC/RO+DBE*C*MM
DA2=2.0*B*DB2-0.0*C*CC-2.0*A*DA*(DTAN(ALPHA)**2)
DB2=DB*2+NPOY*DC*NP*OZ+DA*NP*OX+2.0*DA*C*SYMB*(DTAN(ALPHA)**2)
A3=-0.5*(A2*DB2-B2*CA2)/(A2**2)
B3=SIGN/(2.0*DSQRT(0.25*(B2/A2)**2+C2/A2))
C3=0.25*((A2**2)-(B2*DB2-(B2**2)*2.0*A2*CA2)/(A2**4))-
1C2*CA2/(A2**2)
RDOT=A3+B3*C3
RETURN
END
SUBROUTINE DIRECT CALCULATES THE VECTOR DIRECTION COSINES OF
A SKEW RAY AT THE POINT OF INTERCEPT WITH THE INSIDE SURFACE.
SUBROUTINE DIRECT(CKP,CLP,CMP,DIR,BRX,RRY,NPFZ,PSIR)
DOUBLE PRECISION CKP,CLP,CMP,DIR,RRX,RRY,RRZ,NPFZ,NPFY,NPFY,NPFZ,PSIR,
1EPS,APF,BPP,CPP,AA,BB,CC,AP,BP,CP

```

CCCCC

C

CCCCC

```

EPS=1.0E-06
SIGN=+1.0
IF((DABS(NPFX)) .GT. EPS) GO TO 50
IF((DABS(NPFY)) .GT. EPS) GO TO 30
IF((DABS(RRY)) .LT. EPS) GO TO 25
CMP=C.C
APP=1.C+(FRX/RRY)**2
BPP=DCCLS(PSIR)*RRX/RRY**2
CPP=(CCCLS(PSIR)/RRY)**2-1.0
CKP=BFF/APP+SIGN*DSQRT(BPP**2-APP*CPP)/APP
CLP=(CCGS(PSIR)-RRX*CKP)/RRY
RETURN
25 CONTINUE
CMP=C.C
CKP=CCCLS(PSIR)/RRX
CLP=SIGN*DSQRT(1.0-CKP**2)
RETURN
30 CONTINUE
IF((DABS(NPFZ)) .GT. EPS) GO TO 40
IF((DABS(RRZ)) .LT. EPS) GO TO 35
CLP=C.C
APP=1.C+(RRX/RRZ)**2
BPP=DCCLS(PSIR)*RRX/RRZ**2
CPP=(CCCLS(PSIR)/RRZ)**2-1.0
CKP=BFF/APP+DSQRT(BPP**2-APP*CPP)/APP
CLP=(CCGS(PSIR)-RRX*CKP)/RRZ
RETURN
35 CONTINUE
CLP=C.C
CKP=CCCLS(PSIR)/RRX
CMP=SIGN*DSQRT(1.0-CKP**2)
RETURN
40 CONTINUE
IF((DABS(RRX)) .GT. EPS) GO TO 45
CLP=CCCLS(PSIR)/(RRY-NPFY/NPFZ*RRZ)
CMP=-NPFY/NPFZ*CLP
CKP=SIGN*DSQRT(1.0-CLP**2-CMP**2)
RETURN
45 CONTINUE
IF((DABS(RRY-NPFY/NPFZ*RRZ)) .GT. EPS) GO TO 47
CLP=CCCLS(PSIR)/RRX
CKP=SIGN*DSQRT((1.0-CKP**2)/(1.0+(NPFY/NPFZ)**2))
CMP=-NPFY/NPFZ*CLP
RETURN
47 CONTINUE
SIGN=-1.C
IF(RRY .LT. 0.0 .OR. RRX .LT. 0.0) SIGN=+1.C
IF(RRY .LT. 0.0 .AND. RRX .LT. 0.0) SIGN=-1.0

```

GIS12490
 GIS12500
 GIS12510
 GIS12520
 GIS12530
 GIS12540
 GIS12550
 GIS12560
 GIS12570
 GIS12580
 GIS12590
 GIS12600
 GIS12610
 GIS12620
 GIS12630
 GIS12640
 GIS12650
 GIS12660
 GIS12670
 GIS12680
 GIS12690
 GIS12700
 GIS12710
 GIS12720
 GIS12730
 GIS12740
 GIS12750
 GIS12760
 GIS12770
 GIS12780
 GIS12790
 GIS12800
 GIS12810
 GIS12820
 GIS12830
 GIS12840
 GIS12850
 GIS12860
 GIS12870
 GIS12880
 GIS12890
 GIS12900
 GIS12910
 GIS12920
 GIS12930
 GIS12940
 GIS12950
 GIS12960

```

APP=( ( RRY-NPFY/NPFZ*RRZ)/RRX)**2+1.0*(NPFY/NPFZ)**2
BPP=2.0*CCGS(P SIR)/(RRX**2)*(RRY-NPFY/NPFZ*RRZ)
C P P=(DCGS(P SIR)/RRX)**2-1.0
CLP=EFF/(2.0*APP)+SIGN*DSQRT(BPP**2-4.0*APP*CPP)/(2.0*APP)
CKP=-NPFY/NPFZ*CLP
C K P=(CCGS(P SIR)-(RRY-NPFY/NPFZ*RRZ)*CLP)/FRX
RETURN
5C CONTINUE
SIGN=-1.0
IF (RRY .LT. 0.0) SIGN=+1.0
IF (RRY .LT. 0.0 .AND. NPFZ .LT. 0.0) SIGN=-1.0
AA=NPFY*RRY-NPFY*RRX
BB=NPFY*RRZ-NPFZ*RRX
CC=NPFY*CCGS(P SIR)
IF (DAES(BB)) .GT. EPS) GO TO 60
CLP=CC/AA
AP P=(NPFZ/NPFX)**2+1.0
BP P=NPFY/NPFZ/NPFX**2*CLP
C P P=(NPFY/NPFX)**2+1.0)*CLP**2-1.0
CKP=-EFP/APP+SIGN*DSQRT(BPP**2-APP*CPP)/APP
RETURN
6C CONTINUE
IF (BB .GT. 0.0) SIGN=-1.0
AP=(NPFY/NPFX)**2+1.0
BP=(NPFZ/NPFX)**2+1.0
CP=2.0*NPFY/NPFZ/NPFX**2
AP P=AP+EF*(AA/BB)**2-CP*AA/BB
BP P=2.0*CC*CC*BP/BB**2-CP*CC/BB
C P P=BP*(CC/BB)**2-1.0
CLP=EFF/(2.0*APP)+SIGN*DSQRT(BPP**2-4.0*APP*CPP)/(2.0*APP)
CKP=(CC/BB-AA/BB)*CLP
C K P=(-NPFY*CLP-NPFZ*CMP)/NPF X
RETURN
EN

```

G1S12570
 G1S12580
 G1S12590
 G1S13000
 G1S13010
 G1S13020
 G1S13030
 G1S13040
 G1S13050
 G1S13060
 G1S13070
 G1S13080
 G1S13090
 G1S13100
 G1S13110
 G1S13120
 G1S13130
 G1S13140
 G1S13150
 G1S13160
 G1S13170
 G1S13180
 G1S13190
 G1S13200
 G1S13210
 G1S13220
 G1S13230
 G1S13240
 G1S13250
 G1S13260
 G1S13270
 G1S13280
 G1S13290
 G1S13300
 G1S13310

CHEOC0C10
 CHEOC0C20
 CHEOC0C30
 CHEOC0C40
 CHEOC0C50
 CHEOC0C60
 CHEOC0C70
 CHEOC0C80
 CHEOC0C90
 CHEOC0C100
 CHEOC0C110
 CHEOC0C120
 CHEOC0C130
 CHEOC0C140
 CHEOC0C150
 CHEOC0C160
 CHEOC0C170
 CHEOC0C180
 CHEOC0C190
 CHEOC0C200
 CHEOC0C210
 CHEOC0C220
 CHEOC0C230
 CHEOC0C240
 CHEOC0C250
 CHEOC0C260
 CHEOC0C270
 CHEOC0C280
 CHEOC0C290
 CHEOC0C300
 CHEOC0C310
 CHEOC0C320
 CHEOC0C330
 CHEOC0C340
 CHEOC0C350
 CHEOC0C360
 CHEOC0C370
 CHEOC0C380
 CHEOC0C390
 CHEOC0C400
 CHEOC0C410
 CHEOC0C420
 CHEOC0C430
 CHEOC0C440
 CHEOC0C450
 CHEOC0C460
 CHEOC0C470
 CHEOC0C480

98

```

C C C
C SET CCNANTS
C   PI=2.141592653589793
C   PI2=PI/2
C   N1=1.0
C   N3=1.0
C   BF=F/R
C   RZERO=((X2(1)+QSYMB)**2+Y2(1)**2)**.5
C   IF(CSYMB.LT.-0.6) RZERO=DSQRT(R**2+(CSYMB**2)
C
C C C
C START RAY TRACE
C   DO 10 I=2,K
C   L=J-I
C   TRIP=0
C
C C C
C CALCULATE MIDPCINT
C   S=(Y1(J)-Y1(L))/(X1(J)-X1(L))
C   XD=(X1(J)+X1(L))/2.
C   YD=(Y1(J)+Y1(L))/2.
C
C C C
C CALCULATE LINE CONSTANTS
C   I1=PI2-LATAN(S)
C   THETA0=LATAN(YD/(XC+QSYMB))
C   IF(THETA0.LT.0.0) THETA0=PI+THETA0
C   RO=DSQRT(YD**2+(XC+QSYMB)**2)
C   CALL INDEX(RO,N20,A,B,RZERO)
C   I1P=DARSIN((N1/N20)*DSIN(I1))
C   PSIO=PI-THETA0-I1+I1P
C   EPSLGN=1.0
C   IF(PSIO.GT.PI2.OR.PSIO.LT.0.0) EPSLGN=-1.0
C   F=RO*N20*CSIN(PSIO)
C   ARG=DSQRT(A**2+4*B*(E**2)/(RZERO**2))
C   RSINU=CARSIN((2*(E**2)/(RO**2)-A)/ARG)
C   C=PI2-RSINU
C   TCRTI=1-F*ETAC-(EPSLGN/2.)*(PI2-RSINC)
C   IF(PSIO.LT.PI2) C=0.0
C
C C C
C FIND FIRST GLESS COORDINATES, RADIUS AND ANGLE
C   S1=-DTAN(I1-I1P)
C   S2=DTAN(ALPHA)
C   XI=(YD-S1*XC)/(S2-S1)
C   YI=S2*X1
C   THETAP=LATAN(YI/(XI+QSYMB))

```

```

C      IF (THETAP.LT.0.0) THETAP=THETAP+PI
C      RP=CSQRT(YI**2+(XI+GSYMB)**2)
C      EE=-OSYME
C      SL=1.0
C      GO TO THETA IN TERMS OF RADIUS ITERATION IF ANGLE LARGE
C
C      IF (THETAP.GT.2.0) GC TC 25
C
C      NEWTON-RAPHSON ITERATION
C
C      2C CONTINUE
C      TRIP=TRIP+1
C      IF (THETAP.GE.10.0) THETP=THETAP-C
C      IF (THETAP.LT.10.0) THETP=THETAP
C      IF (THETAP.GE.10.0) EPSLON=+1.0
C      THETA=THETP-THETAO
C      CALL RADYUS(ARG,RSINO,THETAH,RAD,RDG,FILTER,A,EPSLON,E)
C      CALL RADCLS(CSYMB,S2,THETAP,RADD,RDGG)
C      RDF=RAD-RADD
C      RCT=REG-RDGG
C      FACTCR=1.77/(DABS(PSTO))
C      IF (FACTOR.GT.10.0) FACTCR=10.0
C      IF (CSYMB.GT.0.0) FACTOR=1.3
C      THETA=THETAP-RCT/(RDT*FACTCR)
C      DIFF=DABS(THETA-THETAP)
C      EXIT=0.00001*DABS(PSTO)
C      IF (CSYMB.GT.0.0) EXIT=0.000001
C      EXIT2=200
C      IF (DIFF.LT.EXIT) GC TO 30
C      IF (TRIP.GT.EXIT2) GC TO 30
C      THETAP=THETAN
C      GO TC 20
C
C      THETA IN TERMS OF RADIUS ITERATION
C
C      25
C      TRIP=TRIP+1
C      RSIN=CARSIN((2*E**2/RP**2-A)/ARG)
C      THETA=THETAO-EPSLON/2*(RSIN-RSINC)
C      CHECK1=THETA-THETAP
C      RPP=EE/(CSIN(THETA)-SL*DCOS(THETA))
C      CHECK2=DABS(RP-RPP)
C      THETA=THETA
C      RP=RPP
C      IF (TRIP.GT.100) GO TO 26
C      IF (CHECK2.LT.0.000001) GO TO 26
C      GO TC 25
C      RAC=RFP
C
C      26

```

CHEOC570
 CHEOC580
 CHEOC590
 CHEOC600
 CHEOC610
 CHEOC620
 CHEOC630
 CHEOC640
 CHEOC650
 CHEOC660
 CHEOC670
 CHEOC680
 CHEOC690
 CHEOC700
 CHEOC710
 CHEOC720
 CHEOC730
 CHEOC740
 CHEOC750
 CHEOC760
 CHEOC770
 CHEOC780
 CHEOC790
 CHEOC800
 CHEOC810
 CHEOC820
 CHEOC830
 CHEOC840
 CHEOC850
 CHEOC860
 CHEOC870
 CHEOC880
 CHEOC890
 CHEOC900
 CHEOC910
 CHEOC920
 CHEOC930
 CHEOC940
 CHEOC950
 CHEOC960
 CHEOC970
 CHEOC980
 CHEOC990
 CHEOC1000
 CHEOC1010
 CHEOC1020
 CHEOC1030
 CHEOC1040
 CHEOC1050
 CHEOC1060
 CHEOC1070
 CHEOC1080
 CHEOC1090
 CHEOC1100
 CHEOC1110
 CHEOC1120
 CHEOC1130
 CHEOC1140
 CHEOC1150
 CHEOC1160
 CHEOC1170
 CHEOC1180
 CHEOC1190
 CHEOC1200
 CHEOC1210
 CHEOC1220
 CHEOC1230
 CHEOC1240
 CHEOC1250
 CHEOC1260
 CHEOC1270
 CHEOC1280
 CHEOC1290
 CHEOC1300
 CHEOC1310
 CHEOC1320
 CHEOC1330
 CHEOC1340
 CHEOC1350
 CHEOC1360
 CHEOC1370
 CHEOC1380
 CHEOC1390
 CHEOC1400
 CHEOC1410
 CHEOC1420
 CHEOC1430
 CHEOC1440

```

C C
C C      CALCULATE FINAL POINT AT INNER SURFACE COORDINATES AND ANGLES
C C      3C XF=RAD*CCOS(THETAP)-OSYMB
C C          YF=RAD*CSIN(THETAP)
C C          CALL INDEX(RAD,N2,A,B,RZERO)
C C          EONR=E/(N2*RAD)
C C          IF (EONR.GT.1.0) WRITE(6,63C)
C C          IF (EONR.GT.1.0) EONR=1.0
C C          PSIP=DAFSIN(EONR)
C C          IF (THETAP.LT.YCRIT.AND.PSIO.GT.0.0) PSIP=PI-PSIP
C C          ZETA=PI-THETAP-PSIP
C C          I2=PI2-ZETA-ALPHA
C C          EON=(N2/N3)*DSIN(I2)
C C          IF (EON.GT.1.0) WRITE(6,64C)
C C          IF (EON.GT.1.0) EON=1.0
C C          I2P=DAFSIN(EON)
C C          UP=PI2-ALPHA-I2P
C C
C C      CALCULATE Y-COORDINATE AT FOCAL PCINT
C C          S3=-DIAN(UP)
C C          YS(J)=S3*(F-XF)+YF
C C
C C      WRITE CUIPLT
C C      1C CONTINUE
C C          STOP
C C
C C      FORMAT STATEMENTS
C C      10C FORMAT(1X,3F11.7,15,F10.7,3F5.3)
C C      20C FORMAT(1X,4F10.7)
C C      50C FORMAT(1X,16,2F16.8,16)
C C      63C FORMAT(1X,'WHOOOPS PSIP ARSIN')
C C      64C FORMAT(1X,'WHOOOPS AGAIN I2P ARSIN')
C C
C C      END
C C
C C      SUBROUTINE INDEX(R,N,A,B,RZERO)
C C      CF RADII FROM THE CENTER OF SYMMETRY. THE VALUES OF THE INDEX
C C      CONSTANTS A AND B ARE USER INPUTS.
C C
C C      SUBROUTINE INDEX(R,N,A,B,RZERO)
C C      DOUBLE PRECISION N,A,B,R,RZERO
C C      N=CSQRT(A+B*(K/RZERO)**2)
C C      RETURN
C C
CHEO1450
CHEO1460
CHEO1470
CHEO1480
CHEO1490
CHEO1500
CHEO1510
CHEO1520
CHEO1530
CHEO1540
CHEO1550
CHEO1560
CHEO1570
CHEO1580
CHEO1590
CHEO1600
CHEO1610
CHEO1620
CHEO1630
CHEO1640
CHEO1650
CHEO1660
CHEO1670
CHEO1680
CHEO1690
CHEO1700
CHEO1710
CHEO1720
CHEO1730
CHEO1740
CHEO1750
CHEO1760
CHEO1770
CHEO1780
CHEO1790
CHEO1800
CHEO1810
CHEO1820
CHEO1830
CHEO1840
CHEO1850
CHEO1860
CHEO1870
CHEO1880
CHEO1890
CHEO1900
CHEO1910
CHEO1920

```

```

C C C C C
SUBROUTINE RADYUS CALCULATES THE RADIUS FOR THE GRIN EQUATION AT
THE SPECIFIED ANGLE AS WELL AS ITS DERIVATIVE WITH RESPECT TO
ANGLE.
END

SUBROUTINE RADYUS(ARG,RSING,ANGLE,R,DR,FILTER,A,EPSLON,E)
INTEGER FILTER
DOUBLE PRECISION ANGLE,R,DR,ARG,RSINC,A,EPSLON,E,FRACTN
FRACTN=CSIN(-2.0*EPSLON*ANGLE+RSING)
IF((A+AFC*FRACTN).LE.0.0) GO TO 10
R=SQRT(2.0)*DABS(E)/DSQRT(A+ARG*FRACTN)
IF(FRACTN.LE.(-0.989)) GO TO 10
DR=((EPSLON*R**3)/(2.0*E**2))*DCOS(-2.0*EPSLON*ANGLE+RSING)
10 CONTINUE
FILTER=C
IF(FRACTN.LE.(-0.989))FILTER=2
IF((A+AFC*FRACTN).LT.0.0) FILTER=1
RETURN
ENC

C C C C C
SUBROUTINE RADUS CALCULATES THE RADIUS TC THE FRONT SURFACE AT
THE SPECIFIED ANGLE AND ALSO THE DERIVATIVE WITH RESPECT TO
ANGLE.
END

SUBROUTINE RADUS(OSYMB,S,THTAP,RACC,RDGG)
DOUBLE PRECISION THTAP,OSYMB,S
RACC=-OSYMB*S/(DSIN(THTAP)-S*DCCS(THTAP))
RDGG=S*CSYME*(DCCS(THTAP)+S*DSIN(THTAF))/
1(DDSIN(THTAP)-S*DCCS(THTAP))*#2)
RETURN
ENC

```

103

[illegible]

```

C C SET CCNSTANTS
PI=3.14159
PI2=PI/2.
DETP=0.2
NIRP=1.4
THETC=66.C*PI/180
DETS=DETP*TAN(ALPHA)
THEM(1)=THETC-0.20404040
J=C
L=1
CU11=0
TERM1=C
TERM2=0

C C READ IN RAY DATA REJECTING RAYS BLOCKED BY DETECTOR
10 J=J+1
   REAC(5,ZCC) XO(J),YC(J),ZO(J),CK(J),CL(J),CM(J)
   IF(SQR(YC(J)*2+ZO(J)*2).LT.DETS) GO TO 11
   IF(XO(J).EQ.1.0.AND.YO(J).EQ.1.0.ANC.ZO(J).EQ.1.0) TERM1=J
   IF(XO(J).EQ.1.0.ANC.YO(J).EQ.1.0.ANC.ZO(J).EQ.1.0) GO TO 12
   GO TO 1C
11 J=J-1
   CU11=CU11+1
   GO TO 1C
12 CONTINUE

C C INCREMENT MIRROR ANGLE AND CHECK FOR COMPLETED SWEEP
20 L=L+1
   K=L-1
   THEM(L)=THEM(K)+0.00404040
   THEID(L)=THEM(L)*180/PI
   IF(THEID(L).GT.(THETC+0.20)) TERM2=L-1
   WRITE(6,200C) THEM(L),OUT1,TERM1,TERM2
   IF(THEID(L).GT.(THETC+0.20)) GO TO 40
   CU12(L)=0
   CU13(L)=C
   COUNT1(L)=0
   COUNT2(L)=0

C C RAY TRACE FROM BACK LENS SURFACE TO MIRROR AND THEN TO DETECTOR
30 J=J+1

```

```

DET0C490
DET0C500
DET0C510
DET0C520
DET0C530
DET0C540
DET0C550
DET0C560
DET0C570
DET0C580
DET0C590
DET0C600
DET0C610
DET0C620
DET0C630
DET0C640
DET0C650
DET0C660
DET0C670
DET0C680
DET0C690
DET0C700
DET0C710
DET0C720
DET0C730
DET0C740
DET0C750
DET0C760
DET0C770
DET0C780
DET0C790
DET0C800
DET0C810
DET0C820
DET0C830
DET0C840
DET0C850
DET0C860
DET0C870
DET0C880
DET0C890
DET0C900
DET0C910
DET0C920
DET0C930
DET0C940
DET0C950
DET0C960
DET0C970
DET0C980
DET0C990

```


CEI01450
CEI01463
DEI01470
CEI01480
CEI01490
DEI01500

C101C FORMAT(1X,6F10.6)
C200C FURMAT(1X,F10.7,3I10)
C200C FURMAT(1X,F10.6)
C201C FURMAT(1X,F10.6,2I5)
C
ENC

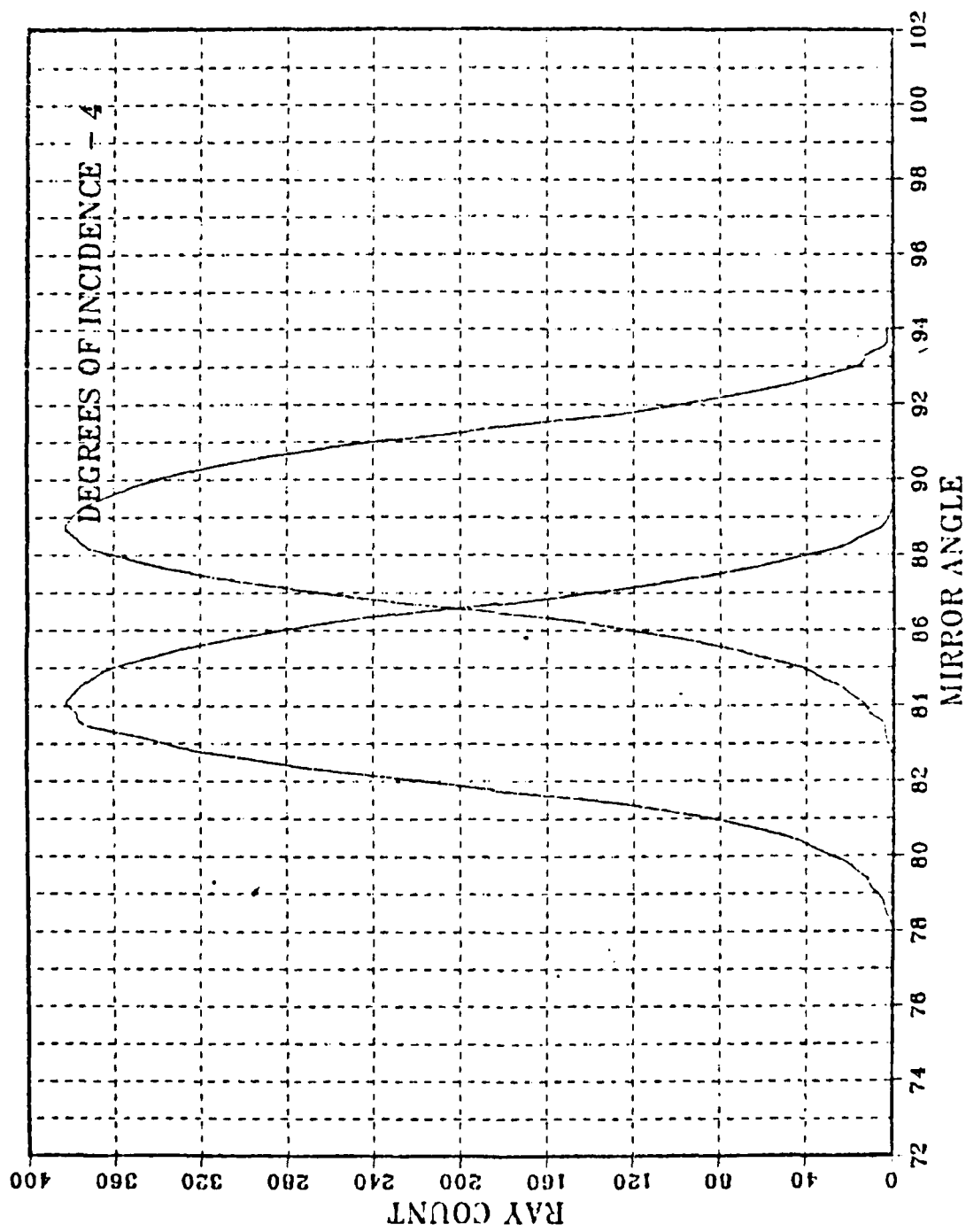


Figure D-1. Detector Signal at 4 Degrees Incidence.

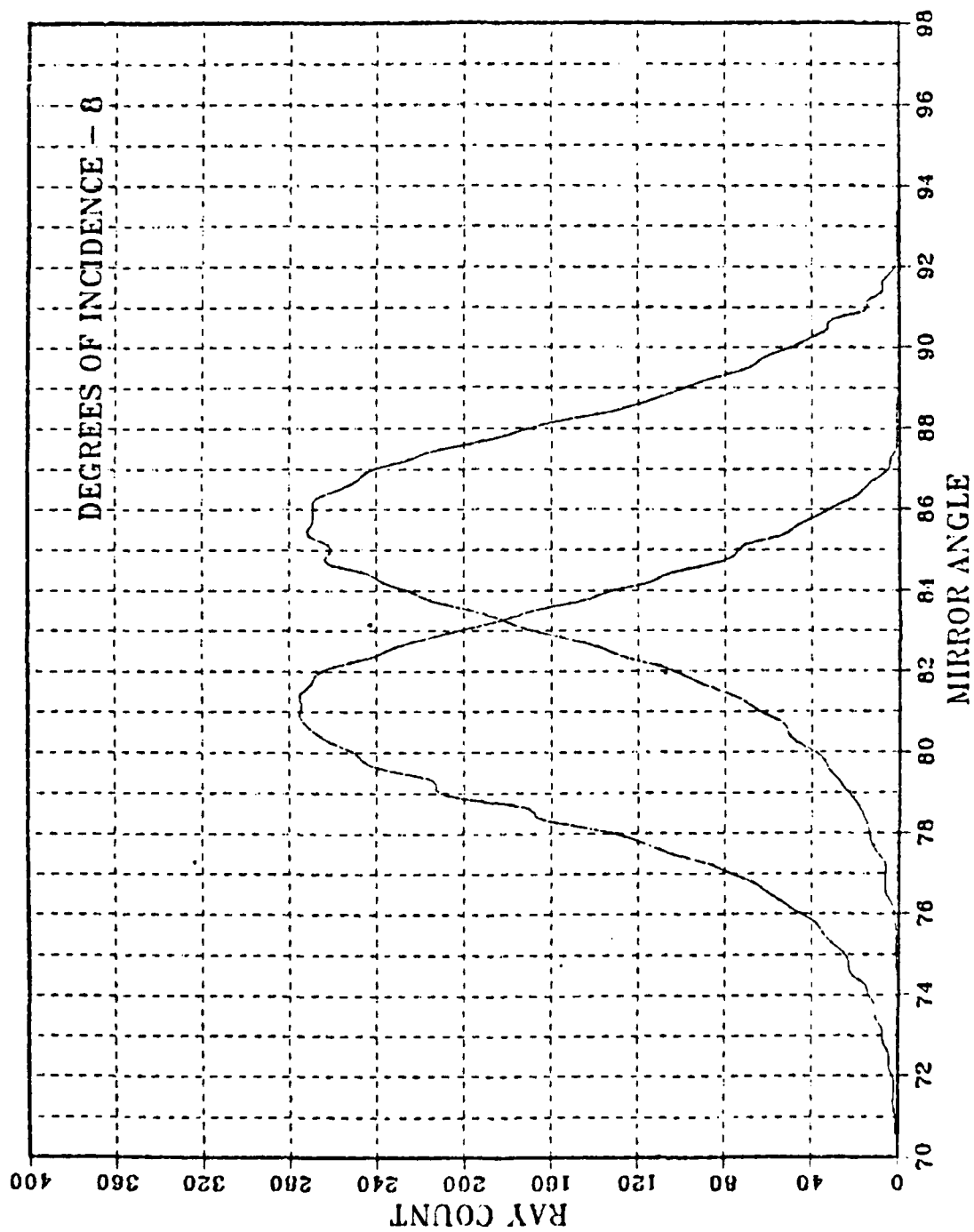


Figure D-2. Detector Signal at 8 Degrees Incidence.

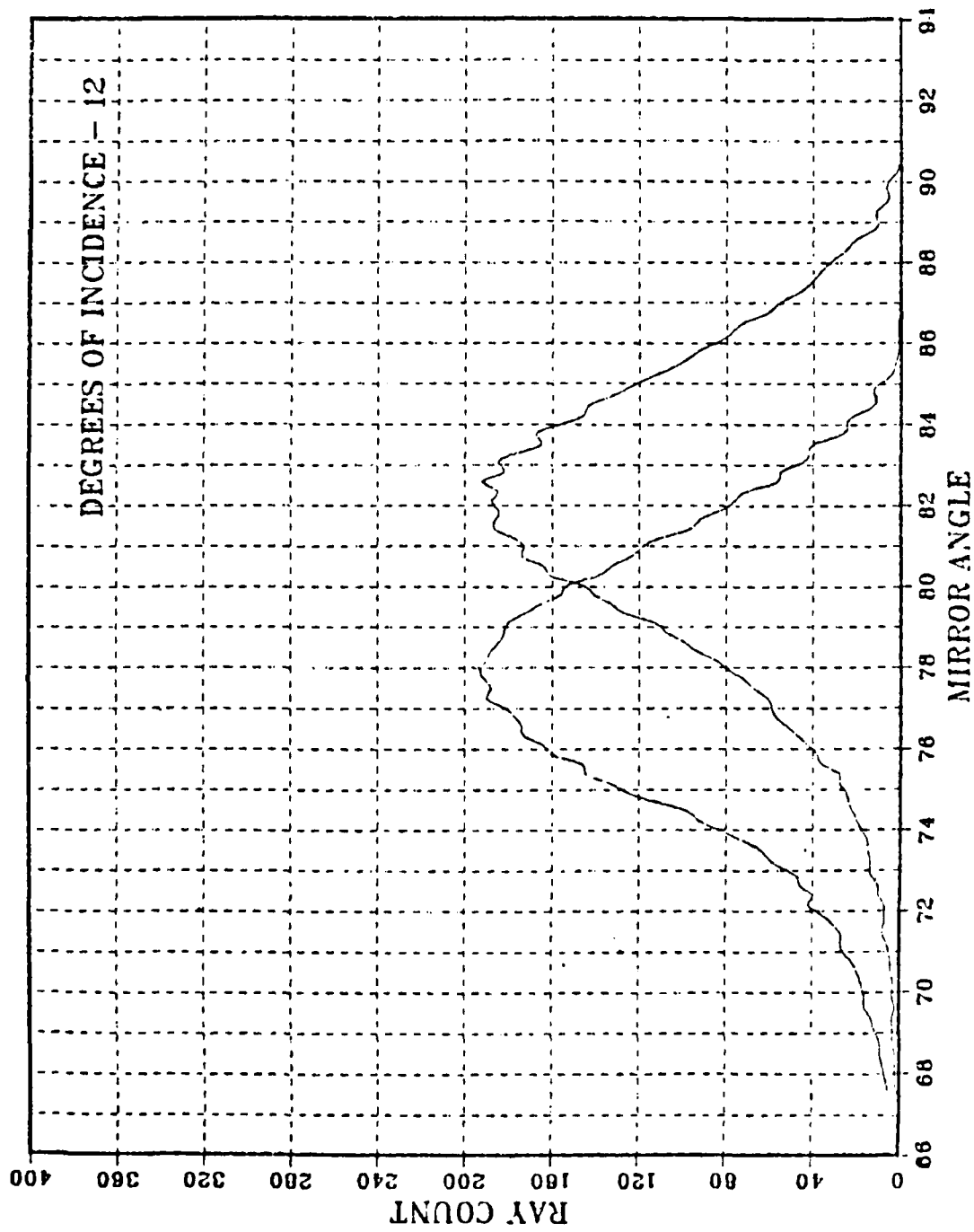


Figure D-3. Detector Signal at 12 Degrees Incidence.

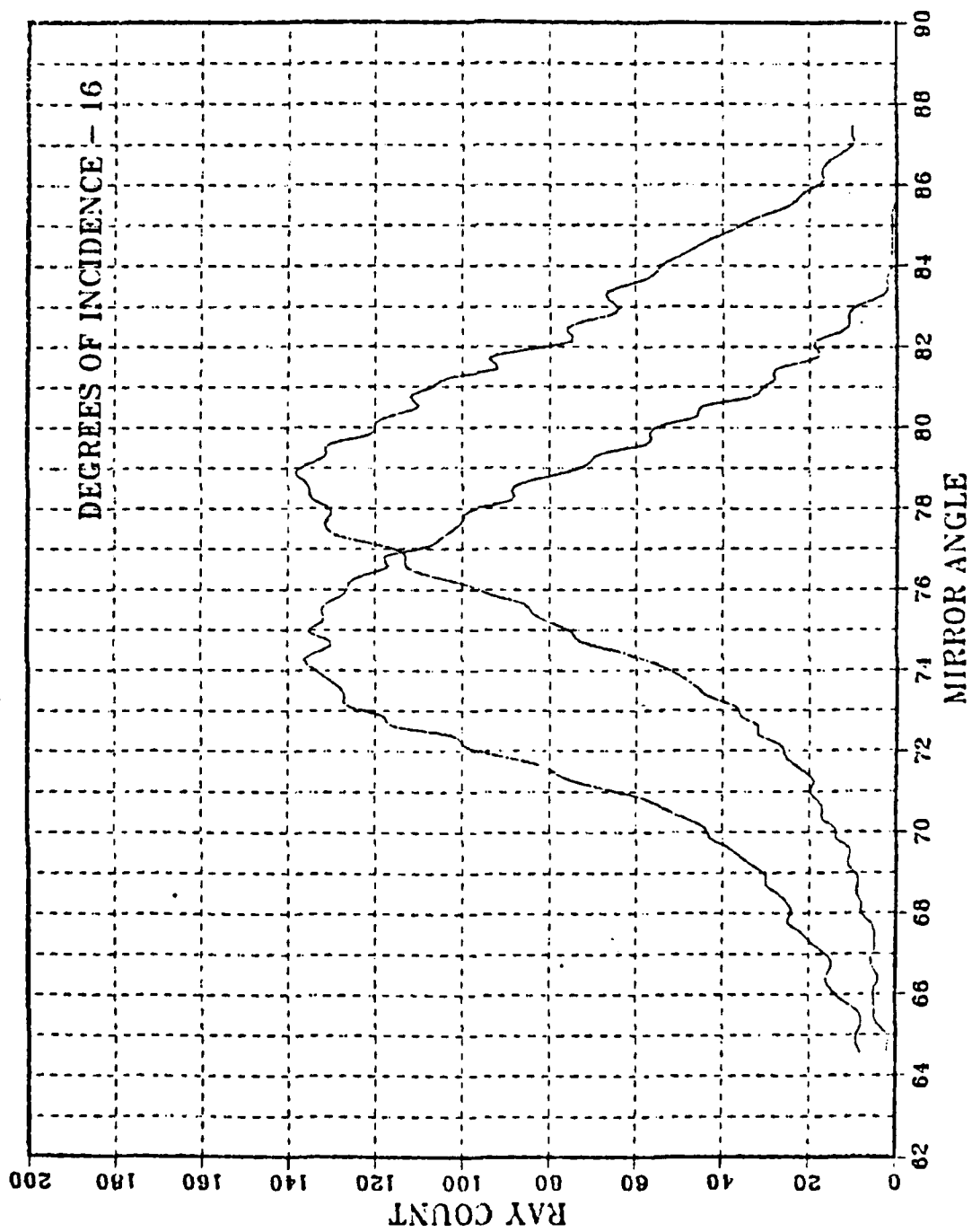


Figure D-4. Detector Signal at 16 Degrees Incidence.

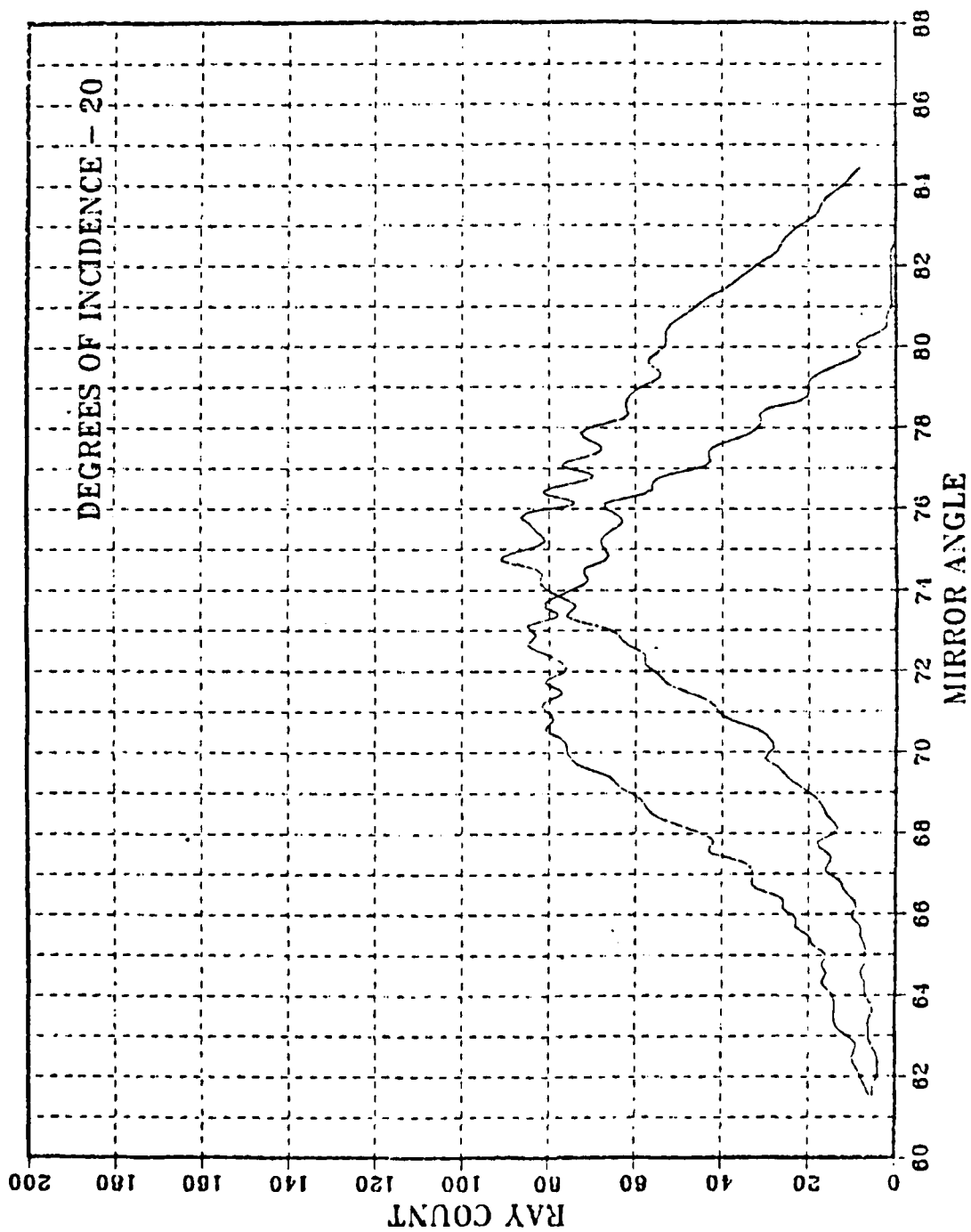


Figure D-5. Detector Signal at 20 Degrees Incidence.

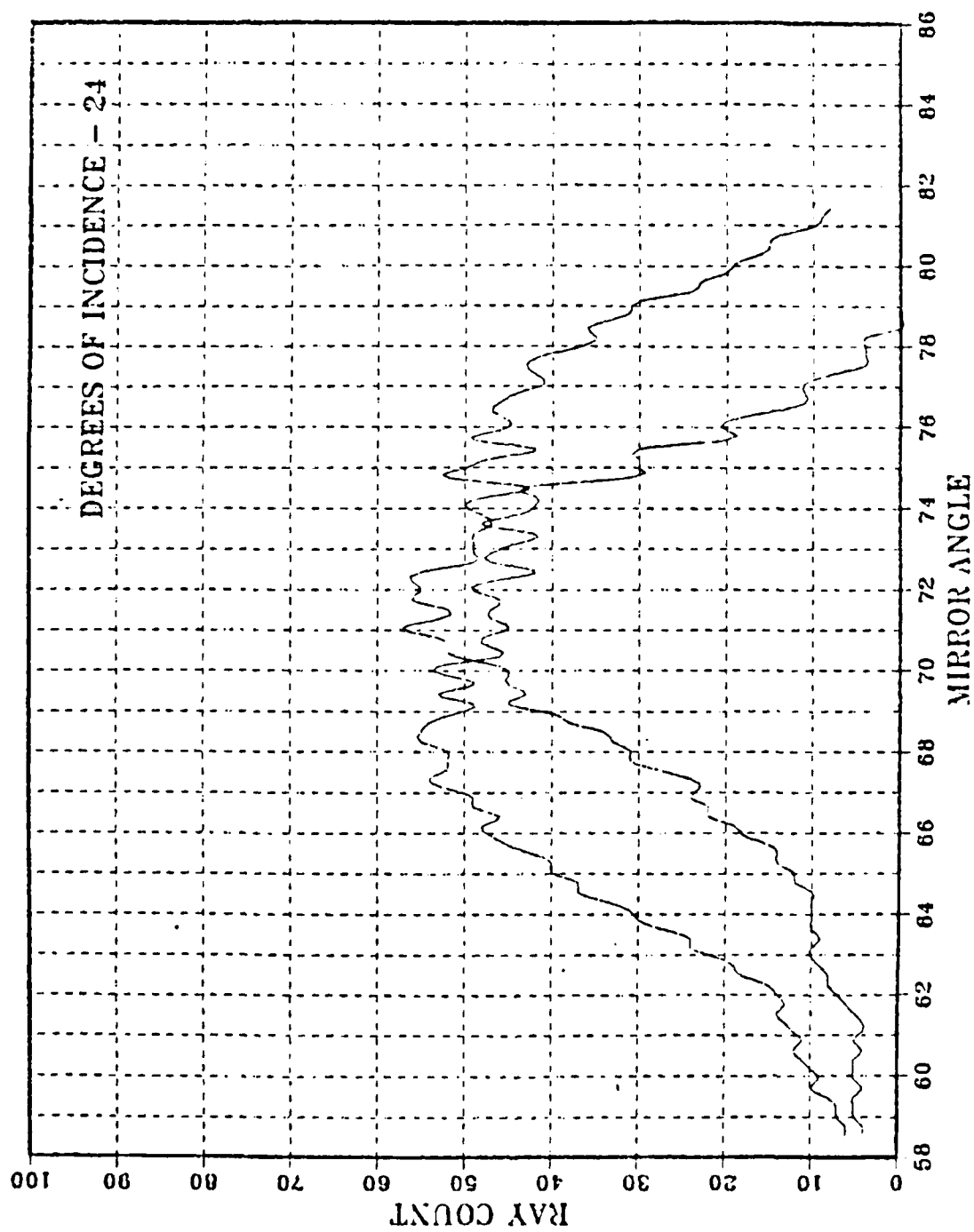


Figure D-6. Detector Signal at 24 Degrees Incidence.

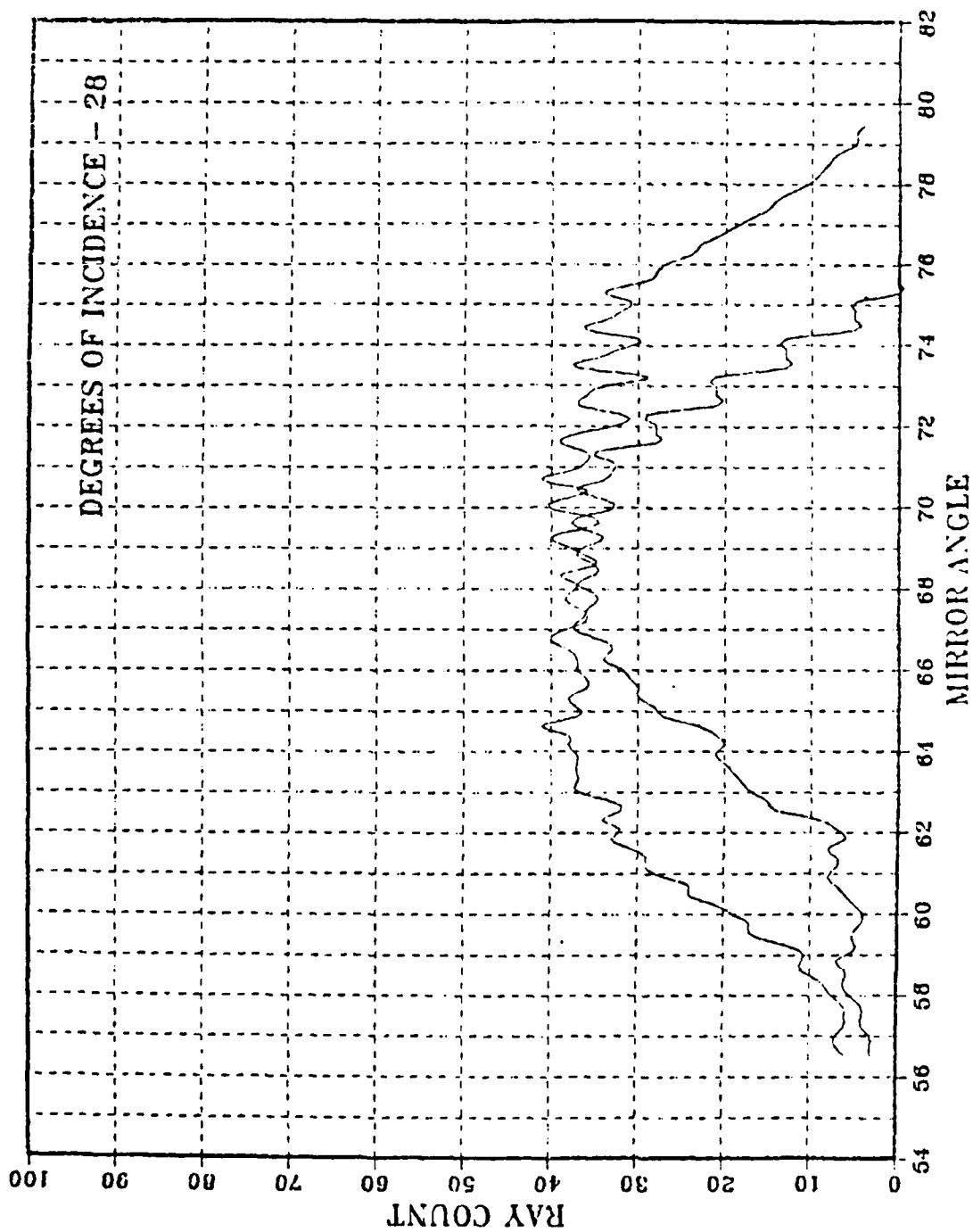


Figure D-7. Detector Signal at 28 Degrees Incidence.

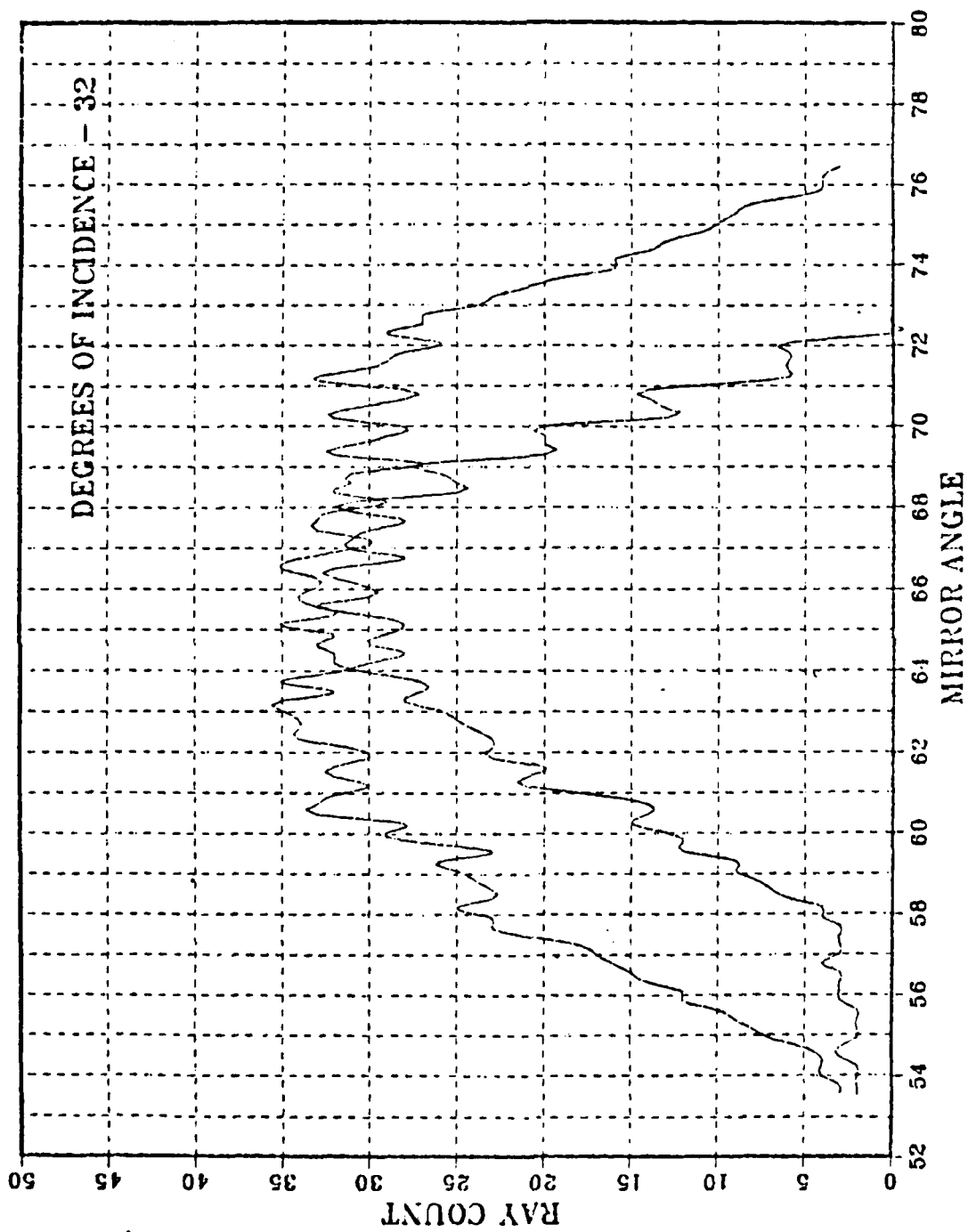


Figure D-8. Detector Signal at 32 Degrees Incidence.

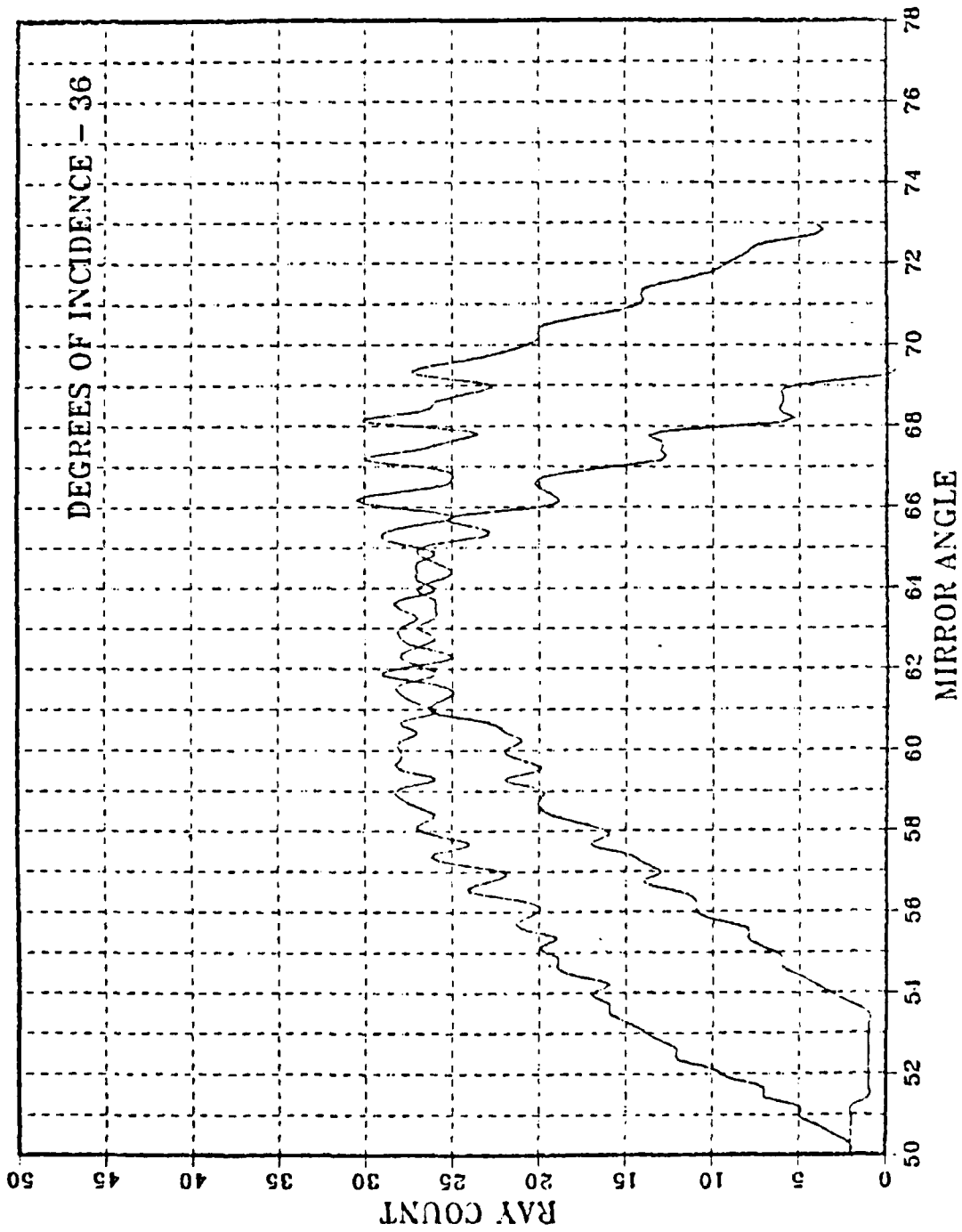


Figure D-9. Detector Signal at 36 Degrees Incidence.

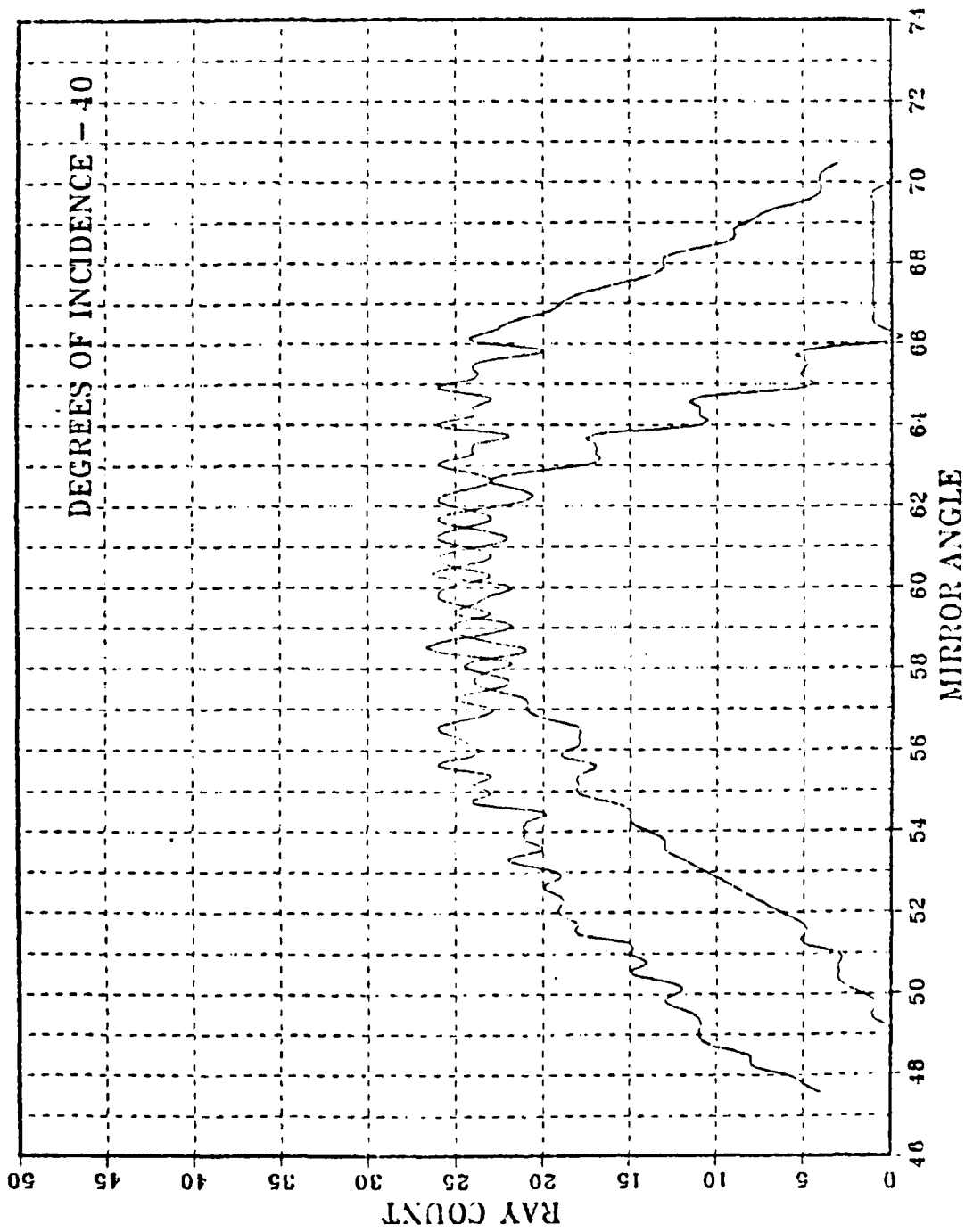


Figure D-10. Detector Signal at 40 Degrees Incidence.

LIST OF REFERENCES

1. Frazier, Robert L., Exterior Ballistics, Guidance Laws and Optics for a Gun-Launched Missile, M. S. Thesis, Naval Postgraduate School, Monterey, California, December 1980.
2. Terrell, James M., Conical Lens for a 5"/54 Gun Launched Missile, M. S. Thesis, Naval Postgraduate School, Monterey, California, June 1981.
3. Amichai, Oded, Sharp Nose Lens Design Using Refractive Index Gradient, Naval Postgraduate School Contractor Report NPS67-82-003CR, Monterey, California, June 1982.
4. Carr, Herbert M., Aerodynamically Efficient Gradient Refractive Index Missile Seeker Lens, M. S. Thesis, Naval Postgraduate School, Monterey, California, October 1982.
5. Marchand, Erich W., Gradient Index Optics, Academic Press, 1978.
6. U. S. Army Missile Command Technical Report RG-CR-32-6, Gradient Index Lens Research-Final Report, by D. T. Moore, p. 36, 19 October 1981.
7. U. S. Army Missile Command Technical Report H8 3-3-BO20-1, Gradient Index Optics Application by R. L. Light, 30 September 1983.
8. D. S. Davidson and A. E. Fuhs, Tracing Nearly Radial Lines in a Spherically Symmetric Gradient Index Lens, Applied Optics, Letter to Editor (in press).

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
3. Directorate of Land Armament and Electronics Engineering and Maintenance Attn: Colonel B. L. Code National Defense Headquarters Ottawa, Ontario, Canada K1A 0K2	2
4. Distinguished Professor A. E. Fuhs Code 67Fu Department of Aeronautics Naval Postgraduate School Monterey, California 93943	4
5. Col. R. Larriva, USMC Defense Advanced Research Projects Agency 1400 Wilson Boulevard Arlington, Virginia 22209	2
6. Charles R. Christensen Research Directorate U. S. Army Missile Command (DRSMI-RRO) Redstone Arsenal, Alabama 35898	1
7. Commander, U. S. Army Missile Command (DRSMI-ROA) Redstone Arsenal, Alabama 35898 Attn: CAPT H. Carr	1
8. Professor A. E. Milne Code 61Mn Naval Postgraduate School Monterey California 93943	1
9. Professor D. T. Moore University of Rochester Rochester, New York 14627	1

10. Dr. Lloyd Smith 1
Code 3205
Naval Weapons Center
China Lake, California 93555
11. Directorate of Land Armament and Electronics 2
Engineering and Maintenance
Attn: Capt. D. Davidson
National Defense Headquarters
Ottawa, Ontario, Canada K1A 0K2
12. Directorate of Personnel Education 1
and Development (DPED)
National Defense Headquarters
Ottawa, Ontario, Canada K1A 0K2