A Study of the Effect of Vertical Stratification of the Atmosphere on the Performance of the GBT

Leonid Kogan
November 20, 1992

1.0 Abstract

The dimensions of the GBT are such that there will be a significant difference in the refractive index of the atmosphere between the top and bottom of the reflector. This will affect the RF performance at short wavelengths and will also introduce errors in the laser metrology system. This study analyzes both these effects. Both possible locations of the focus (arm on top - "focus UP" and arm on bottom - "focus DOWN") have been analyzed. Variance of the phase error at the aperture is less for the "focus UP" configuration. The residual error is negligible in terms of pointing (less than 0.5 arcsec for elevation greater than 20°) and is marginal in terms of surface error (between 60 and 25 microns). The errors introduced in the laser system partially compensate for the phase errors at radio frequencies.

2.0 Introduction

The goal of the closed-loop active surface system is to reduce thermally induced errors in the surface of the telescope to such a level as to permit operation at wavelengths as short as 3 mm.

The aperture of the GBT is so large that the variation in refractive index with height will result in phase errors that may be significant at the shortest operating wavelengths. This variation in refractive index also affects the proposed laser metrology system. Here, we analyze these effects separately and combined. For completeness, the arm on top ("focus UP"), arm on bottom ("focus DOWN") cases are both considered. The analysis results in both pointing errors and surface errors for a standard atmosphere. These errors are also calculated for reasonable variations in atmospheric conditions assuming that a correction for a "standard" atmosphere is applied.

3.0 Method of Analysis

3.1 The RF Performance of the Antenna

A drawing of the parent parabola is shown in Figure 1. For the purpose of analysis we choose the cartesian set of coordinates fixed with respect to the antenna, and the layered atmosphere then rotates about the y axis. Such a representation simplifies the mathematical analysis. The right side of the symmetrical dish then corresponds to the "focus UP" and the left one corresponds to the "focus DOWN". The influence of the atmosphere is seen in the phase error at the aperture. To calculate this error distribution at the aperture, we assume the antenna to be in the transmitting mode. (Relative principle gives the same parameters of the antenna in the receiving mode.) We send the rays from the focus to a point on the dish surface and calculate the time delay to the aperture plane. We receive the distribution of the phase error at the aperture due to *internal* atmosphere. Then we follow the ray from the aperture to the plane of the layer passing through the edge point of the antenna (Figure 2.). The electrical length of the rays above this plane will be equal for all points of the aperture. So, the influence of the *external* atmosphere from the aperture to outer space is defined by the triangle ABC (Figure 2). All delays are multiplied by the speed of light in a vacuum to give surface errors in microns. The length of the central ray is subtracted from all other points, thus giving the relative distribution of the extra lengths with zero at the center of the aperture. To separate the effect of the atmosphere on the pointing and on the gain we fit the plane to the extra length distribution. This fitting minimizes the sum of the squares of the difference between the calculated extra length and points of the plane. We select the linear term for the pointing problem and determine the variance of the phase error at the aperture for the gain reducing the problem. Finally, we receive the distribution of the phase which corresponds to the *internal, external* atmosphere, their sum and residuals after fitting the planes. All calculations are repeated for different elevations.

The difference of the electrical lengths at two sequential layers depends upon the ratio of their refractive index and angle between the ray and the normal vector to the layer. Consider the case where the ray goes from the optically thin layer to the optically thick layer ($n2/n1 = n > 1$). The electrical length of the second layer will be greater than the first layer because of the smaller velocity, but will be less because of the smaller physical path. The final effect of these two factors depends upon the angle between the ray and the normal vector to the layer - z. Analytically, if $(n-1) << \cos^2(z)$ (z is not close to 90°), then the difference of the electrical lengths at two sequential layers - $\Delta\ell$ is determined by Equation 1.

$$\frac{\Delta \ell}{\ell} = (n-1) \cdot (1 - \tan^2 z) \qquad (1)$$

The velocity decrease is the most important factor when z is small. So, the electrical length is decreased at the second layer where the refractive index is larger ($\ell 2 > \ell 1$). When z = 45°, the effect of decreasing the physical length at the second layer is compensated by the effect of decreasing the velocity and $\ell 1 = \ell 2$. For the large z, the factor of physical length decreasing is more important and $\ell 2 < \ell 1$. The graph of Equation 1 is shown in Figure 3. This graph gives the opportunity to estimate approximately the distribution of the extra length due to atmosphere through the aperture. To find the point of the intersection of the ray with the plane of the sequence layer, we have to know such a point for the previous one and the unit vector of the direction. The straight line, passing through point P=(X0, Y0, Z0) and collinear with vector e = (ex, ey, ez) is determined by Equation 2.

$$\frac{X-X0}{ex} = \frac{Y-Y0}{ey} = \frac{Z-Z0}{ez} \qquad (2)$$

The plane corresponding to the border between neighboring layers is described by Equation 3,

$$X \cdot \sin(zen) + (Z-C) \cdot \cos(zen) + \Delta h \cdot k = 0 \qquad (3)$$

where zen = 90° - elevation is the angle between the antenna pointing direction and zenith; C = 60m is the focus distance of the antenna; k is the number of the layer starting from the focus; Δh is the height of a layer. The vector e2, which corresponds to the refracted vector e1, has to be complanar with the plane of the vectors e1 and normal vector to the plane nn = (sin(zen), 0, -cos(zen)). Its direction at this plane is determined by the sine ratio: n1 sin($\alpha 1$) = (n1 + Δn) sin($\alpha 2$), where $\alpha 1$,$\alpha 2$ are the angles between the normal vector nn and the vectors e1 and e2 correspondingly; n1 is the refractive index of the layer and Δn is the gradient of the refractive index between two neighboring layers. Finally, the unit vector of the refracted ray is determined by Equation 4

$$e_2 = \frac{e_1 - \frac{\Delta n}{n_1} \cdot (e_1 \cdot m) \cdot m}{|e_2|} \qquad (4)$$

where **m** is the vector which is complanar with the plane of vectors nn and e1 and perpendicular to vector nn.

$$m = \frac{e_1 - (e_1 \cdot nn) \cdot nn}{|m|} \tag{5}$$

The unit vector of the reflected ray is determined by Equation 6.

$$e_2 = \frac{e_1 - 2 \cdot (e_1 \cdot np) \cdot np}{|e_2|} \quad , \tag{6}$$

where

$$np = \frac{(-x_p,\ -y_p,\ 2c)}{\sqrt{x_p^2 + y_p^2 + 4 \cdot c^2}} \tag{7}$$

is the normal vector to the paraboloid surface at its point x_p, y_p.

Using Equations 2-6, it is possible to trace any ray from the focus through the layered atmosphere and its reflection from the surface and back through the layered atmosphere to the aperture plane. The rays transmitted from the aperture to outer space receive additional phase distortion as indicated by the triangle ABC (Figure 2). To separate the influence of the atmosphere on the pointing, we fit the plane to the calculated phase distribution. This fitting is made by the two dimensional regression method. The variance of the residual can be used to estimate the influence of the atmosphere on the gain of the antenna.

3.2 The influence of the Atmosphere on the Laser Range Measurement System

The system of active surface control of the GBT will be supported by laser measurements of the distances from three points to a point at the surface of the dish (GBT Memo No.36 - Pointing and Surface Control of GBT by John Payne). We have calculated the error of these measurements due to atmosphere using the method described above. It is supposed that the active surface control system will move the panels of the antenna in the normal direction by measured magnitude. So, we have to recalculate the errors of three distance measurements to the errors of determining three coordinates of the point: two at tangent plane x_p, y_p, and the third one at the normal direction to the paraboloid surface at this point. For this goal the problem has been linearized using the partial derivatives $\partial d/\partial xp$, $\partial d/\partial y_p$, $\partial d/\partial np$. The corresponding system of three linear equations has been resolved, and the value of supposed shifting of the point in the normal direction Δnp has been calculated. It is clear from Figure 4 that the path of the ray from the focus will become shorter due to this shifting by $\Delta \ell$ = AB+BC, which is equal to

$$\Delta \ell = 2 \cdot \Delta np \cdot (e \cdot np) \tag{8}$$

where e is the unit vector from the focus to the surface point. Repeating these calculations for the other points at the aperture gives the phase error distribution through the aperture due to atmospheric influence on the range measurements system. The calculation is carried out on a grid of ten meters located inside the circular aperture. The calculated phase error distributions are given in microns.

3.3 The Technique of Programming and Model of the Atmosphere

All calculations have been performed by C- programs whose listings are given in the Appendix. Three main programs have been developed: the first estimates the influence of the atmosphere on the RF property (pointing and gain); the second estimates the influence of the atmosphere on the laser range measurement system; and the third program estimates their difference. They are supported by several functions:

INTERSECT The function determines the point of intersection of a straight line with a plane. The line is given by the point (X0, Y0, Z0) and by the vector (ex, ey, ez).

PARABSEC The function determines the point of intersection of a straight line with a paraboloid. The line is given by the point (X0, Y0, Z0) and by the vector (ex, ey, ez).

REFRACT The function determines a unit vector of the refracted ray (ex, ey, ez) if the incident vector, the plane of refraction and gradient of refraction of the two layers are known.

REFLECT The function determines a unit vector of the ray (ex, ey, ez) reflected from a paraboloid at the point x_p,y_p of its surface.

FITPRINT The function fits the plane z = ax + b into data Z[x,y] minimizing the sum of the squares (Z[x,y] - (ax + b)), prints (and writes in file) the residuals of the fitting. Parameters of the fitting plane give the pointing error and variance of the phase at the aperture.

RANGEF The function calculates atmospheric income to the distance from the rangefinder to the point at the antenna surface.

MATRIX3 The function finds the solution of a system of three linear equations

The three parameters of the atmosphere model: the initial refractive index, its gradient per meter and thickness of the layer are input data for the program. We have run the calculation for two values of the layer height 10 meters and 1 meter and have not found meaningful difference. The final calculation has been done with 1-meter layer of height. The value $3 \cdot 10^{-8}$ has been chosen for the gradient of the refractive index per meter. This value corresponds to the exponential law of refractive index changing with height,

$$e^{-\frac{h[m]}{8500}} \tag{9}$$

taken from the book *Interferometry and Synthesis in Radio Astronomy* by A.R. Thompson, J.M. Moran and G.W. Swenson.

4.0 Results

In this section we present the result of an analysis of the influence of the *internal* and *external* atmosphere on the RF property of the antenna, as well as an analysis of atmospheric influence on the laser range measurements.

We suppose that the average value of the refractive index is known. Its error does not influence the *internal* atmosphere effect, because we compare the difference of phase between a point on the aperture and its central point with an equal length influenced by the atmosphere.

In the case of the *external* atmosphere, the error of the average value of the refractive index affects the pointing error only (not the phase variance), because we compare the influence of the atmosphere for the lengths which is changed proportionally to the x-coordinate of the aperture.

In the case of the laser range measurement system, the error of the average value of the refractive index affects both the pointing error and the phase variance because we compare the influence of the atmosphere for unequal lengths.

4.1 The RF Property of the Antenna

Figure 5 shows the separate effects of the *internal* and *external* atmosphere and their common effect. The fitting of the plane for each of three data has been performed, so the result is represented by three tables for each elevation. Each of three data are followed by parameters of the fitting plane and variance of the residual distributions. This information gives us an idea about the effect of the atmosphere on pointing and gain of the antenna.

The pointing error as a function of the elevation is shown in Figure 6. Analog graphs for the phase variances are shown in Figure 7. The pointing error reaches 2.3 arcsec when the elevation is equal to 15_o and down to 0.165 arcsec for zenith direction. The phase variance is changing from 60 microns when the elevation is equal to 15° to 6 microns at zenith. The GBT antenna is not symmetrical and the atmospheric influence on its parameters depends upon the direction of elevation changing. When we change the direction of elevation rotation, the influence of the *external* atmosphere is changed symmetrically, but the influence of the *internal* atmosphere is not changed symmetrically. So, we can expect that the *internal* and *external* atmosphere will partially compensate each other. We have repeated the calculation for the elevation change corresponding to the case "focus DOWN" and have compared the result with the case "focus UP". The comparison shows some advantage of the "focus UP" version. The result for the "focus DOWN" version is added to Figures 6 and 7.

4.2 The Influence of the Atmosphere on the Laser Range Measurement System

The result of recalculation of the atmospheric influence on the range measurements to phase errors at the aperture is shown in Figure 8. The pointing error decreases from 0.523 arcsec at elevation 15° to 0 at elevation 45° and then increases at the opposite side till -0.18 arcsec. The variance of the phase error at the aperture depends upon elevation very weakly and equals approximately 50 microns. We have used three rangefinders with coordinates, received from John Payne: (0.714, 0.0, 66.795); (10.181, 28.973, 24.095); and (10.181, -28.973, 24.095).

It is understandable that the solution for the position of the point at the surface of the paraboloid results from the intersection of three spheres with centers at the rangefinders' locations. In this case we are discussing a very small region of the point. So, we can speak about the intersection of the tangent planes to the corresponding spheres. From this point of view, it is clear that this intersection is very ambiguous for the points located at the plane of the rangefinders. For the chosen configuration of rangefinders, these "bad" points are located at a distance ~13 meters in x-direction from the inside edge of the dish. And we are really observing some strange points in the second row of the tables (Figure 8). If we have three rangefinders, then it is desirable to have the intersection of their plane with the aperture plane outside the aperture itself.

If the surface control system works in real time, then we can expect that the atmospheric contribution errors in the metrology system will compensate for the phase error due to the atmosphere in RF observations to some degree. To check this, we subtract the tables of our calculation of errors of the rangefinder system (Figure 8) from the tables of error in the sum of the *internal* and *external* atmospheres (Figure 5). The result is shown in Figure 9. There is a small reduction of error compared with the rangefinder system.

5.0 Conclusions

1. The influence of the atmosphere on the RF property of the GBT has been divided for pointing and gain problems. The pointing error and variance of phase error at the aperture reaches 2.3 arcsec and 70 microns accordingly for an elevation as low as 15° and for

chosen parameters of the atmosphere. These values go down to 0.4 arcsec and 6 microns for elevation greater than 45°.

2. Variance of phase error at the aperture for the "focus UP" configuration is less than the "focus DOWN" configuration, because of the effect of compensation of *internal* and *external* atmosphere which takes place in the first case.

3. The influence of the atmosphere on the laser range measurement system is recalculated at 0.5 arcsec pointing error and 50 micron variance of phase at the aperture.

4. The errors recalculated from the laser system measurement partially compensate for the phase errors at RF.

5. The question about the chosen value ($3 \cdot 10^{-8}$ 1/meter) for the gradient of the refractive index is still open. It is not a problem to repeat the calculation using another value. The question about the layer model of the atmosphere also remains open.

6.0 Acknowledgements

John Payne, who supervised the work; James Lamb, who spent a lot of time discussing the different aspects of the study and helping in programming; Jennifer Neighbours, who transformed the original Anglo-Russian dialect of the text to English. All of the staff of the Tucson Division of NRAO who surrounded me with such a warm, kind atmosphere that it was not difficult to imagine the influence of the real atmosphere on the GBT.

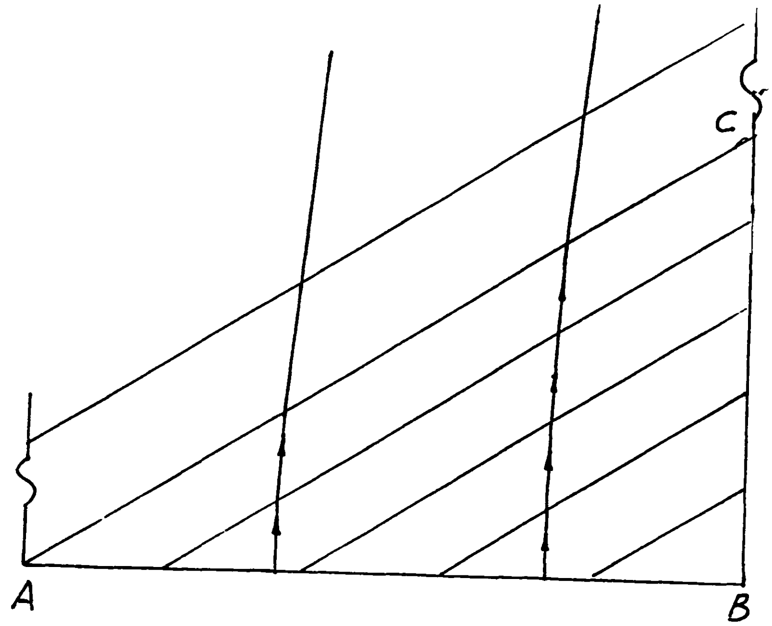

Figure 2. External atmosphere.

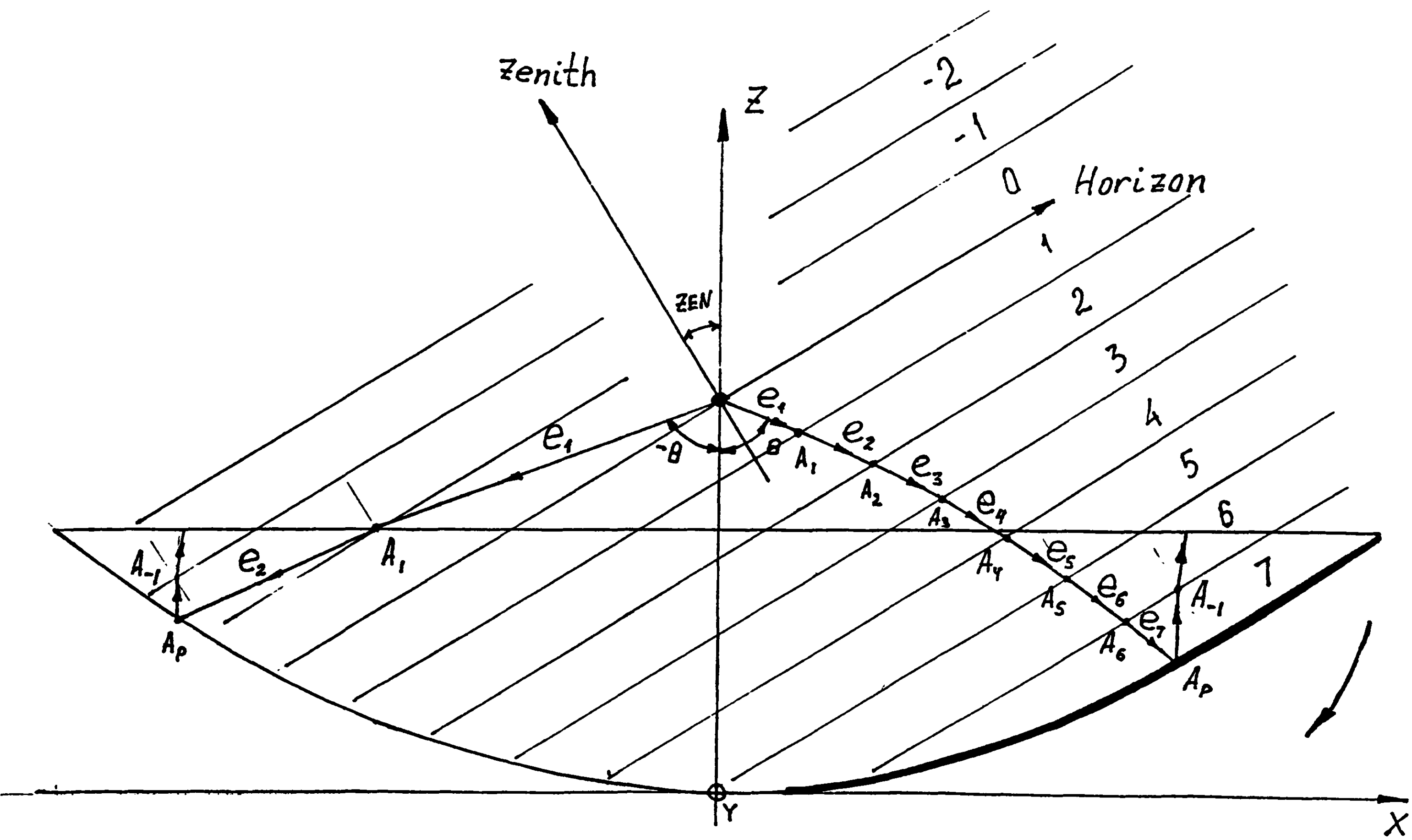

Figure 1. Internal atmosphere. The diagram of the layer model of the atmosphere in the cartesian system of coordinates is fixed with the antenna. The right side corresponds to "focus UP" and the left one to "focus DOWN". The arrow in the lower right corner shows the direction of rotation from zenith to horizon. Two rays at the right and left have the same θ, but they have absolutely different crossing of atmospheric layers.

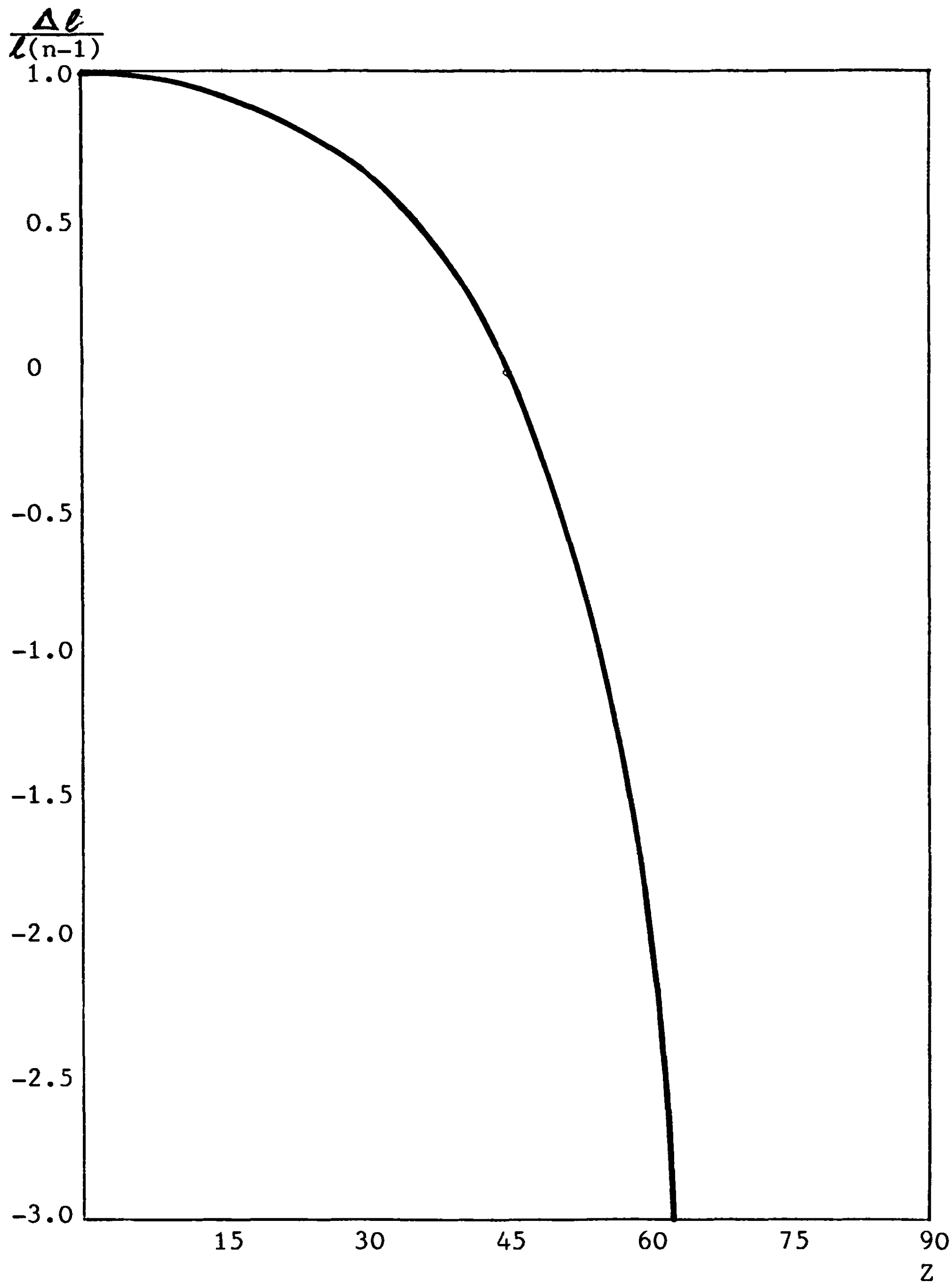

Figure 3. Extra path length through the layer due to the angle between the ray and the normal vector to the layer.

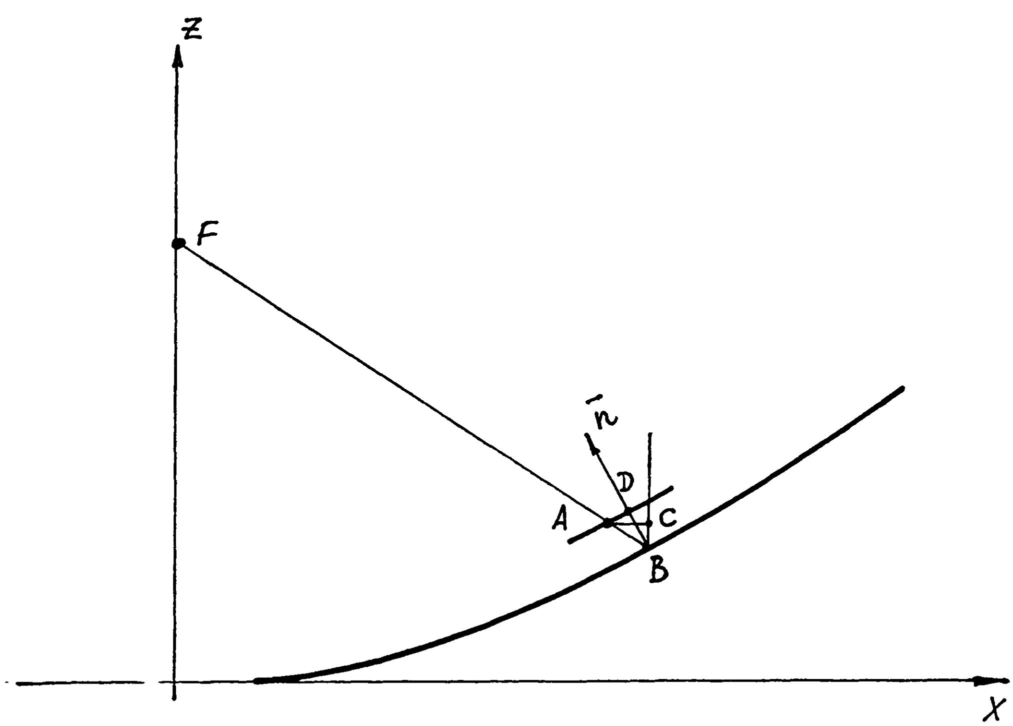

Figure 4. If point B is moved in the normal direction to D as a result of the range measurement control system, then the path of the ray from the focus will be shorter by AB + CB = DB • 2•cos ∠ABD.

* elevation = 15 *

INTERNAL ATMOSPHERE

					-94					
		-75	-74	-73	-72	-73	-74	-75		
	-57	-55	-53	-52	-52	-52	-53	-55	-57	
	-39	-37	-35	-33	-33	-33	-35	-37	-39	
	-23	-20	-17	-16	-16	-16	-17	-20	-23	
-13	-8	-5	-2	-1	0	-1	-2	-5	-8	-13
	4	8	11	13	13	13	11	8	4	
	13	18	21	23	24	23	21	18	13	
	20	25	28	30	31	30	28	25	20	
		28	32	34	35	34	32	28		
					35					

bias=-11 micron; slope 13.3 micron/point; sigma 7 micron;
angle rotation= 0.275 arcsec

EXTERNAL ATMOSPHERE

					-260					
		-250	-250	-250	-250	-250	-250	-250		
	-219	-219	-219	-219	-219	-219	-219	-219	-219	
	-169	-169	-169	-169	-169	-169	-169	-169	-169	
	-95	-95	-95	-95	-95	-95	-95	-95	-95	
0	0	0	0	0	0	0	0	0	0	0
	110	110	110	110	110	110	110	110	110	
	248	248	248	248	248	248	248	248	248	
	407	407	407	407	407	407	407	407	407	
		578	578	578	578	578	578	578		
					779					

bias=66 micron; slope 103.8 micron/point; sigma 65 micron;
angle rotation= 2.141 arcsec

INTERNAL ATMOSPHERE + EXTERNAL ATMOSPHERE

					-353					
		-325	-323	-323	-322	-323	-323	-325		
	-276	-274	-272	-271	-271	-271	-272	-274	-276	
	-208	-205	-203	-202	-202	-202	-203	-205	-208	
	-118	-115	-112	-111	-111	-111	-112	-115	-118	
-13	-8	-5	-2	-1	0	-1	-2	-5	-8	-13
	114	118	121	123	124	123	121	118	114	
	261	266	269	271	271	271	269	266	261	
	427	431	435	437	438	437	435	431	427	
		606	610	612	613	612	610	606		
					814					

bias=55 micron; slope 117.1 micron/point; sigma 59 micron;
angle rotation= 2.416 arcsec

Figure 5. The phase error distribution through the aperture in microns for different elevations. The height of the layer is equal to 1 m. Gradient of the coefficient refraction is equal to $3 \cdot 10^{-8}$. The focus location is "focus UP".

* elevation = 30 *

INTERNAL ATMOSPHERE

					-78					
		-63	-61	-60	-59	-60	-61	-63		
	-50	-47	-44	-42	-42	-42	-44	-47	-50	
	-35	-31	-28	-27	-26	-27	-28	-31	-35	
	-22	-17	-14	-13	-12	-13	-14	-17	-22	
-17	-11	-6	-3	-1	0	-1	-3	-6	-11	-17
	-2	3	7	9	9	9	7	3	-2	
	4	9	13	15	16	15	13	9	4	
	6	12	16	19	19	19	16	12	6	
		11	16	18	19	18	16	11		
					16					

bias=-12 micron; slope 9.8 micron/point; sigma 8 micron;
angle rotation= 0.202 arcsec

EXTERNAL ATMOSPHERE

					-98					
		-94	-94	-94	-94	-94	-94	-94		
	-82	-82	-82	-82	-82	-82	-82	-82	-82	
	-63	-63	-63	-63	-63	-63	-63	-63	-63	
	-36	-36	-36	-36	-36	-36	-36	-36	-36	
0	0	0	0	0	0	0	0	0	0	0
	42	42	42	42	42	42	42	42	42	
	93	93	93	93	93	93	93	93	93	
	153	153	153	153	153	153	153	153	153	
		218	218	218	218	218	218	218		
					293					

bias=25 micron; slope 39.1 micron/point; sigma 25 micron;
angle rotation= 0.806 arcsec

INTERNAL ATMOSRHERE + EXTERNAL ATMOSPHERE

					-176					
		-157	-155	-154	-153	-154	-155	-157		
	-132	-129	-126	-125	-124	-125	-126	-129	-132	
	-98	-94	-92	-90	-89	-90	-92	-94	-98	
	-58	-53	-50	-48	-48	-48	-50	-53	-58	
-17	-11	-6	-3	-1	0	-1	-3	-6	-11	-17
	39	44	48	50	51	50	48	44	39	
	97	102	106	108	109	108	106	102	97	
	159	165	169	172	172	172	169	165	159	
		229	233	236	237	236	233	229		
					309					

bias=12 micron; slope 48.8 micron/point; sigma 18 micron;
angle rotation= 1.007 arcsec

Figure 5. Continued.

* elevation = 45 *

INTERNAL ATMOSPHERE

					-57					
		-48	-45	-43	-42	-43	-45	-48		
	-39	-35	-32	-30	-29	-30	-32	-35	-39	
	-28	-23	-20	-18	-17	-18	-20	-23	-28	
	-19	-14	-10	-8	-8	-8	-10	-14	-19	
-19	-12	-7	-3	-1	0	-1	-3	-7	-12	-19
	-8	-2	2	4	5	4	2	-2	-8	
	-6	-0	4	6	7	6	4	-0	-6	
	-8	-2	3	6	7	6	3	-2	-8	
		-6	-1	2	3	2	-1	-6		
					-5					

bias=-13 micron; slope 5.6 micron/point; sigma 8 micron;
angle rotation= 0.115 arcsec

EXTERNAL ATMOSPHERE

					-39					
		-38	-38	-38	-38	-38	-38	-38		
	-33	-33	-33	-33	-33	-33	-33	-33	-33	
	-25	-25	-25	-25	-25	-25	-25	-25	-25	
	-14	-14	-14	-14	-14	-14	-14	-14	-14	
0	0	0	0	0	0	0	0	0	0	0
	17	17	17	17	17	17	17	17	17	
	38	38	38	38	38	38	38	38	38	
	61	61	61	61	61	61	61	61	61	
		88	88	88	88	88	88	88		
					118					

bias=10 micron; slope 15.7 micron/point; sigma 10 micron;
angle rotation= 0.324 arcsec

INTERNAL ATMOSRHERE + EXTERNAL ATMOSPHERE

					-96					
		-85	-82	-81	-80	-81	-82	-85		
	-72	-68	-65	-63	-62	-63	-65	-68	-72	
	-53	-49	-45	-43	-42	-43	-45	-49	-53	
	-33	-28	-25	-22	-22	-22	-25	-28	-33	
-19	-12	-7	-3	-1	0	-1	-3	-7	-12	-19
	9	15	19	22	22	22	19	15	9	
	31	37	42	44	45	44	42	37	31	
	54	60	64	67	68	67	64	60	54	
		82	87	90	91	90	87	82		
					113					

bias=-3 micron; slope 21.3 micron/point; sigma 6 micron;
angle rotation= 0.439 arcsec

Figure 5. Continued.

* elevation = 60 *

INTERNAL ATMOSPHERE

					-33					
		-29	-25	-23	-23	-23	-25	-29		
	-26	-21	-17	-15	-14	-15	-17	-21	-26	
	-19	-14	-10	-8	-7	-8	-10	-14	-19	
	-15	-10	-6	-3	-3	-3	-6	-10	-15	
-20	-13	-7	-3	-1	0	-1	-3	-7	-13	-20
	-13	-7	-3	-1	0	-1	-3	-7	-13	
	-16	-10	-5	-3	-2	-3	-5	-10	-16	
	-21	-15	-10	-8	-7	-8	-10	-15	-21	
		-23	-18	-15	-14	-15	-18	-23		
					-25					

bias=-13 micron; slope 1.0 micron/point; sigma 8 micron;
angle rotation= 0.020 arcsec

EXTERNAL ATMOSRHERE

					-13					
		-13	-13	-13	-13	-13	-13	-13		
	-11	-11	-11	-11	-11	-11	-11	-11	-11	
	-9	-9	-9	-9	-9	-9	-9	-9	-9	
	-5	-5	-5	-5	-5	-5	-5	-5	-5	
0	0	0	0	0	0	0	0	0	0	0
	6	6	6	6	6	6	6	6	6	
	13	13	13	13	13	13	13	13	13	
	21	21	21	21	21	21	21	21	21	
		30	30	30	30	30	30	30		
					40					

bias=4 micron; slope 5.3 micron/point; sigma 3 micron;
angle rotation= 0.110 arcsec

INTERNAL ATMOSPHERE + EXTERNAL ATMOSPHERE

					-46					
		-42	-38	-36	-35	-36	-38	-42		
	-37	-32	-28	-26	-25	-26	-28	-32	-37	
	-28	-23	-19	-17	-16	-17	-19	-23	-28	
	-20	-15	-11	-8	-7	-8	-11	-15	-20	
-20	-13	-7	-3	-1	0	-1	-3	-7	-13	-20
	-7	-1	3	5	6	5	3	-1	-7	
	-3	3	7	10	11	10	7	3	-3	
	-0	6	11	13	14	13	11	6	-0	
		7	12	15	16	15	12	7		
					15					

bias=-9 micron; slope 6.3 micron/point; sigma 6 micron;
angle rotation= 0.130 arcsec

Figure 5. Continued.

* elevation = 75 *

INTERNAL ATMOSPHERE

					-6					
		-8	-4	-2	-1	-2	-4	-8		
	-10	-5	-1	1	2	1	-1	-5	-10	
	-9	-4	-0	2	3	2	-0	-4	-9	
	-10	-5	-1	2	2	2	-1	-5	-10	
-20	-13	-7	-3	-1	0	-1	-3	-7	-13	-20
	-18	-12	-8	-5	-4	-5	-8	-12	-18	
	-24	-18	-14	-12	-11	-12	-14	-18	-24	
	-33	-27	-23	-20	-20	-20	-23	-27	-33	
		-38	-34	-31	-30	-31	-34	-38		
					-44					

bias=-12 micron; slope -3.7 micron/point; sigma 7 micron;
angle rotation=-0.076 arcsec

EXTERNAL ATMOSPHERE

					-3					
		-2	-2	-2	-2	-2	-2	-2		
	-2	-2	-2	-2	-2	-2	-2	-2	-2	
	-2	-2	-2	-2	-2	-2	-2	-2	-2	
	-1	-1	-1	-1	-1	-1	-1	-1	-1	
0	0	0	0	0	0	0	0	0	0	0
	1	1	1	1	1	1	1	1	1	
	3	3	3	3	3	3	3	3	3	
	4	4	4	4	4	4	4	4	4	
		6	6	6	6	6	6	6		
					8					

bias=1 micron; slope 1.1 micron/point; sigma 1 micron;
angle rotation= 0.022 arcsec

INTERNAL ATMOSPHERE + EXTERNAL ATMOSPHERE

					-8					
		-10	-7	-4	-4	-4	-7	-10		
	-13	-7	-3	-1	-0	-1	-3	-7	-13	
	-11	-6	-2	1	1	1	-2	-6	-11	
	-11	-6	-2	1	2	1	-2	-6	-11	
-20	-13	-7	-3	-1	0	-1	-3	-7	-13	-20
	-16	-11	-7	-4	-3	-4	-7	-11	-16	
	-22	-16	-12	-9	-8	-9	-12	-16	-22	
	-29	-23	-19	-16	-15	-16	-19	-23	-29	
		-32	-28	-25	-24	-25	-28	-32		
					-36					

bias=-11 micron; slope -2.6 micron/point; sigma 6 micron;
angle rotation=-0.054 arcsec

Figure 5. Continued.

* elevation = 90 *

INTERNAL ATMOSRHERE

					22					
		14	17	20	20	20	17	14		
	6	11	15	17	18	17	15	11	6	
	1	6	10	12	13	12	10	6	1	
	-5	1	4	7	7	7	4	1	-5	
-19	-12	-7	-3	-1	0	-1	-3	-7	-12	-19
	-21	-16	-12	-10	-9	-10	-12	-16	-21	
	-31	-26	-22	-20	-19	-20	-22	-26	-31	
	-43	-38	-34	-32	-31	-32	-34	-38	-43	
		-51	-47	-45	-44	-45	-47	-51		
					-59					

bias=-10 micron; slope -8.1 micron/point; sigma 6 micron;
angle rotation=-0.167 arcsec

EXTERNAL ATMOSPHERE

					0					
		0	0	0	0	0	0	0		
	0	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	0	
		0	0	0	0	0	0	0		
					0					

bias=0 micron; slope 0.0 micron/point; sigma 0 micron;
angle rotation= 0.000 arcsec

INTERNAL ATMOSPHERE + EXTERNAL ATMOSPHERE

					22					
		14	17	20	20	20	17	14		
	6	11	15	17	18	17	15	11	6	
	1	6	10	12	13	12	10	6	1	
	-5	1	4	7	7	7	4	1	-5	
-19	-12	-7	-3	-1	0	-1	-3	-7	-12	-19
	-21	-16	-12	-10	-9	-10	-12	-16	-21	
	-31	-26	-22	-20	-19	-20	-22	-26	-31	
	-43	-38	-34	-32	-31	-32	-34	-38	-43	
		-51	-47	-45	-44	-45	-47	-51		
					-59					

bias=-10 micron; slope -8.1 micron/point; sigma 6 micron;
angle rotation=-0.167 arcsec

Figure 5. Continued.

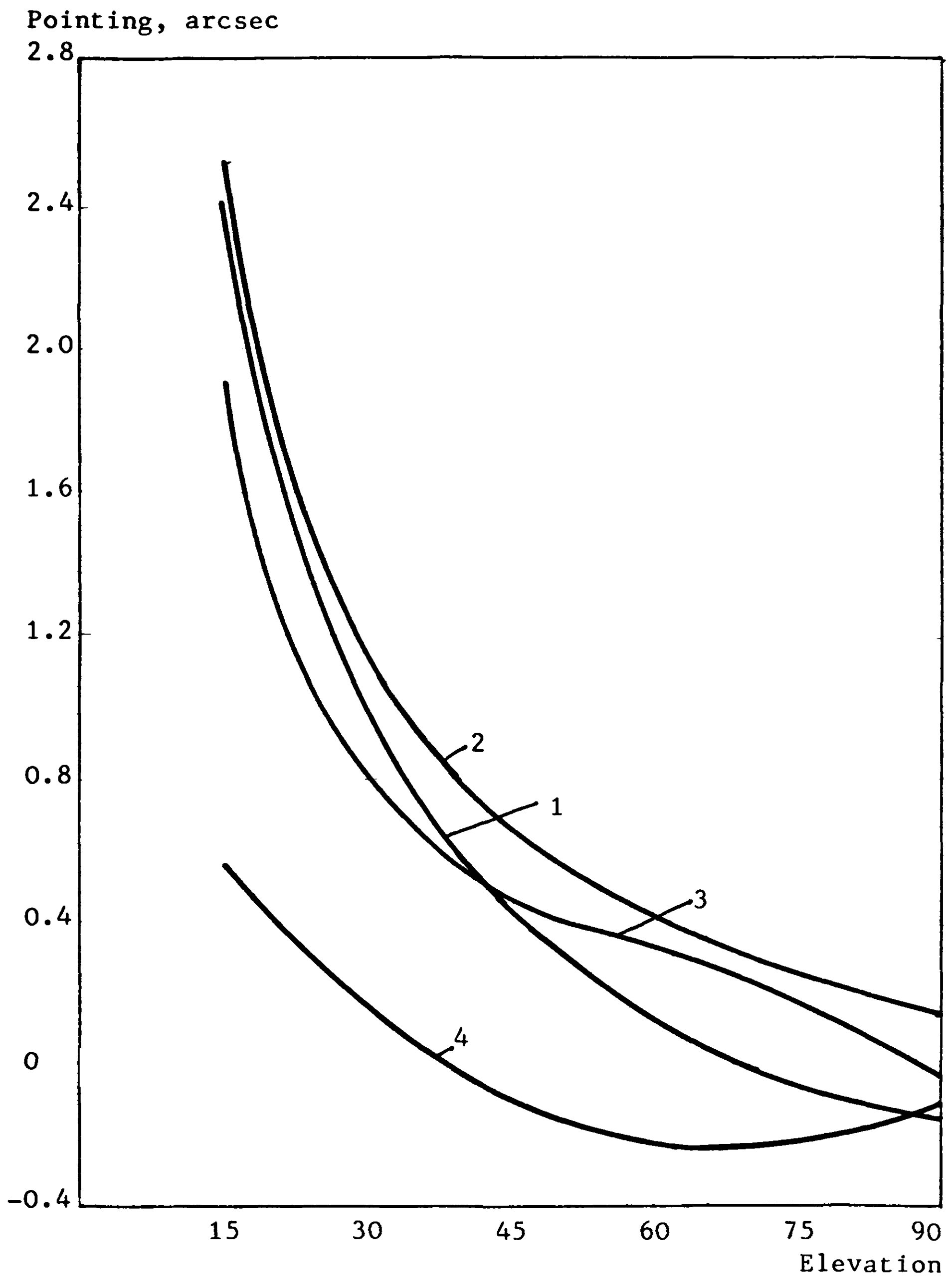

Figure 6. Dependence of pointing error upon elevation.

1. Internal + external atmosphere - "focus UP"
2. The same, but "focus DOWN"
3. Laser range measurment system (RS)
4. Difference between "focus UP" and RS

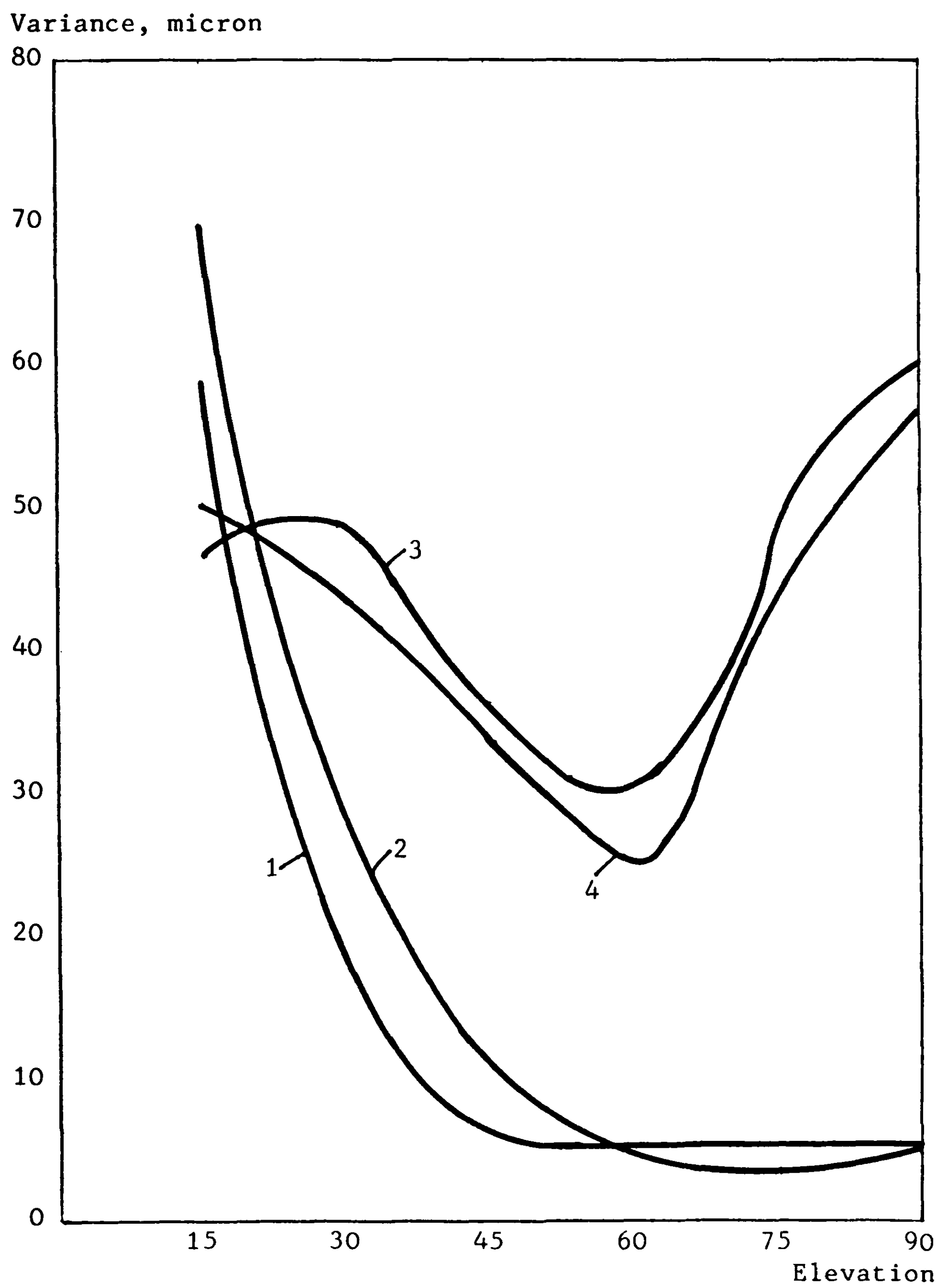

Figure 7. Dependence of phase variance upon elevation.

1. Internal + external atmosphere - "focus UP"
2. The same, but "focus DOWN"
3. Laser range measurement system (RS)
4. Difference between "focus UP" and RS

elev = 15 degrees

					-130					
		130	4	-43	-57	-43	4	130		
	-119	-117	-109	-100	-96	-100	-109	-117	-119	
	-90	-83	-74	-67	-64	-67	-74	-83	-90	
	-61	-51	-42	-35	-33	-35	-42	-51	-61	
-43	-30	-19	-9	-2	0	-2	-9	-19	-30	-43
	1	15	25	32	34	32	25	15	1	
	34	49	61	68	70	68	61	49	34	
	67	84	96	104	107	104	96	84	67	
		118	132	141	143	141	132	118		
					179					

bias=4 micron; slope 24.9 micron/point; sigma 47 micron;
angle rotation= 0.513 arcsec

					-9					
		226	100	53	39	53	100	226		
	-48	-46	-38	-29	-25	-29	-38	-46	-48	
	-44	-37	-28	-21	-18	-21	-28	-37	-44	
	-40	-30	-21	-14	-12	-14	-21	-30	-40	
-46	-34	-22	-12	-6	-4	-6	-12	-22	-34	-46
	-27	-14	-3	4	6	4	-3	-14	-27	
	-19	-4	7	14	17	14	7	-4	-19	
	-11	6	18	26	29	26	18	6	-11	
		15	29	37	40	37	29	15		
					51					

elev = 30 degrees

					-106					
		165	47	-1	-16	-1	47	165		
	-114	-109	-96	-84	-79	-84	-96	-109	-114	
	-91	-78	-64	-53	-49	-53	-64	-78	-91	
	-69	-52	-37	-27	-23	-27	-37	-52	-69	
-70	-50	-30	-14	-4	0	-4	-14	-30	-50	-70
	-33	-12	5	16	20	16	5	-12	-33	
	-21	3	21	33	37	33	21	3	-21	
	-14	12	31	44	48	44	31	12	-14	
		14	35	48	53	48	35	14		
					49					

bias=-17 micron; slope 9.9 micron/point; sigma 49 micron;
angle rotation= 0.204 arcsec

					-40					
		221	103	55	40	55	103	221		
	-67	-62	-50	-37	-32	-37	-50	-62	-67	
	-54	-41	-27	-16	-12	-16	-27	-41	-54	
	-42	-26	-11	-0	4	-0	-11	-26	-42	
-53	-33	-14	3	13	17	13	3	-14	-33	-53
	-26	-5	12	23	27	23	12	-5	-26	
	-24	-0	18	30	34	30	18	-0	-24	
	-27	-1	18	31	35	31	18	-1	-27	
		-8	12	25	30	25	12	-8		
					17					

Figure 8. Phase error distribution through the aperture in microns recalculated from the error of the laser range measurement system. The parameters of the atmosphere are the same as shown in Figure 5.

elev = 45 degrees

					-71					
		103	31	-10	-25	-10	31	103		
	-91	-81	-63	-46	-39	-46	-63	-81	-91	
	-78	-60	-41	-26	-21	-26	-41	-60	-78	
	-68	-47	-27	-13	-8	-13	-27	-47	-68	
-88	-64	-39	-19	-5	0	-5	-19	-39	-64	-88
	-65	-39	-18	-4	1	-4	-18	-39	-65	
	-74	-47	-25	-11	-6	-11	-25	-47	-74	
	-91	-63	-41	-27	-22	-27	-41	-63	-91	
		-89	-66	-52	-47	-52	-66	-89		
					-83					

bias=-40 micron; slope -4.0 micron/point; sigma 36 micron;
angle rotation=-0.083 arcsec

					-51					
		126	55	14	-1	14	55	126		
	-63	-53	-35	-19	-12	-19	-35	-53	-63	
	-46	-28	-9	5	11	5	-9	-28	-46	
	-33	-11	9	23	28	23	9	-11	-33	
-49	-24	0	21	35	40	35	21	0	-24	-49
	-22	4	26	40	45	40	26	4	-22	
	-26	1	23	37	42	37	23	1	-26	
	-39	-12	11	25	30	25	11	-12	-39	
		-33	-11	4	8	4	-11	-33		
					-24					

elev = 60 degrees

					-32					
		-37	-34	-61	-73	-61	-34	-37		
	-55	-42	-19	1	10	1	-19	-42	-55	
	-55	-35	-13	4	10	4	-13	-35	-55	
	-57	-34	-13	2	8	2	-13	-34	-57	
-86	-64	-41	-20	-5	0	-5	-20	-41	-64	-86
	-78	-54	-34	-20	-15	-20	-34	-54	-78	
	-99	-76	-57	-43	-39	-43	-57	-76	-99	
	-127	-105	-87	-75	-71	-75	-87	-105	-127	
		-142	-125	-114	-110	-114	-125	-142		
					-158					

bias=-53 micron; slope -10.8 micron/point; sigma 30 micron;
angle rotation=-0.223 arcsec

					-33					
		-28	-24	-51	-64	-51	-24	-28		
	-35	-21	1	21	30	21	1	-21	-35	
	-24	-4	18	35	41	35	18	-4	-24	
	-15	8	29	44	50	44	29	8	-15	
-33	-11	12	33	48	53	48	33	12	-11	-33
	-14	9	29	43	48	43	29	9	-14	
	-24	-2	18	31	36	31	18	-2	-24	
	-42	-20	-2	10	15	10	-2	-20	-42	
		-46	-29	-18	-14	-18	-29	-46		
					-51					

Figure 8. Continued.

elev = 75 degrees

					3					
		-181	-104	-116	-126	-116	-104	-181		
	-19	-5	19	41	50	41	19	-5	-19	
	-29	-11	10	26	33	26	10	-11	-29	
	-38	-20	-1	13	18	13	-1	-20	-38	
-65	-51	-34	-17	-5	0	-5	-17	-34	-51	-65
	-68	-53	-38	-27	-23	-27	-38	-53	-68	
	-89	-76	-64	-55	-52	-55	-64	-76	-89	
	-114	-104	-94	-87	-85	-87	-94	-104	-114	
		-135	-128	-122	-121	-122	-128	-135		
					-159					

bias=-51 micron; slope -10.0 micron/point; sigma 48 micron;
angle rotation=-0.207 arcsec

					4					
		-170	-93	-105	-115	-105	-93	-170		
	2	16	40	62	71	62	40	16	2	
	2	20	41	57	64	57	41	20	2	
	3	21	40	54	59	54	40	21	3	
-14	0	17	34	47	51	47	34	17	0	-14
	-7	9	23	34	38	34	23	9	-7	
	-18	-5	7	16	19	16	7	-5	-18	
	-33	-23	-13	-6	-3	-6	-13	-23	-33	
		-43	-36	-31	-29	-31	-36	-43		
					-57					

elev = 90 degrees

					30					
		-262	-138	-138	-144	-138	-138	-262		
	9	21	42	62	70	62	42	21	9	
	-7	6	23	37	42	37	23	6	-7	
	-18	-7	7	17	21	17	7	-7	-18	
-38	-32	-21	-11	-3	0	-3	-11	-21	-32	-38
	-46	-39	-30	-25	-23	-25	-30	-39	-46	
	-60	-57	-52	-47	-46	-47	-52	-57	-60	
	-74	-75	-74	-71	-71	-71	-74	-75	-74	
		-94	-95	-96	-96	-96	-95	-94		
					-117					

bias=-38 micron; slope -5.9 micron/point; sigma 60 micron;
angle rotation=-0.122 arcsec

					39					
		-248	-123	-123	-130	-123	-123	-248		
	29	41	62	82	91	82	62	41	29	
	20	33	49	63	69	63	49	33	20	
	15	26	39	50	54	50	39	26	15	
1	6	17	27	35	38	35	27	17	6	1
	-1	5	14	19	22	19	14	5	-1	
	-10	-7	-2	3	4	3	-2	-7	-10	
	-18	-19	-18	-15	-15	-15	-18	-19	-18	
		-32	-33	-34	-34	-34	-33	-32		
					-49					

Figure 8. Continued.

* elevation = 15 *

					223					
		455	327	280	265	280	327	455		
	157	157	163	171	175	171	163	157	157	
	118	122	129	135	138	135	129	122	118	
	57	64	70	76	78	76	70	64	57	
-30	-22	-14	-7	-1	0	-1	-7	-14	-22	-30
	-113	-103	-96	-91	-90	-91	-96	-103	-113	
	-227	-217	-208	-203	-201	-203	-208	-217	-227	
	-360	-347	-339	-333	-331	-333	-339	-347	-360	
		-488	-478	-471	-470	-471	-478	-488		
					-635					

bias=-52 micron; slope -92.3 micron/point; sigma 50 micron; angle rotation=-1.903 arcsec

					-187					
		138	10	-37	-52	-37	10	138		
	-68	-68	-62	-54	-50	-54	-62	-68	-68	
	-15	-11	-4	2	5	2	-4	-11	-15	
	16	23	29	35	37	35	29	23	16	
22	30	38	45	51	52	51	45	38	30	22
	31	41	48	53	54	53	48	41	31	
	9	19	28	33	35	33	28	19	9	
	-32	-19	-11	-5	-3	-5	-11	-19	-32	
		-67	-57	-50	-49	-50	-57	-67		
					-122					

* elevation = 30 *

					70					
		322	202	153	137	153	202	322		
	18	20	30	41	45	41	30	20	18	
	7	16	28	37	40	37	28	16	7	
	-11	1	13	21	25	21	13	1	-11	
-53	-39	-24	-11	-3	0	-3	-11	-24	-39	-53
	-72	-56	-43	-34	-31	-34	-43	-56	-72	
	-118	-99	-85	-75	-72	-75	-85	-99	-118	
	-173	-153	-138	-128	-124	-128	-138	-153	-173	
		-215	-198	-188	-184	-188	-198	-215		
					-260					

bias=-29 micron; slope -38.9 micron/point; sigma 44 micron; angle rotation=-0.803 arcsec

					-95					
		196	76	27	11	27	76	196		
	-70	-68	-58	-47	-43	-47	-58	-68	-70	
	-42	-33	-21	-12	-9	-12	-21	-33	-42	
	-21	-9	3	11	15	11	3	-9	-21	
-24	-10	5	18	26	29	26	18	5	-10	-24
	-4	12	25	34	37	34	25	12	-4	
	-11	8	22	32	35	32	22	8	-11	
	-27	-7	8	18	22	18	8	-7	-27	
		-30	-13	-3	1	-3	-13	-30		
					-36					

Figure 9. Phase error distribution through the aperture in microns as a difference of errors in RF observations (Figure 5) and recalculated errors of the laser range measurement system. (Figure 8)

* elevation = 45 *

					25					
		188	113	71	55	71	113	188		
	-19	-13	2	17	23	17	2	-13	-19	
	-25	-11	4	17	21	17	4	-11	-25	
	-35	-19	-2	9	14	9	-2	-19	-35	
-69	-52	-32	-16	-4	0	-4	-16	-32	-52	-69
	-74	-54	-37	-26	-21	-26	-37	-54	-74	
	-105	-84	-67	-55	-51	-55	-67	-84	-105	
	-145	-123	-105	-94	-90	-94	-105	-123	-145	
		-171	-153	-142	-138	-142	-153	-171		
					-196					

bias=-37 micron; slope -25.3 micron/point; sigma 33 micron;
angle rotation=-0.521 arcsec

					-65					
		124	49	7	-9	7	49	124		
	-58	-52	-37	-22	-16	-22	-37	-52	-58	
	-39	-25	-10	3	7	3	-10	-25	-39	
	-23	-7	10	21	26	21	10	-7	-23	
-32	-15	5	21	33	37	33	21	5	-15	-32
	-12	8	25	36	41	36	25	8	-12	
	-18	3	20	32	36	32	20	3	-18	
	-32	-10	8	19	23	19	8	-10	-32	
		-33	-15	-4	-0	-4	-15	-33		
					-33					

* elevation = 60 *

					14					
		5	4	-25	-38	-25	4	5		
	-18	-10	9	27	35	27	9	-10	-18	
	-27	-12	6	21	26	21	6	-12	-27	
	-37	-19	-2	10	15	10	-2	-19	-37	
-66	-51	-34	-17	-4	0	-4	-17	-34	-51	-66
	-71	-53	-37	-25	-21	-25	-37	-53	-71	
	-96	-79	-64	-53	-50	-53	-64	-79	-96	
	-127	-111	-98	-88	-85	-88	-98	-111	-127	
		-149	-137	-129	-126	-129	-137	-149		
					-173					

bias=-43 micron; slope -17.1 micron/point; sigma 25 micron;
angle rotation=-0.354 arcsec

					-28					
		-20	-21	-50	-63	-50	-21	-20		
	-26	-18	1	19	27	19	1	-18	-26	
	-18	-3	15	30	35	30	15	-3	-18	
	-11	7	24	36	41	36	24	7	-11	
-23	-8	9	26	39	43	39	26	9	-8	-23
	-10	8	24	36	40	36	24	8	-10	
	-18	-1	14	25	28	25	14	-1	-18	
	-32	-16	-3	7	10	7	-3	-16	-32	
		-37	-25	-17	-14	-17	-25	-37		
					-44					

Figure 9. Continued.

* elevation = 75 *

					11					
		-171	-97	-112	-122	-112	-97	-171		
	-6	2	22	42	50	42	22	2	-6	
	-18	-5	12	25	32	25	12	-5	-18	
	-27	-14	1	12	16	12	1	-14	-27	
-45	-38	-27	-14	-4	0	-4	-14	-27	-38	-45
	-52	-42	-31	-23	-20	-23	-31	-42	-52	
	-67	-60	-52	-46	-44	-46	-52	-60	-67	
	-85	-81	-75	-71	-70	-71	-75	-81	-85	
		-103	-100	-97	-97	-97	-100	-103		
					-123					

bias=-40 micron; slope -7.4 micron/point; sigma 44 micron;
angle rotation=-0.153 arcsec

					14					
		-160	-86	-101	-111	-101	-86	-160		
	12	20	40	60	68	60	40	20	12	
	7	20	37	50	57	50	37	20	7	
	6	19	34	45	49	45	34	19	6	
-5	2	13	26	36	40	36	26	13	2	-5
	-4	6	17	25	28	25	17	6	-4	
	-12	-5	3	9	11	9	3	-5	-12	
	-22	-18	-12	-8	-7	-8	-12	-18	-22	
		-33	-30	-27	-27	-27	-30	-33		
					-46					

* elevation = 90 *

					8					
		-276	-155	-158	-164	-158	-155	-276		
	3	10	27	45	52	45	27	10	3	
	-8	0	13	25	29	25	13	0	-8	
	-13	-8	3	10	14	10	3	-8	-13	
-19	-20	-14	-8	-2	0	-2	-8	-14	-20	-19
	-25	-23	-18	-15	-14	-15	-18	-23	-25	
	-29	-31	-30	-27	-27	-27	-30	-31	-29	
	-31	-37	-40	-39	-40	-39	-40	-37	-31	
		-43	-48	-51	-52	-51	-48	-43		
					-58					

bias=-29 micron; slope 2.2 micron/point; sigma 57 micron;
angle rotation= 0.045 arcsec

					48					
		-239	-118	-121	-127	-121	-118	-239		
	38	45	62	80	87	80	62	45	38	
	25	33	46	58	62	58	46	33	25	
	18	23	34	41	45	41	34	23	18	
10	9	15	21	27	29	27	21	15	9	10
	1	3	8	11	12	11	8	3	1	
	-5	-7	-6	-3	-3	-3	-6	-7	-5	
	-9	-15	-18	-17	-18	-17	-18	-15	-9	
		-23	-28	-31	-32	-31	-28	-23		
					-40					

Figure 9. Continued.

```
/*********************************************************************
The program calculate atmospheric contribution to the length of way
from focus to apperture for different elevation. The cartesian
coordinates are used for location of calculated points at the
apperture. The points are located at the knots of a quadratic array
The plane z = slope*x + bias is fitted to the calculated values of
function deltal(x,y). The slope is interpretated as a pointing
problem of the atmosphere. Both function deltal(x,y) and delta(x,y)
-(slope*x+bias) can be typed at the file jout.dat. The points are
calculated and printed if they are inside of the circle apperture.
*********************************************************************/
#include <stdio.h>
#include <math.h >
double x00,y00,z00,slope,bias,sigma,diam,pi;
double ex,ey,ez,zen,zengrad,c,dh,x11,y11,z11,n,dn,xp,yp;
int k ;
[redacted]
{
void fitprint();
int intersect();
int parabsec();
int reflect();
int refract();
int kst,kk,indxy,index,ind,lind,kind,i,j,kx,ky,kkk,ly;
double zxy1[132],zxy[132];
double zk,l,diam,aa,d,dl,xxsurf,xsurf,ysurf,r,h,zkk,ck,lk,ll;
double pi=3.1415926535;
double dn1,n1,teta,fi,dnn,elev,alfa,x,y,z,scal,angle;
FILE *fp;
fp = fopen ("joutlm-.dat", "w");
/********************************************************************
elev - elevation of the antenna ( degree );
zen - zenith angle of the antenna ( degree );
dh - height of a layer ( meter );
dnn - gradient of the refraction coefficient ( 1/meter );
c - focus distance of the antenna (meters );
k - the number of the layers;
*********************************************************************/
alfa = pi/180.0;
/* about antenna */
c = ck = 60.0; diam = 100.0; aa = 4.0;
d = diam +aa; zkk = d*d/(4*c); h = c - zkk;
dl = 10.0;
/* distance between measured points at the apperture at meters */
kx = ky = diam/2/dl;
/* about atmosphere */
dh = 10.0;  dnn = 0.00000003;
printf("height of the layer in meters; \n");
printf("gradient of refraction coefficient in 10 -8  1/meter; \n");
printf("+1 if focus-up version \n");
printf("or: -1 if the focus-down version \n");
scanf ("%lf",&dh); scanf ("%lf",&dnn); scanf ("%d",&lind);
dnn = dnn/100000000.0;
printf("height of the layer =  %4.1f \n",dh);
printf("gradient of refraction coefficient = %3.1e 1/meter\n",dnn);
dn1 = dnn*dh;
if (lind > 0)
{
printf("\n\n\n FOCUS - UP\n");
fprintf(fp,"\n\n\n FOCUS - UP\n");
```

```
}
else
{
printf("\n\n\n FOCUS - DOWN\n");
fprintf(fp,"\n\n\n FOCUS - DOWN\n");
}
printf("\n\n\n gradient of refraction = %3.1e 1/meter;\n",dnn);
printf("height of a layer = %2.0f meters\n",dh);
fprintf(fp,"\n\n\n gradient of refraction = %3.1e 1/meter;\n",dnn);
fprintf(fp,"height of a layer = %2.0f meters\n",dh);
elev =  15.0;
while (elev <= 90.0)
{
printf("*  elevation = %3.0f  *\n",elev);
fprintf(fp,"*  elevation = %3.0f  *\n",elev);
printf("                             INTERNAL  ATMOSPHERE  \n\n");
fprintf(fp,"                               INTERNAL  ATMOSPHERE  \n\n");
for (indxy = 0, lk = 0; indxy <= 1; indxy = indxy + 1)
{
/* We calculate reference point xsurf=ysurf=0, when indxy = 0. */
 for (i = -indxy*kx; i <= indxy*kx; i++)
 {
 ly = sqrt(kx*kx - i*i);
 for (kkk = 1; kkk <= kx-ly; kkk++) { printf("      "); fprintf(fp,"
 xsurf = i*dl;
  for (j = -indxy*ly; j <= indxy*ly; j++)
  {
  ysurf = j*dl;
  n = 1.0000;
  xxsurf = xsurf + diam/2.0 + aa;
  fi = atan2(ysurf, xxsurf);
  r = sqrt(ysurf*ysurf + xxsurf*xxsurf);
  teta = atan (r/(ck - r*r/(4*ck)));
  zen = lind*( 90.0 - elev ) * alfa;
  zengrad = lind*(90.0 - elev);
  kk = 0; index = 0;
  c = 60.0; zk = d*d/(4*c); x00 = y00 = 0; z00 = c;
  /* if the wave moves to parabola kk = 0   */
  /* if the wave moves from parabola kk = 1 */
  ex = sin(teta)*cos(fi); ey = sin(teta)*sin(fi); ez = -cos(teta);
  /*
  scal = -ex*sin(zen) + ez*cos(zen) is scalar product (e.n).
  'n'- normal vector to the layer plane. If this scalar product
  is positive than angle between vectors 'e' and "n" is acute and
  the ray goes from down to up; k,n are decreased from a layer
  to a layer. If this scalar product is negative than we have
  opposite situation;
  */
  scal = -ex*sin(zen) + ez*cos(zen); ind = (scal < 0.0) ? 1 : -1;
  dn = ind*dn1; k = ind;
  l = 0.0;
   do
   {
   intersect(); x = x11; y = y11; z = z11;
       if(!kk)
       {
          if ( k == 1000 ) z = -1;
          /* k = 1000 corresponds to parallel ray and layer
             (look intersect) Negative z = -1 garanties just
             intersection with parabola but not with layer */
```

```
        parabsec();
        if(z > z11)
           {
           n1 = n;
           refract();
           k = k + ind; n = n + dn;
           }
        else
           {
           xp = x11; yp = y11;
           n1 = n;
           reflect(); x = x11; y = y11; z = z11;
           scal = -ex*sin(zen) + ez*cos(zen);
           ind = (scal < 0.0) ? 1 : -1;
           dn = ind*dn1; k = k + ind;
           kk = kk + 1;
           }
       }
       else
       {
        if ( k == 1000 ) z = 1000;
        /*Such a large positive z = 1000 garanties just
        intersection with plane z = zk but not with layer */
        if (z < zk)
        {
        n1 = n;
        refract();
        k = k + ind; n = n + dn;
        }
        else
        {
        n1 = n;
        zen = 0; k = 0; c = zk;
        intersect(); x = x11; y = y11; z = z11;
        }
       }
   l+=sqrt((x-x00)*(x-x00)+(y-y00)*(y-y00)+(z-z00)*(z-z00))*n1;
   x00 = x; y00 = y; z00 = z;
   }
   while (c != zk);
   ll = (l - lk)*1000000.0;
   if(indxy)
   {
   zxy[(i+ kx)*(2*kx +1) + j +kx] = ll;
   printf ("%6.0f", zxy[(i+ kx)*(2*kx +1) + j +kx]);
   fprintf (fp,"%6.0f", zxy[(i+ kx)*(2*kx +1) + j +kx]);
   }
  } /* end for j(y) */
  printf ("\n"); fprintf (fp,"\n");
 } /* end for i(x) */
 if (!indxy) lk = l;
} /* end for indxy */
fitprint (fp, zxy, kx);
printf("\n\n                    EXTERNAL  ATMOSRHERE  \n\n");
fprintf(fp,"                    EXTERNAL  ATMOSRHERE  \n\n");
/***beginning of calculating of atmosphere upper apperture***/
zen = ( 90.0 - elev ) * alfa;
c = 0.0; /* it is to cross of zero-layer (k=0)
            with origin coordinate */
for (indxy = 0; indxy <= 1; indxy = indxy + 1)
```

```
{
 for (i = -indxy*kx; i <= indxy*kx; i++)
 {
 ly = sqrt(kx*kx - i*i);
 for (kkk = 1; kkk <= kx-ly; kkk++) { printf("      ");
                                      fprintf(fp,"      ");}
 xsurf = i*dl; xxsurf = xsurf + diam/2.0;
 kst = xxsurf/dh*sin(zen);
   for (j = -indxy*ly; j <= indxy*ly; j++)
   {
    ysurf = j*dl; y00 == ysurf; x00 = xxsurf; z00 = 0.0;
    ex = ey = 0.0; ez = 1.0;
    n = 1.0000 - dn*kst;
    for (k = kst, l = 0.0; k >= 0; k--)
     {
     intersect(); x = x11; y = y11; z = z11;
     refract();
     n1 = n; n = n + dn;
     l+=sqrt((x-x00)*(x-x00)+(y-y00)*(y-y00)+(z-z00)*(z-z00))*n1;
     x00 = x; y00 = y; z00 = z;
     }
    l -= xxsurf*tan(zen);
    if(indxy)
    {
    ll = (l - lk)*1000000.0;
    zxyl[(lind*i+ kx)*(2*kx +1) + j +kx] = ll;
    printf ("%6.0f", ll); fprintf (fp,"%6.0f",ll);
    }
   }
 printf ("\n"); fprintf (fp,"\n");
 }
 if (!indxy) lk = l;
}
/*** end of calculating of external atmosphere
fit plane into the data - zxy[(i + kx)*(2*ky + 1) +j +ky] ***/
fitprint (fp, zxyl, kx);
printf("\n\n            INTERNAL  ATMOSRHERE + ");
printf("EXTERNAL ATMOSPHERE \n\n");
fprintf(fp,"\n\n            INTERNAL  ATMOSRHERE + ");
fprintf(fp,"EXTERNAL ATMOSPHERE \n\n");
for (i = -kx; i <= kx; i++)
{
ly = sqrt(kx*kx - i*i);
for (kkk = 1; kkk <= kx - ly; kkk++) {printf("      ");
                                      fprintf(fp,"      ");}
for (j = -ly; j <= ly; j++)
 {
 zxy[(i+kx)*(2*kx+1)+j+kx] += zxyl[(i+kx)*(2*kx+1)+j+kx];
 printf ("%6.0f", zxy[(i + kx)*(2*kx + 1) +j +kx]);
 fprintf (fp,"%6.0f", zxy[(i + kx)*(2*kx + 1) +j +kx]);
 }
printf ("\n"); fprintf (fp,"\n");
}
fitprint (fp, zxy, kx);
elev = elev + 15.0;
}
}
```

```
/********************************************************************
The program calculate atmosphere contribution to the measurement
of distances from each of the three rangefinder to points at the
antenna surface and then recalculate it to the phase error at the
apperture.
********************************************************************/
#include <stdio.h>
#include <math.h >
double l,x00,y00,z00,slope,bias,sigma,diam,pi;
double ex,ey,ez,zen,zengrad,c,dh,x11,y11,z11,n,dn,dnn;
int k ;
[redacted]
{
void fitprint();
void rangef();
void matrix3();
int m,indxy,i,j,kx,kkk,ly,num,indzen;
double xrf[] = { 0.714,  10.181,  10.181, 18.047, 1.0 };
double yrf[] = { 0.0,    28.973, -28.973, 41.180, 0.0 };
double zrf[] = { 66.795, 24.095,  24.095, 14.851, 18.2 };
double pi=3.1415926535, zxy[132];
double diam,aa,d,h,dl,xxsurf,xsurf,ysurf,zsurf,r,lk,ll,ls;
double fi,outp,lideal, ck,rck,rsurf2,length,teta;
double elev,alfa,ax[5],ay[5],az[5],xyz[3],range[5],range0[5];
FILE *fp;
fp = fopen ("ranglm.dat", "w");
alfa = pi/180.0;
/* about antenna */
c = ck = 60.0; diam = 100.0; aa = 4.0;
d = diam +aa;
dl = 10.0;
/* distance between points at the apperture at meters */
kx = diam/2/dl;
/* about atmosphere */
dh = 1.0;  dnn = 0.00000003;  indzen = 1;
elev = 15.0;
while (elev <= 90.0)
{
zen = indzen*(90 - elev);
fprintf (fp,"\n\n elev = %4.0f degrees \n",elev);
printf ("\n\n elev = %4.0f degrees \n",elev);
zen = zen*alfa;
for (indxy = 0, lk = 0; indxy <= 1; indxy = indxy + 1)
 {
 /* We calculate reference point xsurf=ysurf=0, when indxy= 0 */
 for (i = -indxy*kx; i <= indxy*kx; i++)
  {
  ly = sqrt(kx*kx - i*i);
  for (kkk = 1; kkk <= kx-ly; kkk++) { printf("      "); fprintf(fp,"
  xsurf = i*dl;
  for (j = -indxy*ly; j <= indxy*ly; j++)
   {
   ysurf = j*dl;
   xxsurf = xsurf + diam/2.0 + aa;
   rsurf2 = (xxsurf*xxsurf + ysurf*ysurf);
   zsurf = rsurf2/(4*ck);
   rck = sqrt (rsurf2 + 4*ck*ck);
   for (m = 0; m <= 2; m++)
   {
   rangef (xrf[m], yrf[m], zrf[m], xxsurf, ysurf);
```

```
   if (indxy)
   {
   range[m] = 1 - range0[m];
   length = sqrt ((xrf[m]-xxsurf)*(xrf[m]-xxsurf) +
         (yrf[m]-ysurf) * (yrf[m]-ysurf) +
         (zrf[m]-zsurf) * (zrf[m]-zsurf));
   /*  Matrix for X0, Y0, n coordinates      */
   ax[m] = -((xrf[m]-xxsurf) + (zrf[m] - zsurf)
               *xxsurf/(2.0*ck))/length;
   ay[m] = -((yrf[m]-ysurf)  + (zrf[m] - zsurf)
               *ysurf/(2.0*ck))/length;
   az[m] = -((xrf[m]-xxsurf)*xxsurf/rck +
              (yrf[m]-ysurf)*ysurf/rck -
              (zrf[m]-zsurf)*2*ck/rck)/length;
    /*  Matrix for X, Y, Z coordinates          */
   /*
   ax[m] = -(xrf[m]-xxsurf)/length;
   ay[m] = -(yrf[m]-ysurf)/length;
   az[m] = +(zrf[m]-zsurf)/length;
   */
   }
   else range0[m] = 1;
   }
   /******************* " LEAST SQUARE " ****************/
   if(indxy)
     {
     matrix3 (ax,ay,az,range,xyz);
     fi = atan2(ysurf, xxsurf);
     teta = atan2(sqrt(rsurf2), (ck - zsurf));
     /*
     Recalculated error at the apperture is equal error at the
     normal direction  multiplied by double scalar product of vector
     e - from the focus to the point (xxsurf,ysurf) at the surface
     and normal vector (-xxsurf/rck, -ysurf/rck, 2*c/rck);
     */
     outp = xyz[2]*2*(sin(teta)*cos(fi)*xxsurf +
                      sin(teta)*sin(fi)*ysurf -
                      cos(teta)*(-2*ck))/rck;
     /*
     Recalculated error at the apperture is equal dz*(1 + cos(teta));
     dz is error at z-direction; teta is an angle between vector e
     and z-axis (2)
     */
     /* outp = xyz[2]*(1 + cos(teta)); */
     zxy[(i + kx)*(2*kx + 1) + j + kx] = outp;
     printf ("%6.0f",outp);
     fprintf (fp,"%6.0f",outp);
     }
   } /* end for j(y) */
  printf ("\n"); fprintf (fp,"\n");
  } /* end for i(x) */
} /* end for indxy */
fitprint (fp, zxy, kx);
elev += 15.0;
}
}
```

```
/**********************************************************************
The program reads (from file joutt.dat) the result of calculating
phase at the aperture from range finder measurements, reads (from
file joutt1.dat) the result of calculating fase at the aperture
from estimating of atmosphere influence on the RF work, subtract
and estimate common effect (joutt2.dat)
**********************************************************************/
#include <stdio.h>
#include <math.h >
double x00,y00,z00,slope,bias,sigma,diam;
double ex,ey,ez,zen,zengrad,c,dh,x11,y11,z11,n,dn,xp,yp;
int k ;
[redacted]
{
void fitprint();
int kst,kk,indxy,index,ind,lind,kind,i,j,kx=5,ky,kkk,ly;
double zxy[132];
int blank,blank1;
double zk,l,diam,aa,d,dl,xxsurf,xsurf,ysurf,r,h,zkk,ck,lk,ll;
double dn1,n1,teta,fi,dnn,elev,alfa,x,y,z,scal,angle;
FILE *fp;
FILE *fp1;
FILE *fp2;
fp = fopen ("joutt.dat", "r");
fp1 = fopen ("joutt1.dat", "r");
fp2 = fopen ("jrf-rg.dat", "w");
printf("      RANGEFINDER  &  RF EFFECTS      \n\n");
fprintf(fp2,"      RANGEFINDER  &  RF EFFECTS      \n\n");
elev = 15.0;
while (elev <= 90.0)
{
  printf("*********************\n");
  printf("*  elevation = %3.0f  *\n",elev);
  printf("*********************\n\n");
  fprintf(fp2,"*********************\n");
  fprintf(fp2,"*  elevation = %3.0f  *\n",elev);
  fprintf(fp2,"*********************\n\n");
  for (i = -kx; i <= kx; i++)
    {
    ly = sqrt(kx*kx - i*i);
    for (kkk = 1; kkk <= kx-ly; kkk++)
    { fprintf(fp2,"      "); printf("      ");}
    for (j = -ly; j <= ly; j++)
        {
        fscanf (fp,"%d", &blank);
        fscanf (fp1,"%d", &blank1);
        zxy[(i+ kx)*(2*kx +1) + j +kx] = blank1 -blank;
        printf ("%6.0f", zxy[(i+ kx)*(2*kx +1) + j +kx]);
        fprintf (fp2,"%6.0f", zxy[(i+ kx)*(2*kx +1) + j +kx]);
        }
    fscanf (fp,"\n");
    fscanf (fp1,"\n");
    printf ("\n");
    fprintf (fp2,"\n");
    }
  fitprint (fp2, zxy, kx);
  elev += 15.0;
}
}
```

```
/*****************************************************************
The program calculate atmosphere contribution to the measurement of
distance from the rangefinder (xr, yr, zr) to the point at the
antenna surface (xsurf, ysurf).
*****************************************************************/
#include <stdio.h>
#include <math.h >
extern double x00,y00,z00,pi;
extern double l,ex,ey,ez,zen,c,dh,x11,y11,z11,dnn,dn,n;
extern int k ;
[redacted] (double xr, double yr, double zr, double xsurf,
             double ysurf)
{
int intersect();
int parabsec();
int refract();
int ind;
double lideal,n0,ck,xs,ys,r,zk;
double dn1,n1,teta,fi,scal,x,y,z;
ck = c;
dn1 = dnn*dh;
ys = ysurf-yr; xs = xsurf-xr;
fi = atan2(ys, xs);
r = sqrt(ys*ys + xs*xs);
teta = atan2(r, (zr - r*r/(4*ck)));
x00 = xr; y00 = yr; z00 = zr;
ex = sin(teta)*cos(fi); ey = sin(teta)*sin(fi); ez = -cos(teta);
parabsec();
n0 = 1.0003;
lideal = sqrt ((xr-x11)*(xr-x11)+(yr-y11)*(yr-y11)
        +(zr-z11)*(zr-z11));
lideal *= n0;
/*
scal = -ex*sin(zen) + ez*cos(zen) is scalar product of vector 'e'
and 'n'- normal vector to the layer plane. If this scalar product
is positive than angle between vectors 'e' and 'n' is acute and
the ray goes from down to up - k,n are decreased from a layer to
a layer. If this scalar product is negative then we have opposite
situation;
*/
scal = -ex*sin(zen) + ez*cos(zen); ind = (scal < 0.0) ? 1 : -1;
dn = ind*dn1;
/* calculate the refraction coefficient at the point of the
rangefinder and number of corresponded layer */
k = -(xr*(-sin(zen)) + (zr - ck)*cos(zen))/dh;
n = n0 + k*dn1;
l = 0.0;
k = k + ind;
do
 {
 intersect(); x = x11; y = y11; z = z11;
 if ( k == 1000 ) z = -1;
 /* k = 1000 corresponds to parallel ray and layer (look intersect)
 Negative z = -1 garanties just intersection with parabola but
 not with layer */
 parabsec();
 if((ex*(x11-x) + ey*(y11-y) + ez*(z11-z)) > 0.0)
    {
    n1 = n;
    refract();
```

```
    k = k + ind; n = n + dn;
    }
 else
    {
    x = x11; y = y11; z = z11;
    n1 = n;
    c = 0;
    }
 l=l+sqrt((x-x00)*(x-x00)+(y-y00)*(y-y00)+(z-z00)*(z-z00))*n1;
 x00 = x; y00 = y; z00 = z;
 }
while (c);
l = (l -lideal)*1000000.0; c =ck;
}
```

```
#include <math.h>
extern double x00,y00,z00,ex,ey,ez,zen,zengrad,c,dh,x11,y11,z11;
extern int k ;
[redacted]
/*
The programm finds the intersection x11, y11, z11
of the line: (x - x00)/ex = (y - y00)/ey = (z - z00)/ez
with the plain: -x*sin(zen) + (z - c)*cos(zen) + dh*k = 0
*/
{
double den;
den = -ex*sin(zen) + ez*cos(zen);
if (den)
 {
 z11=((x00*ez - z00*ex)*sin(zen) + (c*cos(zen) - dh*k)*ez)/den;
 y11 = y00 + ey/ez*(z11-z00);
 x11 = x00 + ex/ez*(z11-z00);
 }
else
/* if den is equal zero then line is parallel to plane */
k = 1000;
}
```

```
#include <stdio.h>
#include <math.h>
extern double slope,bias,sigma,diam,pi;
[redacted] (FILE *fp, double zarray[], int nx)
/*
The programm fits the plane z = ax + b into the data ZARRAY[x,y]
minimizing SUM of squares (ZARRAY[x,y] - (ax + b)).
SLOPE is the found value of "a"
BIAS is the found value of "b"
SIGMA is the found value of "square root from minimum SUM
of squares"; Points are located at knots of quadratic array with
step = 1 and their area is limited by circle;
*/
{
int i,kkk,j,ii,kk,ny,sum5,sum4;
double ss,sum1,sum2,sum3,angle;
diam = 100.0; pi = 3.14159;
for (ii=0, sum1=0, sum3=0, sum4=0, sum5=0; ii <= 2*nx; ii++)
{
  ny = sqrt(nx*nx - (ii - nx)*(ii - nx));
  for (kk = nx-ny, sum2 = 0; kk <= nx + ny; kk++)
    {
    sum1 += zarray[ii*(2*nx+1) + kk];
    sum2 += zarray[ii*(2*nx+1) + kk];
    sum4 += (ii-nx)*(ii-nx);
    sum5++;
    }
  sum3 += (ii - nx)*sum2;
}
bias = sum1/sum5;
slope = sum3/sum4;
/* Now lets find SIGMA = sqrt (sum(sqr(zarray -
(slope*(ii -nx) +bias)))/((2*nx+1)*(2*ny+1)) */
for (ii = 0, sum1 = 0; ii <= 2*nx; ii++)
{
ny = sqrt(nx*nx - (ii - nx)*(ii - nx));
for (kk = nx - ny; kk <= nx + ny; kk++)
  {
  ss = zarray[ii*(2*nx+1) + kk] - (slope*(ii - nx) + bias);
  sum1 += ss*ss;
  }
}
sigma = sqrt(sum1/sum5);
/*************PRINT ARRAY*******************************/
angle = slope*2*nx/(diam*1000000.0)*180.0/pi*60*60;
/* angle - rotation of the ray at argsec */
printf ("\n    bias");
printf ("%-4.0fmicron; slope %-4.1f micron/point;",bias,slope);
printf ("  sigma %-3.0f micron;\n",sigma);
printf ("                   angle rotation=");
printf ("%6.3f arcsec  \n\n", angle);
fprintf (fp,"\n    bias");
fprintf (fp,"%-4.0fmicron; slope %-4.1f micron/point;",bias,slope);
fprintf (fp,"  sigma %-3.0f micron;\n",sigma);
fprintf (fp,"                   angle rotation=");
fprintf (fp,"%6.3f arcsec  \n\n", angle);
for (i = -nx; i <= nx; i++)
{
ny = sqrt(nx*nx - i*i);
for (kkk = 1; kkk <= nx-ny; kkk++) { printf("      ");
```

```
                                        fprintf(fp,"        ");}
for (j = -ny; j <= ny; j++)
  {
  /* print the residual
  zarray[(i + nx)*(2*ky + 1) +j +ky] - slope*x -bias */.
  printf ("%6.0f",zarray[(i+nx)*(2*nx+1)+j+nx]-(slope*i+bias));
  fprintf (fp,"%6.0f",zarray[(i+nx)*(2*nx+1)+j+nx]-(slope*i+bias));
  }
printf ("\n");
fprintf (fp,"\n");
}
}


[redacted] (double *ax, double *ay, double *az,
              double *b, double *xyz)
/*****************************************************************
The program find the solution - xyz of the system of three linear
equation (ax)*x + (ay)*y + (az)*z = b;
*****************************************************************/
{
double d1,d2,d3;
d1 = (*(ay+1))*(*(az+2)) - (*(az+1))*(*(ay+2));
d2 = (*ay)*(*(az+2))     - (*az)*(*(ay+2));
d3 = (*ay)*(*(az+1))     - (*az)*(*(ay+1));
*xyz = (*b)*d1 - (*(b+1))*d2 + (*(b+2))*d3;
*xyz /= ((*ax)*d1 - (*(ax+1))*d2 + (*(ax+2))*d3);
*(xyz+1) = ((*b - (*ax)*(*xyz)) * (*(az+1)) -
   (*(b+1) - (*(ax+1))*(*xyz)) * (*(az))) / d3;
*(xyz+2) = ((*b) - (*ax)*(*xyz) - (*ay)*(*(xyz+1))) / (*az);
}

#include <math.h>
extern double ex,ey,ez,xp,yp,c;
[redacted]
{
/*****************************************************************
The programm finds the coordinates of reflect vector, reflected
from paraboloid surface z = (x*x + y*y)/(4*c) at point A {xp, yp,
(xp*xp +yp*yp)/4c)}
The input: ex, ey, ez - coordinates of input vector E
xp, yp - x,y coordinates of reflection point on paraboloid surface
c - focus distance of the pareaboloid
The output: ex, ey, ez - coordinates of reflacted ray-vector
reflected ray E2 = (E - 2*(E*N)*N)/|E2|
vector N {nx, ny, nz} is normal to the paraboloid surface
*****************************************************************/
double nx,ny,nz,mm,ne,ss;
mm = sqrt(xp*xp + yp*yp +4*c*c);
nx = -xp/mm; ny = -yp/mm;  nz = 2*c/mm;
ne = ex*nx + ey*ny + ez*nz;
ex = ex - 2*ne*nx;  ey = ey - 2*ne*ny;  ez = ez - 2*ne*nz;
mm = sqrt(ex*ex + ey*ey + ez*ez);
ex = ex/mm; ey = ey/mm; ez = ez/mm;
}
```

```
/*****************************************************************
The programm finds the intersection x11, y11, z11
of the line: (x - x00)/ex = (y - y00)/ey = (z - z00)/ez
with paraboloid z = (x*x + y*y)/(4*c)
Input: x00,y00,z00,ex,ey,ez,c
Output: x11,y11,z11
*****************************************************************/
#include <math.h>
extern double x00,y00,z00,ex,ey,ez,c,x11,y11,z11;
[redacted]
{
double a,b,cc,d,x10,x12,y10,z10,z12;
a = ex*ex + ey*ey;
if (a)
{
if (ez)
  {
  b = ez*(ex*x00 + ey*y00 - 2*ez*c);
  cc = ez*ez*(x00*x00 + y00*y00 -4*c*z00);
  d = sqrt(b*b - a*cc);
  z10 = (-b + d)/a;  z12 = (-b - d)/a;
  z11 = (z10/ez > 0.0) ? z10 : z12;
  x11 = ex/ez*z11 + x00;
  y11 = ey/ez*z11 + y00;
  z11 = z11 + z00;
  }
else
  {
  z11 = z00;
  if (ex)
   {
   b = ex*ex*x00 + ex*ey*y00;
   cc = ex*ex*(x00*x00 + y00*y00 - 4*c*z00);
   d = sqrt(b*b - a*cc);
   x10 = (-b + d)/a;  x12 = (-b - d)/a;
   x11 = (x10/ex > 0.0) ? x10 : x12;
   y11 = ey/ex*x11 + y00;
   x11 = x11 + x00;
   }
  else
   {
   x11 = x00; z11 =z00;
   y10 = sqrt(4*c*z00 - x00*x00);
   y11 = ((y10 - y00)/ey > 0.0) ? y10 : -y10;
   }
  }
}
else
{
z11 = (x00*x00 + y00*y00)/(4*c);
x11 = x00;
y11 = y00;
}
}
```

```
#include <math.h>
extern double ex,ey,ez,zen,n,dn;
[redacted]
/*********************************************************************
The programm finds the coordinates of refract vector
The input: ex, ey, ez - coordinates of input vector E
   zen - zenith angle of the normal to the plane NOR
   n - coefficient of refraction of layer 1
   dn =n2 - n1
The output: ex, ey, ez - coordinates of output vector
*********************************************************************/
{
double mx,my,mz,nx,ny,nz,mm,ne,ss;
/*
vector M {mx, my, mz} is at the plane of vectors E and NOR
vectors M and NOR are perpendicular;
scalar product (M*E) > 0
M = (E - (NOR*E)*NOR)/|M|
*/
nx = -sin(zen); ny = 0; nz = cos(zen);
ne = ex*nx + ey*ny + ez*nz;
if (1 - fabs(ne))
{
mx = ex - ne*nx;
my = ey - ne*ny;
mz = ez - ne*nz;
mm = sqrt (mx*mx + my*my + mz*mz);
mx = mx/mm; my = my/mm; mz = mz/mm;
/*
refracted ray E2 = (E - (sin(alfa1) - sin(alfa2)*M)/|M|
alfa1 - angle between NOR and E;  alfa2 - angle between NOR and E2;
*/
ss = sqrt(1 - ne*ne)*dn/n;
ex = ex - ss*mx;
ey = ey - ss*my;
ez = ez - ss*mz;
mm = sqrt(ex*ex + ey*ey + ez*ez);
ex = ex/mm; ey = ey/mm; ez = ez/mm;
/*
}
}
```