T-2047

INVERSION OF TDEM DATA

WITH

CATALOG INTERPOLATION

by

Juan C. Rodriguez-Ovejero

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 – 1346

T-2047

A thesis submitted to the Faculty and the board of Trustees of the Colorado School of Mines in partial fulfillment of the requirements for the degree of Master of Science, Geophysics.

Signed: J.C. Rodriguez-Ovejero

Golden, Colorado

Date: APRIL 21ST, 1978

Approved: George V. Keller
George V. Keller
Thesis Advisor

Golden, Colorado

Date: APRIL 21ST, 1978

Approved: George V. Keller
George V. Keller
Head of Department of Geophysics

Golden, Colorado

Date: APRIL 21ST, 1978

ABSTRACT

Near-zone transient electromagnetic apparent resistivity data is inverted in terms of a three-layer model (H type). Forward solutions are obtained using catalog lookup and interpolation, since forward calculations are expensive. Partial derivaties are obtained using finite differences. Model corrections are determined with a ridge regression estimator. Fastest possible convergence is forced at each iteration by calculating the squared error for several values of the ridge regression parameter, and choosing the value for which the error is the lowest.

The method is tested for both simulated and real data. Convergence takes an average of four iterations in both cases, and thus the method provides a fast and economical interpretation.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to thank my advisor Dr. G.V.Keller, for his encouragements and helpful suggestions in the completion of this thesis. I would like to acknowledge Dr. A.A.Kaufman for his help as a member of my graduate committee. Special thanks go to Dr. Charles Stoyer, whose continued insights, patience and dedication, made the completion of this thesis possible.

During the final stages of my research work, I encountered many difficult problems. I received invaluable help from my friends from many different countries. Since it is impossible to list all them, I just say: thank you Venezuela, Japan, USA, Mexico, South West Africa,... , and Casper.

Finally, I gratefully acknowledge the finantial assistance provided by the Direccion General de Energia (Spanish Government), through their scholarship.

INTRODUCTION

Time Domain Electro-Magnetic (TDEM) sounding is a group of methods which is capable of obtaining information about the electrical properties of the earth at depths of several kilometers or more. Some other electrical methods can be used to perform these soundings, for instance, the expanding array in the D.C. methods, and the Magneto Telluric methods, but it has been shown (Vanyan, 1967) that EM provide a better resolution than any other method.

A basic electromagnetic classification can be time domain, and frequency domain. With the frequency domain measurements, individual frequencies are transmitted and received. In time domain measurements, a step change in current is transmitted and a transient received. Depending on the separation between the transmitter and the receiver, the time domain methods can be classified into NZTDEM and LOTDEM. In the NZTDEM (Near Zone TDEM), the separation (R) between the transmitter and the receiver is considerably less than the thickness of the first layer (H1). In the LOTDEM (long TDEM), the separation R is larger than the thickness H1.

It is known that a single frequency has an effective depth of penetration (skin depth), thus measurements with

different frequencies must be done in order to perform the frequency sounding. The time spent in those soundings could be greatly reduced if many frequencies were recorded at the same time. It is known by the theory of the Fourier transform, that a step function has a large frequency content. Transient methods thus record the information that corresponds to a wide range of frequencies simultaneously. Then the sounding is performed in a very simple manner, when recording one transient.

The theory and interpretation methods for the frequency soundings have been extensively reported in the literature. The theory of the time domain method is reasonable well developed (Keller, 1977) but a great effort must still be done in order to obtain all the information that is contained in a recorded transient.

Several methods are normally used for the interpretation of the TDEM data: maximum voltage method, conventional graphical curve matching, and computer assisted techniques. A first step in the interpretation of the transient, is done by the maximum voltage method, which compares the characters of the response over the area surveyed. This comparison is done by plotting the values of the apparent resistivity which corresponds to the maximum voltage recorded in the transient. A more complete

interpretation can be made with the theoretical derived curves for a layered earth (conventional graphical curve matching). This is a traditional technique very popular in some of the other methods. In the case of TDEM only a few catalogs of curves are available, and in addition to this fact, these curves tend to lack the characteristic shapes that make them so useful for matching purposes. Automatic interpretation techniques are more convenient methods than graphical curve matching, if a computer of moderate capability is available (Keller,1977). Two types of computer-assisted interpretation techniques have been widely used: generalized linear inversion, and optimal simulation. These two methods perform a very accurate interpretation, but a large amount of computer time is required for the calculation of each single curve, and thus the cost of the process is very expensive for a commercial operation (Daniels, 1974).

An economical automatic method of interpretation is catalog interpolation. This method is developed in this work, using a set of theoretical layered-earth curves (NZTDEM) calculated by Kaufman (1970).

MAIN FEATURES OF BEHAVIOUR OF TDEM IN NEAR ZONE

This chapter contains a brief description of some important features of the behaviour of EM fields in the near-zone. The transient EM field is created by passing a step function through a horizontal loop. The electric field at an observation point will be studied in most of the cases, which corresponds the loop-wire array. Application of the reciprocity principle (Vanyan,1967) will make this configuration valid for the wire-loop array.

In the first part of this chapter, expressions for the field of a uniform whole-space are studied. These expressions will give an idea about the propagation of the EM field through the medium. Then the expressions of field in the uniform space are used to define the early and late time asymptotes. The resistivity formulas for the early and late time are obtained from the field expressions. Finally the ideas about propagation of the field are used to explain some important details of the apparent resistivity curves in a horizontal layered earth.

EM FIELD OF MAGNETIC DIPOLE IN UNIFORM SPACE

The uniform space is an ideal earth model where the electrical properties of the medium are homogeneous and isotropic. The electrical properties mentioned are: conductivity, magnetic susceptibility, and dielectric permeability. The description of the ideal model in this chapter is conducted towards an understanding of the propagation effects of the field through the medium.

At the time zero a step current passes through the source loop. Then three different stages (represented in figure 1) can be considered at the observation point (O.P.).

Stage 1 No signal.

The EM waves propagate through the medium with a certain velocity, this velocity of propagation will cause a lag in the arrival time of the EM waves. The time lag depends on the offset distance and some electrical properties of the material, but turns out to have a similar value in most of the cases (microseconds).

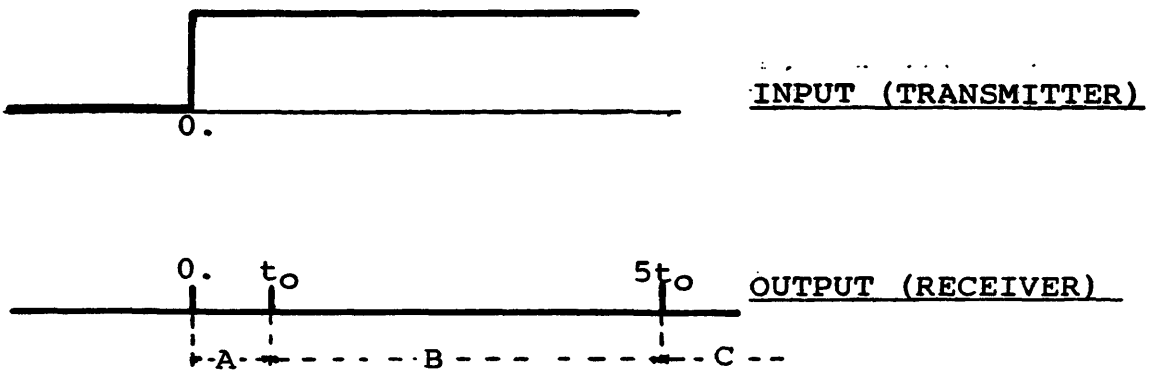
Stage 2 Important effect of displacement curents.

When the signal arrives at the O.P., there is an abrupt change in the electric field. At this time, the major role in the generation of the magnetic field is played by the displacement currents. After this abrupt change, there will be a period where the contribution of the displacement currents into the generation of the magnetic field is still important. The time period will be (approx.) four times the value of the arrival time.

Stage 3 Quasi stationary field.

The influence of the displacement currents outside the range of time mentioned in the previous stage will be very small, which corresponds to the quasi-stationary field situation. In this situation, the main contribution to the magnetic field comes from the conduction currents. Normally some milliseconds after the switch-on the field at the observation point is in this situation. Conduction currents are then the most important factor, in other words, the property that has the main influence in the quasi-stationary stage signal will be the resistivity of the medium.

The expression of the tangential component of the E-field in the quasi-stationary stage is represented in figure 2; two terms can be considered in this expression: geometric factor and parametric function. The parametric function (as it is described in Figure 2) gives an idea about some of the field properties. One of the properties that can be easily visualized is the propagation of the field, which is represented in figure 4 by means of a distribution of the eddy currents induced in the ground. Figure 4 represents the induction currents as a function of the distance, for different times. It can be observed that the maximum intensity of the induced currents is close to the source for small perids of time. As time increases, the maximum intensity decreases in absolute value and it is located farther away from the source. Two effects are thus combined in time: a displacement of the eddy current focus (location of maximum intensity) and a reduction of the maximum amplitude of these currents.



A. WAITING TIME (NO SIGNAL)

EM WAVES PROPAGATE WITH A CERTAIN VELOCITY V_1. THEN THERE IS NO SIGNAL UNTIL t_o.

$$V_1 = \frac{1}{(\varepsilon \mu)^{1/2}} \qquad t_o = \frac{R}{V_1} \simeq (\mu sec.)$$

B. IMPORTANT EFFECT OF DISPLACEMENT CURRENTS

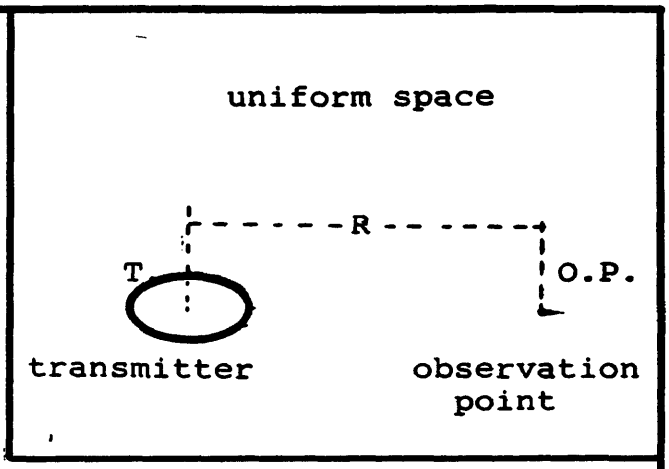
BECAUSE OF THE CHANGES IN THE ELECTRIC FIELD, DISPLACEMENT CURRENTS ARE IMPORTANT COMPARED WITH CONDUCTION CURRENTS.

C. QUASI-STATIONARY SITUATION.

THE DISPLACEMENT CURRENTS ARE SMALL AND CAN BE NEGLECTED. MAIN CONTRIBUTION TO THE MAGNETIC FIELD COMES FROM THE CONDUCTION CURRENTS (RESISTIVITY).

FIGURE 1 EFFECTS AT THE OBSERVATION POINT DUE TO THE EXCITATION OF THE TRASMITTER.

QUASI-STATIONARY E-FIELD OF MAGNETIC DIPOLE IN UNIFORM SPACE.



$$E_\theta = \frac{2}{\pi} \cdot \frac{M \cdot \rho}{2 \cdot \pi \cdot R^4} \quad u^5 \cdot e^{-u^2/2}$$

FOR

$$u = \frac{2 \cdot \pi \cdot R}{T} \qquad T = (2 \cdot \pi \cdot \rho \cdot t \cdot 10^{-7})^{1/2}$$

WHERE

ρ	.....	RESISTIVITY OF THE MEDIUM
M	.....	MAGNETIC MOMENT OF TRANSMITTER
R	.....	OFFSET DISTANCE (T.-O.P.)
t	.....	TIME (sec.)
u	.....	NONDIMENSIONAL PARAMETER
T	.....	TRANSIENT PARAMETER

THE EXPRESSION CONTAINS:

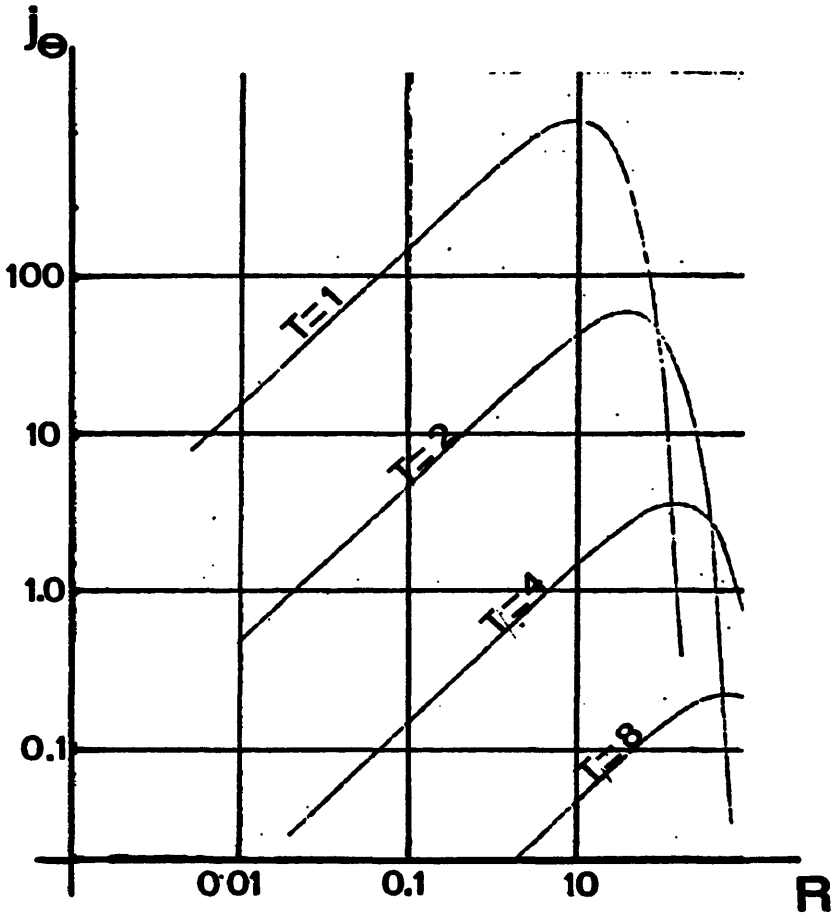
SCALING FACTOR WITH PHYSICAL MEANING OF THE FIELD AND ITS UNITS.

$$K_s = \frac{2}{\pi} \cdot \frac{M \cdot \rho}{2 \cdot \pi \cdot R^5}$$

PARAMETRIC FUNCTION DESCRIBES THE FIELD PROPERTIES AND IT HAS NO DIMENSION.

$$e = u^5 \cdot e^{-u^2/2}$$

FIGURE 2 QUASI-STATIONARY E-FIELD OF MAGNETIC DIPOLE IN UNIFORM SPACE.



$$T = \sqrt{2\pi \cdot \rho \cdot t \cdot 10^{+7}}$$

(AFTER KAUFMAN,1977)

FIGURE 3. PROPAGATION OF THE CURRENTS AS FUNCTION OF THE DISTANCE.

FIELD IN THE UNIFORM SPACE. EARLY AND LATE TIME ASYMPTOTES. APPARENT RESISTIVITY FORMULATIONS.

The half space corresponds to the model created by an interface between free-space (perfect insulator) and a homogeneous and isotropic conductive medium. This model is very useful in electrical prospecting because its similarities with the air-earth interface, where most of the EM data is collected.

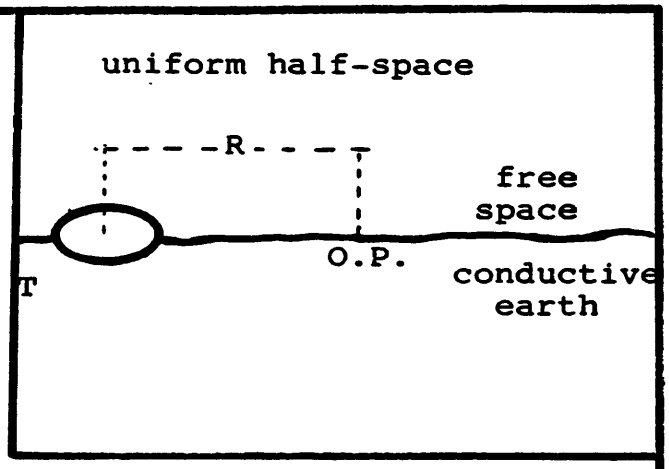
The expression of the tangential component of the eletric field in the uniform space, is shown in figure 4. The expression has a structure similar to the formulas in figure 2, containing a scaling factor and a parametric function. The scaling factor has in itself the physical meaning of the field and its units. The parametric function contains information about the changes in the field, as a function of the nondimensional parameter u.

Two main asymptotic expressions can be defined from the formulas in figure 4: the early stage and the late stage. The early stage, also called the wave zone, is defined for values of u which are significantly greater than one. It is obtained by taking limits of the expression in figure 4 when u goes to infinity. In a similar way, the late stage is defined for small values of

u, and its expression is obtained by taking limits when u goes to zero.

The concepts of early and late time asymptotes are relative concepts in time. Their location depends on the value of the offset distance R, which can be observed from the definition of the parameter u. It is seen that distance (R) and time (t), play opposite roles when the limits are taken. Then in order to locate one of the two asymptotes in time, the offset distance must be taken into account.

The expression for the early stage asymptote is represented in figure 5 for the tangential component of the electric field. From this expression it is possible to obtain a formula for the apparent resistivity in the early stage of the transient. The derivation of this formula is represented in the same figure. The value of the apparent resistivity does not have a physical meaning, but it gives information about how much the measured field differs from uniform earth. A similar expression for the late stage asymptote of the tangential E-field is in figure 6. This figure also contains a formula for apparent resistivity, valid for the late stage.



TANGENTIAL COMPONENT OF THE E-FIELD IN A UNIFORM HALF-SPACE.

THE EXPRESSION IS:

$$F_\theta = \frac{3 \cdot M \cdot \rho}{2 \cdot \pi \cdot R^4} \cdot e_\theta$$

WHERE

$$e_\theta = \phi(u) - \left(\frac{2}{\pi}\right)^{1/2} \cdot u \cdot \left(1 + \frac{u^2}{3}\right) \cdot e^{-u^2/2}$$

AND

$\phi(u)$ IS THE PROVABILITY INTEGRAL,
AND THE OTHER PARAMETERS WERE DEFINED IN FIGURE 2

THIS EXPRESSION CONTAINS (SAME AS FIGURE 2)

SCALING FACTOR

$$K_s = \frac{3 \cdot M \cdot \rho}{2 \cdot \pi \cdot R^4}$$

AND PARAMETRIC FUNCTION (e_θ)

FIGURE 4 EXPRESSION OF THE TANGENTIAL COMPONENT OF THE E-FIELD IN A UNIFORM HALF-SPACE.

EARLY STAGE APPARENT RESISTIVITY.

THE EARLY STAGE ASYMPTOTE IS OBTAINED BY TAKING LIMITS OF THE EXPRESSION OF THE FIELD IN A UNIFORM HALF SPACE (FIGURE 4), WHEN u GOES TO INFINITY.

$$E_\theta = \frac{3 \cdot M \cdot \rho}{4 \cdot \pi \cdot R^4}$$

IN A UNIFORM EARTH, THE VALUE OF THE RESISTIVITY OBTAINED FROM THIS FORMULA WILL BE:

$$\rho = \frac{4 \cdot \pi \cdot R^4}{3 \cdot M} \; E_\theta^{unif.}$$

WHEN THE EARTH IS NOT UNIFORM, THE RESISTIVITY IS:

$$\rho_a = \frac{4 \cdot \pi \cdot R^4}{3 \cdot M} \; E_\theta$$

WHICH CORRESPONDS TO THE EARLY STAGE FORMULATION OF THE APPARENT RESISTIVITY. THIS RESITIVITY, DOES NOT HAVE A PHYSICAL MEANING.

BY DIVIDING THE TWO PREVIOUS EXPRESSIONS,

$$\frac{\rho_a}{\rho} = \frac{E_\theta}{E_\theta^{unif.}}$$

IT CAN BE SEEN, THAT THE APPARENT RESISTIVITY TELLS HOW MUCH THE FIELD MEASURED IN THE REAL EARTH DIFFERS FROM THE FIELD IN A UNIFORM EARTH.

FIGURE 5 EARLY STAGE APPARENT RESISTIVITY.

LATE STAGE APPARENT RESISTIVITY

THE LATE STAGE ASYMPTOTE IS OBTAINED BY TAKING LIMITS OF THE EXPRESSION OF THE FIELD IN A UNIFORM HALF-SPACE WHEN u GOES TO ZERO.

$$E_\theta = \frac{\mu^{5/2} \cdot R \cdot M}{40 \cdot \rho^{3/2} \cdot \pi^{3/2} \cdot t^{5/2}}$$

IN A SIMILAR WAY AS WAS DONE WITH THE EARLY TIME FORMULATION (FIGURE 5), THE APPARENT RESISTIVITY FORMULA WILL BE:

$$\rho_a = \frac{\mu^{5/3}}{\pi \cdot t^{5/3}} \cdot \left(\frac{M \cdot R}{40} \right)^{2/3} \cdot E_\theta^{-2/3}$$

AND COMPARE WITH THE APPARENT RESISTIVITY FOR THE UNIFORM EARTH;

$$\frac{\rho_a}{\rho} = \left(\frac{E_\theta}{E_\theta^{unif.}} \right)^{-2/3}$$

WHICH MEANS THAT THE APPARENT RESISTIVITY TELLS HOW MUCH THE FIELD MEASURED IN A REAL EARTH DIFFERS FROM THE FIELD IN A UNIFORM EARTH.

FIGURE 6 LATE STAGE APPARENT RESISTIVITY.

APPARENT RESISTIVITY CURVES IN HORIZONTAL LAYERED EARTH

In previous paragraphs some important features about the behaviour of the transient EM field in the near-zone were discussed. These features will be the key for the following description of the apparent resistivity curves in the uniform earth.

The set of theoretical apparent resistivity curves were calculated some years ago (1969) in Russia by A.A.Kaufman. For this purpose, the frequency range of integration was divided in three parts: low frequency, intermediate frequency, and high frequency. Later, a specific method of calculation was applied in each of those intervals. With the ideas about propagation of the field in mind, it is possible to explain some important details of these theoretical curves. These details will be described for the two layered earth model in the cases of resistive and conductive basement. Since these two models will characterize every geoelectrical situation, a similar approach can be extended to a three layer case.

Model 1 two layered earth with resistant basement

Figure 7 shows a set of master curves corresponding to this model. The curves are plotted as a function of the apparent resistivity versus time (normalized). Each curve

corresponds to a model with a different second layer resistivity. In order to make the catalog general (independent of the first layer), the parameters have been normalized.

Each of these curves, which characterizes a geoelectrical situation, can be divided into four different parts (A, B, C, D) (figure 8).

Part A.- very early stage of the transient.

This part describes the resistivity in the wave zone using the formulation of the last stage. It can be observed for very small times.

Part B.- First layer asymptote.

The induced currents are confined in the upper layer at this time, then the information obtained at the observation point comes only from the first layer. It can be said that one can see only the first layer from an apparent resistivity point of view.

Part C.- Penetration into the second layer.

The propagation of the eddy currents can be

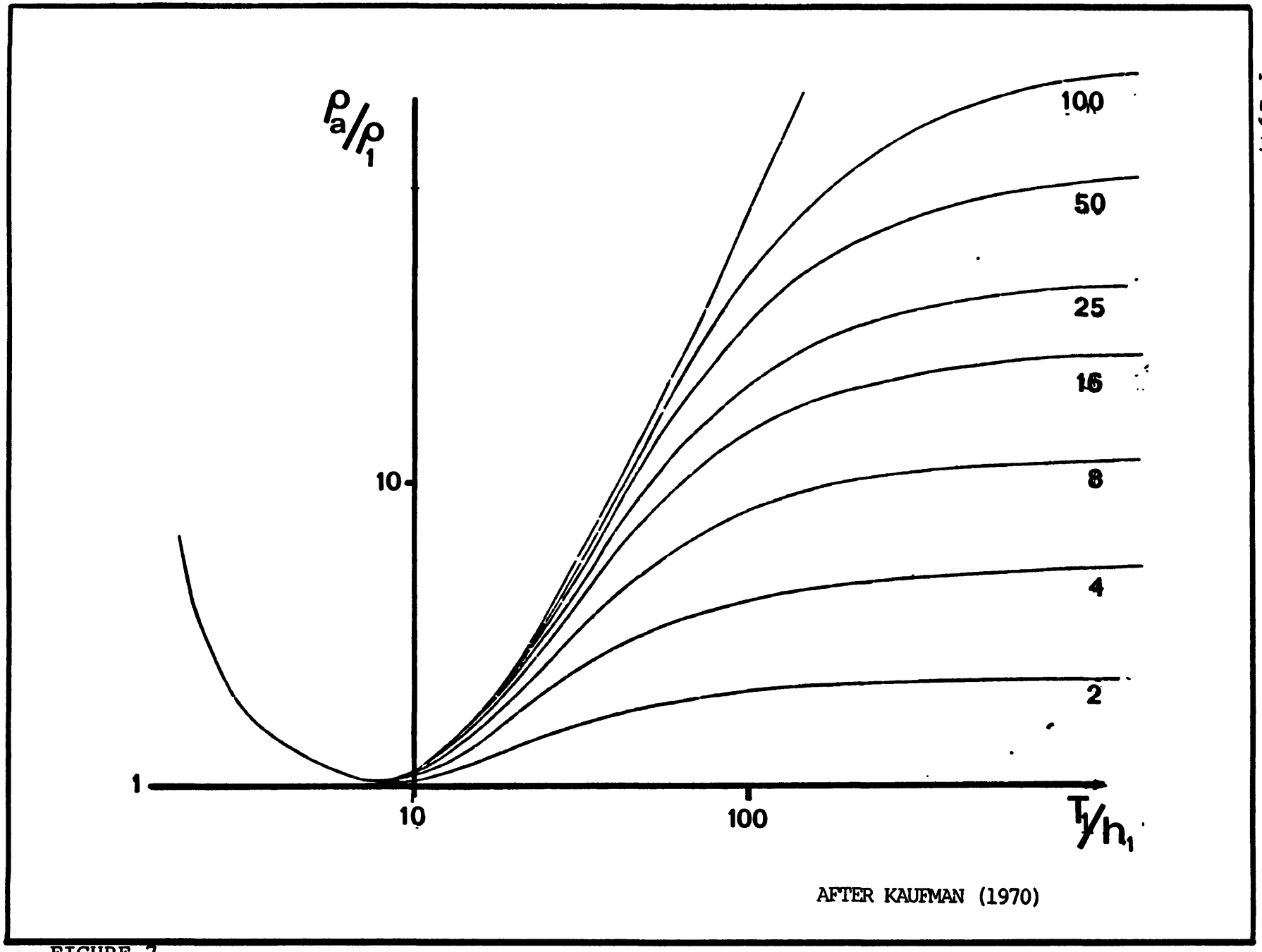


FIGURE 7

TWO-LAYER APPARENT RESISTIVITY CURVES (NEAR-ZONE) FOR RESISTANT BASEMENT.

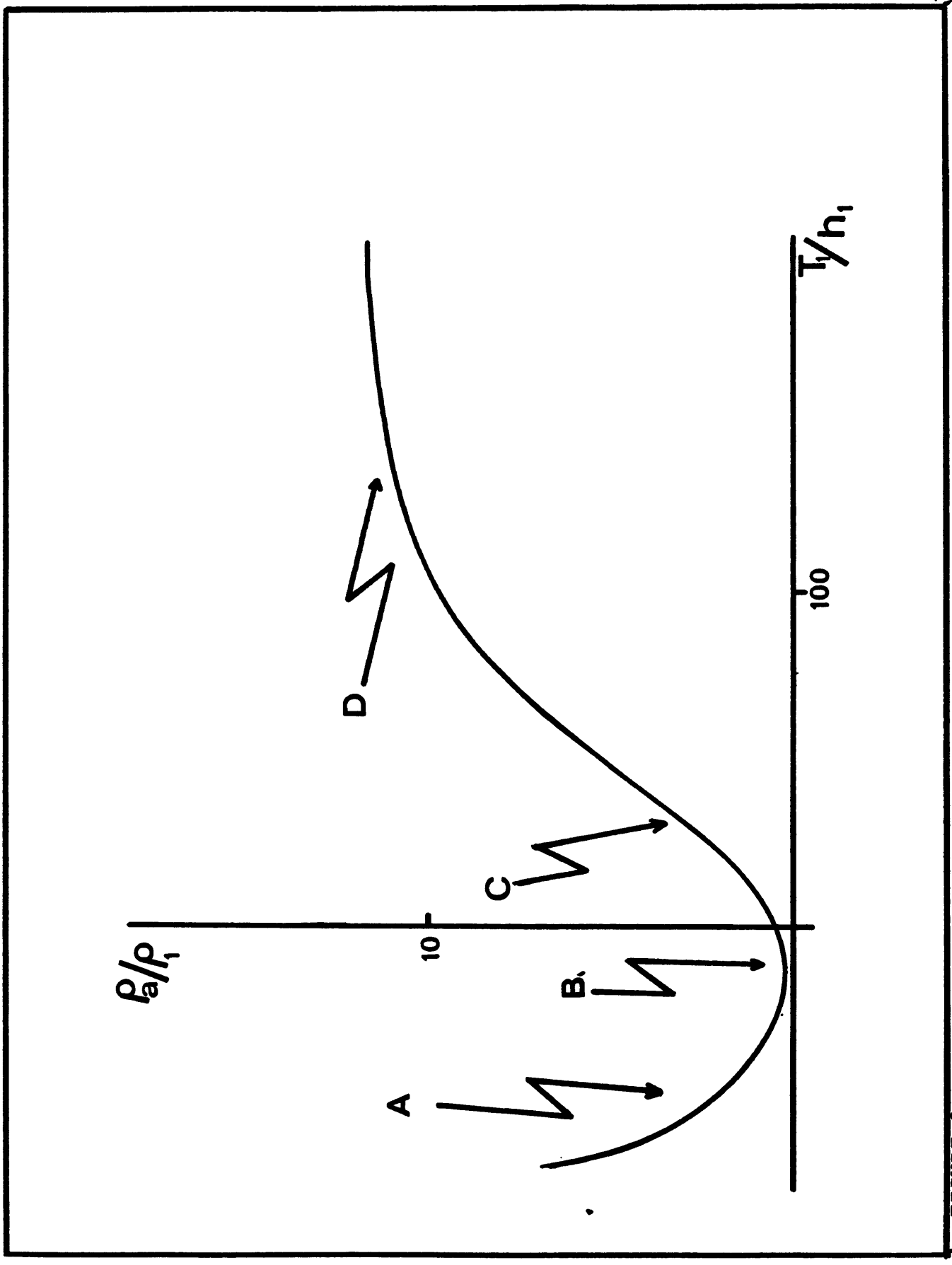


FIGURE 8 DIFFERENT INTERVALS IN A TWO-LAYER APPARENT RESISTIVITY CURVE WITH CONDUCTIVE BASEMENT.

simplified with the idea of "down and away". This idea combines two effects: a deeper penetration on time with a spreading effect in each single layer.

At this time, the currents are beginning to penetrate in the second layer, the magnetic field induced by these currents still has a low density. The low density makes the second layer appear to be an insulator from an apparent resistivity point of view, with a slope of approximately -45 degrees in this part of the curve.

Part D.- Currents located mostly in second layer.

After some time, the maximum intensity of induced currents is located in the second layer. The curve at this time has an asymptote that is similar to the response of a uniform half space with the resistivity of the second layer.

Model 2 <u>two layered earth model with conductive basement</u>

Figure 9 shows another set of apparent resistivity curves, for the case of a conductive basement. In the same way as it was done before, it is possible to divide each

curve in several parts, depending on the behaviour of the field (A, B, B', C, D) (figure 10).

Parts A and B.- These zones, depend only on the first layer resistivity, the explanation on zones A and B of the previous model is valid here.

Part B'.- maximum of the curve.

The eddy currents begin to propagate through the second layer. At the begining, the density of the currents in the second layer is small, and the conductor exerts little influence in the field. Then the value of the apparent resistivity in this interval increases to a maximum and then decreases.

Part C.- Since the resistivity of the second layer is small enough, the second layer looks like a perfect conductor during this period. The slope of the curve is -45 degrees. The effect is similar to the zone C of the first model, where there was a resistor instead of a conductor.

Part D.- Most of the currents are located in the second layer. The asymptote is similar to the last one in the first model.

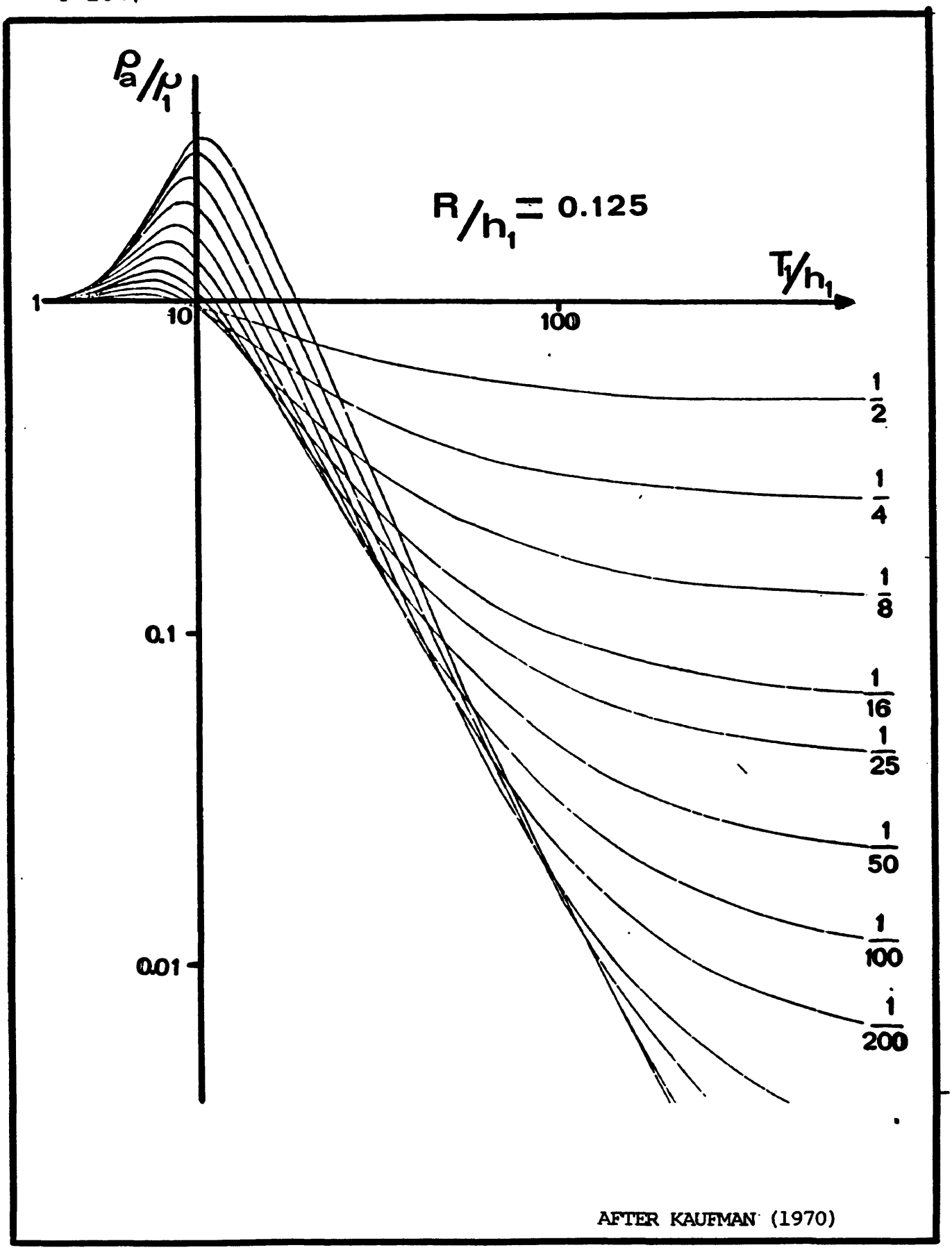


FIGURE 9 TWO-LAYER APPARENT RESISTIVITY CURVES (NEAR-ZONE) FOR CONDUCTIVE BASEMENT.

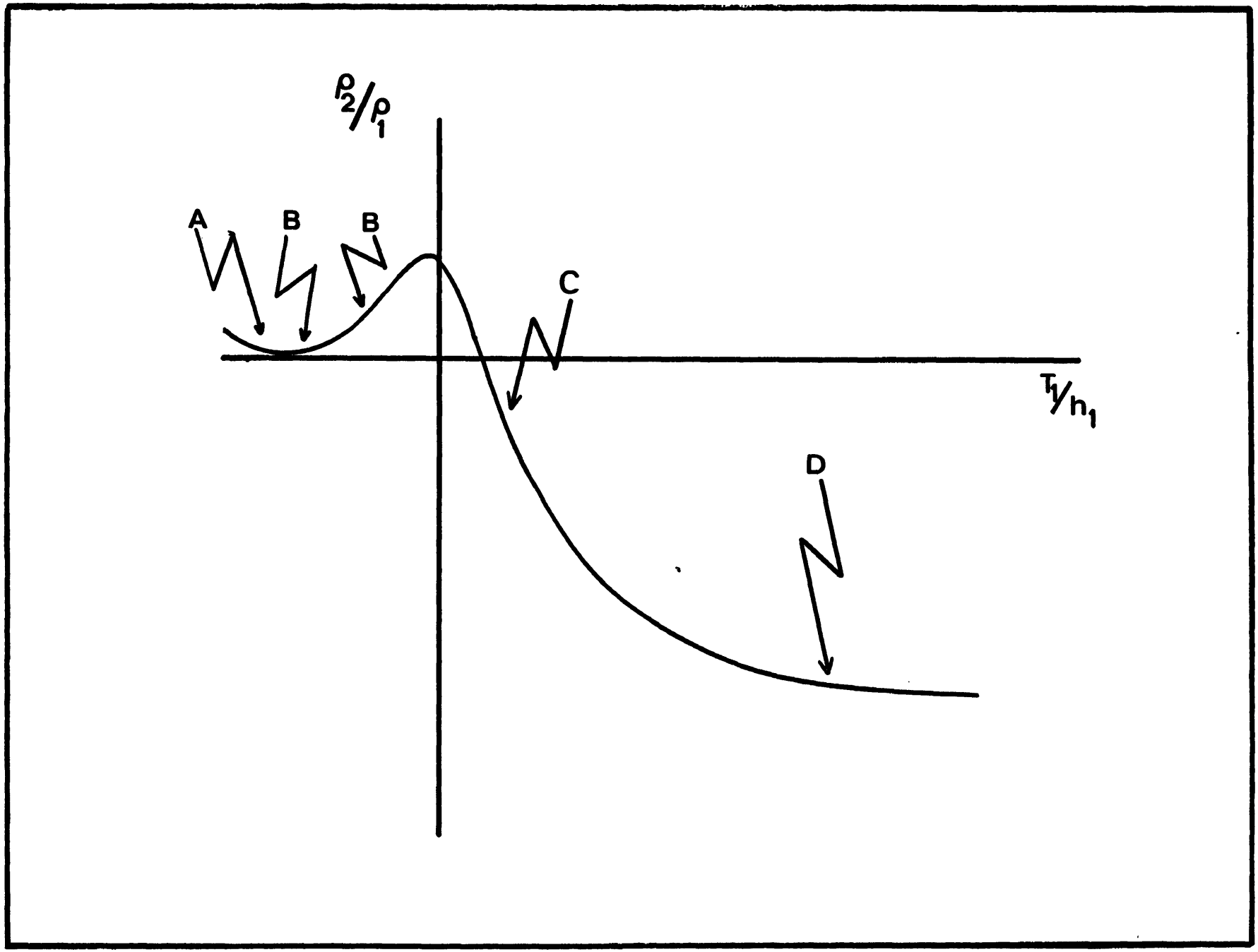


FIGURE 10 DIFFERENT INTERVALS IN A TWO-LAYER APPARENT RESISTIVITY CURVE WITH RESISTIVE BASEMENT.

These two models describe the behaviour of the field for the two possible types of resistsivity contrasts. The ideas of EM wave propagation for more complex geoelectrical sections can be given by an extension of those described above.

In this chapter some of the main features of the behaviour of TDEM in the near zone have been described. The ideas about propagation of the field were shown using the model of a uniform earth. Expression of the field in the uniform half space was used to describe the early and late time asymptotes. Apparent resistivity formulations for those asymptotes were then calculated. Finally, the apparent resistivity curves (late time) in a horizontally layered earth were explained using the model of propagation previously mentioned. A three-layer earth model of these apparent resistivity curves will be used in this work to develop an automatic method of interpretation using the theory of inversion, which is going to be described in the next chapter.

INVERSION THEORY IN GEOPHYSICS

During the last ten years, the improvement in computer facilities has made automatic interpretation methods a popular area of research in most branches of geophysical prospecting. These automatic methods have many advantages compared with the traditional interpretation approach (Keller,1977). It can be said that automatic interpretation represents one of the ultimate goals in most of the geophysical methods.

Geophysical inverse theory is the subject where most of these methods are studied. The theory deals with the mathematical aspects of the interpretation of the indirect measurements collected by the geophysical methods. The starting point for a possible application of this theory, is the existence of a valid mathematical model of the physics, this mathematical model is called the forward solution (FS) of the problem. A class of models is defined, and a particular earth model will be given by values of certain parameters. The calculated values represent the theoretical response for a particular idealized earth model, this response is obtained from the F.S., for the parameters.

The forward solution can be simply represented by a nonlinear functional F that relates the synthetic observations, also called calculated values ($\underline{c}$), with its corresponding vector of parameters ($\underline{p}$);

$$Ci = Fi(P1,P2,..) \qquad (III,1)$$

where

$\bar{p}$ ..... unknown parameters associated with mathematical model

$\bar{c}$ ..... calculated values for a certain set of parameters

This functional F is not a simple mathematical function, but it represents the fact that the forward problem is completely solved. In other words, when a particular value of the parameter vector is given, one can find the appropiate data associated with it.

In a similar manner, the inverse problem deals with estimation of the parameters which correspond to a certain group of observations. The problem can be simply defined as:

> Given the observations obtained with some geophysical method, find as much information as possible about the earth model

parameters.

Figure 11, shows a schematic relation between forward solution and inverse problem. The forward solution relates the parameters to calculated values by a mathematical model. Inverse problem relates the observations (o) to the parameter model using the theory that will be described in the next pages. Both formulations relate similarly to the vectors (observations and parameters) but their action has a different direction.

The following pages contain a brief description of the different steps used to find a solution for the inverse problem. Generalized inverse theory (least-square method) gives a possible value of the solution, but instabilities due to the "ill conditioned" data make this a non-realistic solution. Some linear constraining methods must be used to avoid these problems. The ridge regression estimator uses a feedback principle to ensure that the model obtained in the calculation is close to the possible optimum for that method. The new model obtained with one of the calculations can be improved, if another iterative calculation is performed. Several iterations are needed when using ridge regression until a stable final solution is obtained, this stable model is then considered to be the solution of the inversion.

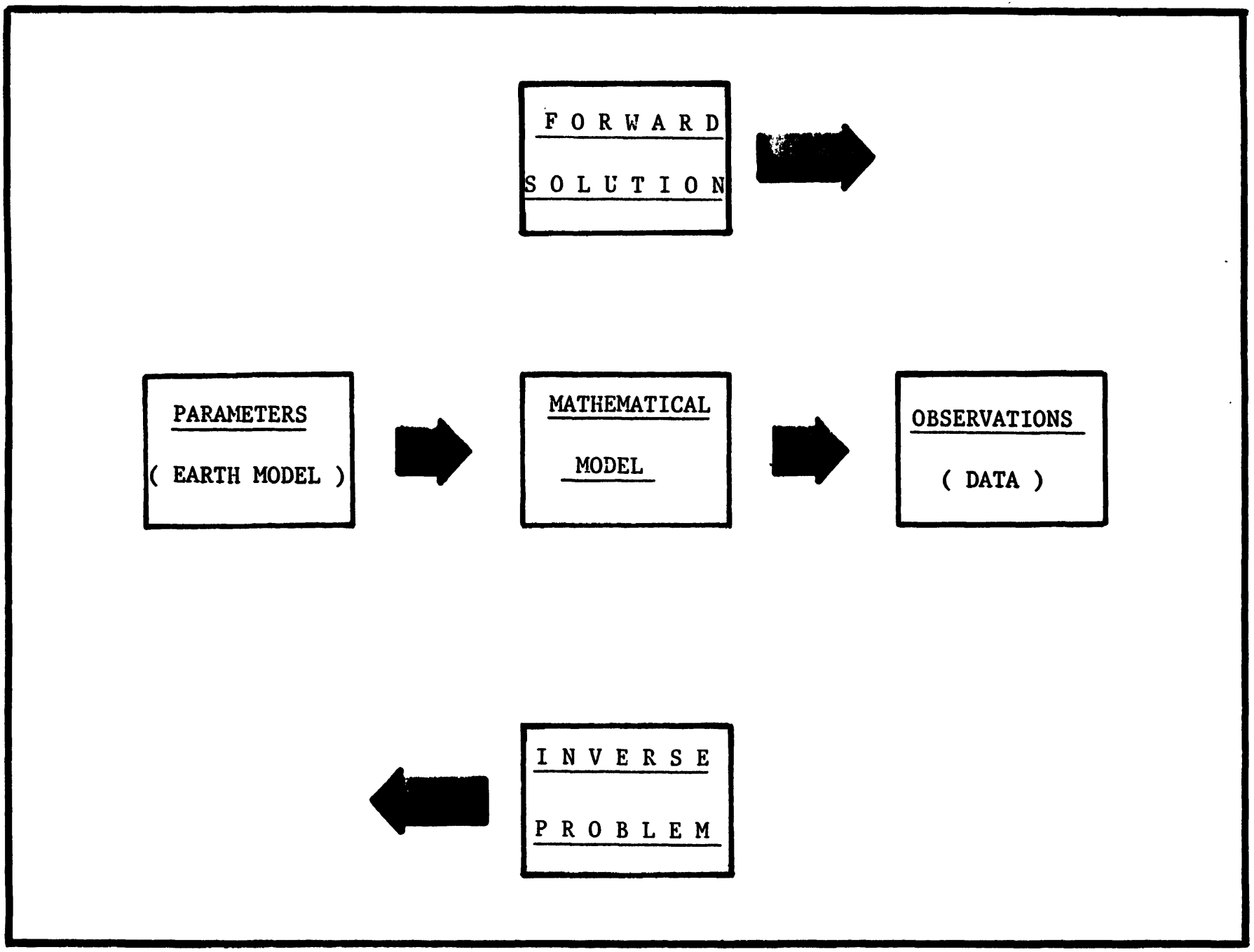


FIGURE 11 SCHEMATIC RELATION BETWEEN FORWARD SOLUTION AND INVERSE PROBLEM

Some ideas about the cost of the final model for a typical inversion are included in the last part of this chapter. The cost turns out to be very high in some cases, which is the reason why catalog interpolation was designed to replace an extremely expensive calculation with an extremely inexpensive one.

Inverse Solution

Assuming that the forward solution is represented by a non-linear functional F, as it was mentioned previously [3,1], the inversion methods try to find out as much information as possible about the M parameters which define the earth model when a set of N observations (data) is given. The first step in order to obtain the inverse solution, will be an expression given by application of the generalized inverse theory (least-square solution).

Given a set of observations ($\underline{o}$) and an estimation (or initial guess) of the earth model in terms of the vector parameter $\underline{p}$, the approximation of the estimation to the real model can be evaluated in terms of an error function E;

$$E = \sum_{1}^{N} (O_i - C_i)^2 \qquad \text{(III,2)}$$

where

Oi ..... observations (real data)

Ci ..... calculated values for the parameter guess, Po.

N ..... number of observations

The best solution for the inversion, in the least-square sense, can be obtained when the error function E is minimized. The result of the minimization is an overdetermined system of N equations with M unknowns. The unknowns in this system of equations are the optimum increment (correction) of the parameter vector estimation. Using the matrix formulation, this system can be written by a simple expression that relates the unknown increment of parameters with the set of observations (Inman, 1973).

$$\bar{D} = \underline{A} \cdot \Delta \bar{P} \qquad \text{(III,3)}$$

$$\bar{D} = \bar{o} - \bar{c}$$

where

$\bar{c}$ ..... calculated vector (observations for the initial model)

$\bar{o}$ ..... Observation vector

$\underline{A}$ ..... Derivative Matrix

The derivative matrix contains the coefficients of the overdetermined system of equations. These coefficients are the derivatives of the functional with respect to the parameters. A general term in the matrix can be defined as:

$$a_{ij} = \left(\frac{\delta Fi}{\delta p_j}\right)_{\overline{P}_o} \qquad \text{(III,4)}$$

where

a_{ij} ..... (i,j) element of A matrix

Fi ..... Functional

Po ..... Initial guess model

The partial derivatives of any functional can be calculated with a sufficient degree of accuracy using the finite difference formulation (F.D.), the expression of this F.D. derivative is:

$$a_{ij} = \frac{Fi(p_1,\ldots,p_j+ pj,..) - Fi(p1,\ldots,pj,..)}{\Delta pj} \qquad \text{(III,5)}$$

Once the matrices $\underline{A}$ and $\underline{d}$ are calculated, a solution of the system (III,3) is needed to perform the inversion. In order to find this solution, an inverse of the nonsquare matrix A is needed. The singular value decomposition method (S.V.D.) (Lanczos,1961) is used to compute this inverse. With this technique, the matrix A can be written as:

$$\underline{A} = \underline{U} \quad \underline{A}^{-1} \quad \underline{V}^t \qquad \text{(III,6.1)}$$

where

$$\underline{U} = \underline{A} \cdot \underline{V} \cdot \underline{\Lambda}^{-1} \qquad \text{(III,6.2)}$$

and defining ATA as the premultiplication of A by its transpose,

$\underline{V}$ ..... orthonormalized eigenvectors of ATA. (eigenvectors row of ATA)

$\underline{\Lambda}$ ..... Diagonal matrix containing the nonnegative square roots of the eigenvalues of ATA.

The generalized inverse of A can be easily calculated from the decomposed matrix (III,1).

$$'A^{-1}' = \underline{V}\ \underline{\Lambda}^{-1}\ \underline{U}^{t} \qquad \text{(III,7)}$$

Finally, the solution given by the least-square method turns out to be:

$$\overline{P} = \underline{V} \cdot \underline{\Lambda}^{-1} \cdot \underline{U}^{t} \cdot \overline{D} \qquad \text{(III,8)}$$

The solution can be written by a more simple expression

(Glen et al, 1972)

$$\Delta\overline{P} = (\underline{A}^{t} \cdot \underline{A})^{-1} \cdot \underline{A}^{t} \cdot \overline{D} \qquad \text{(III,9)}$$

In a few cases, the above expression gives an adequate stable solution for the generalized inverse problem. The main causes of this instability in the solution can be found in some assumption made during the computation of the final formulations. Principal assumptions made that cause problems in the inversion are:

I.- Linearity of forward solution,when computing [3,22]. In the method of minimization, it was assumed that F.S. would have an accurate representation with the first two terms of the Taylor expansion of the functional. However, it does not.

II.- Absence of noise in the data (observation). The observations were considered perfect, i.e., with total absence of noise. Noise, of course is unavoidable.

In most of the real problems of inversion these two assumptions are not completly valid, hence there will be

some noise in the system. Consequently the final solution obtained with these assumptions has a high content of noise, and will be unstable in most cases.

The root cause for the inefectiveness of the method, is the existence of small eigenvalues. These small eigenvalues, represent parameter combinations to which the data are insensitive. But in the inverse problem, the parameters appear to be extremely sensitive to parameter combinations. Because of this sensitivity, the effect of noise is increased in an "ill conditioned system". The final result will have a high percentage of noise, and thus invalid results in most of the cases.

Since the root cause of these invalid results is the existence of small eigenvalues, the problem can be solved in two different ways.

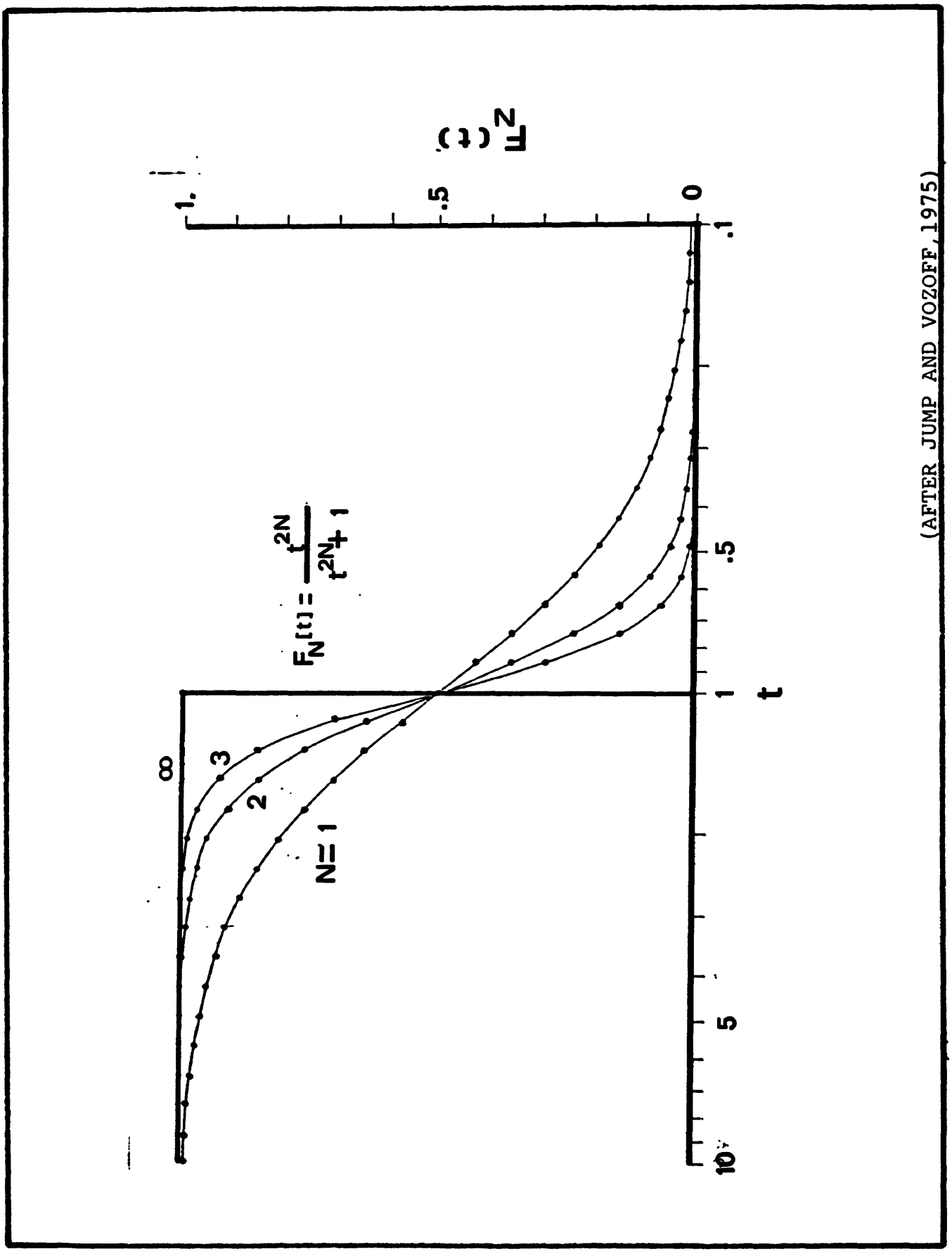
1.- Truncate some of the eigenvalues

When truncating some of the eigenvalues, noise is not a problem anymore, but the information contained in those important and sensitive values, is discarded, this decrement in the information content, will slow down the speed of convergence and sometimes will prevent it.

2.- Increase the magnitude of the smaller eigenvalues

The increment will reduce the degree of sensitivity of those small eigenvalues, this method will attack the problem of noise without decreasing abruptly the information content in the matrix.

The second method appears to be the most adequate to solve the problems caused by an "ill conditioned" system. The method can be performed by application of some attenuation factors to the diagonal matrix of the eigenvalues. Figure 12 shows a family of curves that represent these factors; It can be observed that these factors act like smoothing operators. Its effect is almost null for the low order eigenvalues (large values), and it will affect strongly the higher order eigenvalues (small values), with values of the factor close to zero. Consequently, this strong effect will reduce the sensitivity of the system to the information associated with the very small eigenvalues (noise). Due to this decrement of the sensitivity, the noise is not a problem anymore, and a realistic solution can be obtained by applying these constraining methods to the algorithm of inversion. Feedback is necessary to ensure that the proper



(AFTER JUMP AND VOZOFF, 1975)

FIGURE 12 ATTENUATION FACTORS.

decrement of the sensitivity is chosen.

The process of inversion just described (from generalized inversion to the different constraining methods), can be simply described with an analogy between linear systems and inversion theory (Stoyer, personal communication, 1977). The basic idea of this analogy is the similar concepts used in filter design theory, and inverse solution deduction.

The first idea in order to reverse the polarity of the system would be the Perfect Inverse filter, this filter works well in ideal conditions (figure 13), but most of the inversions are 'ill-conditioned', and additive noise must be taken into account. Perfect inverse does not work properly in a noisy system, and the s/n (signal to noise ratio) is very small in the output (Figure 14). Then an additional filtering effect is needed to obtain the optimum filter in real situations. The filter is in tandem with the perfect inverse, and will give a stable solution in the output even with the presence of high noise content in the system (Figure 14)

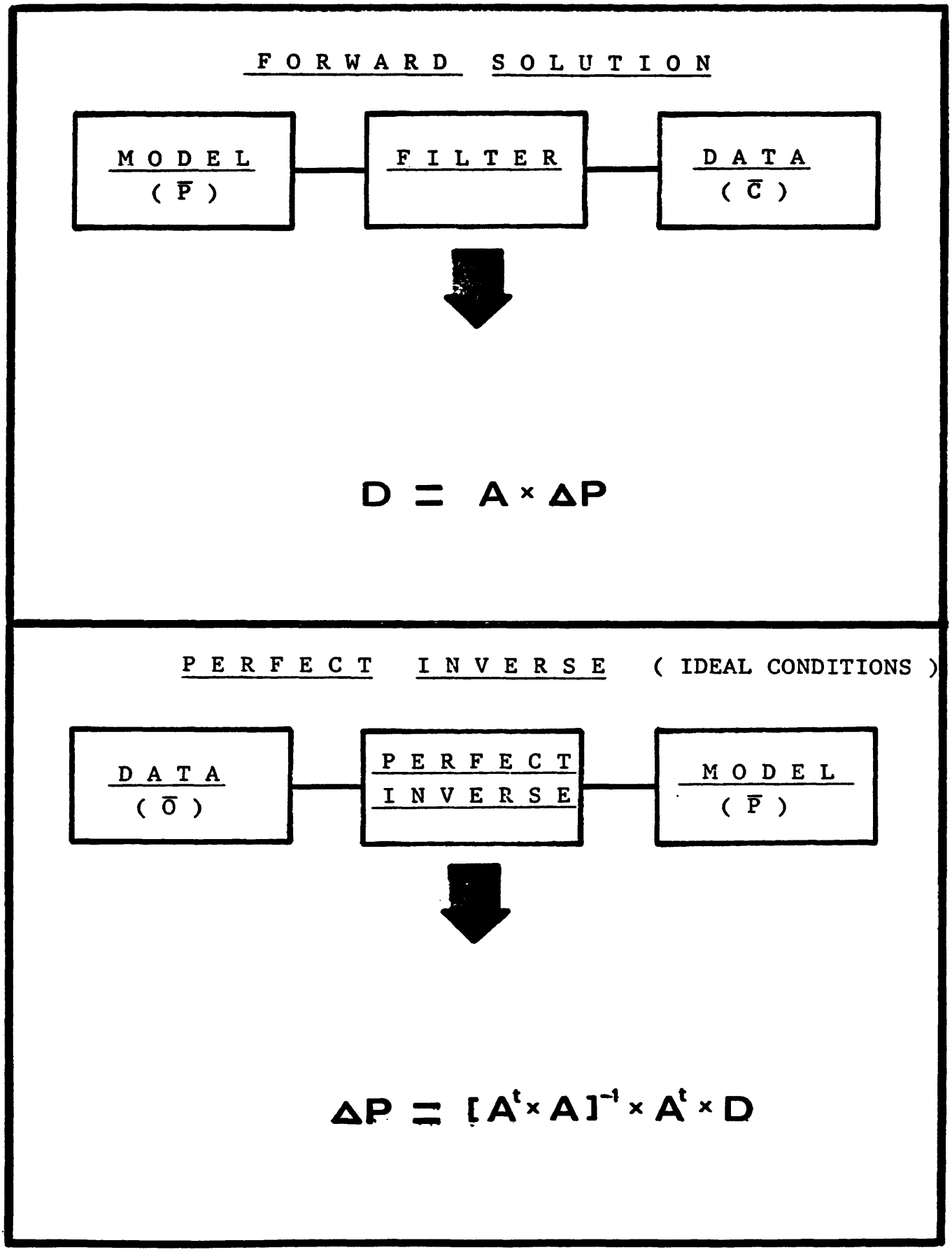


FIGURE 13 FILTERING REPRESENTATION OF INVERSE PROBLEM (I)

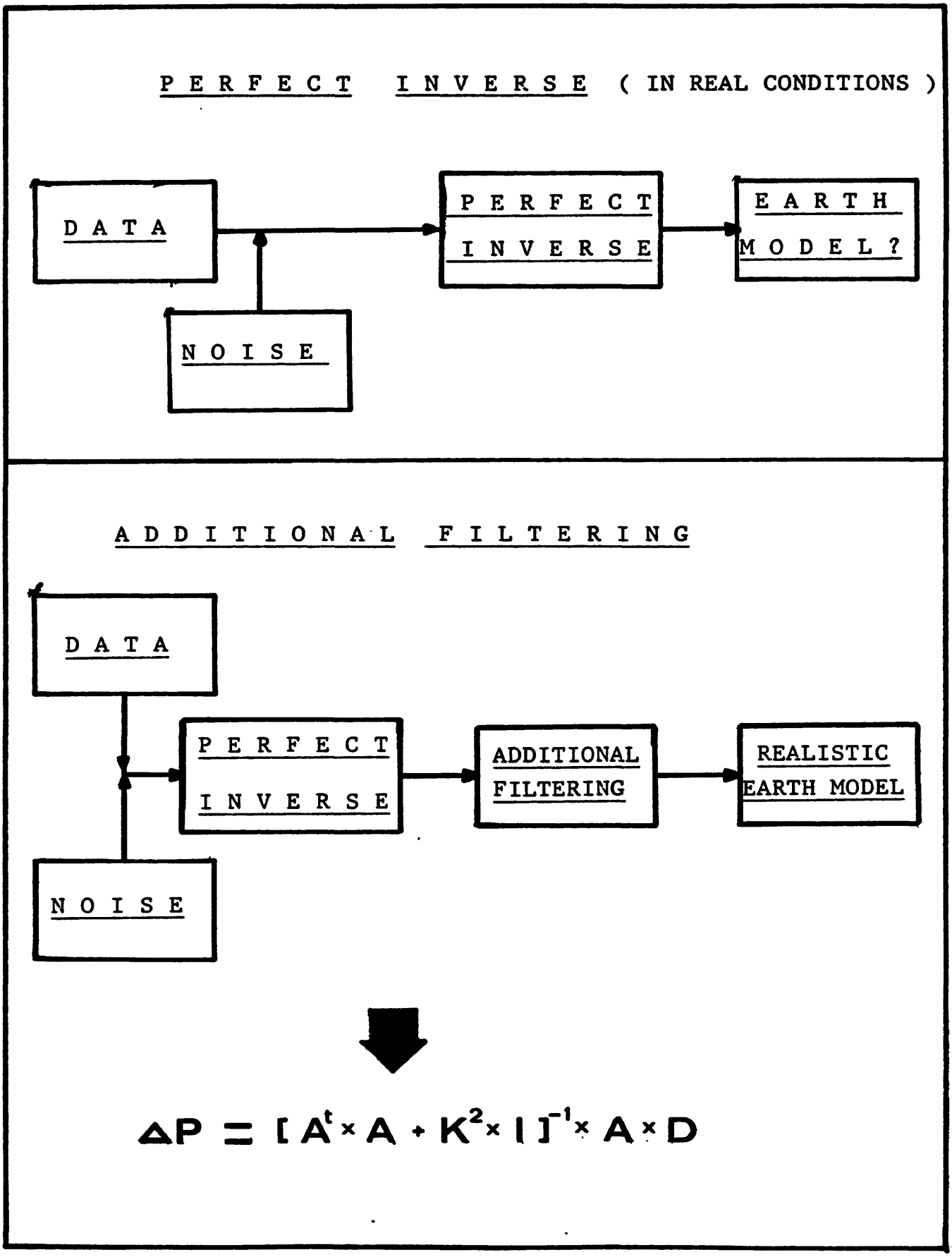


FIGURE 14 FILTERING REPRESENTATION OF INVERSE PROBLEM

Ridge Regression Technique

The ridge regresssion (R.R.) technique is one of the methods which protects the process of inversion against problems derived from the existence of noise. The filtering effect in Ridge Regression is accomplish in a simple manner, by adding to the diagonal of ATA a positive constant K called the Ridge Parameter. Formulation can be expressed as:

$$\Delta\bar{P} = (\underline{A}^t \cdot \underline{A} + K^2 . I)^{-1} \cdot \underline{A}^t \cdot \bar{D} \qquad \text{(III,10)}$$

this expression can be formulated in terms of singular value decomposition as:

$$\Delta \bar{P} = \underline{V} \cdot \underline{\Lambda}^{-1} \cdot \underline{U}^t \cdot \bar{D} \qquad \text{(III,11)}$$

where

$$\underline{\Lambda}'^{-1} = \left[\frac{\lambda}{\lambda^2 + K^2} \right]$$

which appears to be one of the filters described in (Figure 12). The formulation of R.R. has some economical advantages compared to the methods using S.V.D. due to the fact that when using similar constraining methods no factorization of the matrix A in needed.

An important consideration in this formulation is the choice of a value of the ridge parameter K. The optimum value of K is that which gives the minimum error, however this value cannot be determined except by testing the solution.

In order to find the optimum of K it is necessary to examine the solution several times, for different values of K and visualize the plot of the results versus K. The plot obtained, in terms of error (RMS) versus K,is called the Ridge Trace. The plot is always expected to have a shape similar to the curve in Figure 15. The optimum Ridge Parameter corresponds to the minimum in the curve, this minimum is the desired value in the execution of the inversion.

Even in the case where the optimum of K is found, the model determined is not usually the best and another iteration will improve the fit. If the fit does not improve, then convergence is reached. Several iterations may be necessary before the stable solution is obtained; this is then considered to be the result of the inversion.

Finally, although this solution is considered the result of the inversion, sometimes the model obtained is not the only one that fits the data. Additional knowledge of the parameters must be taken into account when this nonuniqueness problem happens.

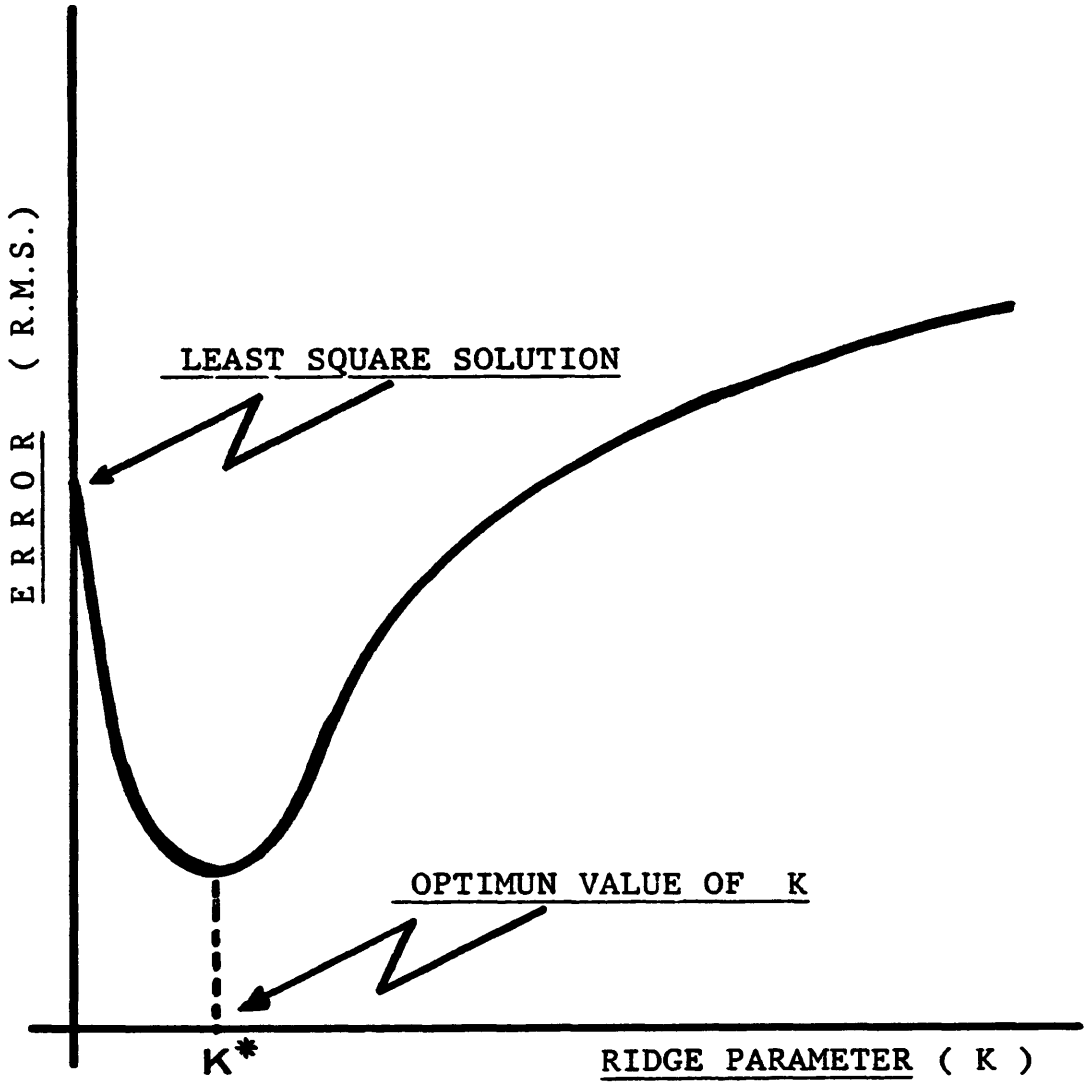


FIGURE 15 IDEAL CURVE OF THE RIDGE TRACE

Economy of the Process

The economy of the inversion process is strongly affected by the cost of each F.S. A high number of calculations of F.S. are necessary to perform an inversion. Figure 16 shows in detail the number of calculations of F.S. in an inversion. The cost of an inversion problem is calculated in this figure for some typical cases, which happens to be forty times the cost of F.S. In some methods where F.S. has a formulation which is easy to compute, the total cost of inversion is still reasonable. However, many of the forward problems are expensive, under these circumstances the algorithm of inversion results in a very expensive method of interpretation.

A more economical approach to the F.S. has to be found in order to make automatic inversion feasible on a commercal basis. In the catalog-interpolation method (described in the next chapter) tedious calculations represented in the idea of functional F, have to be performed only once to generate a catalog of solutions. After the catalog is generated, interpolated F.S. can be easily obtained. The cost of the forward problem will be thus very low, and an economical inversion can be performed in this way.

COST OF AN AVERAGE INVERSION

- THE NUMBER OF FORWARD CALCULATIONS;

CALCULATED VALUE ($\bar{C}$)	1
DERIVATIVES MATRIX ($\underline{A}$) ..... (M is number of parameters)	M
OPTIMIZATION OF RIDGE PARAMETER (estimated)	4
TOTAL F.CALC. PER ITERATION	M + 5

ASSUMING THAT 4 ITERATIONS (AVERAGE) ARE NEEDED BEFORE A FINAL SOLUTION IS OBTAINED, THE TOTAL NUMBER OF CALCULATIONS IN THE INVERSION WILL BE:

TOTAL NUMBER OF F.S. $\underline{4 (M + 5)}$

AND WHEN THE NUMBER OF PARAMETERS IS EQUAL TO FIVE, (M = 5) ;

$$\underline{Icost = 40 \times F.S.cost}$$

THEN, THE COST OF AN AVERAGE INVERSION TURNS OUT TO BE FORTY TIMES THE COST OF A PARTICULAR FORWARD SOLUTION.

FIGURE 16 COST OF AN AVERAGE INVERSION

INVERSION OF TDEM DATA WITH CURVE CATALOG

It was mentioned in chapter II that the transient response of a layered earth (forward solution) requires several integrations over the frequency range. Each of these integrations consumes a large amount of computer time and thus, the cost of a typical forward solution turns out to be quite high. The influence of this cost on the inversion is very important (as it was explained in chapter III), and TDEM inversion will be uneconomical if performed in this way.

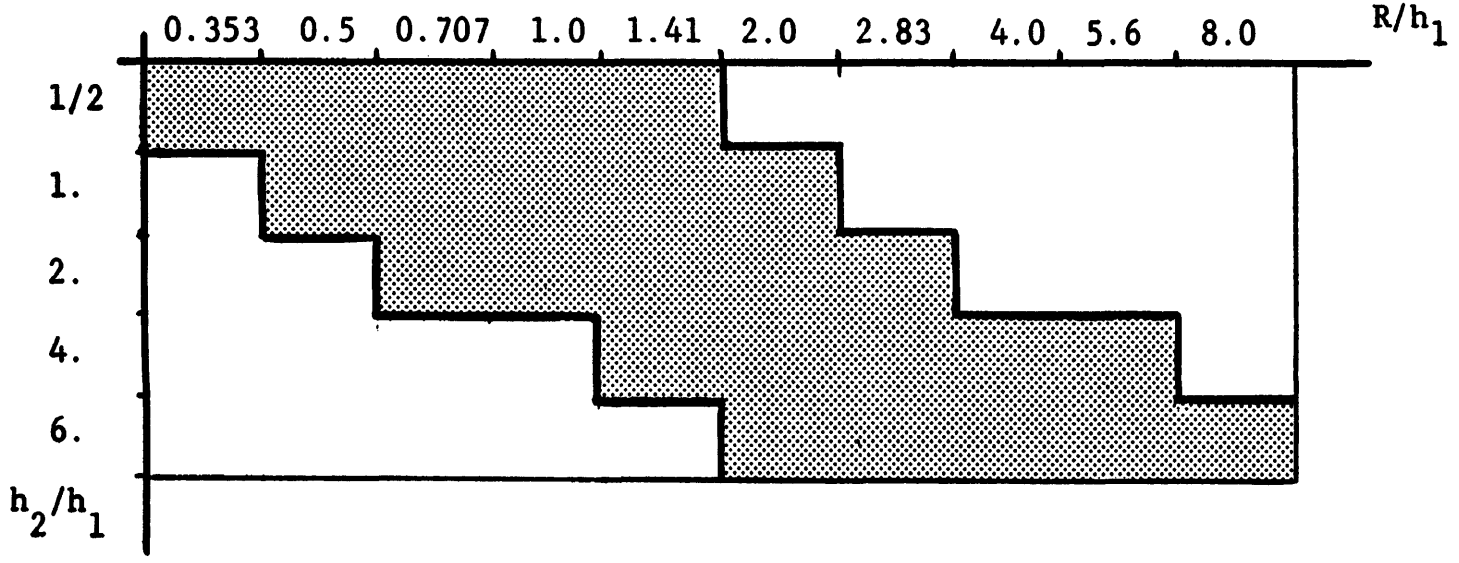
In the present chapter there is a description of the most important features that were taken into account for the design of the inversion program using catalog interpolation. Forward solution (F.S.) is performed by an interpolation process from a set of TDEM near-zone theoretical curves calculated by A.A.Kaufman. The inverse solution is obtained in real time with a ridge regression estimator, which uses some additional calculations to optimize the filter parameter in each iteration, but turns out to be a fast and economical method.

FORWARD SOLUTION

Curve Catalog

Kaufman (1969 and 1970) calculated a set of TDEM master curves in the near-zone. The set includes the most common geoelectrical earth-models for two and three layers. The main features of the behaviour of these curves for different resistivity contrast, were explained in chapter II, where some examples of two-layer earth were described.

The catalog-file used in this inversion process, contains the three layer H type of curves (with resistant basement) from the set calculated by Kaufman. Each curve in this catalog is uniquely defined for constant values of three normalized parameters. These parameters are: resistivity of the second layer (r2/r1), thickness of the second layer (h2/h1), and offset distance transmitter receiver (R/h1), all then normalized with the values of the first layer thickness and resistivity (h1 and r1 respectively). Figure 17 shows the parameter combinations which have a curve represented in catalog. Although each of the parameters has a reasonably broad range of values included, only a few combination of the parameters have a curve represented in this catalog. This limitation in the



COMBINATIONS OF THE NON-DIMENSIONAL PARAMETERS R/h_1, AND h_2/h_1 INCLUDED IN THE CATALOG.

FOR EACH OF THESE COMBINATIONS, THE CATALOG CONTAINS 4 CURVES WITH THE CONTRAST OF RESISTIVITIES:

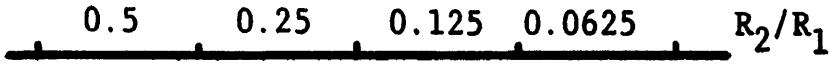


FIGURE 17.-PARAMETER COMBINATIONS WITH CURVE REPRESENTED IN THE CATALOG.

number of catalog models will create some problems in the inversion.

Some examples of these curves are shown in figure 18. The apparent resistivity curves are normally plotted in logarithmic domain (log-log paper) where the ranges of time and lengths differ from one curve to another. The curves have been sampled to give even spacing in the log-domain and then stored in a catalog file. Each curve in the file will be defined by a set of values that include: first sample location in time (To); number of samples in the curve (No), and weighted values of the samples (P1,P2,...). Since multiple access to the file is needed, the shortest possible time must be spent in each of the reading operations. For this purpose a random-access binary file containing several ordered memory locations was used. Each curve of the catalog will be stored in one of these locations, and the corresponding address of that location will be uniquely defined for every combination of the parameters. As a result, this catalog provides an efficient and economic means of retrieving a particular curve.

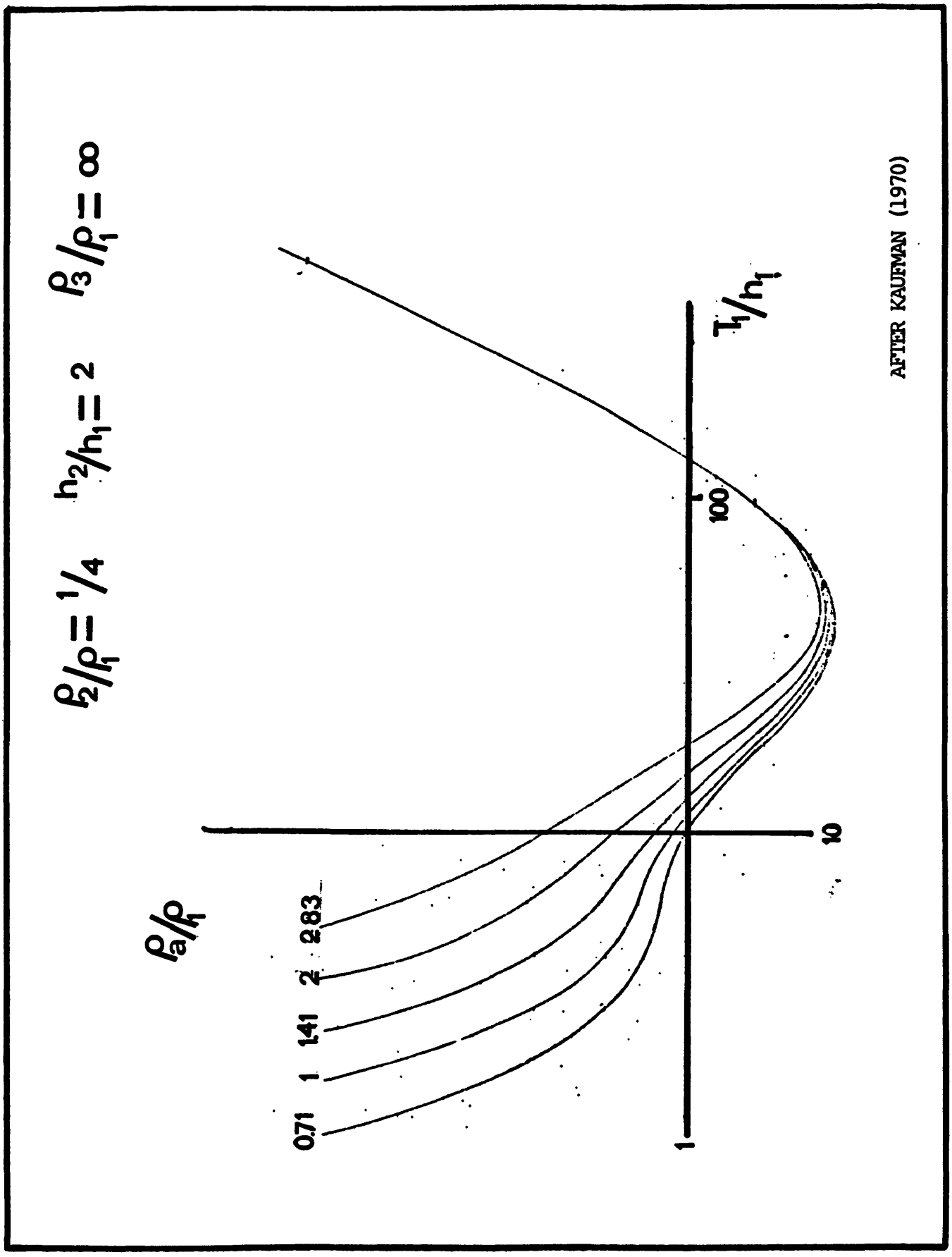


FIGURE 18. NZTDEM APPARENT RESISTIVITY CURVES. EXAMPLE OF THREE-LAYER 'H' TYPE.

Interpolation Procedure

Two interpolations are performed in the process to obtain the forward solution in real time for a given set of parameters. The first interpolation is a multidimensional within the catalog, which calculates the theoretical curve for any given combination of nondimensional parameters which define the catalog. The second interpolation is performed to transform the previous result into a curve of apparent resistivity versus time, at the points where the data was recorded or digitized. Because both normalized resistivity and discreetly sampled normalized time exists in the catalog.

The interpolation within the curve catalog, provides a continuous function of solutions out of the discrete number of curves stored in the catalog file. Several interpolation routines can be adjusted to this problem (e.g.Carnahan et al,1969). A balance between the time spent in the process and the degree of accuracy needed, must be the key factor in choosing a particular method.

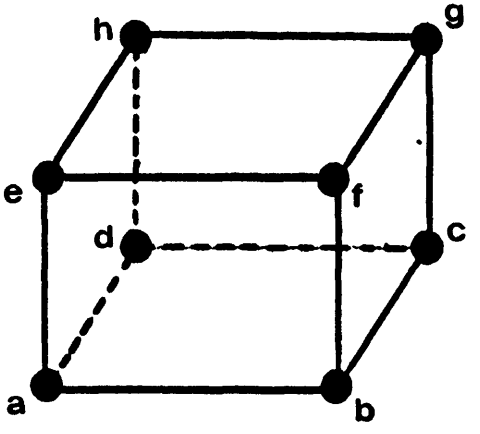
Rodi (1975) describes a linear interpolation procedure in two dimensional space that has a simple formulation. The method is applied here for a three dimensional space, and it assumes that the discrete number of

parameter-combinations with a solution in the original catalog, can be represented in the parameter space by an orthorombic package of elements. In each element of the package, an interpolation function is defined. This function will give a linear approximation to the solution. The accuracy of this approximation depends on the curve density in the catalog. Figure 19 has a brief description of this method and a deduction of a general formula for the interpolation function, this formula has a simple expression, and gives an accurate value of the interpolation for any point within the catalog range, when a proper package of elements is found and translation of the axis is performed.

Once the theoretical curve for the desired values of the normalized parameters is found, the second interpolation must be performed, which will find the values of the apparent resistivity, at the times the data were digitized. Since theoretical curves are expressed as a function of the normalized parameters, a conversion into dimensional parameters (ra vs t) must be done before this second interpolation is accomplished. The result of this interpolator, corresponds to the forward solution calculation used in this inversion.

T-2047

THREE DIMENSIONAL INTERPOLATION



GIVEN:

THE KNOWLEDGE OF THE FUNCTION 'F' IN THE 8 NODES OF AN ELEMENT IN THE ORTHOROMBIC PACKAGE;

ASSUME

INSIDE ELEMENT A FUNCTION OF THE FORM:

$$F(x_1,x_2,x_3) = c_1+c_2x_1+c_3x_1x_2+c_4x_2+c_5x_3+c_6x_2x_3+c_7x_1x_3+c_8x_1x_2x_3$$

THEN, THE VALUE OF THE FUNCTION AT THE NODES WILL DEFINE A SYSTEM OF 8 EQUATIONS. CONSTANTS $c_1,c_2,\ldots,c_8$ ARE THUS UNIQUELY DETERMINED INSIDE THE ELEMENT.

THEREFORE VALUES OF 'F' AT THE NODES WILL DEFINE THE INTERPOLATED FUNCTION INSIDE THE ELEMENT.

FIGURE 19. DESCRIPTION OF THE INTERPOLATION METHOD

Complete forward solution process

Two main routines performed the major calculations described in the previous pages. The subroutine CURVE does the interpolation within the catalog to obtain the theoretical response of the earth for a particular set of nondimensional parameters. The subroutine CALC is the driver of the whole forward solution and performs the second interpolation to obtain the theoretical curve in real time.

The detailed process of CURVE is described in Figure 20. The input to this routine corresponds to a particular value of the three nondimensional parameters. Then the routine looks for the appropiate package of elements. This is done by locating the particular value of each parameter in an ordered vector containing the possible parameter values included in the catalog. When proper package is determined, the curve corresponding to each knot will be read. A reference matrix gives the location of each curve in the random-access binary file. If one of the knots does not have a curve in catalog, then the routine stops its execution, terminating with an error message. The final step corresponds to the interpolation operator. Once the values of the curves in all the knots are known, the three dimensional interpolation can be performed.

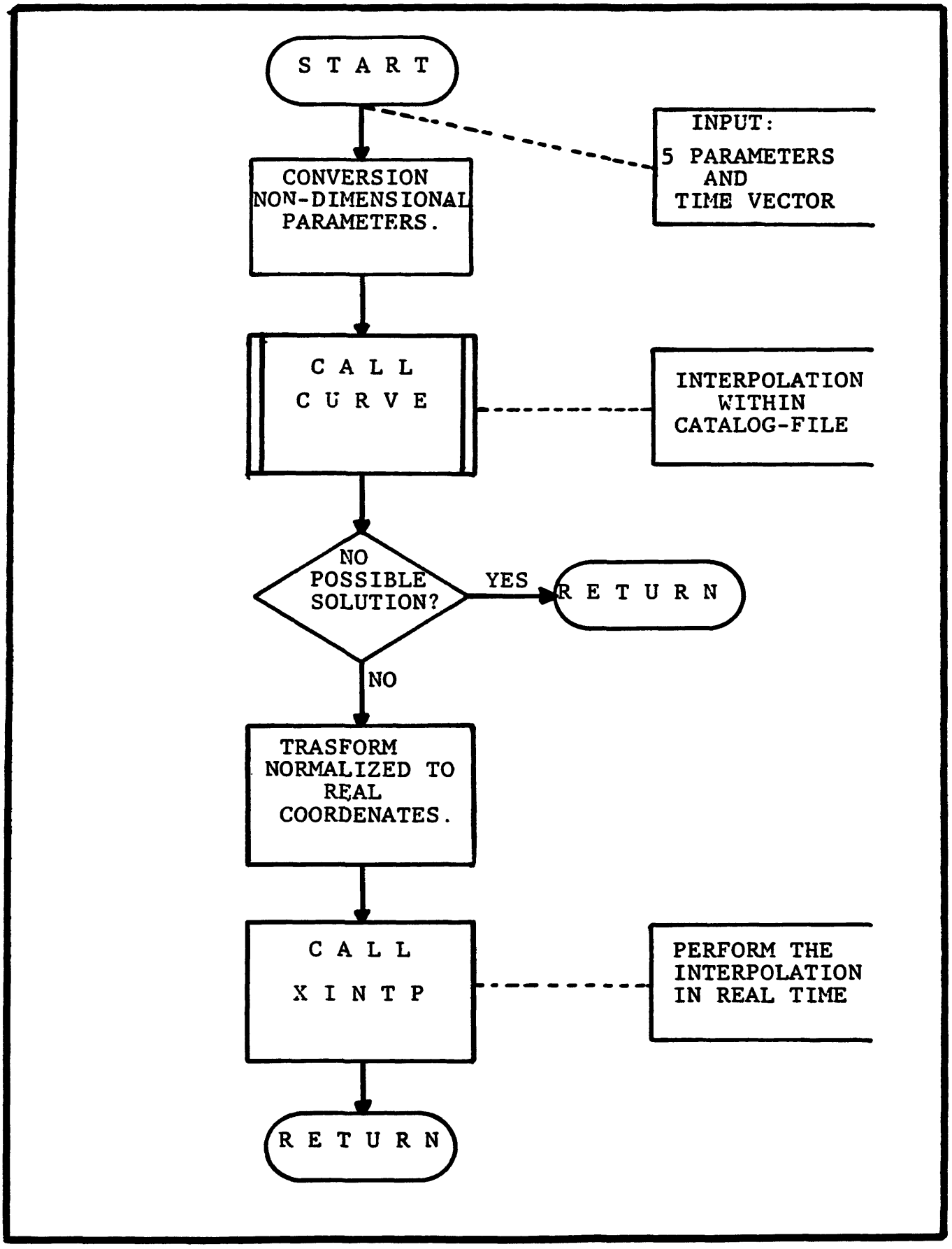


FIGURE 20. FLOW-CHART OF THE SUBROUTINE CURVE.

Final results from this interpolation will be the final results of the routine, that corresponds to an interpolated catalog curve for a particular value of the nondimensional parameters.

CALC (Figure 21) is the routine that controls the whole forward solution process. The input values are a set of geoelecterical parameters, and a vector containing the times at which the data was digitized (or recorded). Then the routine finds the correponding nondimensional parameters that will determine a corresponding theoretical curve in the subroutine CURVE. After this theoretical curve is obtained, the second interpolation will be performed. The final result of this routine, and thus of the whole forward solution, will be the theoretical response for a certain value of the parameters.

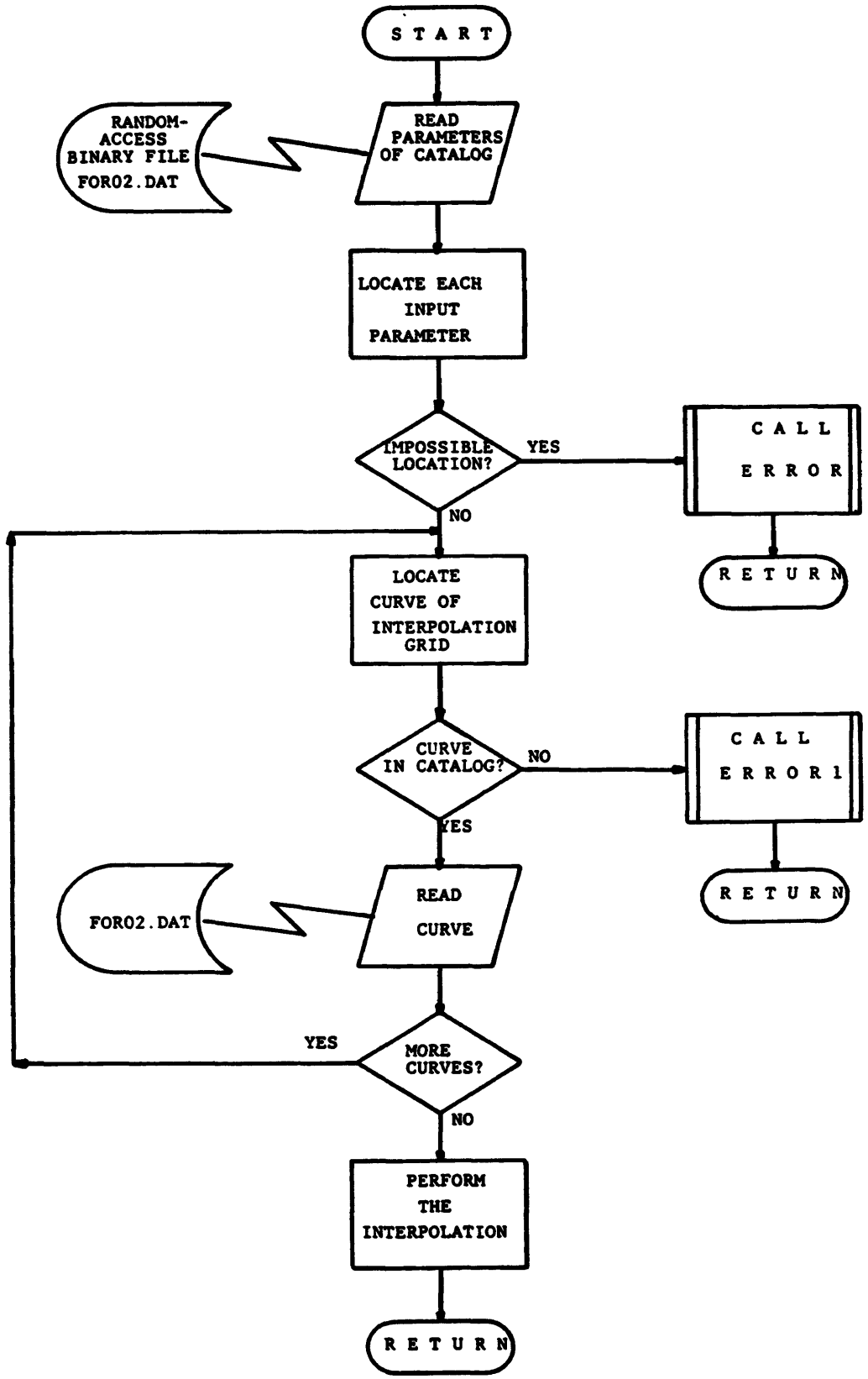


FIGURE 21. FLOW CHART OF THE ROUTINE CALC.

INVERSE PROBLEM

The solution of the inverse problem using the catalog interpolation routine was performed with a ridge regression estimator. Some ideas about the method were described in the previous chapter. A feedback system is used in each iteration, to insure that the filter parameter be close enough to the optimum. Filtering effect was done with the algorithm described in Chapter III [III,10]. This algorithm has a very simple formulation compared with other inversion filters, and gives an excellent result if enough effort is spent in the process of searching for the optimum K.

The sequential calculation involved in the process of inversion, can be visualized with the flow charts of the two main routines of this process, figures 22 and 23.

The first flow diagram corresponds to the routine OPTMO. This routine is the main feature of the ridge regression estimator. Given a set of observations, and a parameter guess, the routine will find the optimum value of the filter parameter K, and its corresponding increment of parameters. This routine can be divided in two main blocks of calculations. The first block does the operations in the algorithm [III,10], previous to the loop of

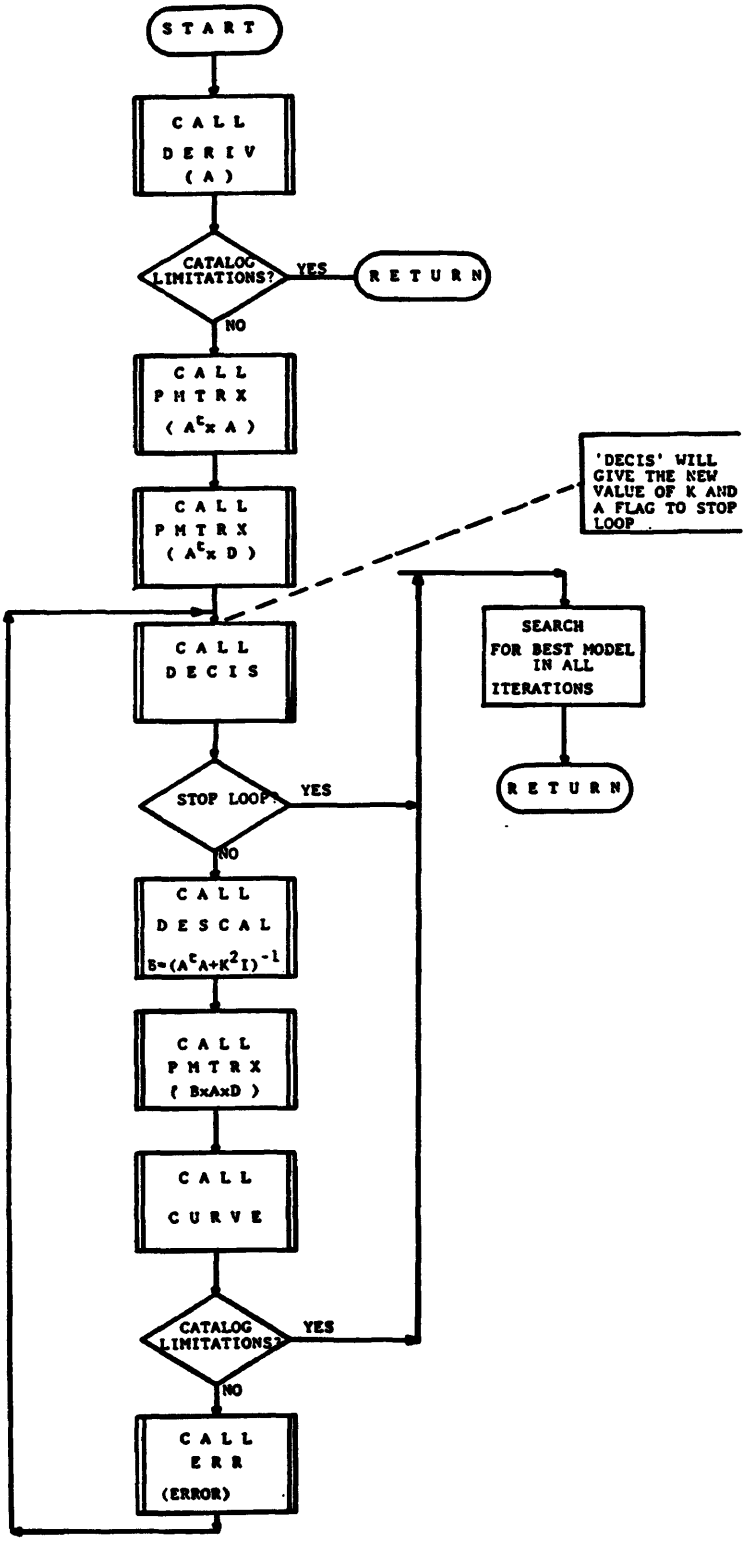


FIGURE 22 FLOW CHART OF THE ROUTINE 'OPTMO'

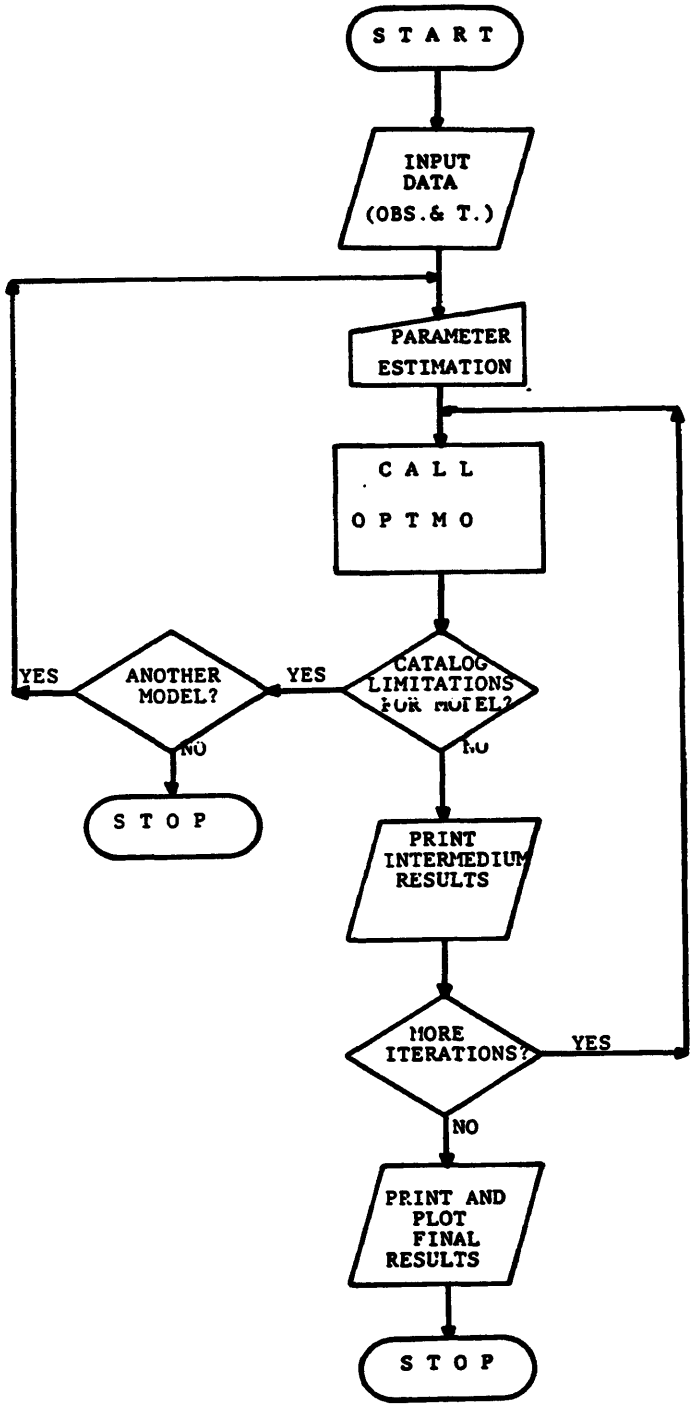


FIGURE 23. FLOW CHART OF THE MAIN PROGRAM.

optimization. These operations are the calculations of the matrix of derivaties A and some subsequent matrix operations. The second group of calculations is where the optimum value of K is determined. This automatic selection is performed by doing several calculations of the algorithm mentioned for values of K (filter parameter) logarithmically spaced (2 per decade). When a point close enough to the minimum of the ridge trace is found, then the optimization iteration will stop. The increment of the parameters associated with that point in the R.T. will be considered to be the optimum. The final result of this routine will be thus, the optimum new-value of the parameters and its corresponding error with the input data.

It was mentioned in the previous chapter, that several iterations with optimum parameter changes may be needed before a stable solution can be found. Figure 23 shows the flow diagram corresponding to the main program of the inversion, where the iterations to find a stable solution are performed.

One important feature in the performace of this inversion that must be finally mentioned, is the limitations of the catalog size. The range of the parameters included in the catalog-file was represented in figure 17. This range of parameters will restrict the

possible catalog models needed during iterations to those included in that small range of values. Some problems due to this fact can be found when performing the calculations of the matrix A and in places where new forward solution models are needed. If problems arise during calculation of A, another finite difference derivative will be made with the incrementation values taken in the opposite direction. In other places of the program the lack of information content in the catalog file will force the operations involved to stop, and sometimes the whole execution of the program will cease.

RESULTS OF THE INVERSION

Four typical transient soundings are shown to ilustrate the results of the inversion. The first two are synthetic data, generated from the catalog file. The last two are real data corresponding to a geothermal survey that was carried out by the geophysics department of C.S.M. in the area of Black Rock desert, Nevada.

In the first example almost perfect fit is obtained after four iterations, with a final error of 0.4%. Figure 24 shows the plot of the error versus the iterations, for this example. It can be observed in this figure, that a significant reduction of the error takes place in the first two iterations. The error declines from an initial value of 54% down to 1.9% after the second iteration. Final solution is obtained in the next two iterations with a final error of 0.4% corresponding to the inverted model. The ridge traces for every iteration are shown in figure 25. It can be observed there, that the shape of those curves is always approximatly the same with some small differences due to the location of the minimum and the curvature. The location of the minimum (K*) can be understood as an indicator of the information content in the matrix. The value of K* for the first two iterations is very small and this agrees with the fact that a great

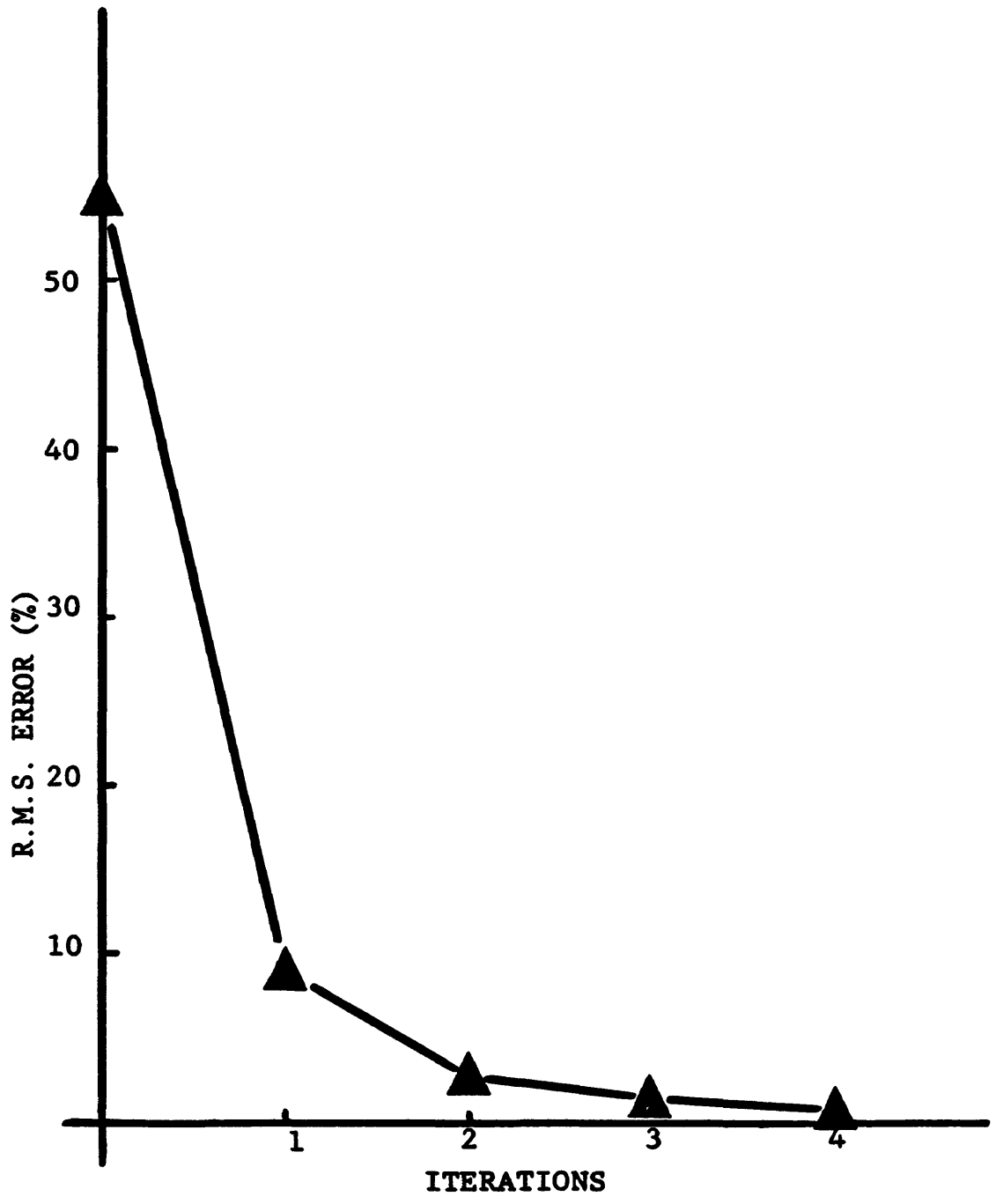


FIGURE 24. EXAMPLE #1 PLOT OF THE ITERATIONS VS. ERROR(RMS)

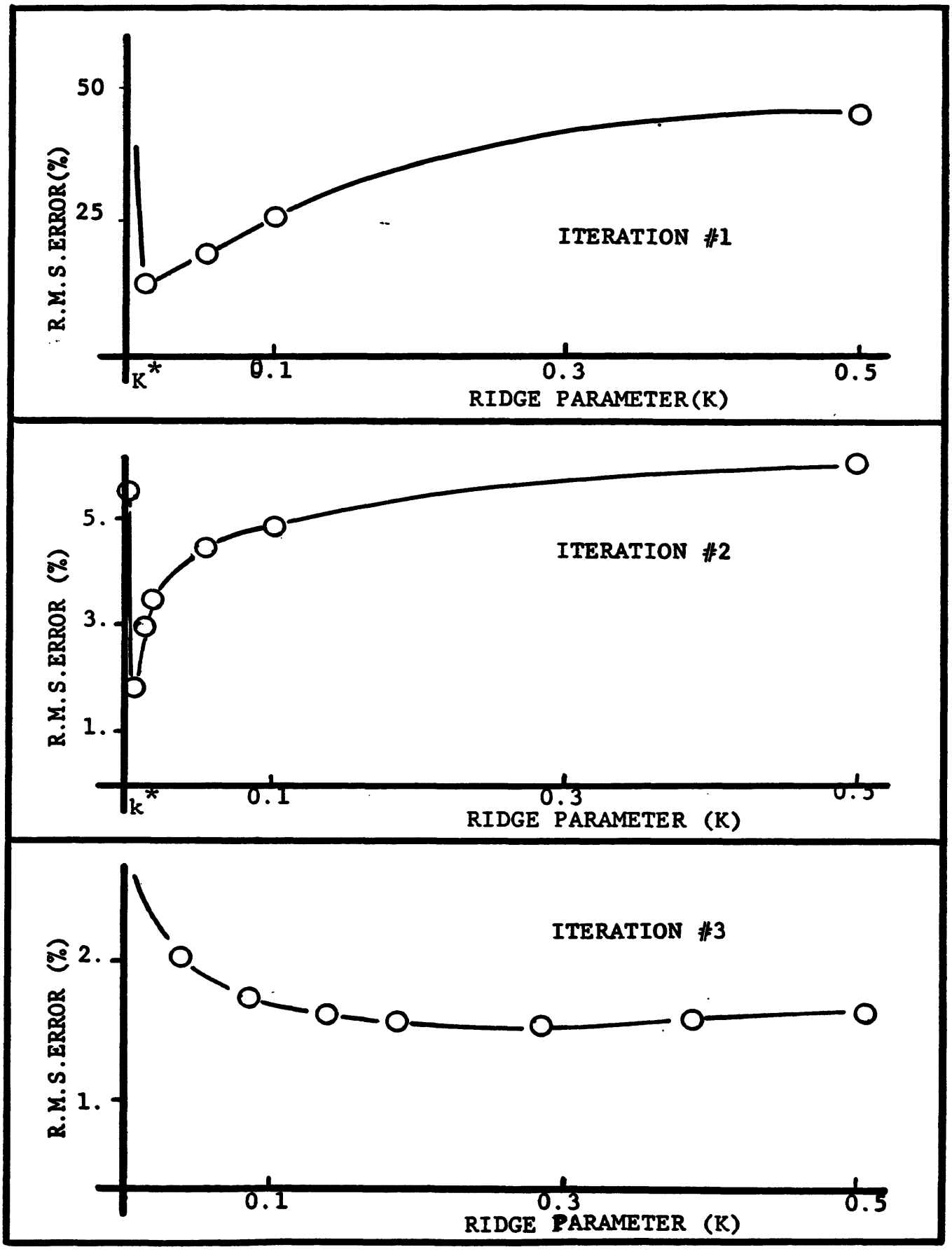


FIGURE 25. EXAMPLE #1 PLOTS OF RIDGE TRACES FOR DIFFERENT ITERATIONS.

reduction of the error took place. The third and forth iterations have large values of K* and the reduction of the error was not very high there.

Figures 26 and 27, represent the parameter changes with iterations. The resistivities are shown in figure 27, and figure 28 contains the corresponding changes for the thickness. These figures reflect a similar trend if compared with the plot of the error (Figure 24) where important changes of the parameters occur during the first two iterations, and relative small changes in the other two. For some parameters, a divergence effect is obtained with the first iteration, but convergence is obtained within the next three iterations. Finally, the changes in the apparent resistivity curves versus the time is shown in figure 28. It can be observed that the difference between the first guess and the observation is great, however, there is almost a perfect match between the final model and the observations.

More impressive results are obtained with the second example, where an error of 120% is reduced to 4% in three iterations. The plot of the error is represented in figure 29, and it can be observed that the first iteration, does not significantly reduce the error. This is due to the limitations of the catalog size, which inhibit the

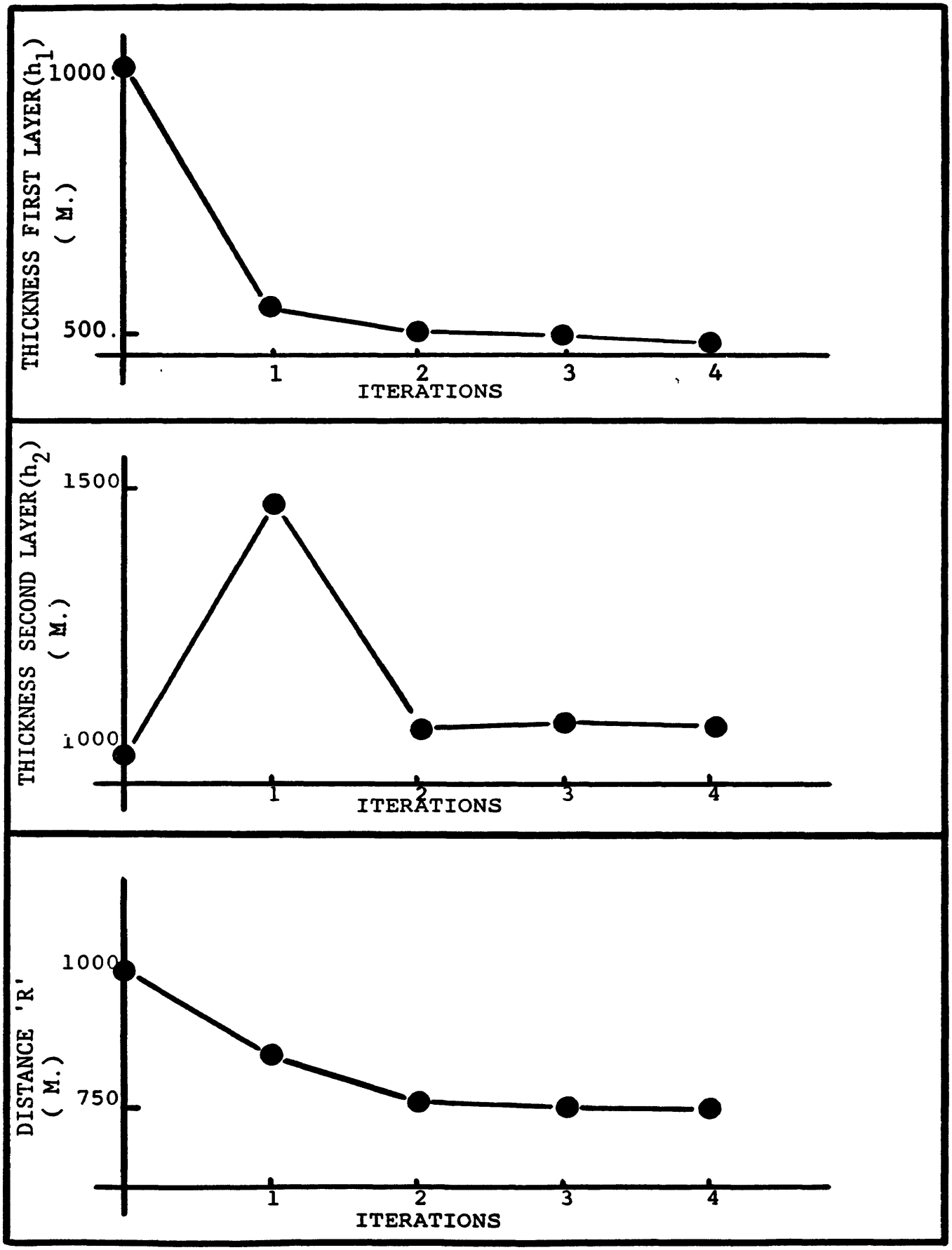


FIGURE 26.- THICKNESS AND DISTANCES CHANGES WITH ITERATIONS EXAMPLE #1.

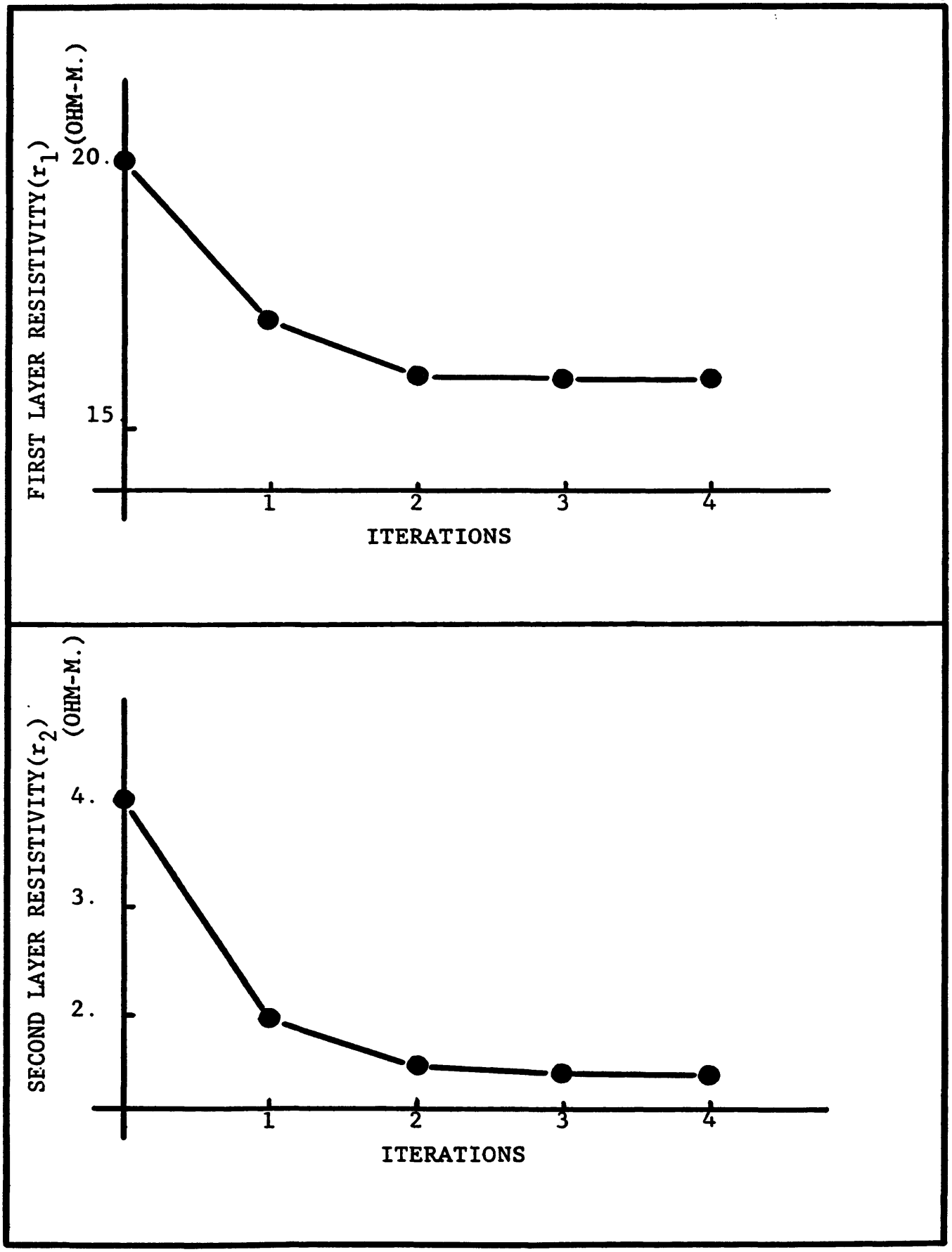


FIGURE 27.- RESISTIVITY CHANGES WITH ITERATIONS EXAMPLE #1.

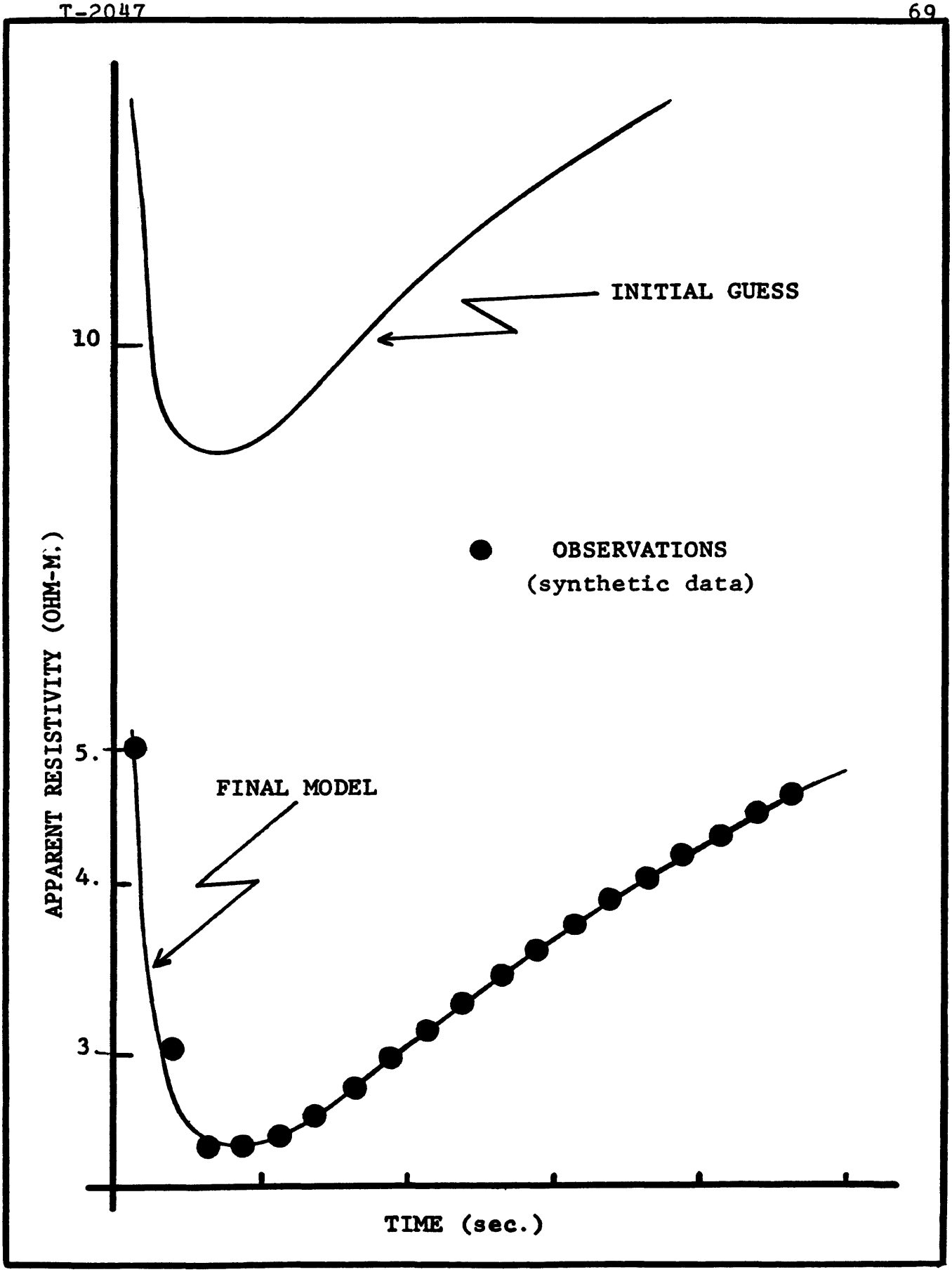
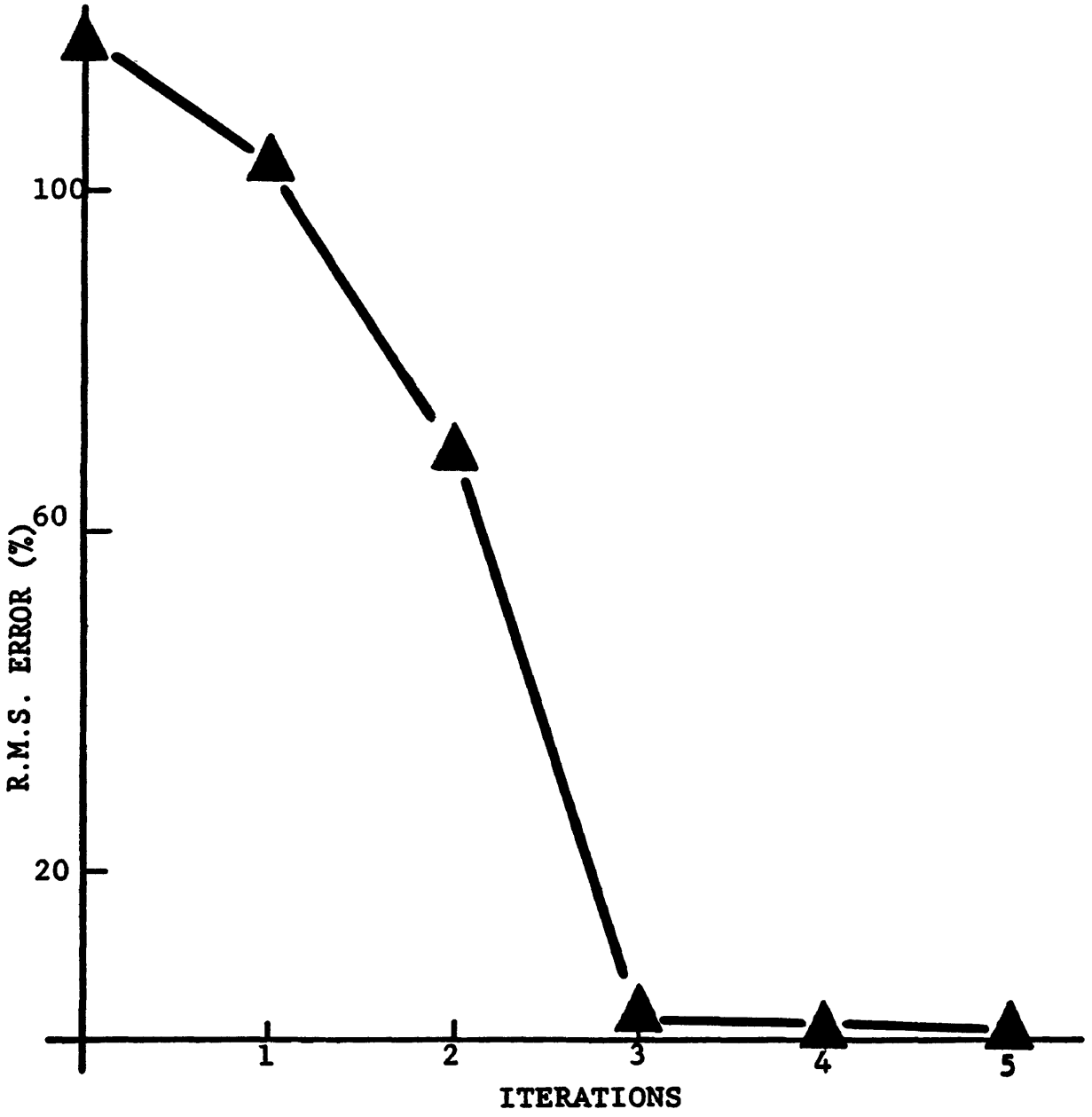


FIGURE 28.- EXAMPLE #1 CHANGE OF THE APPARENT RESISTIVITY CURVES WITH ITERATIONS.

optimization of the ridge parameter (K). The subsequent iterations, operate in a manner similar to the previous models. The apparent resistivity changes with iterations are shown in figure 30.

Similar results can be observed in the other two examples from Black Rock desert. In both cases, an abrupt reduction of the error takes place in the first or the second iteration. After this the noise in the data prevents any further improvement. The plots of apparent resistivity changes, and error versus iteration for those two soundings, are shown in figures 31 through 34. Finally it must be mentioned that the values of resistivities and thicknesses obtained for those two soundings, seem to match very well with the expected geoelectrical section of the area (Keller, personal communication, 1978). The value of the offset distance is considered a parameter here, because it is an indicator of the lateral changes of resistivity, that appear to be very strong in this area (Crewdson, 1976).



	r_1	h_1	r_2	h_2	R
INITIAL MODEL	20.	500.	4.	500.	500.
FINAL MODEL	73	780	18	430	450

FIGURE 29.- EXAMPLE #2 PLOT OF THE ITERATIONS VS.ERROR

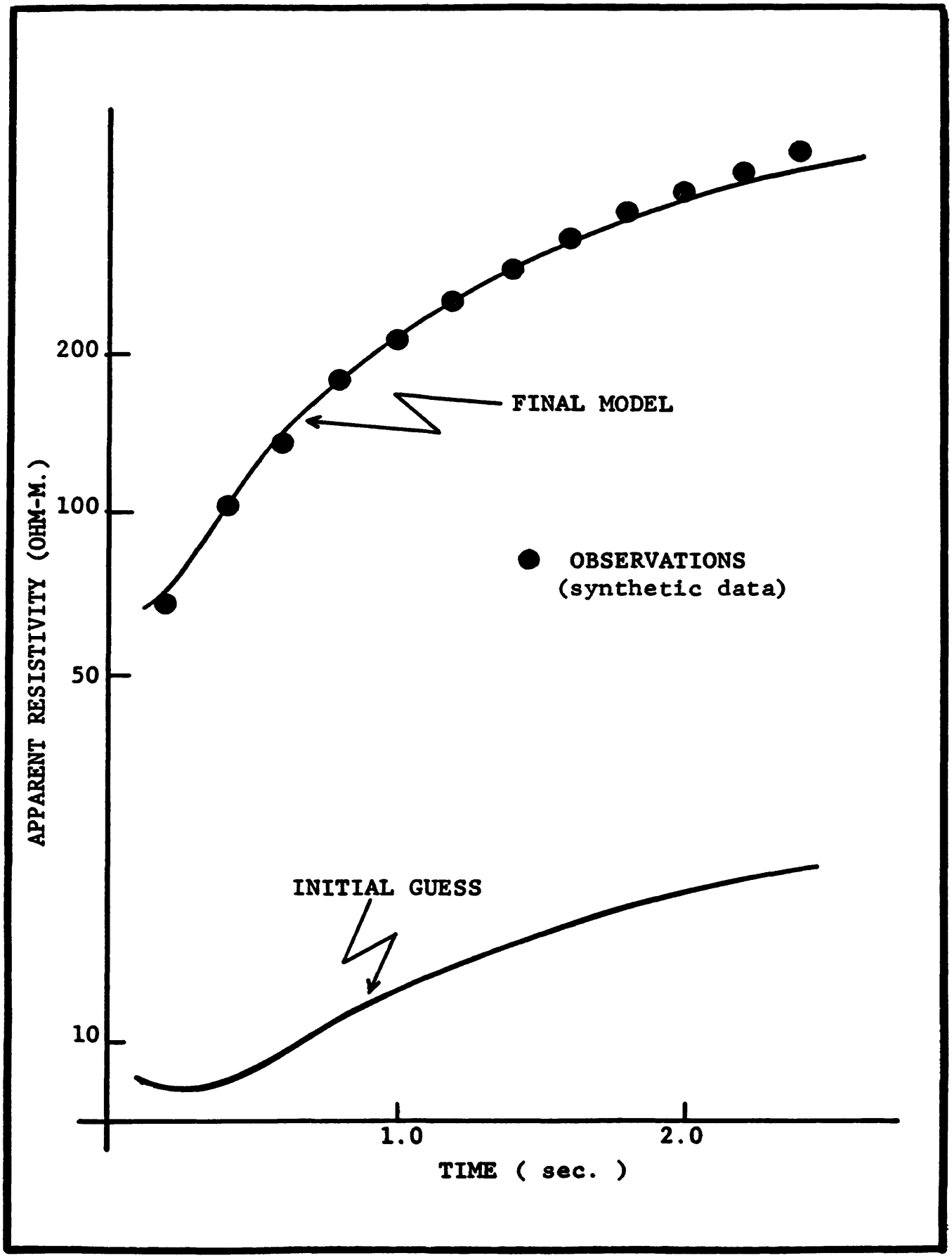


FIGURE 30.- EXAMPLE #2 CHANGE OF THE APPARENT RESISTIVITY CURVES WITH ITERATIONS.

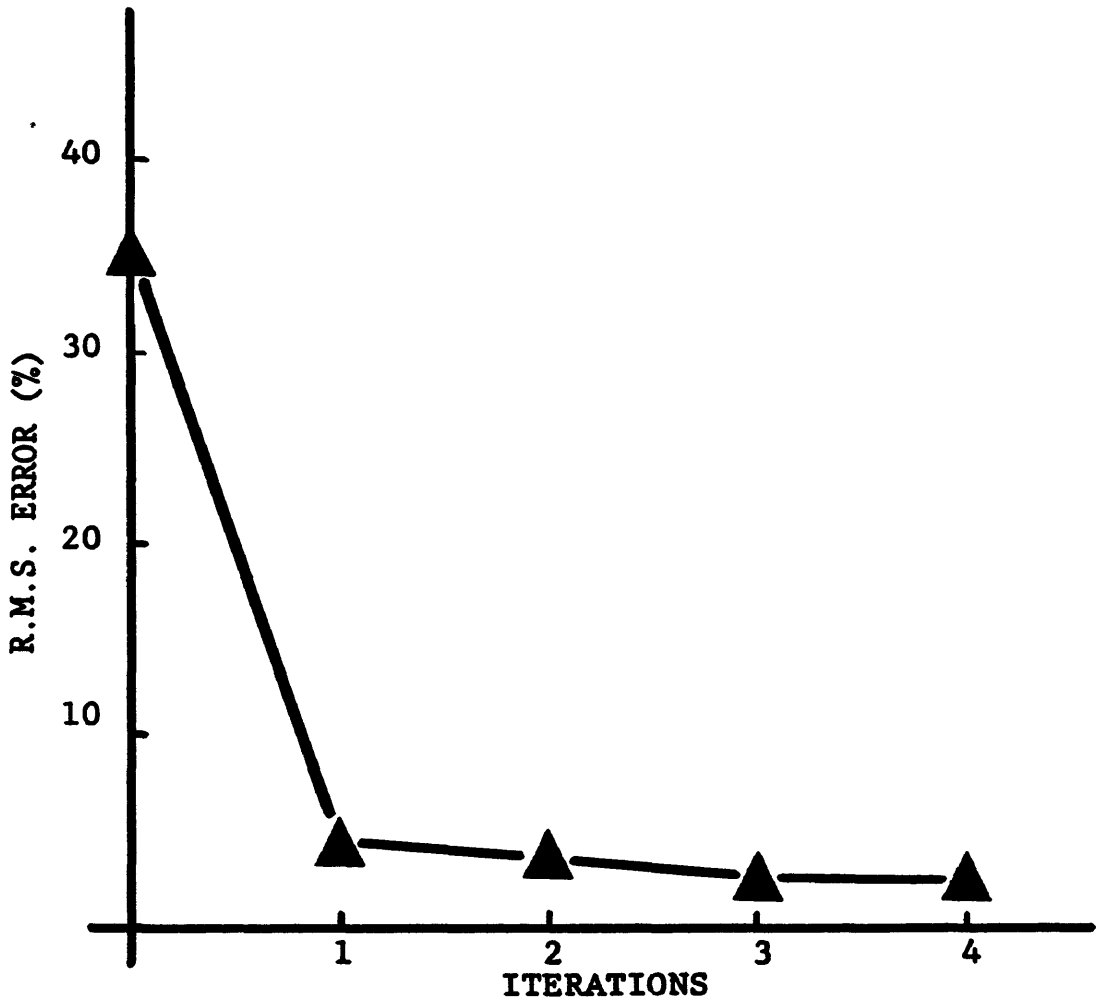


FIGURE 31.- EXAMPLE #3 STATION 13-W (BLACK ROCK DESERT) PLOT OF THE ITERATIONS VS. ERROR.

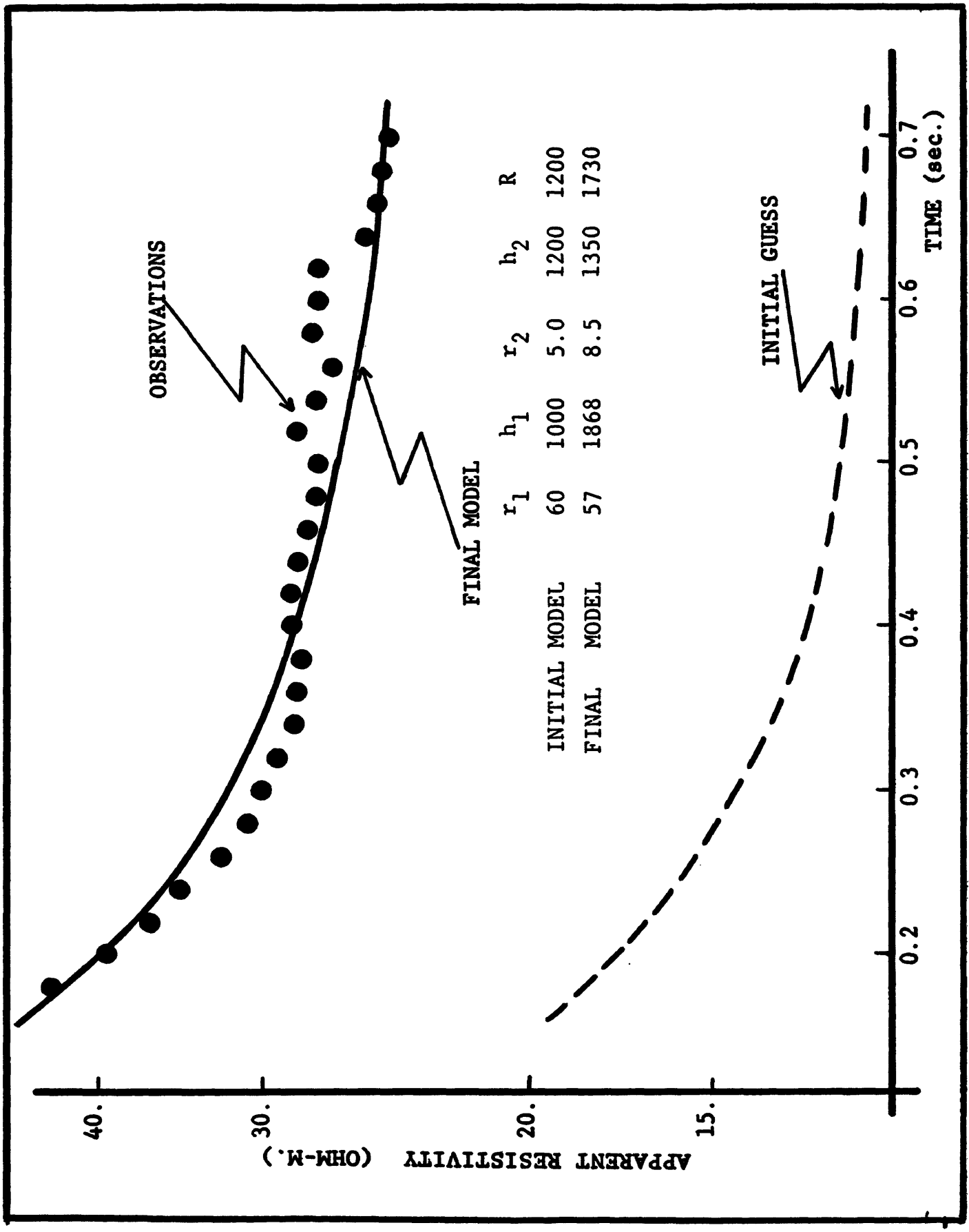


FIGURE 32.- EXAMPLE #3 STATION 13-W (BLACK ROCK DESERT) CHANGES OF THE APPARENT RESISTIVITY CURVES WITH ITERATIONS.

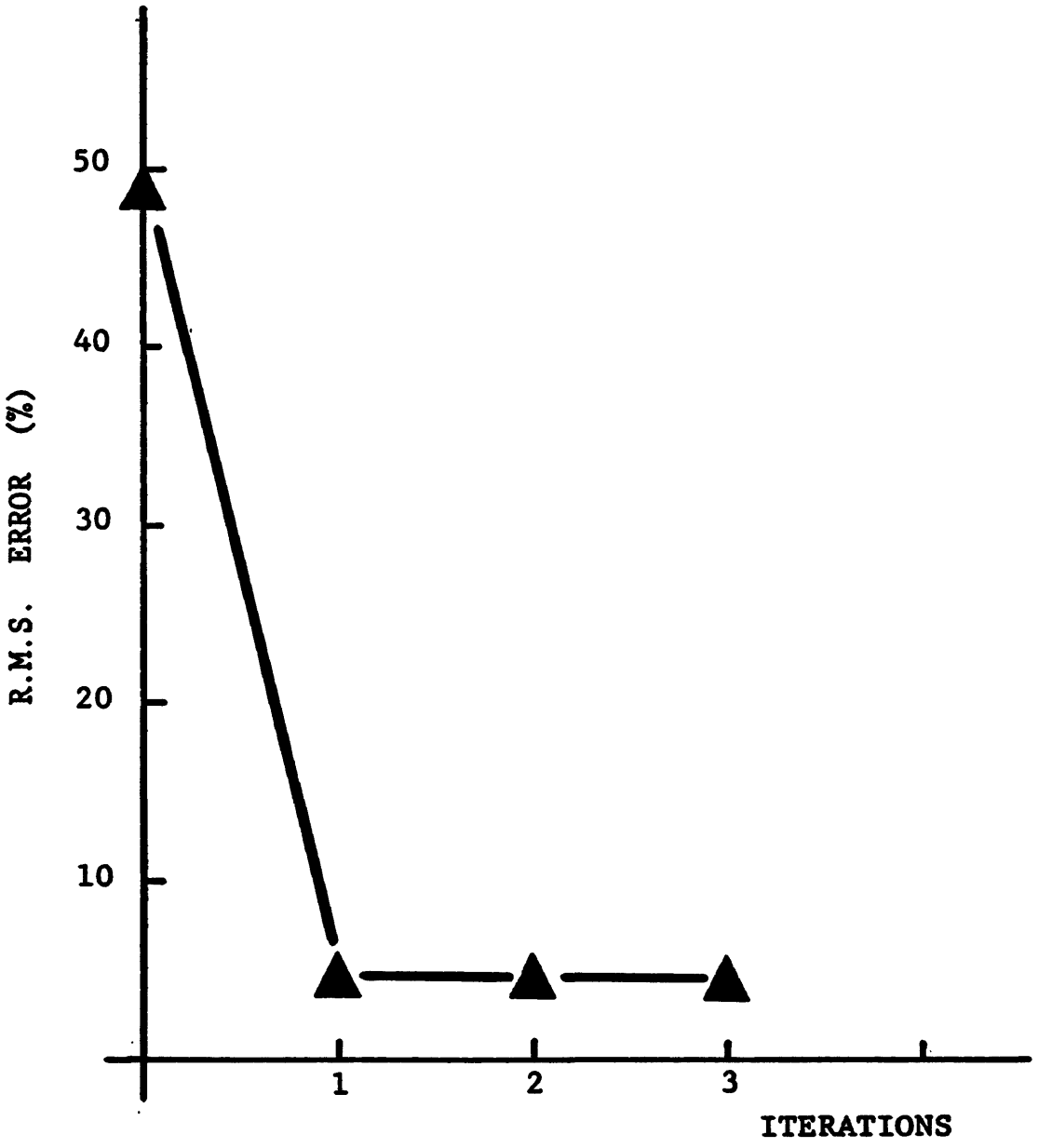


FIGURE 33.-EXAMPLE #4 STATION 63-W (BLACK ROCK DESERT)
PLOT OF THE ITERATIONS VS. ERROR.

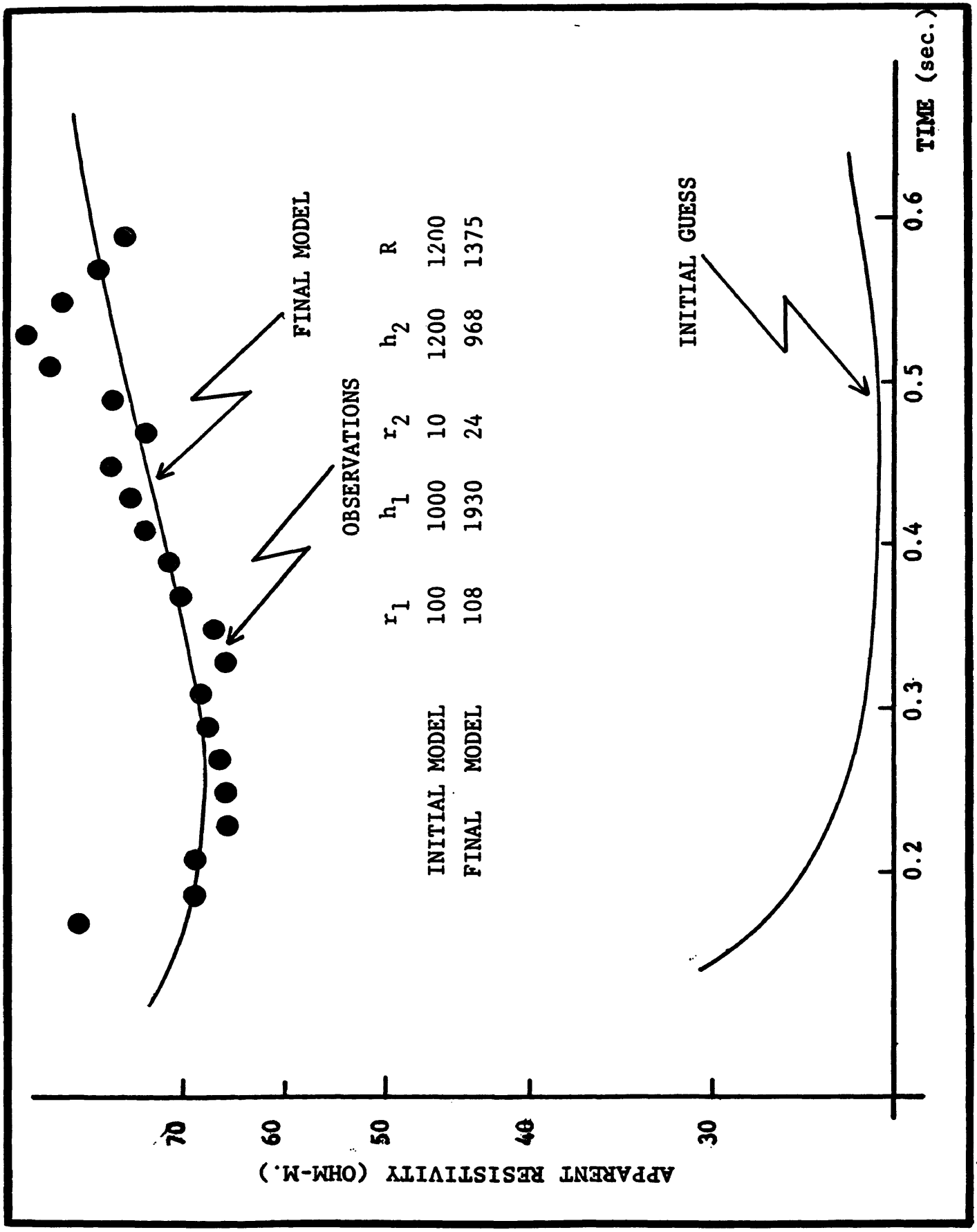


FIGURE 34.- EXAMPLE #4 STATION 63-W (BLACK ROCK DESERT) CHANGES OF APPARENT RESISTIVITY CURVES WITH ITERATIONS.

CONCLUSIONS

Ridge regression estimator with catalog interpolation, provides a fast, accurate and economical automatic method of interpretation for time domain electromagnetic data.

Ridge regression estimator uses a feed-back principle to choose a value of the filter parameter close enough to the optimun. With this value, fastest possible convergence is insured at each iteration and thus the total number of iterations will be the smallest possible.

Since forward calculations of TDEM were expensive, the forward solutions were obtained using catalog look-up and interpolation. This method gives a value of the solution with enough accuracy, and transforms a very expensive method of interpretation into a very inexpensive one.

The most important features of this work can be summarized as follows:

- It has been shown that ridge regression estimator is a powerful tool that insure the fastest possible covergence. Two or three iterations with this method are needed in most of the cases, to reduce the error in the final model down to a low and stable value.

- Forward calculations of TDEM obtained with catalog

look-up and interpolation are several orders of magnitude cheaper than the mathematical model of F.S..

- Interpolation method provides an accurate value of the forward solution if enough density of points is included in the catalog.

Then this algorithm of inversion provides a fast, economical, and accurate method for the automatic interpretation of TDEM data, adequate to operate on a commercial bases. The only problem observed with this powerfull technique, was caused by the limitations due to the catalog size. It is recomended thus, that further works with this algorithm, should include a dense and extensive catalog of solutions. Finally, the aplicability of the algorithm can be extended to any type of geophysical data, provided a catalog of solutions, and specially interesting for the inversion of data whose solution is nonunique.

APPENDIX

```
C
C
C ***************************************************************
C ****       MAIN.FOR                                          ****
C ****  THIS IS THE MAIN PROGRAM OF THE INVERSION OF TDEM ******
C **** (NEAR-ZONE) CURVES USING CATALOG LOOKUP AND INTER- ******
C **** POLATION. THE DETAILED EXPLANATION OF THE ALGORITHM *****
C **** AND FLOW CHARTS OF THE WHOLE PROCESS, CAN BE FOUND ******
C **** IN THE C.S.M. MASTER THESIS T-2047.                 ******
C ****                                                    ******
C ****        INPUT FILE..... FOR15.DAT                   ******
C ****              THIS FILE IS THE OUTPUT OF GFACT.F4    ******
C *****             AND CONTAINS: NUMBER OF DIGITIZED
C *****             POINTS, AND PAIRS (APPAR.RES.,TIME )  *******
C ****              FOR EACH POINT.                      ******
C ****       OUTPUT FILE ..... THE PROGRAM GENERATES      ******
C ****              AN LPT FILE THAT CONTAINS PLOTS (IN LPT)*****
C ****              OF THE RIDGE TRACES, PARAMETER CHANGES,******
C ****              ERROR VS. ITERATION, AND MODEL CHANGES ******
C ****              (ALL THESE PLOTS ARE OPTIONAL).        ******
C ****                                                     ******
C ****  THE PROGRAM WILL WORK IN INTERACTIVE MODE, WITH    ******
C **** SEVERAL OPTIONS: 1.- AUTOMATIC OR MANUAL;           ******
C **** 2.- PARAMETERS TO INVERT 3 OR 5 (3 IN THE CASE OF *******
C **** INVERSION WITH NONDIMEN. PARAMETERS.).             *******
C ****                                                    *******
C ****   ADDITIONAL ROUTINES USED:                        *******
C ****      OPTMO; DECIS; CURVE; CALC; FIND;              *******
C ****      PLOT.FOR; PRINT; PRINT1                       *******
C ****      AND THE MATRIX PACKAGE CONTAINING:            *******
C ****       IRPA; PMTRX; ERROR; DESCAL; DERIV;           *******
C ****       TRACE; MAMIN.                                *******
C ****                                                    *******
C **** THE CATALOG OF CURVES IS INCLUDED IN THE FILE      *******
C **** FOR02.DAT (RANDOM-ACCESS BINARY FILE).              ******
C ****                                                    *******
C ****                                                    *******
C **** ADD.INFO. IN THE CASE OF CATALOG LIMITATIONS,      *******
C **** THE FILE FOR18.DAT, WILL CONTAIN INF. OF THE LOC.  *******
C ****  IN THE CATALOG (NONDIM. PARAM.) USED UNTIL THE    *******
C **** PROBLEM ARISES.                                    *******
C ****                                                    *******
C ***************************************************************
C ****   JUAN CARLOS RODRIGUEZ-OVEJERO 1978, GOLDEN       *******
C ***************************************************************
C
C
C
      PARAMETER MNA=200,NPT=40,NPAR=5
      PARAMETER NNP=21
```

```
      LOGICAL ANSWER,FUERA,AUTO,RTPLT,PRPLT
      DIMENSION PBEST(NPAR),XN(15),STRGP(MNA),OLDAT(NPT)

      COMMON / BLOCK1 / OBS(NPT) ,T(NPT) ,P(NPAR) ,DCALC(NPT),
     &  A(MNA) ,IMAX ,IMIN,MNA1,NPT1,NPAR1,RTPLT

      COMMON / BLOCK2 / ERR,ERRO(15),ERRI(15),INTER,ITER,
     &  CONSK(10),XNEW,IFLAG

      NPT1=21
      NPAR1=3
      MNA1=63
      DATA (XN(I),I=1,15)/1.,2.,3.,4.,5.,6.,7.,8.,9.,10.,11.,12.,
     &  13.,14.,15./
      DATA (OBS(I),I=1,NNP)/1.2,1.17,1.06,0.96,0.88,0.83,0.79,
     &  0.75,0.72,0.70,0.68,0.67,0.66,0.65,0.64,0.63,0.625,
     &  0.62,0.615,0.614,0.613/
      DATA (T(I),I=1,NNP)/0.0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,
     &  1.0,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9,2.0/
C ***************************************************************
C ******      INPUT DATA ; NPAR1,NPT1,T(I),OBS(I)          ********
C ***************************************************************

      WRITE(4,25)

25    FORMAT(1X,'ENTER THE NUMBER OF PARAMETERS ' ,//,
     &  1X,'3......SYNTHETIC DATA (R2/R1,/,H2/H1,/,R/H1) ',/,
     &  1X,'5......REAL DATA (R1,H1,R2,H2,R)  ' ,$ )

      ACCEPT 15,NPAR1
15    FORMAT( I)


      READ (15,26) NPT1
      DO 23 I=1,NPT1
23    READ (15,27) T(I),OBS(I)
      MNA1=NPAR1*NPT1

26    FORMAT ( I)
27    FORMAT (25X,2(F10.4,5X))




      GO TO 6


C ***************************************************************
C ****** WHEN MODEL OUT OF CATALOG ; START AGAIN         ********
C ***************************************************************
```

```
5     CONTINUE

      FUERA=.FALSE.

      WRITE(4,125)
125   FORMAT(1X,/,' DO YOU WANT TO TRY WITH ANOTHER MODEL   ',
     &    $)
126   FORMAT(1X,/, 'DO YOU WANT THE PLOT OF THIS RESULTS  ',$)

      IF(ANSWER(DUMMY)) GO TO 6
      WRITE(4,126)
      IF(ANSWER(DUMMY)) GO TO 280

      STOP

C ***************************************************************
C *****           INPUT PARAMETERS                        ********

C ***************************************************************



6     FUERA=.FALSE.
      AUTO=.FALSE.
      RTPLT=.FALSE.
      PRPLT=.FALSE.
      WRITE(4,8)
      ACCEPT 7,P
      WRITE(4,9) P
C CHECK IF THE PARAMETERS ARE CORRECT
      WRITE(4,105)
105   FORMAT(1X,'DO YOU WANT TO TYPE AGAIN?   ', $)
8     FORMAT(1X,' ENTER THE VALUES OF THE PARAMETERS  ', $)
7     FORMAT(5F)
9     FORMAT(5(F10.4,3X))
      IF(ANSWER(DUMMY)) GO TO 6

      DO 170 I=1,NPAR1
            STRGP(I) = P(I)
170   CONTINUE
C
C
C ***************************************************************
C ******         INITIAL GUESS MODEL AND ERROR           ********
C ***************************************************************
C
C
      IMAX=NPT1
      IMIN=1
      CALL CALC(OCALC,T,NPT1,IMIN,IMAX,P,NPAR1,FUERA)
      IF(FUERA) GO TO 5
```

```
C THIS MATRIX IS USED FOR PLOTTING PURPOSES ONLY(OLDAT).
      DO 115 I=1,NPT1
115   OLDAT(I)=DCALC(I)
C
C NOW TRY TO FIND THE RMS ERROR IN THE INPUT MODEL
C
      CALL ERROR ( OBS,DCALC,NPT1,IMIN,IMAX,ERR)
C
C THIS IS THE ERROR OF THE EXTERNAL ITERATION # 0 (IN THE MATRIX
C WILL BE LOCATED WITH  1 UNIT MORE)

      ERRO(1)=ERR
      ERRMIN=ERR
      ITER=1
      WRITE(4,120) ERR
120   FORMAT(1X,20X,'ERROR IN THE INPUT DATA',F,/)


C ***************************************************************
C *******         AUTO/MANUAL ESTIMATION OF K             ********
C *******       PLOTS OF THE RIDGE TRACE ?                ********
C ***************************************************************
C
C
C WITH THE VARIABLE AUTO(LOGICAL), WE HAVE THE CHOICE OF MANUAL
C OR AUTOMATIC DECISSION ABOUT NEW "K"(IN SUBROUTINE DECIS)


      WRITE(4,128)
      IF(ANSWER(DUMMY)) AUTO=.TRUE.
      WRITE(4,129)
      IF(ANSWER(DUMMY)) RTPLT=.TRUE.
      WRITE(4,131)
      IF(ANSWER(DUMMY)) PRPLT=.TRUE.


128   FORMAT(1X,'DO YOU WANT AUTOMATIC ESTIMATION OF "K"?  ',
     &    $)
129   FORMAT(1X,'DO YOU WANT PLOTS OF THE RIDGE TRACE?  ',$)
131   FORMAT(1X,'DO YOU WANT PLOTS OF PARAMETERS CHANGES ?  ', $)

C
C NOW IT WILL START THE LOOP WHERE THE ITERATION TAKES PLACE
C AND ALSO AN OPTMIZATION OF "K" (RIDGE REGRESION) IN EACH ITERAT.
C
1000  CONTINUE
C ***************************************************************
C ******      OPTIMIZATION OF DELTA DE P                ********
C ***************************************************************
C
```

```
C THE VALUES OF THE PARAMETERS FOR THE NEW MODEL,AND THE CORRES-
C PONDENT ERROR (ERR) ARE THE MAIN CONTRIBUTION OF THIS OPTIMIZA-
C TION SUBROUTINE

      CALL OPTMO(FUERA,AUTO)

      IF (FUERA) GO TO 5

      ITER=ITER+1
      ERRO(ITER)=ERR

C STORAGE THE NEW PARAMETERS IN A  ONE-DIMENSIONAL ARRAY

      DO 171 I=1,NPAR1
             IJ = (ITER-1)*NPAR1+I
             STRGP(IJ) = P(I)
171   CONTINUE

C WE LOOK FOR THE OPTIMUN MODEL(LEAST RMS ERR)

      IF(ERRMIN.LT.ERR) GO TO 40
      ERRMIN=ERR

      DO 30 I=1,NPAR1
             PBEST(I)=P(I)
30    CONTINUE
40    CONTINUE

C IN THE CASE OF THE LAST INTER ITERATION CORRESP TO MIN. ERR.
C THIS CALCULATION WOULD NOT BE NECESSARY.

      CALL CALC(DCALC,T,NPT1,IMIN,IMAX,P,NPAR1,FUERA)

      IF(FUERA) GO TO 5

      NITER=ITER-1
      WRITE(4,130) NITER,ERR
      WRITE(4,140)

      DO 20 I=1,ITER

             J=I-1
             WRITE(4,150) J,ERRO(I)

20    CONTINUE
```

```
130   FORMAT(1X,//,20X,"ITERATION  # ",I2,8X,"ERROR=",F7.4)
140   FORMAT(1X,/,20X," ERRORS IN PREVIOUS ITERATIONS")
150   FORMAT(1X,30X,I3,8X,F7.4)

C ***************************************************************
C *******       MORE ITERATIONS?                          ********

C ***************************************************************



C IF CONTINUE IN THE LOOP, WE HAVE TO CONSIDER THAT THERE ARE NEW
C VALUES OF:DCALC(NPT1),IMAX,IMIN; IN ORDER TO ENTER AGAIN IN OPTMO

      WRITE(4,275)
275   FORMAT(1X,"ANOTHER ITERATION?  ", $)
      IF(ANSWER(DUMMY)) GO TO 1000

280   CALL CALC(DCALC,T,NPT1,IMIN,IMAX,PBEST,NPAR1,FUERA)
      IF(FUERA) GO TO 5
C
C
C ***************************************************************
C *****                  OUTPUT THE RESULTS              ********
C ***************************************************************
C
C
      WRITE(6,300) ERRMIN
      WRITE(6,310) PBEST

      WRITE(6,320)

      KMAX=IMAX-IMIN+1
      DO 350 I=1,KMAX
             J=I-1+IMIN
             D=OBS(J)-DCALC(J)
             WRITE(6,360) OBS(J),DCALC(J),D,OLDAT(J)
350   CONTINUE

      WRITE(6,370)NITER,ERRMIN


C NOW THE VALUES OF THE FIRST GUESS : INPUT DATA, AND
C FINAL ESTIMATION WILL BE PLOTED IN THE SAME GRAPH.

      CHARO=1HO
      CHARI=1HI
      CHARF=1HF
```

```
C  CALCULATION OF THE MAX AND MIN IN THE TIME VECTOR
C  AND IN THE OBSERVATION VECTOR TOGETHER WITH INITIAL GUESS.


      CALL MAXMIN(T,NPT1,XN,XX)
      CALL MAXMIN(OBS,NPT1,YN,YY)
      CALL MAXMIN(OLDAT,NPT1,DMIN,DMAX)
      IF(DMAX.GT.YY) YY=DMAX
      IF(DMIN.LT.YN) YN=DMIN


      CALL PLOT1P(T,OBS,NPT1,CHARO,XN,XX,YN,YY,0)
      CALL MORE1P (T,OLDAT,NPT1,CHARI)
      CALL PLOTPR(6)
      WRITE(6,390)

      CALL PLOT1P(T,OBS,NPT1,CHARO,0.,0.,0.,0.,1)
      CALL MORE1P(T,DCALC,NPT1,CHARF)
      CALL PLOTPR(6)
      WRITE(6,395)


300   FORMAT(1X,///,10X,' OUTPUT  MODEL ',10X,'ERROR= ',F7.4)
370   FORMAT(1X,20X,'NUMBER OF ITERATIONS',I2,10X,'ERROR=',F7.4)
310   FORMAT(1X,20X,5(F10.4,5X))
320   FORMAT(1X,//,20X,'OBSERVATIONS',10X,'CALC.VALUE',10X,
     &  'DIFFERENCE',10X, 'INITIAL GUESS')
360   FORMAT(1X,20X,4(F7.4,15X))
380   FORMAT(1X,//, 20X,'****** PLOT OF THE ERROR V.S.ITERATION*****')
375   FORMAT(1X,/,20X,'****PARAMETER ',I2,' CHANGES *****')
385   FORMAT(1X,10X,F2,5X,F10.4)
390   FORMAT(1X,//,20X,'****** PLOT OF THE OBSERVATIONS (O)',
     &  ' VS. THE FIRST GUESS ******')
395   FORMAT(1X,//, '****** PLOT OF FINAL MODEL (F) VS.',
     &  ' OBSERVATIONS (O) ****** ')



      CHAR=1H*
      IF(PRPLT) GO TO 400
      GO TO 450
400   DO 425 I=1,NPAR1

C EACH OF THE PARAMETERS ARE STORED IN ONE AREA (DCALC) IN ORDER
C TO PLOT THEM IN THE LPT.

      DO 420 J=1,ITER
      JI = (J-1) * NPAR1 + I
420   DCALC(J) = STRGP(JI)

      CALL PLOT1P(XN,DCALC,ITER,CHAR,0.,0.,0.,0.,1)
```

```
      CALL PLOTPR(6)
      WRITE(6,375) I
      WRITE(6,385) ((XN(KK),DCALC(KK)),KK=1,ITER)
425   CONTINUE
450   CALL PLOT1P(XN,ERRO,ITER,CHAR,0.,0.,0.,0.,1)
      CALL PLOTPR(6)
      WRITE(6,380)
      DO 460 I=1,ITER
460   WRITE(6,385) XN(I),ERRO(I)
      STOP
      END
```

```
C
C
C
      SUBROUTINE OPTMO(FUERA,AUTO)
C ***************************************************************
C *** THIS SUBROUTINE CALCULATES THE OPTIMUN INCREMENT  ********
C *** OF THE PARAMETERS TO  THE INPUT MODEL USING       ********
C *** RIDGE REGRESSION ESTIMATOR.                      ********
C ***  THE CALCULATIONS ARE SEQUENTIAL AND SOME SUBROU- ********
C *** TINES ARE NEEDED.:                               ********
C ***    1.- DERIV .......CALCULATES 'A'               ********
C ***    2.- MPACK .......MATRIX PACKAGE               ********
C ***    3.- PLOT.FOR.....PLOTS IN LPT                 ********
C ***    4.- CURVE.FOR....FORWARD SOLUTION(CATAL.INTERP.)********
C****                                                  ********
C ***************************************************************
C ***            JUAN CARLOS R.A.OVEJERO 1978 ,FEB.     ********
C ***************************************************************
C
C
C
      PARAMETER MNA=200,NPT=40,NPAR=5

      LOGICAL ISTOP,FUERA,AUTO


      LOGICAL ANSWER,FLAG


      DIMENSION ATRP(MNA),B(MNA),DIF(NPT),DELP(NPAR),STRGE(MNA),
     &  POLD(NPT)

      COMMON / BLOCK1 / OBS(NPT) ,T(NPT) ,P(NPAR) ,DCALC(NPT),
     &  A(MNA) ,IMAX ,IMIN,MNA1,NPT1,NPAR1,RTPLT

      COMMON / BLOCK2 / ERR,ERRO(15),ERRI(15),INTER,ITER,
     &  CONSK(10),XNEW,IFLAG


C THE CALCULATIONS WILL BE DONE SEQUENTIALLY;
C IN  FIRST PART WE END UP WITH VALUES OF ATRPA AND ATRPD
C LATER THROUGTH AN OPTIMIZATION LOOP A "GOOD" VALUE OF
C DELP WILL BE CALCULATED(WE'LL EMPROVE THE MODEL)
C
C
C ***************************************************************
C ******    CALCULATION OF THE MATRIX "A" (DERIVATIES)  ********
C ***************************************************************
C
C CALCULATE THE MATRIX DERIVATY  "A"
      FUERA=.FALSE.
```

```
      CALL DERIV(FUERA)
      IF(FUERA) RETURN
C DUE TO THE "BOUNDARY LIMITATIONS IN TIME , "A" WILL  HAVE
C ONLY "KA" ELEMENTS.
      KD=IMAX-IMIN+1
      KA=NPAR1*KD
      WRITE(4,312)
CP1   WRITE(15,315)
CP1   CALL PRINT(A,KA)
C ***************************************************************
C ******          TRANSPOSE OF "A"                        ********
C ***************************************************************
C
C NOW CALCULATE THE TRANSPOSE OF "A" AND STORAGE IT IN AN INDE-
C PENDENT AREA====> ATRP
C THE REAL DOUBLE DIMENSION OF THE MATRICES IS:
C       A(KD,NPAR1) <=====> ATRP(NPAR1,KD)
C THEN: THE A(MNA1) CONTAINS ONLY KA ELEMENTS
C
      CALL TRP1(A,NPAR1,KA,ATRP)
CP1   WRITE(15,320)
CP1   CALL PRINT (ATRP,KA)
C
C ***************************************************************
C  ******         TRANSPOSE OF "A"  X  "A"                ********
C ***************************************************************
C
C NOW CALCULATE THE PRODUCT OF THE MATRIX TRANSSPOSE BY "A"
C (ATRPA) AND STORAGE THE RESULT IN THE AREA "B".
C            ATRP(NPAR1,KD) * A(KD,NPAR1) = B(NPAR1,NPAR1)
C
C
      MNC=NPAR1*NPAR1
      CALL PMTRX(ATRP,KA,A,KA,B,MNC,KD)
CP1   WRITE(15,330)
CP1   CALL PRINT(B,MNC)
C
C ***************************************************************
C *****          DIFERENCE MATRIX (OBS - CALC)           ********
C ***************************************************************
C THE DIFERENCE MATRIX IS THE OBSERV.MINUS THE CALCULATED,FROM
C THE POINT IMIN UNTIL IMAX. THE RESULT WILL BE STORED IN THE
C FIRST KD ELEMENTS OF DIF.
      DO 20 I=1,KD
            KX=I-1+IMIN
            DIF(I)= OBS(KX) - DCALC(KX)
20    CONTINUE
CP1   WRITE(15,340)
CP1   DO 342 I=1,KD
CP1342 WRITE(15,341) DIF(I)
C
```

```
C ****************************************************************
C ******        TRANSPOSE OF "A" X  DIFERENCE MATRIX     ********
C ****************************************************************
C
C THE NEXT COMMON POINT BEFORE THE OPT. OF "K" WILL BE FIND THE
C PRODUCT  ATRP(NPAR1,KD) * DIF(KD) =ATRP(NPAR1)  AND STORE IT IN
C THE AREA OF  "A"(THAT IS NOT GOING TO BE USED)
C
      CALL PMTRX(ATRP,KA,DIF,KD,A,NPAR1,KD)
CP1   WRITE(15,350)
CP1   CALL PRINT(A,NPAR1)
C CALCULATE THE TRACE OF THE MATRIX ATRPA, (STORED IN "B")
      CALL TRACE (B,MNC,TRCE)
CP1   WRITE(15,355) TRCE


355   FORMAT(1X,20X,"TRACE=",F)
312   FORMAT(// 1X,20X,"IN THE SUBROUTINE OPTMO")
315   FORMAT(// 20X,"MATRIX OF DERIVATIES")
320   FORMAT(1X,20X,"TRANSPOSE OF A" )
330   FORMAT(1X,20X,"ATRPA")
340   FORMAT(1X,20X,"DIFERENCE MATRIX")
341   FORMAT(1X,40X,F)
350   FORMAT(1X,20X,"ATRP * DIF")

C
C
C THIS IS THE ERROR IN TEH LAST ITERATION (OUTER)
      OLDK=CONSK(INTER)
C THIS IS A REFERENCE VALUE WHEN DOING THE DECISSION OF NEXT "K"
C BOTH MATRICES CONSK AND ERRO ARE STORED IN COMMON AREA
      INTER=0
      NMIN=0
      ERRMIN=TRCE
C BEFORE  BEGINS THE LOOP WE'RE GOING TO STORAGE THE SET OF
C OLD PARAMETERS IN AN ARRAY POLD(I)
C
      DO 10 I=1,NPAR1
10    POLD(I) = P(I)
      IFLAG=0
C
C ****************************************************************
C ****************************************************************
C *******       OPTIMIZATION LOOP                        ********
C ****************************************************************
C ****************************************************************
C
900   CONTINUE
C
C
C
```

```
C
C THIS IS PART OF THE LOOP OF DECISSION OF NEW K ; WITH THE INF.
C CONTAINED IN ERRI(INTER) AND CONSK(INTER),  SUBROUTINE DECIS
C WILL FIND A"GOOD VALUE OF "K"

        IX=INTER
        IF(AUTO)CALL DECIS(ISTOP)
        IF(AUTO)WRITE(4,88) IFLAG,INTER,ERRI(INTER),XNEW
        IF(ISTOP) GO TO 70
        IF(AUTO) GO TO 125

        WRITE(4,360)
        ACCEPT 365, XNEW


365     FORMAT(F)
360     FORMAT(1X,'ENTER THE NEW VALUE OF K  ', $)
88      FORMAT(1X,5X,'SLOPE=',I3,5X,',ITERAT.=',I3,5X,
     &  'ERROR=',F7.4,5X,'NEW K =' ,F10.5)

125     INTER=INTER+1
C
C  ********************************************************************
C  ******         "ATRP = A + K*I"                                ********
C  ********************************************************************
C
C ADD TO THE DIAGONAL ELEMENTES THE: TRACE *XNEW
C AND  THE MATRIX ( ATRPA+KI)=====> STORED IN "B"
C
        XK=XNEW*TRCE
        DO 40 I=1,NPAR1
                INDEX=(I-1)*NPAR1+I
                B(INDEX)=B(INDEX)+XK
40      CONTINUE
C       CALL PRINT1(B,MNC)
C
C  ********************************************************************
C  ******         "(ATRP = A + K*I)**-1                           ********
C  ********************************************************************
C
C
C NOW INVERT THIS MATRIX==> (ATRPA+KI)**-1 AND STORAGE THE
C RESULT IN  "ATRP"
C
        CALL DESCAL (B,ATRP,NPAR1,MNA1)
C       CALL PRINT1(ATRP,MNC)
C
C  ********************************************************************
C  ******         INCREMENT OF THE PARAMETERS (DELP)              ********
C  ********************************************************************
C
```

```
C THE VALUES OF DELP CAN BE OBTAINED MULTIPLYING (ATRPA+KI)**-1
C (STORED IN "ATRP") BY ( ATRP * DIF) (STORED IN "A")
C
      CALL PMTRX(ATRP,MNC,A,NPAR1,DELP,NPAR1,NPAR1)
C
C NOW FIND THE NEW MODEL AND STORAGE IT IN THE STTORAGE MATRIX
C FOR A POSSIBLE RECOVERY.
C
C  POLD IS THE OLD MODEL WITH NO INCREMENT EFFECT
55    DO 60 I=1,NPAR1
            J=I+NPAR1*(INTER-1)
            X1 = 10.**(DELP(I))
            P(I)=POLD(I) * X1
            STORE(J)=P(I)
60    CONTINUE
C
C TRY TO FIND THE ERROR CORRESPONENT TO THE MODEL
C
      FUERA=.FALSE.
      CALL CALC(DCALC,T,NPT1,I1,I2,P,NPAR1,FUERA)
C
C IF THE FLAG FUERA IS TRUE, THEN THAT PARTICULAR CURVE IS NOT
C IN THE CATALOGUE, THEN WE DON'T WANT TO LOOSE THE INFORMATION
C OF PREVIOUS INNER ITERATIONS(INTER) ; SO:
C
      IF((FUERA).AND.(INTER.EQ.1))  RETURN
      IF(FUERA) INTER=INTER-1
      IF(FUERA) WRITE(4,65)
      IF(FUERA) GO TO 80

65    FORMAT(1X,'***THE PROGRAM WILL STOP OPTIMIZATION OF THE',
     &  ' RIDGE TRACE ****'/)
C THE ERROR
C
      CALL ERROR(DCALC,OBS,NPT1,I1,I2,ERR)
      KNEW=(I2-I1+1)
      ERRI(INTER)=ERR
      CONSK(INTER)=XNEW
C
C FLAG TO LOCATE THE MINIMUN OF THE CURVE" ERROR=ERROR(XNEW)"
C
      IF(ERRMIN.LT.ERR) GO TO 25
      NMIN=INTER
      ERRMIN=ERR
25    CONTINUE
C
C THESE STMT ARE WRITTEN TO PROVIDE ENOUGH INFORM.  WHEN
C WORKING AS AN ITERATIVEE SYSTEM
C
      IF(AUTO) GO TO 130
```

```
      WRITE(4,200) INTER
      WRITE(4,210) TRCE,XNEW,ERR,ERRMIN,ERRO(ITER)
      WRITE(4,220)

      DO 230 I=1,NPAR1
      WRITE(4,240) POLD(I),DELP(I),P(I)
230   CONTINUE

C IN CASE THAT WE WANT TO STOP INTER ITERATIONS
C WE' LOOK FOR THE MINIMUN ERROR MODEL
C
      IF(AUTO) GO TO 130

      WRITE(4,85)
      IF (ANSWER(DUMMY)) GO TO 70
130   CONTINUE

200   FORMAT(1X,15X,' OPTIMIZATION ITERATION  # ', I2)
210   FORMAT(1X,10X,' TRACE  ',F7.2,15X,'  K= ',F7.4,/,10X,
     &  ' ERROR =',F7.4,8X,'MIN ERROR=',F7.4,5X,'LAST ERROR',F7.4)
220   FORMAT(1X,15X,'OLD PARAMETERS',10X,'DELP',10X,'NEW PARAM.')
240   FORMAT(1X,15X,3(F10.4,5X))
85    FORMAT(1X,//,'FINISH OPTIMIZATION OF THE RIDGE TRACE? ',$)


C
C IF CONTINUE THE ITERATIONS, LET'S RECOVER THE MODEL
C

      DO 37 I=1,NPAR1
            INDEX=(I-1)*NPAR1+I
            B(INDEX)=B(INDEX)-XK
37    CONTINUE
      GO TO 990

C
C ***************************************************************
C ***************************************************************
C *****          OPTIMIZATION LOOP (LOWER LIMIT)          ********
C ***************************************************************
C ***************************************************************
C
C

70    CONTINUE
      IF(RTPLT) GO TO 75
      GO TO 78
75    CONTINUE
      CHAR=1H*
      CALL PLOT1P(CONSK,ERRI,INTER,CHAR,0.,0.,0.,0.,1)
      CALL PLOTPR(4)
```

```
78      CONTINUE
        IF(NMIN.EQ.INTER) RETURN
C  THE MINIMUN ERROR WILL DEFINE THE BEST MODEL FOR THIS ITER-
C ATION,
30      CONTINUE
        FUERA=.FALSE.
        ERR = ERRI(NMIN)
        XNEW=CONSK(NMIN)
C
C NOW ,THE CORRESPONDENT MODEL WILL BE:
C
        ISEA=(NMIN-1)*NPAR1

C       WRITE(4,45) ISEA
45      FORMAT(1X,20X,'MATRIX OF STORAGE OF PARM',5X,'ISEA=',I2)
        IJ=INTER*NPAR1
C       CALL PRINT1(STORE,IJ)
        DO 90 I=1,NPAR1
                P(I)=0.0
                K=I+ISEA
                P(I)=STRGE(K)
90      CONTINUE
C IT WILL RETTURN WITH THE VALUES OF THE ERROR (ERR),MODEL(P(I))
C
        RETURN
        END
```

```
      SUBROUTINE DECIS(ISTOP)
C ***************************************************************
C ******              SUBROUTINE DECIS                         *******
C ******  THIS ROUTINE CALCULATES THE OPTIMUN VALUE OF THE******
C ******  RIDGE PARAMETER THAT CORRESPONDS AT EACH ITERAT.******
C ******  THE ALGORITHM OF THE CAN BE FOUND IN T-2047      *******
C ******  (C.S.M.). THIS ALGORITHM IS BASED MAINLY IN      *******
C ******  THE CALCULATION OF THE SLOPE OF THE CURVE AT     *******
C ******  EACH PARTICULAR POINT,TO DETERMINE THE MINIMUM.*******
C ******    THE VALUES OF THE STATING POINT (START) AND    *******
C ******  THE INCREMENT OF THE R.P. (CINC) CAN BE MODIFIED.*****
C ***************************************************************
C ******    ORIGINAL PROGRAMER JUAN CARLOS R.OVEJERO,1978 ******
C ***************************************************************
C
C
C
      LOGICAL ISTOP
      COMMON / BLOCK2 / ERR,ERRO(15),ERRI(15),INTER,ITER,
     &   CONSK(10),XNEW,IFLAG
      IF(INTER-2)10,20,30
10    CONTINUE
C
C THIS IS CONCERN WITH THE CERO AND FIRST ITERATINS
C
      ISTOP=.FALSE.
C START IS AN STATISTICAL POINT CLOSE TO THE AVERAGE MINIMUN
C "CINC" IS THE INCREMENT FOR EACH VALUE OF "XNEW"
      START=0.2
C     IF(ITER.EQ.1)START=0.5
      CINC=0.3
      C1=CINC**INTER
      XNEW=C1*START
C NOW BACK TO THE MAIN WITH VALUES OF "ISTOP" AND "XNEW"
      RETURN

20    CONTINUE
C
C FOR THE THIRD ITERATION WE NEED TO COMPARE THE PREVIOUS TWO
C AND DECEIDE WHICH DIRECTION TAKE BASED IN THE SLOPE OF THE
C CURVE: " ERROR= ERROR(XNEW)"
      ISTOP=.FALSE.
      ERRINC=ERRI(INTER-1)-ERRI(INTER)
C
C ERRINC GIVES THE SIGN OF THE SLOPE
C
      IF(ERRINC) 23,26,29
23    CONTINUE
C
C NEGATIVE SLOPE,THEN TAKE BIGGER VALUES OF"XNEW" TO FIND MINIMUN.
C IFLAG MEANS THE OLD VALUE OF THE SLOPE
```

```
      IFLAG=-1
      XNEW=CONSK(1)*1.5
      RETURN

26    CONTINUE
C
C FOR EQUALVALUES OF THE LAST TWO ERRORS THE MINIMUN OF "XNEW"
C WILL BE BETWEEN THE CORRESPONDENT VVALUES OF "XNEW".
C
      IFLAG=0
      A=CONSK(1)-CONSK(2)
      XNEW=A*0.25+CONSK(2)
      RETURN
29    CONTINUE
C FOR POSITIVE SLOPE IN THIS ZONE OF CURVE
C
      IFLAG=1
      XNEW=CONSK(2)*0.3
      RETURN
30    CONTINUE
C CHEK FOR MAX NUMBER OF ITERATIONS
      ITRMAX=8
      IF(INTER.GE.ITRMAX) GO TO 35
C   IF THE ERROR HAS DECREASED ENOUGH THEN STOP "INTER"
C
      PRCMAX=0.75
C     ERRINC=ERRO(ITER)-ERRMIN
C     DECRMT=ERRINC/ERRO(ITER)
C     IF((DECRMT.GT.PRCMAX).AND.(ITER.GT.1)) GO TO 35
      IJ=INTER-1
      ERRINC=ERRI(INTER)-ERRI(IJ)
      X=ERRINC*1000.
      KX=IFIX(X)
      ERRINC=FLOAT(KX)/1000.
      IF(IFLAG)40,50,60
40    CONTINUE
C THE PREVIOUS SLOPE WAS NEGATIVE,THEN CHECK WITH LAST TO DETERMINE
C POSSIBLE MINIMUN
      IF(ERRINC)36,38,35
50    CONTINUE
C THE PREVIOUS SLOPE CERO, THEN CHECK IN MIDDLE
      IF(ERRINC.GT.0.0) WRITE(4,120)
120   FORMAT(1X,20X,' CHECK "XNEW",MAXIMUN IN CURVE')
      GO TO 35
60    CONTINUE
C THE PREVIOUS SLOPE WAS POSITIVE
      IF(ERRINC) 37,38,35
35    CONTINUE
      ISTOP=.TRUE.
      RETURN
36    CONTINUE
```

```
      ISTOP=.FALSE.
      IFLAG=-1
      XNEW=CONSK(INTER)/CINC
      RETURN
37    CONTINUE
      ISTOP=.FALSE.
      IFLAG=1
      XNEW=CONSK(INTER)*CINC
      IF(XNEW.GT.1.) XNEW=1.
      RETURN
38    CONTINUE
      ISTOP=.FALSE.
      IFLAG=0
      DELCTE= (CONSK(IJ)-CONSK(INTER)) *0.5
      XNEW=CONSK(INTER)+DELCTE
      RETURN
      END
```

```
      SUBROUTINE CALC(PA,T,NPT,I1,I2,P,NPAR,FUERA)
C ***************************************************************
C ******          SUBROUTINE CALC                          ********
C ****** THIS SUBROUTINE CALCULATES FOR AN INPUT EARTH     ********
C ****** MODEL, THE CORRESPONDENT VALUES OF THE APPAR.     ********
C ****** RESISTIVITY CURVE, IN REAL TIME.                  ********
C ******          INPUT:                                   ********
C ******            PARAMETERS...RHO1,RHO2,H1,H2,R         ********
C ******            REAL TIME ... T(I),I=1,NPT             ********
C ******          OUTPUT:                                  ********
C ******            APP.RES ......PA(I)                    ********
C ******            I1,I2, ....... LIMITS                  ********
C ****** WHERE THE VALUES OF PA(I) ARE ONLY CALCULATED     ********
C ****** FOR THE RANGE OF VALUES   (I1.LE.I.GE.I2)         ********
C ****** DUE TO THE DIFFERENCE OF THE APP.RES.CURVE        ********
C ****** TIMES AND THE REAL TIME (T(I)).                   ********
C ****** DURING THEE EXECUTION IT WILL CALL THE SUBROUT.   ********
C ****** CURVE,AND THE FUNCTION XINTP.                     ********
C ***************************************************************
C ******          JUAN CARLOS RODRIGUEZ, FEB,1978          ********
C ***************************************************************
C
C
      PARAMETER N=35
      COMMON/BLOCK3/ DRBH,DR2BR,DH2BH
      LOGICAL FUERA
      DIMENSION PA(NPT),T(NPT),P1(N),TT1(N),P(NPAR)
      IF(NPAR.EQ.3)GO TO 3
      RHO1=P(1)
      H1=P(2)
      RHO2=P(3)
      H2=P(4)
      R=P(5)
      DR2BR=RHO2/RHO1
      DH2BH=H2/H1
      DRBH =R/H1
      GO TO 5
3     CONTINUE
      RHO1=10.
      H1=1000.
      DR2BR=P(1)
      DH2BH=P(2)
      DRBH =P(3)
5     CONTINUE
      DELT=0.08
      FUERA=.FALSE.
      CALL CURVE(P1,T1,N1,FUERA)
C THIS FLAG *FUERA WILL COMUNICATE TO THE CALLING ROUTINE THAT
C THE ESTIMATION OF PARAMETERS IS OUT OF THE PROPER RANGE
C (THE RANGE THAT THERE EXISTS IN THE CATALOGUE)
      IF(FUERA) RETURN
```

```
C       CALL TSPLOT(P1,N1,2,N)
C       WRITE(4,250) T1,N1
250     FORMAT(1X,/,10X,"T1=",F7.3,5X,"N1=",I3,/)
C       WRITE(4,255)
255     FORMAT(1X,10X,"APP.RES.CURVE(THEOR.)")
C       WRITE(4,260) P1
260     FORMAT(5X,5F)
        K=1
        I1=1
        I2=NPT
        DO 10 I=1,N1
                X=I-1
                Z=X*DELT
                TT1(I)=T1+Z
10      CONTINUE
        DO 50 I=1,NPT
                TAU=0.0
                IF(T(I).EQ.0.)GO TO 9
                TAU= 3.899089934+0.5*ALOG10(RHO1)+ALOG10(T(I))*0.5-
     &          ALOG10(H1)
C THERE WILL BE A VALUE OF FORMULA(TIME PARAMETER)
                IF(TAU.LT.TT1(1))GO TO 9
                IF(TAU.GT.TT1(N1)) GO TO 8
                PA(I)=XINTP(TAU,P1,TT1,N1,K)
                PA(I)=PA(I)+ALOG10(RHO1)
                GO TO 50
8               I2=I2-1
                GO TO 50
9               I1=I1+1
50      CONTINUE
C       CALL TSPLOT(PA,NPT,2,NPT)
        RETURN
        END
```

```
      FUNCTION XINTP(TAU,P1,T1,N1,K)
C ***************************************************************
C ******      FUNCTION INTERPOLATION                          ********
C ******  THIS FUNTION DOES A LINEAR INTERPOLATION FOR        ********
C ******  THE VALUE OF TAU WHEN THE CURVE P1(T1) IS DE-       ********
C ******  FINED WITH A SET OF N1 POINTS.                      ********
C ******  FIRST, FINDS THE INTERVAL OF TIMES WHERE TAU        ********
C ******  IS INCLUDED, AND THEN DOES A LINNEAR INTERPO        ********
C ******   LATION .                                           ********
C ******         P1(N1).......VECTOR CONT.VALUES OF FUNC.*********
C ******         T1(N1).......CORRESP. TIMES                 ********
C ******         TAU  ........ INTEERP TIME                  ********
C ******         K  ..........FLAG  (LOC OF LAST INTERP.)*********
C ******                J.C.R.1978                           ********
C ***************************************************************
C
      LOGICALC FLAG
      DIMENSION P1(N1),T1(N1)
      FLAG=.TRUE.
      DO 20 J=K,N1
            IF((TAU.GE.T1(J)).AND.(TAU.LE.T1(J+1)))FLAG=.FALSE.
            IF(FLAG) GO TO 20
            DELT=T1(J+1)-T1(J)
            DELP=P1(J+1)-P1(J)
            TINC=TAU - T1(J)
            SLOPE = DELP/DELT
            PINC = SLOPE * TINC
            XINTP = P1(J) + PINC
C           K=J
C           THE VALUE OF K WILL GO BACK TO THE MAIN PROG WITH
C           THIS VALUE.
            RETURN
20    CONTINUE
      STOP ' TAU OUT OF THE TIMES IN CATALOG CURVE'
      END
```

```
      SUBROUTINE CURVE(P1,T1,N1,FUERA)
C ***************************************************************
C ******    SUBROUTINE CURVE                                    ********
C ****** THIS SUBROUTINE WILL CALCULATE THE APPARENT           ********
C ****** RESISTIVITY CURVE CORRESPONDENT TO A GIVEN SET        ********
C ****** OF PARAMETERS (PARAM.,APPA.RES. AND TIME HAVE        ********
C ****** BEEN NORMALIZED).                                    ********
C ******          INPUT:                                      ********
C ******                 DR8H..... R/H1                       ********
C ******                 DR2BR..... R2/R1                     ********
C ******                 DH2BH..... H2/H                      ********
C ******         OUTPUT:                                      ********
C ****** P1(N1)..... VECTOR CONTAINING N1 VALUES OF NOR       ********
C ******              MALIZED APPARENT RESISTIVITY.           ********
C ****** T1 ........ ORIGEN OF NORMALIZED TIMES FOR THE       ********
C ******     APP.RES. CURVE/  T1 <===>P1(1)                   ********
C ****** THE TIME CORRESPONDENT TO THE VALUES OF THE APP      ********
C ****** RES.( T(I) <====> P1(I)) CAN BE OBTAINED:            ********
C ******          T(I)=T1+(I-1)*DELT                          ********
C ****** WHERE DELT IS THE SAMPLE INTERVAL FOR ALL THE        ********
C ****** CURVES OF THECATALOG CONTAINED IN THE FILE.          ********
C ****** OF READING.DURING THE EXCUTION IT WILL READ 8        ********
C ****** CURVES FROM FOR02.DAT(RANDOM ACCESS FILE CONT        ********
C ****** AINING THE CURVE CATALOG); AND WILL CALL THE         ********
C ****** SUBROUTINE FIND.                                     ********
C ******                 JUAN CARLOS RODRIGUEZ                ********
C ***************************************************************
C
C
C THE PARAMETERS WILL BE:
C               NPNT.....# OF POINTS IN EACH CURVE(MAXIMUN)
C               NPAR1.....# OF R/1 PARAM. IN THE CATALOG
C               NPAR2.....#    R2/R1    ""
C               NPAR3.....#    H2/H1    ""
C
      PARAMETER NPNT=35,NPAR1=10,NPAR2=4,NPAR3=5
      COMMON/BLOCK3/ DR8H,DR2BR,DH2BH
      DIMENSION P1(NPNT),CURVEA(NPNT),CURVEB(NPNT),CURVEC(NPNT),
     1 CURVED(NPNT),CURVEE(NPNT),CURVEF(NPNT),CURVEG(NPNT),
     2 CURVEH(NPNT),TREF(NPAR1,NPAR2,NPAR3),R2BR(NPAR2),R8H(NPAR1),
     3 H2BH(NPAR3)
      LOGICAL NEVER,LX,LY,LZ,FUERA
C
C  THE VALUE OF DELT DEPPENDS OF THE RECORD SAMPLE INTERVAL
C
      FUERA=.FALSE.
      DELT=0.08
```

```
C ***************************************************************
C ******  IREF MATRIX CONTAINS THE # OF LOCATION OF THE ********
C ******  SPECIFIC CURVE RECORD IN THE BINARY-RANDOM    ********
C ******  ACCESS FILE.                                  ********
C ******        I.......  R8H(I) = DATA R/H1 OF RECORD  ********
C ******        J.......  R2BR(J)=   "  R2/R1    "      ********
C ******        K.......  H2BH(K)=   "  H2/H1    "      ********
C ******                                                ********
C ******              ILOC=IREF(I,J,K)                  ********
C ******        READ(IDS#ILOC)=> RECORD                 ********
C ***************************************************************
C
C
      DATA(((IREF(I,J,K),I=1,10),J=1,4),K=1,5) /
     &  01,02,03,04,05,00,00,00,00,00,
     &  06,07,08,09,10,00,00,00,00,00,
     &  11,12,13,14,15,00,00,00,00,00,
     &  16,17,18,19,20,00,00,00,00,00,
     &  00,21,22,23,24,25,00,00,00,00,
     &  00,26,27,28,29,30,00,00,00,00,
     &  00,31,32,33,34,35,00,00,00,00,
     &  00,36,37,38,39,40,00,00,00,00,
     &  00,00,41,42,43,44,45,00,00,00,
     &  00,00,46,47,48,49,50,00,00,00,
     &  00,00,51,52,53,54,55,00,00,00,
     &  00,00,56,57,58,59,60,00,00,00,
     &  00,00,00,00,61,62,63,64,65,00,
     &  00,00,00,00,66,67,68,69,70,00,
     &  00,00,00,00,71,72,73,74,75,00,
     &  00,00,00,00,76,77,78,79,80,00,
     &  00,00,00,00,00,81,82,83,84,85,
     &  00,00,00,00,00,86,87,88,89,90,
     &  00,00,00,00,00,91,92,93,94,95,
     &  00,00,00,00,00,96,97,98,99,100/
      DATA NEVER/.TRUE./
18    FORMAT(3F)
      WRITE(18,18)DR8H,DR2BR,DH2BH
      IDS=2
      FLNAM='FOR02.DAT'
      IF(NEVER) OPEN(UNIT=IDS,ACCESS='RANDOM',MODE='BINARY',
     1  DISPOSE='SAVE',FILE=FLNAM,PROTECTION="177,RECORD SIZE=40,
     1  ASSOCIATE VARIABLE=JDS)
      NEVER = .FALSE.
```

```
C
C
C  ********************************************************************
C  *****  READ VECTORS CONTAINING POSSIBLE PARAMETERS IN ********
C  *****  CATALOGUE OF CURVES,LOOK FOR TWO INDEX  THAT   ********
C  *****  WILL LOCATE OUR DATA IN VECTOR(IMAX,IMIN)      ********
C  *****                                                 ********
C  ********************************************************************
C
C
        READ(IDS#1)R2BR
        CALL FIND (J1,J2,R2BR,NPAR2,DR2BR,FUERA)
        IF(FUERA)RETURN
        READ(IDS#2)H2BH
        CALL FIND (K1,K2,H2BH,NPAR3,DH2BH,FUERA)
        IF(FUERA)RETURN
        READ(IDS#3)RBH
        CALL FIND (I1,I2,RBH,NPAR1,DRBH,FUERA)
        IF(FUERA)RETURN
C       WRITE(4,115) I1,I2,J1,J2,K1,K2
115     FORMAT(1X,10X,'VALUES OF IMIN AND IMAX, J,K',6I)
C

C
C  ********************************************************************
C  *****  FOR EACH OF THE 8 CURVES IN THE INTERPOLATION  ********
C  *****          1.- ...LOCATE IT IN IREF MATRIX        ********
C  *****          2,- ...READ ITS VALUES FROM FOR02.DAT  ********
C  *****          3.- ...CHECK FOR MISTAKES IN PARAM.    ********
C  *****                                                 ********
C  *****  TO FIND THE INTERVAL OF TIME COMMON TO THE 8   ********
C  *****  CURVES, THE LARGESTVALUE OF "T1" WILL BE       ********
C  *****  DETERMINE  (T1MAX).                            ********
C  ********************************************************************
C
C
C   EACH OF THE RECORDS CONTAINS:
C        R2/R1;H2/H1;R/H1;T1;N1;CURVE
C    WHERE:
C        (R2/R1,H2/H1,R/H1) ..VALUES OF THE PARAMETERS FOR RECORD
C        CURVE(N1).............N1 VALUES OFAPP.RES.
C
C        T1 ....................NORM.TIME FOR FIRST VALUE OF CURVE
C        N1 ....................# OF POINTS IN THE CURVE
        T1MAX=0.
C IF ONE OF THE CURVES IS NOT IN THE CATALOGUE
C THE EXECUTION WILL COME BACK TO THE MAIN PROG. FOR A PROPER
C ACTION, AFTER PRINTING THE VALUESOF PARAM.
C FOR THE "A" CURVE
        FUERA=.FALSE.
        ILOC = IREF (I1,J1,K1) + 3
```

```
      IF ( ILOC.EQ.3)FUERA=.TRUE.
      IF(FUERA) CALL FALLO
      IF(FUERA) RETURN
      READ (IDS#ILOC) DR2BRA,DH2BHA,DRBHA,T1A,N1A,CURVEA
      IF (( DR2BRA .NE. R2BR(J1)) .OR. (DH2BHA.NE.H2BH(K1))
     1  .OR.(DRBHA.NE.RBH(I1)))STOP'FALLO IN DATA PARM IN FILE'
      IF (T1A .GT. T1MAX) T1MAX=T1A
C     CALL RITEC( DR2BRA,DH2BHA,DRBHA,T1A,N1A,CURVEA,T1MAX)
C
C THE SAME FOR "B"
      FUERA=.FALSE.
      ILOC = IREF (I2,J1,K1) + 3
      IF ( ILOC.EQ.3)FUERA=.TRUE.
      IF(FUERA) CALL FALLO
      IF(FUERA)  RETURN
      READ (IDS#ILOC) DR2BRB,DH2BHB,DRBHB,T1B,N1B,CURVEB
      IF (( DR2BRB .NE. R2BR(J1)) .OR. (DH2BHB.NE.H2BH(K1))
     1  .OR.(DRBHB.NE.RBH(I2)))STOP'FALLO IN DATA PARM IN FILE'
      IF (T1B .GT. T1MAX) T1MAX=T1B
C     CALL RITEC( DR2BRB,DH2BHB,DRBHB,T1B,N1B,CURVEB,T1MAX)
C THE SAME FOR "C"
      FUERA=.FALSE.
      ILOC = IREF (I1,J2,K1) + 3
      IF ( ILOC.EQ.3)FUERA=.TRUE.
      IF(FUERA) CALL FALLO
      IF(FUERA) RETURN
      READ (IDS#ILOC) DR2BRC,DH2BHC,DRBHC,T1C,N1C,CURVEC
      IF (( DR2BRC .NE. R2BR(J2)) .OR. (DH2BHC.NE.H2BH(K1))
     1  .OR.(DRBHC.NE.RBH(I1)))STOP'FALLO IN DATA PARM IN FILE'
      IF (T1C .GT. T1MAX) T1MAX=T1C
C     CALL RITEC( DR2BRC,DH2BHC,DRBHC,T1C,N1C,CURVEC,T1MAX)
C THE SAME FOR "D"
      FUERA=.FALSE.
      ILOC = IREF (I2,J2,K1) + 3
      IF ( ILOC.EQ.3) FUERA=.TRUE.
      IF(FUERA) CALL FALLO
      IF(FUERA) RETURN
      READ (IDS#ILOC) DR2BRD,DH2BHD,DRBHD,T1D,N1D,CURVED
      IF (( DR2BRD .NE. R2BR(J2)) .OR. (DH2BHD.NE.H2BH(K1))
     1  .OR.(DRBHD.NE.RBH(I2)))STOP'FALLO IN DATA PARM IN FILE'
      IF (T1D .GT. T1MAX) T1MAX=T1D
C THE SAME FOR "E"
      FUERA=.FALSE.
      ILOC = IREF (I1,J1,K2) + 3
      IF ( ILOC.EQ.3)FUERA=.TRUE.
      IF(FUERA) CALL FALLO
      IF(FUERA) RETURN
      READ (IDS#ILOC) DR2BRE,DH2BHE,DRBHE,T1E,N1E,CURVEE
      IF (( DR2BRE .NE. R2BR(J1)) .OR. (DH2BHE.NE.H2BH(K2))
     1  .OR.(DRBHE.NE.RBH(I1)))STOP'FALLO IN DATA PARM IN FILE'
      IF (T1E .GT. T1MAX) T1MAX=T1E
```

```
C THE SAME FOR "F"
        FUERA=.FALSE.
        ILOC = IREF (I2,J1,K2) + 3
        IF ( ILOC.EQ.3)FUERA=.TRUE.
        IF(FUERA) CALL FALLO
        IF(FUERA) RETURN
        READ (IDS#ILOC) DR2BRF,DH2BHF,DRBHF,T1F,N1F,CURVEF
        IF (( DR2BRF .NE. R2BR(J1)) .OR. (DH2BHF.NE.H2BH(K2))
     1  .OR.(DRBHF.NE.RBH(I2)))STOP'FALLO IN DATA PARM IN FILE'
        IF (T1F .GT. T1MAX) T1MAX=T1F
C THE SAME FOR "G"
        FUERA=.FALSE.
        ILOC = IREF (I1,J2,K2) + 3
        IF ( ILOC.EQ.3)FUERA=.TRUE.
        IF(FUERA) CALL FALLO
        IF(FUERA) RETURN
        READ (IDS#ILOC) DR2BRG,DH2BHG,DRBHG,T1G,N1G,CURVEG
        IF (( DR2BRG .NE. R2BR(J2)) .OR. (DH2BHG.NE.H2BH(K2))
     1  .OR.(DRBHG.NE.RBH(I1)))STOP'FALLO IN DATA PARM IN FILE'
        IF (T1G .GT. T1MAX) T1MAX=T1G
C THE SAME FOR "H"
        FUERA=.FALSE.
        ILOC = IREF (I2,J2,K2) + 3
        IF ( ILOC.EQ.3) FUERA=.TRUE.
        IF(FUERA) CALL FALLO
        IF(FUERA) RETURN
        READ (IDS#ILOC) DR2BRH,DH2BHH,DRBHH,T1H,N1H,CURVEH
        IF (( DR2BRH .NE. R2BR(J2)) .OR. (DH2BHH.NE.H2BH(K2))
     1  .OR.(DRBHH.NE.RBH(I2)))STOP'FALLO IN DATA PARM IN FILE'
        IF (T1H .GT. T1MAX) T1MAX=T1H
C       CALL RITE( DR2BRH,DH2BHH,DRBHH,T1H,N1H,CURVEH,T1MAX)
C
C ***************************************************************
C ******  DETERMINE THE NUMBER OF COMMON POINTS AFTER   ********
C ******  T1MAX,THAT WILL DEFINE THE COMMON INTERVAL OF ********
C ******  THE 8 CURVES.====> T1MAX,NMIN===> INTERVAL    ********
C ***************************************************************
C
C
        NMIN=NPNT
        KA= ( T1MAX-T1A ) / DELT
        NRES = N1A - KA
        IF ( NMIN .GT. NRES )  NMIN = NRES
        KB= ( T1MAX-T1B ) / DELT
        NRES = N1B - KB
        IF ( NMIN .GT. NRES )  NMIN = NRES
        KC= ( T1MAX-T1C ) / DELT
        NRES = N1C - KC
        IF ( NMIN .GT. NRES )  NMIN = NRES
        KD= ( T1MAX-T1D ) / DELT
        NRES = N1D - KD
```

```
IF ( NMIN .GT. NRES )  NMIN = NRES
KE= ( T1MAX-T1E ) / DELT
NRES = N1E - KE
IF ( NMIN .GT. NRES )  NMIN = NRES
KF= ( T1MAX-T1F ) / DELT
NRES = N1F - KF
IF ( NMIN .GT. NRES )  NMIN = NRES
KG= ( T1MAX-T1G ) / DELT
NRES = N1G - KG
IF ( NMIN .GT. NRES )  NMIN = NRES
KH= ( T1MAX-T1H ) / DELT
NRES = N1H - KH
IF ( NMIN .GT. NRES )  NMIN = NRES
```

```
C ***************************************************************
C ****** INTERPOLATION IN THREE DIMENSION USING THE      ********
C ****** FINITE ELEMENTS METHOD,WITH THE EQ.:            ********
C ******    V(X,Y,Z)=A1+A2 X+A3 Y+A4 XY+A5 Z+A6 YZ      ********
C ******              +A7 XZ + A8 XYZ                    ********
C ******                                                 ********
C ****** THAT DESCRIBES THE VALUE OF THE "V" FUNCTION    ********
C ****** FOR ANY POINT INSIDE THE VOLUME:                ********
C ******                                                 ********
C ******             G I-------------I H                 ********
C ******               I               I                 ********
C ******       E I-------------I F   I                   ********
C ******         I     I         I     I                 ********
C ******         I     I         I     I                 ********
C ******         I  C I--------I----I D  ^ (K-Z)         ********
C ******                  I                I I          ********
C ******       A I-------------I B          I            ********
C ******                                    I-----> (I-X) ********
C ******                           (J-Y)                 ********
C ******                                                 ********
C ****** WHERE THE VALUES OF THE "A" COEFICIENTS ARE     ********
C ****** DETERMINE USING OUR KNOWLEDGE OF THE FUNCTION   ********
C *******IN THE 8 CORNERS(A,B,C,...),(BOUNDARY COND.)    ********
C ***************************************************************
C INTERPOLATION  USING THE FINITE ELEMENTS
C
C TRASLATION OF THE AXIS TO THE A(IMIN,JMIN,KMIN) POINT
      X= R8H(I2) - RBH(I1)
      Y= R2BR(J2) -R2BR(J1)

      Z= H2BH(K2) - H2BH(K1)
C THE VALUE OF THE LOGICALC VARIABLES LX,LY,LZ WILL PRVENT FROM
C POSSIBLE DIVISION BY CERO WHEN CALCULATING A1,...A8
      LX=X.NE.0.0
      LY=Y.NE.0.0
      LZ=Z.NE.0.0
C COORDENATES OF THE DATA POINT IN THE NEW AXIS
      XP= DRBH - RBH(I1)
      YP= DR2BR - R2BR(J1)
      ZP= DH2BH - H2BH(K1)
      DO 50 I=1,NMIN
            M1 = I+KA
            M2 = I+KB
            M3 = I+KC
            M4 = I+KD
            M5 = I+KE
            M6 = I+KF
            M7 = I+KG
            M8 = I+KH
            VA=CURVEA(M1)
```

```
                    VB=CURVEB(M2)
                    VC=CURVEC(M3)

                    VD=CURVED(M4)
                    VE=CURVEE(M5)
                    VF=CURVEF(M6)
                    VG=CURVEG(M7)
                    VH=CURVEH(M8)
C THE FINITE ELEMENTS FINDS THE SOLUTION OF THE LINEAR EQ :
C V(X,Y,Z)=A1+A2*X+A3*Y+A4X*Y+A5*Z+A6*Y*Z+A7*X*Z+A8*XYZ
                    A1=V1
                    IF(LX) A2=(VB-A1)/X
                    IF(LY) A3=(VC-A1)/Y
                    CST= VD-A1-A2*X-A3*Y
                    CSX= X*Y
                    IF(LX.AND.LY) A4=CST/CSX
                    IF(LZ) A5=(VE-A1)/Z
                    CS3= VF-A1-A2*X-A5*Z
                    CS4=X*Z
                    IF(LX.AND.LZ) A7=CS3/CS4
                    CS5= VG-A1-A3*Y-A5*Z
                    CS6= Y*Z
                    IF(LY.AND.LZ) A6= CS5/CS6
                    CS7= VH-A1-A2*X-A3*Y-A4*X*Y
                    CS8= -A5*Z-A6*Y*Z-A7*X*Z
                    CS9= X*Y*Z
                    IF(LX.AND.LY.AND.LZ) A8 = (CS7+CS8)/CS9

C THEN THE VALUE OF THE FUNTION IN THE LOCAITION WILL
C BE : P(XP,YP,ZP)  ====>   VP= P1(I)
                    TOT=A1+A2*XP+A3*YP+A4*XP*YP
                    TOT1= A5*ZP+A6*YP*ZP+A7*XP*ZP
                    TOT2= A8*XP*YP*ZP
                    P1(I)= TOT+TOT1+TOT2
50          CONTINUE
            T1=T1MAX
            N1=NMIN
C           WRITE(4,100)T1,N1
C           WRITE(4,200) P1
100         FORMAT(1X,10X,F,5X,I,/)
200         FORMAT(5X,5F)
            RETURN
            END
```

```
      SUBROUTINE FIND(IMIN,IMAX,A,N,A1,FUERA)
C ***************************************************************
C ******          SUBROUTINE   FIND                          ********
C ******    THIS SUBR. POSITION A SCALCAR IN A VECTOR        ********
C ****** CONTAINING THE POSSIBLE VALUES OF THE PARAME-       ********
C ****** TER (SCALCAR).THE COMPONENTS OF THE VECTOR           ********
C ****** MUST BE ORDERED (INCREASING OR DECREASING).         ********
C ******                                                     ********
C ******       THE ARGUMENTS:                                ********
C ******         A(N)........ REFERENCE VECTOR               ********
C ******         A1 ......... SCALCAR TO LOCATE              ********
C ******         IMAX ....... UPPER LOCATION INDEX           ********
C ******         IMIN ....... LOWER    "       "             ********
C ******                                                     ********
C ******  IF THE VALUE IS OUT OF THE RANGE INCLUDED IN       ********
C ******  THE VECTOR, IT WILL PRINT A MESSAGE.               ********
C ******  IF THE SCALCARS  WITHIN THE 10% OF THE VALUE       ********
C ******  OF APOINT IN THE MATRIX ,THAT WILL BE THE ONLY     ********
C ******  LOCATION (IMAX=IMIN).                             ********
C ******                                                     ********
C ******          JUAN CARLOS RODRIGUEZ. FEB 1978.          ********
C ***************************************************************
C
C
      DIMENSION A(N)
      LOGICAL FUERA
      FUERA=.FALSE.
C THIS THE 10% OF CONFIDENCE INTERVAL
C     EPS=0.01
C XK IS A FLAG TODIFF. INCREAS. FROM DECREAS. SECUENCES
      XK=1.
      IF(A(2).LT.A(1))  XK=-1.
      EPS=EPS*XK
      DO 30 I=1,N
            AMIN=0.
            AMAX=0.
            BMIN=0.
            IX=I
            AMIN=A(I)*(1.- EPS)
            AMAX=A(I)*(1+ EPS)
            BMIN=A(I+1)*(1.- EPS)
            DTA=ABS(AMAX-AMIN)
            DMAXA=ABS(AMAX-A1)
            DMINA=ABS(AMIN-A1)
            IF((DMAXA.LE.DTA).AND.(DMINA.LE.DTA)) GO TO  40
            IF(I.EQ.N) GO TO 30
            DTB=ABS(AMAX-BMIN)
            DMINB=ABS(A1-BMIN)
            IF((DMAXA.LE.DTB).AND.(DMINB.LE.DTB))GO TO 50
30    CONTINUE
      FUERA=.TRUE.
```

```
        CALL FALLO1
        RETURN
40      IMIN=IX
        IMAX=IX
        RETURN
50      IMIN=IX
        IMAX=IX+1
        RETURN
        END
```

```
      SUBROUTINE RITE(A,B,C,T,N,P,TM)
      DIMENSION P(N)
      WRITE(4,100) A,B,C
100   FORMAT(1X,F10X,"R2/R1=",F10.3,"H2/H=",F10.3,"R/H=",F10.3)
      WRITE(4,200)N,T,TM
200   FORMAT(1X,10X,"N=",I3,"T1=",F10.3,"TMAX=",F10.3)
      WRITE(4,300)
300   FORMAT(1X,10X,"VALUES OF THE CURVE=")
      WRITE(4,400) P
400   FORMAT(1X,10X,7F)
      RETURN
```

```
      SUBROUTINE FALLO
C ***************************************************************
C ******          SUBROUTINE FALLO                          ********
C ****** THIS ROUTINE WORKS IN THE CASE OF A LIMITATION ********
C ****** OF THE CATALOG SIZE, GIVING A MESSAGE OF THE  *********
C ******* PARTICULAR LOCATION THAT IS NOT INCLUDED.     ********
C ***************************************************************
C
C
      COMMON/BLOCK3/ DR8H,DR2BR,DH2BH
      WRITE(4,100)
100   FORMAT( /,20X,"*** PARAMETERS IN THE RANGE",
     &  "BUT CURVE NOT IN CATALOGUE *****")
      WRITE(4,110) DR8H,DR2BR,DH28H
110   FORMAT( 20X,"R/H1=",F7.3,5X,"R2/R1=",F7.3,5X,"H2/H1=",
     &  F7.3)
      RETURN
      ENTRY FALL01
      WRITE(4,120)
120   FORMAT(1X,/,20X,"PARAMETERS OUT OF THE CATALOGUE RANGE",/)
      WRITE(4,130) DR8H,DR2BR,DH2BH
130   FORMAT( 20X,"R/H1=",F7.3,5X,"R2/R1=",F7.3,5X,"H2/H1=",
     &  F7.3)
      RETURN
      END
```

```
      SUBROUTINE PLOT1P(X,Y,N,CHAR,XN,XX,YN,YX,ICHEK)
C
C       ONE PAGE PLOT FOR TUBE OR PRINTER
C       MULTIPLE PLOTS AND/OR PRINTS
C
C       C H STOYER PROGRAMMER GOLDEN CO JUNE 1977
C
      IMPLICIT LOGICAL (L)
      LOGICAL ANSWER
      INTEGER LEND
      DIMENSION LEND(23), PLOT(69,23), SAME(1587), X(N), Y(N)
      EQUIVALENCE (PLOT(1),SAME(1))
      DATA BLANK/' '/,DBL/'D'/,LCLEAR/.TRUE./,LNEVER/.TRUE./
      LCLEAR = .TRUE.
      ITOSS = 0
      DO 1 I=1,1587
      SAME(I) = BLANK
      IF(I.LE.23)LEND(I)=0
    1 CONTINUE
      IF(XN.GE.XX) GOTO 2
      IF(YN.GE.YX) GOTO 2
      IF(ICHEK.EQ.0) GOTO 10
    2 XN=X(1)
      XX=X(1)
      YN=Y(1)
      YX=Y(1)
      DO 3 I=1,N
      IF(XN.GT.X(I)) XN=X(I)
      IF(XX.LT.X(I)) XX=X(I)
      IF(YN.GT.Y(I)) YN=Y(I)
      IF(YX.LT.Y(I)) YX=Y(I)
    3 CONTINUE
   10 SX = 68./(XX-XN)
      SY = 22./(YX-YN)
      ENTRY MORE1P(X,Y,N,CHAR)
      IF(CHAR.EQ.BLANK) RETURN
      DO 20 I=1,N
      IX = (X(I)-XN)*SX +1.5
      IY = 23.5 - (Y(I)-YN)*SY
      IF((IX.LE.0) .OR. (IX.GT.69)) GOTO 17
      IF((IY.LE.0) .OR. (IY.GT.23)) GOTO 17
      IF(IX.GT.LEND(IY)) LEND(IY)=IX
      LCLEAR = .FALSE.
      IF(PLOT(IX,IY).EQ.BLANK) GOTO 15
      PLOT(IX,IY) = DBL
      GOTO 20
   15 PLOT(IX,IY) = CHAR
      GOTO 20
   17 ITOSS = ITOSS+1
   20 CONTINUE
      RETURN
```

```
      ENTRY PLOTPR(IDS)
      IF(LCLEAR) GOTO 50
      LTUBE = IDS.EQ.4
      IF(LTUBE.AND.LNEVER)WRITE(4,4000)
      IF(LTUBE.AND.LNEVER) LSOFT=ANSWER(DUMMY)
      IF(LTUBE)LNEVER=FALSE
      IF(LSOFT.AND.LTUBE) WRITE(IDS,4007)
 1000 FORMAT(" ",7("**********"),"*")
 1001 FORMAT("1",7("**********"),"*")
 1002 FORMAT(" *",69A1,"*")
 1007 FORMAT(" ",7("**********"),"*",$)
 1008 FORMAT(" *",69A1)
 4000 FORMAT(" ARE YOU ON A SOFT COPY TERMINAL?",$)
 4007 FORMAT("0TYPE RETURN WHEN DONE VIEWING PLOT")
      IF(LTUBE)GOTO 30
      WRITE(IDS,1001)
      WRITE(IDS,1002)PLOT
      WRITE(IDS,1000)
      IF(LTUBE) GOTO 30
   25 WRITE(IDS,1003) XN,XX,YN,YX
 1003 FORMAT("0XMIN, XMAX, YMIN, YMAX = ",4(1PE11.2))
      AXN=10.**XN
      AXX=10.**XX
      AYN=10.**YN
      AYX=10.**YX
      WRITE(IDS,1003) AXN,AXX,AYN,AYX
      IF(ITOSS.GT.0) WRITE(IDS,1004) ITOSS
 1004 FORMAT("0",I5," POINTS OUT OF RANGE")
      RETURN
   30 CONTINUE
      WRITE(IDS,1000)
      J1=1
      DO35I=1,23
      IF(LEND(I).LE.0) GOTO 32
      J2=J1+LEND(I)-1
      WRITE(IDS,1008) (SAME(J),J=J1,J2)
      GOTO 35
   32 WRITE(IDS,1008)
   35 J1=J1+69
      IF(LSOFT.AND.LTUBE)WRITE(IDS,1007)
      IF(.NOT.(LTUBE.AND.LSOFT))WRITE(IDS,1000)
 1005 FORMAT(A5)
      IF(LSOFT)READ(4,1005) A
      GOTO 25
   50 WRITE(IDS,1006)
 1006 FORMAT("0BLANK PLOT!!")
      RETURN
      END
```

```
      FUNCTION ANSWER(DUMMY)
      LOGICAL ANSWER
      DATA AYES/'YES'/, ANO/'NO'/,STOP/'STOP'/
    1 READ(4,4000) ANS
      IF(ANS.EQ.STOP)GOTO 2
 4000 FORMAT(A5)
      ANSWER=.TRUE.
      IF(ANS.EQ.AYES) RETURN
      ANSWER=.FALSE.
      IF(ANS.EQ.ANO)RETURN
      WRITE(4,4001)
 4001 FORMAT(' WHAT?',$)
      GOTO 1
    2 CONTINUE
      STOP
      END
```

```
C
C
      SUBROUTINE TRPA(A,NCA,MNA,ATRP)
C ***************************************************************
C *****                  SUBROUTINE TRPA                    ********
C ***** THIS SUBROUTINE CALCULATES THE TRASPOSE OF A TWO********
C ***** DIMENSIONAL MATRIX,WHEN IT IS STORED IN A VECTOR********
C ***** FORMED BY SEQUENCES OF ROWS.                       ********
C ***** INPUT MATRIX:
C *****   A(NRA,NCA).....MATRIX WITH NRA ROWS AND ********
C *****                  NCA COLUMNS,STORED IN ONE VECTOR ********
C *****                  OF DIMENSION :MNA=NRA*NCA        ********
C *****          OUTPUT MATRIX:                           ********
C ******   ATRP(NCA,NRA)..... (STORED IN THE SAME MODE)*******
C ***************************************************************
C ******         JUAN CARLOS RODRIGUEZ ,FEB,1978          ********
C ***************************************************************
C
C THE PROPERTY FOR THE TRASPOSE IS A(I,J)= B(J,I)
C AND THE RELATION OF INDEX FROM TWO DIM. STORAGE INTO ONE DIM:
C       IJ = (I-1) * #OF COLUMNS + J
C
C
      DIMENSION A(MNA),ATRP(MNA)
      NFA=MNA/NCA
      DO 20 I=1,NFA
            DO 20 J=1,NCA
C USING THE INDEX RELATION
                  IJ=(I-1)*NCA+J
                  KK=(J-1)*NFA+I
C AND THE CONDITION FOR A TRANSPOSE
                  ATRP(KK)=A(IJ)
20    CONTINUE
      RETURN
      END
```

```
C
C
      SUBROUTINE PMTRX(A,NMA,B,NMB,C,NMC,NCA)
C
C
C ***********************************************************************
C ***** THIS SUBROUTINE CALCULATES THE PRODUCT OF TWO DIM.*******
C ***** MATRICES,WHEN THEY ARE STORED IN VECTORS,EACH    ********
C ***** VECTOR CONTAINING THE ROWS OF EACH MATRIX ONE    ********
C ***** AFTER THE OTHER.                                 ********
C ***** INPUT: ....... A(NMA),B(NMB)                     ********
C *****          WHERE:  NMA=#ROWS A (NRA)*#COL.A(NCA)   ********
C *****          AND     NMB=NRB*NCB                     ********
C ***** OUTPUT ....... C(NMC),WHERE( NMC=NRA*NCB)        ********
C *****                                                 ********
C ***********************************************************************
      DIMENSION A(NMA),B(NMB),C(NMC)
      NFA=NMA/NCA
      NCB=NMB/NCA
C
C
      DO 20 I=1,NFA
            DO 20 J=1,NCB
               X=0.0
C IN THIS DO LOOP THE GENERAL TERM OF THE PRODUCT C(IJ) IS COMP.
                  IJ=(I-1)*NCB+J
                  DO 10 K=1,NCA
C THE RELATION OF THE TWO INDEX (I,J)==> (IJ)
                   IK=(I-1)*NCA+K
                   KJ=(K-1)*NCB+J
                   X=A(IK)*B(KJ)+X
10                 CONTINUE
            C(IJ)=X
20    CONTINUE
      RETURN
      END
```

```
C ***************************************************************
C *****    SUBROUTINE ERROR                                      ********
C ***** THIS SUBR. CALCULATES THE RMS ERROR BETWEEN TWO ********
C ***** SET OF DATA STORED IN TWOVECTORS(SAME DIM.)     ********
C ***** WE ARE INTERESTED IN THE RANGE (I1,I2)ONLY.     ********
C ***** INPUT:     A(N),B(N).....TWO SET OF POINTS      ********
C *****           (I1,I2) ......RANGE OF INTEREST      ********
C ***** OUTPUT:     ERR .........RMS ERROR              ********
C ***************************************************************
C
      SUBROUTINE ERROR(A,B,N,I1,I2,ERR)
      DIMENSION A(N),B(N)
      ERR1=0.0

      DO 30 I=I1,I2

            X=A(I)-B(I)
            Y=X**2
            ERR1=Y+ERR1
30    CONTINUE

      XN=I2-I1+1
      ERR = SQRT(ERR1/XN)
      RETURN
      END
```

```
      SUBROUTINE DESCAL(A,B,N,NSQ)
C ***************************************************************
C ******          SUBROUTINE DESCAL                          ********
C ***************************************************************
C ****** THIS SUBROUTINE ,INVERTS THE UPPER  LEFT-HAND  ********
C ****** CORNER OF SYMMETRIC REAL*8 MATRIX BY ESCALATOR ********
C ****** METHOD . B CAN BE A . LINEAR INDEXEDD          ********
C ******          THE VARIABLES:                       ********
C ******          A..... MATRIX TO BE INVERTED         ********
C ******          B..... INVERSE OF  A                 ********
C ******          N..... DIMENSION OF A AND B (IN THIS SUB)*******
C ******          NSQ.... DIMENSION OF A AND B IN CALLING ********
C ******                  ROUTINE.                     ********
C ******                  I.E. A(NDIM,NDIM), MORE IMPORTANT*******
C ******                  THE FIRST DIMENSION.         ********
C ******                                               ********
C ***************************************************************
C ****** ORIGINAL PROGRAMER BILL RODI                   ********
C ****** MODIFIED BY CHARLES STOYER                     ********
C ***************************************************************
C
C
C
C     IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL LPRINT,ANSWER
      DIMENSION A(NSQ),B(NSQ)
      COMMON /KVAL/ KKK
      NDIM = N
C     WRITE(4,1001) N,KKK
C1001 FORMAT(' DESCAL: ',2I5,' DEBUG?')
      LPRINT=.FALSE.
      IF(A(1).EQ.0.0) GO TO 12
      B(1)=1.D0/A(1)
      IF(N-1) 11,11,2
2     NM1=N-1
      ISTART=0
      DO 9 K=1,NM1
      ISTART=ISTART+NDIM
      KP1=K+1
      INDEX=0
      ICOUNT=KP1
      QF=0.D0
      DO 4 I=1,K
      B(ICOUNT)=0.D0
      DO 3 J=1,K
3     B(ICOUNT)=B(ICOUNT)+B(INDEX+J)*A(ISTART+J)
      QF=QF+B(ICOUNT)*A(ISTART+I)
      INDEX=INDEX+NDIM
4     ICOUNT=ICOUNT+NDIM
      BETA = A(ICOUNT) - QF
      IF(BETA.EQ.0.0) GO TO 13
```

```
      BETA=1.D0/(BETA)
      INDEX=0
      ICOUNT=KP1
      DO 6 I=1,K
      TERM=BETA*B(ICOUNT)
      JCOUNT=KP1
      DO 5 J=1,K
      B(INDEX+J)=B(INDEX+J)+TERM*B(JCOUNT)
5     JCOUNT=JCOUNT+NDIM
      B(ISTART+I)=-TERM
      INDEX=INDEX+NDIM
6     ICOUNT=ICOUNT+NDIM
      ICOUNT=KP1
      DO 7 I=1,K
      B(ICOUNT)=B(ISTART+I)
7     ICOUNT=ICOUNT+NDIM
      B(ICOUNT)=BETA
9     CONTINUE
   11 IF(LPRINT) WRITE(4,1002) B
 1002 FORMAT(1H ,/,(1H ,7(1PE11.2)))
      RETURN
   12 ICOUNT = 1
   13 WRITE (4,1000) ICOUNT
 1000 FORMAT('0**** STOP   ATTEMPT TO DIVDE BY ZERO IN DESCAL; ICOUNT= '
     1I10)
      STOP
      END
```

```
      SUBROUTINE DERIV(FUERA)
C ***************************************************************
C ******          SUBROUTINE DERIV                          ********
C ****** THIS SUBROUTINE CALCULATES THE MATRIX OF DERI-     ********
C ****** VATIES "A" ;                                      ********
C ******                                                   ********
C ******          - USES FINITE DIFFERENCES                ********
C ******          - WORKS IN LOG DOMAIN (PARAM.&DATA)      ********
C ******          - THE DIFFERENT RANGE OF TIME IS SOLVED  ********
C ******            BY A CUTTING EFFECT FROM IMIN TO IMAX  ********
C ******          - IF THE POINT OF DERIV IS CLOSE TO THE  ********
C ******            CORNER OF THE CATALOG THE INCREM FOR   ********
C ******            THE DERIV WILL BE TAKEN IN DIRECTION   ********
C ******            OF THE CATALOG.                        ********
C ******                                                   ********
C ***************************************************************
C ******           PROGRAMER :JUAN CARLOS RODRIGUEZ.(1978)********
C ***************************************************************
C
C
      PARAMETER MNA=200,NPT=40,NPAR=5
      LOGICAL FUERA,RTPLT
      COMMON / BLOCK1 / OBS(NPT) ,T(NPT) ,P(NPAR) ,DCALC(NPT),
     &  A(MNA) ,IMAX ,IMIN,MNA1,NPT1,NPAR1,RTPLT
      DIMENSION B(MNA),CINC(NPT)

      EPS=0.1

      DO 25 I=1,NPAR1
            PK=P(I)
            ID=0
            SIGN=1.
15          CONTINUE
            IF(ID.EQ.1) WRITE(4,124)
            IF(ID.GT.1) WRITE(4,125)
            IF(ID.GT.1) RETURN
            ID=ID+1
            PINC=P(I)*EPS
            PINC=PINC*SIGN
            P(I)=P(I)+PINC
            FUERA=.FALSE.
            CALL CALC(CINC,T,NPT1,I1,I2,P,NPAR1,FUERA)
            IF(FUERA) SIGN=-2.
            IF(FUERA) GO TO 15
C FOR CHECKING PURPOSES
C     WRITE(15,100) I1,I2,P
100   FORMAT(1X,20X,'MODEL OF CURVE',/10X,2(I2,5X),/,20X,5(F,10X))
            IF(IMAX.GT.I2) IMAX=I2
            IF(IMIN.LT.I1) IMIN=I1
            DO 20 J=1,NPT1
                  IJ=(J-1)*NPAR1+I
```

```
                        F1=DCALC(J)
                        F2=CINC(J)
                        D=F2-F1
                        D=D*SIGN
C                       B(IJ)=(F2-F1)/PINC
                        B(IJ)=D*24.15885793
C                       WRITE(15,105)F1,F2,D,B(IJ)
105                     FORMAT(1X,10X,4F)

20              CONTINUE
                P(I)=PK
25      CONTINUE
124     FORMAT(1X,/,1X,'****PROBLEMS DUE TO CATALOG LIMITATIONS',
     &  'WHEN CALCULATING DERIVATIVE MATRIX****',/,1X,'** THE PROGRAM',
     &  ' WILL TRY WITH OPPOSITE INCREMENTATION**')
125     FORMAT(1X,'***THE VALUE IS STILL OUT OF THE CATALOG***',
     &  / ,1X,'***THE PROGRAM WILL STOP THE ITERATIONS ***')
C NOW THERE WILL BE A READJUST OF THE "A" MATRIX ,CUTTING IT FROM
C THE ROW 0 IMIN-1 AND FROM IMAX TO THE NPT1
C THE REMAINING  # OF ROWS WILL BE:
C       IMAX-IMIN +1 =IRE
C AND THE # OF ELEMENTS:
C       # ELEM.= IRE*NPAR1
C
C
C FOR TESTING  THE "CUTTING EFFECT"
        KMAX=(IMAX-IMIN+1)*NPAR1
C THE NUMBER OF ELEMENTS THAT WE DISTROY AT THE FRONT PART
        IDIS=(IMIN-1)*NPAR1
        DO 40 I=1,KMAX
                K=I+IDIS
                A(I)=B(K)
40      CONTINUE
C
        RETURN
        END
```

```
C ***************************************************************
C ******          SUBROUTINE TRACE                          ********
C ******                                                    ********
C ****** THIS SUBROUTINE CALCULATES THE TRACE (SUM OF      ********
C ****** THE DIAGONAL ELEMENTS) , OF A SQUARE MATRIX       ********
C ****** "A(N)", .                                         ********
C ***************************************************************
      SUBROUTINE TRACE(A,N,TRCE)
      DIMENSION A(N)
      X=N
      TRCE=0.0
      M1=SQRT(X)
      DO 20 I=1,M1
            INDEX=(I-1)*M1+I
            TRCE=TRCE+A(INDEX)
20    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MAXMIN(A,N,AMIN,AMAX)
C ***************************************************************
C *****          SUBROUTINE MAMIN                          ********
C ***** THIS SUBROUTINE CALCULATES THE MAXIMUN AND        ********
C ***** MINIMUN VALUES OF A  CERTAIN VECTOR.              ********
C ***** THE VALUE OF CERO WILL NOT BE STUDIED             ********
C ***** (THIS DETAIL CAN BE MODIFIED)                    ********
C *****          JUAN CARLOS MARCH 1978                  ********
C ***************************************************************
C
C
      DIMENSION A(N)

      AMIN = A(1)
      AMAX = A(1)

      DO 20 I = 1 , N
            IF ( A(I) .EQ. 0.0 ) GO TO 20
            IF ( A(I) .GT. AMAX ) AMAX = A(I)
            IF ( A(I) .LT. AMIN ) AMIN = A(I)

20    CONTINUE
      RETURN
      END
```

```
C ***************************************************************
C *****          PROGRAM GFACT.F4                              ********
C ***** THIS PROGRAM READS IN THE DATA OF THE TDEM             ********
C ***** RESPONSE OF THE EARTH FOR THE LOOP-WIRE CONFIG.        ********
C ******AND COMPUTES THE APPARENT RESISTIVITY CURVES           ********
C ***** WITH THE NEAR-ZONE FORMULATION.                       ********
C *****          INPUT DATA:                                   ********
C *****  NPT ......... NUMBER OF DIGIT.POINTS                 ********
C *****  DELT ........ SAMPLE INTERVAL (SEC.)                 ********
C *****  TCERO ....... FIRST SAMPLE LOC. (SEC.)               ********
C *****  SENS ........ SENSITYVITY OF THE RECORD.EQUIP.       ********
C *****  SCALV ....... SCALE FACTOR (REC.EQUI.& DIGT.)        ********
C *****                                                       ********
C *****  R ........... OFFSET DIST. (TRAS.-REC.) (M.)         ********
C *****  S *.......... AREA OF LOOP (SQR. KMTS)               ********
C *****  AB .......... LENGTH OF TRASM. (M.)                  ********
C *****  XI .......... INPUT CURRENT (AMP.)                   ********
C *****  V(I) ........ RECORDED VOLTAJES (MVOL.)              ********
C *****          OUTPUT FILES:
C *****  FOR15.DAT ...... THIS WILL BE THE INPUT OF THE       ********
C *****              INPUT OF THE INVERSION PROGRAM,          ********
C *****              IT WILL CONTAIN NUMBER OF SAMPLES,       ********
C *****              TIME VECTOR, AND APPARENT RES VECTOR     ********
C *****              (IN LOG DOMAIN)                          ********
C *****  -----.LPT........ THIS FILE WILL CONTAIN THE         ********
C *****              PLOTS OF INPUT DATA, THE PLOT OF APP.********
C *****              RES. DATA, AND COORESPNDENT VALUES       ********
C *****              ALSO WILL GIVE THE ESTIATION OF          ********
C *****              S (LONG.COND.) AND  H (H1+H2)            ********
C *****                                                       ********
C ***************************************************************
C *****          JUAN CARLOS  MARCH 1978                       ********
C ***************************************************************
      REAL LT,LRHO
      DIMENSION T(50),RHO(50),V(50),LRHO(50),LT(50)
      DOUBLE PRECISION FILIN, X


      WRITE(4,100)
100   FORMAT(1X,'ENTER THE INPUT FILENAME   ', $ )
      ACCEPT 110,FILIN
110   FORMAT(  A10)


      OPEN (UNIT=12,FILE='FILIN',ACCESS='SEQIN',MODE='ASCII')
      OPEN(UNIT=15,FILE='FOR15.DAT',ACCESS='SEQOUT',MODE='ASCII')
C     OPEN(UNIT=06,FILE='FOR06.DAT',ACCESS='SEQOUT',MODE='ASCII')
```

```
C ***************************************************************
C ******        READ THE INPUT DATA                        ********
C ***************************************************************

      READ(12,201) NPT,DELT,TCERO,SENS,SCALV
      WRITE(4,201) NPT,DELT,TCERO,SENS,SCALV
      READ(12,203) R,S,AB,XI
      WRITE(4,203) R,S,AB,XI
      READ(12,202)(V(I),I=1,NPT)
      WRITE(4,202)(V(I),I=1,NPT)

201   FORMAT( I,4F)
202   FORMAT( 1X,11F7.0)
203   FORMAT(  5F)

C ***************************************************************
C *******       PREPARE THE OUTPUT FILE                  ********
C ***************************************************************

      WRITE(6,301)
      WRITE(6,302)R,S,AB,XI

301   FORMAT(1X,//,20X,'OUTPUT FILE OF GFACT.F4',//)
302   FORMAT(1X,10X,'OFFSET DISTANCE ',F7.1,9X,'LOOP AREA ',F7.3,/,
     &  14X,'AB DISTANCE ',F7.1,9X,'CURRENT INTENSITY ',F7.3)
303   FORMAT(1X,/,20X,'INPUT DATA (TRANSIENT) ',/)
304   FORMAT(1X,//,20X,'TIME AND APPARENT RESISTIVITY VALUES')
305   FORMAT(1X,//,20X,'APPARENT RESISTIVITY CURVE (LOG-LOG)')
306   FORMAT(1X,/,25X,'INPUT DATA (VALUES) ',/)
310   FORMAT(25X,2(F10.4,5X))

C CREATE THE TIME VECTOR FOR PLOTTING PURPOSES.

      DO 20 I=1,NPT
            XX=I-1
            T(I)=TCERO+XX*DELT
            V(I) = V(I) * SENS / SCALV
            V(I) = ABS ( V(I))
20    CONTINUE

      WRITE(6,306)

      DO 25 I = 1 , NPT
25          WRITE(6,310) T(I),V(I)

      ICHAR=1H*

C CALL THE PLOT TO PRINT THE DATA
```

```
      CALL PLOT1P(T,V,NPT,ICHAR,0,0,0,0,1)

C PRINT THE PLOT IN THE OUTPUT FILE FOR6.DAT
      WRITE(6,303)
      CALL PLOTPR(6)


C ***************************************************************
C ******     CALCULATE THE GEOMETRIC FACTOR(NEAR ZONE)  ********
C ***************************************************************

      WRITE(6,304)

      EX=0.666666667
      P = 1.666666667
      X =A8 * R * XI * S
      X = X ** EX
      X = X * 0.39925378 * 0.00001

      K=0
      IF ( T(1) .LT. 0.05 ) K = 1
      IF ( V(1) .LT. 1. ) K = 1
      NN = NPT - K
      WRITE(15,201) NN
      DO 30 I = 1 , NN
            XT    = I + K - 1
            T(I) = DELT * XT + TCERO
            A     = V(I + K) ** EX
            B     = T(I) ** P
            RHO(I)  = X / B / A
            LRHO(I) = ALOG10( RHO(I) )
            LT(I)   = ALOG10( T(I) )
            WRITE (15,310) T(I),LRHO(I)
            WRITE (6,310) T(I),RHO(I)
30    CONTINUE


C PLOT THE CURVE OF APP.RES IN BILOGARITMIC PAPER

      CALL PLOT1P (LT,LRHO,NN,ICHAR,0,0,0,0,1)
      WRITE(6,305)
      CALL PLOTPR(6)

C ***************************************************************
C ****** DETERMINE THE VALUES OF S(LONG.COND.) AND      ********
C ****** H (H1+H2) BASED IN THE LOCATION OF THE MINIMUN ********
C ****** OF THE CURVE. (THIS FORMULA IS VALID ONLY FOR  ********
C ****** THE "H" TYPE OF GEOELECTRICAL SECTIONS)        ********
C ******  MORE INFORM. IN TJEM NEAR-ZONE BY A.A.KAUFMAN ********
C ****** (1977)                                        ********
C ***************************************************************
```

```
C NEXT STEPS TO FIND THE MINIMUN OF THE APPARENT RESISTIVITY
C CURVE

      CALL MINIM(RHO,NN,RMIN,NMIN)

      TMIN = T(NMIN)
C THE TIME CORRESPONDENT TO THE MINIMUN VALUE
      YN = 0.0
      YN = 6.2831853 * TMIN
      XN = YN / RMIN
      XN = SQRT ( XN )
      S  = 497.2 * XN
      A  = RMIN ** 2
      A  = A / YN
      H  = A * S ** 3 * 4.9 / 1000000.


C ***************************************************************
C ******* WRITE THE VALUES OF"S" AND "H" FOR A POSSIBLE ********
C ******* FIRST GUESS ESTIMATION.                       ********
C ***************************************************************


C IN THE FUTURE IT WILL BE POSSIBLE TO GIVE THE FIRST GUESS

      WRITE(6,400)
      WRITE(6,401) S, H

400   FORMAT( ///,20X,"APPROXIMATION VALUES OF BY MINIMUN LOCATION")
401   FORMAT(  / 25X, "LONGITUDINAL CONDUCTANCE  ",6X,F7.2,
     &  /, 20X,"THICKNESS OF FIRST AND SECOND LAYER " F7.2)

      STOP
      END
```

```
      SUBROUTINE MINIM ( A,N,AMIN,NMIN )
C ***************************************************************
C ****** THIS SUBROUTINE CALCULATES THE MINIMUN OF       ********
C ****** A SET OF DATA STORED IN AN ARRAY A(N).          ********
C ****** AMIN MINIMUN OF A(N)AND NMIN ITS LOCATION      ********
C ***************************************************************


      DIMENSION A(N)
      AMIN = A(1)
      NMIN = 1

      DO 30 I= 1 , N

            IF ( AMIN .GT. A(I) ) AMIN = A(I)
            IF ( AMIN .GT. A(I) ) NMIN = I

30    CONTINUE

      RETURN
      END
```

REFERENCES

Carnahan, B, Luther, H. A. and Wilkes, J. O., 1969, Applied numerical methods: John Willey and Sons Inc., N.Y., 604 pp.

Chao, C. Ku, 1976, Numerical inverse magneto telluric problems: Geophys., 41, 276-286.

Corbato, C. E., 1965, A least square procedure for gravity interpretation: Geophys., 33, 596-601.

Crewdson, J. H., 1976, Geophysical studies in the Black Rock Desert geothermal prospect,Nevada: Ph.D. Thesis T-1866, Colorado School of Mines, Goldel, Co., 180 pp.

Daniels, J. J., 1974, Interpretation of electromagnetic soundings using a layered earth model: Ph.D. Thesis T-1627, Colorado School of Mines, Goldel, Co., 86 pp.

Davis, J. C., 1973, Statistics and data analysis in geology: John Willey Sons Inc., N.Y., 550 pp.

Foster, M., 1961, An application off the Weiner-Kolmogrov smooothing theory to matrix inversion: J. Soc. Indust. Appl. Math.

Glenn, Ryu, Ward, Peeples, and Phillips, 1973, The inversion of vertical magnetic dipole sounding data: Geophys., 38, 1109-1129.

Inman, Ryu and Ward, 1973, Resistivitty inversion: Geophys, 38, 1088-1108.

Inman, J. R., Resistivity inversion with ridge regression: Geophys., 40, 789-817.

Johansen, H. K., 1977: A man/computer interpretation system for resistivity soundings over an horizontally stratified earth: Geoph. Prosp., 25, 667-691.

Jump, D. L. B. and Vozoff, K., 1975, Stable methods for the inversion of Geophysical data, Geophys., J. R. Astr. Soc., 42, 957-976.

Kaufman, A. A., 1977, Theoretical basis of the transient EM soundings in the Near-Zone,

----------, 1970, Catalog of transient EM curves in the Near-Zone, (vol.I, II, and III), (in russian)

Keller, G. V., 1968, Electrical prospecting for oil: Quarterly Colorado School of Mines, vol. 63, no. 2, 268 pp.

----------, 1970, Induction methods in prospecting for hot water: Geothermics (Spetial Issue 2), vol.2, pt. 1, pp. 318-332.

Keller, G. V., and Frischknecht, F. c., 1966, Electrical Methods in Geophysical Prospecting: Oxford, Pergamon Press, 527 pp.

King, C. A., 1971, Time-domain electromagnetic coupling: M.Sc. Thesis T-1427, Colorado School of Mines, Goldel, Co.

Lanczos, C., 1961, Linear diferential operators: Van Nostrand, Princenton.

Merchant, M. J. and Sturgul, J. R., 1977, Fortran IV programing with standard fortran watfor, watfiv, and structured watfiv: Wadsworth publish. comp., Ca., 544 pp.

Ofrey, Ofiafate, 1975, Detection and interpretation of various magnetic field components in time-domain electromagnetic sounding: M.Sc. Thesis T-1103, Colorado School of Mines, Goldel, Co.

Parker, R. L., i977, Understanding inverse theory: Ann. Rev. Earth Planet. Sci., 5, 35-64.

Press, F., 1968, Earth models obtained with Monte Carlo Inversion: J. Geophys. Res., 73, 5223-5234.

Rodi, W. L., 1975, A technique for emproving the accuracy of finite element solutions of magneto telluric data: Geophys. J. R. Astr. Soc..

Silva, L. R., 1969, Two-layer master curves for electromagnetic sounding: M.Sc. Thesis T-1250, Colorado School of Mines, Goldel, Co.

Skokan, C. K., 1974, A time domain electromagnetic survey of the East Rift Zone, Kilauea Volcano, Hawaii: Ph.D. Thesis T-1700, Colorado School of Mines, Goldel, Co.

Stoyer, C. H., 1978, Inversion of geophysical data (oral presentation): EM workshop 1978 , Boulder

Twomey, S., 1977, Introduction to the mathematics of

inversion in remote sensing and indirect measurements: Elsevier, Amsterdan, 243 pp.

Vanyan, L. L., 1967, Electromagnetic deep soundings: New York, Consultants Bureau