THE UNIVERSITY OF MANITOBA

"COMPUTER-GRAPHICS REPRESENTATION OF IMAGES TRANSMITTED BY A REFRACTING ATMOSPHERE"

by

MOHAMED EL-BAKRY EL-ARINI

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF ELECTRICAL ENGINEERING

WINNIPEG, MANITOBA, R3T 2N2

February, 1978

COMPUTER-GRAPHICS REPRESENTATION OF IMAGES TRANSMITTED BY A REFRACTING ATMOSPHERE

BY

MOHAMED EL-BAKRY EL-ARINI

A dissertation submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

ABSTRACT

This thesis presents the representation of images transmitted under unusual atmospheric conditions. Interactive computer graphics are used to display directly the effect of atmospheric refraction on the transmitted images of objects at various distances.

Detailed equations are derived, that permit ray trajectories to be calculated for any given atmospheric temperature profiles. Two interpolation methods are discussed for fitting curves to the unequally-spaced temperature data.

The image representation programs (written in BASIC for a PDP11/40 computer with graphics terminal) are explained with the aid of flow charts.

The images of three objects at different horizontal distances from the observer can be represented simultaneously on the display unit, as well as the horizon line of such temperature profile.

ACKNOWLEDGEMENTS

I am especially grateful to my supervisor, Professor W.H. Lehn, who introduced me to the subject of image representation by using minicomputers, for his guidance, interest and encouragement during the work in this thesis.

Also, I would like to thank Mr. Neil-John Morphy of the Computer Center, University of Manitoba, for his assistance for using PDP11/40 minicomputer and its Display unit.

I am also grateful to Ms. Mary-Ann Tyler for her expert typing of the thesis in the minimum time.

I would like to thank all the teaching staff of the Department of Electrical Engineering, University of Manitoba.

The use of the computing facilities at the University of Manitoba, is also acknowledged.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

1.1 Introduction

This thesis develops the use of a minicomputer graphics terminal to reproduce the appearance of the environment under unusual atmospheric conditions. To the author's knowledge, this thesis is the first work that uses interactive graphics to directly display the result of atmospheric refraction on transmitted images of objects at various distances.

Some previous work has been done, representing images by hand, in which it is a tedious task to represent even one image. The only requirement for the computer to represent these images is to feed it the atmospheric temperature profile data and the horizontal distance between the object and observer. Also, by using the light pen of the minicomputer, any object with up to fifty vertices can be drawn on the screen.

This picture of the object is saved in the memory and its image for such a temperature profile is directly represented automatically on the computer's screen. Images for three objects at different horizontal distances from the observer can be represented on the screen at the same time.

A summary of the previous work, which is mostly related to the ray trajectories and refraction of these rays in the atmosphere, is presented in section (1.2).

Chapter 2 will discuss the image transmission under unusual atmospheric conditions. It includes the ray paths in the refracting atmosphere as well as the ray trajectories for four temperature profiles.

Chapter 3 will explain cubic Lagrange interpolation and spline fit methods which are used to interpolate the unequally spaced data of the temperature profile. The comparison between these two methods in our case will be discussed in the same chapter.

The transfer characteristic program flow chart and the results for four temperature profiles will be discussed in Chapter 4.

Chapter 5 discusses the resulting images for the four chosen temperature profiles. Images of some objects for every temperature profile will be represented in this chapter.

The summary, conclusions, limitations in using the programs and suggestions for future work will be explained in chapter 6.

1.2 Summary of Some of the More Important Results of the Previous Work

Nölke[N1] reported the first rigorous mathematical treatment of the Arctic Mirage (1917).

Liljequist[L2] described the temperature inversions and mirages at Maudheim (Antarctica). The important findings of this reference can be summarized as:

(a) If the vertical temperature-gradient is equal to +0.112 C/m for a surface temperature of 0 C, then the curvature for horizontal rays near sea level will be equal to that of the earth's surface. In other words; a ray of light proceeding horizontally from a certain point, will follow a path parallel to the earth's surface. With a temperature gradient of greater magnitude, horizontal rays near sea level will curve towards the earth's surface.

(b) A mathematical model for the light ray trajectories was developed as well as the vertex curve which determines the locus of the vertex points for all rays leaving (or

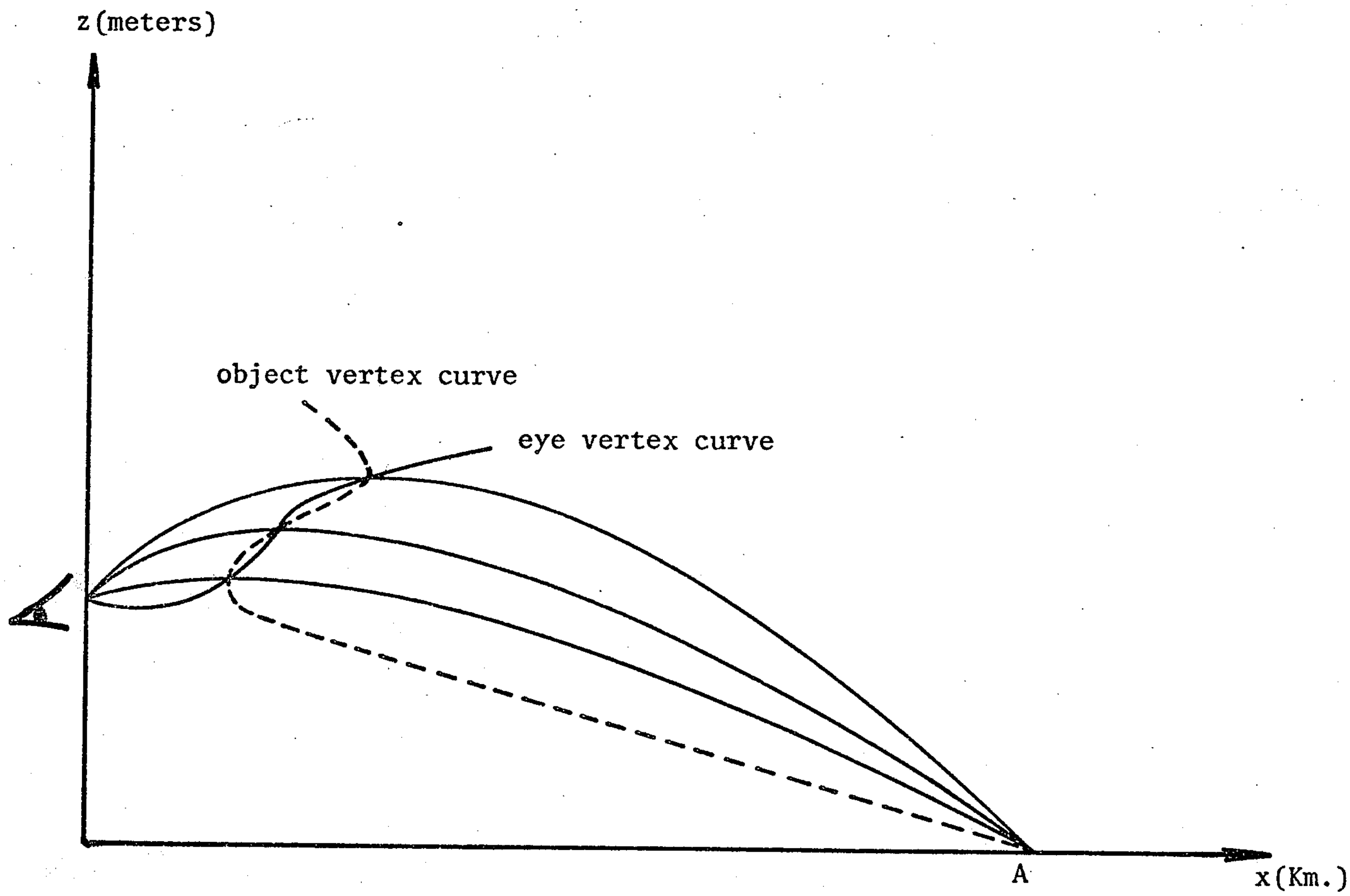


Fig. 1.1: The principle of vertex curves. The vertex curve is the locus of maxima of the light rays

reaching) the origin, Fig. 1-1.

(c) A ray from the object will reach the eye after being totally refracted if the vertex curves of the object and that for the eye intersect. There may even be more than one intersection, each one determining a vertex point of a possible light-trajectory from the object to the eye. If the object and the eye are at the same level, their vertex curves will be identical, though facing in opposite directions.

(d) The following three cases of temperature inversion were discussed:

(i) a linear temperature-increase with height;

(ii) a temperature-increase with height, which decreases in magnitude from the surface and upwards, which he referred to as "young" surface inversion;

(iii) a pronounced thermocline.

Hobbs[H3] discussed the conditions of exceptionally long range visibility within high latitudes, particularly as a result of superior mirage. Numerous cases were cited in which lines of sight ranged from 300 to 400 Km in length.

Lehn and Sawatzky[L1] concluded that the arctic mirage occurs when a temperature inversion produces a steepened density gradient in the lower atmosphere.

Equations relating temperature profile to density and density profile to nearly horizontal ray paths were developed. With the aid of these equations, the appearance of the environment can be predicted for any lower-atmosphere temperature profile. They concluded that the effects of the mirage are noticeable primarily over flat and relatively featureless terrain. For strong inversions near the surface, the observer sees a saucer-shaped earth, with objects near the horizon somewhat distorted. On the other hand, viewing over very great distances is made possible by mild deep inversions, which when located near the surface, produce the appearance of a flat earth.

The subject of the arctic mirage and the early North Atlantic had been discussed by Sawatzky and Lehn[S2]. The authors concluded that, the arctic mirage is a phenomenon that is common in higher latitudes. The authors contend that the arctic mirage was instrumental in the discovery of Greenland by the Icelanders, and in the discovery of North America from Greenland.

Manifestations of the arctic mirage, though largely forgotten in modern times, are described in the earliest accounts of North Atlantic discovery.

Fraser and Mach[F1] showed - with pictures - different types of mirages, for example:

(a) Inferior mirage; in which the image is displaced downward from the object.

(b) Towering; which means image is magnified. Inferior mirage with towering, results when temperature and temperature gradient are greatest at the surface and decrease with height.

(c) Two-image inferior mirage is the same as for inferior mirage with towering except that surface-temperature gradient is larger.

(d) Superior mirage results when the temperature increases with height so that the image is displaced up from the object.

(e) A three-image mirage can result from an inflection-point temperature profile. This profile often exists over an enclosed body of water on a sunny afternoon.

(f) The Fata Morgana mirage requires an inflection-point profile, but the temperature gradient near the inflection-point is slightly smaller than it is for a three-image mirage.

(g) Astigmatic three-image mirage produces an overhanging wall so blurred that no detail can be seen on it. The brightness, however, is redistributed so that the center of the wall is bright. This strip of brightness appears to the eye as a bank of fog and has been called the Fata Bromosa.

CHAPTER 2

IMAGE TRANSMISSION UNDER UNUSUAL ATMOSPHERIC CONDITIONS

The appearance of environment under unusual atmospheric conditions was taken as an image representation problem on a minicomputer display unit.

In this chapter, the ray trajectories for different kinds of unusual atmospheres are discussed, with graphs showing these ray trajectories.

First, the equations will be discussed, that describe the ray trajectory through the atmosphere.

The following sections develop the equations that permit calculation of ray paths when the temperature profile is known.

2.1 Ray Paths in the Refracting Atmosphere

Bertram[B1] and Lehn et al[L1] give the refractive index for visible light as a function of density $\rho(Kg/m^3)$ by:

$$n = 1 + 0.000226\rho \tag{2.1.1}$$

and the propagation velocity v by:

$$v = \frac{c}{n} \tag{2.1.2}$$

where c is the velocity of light in vacuum. Ref. [B1] derives also an expression for the radius of curvature r at a point on the ray, for an atmosphere over a flat earth,

$$\frac{1}{r} = \frac{\sin\theta(z)}{v(z)} \cdot \frac{dv}{dz} \tag{2.1.3}$$

where $\theta(z)$ is the angle between the ray and the vertical, and $v(z)$ is the velocity of propagation at elevation z.

Fig. 2.1 illustrates Snell's law of refraction. It shows a ray traversing a number of parallel layers having indices of refraction n_1, n_2, n_3, ... If we assume that the ray makes an angle θ_1 with respect to the normal to the planes in the first layer, the corresponding angles in the successive mediums are, in accordance with Snell's law, related by the expression:

$$n_1 \sin \theta_1 = n_2 \sin \theta_2 = n_3 \sin \theta_3 = \ldots\ldots$$

Using equation (2.1.2), this can be expressed in the form:

$$\frac{\sin\theta}{v} = A \qquad (2.1.4)$$

where θ is the vertical angle at a point where the velocity of propagation is v, and A is a constant for a given ray. By considering the successive layers to be of infinitesimal thickness, it is readily seen that Eqn. (2.1.4) is valid for the situation where v varies continuously.

The curvature of the ray expressed by Eqn. (2.1.3) provides a useful starting point for a calculation of the ray path.

Coordinates are chosen such that the ray is contained in the xz plane. As shown in Fig. 2.2, an additional set of coordinates facilitates further calculations. One axis (u) is tangent to the ray path at the ray origin and the second axis (w) pointed downward in the the plane of the ray.

Since the velocity is very nearly constant, the ray curvature is very small in the atmosphere; thus,

$$x \simeq u \sin \theta$$

$$z \simeq z_o + u \cos \theta \quad \text{and}$$

$$\left|\frac{dw}{du}\right| << 1$$

Then, the curvature expression becomes in the uw coordinates,

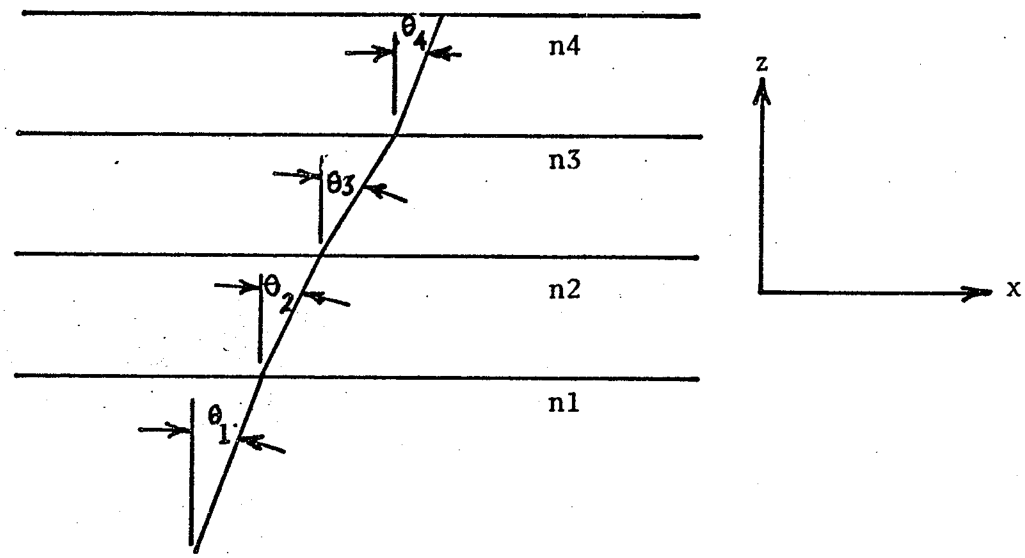


Fig. 2.1: Illustration of Snell's law of refraction
($n1 \sin \theta_1 = n2 \sin \theta_2 = n3 \sin \theta_3 = ...$)

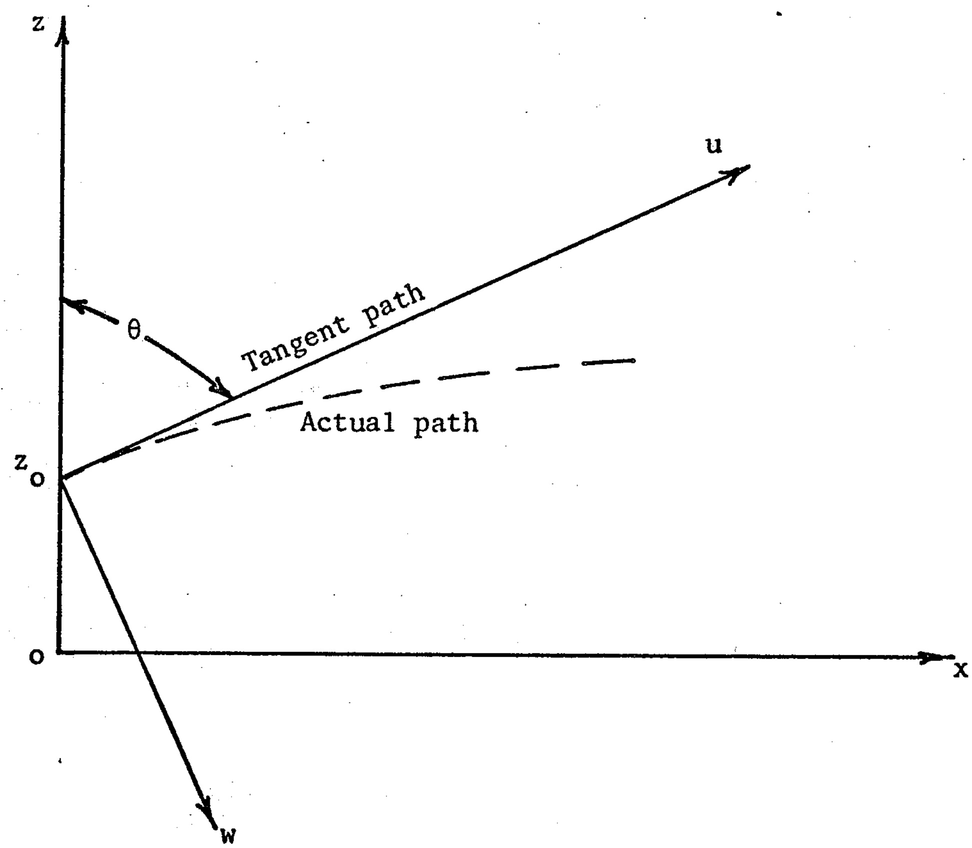


Fig. 2.2: Choice of coordinate systems

refs. [B1] and [L1]:

$$\frac{1}{r} = \frac{\frac{d^2w}{du^2}}{[1 + (\frac{dw}{du})^2]^{1.5}} \simeq \frac{d^2w}{du^2} \tag{2.1.5}$$

From Eqs. (2.1.3) and (2.1.5), therefore:

$$\frac{1}{r} \simeq \frac{d^2w}{du^2} = \frac{\sin\theta}{v} \frac{dv}{dz} \tag{2.1.6}$$

The radius of curvature, given by Eqn. (2.1.6), suggests that the refraction of a nearly horizontal ray over a path of given length should be fairly independent of the ray angle.

For moderate horizontal distances and nearly horizontal rays, the altitude variation of such rays is small. It is then useful to consider the velocity function to be expanded in a Taylor series:

$$\frac{dv}{dz} = v_o' + v_o''(z - z_o) + \frac{v_o'''}{2}(z - z_o)^2 + \ldots. \tag{2.1.7}$$

where:

$$v_o' = \frac{dv}{dz}, \quad v_o'' = \frac{d^2v}{dz^2}, \quad v_o''' = \frac{d^3v}{dz^3}, \ \ldots. \text{ etc.}$$

at the observation point.

Substitution of (2.1.7) into (2.1.6) gives:

$$\frac{d^2w}{du^2} \simeq \frac{\sin\theta}{v}\left(v_o' + v_o''(z - z_o) + \frac{1}{2}v_o'''(z - z_o)^2 + \ldots.\right)$$

Since $\frac{\sin\theta}{v}$ is constant for a given ray, then we can write $\frac{\sin\theta_o}{v_o}$ instead of $\frac{\sin\theta}{v}$ in the last expression.

Therefore;

$$\frac{d^2w}{du^2} \simeq \frac{\sin\theta_o}{v_o}\left(v_o' + v_o''(z - z_o) + \frac{1}{2}v_o'''(z - z_o)^2 + \ldots.\right) \tag{2.1.8}$$

Refs. [B1] and [L1] show the effects of the curvature of the earth on refraction. For horizontal distances under 1000 Km, the equation of the earth's surface is very close to:

$$z = -\frac{x^2}{2R}$$

where R is the radius of the earth.

In Eqn. (2.1.8), the ray elevation above the surface will thus increase by the additional term $\frac{x^2}{2R}$. Also, the angle θ_o between ray and local vertical will decrease by the amount $\frac{x}{R}$. Then Eqn. (2.1.8) will become:

$$\frac{d^2w}{du^2} = \frac{\sin(\theta_o - \frac{x}{R})}{v_o}\left(v_o' + v_o''(z + \frac{x^2}{2R} - z_o) + \frac{v_o'''}{2}(z + \frac{x^2}{2R} - z_o)^2 + \ldots.\right) \quad (2.1.9)$$

For nearly horizontal rays:

$$x \simeq u \quad \text{and}$$
$$z \simeq z_o + u\cos\theta_o - \frac{u^2}{2r_o} \quad (2.1.10)$$

The last term in the z-equation represents the effect of ray curvature; r_o is the radius of curvature at $u = 0$. Replacing r_o by its value from Eqn. (2.1.6), then:

$$z \simeq z_o + u\cos\theta_o - \frac{v_o'\sin\theta_o}{2v_o}u^2 \quad (2.1.11)$$

Also, by using Taylor series:

$$\sin(\theta_o - \frac{x}{R}) \simeq \sin\theta_o - \frac{u}{R}\cos\theta_o$$

Let $\qquad \phi_o = 90° - \theta_o$

$$\therefore \quad \sin(\theta_o - \frac{x}{R}) \simeq \sin\theta_o - \frac{u}{R}\sin\phi_o \quad (2.1.12)$$

By substituting Eqs. (2.1.10), (2.1.11) and (2.1.12) into Eqn. (2.1.9)

$$\frac{d^2w}{du^2} = \frac{(\sin\theta_o - \frac{u}{R}\sin\phi_o)}{v_o}\left[v_o' + v_o''(u\sin\phi_o - \frac{v_o'\sin\theta_o}{2v_o}u^2 + \frac{u^2}{2R}) + \frac{v_o'''}{2}(u\sin\phi_o - \frac{v_o'\sin\theta_o}{2v_o}u^2 + \frac{u^2}{2R})^2\right] \quad (2.1.13)$$

By integrating the last equation twice, the equation of the ray path in the uw coordinate system, can be produced. In each coefficient of u^n, any terms at least 1000 times smaller than their neighbours have been neglected.

$$
\begin{aligned}
w = \frac{\sin\theta_o}{v_o}\Big[& \frac{v_o'}{2}u^2 + \frac{v_o''\sin\phi_o}{6}u^3 \\
& + \frac{1}{12}\Big(\frac{v_o''}{2R} - \frac{v_o' v_o''\sin\theta_o}{2v_o} + \frac{v_o'''\sin^2\phi_o}{2}\Big)u^4 \\
& + \frac{1}{20}\Big(\frac{v_o'''\sin\phi_o}{2R} - \frac{v_o' v_o'''\sin\phi_o\sin\theta_o}{2v_o}\Big)u^5 \\
& + \frac{v_o'''}{60}\Big(\frac{v_o'^2\sin^2\theta_o}{4v_o^2} + \frac{1}{4R^2} - \frac{v_o'\sin\theta_o}{2v_oR}\Big)u^6\Big]
\end{aligned}
\tag{2.1.14}
$$

The subscripts "o" refer to values at the origin of the uw coordinates.

Eqn. (2.1.14) describes the path equation for nearly horizontal rays as a function of propagation-velocity gradient $\frac{dv}{dz}$.

2.2 Relation Between Velocity Gradient and Temperature Profile

The path equation (2.1.14) requires the derivatives of propagation velocity with respect to elevation. These derivatives may be calculated from a given temperature profile.

From Eqs. (2.1.1) and (2.1.2), we can find a relation between propagation velocity and density;

$$v = \frac{c}{1 + 0.000226\rho} \tag{2.2.1}$$

which to an accuracy better than one part in a thousand is equivalent to:

$$v \simeq c(1 - 0.000226\rho) \tag{2.2.2}$$

From this equation, the velocity derivatives can be calculated in terms of density derivatives, therefore:

$$
\begin{aligned}
v' &= -0.000226c\rho' \\
v'' &= -0.000226c\rho'' \quad \text{and} \\
v''' &= -0.000226c\rho'''
\end{aligned}
\tag{2.2.3}
$$

where

$$\rho' = \frac{d\rho}{dz}, \quad \rho'' = \frac{d^2\rho}{dz^2} \quad \text{and} \quad \rho''' = \frac{d^3\rho}{dz^3}$$

In the following paragraphs, the density derivatives can be calculated from a given temperature profile.

Ref. [L1] shows the equation of the pressure in an atmosphere situated in a gravitational field:

$$\frac{dp}{dz} = -g\,\rho(z) \tag{2.2.4}$$

where p is the pressure at elevation z, and g is the acceleration of gravity. Also, Ref. [L1] represents the behaviour of air for small deviations from standard temperature and pressure by the ideal gas equation

$$pV = n\bar{R}T$$

or

$$\rho = \frac{\beta p}{T} \tag{2.2.5}$$

where: T is the temperature in degrees Kelvin, $\bar{R}$ is the gas constant and β is the constant of proportionality.

Substitute from equation (2.2.5) into (2.2.4):

$$\frac{dp}{dz} = -\frac{g\beta p}{T(z)}$$

or

$$\frac{dp}{p} = -g\beta\,\frac{dz}{T(z)} \tag{2.2.6}$$

By integrating equation (2.2.6), then:

$$\int_{p_o}^{p(z)} \frac{dp_1}{p_1} = -g\beta \int_o^z \frac{dz_1}{T(z_1)} \tag{2.2.7}$$

where p_1 and z_1 are dummy variables for integration, and p_o is the atmospheric pressure at earth's surface. Therefore:

$$p(z) = p_o \exp[-g\beta \int_o^z \frac{dz_1}{T(z_1)}] \tag{2.2.8}$$

Substituting (2.2.8) into (2.2.5), then:

$$\rho(z) = \frac{\beta p_o}{T(z)} \exp[-g\beta \int_o^z \frac{dz_1}{T(z_1)}] \tag{2.2.9}$$

Equation (2.2.9) shows the relation between density and the temperature profile $T(z)$.

The next step is to get the first three derivatives of the density, using the notations

$$T' = \frac{dT(z)}{dz}, \quad T'' = \frac{d^2T(z)}{dz^2} \quad \text{and} \quad T''' = \frac{d^3T(z)}{dz^3}$$

for simplicity.

Differentiation of equation (2.2.9) produces:

$$\rho' = \beta p_o[-\frac{T'}{T^2} \exp(-g\beta \int_o^z \frac{dz_1}{T(z_1)}) - \frac{g\beta}{T^2} \exp(-g\beta \int_o^z \frac{dz_1}{T(z_1)})] \quad \text{Ref.[S1]}$$

or

$$\rho' = -(\frac{T' + g\beta}{T})[\frac{\beta p_o}{T} \exp(-g\beta \int_o^z \frac{dz_1}{T(z_1)})] \tag{2.2.10}$$

Substitution of equation (2.2.9) into (2.2.10), therefore:

$$\rho' = -\frac{\rho}{T}(T' + g\beta) \tag{2.2.11}$$

Differentiating (2.2.11) twice to get ρ'' and ρ''':

$$\rho'' = (\frac{T' + g\beta}{T})(\frac{\rho T'}{T} - \rho') - \frac{\rho T''}{T} \tag{2.2.12}$$

and:

$$\rho''' = (\frac{T' + g\beta}{T^3})(2TT'\rho' + \rho TT'' - T^2\rho'' - 2\rho T'^2)$$
$$+ \frac{1}{T^2}(-2TT''\rho' - \rho TT''' + 2\rho T'T'') \tag{2.2.13}$$

The calculations of ρ', ρ'' and ρ''' from the equations (2.2.11), (2.2.12) and (2.2.13) for the temperature profile are as follows:

1. A cubic polynomial is fitted to the neighbouring temperature points for a given temperature profile $T(z)$.

2. Calculate $T'(z)$, $T''(z)$ and $T'''(z)$ at altitude z from the cubic fitting curve.

3. Calculate $\rho(z)$ by using equation (2.2.9) (use numerical integration to evaluate the integral $\int_0^z \frac{dz_1}{T(z_1)}$.)

4. Calculate $\rho'(z)$, $\rho''(z)$ and $\rho'''(z)$ by using the equations (2.2.11), (2.2.12) and (2.2.13)

After calculation of the three derivatives of the density at altitude z, the velocity derivatives can be calculated by using equation (2.2.3).

Table 2.1 shows the numerical values used in calculations, Ref[L1]:

The constant of proportionality:	β	=	0.00349 MKS units
Atmospheric pressure at earth's surface:	p_o	=	1.013×10^5 n/m^2
Surface density of air at 15°C:	ρ_o	=	1.225 Kg/m^3
Effective molecular weight of air:	29.0		
Radius of the earth:	R	=	6400 Km
Velocity of light in vacuum:	C	=	3×10^8 m/sec.

TABLE 2.1

NUMERICAL VALUES FOR STANDARD ATMOSPHERE

2.3 Ray Trajectories for Some Strong Temperature Inversion

Some unusual temperature inversion profiles will be discussed in this section. These examples of strong inversion were chosen from measurement data collected in Reference [L2].

a. Temperature profile I:

This temperature profile starts with cold air on the surface (Elevation = 0 meters). Warm air lies over this cold air so that the temperature increases with increasing elevation up to an elevation of 50 meters. With elevation above 50 meters, the temperature decreases as for a normal atmosphere, with temperature gradient equal to -0.006°C/meter. The change in temperature for the first 50 meters above the surface is 10°C. Table 2.2 and Fig. 2.3 illustrate this temperature profile.

Ray elevation above the earth's surface is plotted against distance along the surface for this profile, as shown in Fig. 2.4. The main equation used for this plot is equation (2.1.14) with the aid of velocity derivatives as explained in section 2.2. Details of calculation techniques are discussed later in the section on transfer characteristics (Chapter 4).

The observer's eye is 3 meters above the surface. The numbers on the paths give ray angles, in degrees above the horizontal, at the observer's station (ϕ_o).

It is noticed from Fig. 2.4, that the rays hit the surface at horizontal distance X Km. For example, at initial elevation angle ϕ_o = 0 degrees, the distance x = 9 Km. The horizontal distance X increased with increasing the initial elevation angle ϕ_o, up to ϕ_o = 0.03°. With increasing ϕ_o above 0.03°, the horizontal surface distance from the observer at which the ray intersects the ground will decrease up to ϕ_o = 0.10°, and then, x will increase again.

It is clear from this figure that, the horizon is located at a horizontal distance greater than 75 Km., and at ϕ_o equal to 0.155 degrees. The comparable values for a normal atmosphere are 6.8 Km. for horizon distance, at an angle ϕ_o of about $-0.05°$.

Any ray that has an initial elevation angle larger than the horizon elevation angle ($\phi_o = 0.155°$), will diverge upward and not return to the surface.

We can notice also from this figure, that many rays intersect together at horizontal distances between 26 Km. and 45 Km. from the observer's station and at altitudes below 18 meters.

TEMPERATURE INPUT DATA

Z(Meters)	T Deg. C
0	5
1	5.6
2	5.9
3	6.1
4	6.24
6	6.52
10	7.1
15	8
20	9
25	10.5
30	12
35	13.2
40	14.1
45	14.7
50	15
55	14.97
60	14.94
65	14.91
70	14.88
80	14.82
100	14.7

TABLE 2.2

TEMPERATURE PROFILE I

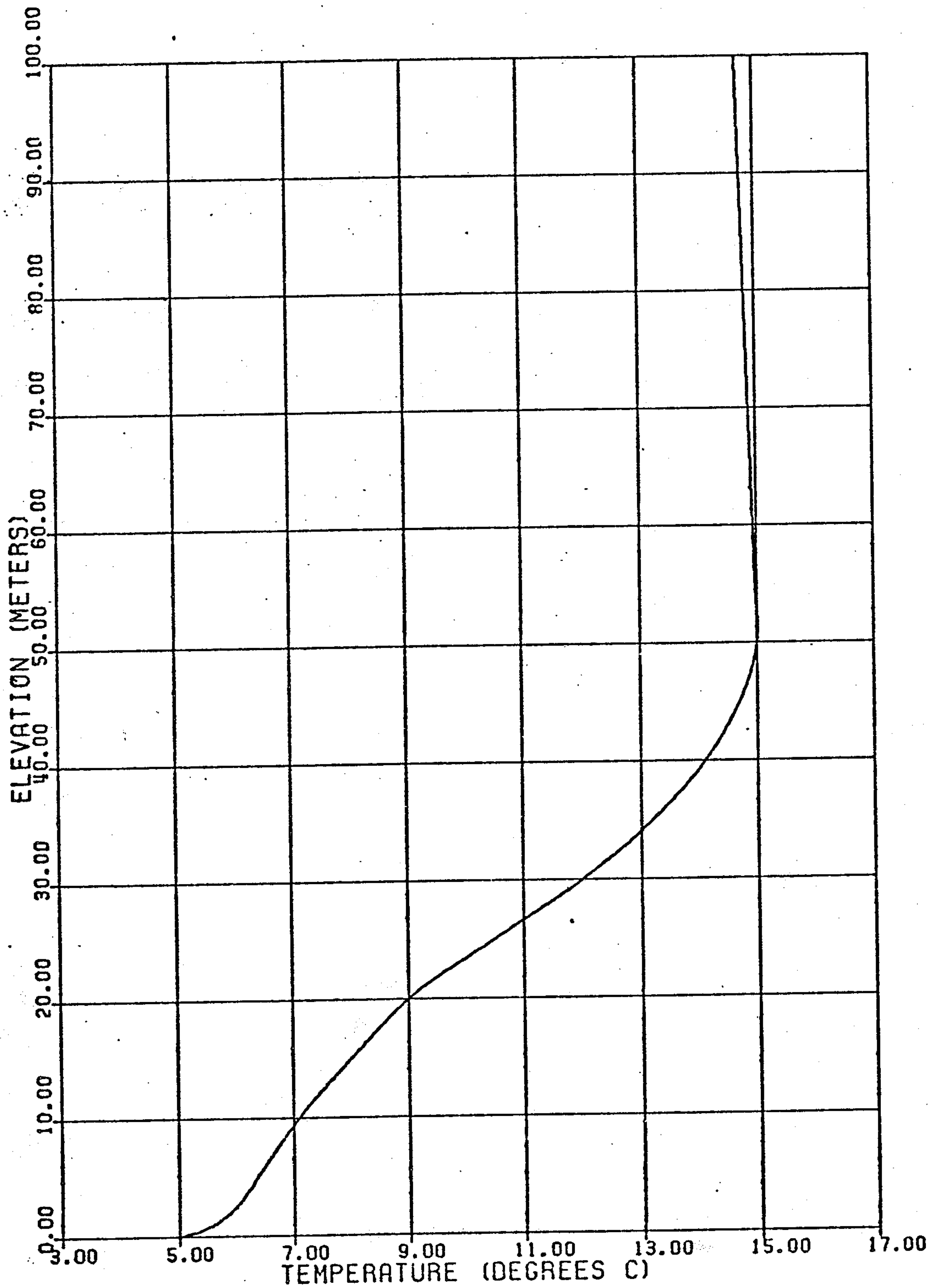


Fig. 2.3: Temperature profile I

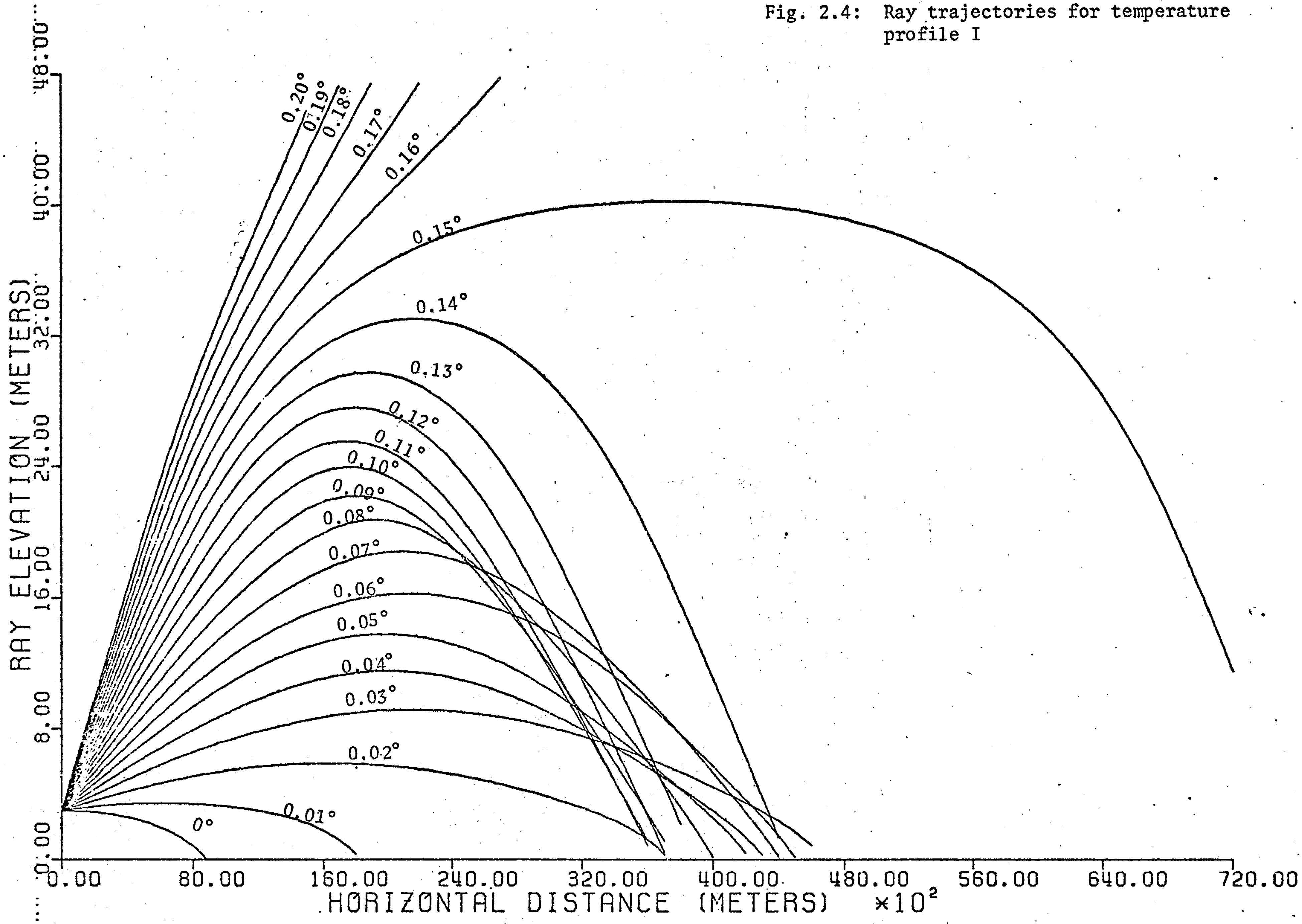


Fig. 2.4: Ray trajectories for temperature profile I

b. Temperature profile II:

This temperature profile is similar to the first one but with a weaker temperature inversion. The change in temperature is 6.78°C in the first 50 meters altitude, whereas, the first profile was 10°C, for the same depth.

The profile starts with cold surface temperature (5°C) and increases rapidly up to 11.80°C at a height of 45 meters.

With altitudes above 50 meters, the temperature decreases normally (-0.006°C/m.) up to 100 meters, the maximum height of interest in this case.

Table 2.3 and Fig. 2.5 show this profile. Fig. 2.6 shows the ray paths with different initial elevation angles at the observer; ϕ_o starts from $\phi_o = 0$ degrees.

The ray starting with $\phi_o = 0°$ intersects the surface at 8 Km. horizontal distance. This distance increases with increasing ϕ_o up to a certain limit, at which $\phi_o = 0.024°$. Above that, the horizontal distance is decreased with increasing ϕ_o (this is clear from Fig. 2.6).

This situation continues to $\phi_o = 0.065$ degrees, after which the rays start again to hit the surface with larger distance if ϕ_o is increased.

We can see easily that the ray at $\phi_o = 0.085°$ represents approximately the horizon, corresponding to a horizontal distance equal to 85 Km. Any ray with ϕ_o greater than 0.085 degrees will diverge upward, as shown in Fig. 2.6.

TEMPERATURE INPUT DATA

Z(Meters)	T Deg. C
0	5
1	5.6
2	5.9
3	6.1
4	6.24
6	6.52
10	7
15	7.5
20	8.3
25	9.3
30	10.3
35	11.1
40	11.6
45	11.8
50	11.78
55	11.75
60	11.72
70	11.66
80	11.6
100	11.48

TABLE 2.3

TEMPERATURE PROFILE II

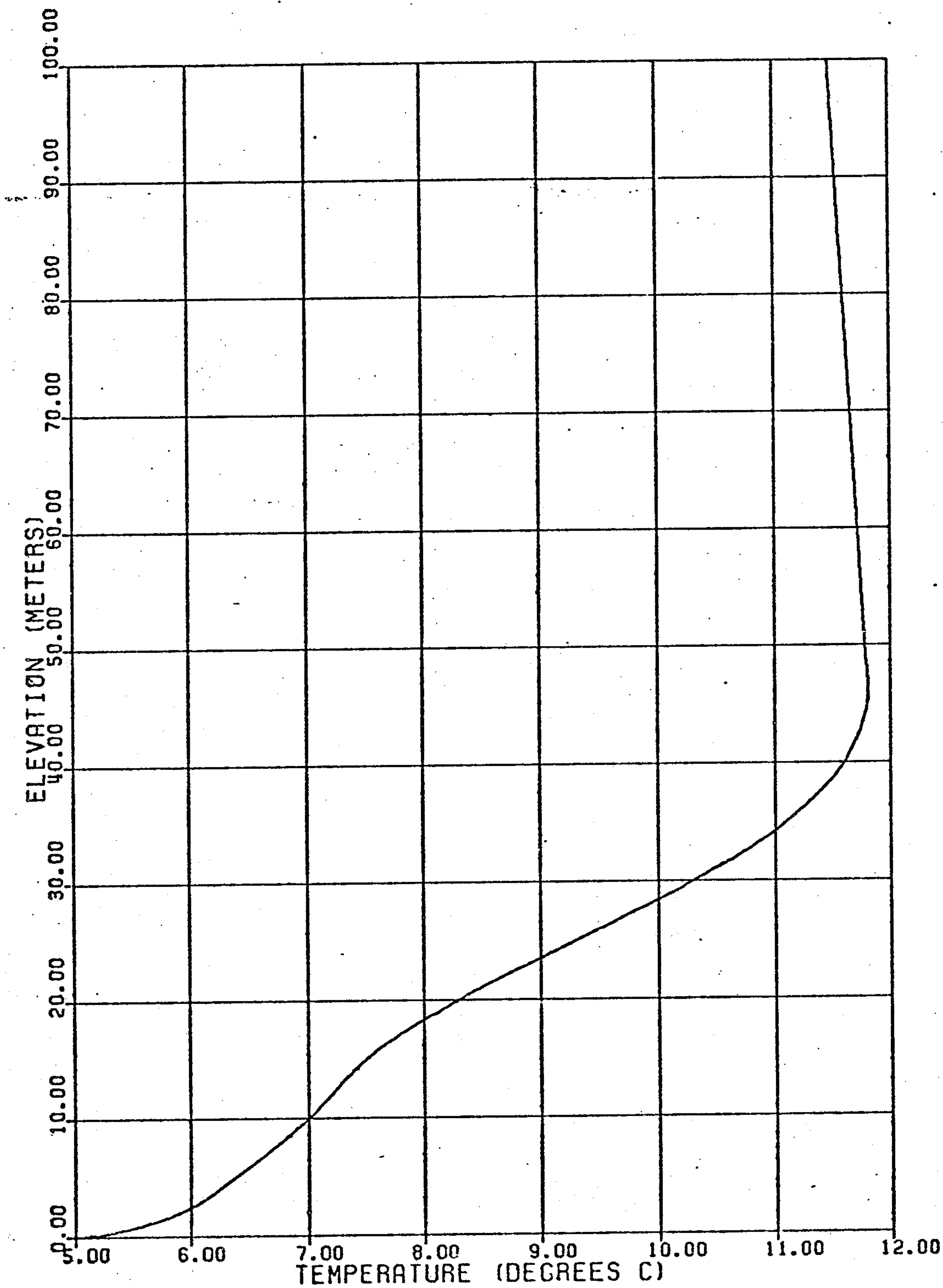
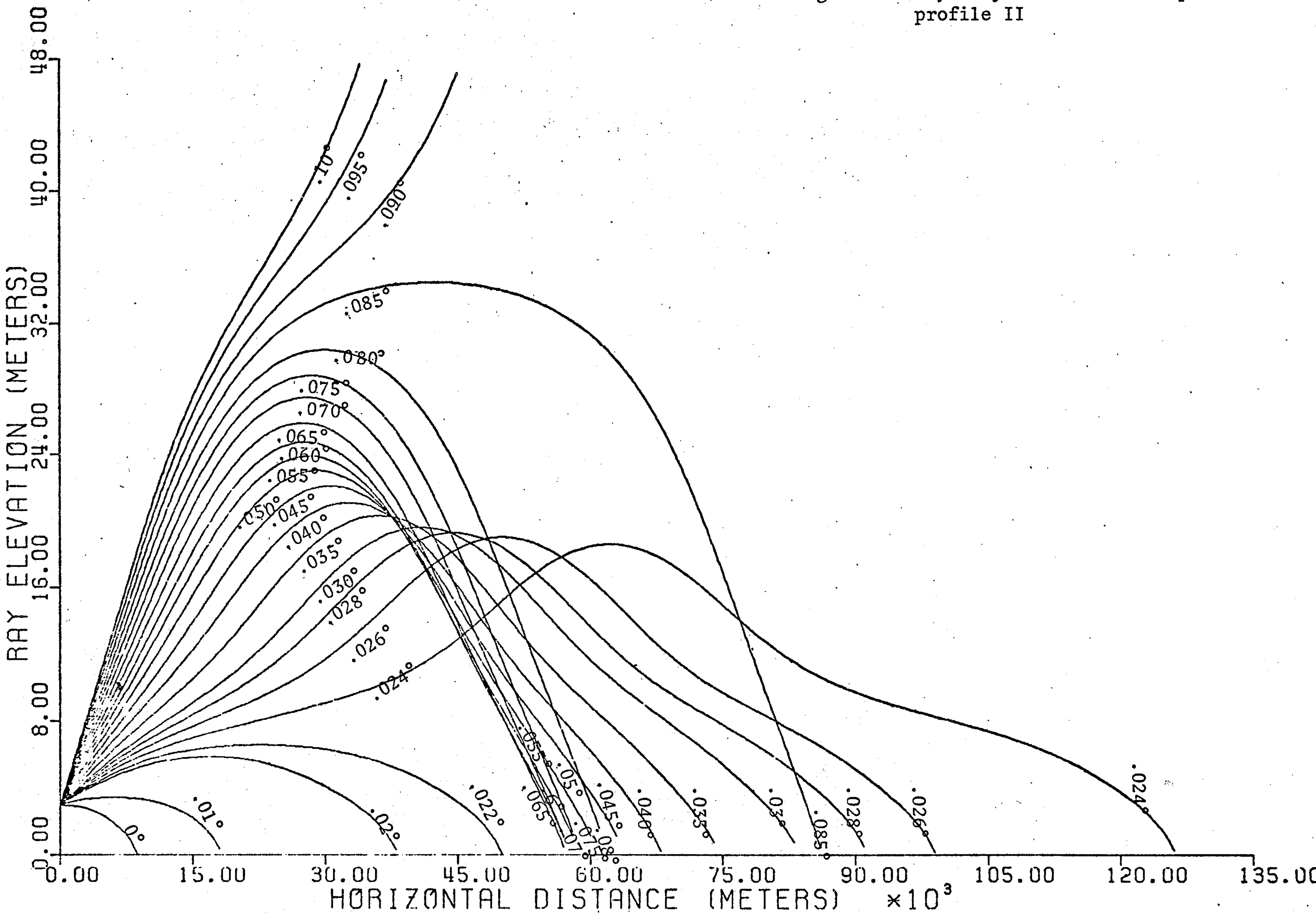


Fig. 2.5: Temperature profile II

Fig. 2.6: Ray trajectories for temperature profile II



c. Temperature profile III;

This profile is characterized by a thermocline at altitude 200 meters above the surface. The thermocline-point of refraction is the point at which the second derivative of the temperature with respect to the elevation (d^2T/dz^2) vanishes. The strength of inversion for this profile is equal to 29.6°C for 320 meters depth. Air layers which lie on the surface are very cold (-30°C at the surface), and the higher layers are warmer, with high temperature gradient. The thermocline occurs at altitude 200 meters. The air layer at elevation 320 meters is the warmest one in this profile. This is clear from Table 2.4 and in Fig. 2.7.

The temperature of the air layers above 320 meters decreases with altitude, with normal temperature gradient -0.006°C/meter up to 700 meters, which is the maximum elevation of interest in this profile.

Ray paths of this profile are calculated by using equation (2.1.14). Fig. 2.8 shows clearly these rays with different initial elevation angle ϕ_o, starting with $\phi_o = -0.03°$ from the horizontal. The incremental change in $\phi_o = 0.01°$. It is clear from Fig. 2.8 that, due to the existence of the thermocline at 200 meters, most of the rays are returned towards the surface from this level. This occurs for all rays that have ϕ_o less than or equal to 0.17 degrees. But with higher elevation angles, the rays diverge upwards. The horizon in this case is at a very long distance from the observer; at 160 Km. approximately with ϕ_o equal to 0.175°.

TEMPERATURE INPUT DATA

Z(Meters)	T Deg. C
0	-30
50	-27.7
100	-25.3
140	-22.5
160	-20.6
180	-17.7
200	-13
220	-7.6
240	-3.5
260	-1.6
280	-.8
300	-.5
320	-.4
350	-.55
400	-.85
450	-1.15
500	-1.45
600	-2.05
700	-2.65

TABLE 2.4

TEMPERATURE PROFILE III

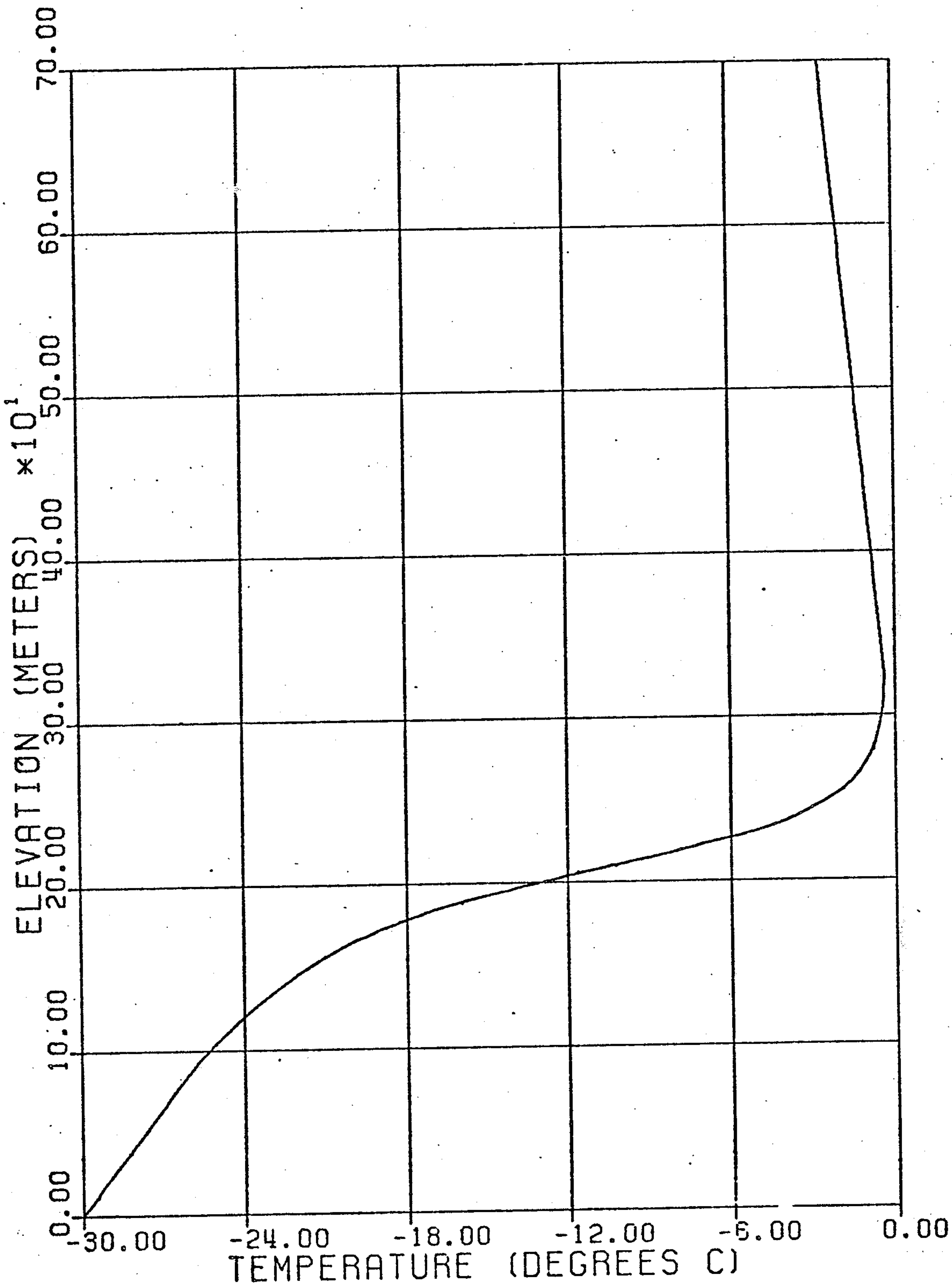


Fig. 2.7: Temperature profile III

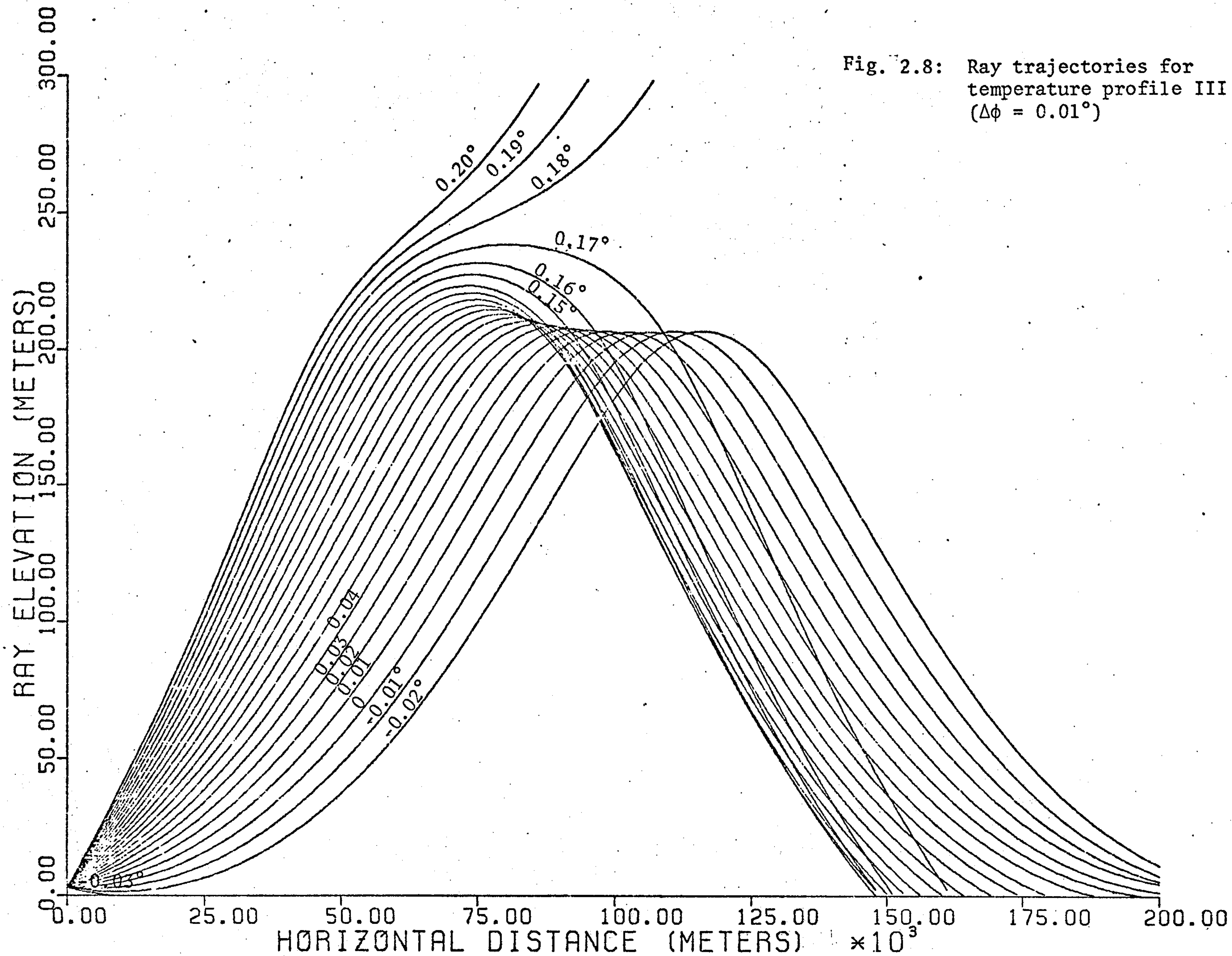


Fig. 2.8: Ray trajectories for temperature profile III ($\Delta\phi = 0.01°$)

d. Temperature profile IV:

This profile is also called the Novaya Zemlya profile, and is defined in Reference [L2]. The profile is characterized by strong inversion, its strength equal to 28.7°C over 860 meters.

The air layers close to the surface are very cold (-35°C) and start to become warmer as the altitude increases. The warmest layer is at 860 meters and the layers above this layer start to become normally colder with the rate of -0.006°C/m. as in Table 2.5, also in Fig. 2.9.

Fig. 2.10 shows the ray paths for this profile. These rays are transmitted for very long horizontal distances, especially the rays with high initial elevation angles ϕ_o. For example, the ray at ϕ_o equal to 0.13° hits the surface at 122 Km., and at ϕ_o equal to 0.14° hits the surface at 162 Km. which is the horizon in this case.

Fig. 2.10 shows ray paths for initial elevation angles in the region $0 \leq \phi_o \leq 0.20°$ with $\Delta\phi = 0.01°$. It is noticeable here, in contrast to the previous profiles, that no rays are intersecting.

TEMPERATURE INPUT DATA

Z (Meters)	T Deg. C
0	-35
35	-31
92	-25
114	-23
128	-22
149	-21
176	-20
207	-19
276	-17
409	-13.5
535	-10
575	-9
633	-8
680	-7.4
740	-6.9
800	-6.5
860	-6.3
920	-6.35
1000	-6.8
1140	-7.6
1300	-8.5

TABLE 2.5

TEMPERATURE PROFILE IV

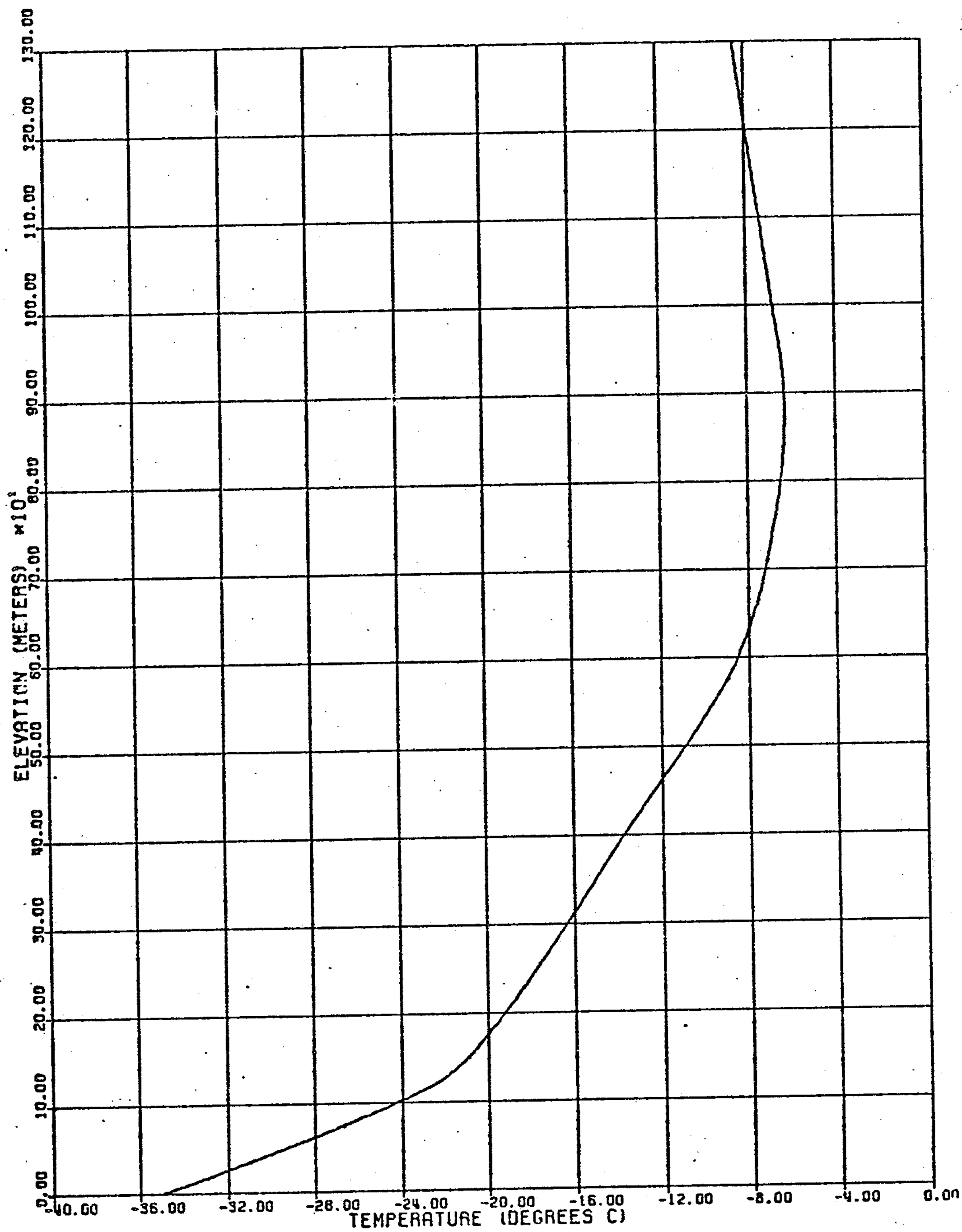


Fig. 2.9: Temperature profile IV

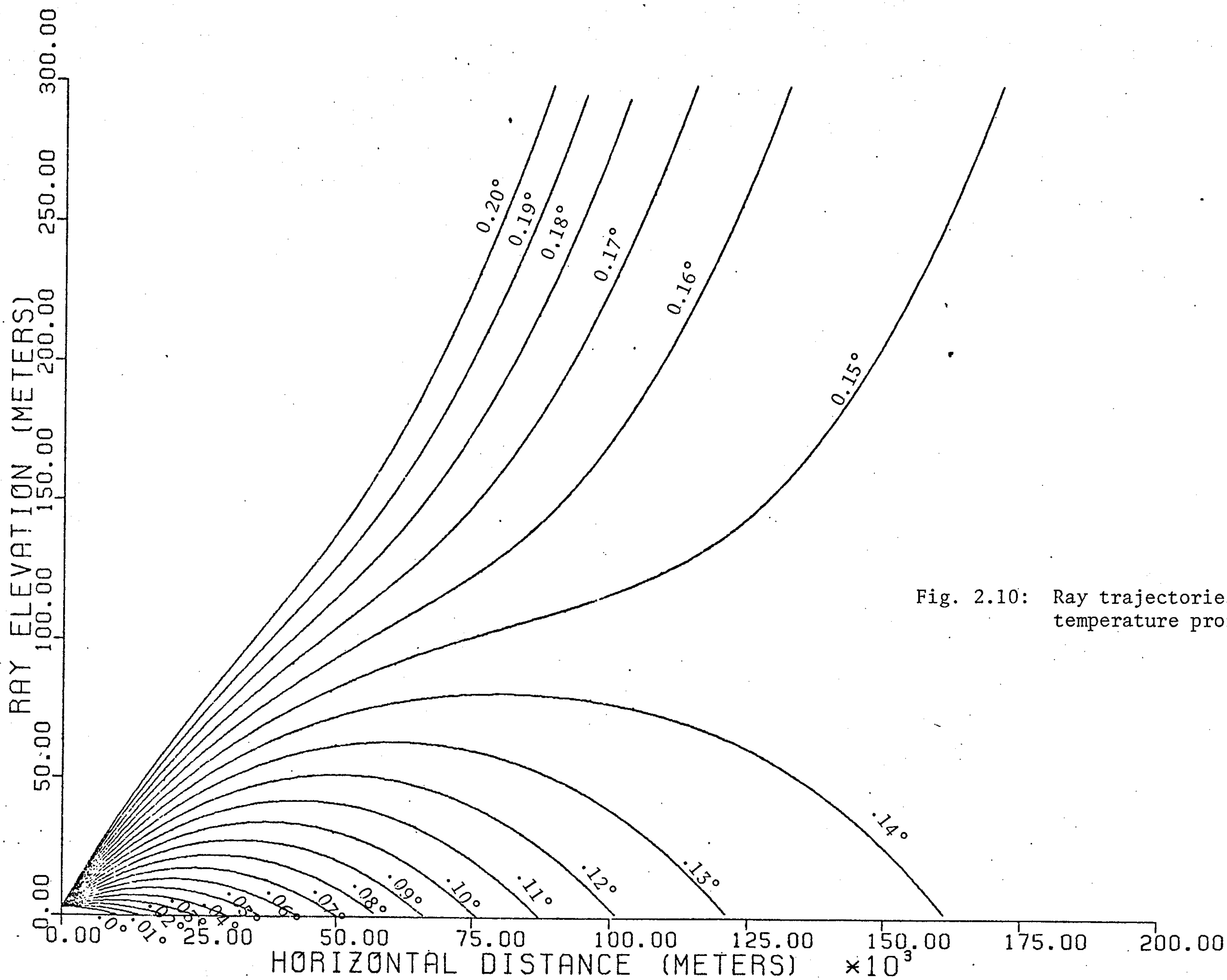


Fig. 2.10: Ray trajectories for temperature profile IV

CHAPTER 3

NUMERICAL METHODS USED IN THE THESIS

Two interpolation methods for unequally spaced data are discussed in this chapter. The first method is a cubic Lagrange interpolation and the second is a spline fit interpolation. The comparison between these two methods for handling unequally spaced temperature data is also discussed. A polynomial of at least third degree was necessary in both methods because we need the first three derivatives of temperature with respect to elevation. For this kind of polynomial, the third derivative is constant between every pair of adjacent data points.

3.1 Lagrange's Interpolation Formula

3.1.1 Theory(Refs.[H1] and [H2])

The Lagrangian interpolation polynomial $t(z)$ of degree n which takes on the same values as a given function $f(z)$ for $(n+1)$ distinct abscissas $z_o, z_1, z_2, \ldots, z_n$ (not necessarily equally spaced), can be written in the form:

$$t(z) = \ell_o(z)\, f(z_o) + \ell_1(z)\, f(z_1) + \ldots + \ell_n(z)\, f(z_n)$$

$$\equiv \sum_{k=o}^{n} \ell_k(z)\, f(z_k) \qquad (3.1.1)$$

where $\ell_o(z), \ldots\ldots, \ell_n(z)$ are polynomials of degree n or less.

We may avoid somewhat lengthy calculation by noticing that the expression (3.1.1) will take on the value $f(z_i)$ if $\ell_i(z_i) = 1$ and if $\ell_i(z_j) = 0$ when $j \neq i$.

With the convenient notation of the Kronecker delta:

$$\delta_{ij} = \begin{cases} 0 & \text{if} \quad i \neq j \\ 1 & \text{if} \quad i = j \end{cases} \qquad (3.1.2)$$

This requirement becomes:

$$\ell_i(z_j) = \delta_{ij} \qquad (i = 0, \ldots, n;\ j = o, \ldots, n) \qquad (3.1.3)$$

Since $\ell_i(z)$ is to be a polynomial of degree n which vanishes when

$$z = z_o, z_1, \ldots, z_{i-1}, z_{i+1}, \ldots, z_n,$$

therefore

$$\ell_i(z) = C_i[(z-z_o)\ldots(z-z_{i-1})(z-z_{i+1})\ldots(z-z_n)] \qquad (3.1.4)$$

where C_i is a constant.

But we have:

$$\ell_i(z_i) = 1$$

Then:

$$C_i = \frac{1}{(z_i-z_o)\ldots(z_i-z_{i-1})(z_i-z_{i+1})\ldots(z_i-z_n)} \qquad (3.1.5)$$

and the desired Lagrangian coefficient functions $\ell_i(z)$ are obtained by introducing (3.1.5) into (3.1.4). Therefore:

$$\ell_i(z) = \frac{(z-z_o)\ldots(z-z_{i-1})(z-z_{i+1})\ldots(z-z_n)}{(z_i-z_o)\ldots(z_i-z_{i-1})(z_i-z_{i+1})\ldots(z_i-z_n)} \qquad (3.1.6)$$

and the Lagrangian interpolation polynomial of degree n is given by the form:

$$t(z) = \sum_{k=o}^{n} \ell_k(z)\, t_k \qquad (3.1.7)$$

as in (3.1.1) where $\ell_i(z)$ defined by (3.1.6) and $t_k = f(z_k)$.

Therefore, this formula is the equation of an nth degree polynomial through $(n+1)$ points: $(z_o,t_o),(z_1,t_1),\ldots,(z_n,t_n)$ which are not necessarily equally spaced.

It can be seen that this formula involves large numbers of multiplications and hence becomes quite slow if n is large.

Without loss in generality, we can use cubic form of Lagrange formula to fit every adjacent four points starting from the first point (z_o,t_o) and ending with (z_n,t_n) by moving one point every time.

If we set $n = 3$ in the equations (3.1.7), and (3.1.6), and if we start from the jth point we can get:

$$t(z) = \sum_{k=j}^{j+3} \ell_k(z)\, t_k$$

$$= \ell_j(z)t_j + \ell_{j+1}(z)t_{j+1} + \ell_{j+2}(z)t_{j+2} + \ell_{j+3}(z)t_{j+3} \tag{3.1.8}$$

where:

$$j = 0,1,2,\ldots,n-3$$

also:

$$\ell_j(z) = \frac{(z-z_{j+1})(z-z_{j+2})(z-z_{j+3})}{(z_j-z_{j+1})(z_j-z_{j+2})(z_j-z_{j+3})}$$

$$j = 0,1,2,\ldots,n-3 \tag{3.1.9}$$

From (3.1.8) and (3.1.9) we can write Lagrange expression in simple form for calculations; therefore:

$$t(z) = b_o+b_1(z-z_j)+b_2(z-z_j)(z-z_{j+1})+b_3(z-z_j)(z-z_{j+1})(z-z_{j+2}) \tag{3.1.10}$$

where:

$$b_o = t_j,$$

$$b_1 = \frac{t_{j+1}-t_j}{z_{j+1}-z_j}$$

$$b_2 = \frac{(t_{j+2}-b_o)-b_1(z_{j+2}-z_j)}{(z_{j+2}-z_j)(z_{j+2}-z_{j+1})} \quad \text{and}$$

$$b_3 = \frac{(t_{j+3}-b_o)-b_1(z_{j+3}-z_j)-b_2(z_{j+3}-z_j)(z_{j+3}-z_{j+1})}{(z_{j+3}-z_j)(z_{j+3}-z_{j+1})(z_{j+3}-z_{j+2})}$$

$$j = 0,1,\ldots,n-3 \qquad (3.1.11)$$

We can see the two expressions (3.1.8) and (3.1.10) are passing through the four adjacent points $(z_j,t_j),(z_{j+1},t_{j+1}),(z_{j+2},t_{j+2})$ and (z_{j+3},t_{j+3}) provided that the conditions (3.1.9) and (3.1.11) are satisfied.

3.1.2 Flow chart for cubic Lagrange Interpolation Formula

Fig. 3.1 shows the flow chart for cubic Lagrange interpolation method. The BASIC language program for this method is in Appendix (3).

3.2 Spline Fit Interpolation

3.2.1 Theory (Ref.[H2,P1])

Instead of fitting one curve of degree n to the $(n+1)$ given data points $(z_o,t_o),(z_1,t_1),\ldots,(z_n,t_n)$ which are arranged in order of increasing values of z, one can use the following alternate approach. One may divide the total interval $[z_o,z_n]$ into n subintervals $[z_o,z_1]$, $[z_1,z_2],\ldots,[z_{n-1},z_n]$ and use the spline fit by connecting each pair of adjacent points (every subinterval) with a section of a third-degree polynomial, matching up the sections so that the first and second derivatives are continuous at each point.

Let $Y_0,Y_1,\ldots,Y_n$ be the values of the second derivative at the points. Then in the interval $[z_k,z_{k+1}]$, the second derivative has the value (using linear interpolation):

$$t'' = Y_k \frac{z_{k+1}-z}{d_k} + Y_{k+1} \frac{z-z_k}{d_k} \qquad (3.2.1)$$

where $d_k = z_{k+1}-z_k$

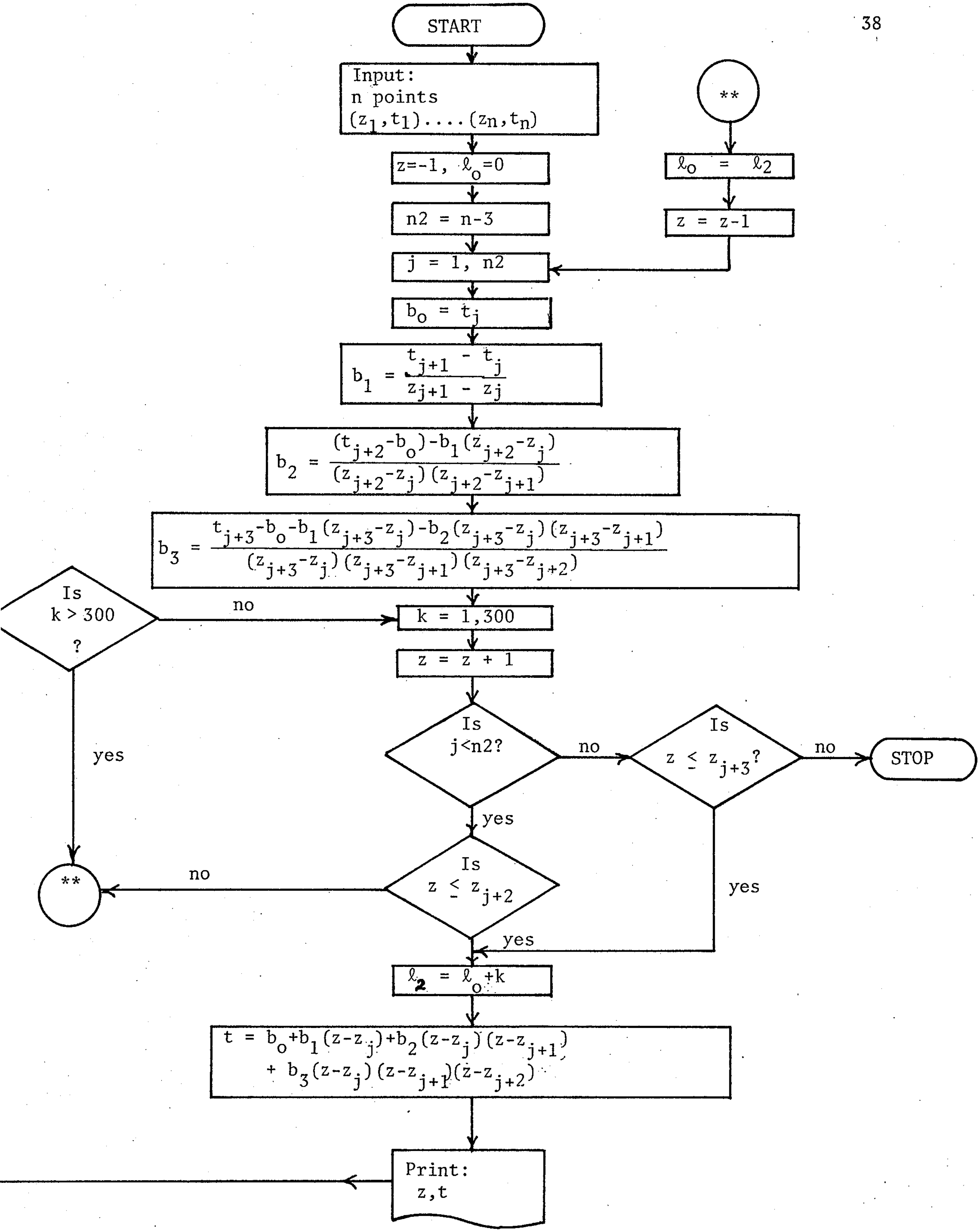


Fig. 3.1: Flow chart of cubic Lagrange interpolation formula.

Integrate (3.2.1) to get the first derivative:

$$t' = -Y_k\left[\frac{(z_{k+1}-z)^2}{2d_k}\right] + Y_{k+1}\left[\frac{(z-z_k)^2}{2d_k}\right] + C_1 \qquad (3.2.2)$$

where C_1 is constant of integration.

By integrating again, we can obtain the equation of the temperature-altitude curve:

$$t = Y_k\left[\frac{(z_{k+1}-z)^3}{6d_k}\right] + Y_{k+1}\left[\frac{(z-z_k)^3}{6d_k}\right] + C_1 z + C_2 \qquad (3.2.3)$$

where C_2 is a constant of integration.

But the curve (3.2.3) passes through the points $(z_k,t_k),(z_{k+1},t_{k+1})$, therefore:

$$C_1 = \frac{t_{k+1} - t_k}{d_k} - \frac{(Y_{k+1} - Y_k)d_k}{6} \qquad (3.2.4)$$

$$C_2 = \frac{t_k z_{k+1} - t_{k+1} z_k}{d_k} - \frac{d_k}{6}(Y_k z_{k+1} - Y_{k+1} z_k) \qquad (3.2.5)$$

substitute C_1 and C_2 into (3.2.3) then,

$$\begin{aligned} t = {} & \frac{Y_k}{6d_k}(z_{k+1}-z)^3 + \frac{Y_{k+1}}{6d_k}(z-z_k)^3 \\ & + (z_{k+1}-z)\left(\frac{t_k}{d_k} - \frac{Y_k d_k}{6}\right) \\ & + (z-z_k)\left(\frac{t_{k+1}}{d_k} - \frac{Y_{k+1} d_k}{6}\right) \end{aligned} \qquad (3.2.6)$$

In this equation all quantities are known except Y_k, Y_{k+1}, the values of the second derivative at the end points of the interval $[z_k, z_{k+1}]$.

We consider the slope at the point (z_k,t_k) as determined from equation (3.2.2) is the same as that determined by the corresponding formula for the interval (z_{k-1}, t_{k-1}) to (z_k, t_k).

Substitute the value of C_1 from equation (3.2.4) in (3.2.2), we can get:

$$t' = -Y_k[\frac{(z_{k+1} - z)^2}{2d_k}] + Y_{k+1}[\frac{(z - z_k)^2}{2d_k}]$$

$$+ \frac{(t_{k+1} - t_k)}{d_k} - (Y_{k+1} - Y_k)\frac{d_k}{6} \qquad (3.2.7)$$

and the corresponding equation for the preceding interval is:

$$t' = -(\frac{Y_{k-1}}{2d_{k-1}})(z_k-z)^2 + (\frac{Y_k}{2d_{k-1}})(z-z_{k-1})^2$$

$$+ \frac{(t_k-t_{k-1})}{d_{k-1}} - (Y_k-Y_{k-1})(\frac{d_{k-1}}{6}) \qquad (3.2.8)$$

collecting the unknowns Y_{k-1}, Y_k and Y_{k+1} on one side of the equation,

Therefore:

$$Y_{k-1}(\frac{d_{k-1}}{6}) + Y_k(\frac{d_{k-1} + d_k}{3}) + Y_{k+1}(\frac{d_k}{6})$$

$$= \frac{t_{k+1} - t_k}{dk} - \frac{t_k - t_{k-1}}{d_{k-1}} \qquad (3.2.9)$$

$$k = 1,2,\ldots,n-1$$

we have similar equation as (3.2.9) for each of the interval points, that is, $k = 1,2,\ldots,n-1$. Therefore, we have $(n-1)$ equations in the $(n+1)$ unknowns $Y_0,Y_1,Y_2,\ldots,Y_n$.

Two more conditions are introduced on Y_0 and Y_n, which require the third derivative to be continuous at (z_1,t_1) and at (z_{n-1},t_{n-1}). From equation (3.2.1):

$$t''' = -\frac{Y_k}{d_k} + \frac{Y_{k+1}}{d_k} \qquad (3.2.10)$$

Equating values for $k=0$ and $k=1$:

$$-\frac{Y_0}{d_0} + \frac{Y_1}{d_0} = -\frac{Y_1}{d_1} + \frac{Y_2}{d_1}$$

or

$$-\frac{Y_0}{d_0} + Y_1(\frac{1}{d_0} + \frac{1}{d_1}) - \frac{Y_2}{d_1} = 0 \qquad (3.2.11)$$

Similarly, equating values for $k = n-2$ and $k = n-1$

$$\therefore \quad -\frac{Y_{n-2}}{d_{n-2}} + \frac{Y_{n-1}}{d_{n-2}} = -\frac{Y_{n-1}}{d_{n-1}} + \frac{Y_n}{d_{n-1}}$$

or

$$-\frac{Y_{n-2}}{d_{n-2}} + Y_{n-1}\left(\frac{1}{d_{n-2}} + \frac{1}{d_{n-1}}\right) - \frac{Y_n}{d_{n-1}} = 0 \qquad (3.2.12)$$

By solving the $(n+1)$ equations (3.2.9), (3.2.11) and (3.2.12) which contain $(n+1)$ unknowns, we can determine Y_k; then equation (3.2.6) can be used directly for finding t for any value of z between z_0, z_n.

We can write equation (3.2.6) in the form:

$$t = C_{1,k}(z_{k+1}-z)^3 + C_{2,k}(z-z_k)^3 + C_{3,k}(z_{k+1}-z) + C_{4,k}(z-z_k) \qquad (3.2.13)$$

$$k = 0,1,\ldots,n$$

where:

$$C_{1,k} = \frac{Y_k}{6d_k} \qquad (3.2.14)$$

$$C_{2,k} = \frac{Y_{k+1}}{6d_k} \qquad (3.2.15)$$

$$C_{3,k} = \frac{t_k}{d_k} - \frac{Y_k\, d_k}{6} \qquad (3.2.16)$$

$$C_{4,k} = \frac{t_{k+1}}{d_k} - \frac{Y_{k+1}\, d_k}{6} \qquad (3.2.17)$$

3.2.2 Determination of constants

Let

$$p_k = \frac{d_k}{6}, \quad k = 0,1,\ldots,n-1$$

$$e_k = \frac{t_k - t_{k-1}}{d_{k-1}}, \quad k = 1,2,\ldots,n$$

$$b_0 = 0$$

$$b_k = e_{k+1} - e_k, \quad k = 1,2,\ldots,n-1$$

$$b_n = 0$$

Then we can write the system of equations (3.2.9),(3.2.11) and (3.2.12) in matrix form as

$$AY = b \tag{3.2.18}$$

where:

$$A = \begin{bmatrix} -\frac{1}{d_0} & \frac{1}{d_0}+\frac{1}{d_1} & -\frac{1}{d_1} & 0 & \text{--------} & 0 & & 0 \\ p_0 & 2(p_0+p_1) & p_1 & 0 & \text{--------} & 0 & & 0 \\ 0 & p_1 & 2(p_1+p_2) & p_2 & \text{--------} & 0 & & 0 \\ - & - & - & & & & & \\ - & - & - & & & & & \\ - & - & - & & & & & \\ - & - & - & & & & & \\ 0 & 0 & 0 & \text{--------} & & p_{n-2} & 2(p_{n-2}+p_{n-1}) & p_{n-1} \\ 0 & 0 & 0 & \text{--------} & & -\frac{1}{d_{n-2}} & \frac{1}{d_{n-2}}+\frac{1}{d_{n-1}} & -\frac{1}{d_{n-1}} \end{bmatrix}$$

Y_i can be found and then C_{ij} can be determined by using equations (3.2.14,15,16 and 17). To solve equation (3.2.18), we can use the elimination methodgiven in Ref. [P1], sect. 10.2.

The flow chart in Fig. 3.2 shows the determination of coefficients C_{ij}.

3.2.3 Computer use of spline fit

Fig. 3.3 shows the flow chart of using computer in spline fit interpolation. The corresponding BASIC program (EA100) is in Appendix (4).

The main idea of this program is to locate the position of the given value of z between the two points (z_k, t_k) and (z_{k+1}, t_{k+1})

START

Input: $z_k, t_k, \quad k = 0,1,\ldots,n$

$$
\begin{aligned}
d_k &= z_{k+1} - z_k \\
p_k &= d_k/6 \\
e_k &= (t_{k+1} - t_k)/d_k \\
k &= 0,1,\ldots\ldots,n-1
\end{aligned}
$$

$b_k = e_k - e_{k-1}, \quad k=1,\ldots n-1$

$$
\begin{aligned}
a_{o,1} &= -1 - d_o/d_1 \\
a_{0,2} &= d_o/d_1 \\
a_{1,2} &= p_1 - p_o \, a_{o,2} \\
a_{1,1} &= 2(p_o + p_1) - p_o \, a_{0,1} \\
a_{1,2} &= a_{1,2}/a_{1,1} \\
b_1 &= b_1/a_{1,1}
\end{aligned}
$$

$$
\begin{aligned}
a_{k,k} &= 2(p_{k-1} + p_k) \\
&\quad - p_{k-1} \, a_{k-1,k} \\
b_k &= b_k - p_{k-1} \, b_{k-1} \\
a_{k,k+1} &= p_k/a_{k,k} \\
b_k &= b_k/a_{k,k} \\
k &= 2,\ldots\ldots, n-1
\end{aligned}
$$

$$
\begin{aligned}
a_{n,n-1} &= 1 + d_{n-2}/d_{n-1} + a_{n-2,n-1} \\
a_{n,n} &= -d_{n-2}/d_{n-1} - a_{n,n-1} \, a_{n-1,n} \\
b_n &= b_{n-2} - a_{n,n-1} \, b_{n-1}
\end{aligned}
$$

$$
\begin{aligned}
Y_n &= b_n/a_{n,n} \\
Y_k &= b_k - a_{k,k+1} \, Y_{k+1}, \quad k = n-1,\ldots,2,1 \\
Y_o &= -a_{o,1} \, Y_1 - a_{o,2} \, Y_2
\end{aligned}
$$

*

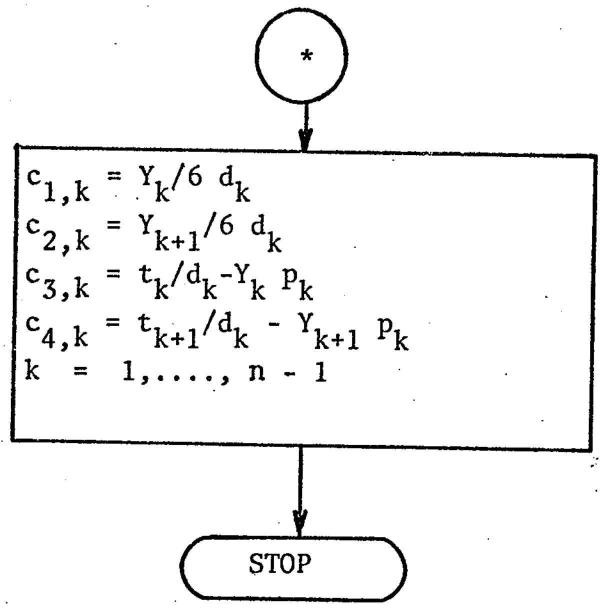


Fig. 3.2: Flow chart for determination of coefficients for a spline fit.

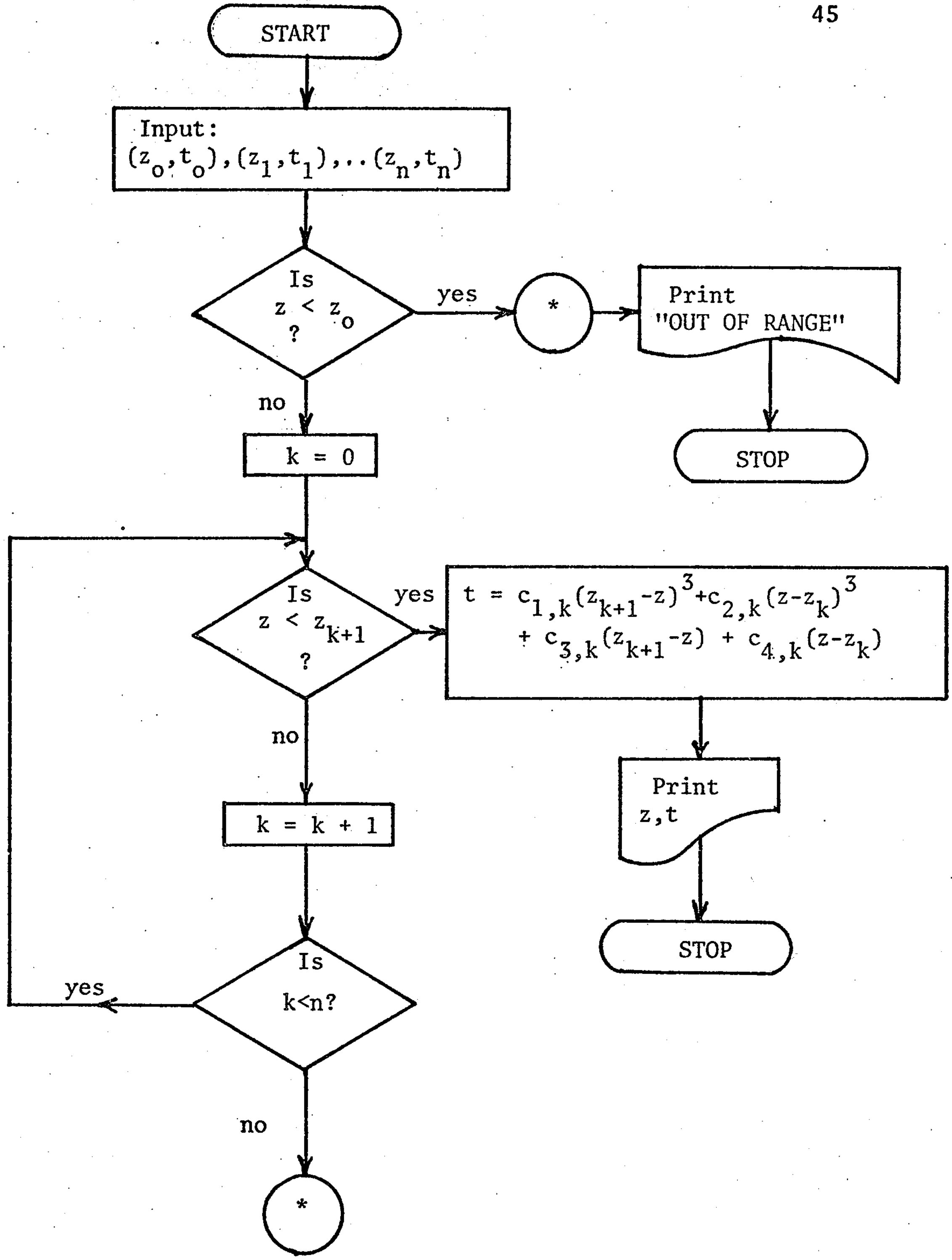


Fig. 3.3: Flow chart of spline fit interpolation

from the input data table: (z_o,t_o), (z_1,t_1), ..., (z_n,t_n) and then determine the value of t corresponding to z by using the equation

$$t = C_{1,k}(z_{k+1}-z)^3 + C_{2,k}(z-z_k)^3 + C_{3,k}(z_{k+1}-z) + C_{4,k}(z-z_k)$$

where the constants $C_{1,k}$, $C_{2,k}$, $C_{3,k}$ and $C_{4,k}$ have been previously computed and stored.

3.3 Comparison Between the Application of Cubic Lagrange and Spline Fit Methods in Temperature Data Profiles

We will take the temperature data profile in Table 3.1 as an example for comparison between the application of cubic Lagrange and spline fit methods for handling this data (unequally spaced). Tables 3.2. and 3.3 show the interpolated values of temperature, density of the atmosphere, velocity of propagation and its first three derivatives at every four meters of altitude, starting from surface level. Table 3.2 uses the cubic Lagrange interpolation formula and Table 3.3 uses the spline fit method.

By comparison of the two tables, we can find by inspection that the temperature, density, the velocity and its first derivative have values that are quite similar, but the second and third derivatives of the velocity (v'', v''') are different. To see the effect of this change in values of v'' and v''' for calculations of w in $(u-w)$ coordinate system as in equation (2.1.14), the sensitivity of w with respect to the 1% change in v''' and then in v'' was examined as follows:

$$S \triangleq \left|\frac{w_2-w_1}{w_1}\right| \times 100\%$$

where w_1 is the value of w in equation (2.1.14) at typical values of v, v', v'', v''', u and ϕ_o

w_2 is the value of w at the same values of v, v', u and ϕ_o but with 1% change in v'' <u>or</u> v'''.

By applying this to the results obtained from cubic Lagrange interpolation and spline fit methods, Table 3.4 is obtained. We can see the effect of 1% change in v''' gives an average .002% change in w. Also the 1% change in v'' gives .03% change in w in average in both methods.

From this result, we can say that in spite of the differences in the v'' and v''' columns in Tables 3.2 and 3.3, and due to the very small effect of the change in v'' and v''' in calculation of parameter w, we can consider either method (cubic Lagrange interpolation or spline fit) equally valid for handling unequally spaced data. The cubic Lagrange method is used in our case.

TEMPERATURE INPUT DATA

Z(meters)	T (Degrees C)
0	5.00
1	5.60
2	5.90
3	6.10
4	6.24
6	6.52
10	7.00
15	7.50
20	8.30
25	9.30
30	10.30
35	11.10
40	11.60
45	11.80
50	11.78
55	11.75
60	11.72
70	11.66
80	11.60
100	11.48

TABLE 3.1

INTERPOLATION OF TEMPERATURE DATA

USING CUBIC LAGRANGE FORMULA

Z(M)	T C	DENS..	V	V'	V''	V'''
0	5	1.27158	2.99914E+08	263.882	-156.643	64.3923
4	6.24	1.26531	2.99914E+08	53.8218	.231714	-.878581
8	6.78533	1.26223	2.99914E+08	46.4585	-3.9116	1.08795
12	7.1576	1.25994	2.99915E+08	39.0198	4.35985	-.253383
16	7.6344	1.25718	2.99915E+08	53.4877	4.31311	-.499059
20	8.3	1.2536	2.99915E+08	66.7452	2.32047	-.492428
24	9.0936	1.24946	2.99915E+08	72.1037	.366493	-.482207
28	9.9184	1.24521	2.99916E+08	69.2221	-2.01495	-.230897
32	10.656	1.24138	2.99916E+08	59.4135	-3.63731	.0122816
36	11.2278	1.23828	2.99916E+08	45.6925	-4.34499	.200293
40	11.6	1.23607	2.99916E+08	29.9091	-3.55136	.194356
44	11.7843	1.23467	2.99916E+08	16.1308	-3.08481	.496424
48	11.7898	1.23406	2.99916E+08	8.7737	-.165308	.0235682
52	11.768	1.23356	2.99916E+08	8.29307	-6.67198E-04	2.84519E-06
56	11.744	1.23307	2.99916E+08	8.2904	-6.32974E-04	-1.20052E-06
60	11.72	1.23258	2.99916E+08	8.28786	-6.37770E-04	-1.19829E-06
64	11.696	1.23209	2.99916E+08	8.2853	-6.51625E-04	3.18418E-07
68	11.672	1.2316	2.99916E+08	8.2827	-6.50353E-04	3.17904E-07
72	11.648	1.23111	2.99917E+08	8.2801	-6.49082E-04	3.17390E-07
76	11.624	1.23062	2.99917E+08	8.2775	-6.47814E-04	3.16877E-07
80	11.6	1.23013	2.99917E+08	8.27491	-6.46548E-04	3.16364E-07
84	11.576	1.22965	2.99917E+08	8.27233	-6.45283E-04	3.15852E-07
88	11.552	1.22916	2.99917E+08	8.26975	-6.44020E-04	3.15339E-07
92	11.528	1.22867	2.99917E+08	8.26718	-6.42760E-04	3.14828E-07
96	11.504	1.22818	2.99917E+08	8.26461	-6.41501E-04	3.14316E-07
100	11.48	1.2277	2.99917E+08	8.26204	-6.40245E-04	3.13805E-07

TABLE 3.2

INTERPOLATION OF TEMPERATURE DATA

USING SPLINE FIT METHOD

Z(M)	T C	DENS.	V	V'	V''	V'''
0	5	1.27158	2.99914E+08	10.6175	-1.30760E-03	1.61036E-07
4	6.24	1.26531	2.99914E+08	51.1089	5.03843	29.7858
8	6.77925	1.26226	2.99914E+08	47.1894	-2.98969	-7.56050E-03
12	7.18949	1.25979	2.99915E+08	38.659	.461883	1.72707
16	7.63186	1.25719	2.99915E+08	53.0478	4.90486	-.706108
20	8.3	1.2536	2.99915E+08	67.0222	2.08992	-.695898
24	9.0937	1.24946	2.99915E+08	72.033	.422046	-.411529
28	9.91369	1.24523	2.99916E+08	69.7777	-1.65406	-.551087
32	10.6521	1.24139	2.99916E+08	59.4488	-3.17669	-.215671
36	11.225	1.23829	2.99916E+08	45.1648	-3.76969	.0451124
40	11.6	1.23607	2.99916E+08	30.4474	-3.59192	.0414452
44	11.7826	1.23468	2.99916E+08	16.7894	-3.23969	.0859049
48	11.7994	1.23401	2.99916E+08	7.56131	-.868035	.761107
52	11.7653	1.23357	2.99916E+08	8.3238	.323191	-.165151
56	11.7446	1.23306	2.99916E+08	8.39898	-.131428	.0404806
60	11.72	1.23258	2.99916E+08	8.19689	.03034	.0404067
64	11.6955	1.23209	2.99916E+08	8.28754	.0150091	-3.82984E-03
68	11.6718	1.2316	2.99916E+08	8.31693	-2.95744E-04	-3.82289E-03
72	11.6481	1.23111	2.99917E+08	8.29393	-6.84070E-03	5.48629E-04
76	11.6242	1.23062	2.99917E+08	8.27096	-4.64862E-03	5.47629E-04
80	11.6	1.23013	2.99917E+08	8.25675	-2.46034E-03	5.46637E-04
84	11.5757	1.22965	2.99917E+08	8.25127	-2.75823E-04	5.45650E-04
88	11.5515	1.22916	2.99917E+08	8.25452	1.90493E-03	5.44662E-04
92	11.5273	1.22867	2.99917E+08	8.2665	4.08190E-03	5.43670E-04
96	11.5035	1.22819	2.99917E+08	8.28718	6.25507E-03	5.42671E-04
100	11.48	1.2277	2.99917E+08	8.31653	8.42440E-03	5.41661E-04

TABLE 3.3

SENSITIVITY OF w WITH RESPECT TO 1% CHANGE IN v'''

			LAGRANGE				SPLINE FIT			
U Km	R m	ϕ_o°	v'	v''	v'''	S%	v'	v''	v'''	S%
1.25	50	.09	8.29307	-.0006672	2.84419×10^{-6}	0	8.3238	.323191	-.165151	-.006634
1.25	8	.09	46.4585	-3.91916	1.08795	.0079	47.1894	-2.98969	-.0075605	-5.693×10^{-5}
1.25	10	.09	39.0198	4.35985	-.253383	-.00197	38.659	.461883	1.72707	.014224
1.25	17.25	.09	53.4877	4.31311	-.499059	-.00282	53.0478	4.90486	-.706108	-.004
1.25	12	.09	39.0198	4.35985	-.253383	-.00197	38.659	.461883	1.72707	.014224
1.25	25.11	.09	72.1037	.366493	-.482207	-.002	72.033	.422046	-.411529	-3.56×10^{-4}
1.25	28.84	.09	69.2221	-2.01495	-.230897	-.001	69.7777	-1.65406	-.551087	-.00249
1.25	32.4653	.09	59.4135	-3.63731	.0122816	7×10^{-5}	59.4488	-3.17669	-.215671	-.001178
1.25	36.15	.09	45.6925	-4.34499	.200293	.0015	45.1648	-3.76969	.0451124	.000335
1.25	40.127	.09	29.9091	-3.55136	.194356	.0023	30.4474	-3.59192	.0414452	.000478

TABLE 3.4

CHAPTER 4

TRANSFER CHARACTERISTIC PROGRAM

The purpose of the transfer characteristic BASIC program is to get mapping functions between the ray elevation in meters and initial elevation angle ϕ_o in degrees. Each mapping function is obtained at a certain horizontal distance between the object and the observer in kilometers, for a certain temperature profile. For every temperature profile given in Chapter (2), we will get the mapping functions at different horizontal distances between the object and the observer.

4.1 Flow Chart for Transfer Characteristic BASIC Program

Simplified and detailed flow charts for transfer characteristic BASIC program are shown in Figs. 4.1 and 4.2, respectively.

The purpose of this program is to find the set of points at which the ray trajectories for a given temperature profile intersect the vertical line at a fixed horizontal distance from the observer. From these intersection points, we can get the transfer characteristic for the corresponding horizontal distance (H Km.) provided that the input temperature and elevation data are available. The transfer characteristic defines the relation between elevation angle in degrees and ray elevation in meters at constant H.

Fig. 4.2 shows the detailed flow chart for the transfer characteristic program. The calculation for the ray path in u-w coordinates using equation (2.1.14), is one of the main purposes of this program. A new u-w coordinate system is chosen if:

a. The u^6 term in equation (2.1.14) exceeds 0.00001 meter.

b. Or; there is a large change in velocity gradient.

c. Or; if the horizontal travelling distance of the ray from the observer (X Km) is equal to five times the horizontal step size (Δu Km) in the axis u.

This shift of origin is done carefully as the new z-axis is not exactly parallel to the original one due to earth's curvature. A corresponding correction must be applied to the new value ϕ_o. Ray elevation above the earth's surface is found by returning to xz coordinates and using the relation $z = -\frac{x^2}{2R}$ to represent the earth's surface. The whole procedure is repeated until the ray intersects the vertical line at the object, intersects the earth's surface, or goes beyond the area of interest.

As we change the input horizontal distance H, a family of transfer characteristic curves is obtained for every temperature profile given in Chapter 2.

The selection of horizontal step size Δu(Km) is important for ray path calculations. Δu is chosen to be 0.25 Km for temperature profiles I and II, and it gives more accurate results than if $\Delta u = 1$ Km or 0.5 Km. As Δu becomes smaller, the calculations become more accurate but it takes more time, which is another important factor. For temperature profiles III and IV, Δu is chosen as 1 Km because it gives results very close to those with $\Delta u = 0.25$ Km. These two profiles use very long horizontal distances, up to 200 Km. For this reason, we need a large horizontal step size to get the results in reasonable time, but not too large to maintain the accuracy of the calculations. Section 4.2 will describe in some detail the transfer characteristic curves for every temperature profile.

Fig. 4,3 shows a simplified flow chart for the subroutine (5000) which is used for the main transfer characteristic program. The purpose of this subroutine is to fit a cubic polynomial to the unequally spaced input data (temperature and elevation) and interpolate the values of temperature every one meter of the elevation. The cubic Lagrange interpolation method, which is explained in detail in Chapter 3, is used in this case. The first three derivatives of the temperature with respect to the elevation ($\frac{dT}{dz}$, $\frac{d^2T}{dz^2}$ and $\frac{d^3T}{dz^3}$) are also calculated at one-meter intervals of z. These results are saved in virtual file VF1(10000), which reserves 2000 values for every variable. This means, the program has a limit for calculations for any temperature profile up to 2000 meters of elevation. The use of the virtual file in this program was made necessary because the memory size of the minicomputer PDP11/40 is only 24 K. words.

Subroutine (5000) also computes the density at every four meters by using equation 2.2.9.

Subroutine (6000) uses the stored values of elevation, temperature and the first three derivatives of the temperature (which were calculated by subroutine (5000) and saved in the virtual file VF1), to calculate the first three derivates of the density. Equations (2.2.11, 12 and 13) are used at the required ray elevation. From the density derivatives, the subroutine calculates the first three derivatives of the velocity using equation (2.2.3).

This subroutine uses a quadratic Lagrange interpolation as in the flow chart of Fig. 4.5 to calculate T, dT/dz and ρ. It uses linear interpolation to calculate d^2T/dz^2. Fig. 4.4 shows the detailed flow chart for subroutine (6000).

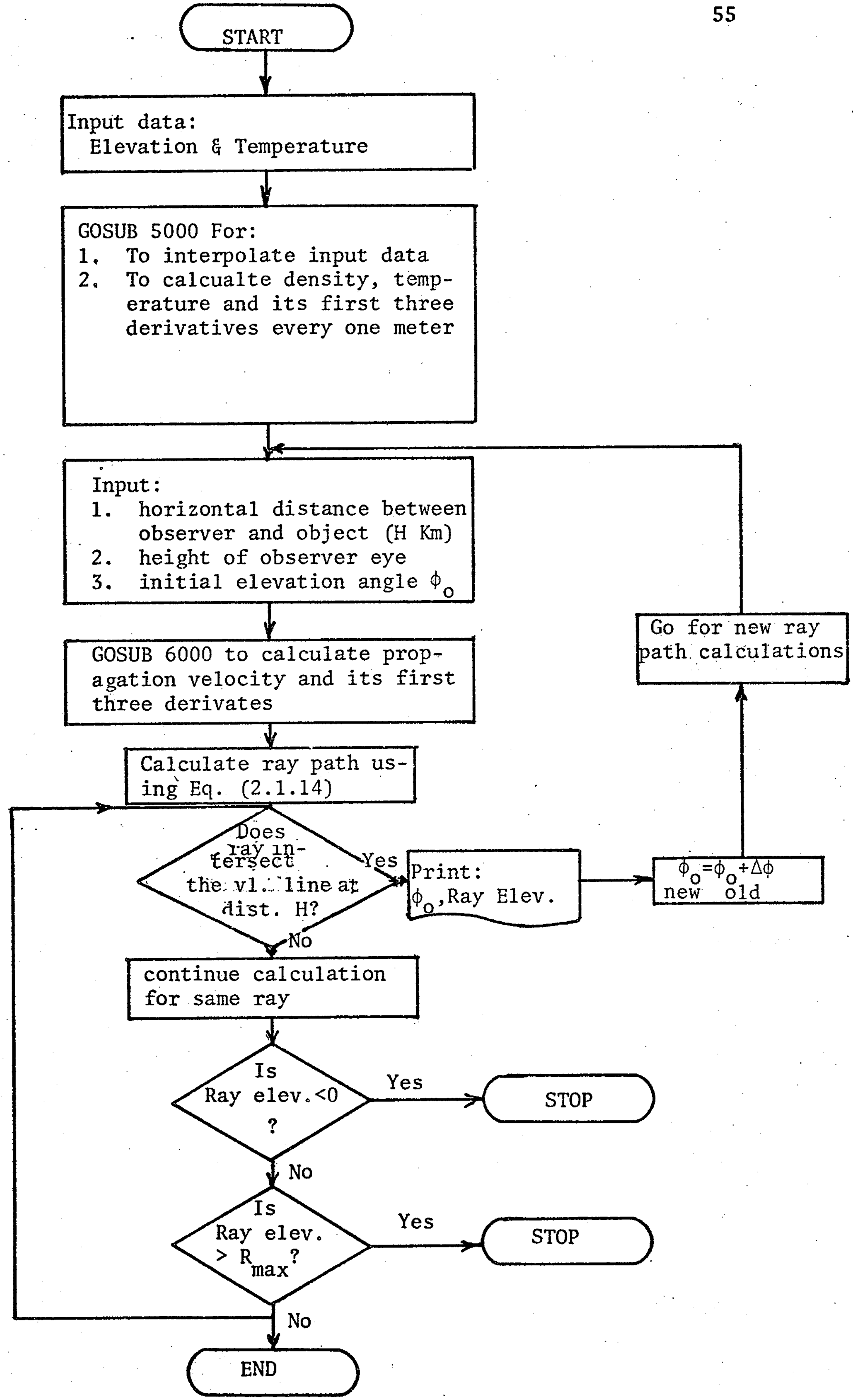


Fig. 4.1: Simplified flow chart for transfer characteristic program

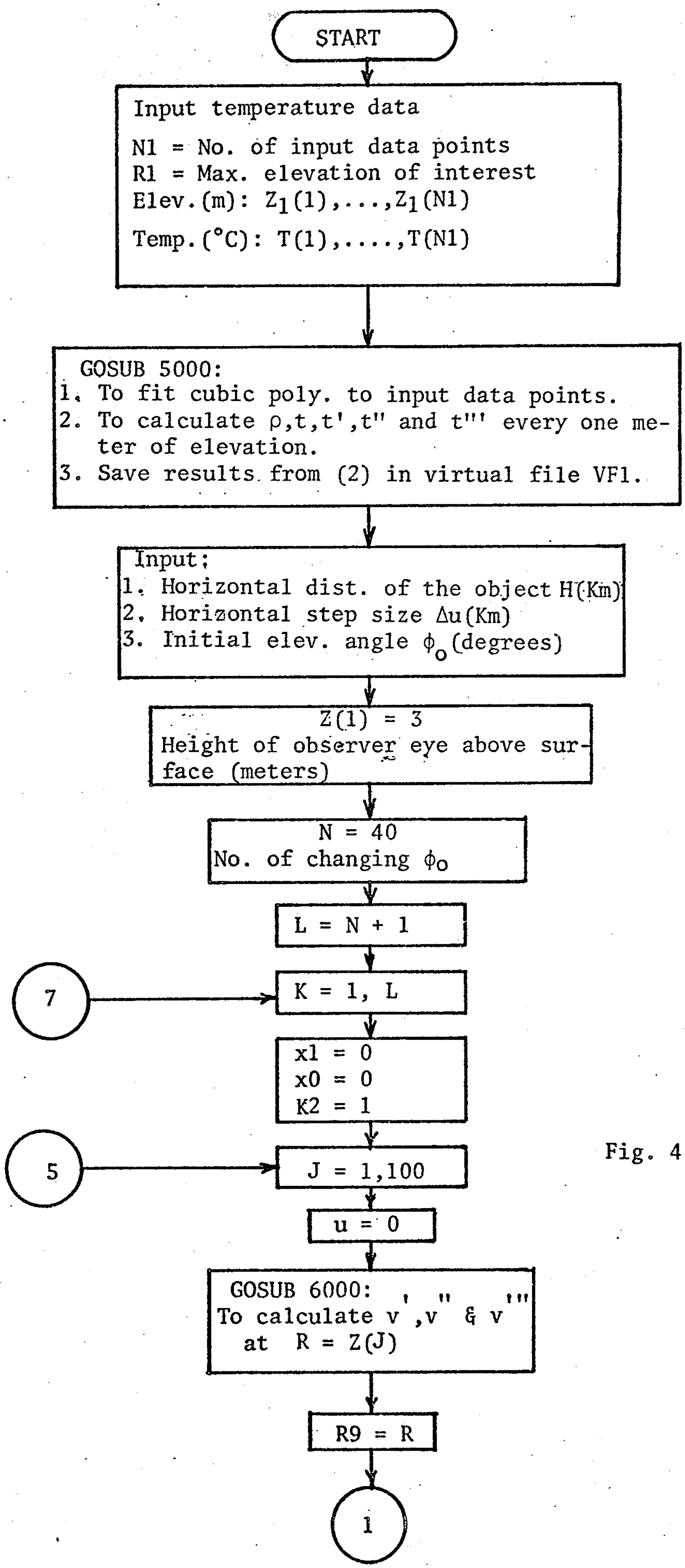
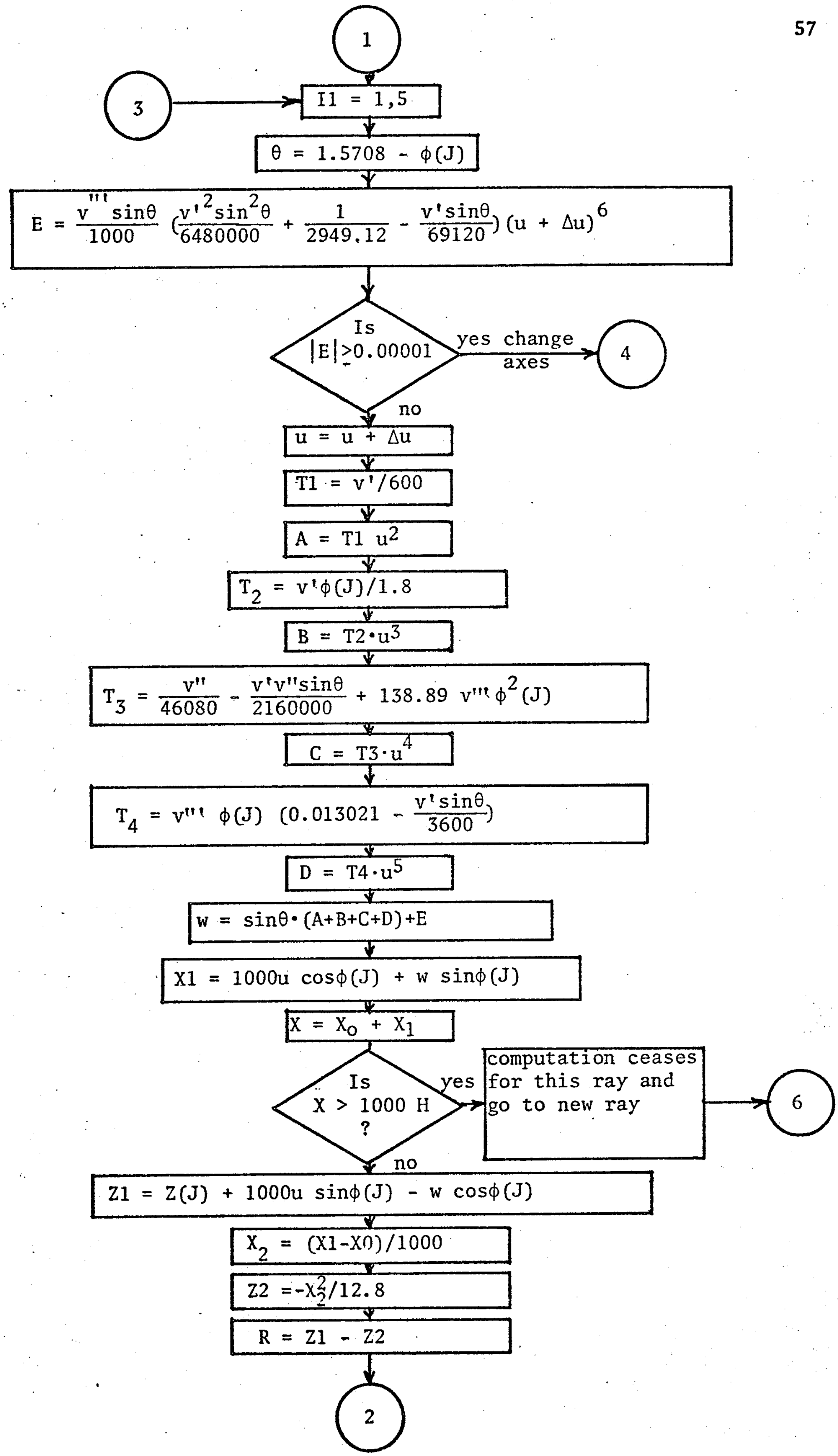
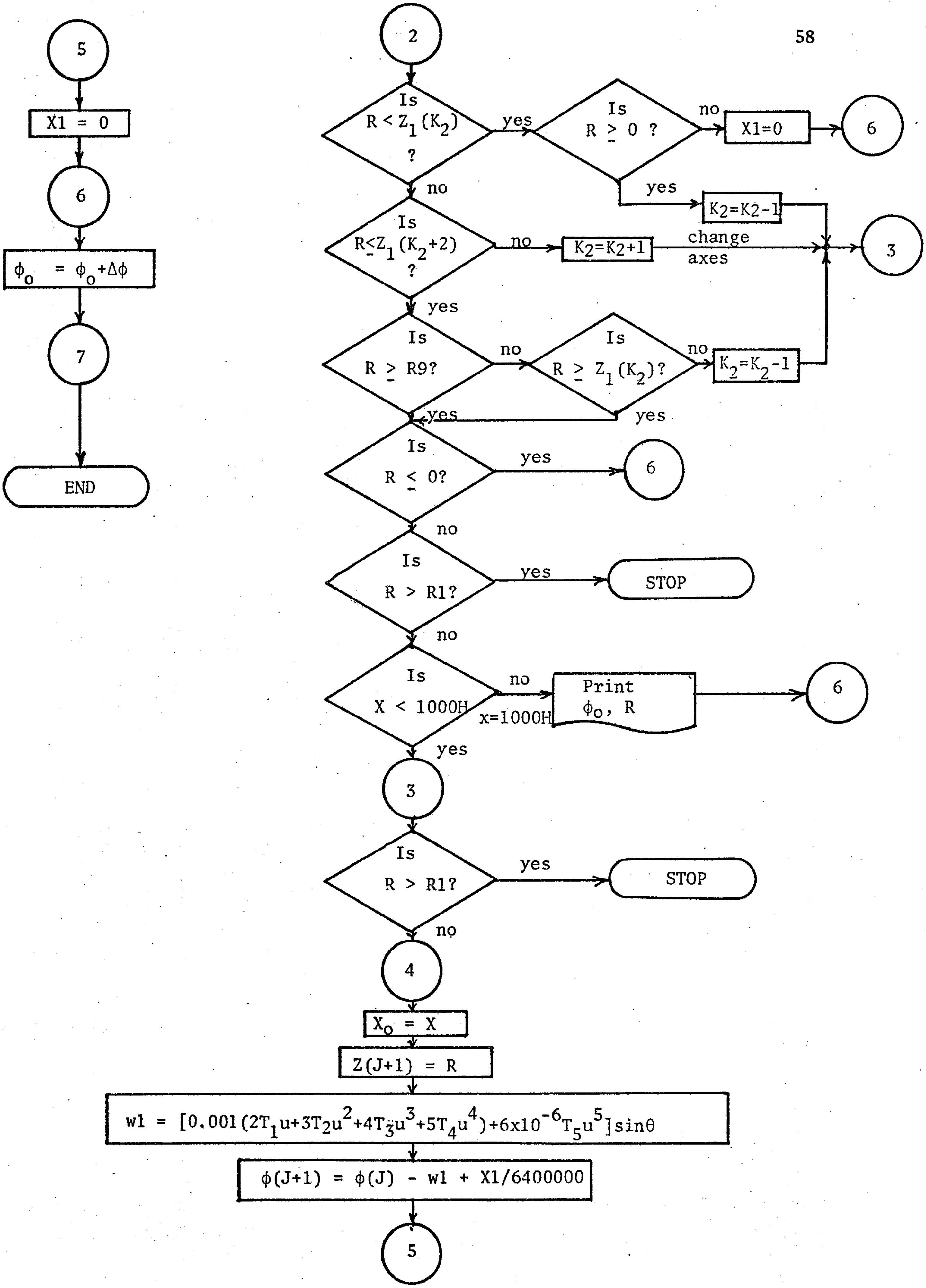


Fig. 4.2: Detailed flow chart for transfer characteristic program


1
3
$I1 = 1,5$
$\theta = 1.5708 - \phi(J)$
$E = \frac{v'''\sin\theta}{1000}\left(\frac{v'^2\sin^2\theta}{6480000} + \frac{1}{2949.12} - \frac{v'\sin\theta}{69120}\right)(u + \Delta u)^6$
Is $|E| \geq 0.00001$
yes change axes
4
no
$u = u + \Delta u$
$T1 = v'/600$
$A = T1\ u^2$
$T_2 = v'\phi(J)/1.8$
$B = T2 \cdot u^3$
$T_3 = \frac{v''}{46080} - \frac{v'v''\sin\theta}{2160000} + 138.89\ v'''\phi^2(J)$
$C = T3 \cdot u^4$
$T_4 = v'''\ \phi(J)\ (0.013021 - \frac{v'\sin\theta}{3600})$
$D = T4 \cdot u^5$
$w = \sin\theta \cdot (A+B+C+D)+E$
$X1 = 1000u\cos\phi(J) + w\sin\phi(J)$
$X = X_o + X_1$
Is $X > 1000\ H$?
yes
computation ceases for this ray and go to new ray
6
no
$Z1 = Z(J) + 1000u\sin\phi(J) - w\cos\phi(J)$
$X_2 = (X1-X0)/1000$
$Z2 = -X_2^2/12.8$
$R = Z1 - Z2$
2


5
X1 = 0
6
φo = φo+Δφ
7
END
2
Is R < Z1(K2) ?
yes
no
Is R ≥ 0 ?
no
X1=0
6
yes
K2=K2-1
Is R≤Z1(K2+2) ?
no
K2=K2+1
change axes
3
yes
Is R ≥ R9?
no
Is R ≥ Z1(K2)?
no
K2=K2-1
yes
yes
Is R ≤ 0?
yes
6
no
Is R > R1?
yes
STOP
no
Is X < 1000H
no
x=1000H
Print φo, R
6
yes
3
Is R > R1?
yes
STOP
no
4
Xo = X
Z(J+1) = R
w1 = [0.001(2T1u+3T2u2+4T3u3+5T4u4)+6x10-6T5u5]sinθ
φ(J+1) = φ(J) - w1 + X1/6400000
5

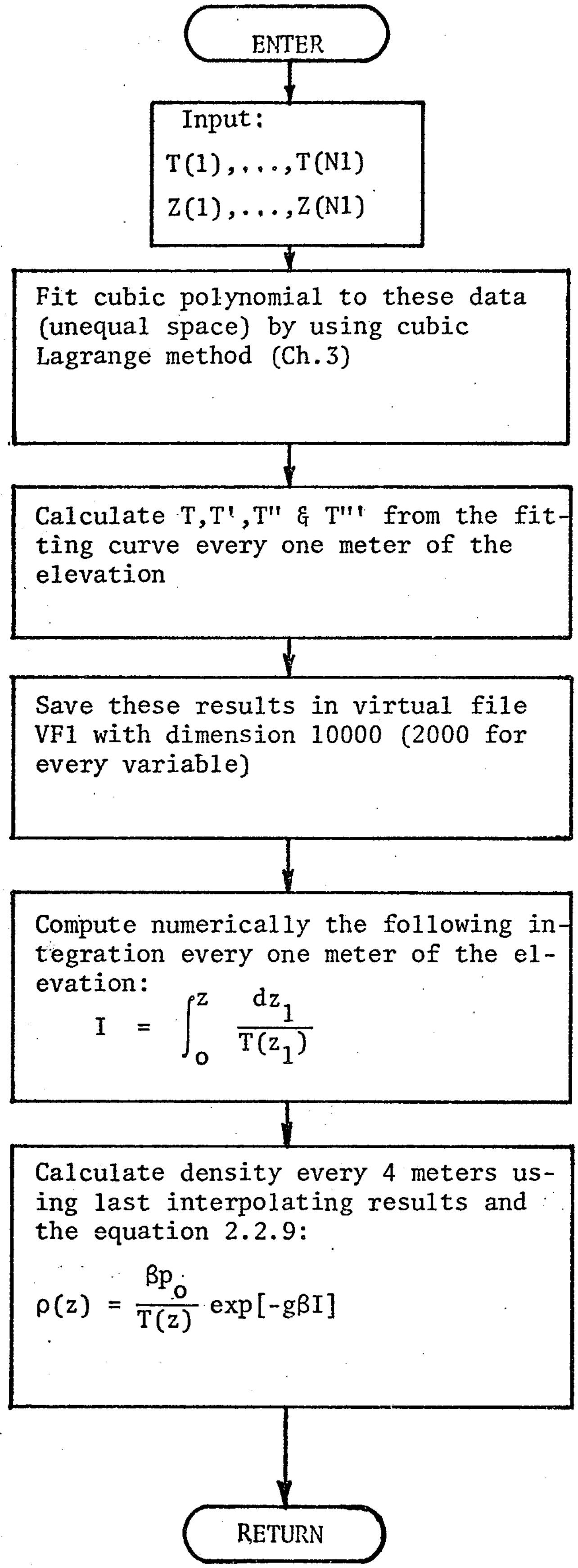
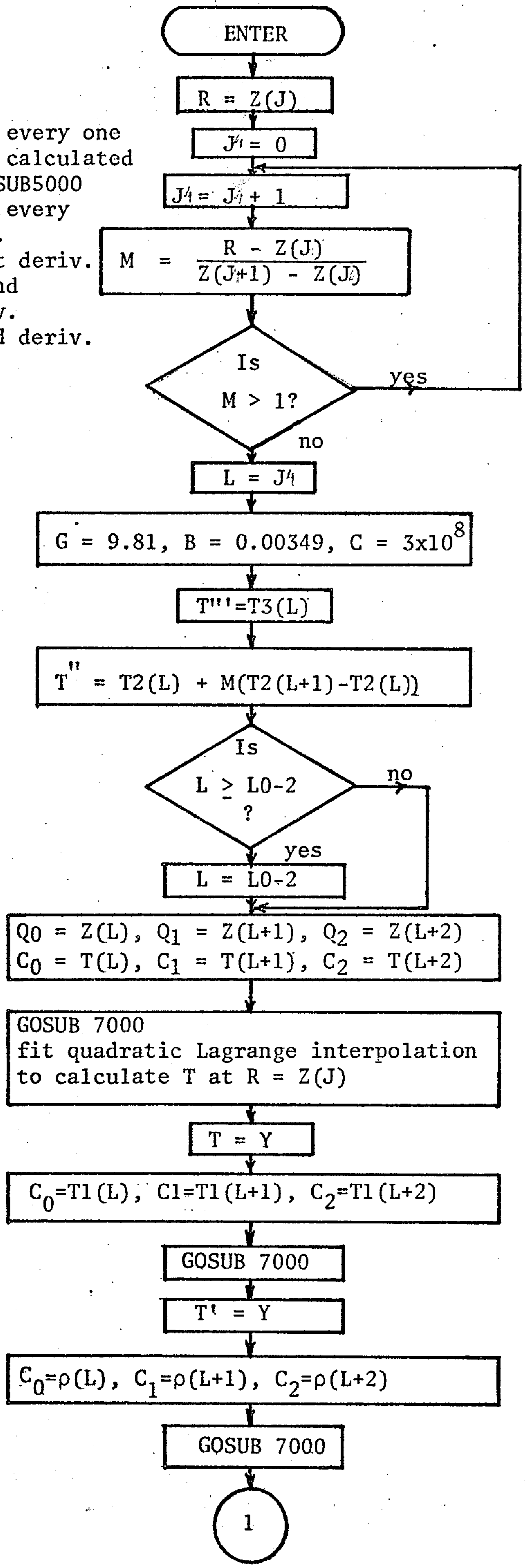


Fig. 4.3: Simplified flow chart for subroutine 5000 (NOTE: See Fig. 3.1 for detailed cubic Lagrange interpolation)

NOTES:

1. Z(1),...,Z(LO) Elev. every one meter calculated in GOSUB5000
2. T(1),...,T(LO) Temp. every one m.
3. T1(1),..,T1(LO) First deriv.
4. T2(1),..,T2(LO) Second deriv.
5. T3(1)...,T3(LO) Third deriv.



1

$$\rho = Y$$

$$\rho' = \frac{-\rho(T' + GB)}{T + 273}$$

$$\rho'' = \left(-\rho' + \frac{\rho T'}{T+273}\right)\left(\frac{T'+GB}{T+273}\right) - \frac{\rho T''}{T+273}$$

$$\rho''' = \frac{-(T'+GB)\rho''}{T+273} + \frac{(T'+GB)\rho' T'}{(T+273)^2} - \frac{\rho' T''}{T+273} + (T'+GB)\frac{(\rho T''+T'\rho')}{(T+273)^2} - \frac{2(T'+GB)\rho T'^2}{(T+273)^3} - \frac{\rho T'''+T''\rho'}{(T+273)} + \frac{2\rho T' T''}{(T+273)^2}$$

$$v = C(1-0.000226\rho)$$

$$v' = -0.000226C\rho'$$

$$v'' = -.000226C\rho''$$

$$v''' = -0.000226C\rho'''$$

RETURN

Fig. 4.4.: Flow chart for subroutine 6000 to calculate density and velocity derivatives.

ENTER

$$B_0 = C_0$$

$$B_1 = \frac{C_1 - B_0}{Q_1 - Q_0}$$

$$B_2 = \frac{C_2 - B_0 - B_1(Q_2 - Q_0)}{(Q_2 - Q_0)(Q_2 - Q_1)}$$

$$Y = B_0 + B_1(R-Q_0) + B_2(R-Q_0)(R-Q_1)$$

RETURN

Fig. 4.5: Flow chart for subroutine 7000 to use quadratic Lagrange interpolation

The print-out of the main transfer characteristic program and its subroutines in the BASIC language is in Appendix (1).

4.2 Transfer Characteristic Curves for Some Temperature Profiles

Transfer characteristic curves, for the four temperature profiles discussed in Chapter 2, are calculated by using the BASIC program mentioned in section 4.1.

4.2.1 Transfer characteristic curves for temperature profile I

Fig. 4.6 shows seven transfer characteristic curves at horizontal distances 10 Km. to 45 Km. from the observer. We can notice that, the curve at 10 Km. is almost linear with relatively low slope. The curve at 20 Km. is non-linear up to elevation 30 meters and linear for elevations higher than this value, but the ray elevation always increases if the initial elevation angle ϕ_o is increased.

The curves at 25 Km., 30 Km. and 35 Km. are almost similar: at some values of ϕ_o, the ray elevation is decreasing with increasing ϕ_o. All curves up to horizontal distance 35 Km. are continuous.

Transfer characteristic curves at 40 Km. and 45 Km. have discontinuous parts between $0.08° < \phi_o < 0.135°$ for the 40 Km. curve and between $0.035° < \phi_o < 0.145°$ for 45 Km.

4.2.2 Transfer characteristic curves for temperature profile II

Six transfer characteristic curves for this profile are shown in Fig. 4.7 at several horizontal distances between 10 Km. and 60 Km.

The curve at 10 Km. is linear, but there is some non-linearity in the curves at 20 Km. and 30 Km. at low elevation (20 meters to 30 meters).

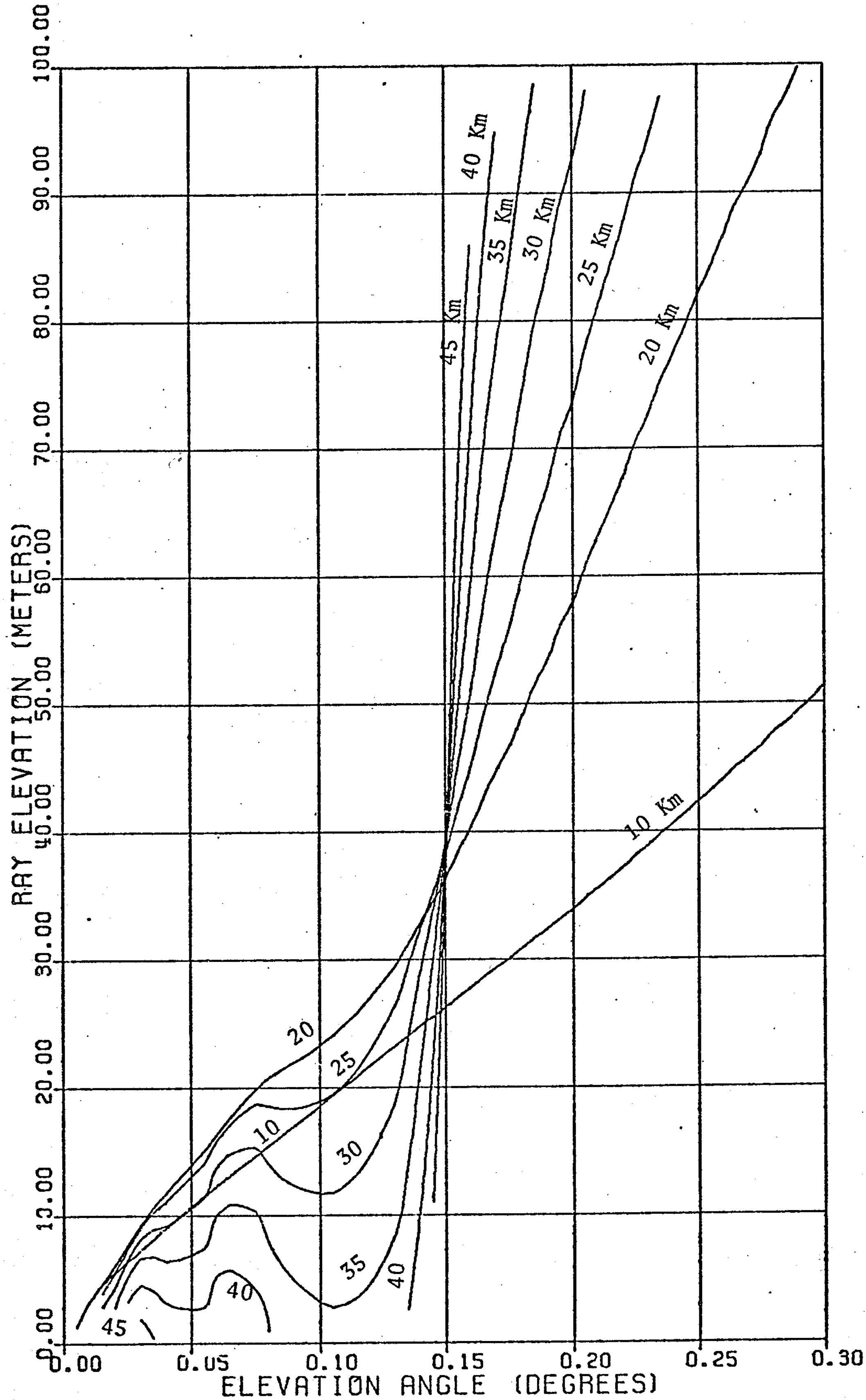


Fig. 4.6: Transfer characteristic curves for Temperature profile I at different horizontal distances (10 to 45 Km.)

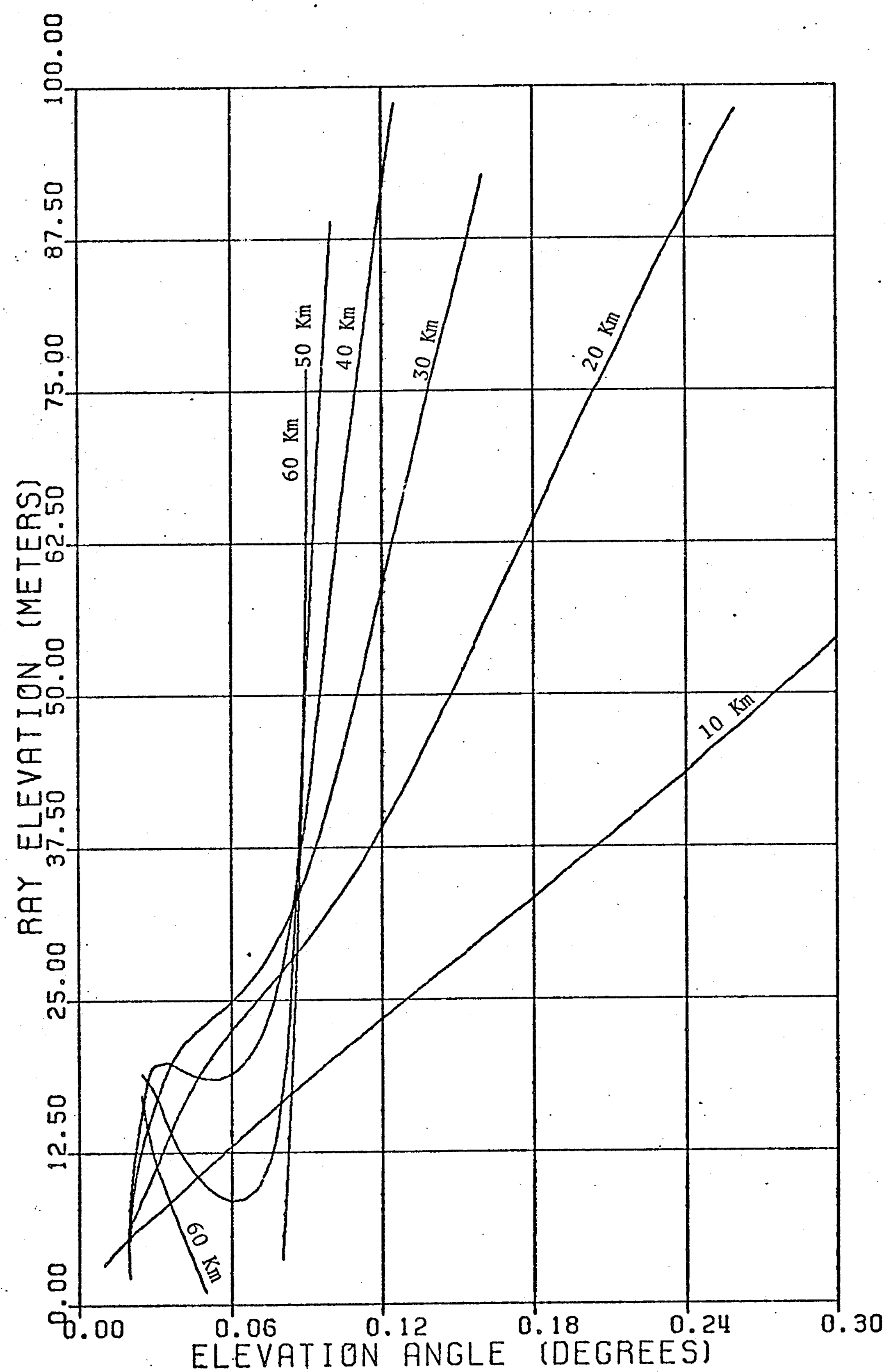


Fig. 4.7: Transfer characteristic curves for Temperature profile II at different horizontal distances (10 to 60 Km.)

The curve at 40 Km, has one local maximum at $\phi_o = 0.035°$ and $z = 19.9$ meters. Also, it has a local minimum at $\phi_o = 0.055°$ and $z = 18.5$ meters. The curve is linear for altitudes above 25 meters. At 50 Km. horizontal distance, the transfer characteristic curve has one local minimum at $\phi_o = 0.06°$ and $z = 8.52$ meters. All transfer characteristic curves up to 50 Km. are continuous. The curve at 60 Km. has two linear parts with a discontinuous part in the region $0.05° < \phi_o < 0.08°$.

4.2.3 Transfer characteristic curves for temperature profile III

Seventeen transfer characteristic curves for temperature profile III are shown in Fig. 4.8 in the range from 10 Km. to 200 Km. horizontal distance. The set of curves between 10 Km. and 50 Km. are almost linear with slope increasing as the horizontal distance is increased. The curves between 60 Km. and 80 Km. have some non-linearities. The group of curves between horizontal distances 90 Km. and 110 Km. have one local maximum and one local minimum. But the curves between 120 Km. and 140 Km. have one local minimum only. All curves up to 140 Km. are continuous. The two curves at 150 Km. and at 160 Km. have some discontinuity at $0.12° < \phi_o < 0.155°$ for the first one and at $0.9° < \phi_o < 0.17°$ for the second one. The curve at 200 Km. is one part only and is linear.

4.2.4 Transfer characteristic curves for temperature profile IV

Ten characteristic curves for horizontal distances between 20 Km. and 200 Km. are shown in Fig. 4.9. It is clear from this figure that all curves are almost linear. They start with low slope at small horizontal distance (20 Km.), and the slope of these curves increases with increasing horizontal distance. Also, it can be noticed that when the transfer characteristic curve is vertical, or at constant initial elevation angle, this angle corresponds to the horizon angle. For this profile, the horizon angle is $0.145°$ at which the transfer characteristic is completely linear and vertical.

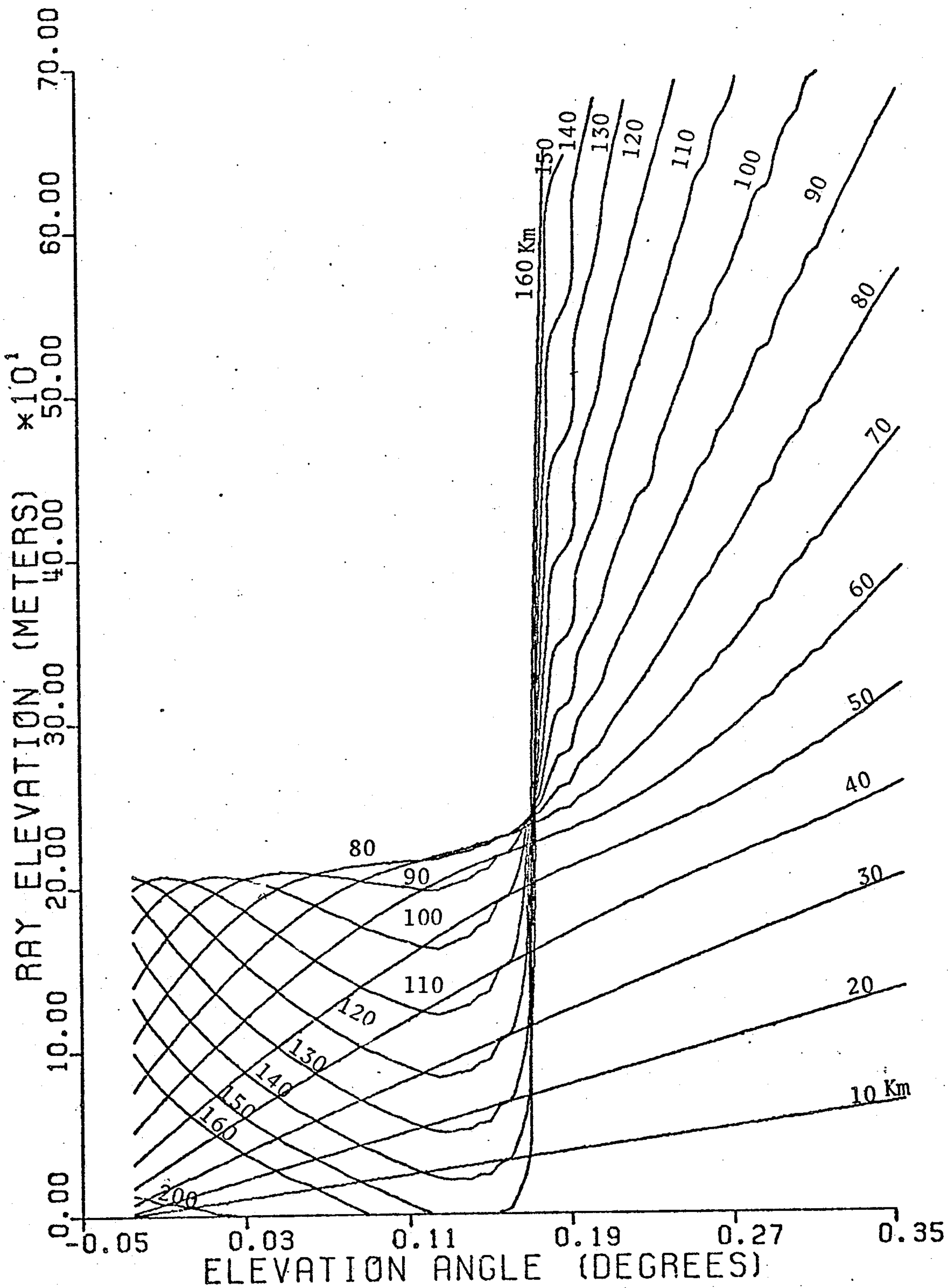


Fig. 4.8: Transfer characteristic curves for temperature profile III at horizontal distances between 10 Km, and 200 Km.

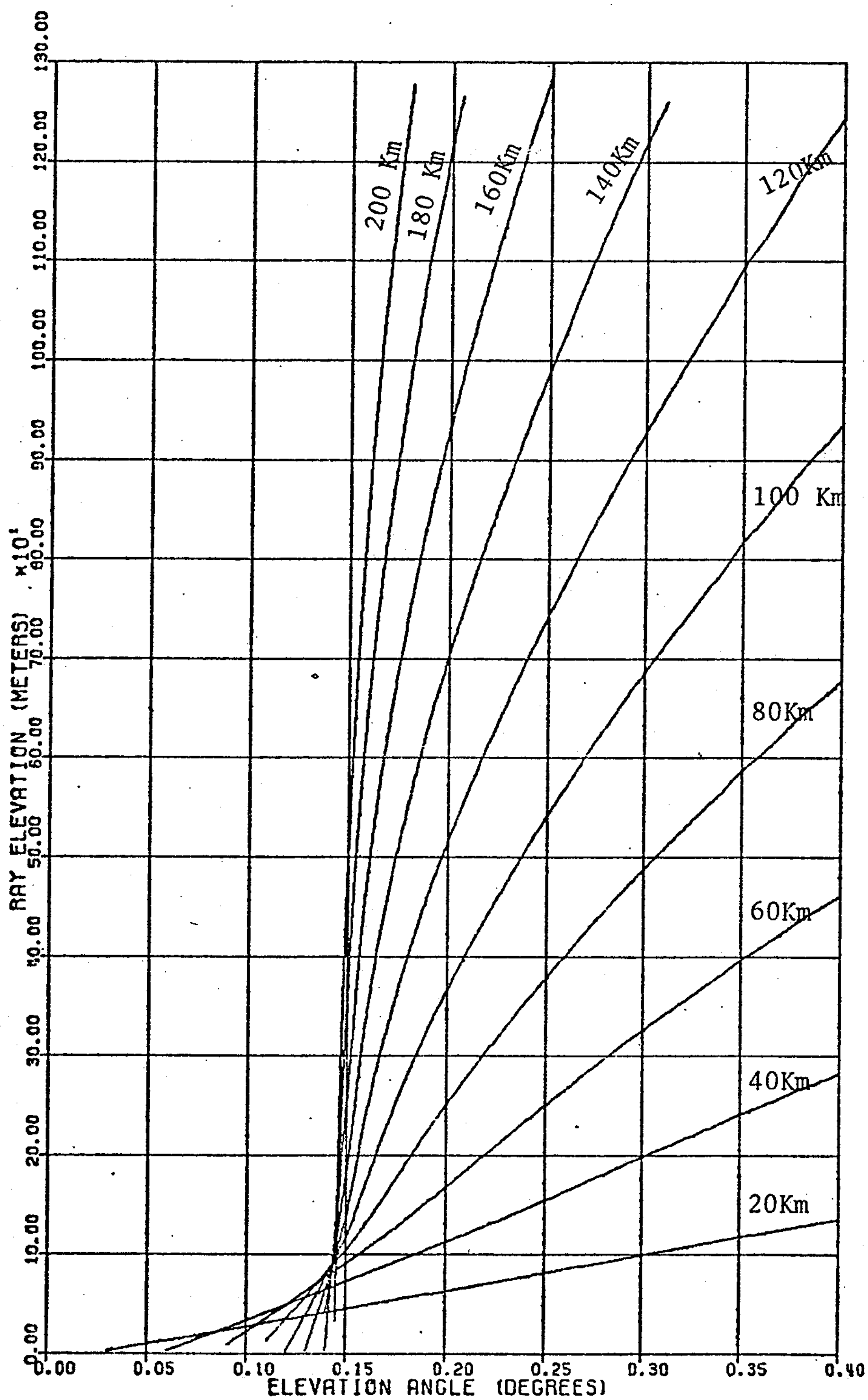


Fig. 4.9: Transfer characteristic curves for temperature profile IV at horizontal distances between 20 Km and 200 Km.

CHAPTER 5

RESULTS AND DISCUSSION

In this chapter, I will explain the BASIC program which gives a computer-graphics representation of images caused by unusual atmospheric conditions. Also, some results with the aid of photographs of the screen will be discussed.

5.1 Flow Chart for Image Representation Program

This program makes use of the transfer characteristic curves, obtained by the Mapping program which is explained in Chapter 4, at any desired horizontal distance between an object and the observer. A picture of the object (in 2 dimensions) is to be drawn directly on the screen of the PDP11/40 graphics terminal by using the light pen.

Once the drawing is complete, the picture of the object is saved in the memory by means of a light pen command. By using the transfer characteristic curve at a certain horizontal distance H(Km.), under any atmospheric conditions entered as a data input to the transfer characteristic program, the image of this object can be represented directly on the computer's screen.

The program can represent the image of objects containing up to fifty vertices, but it is better to use a smaller number of vertices (normally about 20), to leave enough memory space for displaying the image.

Fig. 5.1 shows the simplified flow chart for the image representation program (EA90).

The main idea in image representation is to project the vertical dimensions of the previously-drawn object on the transfer characteristic

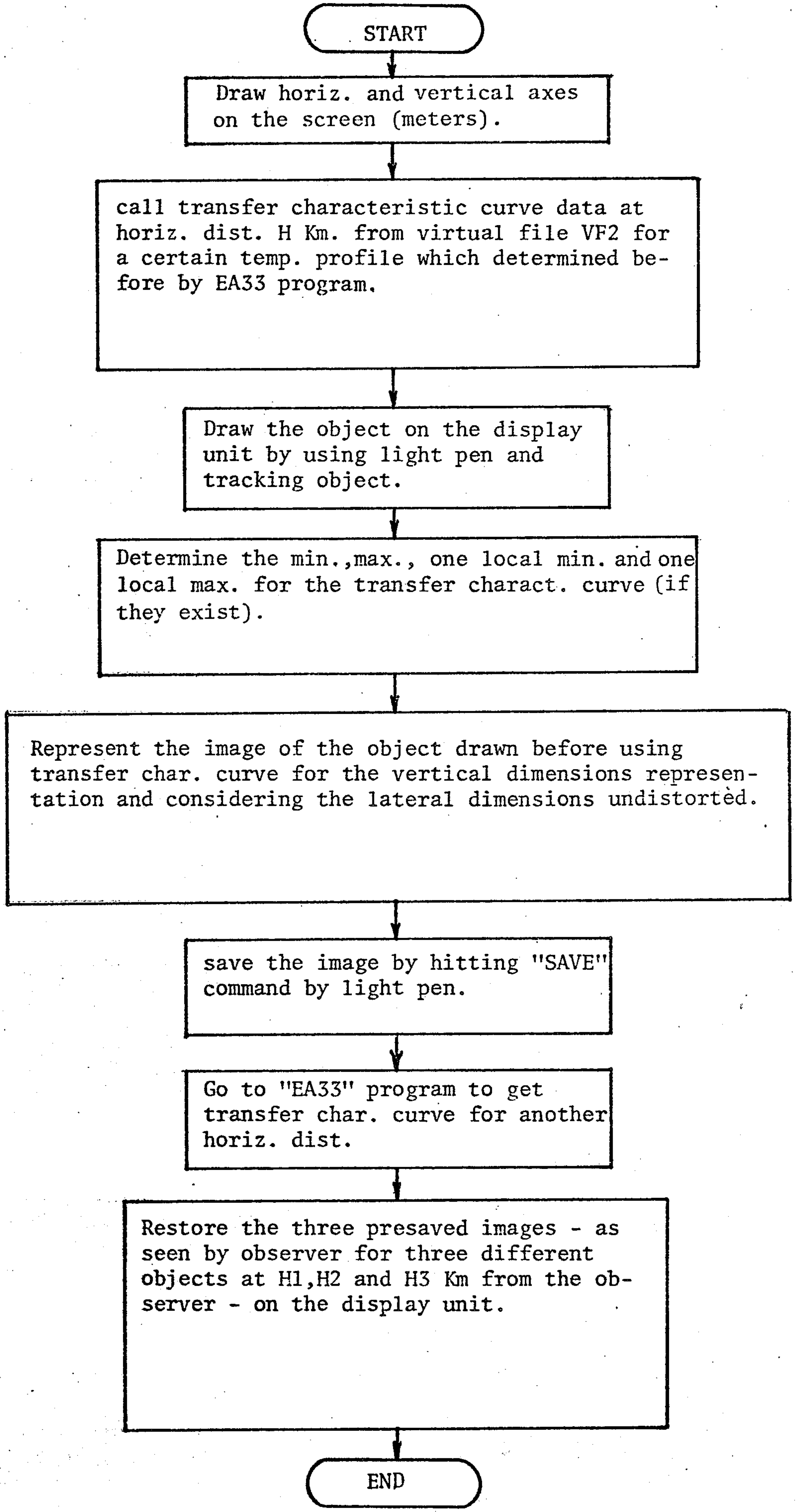


Fig. 5.1: Simplified flow chart for "EA90" image representation program.

curve to get the corresponding elevation angle for every vertex. The lateral dimensions are easily calculated separately, since the atmosphere is considered homogeneous in this direction. The calculation of ϕ_o corresponding to any lateral dimension for a vertex of object with distance X (meters) from a vertical axis at horizontal distance H(Km.) from the observer, is:

$$\phi_o\text{(lateral)} = \frac{\text{X(meters)} \times 57.2958}{1000 \times \text{H(Km.)}} \text{ degrees.}$$

This program can handle several different types of transfer characteristics, as follows:

a. The transfer characteristic curve shown in Fig. 5.2a, which gives one erect image of the object.

b. The curve in Fig. 5.2b gives one inverted image.

c. Fig. 5.2c shows a continuous curve with one local minimum which gives one erect image and one inverted image.

d. Fig. 5.2d shows a curve similar to that of Fig. 5.2c but there is discontinuity in the curve which produces a gap between the two images.

e. Fig. 5.2e shows a transfer characteristic curve with one local minimum and one local maximum. This curve is continuous. It produces three images if the object has dimensions above the local minimum value. Two of these images are erect images and the third one which is in between the erect ones, is inverted.

f. The transfer characteristic curve in Fig. 5.2f gives three images, similar to case (e), but with a gap between the upper two images due to the discontinuity in the curve.

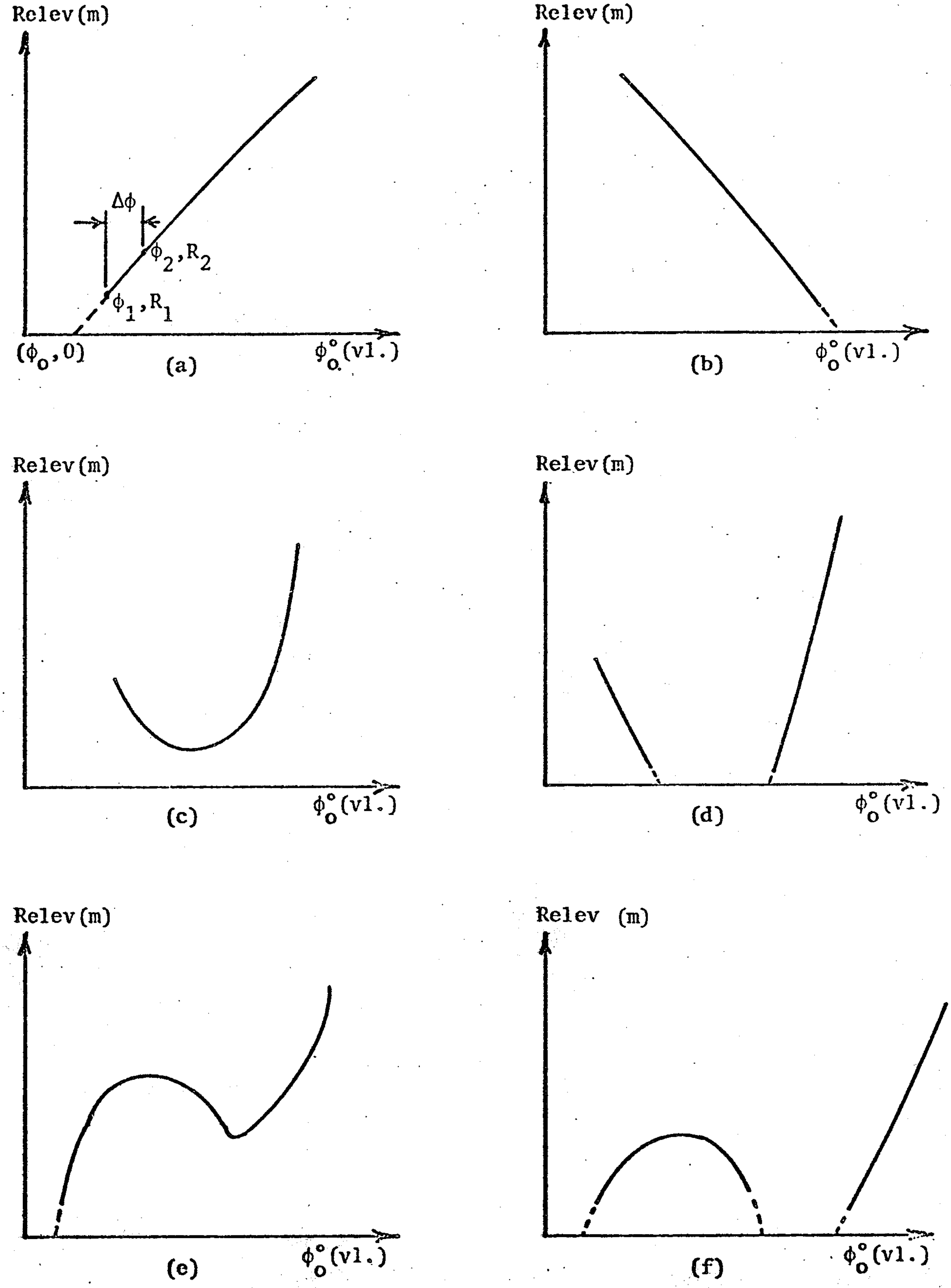


Fig. 5.2: Some transfer characteristic curves which EA90 program can handle to represent images for any object.
(Note: RELEV(m) is ray elevation in meters and ϕ_o°(vl.) is the initial elevation angle at the observer station)

The program cannot handle any curve that has more than one local minimum or maximum due to the limited size of PDP11/40 minicomputer memory.

Fig. 5.3 shows the detailed flow chart for image representation program.

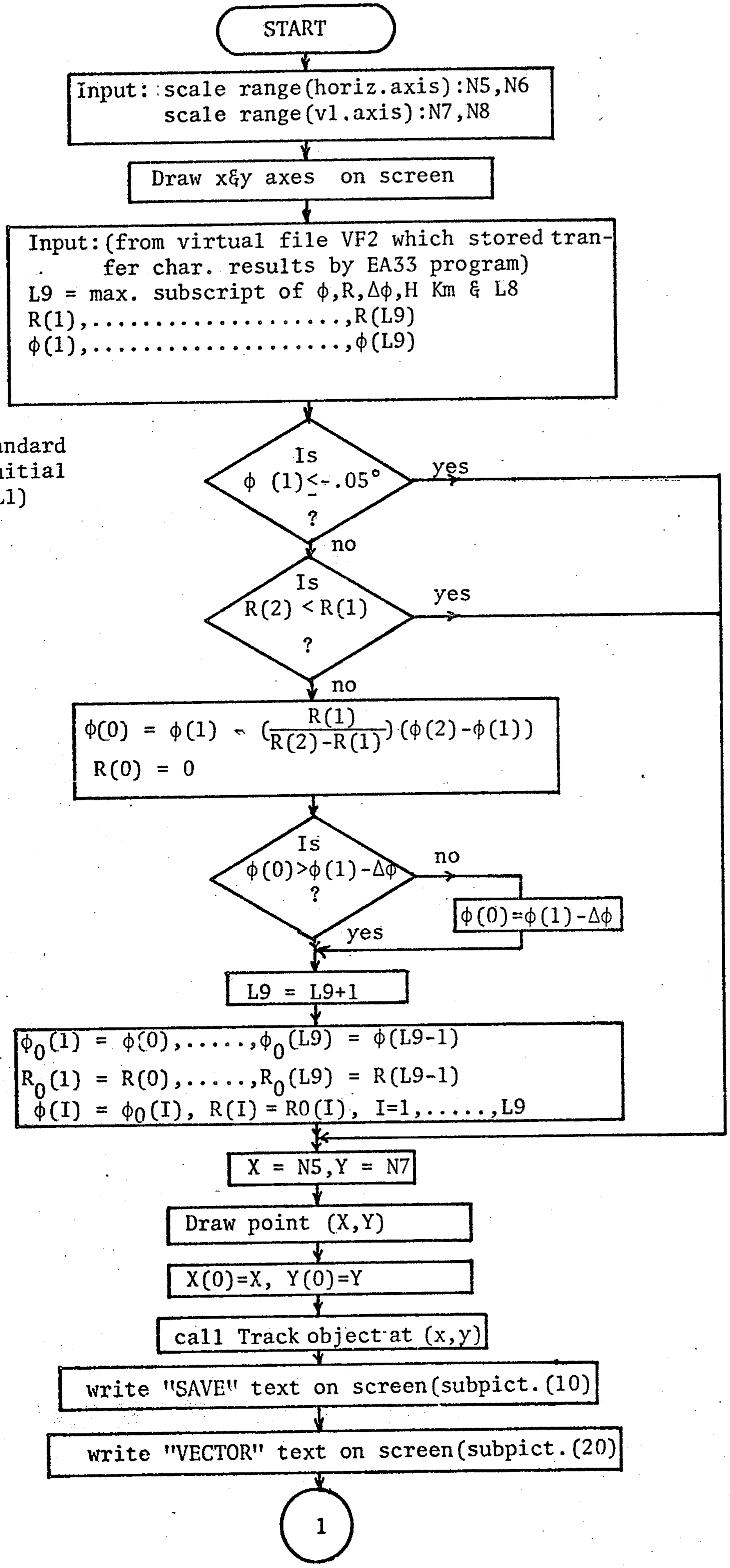
The connection between the two main programs - transfer characteristic program (EA33) and image representation program (EA90) - is shown in the flow chart in Fig. 5.4. The main connector between both programs is through (CHAIN) statements used in every program. That is because of the limited size of PDP11/40 memory (24 K. Words).

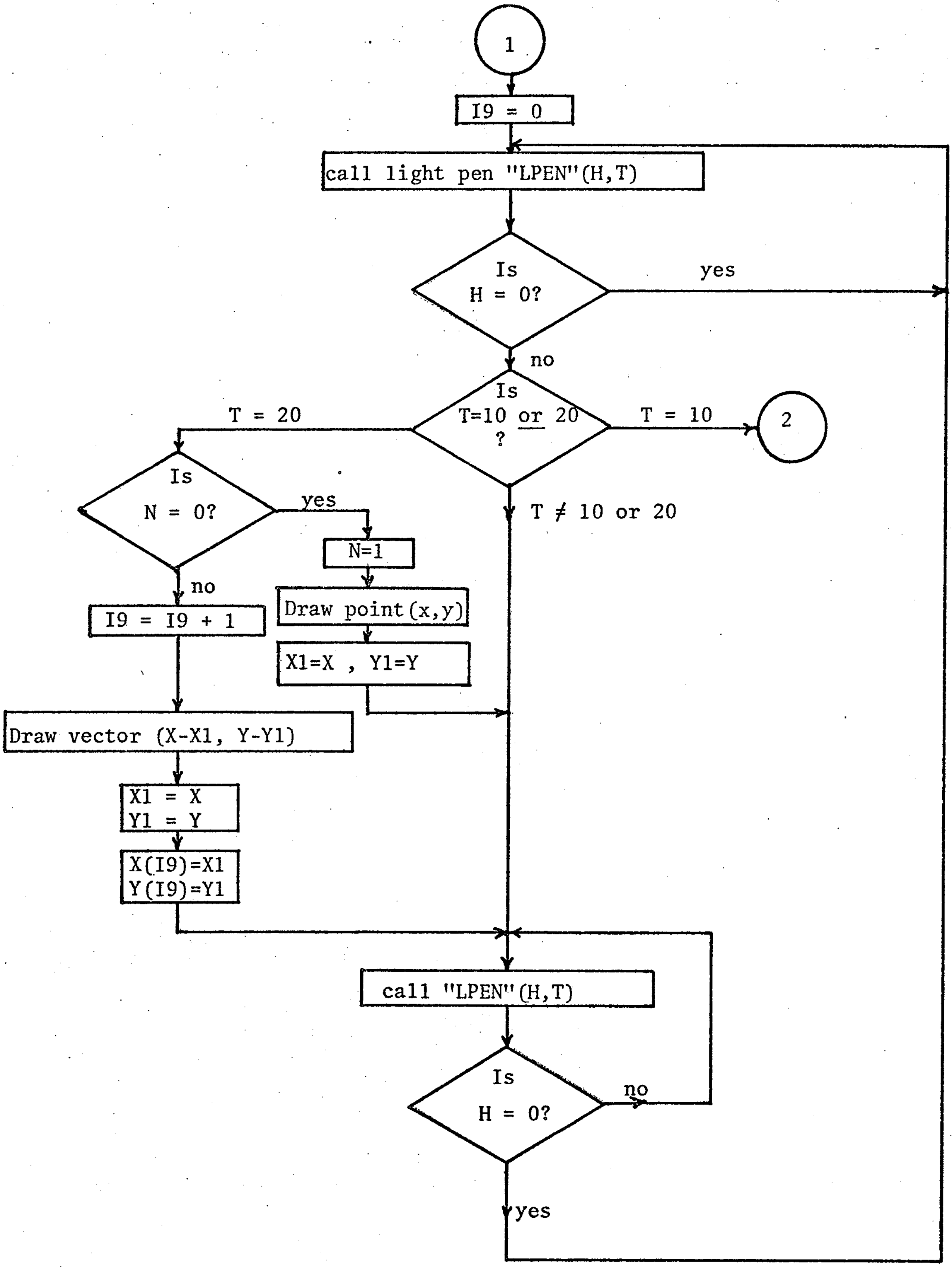
The user of these programs should start with (EA33), the transfer characteristic program. For the first run, he should enter the horizontal distance between the object and observer in kilometers (H), in addition to the temperature profile data. Program (EA33) produces the transfer characteristic curve for this specific temperature profile at this distance. The result will be saved in a virtual file on the disk (due to limitations in memory size).

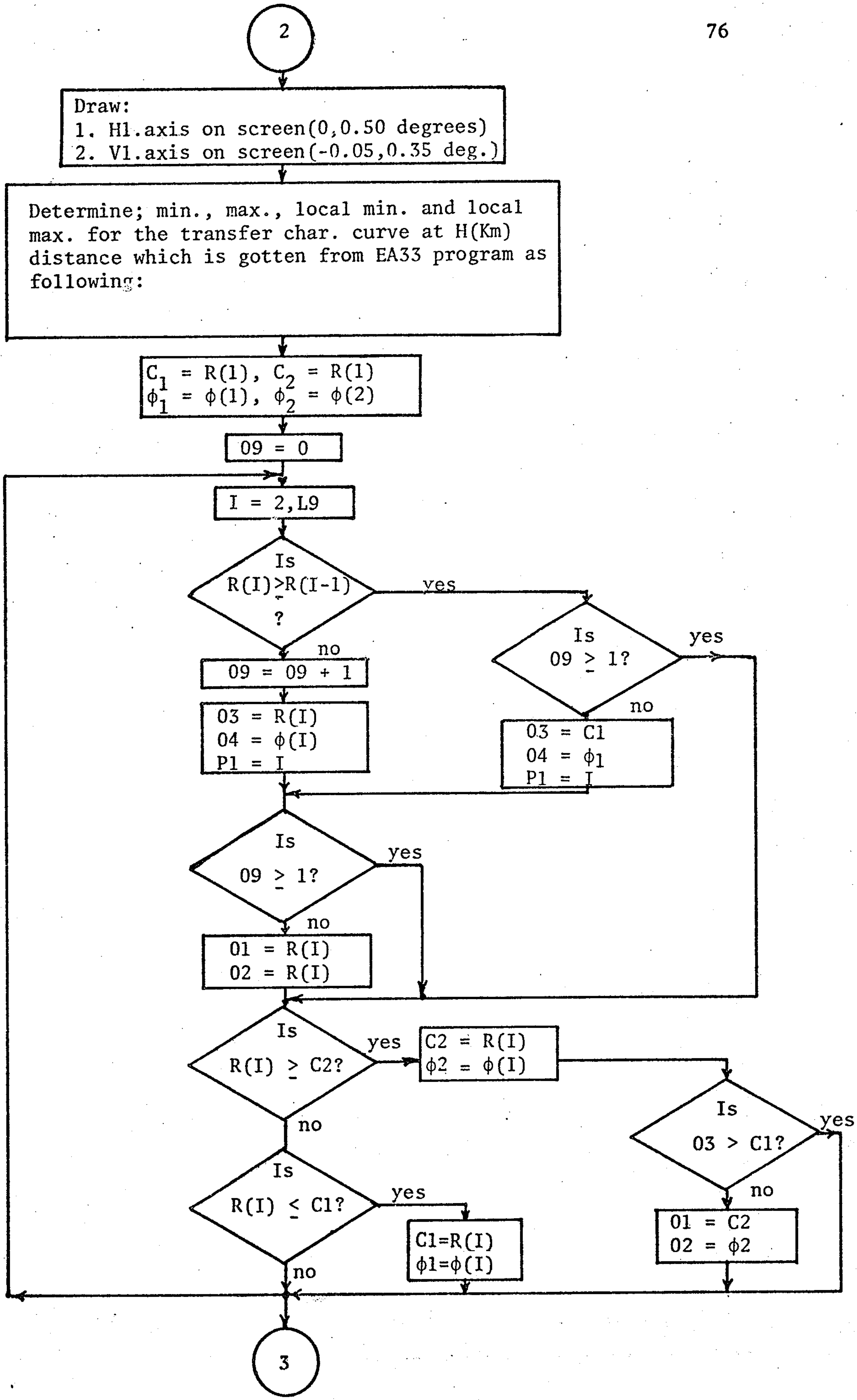
By using the CHAIN statement, these results are transferred to the image representation program (EA90). After the object is drawn on the screen to a chosen scale, by using light pen and tracking object, the image of this object is shown on the graphics terminal. This image can be saved in the memory by hitting the SAVE command on the display unit.

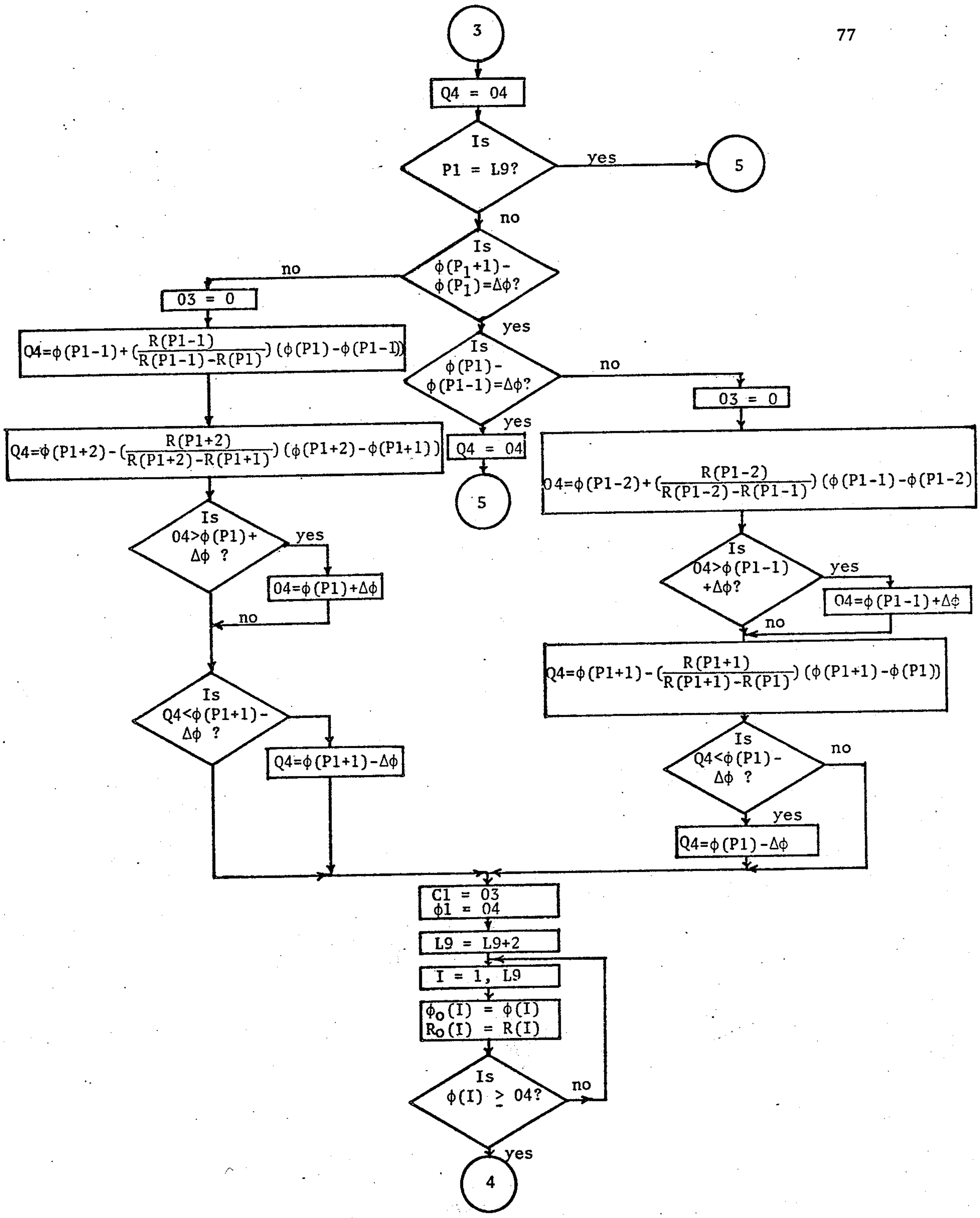
The connection to EA33 program also can be made by using (CHAIN) statement in EA90 program. This means the user can represent the images of three different objects at different horizontal distances from the observer. High light intensity (Z = 6) is used for the image of object in first run, medium intensity (Z = 4) for second run and low intensity (Z = 2) for the last run. For this reason, it is better to represent

NOTE: -0.05° is standard atmosphere initial angle (Ref. L1)




1
I9 = 0
call light pen "LPEN"(H,T)
Is H = 0?
yes
no
Is T=10 or 20 ?
T = 20
T = 10
2
T ≠ 10 or 20
Is N = 0?
yes
no
N=1
Draw point(x,y)
X1=X , Y1=Y
I9 = I9 + 1
Draw vector (X-X1, Y-Y1)
X1 = X
Y1 = Y
X(I9)=X1
Y(I9)=Y1
call "LPEN"(H,T)
Is H = 0?
no
yes


2
Draw:
1. H1.axis on screen(0,0.50 degrees)
2. V1.axis on screen(-0.05,0.35 deg.)
Determine; min., max., local min. and local max. for the transfer char. curve at H(Km) distance which is gotten from EA33 program as following:
C1 = R(1), C2 = R(1)
φ1 = φ(1), φ2 = φ(2)
09 = 0
I = 2,L9
Is R(I)≥R(I-1) ?
yes
no
09 = 09 + 1
03 = R(I)
04 = φ(I)
P1 = I
Is 09 ≥ 1?
yes
no
03 = C1
04 = φ1
P1 = I
Is 09 ≥ 1?
yes
no
01 = R(I)
02 = R(I)
Is R(I) ≥ C2?
yes
C2 = R(I)
φ2 = φ(I)
no
Is 03 > C1?
yes
no
01 = C2
02 = φ2
Is R(I) ≤ C1?
yes
C1=R(I)
φ1=φ(I)
no
3


3
Q4 = 04
Is P1 = L9?
yes
5
no
Is φ(P1+1)-φ(P1)=Δφ?
no
03 = 0
Q4=φ(P1-1)+(R(P1-1)/(R(P1-1)-R(P1)))(φ(P1)-φ(P1-1))
Q4=φ(P1+2)-(R(P1+2)/(R(P1+2)-R(P1+1)))(φ(P1+2)-φ(P1+1))
Is 04>φ(P1)+Δφ ?
yes
04=φ(P1)+Δφ
no
Is Q4<φ(P1+1)-Δφ ?
Q4=φ(P1+1)-Δφ
yes
Is φ(P1)-φ(P1-1)=Δφ?
no
yes
Q4 = 04
5
03 = 0
04=φ(P1-2)+(R(P1-2)/(R(P1-2)-R(P1-1)))(φ(P1-1)-φ(P1-2))
Is 04>φ(P1-1)+Δφ?
yes
04=φ(P1-1)+Δφ
no
Q4=φ(P1+1)-(R(P1+1)/(R(P1+1)-R(P1)))(φ(P1+1)-φ(P1))
Is Q4<φ(P1)-Δφ ?
no
yes
Q4=φ(P1)-Δφ
C1 = 03
φ1 = 04
L9 = L9+2
I = 1, L9
φo(I) = φ(I)
Ro(I) = R(I)
Is φ(I) ≥ 04?
no
yes
4

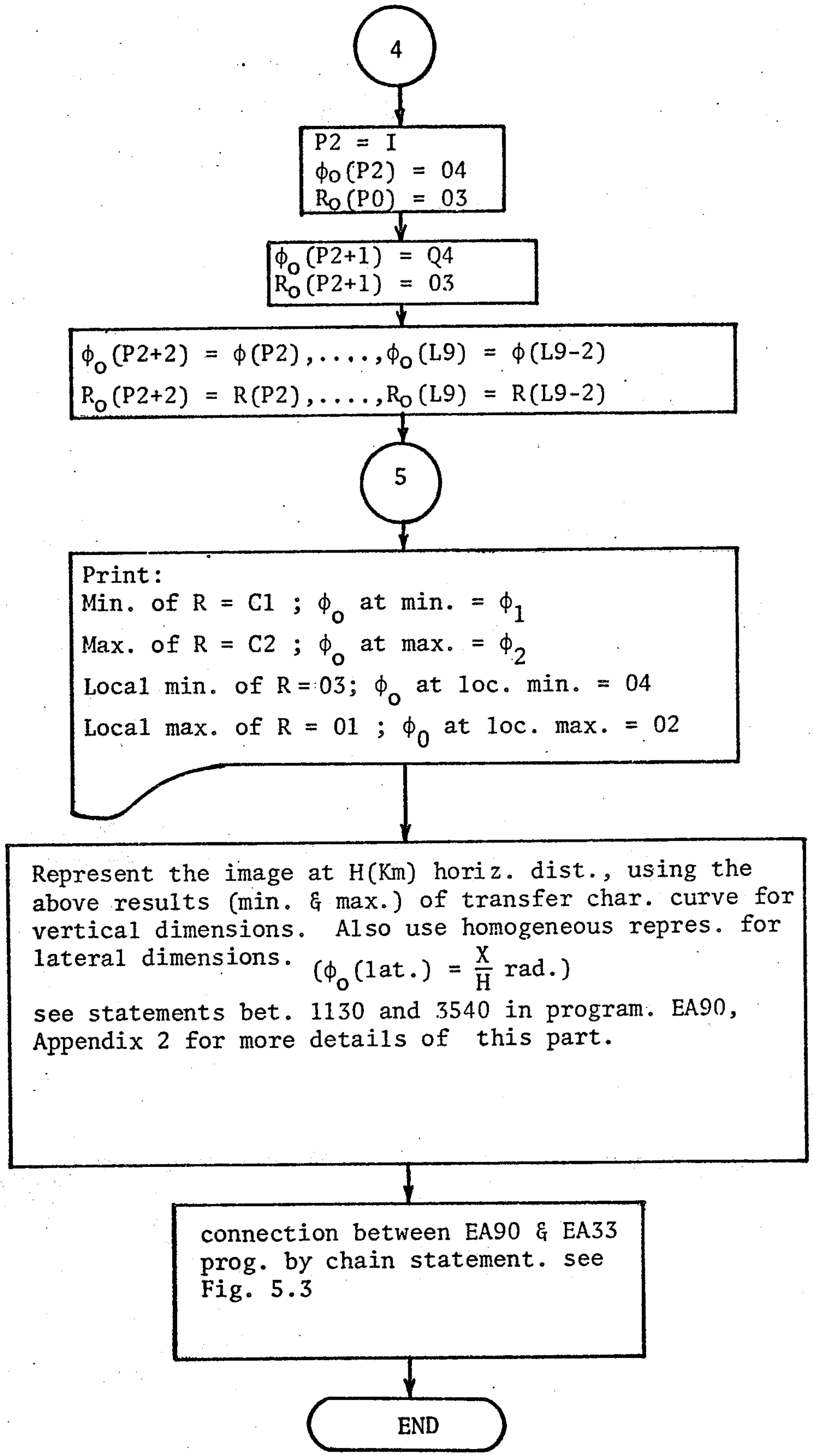


Fig. 5.3: Detailed flow chart for image representation program (EA90).

NOTE: see Appendix (2) for more detail BASIC program.

the nearest object to the observer first, the middle one second and then the farthest one third. After the third run, the images of the three objects at different distances are simultaneously displayed on the screen.

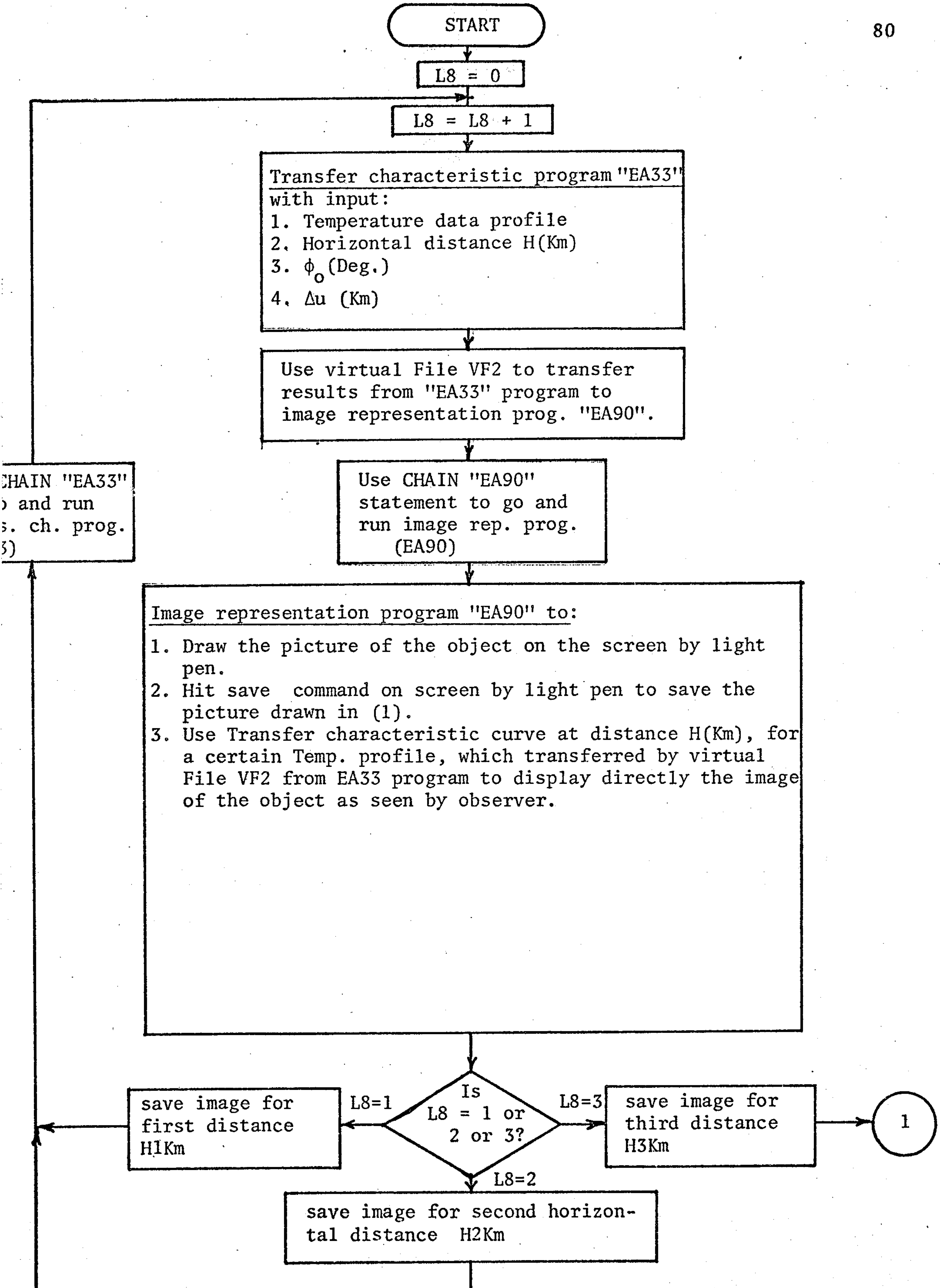
If the horizon elevation corresponding to the temperature profile is known, the user can enter its angle as input. The horizon then appears as a horizontal dash-dot line on the screen, with intensity $(Z = 6)$.

The image representation program (EA90) makes certain approximations in its use of the transfer characteristic curves, as follows:

a. Linear interpolation between transfer characteristic points is used for calculating vertical elevation angles corresponding to all object vertices.

b. For the transfer characteristic curve shown in Fig. 5.2a, the curve starts from the point (ϕ_1, R_1) but we do not have any information before that, and the curve should start at elevation of zero meters. To approximate this, a straight line is connected between the first two points in the curve; its extension will intersect the horizontal axis at the point $(\phi_o, 0)$. If ϕ_o is larger than or equal to $(\phi_1 - \Delta\phi)$, that approximation will be used. But if ϕ_o is less than $(\phi_1 - \Delta\phi)$, then we can take $\phi_o = \phi_1 - \Delta\phi$ at elevation zero as a good approximation.

c. The same approximation technique as in case (b) is used for the transfer characteristic curves in Figs. 5.2b,d,e and f. The approximated sections are shown in these figures by dotted lines.


START
L8 = 0
L8 = L8 + 1
Transfer characteristic program "EA33" with input:
1. Temperature data profile
2. Horizontal distance H(Km)
3. ϕ_o (Deg.)
4. Δu (Km)
Use virtual File VF2 to transfer results from "EA33" program to image representation prog. "EA90".
Use CHAIN "EA90" statement to go and run image rep. prog. (EA90)
CHAIN "EA33") and run ;. ch. prog. ;)
Image representation program "EA90" to:
1. Draw the picture of the object on the screen by light pen.
2. Hit save command on screen by light pen to save the picture drawn in (1).
3. Use Transfer characteristic curve at distance H(Km), for a certain Temp. profile, which transferred by virtual File VF2 from EA33 program to display directly the image of the object as seen by observer.
Is L8 = 1 or 2 or 3?
L8=1
save image for first distance H1Km
L8=3
save image for third distance H3Km
1
L8=2
save image for second horizontal distance H2Km

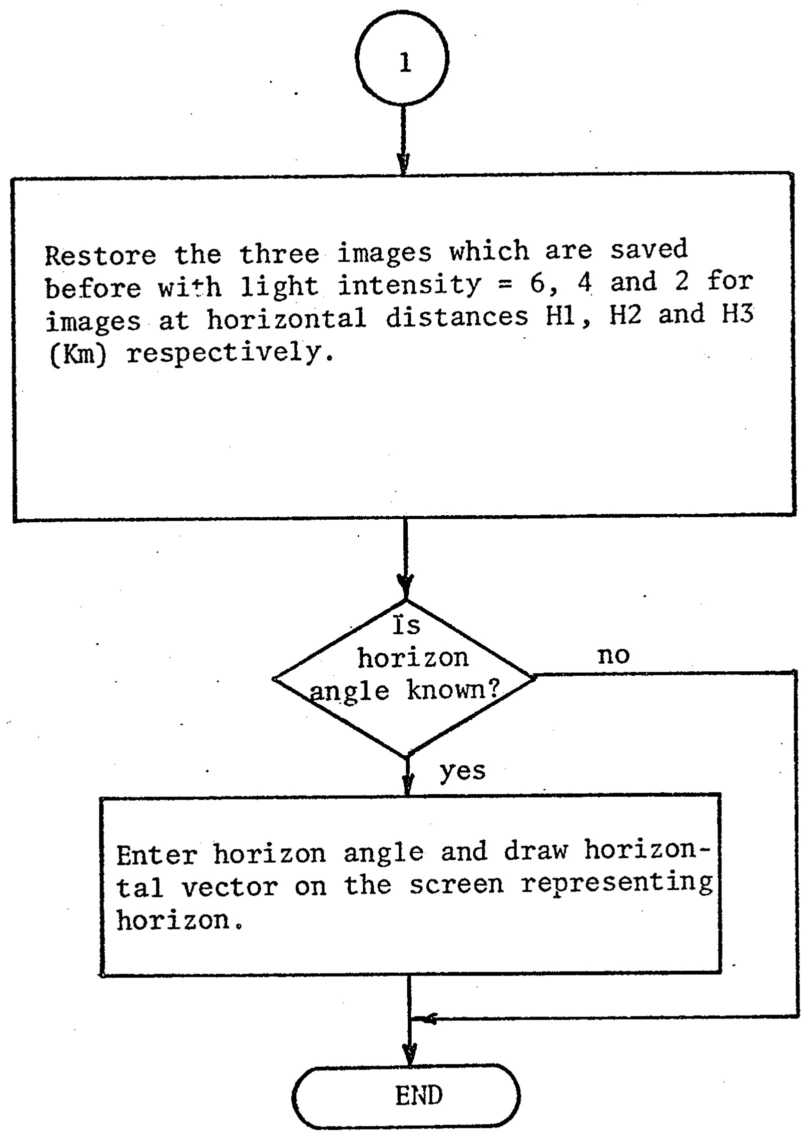


Fig. 5.4: Flow chart for the connection between transfer characteristic program (EA33) and image representation program (EA90) using "CHAIN" statement.

5.2 Results and Discussions

Some results taken by using the two BASIC programs, transfer characteristic and image representation, are presented below. Several different objects are used with different temperature profile:

5.2.1 Image representation in temperature profile I

A ship with 30 meters width and 10 meters mast height is used as an object at 10 Km. distance from the observer. Detailed dimensions of this ship are seen in Fig. 5.5, which is drawn by using light pen and tracking object (diamond shape in Fig. 5.5).

Also two sets of triangular waves are drawn on the display unit. Every set contains four triangles with 10 meters base and 2 meters height. The lateral dimension of the first wave set is between 70 and 110 meters with respect to a reference vertical axis, and it is between 240 and 280 meters for the second set. The two sets are seen at 20 Km. and 40 Km. horizontal distances from the observer. Fig. 5.6 shows the first set of waves drawn on display unit. The second set is the same but with shift in lateral direction to 240 and 280 meters.

The images of these three objects at different horizontal distances are represented on the display unit as seen by the observer, for temperature profile I conditions. These images are shown in Fig. 5.7.

The following observations are clear from Fig. 5.7:

a. The image of the ship at 10 Km. distance is an erect image and all dimensions are clear. The vertical dimensions are compressed with respect to the lateral dimensions. For example; the percentage ratio between the actual height of the mast to the width of the ship $= \dfrac{10\text{(meters)}}{30\text{(meters)}} = 33.33\%$.

   This ratio becomes $= \dfrac{(0.046 - 0.0017)\text{ degrees}}{0.17\text{ degrees}} = 26\%$

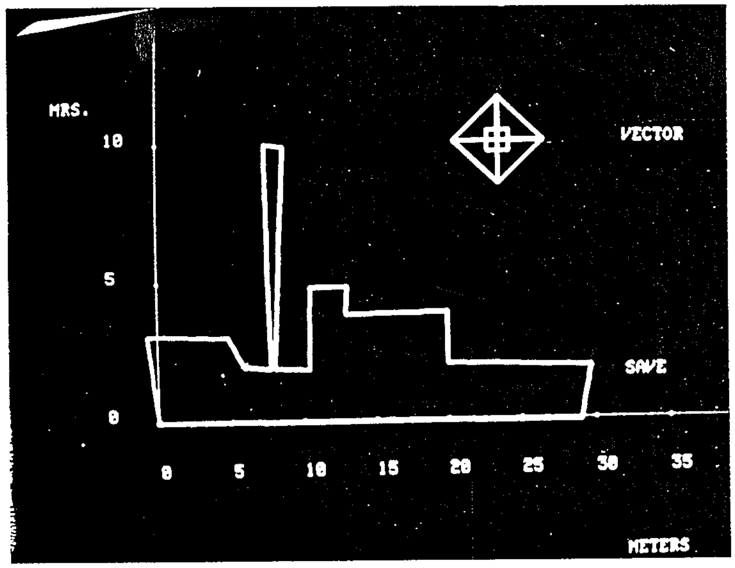


Fig. 5.5: Ship object with 10m height 30m width

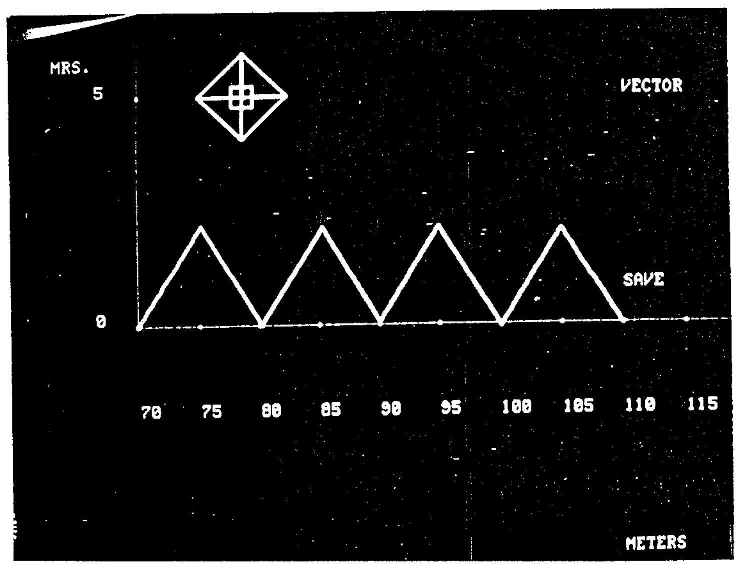


Fig. 5.6: Zigzag waves with 2m height, and 10m width of each one

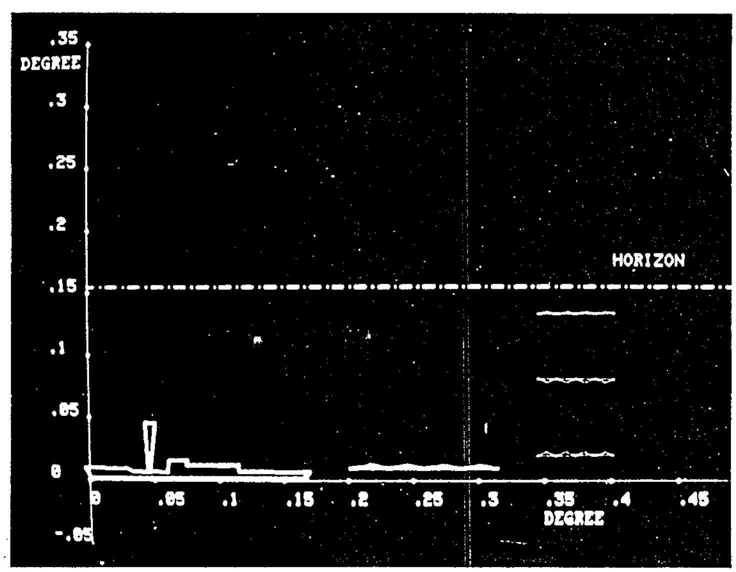


Fig. 5.7: The images of the ship at 10Km, and waves at 20Km and 40Km in temp. profile I

in the image represented in Fig. 5.7.

b. The observer sees one erect image for the waves at 20 Km distance. More distortion occurs in the vertical dimensions. The amount of distortion is found by measuring the ratio between actual height and actual width of the object and comparing it with the observed ones.

$$\frac{\text{actual height of the object}}{\text{actual width}} = \frac{2(\text{meters})}{40(\text{meters})} = 2.50\%$$

$$\frac{\text{apparent height}}{\text{apparent width}} = \frac{0.01233 - 0.01}{0.315 - 0.2} = 2.02\%$$

c. Three images for the object at horizontal distance 40 Km. are observed. Two images are erect but the middle one is inverted. These three images are shown in Fig. 5.7 at the right hand side of the picture. The upper image is more distorted than the middle one, and both are more distorted than the lower one. The upper image is very close to the horizon line which is represented by line at ϕ_o(vertical) = 0.155°

5.2.2 Image representation in temperature profile II

The three objects which were observed in the case of temperature profile I as in Figs. 5.5 and 5.6, are also observed in this profile. The dimensions of each object are the same and they are located at 10 Km., 20 Km. and 40 Km., respectively.

a. One erect image of the ship is observed with percentage ratio between the apparent height of the mast to the apparent width of the ship $= \dfrac{0.044 - 0.00174}{0.1696} = 24.9\%$.

We can notice that this profile exhibits more distortion in the vertical dimensions than temperature profile I at 10 Km.

b. The image of the second object, which is 20 Km. from the observer, is observed as an erect image.

The percentage ratio of height to width of the image =

$$\frac{0.01225 - 0.01}{0.315 - 0.2} = 1.96\%.$$

The distortion in the vertical direction is increased with respect to profile I.

c. Only one completely distorted erect image for the third object is observed in this case, whereas, three images were observed in profile I for the same object and same 40 Km. horizontal distance.

These images are shown in Fig. 5.8 for temperature profile II.

The horizon in this case is at ϕ_0(vertical) equal to 0.087 degrees.

d. Another case for temperature profile II is considered. The object at 10 Km. remains the same ship described above. In addition, a sail-boat (with height 10 meters and width 8 meters, located between 180 and 188 meters on the lateral axis) is placed at 60 Km. from the observer, see Fig. 5.9.

The observer sees one erect image of the ship as described before. But two images for the sail-boat are observed. One of these images is inverted, clearly showing all dimensions of the sail-boat. The second image is erect and above the inverted one with a gap between them. The erect image is completely distorted. The image of the ship and the two images of the sail-boat are shown in the graphics terminal photographs, in Fig.

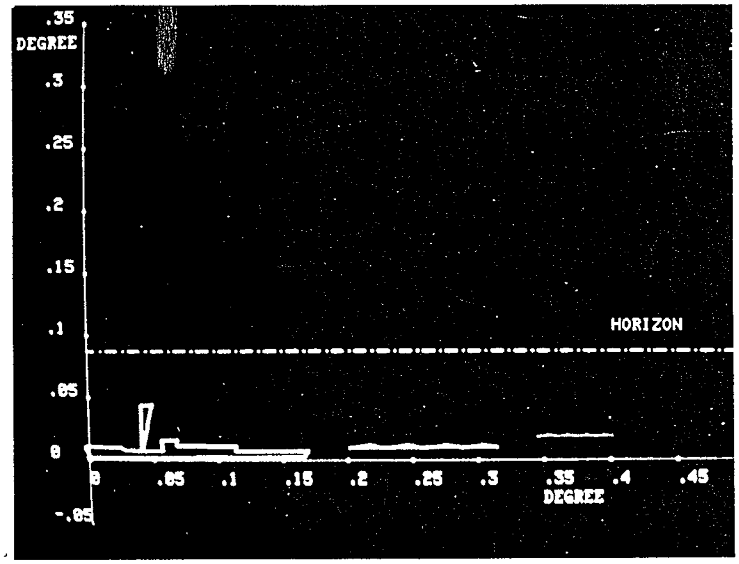


Fig. 5.8: The images of the ship at 10Km, waves at 20Km, and 40Km in temp. profile II

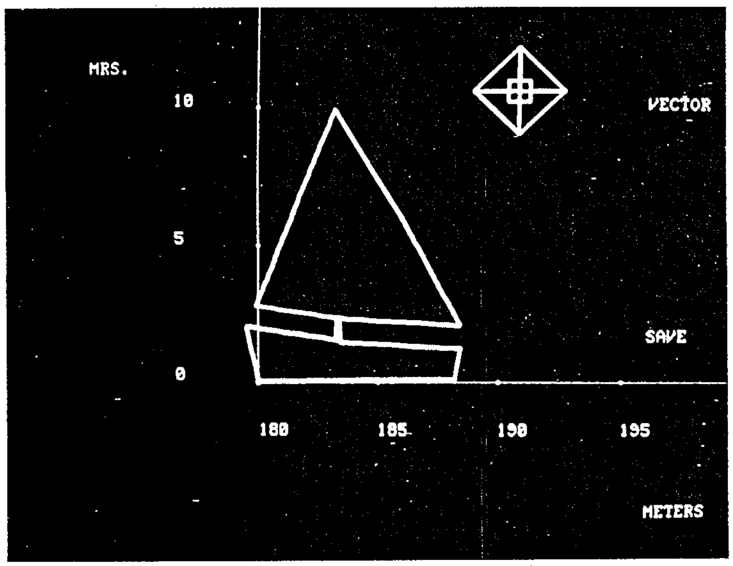


Fig. 5.9: Sailboat object with 10m height and 8m width

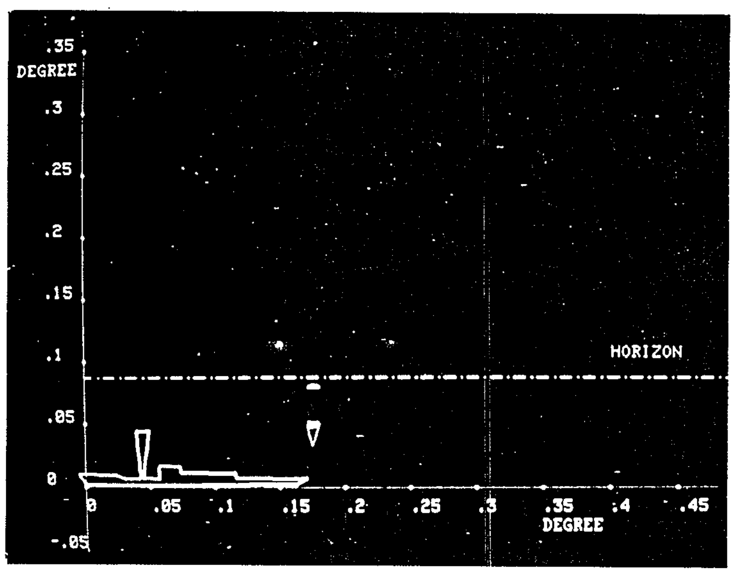


Fig. 5.10: The images of the ship at 10Km and the sailboat at 60Km in temp. profile II

5.10. Notice that the erect image is just below the horizon line.

5.2.3 <u>Image representation in temperature profile III</u>

Since this profile contains a strong inversion layer 300 m. deep, some large objects are chosen to demonstrate the refraction. Three mountains of identical dimensions and shape are observed. Each mountain is a triangle with 150 meters base and 250 meters height. The base of the first object is laterally located between zero and 150 meters, the second between 200 and 350 meters, and the third between 600 and 750 meters. The three objects lie at horizontal distances 60 Km., 80 Km. and 130 Km. respectively. The objects are drawn on the display unit of the minicomputer by using light pen and tracking object. Fig. 5.11 shows the first object; second and third are the same but with a shift in lateral direction. Every leg of the triangle is drawn from five segments of about 50 meters height, to see the effect of refraction for different levels of this kind of atmosphere. The observations are the following:

a. The image for the 60 Km. mountain is a single erect image as seen in the left hand side of the picture in Fig. 5.12. We can notice that the lower segments of the triangle legs are completely compressed but the upper segments are elongated. The ratio between the actual height and actual base of each mountain $= \frac{250}{150} \times 100 = 166.67\%$.
This ratio is still approximately the same for the image of the first object due to compression in one segment, and elongation in the other segments by the same amount approximately.

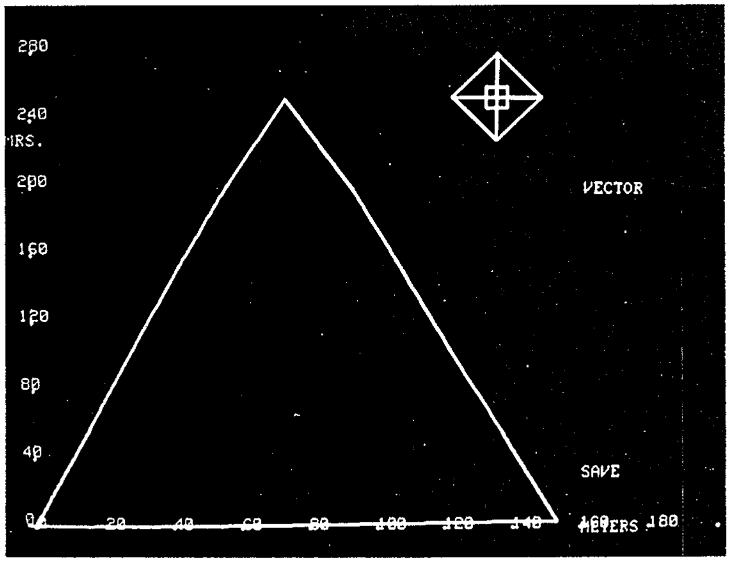


Fig. 5.11: Mountain object (triangle) with 250m in height and 150m base

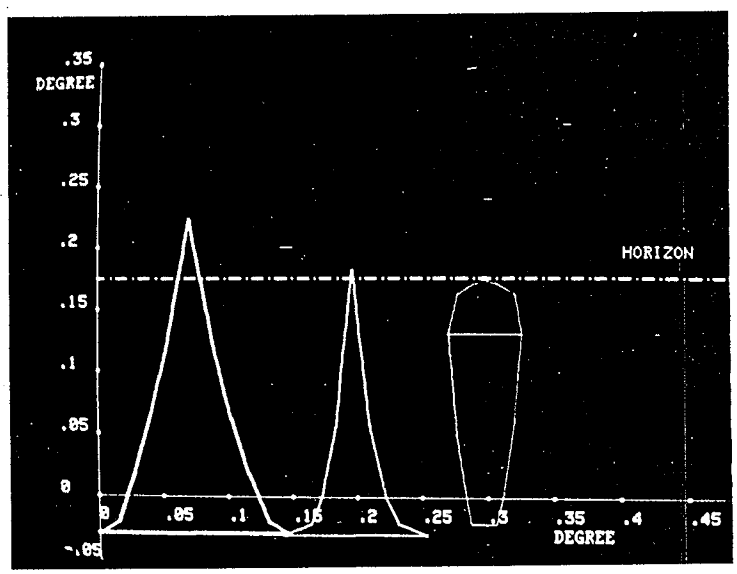


Fig. 5.12: The images of the object in fig. (5.11) at 60Km, 80Km and 130Km in temp. profile III

b) One erect image is observed for the second object which is at 80 Km. distance from the observer. The elongation is concentrated in the middle segments. The ratio of the apparent height to apparent base in this case = 182% approximately; this means the elongation in some segments is larger than the shrinking in the lower segment. We can notice that the images of first and second objects are concave as seen in Fig. 5.12.

c) The observer sees two images of the mountain at 130 Km. distance. The lower image is inverted and all segments are elongated except one, which is completely cut off (segment above 200 meter high). The other image of this third object is erect. The upper four segments of triangle legs are completely distorted as shown in the right hand side of Fig. 5.12. Both images are convex. The top of the upper image is very close to the horizon line, which is at 0.175 degrees.

5.2.4 Image representation in temperature profile IV

In this atmosphere, three objects are observed from 50, 100 and 150 kilometers distance. The objects are the same right angle triangle shape with 50 meters base and 50 meters height. Fig. 5.13 shows the first object as drawn on the minicomputer's screen. The other two objects are the same but with shift in lateral direction. The base of the first object is between zero and 50 meters, for the second between 80 and 130 meters, and it is between 180 and 230 meters for the third. The diagonal of each object consists of three segments, about 17 meters height each. In Fig. 5.14, one erect image of each object is observed below the horizon. The base of each image is at a different level, closest to the horizon for the farthest object. The horizon in this case is at 0.145 degrees, and the earth's surface in this atmosphere has a saucer-shaped appearance. No significant curvature is observed in the diagonal of each image. The most distortion in the vertical dimension occured for the image of the third object. The percentage ratio of height to width for first image is equal to 85.4% but the ratio for actual object is $\frac{50(\text{meters})}{50(\text{meters})} \times 100 = 100\%$. The ratio is decreased to 48% for the second image and 14.1% for the third image which has very high distortion.

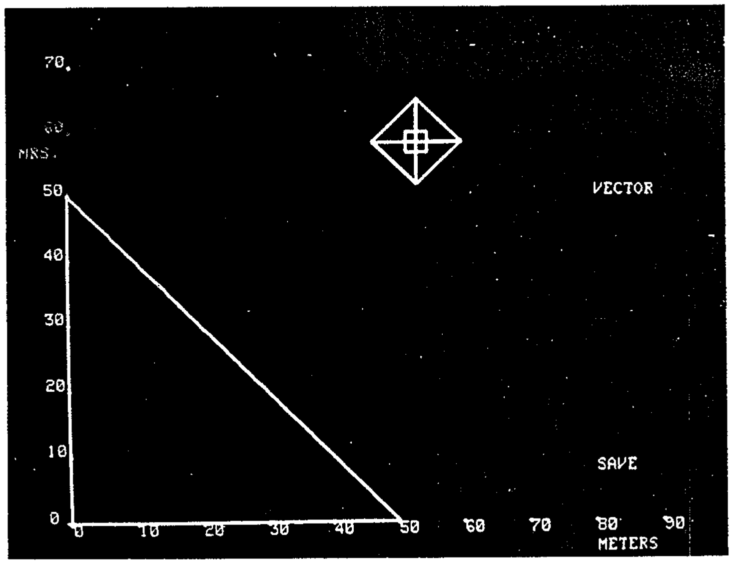


Fig. 5.13: Right angle triangle object with 50m height and 50m base

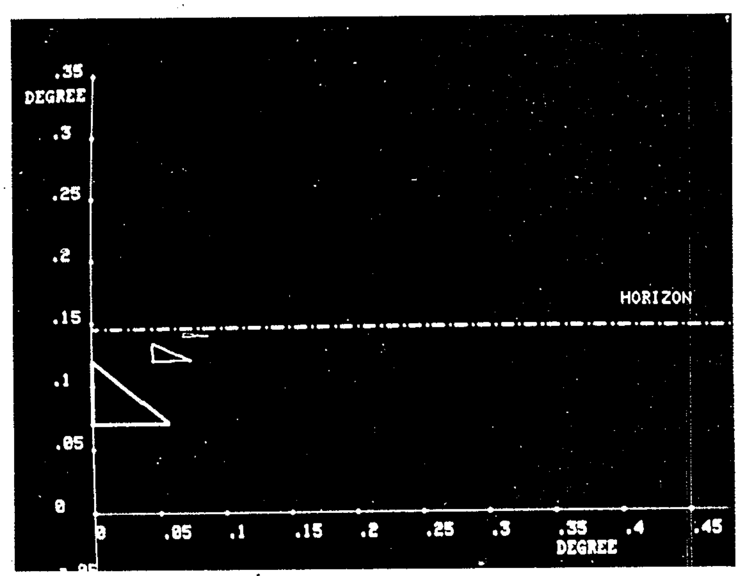


Fig. 5.14: The images of the object in fig. (5.13) at 50Km, 100 Km and 150 Km in temp. profile IV

CHAPTER 6

SUMMARY AND CONCLUSIONS

6.1 Summary

The image representation of objects - as seen by an observer under unusual atmospheric conditions - is discussed in this thesis. The images are displayed directly on the graphics terminal of a minicomputer. Some of the more important results of previous work are summarised.

The equations of the ray paths in the refracting atmosphere as well as the relation between velocity gradient and temperature profile are derived. Ray trajectories are drawn using these equations for four unusual temperature profiles.

Two interpolation methods (Lagrange and spline fit, using cubic polynomials) are discussed along with their flow charts. Also, a comparison between these two methods for handling unequally spaced temperature data in our case is presented. This comparison is based on the percentage change in the value of w in equation (2.1.14) with respect to 1% change in v'' or v''' using the two interpolation methods.

The flow chart of the transfer characteristic program is explained. Also, the results of this program for four unusual temperature profiles are drawn as transfer characteristic curves for each profile at different horizontal distances from the observer.

The use of transfer characteristic curves obtained from this program is important to represent images for any object drawn on the screen of the minicomputer using light pen. Some results are taken using the image representation program for each temperature profile. The images of three objects at different horizontal distances from the observer can be represented simultaneously on the display unit as well as the horizon line of such temperature profile.

6.2 Conclusions

This thesis presents a program that is essential to the analysis of atmospheric optics. It presents a convenient method for calculating image transmission for a wide variety of temperature profiles. The following conclusions are drawn on basis of the minicomputer PDP11/40 results and the data available in the literature:

1. Image of any object as seen by the observer is directly represented on the screen of the minicomputer for the case of unusual atmospheric conditions, provided that the temperature profile data (altitude in meters and temperature in degrees centigrade) are provided as input data to the computer.
2. The images of the three different objects can be represented on the minicomputer's screen at the same time.
3. The calculations of the transfer characteristic curves, particularly for very long horizontal distances, takes a long time. This is because of the small size of the memory of the computer (24 K. Wards). By increasing the computer's memory size, the calculations can be made faster.
4. The use of (CHAIN) statement and virtual files in BASIC language is very important to connect different programs together. But, if the memory size is increased, we can replace these programs with one program.

5. The choice of a numerical technique to handle the unequally spaced temperature data is a difficult decision. The choice of the cubic Lagrange interpolation method is sufficiently accurate in this case. The use of higher degree polynomials with the Lagrange method would be even better. For simplicity, as well as with enough accuracy, a cubic polynomial is used to fit every adjacent four points from the input data. Also, we can conclude that the spline fit method is good and it gives very close results to the Lagrange method.
6. The choice of Δu (horizontal step size) in the ray trajectories calculations is 0.25 km. for the first and second temperature profiles while it is chosen as one kilometer for third and fourth temperature profiles. These values of Δu are chosen after many trials with different values for every temperature profile. With smaller Δu, the accuracy of the calculations will be increased but the execution time becomes very long.

6.3 Limitations for Using Programs

1. The program accepts input data temperature profiles up to elevation = 2000 meters.
2. The program accepts any picture drawn with the light pen on the PDP11/40 screen. The maximum allowable number of vertices is 50, but it is better to draw objects with fewer vertices.
3. The program can calculate the image transmission for transfer

characteristic curves that have at most one local maximum and one local minimum. This restriction is due to the limited memory size of the minicomputer. More complex characteristics have to be interpreted by hand.

4. In the program in Chapter 5, to represent the image, we considered the atmosphere to be laterally homogeneous, and the only spatial variation in density comes with elevation above the earth's surface.

6.4 Suggestion for Future Studies

Some very useful topics for future investigation would be the following extension of the work presented here:

1. Image representation for three dimensional objects in the unusual atmospheres.
2. Study of the ray trajectory equations in the lateral direction instead of considering homogeneous change in this direction.
3. Study of weather conditions for altitudes above 2000 meters, which is the limit of the programs in this thesis due to the limitation in minicomputer memory size.
4. Image representation of moving objects in the unusual atmospheric conditions.
5. Calculation of horizon angle for any temperature profile for direct representation on the screen of minicomputer.

REFERENCES

B1. Bertram,S.,"Compensating for Propagation Errors in Electromagnetic Measuring Systems",I.E.E.E. Spectrum,Vol.8,No.3,(March,1971),pp.58-63.

D1. Digital Equipment Corporation, BASIC/RT11 Language Reference Manual, Vol.1,Dec.-11-LBACA-C-D,Maynard,Massachusetts,(June,1974).

F1. Fraser,A.B.and Mach,W.H.,"Mirages",Scientific American,(Jan.1976), pp.102-111.

H1. Hamming,R.W.,Numerical Methods for Scientists and Engineers,New York, McGraw-Hill Book Co.,1962.

H2. Hildebrand,F.B.,Introduction to Numerical Analysis,2nd.ed.,New York, McGraw-Hill Book Co.,1974.

H3. Hobbs,W.H.,"Conditions of Exceptional Visibility within High Latitudes, Particularly as a result of Superior Mirage",Annals of the American Association of Geographers,(December 1937),pp.229-240.

L1. Lehn,W.H.,and Sawatzky,H.L.,"Image Transmission under Arctic Mirage Conditions",Zeitschrift für Polarforschung,Vol.45,No.2.,(December 1975), pp.120-128.

L2. Liljequist,G.H.,"Refraction Phenomena in the Polar Atmosphere",Norwegian-British-Swedish Antarctic Expedition,Scientific Results,Vol. 2,Part 2B,Norsk Polarinstitutt,Oslo University Press,Oslo,1964,pp. 91-120.

N1. Nölke,V.F.,"Zur Theorie der Luftspiegelungen",Physikalische Zeitschrift,Vol.18,No.7,(March,1917),pp.134-144.

P1. Pennington,R.H.,Introductory Computer Methods and Numerical Analysis, 2nd.ed.London,The MacMillan Co.,1970.

S1. Samuel,M.S.,Standard Mathematical Tables,20th.ed.Cleveland,Ohio,The Chemical Rubber Co.,1972.

S2. Sawatzky,H.L,and Lehn,W.H.,"The Arctic Mirage and the Early North Atlantic,Science,Vol.192,(June,1976),pp.1300-1305.

APPENDIX (1)

```
1 REM THIS PROGRAM IS TO FIND MAPINGS BETWEEN DEPARTURE ANGLE
2 REM    OF RAY TRAJECTORIES AND RAY ELEVATION ABOVE GROUND
3 REM    AT CERTAIN HORIZONTAL DISTANCE
4 L8=0\GO TO 6
5 OPEN "EA42" FOR INPUT AS FILE VF3(2)\L8=VF3(0)
6 L8=L8+1\GO TO 8
7 CHAIN "EA90"
8 CALL "DFIX"(3500)\PRINT \PRINT\CALL"INIT"
9 PRINT "NAME OF TEMPERATURE PROFILE DATA FILE";\INPUT A$
10 OPEN A$ FOR INPUT AS FILE #1
11 INPUT #1:N1\INPUT #1:R1
15 FOR K2=1 TO N1\INPUT #1:Z1(K2)\NEXT K2
17 FOR K2=1 TO N1\INPUT #1:T(K2)\NEXT K2\CLOSE #1\
   IF L8>1GO TO 20
18 PRINT ,"TEMPERATURE INPUT DATA"\PRINT \PRINT ,"Z(METERS)",
19 PRINT "T DEG. C"\FOR K2=1 TO N1\PRINT ,Z1(K2),T(K2)\NEXT K2
20 PRINT\PRINT\DIM Z(100),F(100),A(100),B(100)
21 DIM Z1(28),T(50),D(350),E(350),R(105)
28 GOSUB 5000
30 PRINT \PRINT "HORIZONTAL DISTANCE OF THE OBJECT H(KM)=";
35 INPUT H\ PRINT
40 S=R1
50 Z(1)=3
55 PRINT "HORIZONTAL STEP SIZE (KM.) =";\INPUT U9\PRINT
60 U1=U9
70 PRINT "INITIAL ELEVATION ANGLE F(1)(DEGREES)=";\INPUT F0
71 F(1)=F0/57.2958
75 PRINT \PRINT
80 N=40
90 D1=8.72665E-05
100 D1=2*D1
120 L=N+1
130 I=0
135 I2=0
140 M=1000*57.2958
150 FOR K=1 TO L
151 X1=0
160 X0=0
165 K2=1
170 FOR J=1 TO 100
180 U=0
185 REM
190 REM RESULTS ARE IN METERS, BUT U IN CALCULATIONS IS IN KM.
200 GOSUB 6000
203 R9=R
205 IF J>1GO TO 500
206 IF K>1GO TO 500
210 CALL "NOSC"\CALL "SCAL"(-100,-.05*R1,500,1.5*R1)
220 CALL "APNT"(-100,0,0,-5)
230 CALL "VECT"(600,0,0,2,0,1)
240 CALL "APNT"(0,0,0,-5)
```

```
250 CALL "VECT"(0,R1,0,2,0,1)
260 CALL "APNT"(50,R1,0,-5)
270 CALL "TEXT"("RAY ELEVATION(METERS)")
275 CALL "APNT"(-100,R1,0,-5)
280 CALL "TEXT"("MAPPING AT")
281 CALL "APNT"(-100,.95*R1,0,-5)\CALL "TEXT"("HORIZONTAL")
282 CALL "APNT"(-100,.9*R1,0,-5)
284 CALL "TEXT"("DISTANCE")
286 CALL "APNT"(-100,.85*R1,0,-5)
287 CALL "TEXT"(STR$(H))
288 CALL "APNT"(-70,.85*R1,0,-5)
289 CALL "TEXT"("KM")
290 FOR I0=0 TO R1 STEP (.1*R1)
291 CALL "APNT"(0,I0)
293 CALL "APNT"(5,I0,0,-5)
300 CALL "TEXT"(STR$(I0))
310 NEXT I0
320 CALL "APNT"(350,.05*R1,0,-5)
330 CALL "TEXT"("PHI0 X0.001 DEG")
340 FOR I0=-100 TO 500 STEP 50
345 CALL "APNT"(I0,0)
350 CALL "APNT"(I0-5,-.05*R1,0,-5)
370 CALL "TEXT"(STR$(I0))
380 NEXT I0
500 FOR I1=1 TO 5
510 G=1.5708-F(J)
520 T5=V3*SIN(G)
522 T5=T5*(V1^2*SIN(G)^2/6.48000E+06+1/2949.12-V1*SIN(G)/69120)
525 E1=T5*(U+U1)^6
530 E=E1/1000
540 IF ABS(E)>=1.00000E-05 THEN 800
550 U=U+U1
560 T1=V1/600
561 A=T1*U^2
570 T2=V2*F(J)/1.8
571 B=T2*U^3
580 T3=V2/46080-V1*V2*SIN(G)/2.16000E+06+138.89*V3*F(J)^2
581 C=T3*U^4
590 T4=V3*F(J)*(.013021-V1*SIN(G)/3600)
591 D=T4*U^5
600 W=SIN(G)*(A+B+C+D)+E
610 X1=1000*U*COS(F(J))+W*SIN(F(J))
620 X=X0+X1
630 REM COMPUTATION CEASES IF X EXCEEDS H
640 IF X>1000.1*HGO TO 990
650 Z1=Z(J)+1000*U*SIN(F(J))-W*COS(F(J))
660 X2=(X-X0)/1000
670 Z2=-X2*X2/12.8
675 R=Z1-Z2
681 IF R<Z1(K2)GO TO 992
682 IF R<=Z1(K2+2) THEN 684
683 K2=K2+1\GO TO 750
```

```
684 IF R>=R9GO TO 690
685 IF R>=Z1(K2) THEN 690
686 K2=K2-1\GO TO 750
688 REM
690 IF X<999.9*INT(H)GO TO 700
691 IF H=INT(H) THEN 2000
692 I2=I2+1
693 IF I2<>1GO TO 1990
694 U1=H-INT(H)
695 GO TO 500
700 IF R<=0GO TO 2500
710 REM COMPUTATION CEASES IF RAY ELEVATION EXCEEDS
715 REM   R1(MAXIMUM RAY ELEVATION)
720 IF R>R1GO TO 1505
730 NEXT I1
750 IF R>R1 THEN 1505
800 REM
810 X0=X\U1=U9
815 IF R>R1 THEN 1505
820 Z(J+1)=R
850 W1=(2*T1*U+3*T2*U^2+4*T3*U^3+5*T4*U^4)/1000
    +6.00000E-06*T5*U^5
860 W1=W1*SIN(G)
880 F(J+1)=F(J)-W1+X1/6.40000E+06
890 NEXT J
990 I=I+1\U1=1\B(I)=F(1)*57.2958\R(I)=R\
    CALL "APNT"(1000*B(I),R(I))
991 GO TO 1000
992 IF R>=0GO TO 686
993 X1=0
1000 F(1)=F(1)+D1
1100 U1=U9
1500 NEXT K
1501 REM
1505 REM
1510 L9=I\GO TO 4000
1990 U1=U9
1995 I2=0
2000 IF R<0GO TO 2500
2010 I=I+1
2020 A(I)=F(1)
2025 B(I)=57.2958*A(I)
2045 R(I)=R
2047 L9=I
2048 IF L9<=50 THEN 2050
2049 L9=50
2050 A1=M*A(I)
2100 CALL "APNT"(A1,R)
2105 REM
2110 IF R>SGO TO 3000
2120 LET R3=R\LET F3=F(1)
2150 A3=M*F3
2160 CALL "APNT"(A3,R3)
2170 GO TO 1000
```

```
2500 LET R(0)=R\F1=F(1)
2505 F5=57.2958*F1
2510 REM
2520 F6=M*F1
2530 CALL "APNT"(F6,R1)
2600 GO TO 1000
3000 LET R2=R\LET F2=F(1)
3010 REM
3020 A2=M*F2
3030 CALL "APNT"(A2,R2)
4000 CLOSE VF1\OPEN "EA32" FOR OUTPUT AS FILE VF2(106)
4005 VF2(0)=F5\VF2(51)=H
4007 VF2(105)=L8\VF2(106)=D1*57.2958
4010 FOR I=1 TO L9\VF2(I)=B(I)\NEXT I
4020 FOR I=1 TO L9
4030 VF2(I+52)=R(I)
4040 NEXT I
4050 VF2(103)=R(0)\VF2(104)=L9
4500 CLOSE \GO TO 7
5000 REM THIS SUBROUTINE FOR CONVERSION OF TEMPERATURE PROFILE
5005 REM TO DENSITY TABLE. Z IN METERS, T IN DEGREES C.
5006 REM CALCULATE ALSO DT/DZ,D2T/DZ^2 AND D3T/DZ^3
5008 IF L8>1 THEN 5140
5019 CALL "INIT"
5020 CALL "NOSC"
5030 CALL "SCAL"(-35,-.05*R1,40,1.5*R1)
5035 CALL "APNT"(-35,0,0,-5)
5040 CALL "VECT"(75,0,0,2,0,1)
5045 CALL "APNT"(25,.1*R1,0,-5)
5050 CALL "TEXT"("TEMPERATURE C")
5052 FOR I0=-35 TO 40 STEP 5
5054 CALL "APNT"(I0,0)
5056 CALL "APNT"(I0,-.05*R1,0,-5)
5058 CALL "TEXT"(STR$(I0))
5060 NEXT I0
5062 CALL "APNT"(0,0,0,-5)
5064 CALL "VECT"(0,R1,0,2,0,1)
5066 CALL "APNT"(2,R1,0,-5)
5068 CALL "TEXT"("ELEVATION(M)")
5070 FOR I0=0 TO R1 STEP 50
5071 CALL "APNT"(0,I0)
5072 CALL "APNT"(-4,I0,0,-5)
5073 CALL "TEXT"(STR$(I0))
5074 NEXT I0
5075 CALL "SUBP"(10)
5077 CALL "APNT"(-10,1.05*R1,0,-5)
5078 CALL "TEXT"("TEMPERATURE PROFILE")
5080 FOR K2=1 TO N1\CALL "APNT"(T(K2),Z1(K2))\NEXT K2
5090 CALL "ESUB"
5100 CALL "TIME"(300)
5110 CALL "TIMR"(E3)
5120 IF E3<>0 THEN 5110
```

```
5130 CALL "OFF"(10)
5140 REM    GENERATION OF TEMPERATURE DATA AT 1 METER
5145 REM    INTERVALS(INTERPOLATION)
5150 Z3=-1
5160 L0=0
5170 N2=N1-3
5175 IF L8>1 THEN 5185
5180 CALL "SUBP"(20)
5182 CALL "APNT"(-10,1.05*R1,0,-5)
5183 CALL "TEXT"("INTERPOLATION OF TEMPERATURE DATA")
5185 OPEN "TE" FOR OUTPUT AS FILE VF1(10000)
5190 FOR J2=1 TO N2
5200 B0=T(J2)
5210 B1=(T(J2+1)-B0)/(Z1(J2+1)-Z1(J2))
5220 B2=T(J2+2)-B0-B1*(Z1(J2+2)-Z1(J2))
5230 B2=B2/((Z1(J2+2)-Z1(J2))*(Z1(J2+2)-Z1(J2+1)))
5232 B3=T(J2+3)-B0-B1*(Z1(J2+3)-Z1(J2))
5234 B3=B3-B2*(Z1(J2+3)-Z1(J2))*(Z1(J2+3)-Z1(J2+1))
5236 B3=B3/((Z1(J2+3)-Z1(J2))*(Z1(J2+3)-Z1(J2+1))
     *(Z1(J2+3)-Z1(J2+2)))
5240 FOR K1=1 TO 350
5250 Z3=Z3+1
5260 IF J2>=(N1-3)GO TO 5290
5270 IF Z3<=Z1(J2+2)GO TO 5310
5280 GO TO 5360
5290 IF Z3<=Z1(J2+3)GO TO 5310
5300 GO TO 5360
5310 L2=L0+K1
5315 L3=L2+2000\L4=L2+4000\L5=L2+6000\L6=L2+8000
5320 VF1(L2)=B0+B1*(Z3-Z1(J2))+B2*(Z3-Z1(J2))*(Z3-Z1(J2+1))
5321 VF1(L2)=VF1(L2)+B3*(Z3-Z1(J2))*(Z3-Z1(J2+1))
     *(Z3-Z1(J2+2))
5322 VF1(L3)=B1+B2*((Z3-Z1(J2))+(Z3-Z1(J2+1)))
5323 VF1(L3)=VF1(L3)+B3*(Z3-Z1(J2))*(Z3-Z1(J2+1))
5324 VF1(L3)=VF1(L3)+B3*(Z3-Z1(J2))*(Z3-Z1(J2+2))
5325 VF1(L3)=VF1(L3)+B3*(Z3-Z1(J2+1))*(Z3-Z1(J2+2))
5326 VF1(L4)=2*B2+2*B3*((Z3-Z1(J2))+(Z3-Z1(J2+1))
     +(Z3-Z1(J2+2)))
5327 VF1(L5)=6*B3
5328 VF1(L6)=Z3
5330 REM L2=1 CORRESPONDS TO Z3=0
5335 IF L8>1 THEN 5350
5340 CALL "APNT"(VF1(L2),Z3)
5350 NEXT K1
5360 L0=L2
5370 Z3=Z3-1
5380 NEXT J2
5385 IF L8>1 THEN 5460
5390 CALL "ESUB"
5400 CALL "TIME"(300)
5410 CALL "TIMR"(E5)
5420 REM COMPUTATION OF DENSITY AT 4-METER INTERVALS
5430 IF E5<>0 THEN 5410
5440 CALL "OFF"(20)
5445 CALL "INIT"
```

```
5450 CALL "NOSC"
5451 CALL "SCAL"(-.4,-.05*R1,2,1.5*R1)
5460 T6=0
5470 P=353.5/(VF1(1)+273)
5480 D(1)=P-1.15
5490 E(1)=0
5500 REM    CALCULATE LIMIT ON M FOR SUBSEQUENT FOR LOOP
5510 D2=Z1(N1)
5520 FOR I8=1 TO 350
5530 D2=D2-4
5540 IF D2<0GO TO 5560
5550 NEXT I8
5560 M1=I8
5565 IF L8>1 THEN 5589
5570 CALL "SUBP"(30)
5571 CALL "APNT"(0,0,0,-5)
5572 CALL "VECT"(2,0,0,2,0,1)
5573 CALL "APNT"(1.45,.05*R1,0,-5)
5574 CALL "TEXT"("DENSITY(KG/CU.M)")
5575 FOR I0=0 TO 2 STEP .2
5576 CALL "APNT"(I0,0)
5577 CALL "APNT"(I0-.05,-.05*R1,0,-5)
5578 CALL "TEXT"(STR$(I0))
5579 NEXT I0
5580 CALL "APNT"(0,0,0,-5)
5581 CALL "VECT"(0,.95*R1,0,2,0,1)
5582 CALL "APNT"(.05,R1,0,-5)
5583 CALL "TEXT"("ELEVATION(M)")
5584 FOR I0=0 TO R1 STEP 50
5585 CALL "APNT"(0,I0)
5586 CALL "APNT"(-.17,I0,0,-5)
5587 CALL "TEXT"(STR$(I0))
5588 NEXT I0
5589 FOR M3=2 TO M1
5590 J1=(M3-2)*4
5600 T6=T6+.5/(VF1(J1+1)+273)+1/(VF1(J1+2)+273)
5610 T6=T6+1/(VF1(J1+3)+273)+1/(VF1(J1+4)+273)
5620 T6=T6+.5/(VF1(J1+5)+273)
5630 E2=EXP(-.03424*T6)
5640 M2=(M3-1)*4+1
5650 P=(353.5/(VF1(M2)+273))*E2
5660 D(M3)=P-1.15
5670 E(M3)=(M3-1)*4
5680 IF E(M3)>R1GO TO 5810
5685 IF L8>1 THEN 5700
5690 CALL "APNT"(P,E(M3))
5700 NEXT M3
5705 IF L8>1 THEN 5800
5710 CALL "ESUB"
5720 CALL "TIME"(600)
5730 CALL "TIMR"(E4)
5740 IF E4<>0 THEN 5730
```

```
5750 CALL "OFF"(30)
5760 CALL "INIT"
5800 RETURN
6000 REM THIS SUBROUTINE IS TO CALCULATE THE FIRST THREE
6001 REM DERIVATIVES OF DENSITY AND VELOCITY.
6002 R=Z(J)
6010 J4=0
6020 J4=J4+1
6030 M4=(R-VF1(J4+8000))/(VF1(J4+8001)-VF1(J4+8000))
6040 IF M4>1 THEN 6020
6050 L2=J4\L3=L2+2000\L4=L2+4000
6060 L5=L2+6000\L6=L2+8000
6070 G9=9.81\B9=3.49000E-03\C9=3*10^8
6080 T3=VF1(L5)
6090 T2=VF1(L4)+M4*(VF1(L4+1)-VF1(L4))
6100 IF L2>=L0-2 THEN 6130
6120 GO TO 6140
6130 L2=L0-2\L6=L2+8000\L3=L2+2000
6140 Q0=VF1(L6)\Q1=VF1(L6+1)\Q2=VF1(L6+2)
6150 C0=VF1(L2)\C1=VF1(L2+1)\C2=VF1(L2+2)
6160 GOSUB 7000
6170 T0=Y9
6172 C0=VF1(L3)\C1=VF1(L3+1)\C2=VF1(L3+2)
6174 GOSUB 7000
6176 T1=Y9
6180 J4=0
6190 J4=J4+1
6200 M4=(R-E(J4))/(E(J4+1)-E(J4))
6210 IF M4>1 THEN 6190
6220 L2=J4
6230 IF L2>=M1-2 THEN 6250
6240 GO TO 6260
6250 L2=M1-2
6260 Q0=E(L2)\Q1=E(L2+1)\Q2=E(L2+2)
6270 C0=D(L2)\C1=D(L2+1)\C2=D(L2+2)
6280 GOSUB 7000
6290 P0=Y9+1.15
6300 P1=-P0/(T0+273)*(T1+G9*B9)
6310 P2=(-P1+P0*T1/(T0+273))*(T1+G9*B9)/(T0+273)
6320 P2=P2-P0*T2/(T0+273)
6330 P3=-(T1+G9*B9)*P2/(T0+273)
6340 P3=P3+(T1+G9*B9)*P1*T1/(T0+273)^2
6350 P3=P3-P1*T2/(T0+273)
6360 P3=P3+(T1+G9*B9)*(P0*T2+T1*P1)/(T0+273)^2
6370 P3=P3-2*(T1+G9*B9)*P0*T1^2/(T0+273)^3
6380 P3=P3-(P0*T3+T2*P1)/(T0+273)
6390 P3=P3+2*P0*T1*T2/(T0+273)^2
6400 V=C9*(1-2.26000E-04*P0)
6410 V1=-2.26000E-04*C9*P1
6420 V2=-2.26000E-04*C9*P2
6430 V3=-2.26000E-04*C9*P3
6440 RETURN
```

```
7000 B0=C0
7010 B1=(C1-B0)/(Q1-Q0)
7020 B2=((C2-B0)-B1*(Q2-Q0))/((Q2-Q0)*(Q2-Q1))
7030 Y9=B0+B1*(R-Q0)+B2*(R-Q0)*(R-Q1)
7040 RETURN
8000 END
```

APPENDIX (2)

```
01 REM THIS IS IMAGE REPRESENTATION PROGRAM
02 REM USING TRANSFER CHARACTERISTIC CURVE FROM MAPPING PROGRAM
10 GO TO 30
20 CHAIN "EA33" LINE 5
30 DIM X(50),Y(50),F(50),F1(20),F2(20)
40 DIM R(50),F3(2),F4(2),F8(2),F9(2)
50 DIM F7(50),R7(50),F5(20),F6(20),W5(20),W6(20)
60 CALL "INIT"
70 CALL "SUBP"(100)
80 PRINT \PRINT "SCALE RANGE (HORIZONTAL AXIS)";\INPUT N5,N6
90 PRINT \PRINT "SCALE RANGE (VERTICAL AXIS)";\INPUT N7,N8
100 CALL "SCAL"(N5-10,N7-5,N6,N8)
110 CALL "APNT"(N5,N7,0,-5)
120 V1=.7*(N8-N7)
130 CALL "VECT"(N6-N5,0,0,2,0,1)\CALL "APNT"(N5,N7,0,-5)
140 CALL "VECT"(0,V1,0,2,0,1)
150 FOR I0=N5 TO N6 STEP .1*(N6-N5)\CALL "APNT"(I0,N7)
160 CALL "APNT"(I0,N7-2,0,-5)\CALL "TEXT"(STR$(I0))\NEXT I0
170 FOR I0=N7 TO N8 STEP .1*(N8-N7)\CALL "APNT"(N5,I0)
180 CALL "APNT"(N5-3.5,I0,0,-5)\CALL "TEXT"(STR$(I0))\NEXT I0
190 CALL "APNT"(N5+.8*(N6-N5),N7-5,0,-5)\CALL "TEXT"("METERS")
200 CALL "APNT"(N5-7,N7+.8*V1,0,-5)\CALL "TEXT"("MRS.")
210 CALL "ESUB"
220 OPEN "EA32" FOR INPUT AS FILE VF2(106)
230 L9=VF2(104)\L8=VF2(105)\D1=VF2(106)
240 FOR I=1 TO L9\R(I)=VF2(I+52)\NEXT I
250 FOR I=1 TO L9\F(I)=VF2(I)\NEXT I
260 IF F(1)<=-.05 THEN 370
270 IF R(2)<R(1) THEN 370
280 F(0)=F(1)-(F(2)-F(1))*R(1)/(R(2)-R(1))\R(0)=0
290 D9=F(2)-F(1)
300 IF F(0)>(F(1)-D9) THEN 320
310 F(0)=F(1)-D9
320 L9=L9+1
330 FOR I=1 TO L9\F7(I)=F(I-1)\R7(I)=R(I-1)\NEXT I\PRINT \PRINT
340 FOR I=1 TO L9\F(I)=F7(I)\R(I)=R7(I)
350 NEXT I
360 GO TO 380
370 F(0)=VF2(0)\R(0)=VF2(103)
380 H3=VF2(51)
390 OPEN "EA42" FOR OUTPUT AS FILE VF3(2)\VF3(0)=L8
400 X=N5\Y=N7
410 CALL "APNT"(X,Y)
420 X(0)=X\Y(0)=Y
430 CALL "TRAK"(X,Y)
440 V2=N5+.8*(N6-N5)
450 CALL "APNT"(V2,N7+.1*V1,1,-5)
460 CALL "SUBP"(10)
470 CALL "TEXT"("SAVE")
480 CALL "ESUB"
```

```
490 CALL "APNT"(V2,N7+.7*V1,1,-5)
500 CALL "SUBP"(20)
510 CALL "TEXT"("VECTOR")
520 CALL "ESUB"
530 PRINT \PRINT
540 I9=0
550 PRINT "HIT SUBPICTURE (VECTOR) FIRST BY LIGHT PEN"\PRINT
560 CALL "LPEN"(H,T)
570 IF H=0 THEN 560
580 IF T=10 THEN 800
590 IF T=20 THEN 610
600 GO TO 770
610 CALL "TIME"(120)\CALL "OFF"(20)
620 CALL "TIMR"(E)
630 IF E<>0 THEN 620
640 CALL "DON"(20)
650 Z=1
660 IF N=0 THEN 730
670 I9=I9+1
680 CALL "VECT"(X-X1,Y-Y1,0,0,0,Z)
690 X1=X\Y1=Y
700 X(I9)=X1\Y(I9)=Y1
710 PRINT "X(";I9;")=";X(I9),"Y(";I9;")=";Y(I9)
720 GO TO 770
730 N=1
740 CALL "APNT"(X,Y)
750 X1=X\Y1=Y
760 PRINT "X(";I9;")=";X(I9),"Y(";I9;")=";Y(I9)
770 CALL "LPEN"(H,T)
780 IF H<>0GO TO 770
790 GO TO 560
800 CALL "TIME"(120)\CALL "OFF"(10)
810 CALL "TIMR"(E)
820 IF E<>0 THEN 810
830 CALL "DON"(10)
840 CALL "OFF"(20)
850 CALL "INIT"
860 CALL "NOSC"\CALL "SCAL"(-.05,-.05,.5,.5)
870 CALL "APNT"(0,0,0,-5)\CALL "VECT"(.5,0,0,2,0,1)
880 FOR I0=0 TO .5 STEP .05\CALL "APNT"(I0,0)
890 CALL "APNT"(I0,-.02,0,-5)\CALL "TEXT"(STR$(I0))
900 NEXT I0\CALL "APNT"(.35,-.035,0,-5)\CALL "TEXT"("DEGREE")
910 CALL "APNT"(0,-.05,0,-5)\CALL "VECT"(0,.4,0,2,0,1)
920 FOR I0=-.05 TO .35 STEP .05\CALL "APNT"(0,I0)
930 CALL "APNT"(-.03,I0,0,-5)\CALL "TEXT"(STR$(I0))\NEXT I0
940 I7=0
950 CALL "APNT"(-.05,.33,0,-5)\CALL "TEXT"("DEGREE")
960 REM C1=MIN OF R(I),C2=MAX OF R(I)
970 LET C1=R(1)\C2=R(1)\F1=F(1)\F2=F(1)\K=1
980 O9=0
990 FOR I=2 TO L9
1000 IF R(I)>=R(I-1) THEN 1020
```

```
1010 O9=O9+1\O3=R(I)\O4=F(I)\P1=I\GO TO 1025
1020 IF O9>=1 THEN 1040 \O3=C1\O4=F1
1022 P1=I
1025 IF O9>=1 THEN 1040
1030 O1=R(I)\O2=F(I)
1040 IF R(I)>=C2 THEN 1070
1050 IF R(I)<=C1 THEN 1060 \GO TO 1080
1060 C1=R(I)\F1=F(I)\GO TO 1080
1070 C2=R(I)\F2=F(I)
1075 IF O3>C1 THEN 1080
1077 O1=C2\O2=F2
1080 NEXT I\Q4=O4\IF P1=L9 THEN 1109
1081 IF F(P1+1)-(F(P1)+D1)>=1.00000E-04 THEN 1089
1082 IF F(P1)-(F(P1-1)+D1)>=1.00000E-04 THEN 1083 \
     Q4=O4\GO TO 1109
1083 O3=0\O4=F(P1-2)+R(P1-2)*(F(P1-1)-F(P1-2))/
     (R(P1-2)-R(P1-1))
1084 IF O4>F(P1-1)+D1 THEN 1085 \GO TO 1086
1085 O4=F(P1-1)+D1
1086 Q4=F(P1+1)-R(P1+1)*(F(P1+1)-F(P1))/(R(P1+1)-R(P1))
1087 IF Q4<F(P1)-D1 THEN 1088 \GO TO 1095
1088 Q4=F(P1)-D1\GO TO 1095
1089 O3=0\O4=F(P1-1)+R(P1-1)*(F(P1)-F(P1-1))/(R(P1-1)-R(P1))
1090 Q4=F(P1+2)-R(P1+2)*(F(P1+2)-F(P1+1))/(R(P1+2)-R(P1+1))
1091 IF O4>F(P1)+D1 THEN 1092 \GO TO 1093
1092 O4=F(P1)+D1
1093 IF Q4<F(P1+1)-D1 THEN 1094 \GO TO 1095
1094 Q4=F(P1+1)-D1
1095 PRINT "P1=";P1,"Q4=";Q4,"F(P1)=";F(P1)\C1=O3\F1=O4
1096 L9=L9+2\FOR I=1 TO L9\F7(I)=F(I)\R7(I)=R(I)
1097 IF F(I)>=O4 THEN 1098 \NEXT I
1098 P2=I\F7(P2)=O4\R7(P2)=O3
1099 F7(P2+1)=Q4\R7(P2+1)=O3
1100 FOR I=P2+2 TO L9
1101 F7(I)=F(I-2)\R7(I)=R(I-2)
1102 NEXT I
1103 FOR I=1 TO L9
1104 F(I)=F7(I)\R(I)=R7(I)
1106 NEXT I
1109 PRINT "MIN OF R=";C1;"F1=";F1,"MAX OF R=";C2;"F2=";F2
1110 PRINT "LOCAL MAX OF R=";O1,"PHI AT LOCAL MAX =";O2;"DEG."
1120 PRINT "LOCAL MIN. OF R=";O3,"PHI AT LOCAL MIN.=";O4;"DEG."
1130 L6=0\I8=0\W9=0
1140 FOR I6=1 TO I9
1150 L5=0
1160 IF Y(I6)<C1 THEN 2260
1170 IF Y(I6-1)>C1 THEN 1270
1180 L6=0
1190 I7=I7+1
1200 F2(I6-1)=F1\F4=F2(I6-1)
1210 X9=X(I6-1)+(C1-Y(I6-1))*(X(I6)-X(I6-1))/(Y(I6)-Y(I6-1))
1220 F1(I6-1)=X9*57.2958/(1000*H3)\F3=F1(I6-1)
1230 CALL "APNT"(K*F3,K*F4,0,-5)
1240 PRINT \PRINT
```

```
1250 PRINT "F1(";I6-1;")=";F1(I6-1),"F2(";I6-1;")=";F2(I6-1)
1260 F3(I7)=F1(I6-1)\F4(I7)=F2(I6-1)
1270 I=0
1280 IF R(1)=C1 THEN 1300
1290 GO TO 1400
1300 IF I6>1 THEN 1620 \F1(0)=X(0)*57.2958/(1000*H3)
1310 I=I+1\IF I>=L9 THEN 1330
1320 GO TO 1340
1330 PRINT "CHECK STATMENT NO. 1310"\STOP
1340 G7=ABS(Y(0)-R(I))\G8=ABS(R(I+1)-Y(0))
1350 G9=G7+G8\G6=ABS(R(I+1)-R(I))
1360 IF ABS(G9-G6)>1.00000E-04 THEN 1310
1370 M=G7/G9\F2(0)=F(I)+M*(F(I+1)-F(I))
1380 I=0
1390 GO TO 1620
1400 IF Y(I6)<=R(1) THEN 1530
1410 F2(I6)=F(1)\L5=L5+1\I8=I8+1
1420 IF Y(I6-1)<=R(1) THEN 1460
1430 F1(I6)=X(I6)*57.2958/(1000*H3)
1440 F8(I8)=F8(2)\F9(I8)=F9(2)
1450 GO TO 1480
1460 X7=X(I6-1)+(X(I6)-X(I6-1))*(R(1)-Y(I6-1))/(Y(I6)-Y(I6-1))
1470 F1(I6)=X7*57.2958/(1000*H3)
1480 F8(I8)=F1(I6)\F9(I8)=F2(I6)
1490 IF I8<>2 THEN 1520
1500 GOSUB 2990
1510 I8=0
1520 GO TO 2160
1530 IF Y(I6-1)<=R(1) THEN 1620
1540 IF Y(I6-2)<=R(1) THEN 1570
1550 F8(I8)=F8(2)\F9(I8)=F9(2)
1560 I8=I8+1
1570 GOSUB 2920
1580 GOSUB 2990
1590 GO TO 1620
1600 IF R(2)<R(1) THEN 1620
1605 IF C2=O1 THEN 1624
1610 GO TO 1630
1620 IF Y(I6)<=O1 THEN 1623 \IF Y(I6-1)>O1 THEN 1625
1621 F2(I6)=O2\Y0=O1\GOSUB 3090
1622 F1(I6)=W0\L5=L5+1\GO TO 2160
1623 GOSUB 3500
1624 GOSUB 3230 \GO TO 1630
1625 L5=L5+1\IF L5>1 THEN 1640
1626 F2(I6)=O2
1627 F1(I6)=X(I6)*57.2958/(1000*H3)
1628 GO TO 2160
1630 L5=L5+1
1640 IF L5>=2 THEN 1660
1650 GO TO 2140
1660 IF L5=2GO TO 1680
1670 GO TO 1840
```

```
1680 IF R(2)<R(1) THEN 2200
1690 IF Y(I6-1)>03 THEN 1691 \GO TO 1760
1691 IF Y(I6-1)<=01 THEN 1700 \IF Y(I6)>01 THEN 1716
1692 F3=F5(I6-1)\F4=F6(I6-1)
1693 Y0=01\GOSUB 3090 \F5(I6-1)=W0
1694 F6(I6-1)=02\F5=F5(I6-1)\F6=F6(I6-1)\GOSUB 3120
1695 F3=F5(I6-1)\F4=F6(I6-1)
1700 IF Y(I6)<03 THEN 1720
1701 IF Y(I6)>=01 THEN 1710
1705 GO TO 2070
1710 F6(I6)=02\Y0=01\GOSUB 3090
1712 F5(I6)=W0
1715 GO TO 2100
1716 F6(I6)=02\F5(I6)=X(I6)*57.2958/(1000*H3)
1717 GO TO 2100
1720 Y0=03\GOSUB 3090
1730 F5(I6)=W0\F6(I6)=04
1740 F3=F5(I6-1)\F4=F6(I6-1)
1750 GO TO 2100
1760 IF Y(I6)<03 THEN 1780
1770 GO TO 1830
1780 IF I6<>I9 THEN 1820
1785 IF Y(I6)>03 THEN 1790
1787 IF Y(I6-1)>03 THEN 1790
1788 GO TO 1820
1790 F3=F5(I6-1)\F4=F6(I6-1)
1800 F5=X(I6)*57.2958/(1000*H3)\F6=04
1810 L5=3\GO TO 2670
1820 L5=3\GO TO 2710
1830 IF L5<3 THEN 2000
1840 W9=W9+1\IF W9>=2 THEN 1850 \W5(I6-1)=F5(I6-1)
1841 IF Q4<>04 THEN 1842 \GO TO 1843
1842 W6(I6-1)=Q4\GO TO 1850
1843 W6(16-1)=F6(I6-1)
1850 F3=W5(I6-1)\F4=W6(I6-1)
1860 IF Y(I6)<03 THEN 1880 \IF Y(I6-1)<03 THEN 1912
1870 GOSUB 3230 \GO TO 1940
1880 IF Y(I6-1)<03 THEN 1930
1890 W6(I6)=Q4
1900 Y0=03\GOSUB 3090
1910 W5(I6)=W0\GO TO 1960
1912 W6(I6-1)=04\F4=W6(I6-1)
1914 Y0=03\GOSUB 3090
1916 W5(I6-1)=W0\F3=W5(I6-1)
1920 GOSUB 3230 \GO TO 1940
1930 W6(I6)=04\GO TO 1950
1940 W6(I6)=F(I)+M*(F(I+1)-F(I))
1950 W5(I6)=X(I6)*57.2958/(1000*H3)
1960 PRINT "W5(";I6-1;")=";W5(I6-1);"W6(";I6-1;")=";W6(I6-1);
1970 PRINT "W5(";I6;")=";W5(I6);"W6(";I6;")=";W6(I6)
1980 F5=W5(I6)\F6=W6(I6)\I=0
1990 GO TO 2670
```

```
2000 F6(I6-1)=O4\F4=F6(I6-1)
2010 Y0=O3\GOSUB 3090 \F5(I6-1)=W0\F3=F5(I6-1)
2020 IF Y(I6)<=O1 THEN 2070
2030 F6(I6)=O2\F6=F6(I6)
2040 Y0=O1\GOSUB 3090
2050 F5(I6)=W0\F5=F5(I6)
2060 GO TO 2100
2070 F5(I6)=X(I6)*57.2958/(1000*H3)
2080 GOSUB 3230
2090 F6(I6)=F(I)+M*(F(I+1)-F(I))
2100 PRINT "F5(";I6-1;")=";F5(I6-1);"F6(";I6-1;")=";F6(I6-1);
2110 PRINT "F5(";I6;")=";F5(I6);"F6(";I6;")=";F6(I6)
2120 F5=F5(I6)\F6=F6(I6)
2125 F3=F5(I6-1)\F4=F6(I6-1)
2130 GO TO 2670
2140 F2(I6)=F(I)+M*(F(I+1)-F(I))
2150 F1(I6)=X(I6)*57.2958/(1000*H3)
2160 F5=F1(I6)\F6=F2(I6)
2170 F3=F1(I6-1)\F4=F2(I6-1)
2180 PRINT "F1(";I6;")=";F1(I6),"F2(";I6;")=";F2(I6)
2190 GO TO 2670
2200 L6=L6+1\IF L6>=2 THEN 2210 \C5=F1(I6-1)\C6=Q4
2210 F3=C5\F4=C6
2220 F6=F(I)+M*(F(I+1)-F(I))\C6=F6
2230 F5=X(I6)*57.2958/(1000*H3)\C5=F5
2240 PRINT "F3=";F3,"F4=";F4,"F5=";F5,"F6=";F6
2250 GO TO 2670
2260 IF Y(I6-1)<C1 THEN 2275
2270 GO TO 2320
2275 IF Y(I6)<=C1 THEN 2710
2280 F1(I6)=X(I6)*57.2958/(1000*H3)\F5=F1(I6)
2290 F2(I6)=F1\F6=F2(I6)
2300 F3=F1(I6-1)\F4=F2(I6-1)
2310 L5=2\GO TO 2670
2320 IF R(2)<R(1) THEN 2329
2321 GOSUB 3500
2329 L5=L5+1
2330 I7=I7+1
2340 IF R(1)=C1 THEN 2370
2350 IF Y(I6-1)<=R(1) THEN 2370
2360 GOSUB 2920
2370 F2(I6)=F1\F6=F2(I6)
2380 IF Y(I6-1)>C1 THEN 2440
2390 X8=X(I6)
2400 F1(I6-1)=X(I6-1)*57.2958/(1000*H3)\F3=F1(I6-1)
2410 F2(I6-1)=F1\F4=F2(I6-1)
2420 PRINT "F1(";I6-1;")=";F1(I6-1),"F2(";I6-1;")=";F2(I6-1)
2430 GO TO 2450
2440 X8=X(I6)+(C1-Y(I6))*(X(I6-1)-X(I6))/(Y(I6-1)-Y(I6))
2450 F1(I6)=X8*57.2958/(1000*H3)\F5=F1(I6)
2460 PRINT "F1(";I6;")=";F1(I6),"F2(";I6;")=";F2(I6)
2470 F3(I7)=F1(I6)\F4(I7)=F2(I6)
```

```
2480 IF I7=2 THEN 2500
2490 GO TO 2660
2500 IF L8=1 THEN 2530
2510 IF L8=2 THEN 2540
2520 GO TO 2550
2530 Z2=6\GO TO 2560
2540 Z2=4\GO TO 2560
2550 Z2=2
2560 CALL "APNT"(K*F3(I7-1),K*F4(I7-1),0,-5)
2570 CALL "VECT"(K*(F3(I7)-F3(I7-1)),K*(F4(I7)-F4(I7-1)),0,Z2,0,1)
2580 IF I8<2 THEN 2600
2590 GOSUB 2990
2600 IF L6<1 THEN 2630
2610 CALL "APNT"(K*C5,K*C6,0,-5)
2620 CALL "VECT"(K*(F3(I7)-C5),K*(Q4-C6),0,Z2,0,1)
2625 CALL "VECT"(K*(F3(1)-F3(I7)),0,0,Z2,0,1)
2630 PRINT "I7=";I7;"F3(I7)=";F3(I7);"F4(I7)=";F4(I7);
2640 PRINT "F3(I7-1)=";F3(I7-1);"F4(I7-1)=";F4(I7-1)
2650 F3=F1(I6-1)\F4=F2(I6-1)
2660 I7=0
2670 GOSUB 3120
2680 PRINT "L5=",L5
2690 IF L5>=3 THEN 2710
2700 GO TO 1600
2710 NEXT I6
2715 STOP
2720 IF L8=1 THEN 2750
2730 IF L8=2 THEN 2770
2740 IF L8=3 THEN 2790
2750 CALL "DSAV"("TE1")\PRINT \PRINT
2760 GO TO 2800
2770 CALL "DSAV"("TE2")\PRINT \PRINT
2780 GO TO 2800
2790 CALL "DSAV"("TE3")\PRINT \PRINT
2800 IF L8<3GO TO 2860
2810 CALL "TIME"(300)
2820 CALL "TIMR"(E3)
2830 IF E3<>0 THEN 2820
2840 CALL "RSTR"("TE1")\CALL "RSTR"("TE2")\CALL "RSTR"("TE3")
2850 GO TO 2862
2860 CLOSE \GO TO 20
2862 PRINT "DO YOU KNOW HORIZON (Y OR N)";\INPUT L$
2864 IF L$<>"Y" THEN 2870
2865 PRINT "ENTER HORIZON ANGLE (DEGREES)";\INPUT H9
2866 CALL "APNT"(0,H9,0,-5)
2867 CALL "VECT"(.5,0,0,6,0,4)
2868 CALL "APNT"(.4,H9+.015,0,-6)
2869 CALL "TEXT"("HORIZON")\PRINT \PRINT
2870 PRINT "BEFORE RUNNING THE PROGRAM  AGAIN YOU HAVE TO PRINT,"
2880 PRINT " CONTROL C"
2890 PRINT ".R DELTMP "
2900 PRINT ".R BASGTX "\PRINT \PRINT
```

```
2910 CLOSE \STOP
2920 F2(I6-1)=F(1)
2930 X6=X(I6-1)+(X(I6)-X(I6-1))*(Y(I6-1)-R(1))/
     (Y(I6-1)-Y(I6))
2940 F1(I6-1)=X6*57.2958/(1000*H3)
2950 I8=I8+1
2955 IF I8=2 THEN 2960
2956 I8=1\F8(I8)=F8(2)\F9(I8)=F9(2)\I8=2
2960 F8(I8)=F1(I6-1)\F9(I8)=F2(I6-1)
2980 RETURN
2990 IF L8=1 THEN 3020
3000 IF L8=2 THEN 3030
3010 GO TO 3040
3020 Z2=6\GO TO 3045
3030 Z2=4\GO TO 3045
3040 Z2=2
3045 PRINT ;"F8(";I8-1;")=";F8(I8-1);
3046 PRINT "F9(";I8-1;")=";F9(I8-1);"F8(";I8;")=";F8(I8);
3047 PRINT "F9(";I8;")=";F9(I8)
3050 CALL "APNT"(K*F8(I8-1),K*F9(I8-1),0,-5)
3060 CALL "VECT"(K*(F8(I8)-F8(I8-1)),K*(F9(I8)-F9(I8-1)),0,Z2,0,1)
3070 I8=0
3080 RETURN
3090 X0=X(I6-1)+(Y0-Y(I6-1))*(X(I6)-X(I6-1))/(Y(I6)-Y(I6-1))
3100 W0=X0*57.2958/(1000*H3)
3110 RETURN
3120 IF L8=1 THEN 3150
3130 IF L8=2 THEN 3160
3140 GO TO 3170
3150 Z2=6\GO TO 3180
3160 Z2=4\GO TO 3180
3170 Z2=2
3180 F8=F5-F3
3190 F9=F6-F4
3200 CALL "APNT"(K*F3,K*F4,0,-5)
3210 CALL "VECT"(K*F8,K*F9,0,Z2,0,1)\F3=F5\F4=F6
3220 RETURN
3230 I=I+1
3240 IF I>=L9 THEN 2710
3250 G7=ABS(Y(I6)-R(I))\G8=ABS(R(I+1)-Y(I6))
3260 G9=G7+G8
3270 G6=ABS(R(I+1)-R(I))
3280 IF ABS(G9-G6)>1.00000E-04 THEN 3230
3290 M=G7/G9
3300 RETURN
3500 IF Y(I6-1)<=O1 THEN 3540 \F3=F1(I6-1)\F4=F2(I6-1)
3510 Y0=O1\GOSUB 3090 \F1(I6-1)=W0
3520 F2(I6-1)=O2\F5=F1(I6-1)\F6=F2(I6-1)\GOSUB 3120
3530 F3=F1(I6-1)\F4=F2(I6-1)
3540 RETURN
4000 END
```

APPENDIX (3)

```
01 REM CUBIC LAGRANGE INTERPOLATION FORMULA IS USED HERE
02 REM TO INTERPOLATE UNEQUAL SPACED TEMPERATURE INPUT
03 REM       DATA.
10 CALL "INIT"
20 DIM Z(50),T(50),T0(301),T2(301),T3(301)
25 DIM P0(301),E0(301),Z0(301)
26 DIM T1(301)
30 PRINT \PRINT "DATA FILE NAME";\INPUT A$
40 OPEN A$ FOR INPUT AS FILE #1
50 INPUT #1:N1\INPUT #1:R1
60 FOR I=1 TO N1\INPUT #1:Z(I)\NEXT I
70 FOR I=1 TO N1\INPUT #1:T(I)\NEXT I
80 CLOSE #1
90 CALL "SCAL"(0,0,40,1.5*R1)
100 PRINT \PRINT ,"TEPERATURE INPUT DATA"
110 PRINT \PRINT ,"Z(METERS)","T(DEGREES C)"
120 FOR I=1 TO N1\PRINT ,Z(I),T(I)\NEXT I
130 FOR I=1 TO N1\CALL "APNT"(T(I),Z(I))\NEXT I
140 PRINT \PRINT ,"INTERPOLATION OF TEMPERATURE DATA"
150 PRINT ,"USING CUBIC LAGRANGE FORMULA"
160 REM   GENERATION OF TEMPERATURE DATA AT 1 METER
170 REM   INTERVALS (INTERPOLATION)
180 Z3=-1
190 L0=0
200 N3=N1-3
210 PRINT \PRINT TAB(0);"Z(M)";TAB(5);"T C";TAB(14);"B0";
211 PRINT TAB(23);"B1";TAB(37);"B2";TAB(51);"B3"
220 FOR J=1 TO N3
230 B0=T(J)
240 B1=(T(J+1)-B0)/(Z(J+1)-Z(J))
250 B2=(T(J+2)-B0-B1*(Z(J+2)-Z(J)))
260 B2=B2/((Z(J+2)-Z(J))*(Z(J+2)-Z(J+1)))
270 B3=T(J+3)-B0-B1*(Z(J+3)-Z(J))
275 B3=B3-B2*(Z(J+3)-Z(J))*(Z(J+3)-Z(J+1))
280 B3=B3/((Z(J+3)-Z(J))*(Z(J+3)-Z(J+1))*(Z(J+3)-Z(J+2)))
290 FOR K=1 TO 350
300 Z3=Z3+1
310 IF J>=(N1-3) THEN 340
320 IF Z3<=Z(J+2) THEN 360
330 GO TO 430
340 IF Z3<=Z(J+3) THEN 360
350 GO TO 430
360 L2=L0+K
370 T0(L2)=B0+B1*(Z3-Z(J))+B2*(Z3-Z(J))*(Z3-Z(J+1))
380 T0(L2)=T0(L2)+B3*(Z3-Z(J))*(Z3-Z(J+1))*(Z3-Z(J+2))
381 T1(L2)=B1+B2*((Z3-Z(J))+(Z3-Z(J+1)))
382 T1(L2)=T1(L2)+B3*((Z3-Z(J))*(Z3-Z(J+1))+
    (Z3-Z(J))*(Z3-Z(J+2)))
383 T1(L2)=T1(L2)+B3*(Z3-Z(J+1))*(Z3-Z(J+2))
384 T2(L2)=2*B2+2*B3*((Z3-Z(J))+(Z3-Z(J+1))+(Z3-Z(J+2)))
385 T3(L2)=6*B3
386 Z0(L2)=Z3
```

```
390 REM L2=1 CORRESPONDS TO Z3=0
400 CALL "APNT"(T0(L2),Z3)
410 PRINT TAB(0);Z3;TAB(5);T0(L2);TAB(14);B0;TAB(23);B1;
411 PRINT TAB(37);B2;TAB(51);B3
420 NEXT K
430 L0=L2
440 Z3=Z3-1
450 NEXT J
460 T6=0
470 P0(1)=353.5/(T0(1)+273)
480 E0(1)=0
490 D2=R1
500 FOR I8=1 TO 400
510 D2=D2-4
520 IF D2<0GO TO 540
530 NEXT I8
540 M1=I8
550 FOR M3=2 TO M1
560 J1=(M3-2)*4
570 T6=T6+.5/(T0(J1+1)+273)+1/(T0(J1+2)+273)
580 T6=T6+1/(T0(J1+3)+273)+1/(T0(J1+4)+273)
590 T6=T6+.5/(T0(J1+5)+273)
600 E2=EXP(-.03424*T6)
610 M2=(M3-1)*4+1
620 P0(M3)=(353.5/(T0(M2)+273))*E2
630 E0(M3)=(M3-1)*4
640 IF E0(M3)>R1 THEN 1000
650 NEXT M3
655 PRINT
660 PRINT TAB(0);"Z(M)";TAB(5);"T C";TAB(14);"DENS."
670 PRINT TAB(23);"V";TAB(36);"V1";TAB(45);"V2";TAB(58);"V3"
680 FOR M3=1 TO M1
690 M4=4*M3-3
700 G=9.81\B=3.49000E-03\C=3.00000E+08
710 P1=-P0(M3)/(T0(M4)+273)*(T1(M4)+G*B)
720 P2=(-P1+P0(M3)*T1(M4)/(T0(M4)+273))*(T1(M4)+G*B)/(T0(M4)+273)
730 P2=P2-P0(M3)*T2(M4)/(T0(M4)+273)
740 P3=-(T1(M4)+G*B)*P2/(T0(M4)+273)
750 P3=P3+(T1(M4)+G*B)*P1*T1(M4)/(T0(M4)+273)^2
760 P3=P3-P1*T2(M4)/(T0(M4)+273)
770 P3=P3+(T1(M4)+G*B)*(P0(M3)*T2(M4)+T1(M4)*P1)/(T0(M4)+273)^2
780 P3=P3-2*(T1(M4)+G*B)*P0(M3)*T1(M4)^2/(T0(M4)+273)^3
790 P3=P3-(P0(M3)*T3(M4)+T2(M4)*P1)/(T0(M4)+273)
800 P3=P3+P0(M3)*T1(M4)*T2(M4)/(T0(M4)+273)^2
810 V=C*(1-2.26000E-04*P0(M3))
820 V1=-2.26000E-04*C*P1
830 V2=-2.26000E-04*C*P2
840 V3=-2.26000E-04*C*P3
850 PRINT TAB(0);Z0(M4);TAB(5);T0(M4);TAB(14);P0(M3);
860 PRINT TAB(23);V;TAB(36);V1;TAB(45);V2;TAB(58);V3
870 NEXT M3
1000 END
```

APPENDIX (4)

```
01 REM SPLINE FIT INTERPOLATION METHOD IS USED TO
02 REM FIT CUBIC POLYNOMIAL BETWEEN EVERY TWO ADJACENT
03 REM TEMPERATURE INPUT DATA POINTS.
10 CALL "INIT"
30 DIM X(100),Y(100),D(100),P(100),E(100),C(4,100)
40 DIM P0(300),E0(300)
50 DIM A(100,3),B(100),Z(100)
51 PRINT \PRINT "DATA FILE NAME";\INPUT A$
52 OPEN A$ FOR INPUT AS FILE #1
53 INPUT #1:M\INPUT #1:R1
54 FOR I=1 TO M\INPUT #1:X(I)\NEXT I
55 FOR I=1 TO M\INPUT #1:Y(I)\NEXT I
56 CLOSE #1
60 CALL "SCAL"(0,0,40,1.5*R1)
70 OPEN "EAH101" FOR OUTPUT AS FILE VF2(10000)
80 B=3.49000E-03\P0=101300\G=9.81\C=3.00000E+08
105 PRINT \PRINT ,"TEMPRATURE INPUT DATA"\PRINT \PRINT
107 PRINT ,"Z(M)","T DEG. C"\PRINT \PRINT
108 FOR I=1 TO M\PRINT ,X(I),Y(I)\NEXT I
110 FOR I=1 TO M\CALL "APNT"(Y(I),X(I))\NEXT I
120 M1=M-1
122 PRINT \PRINT ,"INTERPOLATION OF TEMPRATURE DATA"
124 PRINT ,"USING SPLINE FIT METHOD"
125 PRINT,"Z(METERS)","T DEGREES C"
130 FOR K=1 TO M1
140 D(K)=X(K+1)-X(K)
150 P(K)=D(K)/6
160 E(K)=(Y(K+1)-Y(K))/D(K)
170 NEXT K
180 FOR K=2 TO M1
190 B(K)=E(K)-E(K-1)
200 NEXT K
210 A(1,2)=-1-D(1)/D(2)
220 A(1,3)=D(1)/D(2)
230 A(2,3)=P(2)-P(1)*A(1,3)
240 A(2,2)=2*(P(1)+P(2))-P(1)*A(1,2)
250 A(2,3)=A(2,3)/A(2,2)
260 B(2)=B(2)/A(2,2)
270 FOR K=3 TO M1
280 A(K,2)=2*(P(K-1)+P(K))-P(K-1)*A(K-1,3)
290 B(K)=B(K)-P(K-1)*B(K-1)
300 A(K,3)=P(K)/A(K,2)
310 B(K)=B(K)/A(K,2)
320 NEXT K
330 Q=D(M-2)/D(M-1)
340 A(M,1)=1+Q+A(M-2,3)
350 A(M,2)=-Q-A(M,1)*A(M-1,3)
360 B(M)=B(M-2)-A(M,1)*B(M-1)
370 Z(M)=B(M)/A(M,2)
380 M2=M-2
390 FOR I=1 TO M2
```

```
400 K=M-I
410 Z(K)=B(K)-A(K,3)*Z(K+1)
420 NEXT I
430 Z(1)=-A(1,2)*Z(2)-A(1,3)*Z(3)
440 FOR K=1 TO M1
450 Q=1/(6*D(K))
460 C(1,K)=Z(K)*Q
470 C(2,K)=Z(K+1)*Q
480 C(3,K)=Y(K)/D(K)-Z(K)*P(K)
490 C(4,K)=Y(K+1)/D(K)-Z(K+1)*P(K)
500 NEXT K
510 K9=1
520 FOR X=0 TO R1
530 IF X<X(1)GO TO 1150
540 IF X>X(1)GO TO 580
550 Y=Y(1)
560 CALL "APNT"(Y,X)
565 PRINT, X,Y
570 GO TO 760
580 K=1
590 IF X<X(K+1)GO TO 700
600 IF X>X(K+1)GO TO 670
610 Y=Y(K+1)
620 Y1=3*C(2,K)*D(K)^2-C(3,K)+C(4,K)
630 Y2=6*C(2,K)*D(K)
640 Y3=-6*C(1,K)+6*C(2,K)
650 CALL "APNT"(Y,X)
655 PRINT ,X,Y
660 GO TO 760
670 K=K+1
680 IF K<MGO TO 590
690 GO TO 1150
700 Y=(X(K+1)-X)*(C(1,K)*(X(K+1)-X)^2+C(3,K))
710 Y=Y+(X-X(K))*(C(2,K)*(X-X(K))^2+C(4,K))
720 Y1=-3*C(1,K)*(X(K+1)-X)^2+3*C(2,K)*(X-X(K))^2
730 Y1=Y1-C(3,K)+C(4,K)
740 Y2=6*(C(1,K)*(X(K+1)-X)+C(2,K)*(X-X(K)))
750 Y3=6*(-C(1,K)+C(2,K))\CALL "APNT"(Y,X)\PRINT,X,Y
760 REM K9=1 CORRESPONDS TO 0 METER
770 VF2(K9)=X\VF2(K9+2000)=Y\VF2(K9+4000)=Y1
780 VF2(K9+6000)=Y2\VF2(K9+8000)=Y3
790 K9=K9+1\NEXT X
800 T6=0
810 P0(1)=353.5/(VF2(2001)+273)
820 E0(1)=0
830 D2=X(M)
840 FOR I8=1 TO 300
850 D2=D2-4
860 IF D2<0GO TO 880
870 NEXT I8
880 M1=I8
890 FOR M3=2 TO M1
900 J1=(M3-2)*4+2000
```

```
910 T6=T6+.5/(VF2(J1+1)+273)+1/(VF2(J1+2)+273)
920 T6=T6+1/(VF2(J1+3)+273)+1/(VF2(J1+4)+273)
930 T6=T6+.5/(VF2(J1+5)+273)
940 E2=EXP(-.03424*T6)
950 M2=(M3-1)*4+2001
960 P0(M3)=(353.5/(VF2(M2)+273))*E2
970 E0(M3)=(M3-1)*4
980 IF E0(M3)>1024GO TO 1250
990 NEXT M3
995 PRINT TAB(0);"Z(M)";TAB(5);"T C";TAB(14);"DENS.";
997 PRINT TAB(23);"V";TAB(36);"V1";TAB(45);"V2";TAB(58);"V3"
1000 FOR M3=1 TO M1
1010 M4=4*M3-3+2000\L1=M4+2000\L2=M4+4000\L3=M4+6000
1020 P1=-P0(M3)/(VF2(M4)+273)*(VF2(L1)+G*B)
1030 P2=(-P1+P0(M3)*VF2(L1)/(VF2(M4)+273))*(VF2(L1)+G*B)/
     (VF2(M4)+273)
1040 P2=P2-P0(M3)*VF2(L2)/(VF2(M4)+273)
1050 P3=-(VF2(L1)+G*B)*P2/(VF2(M4)+273)
1060 P3=P3+(VF2(L1)+G*B)*P1*VF2(L1)/(VF2(M4)+273)^2
1070 P3=P3-P1*VF2(L2)/(VF2(M4)+273)
1080 P3=P3+(VF2(L1)+G*B)*(P0(M3)*VF2(L2)+VF2(L1)*P1)/
     (VF2(M4)+273)^2
1090 P3=P3-2*(VF2(L1)+G*B)*P0(M3)*VF2(L1)^2/(VF2(M4)+273)^3
1100 P3=P3-P0(M3)*VF2(L3)/(VF2(M4)+273)-VF2(L2)*P1/
     (VF2(M4)+273)
1110 P3=P3+P0(M3)*VF2(L1)*VF2(L2)/(VF2(M4)+273)^2
1120 V=C*(1-2.26000E-04*P0(M3))\V1=-2.26000E-04*C*P1
1130 V2=-2.26000E-04*C*P2\V3=-2.26000E-04*C*P3
1140 PRINT TAB(0);VF2(M4-2000);TAB(5);VF2(M4);TAB(14);P0(M3);
1141 PRINT TAB(23);V;TAB(36);V1;TAB(45);V2;TAB(58);V3
1142 NEXT M3
1145 GO TO 1250
1150 PRINT "OUT OF RANGE FOR INTERPOLATION"
1250 CLOSE \END
```

APPENDIX (5)

CREATION OF TEMPERATURE DATA FILES :

```
05 REM THIS PROGRAM IS TO CREATE DATA FILE FOR EA33 PROGRAM
10 PRINT "DATA FILE NAME IS";\INPUT A$
15 OPEN A$ FOR OUTPUT AS FILE #1
20 PRINT \PRINT "NUMBER OF DATA FOR TEMPRATURE PROFILE";
30 INPUT N1\PRINT #1:N1
40 PRINT \PRINT "MAXIMUM RAY ELEVATION FOR WHICH COMPUTATION
   WILL STOP";
50 INPUT R1\PRINT #1:R1
60 PRINT \PRINT "ENTER ALTITUDE DATA UP TO MAX. ALT.=";
   R1;"METERS"
70 FOR K2=1 TO N1\INPUT X\PRINT #1:X\NEXT K2
80 PRINT \PRINT "ENTER TEMPRATURE DATA IN DEGREES C."
90 FOR K2=1 TO N1\INPUT X\PRINT #1:X\NEXT K2
100 CLOSE #1\END
```

*

APPENDIX (6)

1. A User's Guide to EA33 and EA90

To use EA33 and EA90, follow these steps. Any error messages should be referred to the BASIC/RT-11 Language Reference Manual.

Start the computer according to the instructions attached to its front panel. Load System disk.

Type: (Using correct date):

DAT 15-Nov-77 (Hit CR after any typed line.)

Type: R BASGTX

Machine: * (Hit CR)

Machine: READY

Type: OLD "EA33"

Machine: READY

Type: RUN

Machine: EA33 15-Nov-77 BASIC VØ1-Ø5

NAME OF TEMPERATURE PROFILE DATA FILE?

Type: name of temperature profile data file (which is already created before) here

(e.g. EA500 or EA600 or EA700 or EA800)

Machine: it prints the temperature input data (altitude and temperature), also;

HORIZONTAL DISTANCE OF THE OBJECT H(KM) = ?

Type: horizontal distance between the object and the observer in km here

Machine: HORIZONTAL STEP SIZE (KM) = ?

Type: the horizontal step size Δu in km (e.g. 0.25 or 0.5 or 1.0) here

Machine: INITIAL ELEVATION ANGLE F(1)(DEGREES) = ?

Type: initial angle ϕ_o (e.g. -0.05°) here

Machine: It prints the data of transfer characteristic curve (ϕ_o° and ray elevation R meters) at this particular horizontal distance, and then:

SCALE RANGE (HORIZONTAL AXIS) ? 0,100

Type: left and right limits for drawing the object's picture e.g.

Machine: SCALE RANGE (VERTICAL AXIS) ? 0,50

Type: lower and upper limits, e.g.

Use Light pen control from here on:

Machine: HIT SUBPICTURE (VECTOR) FIRST BY LIGHT PEN

User: Draw the picture of any object on the screen by using light pen. Follow these instruction:

a) hit VECTOR command on the screen by light pen.

b) move the tracking object (diamond shape) from the origin to the next vertex of the object's picture.

c) hit VECTOR command by light pen.

d) move the tracking object again to the next vertex.

e) hit VECTOR again.

f) repeat the previous routine until finishing the drawing of object's picture on the screen.

g) hit SAVE command

h) notice that the machine prints the vertices coordinates in degrees.

Display Unit: it displays the image of the object as seen by the observer at H(Km.) horizontal distance.

Machine: STOP AT LINE 2715

READY

User: You can repeat the drawing of the object's picture if more improvement is required. In this case, type RUN and repeat the previous steps again. If you want to keep the picture:

Type: GO TO 2720

Machine: ** SAVE COMPLETED **

User: You should use previous routine twice to draw two more object's pictures, at different horizontal distances, on the screen. After drawing the third object's picture

Machine: STOP AT LINE 2715

Type: GO TO 2720

Machine: ** SAVE COMPLETED **

to clear the display file after representation of the images of three objects, you should

Type: CONTROL C

R DELTMP

R-BASGTX

2. A User's Guide to EA150 (creation of new data files)

To use EA150, start the computer according to the instructions attached to its front panel. Load system disk.

Type: (using correct date):

DAT 15-Nov-77 (hit CR after any typed line)

Type: R BASGTX

Machine: * (hit CR)

Machine: READY

Type: RUN

Machine: EA150 15-Nov-77 BASIC VØ1-Ø5

DATA FILE NAME IS ? EA900

Type: new data file name e.g., (or EA910,...)

Machine: NUMBER OF DATA FOR TEMPERATURE PROFILE?

Type: the number of pair points in the data here

Machine: MAXIMUM RAY ELEVATION FOR WHICH COMPUTATION WILL STOP ?

Type: max. elevation of interest in meters here

Machine: ENTER ALTITUDE DATA UP TO MAX. ALT. IN METERS

Type: The altitude data, one value per line until the machine prints.

Machine: ENTER TEMPERATURE DATA IN DEGREES C.

Type: the temperature data, one value per line until the machine prints.

Machine: READY