

MATHEMATICAL MODELING IN DISSOLUTION KINETICS facilities and a tutorship during my studies in this department. The help .

AND APPLICATIONS OF A NEW NONLINEAR REGRESSION PROGRAM I received to migrate to Australia and at the beginning of my postgraduate

I am indebted to Mr. K. Murray and M.J. Taylor for their help in

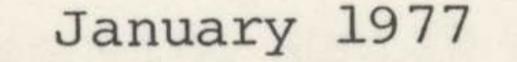
Peter Veng Pedersen

M.Sc., Dip. Ind. Pharm., Cand. Pharm. - as my supervisor during Dr. K.F. Brown's sabbatio/1 year is acknowledged.

construction of some items of equipment used for this work and to Mrs.

A thesis submitted to the University of Sydney as a requirement for the degree of Doctor of Philosophy in the Faculty of Science.

patience, love and understanding of my wife, Connie.



ACKNOWLEDGEMENT

I would like to express my deep gratitude to Dr. K.F. Brown for his encouragement and continued interest in my work. In particular I want to direct my thanks to him for his suggestions and comments in the development of the papers published based on this work.

I would also like to thank Professor T.R. Watson for providing facilities and a tutorship during my studies in this department. The help I received to migrate to Australia and at the beginning of my postgraduate studies is acknowledged with particular gratitude.

My thanks are due to the members of the academic staff, technical staff and fellow postgraduate students, for the pleasant time I have enjoyed in their company. The encouraging discussions with Dr. J.J. Ashley who acted as my supervisor during Dr. K.F. Brown's sabbatical year is acknowledged.

I am indebted to Mr. K. Murray and M.J. Taylor for their help in construction of some items of equipment used for this work and to Mrs. Anita Auzins, Mrs. Anna Marie Pokorny and Ann Broughton for their help in typing my work.

Finally, I would like to express a very deep gratitude for the patience, love and understanding of my wife, Connie.

[REDACTION] ПD

- 0 0 0 -

Poter Venij Pedersen, "Modelling in Pharmacokinetics and some Practical Experiences with NUNLIN and a New Program FUNFIT", J. Pharmadokin. Biopharm. (1976).

PUBLICATIONS

- Peter Veng Pedersen & K.F. Brown, "Dissolution in Relation to 1. Initial Particle Distribution", J. Pharm. Sci., 64, 1192 (1975).
- Peter Veng Pedersen & K.F. Brown, "Size Distribution Effects 2.
 - in Multiparticulate Dissolution", J. Pharm. Sci., 65, 1981 (1976).
- 3. Peter Veng Pedersen & K.F. Brown, "Theoretical Isotropic Dissolution of Nonspherical Particles", J. Pharm. Sci., 65, * (1976).
- Peter Veng Pedersen & K.F. Brown, "Experimental Evaluation of 4.
 - Three Single Particle Dissolution Models", J. Pharm. Sci., 65 * (1976).
- Peter Veng Pedersen, M.J. Crooks & K.F. Brown, "A Method of 5.

obtaining Drug-Macromolecule Binding Parameters directly from

Dynamic Dialysis Data", J. Pharm. Sci., 65, * (1976).

Peter Veng Pedersen & K.F. Brown, "A New Method of Characterizing 6. the Dissolution Properties of Drug Powders", J. Pharm. Sci., 65, *

(1976). ion Program 30 p. 10 copies, University of Sydney Library

Peter Veng Pedersen, "Methods of Enhancing Dissolution". 7.

Proceedings of a Workshop in Formulation and Biopharmaceutics,

University of Sydney, pp. 1-34 (1975).

Peter Veng Pedersen, "Dissolution of Powders as a Tool on Quality 8.

Control of Drugs". Proceedings of a Workshop on Automated Laboratory

Techniques, University of Sydney, pp. 1-16 (1976).

(* = papers in press).

Papers recently submitted for Publication

Peter Veng Pedersen, "Modelling in Pharmacokinetics and some 9.

Practical Experiences with NONLIN and a New Program FUNFIT",

J. Pharmacokin. Biopharm. (1976).

Peter Veng Pedersen, "General Treatment of Linear Pharmacokinetics", 10.

J. Pharm. Sci. (1976).

Peter Veng Pedersen & K.F. Brown, "A Universal Standardization of 11. Dissolution Rate Measurements", J. Pharm. Sci. (1976).

- 12. Peter Veng Pedersen & K.F. Brown, "A General Class of Multiparticulate Dissolution Models", J. Pharm. Sci. (1976).
- S. Mason, P. Veng Pedersen, M.J. Crooks & K.F. Brown, 13.

"The Use of a New Nonlinear Regression Computer Program in Describing

27

36 :

Multiple Equilibria", J. Pharm. Sci. (1976).

J. Whitlam, M.J. Crooks, K.F. Brown & P. Veng Pedersen, 14.

"Aspects of the Binding of Ibuprofen to Serum Albumin",

Biochem. Pharmacol. (1976).

Technical Reports

Peter Veng Pedersen, "Fitting Nonlinear Equations to Data", 15.

99 p. 2 copies, University of Sydney Library (1976).

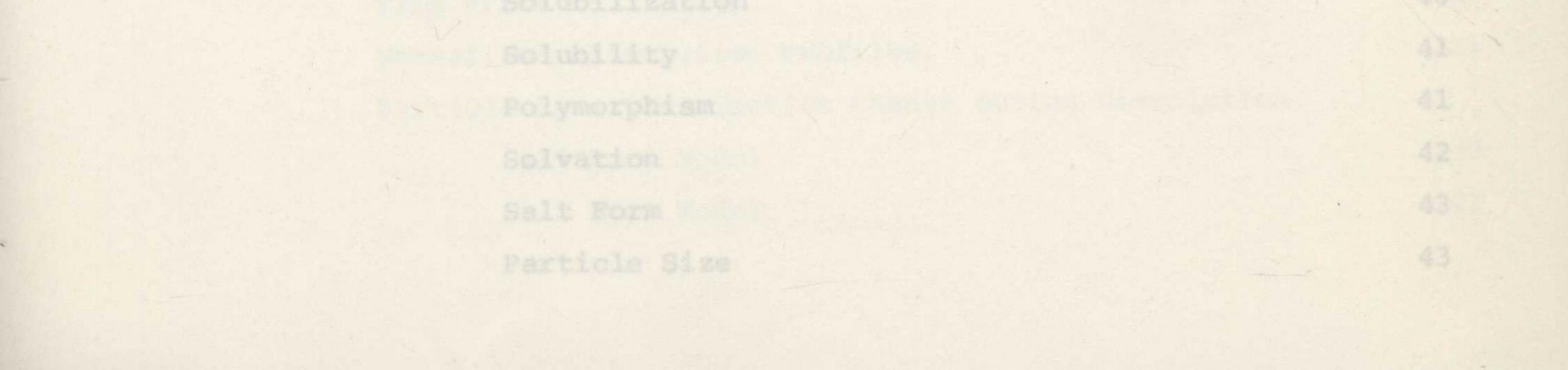
Peter Veng Pedersen, "Guidelines for the use of FUNFIT Nonlinear 16.

Regression Program", 30 p. 10 copies, University of Sydney Library

(1976).

Factors Affecting Drug Dissolution.

- 0 0 0 -



CONTENTS

		Page
Chapter 1	Introduction	1
	Methods for Studying Dissolution Behaviour.	4
	Classification.	5
	Non-sink Conditions.	7
Chapter 2	Experimental of damed on Diffusion Level Assumptions	13
	Dissolution Apparatus.	13
	The Dissolution Cell.	17
	The Heat Exchanger.	18
	Flow Cell.	24
	Analytical.	26
	Dynamic Dialysis	27
	Materials.	30
	Data Treatment.	30
	Particle Distribution Analysis.	31
	Pharmacokinetic Data.	31
Chapter 3	Biological Aspects and Methods of Enhancing Dissolution.	32
	Pathway to Systemic Circulation.	33
	Availability and Rate of Dissolution.	34
	Therapeutic Implications.	35
	Factors Affecting Drug Dissolution.	36
	Disintegration	37
	Agitation	37
	pH	38
	Viscosity	38
	Interfacial Tension	39
	Complexation	39
	Solubilization	40
	Solubility the Troffler.	41
	Port Polymorphism builds Change during Dissolution	41
	Solvation Solution	42
	Salt Form Model	43
	Particle Size	43

	Solid Dispersions.	47
	Particle Size Distribution.	48
Chapter 4	Single Particle Dissolution	49
<u>enapter</u>	Taperimental Evaluation of Three Dissolution Modals	49
	Nernst Film Theory.	50
	Noyes-Whitney's equation.	51
	Danckwerts Theory.	51
	Limited Solvation Rate Theories.	52
	Spherical Dissolution. Approaches not Based on Diffusion Layer Assumptions.	55
	the property of the state of th	56
	Hixson and Crowell's Equation.	56
	Sink-Conditions. Comparison of Models.	56
	Static Dissolution Models.	58
	A Dynamic Model for a Free Falling Spherical Particle.	61
	DIRET AVAILABLE PROTING	65
	Nonspherical Isotropic Dissolution.	68
	Classification	71
	Shape Ratio	83
	Equivalent Spherical Diameter Summary of nonspherical Isotropic Dissolution	90
	AIMIMIZACION,	
Chapter 5	Multiparticulate Dissolution	95
	Monodisperse Powder.	95
	Polydisperse Powder.	96
	General Derivation.	97
	Log-Normal Powders and the Cube Root Model.	101
	Comparison with Earlier Derivation.	107
	Effect of Dispersion Parameter, J.	111
	Effect of Truncation Parameters.	112
Chapter 6	Size Distribution Effects	117
	Theoretical Rules.	122
	Time Scaling.	124
	Time Fraction.	128
	Normalized Dissolution Profiles.	131
	Particle Size Distribution Change during Dissolution	
	1/3 Root Model	139
	1/2 Root Model	141

	2/3 Root Model	143
	The Dispersion Product.	144
Chapter 7	Experimental Evaluation of Three Dissolution Models	151
	Theoretical Considerations.	151
	Spherical Approximation.	153
	Time Scaling Approach.	154
	Particle Distribution Approximation.	159
	Comparison of Models.	167
	Assumptions made.	168
	Alternative Model.	169
Chapter 8	-FUNFIT- A Time Sharing Program for General Nonlinear	171
	Regression and Curvefitting.	
	Other Available Programs.	171
	Theory.	172
	Simultaneous Fitting of Several Systems.	173
	Weighting of Response Systems.	175
	Fitting Implicit Functions.	176
	Minimization.	176
	Lattice Search.	178
	Convergence Criterion.	179
	Statistical Estimation.	181
	Assumptions made.	183
	Confidence Regions.	184
	Use of the Variance-Covariance Matrix.	186
	Use of Transformation Technique.	188
Chapter 11	Examination of Residuals.	188
	Durbin-Watson Statistics.	189
	"Run" - statistics.	189
	"Number" - statistics.	190
	Outliers.	·190
	Comparison of Parameters and Models.	191
	Input to FUNFIT.	192
	Input Check.	199
	Input Error Messages.	201
	in the Presence of Membrane Rinding.	

Page

203

		Page
	Output from FUNFIT.	205
	Defining The Equations to be Fitted.	209
Chapter 9	Curve Fitting and Modelling in Pharmacokinetics	
	Comparison of FUNFIT and NONLIN Computer Programs	217
	Pharmacokinetic Applications of FUNFIT.	219
	Fitting of An Implicit Model.	219
	Discrimination Between Models.	221
	Analysis of Residuals.	223
	The Predictive Power of the Model.	225
	Comparison of FUNFIT and NONLIN.	226
	Bloodlevel Data Used in Comparison.	227
	Least Squares Estimation.	229
	Analysis of NONLIN's Failure.	230
	Effect of the Truncation Option in NONLIN.	233
Chapter 10	A New Method for Characterising the Dissolution Profile	
	of Drug Powders.	236
	Theory.	237
	Monodisperse Powders.	241
	The Specific Dissolution Rate Parameter.	242
	The Effective Particle Size Distribution.	246
	Regression Parameters.	247
	Curvefitting.	247
	Choice of the Mathematical Model.	248
	Characterisation and Quality Control.	265
	Significance of the Dispersion Parameter.	266
Chapter 11	A Method of Obtaining Drug-Macromolecule Binding	
	Parameters Directly from Dynamic Dialysis Data.	267
	Previous Approaches.	267
	Theory.	268
	Binding of Small Molecules by the Dialysis Membrane.	272
	Determination of K _e and Membrane Binding Parameters in	
	Absence of Macromolecule.	272
	Determination of Drug-Macromolecule Binding Parameters	
	in the Presence of Membrane Binding.	275

(iv)

D	3	α	0	
1	a	м	0	

	Data Treatment and Results.	276
	Comparison of Method with Previous Approaches.	281
	Use of Arbitrary Functions.	282
	An Analysis of the Bias in Previous Methods.	284
BIBLIOGRA		293
APPENDIX	Glossary of Symbols.	302
formulatio	Microfiché source listing of FUNFIT.	306

isportant in drug formulation as becomiques in pharmacology, medicine and analysis of drugs in the body improved. Emphasis has been directed toward studying the interactions between drug formulations and the organism and this resulted in the establichment of biopharmaceutics as a distinct field within the pharmaceutical sciences.

One of the sajor highermaceutical problems has been absorption in relation to drog formulation. Many very potent drugs have high lipid/ - 0 0 0 -

water partition coefficients which facilitates their penatration through biological penbranes. While this property lowers the biological berrier and favours absorption, it also paradoxically creates formulation problems bucause such drops are frequently only very slightly soluble in water.

It is generally accepted that limited absorption after oral intake is frequently the result of slow dissolution. This high of absorption problem ocquits so frequently that bioavailability touting has become an important part of modern drug and dosage form design and production control. Namy examples of differences in bioavailability resulting from differences in dissolution behaviour have been given in the literature (1, 2).

CHAPTER 1

- 1 -

INTRODUCTION

Historically pharmacy has been primarily concerned with the forms in which drugs are administered. Most effort has been directed to making formulations of drugs physically and chemically stable and acceptable to the patient with regard to appearance, colour, taste and smell and form of administration. It has become apparent, however, that other factors are important in drug formulation as techniques in pharmacology, medicine and analysis of drugs in the body improved. Emphasis has been directed toward studying the interactions between drug formulations and the organism and this resulted in the establishment of biopharmaceutics as a distinct field within the pharmaceutical sciences.

One of the major biopharmaceutical problems has been absorption in relation to drug formulation. Many very potent drugs have high lipid/ water partition coefficients which facilitates their penetration through biological membranes. While this property lowers the biological barrier and favours absorption, it also paradoxically creates formulation problems because such drugs are frequently only very slightly soluble in water.

It is generally accepted that limited absorption after oral intake is frequently the result of slow dissolution. This kind of absorption problem occurs so frequently that bioavailability testing has become an important part of modern drug and dosage form design and production control. Many examples of differences in bioavailability resulting from differences in dissolution behaviour have been given in the literature (1, 2). Research into dissolution kinetics of drugs is therefore of great importance. The numerous reviews published confirm the importance of the field (1-5).

The main purpose of most dissolution tests is to describe or quantitate the dissolution of a compound or dosage form. The dissolution process is most frequently described in the form of a dissolution profile. Although this is the most exact representation it is useful in practice to summarize and quantitate the dissolution kinetics in terms of one or more parameters e.g. the so-called dissolution rate constant. In order to do so, it is necessary to decide on some kind of mathematical description.

If the purpose is just to summarize dissolution data then a great number of arbitrary approximating functions can be used, e.g. polynomials and multiexponential expressions. Although the original data may be regenerated closely from parameters obtained in this way it is generally not reliable to make predictions or extrapolate on this basis. Furthermore, in the absence of mechanistic understanding and because of the substantial number of parameters involved (as in the case of polynomials), it is not possible to characterize the dissolution in a meaningful way. The use of arbitrary approximating functions is therefore of limited use. It is thus more meaningful to use a mathematical model which has some mechanistic significance at least conceptually and which adequately describes the dissolution behaviour. The extent to which the true mechanism of the system should be approximated by the mathematical model is a somewhat philosphic question, which depends on the purpose of the investigation. For practicability, the model chosen must involve a compromise between its correctness and its simplicity.

Dissolution is a heterogeneous process which is virtually impossible to describe rigorously because of the complexity of the mass transport phenomenon in an agitated heterogeneous system. It is apparent

- 2 -

from Chapter 4 that even when the dissolution mechanism of a free falling single spherical particle is analysed, using several simplifying assumptions, a "rigorous" mathematical model becomes too complex to be of *practical* significance. The description of the kinetics must therefore rely on a simpler model. As shown in Chapter 4 several such models can be derived on the basis of various assumptions about the mass-transport mechanism in the interfacial region.

The model for a *multiparticulate* system must be based on a model for the single particle dissolution behaviour together with a model for the particle size distribution.

Several multiparticulate dissolution models may be postulated. A decision about the model which best characterises the dissolution behaviour must be a compromise between the extent of experimental verification, its simplicity, and whether it is mechanistically meaningful. It may be possible to postulate very simple models which fit a particular set of dissolution data very well. However, oversimplification of the dissolution kinetics may lead to a lack of generality or flexibility. For example it may be possible to derive simple models for the dissolution of monodisperse systems. However, these will usually not fit dissolution data from polydisperse systems adequately; whereas more complex models for polydisperse systems may describe monodisperse systems as a special case.

The evaluation of how well a model fits dissolution data must rely on regression analysis. If the model is linear or can be transformed to linear form then the curve fitting procedure is generally simple. However, the multiparticulate dissolution models for polydisperse systems derived in this thesis are of a nonlinear form which can only be fitted and analysed

- 3 -

properly using a nonlinear regression computer program. As described in Chapter 9 several such programs are available. However, none of these programs are written for interactive time sharing and dedicated to mathematical modeling. The program FUNFIT described in Chapter 8 has been developed with these features. It is a *general* nonlinear regression program which can be used in any branch of science.

Methods for Studying Dissolution Behaviour in vitro.

The large number of dissolution rate measurement procedures and apparatus which have been described indicates that a universally acceptable method has not yet been developed. The many different techniques have been well reviewed by Hersey (6), Wagner (5) and Braun & Walker (7) and more recently by Hersey and Marty (8) and Groves (9).

The objectives of various tests are often very different. One worker may be interested in the fundamental dissolution behaviour, another in the effect of agitation and type of vehicle flow, or perhaps in the thermodynamics of the dissolution process, thus many kinds of dissolution apparatus are adapted to the study of fundamental principles of the process as for example the rotating-disc method, where the experimental conditions are better defined in terms of surface area of drug and agitation. Others are aimed at investigating dissolution behaviour of dosage forms *in vitro* to provide an estimate of their behaviour *in vivo*. The goal in these cases is to establish an *in vitro* - *in vivo* correlation such that a particular drug can be screened *in vitro* to provide an index of its expected behaviour *in vivo*. This requires that the composition of the drug formulation, the materials used and the production techniques are constant. Prediction of *in vivo* performance from *in vitro* data cannot be guaranteed but is only

- 4 -

intended to enable screening of dosage forms more readily and economically than *in vivo* procedures. Other reasons for performing dissolution tests include development purposes to guide the pharmaceutical formulator in the preparation of optimum dosage forms of drugs for clinical trial or for control purposes to ensure that a given pharmaceutical product is essentially uniform from lot to lot.

The various apparatus used for dissolution testing have been reviewed by Hersey (6), who classified them according to the type of agitation (free and forced convection) and the existence of sink or non-sink conditions.

In the past, the term sink conditions has most frequently referred to the situation where the concentration of dissolved drug is kept small, of the order of about 10% of the solubility, such that redeposition of dissolved material onto the dissolving solid is considered negligible. This is somewhat vague and imprecise. In Chapter 4 of this work the sink condition is defined in more precise mathematical terms.

The following section of this chapter is not intended to be a comprehensivereview of apparatus used in dissolution testing, this has already been done by several authors (5-9), but is an attempt to outline briefly the basic principles of their design and discuss their advantages and limitations.

Apparatus with non-sink conditions and constant vehicle volume.

Most apparatus described in the literature belongs to this group because of the simplicity of design and operation. They are based on a constant volume and differ only in the shape of the dissolution container, the way agitation is supplied or in the position of the

- 5 -

dissolving sample. One of the most widely used techniques is the "beaker method" often credited to Levy & Hayes (11) although it was used earlier by Parrott & coworkers (12). It has been used widely for fundamental dissolution rate studies (13, 14) and in modified form for dissolution studies by Shefter & Higuchi (15) and other workers.

The rotating disc apparatus consists of a disc holder to which a tablet or compressed disc of the pure drug is fixed and rotated at suitable constant speed in a round bottomed flask. The method, proposed by Levy & Sahli (19), has been used chiefly for studies of intrinsic dissolution rate and studies concerned with heterogeneous reaction kinetics, diffusion layer theory, effect of agitation on dissolution rate and other fundamental problems and has therefore found wide use (20-22). In interpreting the results of this apparatus it is important to consider that the enormous pressures (for example 50.000 psi on a 0.5 inch diameter disc) often used to make these discs from pure drug powder may introduce changes in crystal form and consequently alter the physical properties of the drug. Clevely & Williams (23) reported that grinding of crystalline barbituric acid derivatives may produce changes in polymorphic form.

The static disc technique employs forced convection in the dissolution liquid and thus has less defined agitation than the rotating disc. Other disc methods include the solvometer technique (24, 25) and the hanging pellet method (13).

There are several disadvantages of the methods mentioned above. They will be discussed under following headings: (1) non-sink conditions, (2) agitation, (3) sampling, (4) introduction of drug sample.

- 6 -

Non-sink conditions

If the object of the dissolution test is to establish an *in vitro* - *in vivo* correlation or to provide a rough guide to the drug release rate of a dosage form *in vivo* then the methods above are of limited value according to Gibaldi & Feldman (26). These authors stated that unless sink conditions are maintained, *in vitro* results will bear little relationship to *in vivo* observations, for drugs that show dissolution rate limited absorption.

- 7 -

Dissolution testing is obviously most relevant for those drugs which represent the greatest dissolution problems. In general such drugs are the least soluble, which means it will frequently be necessary to use exceedingly large volumes of solvent to follow dissolution behaviour for more than just a small fraction of the drug sample used. For example, to follow the dissolution of a tablet containing 5 mg glibenclamide to completion in 0.1 M HCl, it is theoretically necessary to use 500 ml solvent (solubility of glibenclamide is 0.5 mg/100 ml). Dissolution will be very slow, however, because the process slow continuously as saturation is approached. In practice it is usually not convenient to exceed about 20% of saturation. Thus in the example above 2.5 l of solvent would be necessary. This is a rather inconvenient and unwieldy volume for handling and maintaining proper agitation and temperature control.

Agitation of sending, which composide the struct due to the scholling

The large volume of solvent required for apparatus operating under non-sink conditions causes problems in the maintenance of suitable agitation. If the drug is in the form of a powder or disintegrating dosage form a high degree of agitation will be required to suspend the drug particles in the solvent so that they are all exposed to similar conditions of agitation. On the basis of *in vivo* data Hamlin *et.al*. (27) indicated that only low agitation rates could adequately differentiate rates of release from solid dosage forms. These authors studied the rates of dissolution of two polymorphic forms of methylprednisolone and found that sensitivity in distinguishing between the rates decreased at higher agitation intensities. This confirms the importance of using a dissolution apparatus where agitation is well defined and can be varied over a wide range.

Sampling. the vehicle. The powder may also agaregate and float on the

Several problems are encountered with respect to sampling in the types of apparatus mentioned above. It is impossible to sample without disturbing the dissolution process to some extent. Sampling affects solvent volume, which in turn may affect agitation conditions. Replacement of removed solvent is not an entirely satisfactory solution to the problem because it results in a discontinuity or small drop in the concentration of dissolved drug. The agitation conditions may change during additions of replacement liquid and affect the homogeneity of the system by creating "pockets" of fresh solvent which mix only slowly with the bulk. The latter is particularly important under conditions of low agitation. The most serious problem in sampling these systems seems to be the fact that the most precise and detailed characterisation of the dissolution process requires the greatest frequency of sampling, which compounds the errors due to the sampling procedure. The time taken for sampling is difficult to define precisely. In most cases it is considered to be instantaneous rather than a time interval. Such errors may be quite substantial at the initial stages of the experiment, where the concentration is changing very rapidly.

- 8 -

Introduction of drug sample

It has been suggested that only a small amount of solid sample should be used for a complete dissolution test if the solubility is low. Furthermore, slightly soluble drugs are often ground finely or micronised which increases their surface-free energy so that particles may adhere and be difficult to disperse evenly in the solvent. They frequently adhere to the side of the dissolution container and a significant fraction of the powder may be washed up the side of the container where it is no longer exposed to the vehicle. The powder may also aggregate and float on the surface.

- 9 -

It can thus be seen that the most widely used type of dissolution apparatus (non-sink, constant, vehicle volume) has several serious design disadvantages which limits its usefulness in fundamental dissolution kinetics studies.

Dissolution apparatus with non-sink conditions, constant vehicle volume and automatic recording

Schroeter & Wagner (28) appear to be the first to describe an automatic recording dissolution apparatus. They combined a beaker type apparatus with a peristaltic pump. Filtered solvent was pumped to an automatic recording spectrophotometer and returned to the beaker. This procedure has several advantages. It provides unlimited data points, there is no discontinuity in the system due to sampling and the number of sources of error are reduced. Furthermore, it is fast and convenient.

Among the disadvantages are the facts that particles may be trapped in the filter system, the concentration range is limited by the spectrophotometer; however variable path length absorption cells can extend the range. The system is only useful where there is little background absorption from solvent or excipients. The problems associated with the beaker method including introduction of sample, agitation, solvent volume, non-sink conditions are still present.

The technique described by Baum & Walker (7) represents a considerable improvement in the method used by Schroeter & Wagner. Agitation is performed by the solvent as it flows through a colum type dissolution chamber bounded at both ends by mesh screens. Solvent is recycled continuously through the column via a beaker or a flask. Sampling can be done from a collection reservoir or better by passing the filtered solvent through an automatic recording spectrophotometer. This system has several advantages compared to that of Schroeter & Wagner. Firstly, the solvent flow or agitation experienced by the particles is better defined and more easily adjustable. Entrapment of particles by the screen or filter should have little effect since they will still be exposed to almost the same solvent flow. Wetting and dispersion problems are reduced. In addition much larger solvent volumes can be handled readily so that sink conditions can be approached without increasing agitation rate. This system could be improved by altering to a non-recycled open system.

Dissolution apparatus with sink conditions: non-recycled open systems.

These types of apparatus include those most recently developed and represent the most suitable apparatus available for studies of dissolution kinetics. They consist essentially of a dissolution vessel with continuous input of fresh solvent and output of filtered solvent containing dissolved drug. The concentration of drug can be continuously monitored by some automated analytical procedure or solution sample may be

- 10 -

collected in fractions and assayed separately.

The various methods usually differ only in the design of the dissolution vessel and the mode of agitation, which may be produced either by the solvent flow or other means. The most common flow-through dissolution cells are cylindrical, with filters at both ends to enclose the sample and use the solvent flow as the only source of agitation (29-32).

Lapidus & Lordi (33) described a flow-cell which included a holder for compressed drug discs or tablets, enabling drug release measurements under sink conditions from a constant surface area. Their method represents a valuable alternative to the rotating disc technique for studying dissolution kinetics of pure drugs.

There are many advantages to using the continuous-flow column type apparatus. It is a flexible system that permits changes to be made readily, even within a test run, of important factors such as temperature, flow rate and vehicle composition including pH, viscosity, drug concentration and surfactant concentration.

Data obtained using a continuous recording technique with this system are in differential form. Thus a direct recording is made of the variation of release rate with time. This is superior to integral data (cumulative amounts dissolved with time) which tends to obscure small changes in dissolution rate particularly if only relatively few fractions are taken. A continuous recording yields unlimited data points which enable very precise characterisation of the dissolution behaviour. For this reason, it is also suitable for automatic data processing (34). Unlike most other systems, in particular the beaker method, sampling does not influence the dissolution process.

Normally the problem of wetting is minimal. However, when necessary

wetting can be accomplished by introduction of a surfactant solution for a short period initially and allowing it to be washed away immediately by fresh solvent (30).

The dissolution process can be followed to completion provided sufficient solvent is available and that the analytical method is sufficiently sensitive. All particles of a powder experience essentially the same intensity of solvent flow, including those particles which collect on the filters or screen at the ends of the column, thus agitation conditions are related to the solvent flow rate in a meaningful way and the latter is easy to define and control. Complete sink condition can be approximated very well. Thus results are more reproducible and instantaneous initial drug release measurements can be made, in contrast to most other methods. Furthermore, the continuous flow method is fast, convenient and very suitable for routine dissolution tests.

It is apparent that the continuous flow dissolution cell apparatus is a powerful and versatile tool in studies of dissolution kinetics.

The high precission dissolution apparatus described in Chapter 2 belongs to this category.

exceptions of that used in the pump, which was a stlicen tubing (silestic new Corning, 1.d., 0.335 cm and o.d., 0.465 cm). I is an open tube type autor that sonitory the pressure governing the flow rate of liquid.

Figure 2.3 shows a detailed diagram of the dissolution cell constructed for this work. Fowder to be investigated is spread in a this evenly distributed layer, c. in a sandwich-like arrangement between two

CHAPTER 2

EXPERIMENTAL

Dissolution Apparatus

A diagram of the continuous-flow recording dissolution apparatus is shown in Fig. 2.1. R1 and R2 are 20 litre reservoirs containing the dissolution media. P is a peristaltic pump (MHRE Watson-Marlow Ltd., England) transporting the liquid from R₁ or R₂ through a heat exchanger E, which adjusts the liquid to the required temperature before it reaches the dissolution cell D. This is immersed in a water filled, jacketed beaker, B, maintained at the same temperature as the dissolution liquid leaving E. Liquid from the dissolution cell, D, passes through a flow-cell, F, in the spectrophotometer, S, (Perkin-Elmer 124) which was fitted with chartrecorder, R, and finally accumulates in the container, C. V_1 and V_2 are two-way valves which enable by-pass of the dissolution cell, D, for zeroline adjustment of the spectrophotometer with blank liquid from the reservoir. V_3 is a similar valve by means of which liquid can be drawn from either reservoir R1 or R2. Flexible polyethylene tubing (i.d., 0.35 cm) was used throughout with the exceptions of that used in the pump, which was a silicon tubing (silastic, Dow Corning, i.d., 0.335 cm and o.d., 0.465 cm). L is an open tube type meter that monitors the pressure governing the flow rate of liquid.

The Dissolution Cell

Figure 2.2 shows a detailed diagram of the dissolution cell constructed for this work. Powder to be investigated is spread in a thin evenly distributed layer, c, in a sandwich-like arrangement between two Figure 2.1

Diagram of continuous-flow dissolution apparatus: R_1 and R_2 are reservoirs containing the dissolution liquid; P, peristallic pump; E, head exchanger (Fig. 2.3); D dissolution cell (Fig. 2.2); B, water filled jacketed beaker; F, spectophotometer flow-cell (Fig. 2.5); S, spectophotometer with reorder R. V_1 , V_2 and V_3 are two way valves; C, collection vessel; T, thermostat; L, pressure meter.

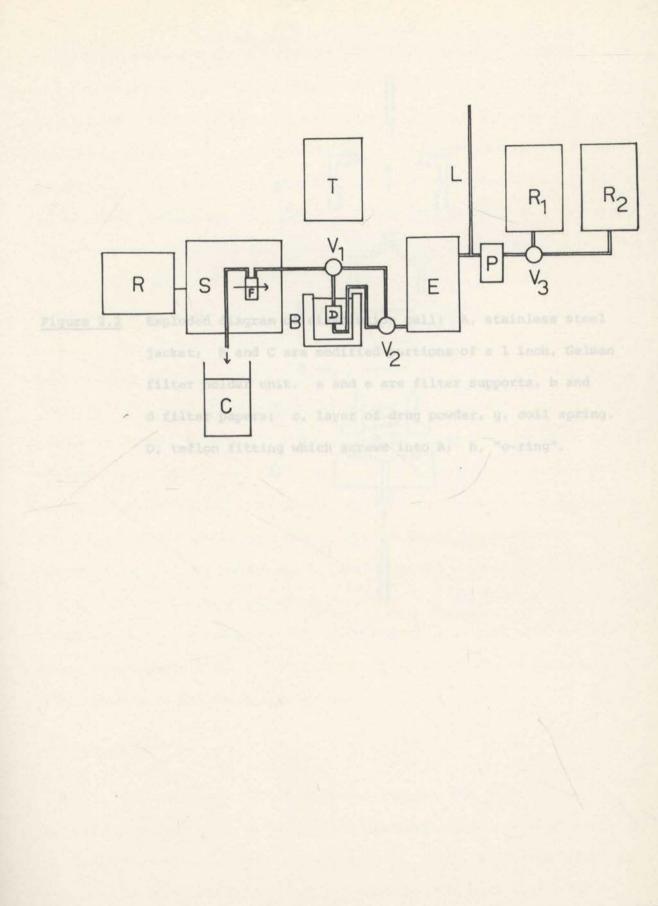
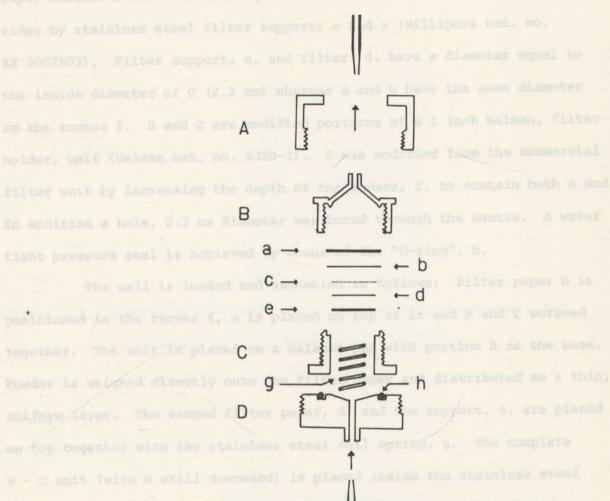


Figure 2.2 Exploded diagram of dissolution cell: A, stainless steel jacket; B and C are modified portions of a l inch, Gelman filter holder unit. a and e are filter supports, b and d filter papers; c, layer of drug powder, g, coil spring. D, teflon fitting which screws into A; h, "o-ring".

- 16 -



jacket, A. The Teflow component, D. is there firmly acrowed to A to form a tight soal with C by means of the "O-ris". The assembled with is placed upright in a tripod stand with the homes connected at both ends and the whole is immersed in the jacketed boaker (5 on Fig. 2.1) for thermal

Reat Exchanges

The equipment for temperature requision was also designed apacially for this work, to overcome the probleme of maintaining large volumes of dissolution fluid at constant temperature. A disgram is shown in Fig. 2.3. It consists of a cylindrical PVC container 40 cm in depth with an paper filters b and d (Whatman quantitative filter paper) supported on both sides by stainless steel filter supports a and e (Millipore cat. no. XX 3002503). Filter support, e, and filter, d, have a diameter equal to the inside diameter of C (2.2 cm) whereas a and b have the same diameter as the recess f. B and C are modified portions of a l inch Gelman, filter holder, unit (Gelman cat. no. 4320-1). C was modified from the commercial filter unit by increasing the depth of the recess, f, to contain both a and b. In addition a hole, 2.2 cm diameter was bored through the centre. A water tight pressure seal is achieved by means of the "O-ring", h.

The cell is loaded and assembled as follows: Filter paper b is positioned in the recess f, a is placed on top of it and B and C screwed together. The unit is placed on a balance pan with portion B as the base. Powder is weighed directly onto the filter paper and distributed as a thin, uniform layer. The second filter paper, d, and the support, e, are placed on top together with the stainless steel coil spring, g. The complete B - C unit (with B still downward) is placed inside the stainless steel jacket, A. The Teflon component, D, is then firmly screwed to A to form a tight seal with C by means of the "O-ring". The assembled unit is placed upright in a tripod stand with the hoses connected at both ends and the whole is immersed in the jacketed beaker (B on Fig. 2.1) for thermal equilibration 10 minutes before any run.

Heat Exchanger

The equipment for temperature regulation was also designed specially for this work, to overcome the problems of maintaining large volumes of dissolution fluid at constant temperature. A diagram is shown in Fig. 2.3. It consists of a cylindrical PVC container 40 cm in depth with an

- 18 -

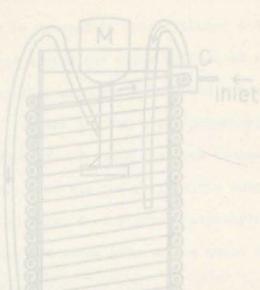
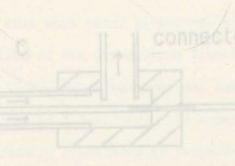
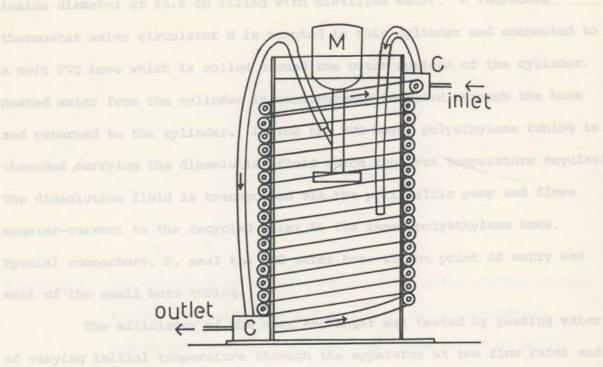


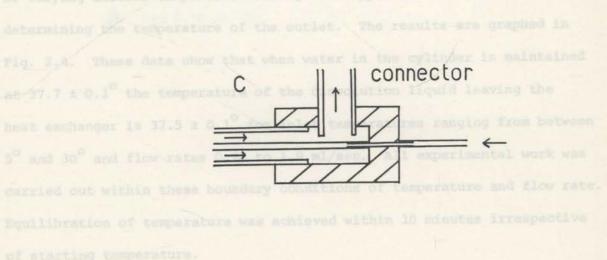
Figure 2.3 Diagram of heat exchanger: M, Thermomix thermostat water

circulator; C, connectors.









The hest exchanger is a very convenient piece of apparatus since it is highly efficient and eliminates the practical difficulties associated with maintaining large volumes of dissolution medium of constant temperature.

An inexpensive flow-cell for the spectrophotometer was also designed and is shown in schematic form in Fig. 2.5. It consists of a standard rectangular quartz cell of 1 cm path length to which is fitted inside diameter of 15.5 cm filled with distilled water. A Thermomix thermostat water circulator M is mounted in this cylinder and connected to a soft PVC hose which is coiled around the outer surface of the cylinder. Heated water from the cylinder is continuously recycled through the hose and returned to the cylinder. Inside the PVC hose, polyethylene tubing is threaded carrying the dissolution fluid which requires temperature regulation. The dissolution fluid is transported via the peristaltic pump and flows counter-current to the recycled water in the inner polyethylene hose. Special connectors, C, seal the PVC water hose at the point of entry and exit of the small bore tubing.

The efficiency of the heat exchanger was tested by passing water of varying initial temperature through the apparatus at two flow rates and determining the temperature of the outlet. The results are graphed in Fig. 2.4. These data show that when water in the cylinder is maintained at $37.7 \pm 0.1^{\circ}$ the temperature of the dissolution liquid leaving the heat exchanger is $37.5 \pm 0.1^{\circ}$ for inlet temperatures ranging from between 5° and 30° and flow rates 0.24 to 1.9 ml/sec. All experimental work was carried out within these boundary conditions of temperature and flow rate. Equilibration of temperature.

The heat exchanger is a very convenient piece of apparatus since it is highly efficient and eliminates the practical difficulties associated with maintaining large volumes of dissolution medium of constant temperature.

An inexpensive flow-cell for the spectrophotometer was also designed and is shown in schematic form in Fig. 2.5. It consists of a standard rectangular quartz cell of 1 cm path length to which is fitted

- 21 -

Figure 2.4 The variation in temperature of water, after passage through the heat exchanger, as a function of inlet temperature. Thermostat seeling 37.7°; key: ●, flow rate 0.24 ml/sec ▲, and 1.90 ml/sec.

- 22 -

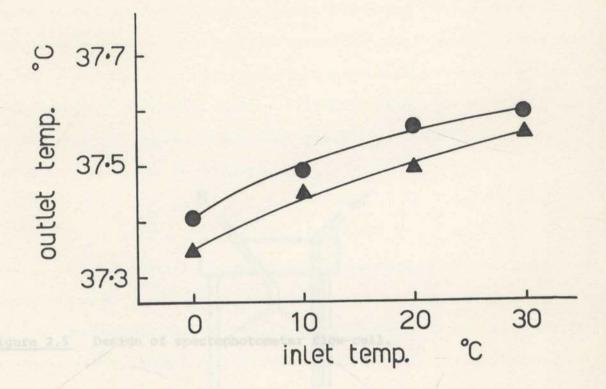
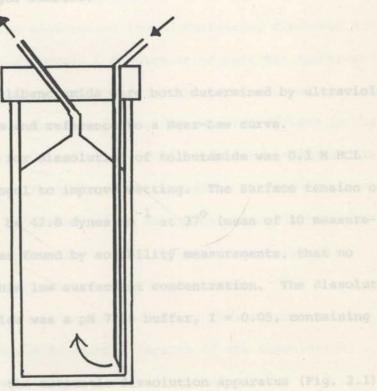


Figure 2.5 Design of spectophotometer flow-cell.

appenzophotometry at 220 nm. containing 10 H actomacrost this solution was found to 20 entity with m.d. 34). oft was solubilization occurs at this liquid used for glibonclamic



atting. The Earlace tension of poffer, I = 0.05, containing

a specially designed rectangular teflon stopper with silicon rubber seal. A narrow bore stainless steel tube passes through the stopper and extends from one side of the cell. This tube is adjustable in length and does not obstruct the light path. Dissolution medium containing dissolved solute is introduced to the bottom of the cell by this means. The solution flows upward across the light path and exits through a cone shaped outlet which assists in the removal of gas bubbles.

Analytical

Tolbutamide and glibenclamide were both determined by ultraviolet spectrophotometry at 220 nm and reference to a Beer-Law curve.

The vehicle used for dissolution of tolbutamide was 0.1 M HCL containing 10^{-5} M cetomacrogol to improve wetting. The surface tension of this solution was found to be 42.8 dynes cm⁻¹ at 37° (mean of 10 measurements with s.d. 5%). It was found by solubility measurements, that no solubilization occurs at this low surfactant concentration. The dissolution liquid used for glibenclamide was a pH 7.25 buffer, I = 0.05, containing 10^{-5} M cetomacrogol.

The operation of the automatic dissolution apparatus (Fig. 2.1) is self-explanatory from the previous description. Values of absorbance were read from the chart recording at 2.5 minutes intervals from the beginning of the experiment. These values together with the volumetric flow rate, the Beer-law constant, the initial amount used, the amount of undissolved drug remaining in the dissolution cell at the end of the experiment and a code for the particular data treatment desired were fed into a digital computer and processed according to a FORTRAN program. The amount of undissolved drug remaining was found by disconnecting the dissolution cell and transferring the filter paper/drug/filter paper "sandwich" to a volumetric flask. The drug was then dissolved in 95% ethanol and the solution assayed. This procedure increases the accuracy of the dissolution curves generated by the computer because it enables corrections to be made on the principle of mass balance.

A test of this accuracy was made using 12.5 mg 60/85 mesh fraction of tolbutamide, a flow rate of 0.149 cm/sec and 0.1 M HCL + 10^{-5} M cetomacrogol as solvent. The dissolution liquid containing dissolved drug was collected at intervals of exactly 5 minutes as it left the spectrophotometer cell (F in Fig 2.1) and assayed for drug. The cube root of the calculated amount undissolved was then plotted versus time as seen in Fig. 2.6. On the same figure are plotted values generated by the computer from the absorbance readings taken from the chart recording. It appears from the plot that the computer generated points are almost coincidental with the points obtained from direct analysis. A chi-square test showed no significant difference between the two methods (P > 0.99).

The accuracy and precision obtained using this experimental technique have made it possible to examine aspects of the dissolution kinetics which would be impossible with most other methods.

Dynamic Dialysis

The data used for the evaluation of the new method of obtaining drug-macromolecule binding parameters described in Chapter 11 was kindly supplied by Dr. M.J. Crooks (35). The dynamic dialysis technique used was that described by Meyer and Guttman (36). Chlorpropamide was dialysed from 1% bovine serum albumin in 0.067 M phosphale buffer pH 7.4 at 37° in the presence of a fixed pre-concentration of warfarin (1.6 x 10^{-5} M).

- 27 -

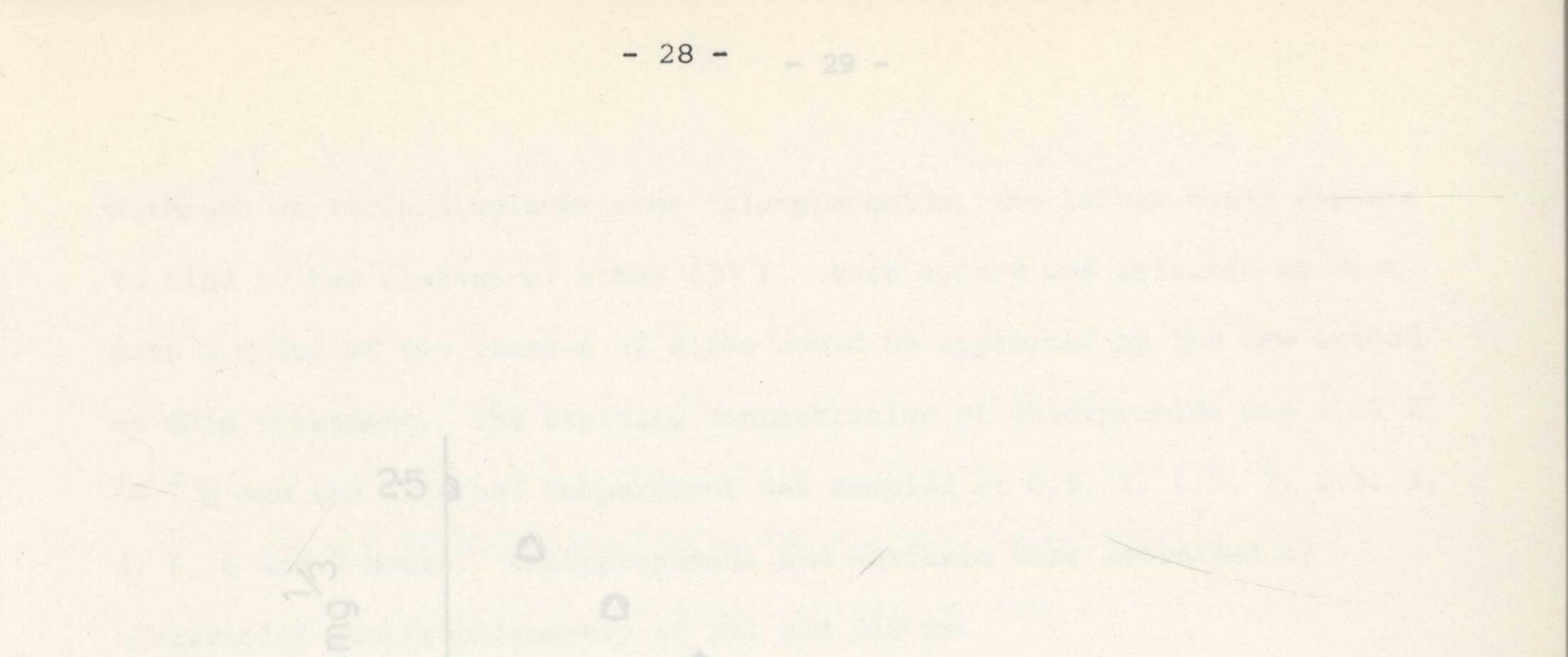
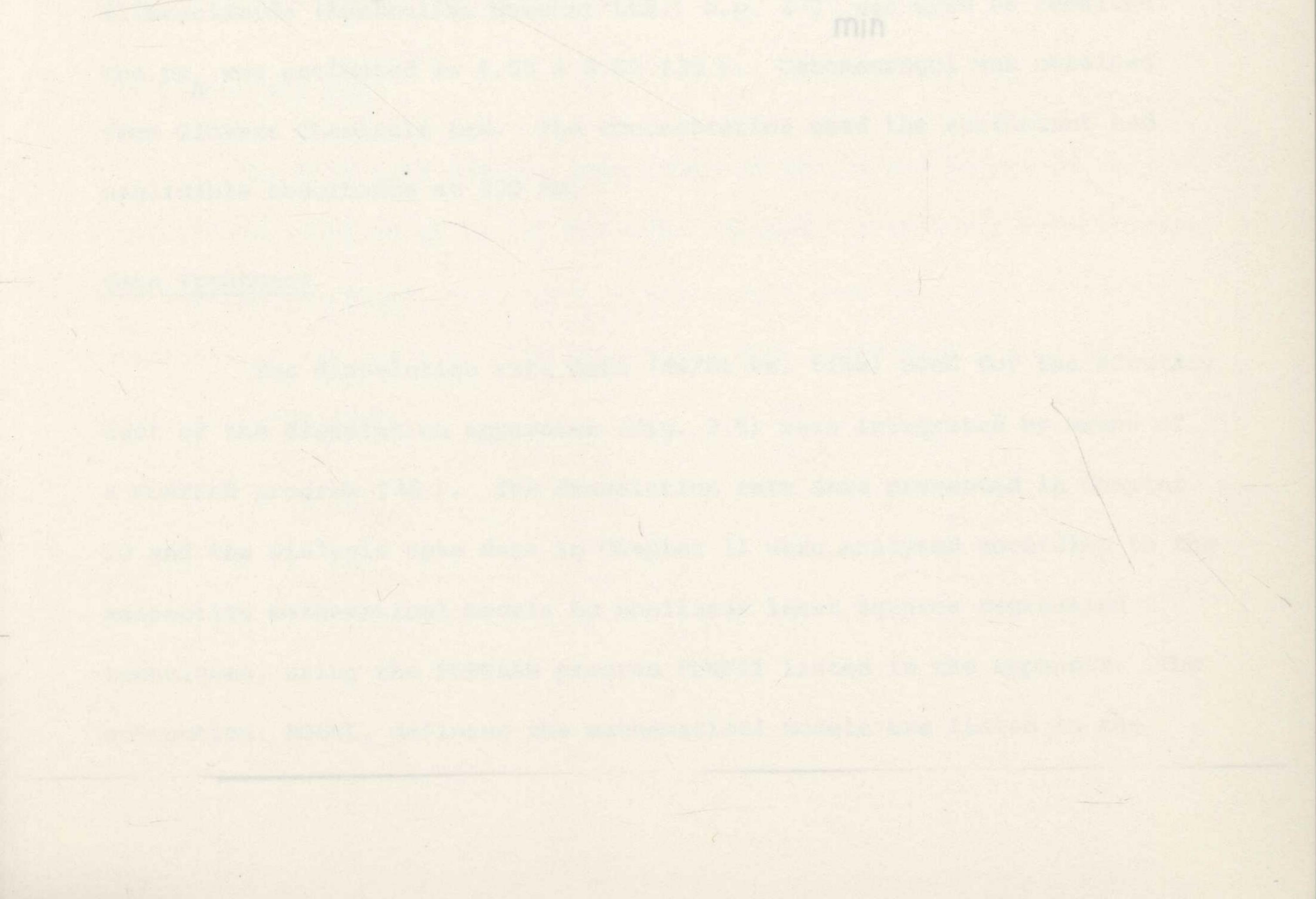
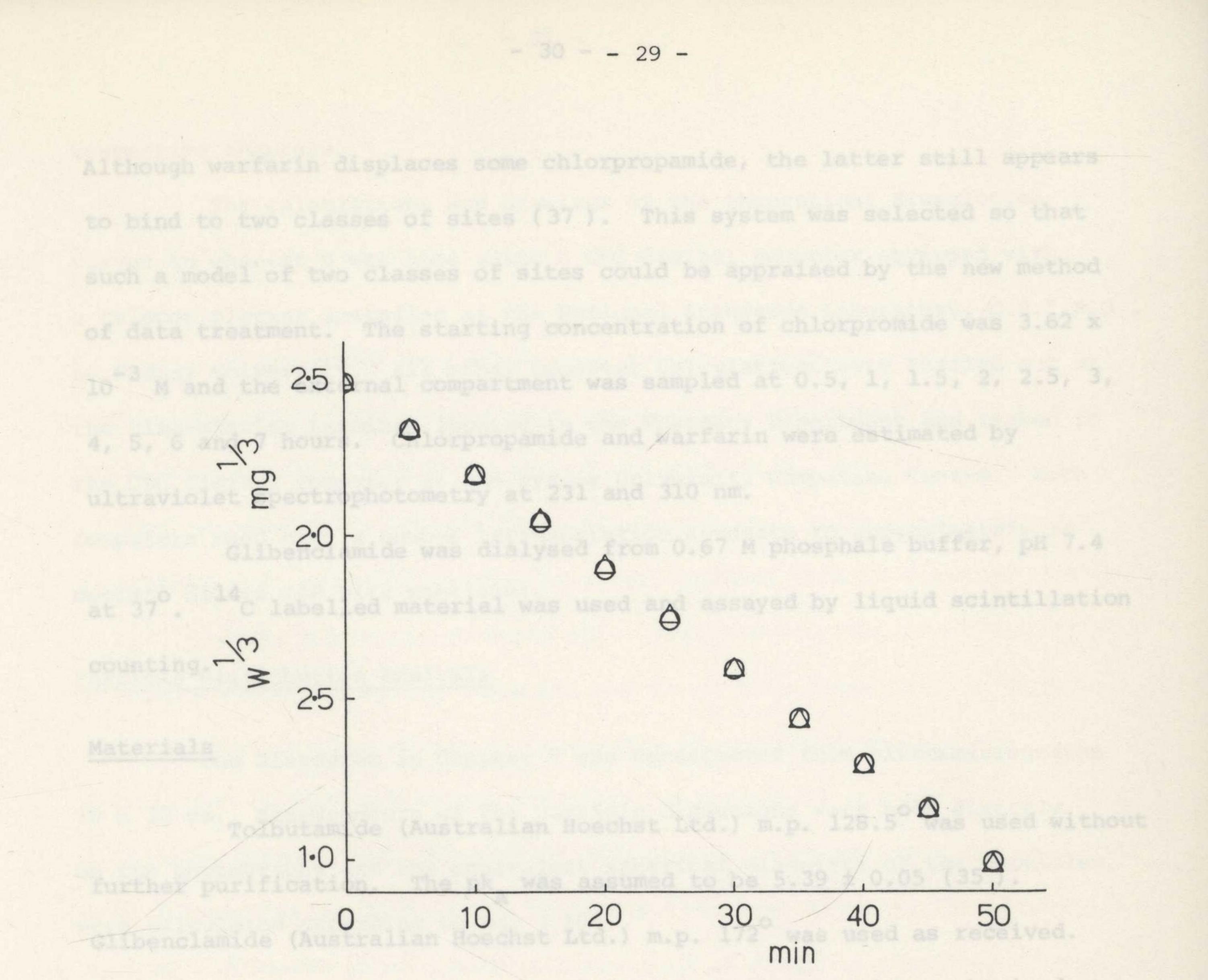


Figure 2.6 Dissolution of 12.5 mg, 60/85 mesh fraction, tolbutamide in 0.1 μ HCl containing 10⁻⁵ μ cetomacrogol using dissolution cell at solvent flow rate 0.149 cm/sec. Key: •, values calculated from intermittent samples collected at 5 min's intervals; •, values generated by digital computer from continuous chart recording of absorbance versus time.



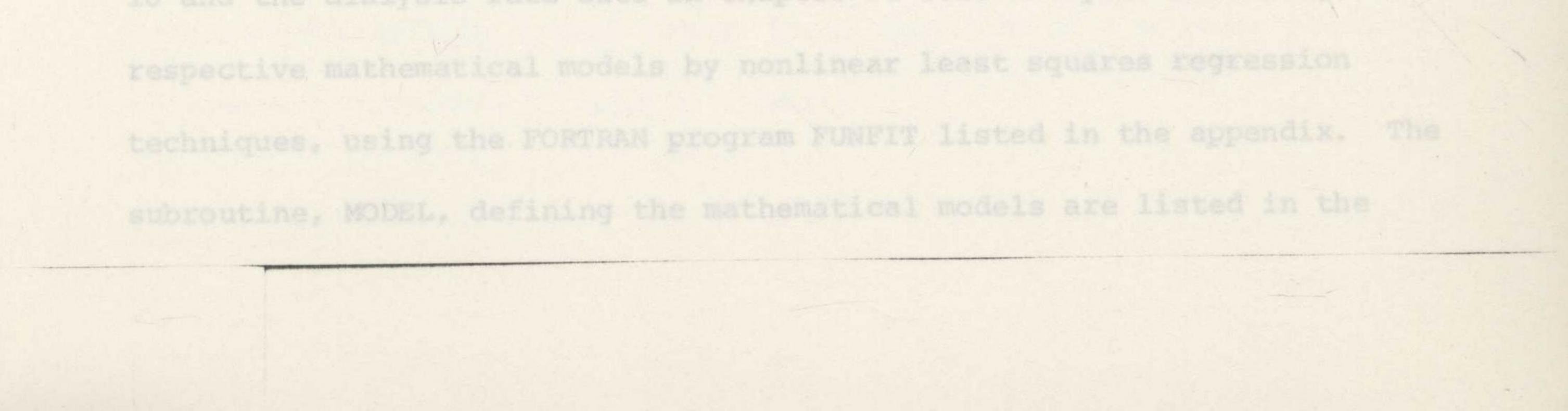


The pk, was estimated as 6.50 ± 0.05 (35). Cetomacrogol was obtained from Glovers Chemicals Ltd. The concentration used the surfactant had

negligible absorbance at 220 nm.

Data Treatment

The dissolution rate data (dw/dt vs. time) used for the accuracy test of the dissolution apparatus (Fig. 2.6) were integrated by means of a FORTRAN program (38). The dissolution rate data presented in Chapter 10 and the dialysis rate data in Chapter 11 were analysed according to the



Although warfarin displaces some chlorpropamide, the latter still appears to bind to two classes of sites (37). This system was selected so that such a model of two classes of sites could be appraised by the new method of data treatment. The starting concentration of chlorpromide was 3.62 x 10^{-3} M and the external compartment was sampled at 0.5, 1, 1.5, 2, 2.5, 3, 4, 5, 6 and 7 hours. Chlorpropamide and warfarin were estimated by ultraviolet spectrophotometry at 231 and 310 nm.

Glibenclamide was dialysed from 0.67 M phosphale buffer, pH 7.4 at 37°. ¹⁴C labelled material was used and assayed by liquid scintillation counting.

Materials

Tolbutamide (Australian Hoechst Ltd.) m.p. 128.5° was used without further purification. The pk_A was assumed to be 5.39 ± 0.05 (35). Glibenclamide (Australian Hoechst Ltd.) m.p. 172° was used as received. The pk_A was estimated as 6.50 ± 0.05 (35). Cetomacrogol was obtained from Glovers Chemicals Ltd. The concentration used the surfactant had negligible absorbance at 220 nm.

Data Treatment

The dissolution rate data (dw/dt vs. time) used for the accuracy test of the dissolution apparatus (Fig. 2.6) were integrated by means of a FORTRAN program (38). The dissolution rate data presented in Chapter 10 and the dialysis rate data in Chapter 11 were analysed according to the respective mathematical models by nonlinear least squares regression techniques, using the FORTRAN program FUNFIT listed in the appendix. The subroutine, MODEL, defining the mathematical models are listed in the respective chapters.

The calculations and drawings of the theoretical dissolution curves in chapter 6 was done using a CDC digital computer equipped with a calcomp plotter installed at the National Standards Laboratory, C.S.I.R.O., at Sydney University. All other computer calculations were carried out at the time-sharing terminal located in the Pharmacy Department and linked to the CDC digital computer at the Sydney University Computing Centre. Both computers used have a number representation accurate to approximately 14 decimal digits (48 bits mantissa).

Particle Distribution Analysis

The histogram in Chapter 7 was constructed from eltronmicrographs 30 x 30 cm. Measurements of the particle dimensions were made directly on the photographs and the equivalent spherical diameters of the particles were calculated according to Eq. 4.104.

Pharmacokinetic Data

The data used for the comparison of NONLIN and FUNFIT in Chapter 9 was kindly supplied by Mr. A. Somogyi. Although artificially generated data containing random noise just as well could have been used in such a comparison it was felt that the biological data would give a more realistic basis for comparison.

for various oral dusage forms. It is expected from the number of stops, that the availability of the dusage forms should be of the order sulution suppression > granulate > copule > tablet.

CHAPTER 3

BIOLOGICAL ASPECTS AND METHODS OF ENHANCING DISSOLUTION

Potent drugs very often have a high lipid/water partition coefficient which facilitates their penetration through biological membranes. While this property lowers the biological barrier and favours absorption it also paradoxically creates formulation problems, since these drugs are frequently only slightly soluble in water, causing them to dissolve slowly.

It is generally accepted that limited absorption after oral intake of poorly soluble drugs is often the result of slow dissolution. This kind of absorption problem occurs so frequently that bioavailability and dissolution testing has become important in evaluation and control of slowly dissolving drugs. Many examples of differences in bioavailability resulting from differences in dissolution behaviour have been given in the literature (1, 39). Research into dissolution kinetics of drugs is therefore of great importance and a better understanding of the influence of the formulation on the dissolution process is necessary to overcome these problems. The numerous reviews published recently confirm the importance of the field (2,3-5).

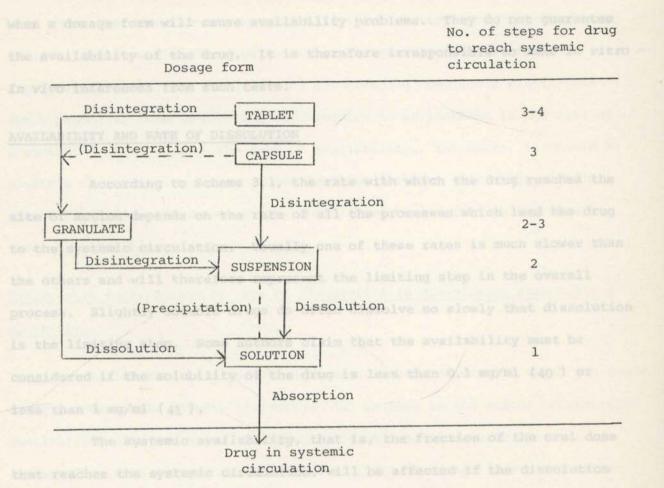
Dissolution as a step in the pathway to systemic circulation.

When a drug is given orally it has to go through a number of steps before it reaches the systemic circulation. Scheme 3.1 illustrates this for various oral dosage forms. It is expected from the number of steps, that the availability of the dosage forms should be of the order solution > suspension > granulate > capsule > tablet.

- 32 -

Scheme 3.1

Pathways to systemic circulation for oral dosage forms



Solid dosage forms are often tested *in vitro* for their disintegration or dissolution behaviour. Such tests essentially provide only information about these processes in relation to the particular experimental conditions. They are primarily useful in evaluating how the disintegration or dissolution is influenced by the drug formulation and by factors such as agitation, pH, vehicle composition etc. and are mainly of value in the development of better formulations.

The aim has often been to use the tests as a measure or indication of how the dosage form will perform *in vivo*. Most conclusions of this nature will however be unreliable because these *in vitro* tests in their very simple forms cannot properly reflect the rather complex in vivo system.

The *in vitro* tests are only able to tell in some but not all cases when a dosage form will cause availability problems. They do not guarantee the availability of the drug. It is therefore irresponsible to make *in vitro in vivo* inferences from such tests.

AVAILABILITY AND RATE OF DISSOLUTION

According to Scheme 3.1, the rate with which the drug reaches the site of action depends on the rate of all the processes which lead the drug to the systemic circulation. Usually one of these rates is much slower than the others and will therefore represent the limiting step in the overall process. Slightly soluble drugs do often dissolve so slowly that dissolution is the limiting step. Some authors claim that the availability must be considered if the solubility of the drug is less than 0.1 mg/ml (40) or less than 1 mg/ml (41).

The systemic availability, that is, the fraction of the oral dose that reaches the systemic circulation, will be affected if the dissolution rate is too slow. Much more easily affected, however, is the rate of availability, that is, the rate with which the drug enters the systemic circulation. This availability for drugs absorbed by passive diffusion will theoretically be affected in any case where the drug is not completely dissolved when it reaches the main absorption site. This implies that the dissolution rate of even relatively rapidly dissolving drugs will affect the rate of availability.

The stomach is the first place where absorption of a swallowed dosage form takes place. The absorption rate at this site is rather poor compared to the intestine due to the relatively small mucosal surface area, the limited agitation and the longer diffusion pathway. The absorption rate first becomes substantial when the drug enters the intestine. Studies have shown that the stomach emptying half-life can vary from 7 to 22 minutes (42,43). The time for the drug to enter the intestine should be significantly less. If the drug does not dissolve completely within this short period of time, any action that results in an increase in the rate of dissolution will increase the rate of availability. Therefore, it should be possible to increase the rate of availability for even relatively rapidly dissolving drugs by a change in the formulation that increases the *in vivo* rate of dissolution.

Therapeutic implications of rate of dissolution in dissolution rate limited absorption.

A slow rate of dissolution results in poor absorption which can cause several undesirable effects including: (a) changes in the extent of systemic availability with decreased therapeutic effect, (b) changes in the rate of availability with delayed onset of activity, (c) increased inter- and intrasubject variability and unpredictability of response, (d) increased residency of drug in gastrointestinal tract with increased damage to mucosal tissue (Aspirin, KCl, steroids) (39). It is therefore nearly always advantageous to formulate both slowly and readily dissolving drugs so they dissolve as fast as possible in the organism. The only exceptions are drugs with a short biological half-life where a slow release dosage form is desirable to avoid the inconvenience of too frequent drug intake. The effect of dissolution rate on pharmacological activity has been discussed by Levy (44).

- 35 -

Factors affecting drug dissolution from solid dosage forms.

The dissolution of a drug from a solid dosage form is influenced by several factors that can be summarised as shown in Scheme 3.2.

Scheme 3.2

Factors affecting drug dissolution from solid dosage forms

- I Disintegration
 - II Environmental factors
 - (1) agitation
- (2) Dissolution media
- (a) drug concentration and gradient
 - (b) pH
 - (c) viscosity
 - (d) interfacial tension
- only of the complexation of the second s
 - (f) solubilization

III Factors related to the drug itself

- (a) solubility
- (b) polymorphism
- (c) solvation
- agatation (45-40) (d) salt form deen to be any way the in with agatation
- (e) particle size
- (f) particle size distribution

It is possible by proper formulation techniques to influence the majority of these factors such that a higher *in vivo* dissolution rate can be achieved.

Disintegration

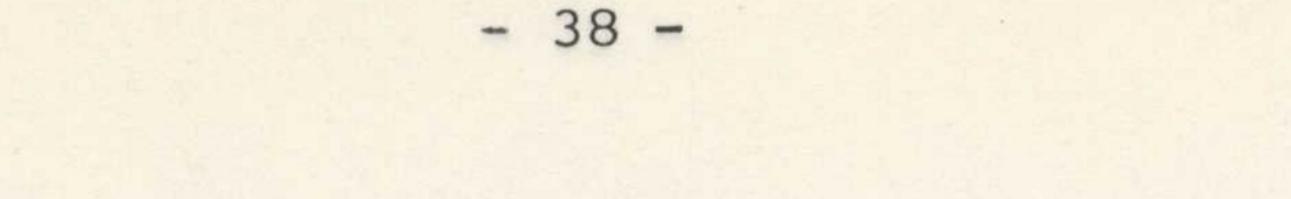
A rapid disintegration of the dosage form (tablet, capsule) is necessary in order to achieve fast dissolution. The factors influencing disintegration have been reviewed by Wagner (45). The aim of disintegration is to make the drug particles available for dissolution as quickly and Noyes and Whitney have shown that the rate of dissolution of a efficiently as possible. Therefore, a disintegration test should include a drug is proportional to its solubility (51). Most drugs are weak acids test for both disintegration time and disintegration efficiency. This or bases and their solubility is dependent on the pH of the vehicle. B efficiency is measured in terms of the availability of disintegrated particles use of the Menderson-Basselbalch equation it can be shown that the for dissolution. A conventional disintegration (time) test is therefore lubility, C, of a weak base having solubility, C , in unionised form is actually only useful when it is combined with a dissolution test. mendent on the hydrogen ion concentration The factors influencing dissolution (Scheme 6.2) can be divided into environmental factors and intrinsic factors, the latter being factors related to the drug itself. The environmental factors can be influenced either by additives in the dosage form or by the administration of the drug e.g. whether it is given before, or after a meal or with or without liquid etc. The intrinsic factors can be manipulated by physical or chemical means The addition of small amounts of alkaline buffer substands only e.g. by change in particle size or by making a salt-form of the drug. of weak acids results in a higher pH in the vicinity of the drug particles

Environmental factors

Agitation absorption should also be considered. It is unfortunate that the effect of

It is well known that dissolution rate increases with increasing pH on the dissolution rates of acids and bases is opposite to the effect of agitation (46-48). There does not seem to be any way the in vivo agitation pH on the intrinsic absorption tates of these weak electrolytes. The absorpt can be influenced by drug formulation. It is possibly too drastic to include of a weak acid from solution is optimal at low pH where the rate of other drugs that increase gastric emptying rate and gastrointestinal dissolution of the weak acid is lowest. On the other hand, the rapid motility. It would be more worthwhile to consider the dependence of agitation on the composition of the food. It has been shown that meals of low viscosity ecause little absorption occurs in this part of the gastrointestinal tract. are emptied more rapidly than meals of high viscosity (49) and that fats

and fatty acids inhibit gastric motility (50). The physical activity of the patient also plays a role. Walking generally produces a higher It has been shown by several investigators (53-55) that the gastrointestinal motility than lying (1). dissolution rate decreases with increasing viscosity of the vehicle. There



pH s not seen to be any way by which the formulation of a solid drug-

Noyes and Whitney have shown that the rate of dissolution of a drug is proportional to its solubility (51). Most drugs are weak acids or bases and their solubility is dependent on the pH of the vehicle. By use of the Henderson-Hasselbalch equation it can be shown that the solubility, C, of a weak base having solubility, C, in unionised form is

dependent on the hydrogen ion concentration as follows:

$$C = C_{0}(1 + K_{b}(H^{+})/K_{w})$$
 (3.1)

where K_{W} is the ion product of water and K_{D} is the dissociation constant of the base. The corresponding expression for a weak acid is:

$$C = C_{0}(1 + K_{a}/(H^{+}))$$
 (3.2)

The addition of small amounts of alkaline buffer substances to formulations of weak acids results in a higher pH in the vicinity of the drug particles which can enhance the dissolution of the drug (52). Buffers can in this way help to increase the dissolution rate, but their effect on drug absorption should also be considered. It is unfortunate that the effect of pH on the dissolution rates of acids and bases is opposite to the effect of pH on the intrinsic absorption rates of these weak electrolytes. The absorption of a weak acid from solution is optimal at low pH where the rate of dissolution of the weak acid is lowest. On the other hand, the rapid dissolution of a weak base in the stomach acid content is not so important because little absorption occurs in this part of the gastrointestinal tract.

Viscosity day relight polyola gras and cellulose derivatives that are able

It has been shown by several investigators (53-55) that the

dissolution rate decreases with increasing viscosity of the vehicle. There

does not seem to be any way by which the formulation of a solid drug (tablet, capsule) can influence significantly the viscosity of the gastrointestinal contents. The implication of viscosity and viscosityenhancing agents used in suspension on drug absorption has been discussed by Gibaldi (56).

Interfacial tension

It is well recognised that the dissolution rate is proportional to the effective surface area of the drug, i.e. the surface area available for dissolution. The effective surface area for hydrophobic drugs is usually considerably less than the real surface area because the interfacial tension between the solid and the liquid does not allow complete wetting. The effective surface area can however be increased by the addition of a surface active agent that facilitates wetting and hence results in an increase in the dissolution rate. It was shown by Finholdt and Solvang (57) that the gastric juice in most humans contains surface active agents and has a surface tension that is considerably less than the 0.1M HCl that is often used in *in vitro* experiments. *In vitro* investigations of the effect of surfactants on the dissolution rate will therefore probably result in overly optimistic expectations about the *in vivo* effect.

Complexation

The effect of complexation on the dissolution rate of drugs has been extensively investigated (58-60). Most soluble macromolecules such as high molecular weight polyols, gums and cellulose derivatives that are able to form complexes with drugs, can increase the water solubility of the drugs. There will not, however, be a proportional increase in the dissolution rate

- 39 -

since the drugs have to dissolve before complex formation can take place. One investigator explains that the increased dissolution rate observed in the presence of certain complex forming agents is possibly caused by a lowering of the energy change for transferring drug molecules from crystal to solution (58). The viscosity increasing properties of macromolecular complex forming agents will, however, reduce the diffusion rate of drugs and often counteract the former effect, causing a possible decrease in the dissolution rate. Complexing agents have primarily been used in dosage forms in order to increase the solubility or stability of a drug rather than to influence its dissolution behaviour. It is important to consider the implications of complex formation on drug absorption before any such agents are used (56,61).

Solubilization

The effect of solubilizing agents on the dissolution process has been investigated by several authors (62-64). The increase in the solubility due to micellar solubilization will usually not result in a proportional increase in the dissolution rate. The significant increase in the rate sometimes found is probably caused by the wetting effect rather than the solubilizing effect. It is therefore apparent considering Finholdt and Solvangs' investigations (57) that these agents are limited in their ability to increase the *in vivo* dissolution rate. They are mainly used for technical reasons in production but are also occasionally used for improving absorption, although there still seems to be controversy as to whether they actually enhance or retard drug absorption (56).

- 40 -

FACTORS RELATED TO THE DRUG ITSELF

Solubility

It was early recognised that the rate of dissolution in a diffusioncontrolled dissolution process is proportional to the solubility (51). Several papers have discussed dissolution rate in relation to drug solubility (65-67). It has been pointed out that the solubility of very small particles increases with decreasing particle size because of an increase in vapor pressure of the solid (66).

The solubility of a drug is determined by the interaction between the solid drug and the solvent. There are therefore two ways in which the solubility and hence the dissolution rate can be manipulated: by changing the vehicle composition, for example by use of buffering agents to alter pH in the vicinity of the drug particles, as discussed previously; or by changing the crystal form.

Polymorphism

Most drugs can exist in at least two crystal forms. In certain classes of compounds the incidence of polymorphism is even greater. The metastable crystal forms have a lower melting point and a higher solubility than the stable forms. The increased solubility generally results in an increased dissolution rate. The amorphous form represents the highest energy level of the molecules in solid form. A pronounced difference in therapeutic activity between the amorphous and crystalline forms of drugs has been observed in several cases (67,68). This difference can only be due to a difference in the *in vivo* dissolution rate of the two forms, because a drug has to dissolve before it can be absorbed and the properties of the dissolved

sophylline and gluterhimids have been observed (75), The

drug do not depend on its original crystal structure. The above observations therefore strongly confirm the role of the dissolution rate in the absorption process. The exploitation of polymorphism often provides an effective way documented that these usually have faster dissolution and absorption rates of enhancing the dissolution rate. Polymorphism of drugs has for this reason been subject to considerable investigations in recent years (17,40,65). The much more rapidly than the corresponding acid forms, regardless of the initial problem in using the most soluble crystal form of a drug to achieve faster dissolution is that this form is often metastable and can change to a stable in terms of the ability of these salts to increase the pH at the drug-liquid and less soluble form, possibly producing severe consequences (67). Drugs in amorphous or other metastable forms should therefore be used only when layer. Some drugs may subsequently re-precipitate in the bulk fluid, but it can be ensured that they will not change to less therapeutically active crystal forms. further dilution or absorption in the gastointestinal tract (44). Chemical Solvation de la case precide the use of the drug in salt form.

The sodium salt of aspirin is very unstable in solution and even the solid A more reliable approach for enhancing the dissolution rate may be form is rather unstable. The socium or potassium salts may react with to use the drug in a more soluble solvated or non-solvated form. Significant

differences in the dissolution rate of anhydrous and hydrated forms of caffeine, theophylline and glutethimide have been observed (75). The and absorption rates (76,77). The alkalinity of some salts may furthermore anhydrous forms dissolved faster in these three cases. The n-pentanol and ethylacetate solvates of fluorocortisone dissolved faster, however, than the non-solvated forms (72). The use of solvated or non-solvated forms of a drug to increase the dissolution rate can be troublesome, since conversion Particle size reduction is undoubtedly the most used and important between forms sometimes occurs during or subsequent to the formulation of a product. Such conversions are temperature-dependent and in the case of hydrates, humidity-dependent as well. Therefore a complete study of the forms should

be done under different temperature and humidity conditions before they are used in pharmaceutical preparations.

Salt form

Several drugs can exist in the form of salts and it is well documented that these usually have faster dissolution and absorption rates than their parent compounds (73-75). Sodium salts of weak acids dissolve much more rapidly than the corresponding acid forms, regardless of the initial

Their fast dissolution in the low pH of the stomach can be explained pH.

in terms of the ability of these salts to increase the pH at the drug-liquid interface, causing a fast release and high concentration in the diffusion layer. Some drugs may subsequently re-precipitate in the bulk fluid, but usually in the form of very fine particles that readily dissolve through further dilution or absorption in the gastointestinal tract (44). Chemical stability can in certain cases preclude the use of the drug in salt form. The sodium salt of aspirin is very unstable in solution and even the solid form is rather unstable. The sodium or potassium salts may react with atmospheric carbon dioxide and water to precipitate out poorly soluble

parent compounds. This occurs on the surface and thereby retards the dissolution

and absorption rates (76,77). The alkalinity of some salts may furthermore

cause epigastric distress after oral intake of the drug in a solid form.

Particle size

Particle size reduction is undoubtedly the most used and important method of increasing the dissolution rate. The surface area to weight ratio of a particle of any shape varies inversely with its diameter. Therefore

the total surface area and hence also the effective surface area of a drug

powder will increase substantially by particle size reduction, in particular

for fine particles. The greater effective surface area of the drug in contact with the gastrointestinal fluid will result in more rapid dissolution and

- 44 -

absorption. This has led most drug firms to micronize (particle size $< 25 \mu$) very slightly soluble drugs for use in oral dosage forms. There are numerous reports of the better dissolution and availability of micronised drug compared to crystalline (78-82). Particle size reduction of a drug does not always influence its systemic availability. It was found for example that 50, 200, 400 and 800 μ powders of chloramphenicol were absorbed to the same extent. The 50 μ and 200 μ powders showed essentially the same absorption

rate or rate of availability with a peak blood level at one hour, while the

400 μ and 800 μ powders were different with peak blood levels at two and

three hours respectively (83). These findings indicate that the in vivo

dissolution rate of the 50 μ and 200 μ powders has been fast enough to enable both powders to dissolve before they reach the main absorption site. However, this was not the case for the 400 μ and 800 μ powders that had different rates of availability. Therefore, it can be concluded that nothing has been gained in therapeutic efficacy by reducing the particle size to less than 200 µ.

Such findings are of value in situations where it is better to use coarse

rather than fine particles because of production or stability reasons. A particle size reduction to enhance dissolution should seriously be considered when the absorption site is in the stomach or upper region of the intestine. If, on the other hand, the site is in the last section of the intestine, drug absorption may be nearly independent of particle size. This results when dissolution occurs before this section is reached, and depends on the stomach emptying rate and the peristaltic activity (2).

Relatively large particle sizes are often needed to give local

action in the terminal part of the gastrointestinal tract (84,85). A

weak organic basic drug will often rapidly dissolve in the form of an

hydrochloride in the acidic content of the stomach. However, when it passes into the slightly basic content of the intestine, it can precipitate out

as the un-ionized compound. The precipitation will be in the form of very fine, rapidly dissolving particles, so the absorption rate of the compound will be independent of the initial particle size (52). The absorption of a slightly soluble weak acid drug should, on the other hand, be much more dependent on particle size. A reduction in particle size will largely facilitate its dissolution in the acidic gastric juice, where it is only slightly soluble. It then forms the more soluble salt when it passes into the duodenum and small intestine. However, the salt is highly ionised and its absorption rate is decreased, since only a small fraction of the drug molecules is in the undissociated state.

Such considerations regarding the effect of particle size reduction in relation to the acidic or basic character of the drug are based on the pH-partition theory for absorption and should therefore be subject to discussion and further investigation. Particle size reduction can in certain cases reduce the therapeutic efficacy of a drug if it is unstable in gastric juice, as for example penicillin G or erythromycin, fast dissolution will enhance degradation in the stomach. The effect of particle size in relationship to absorption and activity has been reviewed by Fincher (2).

Particle size reduction can be performed in several ways: (a) trituration, (b) ball milling, (c) fluid energy micronization or (d) controlled precipitation including spray drying (86). There are, however, limitations as to how much the particles can be reduced in size and how suitable an extreme particle size reduction would be with respect to formulation and dissolution. Fine particles very often show a strong tendency to aggregate and agglomerate due to their increased surface energy and the stronger van der Waals attraction between non-polar molecules. Furthermore, electrostatic charges in fine hydrophobic powders can cause severe technical

- 45 -

difficulties in production. The problems associated with the wettability of fine powders have already been discussed. Drugs with plastic properties are difficult to subdivide by mechanical means (a-c), since they tend to stick together, even if produced by controlled precipitation (d).

Lin et al. (87) found that the *in vitro* dissolution rates of micronized griseofulvin and glutethimide were slower than those of their coarser particles. The opposite finding was reported by Chiou and Riegelman for griseofulvin (88). The results of Lin et al. can be explained by the strong agglomeration and reduced wettability of the micronized powder. The previously mentioned investigations of Finholdt and Solvang (57) indicate that Chiou and Riegelman's results better reflect what is expected *in vivo*. Several reports about the better availability of drugs in micronized form strongly support this (78-80).

Drugs can also be introduced to the gastrointestinal fluids in the form of very fine particles formed by precipitation *in vivo*. This can be done in several ways (52). The drug can be dissolved in a non-aqueous water-miscible solvent from which it precipitates out by dilution in the gastrointestinal tract. Solutions of sodium or potassium salts of an acidic drug will in the same way precipitate out at the low pH existing in the stomach. Formulations resulting in *in vivo* drug precipitation should have better availability than formulations of drugs in micronized form, because agglomeration and wettability problems are avoided and the particle size of the precipitate usually will be smaller than that which can be produced by micronization. However, the many disadvantages of liquid drug formulations compared to solid with respect to stability and convenience do not seem to counteract the advantage of the somewhat better availability.

- 46 -

Solid Dispersions

Drug formulations based on solid dispersions of slightly soluble in a supersaturated form which, through the slow diffusion process occurring drugs have received considerable attention in recent years. The in solids, can precipitate out during storage, particularly at elevated pharmaceutical application of these systems has been extensively reviewed superatures (90). The aging of solid dispersions should be an important by Chiou and Riegelman (89). Solid dispersions are micro-crystalline or rmaceutical scientist, hoperbuly resulting in molecular dispersions of a poorly soluble drug in a solid water-soluble

matrix. The drug either precipitates out in a fine particle form which is readily available for dissolution or becomes solubilized when the matrix rapidly dissolves in the gastrointestinal juice. Solid dispersions may be classified as: (a) simple-eutectic mixtures, (b) solid solutions, (c) glass solutions and glass suspensions, (d) amorphous dispersion in crystalline carriers and various combinations of these systems. Several reports indicate that the availability and dissolution rate of these formulations are superior to formulations containing micronized drugs (90,91). Solid dispersions appear to be as available as the liquid formulations from which the drug

precipitates out in vivo, but they do not have the disadvantages of the latter

with respect to their convenience. The stability also seems to be much

better although very little investigation has been done in this field (89).

Solid dispersions appear, therefore, to be a very promising new

approach in the formulation of slightly soluble, slowly dissolving drugs.

The major problem in designing this formulation form lies in finding a

suitable production method and in finding a matrix material with the right

properties such as low toxicity, high water solubility, high solubilizing

capacity for the drug, thermal and chemical stability and suitability for

processing. Possibly the main hindrance to the introduction of this

Solid dispersions often contain formulation form is the problem of aging.

the drug in a high energy, metastable form which, under prolonged storage,

can be transformed into a stable but less soluble and more slowly dissolving form. Solid solutions (molecular solid dispersions) often contain the drug in a supersaturated form which, through the slow diffusion process occurring in solids, can precipitate out during storage, particularly at elevated temperatures (90). The aging of solid dispersions should be an important research subject for the pharmaceutical scientist, hopefully resulting in the commercial acceptance of this unique dosage form.

Particle size distribution

In recent years there has been increasing interest in the effect of particle size distribution on the dissolution behaviour of drugs (92-96). The reduced systemic availability observed for certain drug formulations can possibly be explained in terms of a "size distribution effect". In most drug powders there is a considerable difference between the sizes of the larger and smaller particles, which produces an important difference in their time for complete dissolution. This time is for a particle, dissolving according to Hixson-Crowell's cube root law (97), proportional to its initial The larger particles can because of their manifold longer diameter. dissolution time in such cases reach the distal part of the intestine and only be partly dissolved. The further dissolution in this section of the intestine with its content of increased viscosity, solid matter and low water content will be rather slow and the drug will possibly get eliminated from the body by defaecation before completion of its dissolution. Therefore in order to avoid such availability problems it would be good practice to have standards for the particle sizes of slightly soluble drugs.

- 0 0 0 -

CHAPTER 4

SINGLE PARTICLE DISSOLUTION

There are three dominant models describing the interfacial mass transport in a dissolving dynamic (agitated) system. These are:

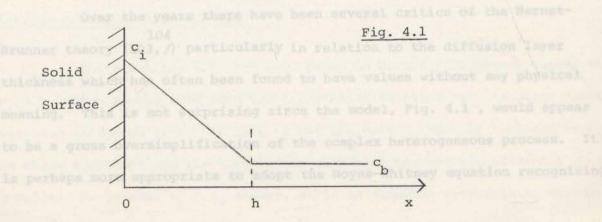
1. Nernst stagnant film theory (98).

- 2. Danckwerts surface renewal theory (99).
- 3. Interfacial solvation rate limited dissolution theory (100,101).

NERNST FILM THEORY

The theory of Nernst assumes a stagnant layer of solvent at the solid-liquid interface. The mass transport of solute through the layer is accomplished by simple molecular diffusion in a steady state fashion following Fick's law of diffusion. Once passed the stagnant layer the solute is then mixed quickly by convection and diffusion in the bulk of the liquid.

The Nernst model predicts a concentration profile from a plane surface in two dimensions as shown schematically in Fig. 4.1, where h refers to the thickness of the so-called diffusion layer, c_i is the interfacial concentration and c_b the bulk concentration. The concentration gradient in the stagnant layer, $o \le x \le h$, is then constant because the diffusion flux,



J, is constant during steady state:

$$J_{\rm D} = - \,{\rm Ddc/dx} = {\rm constant} \tag{4.1}$$

From geometrical considerations (fig. 4.1):

$$dc/dx = -(c_1 - c_1)/h$$
 (4.2)

(4.3)

so that $J_{D} = (D/h) (c_{i}-c_{b})$

J

If it is assumed that the interfacial reaction rate is large compared to the rate of diffusion then c_i can be considered close to the solubility concentration c_o so 4.3 can be approximated by:

$$\sim (D/h) (c_s - c_b)$$
 (4.4)

The dissolution rate from a plane surface of area, A, will then be:

$$dw/dt = -(DA/h)(c_{c}-c_{h})$$
 (4.5)

If in 4.5 D/h is assumed constant, the equation reduces to the Noyes-Whitney equation:

$$dw/dt = -kA (c_{s} - c_{b})$$

$$(4.6)$$

after the workers who appear to be the first to have verified this equation experimentally (51). They did not attach any mechanistic significance to the quantity k. Nernst and Brunner (102) extended the Noyes-Whitney equation to include the concept of diffusion layer thickness and diffusion coefficient as presented in 4.5.

Over the years there have been several critics of the Nernst-104 Brunner theory (103,/) particularly in relation to the diffusion layer thickness which has often been found to have values without any physical meaning. This is not surprising since the model, Fig. 4.1, would appear to be a gross oversimplification of the complex heterogeneous process. It is perhaps more appropriate to adopt the Noyes-Whitney equation recognising that the quantity k depends in some way on D, the hydrodynamic conditions and the geometry of the dissolving object. It has been shown that k in Eq. 4.6 under fixed experimental conditions can often be considered constant and the equation has been successfully applied in several cases.

- 51 -

DANCKWERTS THEORY

Danckwerts (99) rejected the idea of a stagnant film and proposed a model based on the assumption that liquid motion is turbulent and extends to the surface of the dissolving crystal. The physical interpretation is that pockets of fresh solvent reach the interface by turbulence and renew parts of the surface, while 'old' pockets containing solute simultaneously leave the surfaces. The mathematical description of this system includes a parameter, Y, known as the mean rate of surface renewal:

 $dw/dt = A \sqrt{\gamma D} (c_{c} - c_{h})$

(4.7)

and the process is truly reaction rate controlled.

For the dissolution of small particles Goyan (105) proposed the following

model which combines 4.5 and 4.7 :

$$dw/dt = -A (D/r + \sqrt{\gamma D}) (c_s - c_b)$$
 (4.8)

When the radius of the particle, r, diminishes as $\gamma \rightarrow 0$ this equation reduces to 4.5 (in the case r = h).

LIMITED SOLVATION RATE THEORIES

The equations 4.5 and 4.6 based on Nernst film theory assume that

$c_{i} ~ \ c_{s}$ which should hold when the interfacial reaction rate is large compared

with the diffusion rate. If this is not the case c must be replaced by ci

in these equations and the dissolution mechanism becomes significantly more

complex. In general c, < c, always, so it is somewhat difficult to decide

between diffusion and reaction rate controlled dissolution.

Wurster and Taylor (104) proposed a method based on the temperature dependence of k. They claim that the process is diffusion-controlled if the temperature coefficient is approximately 1.3 and interfacial reaction rate controlled if it is close to 2.0. Higuchi (4) derived the following equation for an interfacial reaction rate and diffusion rate controlled process:

$$dw/dt = -\frac{AD}{h + D/k_{i}} (c_{s}-c_{b})$$
(4.9)

where k_i was termed the effective interfacial transport rate constant. This equation was presented in slightly modified form in a later publication (60). It is seen that when the effective interfacial transport rate constant is large i.e. when $k_i >> D/h$ the equation reduces to 4.5 and the process is diffusion rate limited. If $k_i << D/h$ equation 4.9 reduces to:

$$dw/dt = -Ak_i (c_s - c_b)$$
 (4.10)

and the process is truly reaction rate controlled.

SPHERICAL DISSOLUTION

Nernst theory was derived for a plane interface. It is of interest to apply similar assumptions to a spherical particle, i.e. assume (Fig. 4.2):

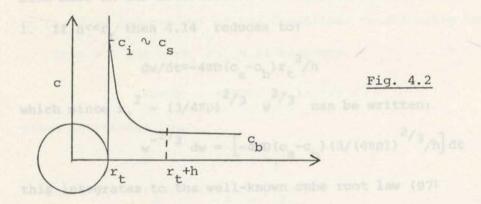
1. Spherical symmetry in dissolution

2.
$$c = c_1 \circ c_s as r = r_+$$

3.
$$c = c_1 at r = r_+ + h$$

- 4. Solute transported only by Ficks diffusion $(J_D = -D\partial c/\partial r)$ in stagnant layer, $r_t \le r \le r_t + h$
- quasi steady state conditions in the sense that the mass transport rate through a spherical surface in the diffusion

or:
$$[d(r^2dc/dr)_t = 0$$
 (4.11)
 $[d(r^2dc/dr)/dr]_t = 0$ (4.12)



Solving 4.12 under boundary conditions 2 and 3 above leads to:

$$(dc/dr)_{t} = -(c_{s}-c_{b}) r_{t} (r_{t}+h)/(hr^{2}) r_{t} \leq r \leq r_{t}+h$$
 (4.13)

From this expression, it is seen that the concentration gradient in the diffusion layer is *not* constant as it was in the case for the plane body (Fig. 4.1). The dissolution rate of the spherical particle according to assumption 4 is thus given by:

$$dw/dt = 4\pi r_t^2 D(dc/dr)_{r=r_t} = -4\pi D(c_s - c_b)(r_t + h)r_t/h$$
 (4.14)

The diffusion layer thickness, h, is related to the intensity of the solvent flow or agitation in the vicinity of the spherical interface. This agitation will depend in some way on the size, r_t, of the particle. Therefore, some functional relationship must exist between the thickness of the diffusion layer and the radius of the dissolving particle.

Several functional relationships between h and r_t can be assumed. On this basis, integration of 4.14 will then lead to different models for single particle dissolution as demonstrated in the following section.

Constant Diffusion Layer Thickness.

Firstly, it would be of interest to investigate the case where h is constant independent of particle size, since this assumption has often been made in the literature. Two cases will be considered:

1. If h<<rt then 4.14 reduces to:

$$dw/dt \simeq -4\pi D(c_s - c_b) r_t^2/h$$
 (4.15)

which since $r_t^2 = (3/4\pi\rho)^{2/3} w^{2/3}$ can be written: $w^{-2/3} dw = \left[-4\pi D (c_s - c_b) (3/(4\pi\rho))^{2/3}/h\right] dt$ (4.16)

this integrates to the well-known cube root law (97)

$$w^{1/3} = w_0^{1/3} - k_1 t$$
 (4.17)

with:
$$k_1 = (4\pi/3)^{1/3} D(c_s - c_b) \rho^{-2/3}/h$$
 (4.18)

2. If h>>rt then 4.14 reduces to:

to the

$$dw/dt \simeq -4\pi D(c_s - c_b)r_t$$
(4.19)

which similarly to above integrates to:

$$w^{2/3} = w^{2/3} - k_2 t$$
 (4.20)

where
$$k_2 = 2(4\pi/3)^{2/3} D\rho^{-1/3} (c_s - c_b)$$
 (4.21)

It is interesting to note that in this case the diffusion layer thickness, h, is not a part of the rate parameter k_2 .

Variable Diffusion Layer Thickness.

It is expected from hydrodynamic considerations that h decreases with increasing r. Any monotonically decreasing functional relationship between h and r could be considered. This investigation will be restricted to simple relationships based on the following power model:

$$k_{p}, p > 0$$
 (4.22)

where k_p and p are some positive constants. Several cases can be considered depending upon the value of p and the size of h relative to r_t . It should first be noted that in the cases where $h^{>>r}_t$ Eq. 4.14 still leads to Eq. 4.20 independent of the functional relationship between h and r_t . 1. If p = 1 and $h^{<r}_t$ Eq. 4.14 becomes:

$$dw/dt = -4\pi D(c_s - c_b) r_t^3 / k_p$$
 (4.23)

which integrates to:

 $h = k_{p} r_{t}^{-p}$

$$w = w EXP (-k_3t)$$
 (4.24)

$$k_{2} = 3D(c_{2}-c_{1})/k_{p}$$
 (4.25)

where

2. If p = 1 and h is not very different in magnitude from $r_t Eq. 4.14$ leads to the following expression:

$$\frac{2}{3} = (w_0^{2/3} + k_4) \text{ EXP } (-k_3 t) - k_4$$
 (4.26)

where $k_4 = k_p (4\pi\rho/3)^{2/3}$ (4.27)

and k₃ is given by 4.25.

3. If p = 2 and $h \le r_t$ the following equation is obtained:

$$v^{1/3} = (w_0^{-1/3} + k_5 t)^{-1}$$
 (4.28)

where
$$k_5 = 4\pi D(c_s - c_b) (4\pi \rho/3)^{4/3}/3k_p$$
 (4.27)

APPROACHES NOT BASED ON DIFFUSION LAYER ASSUMPTIONS

The Noyes-Whitney model applied to a spherical particle takes the form:

$$dw/dt = -kA (c_s - c_b)$$
 (4.28)

where k may or may not be constant or depend on the particle radius. 1. If k = constant 4.28 integrates to yield the model proposed by Hixson and Crowell (97):

$$w^{1/3} = w_0^{1/3} - k_6 t$$
 (4.29)

where
$$k_6 = k(c_s - c_b) (4\pi/3)^{1/3} \rho^{-2/3}$$
 (4.30)

2. If k is inversely proportional to the particle radius, say, $k = k_7/r_t$ then 4.28 integrates to:

$$v_{0}^{2/3} = w_{0}^{2/3} - k_{g}t$$
 (4.31)

where
$$k_8 = 2k_7(c_s - c_b) (4\pi/3)^{2/3} \rho^{-1/3}$$
 (4.32)

3. If k is inversely proportional to the square root of the particle radius, i.e. $k = k_9 r_+^{-l_2}$, Eq. 4.28 integrates to yield:

$$w^{\frac{1}{2}} = w_{0}^{\frac{1}{2}} - k_{10}t$$

$$k_{10} = k_{9}(2\pi)^{\frac{1}{2}}(2\rho/3)^{-1/3} (c_{s}-c_{b})$$
(4.33)
(4.34)

Sink Conditions.

The models for single particle dissolution presented above are derived for dissolution under sink condition. This condition can best be explained in a mathematical context to be a condition under which the change in c_b is so small that the errors, introduced by considering (c_s-c_b) or (c_i-c_b) to be constant, are acceptable. This definition is related to the concept of ignoring any time dependence of the concentration terms in the integration of the dissolution rate equation.

Comparing Models.

It is interesting to note that several of the dissolution models above could be derived both on the basis of the diffusion layer theory and

1. These workers did not specify the composition of k_6 but assumed it did not vary with the progress of the dissolution.

on the more simple model 4.28. In both approaches certain assumptions were made about the relationship between either the diffusion layer thickness, h, and the particle size or the rate parameter k (4.28) and the particle size.

In spite of oversimplification of the dissolution mechanism in the diffusion layer approach, it is appealing since it has greater conceptual value than the other approach. The flexibility of models based on a diffusion layer seems also to be greater.

Of the models considered above the cube root model (4.17, 4.29), the square root model (4.33) and the 2/3-root model (4.20, 4.31) have all been proposed in the literature (97,106,107).

There still seems to be some controversy as to which of these models best describes dissolution of the single particle. This is mainly because there appears to be no accurate experimental information directly concerning the dissolution of single particles.

The inferences made have been based on multiparticle dissolution from which it may be difficult to deduce single particle dissolution behaviour, particularly when the particle size distribution is not completely monodisperse (see Chapter 6).

In the experimental evaluation of the 1/3, 1/2 and 2/3 root models in Chapter 7, the cube root model appears to be the best although further investigation is necessary to confirm this finding. It may well be that a more flexible model applies which is close to the 1/3 root model in the beginning and approaches the 1/2 and 2/3 root model as the dissolution process progresses. Such behaviour is expected if the dissolution follows Eq. 4.14. This equation reduces to the 1/3 model if $h << r_t$, which is likely to be true in the initial stages of dissolution if the particle size then is relatively large. Later when the particle is reduced in size the hydrodynamic activity in the vicinity of the interface will be considerably smaller resulting in an increased h value so that for h>>r+ the dissolution reaction approaches the 2/3 root model. The intermediate state will result

in a "square-root dissolution". This condition should occur when

 $(r_t^{+h})r_t^{/h} \sim r_t^{3/2}$ i.e. when $h \sim r_t^{2/(r_t^{3/2} - r_t)}$.

STATIC DISSOLUTION MODELS

form, (100), i.e., that p = 1. The term c_ is the concentration infinitely Study of single particle dissolution in which there is no agitation is of interest for dissolution in vehicles of extremely high viscosity in which the solute mass transport in most cases is completely diffusion-controlled. If a spherical particle is considered the system is mathematically well defined. Substantial theoretical studies have been performed in this area in chemical engineering (108-113). The many different mathematical approaches presented have been based on different assumptions. For example, the 112 assumption that the process is wholly diffusion controlled (109,/) or partly diffusion controlled (113). Most of the mathematical models have been

described in terms of partial differential equations which for given boundary conditions can only be solved numerically using a digital computer.

It is interesting to note that the radius vs. time relationship

computed in this way by Cable and Evans (109) and Ready and Cooper (112) is approximately linear except for an initial transient period. This means that the dissolution of a spherical particle under static conditions can be approximated (disgregarding the initial phase) by the cube root law. (The 1/3 model implies a linear decrease in radius with time.) Of the equations presented for static spherical dissolution, the

following equations given by Rosner (113) are appealing because of their B.C.1. $\mathbf{c} = \mathbf{c}$ $\mathbf{r} = \mathbf{r}$ $\mathbf{t} > \mathbf{0}$

 $B.C.2. c = 0 z > z_{-} t = 0$

(4.38)

simplicity:

 $dr/dt = (D\alpha/r) \ln ((1-c_{\infty})/(1-c_{i}))$ (4.35)

$$k (c_s - c_i)^P = (D/r) \ln ((1 - c_{\infty})/(1 - c_i))$$
 (4.36)

Where α is the ratio of solvent density/solute density, k is the rate

constant governing the interface kinetics, p is the order of the solvation reaction. It was assumed that this reaction was of the ordinary Berthroud form, (100), i.e., that p = 1. The term c_{∞} is the concentration infinitely

far from the centre of the particle, which is the same as the initial concentration. In most cases therefore $c_m = 0$. The above two equations define, in a computationally simple way, the variation of particle radius with time for given values of D, α , c and r. For example 4.36 can be solved numerically for c; and this value substituted into 4.35 which by numerical integration provides the r_-value for the chosen value of t. Such calculations were performed on a computer by Ridgway and Peacock (114) for various simple organic compounds. The validity of the above two equations depends primarily on the

following extended quasi steady-state assumption: "The instantaneous concentration field surrounding the dissolving sphere is approximated by the steady-state concentration field surrounding a hypothetical spherical solid of the same size, through which solute is being artificially forced at a mass rate equal to the instantaneous rate of dissolution". If the interfacial concentration c; is constant and close to the solubility concentration c the dissolution can be obtained directly by It is interesting and sat plactory to note that this model was also derived solving Fick's law: for the dynamic system (4, 20), considering the diffusion layer theory for (4.37)

the case h>>r, i.e. for a system with little agitation. In fact the rate

$$\partial c / \partial t = DV^{-}c$$

under the following boundary conditions:

B.C.1.
$$c = c_{s}$$
 $r = r_{t}$ $t > 0$ (4.38)
B.C.2. $c = 0$ $r > r_{o}$ $t = 0$ (4.39)
B.C.3. $c = 0$ $r = \infty$ (4.40)

This yields the following expression for the concentration profile in the liquid (115):

$$c(r,t) = c_{s} \frac{r_{t}}{r} \operatorname{erfc} \frac{r-r_{t}}{2\sqrt{Dt}}$$
(4.41)

so that the mass flux per unit surface area at the interface is:

$$J_{i} = -D(\partial c/\partial r)_{r=r_{t}} = Dc_{s} \left[(\pi Dt)^{-\frac{1}{2}} + 1/r_{t} \right]$$
(4.42)

Furthermore:

$$dw/dt = -4\pi r_t^2 J_i$$
 (4.43)

Now since $dw/dt = d (4\pi r_t^3 \rho/3)/dt = 4\pi \rho r_t^2 dr_t/dt$, 4.42 becomes:

$$dr_{t}/dt = -J_{t}/\rho$$
 (4.44)

i.e.
$$dr_t/dt = -\frac{Dc_s}{\rho} \left[(\pi Dt)^{-\frac{1}{2}} + 1/r_t \right]$$
 (4.45)

This equation can readily be integrated numerically to yield the time dependence of r_t and hence a numerical solution for the single particle dissolution for given values of ρ , D and c_c .

As dissolution proceeds the term $(\pi Dt)^{-\frac{1}{2}}$ in the bracket in 4.45. decreases while in the same time the term $1/r_t$ increases. Thus after the initial transition period $((\pi Dt)^{-\frac{1}{2}} + 1/r_t) \sim 1/r_t$ and 4.45 integrates to yield the 2/3-root law:

$$w^{2/3} = w_0^{2/3} - k_{11} t$$
(4.46)
$$k_{11} = 2(4\pi/3)^{2/3} D \rho^{-1/3} c_{5}$$
(4.47)

where

It is interesting and satisfactory to note that this model was also derived for the dynamic system (4.20), considering the diffusion layer theory for the case $h^{>>}r_t$ i.e. for a system with little agitation. In fact the rate parameters k_{11} and k_2 in 4.47 and 4.21 are identical for $c_b = 0$.

A DYNAMIC MODEL FOR A FREE FALLING SPHERICAL PARTICLE

It is very difficult to define, qualitatively or quantitatively, the agitation in the close vicinity of a particle in a mechanically stirred liquid. A better defined experimental system would be the examination of a spherical particle which falls freely through an unstirred liquid. Analysis of such a system should be particularly valuable in the study of single particle dissolution.

In the following section an attempt is made to describe this system in a mathematically rigorous way. In order to do so the following assumptions are made and will be referred to where necessary in the derivation: Al. The particle remains spherical during dissolution.

- A2. The interfacial concentration, c_i, is constant and very close to the saturation concentration, c_c.
- A3. The temperature and densities remain constant.
- A4. There is no c /r dependence.
- A5. The effect of Brownian motions is negligible.

A6. The particle falls without producing turbulence, i.e. the Reynold's number, Re, is less than 0.1:

$$Re = 2r_{+}v_{m}\rho_{1}/\mu < 0.1$$
 (4.48)

(The velocity of the particle in the direction of gravity is denoted v_{∞} , the viscosity of the liquid μ and its density ρ_1 .)

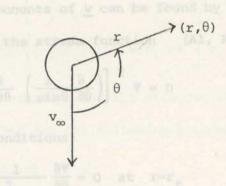
- A7. The fluid is incompressible.
- A8. The fluid flow behaves in a Newtonian way.

A9. Fick's laws of diffusion are obeyed with D=constant.

Spherical symmetry makes it convenient to choose a spherical coordinate system with the particle centre as origin and the θ coordinate

starting counter-clockwise from the direction of the gravity (Fig. 4.3)

Fig. 4.3



First it is necessary to derive an extension of Fick's law valid for incompressible liquids (A7.) in motion: A shell mass balance on an arbitrary volume element, p, in the bulk of the liquid yields (A9.):

$$\iiint \frac{\partial c}{\partial t} dp = \iint (-\underline{v} \cdot \underline{n} c + \underline{D} \nabla c \cdot \underline{n}) ds \qquad (4.49)$$

where s is the surface of the element and \underline{n} is the normal to the surface directed outward and \underline{v} is the fluid velocity vector. The divergence theorem applied on r.h.s. of 4.49 yields:

$$\iiint_{p} \partial c/\partial t \, dp = \iiint_{p} \left(-\underline{\nabla} \cdot \underline{v}c + D\nabla^{2}c \right) dp \qquad (4.50)$$

which also may be written:

$$\iiint_{p} \partial c/\partial t \, dp = \iint_{p} \left(-\underline{v} \cdot \underline{\nabla} c - c \underline{\nabla} \cdot \underline{v} + D \nabla^{2} c \right) dp \qquad (4.51)$$

Equating left and right integrand in 4.51, noting $\nabla \cdot v = 0$ (A7.) it follows that:

 $\partial c/\partial t = D\nabla^2 c - \underline{v} \cdot \underline{\nabla} c$ (4.52)

This is the extended form of Fick's second law which takes into account that the liquid in which diffusion takes place is in motion. The fluid velocity vector function \underline{v} above depends on the geometry of the particle and the velocity v_{∞} with which the particle is falling through the liquid. This velocity can be related through Stokes' equation to the particle radius r_+ .

The r and θ components of <u>v</u> can be found by solving the following differential equation for the stream function (Al, A3, A5-A8):

$$\left[\frac{\partial^2}{\partial r^2} + \frac{\sin\theta}{r^2} \frac{\partial}{\partial \theta} \left(\frac{1}{\sin\theta} \frac{\partial}{\partial \theta}\right)\right]^2 \Psi = 0$$
(4.53)

subject to the boundary conditions:

B.C.1
$$v_r = -\frac{1}{r^2 \sin \theta} \frac{\partial \Psi}{\partial \theta} = 0$$
 at $r = r_t$ (4.54)

B.C.2.
$$v_{\infty} = \frac{1}{r\sin\theta} \frac{\partial \Psi}{\partial r} = 0 \text{ at } r = r_t$$
 (4.55)

B.C.3.
$$\Psi \rightarrow -\frac{1}{2}v_{\infty}r^{2}\sin^{2}\theta$$
 for $r \rightarrow \infty$ (4.56)

The velocity components are then obtained

$$v_{r} = -v_{\infty} \left[1 - \frac{3}{2} \left(\frac{r}{r} \right) + \frac{1}{2} \left(\frac{r}{r} \right)^{3} \right] \cos\theta \qquad 0 \le \theta \le \pi \qquad (4.57)$$
$$v_{\theta} = v_{\infty} \left[1 - \frac{3}{4} \left(\frac{r}{r} \right) - \frac{1}{4} \left(\frac{r}{r} \right)^{3} \right] \sin\theta, \qquad 0 \le \theta \le \pi \qquad (4.58)$$

The r and θ components of ∇c in 4.52 is $\partial c/\partial r$ and $r^{-1} \partial c/\partial \theta$ respectively thus:

$$\underline{\mathbf{v}} \cdot \underline{\nabla} \mathbf{c} = -\mathbf{v}_{\infty} \left[1 - \frac{3}{2} (\mathbf{r}_{t}/\mathbf{r}) + \frac{1}{2} (\mathbf{r}_{t}/\mathbf{r})^{3} \right] \frac{\partial \mathbf{c}}{\partial \mathbf{r}} \cos\theta \qquad (4.59)$$

$$+ \mathbf{v}_{\infty} \left[1 - \frac{3}{4} (\mathbf{r}_{t}/\mathbf{r}) - \frac{1}{4} (\mathbf{r}_{t}/\mathbf{r})^{3} \right] \mathbf{r}^{-1} \frac{\partial \mathbf{c}}{\partial \theta} \sin\theta$$

When there is no ϕ -dependence the ∇^2 operator in spherical coordinates is:

$$\nabla^{2} = \frac{1}{r^{2}} \frac{\partial}{\partial r} \left(r^{2} \frac{\partial}{\partial r} \right) + \frac{1}{r^{2} \sin \theta} \frac{\partial}{\partial \theta} \left(\sin \theta \frac{\partial}{\partial \theta} \right)$$
(4.60)

The velocity v_{∞} is related to r_t by Stokes' law:

$$v_{\infty} = (2\Delta \rho g/9\mu) r_t^2$$
 (4.61)

where g is the acceleration of gravity and $\Delta \rho$ the solid-liquid density difference. Equations 4.52 and 4.59-61 can then be summarized to give:

$$\frac{\partial c}{\partial t} = \frac{D}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial c}{\partial r} \right) + \frac{D}{r^2 \sin \theta} \frac{\partial}{\partial \theta} \left(\sin \theta \frac{\partial c}{\partial \theta} \right)$$

- $(2\Delta \rho g / 9\mu) \left[1 - \frac{3}{2} (r_t / r) + \frac{1}{2} (r_t / r)^3 \right] r_t^2 \frac{\partial c}{\partial r} \cos \theta$ (4.62)
+ $(2\Delta \rho g / 9\mu) \left[1 - \frac{3}{4} (r_t / r) - \frac{1}{4} (r_t / r)^3 \right] r_t^2 \frac{\partial c}{\partial \theta} \sin \theta$

which is the differential equation for the dissolution of spherical particle. The equation is to be solved for the following boundary conditions:

- B.C.1. $c(r_{t}, \theta, 0) = c_{s}$ (4.63)
- B.C.2. $c(\infty, \theta, t) = 0$ (4.64)

B.C.3.
$$c(r, \theta, 0) = 0$$
 $r > r_{+}$ (4.65)

Because 4.62 is a nonlinear partial differential equation, it seems impossible to solve analytically using conventional techniques. The equation can, however, be solved numerically using a digital computer. It would then be convenient first to transform the variables involved to dimensionless quantities such as:

4.66)
4.67)
4.68)
4.69)
	4.67 4.68 4.69

It can then be shown that 4.62 becomes:

$$\frac{\partial c^{*}}{\partial t^{*}} = \frac{\partial}{r^{*}} \left(r^{*} \frac{2\partial c^{*}}{\partial r^{*}} \right) + \frac{\partial}{\partial u} \left((1 - u^{2}) \frac{\partial c^{*}}{\partial u} \right)$$

$$- \frac{1}{2} \operatorname{Sc} \operatorname{Re}^{\circ} \left[1 - \frac{3}{2} (r_{t}^{*}/r^{*}) + \frac{1}{2} (r_{t}^{*}/r^{*})^{3} \right] r_{t}^{*2} u \frac{\partial c^{*}}{\partial r^{*}}$$

$$+ \frac{1}{2} \operatorname{Sc} \operatorname{Re}^{\circ} \left[1 - \frac{3}{4} (r_{t}^{*}/r^{*}) - \frac{1}{4} (r_{t}^{*}/r^{*})^{3} \right] r_{t}^{*} (1 - u^{2}) \frac{\partial c^{*}}{\partial u}$$

$$(4.70)$$

where Sc is the Schmidt number (Sc = $\mu/\Delta\rho D$) and Re^O is the initial Reynold number (Re^O = 2 r_ov_o^O ρ_1/μ). The transformed equation is then to be solved under the following conditions:

B.C.1.	$c^{*}(1, u, 0) = 1$	(4.71)
B.C.2.	$c^{*}(\infty, u, t^{*}) = 0$	(4.72)
B.C.3.	$c^{*}(r^{*}, u, 0) = 0$ $r^{*} > 1$	(4.73)

The numerical solution obtained by solving 4.70 for various values of ScRe^O will then be general in the sense that it encompasses an infinite number of combinations of values for the parameters μ , ρ_1 , $\Delta\rho$, D and c_s.

NONSPHERICAL ISOTROPIC DISSOLUTION

The models for single particle dissolution considered previously in this chapter are all derived for spherical particles. Application of such models to real particle systems is complicated by the fact that pure drug particles are not spherical. The usual approach has been to treat the real particles as if they were spherical having the same surface area or volume. Such approximations may introduce substantial errors.

The influence of shape factors on dissolution kinetics of drugs 117 has been discussed for tablets and controlled release tablets (116,/), but little attention has been given to single drug particles (118,119).

This section presents exact isotropic single-particle dissolution equations for several nonspherical forms and formulas enabling calculation of the diameters of hypothetical spherical particles which closely approximate the dissolution of these forms.

Assume that dissolution takes place istropically, that is, that the rate of dissolution per unit surface area, J, is constant so that the following equation can be written:

$$dw/dt = -JA \tag{4.74.}$$

where w is the amount undissolved and A is the surface area. This equation implies that the boundary of a *plane* interface retreats with constant speed during dissolution such that:

$$ds/dt = -J/\rho \tag{4.75.}$$

where ρ is the solid density, and s is the distance perpendicular to the interface from some fixed reference point in the dissolving solid.

Equation 4.75. integrates to:

S

$$s = s - Jt/\rho \qquad (4.76.)$$

where s is the initial (t=o) distance to the reference point. When this equation is applied to the istropic dissolution of a spherical particle, the following equation arises:

$$w/w_{o} = (1 - Jt/r_{o}\rho)^{3}$$
(4.77.)
(w/w_{o})^{1/3} = 1 - Jt/r_{o}\rho (4.78.)

or

where r is the initial particle radius.

Therefore, when a spherical particle dissolves istropically (J=constant), it obeys the Hixson and Crowell (97) cube root law, that is, a plot of $(w/w_{_{\rm C}})^{1/3}$ vs. is linear.

Dissolution of Prismatic Particles.

Structures I-VI (Fig. 4.4.) are 10 simple forms of the six crystal systems and illustrate the dimensional quantities b_0 , c_0 , l_0 , h_0 and α used in the following derivations. It is assumed, without loss of generality, that $b_0 < c_0 < l_0$.

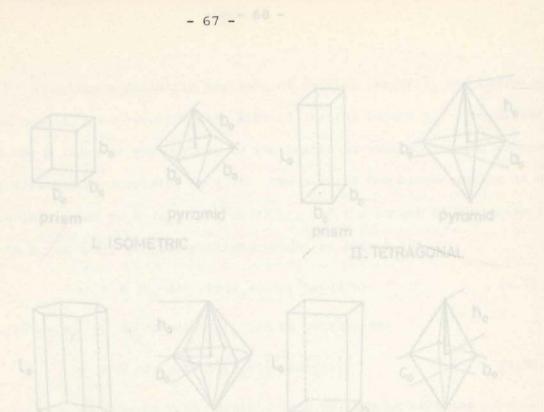
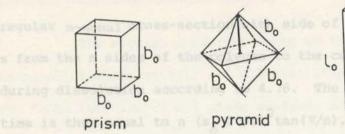
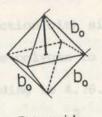
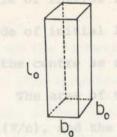


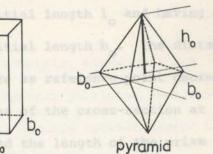
Figure 4.4 Illustration of the six basic crystalforms for which spherical approximations to the theoretical isotropic dissolution are considered.



I. ISOMETRIC

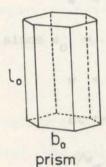


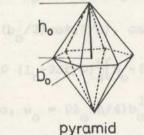


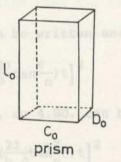


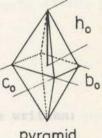
prism

II. TETRAGONAL



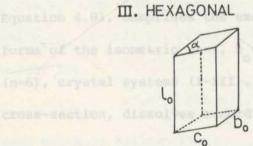




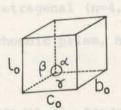


pyramid

IV. RHOMBIC



V. MONOCLINIC



VI. TRICLINIC

 $w/w_0 = (1 - \frac{10^{10}}{10^{10}}) \left[1 + (\frac{20}{000} \min(t) + 1) \left[1 - (\frac{20}{000} \min(t) + 1) \right] \right]$

Consider a prismatic particle of initial length l_0 and having a regular n-gonal cross-section with side of initial length b_0 . The distance, s from the n sides of the polygan to the centre as reference point decreases during dissolution according to 4.76. The area of the cross-section at any time is then equal to n $(s_0^{-Jt/\rho})^2 \tan(\pi/n)$, and the length of the prism is equal to $l_0^{-(2Jt/\rho)}$, so that particle weight at time t is:

$$w = \rho (1 - 2Jt/\rho) n(s - Jt/\rho)^{2} tan(\pi/n)$$
(4.79.)

which, since $s_0 = (b_0/2) \cot(\pi/n)$, can be written as:

$$a = \rho \left(1_{0}^{-2Jt} / \rho \right) \frac{n}{4} \left[b_{0}^{-} \left(\frac{2J}{\rho} \tan \frac{\pi}{n} \right) t \right]^{2}$$
(4.80.)

at t=o, $w_0 = \rho l_0 (n/4) b_0^2$, so 4.80. can be written:

$$w/w_{o} = (1 - \frac{2J}{l_{o}\rho}t) \left[1 - (\frac{2J}{b_{o}\rho}tan\frac{\pi}{n})t\right]^{2}$$
 (4.81.)

Equation 4.81. comprises the exact dissolution equation for the prismatic forms of the isometric (n=4, $b_0 = 1_0$), tetragonal (n=4, $b_0 = 1_0$), and hexagonal (n=6), crystal systems (I-III). The rhombic prism, having a rectangular cross-section, dissolves according to:

$$w = \rho(b_0 - 2Jt/\rho) (c_0 - 2Jt/\rho) (1_0 - 2Jt/\rho)$$
(4.82.)

because of the istropic retreat of all surfaces. Similarly to 4.81., this equation can be written:

$$w/w_{o} = (1-2Jt/b_{o}\rho) (1-2Jt/c_{o}\rho) (1-2Jt/1_{o}\rho)$$
 (4.83.)

The monoclinic prismatic particle, having a parallelogram crosssection with an acute angle α , at any time has a cross-sectional area equal to bc sin α , where b=b_o - [(2J/ ρ)sin α]t and c=c_o - [(2J/ ρ)sin α]t, so it dissolves according to:

$$w = \rho \left(1_{o}^{-2Jt}/\rho\right) \left[b_{o}^{-}\left(\frac{2J}{\rho}\sin\alpha\right)t\right] \left[c_{o}^{-}\left(\frac{2J}{\rho}\sin\alpha\right)t\right]\sin\alpha \qquad (4.84.)$$

$$w/w_{o} = \left(1-\frac{2Jt}{1_{o}\rho}t\right) \left[1-\left(\frac{2J}{b_{o}\rho}\sin\alpha\right)t\right] \left[1-\left(\frac{2J}{c_{o}\rho}\sin\alpha\right)t\right] \qquad (4.85.)$$

or:

Initially. The,

Dissolution of Pyramidal Particles.

The regular pyramidal forms of the isometric, tetragonal and hexagonal systems (I-III) all dissolve like spherical particles, following the "cube root law": all plane surfaces of the pyramid retreat toward its centre of symmetry with the same constant speed during istropic dissolution. Therefore, the shape of the pyramid remains the same while its size diminishes. For example, it can be shown geometrically that all lengths of the prism decrease by a factor of $(1-Jt/r_0\rho)$, where r_o , given by:

$$f_{0} = \frac{1}{2} h_{0} b_{0} (h_{0}^{2} + b_{0}^{2}/4)^{-\frac{1}{2}}$$
 (4.86.)

is the radius of the largest sphere that can be contained in the pyramid initially. The weight of the regular n-gonal prism at time t is equal to $\frac{1}{6} \rho nh_t b_t^2$ cot (π/n), where the height, h_t , and side, b_t , are $h_t = h_0 (1 - Jt/r_0 \rho)$ and $b_t = b_0 (1 - Jt/r_0 \rho)$, respectively, according to the above theory. Thus, its weight is:

$$w = \frac{1}{6} \rho nh_{o} b_{o}^{2} (1 - Jt/r_{o} \rho)^{3} \cot(\pi/n)$$
 (4.87.)

from which it follows that:

 $w/w_{0} = (1-Jt/r_{0}\rho)^{3}$ (4.88.)

This equation is identical to 4.77. Therefore, a regular pyramidal crystal form dissolves in identical manner to the largest (hypothetical) spherical particle that can be contained within its boundaries initially. This is also approximately true for an irregular pyramidal form such as the rhombic pyramid when the irregularity is not too extreme. It can be shown, using a double integration approach, that this crystal form dissolves according to:

$$w/w_{0} = (1-Jt/\rho r_{1})^{2} \left[1-(\frac{1}{4r_{1}}+\frac{3}{4r_{2}})\frac{J}{\rho}t \right]$$
 (4.89.)

where

$$r_{1} = \frac{1}{2}h_{o}b_{o}(h_{o}^{2} + b_{o}^{2/4})^{-\frac{1}{2}}$$
(4.90.)
$$r_{2} = \frac{1}{2}h_{o}c_{o}(h_{o}^{2} + c_{o}^{2/4})^{-\frac{1}{2}}$$
(4.91.)

The deviation from spherical particle dissolution (4.77.) arises from the fact
$$b_0 \neq c_0$$
. If $b_0 = c_0$, then 4.89. reduces to the special cases 4.88. as

expected.

The

To evaluate these single-particle dissolution equations, it is convenient to present them in a transformed simplified form which better illustrates their intrinsic dissolution profile (96). For example 4.81. can be transformed to:

$$w/w_{0} = (1-Ft^{*}) (1-t^{*})^{2}$$
 (4.92.)

$$(w/w_{o})^{1/3} = (1-Ft^{*})^{1/3} (1-t^{*})^{2/3}$$
 (4.93.)

where:

or

$$t^* = (\frac{2J}{b_0 \rho} \tan \frac{\pi}{n}) t$$
 (4.94.)

is denoted time length and:

$$F = \frac{b}{l_0} \cot (\pi/n)$$
 (4.95.)

is denoted the shape ratio. This form of the equation clearly shows that the intrinsic dissolution profile depends only on the value of the dimensionless shape ratio, F, which defines the particle shape. Furthermore, the transformation makes it more convenient to evaluate the extent to which dissolution of the prismatic particle deviates from spherical particle dissolution (i.e. from the cube root law). For F=l i.e. when $(w/w_{2})^{1/3} = 1-t^{*}$, there is no such deviation; however, as F decreases, the deviation becomes more significant, i.e., when the length of the particle relative to its side length or "diameter" becomes more extreme.

It is seen (Fig. 4.5.) that as F decreases, the deviation from the cube root law becomes larger. Dissolution then approaches "the square root law", that is, a linear relationship between $(w/w_0)^{\frac{1}{2}}$ and time length (or time). This is in agreement with the fact that, for small F values, 4.92. approximates $(w/w_0)^{\frac{1}{2}} = 1-t^*$.

The cube root law and the square root law were each postulated previously as a model for the dissolution of spherical particles under sink conditions (97,106). Pure drug particles are not spherical, however, but are often prismatic in shape. Therefore, the particle shape effect should be considered in any experimental evaluation of such models.

The dissolution equation for a rhombic pyramidal particle, 4.89., can also be transformed to 4.92. where:

then:

$$t^{*} = \frac{2J}{\rho} (h_{o}^{2} + b_{o}^{2/4})^{-\frac{1}{2}} t \qquad (4.96.)$$
and:

$$F = \frac{1}{4} + \frac{3}{4} \left(\frac{h_{o}^{2} + c_{o}^{2/4}}{h_{o}^{2} + b_{o}^{2/4}} \right)^{\frac{1}{2}} \qquad (4.97.)$$

It is seen (Fig. 4.6.) that the shape ratio, F, for this pyramidal particle form does not deviate much from 1 for most shapes, indicating that in most cases dissolution closely approximates that of a spherical particle.

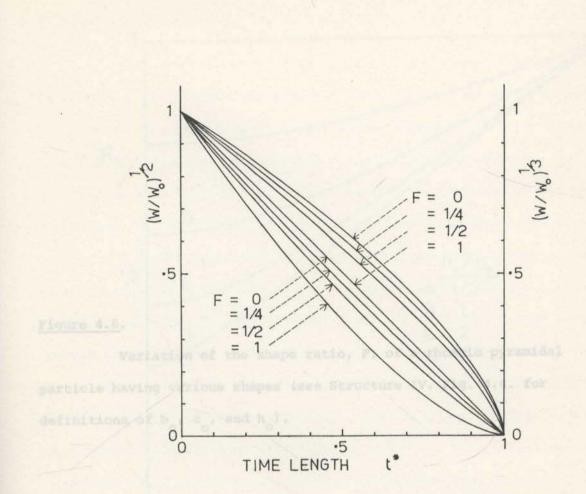
Dissolution equations for either a rhombic (4.83.) or a monoclinic (4.85.) particle can similarly be written in a common transformed form as:

$$w/w_{0} = (1-F_{1}t^{*})(1-F_{2}t^{*})(1-t^{*})$$
 (4.98.)

where $F_1 = b_0/c_0$, $F_2 = b_0/1_0$, and $t^* = 2Jt/b_0\rho$ for a rhombic particle and $F_1 = b_0/c_0$, $F_2 = (b_0/1_0)\sin\alpha$, and $t^* = (2J/b_0\rho)\sin\alpha$ t for a monoclinic particle.

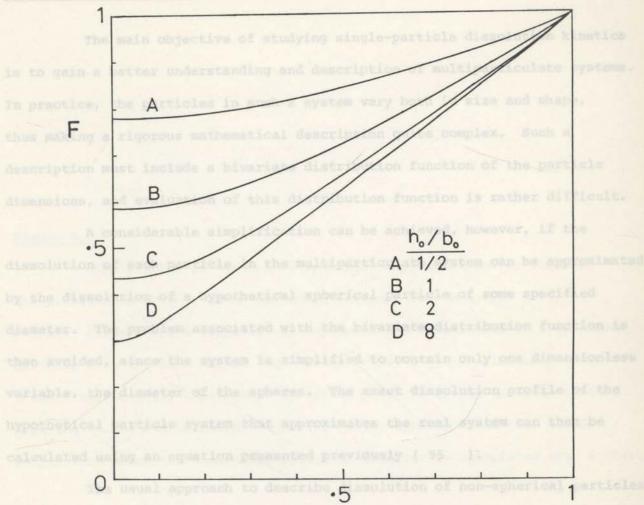
- 72 -

Influence of the shape ratio, F, on the intrinsic dissolution profile of an n-gonal prismatic particle (tetragonal or hexagonal) or a rhombic pyramidal particle. The four curves below and including the diagonal are square root plots. The four curves above and including the diagonal are cube root plots. The dissolution deviates increasingly from the w/w_0 $^{1/3}$ versus t* linear relationship (the cube root law) as the shape ratio becomes less than 1 and approaches a linear $(w/w_0)^{1/2}$ versus t* relationship (the square root law).



Variation of the shape ratio, F, of a rhombic pyramidal particle having various shapes (see Structure IV, Fig. 4.4. for definitions of b_0 , c_0 , and h_0).





has been to approximate them by spherical particles $\underline{b_o}$ ring the same surface area or volume. It is of interest to evaluate the area in such approximations. A spherical particle, having the same surface area as an m-gonal primatic particle with shape ratio P, dissolves according to:

$$(w/w_0)^{3/3} = 1 - \left(\frac{2\pi \cot \frac{\pi}{n}}{\pi (1+2/T)}\right)^{\frac{1}{2}} t^{\frac{\pi}{2}}$$

or if it has the same volume, it dissolves according to

$$(w/w_{\rm q})^{1/3} = 1 - \left(\frac{2\pi r \cot \frac{\pi}{n}}{3\pi}\right)^{1/3} t^{*}$$
 (4)300.3

where c^* and F are defined by 4.94. and 4.95. Figure 4.7. shows the substantian acrors introduced by such approximations based on equal surface area or volume. This is not only the case for F = 1/4 but for all other values of the shape

Spherical Approximations.

The main objective of studying single-particle dissolution kinetics is to gain a better understanding and description of multiparticulate systems. In practice, the particles in such a system vary both in size and shape, thus making a rigorous mathematical description quite complex. Such a description must include a bivariate distribution function of the particle dimensions, and evaluation of this distribution function is rather difficult.

A considerable simplification can be achieved, however, if the dissolution of each particle in the multiparticulate system can be approximated by the dissolution of a hypothetical *spherical* particle of some specified diameter. The problem associated with the bivariate distribution function is then avoided, since the system is simplified to contain only one dimensionless variable, the diameter of the spheres. The exact dissolution profile of the hypothetical particle system that approximates the real system can then be calculated using an equation presented previously (95).

The usual approach to describe dissolution of non-spherical particles has been to approximate them by spherical particles having the same surface area or volume. It is of interest to evaluate the errors in such approximations. A spherical particle, having the same surface area as an n-gonal prismatic particle with shape ratio F, dissolves according to:

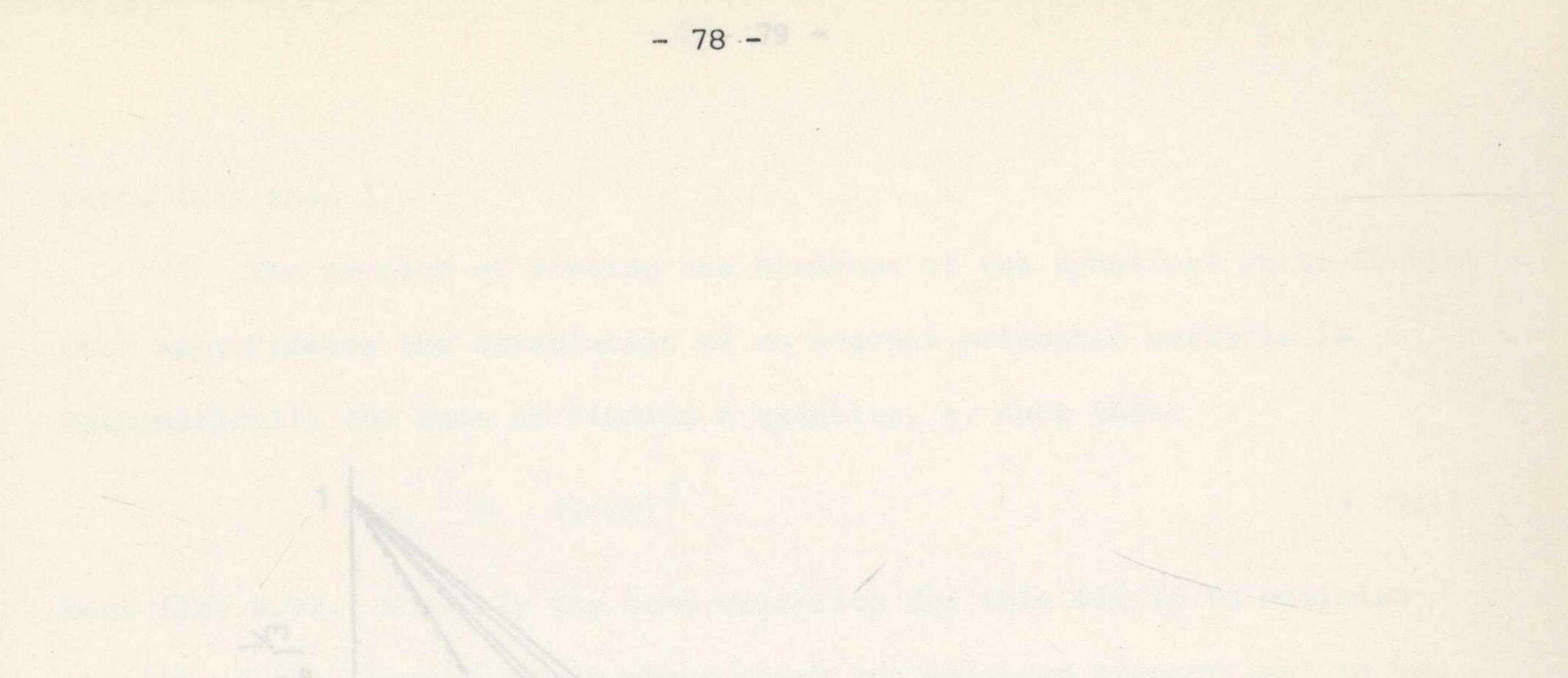
$$(w/w_0)^{1/3} = 1 - \left(\frac{2\pi \cot \frac{\pi}{n}}{n(1+2/F)}\right)^{\frac{1}{2}} t^*$$
 (4.99.)

Or if it has the same volume, it dissolves according to:

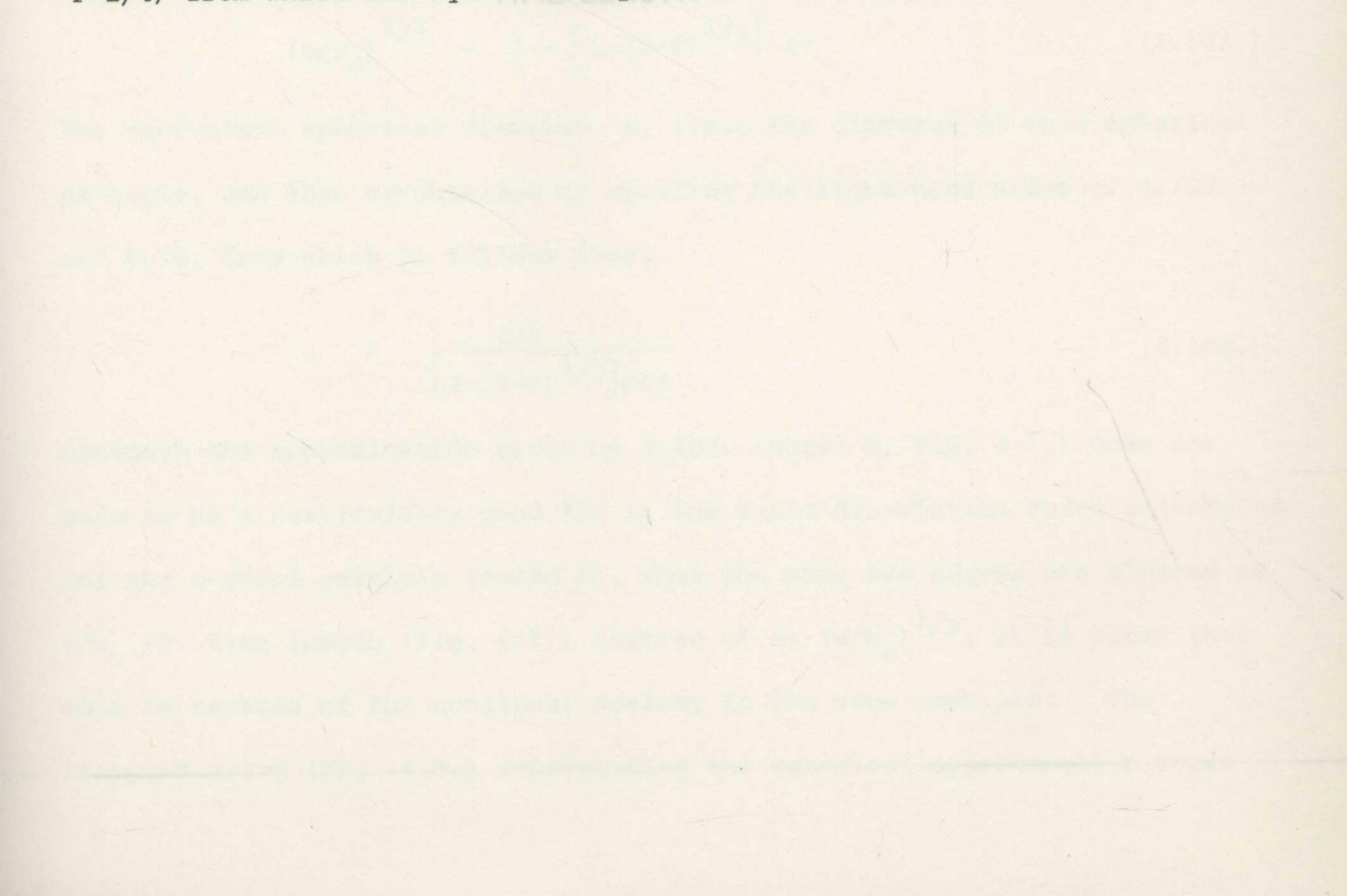
$$(w/w_{o})^{1/3} = 1 - \left(\frac{2\pi F \cot \frac{\pi}{n}}{3n}\right)^{1/3} t^{*}$$
 (4.100.)

where t* and F are defined by 4.94. and 4.95. Figure 4.7. shows the substantial errors introduced by such approximations based on equal surface area or volume. This is not only the case for F = 1/4 but for all other values of the shape

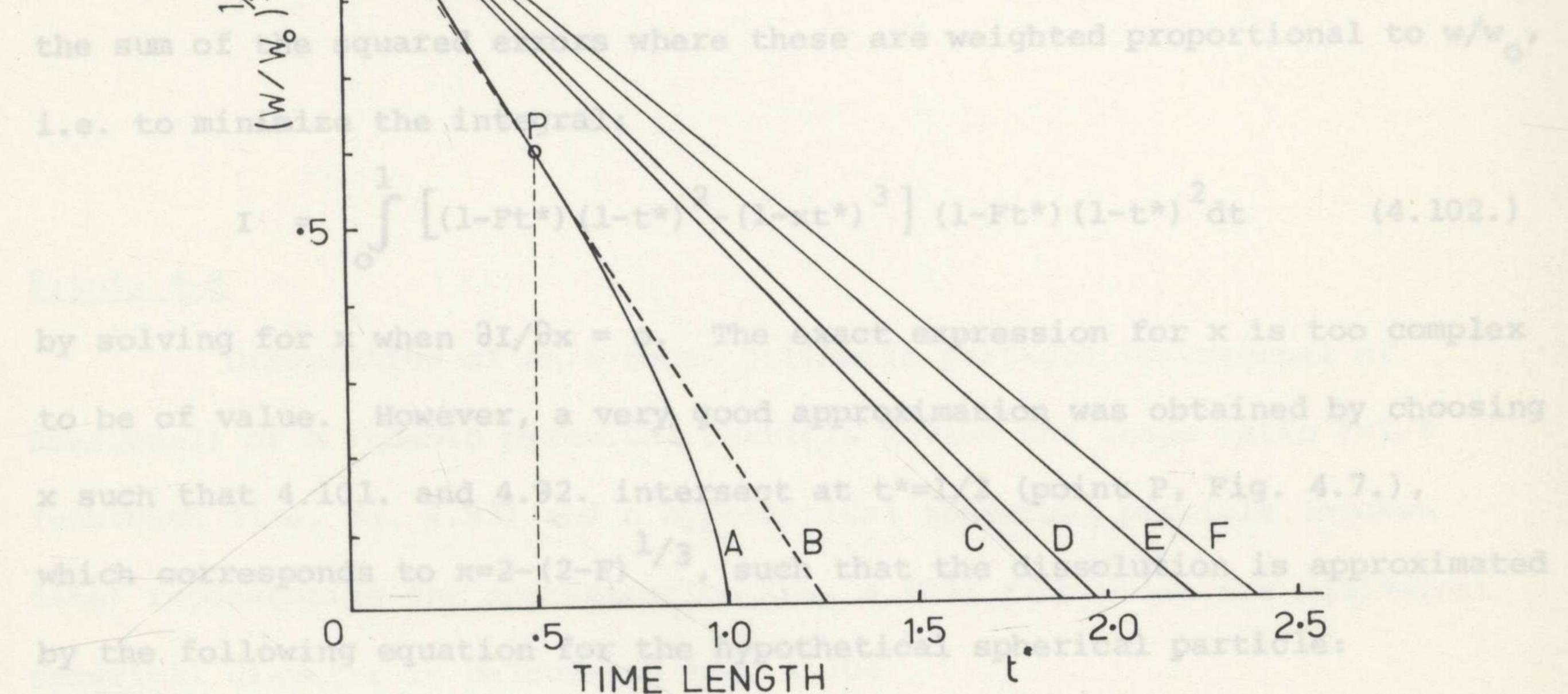
- 77 -

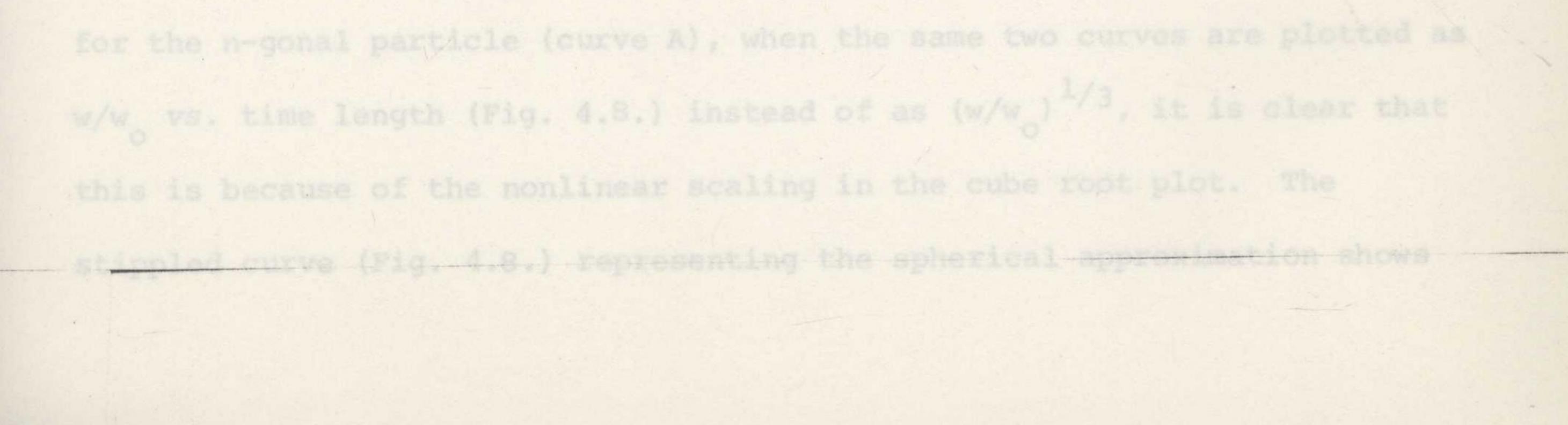


Application of the use of spherical approximations to describe the dissolution of an n-gonal prismatic particle (curve A, Eq. 4.93, F=1/4). Curve C and E respectively represent the dissolution of a spherica particle with the same volume (Eq. 4.100, F=1/4, n=4,6). Similarly curve D and F respectively represent a spherical particle having the same surface area (Eq. 4.99, F=1/4, n=4,6). Curve B is the approximation (Eq. 4.103, F=1/4) from which the equivalent spherical diameter is calculated (Eq. 4.104).



- 79 best approximates the dissolution of an n-gonal prismatic particle is the best criterion for this fit is to minimize m





ratio less than 1.

The problem of finding the diameter of the spherical particle that best approximates the dissolution of an n-gonal prismatic particle is mathematically the same as finding a quantity, x, such that:

$$w/w_{o} = (1-xt)^{3}$$
 (4.101.)

best fits 4.92. Possibly the best criterion for this fit is to minimize the sum of the squared errors where these are weighted proportional to w/w_{o} , i.e. to minimize the integral:

$$I = \int_{0}^{1} \left[(1 - Ft^{*}) (1 - t^{*})^{2} - (1 - xt^{*})^{3} \right] (1 - Ft^{*}) (1 - t^{*})^{2} dt \qquad (4.102.)$$

by solving for x when $\partial I/\partial x = o$. The exact expression for x is too complex to be of value. However, a very good approximation was obtained by choosing x such that 4.101. and 4.92. intersect at t*=1/2 (point P, Fig. 4.7.), which corresponds to x=2-(2-F)^{1/3}, such that the dissolution is approximated by the following equation for the hypothetical spherical particle:

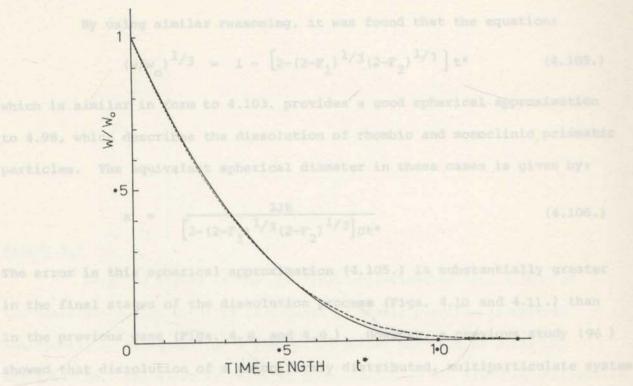
$$(w/w_0)^{1/3} = 1 - [2 - (2 - F)^{1/3}] t^*$$
 (4.103.)

The equivalent spherical diameter, a, i.e., the diameter of this spherical particle, can then be obtained by equaling the right-hand sides of 4.103. and 4.78. from which it follows that:

$$a = \frac{2Jt}{\left[2-(2-F)^{1/3}\right]\rho t^{*}}$$
(4.104.)

Although the approximation given by 4.103. (curve B, Fig. 4.7.) does not seem to be a particularly good fit to the exact dissolution curve calculated for the n-gonal particle (curve A), when the same two curves are plotted as w/w_o vs. time length (Fig. 4.8.) instead of as $(w/w_o)^{1/3}$, it is clear that this is because of the nonlinear scaling in the cube root plot. The stippled curve (Fig. 4.8.) representing the spherical approximation shows

Dissolution of an n-gonal prismatic particle (tetragonal or hexagonal) or a rhombic pyramidal particle having the shape ratio F=1/4 (unbroken line, Eq. 4.93) and a hypothetical spherical particle (broken line) representing the approximation (Eq. 4.103) from which the equivalent spherical diameter is calculated (Eq. 4.104). approximation were calculated for various values of the shape ratio F and showed (Fig. 4.9.) that this choice for the approximation was patisfactory.



is only singhtly effected by the effect of the lower and of the particle-size distribution had very little offect on the dissolution profile calculated (95) Thus, approximation error in the later state of the single-particle dissolution does not introduce the same degree of error when applied to a nominiformly distributed, multiparticulate system. The approximation (4.105.) should, therefore, yield considerably better results when applied to a multiparticulate system than might be judged from Eig. 4.10. This explains the theirs of the particular weighting of the errors in the approximation procedure.

Table 4.1 summarizes the dissolution of the particle forms shown in Structures 1-VI and gives formulas for the calculation of the equivalent spherical disaster in each case. The dissolution of these 10 crystal shopen excellent fit to the true dissolution. The weighted errors of the spherical approximation were calculated for various values of the shape ratio F and showed (Fig. 4.9.) that this choice for the approximation was satisfactory.

By using similar reasoning, it was found that the equation:

$$(w/w_0)^{1/3} = 1 - [2 - (2 - F_1)^{1/3} (2 - F_2)^{1/3}] t*$$
 (4.105.)

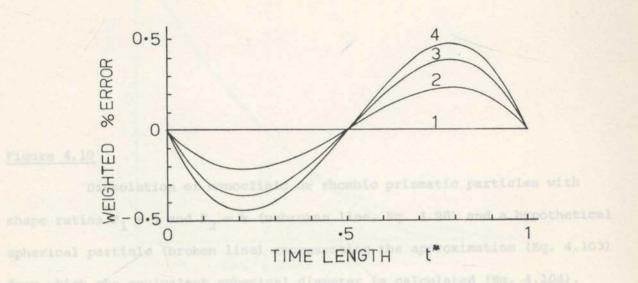
which is similar in form to 4.103. provides a good spherical approximation to 4.98, which describes the dissolution of rhombic and monoclinic prismatic particles. The equivalent spherical diameter in these cases is given by:

$$a = \frac{2Jt}{\left[2 - (2 - F_1)^{1/3} (2 - F_2)^{1/3}\right] \rho t^*}$$
(4.106.)

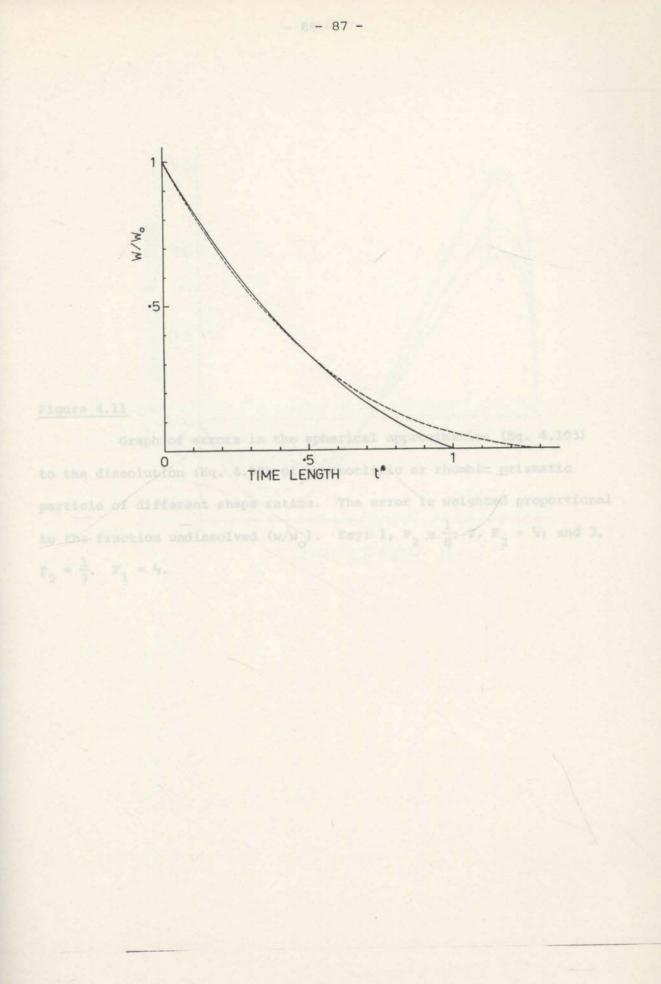
The error in this spherical approximation (4.105.) is substantially greater in the final stages of the dissolution process (Figs. 4.10 and 4.11.) than in the previous case (Figs. 4.8. and 4.9.). However, a previous study (96) showed that dissolution of a nonuniformly distributed, multiparticulate system is only slightly affected by the dissolution behaviour of the smallest particles. Substantial truncation at the lower end of the particle-size distribution had very little effect on the dissolution profile calculated (95). Thus, approximation error in the later stage of the single-particle dissolution does not introduce the same degree of error when applied to a nonuniformly distributed, multiparticulate system. The approximation (4.105.) should, therefore, yield considerably better results when applied to a multiparticulate system than might be judged from Fig. 4.10. This explains the choice of the particular weighting of the errors in the approximation procedure.

Table 4.1 summarizes the dissolution of the particle forms shown in Structures I-VI and gives formulas for the calculation of the equivalent spherical diameter in each case. The dissolution of these 10 crystal shapes can be described by three basic transformed equations of cubic form in time-

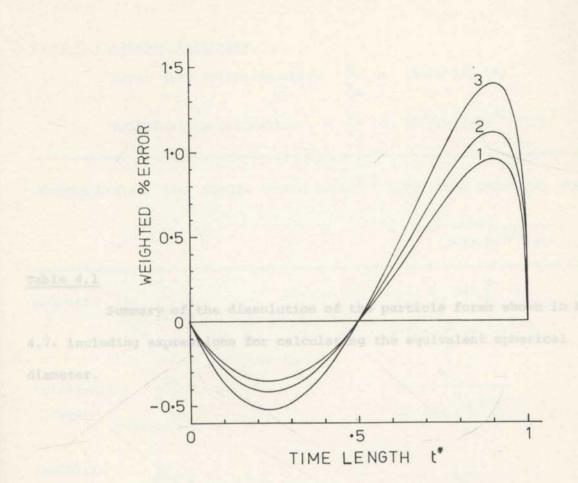
Graph of errors in the spherical approximation (Eq. 4.103) of the dissolution (Eq. 4.93) of an n-gonal prismatic particle (isometric, tetragonal, or hexagonal) or a rhombic pyramidal particle of different shape ratios. The error is weighted proportional to the fraction undissolved (w/w_0) . Key: 1, F = 1;2, F = $\frac{1}{2}$; 3, F = $\frac{1}{3}$, and 4, F = $\frac{1}{4}$.



Dissolution of monoclinic or rhombic prismatic particles with shape ratios $F_1 = \frac{1}{2}$ and $F_2 = \frac{1}{4}$ (unbroken line, Eq. 4.98) and a hypothetical spherical particle (broken line) representing the approximation (Eq. 4.103) from which the equivalent spherical diameter is calculated (Eq. 4.104).



Graph of errors in the spherical approximation (Eq. 4.103) to the dissolution (Eq. 4.98) of a monoclinic or rhombic prismatic particle of different shape ratios. The error is weighted proportional to the fraction undissolved (w/w_o). Key: 1, $F_2 = \frac{1}{8}$; 2, $F_2 = \frac{1}{4}$; and 3, $F_2 = \frac{1}{3}$. $F_1 = \frac{1}{2}$.



- 90 -

n-Conal Prismatic Particles

Table 4.1

Summary of the dissolution of the particle forms shown in Fig. 4.7. including expressions for calculating the equivalent spherical diameter.

Soberical approximation.

CHYPTAL SYNTEM TIME LINGTH CHAPT ANTIOS

DISSOLUTION OF PRISMATIC PARTICLES (a)

n-Gonal Prismatic Particles

Exact dissolution equation	$\frac{W}{W_{O}} =$	(1-Ft*) (1-t*) ²
Spherical approximation	$\frac{W}{W_{O}} =$	$(1-(2-(2-F)^{1/3})t^*)^3$

CRYSTAL SYSTEM TIME LENGTH SHAPE RATIO (b) EQUIVALENT SPHERICAL DIAMETER

$$t^{*} \qquad F \qquad a = \frac{2Jt}{(2-(2-F)^{1/3})\rho t^{*}}$$

$$n-Gonal \quad \left(\frac{2J}{b_{0}\rho} \tan \frac{\pi}{n}\right)t \qquad \frac{b}{0} \cot \frac{\pi}{n} \qquad \qquad \frac{b}{2-(2-\frac{b}{10}\cot \frac{\pi}{n})^{1/3}}{2-(2-\frac{b}{10}\cot \frac{\pi}{n})^{1/3}}$$

$$Tetragonal \qquad \frac{2J}{b_{0}\rho}t \qquad \frac{b}{10} \qquad \qquad \frac{b}{2-(2-\frac{b}{10}\cot \frac{\pi}{n})^{1/3}}$$

$$Isometric \qquad \frac{2J}{b_{0}\rho}t \qquad 1 \qquad \qquad \frac{b}{0} \qquad \qquad \frac{b}{2-(2-\frac{b}{10}\cot \frac{\pi}{n})^{1/3}}{\frac{b}{0}}$$

$$Isometric \qquad \frac{2J}{b_{0}\rho}t \qquad 1 \qquad \qquad \frac{b}{0} \qquad \qquad \frac{\sqrt{3}b_{0}}{2-(2-\frac{\sqrt{3}b_{0}}{1})^{1/3}}$$

$$Hexagonal \qquad \frac{2\sqrt{3}J}{3b_{0}\rho}t \qquad \sqrt{3}\frac{b}{10} \qquad \qquad \frac{\sqrt{3}b_{0}}{2-(2-\sqrt{3}\frac{b}{10})^{1/3}}$$

$$Rhombic and Monoclinic Particles$$

$$Exact dissolution equations \qquad \frac{w}{w_{0}} = (1-F_{1}t^{*})(1-F_{2}t^{*})(1-t^{*})$$

$$spherical approximation \qquad \frac{w}{w_{0}} = (1-(2-(2-F_{1})^{1/3}(2-F_{2})^{1/3})t^{*})^{3}$$

CRYSTAL SYSTEM TIME LENGTH SHAPE RATIOS EQUIVALENT SPHERICAL DIAMETER

t* F₁ F₂ a

$$= \frac{2Jt}{(2-(2-F_1)^{1/3}(2-F_2)^{1/3})\rho t^*}$$

Rhombic
$$\frac{2J}{b_{o}} \qquad \frac{b_{o}}{c_{o}} \qquad \frac{b_{o}}{l_{o}} \qquad \frac{b_{o}}{l_{o}} \qquad \frac{b_{o}}{2-(2-\frac{b_{o}}{c_{o}})^{1/3}(2-\frac{b_{o}}{l_{o}})^{1/3}}$$
Monoclinic
$$\frac{2J}{b_{o}\rho\sin\alpha} t \qquad \frac{b_{o}}{c_{o}} \qquad \frac{b_{o}}{l_{o}}\sin\alpha}{\frac{b_{o}\sin\alpha}{2-(2-\frac{b_{o}}{c_{o}})^{1/3}(2-\frac{b_{o}}{l_{o}})^{1/3}}}$$

- 92 -

DISSOLUTION OF PYRAMIDAL PARTICLES

n-Gonal Pyramidal Particles

(Isometric, Tetragonal, Hexagonal)

Exact dissolution equation $\frac{W}{W_{O}} = (1-t^{*})^{3}$ Time length $t^{*} = \frac{2J\sqrt{h_{O}^{2} + b_{O}^{2}/4}}{\rho}t$

Equivalent spherical diameter^(c)
$$a = \frac{h_{o}b}{\sqrt{h_{o}^{2} + b_{o}^{2}/4}}$$

Rhombic Pyramidal Particle

Exact dissolution equation	$\frac{W}{W_{O}} =$	(1-Ft*) (1-t*) ²
Spherical approximation	$\frac{W}{W_{O}} =$	$(1-(2-(2-F)^{1/3})t*)^3$
Time length	t* =	$\frac{2J\sqrt{h_o^2 + b_o^2/4}}{\rho}t$
Shape ratio	F =	$\frac{1}{4} + \frac{3}{4} \sqrt{\frac{h_o^2 + c_o^2/4}{h_o^2 + b_o^2/4}}$

Equivalent spherical diameter $a = \frac{1}{(2-(2-F)^{1/3})\sqrt{h_0^2 + b_0^2/4}}$

- (a) Figure 1 defines the quantities b_0 , c_0 , 1_0 and α used.
- (b) When F=1, i.e. $\frac{b_0}{l_0} = \tan \frac{\pi}{n}$ then the equivalent spherical diameter is equal to the biggest sphere that can be contained in the prismatic body. The spherical approximation of the dissolution will then become exact.
- (c) This diameter is equal to the biggest sphere the pyramid can contain.

(stippled line, Fig. 4,8. and 4.10) is above the calculated dissolution curve in the later stages. The true dissolution curve, because of the rounds offert, is above the calculated curve and hence closer to the approximation. Excellent egreement between experimental and calculated results was obtained for the dissolution of a multiparticulate system of perticles, approximately tetragonal primetic in shape, when the respective spherical approximations were applied (120). length or time. Dissolution can thus be described exactly by a third degree polynamial in time in all cases.

- 94 -

The basic assumptions behind these derivations is that the rate of dissolution per unit surface area, J, remains constant during dissolution and is the same everywhere at the interface of the dissolving crystal. This assumption can only be approximately true in practice under complete sink conditions. The higher activity at the crystal edges results in a larger J value in these areas and, therefore, a "rounding off" of the shape, so that during the later stages dissolution is slower than calculated. However, this should result in an improvement in the fit of the spherical approximation and could result in a closer fit to the real dissolution than the exact expression given for isotropic conditions. Thus, the approximating curve (stippled line, Fig. 4.8. and 4.10) is above the calculated dissolution curve in the later stages. The *true* dissolution curve, because of the rounding effect, is above the calculated curve and hence closer to the approximation.

Excellent agreement between experimental and calculated results was obtained for the dissolution of a multiparticulate system of particles, approximately tetragonal prismatic in shape, when the respective spherical approximations were applied (120).

- 0 0 0 -

CHAPTER 5

MULTIPARTICULATE DISSOLUTION

The dissolution of a multiparticulate system is considerably more complex to characterize than a single particle system. There are two main reasons for this. One reason is that it may be difficult to account for the particle size distribution effect (see Chapter 6) which is present when the powder is not monodisperse. Another reason is that there may be significant interaction between the dissolving particles; the dissolution of a particle in the powder will, in general, influence the dissolution of other particles and often to varying degrees. In evaluating the intrinsic dissolution properties of a powder it is therefore necessary to establish experimental conditions which minimize such interactions. This is made possible by the "single layer-flow through" principle employed in the dissolution cell used in these experiments (see Chapter 7 and 10).

Under such sink conditions it is possible to establish a multiparticulate dissolution model based on a single particle dissolution model and the initial particle size distribution. Such a model should describe the intrinsic dissolution properties of a powder.

Monodisperse powders.

If the initial particle size distribution can be considered to be infinitely narrow, i.e. if all the particles are of the same initial size, r_o, then the multiparticulate model becomes identical to the simple particle model when considered on a "fraction undissolved *versus* time basis".

For example if the *single* particle dissolution follows the cube root model:

$$w^{1/3} = w^{1/3} - kt$$
 (5.1.)

Then the multiparticulate model becomes:

$$W^{1/3} = W_0^{1/3} - n^{1/3} kt$$
 (5.2.)

(5.3.)

where $n = W_0/W_0$

is the number of particles in the powder. Eqs. 5.1. and 5.2. are thus identical in the sense that:

$$W/W_{o} = W/W_{o} = (1 - \frac{k}{W_{o}} t)^{3}$$
 (5.4.)

Polydisperse powders.

-94

Several investigators (107,92) have considered the problem of exactly describing the dissolution profile of powders in relation to their particle size distribution. These authors have been concerned with powders initially consisting of particles with log-normal size distribution. Earlier attempts made use of approximations (107) or computer simulations (92). More recently, Brooke (93,94) developed an equation that permits calculation of the dissolution profile of such powders without the aid of a computer. This equation was later presented in a form to account for trimcated log-normal distributions.

A general equation that exactly describes the entire dissolution profile of powders under sink conditions is derived in the following section. The equation is valid for particles having any initial size distribution and dissolving according to any explicit single particle model. The equation is then applied to obtain an exact expression for a log-normal powder considering the cube root model. Consider a powder consisting of particles which initially (t=o) have a weight density (probability distribution) $f_0(w_0)$. Let the particles dissolve independently of each other according to:

$$w = g(w_{0}, t, A)$$
 (5.5.)

where w and w_0 are the particle weights at time t and t=0, respectively and A collectively represents dissolution parameters such as solubility, particle density and particle shape factors. The inverse dissolution function is defined as:

$$w_{o} = g^{-1} (w, t, A)$$
 (5.6.)

Using the rules of transformation of independent variables (121) the particle weight density function at time t becomes:

$$f_{0}(w) = f_{0}\left[g^{-1}(w,t,A)\right] \frac{d}{dw} g^{-1}(w,t,A)$$
(5.7.)

For 5.7. to hold, the following conditions for g must be satisfied: (a) g^{-1} must be a strictly increasing function of w for all t values, (b) g must decrease strictly with time until equal to zero, (c) g must remain equal to zero beyond that time. The latter two conditions ensure that the dissolution function reflects the actual physical conditions of the dissolution process. The first condition will rarely be violated because in application, g is nearly always a strictly increasing function of w_o for all t values. It is obvious that the second condition must be met by any dissolution equation.

The third condition is not satisfied for most equations in the 107 literature (97,106,/). To overcome this problem, it is necessary to redefine the particle weight density function such that it is generally applicable:

giving the bast fit when t

 $f(w) = \frac{h(w)}{\int_{0}^{\infty} h(w) dw} \quad \text{for } w > o \quad (5.8.)$

$$f(w) = o \qquad \text{for } w \leq o \qquad (5.9.)$$

where division by the integral is necessary to satisfy the condition that the total integral (from - ∞ to + ∞) of f(w) must be equal to 1.

- 98 -

The weight of undissolved powder, W, at any time, t, is equal to the product of the number of particles remaining, N_t , and the mean particle weight, which for a large number of particles is the same as the expected value of w, $E_t(w)$. Therefore, the following general equation can be written: $W = N_{\pm}E_{\pm}(w)$ (5.10.)

The number of particles remaining at time t is:

$$N_{t} = N_{o} \int_{O}^{\infty} h(w) dw$$
 (5.11.)

where the initial numbers of particles N_{O} is equal to the initial powder weight, W_{O} , divided by the initial mean particle weight:

$$N_{o} = \frac{W_{o}}{\int_{0}^{\infty} W_{o} f_{o}(W_{o}) dW_{o}}$$
(5.12.)

The mean particle weight as time t is given by:

$$E_{t}(w) = \int_{-\infty}^{\infty} wf(w) dw \qquad (5.13.)$$

which according to 5.8. can be written:

$$E_{t}(w) = \frac{\int_{0}^{\infty} wh(w) dw}{\int_{0}^{\infty} h(w) dw}$$
(5.14.)

substituting Eqs. 5.11., 5.12. and 5.14. into Eq. 5.10. yields:

$$\frac{W}{W_{o}} = \frac{\int_{0}^{\infty} wh(w) dw}{\int_{0}^{\infty} w_{o} f_{o}(w_{o}) dw_{o}}$$
(5.15.)

This equation relates to unbounded particle weight distributions. In practice, the distribution is always bounded, so the limits of the integration must be changed accordingly.

Let m and M denote the initial weights of the smallest and largest particle, respectively.¹ These values then represent the lower and

^{1.} These values are not intended to be absolute but rather represent limits giving the best fit when the actual particle distribution is approximated by any particular function. Therefore, they also represent truncation limits of the function.

upper boundaries, respectively, of $f_{O}(w_{O})$; f(w) will be correspondingly bounded by $Pg(m_{O},t,A)$ and $Pg(M_{O},t,A)$. The operator P is introduced to ensure that the limits are never negative. Therefore, by definition, P is equal to 1 in the time period before the operand becomes zero and is equal to zero beyond that time. When these limits are introduced into 5.15. and h(w) is written according to 5.7., the following expression is obtained:

$$\frac{W}{W_{o}} = \frac{\int_{w=Pg(M_{o},t,A)}^{w=Pg(M_{o},t,A)} wf_{o} [g^{-1}(w,t,A)] \frac{d}{dw} g^{-1}(w,t,A) dw}{\int_{w=Pg(M_{o},t,A)}^{M} w_{o} f_{o}(w_{o}) dw_{o}}_{m_{o}}$$
(5.16.)

If $g^{-1}(w,t,A)$ and w_0 in the integrals are considered dummy variables and called x, 5.16. simplifies to:

$$\frac{W}{W_{o}} = \frac{\int_{L_{1}}^{L_{2}} g(x,t,A) f_{o}(x) dx}{\int_{m_{o}}^{M} x f_{o}(x) dx}$$
(5.17.)

Because of this transformation and the properties of the operator P, the limits of integration, L_1 and L_2 have the following values:

$$L_{1} = m_{o} \qquad \text{for t such that } Pg(m_{o}, t, A) > 0 \qquad (5.18.)$$
$$L_{1} = g^{-1}(o, t, A) \qquad " " " " " = 0 \qquad (5.19.)$$
$$L_{2} = M_{o} \qquad \text{for t such that } Pg(M_{o}, t, A) > 0 \qquad (5.20.)$$

$$L_2 = g^{-1}(o,t,A)$$
 """ = 0 (5.21.)

The time at which $g(m_0, t, A) = 0$ is the critical time, that is, the time when the dissolving particles begin to disappear. When $g(M_0, t, A) = 0$ all particles are dissolved. Beyond that time, $L_1 = L_2$, thus making W/W equal to zero. Equation 5.17. requires knowledge of the initial weight distributing $f_o(w_o)$, but more often the initial size distribution is of greater interest. Consider particles that are spherical and remain so during dissolution; in this case $w = \rho \pi a^3/6$, where ρ and a are the particle density and diameter, respectively.² Transformation in 5.16. from the initial weight distribution to the initial size distribution is then easily achieved; and by a similar procedure to that used to obtain 5.17. from 5.16., the following equation is

derived:

$$\frac{W}{W_{o}} = \frac{\int_{R_{1}}^{R_{2}} g(\frac{\rho\pi}{6} x^{3}, t, A) l_{o}(x) dx}{\int_{d}^{D} \frac{\pi\rho}{6} x^{3} l_{o}(x) dx}$$
(5.22.

where:

$$R_{1} = d_{o} \quad \text{for t such that } Pg(\frac{\rho\pi}{6}d_{o}^{3}, t, A) > 0 \quad (5.23.)$$

$$R_{1} = \left[\frac{6}{\rho\pi}g^{-1}(o, t, A)\right]^{1/3} \quad " " " " = 0 \quad (5.24.)$$

$$R_{1} = D_{o} \text{ for t such that } Pg(\frac{\rho\pi}{6}D_{o}^{3}, t, A) > 0 \quad (5.25.)$$

$$R_{1} = \left[\frac{6}{\rho\pi}g^{-1}(o, t, A)\right]^{1/3} \quad " " " " = 0 \quad (5.26.)$$

In Eq. 5.22, d_0 and D_0 are initial diameters of the smallest and largest particles, respectively, 1_0 is the initial particle-size density function.

Eqs. 5.17. and 5.22. rigorously describe the entire dissolution profile of any powder if its initial particle weight density function, f_o , or initial particle-size density function, l_o , is known, together with the particle dissolution function, which can be any explicit expression. With appropriate choice of limits of integration, the equations are applicable to truncated as well as "ideal" distributions. The time-dependent integral

^{2.} The derivation is valid for particles of other shapes as long as they remain unchanged during dissolution and an appropriate shape factor is used in place of $\pi/6$.

in the numerator of Eqs. 5.17. and 5.22. reduces to the constant integral in the denominator of zero time, thus making the ratio W/W_{o} equal to 1 as expected.

The general mathematical models expressed by 5.17. and 5.22. require the use of a computer for numerical evaluation because they are in integral form. However, if the initial particle weight or particle-size distribution can be approximated by some simple function, then the model can often be solved in terms of an expression suitable for evaluating without the use of a computer.

Special Case: Log-Normal Powders and the Cube Root Model.

Carstensen and Musa (92) have pointed out in their review of the literature that procedures such as milling, grinding and precipitation, which are based on random processes, produce particles having skewed distribution functions which often can be approximated by a log-normal distribution. It is therefore of interest to examine this special case.

Consider a powder consisting of spherical particles of initial diameters a_0 , distributed such that $\ln a_0$ approximates a normal distribution with mean μ and standard deviation σ , truncated at $\ln d_0 = \mu - i\sigma$ and $\ln D_0 = \mu + j\sigma$, (Fig. 5.1.) where i and j are trimcation parameters. The density function of $\ln a_0$ is then given by:

$$u(\ln a_{o}) = \frac{N(\ln a_{o}, \mu, \sigma)}{\ln a_{o} = \mu + j\sigma}$$
(5.27.)
$$\int_{\ln a_{o}}^{0} = \mu - i\sigma$$
$$\mu - i\sigma \leq \ln a_{o} \leq \mu + j\sigma$$

where the normal density function N is defined as:

$$N(x,\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} EXP\left[-(x-\mu)^2/2\sigma^2\right]$$
(5.28.)

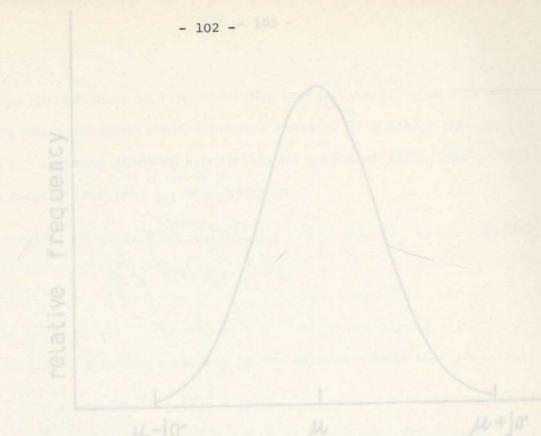
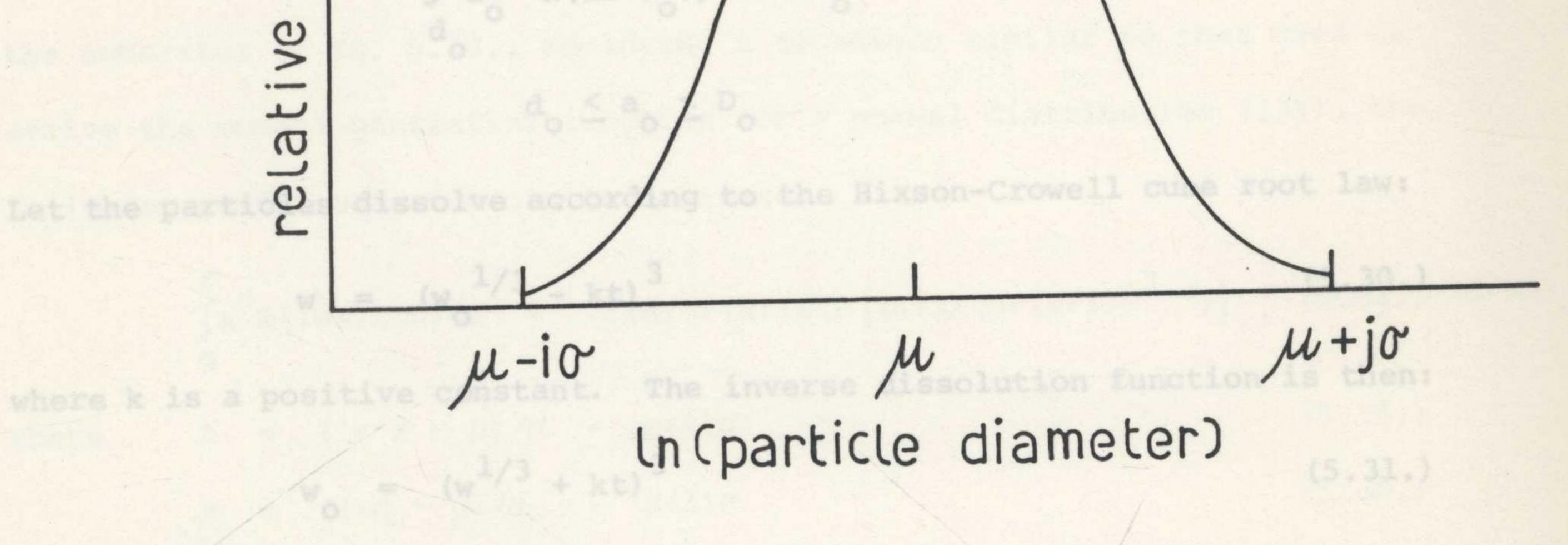


Figure 5.1. Illustration of the parameters in a truncated log-normal distribution.

- 103 -Because the distribution is truncated, the it tegral of u (lna) must be c ion technique (121), the initial equal to 1. BC on unction, 1, then become req L



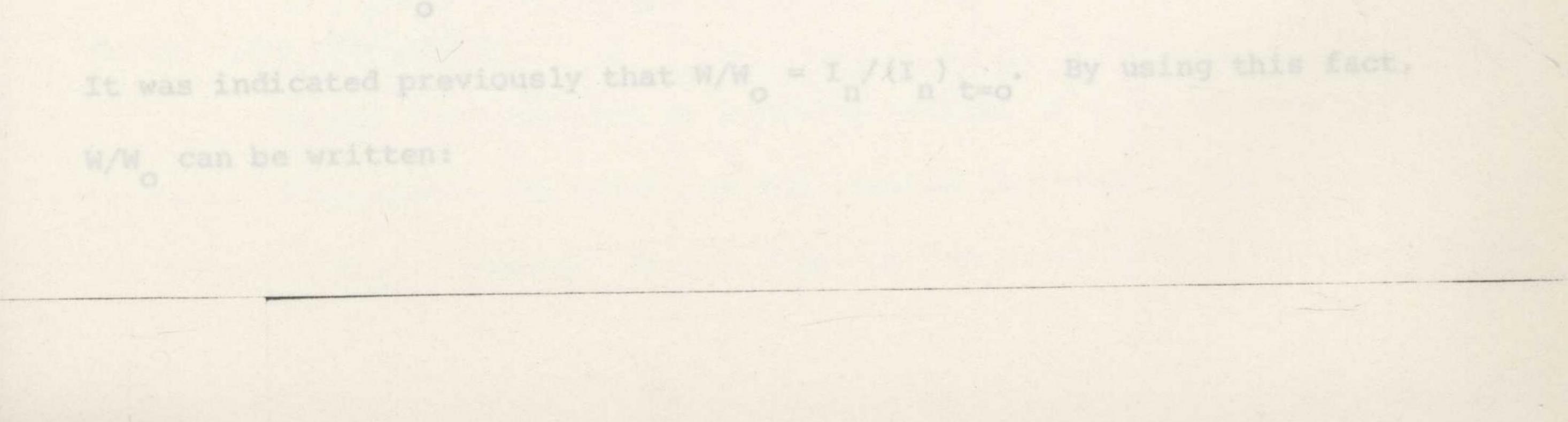
relationship giving the dissolution profile can then be derived by means or

After the initial particle-size distribution (5.29.) and the particle

5.22. in the following way

After inserting Eqs. 5.29.-5.31. into Eq. 5.22., the integral in

the numerator, I of 5.22. becomes:



Because the distribution is truncated, the integral in 5.27. is introduced to satisfy the requirement that the total integral of u (lna) must be equal to 1. By using standard transformation technique (121), the initial particle density function, 1, then becomes: $l_{0}(a_{0}) = \frac{a_{0}^{-1}N(\ln a_{0},\mu,\sigma)}{a_{0}}$ (5.29.) $\int_{a}^{o} -1 N(\ln a, \mu, \sigma) d a$

0 0 the numerator of Eq. 5.03., employing a technique similar to that used to derive the moment-generation $d_0 \leq a_0 \leq D_0$ a normal distribution (121), the

Let the particles dissolve according to the Hixson-Crowell cube root law:

$$w = (w_0^{1/3} - kt)^3$$
 (5.30.)

where k is a positive constant. The inverse dissolution function is then: where A = (ln 2 - j)/0 - (s+1)0

$$w = (w^{1/3} + kt)^3$$
 (5.31.)

After the initial particle-size distribution (5.29.) and the particle dissolution equation (5.30. and its inverse 5.31.) are defined, the

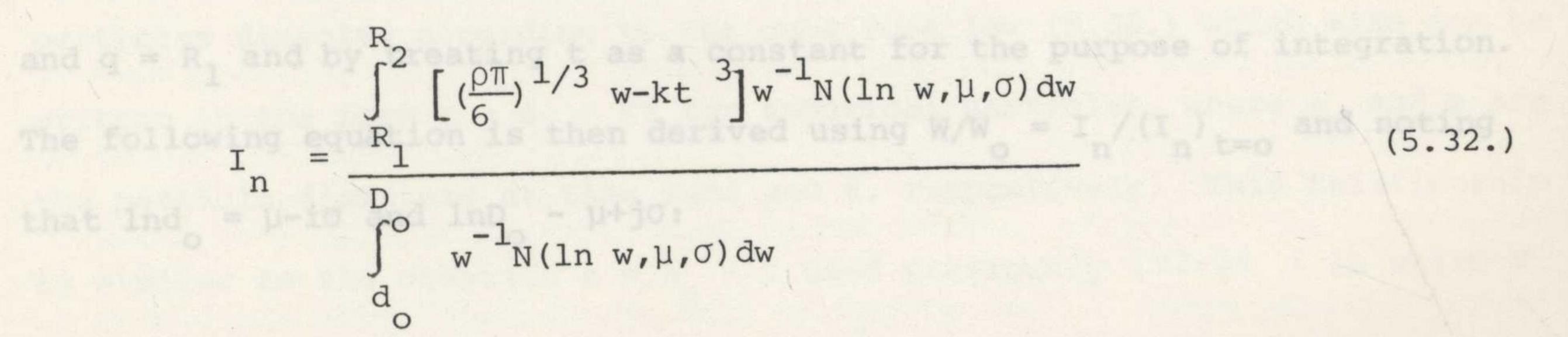
relationship giving the dissolution profile can then be derived by means of

as w -3(Kt) 33(Kt) w [letting K = (6/pn) K for simplicity]. The above

5.22. in the following way:

Thus, to evaluate the numerator, I, in 5.33., the term (w-Kt) w is expanded After inserting Eqs. 5.29.-5.31. into Eq. 5.22., the integral in

the numerator, I_n of 5.22. becomes: formula is then applied term by term by putting s = 2,1,0, and -1; r = R.



It was indicated previously that $W/W_0 = I_n/(I_n)_{t=0}$. By using this fact,

W/W can be written:

$$\frac{W}{W_{o}} = \frac{\prod_{R_{1}}^{R_{2}} \left[w - \left(\frac{6}{\rho \pi}\right)^{1/3} kt \right]^{3} w^{-1} N (\ln w, \mu, \sigma) dw}{\prod_{d_{o}}^{D} w^{2} N (\ln w, \mu, \sigma) dw}$$
(5.33)

In 5.33., the time-independent integral in the denominator is equal to the time-dependent integral in the numerator evaluated at zero time. To evaluate the numerator of Eq. 5.33., employing a technique similar to that used to derive the moment-generating function for a normal distribution (121), the following useful equation can be obtained:

$$\int_{q}^{r} S_{N}(\ln x, \mu, \sigma) dx = (F(A) - F(B)) EXP[(s+1)(\mu + (s+1)\sigma^{2/2})]$$
(5.34.)

where

$$A = (\ln r - \mu)/\sigma - (s+1)\sigma$$
 (5.35.)

$$B = (\ln q - \mu)/\sigma - (s+1)\sigma$$
 (5.36.)

and the function F() is the area under the standard normal curve function given by:

$$F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-x^2/2} dx$$
 (5.37.)

Thus, to evaluate the numerator, I, in 5.33., the term $(w-Kt)^{3}w^{-1}$ is expanded as $w^{2}-3(Kt)^{2}+3(Kt)^{3}w^{-1}$ (letting $K = (6/\rho\pi)^{1/3}k$ for simplicity). The above formula is then applied term by term by putting s = 2,1,0, and -1; $r = R_{2}$ and $q = R_{1}$ and by treating t as a constant for the purpose of integration. The following equation is then derived using $W/W_{0} = I_{n}/(I_{n})_{t=0}$ and noting that $\ln d_{0} = \mu - i\sigma$ and $\ln D_{0} - \mu + j\sigma$:

$$W/W_{o} = \left[\frac{F\left(\frac{2}{\sigma}^{-\mu} - 3\sigma\right) - F\left(\frac{1}{\sigma}^{-\mu} - 3\sigma\right)}{F\left(j-3\sigma\right) - F\left(-1-3\sigma\right)} \right]$$

-3 (Kt)
$$\left[\frac{F\left(\frac{2}{\sigma}^{-\mu} - 2\sigma\right) - F\left(\frac{1}{\sigma}^{-\mu} - 2\sigma\right)}{F\left(j-3\sigma\right) - F\left(-1-3\sigma\right)} \right] \exp\left[-\mu-5\sigma^{2}/2\right] (5.38.)$$

$$-3 (\text{Kt})^2 \left[\frac{F(\frac{2}{\sigma} - \sigma) - F(\frac{1}{\sigma} - \sigma)}{F(j-3\sigma) - F(-1-3\sigma)} \right] \exp \left[-2\mu - 4\sigma^2 \right]$$

$$- (Kt)^{3} \left[\frac{F(\frac{T_{2}-\mu}{\sigma}) - F(\frac{T_{1}-\mu}{\sigma})}{F(j-3\sigma) - F(-i-3\sigma)} \right] EXP\left[-3\mu - 9\sigma^{2}/2 \right]$$

where:	т1	=	μ - iσ f	for ln Kt $\leq \mu$ - io	(5.39.)
	Т			for ln Kt > μ - io	(5.40.)
particle dimo	т2	=	μ + jσ i	For ln Kt $\leq \mu$ + j σ	(5.41.)
	т2		ln Kt i	for ln Kt > μ + jσ	(5.42.)
and			(6/ρπ) ^{1,}		(5.43.)

The change in T, at timelength Kt = EXP $(\mu-i\sigma)$ (critical time) coresponds to the time when the smallest particles, initially having a diameter $d_o = EXP$ $(\mu-i\sigma)$, begin to disappear. The change at Kt = EXP $(\mu+j\sigma) = D_o$ signifies the end of the dissolution process, so W/W_o becomes zero after that timelength.

Equation 5.38. describes the complete dissolution profile of lognormal powders and any sieve fraction of such powders. It assumes that the particles dissolve according to the cube root law (5.30.) which also can be written in the form $a = a_0 - Kt$ for spherical particles, where a_0 and a are the particle diameters at time zero and t, respectively. This relationship is similar to the equation $a = a_0 - \tau$ used previously (92-94) in which τ denotes the timelength.

Brooke (94) derived an equation similar to 5.38 for log-normal powders. Directions were given for the changes required in the equation at the critical time. However, in his equation the first term is constant, equal to 1. Therefore, his equation is incorrect if applied to dissolution after the critical time. The error so introduced becomes quite substantial for large timelengths and large values of σ .

Brooke calculated values of W/W_o for various values of $\tau/EXP(\mu)$ (the latter terminology corresponds to $EXP(-\mu)Kt$ used here). This procedure represents an ingenious method of "scaling time" (by the factor $EXP(-\mu)K$) so the dissolution profile becomes independent of the parameters μ and k (Eq. 5.30.) and only depends on σ , enabling the effect of σ alone to be assessed. In the present work, however, cube root type plots of $(W/W_o)^{1/3}$ versus $EXP(-\mu)Kt$ are used for better comparison with the fundamental particle dissolution equation (5.30) which obeys the cube root law.

Figure 5.2. shows such a plot for powders truncated at $\mu \pm 2\sigma$ (i=j=2) and having σ =0.3. The curvature of the unbroken line, calculated according to Eq. 5.38. is logically expected. The stippled line represents the dissolution profile calculated according to Brooke's equation (Eq. 4 of Ref. 94). The two profiles are, as expected, identical until critical time (EXP(- μ)Kt=0.5488), but the later part clearly demonstrates the limitation of his equation.

Figure 5.3. demonstrates the effect of σ on the dissolution profile for a powder initially having an ideal distribution (i=j=10). Powders of uniform particle size, i.e., $\sigma=0$, give linear cube root plots as expected, while the deviation from linearity is significant for larger σ values. An increase in σ results in a decrease in the initial slopes of the curves, which is consistent with calculations made by Brooke (94). Among powders having the same logarithmic mean diameter, μ , those with broadest distribution will have the slowest initial release rate.

- 107 -

A Cyber 76 digital computer equipped with Calcomp platter was used for calculations and plots. Numerical evaluations were tested to six digits.

Figure 5.2.

Plot of the cube root of fraction undissolved versus $EXP(-\mu)Kt$ for a hypothetical log-normal powder, with $\sigma=0.3$ and truncation at $\mu\pm2$ (i=j=2). The unbroken curve is calculated according to Eq. 5.38, and the stippled curve is calculated according to an equation given by Brooke.

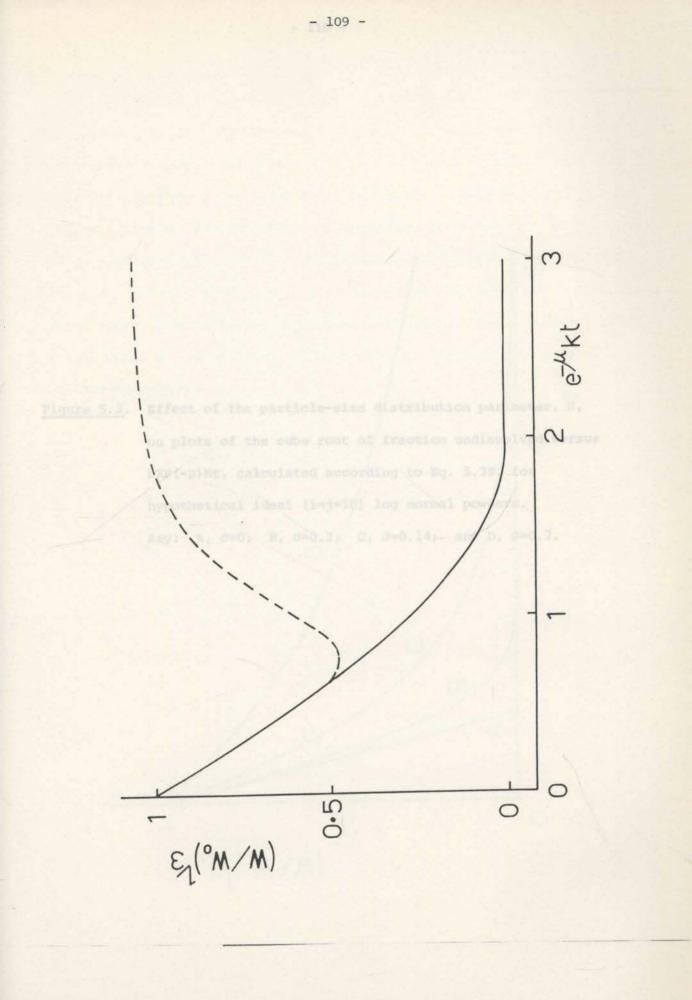
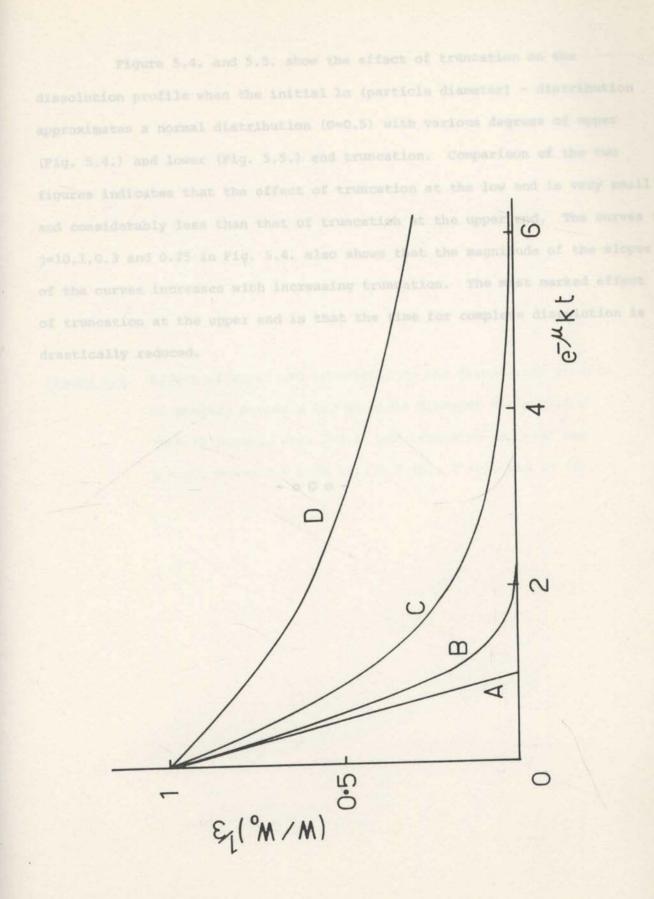


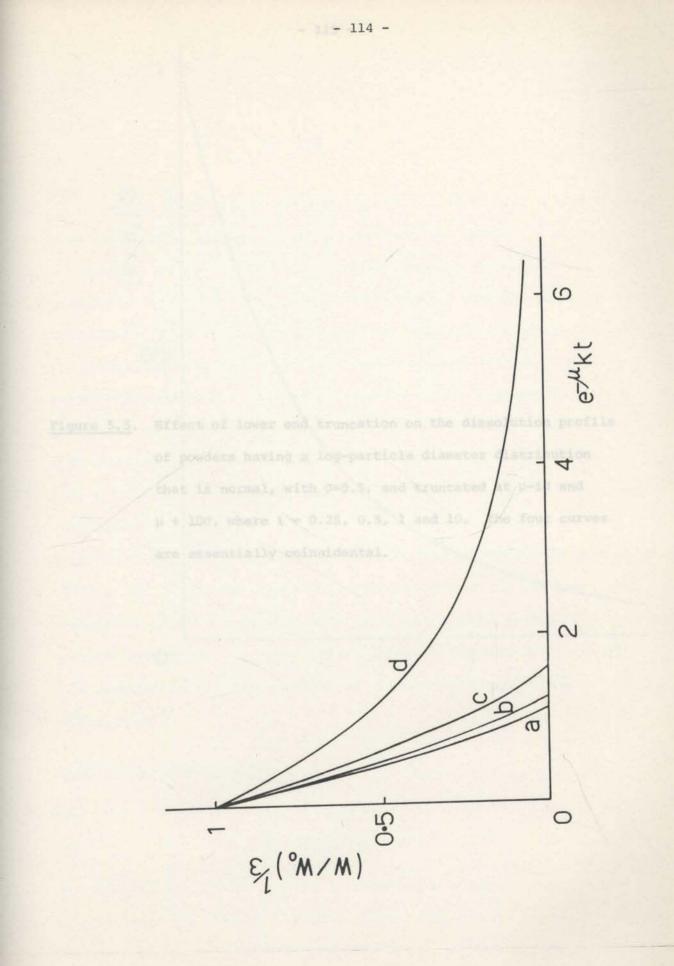
Figure 5.3. Effect of the particle-size distribution parameter, σ , on plots of the cube root of fraction undissolved versus EXP($-\mu$)Kt, calculated according to Eq. 5.38. for hypothetical ideal (i=j=10) log normal powders. Key: A, $\sigma=0$; B, $\sigma=0.2$; C, $\sigma=0.14$; and D, $\sigma=0.7$.

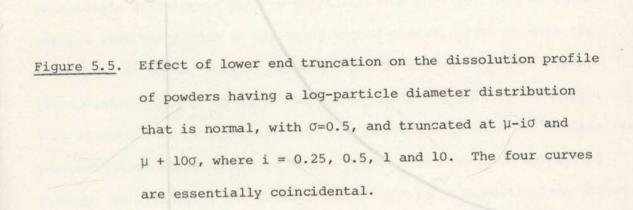


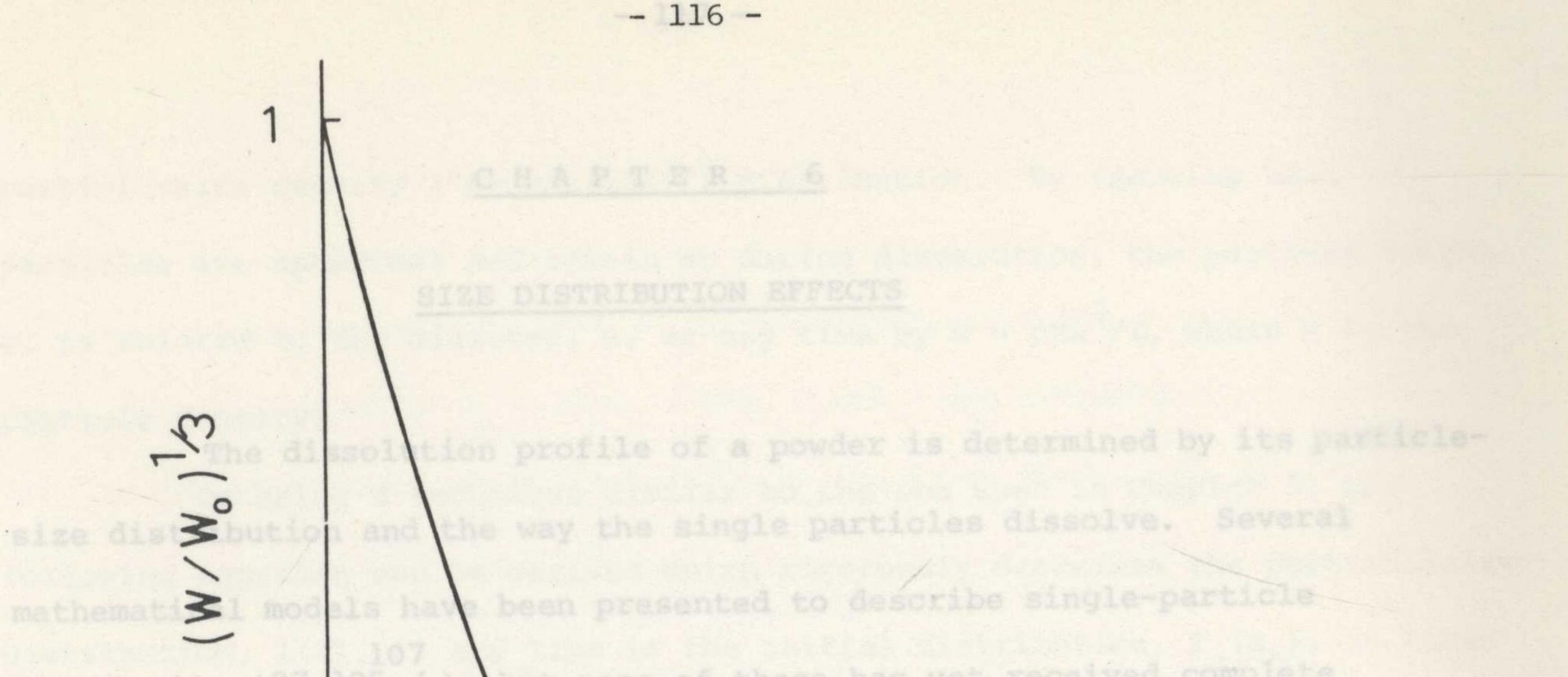
- 111 -

Figure 5.4. and 5.5. show the effect of truncation on the dissolution profile when the initial ln (particle diameter) - distribution approximates a normal distribution (σ =0.5) with various degrees of upper (Fig. 5.4.) and lower (Fig. 5.5.) end truncation. Comparison of the two figures indicates that the effect of truncation at the low end is very small and considerably less than that of truncation at the upper end. The curves for j=10,1,0.3 and 0.25 in Fig. 5.4. also shows that the magnitude of the slopes of the curves increases with increasing truncation. The most marked effect of truncation at the upper end is that the time for complete dissolution is drastically reduced.

Plante 3.4. Effect of upper and truncation on the dissolution profile of powders having a log-particle dismeter distribution what is normal, with 0=0.3, and transsed at u-100 and $\mu + j_0$, where j = 0.23 (a). 0.5 (b). 1 (c) and 10 (d). Figure 5.4. Effect of upper end truncation on the dissolution profile of powders having a log-particle diameter distribution that is normal, with σ =0.5, and truncated at μ -10 σ and μ + j σ , where j = 0.25 (a), 0.5 (b), 1 (c), and 10 (d).

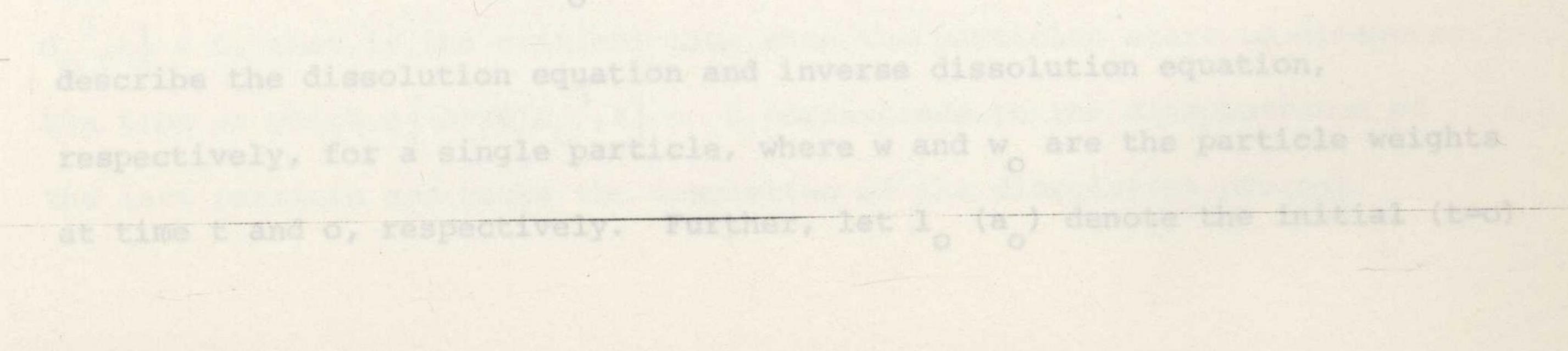






dissolution (97, 105-7) but none of these has yet received complete acceptance. Experimental valuation of the models on the basis of multiparticulate 0.5 lution data is complicated by the distribution effect present when the powder is no truly monodispersed. Such powders are impossible to obtain in most car (122). Processes such as sieving, precipitation, a d grinding do not visit completely uniform particular. This situation particularly true to time powders which are of greatest pharmaceutical importance. In recent your there has been increasing interest in evaluating the distribution effect in Utiparticulate dissolution

(92-96, 107). In this chapter distribution effects for 1 - systems are considered together with three single-particle dissolution models from the lit Osture. By using siglated dissolutio4 data and partic 6 size distributions, the possibility of distinguishing be e-"Kte models is investigated.



CHAPTER 6

SIZE DISTRIBUTION EFFECTS

The dissolution profile of a powder is determined by its particlesize distribution and the way the single particles dissolve. Several mathematical models have been presented to describe single-particle 107 dissolution (97,105-/), but none of these has yet received complete acceptance. Experimental evaluation of the models on the basis of multiparticulate dissolution data is complicated by the distribution effect present when the powder is not truly monodispersed. Such powders are impossible to obtain in most cases (122). Processes such as sieving, precipitation, and grinding do not yield completely uniform particles. This situation is particularly true for fine powders which are of greatest pharmaceutical importance. In recent years, there has been increasing interest in evaluating the distribution effect in multiparticulate dissolution (92-96, 107). In this chapter distribution effects for log-normal powder systems are considered together with three single-particle dissolution models from the literature. By using simulated dissolution data and particlesize distributions, the possibility of distinguishing between the models is investigated.

Theoretical Considerations

Let:

and

$$w = g(w_0, t)$$
 (6.1)
 $w_0 = g^{-1}(w, t)$ (6.2)

describe the dissolution equation and inverse dissolution equation, respectively, for a single particle, where w and w_o are the particle weights at time t and o, respectively. Further, let l_o (a_o) denote the initial (t=o) particle-size density ("probability") distribution. By assuming that particles are spherical and remain so during dissolution, the particle weight, w, is related to the diameter, a, at any time by $w = \rho \pi a^3/6$, where ρ is the particle density.

By using a technique similar to the one used in Chapter 5, the following equation can be derived which rigorously describes the particle-size distribution, 1(a), at any time if the initial distribution, $1_0(a_0)$, is known together with the particle dissolution function, g:

$$l(a) = \frac{1_{o} \left(\left[\frac{6}{\pi \rho} g^{-1} \left(\frac{\pi \rho}{6} a, {}^{3} t \right) \right]^{1/3} \right) \frac{d}{da} \left[\frac{6}{\pi \rho} g^{-1} \left(\frac{\pi \rho}{6} a^{3}, t \right) \right]^{1/3}}{\int_{L_{1}}^{L_{2}} 1_{o}(a) da}$$
(6.3)

$$P\left[\frac{6}{\pi\rho}g\left(\frac{\pi\rho}{6}d_{o}^{3},t\right)\right]^{1/3} \leq a \leq P\left[\frac{6}{\pi\rho}g\left(\frac{\pi\rho}{6}D_{o}^{3},t\right)\right]^{1/3} \quad (6.4)$$

The integration limits L1 and L2 depend on time as follows:

For

$$L_{1} = d_{o} \text{ for t such that } Pg(\frac{\rho\pi}{6}d_{o}^{3},t) > 0$$
 (6.5)

$$L_{1} = \left[\frac{6}{\pi\rho} g^{-1}(o,t)\right]^{1/3} " " " " " = 0$$
 (6.6)

$$L_2 = D_0 \text{ for t such that } Pg \ (\frac{\rho\pi}{6} D_0^3, t) > 0 \tag{6.7}$$

$$L_2 = \left[\frac{6}{\pi\rho} g^{-1}(o,t)\right]^{1/3} " " " " = 0$$
 (6.8)

where d_0 and D_0 denote the initial diameters of the smallest and largest particles, respectively. The operator P has been introduced to make the expression generally applicable (95). It is defined to be equal to one in the time period before the operand becomes zero and is equal to zero beyond that time. The lower integration limit L_1 changes value at $g[(\rho\pi/6)$ $d_0^3, t] = 0$, that is the critical time when the particles start to disappear. The time at which $g[(\rho\pi/6)D_0^3, t] = 0$ corresponds to the disappearance of the last particle and marks the completion of the dissolution process. Many powders have size distributions that are approximately lognormal (122). Consider such a powder distributed such that ln a approximates a normal distribution with mean μ and standard deviation σ truncated at ln d = μ -i σ and at ln D = μ +j σ , where i and j are truncation parameters. The initial particle-size distribution, 1 (a), is then given by (96):

$$a_{0}(a_{0}) = \frac{a_{0}^{-1} N(\ln a_{0}, \mu, \sigma)}{D}$$
 (6.9)

 $\int_{0}^{0} a_{0}^{-1} N(\ln a_{0}, \mu, \sigma) d a_{0}$ $d_{o} \leq a_{o} \leq D_{o}$ where N() is the normal distribution with ln a as the variable. The change in the particle-size distribution during dissolution depends on the way the individual particles dissolve. Three widely known models for single-particle dissolution are considered. When written in the same form as 6.1, the cube root law (97) can be expressed as:

$$w = (w^{1/3} - k_t)^3$$
 (6.10)

For Model 1, m = 3/2; for Model 2, $m^3 - 2$; and for Model 3, m = 3. The In a similar way, the equation presented by Niebergal et al. (106) can be constants k and k for each model should be k. F K. K. and K. A. A. A.

written simply:

$$w = (w_0^{\frac{1}{2}} - k_w t)^2$$
(6.11)

and the model proposed by Higuchi and Hiestand (107) can be written:

$$w = (w_0^{2/3} - k_1 t)^{3/2}$$
(6.12)

For simplicity and because their evaluation is not important to the theoretical discussion, the constants k1, k2 and k3 are used in place of the original time coefficients which included parameters such as the shape

factor, particle density, and diffusion coefficient. In the following section,

Eqs. 6.12, 6.11 and 6.10 will be referred to as models 1, 2 and 3, respectively.

By having defined the initial size distribution 1 (a) (Eq. 6.9),

and the dissolution function (Eqs. 6.10-12) the size distribution at time t, 1(a), can be expressed applying 6.3:

$$(a) = \frac{\left(\frac{3}{m} - 1\right)\left(\frac{3}{m} + Kt\right)^{-1} \left[\left(\frac{3}{m} + Kt\right)^{\frac{3}{m}}, \frac{3}{m}, \frac{3}{m}$$

$$P(EXP[3(\mu-i\sigma)/m]-Kt)^{\frac{m}{3}} \leq a \leq P(EXP[3(\mu+j\sigma)/m]-Kt)^{\frac{m}{3}}$$

and 1(a) = 0 elsewhere. Also:

- $T_1 = \mu i\sigma$ for $\frac{m}{3} \ln (Kt) < \mu i$ (6.14)
- $T_{1} = \frac{m}{3} \ln (Kt) \qquad """ \geq \mu i \qquad (6.15)$

$$T_2 = \mu + j\sigma$$
 for $\frac{\pi}{3} \ln (Kt) < \mu + j$ (6.16)

$$T_{2} = \frac{m}{3} \ln (Kt) \qquad " " " \geq \mu + j \qquad (6.17)$$

$$K = (6/\rho \pi)^{\frac{1}{m}} k \qquad (6.18)$$

and

Shere T. e

For

Equation 6.13 describes the size distribution for all three models. For Model 1, m = 3/2; for Model 2, m = 2; and for Model 3, m = 3. The constants k and K for each model should be k_1 , k_2 , k_3 and K_1 , K_2 , K_3 , respectively. The function F() is the commonly tabulated area under the standard normal curve function defined earlier (5.37).

The main particle size (diameter), a can be obtained by applying the usual integration approach used in mathematical expectation:

$$\overline{a} = \frac{\int_{R} \frac{3}{a^{m}} \left(a^{m} + Kt\right)^{-1} N\left[ln\left(a^{m} + Kt\right)^{m}, \mu, \sigma\right] da}{F\left(\frac{T_{2}-\mu}{\sigma}\right) - F\left(\frac{T_{1}-\mu}{\sigma}\right)}$$
(6.19)

The integration interval R in 6.19 is the same as the interval for a defined in 6.13. Equation 6.19 considers Model 1 and 2 (m = 3/2 and m = 2, respectively). The mean particle size for the third model (m = 3) simplies further to:

$$= \frac{F\left(\frac{T_{2}-\mu}{\sigma} - \sigma\right) - F\left(\frac{T_{1}-\mu}{\sigma} - \sigma\right)}{F\left(\frac{T_{2}-\mu}{\sigma}\right) - F\left(\frac{T_{1}-\mu}{\sigma}\right)} EXP\left(\mu+\sigma^{2}/2\right) - K_{3}t \qquad (6.20)$$

Where T_1 and T_2 are still defined as in Eqs. 6.14-17. The exact dissolution profile of a log-normal powder with single particles dissolving according to each of the three models can be derived using 5.22 presented earlier(96):

$$\frac{W}{W_{O}} = \sum_{n=0}^{m} {m \choose n} (-Kt) \frac{(m-n) F(A) - F(B)}{F(j-3\sigma) - F(-i-3\sigma)} EXP (C)$$
(6.21)

For which

 $A = (T_2^{-\mu})/\sigma - 3n\sigma/m$ (6.22) $B = (T_1^{-\mu})/\sigma - 3n\sigma/m$ (6.23) $C = \frac{3}{m} (n-m) (\mu + \frac{3}{m} (n+m)\sigma^{2}/2)$ (6.24)

and

Where W and W_o are the amounts of undissolved powder at time t and 0, respectively. Equation 6.21 with m = 3, although presented in a more compact form, is identical to 5.38 derived earlier. Equation 6.21 is not defined for Model 1 (m = 3/2) which must be considered separately:

$$= \frac{\int_{R_1}^{R_2} (w^2 - K_1 t)^{3/2} w^{-1} N (\ln w, \mu, \sigma) dw}{F(j - 3\sigma) - F(-i - 3\sigma)} EXP (-3\mu - 9\sigma^{2/2})$$
(6.25)

where

 $R_{1} = EXP(\mu-i\sigma) \text{ for } (K_{1}t)^{\frac{1}{2}} < EXP(\mu-i\sigma)$ (6.26) $R_{1} = (K_{1}t)^{\frac{1}{2}} "" \geq EXP(\mu-i\sigma)$ (6.27) $R_{2} = EXP(\mu+j) \text{ for } (K_{1}t)^{\frac{1}{2}} < EXP(\mu+j\sigma)$ (6.28) $R_{2} = (K_{1}t)^{\frac{1}{2}} "" \geq EXP(\mu+j\sigma)$ (6.29)

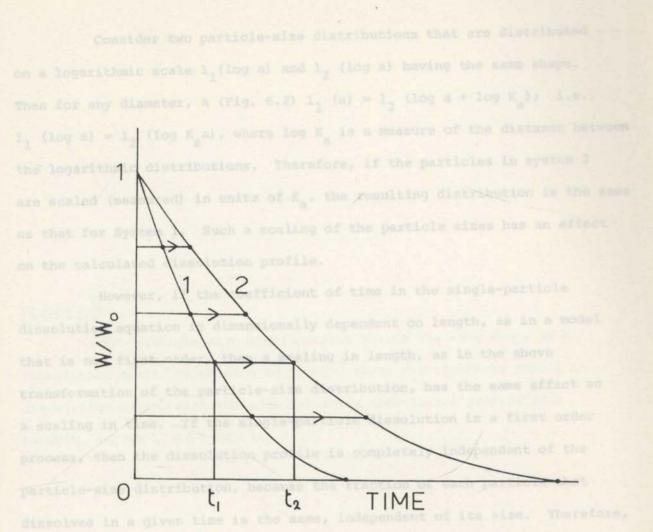
The derivation of these equations are based on two assumptions: (a) that the particles in the multiparticulate system dissolve independently of each other, which will be approximated well under sink conditions; and (b) that they dissolve according to the same single-particle dissolution model having fixed parameters (for these cases, k_1 , k_2 and k_3 are the same for all particles and do not vary during dissolution). If these conditions exist, then it is possible to propose some general rules concerning the dissolution process. These rules are explained in relation to what will be termed "the intrinsic dissolution profile", which can be defined in the following way: Dissolution curves have the same intrinsic dissolution profile if, by a suitable scaling of time, they can be brought into each other in a W/W_{a} versus time plot (Fig. 6.1).

It should be clear from observation of Eqs. 5.17 and 5.22 that the coefficient of time in an expression correctly defining the multiparticulate dissolution profile originates directly from the coefficient of time in the single-particle dissolution equation. Thus a different value of the rate parameter, that is, a different coefficient of time in the singleparticle dissolution equation, has the same effect as a different scaling of time. Therefore, the intrinsic dissolution profile will still be the same. The following rule can thus be stated:

1. The intrinsic dissolution profile is independent of the value of the rate parameter, that is, the coefficient of time in the singleparticle dissolution equation.

According to this rule, the rate parameter k_1 , k_2 and k_3 (6.10-12) have no influence on the intrinsic dissolution profile. Furthermore, there will always be a proportional relationship between the coefficient of time in the multiparticulate dissolution equation and the rate parameter. The following rule can therefore be stated:

2. In two systems having identical particle-size distributions, the time-scaling factor that brings one dissolution curve into another is equal to the factor with which the rate parameters are proportionally related in the two systems. (Fig. 6.1) Two dissolution curves having the same intrinsic dissolution profile. Curve 1 can be brought into curve 2 by a time-scaling factor, t_2/t_1 . Rule 2 states that $k_{(1)} = (t_2/t_1)k_{(2)}$, where $k_{(1)}$ and $k_{(2)}$ are the rate parameters in the single-particle dissolution equation for Systems 1 and 2 having the same particle-size distribution. These statements also include plots where W/W_o is raised to any other exponent.



it can be concluded that system 1 and 1 have the same intrinsic dissolution profile, and the following rule can be given:

 two powders discolving according to the same single-particle dissolution model have the same intrinsic dissolution profile if their particle-size distributions are of the same shape on a logarithmic scale.
 (Fig. 5.2)

It follows from this rule that it is not the "position" of the distribution, that is, not the actual size of the particles. But the shape of the distribution that affects the intrinsic profile. Thus it can be stated that:

 The intrinsic dissolution profile does not depend on the actual size of the particles but on the shape of their distribution. Consider two particle-size distributions that are distributed on a logarithmic scale $l_1(\log a)$ and $l_2(\log a)$ having the same shape. Then for any diameter, a (Fig. 6.2) $l_1(a) = l_2(\log a + \log K_a)$; i.e., $l_1(\log a) = l_2(\log K_a a)$, where $\log K_a$ is a measure of the distance between the logarithmic distributions. Therefore, if the particles in system 2 are scaled (measured) in units of K_a , the resulting distribution is the same as that for System 1. Such a scaling of the particle sizes has an effect on the calculated dissolution profile.

However, if the coefficient of time in the single-particle dissolution equation is dimensionally dependent on length, as in a model that is not first order, then a scaling in length, as in the above transformation of the particle-size distribution, has the same effect as a scaling in time. If the single-particle dissolution is a first order process, then the dissolution profile is completely independent of the particle-size distribution, because the fraction of each particle that dissolves in a given time is the same, independent of its size. Therefore, it can be concluded that system 1 and 2 have the same intrinsic dissolution profile, and the following rule can be given:

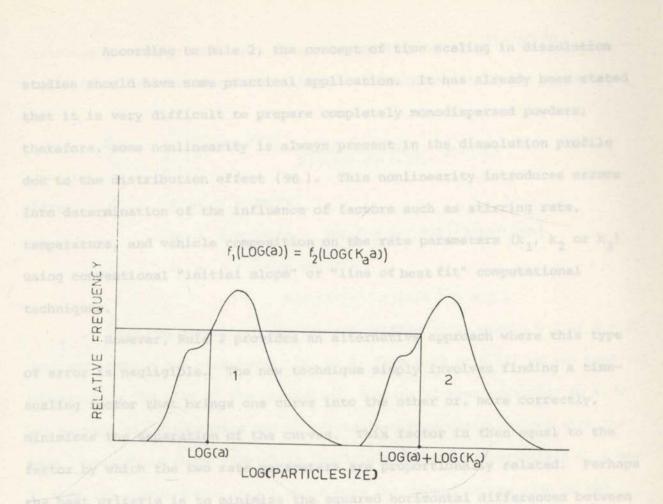
3. Two powders dissolving according to the same single-particle dissolution model have the same intrinsic dissolution profile if their particle-size distributions are of the same shape on a logarithmic scale. (Fig. 6.2)

It follows from this rule that it is not the "position" of the distribution, that is, not the actual size of the particles, but the shape of the distribution that affects the intrinsic profile. Thus it can be stated that:

4. The intrinsic dissolution profile does not depend on the actual size of the particles but on the shape of their distribution.

- 125 -

Two particle-size distributions having the same shape on a logarithmic scale. According to Rule 3, these distributions result in dissolution curves with the same intrinsic dissolution profile (for example, curves 1 and 2 of Fig. 7.1).



the curves.

According to Bales 1 and 3, it should be possible to normalize the calculated dissolution publies for log-normal powders by appropriate exking of time to a form that does not depend on either the rate parameter $(k_1, k_2 \text{ or } k_3)$ or the actual sizes of the particles. One approach is to evaluating in the W/W varsus time plot as the time fraction ψ , defined as the fraction of the time necessary for complete dissolution. The expression dufining the resulting normalized dissolution profile can be physical in the following way, using Models 2 and 3 as examples. The time for complete dissolution, t₀, is given by

(m/3) in (RE.) = projet, then since 0 = t/t, it follows that:

- REIDE -3(0+j0)/m

- 127 -

According to Rule 2, the concept of time scaling in dissolution studies should have some practical application. It has already been stated that it is very difficult to prepare completely monodispersed powders; therefore, some nonlinearity is always present in the dissolution profile due to the distribution effect (96). This nonlinearity introduces errors into determination of the influence of factors such as stirring rate, temperature, and vehicle composition on the rate parameters $(k_1, k_2 \text{ or } k_3)$ using conventional "initial slope" or "line of best fit" computational techniques.

However, Rule 2 provides an alternative approach where this type of error is negligible. The new technique simply involves finding a timescaling factor that brings one curve into the other or, more correctly, minimizes the separation of the curves. This factor is then equal to the factor by which the two rate parameters are proportionally related. Perhaps the best criteria is to minimize the squared horizontal differences between the curves.

According to Rules 1 and 3, it should be possible to normalize the calculated dissolution profiles for log-normal powders by appropriate scaling of time to a form that does not depend on either the rate parameter $(k_1, k_2 \text{ or } k_3)$ or the actual sizes of the particles. One approach is to scale time in the W/W_o versus time plot as the time fraction ψ , defined as the fraction of the time necessary for complete dissolution. The expression defining the resulting normalized dissolution profile can be obtained in the following way, using Models 2 and 3 as examples. The time for complete dissolution, t_o , is given by

 $(m/3)\ln(Kt_0) = \mu + j\sigma$; then since $\psi = t/t_0$ it follows that:

 $\psi = \text{KtEXP}\left[-3(\mu+j\sigma)/m\right]$

(6.30)

Accordingly, the Kt terms in 6.21 can be substituted by $\psi^{3(\mu+j\sigma)/m}$, which causes the μ term to cancel out. After rearrangement, 6.21 can be written:

$$\frac{W}{W_{O}} = \sum_{n=0}^{m} {m \choose n} (-\psi)^{(m-n)} \frac{F(A) - F(B)}{F(j-3\sigma) - F(-i-3\sigma)} EXP (C)$$
(6.31)

where:

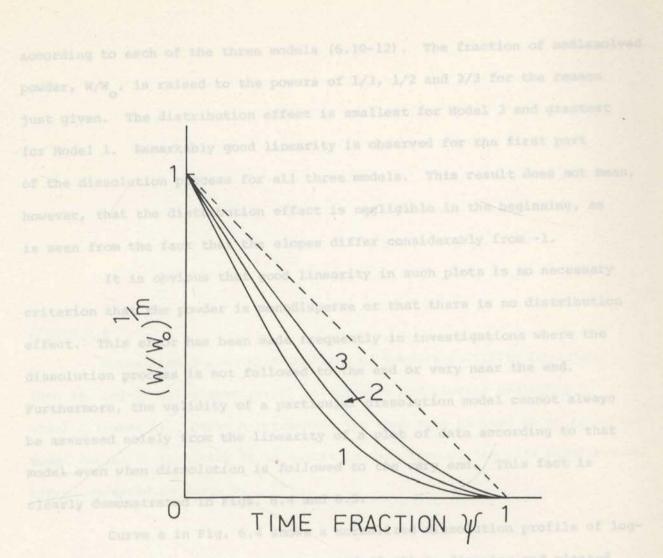
$$B = i - 3n\sigma/m \quad \text{for } 0 \le \psi \le \exp\left[-3(i+j)\sigma/m\right]$$
$$B = j + \frac{m}{3} \ln \psi - 3n\sigma/m \quad \text{for } \exp\left[-3(i+j)\sigma/m\right] \le \psi \le 1$$

Thus the normalized dissolution profile (6.31 with m = 2 or m = 3) does not contain any rate terms $(k_3 \text{ or } k_2 \text{ from 6.10 or 6.11})$ or any term (μ) representing the size of the particles. Scaling of time according to 6.30 has brought all dissolution curves originating from distributions with the same "logarithmic shape" (which is completely defined by parameters σ , i and j) into one single curve (6.31) which does not depend on the size of the particles or the rate parameter from the single-particle equation. This confirms Rule 1 and 3.

The transformation has essentially normalized all possible systems having the same intrinsic dissolution profile into one single curve. This curve is unique in that it makes it possible to evaluate the isolated distribution effect. This evaluation is best done by plotting in a way that linearizes the underlying single-particle dissolution equation (6.10-12), by using $(W/W_0)^{1/m}$ instead of W/W_0 in the plot. Such a plot will be linear with slope = -1 for a true monodisperse system. Any deviation from this linearity and slope will be due solely to the distribution effect.

Figure 6.3 shows such normalized dissolution profiles, calculated according to 6.21 and 6.31, for powders initially log-normal, having distribution (shape) parameters $\sigma = 0.2$ and i = j = 2, for particles dissolving

Normalized dissolution profiles of powders initially log-normal $(\sigma=0.2, i=j=2)$ calculated (Eq. 6.21, m=3,2, Eq. 6.25) to dissolve and plotted according to models 1,2 and 3. The stippled line represents monodisperse powders $(\sigma=0)$.



normal powders (0=0.18, 1=9=2) calculated (5.25) to dissolve and provide (W/W to power of 2/3) eccording to Model 1. The size distribution effect is clearly reflect in the hominearity of the curve. By plotting the same data according to an incorrect model. Model 3 (W/W to power of 1/3), the size distribution effect is almost entirely cancelled and surprisingly pool linearity is obtained that extends to the very end of the dissolution process (curve b).

Figure 5.3 shows the same phenomenon for powders to = 0.12, = j = 2) where the particles are calculated to dissolve according to Hodel (6.31, m = -7).

A judgment based solwly on the linearity of one picts what prove

according to each of the three models (6.10-12). The fraction of undissolved powder, W/W_o, is raised to the powers of 1/3, 1/2 and 2/3 for the reason just given. The distribution effect is smallest for Model 3 and greatest for Model 1. Remarkably good linearity is observed for the first part of the dissolution process for all three models. This result does not mean, however, that the distribution effect is negligible in the beginning, as is seen from the fact that the slopes differ considerably from -1.

It is obvious that good linearity in such plots is no necessary criterion that the powder is monodisperse or that there is no distribution effect. This error has been made frequently in investigations where the dissolution process is not followed to the end or very near the end. Furthermore, the validity of a particular dissolution model cannot always be assessed solely from the linearity of a plot of data according to that model even when dissolution is followed to the very end. This fact is clearly demonstrated in Figs. 6.4 and 6.5.

Curve a in Fig. 6.4 shows a normalized dissolution profile of lognormal powders (σ =0.14, i=j=2) calculated (6.25) to dissolve and plotted (W/W_o to power of 2/3) according to Model 1. The size distribution effect is clearly reflect in the nonlinearity of the curve. By plotting the same data according to an incorrect model, Model 3 (W/W_o to power of 1/3), the size distribution effect is almost entirely cancelled and surprisingly good linearity is obtained that extends to the very end of the dissolution process (curve b).

Figure 6.5 shows the same phenomenon for powders ($\sigma = 0.12$, i = j = 2) where the particles are calculated to dissolve according to Model 2 (6.31, m = 2).

A judgment based solely on the linearity of such plots will often lead to false conclusions about the validity of the model, even where

- 132 -

Figure 6.4

Normalized dissolution profiles of powders initially log-normal $(\sigma=0.14, i=j=2)$ calculated (6.25) to dissolve and plotted according to model 1 (curve a). Curve b represents the same data plotted to an incorrect model (model 3) resulting in an almost cancellation of the size distribution effect observed in curve a.

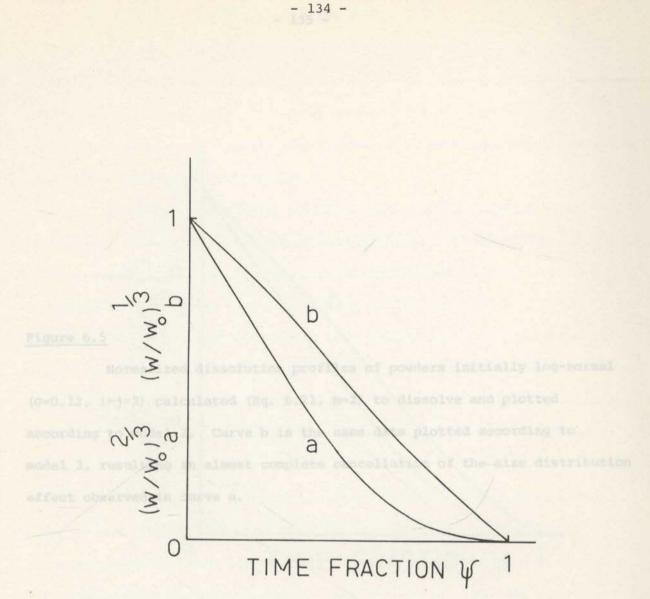
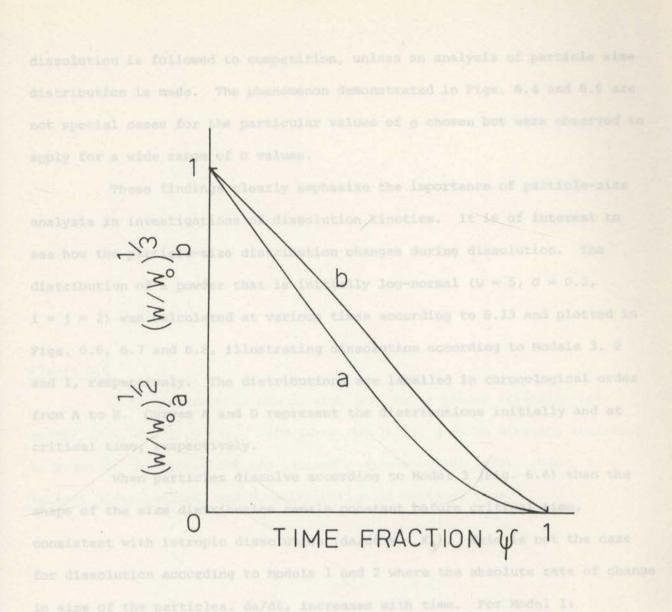


Figure 6.5

Normalized dissolution profiles of powders initially log-normal $(\sigma=0.12, i=j=2)$ calculated (Eq. 6.21, m=2) to dissolve and plotted according to model 2. Curve b is the same data plotted according to model 3, resulting in almost complete cancellation of the size distribution effect observed in curve a.



- 136 -

and for Model 2+

As a result, the distribution broadens before critical the (Figs. 6.7 and 5.8) and is particularly affected at the small particles and as zero is approached where de/dt takes extreme values. Receive er me latter effort, near the end of the dissolution process the relative frequency of the very small particles increases with increasing size for Models 1 and 2 (Fig. 5.8 and 5.7) while it desreases for Model 1 (Fig. 5.6). This information dissolution is followed to competition, unless an analysis of particle size distribution is made. The phenomenon demonstrated in Figs. 6.4 and 6.5 are not special cases for the particular values of σ chosen but were observed to apply for a wide range of σ values.

These findings clearly emphasize the importance of particle-size analysis in investigations of dissolution kinetics. It is of interest to see how the particle-size distribution changes during dissolution. The distribution of a powder that is initially log-normal ($\mu = 5$, $\sigma = 0.2$, i = j = 2) was calculated at various times according to 6.13 and plotted in Figs. 6.6, 6.7 and 6.8, illustrating dissolution according to Models 3, 2 and 1, respectively. The distributions are labelled in chronological order from A to H. Curves A and D represent the distributions initially and at critical time, respectively.

When particles dissolve according to Model 3 (Fig. 6.6) then the shape of the size distribution remain constant before critical time, consistent with istropic dissolution $(da/dt = -K_3)$. This is not the case for dissolution according to Models 1 and 2 where the absolute rate of change in size of the particles, da/dt, increases with time. For Model 1:

$$da/dt = -\frac{K_1}{2} (a_0^2 - K_1 t)^{-\frac{1}{2}}$$
 (6.32)

and for Model 2:

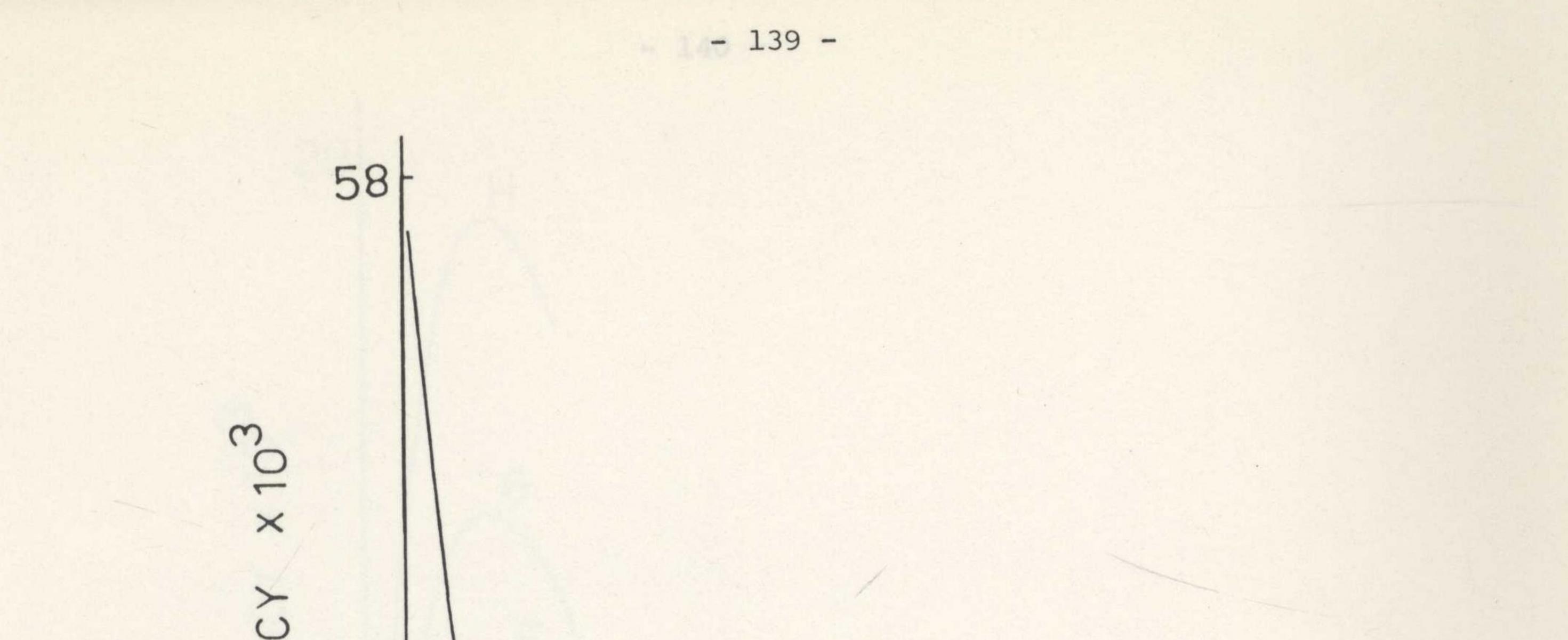
$$da/dt = -\frac{2}{3} \kappa_2 (a^{3/2} - \kappa_2 t)^{-1/3}$$
 (6.33)

As a result, the distribution broadens before critical time (Figs. 6.7 and 6.8) and is particularly affected at the small particle end as zero is approached where da/dt takes extreme values. Because of the latter effect, near the end of the dissolution process the relative frequency of the very small particles increases with increasing size for Models 1 and 2 (Fig. 6.8 and 6.7) while it decreases for Model 3 (Fig. 6.6). This information

Figure 6.6

Size distribution change with time for a powder initially lognormal (μ =5, σ =0.2, i=j=2), calculated (Eq. 6.13, m=3) to dissolve according to Model 3. Distributions are labeled in chronological order. Key: A, initial distribution; and D, distribution at critical time.

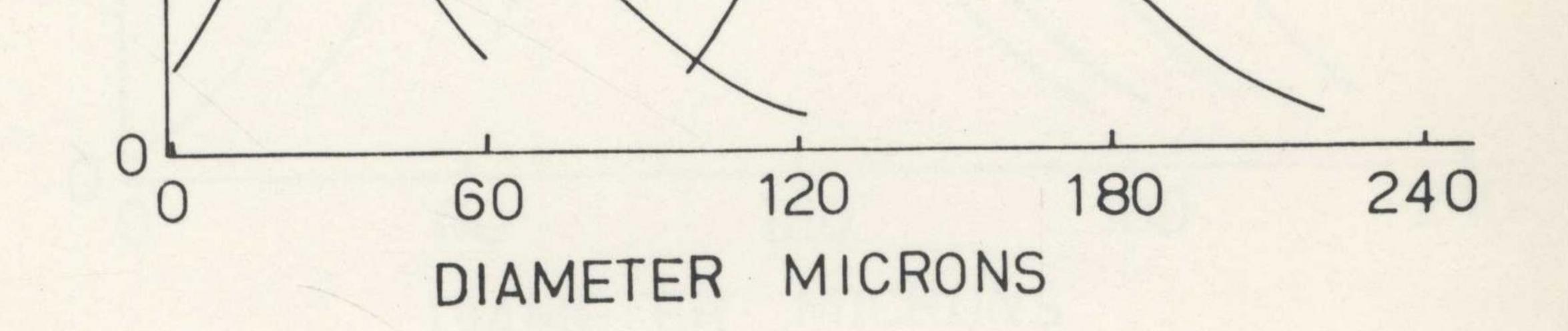
- 138 -



0 Ц Ц 29 Size distribution change with time for a powder initially lognormal (µ 2 0=0) ш ix æ

H

F

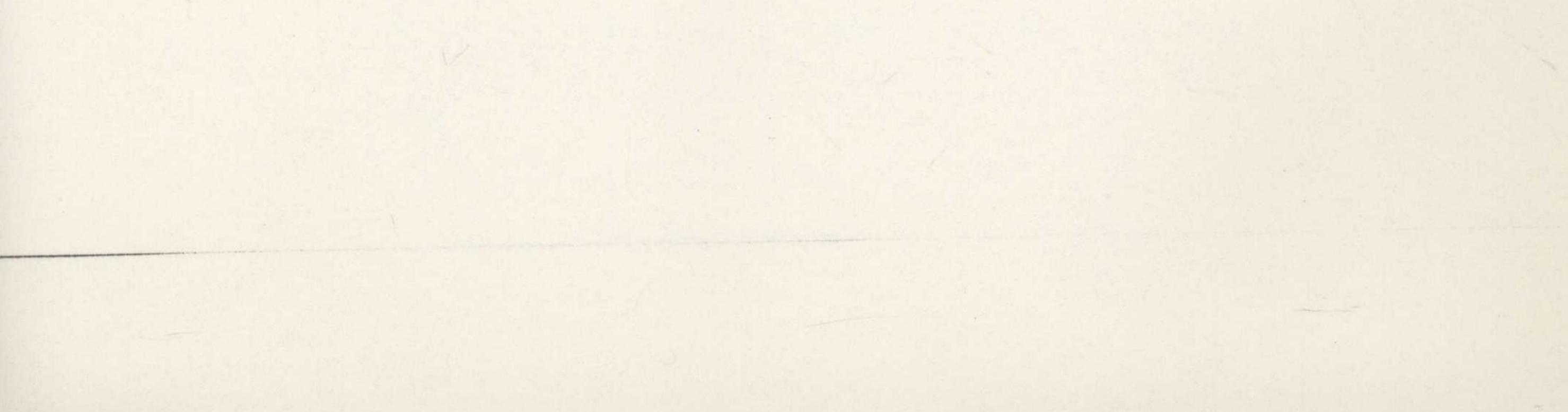


=2), calculated (Eq. 6.13, m=2) to dissolve

istribition; and D, distribution at critical time.

listributions are labeled in chronological order.

A



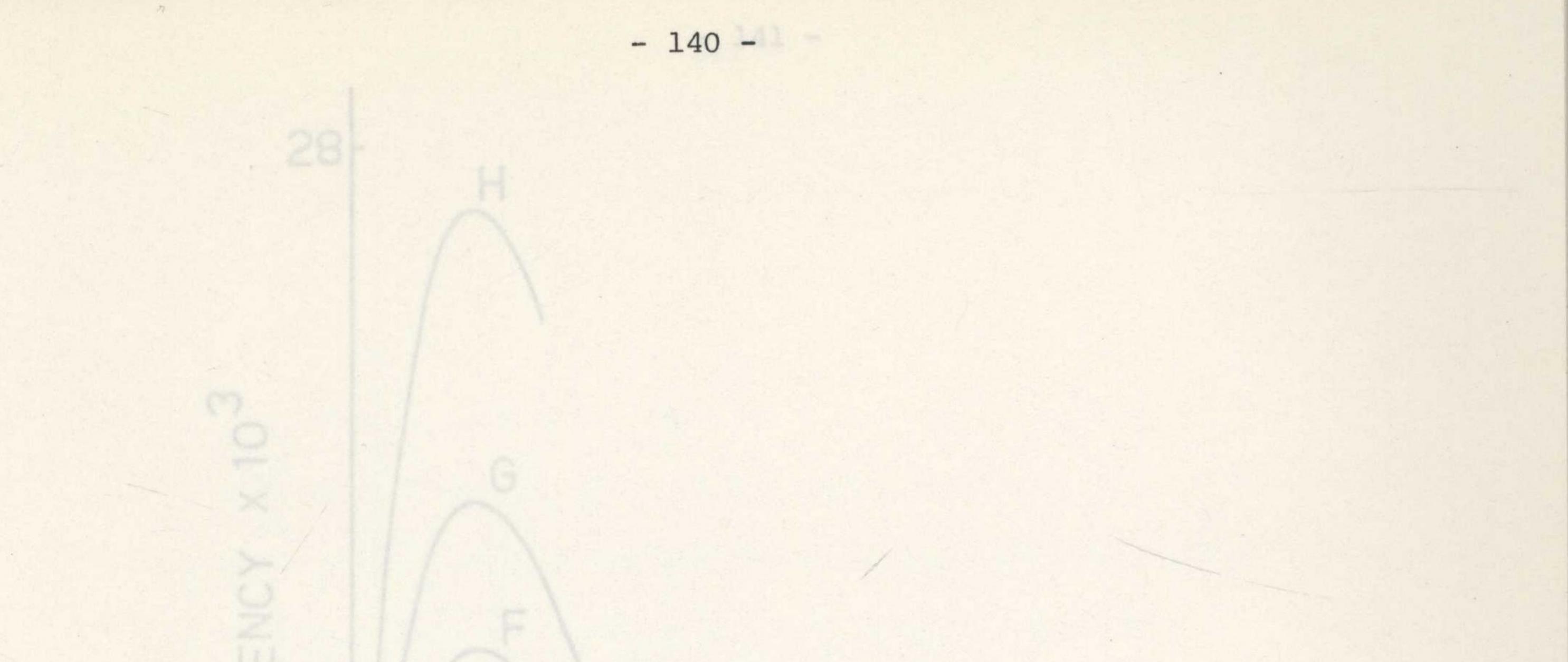
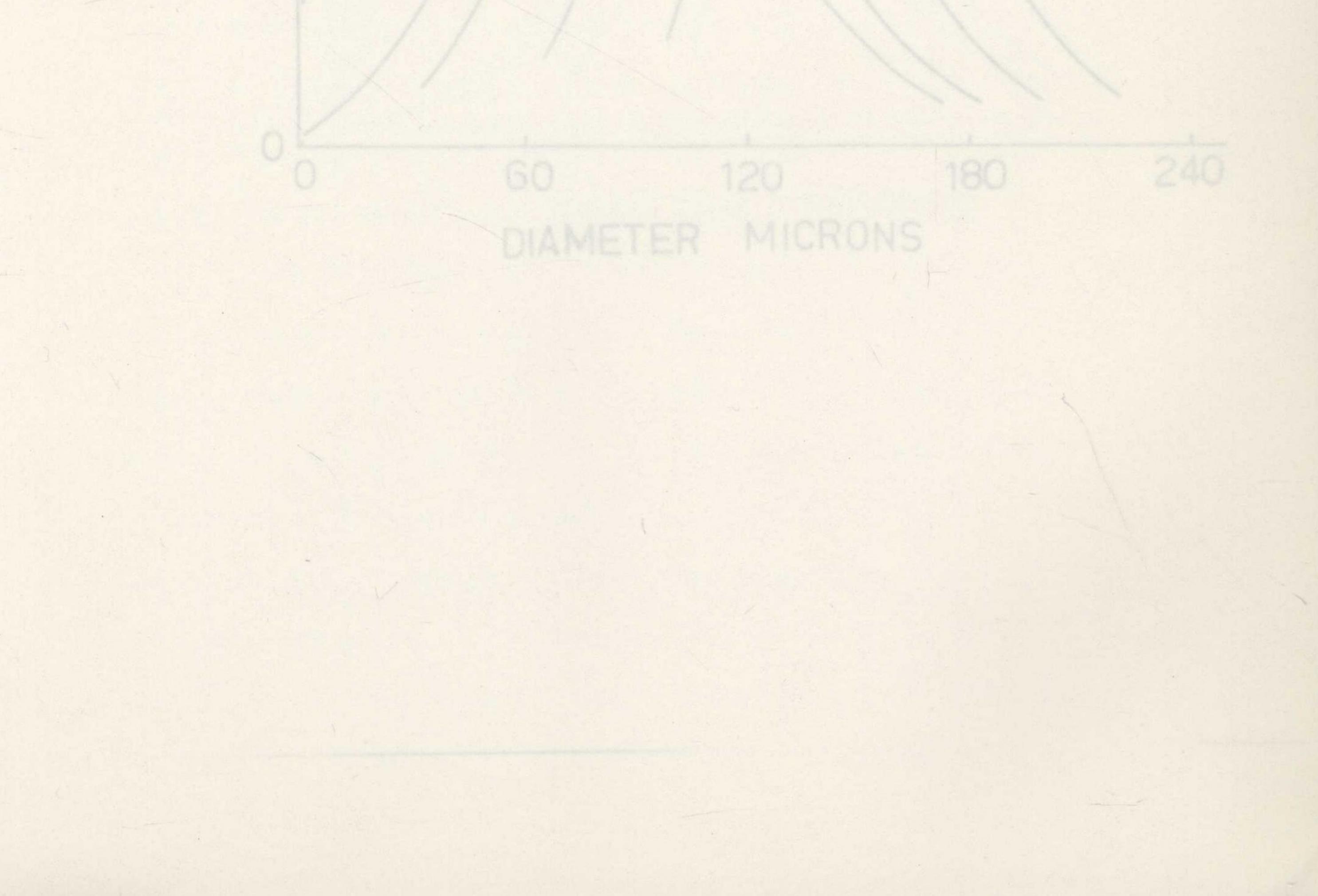
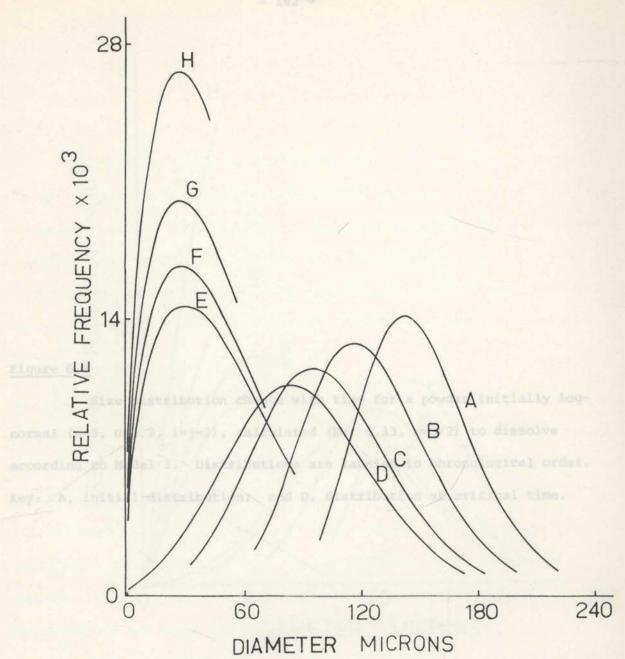


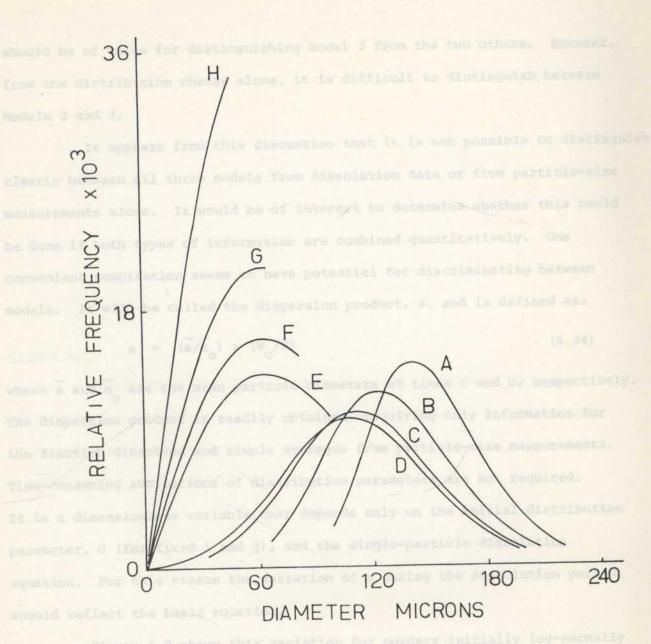
Figure 6.7.

Size distribution change with time for a powder initially lognormal (μ =5, σ =0.2, i=j=2), calculated (Eq. 6.13, m=2) to dissolve according to Model 2. Distributions are labeled in chronological order. Key: A, initial distribution; and D, distribution at critical time.





Size distribution change with time for a powder initially lognormal (μ =5, σ =0.2, i=j=2), calculated (Eq. 6.13, m=3/2) to dissolve according to Model 1. Distributions are labeled in chronological order. Key: A, initial distribution; and D, distribution at critical time.



distributed (0 = 0.2, t = j = z). The curves representing the three dissolution models are significantly different. The basic shape of the curve remains the same for varying values of 0, eithough the minime chiff to the right and to higher values for yery narrow distributions. All three curves approach s > 1 (stippled line) when 0 approaches zero as reperted for a completely monodisperse powder. The values of the three minime remain approximately constant. When 0 ranges from scenachat less than 0.1 to at least 0.2 (Fig. 5.10), which exceptions much five powder distributions should be of value for distinguishing model 3 from the two others. However, from the distribution change alone, it is difficult to distinguish between Models 2 and 3.

It appears from this discussion that it is not possible to distinguish clearly between all three models from dissolution data or from particle-size measurements alone. It would be of interest to determine whether this could be done if both types of information are combined quantitatively. One convenient combination seems to have potential for discriminating between models. It will be called the dispersion product, s, and is defined as:

$$s = (\overline{a}/\overline{a}) \times (W_{O}/W)$$
(6.34)

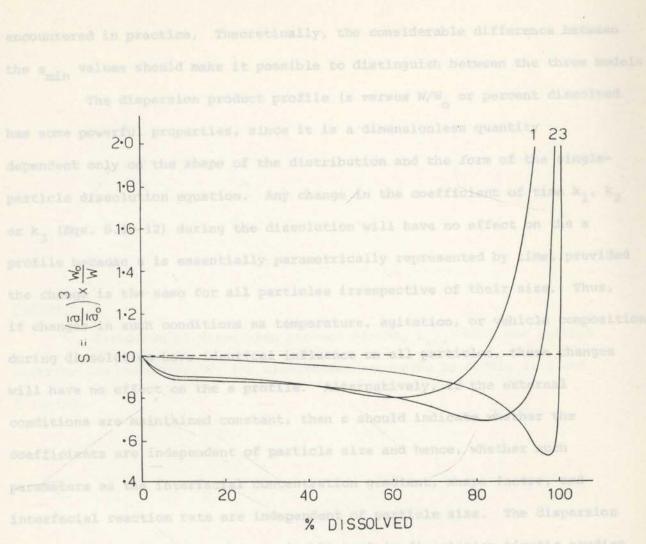
where \bar{a} and \bar{a}_{0} are the mean particle diameters at times t and 0, respectively. The dispersion product is readily obtained, requiring only information for the fraction dissolved and simple averages from particle-size measurements. Time-consuming evaluations of distribution parameters are not required. It is a dimensionless variable that depends only on the initial distribution parameter, σ (for fixed i and j), and the single-particle dissolution equation. For this reason the variation of s during the dissolution period should reflect the basic equation.

Figure 6.9 shows this variation for powders initially log-normally distributed ($\sigma = 0.2$, i = j = z). The curves representing the three dissolution models are significantly different. The basic shape of the curve remains the same for varying values of σ , although the minima shift to the right and to higher values for very narrow distributions. All three curves approach s = 1 (stippled line) when σ approaches zero as expected for a completely monodisperse powder. The values of the three minima remain approximately constant. When σ ranges from somewhat less than 0.1 to at least 0.2 (Fig. 6.10), which encompasses most fine powder distributions

- 144 -

Figure 6.9

Variation of the dispersion product (Eq. 6.34) with progress of dissolution of powders initially log-normal (σ =0.2, i=j=2) calculated to dissolve according to model 1, 2 and 3. The stippled line represents a monodisperse powder (σ =0).



The extent to which these methematical models can be applied to

1. An montioned earlier, it was assumed that the particles discolve independently of each other. This should be approximated well under sink conditions.

2. It was assumed that the dissolution of each particle in the peader can be described by an equation having the same parameter when this for all the particles. In practice, this scamption is rerely welld however of differences in individual particle shapes, crystal structure, and interaction with the vehicle. However, these types of effects protobly can be averaged to produce a parameter value for the single-particle dissolution encountered in practice. Theoretically, the considerable difference between the s_{min} values should make it possible to distinguish between the three models.

The dispersion product profile (s versus W/W_o or percent dissolved has some powerful properties, since it is a dimensionless quantity dependent only on the *shape* of the distribution and the *form* of the singleparticle dissolution equation. Any change in the coefficient of time k_1 , k_2 or k_3 (Eqs. 6.10-12) during the dissolution will have no effect on the s profile because s is essentially parametrically represented by time, provided the change is the same for all particles irrespective of their size. Thus, if changes in such conditions as temperature, agitation, or vehicle composition during dissolution have identical influence on all particles, these changes will have no effect on the s profile. Alternatively, if the external conditions are maintained constant, then s should indicate whether the coefficients are independent of particle size and hence, whether such parameters as the interfacial concentration gradient, shape factor, and interfacial reaction rate are independent of particle size. The dispersion product should, therefore, be a valuable tool in dissolution kinetic studies.

The extent to which these mathematical models can be applied to describe the dissolution of a "real" powder depends on three assumptions.

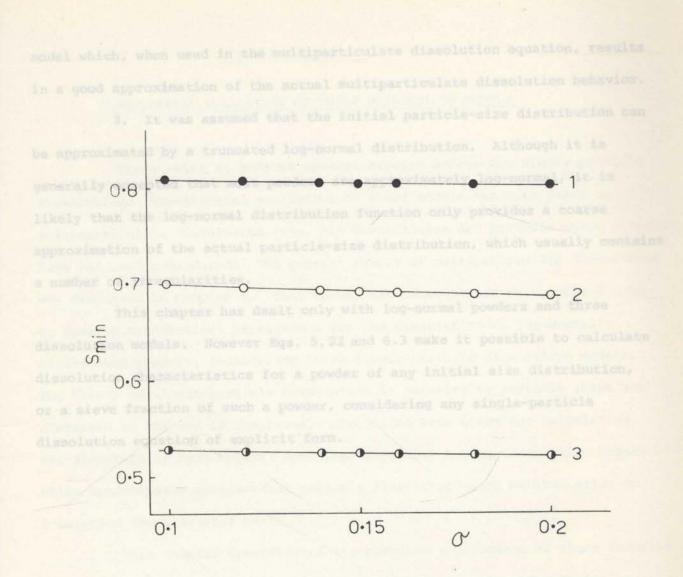
 As mentioned earlier, it was assumed that the particles dissolve independently of each other. This should be approximated well under sink conditions.

2. It was assumed that the dissolution of each particle in the powder can be described by an equation having the same parameter value (k) for all the particles. In practice, this assumption is rarely valid because of differences in individual particle shapes, crystal structure, and interaction with the vehicle. However, these types of effects probably can be averaged to produce a parameter value for the single-particle dissolution

- 147 -

Figure 6.10

Variation of dispersion product minima, S_{min} , with the initial distribution parameter, σ , for dissolution according to Models 1, 2 and 3 (i=j=2).



model which, when used in the multiparticulate dissolution equation, results in a good approximation of the actual multiparticulate dissolution behavior.

3. It was assumed that the initial particle-size distribution can be approximated by a truncated log-normal distribution. Although it is generally accepted that most powders are approximately log-normal, it is likely that the log-normal distribution function only provides a coarse approximation of the actual particle-size distribution, which usually contains a number of irregularities.

This chapter has dealt only with log-normal powders and three dissolution models. However Eqs. 5.22 and 6.3 make it possible to calculate dissolution characteristics for a powder of any initial size distribution, or a sieve fraction of such a powder, considering any single-particle dissolution equation of explicit form.

a weighted least equires hosis. This chapter demonstrates the combined application of these the to explain the discolution kinetics of 50/85-secch fraction of tolbutemide such that both size distribution effects and particle, shape effects are considered. By using a time-scaling approach, the three single-perticin

Theoretical Considerations

The three equations for dissolution of spherical particular under sink conditions considered can be written in compon form sur

- 150 -

CHAPTER 7

EXPERIMENTAL EVALUATION OF THREE DISSOLUTION MODELS

There exist at present several kinetic models for single particle dissolution. Experimental evaluation of these models has been based on multiparticulate dissolution data, but distribution and particle shape effects have not been considered. The general theory of multiparticulate dissolution was discussed in Chapter 5. This theory was subsequently in Chapter 6 used to develop mathematical expressions for the dissolution of log-normally distributed powders, considering three single-particle dissolution models. The theory of single-particle dissolution in relation to particle shape was discussed at the end of Chapter 4. Directions were given for calculating the diameters of hypothetical spherical particles having dissolution behaviour which approximates nonspherical particle dissolution with minimum error on a weighted least squares basis.

and shape factor. This simplification is

This chapter demonstrates the combined application of these theories to explain the dissolution kinetics of 60/85-mesh fraction of tolbutamide such that both size distribution effects and particle shape effects are considered. By using a time-scaling approach, the three single-particle dissolution kinetic models are evaluated.

Theoretical Considerations

The three equations for dissolution of spherical particles under sink conditions considered can be written in common form as:

$$w = (w_0^{\frac{1}{m}} - kt)^m \qquad m = 3, 2, \frac{3}{2}$$
(7.1)
$$k = k_0, k_0, k_0$$

- 152 -

("-RE) 3/2 w 1 N(1n W, 11,0) dw representing the cube root, the square root and the 3/2-root models, respectively (97,106,107). The positive constants k1, k2 and k3 replace the original coefficients of time. These contained quantities such as density, diffusion coefficient and shape factor. This simplification is made because the aim is not to evaluate the theoretical basis of the three equations but solely to assess them as models for describing the dissolution

kinetic data.

These equations (7.1) do not strictly describe the dissolution

correctly in their present form since w does not vanish for t $\rightarrow \infty$. A more

correct formulation would therefore be:

$$w = (w_{o}^{\frac{1}{m}} - kt)^{m} \text{ for } t \leq w_{o}^{\frac{1}{m}} / k$$

$$w = 0 \qquad \text{ for } t \geq w_{o}^{\frac{1}{m}} / k$$

$$(7.2)$$

The general multiparticulate dissolution equation presented earlier (5.22) however was developed such that it accepts the single particle dissolution function in the forms of both 7.1 and 7.2. Equation 7.2 will be used for end at utjo, where i and j are truncation parameters (Fig. 5.1).

simplicity. The multiparticulate dissolution equations above consider spherical In Chapter 6 the following equations were presented to describe the drug particles. Such particles are only encountered when the drug exists in dissolution profile of a multiparticulate log-normal system where the Louid form as an emulsion. In solid form the particles are not spherical. spherical particles dissolve according to each of the three models (7.1): dissolution tests was a 60/85 mesh fraction of

$$\frac{W}{W_{O}} = \sum_{n=0}^{m} {m \choose n} (-Kt)^{(m-n)} \frac{F(A) - F(B)}{F(j - 3\sigma) - F(-i - 3\sigma)} EXP (C)$$
(7.3)
where $A = (T_2 - \mu)/\sigma - 3n\sigma/m$
 $B = (T_2 - \mu)/\sigma - 3n\sigma/m$

$C = EXP\left[\frac{3}{m}(n-m)(\mu+\frac{3}{m}(n+m)\sigma^2/2)\right]$ $T_1 = \max\left[\frac{m}{3}\ln(Kt), \mu-i\sigma\right]$ $T_2 = \max\left[\frac{m}{3}\ln(Kt), \mu+j\sigma\right]$

 $2 - (2 - \frac{2}{3})^{1/3}$

$$- \frac{153}{m} - \frac{$$

where $R_1 = \max \left[(Kt)^{\frac{1}{2}}, EXP(\mu-i\sigma) \right]$ $R_2 = \max \left[(Kt)^{\frac{1}{2}}, EXP(\mu+j\sigma) \right]$

The function F() above is the area under standard normal curve function defined earlier (5.37). The function N(ln w, μ , σ ,) is the normal distribution function with ln w as variable, defined by:

N
$$(\ln w, \mu, \sigma) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{\ln w - \mu}{\sigma} \right)^2 \right]$$

The constant K is related to k1, k2 and k3 by: Constant K is related to k1, k2 and k3 by:

$$K = (6/\rho\pi)^{\frac{1}{m}} k_{i}$$
 $i = 1, 2, 3$ (7.6)

where ρ is the particle density.

It is assumed that the particle diameter (a) distribution is "lognormal", that is, ln a can be approximated by a normal distribution (mean = μ , standard deviation = σ) truncated at the lower end at μ -i σ and at the upper end at μ +j σ , where i and j are truncation parameters (Fig. 5.1).

The multiparticulate dissolution equations above consider spherical drug particles. Such particles are only encountered when the drug exists in liquid form as an emulsion. In solid form the particles are not spherical. The drug used for the dissolution tests was a 60/85 mesh fraction of Tolbutamide consisting of particles approximately tetragonal prismatic in shape. It was shown in Chapter 4 that the dissolution of such particles can be approximated well by the dissolution of hypothetical spherical particles. The equivalent spherical diameter, a, is the diameter of the spherical particle that best approximates the dissolution of the non-spherical particle, and is given by:

$$a = \frac{b_{o}}{2 - (2 - \frac{b_{o}}{1_{o}})^{1/3}}$$

(7.7)

where 1_o and b_o are the length and side respectively of the tetragonal particle. In this way dissolution of the non-spherical particle system can be suitably described by the dissolution of a hypothetical spherical particle system that can be rigorously treated using the equations above.

It is evident from these that k, μ , σ , i, j and ρ must be known to calculate the dissolution profile. The distribution parameters μ , σ , i and j can be obtained from micrographs and ρ by a standard method, but the single particle rate parameter k is unknown. It is possible however to calculate the exact intrinsic dissolution profile with much less information. According to the rules given previously (Chapter 6) for multiparticulate dissolution only the *shape* of the initial distribution, that is, of the above six parameters only σ , i and j are required to calculate the intrinsic dissolution profile when the single particle dissolution model is known. The concept of time scaling was discussed in Chapter 6. By such an approach it is possible to evaluate quantitatively the difference between the actual dissolution data and the calculated intrinsic dissolution profile.

Figure 7.1 illustrates this application of time scaling. Curve B through the experimental data points represents the dissolution curve $((w/w_0)^{\frac{1}{m}} versus t)$ and curve A the corresponding calculated normalised intrinsic dissolution profile $((w/w_0)^{\frac{1}{m}} versus \psi)$.

Let, N, be the number of data points and, f, the time scaling factor that brings curve B "into" curve A such that the sum of the squared deviations between the curves given by:

ss =
$$\sum_{i=1}^{N} (ft_i - \psi_i)^2$$
 (7.8)

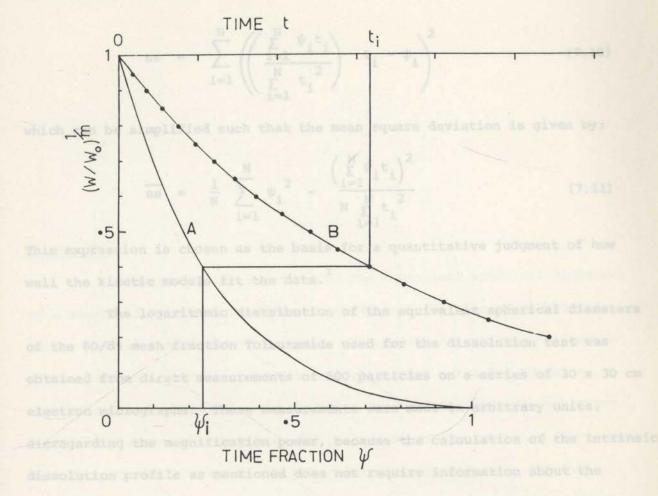
is a minimum. This means f is obtained from $\partial ss/\partial f = 0$ which gives:

$$f = \frac{\sum_{i=1}^{N} \psi_i t_i}{\sum_{i=1}^{N} t_i^2}$$
(7.9)

- 154 -

Figure 7.1

Time-scaling approach (Eqs. 7.8-7.11) used in Figs. 7.3-7.5 to evaluate the agreement between the observed dissolution data and the theoretical dissolution curve calculated in form of the normalized intrinsic dissolution profile, A, on the basis of particle-size analysis.



a minimum of the sum of squares is therefore

actual alars of the particles but the shape of their distribution (that is, 9, 1 and 3 for a log-normal powder). Each particle was approximated by the tetragonal prismatic body which fitted best, and its equivalent spherical diameter was calculated using Sq. 7.7.

The histogram of the logarithm of these dismeters (Fig. 7.2) shows a good fit to a normal distribution with standard deviation $\sigma = 0.395$ and mean

 The alternative approach, to bring curve A "into" surve B, that has the character of a curve fitting to the data points would yield the same result in a comparison of the models. The above time scaling of the date is used for the convenience of plotting and to better illustrate the predicted time for complete dissolution. The minimum of the sum of squares is therefore:

ss =
$$\sum_{i=1}^{N} \left(\left(\frac{\sum_{i=1}^{N} \psi_{i} t_{i}}{\sum_{i=1}^{N} t_{i}^{2}} \right) t_{i} - \psi_{i} \right)^{2}$$
(7.10)

which can be simplified such that the mean square deviation is given by:

$$\overline{ss} = \frac{1}{N} \sum_{i=1}^{N} \psi_i^2 - \frac{\left(\sum_{i=1}^{N} \psi_i t_i\right)^2}{\sum_{i=1}^{N} t_i^2}$$
(7.11)

This expression is chosen as the basis for a quantitative judgment of how well the kinetic models fit the data.

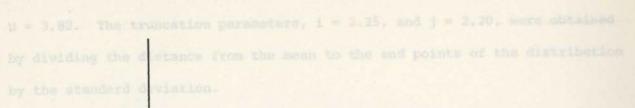
The logarithmic distribution of the equivalent spherical diameters of the 60/85 mesh fraction Tolbutamide used for the dissolution test was obtained from direct measurements of 500 particles on a series of 30 x 30 cm electron micrographs. These measurements were made in arbitrary units, disregarding the magnification power, because the calculation of the intrinsic dissolution profile as mentioned does not require information about the actual sizes of the particles but the shape of their distribution (that is, σ , i and j for a log-normal powder). Each particle was approximated by the tetragonal prismatic body which fitted best, and its equivalent spherical diameter was calculated using Eq. 7.7.

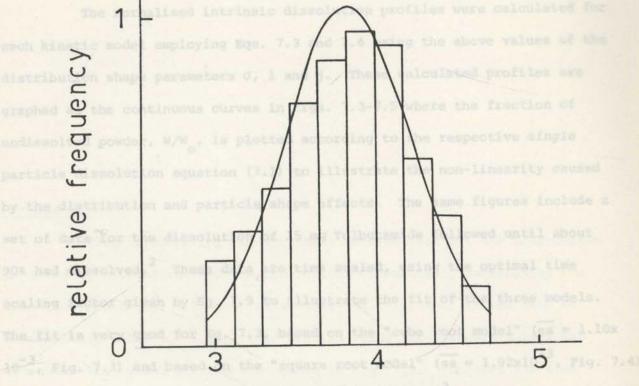
The histogram of the logarithm of these diameters (Fig. 7.2) shows a good fit to a normal distribution with standard deviation $\sigma = 0.395$ and mean

The alternative approach, to bring curve A "into" curve B, that has the character of a curve fitting to the data points would yield the same result in a comparison of the models. The above time scaling of the data is used for the convenience of plotting and to better illustrate the predicted time for complete dissolution.

Figure 7.2

Histogram of the logarithm of the equivalent spherical diameters of a sample of 500 particles from the 60-85 mesh fraction of tolbutamide powder used in the dissolution tests. The equivalent spherical diameters of the particles that are approximately tetragonal prismatic in shape were calculated according to Eq. 7.7 from measurements made (in arbitrary units) on electron micrographs. The parameters of the truncated log-normal distribution which best fits this diameter distribution are: $\sigma = 0.395$, $\mu = 3.82$, i = 2.25, and j = 2.20.





ln (equivalent diameter) arbitrary units

the "cube rout model" describes the single particle dissolution best with the "square root model" sincet as good, however, the "2/3-root model" is relatively poor.

In order to make the above evaluations of the three models by comparing dissolution data with theoretical calculations it is necessary that the experimental conditions are consistent with the assumptions behind these calculations. The three assumptions on which Eqs. 7.3 and 7.4 are based are:

The ansorbance after 90% discolution was so shall that enhanced in an would have been introduced if the process were followed much further.

 μ = 3.82. The truncation parameters, i = 2.25, and j = 2.20, were obtained by dividing the distance from the mean to the end points of the distribution by the standard deviation.

The normalised intrinsic dissolution profiles were calculated for each kinetic model employing Eqs. 7.3 and 7.4 using the above values of the distribution shape parameters σ , i and j. These calculated profiles are graphed as the continuous curves in Figs. 7.3-7.5 where the fraction of undissolved powder, W/W_{σ} , is plotted according to the respective *single* particle dissolution equation (7.1) to illustrate the non-linearity caused by the distribution and particle shape effects. The same figures include a set of data for the dissolution of 15 mg Tolbutamide followed until about 90% had dissolved.² These data are time scaled, using the optimal time scaling factor given by Eq. 7.9 to illustrate the fit of the three models. The fit is very good for Eq. 7.3, based on the "cube root model" ($\overline{ss} = 1.10x$ 10^{-3} , Fig. 7.3) and based on the "square root model" ($\overline{ss} = 1.92x10^{-3}$, Fig. 7.4) but not so good for the "2/3-root model" ($\overline{ss} = 6.70x10^{-3}$, Fig. 7.5).

The dissolution test was done several times using different amounts of Tolbutamide. The \overline{ss} - values (Table 7.1) indicate (F-test, P < .05) that the "cube root model" describes the single particle dissolution best with the "square root model" almost as good, however, the "2/3-root model" is relatively poor.

In order to make the above evaluations of the three models by comparing dissolution data with theoretical calculations it is necessary that the experimental conditions are consistent with the assumptions behind these calculations. The three assumptions on which Eqs. 7.3 and 7.4 are based are:

- 160 -

^{2.} The absorbance after 90% dissolution was so small that substantial error would have been introduced if the process were followed much further.

Figure 7.3

Plot illustrating the agreement between data for the dissolution of 15 mg of tolbutamide and the theoretical dissolution, considering the cube root model (Eq. 7.1, m=3) calculated (Eq. 7.3, m=3) in the form of a normalized intrinsic dissolution profile using the parameters from the truncated log-normal distribution shown in Fig. 7.2. The data are time scaled using the scaling factor given by Eq. 7.9. The mean square deviation $\overline{ss} = 1.10 \times 10^{-3}$.

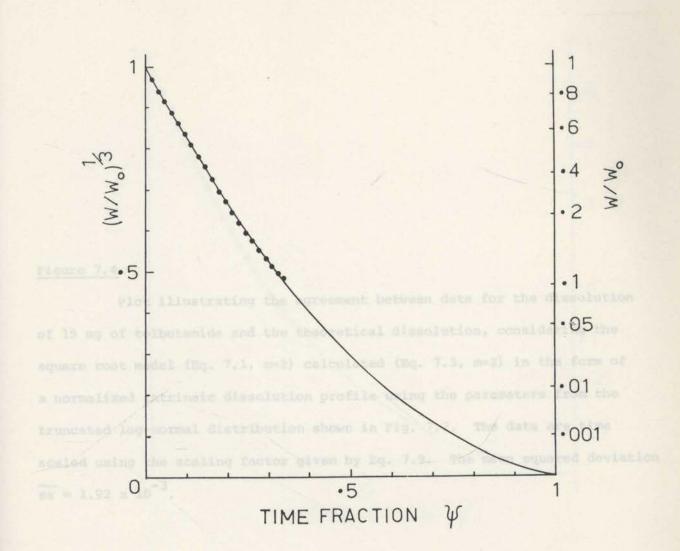
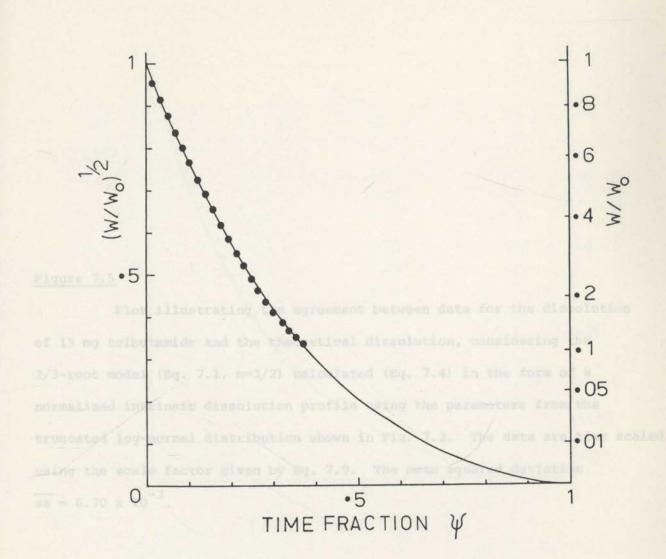


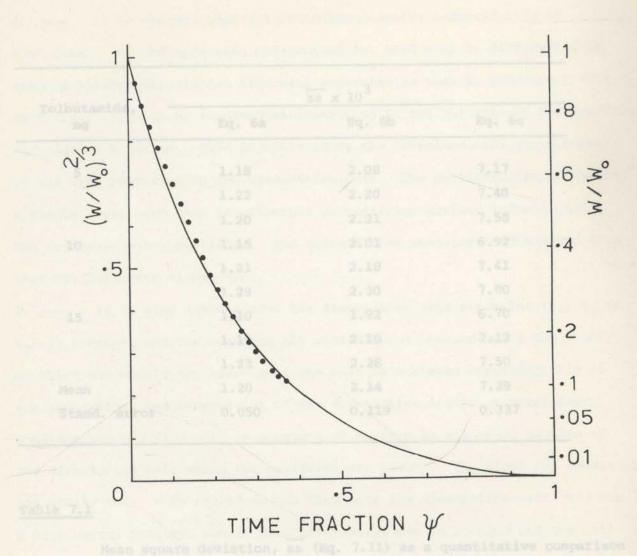
Figure 7.4

Plot illustrating the agreement between data for the dissolution of 15 mg of tolbutamide and the theoretical dissolution, considering the square root model (Eq. 7.1, m=2) calculated (Eq. 7.3, m=2) in the form of a normalized intrinsic dissolution profile using the parameters from the truncated log-normal distribution shown in Fig. 7.2. The data are time scaled using the scaling factor given by Eq. 7.9. The mean squared deviation $\overline{ss} = 1.92 \times 10^{-3}$.

- 163 -



Plot illustrating the agreement between data for the dissolution of 15 mg tolbutamide and the theoretical dissolution, considering the 2/3-root model (Eq. 7.1, m=3/2) calculated (Eq. 7.4) in the form of a normalized intrinsic dissolution profile using the parameters from the truncated log-normal distribution shown in Fig. 7.2. The data are time scaled using the scale factor given by Eq. 7.9. The mean squared deviation $\overline{ss} = 6.70 \times 10^{-3}$.



of the fit of the three multiparticulate dissolution equations. Eqs. 7.3 (m=2,3) and Eq. 7.4 (based on the single-particle dissolution models of Eq. 7.1 (m=3,2, 3/2) to the data from the dissolution of various encunts of 60/85 mesh fraction of tolbutamide.

Tolbutamide,	ss x 10 ³		
	Eq. 6a	Eq. 6b	Eq. 6c
Lanolution kinetics.	This is achie	out by the "abeo	lute sink arising
t the drug particity	1.18	2.08	7.17
	1.22	2.20	7.40
	1.20	2.21	7.58
10	1.15	2.01	6.92
	1.21	2.18	7.41
	1.29	2.30	7.80
15 It is also	1.10	1.92	6.70
	1.19	2.10	7.12
	1.23	2.26	7.50
Mean	1.20	2.14	7.29
Stand. error	0.050	0.119	0.337

ad that the particles discolve independencia

Table 7.1

Mean square deviation, \overline{ss} (Eq. 7.11) as a quantitative comparison of the fit of the three multiparticulate dissolution equations. Eqs. 7.3 (m=2,3) and Eq. 7.4 (based on the single-particle dissolution models of Eq. 7.1 (m=3,2, 3/2) to the data from the dissolution of various amounts of 60/85 mesh fraction of tolbutamide. - 168 -

It is assumed that the particles dissolve independently of each other. - The apparatus constructed for this work is different from other published dissolution flow cell apparatus in that it guarantees this condition, which is of fundamental importance in the analysis of multiparticulate dissolution kinetics. This is achieved by the "absolute sink arrangement" of the drug particles in the dissolution cell. The particles are placed in a single layer such that in principle no particle receives solvent that

has contacted other particles. The solvent thus contains no dissolved drug

that may influence dissolution. -

2. — It is also assumed that the dissolution rate parameter (k1, k2 or k₃) is constant and the same for all particles. - According to the theory on which the models are based this can only be achieved experimentally if the temperature and composition of the dissolution liquid is maintained constant and the flow rate is constant or uniform in the cross section of the dissolution cell where the particles are placed. The first two conditions

predominates, the apparent

are easily met. With regard to the flow rate the dissolution cell used has

a very useful feature, namely that the process can be stopped and the cell

rapidly disconnected allowing the particles to be inspected at any stage

of the dissolution process. Such inspections showed (after microscopic

measurements) uniform dissolution over the whole particle layer indicating

a uniform flow rate. The fact that the particles in the dissolution cell

can be inspected in this way makes it possible for the dissolution data to

be combined with particle size measurements.

3. ____ It is further assumed that the initial particle size (diameter)

distribution can be approximated by a truncated log-normal distribution

function. - Figure 7.2 shows that this is a good approximation for the 500

particles measured, however, it does not guarantee the correctness of the

assumption that this small sample represents the particle size distribution in the samples used for the dissolution tests, although the uniformity of the powder supported this assumption.

These investigations indicate that it is possible to describe mathematically, the dissolution of a multiparticulate system with a high degree of accuracy by considering both the particle size distribution effect and the particle shape effect discussed earlier. It is evident from the dissolution data obtained, that among the three models investigated, the cube root model describes the kinetics best.

It is possible that more complex and flexible models for single particle dissolution could describe the dissolution more adequately. The fact that the ss values for the cube root and the square root model are almost the same suggests a model with properties between these two. The Danckwerts model as discussed by Goyan (105) is given by:

$$-dw/dt = A((Dp)^{\frac{1}{2}} + D/a)C_{a}$$
 (7.12)

where

w =

D

= the surface area

= the diffusion coefficient

weight undissolved

= a quantity related to stirring p

= the radius of the particle a

C = the steady state concentration.

This model is very flexible. When (Dp) predominates, the apparent model would be the cube root model. As the quantity D/a becomes more important, then the square root model will become the apparent model. Finally, as the quantity D/a predominates the "squared cube root model" will become the apparent model. However, the Danckwerts model, when applied to the log-normal case, results in a mathematical expression which is much more complex than Eqs.

7.3 and 7.4.

The fit of the dissolution data to the cube root model is excellent. Therefore, if an application of the Danckwerts model results in an even better fit, this improvement will likely be statistically insignificant considering the magnitude of the experimental errors. In such a case the Hixson-Crowell model should be preferred because of its simplicity.

to a linear form¹ it is necessary to use a computer to cope with the complexity of computations. Several computer programs are available for the treatment of membrane regression (123-137). Considerable time and effort is often agent in applying the more powerful and vergetile of these programs and their complexity frequently inhibits the loss experienced. Even for the experiment user their inflexible and strict input structure may often lased to errors that can be difficult to find because of the lack of specific error passages. The time spent on correcting such errors is of considerable inconvenience, particularly when the program used is not written for inferentive time-there.

This chapter describes a powerful and versatile FORTRAN computer program. FURFIT, for nonlinear regression and curve fitting that does not have the disedvantages described above. It is written for the actious communicates of the user and utilizes the many advantages of interactive

The expression y = as is linear in the perspector a sud nonlinear in b. By transforming to in y = in a + built is linear in (in a) and b. The expression y = as + ce is nonlinear in b and d and cannot be transformed to a linear form by standard space.

CHAPTER 8

- FUNFIT - A TIME SHARING PROGRAM FOR GENERAL

NONLINEAR REGRESSION AND CURVEFITTING

Many investigators must deal with the problems of evaluating how well one or more mathematical models describe a certain physical system or of obtaining estimates of the parameters in a particular model. If the mathematical model is nonlinear in its parameters and cannot be transformed to a linear form¹ it is necessary to use a computer to cope with the complexity of computations. Several computer programs are available for the treatment of nonlinear regression (123-132). Considerable time and effort is often spent in applying the more powerful and versatile of these programs and their complexity frequently inhibits the less experienced. Even for the experienced user their inflexible and strict input structure may often lead to errors that can be difficult to find because of the lack of specific error messages. The time spent on correcting such errors is of considerable inconvenience, particularly when the program used is not written for interactive time-sharing.

This chapter describes a powerful and versatile FORTRAN computer program, FUNFIT, for nonlinear regression and curve fitting that does not have the disadvantages described above. It is written for the maximum convenience of the user and utilizes the many advantages of interactive

[P1.P2.P1.P1] is the dermoter vector and x the indervoter

^{1.} The expression $y = ae^{bx}$ is linear in the parameter a and nonlinear in b. By transforming to ln y = ln a + bx it is linear in (ln a) and b. The expression $y = ae^{bx} + ce^{dx}$ is nonlinear in b and d and cannot be transformed to a linear form by standard means.

time-sharing. The input occurs essentially as a communication with the computer. Questions are asked about which particular data treatment is desired and instructions are given how to enter data. Every input is extensively checked for numerical, logical and typing errors, so these can be corrected immediately. A special command 'BACK' makes it easy to edit previous inputs so multiple runs under various conditions can be made quickly. The program offers an extensive analysis of residuals that most other programs neglect, and a lattice search to obtain suitable initial parameter estimates. A lattice search combined with contour maps makes it possible to investigate whether a better solution of the nonlinear regression problem may exist.

THEORY rungtion to be firted could, in fact, have 2 independent whilehigh

The Central Limit Theorem of probability theory justifies the assumption that random errors in a set of observations are normally distributed. If the mathematical model is correct, the independent variable exact, and the errors are independent and normally distributed with zero mean and the same variance, then the method of least squares is the best choice for the estimation of the parameters because the estimates obtained will be maximum likelihood estimates.

For example the equation to be fitted to some observations might be:

$$y = p_1 e^{p_2 x} + p_3 e^{p_4 x} = f(\underline{p}, x)$$
 (8.1)

where $\underline{P} = \left[p_1, p_2, p_3, p_4 \right]^T$ is the parameter vector and x the independent variable. The problem is then to minimize the sum of the squared residuals, that is:

$$SS = \sum_{i=1}^{NOBS} e_i^2 = \sum_{i=1}^{NOBS} \left[y_i - f(\underline{P}, x_i) \right]^2$$
(8.2)

where $[(y_i, x_i), i = 1, 2, ..., NOBS]$ are the observations. There may also be reasons to weight the observations in a certain way. The weighted residual sum of squares is then defined as²

$$SS = \sum_{i=1}^{NOBS} e_{w'i}^{2} = \sum_{i=1}^{NOBS} w_{i} [y_{i} - f(\underline{p}, x_{i})]^{2}$$
(8.3)

where w_i is the weight of the i-th observation. Unweighted data are in fact data having the same weight i.e., $w_i = 1$. The relationship between weighted and unweighted residuals is

 $e_{w,i} = \sqrt{w_i} e_i$ (8.4)

The above example represents a simple case with only 1 independent variable, x. The function to be fitted could, in fact, have 2 independent variables i.e., $\underline{X} = \begin{bmatrix} x_1, x_2 \end{bmatrix}^T$. The observations are then $\begin{bmatrix} (y_1, x_{1,i}, x_{2,i}), i = 1, 2, ..., NOBS \end{bmatrix}$ (e.g. concentration, time temperature) and are represented in 3 dimensions. The problem is then no longer a least squares *curve*-fitting but a least squares *surface*-fitting. 'FUNFIT' can fit 'hypersurfaces' with up to 9 independent variables.

SIMULANTEOUS FITTING OF SEVERAL RESPONSE SYSTEMS

The system under investigation can sometimes be measured for more than one response or dependent variable, y. For example blood levels and urinary excretion of a drug could be measured in the same pharmacokinetic experiment or, in an experiment in chemical reaction kinetics more than one reaction product could be followed. In general, if there is a functional

In FUNFIT the weights are normalized i.e. scaled so that their sum is equal to the number of observations, NOBS. This is done to achieve a better comparison of weighted and unweighted sum-of-squares values.

expression for each kind of measured response and these expressions contain one or more *common* parameters it is likely that a simultaneous least squares fit of all the functions would provide more reliable parameter estimates than would be obtained by fitting each function individually. If, for example, 2 different kinds of response are measured in a system, the simultaneous least squares fitting problem is to minimize 'the sum of the sum of squared residuals' given by:

$$ss = \sum_{i=1}^{NOBS_{1}} w_{1,i} [y_{1,i} - f_{1}(\underline{P}_{1}, \underline{X}_{1,i})]^{2}$$

$$+ \sum_{i=1}^{NOBS_{2}} w_{2,i} [y_{2,i} - f_{2}(\underline{P}_{2}, \underline{X}_{2,i})]^{2}$$
(8.5)

where the parameter vectors \underline{P}_1 and \underline{P}_2 contain one of more common parameters. The symbols are used in analogy with equation 8.3 and subscripts 1 and 2 denote response systems 1 and 2. Hence, functions f_1 and f_2 are to be fitted simultaneously to NOBS₁ observations from response system 1 and NOBS₂ observations from response system 2 respectively. It is possible using 'FUNFIT' to fit simultaneously up to 10 functions (response systems) each containing up to 20 parameters and 9 independent variables, with or without weighting of the single observations in each response system. The sum-ofsquares function can therefore be summarized by a general extension of

ss =
$$\sum_{j=1}^{NFUNC} \sum_{i=1}^{NOBS_{j}} w_{j,i} [y_{j,i} - f_{j}(\underline{P}_{j}, \underline{X}_{j,i})]^{2}$$
 (8.6)

(8.7)

The total number of observations is

NOBS =
$$\sum_{j=1}^{NFUNC} NOBS_{j}$$

- 174 -

where NOBS, is the number of observations for the j-th function.

WEIGHTING OF RESPONSE SYSTEMS

The problem of weighting is of particular importance in simultaneous fitting for two reasons: 1. There may be a large difference in the orders of magnitude of the values of the dependent variables in each response system. 2. The variances of the errors may differ considerably between the systems.

signifies the weights of the j-th response system and w. .

The effect of condition 1. can be reduced by the proper choice of units for the dependent variables. Equation 8.6 assumes that the residual variance between each individual response system is the same. However this is seldom the case (i.e. condition 2. applies) and each response system must be given its own weight. It is generally accepted that, in a statistical sense, the best weighting scheme is to make the weight of each observation inversely proportional to the variance of the error (as estimated by the residual variance). Therefore the weight of each response system should be made proportional to the reciprocal of its residual variance which is obtained when the system is fitted individually by least squares (i.e. not simultaneously with other systems). Therefore the weight for the j-th response system should be:

$$W_{j} = \frac{NOBS_{j} - NPAR_{j}}{\left[\sum_{i=1}^{NOBS_{j}} W_{j,i} \left[Y_{n,i} - f_{j} \left(\frac{P}{j}, \frac{X}{j}, i\right)\right]^{2}\right]_{min}}$$
(8.8)

where NPAR, is the number of parameters in the function f_j and (NOBS - NPAR) j is the residual degrees of freedom. Equation 8.6 can therefore be written in an improved form:

- 175 -

$$SS = \sum_{j=1}^{NFUNC} W_{j} \sum_{i=1}^{NOBS} W_{j,i} [Y_{j,i} - f_{j}(\underline{P}_{j}, \underline{X}_{j,i})]^{2}$$
(8.9)

where W_j signifies the weights of the j-th response system and w_{j,i} the weights of the observations within that system. For example, if it is desired in a pharmacokinetic experiment to fit blood and urine data simultaneously, an individual fitting of each of the two systems should be done first. The weights to be used for each system in the simultaneous fitting are calculated as the residual degrees of freedom divided by the (weighted or unweighted) residual sum-of-squares obtained from the individual fittings.

Fitting implicit functions

It is necessary to apply a special approach if a function to be fitted is of implicit form; i.e., if the dependent variable, y, cannot be expressed explicitly as a function of the independent variable(s), \underline{X} and the parameters, \underline{P} .

The function may be described by:

$$q(y, X, P) = 0$$
 (8.10)

(e.g. $y^2 + x^{p1} + (x + y)^{p2} = 0$) from which y can be found by an iterative procedure only when numerical values of <u>X</u> and <u>P</u> are given. The calculated values of y are found by including in the user-supplied subroutine 'MODEL' a suitable algorithm for finding the root of g. This algorithm will serve the purpose of solving Eq. 8.10 for y so the subroutine indirectly defines the functional relationship between y and <u>X</u>, <u>P</u>.

MINIMIZATION

It should be noted that for a given set of observations the residual sum of squares, SS, depends only on the parameters because in Eq's

8.2, 8.6 and 8.9, the quantities y, X, w, W, NFUNC and NOBS are all numbers that are given. These equations can therefore be written in a shorter form as the importance of good (i.e. close) initial parameter estimated $SS = SS(\underline{P})$ (8.11) indicating that SS is a function of the parameters only. The least squares fitting and parameter estimation is then simply reduced to a function

minimization problem, namely to find the particular values of the elements

of the parameter vector, P, that minimize SS.

If the sum-of-squares function is strictly convex in a specified

convex parameter space then it will only have one minimum (133). This

condition is guaranteed when fitting linear but not nonlinear functions

for which there may exist more than one minimum. There is, in general, in

the latter case no guarantee that the minimum found by any nonlinear regression program is the smallest, giving the best possible fit and parameter estimates.

This problem can be illustrated in the case of two parameters where Eq. 8.11

describes a surface in 3 dimensions: Computer programs find the minimum by some iterative procedure and therefore require initial estimates of the parameters p_1 and p_2 together with their initial step sizes Δp_1 and Δp_2 . For example if the SS surface has two troughs and the initial starting point (p, p) lies on the slope of one of these and the same is the case for the points $(p_1 + \Delta p_1, p_2)$, $(p_1, p_2 + \Delta p_2)$ or $(p_1 + \Delta p_1, p_2 + \Delta p_2)$ then the strategy of the minimization algorithms employed is to 'run downhill' so that the parameter estimates in the next iteration step will usually be closer to the bottom of that trough while the other trough which may contain a smaller

minimum, is overlooked. Mathematically this problem exists also for sum-of-

a grid or lattice that spans over the whole parameter space. The computation

3. If the initial step sizes are not included in the input they are chosen in relation to the upper and lower limits given for the parameters in ters and divisions per parameter. For example a lattice FUNFIT.

squares surfaces in more than 3 dimensions (hyper surfaces) but is then impossible to visualize.

The importance of good (i.e. close) initial parameter estimates is obvious. The choice of small initial parameter step sizes will, as seen above, result in an increased probability of overlooking a smaller minimum. Large initial step sizes require more iterations and computation time but reduce this probability because a larger part of the parameter space is evaluated.

Most minimization algorithms in nonlinear regression programs are based on the gradient method of Gauss-Newton (134). These are particularly efficient in finding a minimum for a sum-of-squares function but are more likely to overlook a better minimum than non-gradient methods that usually take longer to converge. The simplex method of Nelder and Mead (135) used in 'FUNFIT' appears to be a particularly suitable non-gradient method. It has statistical application, is rapidly convergent and, on the way to the minimum, covers a large section of the parameter space, reducing the probability of missing a global (smallest) minimum. It has proven to be a very robust method that will always find a minimum and further has the advantage over gradient methods that nonlinear parameter constraints can more easily be employed. The economic disadvantage of the somewhat slower minimization procedure is insignificant in most cases.

LATTICE SEARCH

The most reliable way to search for a minimum smaller than that found by some minimization procedure is to evaluate SS (<u>P</u>) at each point in a grid or lattice that spans over the whole parameter space. The computation time for such a procedure increases exponentially and very rapidly with the number of parameters and divisions per parameter. For example a lattice

- 178 -

consisting of just 6 parameters with 10 divisions requires 10⁶ SSevaluations each of which requires NOBS function evaluations. If SS (P) is a surface in 3 dimensions (2 parameter case) its approximate shape can be visualized from a contour plot constructed from the SS-values of the points in a grid of the two parameters. If there are more than two parameters a composite picture of the sum-of-squares surface can be built up by "slicing" the parameter space, that is by fixing all except two parameters at a time. Contour plots help greatly in finding a global minimum and considerably reduce the number of function evaluations that otherwise would be required in such a search.

CONVERGENCE CRITERION

The accuracy with which the exact position of the minimum of SS (P) can be determined is always limited. After a certain number of iteration steps the SS-values of the points used in calculating the position of the next P estimate differ so little from each other that further iterations will introduce round-off errors. It is because of this and for reasons of economy, necessary to specify a stopping criterion for the minimization process. A necessary (but not sufficient) condition for P to be close to a minimum is that the SS-surface in the proximity of P is "sufficiently flat". The minimization process is often considered to have converged when the SS (P) value does not change more than a certain amount from one iteration step to the next. This guarantees a certain degree of 'flatness' as long as there is a significant change in the parameter vector P between the iterations. However such a stopping criterion evaluates the 'flatness' in a somewhat imprecise way and since SS (P) is not usually dimensionless, depends on the units used for the experimental data. The stopping criterion

- 179 -

used in 'FUNFIT' checks the 'flatness' in terms of the coefficient of

variation (percent) of the SS-values of some points closely surrounding P. For many applications it is important to get statistical estimated The flatness is in this way defined more strictly and does not depend on the units used. Results can therefore be compared on a more consistent basis information is found in the shape of the SS-surface near the minimum. The than in the former case. complexity of the SS(P) expression in the nonlinear case does not allow

The stopping criterion determines how 'accurately' the parameters exact statistical estimates to be obtained. These are obtained instead are determined. Its value should be chosen in relation to the variability

approximating the surface in the region of the minimum with a more simple

of the parameters. There is no reason to choose a very small stopping criterion surface for which exact. i.e. unbiased, statistical estimates can les

to get very 'accurate' parameters if their values are statistically rather

calculated. The SS (P) surface is approximated in the region of the final

uncertain. The variability of the parameters is related to the 'variability' parameter estimates, P, by a quadratic surface from a Taylor series of the SS-surface in the region of the minimum because a flat surface allows the parameters to have a wide range of values resulting in only a small variation of the SS (P) value. The stopping criterion used in 'FUNFIT' therefore has an advantage over other commonly used criteria because it is based on the 'variability' of the surface and hence determines the 'accuracy'

of the parameters in relation to their variability. To avoid termination on

a flat plateau of the SS-surface or near a 'saddle point' it must further be ensured that the surface around the point found at convergence is convex. In gradient methods this is often done by checking whether the matrix of the second partial derivatives (the Hessian matrix) is positive definite as is the case if all its eigenvalues are positive (133). In 'FUNFIT' the minimization process is allowed to continue a certain number of iteration steps after the 'flatness criterion' has been satisfied and the process is considered to have converged if the new parameter values and SS-values are

sufficiently close to the previous values.

STATISTICAL ESTIMATION

For many applications it is important to get statistical estimates of the variability of the parameters obtained. As indicated above, this information is found in the shape of the SS-surface near the minimum. The complexity of the SS(\underline{P}) expression in the nonlinear case does not allow exact statistical estimates to be obtained. These are obtained instead by approximating the surface in the region of the minimum with a more simple surface for which exact, i.e. unbiased, statistical estimates can be calculated. The SS (\underline{P}) surface is approximated in the region of the final parameter estimates, $\underline{\hat{P}}$, by a quadratic surface from a Taylor series expansion where third- and higher order terms are neglected; i.e.,

$$\mathrm{SS}(\underline{\mathrm{P}}) \approx \mathrm{SS}(\underline{\hat{\mathrm{P}}}) + \nabla^{\mathrm{T}} \mathrm{SS}(\underline{\hat{\mathrm{P}}}) (\underline{\mathrm{P}} - \underline{\hat{\mathrm{P}}}) + \frac{1}{2} (\mathrm{P} - \underline{\hat{\mathrm{P}}}) \nabla^{2} \mathrm{SS}(\underline{\hat{\mathrm{P}}}) (\mathrm{P} - \underline{\hat{\mathrm{P}}})$$
(8.12)

where $\nabla^{T} = (\partial/\partial p_1 + \partial/\partial p_2 + ... + \partial/\partial p_n)$ and n is the number of parameters. The statistical information is found in the term $\nabla^2 SS(\hat{P})$, that is the square matrix of the second partial derivatives of SS(P) evaluated at \hat{P} , the so-called Hessian matrix:

$$H(\hat{\underline{P}}) = \nabla^{2} ss(\hat{\underline{P}}) = \begin{bmatrix} \frac{\partial^{2} ss(\hat{\underline{P}})}{\partial p_{1}^{2}} \cdots \frac{\partial^{2} ss(\hat{\underline{P}})}{\partial p_{1} \partial p_{n}} \\ \frac{\partial^{2} ss(\hat{\underline{P}})}{\partial p_{1} \partial p_{n}} \cdots \frac{\partial^{2} ss(\hat{\underline{P}})}{\partial p_{1} \partial p_{n}} \end{bmatrix}$$
(8.13)

The sample variance-covariance matrix of the parameters, V, is the inverse of the Hessian matrix multiplied by the residual variance estimate:

$$V = \frac{SS(\underline{P})}{NOBS-n} H^{-1}(\underline{\hat{P}})$$
(8.14)

It can be seen from this equation that in order to decrease the variances of the parameters, the number of observations, NOBS, should be relatively large compared to the number of parameters, n. It is recommended that NOBS should not be less than about 3n in statistical estimations. The covariance between the i-th and the j-th parameter is the (i,j)-th element of the variance-covariance matrix:

$$\operatorname{cov}(\hat{\mathbf{p}}_{i}, \hat{\mathbf{p}}_{j}) = \mathbf{v}_{i,j}$$
(8.15)

and the variance of the i-th parameter is the i-th diagonal element:

$$var(\hat{p}_{i}) = COV(\hat{p}_{i}, \hat{p}_{i}) = v_{i,i}$$
 (8.16)

The elements of the correlation matrix, $\rho_{i,j}$, are calculated from the variance-covariance matrix by:

$$\rho_{i,j} = \frac{\operatorname{COV}(\hat{p}_{i}, \hat{p}_{j})}{\sqrt{\operatorname{Var}(\hat{p}_{i})} \operatorname{Var}(\hat{p}_{j})}$$
(8.17)

The joint probability distribution of the estimated parameters is multivariate normal and is given by (136).

$$g(\underline{P}) = \frac{1}{(2\pi)^{n/2} \sqrt{|v|}} \exp \left[-\frac{(\underline{P}-\underline{\hat{P}})^{T} v^{-1} (\underline{P}-\underline{\hat{P}})}{2}\right] (8.18)$$

ASSUMPTIONS MADE IN THE STATISTICAL EVALUATIONS

The reliability of the statistical estimates above depends on the following assumptions:

1. The independent variable(s) is without error.

2. The errors are independent, have zero mean and the same variance, σ^2 , i.e.:

$$cov(\varepsilon_{i},\varepsilon_{j}) = 0$$

$$i \neq j$$
(8.19)

$$E(\varepsilon_{i}) = 0 \tag{8.20}$$

$$Var(\varepsilon_i) = 0 \tag{8.21}$$

For the F- test, t- test and confidence limits to be valid the errors must furthermore be normally distributed with zero mean and the same variance:

$$\varepsilon_i \sim N(0,\sigma^2) \tag{8.22}$$

3. The mathematical model is correct.

4. Equation 8.12 is exact.

When conditions 1., 2. and 3. are satisfied the least squares estimate of <u>P</u> is also the maximum likelihood estimate of <u>P</u> because the likelihood function, L, can be written (137):

$$L(\underline{P},\sigma^{2}) = (2\pi\sigma^{2})^{-\frac{11}{2}} \exp[-SS(\underline{P})/2\sigma^{2}]$$
(8.23)

which is maximized when $SS(\underline{P})$ is minimized (σ^2 = constant). Therefore the least squares parameter estimates are unbiased under these 3 conditions.

Assumption 4. is only true when linear but not nonlinear functions are fitted. The most crucial point in nonlinear regression appears to be how good an approximation Eq. 8.12 is, because the statistical estimates (variances, covariances) are based on formulae and analysis from linear regression theory. The estimates are therefore biased and often called 'asymptotic estimates' because of the asymptotic property of the Taylor series expansion (Eq. 8.12). It has been found in simulation studies that the standard deviations of parameters in nonlinear regression can be two- to three-fold different from their true values (138). Parameter estimates are usually less biased because they are less sensitive to violation of the above conditions.

WEIGHTING

Where assumption 2. is not true because the error variance, σ^2 , is not constant, it is recommended that weights inversely proportional to

the individual variances be used, according to the following weighting scheme:

$$w_{i} = \frac{1}{\sigma_{i}^{2}}$$
, $i = 1, 2, ..., NOBS$ (8.24)

Thus if,

and

$$\varepsilon_i \sim N(o, \sigma_i^2)$$
 (8.25)

then the weighted errors will fulfil condition 2. because

$$\varepsilon_{w,i} = w \varepsilon_{i} = \frac{\varepsilon_{i}}{\sigma_{i}} \sim N(0,1)$$
(8.26)

so that this weighting will provide improved estimates. The error variances, σ_i^2 , can only be estimated by repeated experiments. However in some weighting situations a certain functional relationship is assumed between the error and the dependent or independent variable. For example in pharmacokinetics it is often assumed that the standard deviation of the errors is proportional to the plasma concentration, so the data are assigned weights inversely proportional to the square of the concentration. It is not valid statistically to use weights just to get a better fit. There must be a sound basis for the weighting scheme used.

CONFIDENCE REGIONS AND CONFIDENCE LIMITS

When assumptions 1. to 4. are true it can be shown that $SS(\underline{\hat{P}})$ is distributed as chi-square with (NOBS-n) degrees of freedom:

$$ss(\hat{\underline{P}}) \sim \sigma^2 \chi^2_{NOBS-n}$$
 (8.27)

$$ss(p) - ss(\hat{p}) \sim \sigma^2 \chi_p^2$$
 (8.28)

Both SS(\underline{P}) - SS($\underline{\hat{P}}$) and SS($\underline{\hat{P}}$) follow a χ^2 distribution and are independently distributed. The ratio

$$\frac{\left[SS\left(\underline{P}\right) - SS\left(\underline{\hat{P}}\right)\right]/n}{SS\left(\underline{\hat{P}}\right)\lambda NOBS-n)} \sim F(n, NOBS-n)$$
(8.29)

is therefore distributed as an F-distribution with n and (NOBS-n) degrees of freedom respectively (136). The term $SS(\underline{P})$ can be isolated from this equation so that the critical points of the F-distribution provide the exact (1- α) 100% confidence region of the parameters:

$$SS(\underline{P}) = SS(\underline{\hat{P}}) \left[1 + \frac{n}{NOBS-n} F_{\alpha,n,NOBS-n} \right]$$
(8.30)

The set of parameter values, \underline{P} , for which this equation is satisfied forms a closed contour line (two parameter case) or a contour surface inside which the probability of *simultaneously* finding the true (population) parameter values is $(1-\alpha)$ 100%. In the nonlinear case where assumption 4. is violated, the confidence contours are still exactly defined by Eq. 8.30 but the $(1-\alpha)$ 100% confidence level is only approximated.

When linear equations are fitted the confidence region is ellipsoidal in shape and given by:

$$(\underline{\mathbf{P}}-\underline{\hat{\mathbf{P}}})^{\mathrm{T}} \mathbf{H}^{-1}(\underline{\mathbf{P}}-\underline{\hat{\mathbf{P}}}) \leq \frac{\mathbf{n}}{\mathrm{NOBS-n}} \operatorname{SS}(\underline{\hat{\mathbf{P}}}) \mathbf{F}_{\alpha,n,\mathrm{NOBS-n}}$$
(8.31)

Because the parameters are correlated it is not possible to define confidence limits for each individual parameter. Limits are however often defined in terms of tangent points of 'support planes' i.e. planes parallel to the parameter coordinate axis and tangent to the ellipsoidal region defined by Eq. 8.31. The approximate $(1-\alpha)$ 100% 'support plane' confidence limits of the parameters are given by (123)

$$\hat{\rho}_{i} \pm SD_{i} \sqrt{F_{\alpha,n,NOBS-n} \times n}$$
(8.32)

where SD_i is the standard deviation of the i-th parameter.

- 185 -

Assuming no correlation among the parameters the 'uni-plane' confidence limits are given by (123)

$$\hat{p}_{i} \pm t_{\alpha, \text{NOBS-n}} \times SD_{i}$$
 (8.33)

where $t_{\alpha,NOBS-n}$ is the t-statistic with probability level α and (NOBS-n) degrees of freedom. Confidence limits are sometimes desired for the ratio of two parameters, for example p_1/p_2 . This can be calculated using the following expression (139):

$$\frac{\hat{p}_{1} \hat{p}_{2} - t_{\alpha,NOBS-n}^{2} \operatorname{cov}(\hat{p}_{1}, \hat{p}_{2}) \pm t_{\alpha,NOBS-n}^{A}}{\hat{p}_{2}^{2} - t_{\alpha,NOBS-n}^{2} \operatorname{Var}(\hat{p}_{2})}$$
(8.34)

where

$$A = \hat{p}_{1}^{2} \operatorname{Var}(\hat{p}_{2}) + \hat{p}_{2}^{2} \operatorname{Var}(\hat{p}_{1}) - 2 \hat{p}_{1} \hat{p}_{2} \operatorname{Cov}(\hat{p}_{1}, \hat{p}_{2})$$

$$- t_{\alpha, \text{NOBS-n}}^{2} \left[\operatorname{Var}(\hat{p}_{1}) - \frac{\operatorname{Cov}(\hat{p}_{1}, \hat{p}_{2})^{2}}{\operatorname{Var}(\hat{p}_{2})} \right]$$

$$(8.35)$$

STANDARD DEVIATION OF A FUNCTION OF THE PARAMETERS

Use of variance-covariance matrix

The variance-covariance matrix Eq. 8.14 that is printed out in most computer programs allows the standard deviation to be calculated for any quantity that is expressed as a function of one or more of the parameters. Let such a quantity be denoted

$$g = g(\hat{p}_1, \hat{p}_2, \dots, \hat{p}_k)$$
 (8.36)

then its standard deviation is given by the following formula which is based on a Taylor series expansion (140)

$$(SD_{g})^{2} \approx \sum_{i,j=1}^{K} \frac{\partial g}{\partial p_{i}} \frac{\partial g}{\partial p_{j}} COV(\hat{p}_{i}, \hat{p}_{j})$$
(8.37)

where the summation extends over all k^2 choices of the two indices i and j. The formula is exact where g is linear in the parameters, but is otherwise an approximation which is sufficiently accurate provided the coefficient of variation of the parameters does not exceed about 20%.

- 187 -

Three special cases are of interest: 1. If g is a function of one parameter p_i only, then Eq. 8.37 reduces to

$$SD_g \approx \left| \frac{dg}{dp_i} \right| SD_i$$
 (8.38)

In particular if g is linear and of the form ap + b (a and b are constants) then $SD_g = aSD_p$ exactly. 2. If g is a linear function of the form:

$$g = \sum_{i=1}^{k} a_{i} \hat{p}_{i}$$
(8.39)

then Eq. 8.37 becomes

$$(SD_g)^2 = \sum_{i=1}^k a_i^2 \operatorname{Var}(\hat{p}_i) + \sum_{i \neq j}^k a_i^a \operatorname{COV}(\hat{p}_i, \hat{p}_j)$$
 (8.40)

3. If g is a ration of two parameters, for example $g = \hat{p}_1 / \hat{p}_2$, Eq. 8.37 becomes

$$(SD_{g})^{2} \approx \left(\frac{\partial g}{\partial p_{1}}\right)^{2} Var(\hat{p}_{1}) + \left(\frac{\partial g}{\partial p_{2}}\right)^{2} Var(\hat{p}_{2}) + 2 \frac{\partial g}{\partial p_{1}} \frac{\partial g}{\partial p_{2}} Cov(\hat{p}_{1}, \hat{p}_{2})$$

$$(8.41)$$

which in this case gives

$$(SD_g)^2 \approx \frac{Var(\hat{p}_1)}{\hat{p}_2} + \frac{\hat{p}_1^2 Var(\hat{p}_2)}{\hat{p}_2} - \frac{2\hat{p}_1 COV(\hat{p}_1, \hat{p}_2)}{\hat{p}_2^3}$$
 (8.42)

Use of the transformation technique

Equation 8.37 often leads to rather complex expressions if g is a nonlinear function of more than a few parameters. However, this problem can be avoided in most cases and the standard deviation of g obtained directly without the truncation errors that may be introduced when Eq. 8.37 is used. This can be done if, as is usually the case, Eq. 8.36 can be solved explicitly or numerically for one of its parameters, so that this parameter, \hat{p}_i , can be expressed as a function of g and the remaining parameters, i.e.

$$\hat{p}_{i} = h(g, \hat{p}_{1}, \hat{p}_{2} \cdots , \hat{p}_{k-1})$$
(8.43)

The function to be fitted to the data points can then be transformed so that it contains g as a parameter essentially replacing p_i. To evaluate the function, p_i is then simply calculated using Eq. 8.43 (where g and the (k-1) parameters are among the k input parameters). Initial estimates must be obtained for g since it acts as an input parameter. This is done by calculating it from Eq. 8.36 using the initial estimates of the k parameters or by using the final least squares estimates obtained from curve-fitting based on the original untransformed function.

This transformation technique is of considerable importance and highly recommended. For example, it allows the standard deviation to be calculated for essentially any pharmacokinetic quantity that is expressed in terms of the macro- or microparameters in the model.

EXAMINATION OF RESIDUALS

It is of fundamental importance to analyse the pattern of the residuals because this is essentially the only way of examining whether the

basic assumptions behind the nonlinear estimation are violated. Most general nonlinear programs seem to ignore this. Perhaps the best way to examine the residuals is to plot them against the independent and dependent variables (141). Significant systematic deviations can be visualized in this way. The assessment is somewhat complicated by the fact that there will always be a correlation between the residuals because NOBS residuals are only associated with (NOBS-n) degrees of freedom. 'FUNFIT' includes the Durbin-Watson statistic, given by (142):

$$a = \frac{\sum_{i=2}^{NOBS} (e_i - e_{i-1})^2}{\sum_{i=1}^{NOBS} e_i^2}$$
(8.44)

to test for excessive serial correlation (systematic deviation) among the residuals. The statistic is compared with tabulated critical points at a given significance level.

The fundamental assumption of random errors is also tested using two nonparametric tests that will be called the 'group' statistic and the 'number' statistic. The 'group' statistic is based on an analysis of groups (runs) of residuals of equal sign. For example the sequence of residuals (+++) (--) (+) (--) (+) forms r = 5 groups. The least number of residuals with the same sign is L = 4. If NOBS residuals, with equal probability of being negative and positive, form a sequence with r groups, then the probability of getting \leq r groups is (143)

$$P(\leq r) = \binom{NOBS}{L} -1 \sum_{i=2}^{r} f_r$$
(8.45)

where $f_r = 2 \begin{pmatrix} L-1 \\ \frac{i}{2} - 1 \end{pmatrix} \begin{pmatrix} NOBS-L-1 \\ \frac{i}{2} - 1 \end{pmatrix}$ when i is even (8.46)

and
$$f_r = \begin{pmatrix} L-1 \\ \underline{i-1} \\ 2 \end{pmatrix} \begin{pmatrix} NOBS-L-1 \\ \underline{i-3} \\ 2 \end{pmatrix} + \begin{pmatrix} L-1 \\ \underline{i-3} \\ 2 \end{pmatrix} \begin{pmatrix} NOBS-L-1 \\ \underline{i-1} \\ 2 \end{pmatrix}$$
 (8.47)

when i is odd. The 'number' statistic is defined as the probability of getting L or less residuals of same sign and is given by

$$P(\leq L) = 2^{-NOBS} \sum_{i=1}^{L+1} {NOBS \choose i-1}$$
(8.48)

where the large brackets denote binomial coefficients i.e. $\binom{x}{y} = x!/(y!(x-y)!)$.

If $P(\leq r) < 0.05$ or $P(\leq L) < 0.05$ then the hypothesis that the residuals are random should be rejected (with α -error < 0.05). This can be used to reject the basic assumption of normal errors provided the model is correct. However, it can also mean that the mathematical model is wrong if it is assumed that the errors are in fact random with zero mean.

OUTLIERS

Several repeated experiments will be required to examine whether the errors are all normally distributed. When only one experiment is available a test must rely on the fact that if all the errors are deviates from the same normal distribution, they will then collectively be normally distributed. If the mathematical model is correct and it is assumed that $E(\varepsilon_i) = 0$ and $Var(\varepsilon_i) = \sigma^2$ then the residuals are unbiased estimates of the errors, i.e. $E(\varepsilon_i) = \varepsilon_i$ and the residual variance given by

$${}^{2} = \frac{\sum_{i=1}^{\text{NOBS}} (e_{i} - \overline{e})^{2}}{\text{NOBS-n}} \approx \frac{\sum_{i=1}^{\text{NOBS}} e_{i}^{2}}{\text{NOBS-n}} = \frac{\text{SS}(\hat{\underline{p}})}{\text{NOBS-n}}$$
(8.49)

estimates σ^2 . Further, if it is assumed that $\varepsilon_i \sim N(0, \sigma^2)$ it follows that $\varepsilon_i / \sigma \sim N(0, 1)$. Now, if the 4 basic assumptions mentioned previously are true then ε_i / s , the unit normal deviate form of the residuals, estimates ε_i / σ and is N(0, 1).

The normal deviate form of the residuals can be used to test for outliers, that is, data-points which, in a statistical sense, are not typical of the rest of the data. The i-th observation is an outlier if the normal deviate e_i/s falls in the critical region of the N(0,1) distribution, that is if $|e_i/s| > 1.96$ ($\alpha < 0.05$). If NOBS-n is smaller than 30 the normal deviate should be compared with the critical values of the t- distribution with (NOBS-n) degrees of freedom.

Outliers should be submitted to particularly careful examination since they may provide information of vital interest. They should only be rejected if they are caused by errors in the recording or the experimental technique (144). Outliers may also occur if the mathematical model is incorrect.

If there are many observations it may be useful to construct a half-normal plot of the residuals. The linearity of such a plot gives information about outliers and how normally the residuals are distributed (141). The above analysis extends also to weighted residuals.

COMPARISON OF PARAMETERS AND MODELS

theur to runrint =

It can be of interest to investigate whether there is a significant difference in a certain parameter, \hat{p}_i , between two experiments (1 and 2).

If it is assumed the parameter is normally distributed a t-test can then be applied.

$$\frac{\hat{p}_{i,1} - \hat{p}_{i,2}}{\sqrt{\operatorname{Var}(\hat{p}_{i,1}) + \operatorname{Var}(\hat{p}_{i,2})}}$$
(8.50)

If t falls in the critical region $|t| > t_{\alpha/2}$ of a t- distribution with (NOBS₁ + NOBS₂ - 2n) degrees of freedom, the null hypothesis that $\hat{p}_{i,1}$ and $\hat{p}_{i,2}$ are the same should be rejected at significance level α . USER - COMPUTER INTERACTION DURING EXECUTION

For any particular system there may be more than one mathematical model which could be appropriate. It is generally accepted that if two models give approximately "equal" fits, the simpler model should be chosen unless the more complex model can be justified on other criteria. At present there does not seem to be any simple, rigorously based statistical test which makes it possible to distinguish between alternative models. The following test, based on a comparison of residual variances, is no exception. It can however be used as a guide in the absence of other methods.

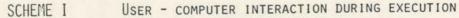
Let $SS(\hat{\underline{P}}_{(1)})/(NOBS_1-n_1) > SS(\hat{\underline{P}}_{(2)})/(NOBS_2-n_2)$ be the residual variances of models 1 and 2 respectively. If it is assumed $SS(\hat{\underline{P}}_{(1)})$ and $SS(\hat{\underline{P}}_{(2)})$ are independent random variables having chi-square distributions with $(NOBS_1-n_1)$ and $(NOBS_2-n_2)$ degrees of freedom, the variance ratio given by:

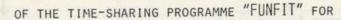
$$r = \frac{SS(\hat{P}(1))}{SS(\hat{P}(2))} \frac{NOBS_2 - n_2}{NOBS_1 - n_1}$$
(8.51)

follows an F- distribution with $(NOBS_1-n_1)$ and $(NOBS_2-n_2)$ degrees of freedom. Therefore, if $F > F_{\alpha,NOBS_1-n_1,NOBS_2-n_2}$, the null hypothesis that the residual variances are identical should be rejected. The hypothesis that the residual variance of model 2 is significantly smaller than that of model 1, (i.e. that model 2 fits the experimental data better than model 1) can therefore be accepted. If F does not exceed the critical value, further investigations are required or the simpler of the two models should be chosen.

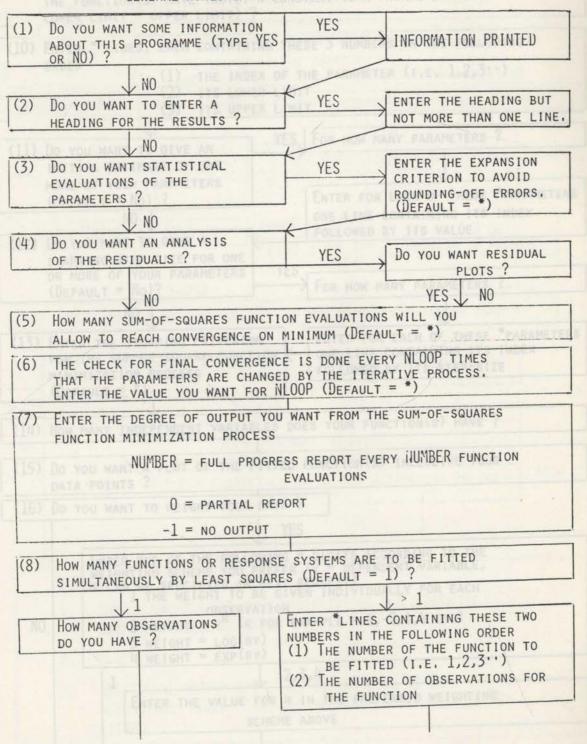
INPUT TO FUNFIT

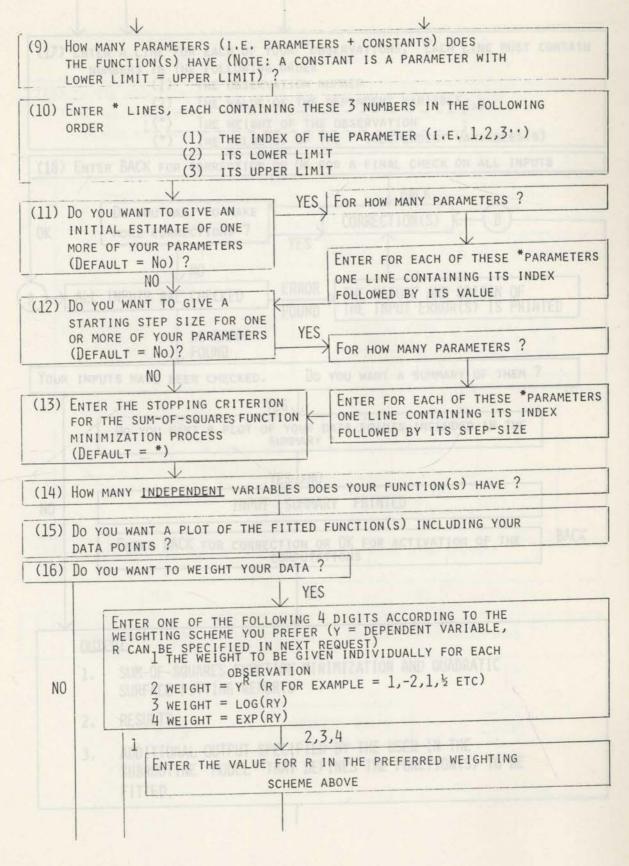
The input is provided in a communication between the user and the computer (scheme 8.1). Questions are asked about which data treatment is wanted and depending on the answers, the next questions are given in relation

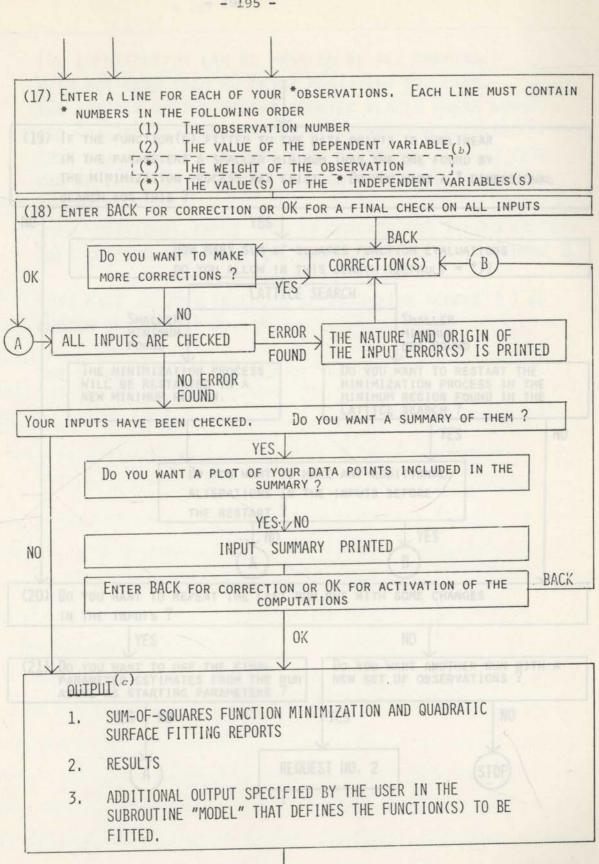




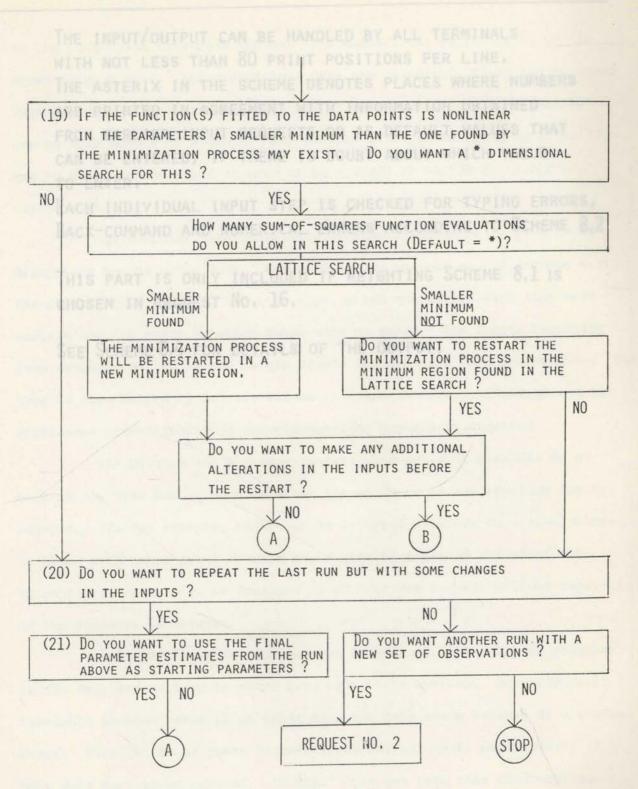
GENERAL NONLINEAR REGRESSION AND CURVE FITTING (a)







- 195 -



(a)

(b)

(c)

The input/output can be handled by all terminals with not less than 80 print positions per line. The asterix in the scheme denotes places where numbers are printed in agreement with information obtained from earlier input requests or as default values that can be entered, if there is doubt about which value to enter. Each individual input step is checked for typing errors, Back-command and numerical errors according to Scheme 8.2

THIS PART IS ONLY INCLUDED IF WEIGHTING SCHEME 8.1 IS CHOSEN IN REQUEST No. 16.

user is also helped by default values in requests no.3,5,6,8,12,13 and 19

SEE SCHEME 8.4 FOR DETAILS OF THE OUTPUT.

where some miderstanding of the minimization process is required.

The program has a unique feature that makes it possible to go

request. If, for example, BACM or B4 is typed anywhere on a line, alone or along with other data, request no. 4 will be repeated and after the appropriate input, will be followed by the current request without repetition of the requests in between. Possibly the most inconvenient feature of most computer programs is the very strict form in which data have to be entered. Execution will terminate in most cases if an input does not make sense because of a typing error. This can be of great inconvenience for the user, particularly if many data have to be entered. 'FUNFIT' does not have this disadvantage.

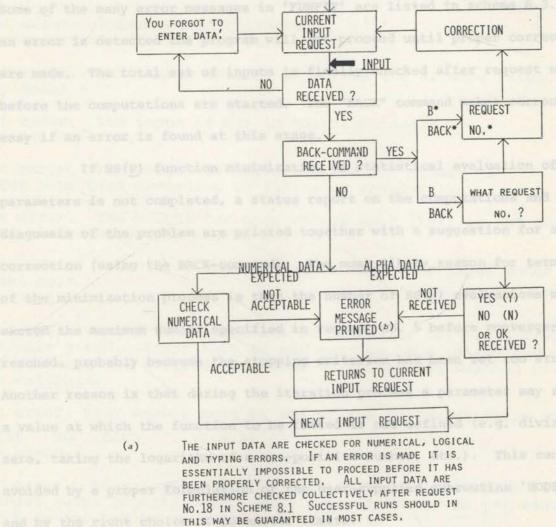
It will not terminate for any kind of typing error. Instead every input is checked in several ways (scheme 8.2). Typing errors, logical errors and numerical errors are all detected. It is, for example, not uncommon to to the particular treatment chosen. The answers can be YES or NO or, in short form, Y or N. Where numerical data are requested, these can be typed anywhere on a line in integer, decimal or exponential form separated by one or more blanks or a comma. For example 17.0 can be typed as 17 or 17. and -0.00153 can be typed as such or as -.00153 or -1.53E-3, -1.53E-03, -153E-5, etc.

The computer uses previously entered information to give exact directions how to enter data. For example, in request no. 17 (scheme 8.1), the user is told how many lines to type, which quantities each line must contain and the order in which these must be typed. The inputs resulting from requests no. 8, 14 and 16 are stored and used for this instruction. The user is also helped by default values in requests no.3,5,6,8,12,13 and 19 where some understanding of the minimization process is required.

The program has a unique feature that makes it possible to go back at any time and correct or alter the input under any previous input request. If, for example, BACK4 or B4 is typed anywhere on a line, alone or along with other data, request no. 4 will be repeated and after the appropriate input, will be followed by the current request without repetition of the requests in between.

Possibly the most inconvenient feature of most computer programs is the very strict form in which data have to be entered. Execution will terminate in most cases if an input does not make sense because of a typing error. This can be of great inconvenience for the user, particularly if many data have to be entered. 'FUNFIT' does not have this disadvantage. It will not terminate for any kind of typing error. Instead every input is checked in several ways (scheme 8.2). Typing errors, logical errors and numerical errors are all detected. It is, for example, not uncommon to SCHEME 8.2

STRUCTURE OF THE DATA CHECK PERFORMED AT EACH INDIVIDUAL INPUT STEP IN SCHEME 8.1 (a)



(b)

SCHEME 8.3 GIVES EXAMPLES OF SOME OF THE MANY INPUT ERROR MESSAGES IN FUNFIT. type a comma in place of a decimal point or an "O" instead of a zero; such errors are clearly indicated by the program and the last input line is repeated for immediate correction. An error message is also printed so corrections can be made if an input does not agree with an earlier input. Some of the many error messages in 'FUNFIT' are listed in scheme 8.3. If an error is detected the program will not proceed until proper corrections are made. The total set of inputs is finally checked after request no. 18 before the computations are started. The "BACK" command makes corrections easy if an error is found at this stage.

If SS(P) function minimization or statistical evaluation of the parameters is not completed, a status report on the computations and a diagnosis of the problem are printed together with a suggestion for a suitable correction (using the BACK-command). The most likely reason for termination of the minimization process is that the number of SS(P) evaluations may exceed the maximum number specified in request no. 5 before convergence is reached, probably because the stopping criterion has been set too strictly. Another reason is that during the iteration process a parameter may reach a value at which the function to be fitted is not defined (e.g. division by zero, taking the logarithm of a non-positive number, etc.). This can be avoided by a proper formulation of the user-supplied subroutine 'MODEL' and by the right choice of parameter limits.

At convergence, the point in n-dimensional parameter space defined by $\hat{\underline{P}}$ is surrounded by other points which form a polygon with $\hat{\underline{P}}$ as centroid. The size of this polygon depends on the accuracy with which the SS(\underline{P}) minimum is determined and diminishes for small values of the stopping criterion. The SS-values at the vertices, the centroid and mid-points of the sides of the polygon provide the surface points used in the quadratic surface fitting for estimating the variability of the parameters (135). Because of the small

- 200 -

SCHEME 8.3 SOME INPUT ERROR MESSAGES IN FUNFIT (a)

Input error, too many numbers on line above.* are expected. Reenter last line in correct form according to request. Input error, too few numbers on line above.* are expected. Reenter last line in correct form according to request.) (b) Input error. Unrecognized character (Reenter last line in correct form. Input error. The number entered is not in the allowed range. - Try again -Input error. The input is not in the right order. - Try again -Input error. You forgot to enter observation number *. Reenter your data according to following request: Input error. The weight for observation no.* (Y=*) is not defined by the chosen weighting scheme no.* Input error. Your input does not agree with your input under request no.* - Try again -Input error. The initial estimate of parameter no.* is not within the limits given. Enter BACK if you want to make correction or OK if you want the program to choose an acceptable initial parameter estimate. Input error, your answer must be one of the following three only YES NO BACK (Y, N, B)- Try again -Input error. Your input under request no.* does not agree with the input under request no.*. What request number do you want to go back to for making corrections? Input error. You forgot to enter data. - Try again -

Input error. The weight of the dependent variable is negative. Reenter last line in correct form.

(a) The appropriate numbers are printed where * appears in the text.

(b) The mistyped symbol is printed in the bracket. This feature prevents termination of the program when for example common errors such as typing ',' instead of '.' or the letter O instead of zero are committed. size of the polygon at convergence these points will be so close together that numerical rounding-off errors can be significant in the fitting procedure. To prevent this, the polygon is expanded around the centroid before the quadratic surface is fitted until the SS-values at the ventices exceed that at the centroid by more than a given value of the expansion criterion (request no. 3). Two tests are then done to ensure that the statistical evaluation of the parameters is not significantly subject to numerical errors: The minimum of the fitted surface is compared with the minimum $SS(\hat{P})$ found by the minimization process to check for round-off errors; also, the surface points used in the fitting procedure must lie in a convex region of the SS(P) surface in order to be close to the true minimum and this is checked by evaluating whether the Hessian matrix (Eq. 8.13) is positive definite.

It is however most unlikely that there will be any computational difficulties either in the minimization process or in the statistical evaluation of the parameters if the recommended (default) values are chosen in requests no. 3,5 and 13.

Lower and upper parameter limits are specified in request no. 10 to prevent convergence on an SS(P) minimum that may give unrealistic parameter estimates. If an initial estimate is not given for a parameter in request no. 11 the mid-point of the range defined by its lower and upper limits is chosen. Default values for the initial step sizes (request no. 12) are chosen in relation to the initial parameter estimates and their lower and upper limits so that a large part of the parameter space is searched in the SS(P) minimization process and the chance of overlooking a smaller minimum is reduced.

Requests no. 11 and 12 also provide the option of choosing one or more initial parameter estimates and their respective step sizes independently of their lower and upper limits. This option can be useful if it is desirable to increase the chance of finding a minimum that gives final estimates close to the initial estimates instead of finding another even smaller minimum that may exist but would give less realistic parameter estimates.

By entering 1 in request no. 5 a lattice search can be made before the start of a run. The coordinates of the point in the lattice which has the smallest $SS(\underline{P})$ value can then be chosen as the initial estimates of the parameters, and the run can be started after input no. 5 has been modified (using a B5 command). This 'pre-search' for the best starting values increases the chance of finding the global minimum.

APPLICATIONS OF THE 'BACK' COMMAND

The B** command is not only useful for correcting input errors but has wider application. For example, if the analysis of the residuals suggests a particular weighting scheme, a new run with weighted data can easily be made. By typing Bl6, request no. 16 is repeated and the desired weighting scheme can be chosen (e.g. R = -2 will give weights proportional to y^{-2}). The 'communication' then continues from the current request and a new run, this time with weighted data, will be started by typing Bl8 followed by OK and N in succession. A new run in which one or more of the parameters is kept constant can, in a similar way, be made by repeating request no. 10 and fixing the parameters in question by setting upper limit=lower limit=the fixed value.

The lattice search that can be chosen in request no. 19 includes, in the case of two parameters (2-dimensional search), a contour map of the SS(P) surface, together with matrices of the $SS(\underline{P})$ values and the sign of the partial derivatives of $SS(\underline{P})$ with respect to each of the two parameters at the points in the search grid. In the case of only one parameter (all others fixed) a plot will be printed of $SS(\underline{P})$ versus the parameter.

The B** command is particularly useful in connection with a lattice search involving more than 2 parameters if a composite picture of the SS(P) surface is desired. This is done as follows: The command BlO is typed in response to request no. 19 (or after a lattice search) and all but two parameters are fixed. Then after the following 2-dimensional search, Bl9 is typed, followed again by BlO. This procedure can be repeated as many times as desired, each time 'slicing' the parameter space in a different way.

If two or more response systems are fitted simultaneously the data for these are stacked and numbered in succession (in request no. 17). Therefore, a new run where only the first response system is fitted individually can easily be made by typing B8 followed by 1, B18 and N in succession. If the first run was a simultaneous fitting of a pharmacokinetic model using both blood and urine data the effect of including or excluding the urine response system can quickly be evaluated.

Every time the B** command is used the computer stores the current request position and returns to this immediately after correction of a previous input. It is however, possible to erase this memory and continue without returning from a previous request by typing the same B** command twice in succession. This procedure is useful if, for example, the first 7 inputs for a new run with different observations are identical to those of a previous run. These inputs are then taken to be the same and need not be repeated for the new run if B8 is typed twice in succession after completion of the previous run.

- 204 -

Scheme 8.4 summarizes the maximum output possible from 'FUNFIT' for a single run. The sections printed in italics denote the standard (minimum) output that is always printed (or displayed on a terminal screen). The degree of output will range from this to the maximum output according to the user's specifications.

The optional <u>input summary</u> is useful as a check for numerical errors before activation of the computations or may serve as an extra copy of the experimental data. A <u>minimization report</u> should be chosen if the user-supplied subroutine 'MODEL' specifies parameter constraints that may cause convergence problems. The report is also of value because it provides a table of SS(<u>P</u>) values and parameter values that gives some information about the sum-of-squares surface and how large a section of the parameter space has been searched on the way to the minimum.

The <u>quadratic surface fitting report</u> is only printed in conjunction with the minimization report when statistical evaluation of the parameters has been requested (no. 3). The variance-covariance and correlation matrices of the parameters printed in this section will also appear in the general result section. The plots printed in the <u>result section</u> are line plots and are therefore of low accuracy but provide the essential information. The exact coordinates of the points in the plots are tabulated so precise plots can be produced manually if desired.

In fitting pharmacokinetic models it is often necessary to calculate from the final parameter estimates, quantities such as half-lives, clearances and volumes of distribution or to make a plot of the amount of drug in a peripheral compartment versus time. It will be demonstrated in the following section how the subroutine 'MODEL' which describes the equation(s) to be - 206 -

SCHEME 8.4 SUMMARY OF OUTPUT FROM 'FUNFIT' (a)

INPUT SUMMARY (b)

- Heading for problem. - Number of: variable parameters, constant parameters, observations and independent variables for each response system. Table of parameters with lower and upper limits, initial estimates and step sizes. - Table of observations with dependent variable(s), independent variable(s), weight (if any) and normalized weight. - Plot of dependent variable(s) <u>versus</u> independent variable(s). - Weighting scheme used, expansion and stopping criterion. -

SUM-OF-SQUARES FUNCTION MINIMIZATION REPORT (c)

- Table of evaluation no., sum-of-squares values and parameters.sum-of-square value and parameter estimates at convergence and number of evaluations used to reach convergence.-

FITTING OF QUADRATIC SURFACE IN REGION OF MINIMUM (c)

- Minimum of quadratic surface and parameter values at minimum.-Generalised inverse of information matrix (H^{-1}) . - Information matrix (H). - Correlation matrix (Eq.8.17).- Number of evaluations used in the fitting.-

function values yersus the parameter in its specified renge

RESULTS of coordinates of points in the plot. - Sum-of-Equares and

- Heading for problem. - Table of parameters with their lower and upper limits, initial and final estimates. - Table of standard deviation, coefficient of variation and 95% confidence limits of the parameters. - Graphical illustration of the relative position of

the calculated parameters in their specified range .- Residual sum of squares, regression sum of squares, sum of squared response, mean of response, residual mean square, regression mean square, mean of residuals and correlation coefficient .- Weighted residual sum of squares, weighted residual mean squares, mean of weighted residuals. - Table of dependent variable(s), independent variable(s), observed and calculated responses, difference in response, differences expressed as percentages and as normal deviates .- Table of weights, normalized weights, weighted residuals and normal deviate form of weighted residuals .- Plot of calculated (fitted) curve including experimental data points. Table of coordinates in plot .-Analysis of variance table. - Durbin-Watson statistics for serial correlation of residuals. 'Run test' and 'number test' for randomness of residuals .- Residual plots: residuals versus dependent and independent variables. Weighted residuals versus dependent and independent variables .- Variance-covariance matrix .- Correlation matrix .- OUTPUT SPECIFIED IN THE USER-SUPPLIED SUBROUTINE 'MODEL' .-

LATTICE SEARCH FOR GLOBAL MINIMUM (d)

The output in this section depends on the number of <u>variable</u> parameters as follows: (1 PARAMETER) - Plot of sum-of-squares function values <u>versus</u> the parameter in its specified range.-Table of coordinates of points in the plot.- Sum-of-squares and parameter values at minimum found in the unidimensional search.-(2 PARAMETERS) - Residual sum-of-squares matrix.

Matrices of the sign of the partial derivatives of
the sum-of-squares function with respect to each parameter.
Sum-of-squares contour map. - Parameter values at the grid point

that has the smallest sum-of-squares value.- (> 2 PARAMETERS).-Total number of lattice points evaluated. Number of divisions of each parameter interval.- Parameter values of the grid point with the smallest sum-of-squares value.

- (a) The scheme summarizes the maximum degree of output possible from FUNFIT for a single run. The sections printed in italics denote the standard, minimum output that is always printed. The degree of output will vary in this range according to the user's specifications.
- (b) The input summary can be chosen to check all the inputs before the activation of the computations or may serve as an extra copy of the experimental data.
- (c) It is useful to choose a minimization report if special parameter constraints have been specified in the user supplied subroutine 'MODEL' that specifies the function(s) to be fitted.
- (d) The lattice search can be used to evaluate the contour of the sum-of-squares surface when there are only two parameters. A composite picture of the surface can be built up if there are more than two by fixing all but two parameters at a time (see text).

fitted, can include a special section containing such additional calculations and plots. This 'user-supplied output' will then be printed as the last part of the results section. The program includes an easily applicable subroutine, 'PLOT', which enables the user to make special plots in this section.

DEFINING THE EQUATION(S) TO BE FITTED

In order to use FUNFIT the function(s) to be fitted must first be defined by the user in a special subroutine called MODEL. The structure of this subroutine is:

SUBROUTINE MODEL (Y, X, P, IPRINT)

where Y denotes the dependent variable(s); X the independent variable(s); P the parameter vector and IPRINT is an integer variable which is controlled by FUNFIT and is used if additional output is desired. The subroutine must define Y as a function of X and P.

The structure can best be illustrated by some examples, 1. A single Equation, one Dependent and Independent Variable: To fit the two-exponential equation:

$$= p_1 e^{-p_2 t} + p_3 e^{-p_4 t}$$

(8.52)

the subroutine can be written simply as:

the coordinates C

SUBROUTINE MODEL (C,T,P,IPRINT)

DIMENSION P(4)

C = P(1) * EXP(-P(2) * T) + P(3) * EXP(-P(4) * T)

RETURN

END

It is recommended to use WPOINTE-75 and MLINES-50

Special Output

If it is desired to calculate a quantity, say:

$$A = p_1/p_2 + p_3/p_4$$
(8.53)

from the final least squares parameter estimates this can be done by the computer by including the following special output section in the subroutine:

IF (IPRINT .EQ. \emptyset) RETURN A = P(1)/P(2) + P(3)/P(4) WRITE (6,1)A

1 FORMAT(" A=", E12.6)

just before the RETURN statement. It is seen that the IPRINT parameter controls the execution of the special output section. This section is printed *once* just after the general output section described in scheme 8.4.

An easily applicable plotting routine is available for special plots the user may wish to make in the special output section. Its structure is

SUBROUTINE PLOT (XARRY, YARRY, NPOINS, NLINES)

where XARRAY and YARRAY are two arrays of dimension NPOINTS which contain the coordinates to the points to be plotted using NLINES of the output device.³

In calculating the points for such plots it is frequently convenient to know the interval over which the observations are taken, i.e. the smallest and largest X value. This information can be made available by including the common statement:

COMMON XMIN, XMAX

3. It is recommended to use NPOINTS=75 and NLINES=50.

The following subroutine illustrates the use of PLOT in terms of a plot (75 points) of the function:

$$f_2 = |e^{-P_4 t} - e^{-P_2 t}|$$
 (8.54)

over the range of the t values, i.e. from XMIN to XMAX:

SUBROUTINE MODEL (C,T,P,IPRINT)

DIMENSION P(4), F2(75), TT(75)

COMMON XMIN, XMAX

C=P(1)*EXP(-P(2)*T) + P(3)*EXP(-P(4)*T)

IF (IPRINT.EQ.Ø) RETURN

DO 1 I = 1,75

TT(I) = XMIN + (I-1) * XMAX/74.

1 F2(I) = ABS(EXP(-P(4)**TT(I)) - EXP(-P(2)*TT(I)))
CALL PLOT (TT,F2,75,50)

RETURN

and added to the END evicus equared meridual is the process of calculating

Fitting Several Functions Simultaneously

Example

The general structure of the user supplied subroutine MODEL when N response systems or equations are to be fitted simultaneously can be illustrated schematically as follows:

SUBROUTINE MODEL (Y,X,P,IPRINT)

DIMENSION P()

COMMON/FUNNUM/ITHFUN

GO TO (1,2,...,N) ITHFUN

l Y = (1st function)

RETURN

It is desired to estimate p, and p, from the set of chaervations of (y, t)

2 Y = (2nd function) RETURN

N Y = (Nth function) RETURN

END

The basis of this structure can be explained in the following way: During the input procedure each observation point is numbered consecutively and automatically assigned a label ITHFUN (<u>i-th fun</u>ction) indicating which response system it belongs to. The user supplied subroutine MODEL is called by FUNFIT for each observation point. The parameter ITHFUN that is introduced by the COMMON block named FUNNUM (<u>function number</u>) transfer control to the i-th function which is then calculated. Control then returns to FUNFIT where the squared residual at that particular point is calculated, and added to the previous squared residual in the process of calculating the sum of squared residuals.

Example ______ the second to the second sector velue in the input, or it can

Consider the following linear compartmental system:

$$\begin{bmatrix} \mathbf{y}_1 \end{bmatrix} \xrightarrow{\mathbf{p}_1} \begin{bmatrix} \mathbf{y}_2 \end{bmatrix} \xrightarrow{\mathbf{p}_2} \xrightarrow$$

where y_1 and y_2 , which are measured at various times t, are given by:

$$dy_1/dt = -p_1y_1$$
 (8.55)

$$dy_2/dt = p_1 y_1 - p_2 y_2$$
(8.56)

which (for $y_1 = y_0$ and $y_2 = 0$ at t=0) integrates to:

$$y_1 = y_0 EXP(-p_1t)$$
 (8.57)

$$y_{2} = \frac{y_{0}p_{1}}{p_{2}^{-p_{1}}} (EXP(-p_{1}t) - EXP(-p_{2}t))$$
(8.58)

It is desired to estimate p_1 and p_2 from the set of observations of (y_1,t) and (y_2,t) . This can be done by fitting simultaneously 8.57 and 8.58 to the observations.

These equations can conveniently be defined as follows:

SUBROUTINE MODEL (Y,T,P,IPRINT)

DIMENSION P(3)

COMMON/FUNNUM/J

GO TO (1,2) J

1 Y = P(3) * EXP(-P(1) * T)

RETURN

Y = P(3)*P(1)/(P(2)-P(1))*(EXP(-P(1)*T)-EXP(-P(2)*T))RETURN

END

The quantity y_0 can either be defined as a constant by setting its upper and lower limits equal to the same (constant) value in the input, or it can be defined as a parameter that is to be estimated by assigning appropriate bounds for P(3).

Fitting an Equation with 2 Independent Variables, a Simple Example

Consider a first-order reaction:

where the fraction remaining, f_A , of A is measured at various times, t, at various temperatures T (⁰K). A simplified model for this reaction may be written as:

$$f_{A} = EXP (-p_{1} t EXP (-p_{2}/T))$$
 (8.59)

where p_1 and p_2 are parameters to be determined from a set of (f_A, t, T) data points. This model can be defined in the following way for FUNFIT:

where X(1) acts as the time, t, and X(2) as the absolute temperature $({}^{0}K)$.

Fitting of Implicit Equations, A Simple Example

It is, in the following equation:

$$p_1 x = \ln y + p_2 y$$
 $x, y, p_1, p_2 > 0$ (8.60)

not possible to express y explicitly as a function of the independent variable x so a special technique must be used to express and fit the function:

$$y = f(x, p_1, p_2)$$
 (8.61)

The numerical problem of defining 8.60 in the form of 8.61 is the same as finding the root of the equation:

 $p_1 x - \ln y - p_2 y = 0$ (8.62)

for given values of x, p_1 and p_2 ; where $p_{1,2}$ are the parameters to be determined from a set of (x,y) observations. Let the left-hand side of 8.62 be denoted g(y) then y can be found by the Newton-Raphson iteration:

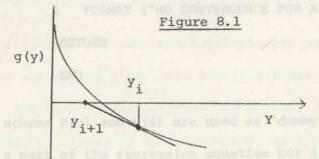
$$y_{i+1} = y_i - \frac{g(y_i)}{g'(y_i)}$$
 (8.63)

$$g'(y) = \frac{\partial g(y)}{\partial y} = -\frac{1}{y} - p_2$$
 (8.64)

where

i.e. 8.63 becomes: $y_{i+1} = y_i + \frac{p_1 x - \ln y_i - p_2 y_i}{\frac{1}{y_i} + p_2}$ (8.65)

Since y and $p_2 > 0$ then g'(y) < 0 so the function g(y) will be monotonously decreasing:



In the iteration 8.65 y_{i+1} may become non-positive for which g(y) will not be defined in the next iteration. To prevent this to occur it is convenient to define:

$$y_{i+1} = y_i/2$$
 if $y_{i+1} \le 0$ (8.66)

The iteration 8.65 may be considered to have converged when the relative change in y between iterations is less than 0.0001% i.e. when $|\langle y_{i+1} - y_i \rangle / y_i| < 10^{-6}$.

The subroutine to define 8.61 (i.e. 8.60) may thus be defined in the following way:

SUBROUTINE MODEL (Y, X, P, IPRINT)

DIMENSION P(4)

C P(3) = CONVERGENCE CRITERION

C P(4) = MAX.NO.OF ITERATIONS

MAX = IFIX (P(4))

DO 1 I = 1, MAX

YSAVE = Y

Y = Y + (P(1) *X-ALOG(Y) - P(2) *Y) / (1/Y+P(2))

IF(Y.LE.Ø.)Y = YSAVE/2.

IF (ABS ((Y-YSAVE)/YSAVE).LT.P(3)) RETURN

1 CONTINUE

WRITE(6,2)X,Y,P

2 FORMAT ("NO CONVERGENCE FOR X,Y,P =", 6E12.6)

RETURN

In this scheme P(3) and P(4) are used as "dummy" constant parameters which are not a part of the regression equation but included of computational reasons.

user is choosing a mitable value for the step size used in the finite difference approximation of derivatives. The default value specified for this step size is a program may unfortunately apply successfully only is a limited number of eases. Such practical experiences are illustrated in this paper ip the application of the program MONLER (123) which is based on Hartley's modification of the Gauss-Newton algorithm (148).

Such problems are eliminated in TURFT, This program has implemented the adaptive simples method of Belder and Mend (135,50) which is a progradient method that does not require evaluation of derivatives. This method is less efficient then Gauss-Newton based methods but considerably more robust and reliable. It will sever fail even under extreme conditions where the gradient methods may be unstable due to ment alegulatity and ill-conditioning of the matrices used in the iteration extreme.

Humarical techniques in nonlinear parameter estimation have been reviewed by Chambers (149) and an excellent discussion has been given by Dennis (146).

CHAPTER 9

CURVE FITTING AND MODELLING IN PHARMACOKINETICS

- 217 -

A Comparison of FUNFIT and NONLIN computer programs.

Several programs are available for nonlinear least squares parameter estimation (123-128). Nearly all are based on the Gauss-Newton or other related gradient methods since these are usually rapidly convergent and provide estimates of the variance-covariance matrix. However, such gradient methods may fail when the residuals are large (145,146), as is often the case in fitting equations to biological data, and they may converge on a non-stationary point (147) if great care is not taken by the user in choosing a suitable value for the step size used in the finite difference approximation of derivatives. The default value specified for this step size in a program may unfortunately apply successfully only in a limited number of cases. Such practical experiences are illustrated in this paper in the application of the program NONLIN (123) which is based on Hartley's modification of the Gauss-Newton algorithm (148).¹

Such problems are eliminated in FUNFIT. This program has implemented the adaptive simplex method of Nelder and Mead (135,150) which is a nongradient method that does not require evaluation of derivatives. This method is less efficient than Gauss-Newton based methods but considerably more robust and reliable. It will never fail even under extreme conditions where the gradient methods may be unstable due to near singularity and ill-conditioning of the matrices used in the iteration procedure.

Numerical techniques in nonlinear parameter estimation have been reviewed by Chambers (149) and an excellent discussion has been given by Dennis (146).

The possibility of multiple solutions (multiple sums of squares minima) undoubtedly represents the greatest problem in nonlinear estimation. The problem is expected to be particularly pronounced in pharmacokinetic studies, because these often involve the fitting of multiexponential equations to rather variable biological data, and because the ratio of number of data points to number of parameters is often quite small. Parameter estimation under such conditions may produce spurious results and discrimination between pharmacokinetic models may be very difficult and unreliable. The problem can be reduced, but usually not eliminated, if a graphical or numerical method is available which provides good initial parameter estimates for the iteration procedure. This is seldom the case for models describing nonlinear pharmacokinetics. The best approach should therefore be to use an algorithm which is effective in finding the (statistically) best solution in terms of the smallest residual sum-ofsquares value.

It is generally accepted that the nongradient search methods perform better than the gradient methods in this respect. In particular, the adaptive simplex method used in FUNFIT appears to be very suitable because of its unique minimization method.

Regardless of the choice of algorithm the question of which starting values the parameters should be given still seems to be the greatest practical problem the user faces in nonlinear estimation. Frequently, when no preliminary estimation technique is available the initial values are simply guesses which all too often produce unacceptable results in the first run. However, by studying these results, corrections can often be made so that acceptable results can be obtained in subsequent

runs.

- 218 -

Interactive programs are most convenient in such cases. In

particular, FUNFIT has been designed so that it allows a highly interactive

and flexible editing of input at any stage.

Pharmacokinetic applications of FUNFIT

1. The classical linear, compartmental models are still the most often used

models in pharmacokinetic studies. The evaluation of such models is

well documented and has become a routine procedure in many investigations.
2. There has, however, been an increasing awareness that linear models cannot adequately describe certain drug disposition phenomena (151,152) and various nonlinear models have been postulated. These mathematical

models are often of a form which requires a special technique for least squares fitting.

3. Often several possible models are investigated to explain a pharmacokinetic

phenomenon. There has been increasing interestin in discriminating

between such models (152,153).

It is appropriate to discuss points 2. and 3. above:

Fitting of Implicit Functions: a simple example

Several of the models describing nonlinear pharmacokinetic phenomena can be expressed in an implicit form which can be fitted by defining the functional relationship between the variables explicitly by an iterative procedure.

Consider, for example, a simple one-compartmental model with

intravenous injection in which the drug is eliminated partly by conversion to a single metabolite according to Michaelis-Menten kinetics and partly by excretion unchanged in the urine (154). The concentration of drug in plasma, c, is given *implicitly* by:

$$\ln \frac{c}{c_{o}} = \frac{V_{m}}{k_{1u}m} \ln \left[\frac{k_{1u}m + V + k_{1u}c}{k_{1u}m + u + k_{1u}c} \right] - \left[\frac{k_{1u}m + V}{K_{m}} \right] t$$
(9.1)

where V_{m} and K_{m} are the Michaelis-Menten parameters, $C_{o} = Dose/V_{1}$ and k_{1u} is the urinary elimination constant. This equation can be written more simply:

 $\ln c + A_1 \ln (A_2 + A_3 c) + A_4 = 0$ (9.2)

where A1

be prevent the parameter,

$$= \frac{V_{m}}{k_{lu} k_{m}}$$
(9.3)

$$A_{2} = \frac{k_{1} u m m}{k_{1} u m m k_{1} u c}$$
(9.4)

$$A_{3} = \frac{k_{1u}}{k_{1u} m m m k_{1u} c_{0}}$$
(9.5)

$$A_4 = \frac{k_1 K_1 + V_m}{K_m} t - \ln c_0$$
 (9.6)

The dependent variable, c, cannot be isolated from 9.1 but must be found by an iterative procedure by solving 9.2 for c.² The Newton-Raphson algorithm provides a simple and rapidly convergent method. If the expression in 9.2 is denoted g(c) then c can be determined by the following iteration:

$$c_{i+1} = c_i - \frac{g(c_i)}{g'(c_i)}$$
 (9.7)

(c) =
$$\frac{\partial g}{\partial c} = \frac{1}{c} + \frac{A_1 A_2}{A_2 + A_3 c}$$
 (9.8)

2. It would be incorrect, as is sometimes done for equations of similar type, to fit directly the equation where t is expressed as a function of the dependent variable. The result would be unreliable because the dependent variable which accounts for nearly all of the errors in the data, is treated as an independent variable without error. It is important that c_{i+1} in this iteration does not take a nonpositive value since this will terminate the execution of the program because g(c) (ln c) is not defined for $c \leq 0$. To prevent this it is most convenient to define

$$c_{i+1} = c_i/2$$
 if $c_{i+1} \leq 0$ (9.9)

This is an acceptable approach because g(c) is strictly increasing for c>0 since A_1 , A_2 , A_3 , $A_4 > 0$ and therefore, g'(c) > 0. The term $\ln (A_2 + A_3 c)$ in g(c) will also be defined under these conditions.

The above procedure can be used for most implicit mathematical models in nonlinear pharmacokinetics. However special care must be taken to prevent the parameters wandering into a parameter space where the function(s) is not defined (e.g. logarithm of a nonpositive number, division by zero etc.).

Discrimination between models

The best criterion to use in discriminating between alternate pharmacokinetic models depends on the aim of the investigation and the 157 application of the results (138,155-/). If the main aim is to discriminate between models, the experiment should be designed so that the hypothesized models are placed in as much jeopardy as possible.

The problem is nevertheless considerably complicated by the substantial variation and low reproducibility of measurements in a biological system and the limited number of sample points available. Discrimination on a statistical basis requires information about the variability of the observations which can only be estimated by repeated experiments. A likely outcome of such experiments would often be that the system is "ill-conditioned" i.e. the variability of the data is too large to allow a discrimination on a significant probability level.

Since the macroparametric representations of linear, compartmental models are all of the multi-exponential form:

$$= \sum_{i=1}^{n} A_{i}e^{-\alpha_{i}t} \qquad (A_{i}, \alpha_{i} > 0) \qquad (9.10)$$

it appears appealing in routine investigations of raw pharmacokinetic data to apply a "multiple regression approach", similar to that used for linear systems, to determine the order, n, of the system.

- This seems to be unreliable, however, for several reasons. 1. There exists no computer program which will inevitably find the "best" solution (smallest residual sum of squares) in a nonlinear least squares estimation which may have several minima.
- Measurements in biological systems often produce substantial residual values which may give rise to multiple minima.
- The number of minima will increase very rapidly as the number of exponential terms, n, to be fitted increases.

The problem of multiple minima can be reduced but not eliminated by a suitable procedure which gives good initial estimates, or by multiple runs with initial parameter values randomly taken from the parameter space, or by performing a lattice search. The interactive structure of FUNFIT makes it particularly suitable for performing multiple runs and lattice searches.

So-called "back-projection" or "stripping" is the technique most frequently used to obtain initial parameter estimates for models of the form described by 9.10. The stripping is either done graphically or automatically by the computer, in some cases employing a spline function representation of the data (152). However, it is important to realise that this particular technique assumes that one or more exponential terms vanish in certain regions of the total drug level-time curve. In other words the method tends to disregard cases where two or more exponential terms dominate fairly equally throughout the whole time space investigated. Hence, the method may produce biased results. Discrimination between alternative models (9.10) on this basis must therefore be considered unreliable.

It would be appropriate in this connection to refer to a different method which does not introduce such a bias (158). This method is based on a linear shift operator technique which appears not to have been used previously in pharmacokinetic studies. Currently it seems to be the most suitable to use in obtaining initial parameter estimates in linear compartmental models.

In evaluating how well a model describes some data, three points must be considered:

1. How well do the calculated values agree with those observed, i.e. what is the sum of squared residuals or the correlation coefficient? 2. Does the fit agree with the basic assumptions made about the errors? 3. How predictive is the model?

A comparison of fits entirely in terms of sum of squared residuals (such as an F-test) must be considered insufficient.

Analysis of Residuals

The importance of an analysis of residuals (137) seems to have been completely ignored in most computer programs. The basic assumptions in nonlinear least squares are: 1. The independent variable(s) is without error. 2. The errors, ε_i in the dependent variable are independent

- 223 -

 $\binom{\text{COV}}{i\neq j}(\varepsilon_i,\varepsilon_j)=0$ and normally distributed with zero mean and the same variance $(\varepsilon_i^{N}(0,\sigma^2))$.

The theory of least squares predicts that *if* the model is correct then the *residuals* should reflect the above properties of the *errors*, in an unbiased way, provided the errors possess these properties³ (140). The assessment of a model on the basis of a residual analysis must rely on the converse principle: If the *residuals* appear to be from the same normal distribution then, under assumption 2, the model should not be rejected.

To investigate whether the residuals are in fact equally normally distributed requires repeated experiments. When only one experiment is available a test must rely on the fact that if the residuals are deviates from the same normal distribution, then collectively they will be normally distributed. The converse, however, is not true in general (140). Therefore, in the absence of repeated experiments it is necessary to make the additional assumption that if the residuals are collectively normally distributed then they are individually normally distributed also.

Possibly the best way to examine the residuals is to plot them against the independent and dependent variables (137,138). Significant systematic deviations can be visualized in this way. FUNFIT includes such plots and the following statistics which may be helpful in the assessment and comparison of models.

The Kolmogorov-Smirnov statistic (159) is used to test for normality of the residuals. The procedure is as follows: Given N residuals, the program calculates

ultimate goal in mathematical modelling in pharmacokination

3. This is only strictly true in the linear case but approximately true for the nonlinear case.

- 224 -

$$= \max_{X} |F^{*}(X) - S_{N}(X)|$$
(9.11)

where $S_N(X)$ is the cumulative distribution of the residuals and $F^*(X)$ is the cumulative normal distribution function with the same mean and variance as the residual sample. The calculated value of D is compared with the critical value obtained from a Monte Carlo calculation at a given significance level.

The fundamental assumption of random errors is also tested in FUNFIT using the "run test" and the "number test" as discussed in Chapter 8. The Durbin-Watson statistic to test for serial correlation among the residuals was also presented in that chapter together with the test for "outliers".

If the residual analysis reveals that the residuals do not appear to be significantly random or normally distributed then this does not necessarily mean that the model is incorrect. More exactly it means one is faced with the problem of either rejecting the hypothesis that the model is "correct", rejecting the assumption made about the errors, or rejecting the assumption that the computer program has found the "best" solution in the case of multiple sum of squares minima. In the last two cases the model cannot be verified. This clearly emphasizes the need for a computer program which is efficient in finding a global minimum, the need for accurate data to reduce or eliminate multiple minima, and the need for carefully designed experiments which do not introduce systematic or cumulative errors.

The predictive power of the model

The ultimate goal in mathematical modelling in pharmacokinetics is to establish models with significant predictive power. A similar goal

exterts in modellin D

exists in modelling of economic systems. The voluminous literature in this area can undoubtedly give inspiration to future approaches in pharmacokinetics.

A very useful test of the predictive capabilities of a model is to test the hypothesis that the parameters do not depend on the model variables. This can be done readily if sufficient data are available. The total set of data is first partitioned into 2 or more subsets. The parameters are then estimated separately for each subset and the parameter subsets are tested for any trend or for a functional relationship with the independent variable(s) by suitable correlation analysis. A test to establish whether the parameter subsets are significantly different from the parameter set obtained for the whole sample can also be employed (60,161). The highly interactive structure of FUNFIT readily facilitates such partitioning of the data enabling the above tests to be made.

A Comparison of FUNFIT and NONLIN

FUNFIT was applied to obtain parameter estimates of the following simplified 2- and 3-compartment models:

 $c = p_1^{e} + p_3^{e} \qquad p_1^{>0} \qquad (9.12)$

$$c = p_1^{e} + p_3^{e} + p_5^{e} p_i^{>0}$$
 (9.13)

which were used to describe the plasma profile of pancuronium after I.V. bolus injection in 4 human subjects. The data to which 9.12 and 9.13 were fitted are shown in table 9.1.

Identical parameter limits and initial estimates were chosen to those used in applying the 1969 version of NONLIN which appears to be the

- 226 -

Table 9.1 Pancuronium Bloodlevel Data to which Eqs. 9.12 and 9.13 are

fitted using NONLIN and FUNFIT.

	I.A.		в.	А.	M.C.		J.C.	
	TIME	CONC.	TIME	CONC.	TIME	CONC.	TIME	CONC.
No.	min	µg/ml	min	µg/ml	min	µg/ml	min	µg/ml
1	7.5	1.120	5.0	.600	5.0	1.033	5.0	1.440
2	10.0	.775	10.0	.556	10.0	.830	10.0	1.000
3	20.0	.545	15.0	.550	15.0	.800	15.0	.945
4	30.0	.510	20.0	.480	20.0	.680	20.0	.805
5	60.0	. 395	30.0	. 370	30.0	.555	30.0	.620
6	95.0	.416	60.0	. 200	60.0	.255	60.0	.463
7	120.0	.166	90.0	.160	91.0	.235	90.0	.365
8	152.0	.200	120.0	.150	120.0	.220	120.0	.355
9	180.0	.200	255.0	.090	240.0	.143	143.0	.270
10	190.0	.168	361.0	.100	413.0	.095	257.0	.160
11	250.0	.130	verall,	n the PU	ert reat	its.	408.0	.083
12	400.0	.067	IN's rea	alts in !	the cases	whore r	eiulte d	

identical to MONLIN's final estimates and with initial parameter atop al 0.1% of these parameter values. At the first iteration Pinsfr give eras the same SS-value as NONLIN's value of convergence but it did for accord this solution as a staticentry point and converged to a significantly different solution (Table 9.2).

- 227 -

most commonly used nonlinear regression program in pharmacokinetic investigations. The stopping criterion for FUNFIT was 0.1 (per cent) which gives approximately the same relative change in the SS value at convergence as NONLIN with its differently defined stopping criterion set at TEST=0.0001. The step size used in NONLIN to approximate derivatives was DEL=0.001; the same value as that used in the test problems given in the NONLIN user's manual. The experimental plasma levels were recorded to 3 significant digits after the decimal point so the precision factor in NONLIN was chosen as IDIG=-9 to avoid significant truncation of calculated values. This NONLIN and FUNFIT should give identical SS values for identical parameter values. This was verified in all runs.

FUNFIT found a different solution than NONLIN in every case where a 3-compartment model (9.13) was fitted, and in half of the cases where a 2-compartment model (9.12) was fitted (Table 9.2). The residual sum of squares values obtained using FUNFIT were substantially lower than those obtained using NONLIN in all cases where there was a difference. The average percentage difference was -55% and -29% when fitting Eqs. 9.13 and 9.12 respectively. The differences were also reflected in the parameter values. Furthermore, the run test indicates that the residuals are more randomly distributed overall in the FUNFIT results.

To test NONLIN's results in the cases where results differed for the two programmes FUNFIT was started with initial parameter values identical to NONLIN's *final* estimates and with initial parameter step sizes = 0.1% of these parameter values. At the first iteration FUNFIT gave exactly the same SS-value as NONLIN's value at convergence <u>but it did not accept</u> <u>this solution as a stationary point</u> and converged to a significantly different solution (Table 9.2). Table 9.2: Least squares fitting of Eq's 9.12 and 9.13 using NONLIN (N) and FUNFIT (F).

PATIENT /POINTS		SS	ASS% (a)	RUNS ^(b)	P1	P ₂	P ₃	P ₄	P ₅	P ₆
IA/12	N	.7131E-1	S D L	6	.1452E+1	.2871	.4553	.3022E-1	. 3986	.4165E-
	F	.2410E-1	-66.2	9	.4924E+1	.3026	.4081E-1	.1686	.6013	.6632E-
	F ^(c)	.2869E-1	-59.8	9	.2940E+1	.2370	.4855E-3	.3568E-1	.6025	.6712E-3
JC/11	N	.1825E-1		8	.1038E+2	.9543	.1139E+1	.7307E-1	.6039	.5048E-
	F	.6631E-2	-63.7	9	.4649E+1	.5826	.8223	.5242E-1	.5597	.4721E-
	F ^(c)	.6074E-2	-66.7	9	.1303E+2	.7920	.8369	.5411E-1	.5647	.4744E-
BA/10	N	.8973E-2		6	.2614	.4341E-1	.3327	.3473E-1	.1697	.1873E-
	F	.4653E-2	-48.1	6	.5603E-1	.5668	.5911	.2526E-1	.1041	.2758E-
	F ^(c)	.4447E-2	-50.4	6	.7253E-1	.2467E-1	.5253	.2478E-1	.9702E-1	.3492E-
MC/10	N	.1382E-1		4	.8224	.4488E-1	.7388E-1	.6326E-1	.3148	.2521E-
di Ei	F	.7844E-2	-43.2	6	.1658E+1	.9200	.9374	.3623E-1	.2283	.1936E-
	F ^(c)	.8346E-2	-39.6	6	.8740	.3600E-1	.818E-1	.6357E-1	.2336	.2035E-
IA/12	N	.5102E-1	- DEC	5	.1983E+1	.2367	.6586	.7449E-2		
	F	.2199E-1	-55.9	8	.1066E+2	. 3999	.6202	.6858E-2		
	F ^(c)	.2199E-1	-55.9	. 9	.1049E+2	.3978	.6198	.6853E-2		
JC/11	N	.1621E-1		7	.1240E+1	.9424E-1	.6496	.5616E-2		
	F	.1621E-1	0	7	.1240E+1	.9425E-1	.6497	.5616E-2		
BA/10	N	.4528E-2		6	.5998	.2569E-1	.1012	.2100E-3		
	F	.4446E-2	- 1.8	6	.5975	.2470E-1	.9677E-1	.5866E-5		
	F ^(c)	.4449E-2	- 1.8	6	.5962	.2484	.9835E-1	.5402E-4		
MC/10	N	.8194E-2		6	.9464	.3760E-1	.2373	.2093E-2		
	F	.8194E-2	0	6	.9464	.3761E-1	.2374	.2095E-2		
							and the second			

(a) ASS% = 100(SS_{FUNFIT} - SS_{NONLIN})/SS_{NONLIN}

(b) See text for definition of runs.

(c) FUNFIT was in these cases started using NONLINS final parameter estimate as initial estimates and with initial parameter step sizes = 0.1% of these parameter values.

111-2

of E is the relative error in the computed I-value and U,

aconal element of the Housian mate

The detailed minimization report chosen in the investigation of this phenomenon, in fact, showed that the SS-function in NONLIN's convergence region had a significant gradient value indicating that NONLIN's solutions in these cases were not sufficiently close to the true sum of squares minimum.

The most likely reason for the failure of NONLIN to find a satisfactory solution in these cases appears to result from substantial errors in the approximation of derivatives.

The derivative with respect to the i-th parameter, p_i , of the function $f(\underline{X},\underline{P})$ to be fitted in NONLIN is approximated by a one-sided difference formula:

$$\frac{\partial f}{\partial p_{i}} \sim \frac{f(\underline{x}, p_{1}, p_{2} \cdots p_{i}^{+\delta p_{i}}, \cdots p_{n}) - f(\underline{x}, \underline{p})}{\delta p_{i}}$$
(9.14)

where the step size δ , (DEL) is chosen by the user. The value of this quantity is critical to the accuracy of the derivative. In choosing a proper value for δ , one has to steer between two hazards: 1. If the value chosen for δ is too *small* the derivative will be substantially inaccurate because of the *rounding error* which arises when the two f-values in (9.14) are too close. 2. If δ is set too *large* the derivative approximation will be too inaccurate because of the *truncation error* (9.14 is only accurate in the limit as $\delta \rightarrow 0$). This indicates that there must be an optimal value for δ . It can be shown that this value, for the i-th parameter, is approximately given by:

$$\left|\delta_{\mathbf{i}}\right| \simeq \left|\frac{4 \mathbf{f}}{\mathbf{p}_{\mathbf{i}}\mathbf{H}_{\mathbf{i}\mathbf{i}}}\right| \tag{9.15}$$

where ε is the relative error in the computed f-value and H_{ii} is the i-th diagonal element of the Hessian matrix. This formula shows that the optimal

 δ -value differs from parameter to parameter. It is not uncommon to find a very large value for the ratio, $\max(p_i H_{ii})/\min(p_i H_{ii})$, indicating that the choice of a *single* common δ -value as is done in NONLIN may not be adequate for all derivative evaluations and may cause convergence to a non-stationary point (147).

The above problem can be overcome in several ways:

- By abolishing difference approximations and using exact analytical derivatives. This, however, may be of considerable inconvenience for the user who must define the analytical derivatives. It also limits the use of the program to equations for which analytical derivatives can be obtained.
- By modifying the initial choice of δ according to Eq. 19 or by other means (162). However, even if alterations are made according to 1. or
 the Gauss-Newton methods may still converge in some cases to a point at which the gradient does not vanish (145).
- By abolishing the linearization approach in the Gauss-Newton methods and using a general function minimization approach (163-165).
- 4. By the use of an algorithm which is not based on derivatives or derivative approximations as is done in FUNFIT. The disadvantage of the last approach is that more function evaluations are required to reach convergence. For most pharmacokinetic applications this disadvantage is not significant. However, in cases where many parameters (> about 12) are to be estimated or where the equation(s) to be fitted is very time-consuming to evaluate (for example in the fitting of a functional relationship described by a system of differential equations) the disadvantage may become significant.

The presence of multiple SS-minima in fitting the 3-compartment model is evident from the fact that in the cases where FUNFIT was started with NONLIN's final estimate it converged on a different solution (Table 9.2). The difference between SS-values found by FUNFIT in consecutive runs was much smaller than the difference between NONLIN and FUNFIT's values.

To test for multiple minima in fitting the 2-compartment model to IA's data, FUNFIT was started randomly 10 times in the chosen parameter space. In 9 of these cases it found the same solution (Table 9.2) but in one case it converged to:

SS = .2209E-1 and p_{1-4} = .8780E+1, .3744, .6162, .6797E-2.

It is encouraging that this minimum is larger than that found in the 9+2 other cases. The more frequent occurrence of different solutions in the 3-compartment fitting confirms that the problem of multiple minima increases with an increasing number of parameters.

In only about half the cases investigated did FUNFIT and NONLIN find a smaller SS value for the 3-compartment model than for the 2-compartment model. This clearly emphasises the problems in discriminating between nonlinear mathematical models as discussed.

If, in fitting linear compartmental models, 9.10, the lower limits for the coefficient parameters A_i are set to zero then, in theory, the fit in terms of SS of a higher order model (e.g. n=3 <u>vs</u> n=2) should always be better or at least as good as the fit of a lower order model.⁴ Therefore, if under such conditions it is found that the higher order model does not improve the fit (SS) then there is reason to believe a better minimum exists. In such cases, the higher order model should be refitted with initial estimates of the common parameters equal to the final parameter estimates of the lower order models. If this procedure also fails to give a lower SS-value, then a third run should be made where the common parameters

4. Provided of course that the parameter space for the lower order model is a subset of that of the higher order model. are restricted within narrow limits around the optimal values for the lower order model, and where the parameters in the higher order term (e.g. p₅ and p₆ above) are much less restricted. The above approach will be successful in most cases and its use is recommended not only for fitting to sums of exponentials but also for any other "order system" where the terms are allowed to vanish. It should reduce significantly the problems associated with multiple minima and the problems of finding suitable initial estimates.

Truncation

In the NONLIN program it is possible by using the IDIG parameter to specify various degrees of truncation of the calculated values for the dependent variable. For example, if IDIG is set at -3 then all calculated values of the dependent variable will be truncated to 3 significant digits after the decimal point. The philosophy behind the use of this parameter is that there is no reason to calculate the predicted values to any higher precision than the observed values. In adopting such a philosophy it must be realised that the results so obtained will be specific to the NONLIN program and in general cannot be compared with results obtained using other nonlinear regression programs. The difference between results obtained specifying virtually no truncation (IDIG= -9) and specifying truncation to the precision of the observations (IDIG= -3) was found to be very pronounced (Table 9.3). The substantial difference was reflected not only in the SS-values and parameter values but also in the randomness of the residuals.

Truncation (IDIG= -3), furthermore, strongly affects the errors in the derivative approximations since the value of ε and hence, δ in Eq. 9.15 will be affected. This may explain why NONLIN in one case (Table 9.3)

- 233 -

PATIENT /POINTS	PRECISION- FACTOR	SS	ASS% (a)	RUNS ^(b)	P ₁	P2	P ₃	P4	P5	P ₆
IA/12	-9	.7131E-1	48.7	6	.1452E+1	.2871	.4553	.3022E-1	.3986	.4165E-2
	-3	.3658E-1		8	.1793E+1	.1807	.3155	.1097E-1	.2949	.3697E-2
JC/11	-9	.1825E-1	35.2	8	.1028E+2	.9543	.1139E+1	.7307E-1	.6039	.5048E-2
F - 3	-3	.1183E-1		6	.112E+1	.1813	.5571	.3310E-1	.5060	.4237E-2
BA/10	-9	.8973E-2	68.8	6	.2614	.4341E-1	.3327	.3473E-1	.1697	.1873E-2
1	-3	.1515E-1		4	.1396	.5172E-1	.4649	.4265E-1	.1734	.1953E-2
MC/10	-9	.1382E-1	582 (c)	4	.8224	.4488E-1	.7388E-1	.6326E-1	.3148	.2521E-2
,	-3	.9427E-1		3	.8000	.6000	.3000	.1000	.4000E-1	.2000E-2
IA/12	-9	.5102E-1	74.9	5	.1983E+1	.2367	.6586	.7449E-2		
111/ 12	-3	.8924E-1		4	.8795	.4713E-1	.3312	.2706E-2		
JC/11	-9	.1621E-1	165	7	.1240E+1	.9424E-1	.6496	.5616E-2		
	-3	.4306E-1		4	.1114E+1	.4835E-1	.4251	.2649E-2		
BH/10	-9	.4528E-2	17.1	6	.5998	.2569E-1	.1012	.2100E-3	8 F.	
	-3	.5303E-2		6	.5547	.3019E-1	.1610	.2121E-2		
MC/10	-9	.8194E-2	2.78	6	.9464	.3760E-1	.2373	.2093E-2		
,	-3	.8422		5	.9226	.4107E-1	.2760	.2779E-2		

Table 9.3: Least squares fitting of Eq's 9.12 and 9.13 using NONLIN with precision factor -3 and -9.

(a) $|\Delta SS_{1}| = 100 \times |S_{(-9)} - S_{(-3)}| / S_{(-9)}$

(b) See text for definition of 'runs'.

(c) NONLIN failed to find a proper solution. At convergence after 5 iterations the final parameter estimates were identical to the initial estimates.

. 234

failed to converge properly.

The relative precision with which the parameter can be calculated will also be affected by truncation since it will not be possible to improve their estimates further when the computer has reduced the residual sum of squares to a value, SS, for which

$$\frac{SS}{SS(true)} < 1 + \varepsilon$$
 (9.16)

where the error, E, depends on the degree of truncation chosen by IDIG (149). Furthermore the truncation procedure cannot be justified on a statistical basis since the theory of least squares assumes no errors in the calculated values. In fact great effort is often made to program the function to be fitted so that it can be evaluated with minimum errors to avoid biased results.

The user of the program NONLIN is, therefore, advised to use a value for the precision factor IDIG such that minimum truncation takes place.

The many nonlinear regression programs available have provided the scientist with a powerful tool useful for a great variety of problems. However, the results obtained have too often been accepted and used without an awareness of the limitations and possible unreliability of the program used, ignoring the numerical problems involved.

The complex structure of the program used has often resulted in an authoritative attitude which may go so far as using the program as a substitute for rational thought. There is a definite need for programs which are more reliable and which allow greater interaction between the user and the program, with the user in a more dominant role.

CHAPTER 10

A NEW METHOD FOR CHARACTERISING THE DISSOLUTION

PROPERTIES OF DRUG POWDERS

Chapter 7 demonstrated good agreement between the theoretical dissolution profile and experimental data for tolbutamide. The intrinsic dissolution profile was calculated, using the rigorous mathematical approach present in Chapter 5 and 6, from optical analysis of the size distribution of the 60/85 mesh fraction powder. Such an analysis is complicated to perform for micronized powders because of the highly irregular particle shapes and the degree of aggregation often found in such powders.

However it should be possible to determine by nonlinear regression analysis whether the dissolution of such a powder can be adequately described by one of the multiparticulate dissolution models presented earlier.

In this chapter various such models, based on a log-normal particle size distribution, are fitted by nonlinear least squares regression to data from the dissolution of micronized glibenclamide using the FUNFIT program. Estimates of parameters describing the *effective* initial particle size distribution are obtained together with estimates of a quantity defined as the specific dissolution rate parameter. A dissolution equation based on an ideal, untruncated log-normal distribution with the single particles dissolving according to the cube root law best describes the dissolution kinetics. Dissolution behaviour of glibenclamide can be well described by this model in terms of the specific dissolution rate parameter and one other parameter which accounts for the distribution effect termed the dispersion parameter. Estimation of these two parameters provides a means of describing dissolution characteristics of drug powders apparently more correctly than previous approaches. The method should be of interest in the quality control of drugs likely to cause bioavailability problems because of dissolution rate limited absorption.

Theoretical

where

The dissolution equation for a log-normal powder, considering the cube root and the square root model, was given earlier by Eq. 6.21. This equation appears to contain 5 parameters, namely σ , i, j, μ and K_m which define the dissolution profile, W/W_o versus time. However, an attempt to obtain least squares estimates of all these 5 parameters from W/W_o vs. time data may fail because μ and K_m can be fused into a single parameter:

$$K_{m}^{*} = e^{-\frac{3}{m}\mu} K_{m}$$
 (10.1)

which will be called the specific dissolution rate parameter.

The uniqueness of these 4 parameters, σ , i, j and K^* , in defining the dissolution profile can be seen by substituting $K_m = e^{\frac{3}{m}\mu} K^*_m$ into Eq. 10.1 resulting in total cancellation of μ , in full agreement with the theory discussed previously:

$$\frac{W}{W_{O}} = \sum_{n=0}^{m} {m \choose n} (-K_{m}^{*}t)^{(m-n)} \frac{F(j-\frac{3n\sigma}{m})-A}{F(j-3\sigma)-F(-i-3\sigma)} e^{\frac{9}{2}\left(\frac{n}{m}\right)^{2}-\frac{3}{2}\sigma^{2}} (10.2)$$

$$A = F(-i-\frac{3n\sigma}{m}) \text{ for } t < \frac{1}{K_{m}^{*}} e^{\frac{-3i\sigma}{m}} (10.3)$$

and A = F(
$$\frac{m}{3\sigma}$$
 ln(K_m^* t) $-\frac{3n\sigma}{m}$) for $\frac{1}{K_m^*}$ e $m > t > \frac{1}{K_m^*}$ e m (10.4)

and

The continuous flow, recording apparatus used provides dissolution rate data. The fraction undissolved, W/W_o , versus time can be obtained by integrating. The parameters σ , i, j and K_m^* can then be estimated by nonlinear least squares regression analysis using Eq. 10.2. However the integrated data will contain integration errors. The integration also tends to "smooth" the original data so estimates of the variability of the parameters will be less reliable than if the original *rate* data were used. It is therefore useful to derive an expression for the release rate in order to estimate the 4 parameters directly from the original (rate) data.

(10.5)

By applying Eq. 5.33 of Chapter 5 the following equation is obtained:

$$\frac{W}{W_{o}} = \frac{\int_{R_{1}}^{R_{2}} \left[x^{\frac{3}{m}} - \kappa_{m} t\right]^{m} x^{-1} N(\ln x, \mu, \sigma) dx}{\int_{0}^{D} x^{2} N(\ln x, \mu, \sigma) dx}$$
(10.6)

where $d_o = e^{\mu - i\sigma}$ (10.7) $D_o = e^{\mu + j\sigma}$ (10.8) $R_1 = (\max[K_m t, d_o^{\frac{3}{m}}])^{\frac{m}{3}}$ (10.9) $R_2 = (\max[K_m t, D_o^{\frac{3}{m}}])^{\frac{m}{3}}$ (10.10)

An expression for the dissolution rate of drug, Q = -dW/dtcan now be found by differentiating Eq. 10.6 with respect to time (using Leibnitz's rule). For abbreviation, let B denote the (constant) denominator of Eq. 10.6; differentiation then gives:

 $\frac{W}{W_{0}} = 0 \quad \text{for} \quad t \ge \frac{1}{K_{m}^{*}} e^{\frac{3j\sigma}{m}}$

 $Q = mK_{m} \int_{-\infty}^{R_{2}} (x^{-Kt})^{(m-1)} x^{-1} N(\ln x, \mu, \sigma) dx \qquad (10.11)$

$$\left(R_{2}^{\frac{3}{m}}-K_{m}t\right)^{m}R_{2}^{-1}N\left(\ln R_{2}, \mu, \sigma\right)\frac{dR_{2}}{dt}$$

$$\left(R_{1}^{\frac{3}{m}-K_{m}t}\right)^{m}R_{1}^{-1}N\left(\ln R_{1},\mu,\sigma\right)\frac{dR_{1}}{dt}$$

Before critical time¹ $R_1 = d_0$, $R_2 = D_0$ so $dR_1/dt = dR_2/dt = 0$, and the last two terms of the right-hand side of Eq. 10.11 vanish. After the critical time, dR_2/dt is still zero but $R_1 = (K_m t)^{\frac{m}{3}}$ (according to 10.9), thus the last term also vanishes because then:

$$R_1^{\frac{3}{m}} - K_m^{t} = ((K_m^{t})^{\frac{m}{3}})^{\frac{3}{m}} - K_m^{t} =$$

Equation 10.11 can therefore be simplified to:

$$Q = mK_{m}W_{o} = \frac{\int_{R_{1}}^{R_{2}} \frac{3}{(x^{m}-K_{m}t)} (m-1) x^{-1} N(\ln x,\mu,\sigma) dx}{\int_{0}^{D} x^{2} N(\ln x,\mu,\sigma) dx}$$
(10.12)

oters (ovir) and () are

(10.13)

The term $(x^{m}-K_{m}t)^{(m-1)}x^{-1}$ under the integral sign can be expanded to:

$$x - 2K_3t + (K_3t)^2 x^{-1}$$
 (for m=3)
 $x^{\frac{1}{2}} - K_2t x^{-1}$ (for m=2)

and the formula given previously (5.34) can be applied to express the integrals in 10.11 in terms of the function F. This leads to the following

 The critical time is the time when the first particles begin to disappear in the dissolution process, i.e.
 <u>3</u>(µ-iσ)

$$t = \frac{1}{K_{m}} e^{\frac{3}{m}(\mu - \mu)}$$

$$Q = m K_{m} \bigotimes_{n=0}^{(m-1)} {\binom{m-1}{n}} (-K_{m}t)^{(m-n-1)} \frac{F(\frac{T_{2}-\mu}{\sigma} - \frac{3n\sigma}{m}) - F(\frac{T_{1}-\mu}{\sigma} - \frac{3n\sigma}{m})}{F(j-3\sigma) - F(-1-3\sigma)} \times \frac{\frac{3}{m}(n-m)}{\binom{\mu+3}{\sigma}} (n+m)\sigma^{2})$$
(10.14)

where T_1 and T_2 are defined:

$$T_{1} = \max\left[\frac{m}{3}\ln(Kt), \mu - i\sigma\right]$$
(10.15)
$$T_{2} = \max\left[\frac{m}{3}\ln(Kt), \mu + j\sigma\right]$$
(10.16)

This expression for the dissolution rate Q contains 5 parameters $(\sigma, i, j, \mu \text{ and } K_m)$. However, as before only 4 parameters $(\sigma, i, j \text{ and } K_m^*)$ are needed to define uniquely the dissolution rate profile:

$$Q = m K_{m 0}^{*W} \sum_{n=0}^{(m-1)} {\binom{m-1}{n}} (-K_{m}^{*t})^{(m-n-1)} \frac{F(j-\frac{3n\sigma}{m})-A}{F(j-3\sigma)-F(-i-3\sigma)} \times (10.17)$$

$$= \frac{9}{e^2} \left[\left(\frac{n}{m}\right)^2 - 1 \right] \sigma^2$$

This equation is the differential form of Eq. 10.2. The quantity A is defined as previously (10.3 and 10.4) and $\frac{3j\sigma}{\sigma}$

$$Q = 0 \text{ for } t \ge \frac{1}{K_m^*} e^{\frac{3j0}{m}}.$$

The above expression is of considerable value since it allows the effective initial particle size distribution parameters σ , i and j to be determined together with the specific dissolution rate parameter, K_m^* , by nonlinear regression analysis of the dissolution rate profile (dW/dt versus time).

It is also of interest to investigate how well the dissolution behaviour can be described if the particle size distribution is considered Noting that $F(\infty) = 1$ and $F(-\infty) = 0$, Eq. 10.17 becomes, for an ideal distribution:

$$Q = m K_{m 0}^{*W} \sum_{n=0}^{(m-1)} {\binom{m-1}{n}} (-K_{m}^{*t})^{(m-n-1)} \left[1 - F \left(\frac{m}{3\sigma} \ln(K_{m}^{*t}) - \frac{3n\sigma}{m}\right) \right] X$$

$$= \frac{9}{2} \left[\left(\frac{n}{m}\right)^{2} - 1 \right] \sigma^{2}$$
(10.18)

If the distribution is considered ideal at the lower end $(i=\infty)$ but truncated at the higher end, the expression becomes:

$$Q = m \underset{m \circ o}{K^{*}W} \sum_{n=o}^{(m-1)} {m-1 \choose n} (-K_{m}^{*}t)^{(m-n-1)} \frac{F(j-\frac{3n\sigma}{m}) - F(\frac{m}{3\sigma} \ln(K_{m}^{*}t) - \frac{3n\sigma}{m})}{F(j-3\sigma)} x$$

$$= \frac{9}{2} [(\frac{n}{m})^{2} - 1] \sigma^{2}$$
(10.19)

and if it is considered truncated at the lower end but not at the higher end $(j=\infty)$:

$$Q = m K_{m}^{*W} \circ \sum_{n=0}^{(m-1)} {m-1 \choose n} (-K_{m}^{*}t)^{(m-n-1)} \frac{1-A}{1-F(-i-3\sigma)} e^{\frac{9}{2} \left[\left(\frac{n}{m}\right)^{2} - 1 \right] \sigma^{2}}$$
(10.20)

Monodisperse powder

It is of interest to look at the limiting case where the particle size distribution is infinitely narrow, i.e. $\sigma=0$, since this case allows a better understanding of the specific dissolution rate parameter, K_m^* . It also provides a method of obtaining a suitable initial estimate of this parameter to use in nonlinear curve fitting.

When $\sigma=i=j=o$, Eq. 10.17 becomes:

$$Q = m K_{mo}^{*W} \sum_{n=0}^{(m-1)} {m-1 \choose n} (-K_{m}^{*t})^{(m-n-1)}$$
(10.21)

which can be written more simply as:

$$Q = - \frac{dW}{dt} = m K_{mo}^{*W} (1 - K_{m}^{*t})^{(m-1)}$$
(10.22)

which, after integration, can be written:

$$\frac{W}{W_{O}}$$
) ^{$\frac{1}{m}$} = 1 - K^{*}_mt (10.23)

As expected, since there is no size distribution effect this equation predicts that the powder will dissolve strictly according to the cube root or square root model.

When the powder is monodisperse $\mu = \ln a_0$ and Eq. 10.1 becomes:

$$K_{m}^{*} = e^{-\frac{3}{m} \ln a} K_{m} = a_{o}^{-\frac{3}{m}} K_{m}$$
(10.24)
spherical particles $K = (60\pi)^{\frac{1}{m}} k$ where k_{m} is the rate parameter in

For spherical particles $K_m = (6\rho\pi)^m k_m$ where k_m is the rate parameter in the single particle dissolution model:

$$\frac{1}{m} = w_{0}^{m} - k_{m}^{t}$$
(10.25)

thus 10.24 becomes:

K*

$$= a_{o}^{-\frac{3}{m}} (6/\rho\pi)^{\frac{1}{m}} k_{m} = w_{o}^{-\frac{1}{m}} k_{m}$$
(10.26)

When $k_m = w_0^{\frac{1}{m}} K_m^*$, from this expression is inserted in 10.25 the single particle dissolution model can be written:

$$\left(\frac{w}{w}\right)^{\frac{1}{m}} = 1 - K_{m}^{*t}$$
 (10.27)

A comparison of 10.27 and 10.23 shows that for a monodisperse powder the specific dissolution rate parameter, K_m^* , is common to both multiparticulate

(10.23) and single particle dissolution (10.27). For both m=2 and 3 it has the dimension of time⁻¹.

Equation 10.22 can be written:

$$Q^{\frac{1}{m-1}} = (mW_{O}K_{m}^{*})^{\frac{1}{m-1}} - (mW_{O})^{\frac{1}{m-1}} (K_{m}^{*})^{\frac{m}{m-1}} t$$
(10.28)

This equation can be used to obtain an initial estimate of K_m^* from the linear regression of $Q^{\overline{m-1}}$ on t, using dissolution rate data. The intercept value, $(mW_OK_m^*)^{\frac{1}{m-1}}$, divided by $(mW_O)^{\frac{1}{m-1}}$ gives $K_m^{*\frac{m-1}{m-1}}$. (Fig. 10.1)

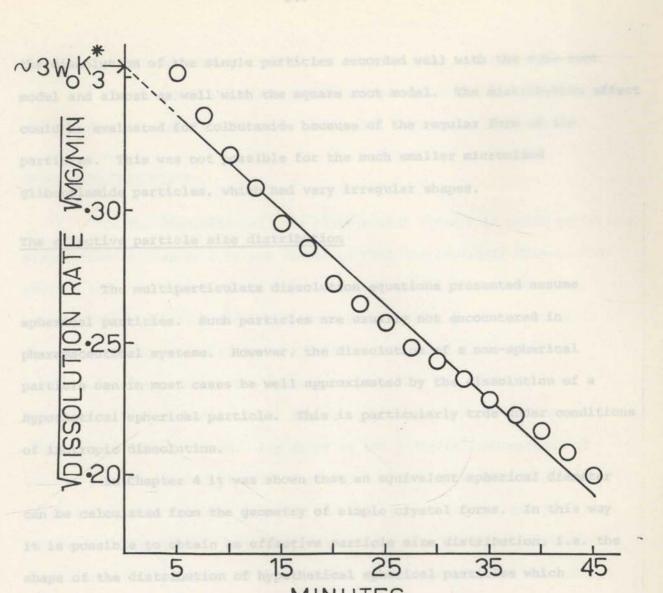
For a monodisperse powder 10.28 predicts a linear relationship between \sqrt{Q} and t when the single particles dissolve according to the cube root model (m=3) and a linear relationship between Q and t when they dissolve according to the square root model (m=2).

Significant deviations from linearity were observed when dissolution rate data for micronized glibenclamide were plotted in either of these ways (Figs. 10.1-10.5). Such deviations can arise if the powder is not monodisperse or if the single particles do not dissolve according to the single particle model given by 10.25.

Under an electron microscope the micronized glibenclamide used appears to be quite polydisperse. The observed deviation from linearity in the rate plots can thus be explained as a particle size distribution effect, assuming 10.25 to be valid. However, the rate data can also be explained by other single particle dissolution models in combination with a size distribution effect. Conclusions about the validity of a single particle model can only be made when the size distribution effect can be taken into account.

Chapter 7 dealing with the dissolution of 60/85, mesh fraction tolbutamide in relation to its particle size distribution indicated that

Illustration of graphical method for obtaining an initial estimate of the specific dissolution rate parameter, K_m^* , from rate data from the dissolution of 5 mg micronized glibenclamide. Equation 10.28 predicts for a monodisperse powder in which the single particles dissolve according to the cube root law a linear relationship between $\sqrt{2}$ and t. The deviation from linearity in the graph is caused by the size distribution effect.



MINUTES . For tolloutenide it was found that the effective distribution are approximately low-normal. Ancause of the irregular particle shapes it was not possible to make a priori continuious about the effective particle size distribution for plibenclamide. In these investigations the distribution is any mode to be log-normal, consistent with the results in Chapter 7 and the feat that providers are place found to have a low-normal particle size distribution (122,166). The distribution in Fig.5.1 cherefore illustrates a descented approximation to the offective particle size distribution. De logical particle dimeters are the disactors of the hyperbalical approximation corried the dissolution of the single particles accorded well with the cube root model and almost as well with the square root model. The distribution effect could be evaluated for tolbutamide because of the regular form of its particles. This was not possible for the much smaller micronized glibenclamide particles, which had very irregular shapes.

The effective particle size distribution

The multiparticulate dissolution equations presented assume spherical particles. Such particles are usually not encountered in pharamaceutical systems. However, the dissolution of a non-spherical particle can in most cases be well approximated by the dissolution of a *hypothetical* spherical particle. This is particularly true under conditions of isotropic dissolution.

In Chapter 4 it was shown that an equivalent spherical diameter can be calculated from the geometry of simple crystal forms. In this way it is possible to obtain an effective particle size distribution, i.e. the shape of the distribution of hypothetical spherical particles which approximate the dissolution behaviour of real particles. For tolbutamide it was found that the effective distribution was approximately log-normal. Because of the irregular particle shapes it was not possible to make a priori conclusions about the effective particle size distribution for glibenclamide. In these investigations the distribution is assumed to be log-normal, consistent with the results in Chapter 7 and the fact that powders are often found to have a log-normal particle size distribution (122,166). The distribution in Fig.5.1 therefore illustrates a log-normal approximation to the effective particle size distribution. The initial particle diameters are the diameters of the hypothetical spherical particles

- 246 -

that approximate the dissolution of the real nonspherical particles.

If the lower and upper truncation parameters, i or j, are finite the log-distribution is said to be truncated; otherwise it is ideal.

Regression parameters

In the discussion of size distribution effects in multiparticulate dissolution in Chapter 6 it was indicated that the *intrinsic dissolution* profile does not depend on the actual size of the particles but on the shape of their distribution. For the same reason it is not possible to determine by regression analysis the scale parameter, μ , as might be erroneously expected from the appearance of 6.21. If, in fact, this equation is used to obtain least squares estimates of μ and K_m , these would not be unique values. Any other of the infinite combinations of the two parameters that give the same value of K_m^* (10.1) will according to 10.2 result in the same fit (for the same values of σ , i and j). The regression analysis can thus only provide estimates of the *dimensionless* distribution parameters σ , i and j, which define the *shape* of the initial distribution, and K_m^* . The scale or position of the distribution, given by μ , is hidden in the specific dissolution rate parameter, K_m^* .

Curvefitting

Equation 10.17 was fitted by least squares to dissolution rate data from dissolution of 5 and 10 mg micronized glibenclamide using FUNFIT. The kinetic models given by 10.17-10.19 are defined in schemes 10.1-10.3

respectively. The "model parameter", m. can be but at a por a depending on whether a model based on the cube- or expert cout is to be fitted. The use of m is of great convenience since a fitting to both models can quickly be done with a single input of dissolution data.

The estimates obtained for the truncation parameters, i and j (Table 10.1, Eq. 10.17), were all larger than 2 and in most cases exceeded 4, indicating that the effective initial particle size distribution was close to ideal. It was shown in Chapter 5 that the effect on the dissolution profile of the lower truncation parameter, i, is negligible. Simulation studies also show that the influence of an increase in the upper truncation parameter, j, becomes insignificant when j is larger than 2. It is therefore expected from the values of i and j obtained using 10.17 that the simpler model, 10.18, which assumes an ideal distribution $(i=j=\infty)$ should fit the same data nearly as well. The values in table 10.1 and the curves fitted in Figs. 10.2-10.5 confirm this expectation. There does not seem to be any significant difference in either the K_m^* , the σ or the r-values for the two models. The residual plots in Figs. 10.2, 10.3 and Figs. 10.4, 10.5 also seem to be very similar. Equation 10.19 (upper truncation) and 10.20 (lower truncation) also gave similar results (and therefore, these have not been included).

Choice of the mathematical model

In agreement with the general principles of mathematical modelling, Eq. 10.17, should be considered as the model that best describes the dissolution of the micronized glibenclamide. This is because it is the

^{2.} The user of FUNFIT does not need to specify the binomial coefficients or the cumulative standard normal distribution function, F, since these are present in FUNFIT as function routines named NBC and SDF respectively. The "model parameter", m, can be set at 3 or 2, depending on whether a model based on the cube- or square root model is to be fitted. The use of m is of great convenience since a fitting to both models can quickly be done with a single input of dissolution data.

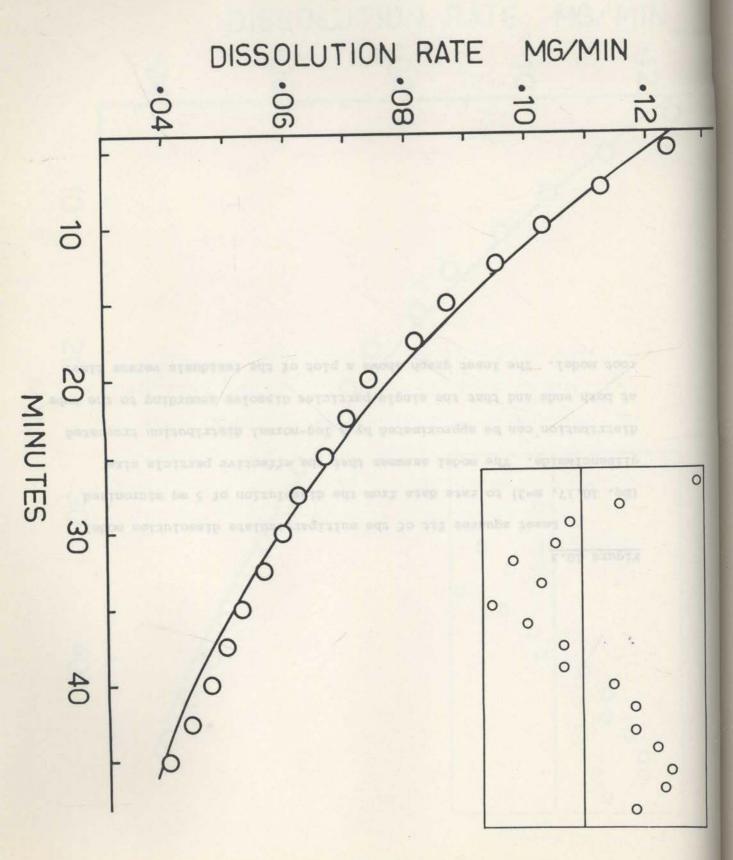
W _o (mg)	SQUARE ROOT MODEL (m=2)							CUBE ROOT MODEL (m=3)					
		Eq.10.18 ^(b) Scheme 10.2		Eq.10.19 ^(c) Scheme 10.3		Eq.10.17 ^(d) Scheme 10.1		Eq.10.18 (b) Scheme 10.2		Eq.10.19 (c) Scheme 10.3		Eq.10.17 ^(d) Scheme 101	
	$\kappa_m^*(\min^{-1})$	0.02858	(6.39)	0.02858	(5.12)	0.02859	(6.51)	0.02367	(4.55)	0.02367	(4.44)	0.02449	(3.96)
	σ	0.4759	(2.93)	0.4760	(2.70)	0.4760	(3.28)	0.6184	(2.47)	0.6184	(2.11)	0.6315	(1.71)
	i j	30 30		∞ 5.299		5.136 5.655		8		∞ 9.807		2.364	
	r ^(e)	0.9935		0.9935		0.9935		0.9952		0.9952		0.9954	
	K _m [*] (min ⁻¹)	0.02633	(7.18)	0.02633(10.6)	0.02633	(9.35)	0.02147	(9.45)	0.02147	(4.74)	0.02176	(4.03)
	σ	0.4955	(3.20)	0.4956	(7.24)	0.4955	(5.14)	0.6374	(3.29)	0.6374	(2.03)	0.6446	
10	i	00				8.912		8				4.577	
	j			5.299		6.732		00		7.060		4.200	
	r ^(e)	0.9938		0.9938		0.9938		0.9957		0.9957		0.9958	

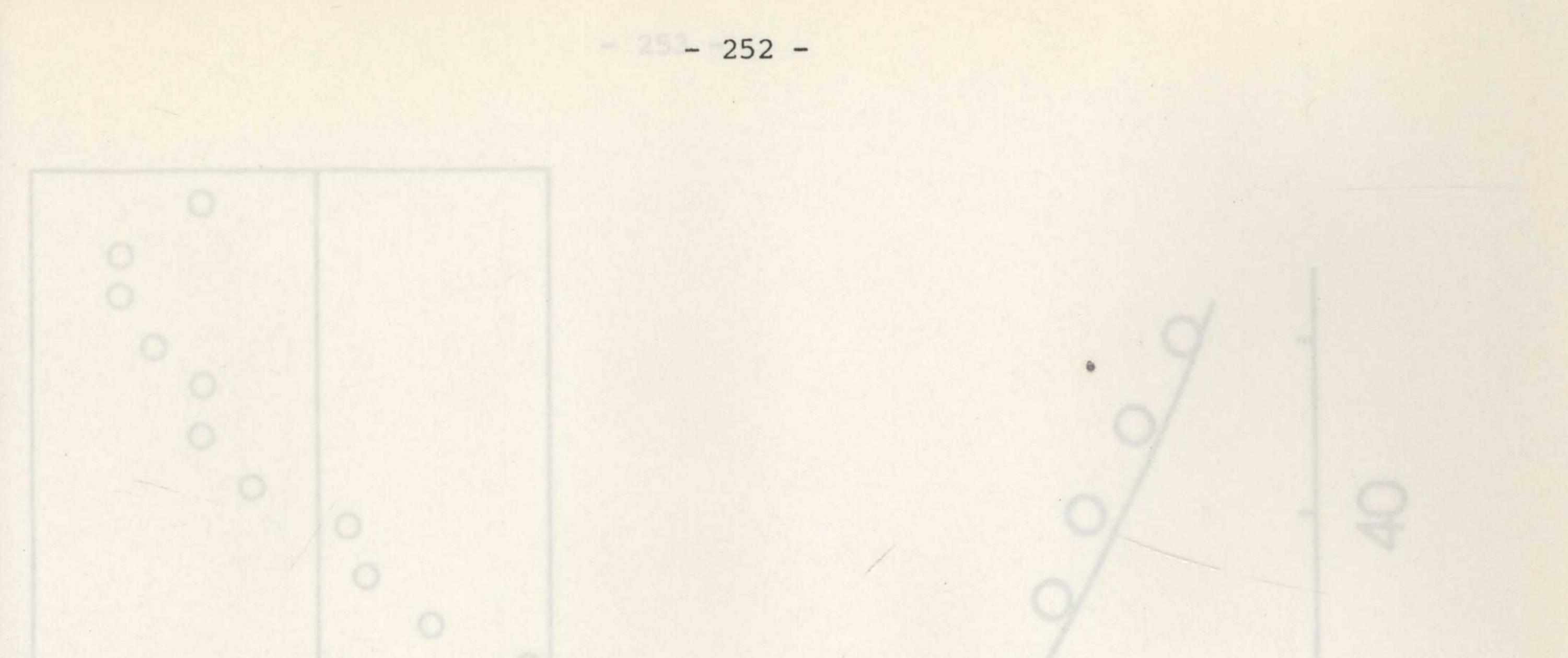
TABLE 10.1 Least squares estimates of rate- and distribution parameters obtained from nonlinear regression analysis of data from the dissolution of micronized glibenclamide, considering various models for multiparticulate dissolution kinetics.^(a)

(a) The values in brackets are relative standard deviations (percent).
(b) The initial size distribution is considered ideal (i=j=∞,Fig.l).
(c) The initial size distribution is truncated at the upper end (i=∞, j<∞,Fig.l).
(d) The initial size distribution is truncated at both ends (Fig.l).

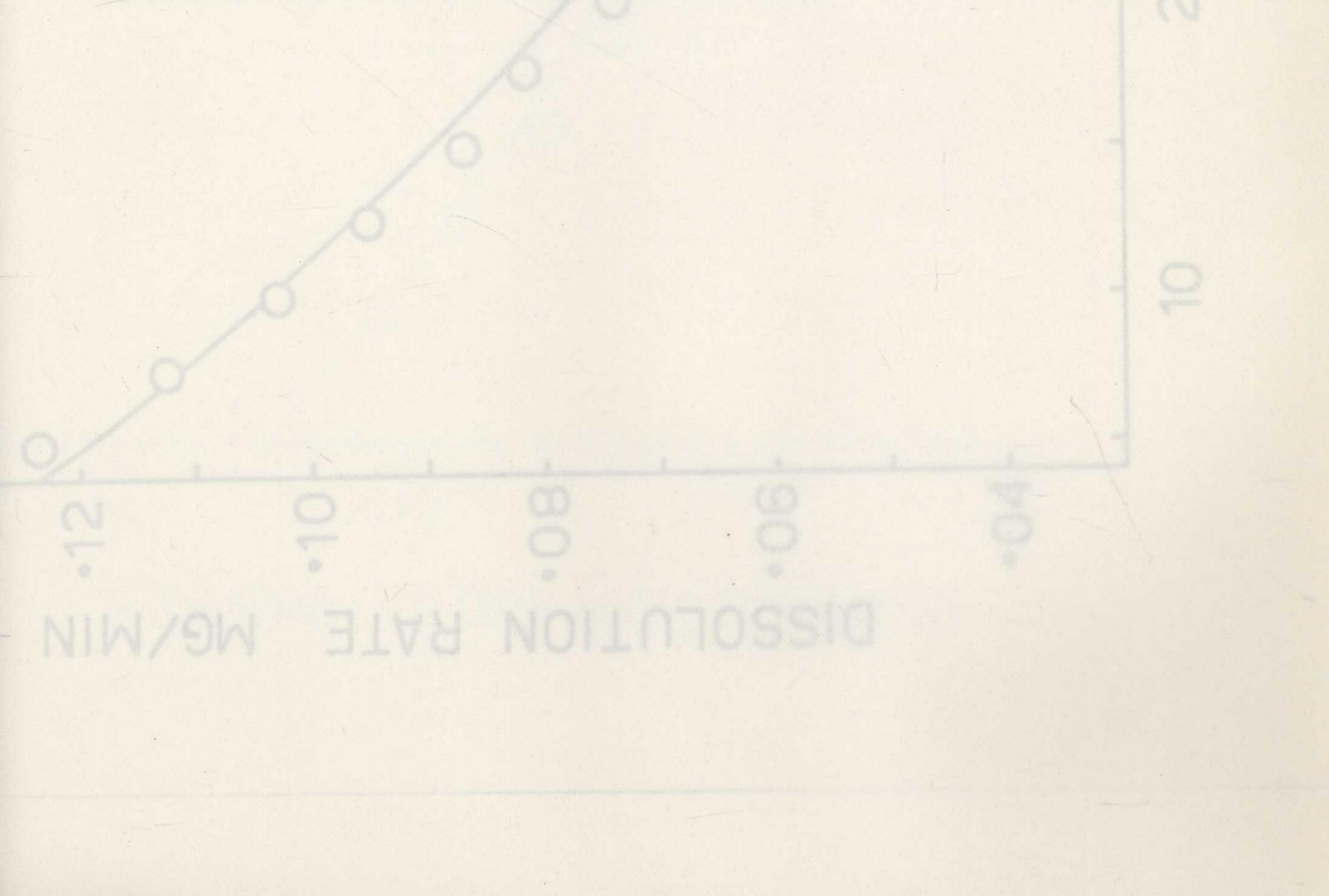
(e) Correlation coefficient.

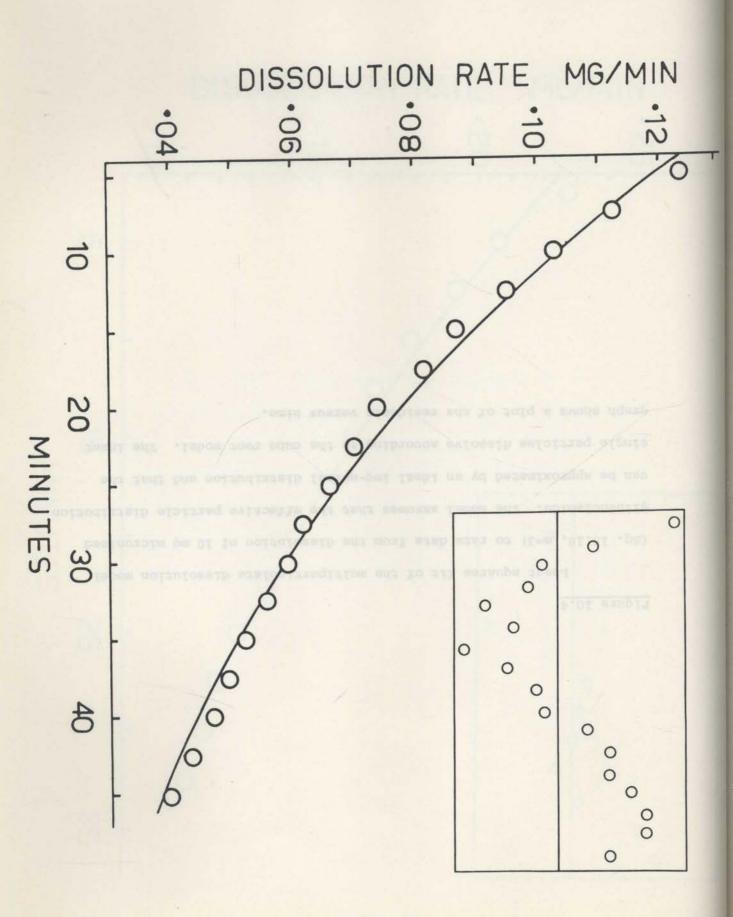
Least squares fit of the multiparticulate dissolution model (Eq. 10.18 m=3) to rate data from the dissolution of 5 mg micronized glibenclamide. The model assumes that the *effective* particle size distribution can be approximated by an *ideal* log-normal distribution and that the single particles dissolve according to the cube root model. The inset graph shows a plot of the residuals *versus* time.





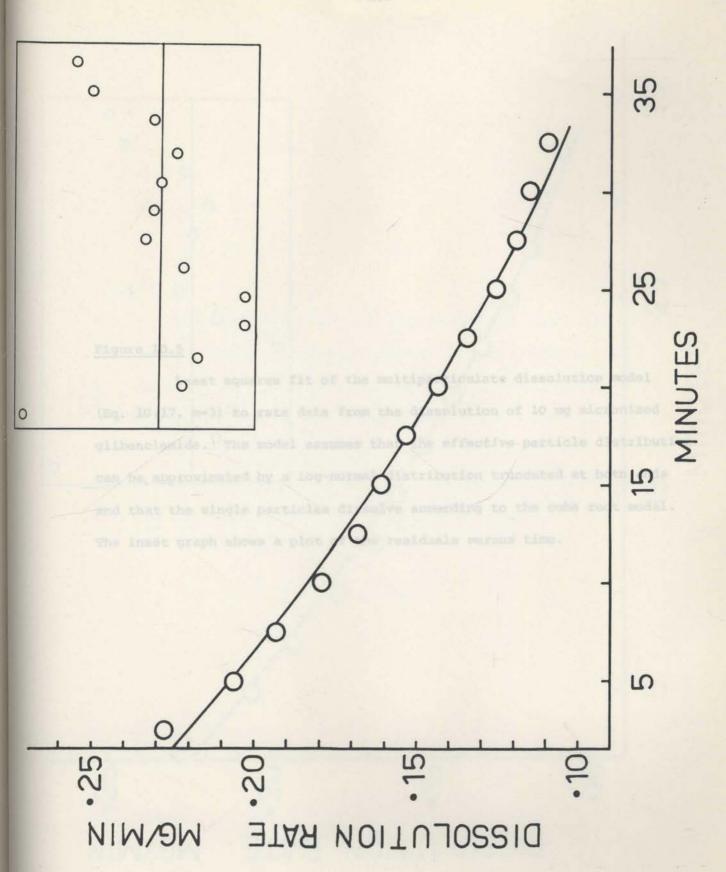
Least squares fit of the multiparticulate dissolution model (Eq. 10.17, m=3) to rate data from the dissolution of 5 mg micronized glibenclamide. The model assumes that the *effective* particle size distribution can be approximated by a log-normal distribution truncated at both ends and that the single particles dissolve according to the cube root model. The inset graph shows a plot of the residuals *versus* time.





Least squares fit of the multiparticulate dissolution model (Eq. 10.18, m=3) to rate data from the dissolution of 10 mg micronized glibenclamide. The model assumes that the *effective* particle distribution can be approximated by an ideal log-normal distribution and that the single particles dissolve according to the cube root model. The inset graph shows a plot of the residuals *versus* time.

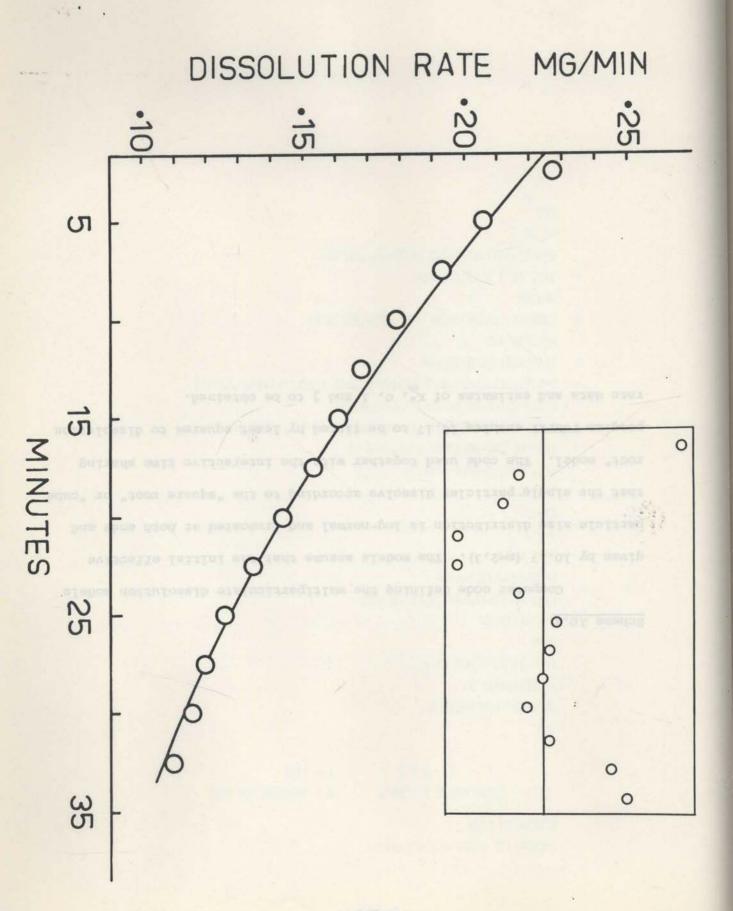
DISSOLUTION RATE MGVMIN



- 255 -

Least squares fit of the multiparticulate dissolution model (Eq. 10.17, m=3) to rate data from the dissolution of 10 mg micronized glibenclamide. The model assumes that the *effective* particle distribution can be approximated by a log-normal distribution truncated at both ends and that the single particles dissolve according to the cube root model. The inset graph shows a plot of the residuals *versus* time.

NIM\DM BTAR NOITUJO2210



SUBADUTTIE PODEL CO. T.P. (PRINT) DIVISION POR

+ SIGPR, KSTAR, I.J., VO.M Q + DISSOLUTION MIT

Scheme 10.1

Computer code defining the multiparticulate dissolution models given by 10.17 (m=2,3). The models assume that the initial effective particle size distribution is log-normal and *truncated at both ends* and that the single particles dissolve according to the "square root" or "cube root" model. The code used together with the interactive time sharing program FUNFIT enables 10.17 to be fitted by least squares to dissolution rate data and estimates of K_m^* , σ , i and j to be obtained.

> WELTE (6,5)? FORMAT (* STORM, KSTAN, 1, J, MI, M-74552.4 AETUGA IF(T, GE, 1, 77(2)) AETUGH G-P(G)*2(2)*P(S)*((1, -P(2)*T)**(1-1)) GO TO 2 END

```
SUBROUTINE MODEL(Q, T, P, IPRINT)
          DIMENSION P(6)
                                    Q = DISSOLUTION RATE
          P(I) = SIGMA, KSTAR, I, J, WØ, M
                      2 3 4 5 6
                                     T = TIME
                 1
          Q=Ø.
          IF(P(2), LE, Ø, )RETURN
M=IFIX(P(6)+.1)
          IF(P(1).EQ.Ø.)GO TO 4
          AM=M
B=3.*P(1)/AM
          IF(T.GE.EXP(B*P(4))/P(2))RETURN
          C = EXP(-B*P(3))/P(2)
dissolve a DO 1 L=1,M to the "square poot" or "cube root" model. This code
         H=L-1 the treatmontive time sharing program FUNFIT enables
          Ail=11
10.18 to bp=B*AN and by least squares to dissolution rate data and ostimates
IF(T.LT.C)A=SDF(-P(3)-D)
          IF(T.GE.C)A=SDF(ALOG(P(2)*T)/B-D)
       1 Q=Q+FLOAT(NBC(M-1,N))*((-P(2)*T)**(M-N-1))*(SDF(P(4)-D)-A)*
        + EXP(4.5*(AN*AN/AM/AM-1.)*P(1)*P(1))
          Q=P(6)*P(2)*P(5)*Q/(SDF(P(4)-3,*P(1))-SDF(-P(3)-3,*P(1)))
       2 IF(IPRINT.EQ.Ø)RETURN
          WRITE(6,3)P
       3 FORMAT(' SIGMA, KSTAR, 1, J, WØ, M='/6E12.4)
          RETURN
       4 IF(T.GE.1./P(2))RETURN
          Q=P(6)*P(2)*P(5)*((1, -P(2)*T)**([1-1))
          GO TO 2
          END
```

Scheme 10.2

Computer code defining the multiparticulate dissolution models given by 10.18 (m=2,3). The models assume that the initial effective particle size distribution is *ideal* log-normal and that the single particles dissolve according to the "square root" or "cube root" model. This code used together with the interactive time sharing program FUNFIT enables 10.18 to be fitted by least squares to dissolution rate data and estimates of K_m^* and σ to be obtained.

> A-SERFEA*ALOS (P(2)*13-AN/A) 1 0-0-FLOAT (IBC OI-1,N))*(6-P(2)*1)**0+A-1))*(1 +EXP(4.5*(AN*AN/NVAH-3,)*P(1)*P(1)) 0-P(4)*P(2)*P(3)*0 2 IF(1PRUIT.E9.0)RETURN WRITE(6,3)P 3 FORMAT(* SIGNA, #STARLND, D**NE11.4) RETURN 4 IF(T.GE.1./P(2))RETURN 0+P(4)*P(2)*P(3)*((1.-P(2)*T)**05-D)) 50 TO 2 ED0

- 260 -

```
SUBROUTINE MODEL(Q, T, P, IPRINT)
        DIMENSION P(4)
     CCCC
          P(1) = SIGMA, KSTAR, WØ, M = DISSOLUTION RATE 
1 2 3 4 T = TIME
        Q=Ø.
        IF(P(2), LE.Ø.) RETURN
        1=1F1X(P(4)+,1)
        IF(P(1).EQ.Ø.)GO TO 4
        AM=M
        A=0.
        DO 1 L=1, M a single particles dissolve according to the "square
        N=L-1
        AN=N
IF(T.EQ.Ø.)GO TO 1
A=AM/(3.*P(1))
        A=SDF(A*ALOG(P(2)*T)-AN/A)
     1 Q=Q+FLOAT(NBC(M-1,N))*((-P(2)*T)**(M-N-1))*(1.-A)*
        +EXP(4,5*(AN*AN/AM/AM-1,)*P(1)*P(1))
        Q = P(4) * P(2) * P(3) * Q
        IF (IPRINT, EQ, Ø) RETURN
      2
         WRITE(6,3)P
        FORMAT(' SIGMA, KSTAR, WØ, M='4E11.4)
      3
         RETURN
        IF(T,GE,1./P(2))RETURN
      4
         Q=P(4)*P(2)*P(3)*((1,-P(2)*T)**(M-1))
         GO TO 2
         END
```

- 261 -

Scheme 10.3

Computer code defining the multiparticulate dissolution models given by 10.19 (m=2,3). The models assume that the initial effective particle size distribution is log-normal and truncated at the upper end $(i=\infty,j<\infty)$ and that the single particles dissolve according to the "square root" or "cube root" model. The code used together with the interactive time sharing program FUNFIT enables 10.19 to be fitted by least squares to dissolution rate data and estimates of K_m^* , σ and j to be obtained.

> 0-P(S) P(2) POY 0750F (P) (F()P1(H1,EQ,A)ET(H1 WRITE(5,3)P FORM(1(') STORA, ISTAN, J.W

1E(1.0E.1./P(2))RETURN 0-P(5)*P(2)*P(4)*(41.-P(2)*T)** 04 60 TO 2

	SUBROUTINE MODEL(Q,T,P,IPRINT)
	DIMENSION P(5)
	P(I) = SIGMA, KSTAR, J, WØ, M = DISSOLUTION RATE 1 2 3 4 5 T = TIME
	Q=0. butlon waing atthes 5 by 10 by of powdays (fable 10.1); of
	IF(P(2).LE.Ø.)RETURN
	M=IFIX(P(5)+,1)
	IF(P(1).EQ.Ø.)GO TO 4
	AM=M
	B=3.*P(1)/AM
	IF(T.GE.EXP(B*P(3))/P(2))RETURN
	A=Ø.
	DO 1 L=1,M
	N=L-1
	AN=N (10.3
	IF(T.EQ.Ø.)GO TO 1
	A=SDF(ALOG(P(2)*T)/B-B*AN)
1	Q=Q+FLOAT(NBC(M-1,N))*((-P(2)*T)**(M-N-1))*(SDF(P(3)-B*AN)-A)*
	+EXP(4.5*(AN*AN/AM/AM-1.)*P(1)*P(1))
	Q=P(5)*P(2)*P(4)*Q/SDF(P(3)-3.*P(1))
2	IF(IPRINT.EQ.Ø)RETURN
	WRITE(6,3)P
3	FORMAT(' SIGMA, KSTAR, J, WØ, M= '5E10.4)
	RETURN Declares the residuals. The Durbin-Retson
4	IF(T.GE.1./P(2))RETURN
	$\Omega = P(5) * P(2) * P(4) * ((1 - P(2) * T) * * ([1 - 1]))$
	GO TO 2
	END counted by a stand ficant departure from the assessed stauts

8

- 263 -

simplest of the models, containing only the two parameters K^{*} and σ , and fits the dissolution data just as well as the other models that contain more parameters (i and j). Furthermore, it is clear (Table 10.1) that the multiparticulate dissolution model based on the cube root model (m=3) agrees best with the dissolution data. This is true for either a truncated or ideal distribution using either 5 or 10 mg of powder (Table 10.1). Of the 8 multiparticulate models investigated it can therefore be concluded that the following equation:

$$\frac{dW}{dt} = -3K_{3}^{*} W_{0} \sum_{n=0}^{2} {\binom{2}{n}} (-K_{3}^{*}t)^{(2-n)} \left[1 - F\left(\frac{1}{\sigma} \ln(K_{3}^{*}t) - n\sigma\right)\right] e^{\frac{\sigma^{2}}{2} \left[n^{2} - 9\right]}$$
(10.29)

which can be written in integrated form as:

$$\frac{W}{W_{o}} = \sum_{n=0}^{3} {\binom{3}{n}} (-K_{3}^{*}t)^{(3-n)} \left[1 - F\left(\frac{1}{\sigma} \ln(K_{3}^{*}t) - n\sigma\right) \right] e^{\frac{\sigma^{2}}{2} \left[n^{2} - 9 \right]}$$
(10.30)

best describes the dissolution kinetics of the micronized glibenclamide. These two equations uniquely characterise the dissolution behaviour in terms of the rate parameter, K_3^* , and the distribution parameter, σ .

It appears from the residual plots in Figs. 10.2-10.5 that there is a serial correlation between the residuals. The Durbin-Watson statistic indicates ($\alpha < 0.05$) that this correlation is significant (142). Systematic deviation can be caused by non-random experimental errors. It can also be caused by a significant departure from the assumed single particle dissolution model 10.25 or by a deviation from log-normality. The residual values are however so small in relation to the accuracy of the experimental technique that the correlation seems of little importance.

The specific dissolution rate parameter, K_3^* , should theoretically

be independent of the initial amount used, W_{O} . This is only found to be approximately true. Values of the rate parameter, K_{m}^{*} , obtained for $W_{O}^{=10}$ mg are consistently lower than for $W_{O}^{=5}$ mg (Table 10.1). The dissolution models consider dissolution under *complete sink conditions* i.e. conditions where there is no interaction between the dissolving particles. Using 10 mg of the very fine powder it has not been possible to load the dissolution cell with a single "layer" of particles in such a way that dissolved drug from any particle does not pass over other particles. The slightly larger K_{m}^{*} values observed when less powder is used in the cell agree with an expected smaller particle interaction.

Characterisation and quality control of drug powders

Previous approaches to characterizing the dissolution properties of drug powders have been based on equations describing monodisperse systems. In many cases, the so-called "dissolution rate constant" evaluated using such equations will not be sufficient to characterize the dissolution behaviour because the size distribution effect is not accounted for. This is particularly true for pharmaceutical systems which frequently involve highly polydisperse fine powders.

The use of nonlinear regression analysis to evaluate the specific dissolution rate parameter, K_m^* , and the distribution parameter, σ , represents a more exact and meaningful approach.

The properties of K^{*}_m make its interpretation particularly meaningful. These properties are best understood in relation to the concepts of *time scaling* and *the intrinsic dissolution profile* from which the following conclusion can be made:³

^{3.} For example if it takes x min for a powder to dissolve, say 30% for a given K_{m}^{*} value, then it will take the powder x/2 minutes to dissolve to the same extent if the value of K_{m}^{*} is doubled (for illustration see Fig. 6.1).

If K_m^* is changed by a factor α then the time for complete dissolution, or the time for any particular fraction to dissolve, is changed by a factor of 1/ α . This simple property makes K_m^* a particularly useful parameter.

The distribution parameter, σ , which appropriately should be called the dispersion parameter, is a single measure of how polydisperse a powder is, or more exactly, how much the dissolution behaviour deviates from that expected if the powder were completely monodisperse. A value close to zero characterizes a nearly monodisperse powder, while higher values indicate increasing "degrees of dispersion". Probably the most important property of σ is that it is a measure of how long it takes the last fraction of a polydisperse powder to dissolve. For example, it is seen from 10.5 that the time for complete dissolution increases exponentially with σ . For this reason it is most likely that very slightly soluble drugs, which exhibit dissolution rate limited absorption, will show a significant correlation between σ and systemic availability. Research in this area should be of considerable pharmaceutical interest.

Although the multiparticulate dissolution model defining K_m^* and σ may seem complex, the interpretation of these parameters is simple and they can be readily obtained. The experimental technique used requires a high precision, flow-through dissolution apparatus which is easy to standardize in combination with a nonlinear regression program.

The method could well become established as a routine procedure in quality control and further investigation could eventually result in improved standards for drug dissolution.

- 266 -

CHAPTER 11

A METHOD OF OBTAINING DRUG-MACROMOLECULE BINDING PARAMETERS DIRECTLY FROM DYNAMIC DIALYSIS DATA

The dynamic dialysis method for characterizing interactions of small molecules with macromolecules is well established (167-181). It has a number of advantages compared to equilibrium dialysis and ultrafiltration. A complete binding profile can be obtained rapidly in one experiment and the method utilizes only a small sample of macromolecule. As the method is based on a dynamic process, an equilibrium state does not need to be defined and compared to ultrafiltration there is no change in concentration of macromolecule.

Meyer and Guttman (169) designed a dynamic dialysis method to characterize drug-protein interactions. However this method has a number of limitations. The experimental data must be differentiated to evaluate binding parameters. It is recognised in numerical analysis that differentiation of discrete data may introduce substantial errors particularly if the number of data points is limited. An empirical equation was used to fit dialysis data to obtain instantaneous rates. A recent publication (179) has shown that the various empirical equations used can yield substantially different binding parameters.

The technique of Meyer and Guttman (169) requires that the rate constant for dialysis be determined in a separate experiment in the absence of macromolecules. It is assumed that the same rate constant will apply in the presence of macromolecules. This may be an unreasonable assumption as the rate constant depends on several factors such as the physico-chemical

- 267 -

183

state of the dialysis membrane (182,/) which may change between runs.

A number of compounds are significantly bound to the membrane material (171,177). Using previous methods it has not been possible to determine the dialysis rate constant and account for the membrane binding in the determination of macromolecule binding parameters of such compounds.

In this chapter a new approach is presented which rigorously describes the total kinetics of the system in a form that enables binding parameters to be estimated accurately, directly from dialysis data. It eliminates the need to determine an accurate dialysis rate constant in a separate experiment. The method does not rely on differentiation of experimental data and should be applicable to compounds that are membrane bound.

THEORY

Consider an interaction between small molecules and macromolecules which can be described by the general binding expression:

$$\overline{D} = \sum_{i=1}^{j} \frac{n_i K_i D_f}{1 + K_i D_f}$$
(11.1)

where $\bar{\nu}$ is the number of moles of small molecules bound per mole macro molecule, n is the number of binding sites in the i-th class of sites, K_{i} is the association constant for the interaction and D_{f} is the molar concentration of unbound small molecules. If a model with two classes (j=2) is assumed then the total concentration of drug, D_{t} , in the protein compartment is given by (171):

$$D_{t} = D_{f} + P_{t}D_{f} \left[\frac{n_{1}K_{1}}{1+K_{1}D_{f}} + \frac{n_{2}K_{2}}{1+K_{2}D_{f}} \right]$$
(11.2)

If sink conditions prevail the small molecules will leave the

protein compartment by a first order process:

$$\frac{dD_t}{dt} = -K_e D_f$$
(11.3)

where K_e is the dialysis rate constant.

dt dt

It is convenient to introduce a variable, s, defined as:

$$= - \frac{dD_t}{dt}$$
(11.4)

so that $D_f = s/K_e$ and Eq. 11.6 can be written:

$$D_{t} = \frac{s}{K_{e}} + P_{t}s \left[\frac{n_{1}K_{1}}{K_{e} + K_{1}s} + \frac{n_{2}K_{2}}{K_{e} + K_{2}s} \right]$$
(11.5)

Taking the differential of this equation, noting $dD_t = -sdt$, it becomes

$$= -\left[\frac{1}{K_{e}s} + P_{t}K_{e}\left(\frac{n_{1}K_{1}}{s(K_{e}+K_{1}s)^{2}} + \frac{n_{2}K_{2}}{s(K_{e}+K_{2}s)^{2}}\right)\right]ds$$
(11.6)

which integrated from t=0 to t corresponding to $s = s_0$ to s yields:

$$t = \left| \frac{1}{K_{e}} \ln s + P_{t} n_{1} K_{1} \left[\frac{1}{K_{e} + K_{1} s} + \frac{1}{K_{e}} \ln \left(\frac{s}{K_{e} + K_{1} s} \right) \right] (11.7) + P_{t} n_{2} K_{2} \left[\frac{1}{K_{e} + K_{2} s} + \frac{1}{K_{e}} \ln \left(\frac{s}{K_{e} + K_{2} s} \right) \right] \right|_{s}^{s_{o}}$$

If the following function is defined:

$$f(x) = \frac{1}{K_{e}} \ln x + P_{t} n_{1} K_{1} \left[\frac{1}{K_{e} + K_{1} x} + \frac{1}{K_{e}} \ln \left(\frac{x}{K_{e} + K_{1} x} \right) + P_{t} n_{2} K_{2} \left[\frac{1}{K_{e} + K_{2} x} + \frac{1}{K_{e}} \ln \left(\frac{x}{K_{e} + K_{2} x} \right) \right]$$
(11.8)

then Eq. 11.7 can be written more simply as:

- 269 -

The exact functional relationship describing the change of D_t with t is now described by Eq's 11.5 and 11.9 in *parametric form* where the variable s is the parameter. Each value of s defines by these equations a *unique* pair of D_t and t values.

The quantity s_0 is the initial (t=0) value of $-dD_t/dt$ (Eq. 11.4), which would normally be determined by extrapolation. To avoid the errors and problems of such an extrapolation *it is convenient to define t=0* at the first sampling time. In this way s_0 is $-dD_t/dt$ at first sampling.

In order to determine the binding parameters by nonlinear regression it is necessary to define the exact functional relationship between D_t and t, for any values of n_1 , K_1 , n_2 , K_2 , s_0 and K_e which are changing during the nonlinear fitting procedure. This can be done by determining the particular values of s which satisfy Eq. 11.9. These values are then used to determine the corresponding values of D_t by Eq. 11.5.

However Eq. 11.9 cannot be expressed explicitly in terms of s so some iterative procedure is needed to solve for s. The Newton-Raphson algorithm is particularly suitable because it is computationally compact and exhibits quadratic convergence.

If

$$\phi(s) = t - f(s_{-}) + f(s)$$
(11.10)

then Eq. 11.9 can be solved by the Newton-Raphson method using the following iteration¹:

$$s_{i+1} = s_{i} - \frac{\phi(s_{i})}{\phi'(s_{i})}$$
(11.11)

where the functions φ and φ' are defined by:

1. The function ϕ is not defined for $s \le 0$ so it is necessary during the iteration procedure to prevent s from taking a nonpositive value. This is conveniently done by defining $s_{i+1} = s_i/2$ if $s_{i+1} \le 0$ since ϕ is a monotone increasing function of s because $\phi' = f' > 0$ (s>0).

- 270 -

$$\phi'(s) = f'(s) = \frac{1}{K_e s} + \frac{P_t K_e}{s} \left[\frac{n_1 K_1}{(K_e + K_1 s)^2} + \frac{n_2 K_2}{(K_e + K_2 s)^2} \right] \quad (11.12)$$

$$\phi(s) = t + \frac{1}{K_e} \ln \frac{s}{s_o} + P_t n_1 K_1 \left[\frac{K_1 (s_o - s)}{(K_e + K_1 s) (K_e + K_1 s_o)} + \frac{1}{K_e} \ln \frac{s}{s_o} \frac{(K_e + K_1 s_o)}{(K_e + K_1 s)} \right] \quad (11.13)$$

$${}^{P}t^{n}2^{K}2\left[\frac{K_{2}(s_{0}-s)}{(K_{e}+K_{2}s)(K_{e}+K_{2}s_{0})} + \frac{1}{K_{e}}\ln\frac{s}{s_{0}}\frac{(K_{e}+K_{2}s_{0})}{(K_{e}+K_{2}s)}\right]$$

In order for the algorithm (11.11) to converge within a given number of iterations it is necessary that the initial estimates of s are not too far from the s values for which 11.9 is satisfied. These initial estimates can, according to 11.4, be obtained as the (positive) values of $-dD_t/dt$ estimated from the observed values of D_t vs. t. Any simple technique for slope estimation can be used since the actual accuracy of the estimates is of no importance for the final result.

The above derivations for 2 binding classes can easily be extended to any number (j) of classes for which then:

$$D_{t} = \frac{s}{K_{e}} + P_{t}s \sum_{i=1}^{j} \frac{n_{i}K_{i}}{K_{e}+K_{i}s}$$
(11.14)
$$t = \left| \frac{1}{K_{e}} \ln s + P_{t} \sum_{i=1}^{j} n_{i}K_{i} \left[\frac{1}{K_{e}+K_{i}s} + \frac{1}{K_{e}} \ln \left(\frac{s}{K_{e}+K_{i}s}\right) \right] \right|_{s}^{s_{o}}$$

(11.15)

and the iteration, 11.11, can still be used with

$$\phi'(s) = \frac{1}{K_{e}s} + \frac{P_{t}K_{e}}{s} \sum_{i=1}^{j} \frac{n_{i}K_{i}}{(K_{e}+K_{i}s)^{2}}$$
(11.16)

and to be is the amount of small solecules bound por another of available

Binding of Small Hole

$$p(s) = t + \frac{1}{K_{e}} \ln \frac{s}{s_{o}}$$

$$+ P_{t} \sum_{i=1}^{j} n_{i}K_{i} \left[\frac{K_{i}(s_{o}-s)}{(K_{e}+K_{i}s)(K_{e}+K_{i}s_{o})} + \frac{1}{K_{e}} \ln \frac{s(K_{e}+K_{i}s_{o})}{s_{o}(K_{e}+K_{i}s)} \right]$$
(11.17)

Binding of Small Molecules by the Dialysis Membrane

The treatment outlined above is based on an assumption that binding occurs only to protein. However some drugs, particularly those strongly protein bound can become significantly bound to the dialysis membrane. This means that K_e cannot be estimated from plots of $\ln D_t$ vs. t in the absence of protein using previous techniques because of curvature (Fig. 11.1).

A special technique is therefore required to estimate the dialysis rate constant and the membrane binding must be taken into account in the treatment of the dialysis behaviour of the small molecule-macromolecule system.

Determination of K_e and membrane binding parameters in absence of macromolecule.

The binding of small molecules to the membrane can often be considered as a Langmuir type absorption phenomenon (171), which is mathematically analogous to binding to a single class of sites, and can be described by the equation:

$$\overline{\nu}^* = \frac{n^* K^* D_f}{1 + K^* D_f}$$
(11.18)

where \bar{v}^* is the amount of small molecules bound per amount of available membrane material and K* is the association constant for membrane binding.

Figure 11.1

Dynamic dialysis of glibenclamide in 0.067M phosphate buffer in the absence of macromolecules at pH 7.4 and 37° . The tangents illustrate a difference in the slope at t=0 and t=3 hours of 15%.

- 274 -3 2 HOURS e and V is the volume of the pr 11.20 can then be used to determin C 5.5 4.5 - 5.0 - 4.0 MG/ML Ln Dt

This leads to the following relationship between D, and t similar to 11.14 and 11.15 (j=1):

t

$$D_{t} = \frac{s}{K_{e}} + \frac{k's}{K_{e} + K^{*}s}$$
(11.19)

$$= \frac{1}{K_{e}} \ln s + \frac{K'}{K_{e} + K^{*}s} + \frac{K'}{K_{e}} \ln \frac{s}{K_{e} + K^{*}s} \int_{s}^{s_{o}} (11.20)$$

Mn*K* (11.21)K' where:

and M is the amount of membrane material available for binding, W is the molecular weight of the small molecule and V is the volume of the protein compartment. Equations 11.19 and 11.20 can then be used to determine K and the membrane binding parameters K* and K' in the absence of protein using the technique described above.

Determination of drug-macromolecule binding parameters in the presence of membrane binding.

Once the parameters K, K' and K* have been determined using the above approach it is possible to account for membrane binding and determine parameters for binding to the macromolecule.

Simultaneous binding to the membrane and the macromolecule leads to the following expressions:

$$D_{t} = \frac{K's}{K_{e}+K^{*}s} + \frac{s}{K_{e}} + P_{t}s \sum_{i=1}^{i} \frac{n_{i}K_{i}}{K_{e}+K_{i}s}$$
(11.22)

$$t = \left| \frac{K'}{K_{e}+K^{*}s} + \frac{K'}{K_{e}} \ln \frac{s}{K_{e}+K^{*}s} \right|$$
(11.23)

$$+ \frac{1}{K_{e}}\ln s + P_{t} \sum_{i=1}^{j} n_{i}K_{i} \left[\frac{1}{K_{e}+K_{i}s} + \frac{1}{K_{e}}\ln (\frac{s}{K_{e}+K_{i}s}) \right] \right|_{s}^{s_{0}}$$

10

- 275 -

which enables the D_t, t functional relationship to be evaluated using the iterative procedure discussed.

By comparing Eqs. 11.22 and 11.23 with Eqs. 11.14 and 11.15 it is seen that in the presence of membrane binding the dialysis behaviour is mathematically analogous to a system where the small molecule is binding to two macromolecular species.

DATA TREATMENT AND RESULTS

Data for dialysis of chlorpropamide from bovine serum albumin (BSA) in the presence of 1.6 x 10^{-5} M free warfarin were treated according to the proposed method. Binding parameters n_1 , K_1 , n_2 , K_2 and the dialysis rate constant K_e were estimated using the function relating D_t and t given in Eqs. 11.5 and 11.9 and evaluated using Eqs. 11.11-11.13 and 11.5. This function was programmed in a subroutine (scheme 11.1) which was executed with FUNFIT.

The least squares fit of the model using two classes of binding sites (Fig. 11.2) agreed very well with the experimental data (r = 0.99986). The binding parameters estimated by the method are summarised in Table 11.1. The values for the number of binding sites n_1 and n_2 were found to be so close to 2 and 9 in a preliminary computation that their values were fixed as integers. Such a model has greater conceptual value and the slight change in n_1 and n_2 did not substantially alter the values of the other parameters.

Also included in Table 11.1 are values for binding parameters estimated by fitting dialysis data to a fourth order polynomial and then evaluation using a modification of the method of Hart (184) as described elsewhere (175). This method will be denoted derivative method I. A Scheme 11.1

The subroutine MODEL defines the functional relationship between D, and t in a dynamic dialysis process. This function, given in parametric form by Eqs. 11.5 and 11.9, is calculated according to Eqs. 11.11-11.13 and 11.5. The functions PHI and PHIDER used in MODEL defines $\phi(s)$ and $\phi'(s)$ as given by Eqs. 11.13 and 11.12. The subroutine which is

written specifically for FUNFIT enables least squares estimates of the binding parameters n1, K1, n2, K2 (Eq. 11.1) and the dialysis rate constant K (Eq. 11.3) to be determined directly from dynamic dialysis data (Fig. 11.2). The subroutine specifies s as a second "independent variable", x(2), which is used as an initial value in the iteration, Eq. 11.11, that in less than MAX cycles should determine s to an accuracy corresponding to an accuracy of t better than 10^{-6} %. The last two parameters (P(7) = total macromolecule concentration and P(8) = MAX) of the eight formal parameters are fixed during the least squares fitting procedure. A value of 50 for

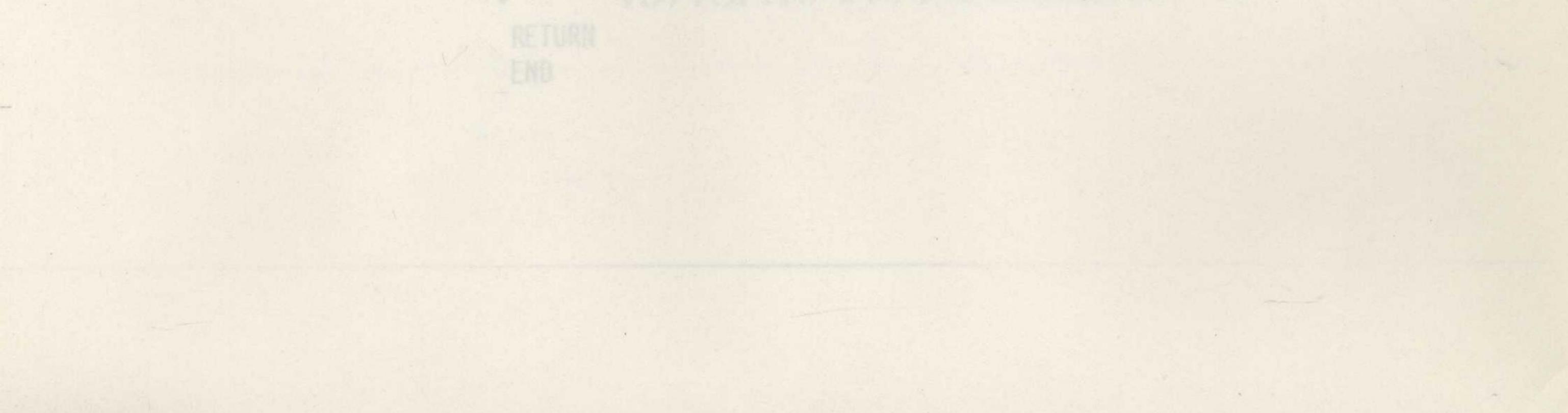
MAX should be sufficient to reach convergence (Eq. 11.11) even for pure

starting values of P(1) to P(6). However if convergence is not reached

IPRINT is made equal to 1, the current values of the parameters are printed

and the control is returned to the main program so that appropriate action

can be taken.



```
- 278 -
                      SUBROUTINE MODEL (DT, X, P, IPRINT)
                      DIMENSION X(2), P(8)
                  C
                                             DT = TOTAL CONC. OF SMALL MOLECULES
                  C
                    P=N1,K1,N2,K2,SØ,KE,PT,MAX
                                              \chi(1) = TIME
                     1 2 3 4 5 6 7 8
                  C
                                              \chi(2) = SLOPE ESTIMATE
                  C
                  C
                       S=X(2)
                       MAX=IFIX(P(8))
                       DO 1 I=1, MAX
                       SSAVE=S
                       PHIVAL=PHI(S,P,X(1))
                       S=S-PHIVAL/PHIDER(S, P)
                       IF(S.LE.Ø.)S=SSAVE/2.
                       TIME = X(1)
                       IF(TIME.EQ.Ø.)TIME=1.
                       IF(ABS(PHIVAL)/TIME.LT.1E-8) GO TO 2
                     1 CONTINUE
                       IPRINT=1
                     2 DT=S/P(6) + P(7)*S*(P(1)*P(2)/(P(6)+P(2)*S)+
                                        P(3)*P(4)/(P(6)+P(4)*S))
           Bynamic ditionis date to
                       IF(IPRINT.EQ.Ø)RETURN
WRITE(6,3)P
                     3 FORMAT(' N1, K1, N2, K2, SØ, KE, PT, MAX =', (4E11.5))
RETURN
                       END
                   С
                   C
                       FUNCTION PHIDER(S,P)
                       DIMENSION P(8)
                       PHIDER=1./(P(6)*S) + (P(7)*P(6)/S)*(P(1)*P(2)/((P(6)+P(2)*S)**2)+
                                                       P(3)*P(4)/((P(6)+P(4)*S)**2))
                      +
                       RETURN
                       END
                   С
```

С

```
FUNCTION PHI(S,P,T)

DIMENSION P(8)

A=ALOG(S/P(5))/P(6)

B=P(5)-S

C=P(6)+P(2)*P(5)

D=P(6)+P(2)*S

E=P(6)+P(4)*P(5)

F=P(6)+P(4)*S

PHI=T+A+P(7)*P(1)*P(2)*(P(2)*B/(C*D)+A+ALOG(C/D)/P(6))+

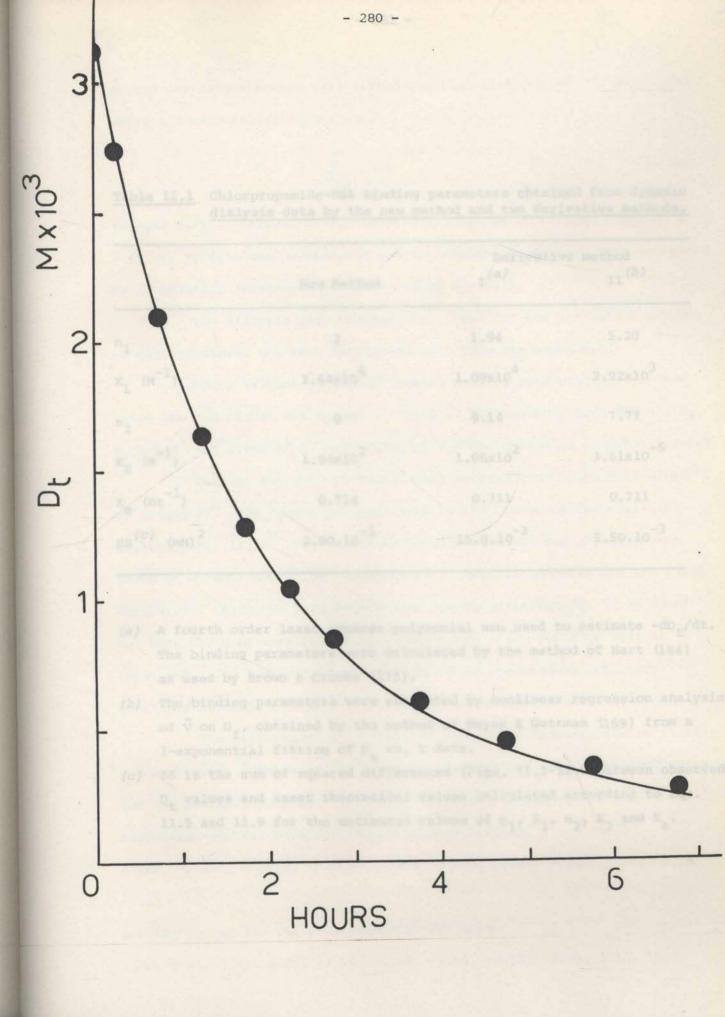
+ P(7)*P(3)*P(4)*(P(4)*B/(E*F)+A+ALOG(E/F)/P(6))

RETURN

END
```

Figure 11.2

Dynamic dialysis data for the binding of chlorpropamide to 1% BSA at pH 7.4 and 37[°] in the presence of 1.6 x 10^{-5} M free warfarin. The curve fitted by least squares is the function relating D_t and t (Eqs. 11.5 and 11.9) which provides estimates of n₁, K₁, n₂, K₂ and K_e.



V val D, profile was	constructed and A	Derivativ	
	New Method	I ^(a)	II ^(b)
n ₁	2	1.94	5.28
κ ₁ (M ⁻¹)	2.64×10 ⁴	1.09x10 ⁴	2.92x10 ³
n2 me he her method	9	9.14	7.77
к ₂ (м ⁻¹)	1.94×10 ²	1.86x10 ²	3.61x10 ⁻⁵
K _e (Hr ⁻¹)	0.714	0.711	0.711
$ss^{(c)}$ (mM) ²	2.60.10 ⁻³	15.0.10-3	5.50.10 ⁻³

Table 11.1 Chlorpropamide-BSA binding parameters obtained from dynamic dialysis data by the new method and two derivative methods.

- (a) A fourth order least squares polynomial was used to estimate -dD_t/dt. The binding parameters were calculated by the method of Hart (184) as used by Brown & Crooks (175).
- (b) The binding parameters were estimated by nonlinear regression analysis of $\bar{\nu}$ on D_f , obtained by the method of Meyer & Guttman (169) from a 3-exponential fitting of D_+ vs. t data.
- (c) SS is the sum of squared differences (Figs. 11.3-11.5) between observed D_t values and exact theoretical values calculated according to Eqs. 11.5 and 11.9 for the estimated values of n₁, K₁, n₂, K₂ and K_e.

second derivative method (II) tested was that of Meyer and Guttman (169) where a 3-exponential expression:

$$D_{t} = \sum_{i=1}^{3} A_{i}e^{-\alpha_{i}t}$$
(11.24)

is used to fit dialysis data. By differentiation at various t values a $\bar{\nu}$ vs. D_{f} profile was constructed and the binding parameters were estimated by a nonlinear regression technique using Eq. 11.1.

The dialysis rate constant, K_e , used for the derivative methods of data treatment was that determined in a separate experiment.

There is good agreement between binding parameters determined using the new method and method I. This is particularly true for n_1 , n_2 , K_2 and K_2 and although K_1 determined by the new method is somewhat greater.

However entirely different binding parameters were determined by method II. The reason probably lies in the choice of the empirical equation, Eq. 11.24. The selection of this equation has probably been based on a requirement for "smoothness" of the fitted curve and its first derivative. Although this requirement may be satisfied, Eq. 11.24 does not have the same flexibility as a polynomial which could be just as important. The flexibility of polynomials to approximate arbitrary functions is explained by the well-known Taylor series theorem.

The difference in flexibility is clearly demonstrated by fitting a fourth order polynomial and 11.24 to exact dialysis data obtained using the new method (Table 11.1). The D_t residuals (expressed as % of calculated values) were for the polynomial fitting: 0.286, -0.217, -0.519, -0.0609, 0.466, 0.779, 0.381, -1.49, -1.42, 3.14, -1.12

and the values for the triexponential fit were: 1.73, 1.91. 1.46, 3.38, -1.19, -2.65, -3.91, -4.55, -2.19, 3.10, 10.7. This shows that the polynomial is considerably more flexible. The triexponential fitting resulted in significant systematic deviation in residuals leading to bias in the slope values and D_t values and therefore a bias in the final results. The residual sum of squares (mM²) were 5.85.10⁻⁴ and 1.04.10⁻² for the polynomial and triexponential fitting respectively.

A further disadvantage of using 11.24 is that multiple solutions are possible because this equation is nonlinear (in α) and therefore may result in several sum-of-squares minima. This is not the case with a polynomial which has a *unique* least squares solution.

The fitting of 11.24 to the generated, exact D_t , t data was repeated several times with different initial estimates for A_i and α_i but the same solution was obtained each time suggesting that the above fit is the best possible using the triexponential equation.

The use of a lease squares polynomial to represent dialysis data, is on the other hand, expected to be less suitable than the triexponential when the experimental errors are large or where there are significant 'gaps' between observation points. This is due to the fact that the ordinary least squares polynomial fitting completely disregards derivative values. The derivative values are commonly found to be in large error at first and last observation points and just before or after 'gaps' in the data. This disadvantage of polynomials can be reduced considerably by imposing constraints on the derivative values by using least squares spline polynomials.

Such polynomials will compete favourably with 11.24 on data with large errors and 'gaps' particularly considering the fact that the problem of multiple minima, using 11.24, is much larger for large residual problems.

- 283 -

However, no matter which empirical equation is used the results obtained will theoretically never be as exact as obtained using the *true* equation as in fact done in the proposed method.

To investigate the bias introduced by using method I and II the exact D_t, t profile was calculated using the parameter values (Table 11.1) obtained using the two methods. The differences between the observed D_t values and the calculated values (Figs. 11.4 and 11.5) shows that the residuals are significantly larger in those methods than in the new method (Fig. 11.3). The residuals from method I are particularly biased in a positive direction although their pattern resembles that from the new method (Fig. 11.3).

On the other hand the residuals from method II (Fig. 11.5) show an entirely different pattern consistent with the fact that the binding parameters obtained using method II represent an entirely different solution.

A final check on the bias introduced by method I and II was made by applying them to *exact* D_t, t data generated from the parameter values (Table 11.1) obtained using the new method.

Method I found binding parameters relatively close to the true values although K_1 seems to be somewhat different (Table 11.2). Method II found however an entirely different solution that is similar to the solution obtained using the real experimental data.

The value for the dialysis rate constant, K_e , estimated by the new method agrees very well with the value determined experimentally (Table 11.1). As stated previously this may not always be true as the permeability of the membrane may change between experiments.

Chlorpropamide does not appear to be significantly membrane bound.

- 284 -

Figure 11.3

Differences between observed D_t values and exact theoretical values calculated according to Eqs. 11.5 and 11.9 for the values of n_1 , K_1 , n_2 , K_2 and K_e (Table 11.1) estimated by the proposed method.

- 285 -

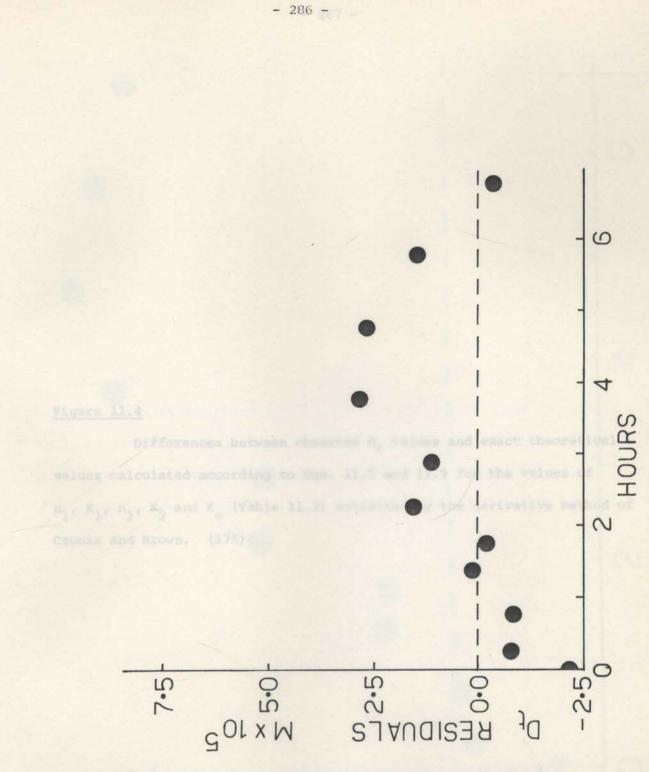
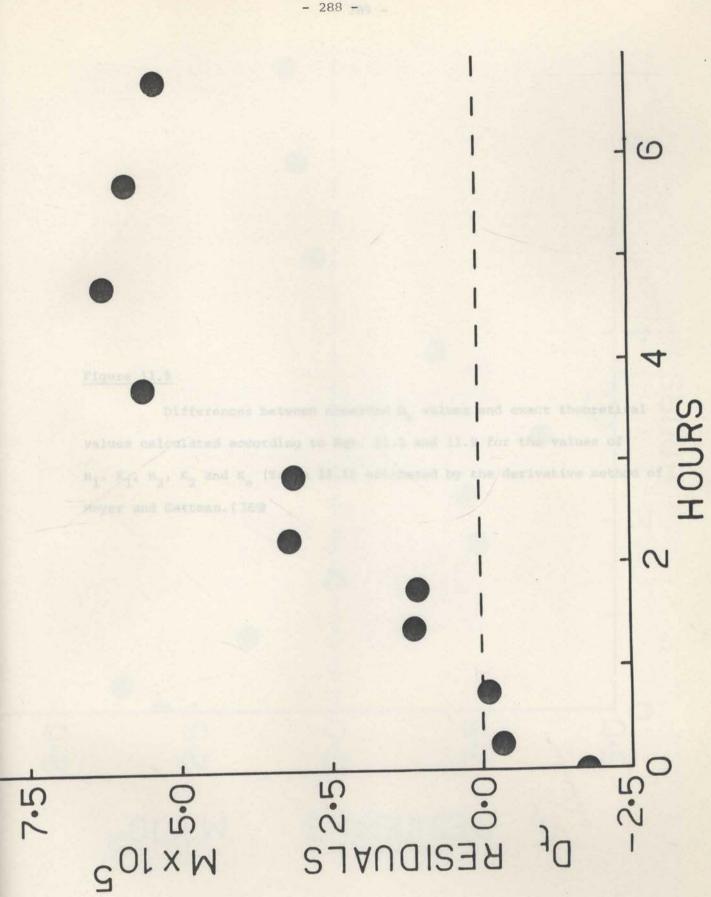


Figure 11.4

Differences between observed D_t values and exact theoretical values calculated according to Eqs. 11.5 and 11.9 for the values of n_1 , K_1 , n_2 , K_2 and K_e (Table 11.1) estimated by the derivative method of Crooks and Brown. (175)



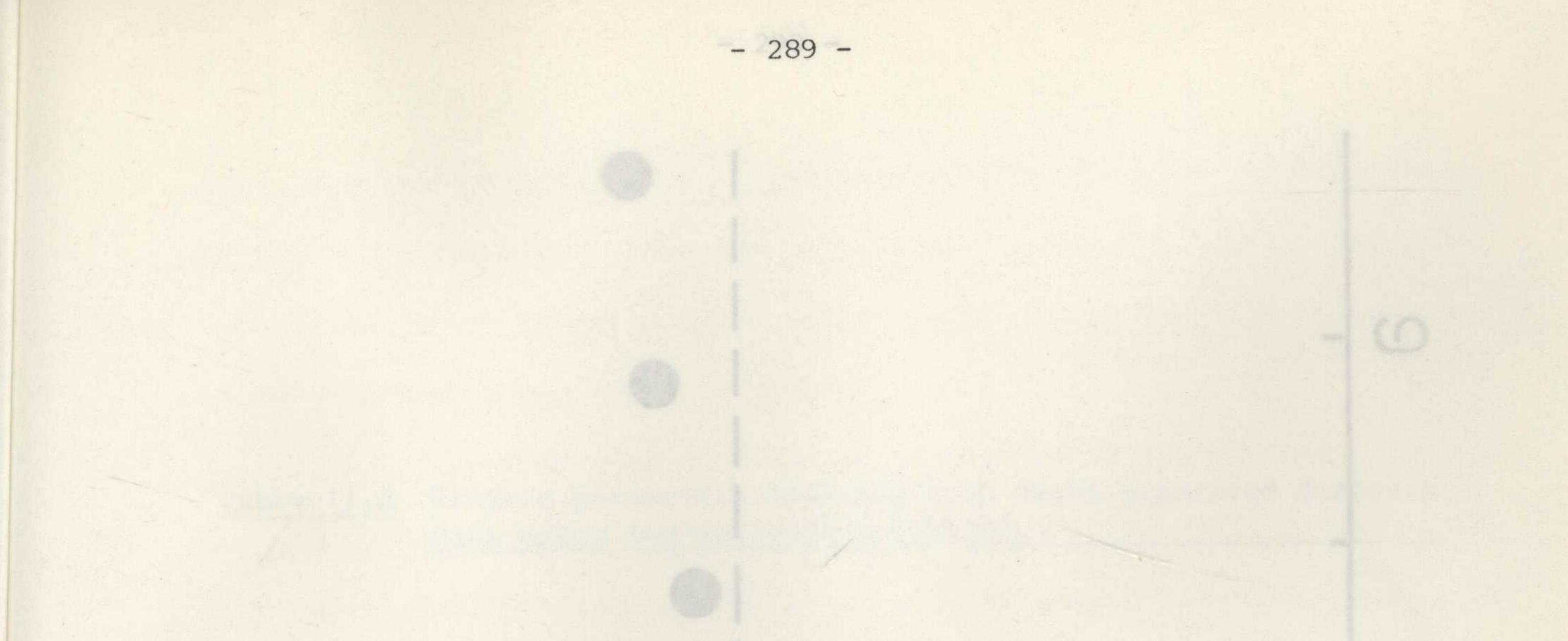
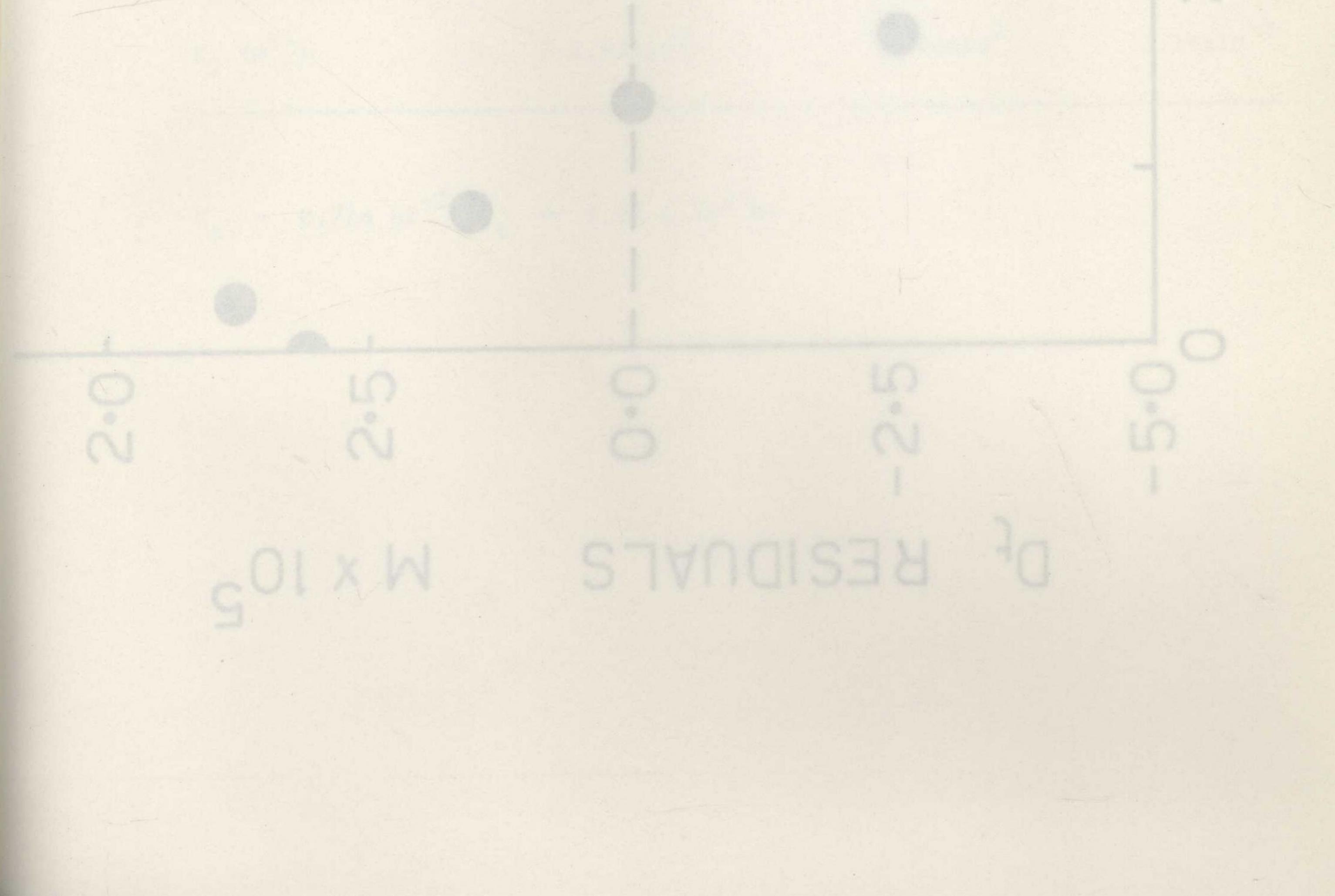
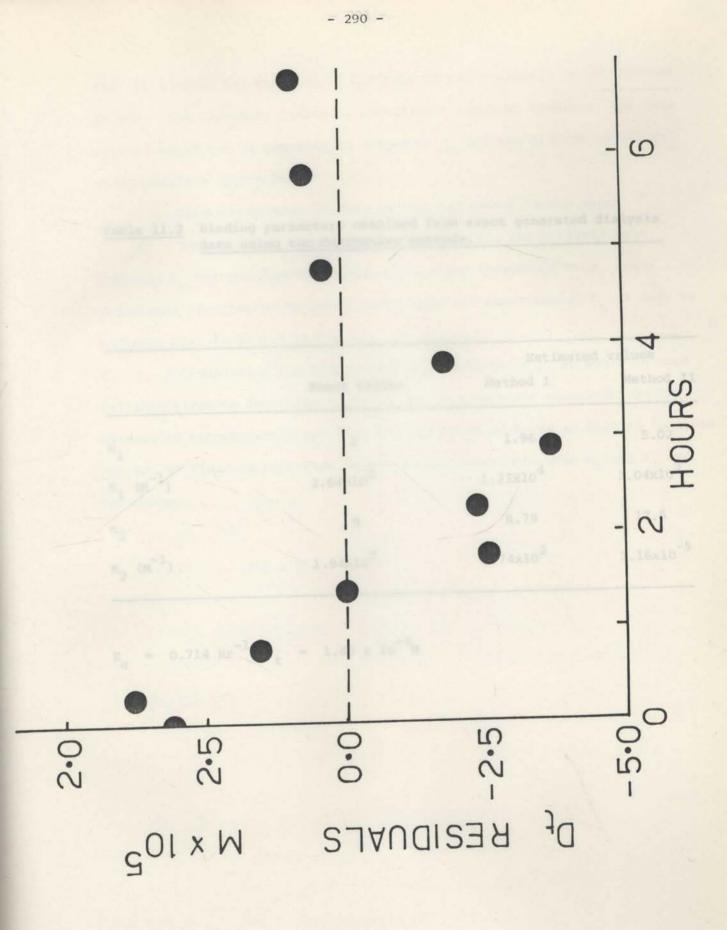
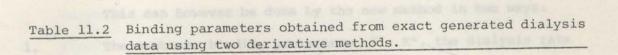


Figure 11.5

Differences between observed D_t values and exact theoretical values calculated according to Eqs. 11.5 and 11.9 for the values of n_1 , K_1 , n_2 , K_2 and K_e (Table 11.1) estimated by the derivative method of Meyer and Guttman.(169)







applying Sept. 11.42		Estimated values			
	Exact values	Method I	Method II		
n n n n n n n n n n n n n n n n n n n	ection and the state	1.96	5.02		
к ₁ (м ⁻¹)	2.64×10 ⁴	1.21×10 ⁴	3.04x10 ³		
n ₂	9	8.78	12.5		
к ₂ (м ⁻¹)	1.94x10 ²	1.74x10 ²	3.16x10 ⁻⁵		

 $K_e = 0.714 \text{ Hr}^{-1}, P_t = 1.45 \times 10^{-4} \text{M}$

Fig. 11.1 shows the kinetics of dialysis of glibenclamide in the absence of BSA. The curvature indicates significant membrane binding. For this drug it would not be possible to determine K_e and the binding parameters using previous approaches.

This can however be done by the new method in two ways:

1. The membrane binding parameters, K', K*, the dialysis rate constant K_e and the drug-macromolecule binding parameters n_i, K_i can all be determined simultaneously and directly from the experimental D_t , t data by applying Eqs. 11.22 and 11.23.

2. Considering the many parameters involved in 1. it would be more reliable first to determine K_{e} , K' and K* in a separate experiment in the absence of macromolecule and then use the value of K_{e} as an initial estimate and K', K* fixed as constants in the second experiment when n_{i} and K_{i} are determined.

Malson, R., 1010., 46, 607 (1987).

S. Shalfar, E. and Highohi, T., 1010., H. THI IIPON.

16. Brudney, M., Frewart, D.J. and Putters, B.T., Con. Hed. Asroc. J., 90, 960 (1976).

17. Lavy, G. and Procknall, J.A., J. Pharm. Bcl., 53, 656 (1964).

18. La, P.C., Mice. S.R. and Maineille, C.M., Con. Med. Assoc. J., 57.

19. Love, G. and Sabil, B.K., J. Cheven, Pols, 51, 52 (1982).

20. Devy, G. and Tenski, W., Mid., 51, 571 (1964).

21. Wood, d.H., Svarto, d.W. and Laberwan, S., 5018. 14, 1065 (1965).

22. Valkoff, H. and Drum, G., 1810., 54, 1893 (1965).

BIBLIOGRAPHY

1.	Wagner, J.G., <u>J. Pharm. Sci.</u> , <u>50</u> , 359 (1961)
2.	Fincher, J.H., <u>ibid.</u> , <u>57</u> , 1825 (1968)
3.	Wurster, D.E. and Taylor, P.W., <u>ibid.</u> , <u>54</u> , 169 (1965).
4.	Higuchi, W.I., <u>ibid.</u> , <u>56</u> , 315 (1967)
5.	Wagner, J.G., Drug Intelligence Clin. Pharm., 4, 17 (1970).
6.	Hersey, J.A., Mfg. Chemist Aerosol News, 40, 32 (1969).
7.	Baun, D.C. and Walker, <u>G.C., J. Pharm. Sci.</u> , <u>58</u> , 611 (1969).
8.	Hersey, J.A. and Marty, J.J., Manuf. Chemist, 46, 43 (1975).
9.	Groves, M.J., <u>ibid</u> ., 46, 37 (1975).
10.	Dare, J.G., <u>Australian J. Pharm.</u> , <u>45</u> , 5 (1967).
11.	Levy, G. and Hayes, B.A., <u>New Engl. J. Med.</u> , <u>262</u> , 1053 (1960).
12.	Parrot, E.L., Wurster, D.E. and Higuchi, T., J. Pharm. Sci., 44,
	269-272 (1955).
13.	Nelson, E., <u>ibid.</u> , <u>46</u> , 607 (1957).
14.	Mitchell, A.G. and Saville, D.J., <u>ibid.</u> , <u>19</u> , 729 (1967)
15.	Shelfer, E. and Higuchi, T., ibid., 52, 781 (1963).
16.	Brudney, N., Stewart, D.J. and Eustace, B.T., Can. Med. Assoc. J.,
	<u>90</u> , 980 (1976).
17.	Levy, G. and Procknall, J.A., <u>J. Pharm. Sci.</u> , <u>53</u> , 656 (1964).
18.	Lu, F.C., Rice, W.B. and Mainville, C.W., Can. Med. Assoc. J., 92,
	1166 (1965).
19.	Levy, G. and Sahli, B.A., <u>J. Pharm. Sci.</u> , <u>51</u> , 58 (1962).
20.	Levy, G. and Tauski, W., <u>ibid.</u> , <u>53</u> , 679 (1964).
21.	Wood, J.H., Syarto, J.E. and Leherman, H., <u>ibid.</u> , <u>54</u> , 1068 (1965).
22.	Paikoff, M. and Drum, G., <u>ibid.</u> , <u>54</u> , 1693 (1965).

 Cleverly, B. and Williams, P.P., <u>Chem. & Ind. London</u>, Jan. 10, p. 49 (through Wagner J.G., <u>J. Pharm. Sci.</u>, <u>50</u>, 359 (1961)).
 Klein, L., <u>Bull. Biol. Pharm.</u>, 273 (1932).
 Elliott, G.H., <u>Pharm. Z.</u>, <u>131</u>, 514 (1933).
 Gibaldi, M. and Feldman, S., <u>J. Pharm. Sci.</u>, <u>56</u>, 1238 (1967).
 Hamlin, W.E., Nelson, E., Ballard, B.E. and Wagner, J.G., <u>ibid.</u>, <u>51</u>,

432 (1962).

- 28. Schroeter, L.C. & Wagner, J.G., ibid., 51, 957 (1962).
- 29. Langenbucher, F., ibid., 58, 1265 (1969).
- 30. Tingstad, J.E. and Riegelman, S., ibid., 59, 692 (1970).
- 31. Tingstad, J.E., Gropper, E., Lachman, L. and Shami, E., <u>ibid.</u>, <u>61</u>, 1985 (1972).
- Needham, T.E., Luzzi, L.A. and Mason, W.D., <u>ibid.</u>, <u>62</u>, 1860 (1973).
 Lapidus, H. and Lordi, N.G., <u>ibid.</u>, <u>55</u>, 840 (1966).
- 34. Fatton, R., Fielding, D., Khanna, S.C. and Richler, A., Pharm. Acta. Helv.,

47, 487 (1971).

35. Crooks, M.J., Ph.D. thesis, Sydney University (1973).

36. Meyer, M.C. and Guttman, D.E., J. Pharm. Sci., 57, 1627 (1968).

37. Brown, K.F. and Crooks, M.J., Biochem. Pharmacol., 25, 1175, (1976).

38. Veng Pedersen, P., M.Sc. thesis, Sydney University (1975).

39. Barr, W.H., Drug Inf. Bull., 3, 27-45 (1969).

40. Castillo, R.A. & Grim, W.M., Paper presented to the Industrial

Pharmaceutical Technology Section, AphA Academy of Pharmaceutical

Science, Las Vegas Meeting, April, 1967.

41. Dare, J.G., Australian J. Pharm., 45, 5 (1967).

42. Hunt, J.N. & McDonald, I., J. Physiol. 126, 459 (1954).

43. Hunt, J.N. & Spurrel, W.R., ibid., 113, 159 (1951).

- Levy, G. "Physio-chemical Aspects of Drug Actions" Pergamon Press, Oxford, New York (1968).
- 45. Wagner, J.G., "Biopharmaceutics and Relevant Pharmacokinetics" Drug Intelligence publications, Illinois (1970).
- 46. Hixson, A. & Baum, S., Ind. Eng. Chem., 33, 478 (1941).
- 47. ibid., 34, 120 (1942).
- 48. Abramson, M. & King, C.V., J. Am. Chem. Soc. 61, 2290 (1939).
- 49. Levy, G. & Jusko, W., J. Pharm. Sci., 54, 219 (1965).
- 50. Card, W.I., <u>Am. J. Digest. Dis.</u>, <u>8</u>, 47 (1941).
- 51. Noyes, A. & Whitney, W., Z. Physik. Chem., 23, 689 (1897).
- 52. Levy, G., J. Pharm. Sci., 52, 1039 (1963).
- 53. Wagner, C., J. Phys. Chem., <u>53</u>, 1030 (1949).
- 54. Roehl, E., King, C.V. & Kipness, S., J. Am. Chem. Soc., 61, 2290 (1939).
- 55. Roller, O., J. Phys. Chem., 39, 221 (1935).
- 56. Gibaldi, M., "Introduction to Biopharmaceutics" Lea & Febiger, Philadelphia (1971).
- 57. Finholt, P. & Solvang, S., J. Pharm. Sci., 57, 1322 (1968).
- 58. Elworthy, P.H. & Lipscombe, F.J., J. Pharm. Pharmacol., 20, 923 (1968).
- 59. Walter, V., ibid., 20, suppl. 228s (1968).
- Higuchi, W.I., Prakonpau, S., Surpuriya, V. & Young, F. <u>Nature</u>, <u>178</u>, 633 (1972).
- 61. Krondi, A., J. Can. Med. Assoc., 29, 360 (1970).
- 62. Bates, T.R., Gibaldi, M. & Kanig, J.L., J. Pharm. Sci., 55, 191 (1966).
- 63. Bates, T.R., Lin, S.L. & Gibaldi, M., ibid., 56, 1492 (1967).
- 64. Thakker, A.Z. & Hall, N.A., ibid., 58, 68 (1969).
- 65. Higuchi, W.I., Lau, P.K., Higuchi, T. & Shell, J.W., ibid., 150 (1963).
- 66. Leonards, J.R., Clin. Pharmacol. Therap., 4, 476 (1963).

- 67. Higuchi, W.I., Nelson, E. & Wagner, J.G., J. Pharm. Sci., 53, 333 (1964).
- 66. Smolen, V.F. & Kildsig, D.O., ibid., 60, 130 (1971).
- 67. Mullins, J.D. & Macek, T.J., J. Am. Pharm. Assoc., 49, 245 (1960).
- 68. Alimirante, L., De Carneri, I., & Coppi, G., Ill. Pharmaco. Ed. Prat.,
 - 15, 471 (1960).
- 69. Taylor, P.W. & Wurster, D.E., J. Pharm. Sci., 54, 1654 (1965).
- 70. Higuchi, W.I., Bernado, P.D. & Mehta, S.C., ibid., 56, 200 (1967).
- 71. Tawaski, L., Science, 160, 76 (1968).
- 72. Shefter, E., & Higuchi, T., J. Pharm. Sci., 52, 781 (1963).
- 73. Nelson, E., Knoechel, E.L., Hamlin, W.E., & Wagner, J.G., <u>ibid.</u>, <u>51</u>, 509 (1962).
- 74. Unger, R.H. & Madison, L.L., J. Clin. Invest., 37, 627 (1958).
- 75. Middleton, E.J., Shang, H.S., & Cook, D., Can. J. Pharm. Sci., 4, 97 (1969).
- 76. Mullins, J.D. & Macek., T.J., J. Am. Pharm. Ass. Sci. Ed., 49, 249 (1960).
- 77. Feinberg, M., ibid., 9, 113 (1969).
- 78. Reinhold, J.G., Phillips, F.J. & Flipper, H.J., <u>Am. J. Med. Sci.</u>, <u>210</u>, 171 (1945).
- 79. Atkinson, R.M., Bedford, C.B., Child, K.J. & Tamic, E.G., Nature,
 - 193, 588 (1962).
- 80. Levy, G., Lancet, 2, 723 (1962).
- 81. Bauer, G., Rieckman, P. & Shanmann, W., Arzneim. Forsch. 12, 487 (1962).
- 82. Boyed, E.M. & Dingwall, R.W., Am. J. Med. Sci., 213, 549 (1947).
- 83. Kakani, K., Arita, T. & Okashi, S., "Absorption and exretion of
 - chloramphenical" Paper presented to the Am. Pharm. Assoc. National

Mtg., Las Vegas, Nev. (1962) from ref. 56.

84. Lees, K.A., Pharm. J., 289 (1963).

85. Dare, J.G., Australian J. Pharm., 45, 5-58 (1964).

86. Kornblum, S.S., & Hirschorn, <u>J. Pharm. Sci.</u>, <u>59</u>, 606, (1970).
87. Lin, S.L. & Lachman, L., <u>ibid.</u>, <u>57</u>, 2143 (1968).
88. Chiou, W.L. & Riegelman, S., <u>ibid.</u>, <u>58</u>, 1505 (1969).
89. <u>ibid.</u>, <u>60</u>, 1281 (1971).
90. Goldberg, A.H., Gibaldi, M. & Kanig, J.L., <u>ibid.</u>, <u>55</u>, 487 (1966).
91. Gibaldi, M., Feldman, S., & Bates, T.R., <u>ibid.</u>, <u>57</u>, 708 (1968).

- 92. Carstensen, J.T. & Musa, M.N., ibid., 61, 223 (1972).
- 93. Brooke, D., ibid., 62, 795 (1973).
- 94. <u>ibid.</u>, <u>63</u>, 344 (1974).
- 95. Veng Pedersen, P. & Brown, K.F., ibid., 64, 1192 (1975).
- 96. <u>ibid.</u>, <u>65</u>, 1981 (1976).
- 97. Hixson, A.W. & Crowell, J.H., Ind. Eng. Chem., 23, 923 (1931).
- 98. Nernst, W., Z. Phys. Chem., <u>47</u>, 52 (1904).
- 99. Dankwerts, P.V., Ind. Eng. Chem. 43, 1460 (1951).
- 100. Bertroud, A., J. Phys. Chem., 10, 624 (1912).

101. Nedich, R.L. and Kildsig, D.O., <u>J. Pharm. Sci.</u>, <u>61</u>, 214 (1972).
102. Nernst, W. and Brunner, E., <u>Zeit. Phys. Chem.</u>, <u>47</u>, 52 (1904).
103. Wilderman, M., <u>ibid.</u>, <u>60</u>, 445 (1909).
104. Wurster, D.E. and Taylor, P.W., <u>J. Pharm. Sci.</u>, <u>54</u>, 169 (1965).
105. Goyan, J.E., <u>ibid.</u>, <u>54</u>, 645 (1965).
106. Niebergall, P.J., Milosovich, G. & Goyan, J.E., <u>ibid.</u>, <u>52</u>, 236 (1963).
107. Higuchi, W.I. & Hiestand, E.N., <u>ibid.</u>, <u>52</u>, 67 (1963).
108. Rosner, D.E. and Epstein, M., <u>J. Phys. Chem.</u>, <u>74</u>, 4001 (1970).
109. Cable, M. and Evans, D.J., J. Appl. Phys. 37, 2377 (1966).

110. Scriven, L.E., Chem. Engng. Sci., 10, 1 (1959).

111. Duda, J.L. and Vrentas, J.S., AI Ch.E.J., 15, 351 (1969).

112. Ready, D.W. and Cooper, A.R., Chem. Engng. Sci., 21, 917 (1966).

113. Rosner, D.E., J. Phys. Chem., 73, 383 (1969).

114. Ridgeway, K. & Peacock, D.G., J. Pharm. Pharmac., 23, 1215 (1971).

115. Crank, J., "The Mathematics of Diffusion", Oxford University Press, London (1956).

116. Cleave, J.P., J. Pharm. Pharmacol., 17, 698 (1965).

117. Cobby, J., Mayersohn, M. and Walker, G.C., J. Pharm. Sci., 63, 725 (1974).

118. Edmundson, I.C. and Lees, K.A. J. Pharm. Pharmacol., 17, 193 (1965).

119. Withey, R.J., ibid., 23, 573 (1971).

120. Veng Pedersen, P. & Brown, K.F., J. Pharm. Sci., in press.

121. Walpole, R.E. and Myers, R.H. "Probability and Statistics for Engineers

and Scientists", Macmillan, New York, N.Y., 1972.

122. "Symposium on Particle Size Measurement", American Society for Testing

Materials Special Technical Publication No. 234, Philadelphia, Pa., (1958).

123. Metzler, C.M., 'A Users Manual for NONLIN', The Upjohn Co.,

Technical Report 7292/69/7292/005, Kalamazoo, Mich. (1969).

124. Booth, G.W., & Peterson, T.L., 'Nonlinear Estimation', I.B.M. share

program., no. 687 WLNL7 (1958).

125. Marquardt, D.W., 'Least Squares Estimation of Nonlinear Parameters',

I.B.M. share library program, Distribution no. 309401 (1966).

126. Meeter, D.A., 'Program GAUSHAUS', Numerical Analysis Laboratory,

University of Wisconsin, Madison, Wisconsin (1966).

127. Pfeffer, M., 'J. Pharmacokin. Biopharm.'2, 137 (1973).

128. Berman, M. & Weiss, M.F., 'Users Manual for SAAM', National Institutes

of Health, Bethesda, Md. (1968).

129. Chamber, J.M., Appl. Statist., 18, 249 (1969).

130. Fletcher, R., Report AERE, R6799, London: H.M.S.O. (1971).

131. Niebergall, P.J., Schnaare, R.L. and Sugita, E.T., J. Pharm. Sci., 60,

1393 (1971).

132. Atkins, G.L., Biochim. Biophys. Acta, 252, 405 (1971).

- 133. Himmelblau, D.M., 'Applied Nonlinear Programming', McGraw-Hill, New York, (a) p. 9 (b) p. 474 (1972).
- 134. Gauss, C.F., 'Theory of Least Squares'. English translation by H.F. Trotter of original from 1821. Princeton University, Statistical Techniques Research Group. Technical Report No. 5 (1959).

135. Nelder, J.H. and Mead, R., Comput. J., 7, 308 (1965).

- 136. Hogg, R.V., & Craig, A.T., 'Introduction to Mathematical Statistics'. McMillan, New York, (a) p. 383 (b) p. 138 (1970).
- 137. Draper, N.R., & Smith, H., 'Applied Regression Analysis'. John Wiley & Sons, New YOrk. p. 265 (1966).
- 138. Boxenbaum, H.G., Riegelman, S., and Elashoff, R.M., J. Pharmacokin. Biopharm., 2, 123 (1974).
- 139. Davis, O.L., & Goldsmith, P.L., 'Statistical Methods in Research and Production', Oliver and Boyed, Edinburgh, p. 236 (1972).
- 140. Kendall, M.G., 'The Advanced Theory of Statistics', Charles Griffin & Co., London, Chapter 10 (1963).
- 141. Daniel, C., & Wood, F.S., 'Fitting Equations to Data', Wiley & Sons, New York, p. 28 (1971).
- 142. Durbin, J., & Watson, G.S., Biometrica, 37, 409 (1950).
- 143. Sweed, F.S., & Eisenhart, C., Ann. Math. Stats., 14, 66 (1943).
- 144. Anscombe, F.J., Technometrics, 2, 123 (1960).
- 145. Powell, M.J.D. in Rabinowitz, P. (ed.) "Numerical Methods for Non-Linear Algebraic Equations", Gordon & Breach, p. 87 (1970).
- 146. Dennis, J.E., <u>in</u> Byrne, G.D. & Hall, C.H. (eds.) "Numerical Solution of System of Nonlinear Algebraic Equations", Academic Press, New York, p. 157 (1973).

147. Dennis, J.e., Proc. ACM Natl. Conf. 1970, ACM, N.Y. (1971).

148. Hartley, H.O., Technometrics, 3, 269 (1961).

149. Chambers, J.M., Biometrica, 60, 1 (1973).

150. Spendley, W. in Fletcher, R. (ed.), "Optimization", Academic Press,

New York, 259 (1969).

151. Kruger-Thiemer, E. & Levine, R.R., Arzneimittel-Forschung, 18, 1575 (1968).

152. Wagner, J., "Fundamentals of Clinical Pharmacokinetics", Illinois, Drug

Intelligence Publications (1975).

153. Wagner, J., J. Pharmacokin. Biopharm., 3, 457 (1975).

154. Wagner, J., "Biopharmaceutics and Relevant Pharmacokinetics", The

Hamilton Press, Hamilton, Illinois (1971).

155. Berman, M., Ann. N.Y. Acad. Sci., 108, 182 (1963).

156. Nooney, G.C., J. Chron. Dis., 19, 325 (1966).

157. Berman, M., Shahn, E. & Weiss, M.F., Biophys. J., 2, 275 (1962).

158. Hiller, J. in, Andersen, R.S. & Osborne, M.R. (Eds.) "Data Representation"

University of Queensland Press, Queensland, Australia, 62 (1970). 159. Killifors, H.W., J. Amer. Stat. Ass., 62, 399 (1967). 160. Hunter, W.G. & Mezaki, R., A.I. ChE.J., 10, 315 (1964). 161. Box, G.E.P. & Hunter, W.G., Technometrics, 7, 23 (1965). 162. Stewart, G.W., J. Assoc. Comput. Mach., 14, 72 (1967). 163. Fletcher, R. (ed.) "Optimization", Academic Press, New York, p. 1 (1969). 164. Brown, K.M. & Dennis, J.E., Yale University Report 71-6 (1971). 165. Powell, M.J.D., in Murray, W. (ed.) "Numerical Methods for Unconstrained Optimization", Academic Press, p. 29 (1972).

166. Herdan, G. & Smith, M.L., "Small Particle Statistics", Elsevier, New

York, N.Y. (1953).

167. Stein, H.H., Analyt. Biochem., 13, 605 (1965).

168. Agren, A., Acta. Pharm. Suec., 5, 37 (1968).

169. Meyer, M.C. & Guttman, D.E., J. Pharm. Sci., <u>57</u>, 1627 (1968).

170. Meyer, M.C. & Guttman, D.E., J. Pharm. Sci., 59, 33 (1970).

171. Meyer, M.C. & Guttman, D.E., J. Pharm. Sci., 59, 39 (1970).

172. Dearden, J.C. & Tomlinson, E., J. Pharm. Pharmac., 22, Suppl., 53S. (1970)

173. Goto, S., Ohki, T., Kiryu, S. & Iguchi, S., Yakuzaigaka, 31, 247 (1971).

174. Brown, K.F. & Crooks, M.J., Pharm. Acta. Helv., 48, 494 (1973).

175. Crooks, M.J. & Brown, K.F., J. Pharm. Sci., 62, 1904 (1973).

176. Crooks, M.J. & Brown, K.F., J. Pharm. Pharmac., 26, 235 (1974).

177. Crooks, M.J. & Brown, K.F., J. Pharm. Pharmac., 26, 304 (1974).

178. Brown, K.F. & Crooks, M.J., Biochem. Pharmacol., In the press (1976).

179. Kanfer, I. & Cooper, D.R., J. Pharm. Pharmac., 28, 58 (1976).

180. Kazmi, S.J.A. & Mitchell, J. Pharm. Sci., 62, 1299 (1973).

- 181. Bottari, F., Di Colo, G., Nannipieri, E., Saettone, M.F. & Serafini, M.F., J. Pharm. Sci., 64, 946 (1975).
- 182. Craig, L.C., Advan. Anal. Chem. Instr., 4, 35 (1966).
- 183. Matsumoto, H., Matsumura, H. & Iguchi, S., <u>Chem. Pharm. Bull.</u>, <u>14</u>, 385 (1966).

184. Hart, H.E., Bull. Math. Biophys., 27, 87 (1965).

- 0 0 0 -

- 302 -

Table of Symbols

a	diameter or equivalent spherical diameter.
a	mean diameter.
A	surface area.
α	acute angle of crystal (p. 69).
bo	side length of crystal (t=0).
co	solubility of compound in unionised form.
c _b	solute bulk concentration.
° _i	interfacial concentration of solute.
с*	dimensionless concentration.
COV()	covariance.
do	initial diameter of the smallest particle in a powder.
D _o ·	initial diameter of the largest particle in a powder.
D	diffusion coefficient.
Dt	total concentration of small molecule.
Df	free concentration of small molecule.
∇ ²	Laplacian operator.
e _i	i-th residual value.
E()	expected value.
E _t (w)	mean particle weight of time t.
	1-erf(x) complementary error function.
erf(x) =	$\frac{2}{\sqrt{\pi}} \int_{0}^{x} e^{-u^2} du$ error function.
f()	particle weight density function of time t.
	shape ratio.
$F(x) = \frac{1}{\sqrt{2}}$	$ \int_{-\infty}^{x} e^{-u^{2}/2} du $ cumulative normal distribution function.
^F α,n,p	critical point of F-distribution.

ψ	stream function, time function.
g	acceleration of gravity.
g()	single particle dissolution function (p. 97).
g ⁻¹ ()	inverse single particle dissolution function (p. 97).
γ	mean rate of surface renewal (p. 51).
h	diffusion layer thickness.
н	Hessian matrix.
H+	Htg concentration.
H _{ii}	i-th diagonal element of Hessian matrix.
i	lower truncation parameter (Fig. 5.1)
j	upper truncation parameter (Fig. 5.1)
J	interfacial mass flux (p. 66).
JD	diffusion flux.
k _i	effective interfacial transport rate constant (p. 52).
k _m	rate parameter for model m.
Ka	dissolution constant of acid.
ĸ	dissociation constant of base.
ĸe	dialysis rate constant (Chapter 11).
ĸ	association constant for i-th class (Chapter 11)
ĸ	Michaelis Menten parameter (Chapter 11), rate parameter for
	model m.
Kw	ion product of water.
к*	specific dissolution rate parameter, association constant
	for membrane binding.
10	sidelength of crystal (Fig. 4.4).
1()	initial particle size distribution.
L()	likelihood function.

m	model parameter.
mo	initial weight of the smallest particle.
M	initial weight of the largest particle.
μ	logarithmic mean (Fig. 5.1). viscosity (Chapter 4).
n _i	number of sites in i-th class (Chapter 11)
N()	normal distribution function (p. 101).
Nt	number of particles at time t.
\overline{v}	number of moles of small molecules bound per mol macromalecule.
<u>√</u> *	amount of small molecule bound per amount of available membrane
	material.
P	parameter vector.
Р	operator (defined p. 99).
Pt	total concentration of macromolecule (Chapter 11).
π =	3.1415
rt	radius of particle at time t.
r*	dimensionless radius.
ρ	density
ρ _{i,j}	correlation coefficient between i-th and j-th element.
Re	Reynolds number.
S	distance to fixed point (Chapter 4)
	dispersion product (Chapter 6)
	variable defined p. 269.
Sc	Schmidt number.
SDi	standard deviation of i-th element.
SS	sum of squared residuals.
s _N ()	cumulative distribution of residuals.
σ	standard deviation of normal distribution (Fig. 5.1.)

t	time.
t*	dimensionless time (Chapter 4).
t _{α,n}	critical value of t-distribution.
V _∞	vertical velocity of free falling spherical particle.
v	variance-covariance matrix.
v _m	Michaelis Menten parameter.
Var()	variance.
w	weight of single particle.
W	weight of undissolved powder.
θ	spherical coordinate (Chapter 4).
(^x) =	x!/(y!(x-y)!) binomial coefficient.

- 0 0 0 -

C FUN NON THE C SUB C C C FUN SUB C C C C C C C C C C C C C C C C C C C	GRAM FUNFIT(INPUT,OUTPUT,TAPE5=: PUT,TAPE6=OUTPUT) FIT IS AN INTERACTIVE TIME-SHARING PROGRAM FOR GENERAL LINEAR REGRESSION AND CURVE FITTING PROGRAM CONSISTS OF- N PROGRAM / FUNFIT ROUTINES / NELDR.CHOL,SYMINV,LSO,XYPLOT,PLOT, / SEARCH,READ,PLACE,PROBI,KOLMIR CTIONS / SIGNOF,WEIGHT,SEOPRO,NBC,SDF UBROUTINE MODEL(Y,X,P,IPRINT) NUST BE SUMPLIED BY THE USER DEFINE THE FUNCTION(S) TO BE FITTED. FOR DETAILS ABOUT THIS ROUTINE CONTACT P.VENG PEDERSEN,THE PHARMACY DEPARTMENT UNIVERSITY OF SYDNEY,N.S.H. 2006,AUSTRALIA.	1/0	1
DIM STP VII VAR VAR VAR VAR VAR VAR COM COM COM COM COM COM COM COM	ENSION F(20), F1(20), F2(20), FSAVE(20), WF(20), FS(20), NT(20), S(20), STEP(20), STEP5(20), VAR(20), TER'T(30), X(100), Z(100), COV(55), INDEX(100), RES(100), DIER'T(30), YEST(100), IOUTL(10), COV(55), INDEX(100), WRES(100), WRES(10), STPWEW(7), NRECOM(7), IT(7), V(C'55), A(60), FMIN(20), AR(12), XX(9, 100), G'9), LABEL(100), SET(10), LIMI(10), LIM2(10), RSS(10), WRSS(10) BLE PRECISION VC MON XMIN, XMAX MON /FUNNEM/ITHFUN MON /DATA/ XX,Y, WNN, NOBS MON /PARLIM/ F1,F2 MON /CONSTR/LIMITS MON /B2/ 2 MON /B2/ XARCOV,VC, NEVAL1, NEVAL2, AMING MON /B4/ NINO MON /B4/ NINO MON /B4/ NINO MON /B6/ NOP MON /B7/ ITABLE MON /B6/ NOP MON /B1/AR MON /	150 160 170 180 200 210 220 230 240 250 260 250 260 270 280 290 300 310 320 330 340 350 360 370 380 370 380 390 400 410 420 430 400 510 520 550 560 570 580 550 560 570 580 590 600 610	

 IOUAD=1
 620

 SIMP=1E-6
 630

 ANS4=2HNO
 640

 PLOTRS=2HNO
 650

 MAX=800
 660

 NLODP=20
 660

 IPRINT=-1
 680

 ANS11=2HNO
 690

 ANS12=2HNO
 700

 STOPCR=0.001
 710

 NIMD=1
 720

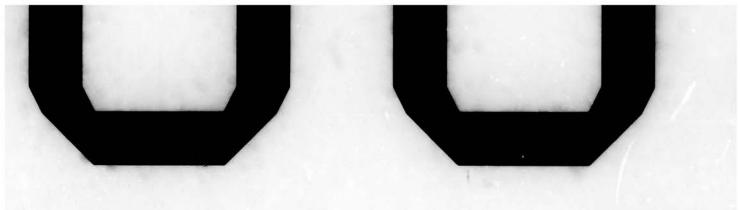
 PLT=2HNO
 730

 ANS16=2HNG
 740

 SM0=1
 750

 HRITE(6,505)
 760

 CALL READ(1,AR)
 780



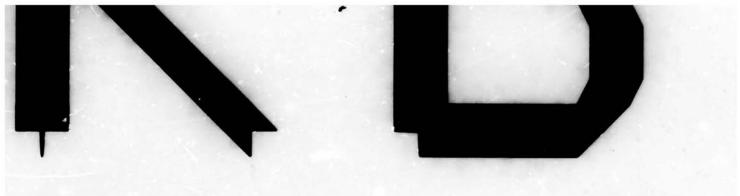
370	J=IOUTL(I) WRITE(6,822)J	9300 9310
	ir(NOUTL.EQ.1) WRITE(6,824) IF(NOUTL.GT.1) WRITE(6,826)	9320 9330
3701	CONTINUE IF (ANS16.EQ.2HND.OR.NOWGHT.EQ.1) GO TO 3704	9340 9350
	WRITE(6,8241)T	9360
	NOUTL=0 STAR=2H *	9370 9380
	NSAVE=1	9390
	DO 3703 I=1,NOBS N=LABEL(I)	9400 9410
1	IF(N.EQ.NSAVE+1) WRITE(6,8240) N	9420
1	NSAVE=N	9430
	WNRDV=WRES(I)/WSDVRS IF(ABS(WNRDV).LT.T) GO TO 3702	9440 9450
	NOUTL=NOUTL+1	9460
	WRITE(6,8242)1,STAR,WY(1),WYN(1),WRES(1),WNRDV G0 T0 3703	9470 9480
	WRITE(6,8243)1,WY(I),WYN(I),WRES(I),WNRDV	9490
3703	CONTINUE WRITE(6,8244)	9500 9510
	IF(NOUTL.EQ.1) WRITE(6,8245)	9520
770/	IF(NOUTL.EQ.1) WRITE(6,8245) IF(NOUTL.GT.1) WRITE(6,8246)	9530
5/04	CONTINUE IF(NIND.GT.1.OR.PLT.NE.3HYES) GO TO 371	9540 9550
	CALL XYPLOT(F)	9560
	20 3709 11=1,NFUNC N1=LIM1(11)	9570
	N2=N1+1	9590
	N3=LIM2(11) NN=N3-1	9600
	XMAX=X(N1)	9610 9620
	XMIN=X(N1)	9630
	DC 3705 I=N2,N3 IF(X(I).GT.XMAX) XMAX=X(I)	9640 9650
3705	IF(X(I).LT.XMIN) XMIN=X(I)	9660
	DELTA=(XMAX-XMIN)/50. IF(II.E0.1) WRITE(6,8261)	9670 9680
	IF(II.GT.1) WRITE(6,8264)II	9690
	D0 3706 1=1,26 J=26+1	9700 9710
	XP1=XMIN+FLOAT(I-1)+DELTA	9720
	XP2=XMIN+FLOAT(J-1)+DELTA	9730
	D0 3708 L=N1,NN IF(XP1.GT.X(L).AND.XP1.LE.X(L+1))	9740 9750
	YP1=Y(L)+(Y(L+1)-Y(L))(XP1-X(L))/(X(L+1)-X(L))	9760
	1F(XP2.GT.X(L).AND.XP2.LE.X(L+1)) •YP2=Y(L)+(Y(L+1)-Y(L))•(XP2-X(L))/(X(L+1)-X(L))	9770
	ITHEIN=11	9790
	CALL MODEL(Y/1,XP1,F,0) CALL MODEL(Y/2,XP2,F,0)	9800
	IF(1.EQ.26) 69 TO 3707	9810 9820
3706	WRITE(6,8262) 1,XP1,YP1,J,XP2,YP2	9830
	WRITE(6,8262) I,XP1,YP1 CONTINUE	9840 9850
	WRITE(6,8263)	9860
371	CONTINUE IF(IQUAD.NE.1) GC TO 376	9870 9880
	WRITE(6,827)	9890
	DO 375 K=1,2 1F(K.EQ.2) WRITE(6,8270)	9900
	IFIN.EU.27 MAIIE(0,02/07	9910

L=		9920
372 IF	(L.GT.NPR) GO TO 375	9930
II:	L*(L-1)/2	9940
DO	373 1=L, NPR	9950
IA	-11+L	9960
	ii•i	9970
	MINO(11,1A+5)	9980
İF	(K.EQ.2) GO TO 3721	9990
no.	3720 J=1A, IB	10000
3720 VA	COV(J)=2.*FUNC*VARCOV(J)/FLOAT(NRSDF)	10010
HR.	TE(6,829)(VARCOV(J), J=1A,1B)	10020
	TO 373	10030
	3722 KK=IA, IB	10040
3722 VAL	COV(KK)=VC(KK)	10050
	TE(6,829)(VARCOV(J), J=1A, 1B)	10060
373 CO	ITINUE	10070
	+6	10080
		10080

-		
	1	

	L=1*(1-1)/2+J	18600		
	BMAT(L)=2D0+((HSTST+A0)-(AVAL(1)+AVAL(J)))	18610		
	CONTINUE	18620		
207	CONTINUE L=0	18630		
	DO 210 1=1,NAP	18640		
	11=1+1	18660		
	L=L+I	18670		
210	BMAT(L)=2D0*((H([1)+A0)-2D0*AVAL(I)) CONTINUE	18680		
210	DO 237 I=1,NAP	18690 18700		
	11=1-1	18710		
237	AVAL(1)=2D0*AVAL(1)-(H(11)+3D0*A0)/2D0	18720	1	
	D0 219 I=1,NOP	18730	1	
219	PMIN(I)=G(1,I) DO 211 I=1,NAP	18740 18750		
	11=1+1	18760		
	DO 211 J=1,NOP	18770		
	G(I1, J) = G(I1, J) = G(1, J)	18780		
211	CONTINUE DO 212 I=1, NAP	18790		
	[1=]+1	18810		
	DO 212 J=1,NOP	18820		
	G(1, J) = G(11, J)	18830		
212	CONTINUE CALL SYMINV(BMAT,NAP,BMAT,TEMP,NULLTY,IFAULT)	18840 18850		
	IF(IFAULT.NE.O) GO TO 450	18860		
	IRANK = NAP-NULLTY	18870		
	GO TO 441	18880		
	WRITE(IW, 302)	18890		
302	FORMAT(49H MATRIX TO BE INVERTED NOT POSITIVE SEMI-DEFINITE) IF(NVIOL.GT.0.AND.IPRINT.EQ1) WRITE(IW,8031)NVIOL	18900		
	IF(NFIX.GT.0) WRITE(6,3020) NFIX	18920		
3020	FORMAT(" NO. OF VERTICES OF SIMPLEX CONSTRAINED BY THE",	18930		
	• PARAMETER LIMITS = ",12/) IFAULT = 2	18940		
	RETURN	18950 18960		
441	DO 213 I=1,NAP	18970		
	H(1)=0D0	18980		
	D0 214 J=1, NAP	18990		
216	IF(J-I)216,216,215 IJ=I+(I-1)/2+J	19000 19010		
	GO TO 217	19020		
	[J=J*(J-1)/2+]	19030		
	H(1)=H(1)+BMAT(1J)*AVAL(J)	19040		1
	CONTINUE	19050 19060		
2.5	YMIN=0D0	19070		
200	DO 218 I=1,NAP	19080		
218	YMIN=YMIN+H(I) *AVAL(I)	19090		
	YMIN=AO-YMIN DO 220 J=1,NOP	19100 19110		
	PSTST(1)=0.0	19120		
	D0 220 J=1, NAP	19130		
220	PSTST(1)=PSTST(1)+H(J)+G(J,1)	19140		
221	DO 221 I=1,NOP PMIN(I)=PMIN(I)-PSTST(I)	19150		
ee i	AMINO=YMIN	19160 19170		/
	IF(IPRINT)223,222,222	19180		
222	WRITE(IW, 303) YMIN, (PMIN(1), I=1, NPR)	19190		
303	FORMAT(42H MINIMUM OF FITTED QUADRATIC SURFACE IS ,D15.8,3H AT// (SE14.6))	19200		
		19210		

WRITE(IW,304)FUNC_(F(I),I=1,NPR)	19220
304 FORMAT(/42H COMPARE WITH MINIMUM FOUND BY ITERATION ,D15.8,3H AT/	19230
*/(5E14.6))	19240
D=SNGL(100-YMIN/FUNC)	19250
IF(ABS(D).GT1) WRITE(IW,305)	19260
305 FORMAT(/	19270
* IF DIFFERENCE IS LARGE,INFORMATION MATRIX IS INACCURATE AND A"/	19280
*" NEW RUN WITH A DIFFERENT STOPPING 'RITERION, EXPANSION',	19290
CRITERION'/ OR PARAMETER RANGE IS RECOMMENDED.'/)	19300
223 CONTINUE	19310
DO 2230 1=1,NOP	19320
IF(PMIN(1),LT,F1(1))PMIN(1)=F1(1)	19330
2230 IF(PMIN(1),GT,F2(1))PMIN(1)=F2(1)	19340
CALL FUNCTN(PMIN,OMIN)	19350
IF(QMIN.GE.FUNC) GO TO 2234 FUNC=QMIN DO 2231 [=1,NOP 2231 F(1)=PMIN(1)	19360 19370 19380



DO 260 17=1,NP	27900
PP(7,17)=F1(7)+FLOAT(17-1)+DE'(7) FF(7)=PP(7,17)	27910 27920
D0 250 16=1,NP	27930
PP(6,16)=F1(6)+FLOAT(16-1)+DEL(6)	27940
FF(6)=PP(6,16)	27950
D0 240 15=1,NP	27960
PP(5,15)=F1(5)+FLOAT(15-1)+DEL(5) FF(5)=PP(5,15)	27970 27980
DO 230 14=1,NP	27990
PP(4,14)=F1(4)+FLOAT(14-1)+DEL(4)	28000
FF(4)=PP(4,14)	28010
D0 220 13=1,NP	28020
PP(3,13)=F1(3)+FLOAT(13-1)+DEL(3) FF(3)=PP(3,13)	28030 28040
DO 210 12=1.NP	28050
PP(2,12)=F1(2)+FLOAT(12-1)+DEL(2)	28060
FF(2)=PP(2,12)	28070
DO 200 11=1.NP PP(1,11)=F1(1)+FLOAT(11-1)*DEL(1)	28080 28090
FF(1)=PP(1,11)	28090
CALL LSQ(FF,SUM)	28110
NEVAL=NEVAL+1	28120
IF(SUM.GE.SSMIN) GO TO 195 SSMIN=SUM	28130
IS(1)=11	28140 28150
15(2)=12	28160
1\$(3)=13	28170
15(4)=14	28180
15(5)=15 15(6)=16	28190 28200
IS(7)=17	28210
195 IF(F1(1).EQ.F2(1)) GO TO 205	28220
200 CONTINUE	28230
205 IF(F1(2).E0.F2(2)) GO TO 215 210 CONTINUE	28240 28250
215 IF(F1(3).E0.F2(3)) GO TO 225	28260
220 CONTINUE	28270
225 IF(NOP.EQ.3) GO TO 270	28280
IF(F1(4).EQ.F2(4)) GO TO 235 230 CONTINUE	28290
235 IF(NOP.EQ.4) GO TO 270	28300 28310
IF(F1(5).EQ.F2(5)) GO TO 245	28320
240 CONTINUE	28330
245 IF(NOP.EQ.5) GO TO 270	28340
IF(F1(6).EQ.F2(6)) GO TO 255 250 CONTINUE	28350 28360
255 IF(NOP.EQ.6) GO TO 270	28370
IF(F1(7).EQ.F2(7)) GO TO 270	28380
260 CONTINUE	28390
270 CONTINUE DO 280 1=1,7	28400
FNEW(1)=PP(1, IS(1))	28410 28420
280 STPNEW(1)=DEL(1)/2.	28430
RETURN	28440
1000 FORMAT(// 11X,	28450
* PLOT OF RESIDUAL SUM OF SQUARES VERSUS PARAMETER NO.",12/ 11X, RANGING FROM ",E11.4," TO ",E11.4)	28460 28470
1010 FORMAT(// 6X.	28480
* PLOT OF WEIGHTED RESIDUAL SUM OF SQUARES VERSUS PARAMETER , * NO. ,12/6X, RANGING FROM ,E11.4, TO ,E11.4 ?	28490
* NO. , 12/6X, RANGING FROM , E11.4, TO , E11.4)	28500
1020 FORMAT(/ " TABLE OF PLOT ABOVE //	28510

B04

	and the second se
*2(" NO. SUM OF SQUARES PARAMETER ")/1X,68(1H-))	28520
1025 FORMAT(/ TABLE OF PLOT ABOVE"//	28530
*2(" NO. WGHT.SUM OF SOUARES PARAMETER ")/1X,68(1H-))	28540
1026 FORMAT(14,2E14.4,3X,14,2E14.4)	28550
1027 FORMAT(1X,68(1H-))	28560
1030 FORMAT(//" THE SEARCH GRID CONSISTS OF 100 POINTS GIVEN BY",	28570
*" PAIRED"/" COMBINATION OF FOLLOWINC PARAMETER VALUES"//	28580
* PAR.NO.", I1," (J=1-10)"//(5("(",I2,")",E10.4)))	28590
1040 FORMAT(/" PAR.NO.",11," (1=1-10)" //(5(" (",12,")",E10.4)))	28600
1050 FORMAT(/	28610
 MATRIX OF RESIDUAL SUM OF SQUARES AT POINTS IN SEARCH GRID. 	28620
1060 FORMATC / MATRIX OF WEIGHTED RESIDUAL SUM OF SQUARES AT POINTS",	28630
•" IN SEARCH GRID.")	28640
1070 FORMAT(28650
* THE I, J-TH ELEMENT IN THE MATRIX CORRESPONDS TO THE J-TH /	28660
* VALUE OF PAR.NO. , I1, AND THE 1-TH VALUE OF PAR.NO. , I1//)	28670
1075 FORMAT(5(2X, " J=",12),3X,5(2X, " J=",12) /)	28680
1080 FORMAT(5(E10.4.5X))	28690

12	COMMON /B11/AR	370	the second second second second	
	COMMON /B12/NPR	380		
	COMMON /B13/IGNORE	390		
	COMMON /B14/NVICL	400		
	COMMON /B15/SKIP	410		
	LOGICAL SKIP, SKIPS	420		
	EQUIVALENCE (AR(1), ANS)	430		
	SXTERNAL LSO	440		
	+2 20 2 11 2 14 2 14 2 14 2 13 2 12 2 11 2 10 2 00 2 00 2 00	450 460		
	DATA TCRIT/12.7,4.30,3.18,2.78,2.57,2.45,2.36,2.31,2.26,2.23, *2.20,2.18,2.16,2.14,2.13,2.12,2.11,2.10,2.09,2.09,2.08, *2.07,2.06,2.06,2.06,2.05,2.05,2.05,2.04,2.04/ DATA NRECOM /0.0,343,625,1024,729,2187 /	470		
	DATA NRECOM /0.0.343.625.1024.729.2187 /	480		
	UNIN NUNII / U, U, O, 10, 32, 04, 120 /	490		
	HRITE(6,500)	500		
	LINES=30	510		
	NRUN=0	520		
	ITABLE=0 LIMITS=0	530		
	NCORR=0	540		
	NDUMMY=0	550 560		
	NS=1	570		
	NFUNC=-1	580		
	NOP=-1	590		
	NOP2=-1	600		
	NOBS2=-1	610	2.5	

B01

IQUAD=1	170
SIMP=1E-6	620 630
ANS4=2HNO	640
PLOTRS=2HNO	650
MAX=800	660
NLOOP=20	670
IPRINT=-1	680
ANS11=2HNO	690
ANS12=2HNO	700
STOPCR=0.001	710
NIND=1	720
PLT=2HNO ANS16=2HNO	730
5 NQ=1	740
WRITE(6,505)	750 760
IGNORE=0	770
CALL READ(1, AR)	780
GO TO (1000,1001) IGO	790
6 IF(ANS.NE.3HYES) GO TO 8	800
WRITE(6,511)	810
WRITE(6,512) 8 IF(NS.GT.NG) GO TO 1006	820
8 IF(NS.GT.NG) GO TO 1006	830
9 CONTINUE	840
10 NG=2	850
WRITE(6,520) READ(5,525)	860 870
IF(E0F(5))10,11,10	870
11 IF(NS.GT.NQ) GO TO 1006	890
15 NQ=3	900
	910
WRITE(6,530) CALL READ(1,AR)	920
GU TO (1000,1001) GO	930
16 IF(ANS.EQ. 3HYES) IG AD=1	940
IF(ANS.EQ.2HNO) IQUAD=0	950
IF(ANS.EU.2HNO) GO TO 18	960
WRITE(6,535)	970
CALL READ(1,AR) IF(IGO.E0.2) GD TO 1001	980
IF(ANS.EQ.4HBACK) GO TO 1004	990
SIMP=ANS	1000 1010
IF(SIMP.LT.0 OR.SIMP.GT. 10.)G0 TO 1007	1020
18 IF(NS.GT.NQ) GO TO 1006	1030
20 NQ=4	1040
WRITE(6,540)	1050
CALL READ(1,AR)	1060
GO TO (1000,1001) IGO	1070
21 ANSC-ANS	1080
IF (ANS4.EQ.2HNO) GO TO 24	1090
WRITE(6,541) 22 CALL READ(1,AR)	1100
IF(IGO.EQ.2) GO TO 1001	1110
IF (ANS.EQ. 4HBACK) GO TO 1004	1120
PLOTRS=ANS	1140
IF (PLOTRS.NE.2HNO.AND.PLOTRS.NE.3HYES) GO TO 23	1150
GO TO 24	1160
23 WRITE(6,1020)	1170
GO TO 22	1180
24 IF(NS.GT.NG) GO TO 1006	1190
25 NO=5	1200
WRITE(6,545) CALL READ(1,AR)	1210
G0 TC (1003,1001)IG0	1220
00 10 11003,1001/100	1230

C01

The second se	
26 MAX=1F1X(ANS+.1)	1240
IF(MAX.LT.1.OR.MAX.GT.2000) GO TO 1007	1250
IF(NS.GT.NQ) GO TO 1006	1230
	1269 1270
30 NQ=6	1270
WRITE(6,550)	1280 1290
CALL READ(1, AR)	1290
GO TO (1003,1001)IGO	1300
GO TO (1003,1001)IGO 31 NLOOP=IFIX(ANS+.1)	1310
IF(NLOOP.LT.1) GO TO 1007 IF(NLOOP.GT.MAX.AND.MAX.NE.1) GO TO 32	1300 1310 1320 1330 1340
IF (NLOOP, GT. MAX, AND, MAX, NE, 1) GO TO 32	1330
GO TO 33	1340
GO TO 33 32 NAGR=5	1350
60 10 1008	1360
GO TO 1008 33 IF(NS.GT.NO) GO TO 1006	1370
33 IF(NS.GT.NQ) GO TO 1006 35 NQ=7	1370 1380
35 NO=7	1200

UELIA=(XRAX-XRIN)/30. IF(II.E0.1) WRITE(6,8261) IF(II.GT.1) WRITE(6,8264)II D0 3706 I=1,26	9670 9680 9690 9700	
J=26+I XP1=XMIN+FLOAT(I-1)+DELTA	9710 9720	
XP2=XMIN+FLOAT(J-1)+DELTA	9730	
DO 3708 L=N1,NN IF(XP1.GT.X(L).AND.XP1.LE.X(L+1))	9740 9750	
YP1=Y(L)+(Y(L+1)-Y(L))(XP1-X(L))/(X(L+1)-X(L)) 3708 IF(XP2.GT.X(L).AND.XP2.LE.X(L+1))	9760	
YP2=Y(L)+(Y(L+1)-Y(L))(XP2-X(L))/(X(L+1)-X(L))	97.'0 9780	
ITHFUN=II CALL MODEL(Y/1,XP1,F,0)	9790 9800	
CALL MODEL(Y)2, XP2, F, 0) IF(I.EQ.26) 67 TO 3707	9810	
3706 WRITE(6,8262) 1,XP1,YP1,J,XP2,YP2	9820 9830	
3707 WRITE(6, 5262) I, XP1, YP1 3709 CONTINUE	9840 9850	
WRITE(6,8263)	9860	
371 CONTINUE IF(IQUAD.NE.1) GU TO 376	9870 9880	
WRITE(6,827)	9890	
DO 375 k=1,2 1F(k.EQ.2) WRITE(6,8270)	9900 9910	

	1.4
172	L=1 IF(L.GT.NPR) GO TO 375
3/2	11=L*(L-1)/2
	DO 373 I=L,NPR
	IA=II+L
	11=11+1
	IB=MINO(II, IA+5)
	IF(K.EQ.2) GO TO 3721
	DO 3720 J=1A,1R
3720	VARCOV(J)=2.*FUNC+VARCOV(J)/FLOAT(NRSDF)
	WRITE(6,829)(VARCOV(J), J=1A, IB)
7774	GO TO 373
3722	DO 3722 KK=IA,IB VARCOV(KK)=VC(KK)
3/22	WRITE(6,829)(VARCOV(J), J=IA, IB)
373	CONTINUE
	L=L+6
	GO TO 372
	CONTINUE
376	1F(ANS4_EQ.2HNO) GO TO 395
	DWSTAT=0.
790	D0 380 1=2,NOBS DWSTAT=DWSTAT+(RES(1-1)-RES(1))**2
200	DWSTAT=DWSTAT/SSRES
	WRITE(6,830)DWSTAT
	N=0
	DO 381 I=1,NOBS
	IF(WYN(I).EQ.0.) GO TO 381
	N=N+1
	VARCOV(N)=RES(I)
581	CONTINUE PR=SEOPRO(VARCOV,N)
	WRITE(6,832)PP
	IF(PR.LT.0.05) WRITE(6,833)
	IF(NOBS.GT.3) CALL KOLMIR(VARCOV, N, DN, CRIT, 1)
	CALL PROB1(VARCOV, N, MM, PR)
	WRITE(6,8320)MM,N,PR
	IF(PR.LT.0.C5) WRITE(6,833)
	RATID=FLOAT(NRSDF)/ FLOAT(NOBS-NDUMMY)
	IF(RATIO.LE5) WRITE(6,834)RATIO IGNORE=1
	IF(PLOTRS.NE.3HYES) GO TO 395
	IF(NIND.GT.1) 0 TO 390
	WRITE(6,836)
	CALL PLOT(X,RES,NOBS,LINES)
	WRITE(6,838)
	CALL PLOT(YEST, RES, NOBS, LINES)
700	GO TO 391
370	WRITE(6,840) CALL PLOT(X,RES,NOBS,LINES)
	WRITE(6,842)
	CALL PLOT(Z, RES, NOBS, LINES)
	WRITE(6,838)
	CALL PLOT(YEST, RES, NOBS, LINES)
391	IF(ANS16.EO.2HNO.OR.NOWGHT.EO.1) GO TO 395
	IF(NIND.GT.1) GO TO 392
	WRITE(6,8360) CALL PLOT(X,WRES,NOBS,LINES)
	WRITE(6,8380)
	CALL PLOT(YEST, WRES, NOBS, LINES)
10000	GO TO 395
392	WRITE(6,8400)

COS

	CALL PLOT(X,WRES,NOBS,LINES) WRITE(6,8420) CALL PLOT(Z,WRES,NOBS,LINES) WRITE(6,8380)		
	CALL PLOT(YEST, WRES, NOBS, LINES)		
395			
	WRITE(6,8401)STOPCR		
	IF(IQUAD.NE.1) GO TO 396		
	WRITE(6,8402)SIMP		
	D=ABS(1AMING/FUNC)		
	IF(IPRINT.EQ1.AND.D.GT.0.1)	WRITE(6,8403)AMING,FUNC	
396	NEVAL I=NEVAL1-NVIOL		
	WRITE(6,8423)NEVAL1,NVIOL		
	IF(IQUAD.EO.1.AND.IPRINT.EO1)	WRITE(6,8424)NEVAL2	
	ITABLE=1		
	ITHFUN=NFUNC		

	H(1)=0D0		18980	1		
	DO 214 J=1,NAP		18990			
214	IF(J-I)216,216,215 5 IJ=I+(I-1)/2+J		19000			
210	GO TO 217		19010 19020			
215	5 [J=J*(J-1)/2+]		19030			
	<pre>7 H(1)=H(1)+BMAT(1J)+AVAL(J)</pre>		19040		1	
214	CONTINUE		19050			
215	S CONTINUE YMIN=0D0		19060 19070			
	DO 218 I=1, NAP		19080			
218	B YMIN=YMIN+H(I)*AVAL(I)		19090			
	YMIN=A0-YMIN		19100			
	D0 220 1=1,NOP PSTST(1)=0.0		19110 19120			
	DO 220 J=1,NAP		19130			
220) PSTST(1)=PSTST(1)+H(J)+G(J,1)		19140			
	DO 221 I=1,NOP		19150			
221	PMIN(1)=PMIN(1)-PSTST(1) Aming=ymin		19160 19170		1	
	IF(IPRINT)223,222,222		19180			
222	2 WRITE(IW, 303) YMIN, (PMIN(I), 1=1, NPR)		19190			
	FORMAT(42H MINIMUM OF FITTED QUADRATIC SURFACE IS	,D15.8,3H AT//	19200			
	•(5E14.6))		19210			

	the second s
WRITE(IW, 304)FUNC, (F(I), I=1,NPR)	19220
504 FORMAT(/42H COMPARE WITH MINIMUM FOUND BY ITERATION ,D15.8,3H AT/	19230
*/(SE14.6)) D=SNGL(1D0-YMIN/FUNC)	19240 19250
IF(ABS(D).GT1) WRITE(IW, 305)	19260
SOS FORMAT(/	19270
 *" IF DIFFERENCE IS LARGE, INFORMATION MATRIX IS INACCURATE AND A"/ *" NEW RUN WITH A DIFFERENT STOPPING "RITERION, EXPANSION", 	19280 19290
* CRITERION"/" OR PARAMETER RANGE IS RECOMMENDED. /)	19300
23 CONTINUE DO 2230 I=1,NOP	19310 19320
IF(PMIN(1),LT,F1(1))PMIN(1)=F1(1)	19330
30 IF(PMIN(I).GT.F2(I))PMIN(I)=F2(I)	19340
CALL FUNCTN(PMIN, QMIN) IF(QMIN.GE.FUNC) GO TO 2234	19350 19360
FUNC=QMIN	19370
DO 2231 I=1,NOP 31 F(I)=PMIN(I)	19380 19390
IF(IPRINT.LT.0) GO TO 2234	19400
WRITE(IW,2232)FUNC	19410
32 FORMAT(/ IMPROVED FUNCTION VALUE =",D15.8/) IF(NPR.LE.5) WRITE(IW,1080)(F(I),I=1,NPR)	19420 19430
IF(NPR.GT.5) WRITE(IW, 108)(F(I), I=1, NPR)	19440
34 CONTINUE DO 224 I=1,NOP	19450 19460
D0 225 J=1,NAP	19470
H(J)=000	19480
D0 226 K=1,NAP IF(K-J)227,227,228	19490 19500
27 JK=J*(J-1)/2+K	19510
GD TD 229 28 JK=K+(K-1)/2+J	19520 19530
29 H(J)=H(J)+BMAT(JK)+DBLE(G(K,I))/ZD0	19540
26 CONTINUE	19550
25 CONTINUE DO 230 J=1,NOP	19560 19570
IJ = J * (J - 1)/2 * I	19580
VC(IJ)=0D0 D0 231 K=1.NAP	19590 19600
31 VC(IJ)=VC(IJ)+H(K)+DBLE(C(K,J))	19610
30 CONTINUE	19620
24 CONTINUE J = 0	19630 19640
DO 234 (=1,NOP	19650
J = J+1 34 VAR(1)=VC(J)	19660 19670
36 IF(IPRINT.GE.O) WRITE(IW, 306) IRANK	19680
06 FORMAT(/" INFORMATION MATRIX HAS RANK",13// *"GENER/ IZED INVERSE OF INFORMATION MATRIX"/)	19690
IJK=1	19700 19710
GO TO 710	19720
17 CONTINUE IF(IPRINT.GE.O) WRITE(IW,308)	19730 19740
08 FORMAT(19750
* THE MATRIX ABOVE MUST BE MULTIPLIED BY TWICE THE ESTIMATE OF * THE RESIDUAL // VARIANCE TO GIVE THE VARANCE-COVARIANCE MATRIX	19760
* OF THE PARAMETERS. ()	19770 19780
CALL SYMINV(VC, NAP, BMAT, TEMP, NULLTY, IFAULT)	19790
IF(IPRINT.GE.O) WRITE(IW,320) 20 FORMAT(/ INFORMATION MATRIX (HESSIAN MATRIX)" /)	19800
IJK=3	19820
GO TO 710	19830

C03

 711
 IJK=2
 19840

 IJ = 0
 19850

 D0
 701
 1=1,NPR

 II = 1I+1
 19870

 IF(VC(II),LE,000)
 G0 T0 702

 VC(II)=DD/DSORT(VC(II))
 19990

 VC(II)=D0/DSORT(VC(II))
 19990

 702
 19830

 IM1 = 1-1
 19920

 703
 IF(I,E0,1) GO TO 701
 19930

 IM1 = 1-1
 19940

 JJ = 0
 19950

 D0 705
 J=1,IM1
 19960

 JJ = J+1
 19970

 IJ = IJ+1
 19970

 IJ = IJ+1
 19970

 IJ = IJ+1
 19970

 IJ = VC(IJ) *VC(II) *VC(IJ)
 19980

225 IF(NUP.EU.S) GU IU 2/0	28280	
IF(F1(4).EQ.F2(4)) GO TO 235	28290	
230 CONTINUE	28300	
235 IF(NOP.E0.4) GO TO 270	28310	
IF(F1(5).EQ.F2(5)) GO TO 245	28320	
240 CONTINUE	28330	
245 IF(NOP.EQ.5) GO TO 270	28340	
IF(F1(6).EQ.F2(6)) GO TO 255	28350	
250 CONTINUE	28360	
255 IF(NOP.EQ.6) GO TO 270	28370	
IF(F1(7),E0,F2(7)) GO TO 270	28380	
260 CONTINUE	28390	
270 CONTINUE	28400	
DO 280 I=1,7	28410	
FNEW(I)=PP(I,IS(I))	28420	
280 STPNEH(1)=DEL(1)/2.	28430	
RETURN	28440	
1000 FORMAT(// 11X,	28450	
*" PLOT OF RESIDUAL SUM OF SQUARES VERSUS PARAMETER NO. , 12/	28460	
11X, RANGING FROM ",E11.4, TO ",E11.4)	28470	
1010 FORMAT(// 6X,	28480	
*" PLOT OF WEIGHTED RESIDUAL SUM OF SQUARES VERSUS PARAMETER",	28490	
* NO. , 12/6X, RANGING FROM , E11.4, TO , E11.4)	28500	
1020 FORMAT(/ TABLE OF PLOT ABOVE //	28510	
	20310	

WK11E(0,555) CALL READ(1,AR)	970 980	
IF(IGO.EO.2) GO TO 1001	990	
IF(ANS.ED.4HBACK) GO TO 1004	1000	
SIMP=ANS	1010	
IF(SIMP.LT.0 OR.SIMP.GT. 10.)GO TO 1007	1020	
18 IF(NS.GT.NG) GO TO 1006	1030	
20 NO=4	1040	
WRITE(6,540)	1050	
CALL READ(1,AR)	1060	
GO TO (1000,1001) IGC	1070	
21 ANS4=ANS	1080	
IF (ANS4.EQ.2HNO) GO TO 24	1090	
WRITE(6,541)	1100	
2 CALL READ(1, AR)	1110	
IF(IGD.EQ.2) GO TO 1001	1120	
IF(ANS.EQ.4HBACK) GO TO 1004	1130	
PLOTRS=ANS	1140	
IF(PLOTRS.NE.2HNO.AND.PLOTRS.NE.3HYES) GO TO 23	1150	
GO TO 24	1160	
23 WRITE(6,1020)	1170	
GO TO 22	1180	
24 IF(NS.GT.NG) GO TO 1006	1190	
25 NQ=5	1200	
WRITE(6,545)	1210	
CALL READ(1, AR)	1220	
GO TC (1003,1001)IGO	1230	

C01

24	MAN-TETUTING AN	
26	MAX=IFIX(ANS+.1)	1240
	IF(MAX.LT.1.OR.MAX.GT.2000) GO TO 1007	1250
	IF(NS.GT NQ) GO TO 1006	1260
30	NQ=6	1270
	WRITE(6,550)	1280
	CALL READ(1, AR)	1290
	GO TO (1003,1001)IGO	1300
71	NLOOP=IFIX(ANS+.1)	
21		1310
	IF(NLOOP.LT.1) GO TO 1007	1320
	IF(NLOOP.GT.MAX.AND.MAX.NE.1) GO TO 32	1330
	GO TO 33	1340
32	NAGR=5	1350
	GO TO 1008	1360
33	IF(NS.GT.NQ) GO TO 1006	1370
35	NG=7	1380
33	WRITE(6,555)	
		1240
	CALL READ(1, AR)	1400
	GO TO (1003,1001)IGO	1410
36	IPRINT=IFIX(ANS+.1)	1420
	IF(ANS.LT.OAND.ANS.GE1.) IPRINT=-1	1430
	IF(IPRINT.LT1.OR.IPRINT.GE.MAX.AND.MAX.NE.1) GO TO 1007	1440
	IF(NS.GT.NQ) GO TO 1006	1450
10	NG=8	
40		1460
	WRITE(6,560)	1470
	CALL READ(1,AR)	1480
	GO TO (1003,1001)IGO	1490
41	NFUNC=IFIX(ANS+.1)	1500
	IF(NFUNC.LT.1.OR.NFUNC.GT.10) GO TO 1007	1510
	IF(NFUNC.EQ.1) GO TO 4405	1520
	WRITE(6,561)NFUNC	1530
	NOBS=0	1540
223	D0 4400 I=1,NFUNC	1550
42	CALL READ(2,AR)	1560
	IF(IG0.EQ.2) GO TO 1001	1570
	IF(ANS.EQ.4HBACK) GO TO 1004	1580
	J=IFIX(ANS+.1)	1590
	IF(J.LT.1.0R.J.GT.10) GO TO 43	1600
	G0 T0 44	
17		1610
43	WRITE(6,1024)	1620
100	GO TO 42	1630
44	ITHSET(J)=IFIX(AR(2)+.1)	1640
	IF(ITHST(J).LT.1.OR.ITHSET(J).GT.100) GO TO 43	1650
4400	NOBS=NOBS+ITHSET(J)	1660
	GO TO 4410	1670
4405	WRITE(6,562)	1680
1403	CALL READ(1, AR)	
		1690
	!F(IG0.E0.2) GO TO 1001	1700
	IF(ANS.EQ.4HBACK) GO TO 1004	1710
	NOBS=IFIX(ANS+.1)	1720
	ITHSET(1)=NOBS	1730
4410	IF(NOUS.LT.2.OR.NOBS.GT.200) GO TO 1007	1740
	N=1	1750
	DO 4415 I=1,NFUNC	1760
	LIM1(I)=N	
		1770
	N1=N	1780
	LIM2(1)=LIM1(1)+ITHSET(1)-1	1790
	N2=LIM2(1)	1800
	N=N2+1	1810
	DO 4415 J=N1,N2	1820
4415	LABEL(J)=1	1830
	IF(NS.GT.NO) GO TO 1006	1840
45	N0=9	
43	nu=7	1850

D01

			Г
	WRITE(6,565)	1860	
	CALL FEAD(1,AR)	1870	
	GO TO (1003,1001)IGO	1880	
46	NOP=IFIX(ANS+.1)	1890 1900	
	IF(NOF.LT.1) GO TO 1007 IF(NS.GT.NO) GO TO 1006	1900	
	IF(NS.GT.NO) GO TO 1006	1910 1920	
50	NQ=10	1920	
	WRITE(6,570)NOP	1930	
	NOP2=NOP+NOP/2	1940 1950	
	DO 54 I=1,NOP2 CALL READ(3,AR)	1950	
	GO TO (1003, 1001) IGO	1960 1970	
51		1980	
	IF(AR(2).GT.AR(3)) GO TO 1009	1990	
		1770	

LALL PRUBILVARLUY, N, NH, PR/	102/0	
WRITE(6,8320)MM,N,PR	10280	
IF(PR.LT.O.C5) WRITE(6,833) RATIO=FLOAT(NRSDF)/ FLOAT(NOBS-NDUMMY)	10290	
RATIO=FLOAT(NRSDF)/ FLOAT(NOBS-NDUMMY)	10300	
IF(RATIO.LE5) WRITE(6,834)RATIO	10310	
IGNORE=1	10320	
IF(PLOTRS.NE.3 (ES) GO TO 395	10330	
IF(NIND.GT.1) GO TO 390	10340	
WRITE(6,836)	10350	
CALL PLOT(X, RES, NOBS, LINES)	10360	
WRITE(6,838)	10370	
CALL PLOT (YEST, RES, NOBS, LINES)	10380	
GO TO 391	10390	
390 WRITE(6,840)	10400	
CALL PLOT(X, RES, NOBS, LINES)	10410	
WRITE(6,842)	10420	
CALL PLOT (Z, RES, NOBS, LINES)	10420	
WRITE(6,838)	10430	
CALL DIOT/VECT DEC NODE LINES)		
CALL PLOT (YEST, RES, NOBS, LINES)	10450	
391 IF(ANS16.E0.2HNO.OR.NOWGHT.E0.1) GO TO 395	10460	
IF(NIND.GT.1) GO TO 392	10470	
WRITE(6,8360)	10480	
CALL PLOT(X, WRES, NOBS, LINES)	10490	
WRITE(6,8380)	10500	
CALL PLOT(YEST, WRES, NOBS, LINES)	10510	
GO TO 395	10520	
392 WRITE(6,8400)	10530	

COZ

CALL PLOT(X, MRES, NOBS, LINES) MFITE(6, 8420) CALL PLOT(Z, MRES, NOBS, LINES) MFITE(6, 830) CALL PLOT(YEST, MRES, NOBS, LINES) 395 CONTINUE MFITE(6, 8401) DFABS(1-AMINO/FUNC) IF(IDAD.ME.1 J GO TO 396 MFITE(6, 8402)SIMP OFABS(1-AMINO/FUNC) IF(IDAD.ME.1 J GO TO 396 MFITE(6, 8422)MEVALI, MVIOL IF(IDAD.ME.1-MVIOL D.G.T.O.1) MRITE(6, 8403)AMINO, FUNC 396 MEVAL: NEVALI-MVIOL MRITE(6, 8422)MEVALI, MVIOL IF(IDAD.CE.1.AND.IPRINT.EO.-1)MRITE(6, 8424)NEVAL2 ITABLE-1 THFUM=NFUNC MFITE(6, 8425) MG(1).GT.XMAX) XMAX=X(1) 300 3960 I=2, NOBS IF(X(1).LT.XMIN) XMIM=X(1) DO 3960 J=2, NOBS IF(X(1).LT.XMIN) XMIM=X(1) DO 397 J=1, NIND 397 G(J)=XX(J,N) MRITE(6, 8425) IGNORE=0 IF(NIND.GT.1) CALL MODEL(AA,G(1),F,1) IF(NIND.GT.1) CALL MODEL(AA,G(1),F,1) IF(NIND.GT.1) CALL MODEL(AA,G(7),F,1) ITABLE=0 MRITE(6, 844)NRUN ANSI2=2HNO ANSI2=2HNO ANSI2=2HNO ANSI2=2HNO ANSI2=2HNO ANSI2=COR.HO YPE OUESTIONS THAT FOLLOW ONLY FOUR ANSWERS"/ * OF NUMPERICAL DATA. IT ALLONS THAT FOLLOW ONLY FOUR ANSWERS"/ * OR IN SHORT FORM Y YES NO HACK REQUESTS THAT ASK?/ * OR NUMPERICAL DATA. IT ALLONS ANY THEOR TO BE ON ANSWERS"/ * OR NUMPERICAL DATA. IT ALLONS ANY THEOR TO BE ON ANSWERS"/ * OR NUMPERICAL DATA. IT ALLONS ANY THEOR TO BE ON ANSWERS '/ * OR NUMPERICAL DATA. IT ALLONS ANY THEOR TO BE ON ANSWERS '/ * OR NUMPERICAL DATA. IT ALLONS AND CORFECT THE INFORMATION ABOUT TINE THAT ASK?/ * OR NUMPERICAL DATA. IT ALLONS AND CORFECT THE ANY TO EDBCK AND CORF.'T ANY '/ * TRAMERED /* BACKAAND OR FORMARD TO REDUEST NUMPER '/'' * TRAMERED /* BACKAAND OR FORMARD TO REDUEST NUMPER '/'' * TRAMERED /* BACKAAND CORFECT THE INFORMATION THE ANN THE ANY '/ * TRAMERED /* BACKAAND OR FORMARD TO REDUEST NUMPER */'' * TRAMERED /* BACKAAND CORFECT THE INFORMATIO REDUEST NUMPER */'' * TRAMERED /* 10550 10560 10570 10740 10750 10920 10930 10950 10960 10970 10980 10990 11000 11010 11090 11109 11110 11110 11120 2 FORMAT(* " A PARAMETER CAN BE CHANGED TO A CONSTANT IN REQUEST NO.10 BY"/ * " ASSIGNING IT THE SAME LOWER AND UPPER LIMITS. THIS ENABLES ANY"/ * " SUM-OF-SQUARES CONTOUR TO BE EVALUATED (REQUEST NO.19) BY "/ * " FIXING ALL EXECEPT 2 PARAMETERS."/ * " A LATTICE SEARCH FOR OPTIMAL INITIAL PARAMETER ESTIMATES CAN"/ 11140 11150

DOZ

*" BE MADE BEFORE THE START OF THE FITTING PROCEDURE BY ENTERING"/	11160
* 1 AS THE MAX. NO. CF SUM-OF-SQUARES FUNCTION EVALUATIONS"/ * ALLOWED IN REQUEST NO.5."/	11170
. THE DEFAULT VALUE SPECIFIED IN SOME REQUESTS CAN BE ENTERED IF /	11190
*" THERE IS DOUBT ABOUT WHICH VALUE TO ENTER."/	11200
* FOR MORE DETAILED INFORMATION ABOUT THE PROGRAM CONTACT"/ * P.VENG PEDERSEN. //)	11210 11220
O FORMAT(" (2) ENTER A HEADING BUT NOT MORE THAN ONE LINE"//)	11230
5 FORMAT(80H	11240
O FORMAT(" (3) DO YOU WANT STATISTICAL EVALUATIONS OF THE",	11250 11260
*" PARAMETERS (DEFAULT=YES)"//)	11270
5 FORMAT(ENTER THE EXPANSION CRITERION TO AVOID ROUNDING-OFF , * ERRORS // IN THE STATISTICAL EVALUATION OF THE PARAMETERS .	11260
* ERRORS / IN THE STATISTICAL EVALUATION OF THE PARAMETERS .	11290

8	1J = J+(J-1)/2+1	19580	d. et Al al 1	
	VC(IJ)=0D0 D0 Z31 k=1,NAP	19590 19600		
	231 VC(IJ)=VC(IJ)+H(K)+DBLE(C(K,J))	19610		
	230 CONTINUE	19620		
	224 CONTINUE	19630		
	J_= 0	19640		
	DO 234 1=1,NOP	19650		
	J = J + I $234 VAR(I) = VC(J)$	19660		
	236 IF(IPRINT.GE.O) WRITE(IW, 306) IRANK	19670 19680		
	306 FORMAT(/ INFORMATION MATRIX HAS RANK 13//	19690		
	306 FORMAT(/" INFORMATION MATRIX HAS RANK",13// *" GENER/ IZED INVERSE OF INFORMATION MATRIX"/)	19700		
	1JK=1	19710		
	GO TO 710	19720		
	717 CONTINUE IF(IPRINT.GE.O) WRITE(IW,308)	19730		
	308 FORMAT(19740 19750		
	. THE MATRIX ABOVE MUST BE MULTIPLIED BY TWICE THE ESTIMATE OF ,	19760		
	* THE RESIDUAL // VARIANCE TO GIVE THE VARANCE-COVARIANCE MATRIX , * OF THE PARAMETERS. /)	19770		
	*" OF THE PARAMETERS. "/)	19780		
	CALL SYMINV(VC, NAP, BMAT, TEMP, NULLTY, IFAULT)	19750		
	IF(IPRINT.GE.O) WRITE(IW, 320)	19800		
	320 FORMAT(/ INFORMATION MATRIX (HESSIAN MATRIX)" /) IJk=3	19810		
	GO TO 710	19830		

C03

711	1 8-3	101/4
	IJK=2 II = 0	19840 19850
	ij = o	19860
	DO 701 I=1,NPR	19870
	11 = 11+1	19880
	IF(VC(II).LE.0D0) GO TO 702	19890
	VC(II)=1D0/DSORT(VC(II))	19900
	GO TO 703	19910
702	VC(II)=0D0 IF(I.E0.1) GO TO 701	19920
	IM1 = I-1	19940
	0 = LL	19950
	DO 705 J=1,IM1	19960
	JJ = JJ+J	19970
		19980
705	VC(IJ) = VC(IJ) * VC(II) * VC(JJ) IJ = IJ+1	19990 20000
	IF(IPRINT.GE.0) WRITE(IW, 312)	20010
312	FORMAT(CORRELATION MATRIX /)	20020
	II = 0	20030
	D0 706 I=1,NPR	20040
		20050
706	IF(VC(II).NE.ODO) VC(II)=1D0 CONTINUE	20060 20070
	GO TO 710	20080
712	IF(IPRINT.GE.O) WRITE(IW, 310)NEVAL	20090
310	FORMAT(" A FUTHER", 14, " FUNCTION EVALUATIONS HAVE BEEN USED "/)	20100
	NEVAL2=NEVAL	20110
375	IF(NFIX.GT.O.AND.IPRINT.GE.O) WRITE(6,3020)NFIX	20120
710	RETURN	20130 20140
	IF(L.GT.NPR) GO TO (717,712,711), IJK	20150
	11=L*(L-1)/2	20160
	DO 713 I=L,NPR	20170
	11 = 11+L	20180
		20190
	12=MINO(II,11+5) IF(IJK.EQ.3) GO TO 718	20200 20210
	IF(IPRINT.GE.0) WRITE(IW,714)(VC(J), J=11,12)	20220
	IF(IJK.NE.1) GO TO 713	20230
	D0 500 J=I1,I2	20240
	VARCGV(J)=VC(J)	20250
71.0		20260
713	IF(IPR'NT.GE.O) WRITE(IW,714)(BMAT(J),J=11,I2) CONTINUE	20270 20280
	FORMAT(1X,6013 6)	20290
	WRITE(IW, 715)	20300
	FORMAT(1H ,/)	20310
	L=L+6	20320
	GO TO 716 END	20330
		20340 20350
	SUBROUTINE CHOL (A, N, U, NULLTY, IFAULT)	20360
3	DIMENSION A(55),U(55)	20370
3	DOUBLE PRECISION A, U, W, ETA	20380
	ETA=1D-16	20390
	NN=0 IFAULT=1	20400
	IF (N.LE.O) GO TO 100	20410 20420
	IFAULT=2	20420
1	NULLTY=0 J=1	20440

K=0 D0 10 ICDL = 1,N	20460 20470 20480	
DO 10 ICOL = 1, N	20470	
L=0 DO 11 IROW = 1,ICOL	20480	
DO 11 IROW = 1,ICOL	20490 20500 20510 20520 20530 20540	
K=K+1	20500	
H=A(K)	20510	
M=J DO 12 I = 1,IROW	20520	
DO 12 I = 1, IROW	20530	
L=L+1	20540	
IF (I.EQ.IROW) GO TO 13	20550	
H=H-U(L)+U(M)	20560	
M=H+1	20570	
2 CONTINUE	20580	
IF (IROW.EO.ICOL) GO TO 14	20550 20560 20570 20580 20590	

00	TO 44	1610	
	ITE(6,1024)	1620	
GO	TO 42	1630	
44 ITH	HSET(J)=IFIX(AR(2)+.1)	1640	
IFO	(ITHSET(J).LT.1.OR.ITHSET(J).GT.100) GO TO 43	1650	
4400 NOE	3S=NOBS+ITHSET(J)	1660	
	TO 4410 ITE(6,562)	1670	
4405 MR	L READ(1, AR)	1680 1690	
IF	(IGO.EQ.2) GO TO 1001	1700	
IFO	(ANS.EQ. 4HBACK) GO TO 1004	1710	
NOE	BS=IFIX(ANS+.1)	1720	
17	HSET(1)=NOBS	1730	
4410 IF	(NOBS.LT.2.OR.NOBS.GT.200) GO TO 1007	1740	
N=1		1750	
00	4415 I=1,NFUNC 41(I)=N	1760 1770	
N1=		1780	
	12(1)=LIM1(1)+ITHSET(1)-1	1790	
N2:	LIMZ(I)	1800	
N=M	12+1	1810	
	4415 J=N1,N2	1820	
4415 LAE		1830	
	NS.GT.NQ) GO TO 1006	1840	
45 NG=	-7	1850	

D01

H	RITE(6,565)	1860
C	ALL FEAD(1, AR)	1870
G	0 TO (1003,1001)IGO	1880
46 N	OP=IFIX(ANS+.1)	1890
I	F(NOP.LT.1) GO TO 1007	1900
I	F(NS.GT.NQ) GO TO 1006	1910
50 N	Q=10	1920
H	RITE(6,570)NOP	1930
	OP2=NOP+NOP/2	1940
D	0 54 I=1,NOP2	1950
C	ALL READ(3, AR)	1960
G	0 TO (1003,1001)IGO	1970
51 J	=IFIX(ANS+.1)	1980
I	F(AR(2).GT.AR(3)) GO TO 1009	1990
1	F(J.LT.1.OR.J.GT.NOP) GO TO 52	2000
G	O TO 53	2010
	AGR=9	2020
	D TO 1008	2030
	1(J)=AR(2)	2040
	2(J)=AR(3)	2050
	F(J.EQ.NOP) GO TO 5400	2060
	ONTINUE	2070
H	RITE(6,571)NOP	2080
E LOO G	0 TO 50	2090
5400 1	F(NS.GT.NO) GO TO 1006	2100
55 N		2110
	RITE(6,575)	2120
	ALL READ(1, AR)	2130
54 4	0 TO (1000,1001) IGO NS11=ANS	2140
	F(ANS.EQ.2HNO) GO TO 66	2150
		2160
	RITE(6,580) ALL READ(1,AR)	2170
TI I	F(IGO.EO.2) GO TO 1001	2180
1	F(ANS.EQ. 4HBACK) GO TO 1004	2190
Ň	11=1FIX(ANS+.1)	2200 2210
Ĩ.	F(N11.LT.1) GO TO 1007	2220
Î	F(N11.GT.NOP) GO TO 57	2230
	0 TO 58	2240
	AGR=9	2250
	0 TO 1008	2260
	F(N11.GT.1) GO TO 59	2270
Ú.	RITE(6,585)	2280
P.4	TO 60	2290
59 W	RITE(6,590)N11	2300
60 C	ONTINUE	2310
	0 65 I=1,N11	2320
	ALL READ(2, AR)	2330
11	F(IGO.EQ.2) GO TO 1001	2340
11	F(ANS.EQ.4HBACK) GO TO 1004	2350
J	=IFIX(ANS+.1)	2360
	F(J.LT.1) GO TO 1007	2370
	F(J.GT.NOP) GO TO 61	2380
G	0 TU 62	2390
	AGR=9	2400
	0 TO 1008	2410
	F(AR(2).LT.F1(J).OR.AR(2).GT.F2(J)) GO TO 63	2420
	0 TO 64	2430
	AGR=10	2440
G	0 TO 1008	2450
6/ NI	F(1)=J	2460
	S(J)=AR(2)	

E01

66 IF(NS.GT.NQ) GO TO 1006	2480	
70 NQ=12	2490	- 1
WRITE(6,595)	2500	- 1
CALL READ(1,AR)	2510	- 1
GO TO (1000,1001) IGP	2520	
71 ANS12=ANS	2530	
IF(ANS.EQ.2HNO) GO TO 81	2540	- 1
WRITE(6,600)	2550	
CALL READ(1, AR)	2560	- 1
IF(IGO.EQ.2) GO TO 1001	2570	
IF(ANS.ED. 4HBACK) GO TO 1004	2580	- 1
N12=1F1X(ANS+.1)	2590	
1F(N12.LT.1) GO TO 1007 1F(N12.GT.NOP) GO TO 72	2600	
CO TO 77	2610	
GO TO 73	2620	

+66X," VERSION 7-76"/) 505 FORMAT(" (1) DO YOU WANT SOME INFORMATION ABOUT THIS PROGRAM",	10910
*" (TYPE YES OR NO)" //)	10920
511 FORMAT(/	10940
*" IN THE YES-OR-NO TYPE QUESTIONS THAT FOLLOW, ONLY FOUR ANSWERS"/	10950
* ARE ALLOWED, NAMELY YES NO BACK REQUEST 7	10960
*" OR IN SHORT FORM Y N B R"/	10970
 *" BACK (B) CAN ALSO BE ENTERED AT ANY TIME ON REQUESTS THAT ASK"/ *" FOR NUMERICAL DATA. IT ALLOWS YOU TO GO BACK AND CORRECT ANY"/ 	10980
* OF YOUR PREVIOUS INPUTS AND THEN CONTINUE FROM CURRENT INPUT"/	11000
*" REQUEST. SIMPLY TYPE BACK** (OR B**) ANYWHERE ON A LINE IF YOU"/	11010
*" WANT TO GO BACK AND CORRECT THE INPUT UNDER REQUEST NUMPER **."/	11020
*" THE B** FEATURE ALSO MAKES IT EASY TO REPEAT A RUN WITH ONLY"/ *" A FEW CHANGES IN THE INPUTS. BY TYPING R** YOU WILL BE ",	11030
*"TRANFERED"/" BACKWARD OR FORWARD TO REQUEST NUMBER **"/	11050
* THE R-COMMAND AND THE DEFAULT VALUES ENABLE A FAST INPUT. "/	11060
 INPUT ERRORS IN REQUEST NO.10 OR 17 CAN BE CORRECTED", 	11070
* BY RETYPING / THE WRONG LINE(S) IN CORRECT FORM. THE LAST LINE	11080
*," STARTS THE NEXT REQUEST."/) 512 FORMAT(11090
*" A PARAMETER CAN BE CHANGED TO A CONSTANT IN REQUEST NO.10 BY"/	11110
*" ASSIGNING IT THE SAME LOWER AND UPPER LIMITS. THIS ENABLES ANY"/	11120
* SUM-OF-SQUARES CONTOUR TO BE EVALUATED (REQUEST NO. 19) BY "/	11130
 FIXING ALL EXECEPT 2 PARAMETERS."/ A LATTICE SEARCH FOR OPTIMAL INITIAL PARAMETER ESTIMATES CAN"/ 	11140
A LATTICE SEARCH FOR OFTIMAL INITIAL PARAMETER ESTIMATES LAN /	11150

DOZ

THE MADE REFORE THE ATART OF THE FITTING PROPERTIES BY ENTERING .	
*" BE MADE BEFORE THE START OF THE FITTING PROCEDURE BY ENTEPING"/	11160
*" 1 AS THE MAX. NO. CF SUM-OF-SQUARES FUNCTION EVALUATIONS"/	11170
* ALLOWED IN REQUEST NO.5. 7	11180
* THE DEFAULT VALUE SPECIFIED IN SOME REQUESTS CAN BE ENTERED IF "/	11190
*" THERE IS DOUBT ABOUT WHICH VALUE TO ENTER."/	11200
* FOR MORE DETAILED INFORMATION ABOUT THE PROGRAM CONTACT /	11210
* P.VENG PEDERSEN. //)	11220
520 FORMAT(" (2) ENTER A HEADING BUT NOT MORE THAN ONE LINE"//)	11230
525 FORMAT(80H	11240
	11250
530 FORMAT(" (3) DO YOU WANT STATISTICAL EVALUATIONS OF THE",	11260
* PARAMETERS (DEFAULT=YES) //)	11270
TO THE PERSON THE PERSON OF TERMON TO MARK DOWNSTAND OFFIC	
535 FORMAT(" ENTER THE EXPANSION CRITERION TO AVOID ROUNDING-OFF",	11280
*" ERRORS 7" IN THE STATISTICAL EVALUATION OF THE PARAMETERS",	11290
* (DEFAULT=1E-6) //)	
	11300
540 FORMAT(" (4) DO YOU WANT AN ANALYSIS OF THE RESIDUALS".	11310
*" (DEFAULT=NO)" //)	11320
541 FORMAT(" DO YOU WANT PLOTS OF THE RESIDUALS"//)	11330
545 FORMAT(" (5) HOW MANY SUM-OF-SQUARES FUNCTION EVALUATIONS WILL",	11340
* YOU ALLON / TO REACH CONVERGENCE ON MINIMUM (DEFAULT=800) //)	11350
550 FORMAT(" (6) THE CHECK FOR FINAL CONVERGENCE IS DONE EVERY NLOOP"/	11360
* TIMES THAT THE PARAMETERS ARE CHANGED BY THE ITERATIVE PROCESS /	11370
*" ENTER THE VALUE YOU WANT FOR NLC (DEFAULT=20)"//)	11380
555 FORMAT(" (7) ENTER THE DEGREE OF OUTPUT YOU WANT FROM"/	11390
* THE FUNCTION MINIMIZATION PROCESS (DEFAULT=-1)"//	
	11400
*1X, "NUMBER= FULL PROGRESS REPORT EVERY *NUMBER* FUNCTION",	11410
•" ÉVALUATIONS"/	11420
*5X, 0= PARTIAL REPORT"/	11430
<pre>*5X, 0= PARTIAL REPORT"/ *5X, -1= NO DUTPUT //> 560 FORMAT(*(8) HOW MANY FUNCTIONS OR RESPONSE SYSTEMS ARE TO BE".</pre>	11440
540 FORMAT("+(1) HOU MANY CUNCTIONS OF RECOONSE SYSTEMS ADD TO DE"	
	11450
*" FITTED"/" SIMULTANEOUSLY BY LEAST SQUARES (DEFAULT=1)"//)	11460
561 FORMAT(" ENTER", 12, "LINES CONTAINING THESE THO NUMBERS IN THE",	11470
*" FOLLOWING ORDER //5X," (1) THE NUMBER OF THE FUNCTION TO BE",	11480
*" FITTED (1.E. 1,2,3)"/5X," (11) THE NUMBER OF OBSERVATIONS",	11490
* FOR THE FUNCTION //)	11500
562 FORMAT(" HOW MANY OBSERVATIONS DO YOU HAVE"//)	11510
565 FORMAT("*(9) HOW MANY PARAMETERS (I.E. PARAMETERS+CONSTANTS)",	11520
+" DOES THE FUNCTION(S)"/" HAVE (A CONSTANT IS A DADAMETED"	
DOLD THE TORUST I THATE. TA CONSTANT IS A PARALETER ,	11530
• " WITH LOWER LIMIT = UPPER LIMIT)"//)	11540
570 FORMAT("*(10) ENTER", 12," LINES EACH CONTAINING THESE 3 NUMBERS",	11550
The FOLLOWING OPDER"//	
*" IN THE FOLLOWING ORDER"//	11560
*,5X," (I) THE INDEX OF THE PARAMETER (I.E. 1,2,5ETC)"/	11570
*,5X," (II) ITS LOWER LIMIT"/5X," (III) ITS UPPER LIMIT"//)	
13A, 111/ 113 LUNER LINIT /JA, 1111/ 113 UPPER LINIT //)	11580
571 FORMAT(/" YOU FORGOT TO ENTER THE LAST PARAMETER (NO.", 12,	11590
•") -TRY AGAIN-"/)	11600
575 FORMAT(" (11) DO YOU WANT TO GIVE AN INITIAL ESTIMATE OF ONE OR"/	11610
* MORE OF YOUR PARAMETERS (DEFAULT=NO)"//)	11620
SEO EODMATI" EOD LICU WANY DADAMETEDS" / /)	11630
585 FORMAT(ENTER O N E LINE CONTAINING THE INDEX OF THE PARAMETER"/	
585 FURMATC ENTER UN E LINE CONTAINING THE INDEX OF THE PARAMETER"/	11640
	11650
590 FORMAT(" ENTER FOR EACH OF THESE", 12, " PARAMETERS ONE"/	
JTO FURNALL ENTER FUR EACH UP THESE , 12, PARAMETERS UNE /	11660
*" LINE CONTAINING ITS INDEX FOLLOWED BY ITS VALUE."//)	11670
595 FORMAT(" (12) DO YOU WANT TO GIVE A STARTING STEP SIZE FOR ONE"/	11680
" OD MODE OF YOUR DUBANTTOO (DEE A STANTING STEP SIZE FOR ONE /	
* OR MORE OF YOUR PARAMETERS (DEFAULT=NO)"//)	11690
600 FORMAT(" FOR HOW MANY PARAMETERS"//)	11700
ANS ENDWAT /" ENTED O N E LINE CONTAINING THE INDEX OF THE DADAMETER	
605 FORMAT(" ENTER O N E LINE CONTAINING THE INDEX OF THE PARAMETER "/	11710
*" FOLLOWED BY ITS STEP SIZE."//)	11720
610 FORMAT(" ENTER FOR EACH OF THESE", 12," DIFFERENT PARAMETERS ONE"/	11730
* LINE CONTAINING ITS INDEX FOLLOWED BY ITS STEP SIZE. "//)	11740
615 FORMAT(" (13) ENTER THE CONVERGENCE CRITERION FOR THE SUM OF",	11750
*" SQUARES / MINIMIZATION PROCESS (DEFAULT=0.001) //)	
	11760
620 FORMAT(" (14) HOW MANY INDEPENDENT VARIABLES DOES YOUR FUNCTION",	11770

E02

_		
	<pre>*" HAVE (DEFAULT=1)" //)</pre>	11780
	625 FORMAT(" (15) DO YOU WANT A PLOT OF THE FITTED CURVE (DEFAULT=NO)"	11790
	*//)	11800
	635 FORMAT(" (16) DO YOU WANT TO WEIGHT YOUR DATA (DEFAULT=NO)" //)	11810
	636 FORMAT(" ENTER ONE OF FOLLOWING 4 DIGITS ACCORDING TO THE",	11820
	* WEIGHTING YOU PREFER "/" (Y= DEPENDENT VARIABLE. YOU CAN",	11830
	*" SPECIFY R IN NEXT INPUT REQUEST)"// *" 1 THE WEIGHT TO BE SPECIFIED FOR EACH ORSERVATION"	11840
		11850
	*/" 2 WEIGHT = Y**R (R FOR EXAMPLE = -2,1,1/2 ETC.)"/ *" 3 WEIGHT = LOG(R*Y)	11860
	*/" 2 WEIGHT = Y**R (R FOR EXAMPLE = -2,1,1/2 ETC.)"/ *" 3 WEIGHT = LOG(R*Y) *" 4 WEIGHT = EXP(R*Y) "/)	11870
	637 FORMAT(" ENTER THE VALUE FOR R IN THE PREFERRED WEIGHTING SCHEME",	11890
	*" NO.", 12," ABOVE" //)	11900
	640 FORMAT("*(17) ENTER A LINE FOR EACH OF YOUR", 13, " OBSERVATIONS."/	11910
	* EACH LINE MUST CONTAIN ,12, NUMBERS IN THE FOLLOWING ORDER	11920
	+//10Y " (1) THE OBCEDVATION NUMBED"/	11070

IF(IJK.ME.1) GO TO 713 DO 500 J=11,I2 S00 VARCGV(J)=VC(J) GO TO 713 718 IF(IPR'NT.GE.0) WRITE(IW,714)(BMAT(J),J=11,I2) 713 CONTINUE 714 FORMAT(1X,6013 6) WRITE(IW,715) 715 FORMAT(1H ,/) L=L+6 GO TO 716 END C SUBROUTINE CHOL (A,N,U,NULLTY,IFAULT) DIMENSION A(55),U(55) DOUBLE PRECISION A,U,W,ETA ETA=1D-16 NN=0 IFAULT=1 IF (N.LE.0) GO TO 100 IFAULT=2 NULLTY=0	20220 20230 20230 20240 20250 20250 20260 20280 20290 20300 20310 20310 20310 20310 20350 20550
---	---

D03

K=0	20460
DO 10 ICOL = 1.N	20470
L=0 DO 11 IROW = 1,ICOL	20480
K=K+1	20490 20500
W=A(K)	20510
M=J	20520
DO 12 I = 1, IROW	20530
L=L+1 IE (I EO IDON) CO TO 17	20540
IF (I.EO.IROW) GO TO 13 W=W-U(L)+U(M)	20550 20560
M=H+1	20570
12 CONTINUE	20580
13 IF (IROW.EQ.ICOL) GO TO 14	20590
IF(U(L).EQ.0D0) GO TO 21	20600
U(K)=W/U(L) GO TO 11	20610 20620
21 U(K)=0D0	20630
11 CONTINUE	20640
14 IF(W.LE.DABS(ETA+A(K)))GO TO 20	20650
U(K)=DSQRT(W)	20660
GO TO 15 20 U(K)=0D0	20670
IF(W.LT.ODO)NN=NN+1	20680 20690
NULLTY=NULLTY+1	20700
15 J=J+ICOL	20710
10 CONTINUE	20720
IFAULT=0	20730
IF(N.LE.4.AND.NN.GT.1) IFAULT=2	20740
IF(N.LE.6.AMD.NN.GT.2) IFAULT=2 IF(NN.GT.3.OR.N.EQ.NN) IFAULT=2	20750 20760
100 RETURN	20770
END	20780

SUBROUTINE SYMINY (A, N, C, W, NULLTY, IFAULT)	20800
DIMENSION A(55),C(55),W(20) DOUBLE PRECISION A,C,W,X	20810 20820
NROW=N	20820
IFAULT=1	20840
IF (NROW.LE.O) GO TO 100	20850
IFAULT=0	20860
CALL CHOL(A,NROW,C,NULLTY,IFAULT) IF (IFAULT.NE.0) GO TO 100	20870
NN=(NROW+())/2	20880 20890
IROW=NROW	20900
NDIAG=NN	20910
16 IF(C(NDIAG).EQ.0D0) GO TO 11	20920
	20930
DO 10 I=IROW,NROW W(I)=C(L)	20940 20950
	20960
10 CONTINUE	20970
I COL =NROW	20980
JCOL=NN	20990
MDIAG=NN 15 L=JCOL	21000
X=0D0	21010 21020
IF(ICOL.EQ.IROW)X=1D0/H(IROW)	21030
K=NROW	21040
13 IF (K.EQ.IROW) GO TO 12	21050
X=X-W(K)+C(L)	21060
K=K-1	21070

E03

L=L-1 IF (L.GT.MDIAG) L=L-K+1 GO TO 13 12 C(L)=X/N(IROW) IF (ICOL.EC.IROW) GO TO 14 MOIAG=MDIAG-ICOL ICOL=JCOL-1 JCOL=JCOL-1 GO TO 15 11 L=NDIAG DO 17 J=IROW, NROW C(L)=ODO L=L+J 17 CONTINUE 14 MDIAG=NDIAG-IROW IROW=IROW-1

GO TO 58	2240	
57 NAGR=9 GO TO 1008	2250 2260	
58 IF(N11.GT.1) GO TO 59	2270	
WRITE(6.585)	2280	
59 WRITE(6,590)N11	2290	
59 WRITE(6,590)N11	2300	
60 CONTINUE	2310	
DO 65 I=1,N11 CALL READ(2,AR)	2320 2330	
LE(IGO EO 2) GO TO 1001	2330	
IF(IGO.EO.2) GO TO 1001 IF(ANS.EO.4HBACK) GO TO 1004	2350	
J=IFIX(ANS+.1)	2360	
IF(J.LT.1) GO TO 1007	2370	
IF(J.GT.NOP) GO TO 61 GO TO 62	2380	
61 NAGR=9	2390 2400	
GO TO 1008	2410	
62 IF(AR(2).LT.F1(J).OR.AR(2).GT.F2(J)) GO TO 63	2420	
GO TO 64	2430	
63 NAGR=10	2440	
GD TO 1008 64 NF(I)=J	2450	
65 FS(J)=AR(2)	2460 2470	
05 1000-ARE/	24/0	

E01

and the second se			
66 IF(NS.GT.NO) GO TO	1006	2480	
70 NQ=12		2490	
WRITE(6,595)		2500	
CALL READ(1,AR) GO TO (1000,1001) I	· •	2510	
71 ANS12=ANS	36	2520 2530	
IF(ANS.EQ.2HNO) GO	TO 81	2540	
WRITE(6,600)		2550	
CALL READ(1, AR)	****	2560	
IF(IGO.EO.2) GO TO IF(ANS.EO.4HBACK) G	0 TO 1004	2570 2580	
N12=IFIX(ANS+.1)	3 10 1004	2590	
IF(N12.LT.1) GO TO	1007	2600	
IF(N12.GT.NOP) GO T	0 72	2610	
GO TO 73 72 NAGR=9		2620	
GO TO 1008		2630 2640	
73 IF(N12.GT.1) GO TO	74	2650	
WRITE(6,605)		2660	
GO TO 75		2670	
74 WRITE(6,610)N12 75 CONTINUÉ		2680 2690	
DO 80 1=1 N12		2700	
CALL READ(2, AR)		2710	
IF(IGO.E0.2) GO TO	1001	2720	
IF(ANS.EQ.4HBACK) G J=IFIX(ANS+.1)	J TO 1004	2730	
IF(J.LT.1) GO TO 10	07	2740 2750	
IF(J.GT.NOP) GO TO	76	2760	
GO TO 77		2770	
76 NAGR=9		2780	
GO TO 1008 77 IF(ABS(AR(2)).GE.(F	2(1)-E1(1))) CO TO 78	2790 2800	
GO TO 79		2810	
78 NAGR=10		2820	
GO TO 1008		2830	
79 NT(1)=J 80 STPS(J)=AR(2)		2840 2850	
81 IF(NS.GT.NG) GO TO	1006	2360	
85 NQ=13		2870	
WRITE(6,615)		2880	
CALL READ(1,AR) GO TO (1003,1001)IG	•	2890	
86 IF (ANS.LT.OOR.ANS	GT 1) GO TO 1007	2900 2910	
STOPCR=ANS		2920	
IF(NS.GT.NQ) GO TO	1006	2930	
90 NG=14 WRITE(6,620)		2940	
92 CALL READ(1.AR)		2950 2960	
GO TO (1003,1001) IG 91 NIND=IFIX(ANS+.1)	0	2970	
91 NIND=IFIX(ANS+.1)		2980	
IF(NIND.LT.1.OR.NIN	J.GT.9) GO TO 93	2990	
GO TO 94 93 HRITE(6, 1024)		3000 3010	
93 WRITE(6,1024) GO TO 92		3020	
94 IF(NS.GT.NQ) GO TO	1006	3030	
95 NG=15		3040	
IF(NIND.GT.1) GO TO WRITE(6,625)	41	3050	
CALL READ(1, AR)		3060 3070	
GO TC (1000,1001) I	30	3080	
96 PLT=ANS			

F01

 97
 IF(NS.GT.NQ) GO TO 1006
 3100

 100
 NQ=15;
 3110

 WRITE(6,635)
 3120

 CALL READ(1,AR)
 3130

 GO TO (1000,1001) IGO
 3140

 101
 AMS16=ANS

 IMCHT=0
 3160

 IMCHT=0
 3160

 IF(AMS16.EQ.2HNO) GO TO 102
 3160

 IF(G,636,
 3180

 CALL READ(1,AR)
 3190

 IF(IGO.EQ.2) (ABACK) GO TO 1001
 3200

 IF(AMS16.EQ.4HBACK) GO TO 1004
 3210

 IF(IGO.EQ.2) (AHBACK) GO TO 1004
 3210

 IF(IGO.EQ.2) (GO TO 1004
 3220

 IF(IMGHT.LT.1.OR.INGHT.CT.4) GO TO 1007
 3230

 IF(IMGHT.EQ.1) GO TO 102
 3240

 WRITE(6,637) IMGHT
 3250

 IF(IMCHT.EQ.1) ADA
 3250

* WITH LOWER LIMIT = UPPER LIMIT) //)	11540
570 FORMAT("*(10) ENTER", 12," LINES EACH CONTAINING THESE 3 NUMBERS",	11550
• IN THE FOLLOWING ORDER //	11560
•,5X, (1) THE INDEX OF THE PARAMETER (I.E. 1,2,3ETC)"/ •,5X," (II) ITS LOWER LIMIT"/5X," (III) ITS UPPER LIMIT"//)	11570
571 FORMAT(/" YOU FORGOT TO ENTER THE LAST PARAMETER (NO.", 12,	11580 11590
*") -TRY AGAIN-"/)	11600
575 FORMAT(" (11) DO YOU WANT TO GIVE AN INITIAL ESTIMATE OF ONE OR"/	11610
* MORE OF YOUR PARAMETERS (DEFAULT=NO)"//)	11620
580 FORMAT(" FOR HOW MANY PARAMETERS"//) 585 FORMAT(' ENTER O N E LINE CONTAINING THE INDEX OF THE PARAMETER"/	11630 11640
*" FOLLOWED BY ITS VALUE."//)	11650
590 FORMAT(" ENTER FOR EACH OF THESE", 12, " PARAMETERS ONE"/	11660
*" LINE CONTAINING ITS INDEX FOLLOWED BY ITS VALUE."//) 595 FORMAT(" (12) DO YOU WANT TO GIVE A STARTING STEP SIZE FOR ONE"/	11670
*" OR MORE OF YOUR PARAMETERS (DEFAULT=NO)"//)	11680 11690
600 FORMAT(" FOR HOW MANY PARAMETERS"//)	11700
605 FORMAT(" ENTER O N E LINE CONTAINING THE INDEX OF THE PARAMETER "/	11710
*" FOLLOWED BY ITS STEP SIZE."//) 610 FORMAT(" ENTER FOR EACH OF IMESE", 12," DIFFERENT PARAMETERS ONE"/	11720
*" LINE CONTAINING ITS INDEX FOLLOWED BY ITS STEP SIZE."//)	11740
615 FORMAT(" (13) ENTER THE CONVERGENCE CRITERION FOR THE SUM OF",	11750
* SQUARES // MINIMIZATION PROCESS (DEFAULT=0.001) //)	11760
620 FORMAT(" (14) HOW MANY INDEPENDENT VARIABLES DOES YOUR FUNCTION",	11770

EOZ

 HAVE (DEFAULT-1). ///.) COST (ADD) TOU MANT A PLOT OF THE FITTED CURVE (DEFAULT-NO). 11700 COST (ADD) TOU MANT TO WEIGHT YOUR DATA (DEFAULT-NO). 11810 COST (ADD) TOU MANT TO WEIGHT YOUR DATA (DEFAULT-NO). 11810 COST (ADD) TO TO USE SPECIFIC FOR EACH (DEFAULT-NO). 11810 SECTIFY B IN MEXT INFUT REQUEST) // 11810 SECTIFY B IN MEXT INFUT REQUEST // 11810 SECTIFY B IN MEXT INFUT REQUEST // 11810 SECTIFY B INFUT REGIFY REPORT VARIABLE (SECTIFY) 11800 SECTIFY B INFUT REGIFY D INFUT REGIFY VARIABLE (SECTIFY) 11800 SECTIFY B INFUT REGIFY D INFUT REGIFY VARIABLE (SECTIFY) 11800 SECTIFY B INFUT REGIFY D INFUT REGIFY VARIABLE (SECTIFY) 11800 SECTIFY B INFUT REGIFY D INFUT REGIFY VARIABLE (SECTIFY) 11800 SECTIFY B INFUT REGIFY D INFUT REGIFY VARIABLE (SECTIFY) 11800 SECTIFY B INFUT REGIFY VARIABLE (SECTIFY) 11800 SECTIFY B INFUT REGIFY D INFUT REGIFY VARIABLE (SECTIFY) 11800 SECTIFY B INFUT REGIFY D INFUT REGIFY VARIABLE (SECTIFY) 11800 SECTIFY B INFUT REGIFY D INFUT REGIFY VARIABLE (SECTIFY) 11800 SECTIFY B INFUT REGIFY D INFUT RE		
625 FORMATC (15) DO YOU MANT A PLOT OF THE FITTED CURVE (DEFAULT-ND): 1176 637 ///DV 643 FORMATC - ENTER ONE OF FOLLOWING 4 DIGITS ACCOMDING TO THE - // 1180 75 SPECIFY R IN MEXT THOUT REQUESTS // 1180 75 SPECIFY R IN MEXT THOUT REQUESTS // 1180 76 SPECIFY R IN MEXT THOUT REQUESTS // 1190 76 SPECIFY R IN MEXT SPECIFY R IN THE FOLLOWING ONDER 77 SPECIFY R IN MEXT SPECIFY R IN THE FOLLOWING ONDER 76 SPECIFY R IND PART SPECIFY R IN THE FOLLOWING ONDER 77 SPECIFY R IND PART SPECIFY R IN THE FOLLOWING ONDER 77 SPECIFY R IND PART SPECIFY R IN THE FOLLOWING ONDER 77 SPECIFY R IND PART SPECIFY R IN THE FOLLOWING ONDER 77 SPECIFY R IND PART SPECIFY R IN THE FOLLOWING ONDER 77 SPECIFY R IND PART SPECIFY R IN THE FOLLOWING ONDER 77 SPECIFY R IND PART SPECIFY R IN MERES IN THE FOLLOWING ONDER 77 SPECIFY R IND PART SPECIFY R IN MERES IN THE FOLLOWING ONDER 77 SPECIFY R IND PART SPECIFY R IN MERES IN THE FOLLOWING ONDER 77 SPECIFY R IND PART SPECIFY R IN MERES IN THE FOLLOWING ONDER 78 SPECIFY R IND PART SPECIFY R IN MERES IN THE FOLLOW		
625 FORMATC (15) DO YOU MANT A PLOT OF THE FITTED CURVE (DEFAULT-ND): 1176 637 FORMATC - ENTER ONE OF FOLLOWING 4 DIDTS ACCOMDING TO THE: // 1180 648 FORMATC - ENTER ONE OF FOLLOWING 4 DIDTS ACCOMDING TO THE: // 1180 649 FORMATC - ENTER ONE OF FOLLOWING 4 DIDTS ACCOMDING TO THE: // 1180 649 FORMATC - ENTER THE OF FACT (15) ENDER EACH DESERVATION: // 1180 74 FORMATC - ENTER THE UTREDUEST // 1180 640 FORMATC - ENTER THE VALUE FOR R IN THE PREFERRED MEIGHTING SOLMER. / 1180 640 FORMATC - ENTER THE VALUE FOR R IN THE PREFERRED MEIGHTING SOLMER. / 1180 640 FORMATC - ENTER THE VALUE FOR R IN THE PREFERRED MEIGHTING SOLMER. / 1180 640 FORMATC / ENTER THE VALUE FOR R ALIN ENDERFERRED MEIGHTING SOLMER. / 1180 640 FORMATC / ENTER THE VALUE SOL THE OBSERVATION // 1190 641 FORMATC / ENTER THE INTER ALINE FOR R ALIN ENDERFERRED MEIGHTING SOLMER. / 1190 641 FORMATC / ENTER THE INTER ALINE FOR REACH OF YOUR '15, OBSERVATIONS. / 1190 641 FORMATC / ENTER NO. INF. 11, 12, NUMBERS IN THE FOLLOWING ONDER / 1190 641 FORMATC / ENTER NO. INF. 11, 13, OBSERVATION MY/OF THIS RESONAGE / 1190 641 FORMATC / ENTER NO. INF. 11, 13, OBSERVATION MY/OF THIS RESONAGE / 1190 641 FORMATC / ENTER NO. INF. 11, 10 FORMATION / 1190 641 FORMATC / ENTER NO. INF. 11, 10 FORMATION / 1190 641 FORMATC / ENTER NO. INF. 11, 10 FORMATION / 1190 641 FORMATC / ENTER NO. INF. 11, 10 FORMATION / 1190 641 FORMATC / 100 FORMAT / 100 FORMATION NO. 13, 17/ / 100 641 FORMATC / 100 FORMATOR / 100 FORMATION NO. 13, 17/ / 100 641 FORMATC / 100 FORMATOR / 100 FORMATION NO. 13, 17/ / 100 642 FORMATC / 100 FORMATOR / 100 FORMATION NO. 13, 17/ / 100 644 FORMATC / 100 FORMATOR / 100 FORMATION NO. 13, 17/ / 100 645 FORMATC / 100 FORMATOR / 100 FORMATION NO. 13, 17/ / 100 646 FORMATC / 100 FORMATOR / 100 FORMATION NO. 13, 17/ / 100 647 FORMATC / 100 FORMATOR / 100 FORMATION NO. 13, 17/ / 100 648 FORMATC / 100 FORMATOR / 100 FORMATION NO. 13, 17/ / 100 649 FORMATC / 100 FORMATOR / 100 FORMATOR / 100 FORMATOR / 100 640 FORMATC / 100 FORMATOR / 100 FO		
 *//		
 635 FORMATIC (16) DO YOU MART TO MEGAT YOUR DATA (DEFAULT-MO) ///) 11810 636 FORMATIC ENTER OUE FOLLOWING COURDING TO HEY (1820) 640 FORMATIC ENTER THE VERT HEY TO REGREST? // VARIABLE. YOU DAT: 11800 77 THE WEIGHT E VERT ON YOUR FOREST? // VARIABLE. YOU DAT: 11800 78 FORMATIC ENTER THE VALUE FOR R IN THE PREFERRED MEIGHTING SCHEME. 11800 79 FORMATIC ENTER THE VALUE FOR R IN THE PREFERRED MEIGHTING SCHEME. 11800 70 FORMATIC ENTER THE VALUE FOR R IN THE PREFERRED MEIGHTING SCHEME. 11900 640 FORMATIC HEY ENTER THE VALUE FOR R IN THE PREFERRED MEIGHTING SCHEME. 11900 640 FORMATIC HEY ENTER THE VALUE FOR R IN THE PREFERRED MEIGHTING SCHEME. 11900 640 FORMATIC HEY ENTER THE VALUE FOR R IN THE PREFERRED MEIGHTING SCHEME. 11900 640 FORMATIC HEY ENTER THE LEGAT YOUR DATA SCHEMET YOR AND THE TO THE TO SCHEMATION. 11900 640 FORMATIC HEY ENTER THE LEGAT SCHEMA TO THE FOLLOWING FORES. 7 11900 641 FORMATIC FORTER THE FIRST DOBERVATION HUMBER? (11) THE VALUE OF THE DEFENDER VARIABLE (RESPONSE) 7 11900 641 FORMATIC FORTER MAIN HEY I. 1, OBSERVATION HUMBER? (11) THE VALUE OF THE DEFENDER VARIABLE (RESPONSE) 7 11900 641 FORMATIC FORTER MAIN HEY I. 1, OBSERVATION HUMBER? (11) THE VALUE OF THE DEFENDER VARIABLE (RESPONSE) 7 11900 641 FORMATIC FORTER HEF FIRST CARONING TO THE FORLOWING CORES. 11900 641 FORMATIC FORTER HER HER FOR CARON HEY ANTANATICHS (11) THE VALUE OF THE DEFENDER VARIABLE (RESPONSE) 7 1100 642 FORMATIC (12) ENTER MON HEY I. 1, OBSERVATION (10, 1, 2, 2) / 1000 644 FORTER VAND ANTA STRICTLY ACCOMDING TO THE FORLOWING CORES. 7 1100 645 FORMATIC (12) ENTER MERETING TO FOR THE THE FOLLOWING GORES. 7 1100 645 FORMATIC (12) ENTER MERETING TO FOR CARON FOR A FINIS RESPONSE S / 1 1000 645 FORMATIC (12) ENTER MERETING TO FOR CARON FOR A FINIS RESPONSE / 1 1000 645 FORMATIC (12) ENTER ALL HEY MEDICALLE (12) OBTAINED (12		
635 FORMATL - ENTER ONE OF FOLLOWING 4 DIGITS ACCOMDING TO THE - 1120 • ECTIVE TO UNERFORT / C2 DEPENDENT VARIABLE. YOU CAL: • STUTH THE WEIGHT TO BE SPECIFIED FOR EACH OBSERVATION - 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11800 • C 2 MEIGHT = U.GC RW) / 11900 • C 2 MEIGHT = U.GC RW) / 11900 • C 2 MEIGHT = U.GC RW) / 11900 • C 2 MEIGHT = U.GC RW) / 11900 • C 2 MEIGHT = U.GC RW) / 11900 • C 2 MEIGHT = U.GC RW) / 11900 • C 2 MEIGHT = C FW & DEPENDENT VARIABLE (RESPONSE) / 11900 • 1000. (12 / MENGENDE / 12 / INDEPENDENT VARIABLE (S) / 11900 • 1000. (12 / MENGENDE / 12 / INDEPENDENT VARIABLE (S) / 11900 • 1000. (12 / MENGENDE / 12 / INDEPENDENT VARIABLE (S) / 11900 • 0 MEIGHT = U.GC RW / 12 / 1000 • 0 MEIGHT = U.GC RW / 1000 • 0 MEIGHT = U.GC RW / 1000 • 0 MEIGHT =		
 HELGHING YOU PREFER, 7, G2 DEPENDENT VARIABLE, YOU CAM., 11830 ST. THE HELGHT TO BE SECTIFED OR REACH OBSERVATION; 11830 MELGHT E GOG RAY, 7 MELGHT E GOG RAY, 7 MELGHT E COG RAY, 7 MELGHT E COG RAY, 7 MELGHT E COG RAY, 7 TA MELGHT E COM CHARGE T COG RAY, 7 TA MELGHT E COM CHARGE T COG RAY, 7 TA MELGHT E COM CHARGE T COM CHARGE	636 FORMAT("ENTER ONE OF FOLLOWING 4 DIGITS ACCORDING TO THE".	
 SPECIFY R IN MEXT IMPUT REQUESTS/// AD EACH OBSERVATION THE REFINIT TO BE SPECIFIED FOR EACH OBSERVATION THE SPECIFIED FOR ACCOUNT OF THE SPECIFIED FOR EACH OF SPECIFIED FOR ACCOUNT OF THE SPECIFIED FOR EACH OF SPECIFIED FOR ACCOUNT OF THE SPECIFIED TO ACCOUNT OF THE SPECIFIES TO ACCOUNT OF THE SP	• WEIGHTING YOU PREFER "/" (Y= DEPENDENT VARIABLE. YOU CAN",	
<pre>// 2 WEIGHT = Y+R (R FOR EXAMPLE = -2,1,1/2 ETC. 7/ 11800 //</pre>	*" SPECIFY R IN NEXT INPUT REQUEST)"//	
 3 MEIGHT = LOG (RY) 7 1180 637 FORM C. ENERG THE VALUE FOR R IN THE PREFERRED MEIGHTING SCHEME. 7000 C. /li>		
 *. 1 <li< td=""><td></td><td></td></li<>		
637 FORMATC - ENTER' THE VALUE FOR 'R IN THE PREFERED WEIGHTING SCHEME', 1180 640 FORMATC - 112, ABOVE //) 640 FORMATC - 112, ABOVE //) 7100, (1) THE OBSERVATION RUBBER, VARIABLE (RESPONSE) / 1190 7100, (1) THE OBSERVATION RUBBER, VARIABLE (RESPONSE) / 1190 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (RESPONSE) / 1190 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) THE VALUE(S) OF THE /IZ, INDEFENDENT VARIABLE (S) //) 7100, (1) AUTON AUTON AUTON AUTON AUTON AUTONS INCLUED IN , 7100, (1) AUTON AUTON AUTON AUTON AUTON AUTONS INCLEDED IN , 7100, (1) SEACCHAR ED //) 7100, (1) SEACCHAR ED //) 7100, (1) SEACCHAR ED //) 7100, (1) SEACCHAR E		
<pre>* M0.7[2], ABOVE //) 600 FORMATC* (47) ENTER A LINE FOR EACH OF YOUR 13.7 OBSERVATIONS.7 11000 * EACH LINE MLST CONTAIN 12. MURRERS IN THE FOLLOWING ORDER 11020 * TOX, (11) THE VALUE OF THE OPERMEENT YARIABLE (RESPONSE) / 11000 * TOX, (11) THE VALUE OF THE OPERMEENT YARIABLE (RESPONSE) / 11000 * TOX, (11) THE VALUE OF THE OPERMEENT YARIABLE (RESPONSE) / 11000 * TOX, (11) THE VALUE OF THE OPERMEENT YARIABLE (RESPONSE) / 11000 * TOX, (11) THE VALUE OF THE OPERMEENT YARIABLE (RESPONSE) / 11000 * TOX (12) THE VALUE OF THE OPERMEENT YARIABLE (RESPONSE) / 11000 * TOX (12) THE WALLES() OF THE '12, 'INDEPENDENT YARIABLE (S) //) * NO. 12/ REPREMEENT HE IST OBSERVATION FOR THIS RESPONSE * OPERMENT HE IST OBSERVATION FOR THIS RESPONSE * OPERMENT HE IST OBSERVATION FOR THIS RESPONSE * OPERMENT // YOU FOROT '16 CHET THE LAST OBSERVATION OF THE PREVIOUS * OPERMENT // YOU FOROT '16 CHET THE LAST OBSERVATION NO. 13, '17 '1000 * CHET YOUR DATA STRICTLY ACCORDING TO THE FOLLOWING REQUEST // 12000 * CHET YOUR DATA STRICTLY ACCORDING TO THE FOLLOWING REQUEST // 12000 * CHET YOUR DATA STRICTLY ACCORDING TO THE FOLLOWING REQUEST // 12000 * CHET YOUR DATA STRICTLY ACCORDING TO THE FOLLOWING ROBER '12000 * TOX (11) THE VALUE OF THE DEFENDENT VARIABLE (RESPONSE) / 12000 * CHECK ON ALL INPUTS 'NA WE BEEN CHECKED. DO YOU MANT A SUMMARY', 12100 * OF THEM'/) 662 FORMATC ' WOU WANT A PLOT OF YOUR DATA POINTS INCLUDED IN', 1210 * OF THEM'/) 663 FORMATC 'WOU WANT A PLOT OF YOUR DATA POINTS INCLUDED IN', 1210 * OF THEM'/) 664 FORMATC 'WOIN ON HAAT A PLOT OF YOUR DATA A SUMMARY', 1210 * OF THEM'/) 670 FORMATC 'WOIN WANT A PLOT OF YOUR DATA POINTS INCLUDED IN', 1210 * OF THEM'/) 670 FORMATC 'WOIN WANT A PLOT OF YOUR DATA POINTS INCLUDED IN', 1210 * OF THE SUMMARY // WOI TON THE RESULTS //) 670 FORMATC // TOY OW WANT A PLOT OF YOUR DATA A SUMMARY', 1210 * OF THE SUMMARY // WOI TON THE RESULTS //) 670 FORMATC // TOY WANT A PLOT OF YOUR DATA A SUMMARY', 1210 * OF THE SUMMARY // YOU WANT A SI ON A L SE AR CH ', 1210 * OF THE SUMMARY // Y</pre>		
<pre>660 FORMATC**(12) ENTER A LINE FOR EACH OF YOUR 113, 'OBSERVATIONS.'/ 1101 FACULINE MUST CONTAIN 'LINE TOUR MURDER' TOOK, '(1) THE VORDE TOUR MURDER' TOOK, '(1) THE VORDE TOUR MURDER' TOOK, '(1) THE VORDE TOUR MURDER' TOOK, '(1) THE VALUE(3) OF THE '12,' INDEPENDENT VARIABLE (RESPONSE)'/ 1100 TOOK, '(1) THE VALUE(3) OF THE '12,' INDEPENDENT VARIABLE (S)'//) 1160 TOOK, '(1) THE VALUE(3) OF THE '12,' INDEPENDENT VARIABLE (S)'//) 1160 TOOK, '(1) THE VALUE(3) OF THE '12,' INDEPENDENT VARIABLE (S)'//) 1160 TOOK, '(1) THE VALUE(3) OF THE '12,' INDEPENDENT VARIABLE (S)'//) 1160 TOOK THE '12,' INDEPENDENT VARIABLE (S)'//) 1160 TOOK THE '12,' INDEPENDENT VARIABLE (S)'/) 1160 TOOK THE VALUE(3) OF THE '12,' INDEPENDENT VARIABLE (S)'/) 1200 TOOK THE '12,' INDEPENDENT VARIABLE (S)'/) 1200 TOOK THE '12,' INDEPENDENT VARIABLE (RESPONSE)'/ 1200 TOOK, '(1) THE VORDER'ION NUMBER'/ VARIABLE (RESPONSE)'/ 1200 TOOK ON ALL INPUTS'//) 100 YOUW DATA POINTS INCLUDED IN', 1210 THE SUMMARY //) 100 YOU MAAT A POINTS INCLUDED IN', 1210 THE SUMMARY //) 100 YOU MAAT A POINTS INCLUDED IN', 1210 THE SUMMARY //) 100 YOU MAAT A POINTS INCLUDED IN', 1210 THE SUMMARY //) 100 YOU MAAT A '11'-'DIMENSIONAL SEARCH //) 1210 THE SUMMARY //) 100 YOU MAAT A '11'-'DIMENSIONAL SEARCH //) 1210 TOOK IN SEARCH = 'EOL OF OD YOU SOUALDEN IN THE'. (2200</pre>	"NO IZ. AROVE //)	
<pre>* Fach Like Rust Containt, 12, Numbers In The FollOwing ORDER 11920 */10X, C1D THE VALUE OF THE DEPEndent VARIABLE (RESPONSE) / 11940 *10X, C1D THE VALUE OF THE DEPEndent VARIABLE (RESPONSE) / 11940 *10X, C1D THE VALUE OF THE DEPEndent VARIABLE (RESPONSE) / 11940 *10X, C1D THE VALUE OF THE DEPEndent VARIABLE (RESPONSE) / 11940 *10X, C1D THE VALUE (DITTO FILE DEPENDENT VARIABLE (ST//) 11940 *10X, C1D THE VALUE (DITTO FILE DEPENDENT) RESPONSE SYSTEM . 11970 * No. 12/ REFERRED 13/ IN CONTINUATION FOR RESPONSE SYSTEM . 11970 * SYSTEM FUST BE NUMBERED . 13/ IN CONTINUATION OF THE PREVIOUS . 11990 * OBSERVATION HUMBER. // . 100 ENTER THE LAST OBSERVATION (NO13, 25/ / 12010 * SYSTEM FUST BE NUMBERED . 13/ IN CONTINUATION OF THE PREVIOUS . 11990 * OBSERVATION HUMBER. // . 100 ENTER THE LAST OBSERVATION (NO13, 25/ / 12010 * FACH LINE HUST CONTAIN . 12/ NUMBERS IN THE FOLLOWING ORDER . 12000 * CONSERVATION HUMBER. / . 100 ENTER THE LAST OBSERVATION (NO13, 25/ / 12010 * FACH LINE HUST CONTAIN . 12/ NUMBERS IN THE FOLLOWING ORDER . 12000 * (10X, C1D) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE) / 12000 * (10X, C1D) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE) / 12000 * (10X, C1D) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE) / 12000 * (10X, C1D) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE) / 12000 * (10X, C1D) THE VALUE ON OF THE J2. INDEPENDENT VARIABLE (S7) /) 12070 * (10X, C1D) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE) / 12000 * (10X, C1D) THE NALE SEEN CHECKED. DO YOU MANT A SUMMAY , 1210 * (10X, C1D) THE NALE SEEN CHECKED. ON YOU MANT A SUMMAY , 1210 * (10X, C1D) THE NALE SEEN CHECKED. ON YOU MANT A SUMMAY , 1210 * (10X, C1D) THE RACK - FOR CORRECTION OR -OK- FOR ACTIVATION OF , 1210 * (10X, C1D) THE SEARCH - 11, -101ENSIONAL SEARCH //) (110 * (10X, C1D) THE SEARCH - 11, -101ENSIONAL SEARCH //) (110 * (10X, C1D) THE SEARCH - 11, -101ENSIONAL SEARCH - //) (110 * (10X, C1D) THE SEARCH - 200, AT /) (110 * (10X, C1D) THE SEARCH - 200, AT /) (110X, C1D) SEARCH - 2000 * (20X, COMAT(/) THE S</pre>	640 FORMAT(*(17) ENTER A LINE FOR EACH OF YOUR	
<pre>*//10X,- (11) THE OBSERVATION NUMBER?/ 11930 *10X,- (11) THE WAILE OF THE OFFENDENT VARIABLE (RESPONSE) / 11930 *10X,- (11) THE WEIGHT OF THE OFFENDENT VARIABLE (RESPONSE) / 11950 *10X,- (11) THE WEIGHT OF THE OFFENDER RESPONSE SYSTEM *00,- (12) THE WEIGHT OF THE OFFENDER RESPONSE SYSTEM *00,- (12) THE WEIGHT OF THE OFFENDER RESPONSE SYSTEM *00,- (12) THE NUMBERCO, (13/) IN CONTINUATION OF THIS RESPONSE SYSTEM *00,- (12) THE NUMBERCO, (13/) IN CONTINUATION OF THE PREVIOUS SYSTEM *00,- (12) THE NUMBERCO, (13/) IN CONTINUATION OF THE PREVIOUS *000 *00,- (11) THE NUMBERCO, (13/) IN CONTINUATION OF THE PREVIOUS *000 *00,- (11) THE NUMBERCO, (13/) IN CONTINUATION OF THE PREVIOUS *000 *100X,- (11) THE NUMBERCO, (13/) IN CONTINUATION OF THE PREVIOUS *100X,- (11) THE VALUE OF THE OF EXAL OF YOUR *100X,- (11) THE VALUE OF THE OF EXAL OF YOUR *100X,- (11) THE VALUE OF THE OF EXAL OF YOUR *100X,- (11) THE VALUE OF THE OF EXAL OF YOUR *100X,- (11) THE VALUE OF THE OF EXAL OF YOUR *100X,- (11) THE VALUE OF THE OFFENDENT VARIABLE (RESPONSE) / *100X,- (11) THE VALUE (S) OF THE *100X,- (11) THE VALUE (S) OF THE *100 *0 OF THEM //) *0 OF THEM //</pre>		
 10X, "(IV) THE VALUE(S) OF THE "12," INDEPENDENT VARIABLE(S)"//) 11970 10X, "(IV) THE VALUE(S) OF THE "12," OSEAVATION FOR THIS RESPONSES", 11980 NO. "(IZ) REMEMBER THE FIRST OBSERVATION FOR THIS RESPONSES", 11980 OSEERVATION NUMBER", "//) * OBSERVATION NUMBER", "//) * OBSERVATION NUMBER", "/) * OBSERVATION ALL DE OFENDENT VARIABLE (S) '/) * OBSERVATION ALL DE OFENDENT VARIABLE (RESPONSE)', "/) * OBSERVATION ALL DE OFENDENT VARIABLE (S) '/) * OBSERVATION ALL INFUTES '/) * OFTER CALC. FOR CORRECTION OR OR-FOR A FINAL", "2080 * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "/) * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "100 * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "1100 * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "1100 * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "1100 * OFTER 'DO YOU ANAT A A TA ', "OTTERPENDENT VARIABLE ', "1100 * OFTER'', "NO FOR THE RESULTS'//) * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "1210 * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "1210 * OFTER'', 'N', 'N', 'N', 'N', 'N', 'N', 'N',	AVIAN " (I) THE ODCEDIATION NUMBED"/	
 10X, "(IV) THE VALUE(S) OF THE "12," INDEPENDENT VARIABLE(S)"//) 11970 10X, "(IV) THE VALUE(S) OF THE "12," OSEAVATION FOR THIS RESPONSES", 11980 NO. "(IZ) REMEMBER THE FIRST OBSERVATION FOR THIS RESPONSES", 11980 OSEERVATION NUMBER", "//) * OBSERVATION NUMBER", "//) * OBSERVATION NUMBER", "/) * OBSERVATION ALL DE OFENDENT VARIABLE (S) '/) * OBSERVATION ALL DE OFENDENT VARIABLE (RESPONSE)', "/) * OBSERVATION ALL DE OFENDENT VARIABLE (S) '/) * OBSERVATION ALL INFUTES '/) * OFTER CALC. FOR CORRECTION OR OR-FOR A FINAL", "2080 * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "/) * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "100 * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "1100 * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "1100 * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "1100 * OFTER 'DO YOU ANAT A A TA ', "OTTERPENDENT VARIABLE ', "1100 * OFTER'', "NO FOR THE RESULTS'//) * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "1210 * OFTER 'DO YOU ANAT A PLOT OF YOUR DATA POINTS INCLUDED IN", "1210 * OFTER'', 'N', 'N', 'N', 'N', 'N', 'N', 'N',	*10X," (II) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE)"/	
641 FORMAT(*, ENTER NOW THE", 13, _ OBSERVATIONS FOR RESPONSE SYSTEM", 11970 * NO. 12/ REPRETENDENTED (13/* IN CONTINUATION FOR THE PREVIOUS", 11990 * SYSTEM MUST BE NUMBER.//, 10R THE LAST OBSERVATION NO FINE REPORTSONS // 12010 644 OBSERVATION NUMBER.//, 10R THE LAST OBSERVATION NO, 15, 7/* / 12010 645 FORMAT(* (17) ENTER AL LINE FOR EACH OBSERVATION (NO. 11, 7, 7/* / 12010 645 FORMAT(* (17) ENTER AL LINE FOR EACH OF SUBERIAL TO MONE, 12, 7/* / 12010 645 FORMAT(* (17) ENTER AL LINE FOR EACH OF YOUR, 13, 7/* 0556RVATIONS // 12010 646 FORMAT(* (17) ENTER AL LINE FOR EACH OF YOUR, 13, 7/* 0556RVATIONS // 12010 * CACH LINE MUST CONTAINT, 12, * NUMBERS IN THE FOLLOHING ORDER ' 12000 * (10X,* (11) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE)'/ 12000 * (10X,* (11) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE)'/ 12000 * (11) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE)'/ 12000 * (12,* (11) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE)'/ 12000 * (10X,* (11) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE)'/ 12000 * (10X,* (11) THE VALUE OF THE DEPENDENT VARIABLE (NESPONSE)'/ 12000 * (10X,* (11) THE VALUE OF THE DEPENDENT VARIABLE (NESPONSE)'/ 12000 * (10X,* (11) THE VALUE OF THE DEPENDENT VARIABLE (NESPONSE)'/ 12000 * (10X,* (11) THE VALUE OF THE DEPENDENT VARIABLE (NESPONSE)'/ 12000 * (10X,* (11) THE VALUE OF THE RESULTS'//) * (10) * (10X FOR THE RESULTS'//) * (10X FORMAT(* (10X FOR THE RESULTS'//) * (10X FORMAT(* (10X FOR THE RESULTS'//) * (10X FORMAT(* (10X FOR THE RESULTS'/)) * (2000 FORMAT(* (10X FOR THE RESULTS')) * (2000 FORM		
* SYSTEM MUST BE NUMBERED', 13/" IN CONTINUATION OF THE PREVIOUS", 11990 * DESERVATION NUMBER./', 2 642 FORMAT(-' YOU FORGOT TO ENTER THE LAST OBSERVATION (NO.", 13, ") / 12010 * REFUTER YOUR DATA STRICTLY ACCORDING TO THE FOLLOWING REQUEST // 12020 655 FORMAT(-'(17) ENTER A LINE FOR EACH OF YOUR, 13, OBSERVATIONS./' 12030 * TEACH LINE MUST CONTAIN 12, NUMBER // 12, NUMBER // 12000 * 100X, (11) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE) // 12000 * 100X, (11) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE) // 12000 * 100X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) ///) 12000 * 100X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) ///) 12000 * 00X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00Y, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00Y, (11) THE VALUE OF YOUR DATA POINTS INCLUDED IN , 12100 * THE SUMMARY //) * THE SUMMARY //) 1210 * THE COMPUTATIONS //) 100 YOU MANT A PLOT OF YOUR DATA POINTS INCLUDED IN , 12100 * THE COMPUTATIONS //) 100 YOU MANT A '. (11) - '. '. '. '. '. '. '. '. '. '. '. '. '.	ALL ENDMAT(/" ENTER NOW THE" 13 " OPERIVATIONS FOR DESCONCE EVETEN"	
* SYSTEM MUST BE NUMBERED', 13/" IN CONTINUATION OF THE PREVIOUS", 11990 * DESERVATION NUMBER./', 2 642 FORMAT(-' YOU FORGOT TO ENTER THE LAST OBSERVATION (NO.", 13, ") / 12010 * REFUTER YOUR DATA STRICTLY ACCORDING TO THE FOLLOWING REQUEST // 12020 655 FORMAT(-'(17) ENTER A LINE FOR EACH OF YOUR, 13, OBSERVATIONS./' 12030 * TEACH LINE MUST CONTAIN 12, NUMBER // 12, NUMBER // 12000 * 100X, (11) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE) // 12000 * 100X, (11) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE) // 12000 * 100X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) ///) 12000 * 100X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) ///) 12000 * 00X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00X, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00Y, (11) THE VALUE OF THE DEPENDENT VARIABLE (S) //) 12000 * 00Y, (11) THE VALUE OF YOUR DATA POINTS INCLUDED IN , 12100 * THE SUMMARY //) * THE SUMMARY //) 1210 * THE COMPUTATIONS //) 100 YOU MANT A PLOT OF YOUR DATA POINTS INCLUDED IN , 12100 * THE COMPUTATIONS //) 100 YOU MANT A '. (11) - '. '. '. '. '. '. '. '. '. '. '. '. '.	*" NO " 12/" REMEMBER THE FIRST ORSERVATIONS FOR RESPONSE STSTEM ,	
<pre>* DBSERVATION NUMBER.7//) for a constraint of a strict r the LAST OBSERVATION (NO,13, -)-/ * REARTER YOUR DATA STRICTLY ACCORDING TO THE FOLLOHING REQUEST/) for a constraint of a strict r the constraint of the strict r the strict</pre>	A SYSTEM MUST BE NUMBERED . 13/" IN CONTINUATION OF THE PREVIOUS"	
<pre>642 FORMAT(/' YOU FORGOT TO ENTER THE LAST OBSERVATION (NO_"13,")' 12010 ** REPITER YOUR DATA STRILTY ACCOMENG TO THE FOLLOWING REQUEST//) ** CACH LINE WIST CONTAIN 12," MUMBER' IN THE FOLLOWING GORDER ** CACH LINE WIST CONTAIN 12," MUMBER' IN THE FOLLOWING GORDER ** CACH LINE WALLE OF THE DEPENDENT VARIABLE (RESPONSE)'/ 12060 ** CACH LINE WALLE OF THE DEPENDENT VARIABLE (RESPONSE)'/ 12060 ** CACH LINE WALLE OF THE DEPENDENT VARIABLE (RESPONSE)'/ 12060 ** CACH LINE WALLE OF THE DEPENDENT VARIABLE (RESPONSE)'/ 12060 ** CACH LINE WALLE OF THE DEPENDENT VARIABLE (S)'/ 12060 ** CACH LINE WALLE OF THE DEPENDENT VARIABLE (S)'/ 12070 ** CACH LINE WALLE OF THE DEPENDENT VARIABLE (S)'/ 12070 ** CACH LINE WINT ANY BEEN CHECKED. DO YOU WANT A SUMMARY', 12080 ** CHECK ON ALL INFUTS'//) ** CACH LINE WINT ANY BEEN CHECKED. DO YOU WANT A SUMMARY', 12100 ** CHECK ON ALL INFUTS'//) ** CACH LINE SUMMARY'//) ** CACH CONTAIL OF YOUR DATA POINTS INCLUDED IN', 12100 ** CHECK ON ALL INFUTS'//) ** CHECK TOON OF A FOR CORRECTION OR -OK- FOR ACTIVATION OF', 12100 ** CHECK MAIT NON FOR THE RESULTS'//) ** CACH CORFNATC' (ADI NANY A 'DI 'D' OU WANT A 'S' CACH CORFNATC'') 12170 ** CACH CORFNATC'' ADI NANY A 'D' 'D' O'O'O'O'O'O'O'O'O'O'O'O'O'O'O'O</pre>	*" OBSERVATION NUMBER. "//)	
<pre>** REENTER YOUR DATA STRICTLY ACCORDING TO THE FOLLOWING REQUEST //) 12020 655 FORMAT(**(17) ENTER A LINE FOR EACH OF YOUR."13, "OBSERVATIONS/ 12040 ** EACH LINE MUST CONTAIN",12," NUMBERS IN THE FOLLOWING ORDER ** CACH LINE MUST CONTAIN",12," NUMBERS IN THE FOLLOWING ORDER ** CACH LINE MUST CONTAIN",12," NUMBERS IN THE FOLLOWING ORDER ** CACH LINE MUST CONTAINT,12," NUMBERS IN THE FOLLOWING ORDER ** CACH CACH CACH CACH CACH CACH CACH CA</pre>		
<pre>* EACH LINE MUST CONTAIN., 12, "NUMBERS IN THE FOLLOWING ORDER" 12040 */10X, "(11) THE OBSERVATION NUMBERS IN THE FOLLOWING ORDER" 12050 *10X, "(111) THE VALUE OF THE DEPENDENT VARIABLE (S)"//) 12070 600 FORMAT(" (18) ENTER-EACK-FOR CORRECTION OR-OK-FOR A FINAL", 12080 * CHECK ON ALL INPUTS THAVE BEEN CHECKED. DO YOU WANT A SUMMARY", 12100 * OF THEM '//) ANT A PLOT OF YOUR DATA POINTS INCLUDED IN", 1210 63 FORMAT(" YOUR INPUTS HAVE BEEN CHECKED. DO YOU WANT A SUMMARY", 12100 * OF THEM '//) 643 FORMAT(" DO YOU WANT A PLOT OF YOUR DATA POINTS INCLUDED IN", 1210 * THE SUMMARY'/) 644 FORMAT(" ENTER -BACK-FOR CORRECTION OR -OK-FOR ACTIVATION OF", 12140 * THE COMPUTATIONS '//) 6570 FORMAT(" ENTER -BACK-FOR CORRECTION OR -OK-FOR ACTIVATION OF", 12160 * THE COMPUTATIONS '//) 6700 FORMAT('/ NAIT A PLOT OF YOUR DATA POINTS INCLUDED IN", 12150 6707 FORMAT('/ OF) DO YOU WANT A ,11,"-DIMENSIONAL SEARCH" //) 12170 6708 FORMAT('/ OF) DO YOU WANT A ,11,"-DIMENSIONAL SEARCH" //) 12170 6709 FORMAT('/ OF) DO YOU WANT A ,11,"-DIMENSIONAL SEARCH" //) 12170 6705 FORMAT('/ OF) DO YOU WANT A ,11,"-DIMENSIONAL SEARCH" //) 12170 6706 FORMAT('//20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12200 6707 FORMAT('/20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12200 6715 FORMAT('/20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12200 6720 FORMAT('/20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12200 6720 FORMAT('/20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12200 6720 FORMAT('/20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12200 6720 FORMAT('/20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12200 6721 FORMAT('/20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12200 6725 FORMAT('/A WHICH IS SMALLER THAN THE VALUE (C'EIO,4,") OBTAINED', 12230 6725 FORMAT('/A WHICH IS SMALLER THAN THE VALUE (C'EIO,4,") FOUNDS, 12230 6725 FORMAT('/ MHICH IS SMALLER THAN THE VALUE (C'EIO,4,") OBTAINED', 12230 6735 FORMAT('/ MHICH IS SMALLER THAN THE VALUE (C'EIO,4,") FOUNDS, 12330 6745 FORMAT('/ " HHICH IS SMALLER THAN THE VALUE (C'EIO,4,"</pre>	* REENTER YOUR DATA STRICTLY ACCORDING TO THE FOLLOWING REQUEST //	12020
<pre>*//10x," (1) THE USAUE OF THE DEPENDENT VARIABLE (RESPONSE)"/ 12050 *10x," (11) THE VALUE (S) OF THE DEPENDENT VARIABLE (S) //) 12070 660 FORMAT(" (18) ENTER-EACK-FOR CORRECTION OR-OK-FOR A FIMAL", 12080 * CHECK ON ALL INFUTS'//) 662 FORMAT(" OVU INPUTS HAVE BEEN CHECKED. DO YOU MANT A SUMMARY", 12100 663 FORMAT(" OVU MANT A PLOT OF YOUR DATA POINTS INCLUDED IN", 12110 663 FORMAT(" OVU MANT A PLOT OF YOUR DATA POINTS INCLUDED IN", 12130 664 FORMAT(" C) YOU MANT A PLOT OF YOUR DATA POINTS INCLUDED IN", 12130 664 FORMAT(" ENTER -BACK- FOR CORRECTION OR -OK- FOR ACTIVATION OF", 12140 * THE COMPUTATIONS' //) 12150 670 FORMAT(" (19) DO YOU MANT A ",11."-DIMENSIONAL SEARCH" //) 12170 6705 FORMAT(" (19), 12."-O I M E N S I O N A L S E A R C H ", 1220 * SEARCH (DEFAULT=",14.") '/) 6710 FORMAT('/,15," LATTICE POINTS HAVE BEEN EVALUATED (", 1220 * 12." POINTS PER PARAMETER) ' N A L S E A R C H ", 1220 * 20(1H+), 12."-O I M E N S I O N A L S E A R C H ", 1220 * 20(1H+), 12."-O I M E N S I O N A L S E A R C H ", 1220 * 700 FORMAT('/, 15," LATTICE POINTS HAVE BEEN EVALUATED (", 1220 * 700 FORMAT('/, 15," LATTICE POINTS HAVE BEEN EVALUATED (", 1220 * 700 FORMAT('/, 15," LATTICE POINTS HAVE BEEN EVALUATED (", 1220 * 700 FORMAT('/, 15," LATTICE POINTS HAVE BEEN EVALUATED (", 1220 * 700 FORMAT('/, 15," LATTICE POINTS HAVE BEEN EVALUATED (", 1220 * FOUND IN SEARCH = "E10.4." AT') * 201H= 10.NEGRATER * ESIDUAL SUM OF SOUARES VALUE", 1220 * FOUND IN SEARCH = "E10.4." AT') * 2020 * FOUND IN THE SHALLER THAN THE VALUE (", E10.4.") OBTAINED", 1220 * FOUND IN THE SHALLER THAN THE VALUE (', E10.4.") FOUND", 12310 * THE MINIMIZATION 'PROCESS' AT') * 700 FORMAT('" MHICH IS SMALLER THAN THE VALUE (', E10.4.") FOUND", 1230 * 704 FORMAT('" MHICH IS SMALLER THAN THE VALUE (', E10.4.") FOUND", 1230 * 704 FORMAT('' MHICH IS SMALLER THAN THE VALUE (', E10.4.") FOUND", 1230 * 704 FORMAT('' MHICH IS SMALLER THAN THE VALUE (', E10.4.") FOUND", 1230 * 704 FORMAT('' MHICH IS SMALLER THAN THE VALUE (', E10.4.") FOUND", 1230 * 704 FORMAT(</pre>	655 FORMAT("+(17) ENTER A LINE FOR EACH OF YOUR", 13, " OBSERVATIONS."/	
<pre>*10%," (11) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE)"/ 12060 *0 (11) THE VALUE (S) OF THE 12. 'INDEPENDENT VARIABLE(S)'/) 12070 660 FORMAT(" (18) ENTER-EACK-FOR CORRECTION OR-OK-FOR A FIMAL", 12080 *0 (ECK ON ALL IMPUTS HAVE BEEN CHECKED. DO YOU HANT A SUMMARY", 1210 *0 (0) THEM"/) 1210 *0 (1) THE SUMMARY //) 12 (1) THE SUMMARY //) 1210 *0 (1) THE SUMMARY //) 12 (1) THE SUMMARY //) 1210 *0 (1) THE SUMMARY //) 12 (1) THE SUMMARY //) 1210 *0 (1) THE SUMMARY //) 12 (1) THE SUMMARY //) 1210 *0 (1) THE SUMMARY //) 12 (1) THE SUMMARY //) 12 (1) *0 (1) THE SUMMARY //) 12 (1) THE SUMMARY //) 12 (2) *0 (1) THE SUMMARY //) 12 (2) (2) (2) (2) (2) (2) (2) (2) (2) (2</pre>		
<pre>*10X, (111) THE VALUE(S) OF THE 12, INDEPENDENT VARIABLE(S) //) 12070 660 FORMAT(* (111) FHEVARLE-FOR CORRECTION OR-OK-FOR A FINAL*, 12090 ** CHECK ON ALL IMPUTS //) CHECKED. DO YOU MANT A SUMMARY*, 12100 662 FORMAT(* OVU IMPUTS HAVE BEEN CHECKED. DO YOU MANT A SUMMARY*, 12100 663 FORMAT(* OVU IMPUTS HAVE BEEN CHECKED. DO YOU MANT A SUMMARY*, 12100 664 FORMAT(* CHECKED. TO YOU MANT A PLOT OF YOUR DATA POINTS INCLUDED IN*, 12120 664 FORMAT(* CHECKED. FOR CORRECTION OR -OK- FOR ACTIVATION OF*, 12160 670 FORMAT(* MAIT NOM FOR THE RESULTS*//) 12170 670 FORMAT(* MAIT NOM FOR THE RESULTS*//) 12170 670 FORMAT(* HAIT NOM FOR THE RESULTS*//) 12170 671 FORMAT(* HAIT NOM FOR THE RESULTS*//) 12170 671 FORMAT(* HAIT NOM FOR THE RESULTS*//) 12170 672 FORMAT(* HAIT NOM FOR THE RESULATIONS DO YOU ALLON IN THE*, 12180 ** SEARCH (DEFAULT=*, 14, *) / 12 NOM ALL S E A R C H *, 12200 ** ORMAT(// 15,* LATTICE POINTS HAVE BEEN EVALUATED (*, 12220 ** THE SMALLEST MEIGHTED RESIDUAL SUM OF SOUARES VALUE*/ 12220 ** THE SMALLEST MEIGHTED RESIDUAL SUM OF SOUARES VALUE*/ 12220 ** FORMAT(/* THE SMALLEST MEIGHTED RESIDUAL SUM OF SOUARES VALUE*/ 12220 ** ORMAT(/* THE SMALLEST MAILEST MAITE* VALUE (*, E10, 4,*) OBTAINED*, 12290 ** USING THE FOLLOWING*/* INITIAL PARAMETER SY ALUE*/ 1220 ** THE MINITIZATION '* PORCESS THE MINITIZATION IS THEREFORE*, 12300 673 FORMAT(/* MHICH IS SMALLER THAN THE VALUE (*, E10, 4,*) OBTAINED*, 12300 673 FORMAT(/* MHICH IS SMALLER THAN THE VALUE (*, E10, 4,*) FOUND M*, 12300 673 FORMAT(/* MHICH IS SMALLER THAN THE VALUE (*, E10, 4,*) FOUND M*, 12300 ** USING THE FOLLOWING*/* INITIAL PARAMETER SYIMATES*/) 12300 673 FORMAT(/* MHICH IS SMALLER THAN THE VALUE (*, E10, 4,*) FOUN</pre>	*//10X, (1) THE UBSERVATION NUMBER /	
660 FORMATC" (13) ENTER-EACK-FOR CORRECTION OR-OK-FOR A FINAL",12080** CHECK ON ALL IMPUTS HAVE BEEN CHECKED. DO YOU HANT A SUMMARY",12100** OF THEM'//)12110663 FORMATC" DO YOU HANT A PLOT OF YOUR DATA POINTS INCLUDED IN",1210** THE SUMMARY"//)12130664 FORMATC" ENTER-BACK-FOR CORRECTION OR -OK-FOR ACTIVATION OF",121306706 FORMATC" ALT NOM FOR THE RESULTS"//)121506707 FORMATC" HAIT NOM FOR THE RESULTS"//)121606708 FORMATC" HAIT NOM FOR THE RESULTS"//)121606709 FORMATC" HOM MANY FUNCTION EVALUATIONS DO YOU ALLOH IN THE",121806709 FORMATC/" (/2014+),IZ; "-D I ME NS I O N A L S E A R C H ",12200*200 TH#3'/'15," LATTICE POINTS HAVE BEEN EVALUATED (",12200*200 TH#3'/'15," LATTICE POINTS HAVE BEEN EVALUATED (",12200*200 TH#3'/'15," LATTICE POINTS HAVE BEEN EVALUATED (",12200*200 TH#3'/'16," AT'')121006720 FORMATC/" THE SMALLEST MESIDUAL SUM OF SOUARES VALUE",12200*200 TH#3'/'11," -E', E'1.4, 'S', 'A'')122306720 FORMATC/" THE SMALLEST RESIDUAL SUM OF SOUARES VALUE",12200*200 TH#3'/'210.4, 'AT'')122306731 FORMATC/" THE SMALLEST HEASIDUAL SUM OF SOUARES VALUE",12200*201 THE SUMLEST HEAST HEAST HAN THE VALUE (", E10.4, ') OBTAINED",123006731 FORMATC/" HHICH IS SMALLER THAN THE VALUE (', E10.4, ') OBTAINED',123006735 FORMATC/" HHICH IS SMALLER THAN THE VALUE (', E10.4, ') FOUND BY',123006731 FORMATC/" HHICH IS SMALLER THAN THE VALUE (', E10.4, '	+10X, (11) THE VALUE OF THE DEPENDENT VARIABLE (RESPONSE) /	
<pre>* CHECK ON ALL INPUTS '//) 662 FORMAT('VOUR INPUTS HAVE BEEN CHECKED. DO YOU MANT A SUMMARY', 1210 * OF THEM'//) 663 FORMAT('DO YOU MANT A PLOT OF YOUR DATA POINTS INCLUDED IN', 1210 664 FORMAT('ENTER -BACK - FOR CORRECTION OR -OK - FOR ACTIVATION OF', 1210 * THE SUMMARY'/) 1210 6706 FORMAT('AIT NON FOR THE RESULTS'/) 1210 6700 FORMAT(', 'IND FOR THE RESULTS'/) 1210 6701 FORMAT(', 'IND FOR THE RESULTS'/) 1210 6702 FORMAT(', 'IND FOR THE RESULTS'/) 1210 6710 FORMAT(', 'IND FOR THE STALLATIONS DO YOU ALLON IN THE', 'III' 1210 6711 FORMAT(', 'IS', 'I'', 'I''', 'I'', 'I''', 'I'', '</pre>		
<pre>662 FORMAT('' YOUR INPUTS HAVE BEEN CHECKED. D0 YOU MANT A SUMMARY'', 12100 (* OF THEM''/') 663 FORMAT(' D0 YOU MANT A PLOT OF YOUR DATA POINTS INCLUDED IN', 12100 (* THE SUMMARY'/) 664 FORMAT(' ENTER -BACK- FOR CORRECTION OR -OK- FOR ACTIVATION OF', 12160 * THE SUMMARY'/) 700 FORMAT(''ALIT NON FOR THE RESULTS'/) 710 FORMAT('/', (19) D0 YOU MANT A ', 11, '-DIMENSIONAL SEARCH' //) 710 FORMAT('/', (19) D0 YOU MANT A ', 11, '-DIMENSIONAL SEARCH' //) 710 FORMAT('/', (19) D0 YOU MANT A ', 11, '-DIMENSIONAL SEARCH' //) 710 FORMAT('/', (19) D0 YOU MANT A ', 11, '-DIMENSIONAL SEARCH' //) 710 FORMAT('/', (19) D0 YOU MANT A ', 11, '-DIMENSIONAL SEARCH' //) 710 FORMAT('/', (19) D0 YOU MANT A ', 11, '-DIMENSIONAL SEARCH' //) 710 FORMAT('/', (19) D0 YOU MANT A ', 11, '-DIMENSIONAL SEARCH' //) 710 FORMAT('/', (19) D0 YOU MANT A ', 11, '-DIMENSIONAL SEARCH' /') 710 FORMAT('/' (19, 'LATTICE POINTS HAVE BEEN EVALUATED (', 12200 *'SOLATH'', '' THE SMALLEST MEIGHTED RESIDUAL SUM OF SQUARES VALUE'', 12230 6720 FORMAT('/' THE SMALLEST MEIGHTED RESIDUAL SUM OF SQUARES VALUE'', 12250 *'FOUND IN SEARCH = ',E10.4, ''' / A041) 7276 730 FORMAT('/' THE SMALLEST THAN THE VALUE ('',E10.4, '') OBTAINED'', 12290 *'' USING THE FOLLOWING ''' INITIAL PARAMETER ESTIMATES' /) 730 FORMAT(', SMALLER THAN THE VALUE ('',E10.4, '') OBTAINED'', 12300 *'' BY THE MINIMIZATION ''' PROCESS.THE MINIMIZATION IS THEREFORE'', 12300 *''' BY THE MINIMIZATION ''''''''''''''''''''''''''''''''''''</pre>	* CHECK ON ALL INPUTS (/)	
<pre>** OF THEM"//) 663 FORMAT(" DO YQU MANT A PLOT OF YQUR DATA POINTS INCLUDED IN", 12130 664 FORMAT(" ENTER -BACK- FOR CORRECTION OR -OK- FOR ACTIVATION OF", 12140 ** THE COMPUTATIONS" //) 6706 FORMAT(" HAIT NOH FOR THE RESULTS"//) 6706 FORMAT(" (19) DO YOU MANT A ",11,"-DIMENSIONAL SEARCH" //) 12160 6706 FORMAT(" (19) DO YOU MANT A ",11,"-DIMENSIONAL SEARCH" //) 12170 6705 FORMAT(" (19) DO YOU MANT A ",11,"-DIMENSIONAL SEARCH" //) 6710 FORMAT(//20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12190 6710 FORMAT(//20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12200 *20(1H*) 6715 FORMAT(// 15," LATTICE POINTS HAVE BEEN EVALUATED (", 12220 *12," POINTS PER PARAMETER) 1220 *12," POINTS PER PARAMETER '=) 1220 *12," POINTS PER PARAMETER '= EIGHTED RESIDUAL SUM OF SQUARES VALUE", 12200 *12," FOUND IN SEARCH =",EIG.4," AT' /) 6720 FORMAT(// ' THE SMALLEST HEIGHTED RESIDUAL SUM OF SQUARES VALUE", 12200 ** FOUND IN SEARCH =",EIG.4," AT' /) 6731 FORMAT(/ ', WHICH IS SMALLER THAN THE VALUE (',EIG.4,'') OBTAINED', ** BY THE MINIMIZATION'' PROCESS.'HE MINIMIZATION J 2300 6740 FORMAT(// MHICH IS LARGER THAN THE VALUE (',EIG.4,'') OBTAINED', ** BY THE MINIMIZATION'' PROCESS.'HE MINIMIZATION BY THE PARAMETER SADVE') 2300 6740 FORMAT(// 'MHICH IS LARGER THAN THE VALUE (',EIG.4,'') FOUND BY', ** BY THE MINIMIZATION'' PROCESS.'HE MINIMIZATION BY THE PARAMETER'') 6740 FORMAT(// 'MHICH IS LARGER THAN THE VALUE (',EIG.4,'') OBTAINED', ** BY THE MINIMIZATION FOR THAN THE VALUE (',EIG.4,'') FOUND BY', ** BY THE MINIMIZATION FOR THAN THE VALUE (',EIG.4,'') FOUND BY', ** DISING THE FOLLOWING''' INITIAL PARAMETER ESTIMATES'/) 6740 FORMAT(/'' MHICH IS LARGER THAN THE VALUE (',EIG.4,'') FOUND BY', ** BY THE MINIMIZATION FOR THAN THE VALUE (',EIG.4,'') FOUND BY', ** DISING THE FOLLOWING''' INITIAL PARAMETER SETIMATES'/) ** DISING THE FOLLOWING''' INITIAL PARAMETER SETIMATES'/) ** DISING THE FOLLOWING''' INITIAL PARAMETER SETIMATES'/) ** THE MINIMIZATION FOR THAN THE VALUE (',EIG.4,'') FOUND BY', ** DISING THE FOLLOWING'</pre>	662 FORMAT(YOUR INPUTS HAVE BEEN CHECKED, DO YOU HANT A SUMMARY".	
<pre>** THE SUMMARY"//) 664 FORMAT("ENTER - BACK- FOR CORRECTION OR -OK- FOR ACTIVATION OF", 71150 ** THE COMPUTATIONS" //) 6706 FORMAT('', 19) DO YOU HART A '', 11, "-DIMENSIONAL SEARCH" //) 7116 7070 FORMAT('', (19) DO YOU HAAT A '', 11, "-DIMENSIONAL SEARCH" //) 710 FORMAT('', (19) DO YOU HAAT A '', 11, "-DIMENSIONAL SEARCH" //) 710 FORMAT('', 19) DO YOU HAAT A '', 11, "-DIMENSIONAL SEARCH" //) 710 FORMAT('', 19) DO YOU HAAT A '', 11, "-DIMENSIONAL SEARCH" //) 710 FORMAT('', 19) DO YOU HAAT A '', 11, "-DIMENSIONAL SEARCH" //) 710 FORMAT('', 19) DO YOU HAAT A '', 11, "-DIMENSIONAL SEARCH" //) 710 FORMAT('', 19) DO YOU HAAT A '', 11, "-DIMENSIONAL SEARCH" //) 710 FORMAT('//20(1H+), 12, "-D I M E N S I O N A L S E A R C H '', 12200 710 FORMAT('/, 15, " LATTICE POINTS HAVE BEEN EVALUATED ('', 12210 710 FORMAT('/ 15, " LATTICE POINTS HAVE BEEN EVALUATED ('', 12220 711 *'', " FORMAT('', THE SMALLEST REIGHTED RESIDUAL SUM OF SQUARES VALUE'', 12240 710 FORMAT('', THE SMALLEST REIGHTED RESIDUAL SUM OF SQUARES VALUE'', 12250 720 FORMAT(', '' THE SMALLEST HEIGHTED RESIDUAL SUM OF SQUARES VALUE'', 12260 720 FORMAT(', '' THE SMALLEST HEIGHTED RESIDUAL SUM OF SQUARES VALUE'', 12260 720 FORMAT(', '' THE SMALLEST HAAT THE VALUE ('', E10.4, '') OBTAINED'', 12270 720 FORMAT(', '' THE SMALLEST THAA THE VALUE ('', E10.4, '') OBTAINED'', 12280 721 FORMAT(', '' HHICH IS SMALLEST THAA THE VALUE ('', E10.4, '') OBTAINED'', 12290 ** USING THE FOLLOWING /'' INITIAL PARAMETER ESTIMATES'/) 735 FORMAT(''' HHICH IS LARGER THAAN THE VALUE ('', E10.4, '') FOUND B'', 12310 ** BY THE MINIMIZATION '' PROCESS.'THE MINIMIZATION IS THEREFORE'', 12320 ** THE MINIMIZATION ''' DROCESS'.'THE MINIMIZATION B''' INITIAL PARAMETER SABOVE'/) 7330 741 FORMAT(''''''''''''''''''''''''''''''''''''</pre>	*" OF THEM"//)	
664 FORMAT(" ENTER -BACK- FOR CORRECTION OR -OK- FOR ACTIVATION OF", 12150 12150 *"THE COMPUTATIONS"//) 12150 6700 FORMAT(" MAIT NOM FOR THE RESULTS"//) 12160 6700 FORMAT(" MAIT NOM FOR THE RESULTS"//) 12160 6700 FORMAT(" HOM TANY FUNCTION EVALUATIONS DO YOU ALLOW IN THE", 12180 12170 6705 FORMAT(" HOM MANY FUNCTION EVALUATIONS DO YOU ALLOW IN THE", 12180 12190 6710 FORMAT(//20(1H*),12,"-01 M E N S I O N A L S E A R C H ", 12200 12200 * "SEARCH (DEFAULT=",14,")" //) N A L S E A R C H ", 12200 12200 * "SCOTH#*) 15," LATTICE POINTS HAVE BEEN EVALUATED (", 12220 12220 * *15 FORMAT(// TIS PER PARAMETER ")") 12230 6720 FORMAT(// THE SMALLEST MEIGHTED RESIDUAL SUM OF SQUARES VALUE"/ 12240 12250 * *5 FORMAT(// THE SMALLEST MEIGHTED RESIDUAL SUM OF SQUARES VALUE", 12250 12250 6720 FORMAT(// THE SMALLEST MEIGHTED RESIDUAL SUM OF SQUARES VALUE", 12250 12270 6731 FORMAT(// THE SMALLEST MEIGHTED RESIDUAL SUM OF SQUARES VALUE", 12250 12270 6731 FORMAT(// THE SMALLEST THE VALUE (",EI0.4,") OBTAINED", 12230 12230 6731 FORMAT(// THE SMALLEST THAN THE VA		
<pre>* THE COMPUTATIONS" //) 12150 6705 FORMAT(" MAIT NOM FOR THE RESULTS"//) 12160 6705 FORMAT(" HOW MANY FUNCTION EVALUATIONS DO YOU ALLON IN THE", 12180 * SEARCH (DEFAULT=",14,") // 12190 6710 FORMAT(//20(1H+),12,"-D I M E N S I O N A L S E A R C H ", 12200 * 20(1H+)) 12190 6715 FORMAT(//15," LATIICE POINTS HAVE BEEN EVALUATED (", 12220 * 12," POINTS PER PARAMETER)) 12190 6720 FORMAT(// THE SMALLEST WEIGHTED RESIDUAL SUM OF SOUARES VALUE"/ 12240 * 12," POINTS PER PARAMETER) 12250 6720 FORMAT(// THE SMALLEST WEIGHTED RESIDUAL SUM OF SOUARES VALUE"/ 12250 6720 FORMAT(// THE SMALLEST WEIGHTED RESIDUAL SUM OF SOUARES VALUE", 12250 6720 FORMAT(// THE SMALLEST WEIGHTED RESIDUAL SUM OF SOUARES VALUE", 12260 * FOUND IN SEARCH = ",E10.4," AT') 12270 6731 FORMAT(//" WHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12290 * USING THE FOLLOWING'" INITIAL PARAMETER ESTIMATES') 12300 6731 FORMAT(/" WHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND , 12300 * USING THE FOLLOWING'" INITIAL PARAMETER ESTIMATES') 2300 6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND , 12300 6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12360 * " BY THE MINIMIZATION"/" PROCESS.THE MINIMIZATION IS THEREFORE", 12300 6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12360 * " USING THE FOLLOWING'" INITIAL PARAMETER ESTIMATES') 2330 6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12360 * " USING THE FOLLOWING'" INITIAL PARAMETER ESTIMATES') 2330 6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12360 * " USING THE FOLLOWING'" INITIAL PARAMETER ESTIMATES') 12330 6741 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12360 * " USING THE FOLLOWING'" INITIAL PARAMETER ESTIMATES') 12330 6741 FORMAT(/ THICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * " USING THE FOLLOWING'" INITIAL PARAMETER ESTIMATES') 12370</pre>		
670FORMAT(' MAIT NOW FOR THE RESULTS'//)121606700FORMAT(/'' (19) DO YOU WANT A ",11,"-DIMENSIONAL SEARCH" //)121706705FORMAT('' HOW MANY FUNCTION EVALUATIONS DO YOU ALLOW IN THE",12180* "SEARCH (DEFAULT=",14,")" //)1121906710FORMAT('/20(1H+),12,"-D I M E N S I O N A L S E A R C H ",12200* 20(1H+))15," LATTICE POINTS HAVE BEEN EVALUATED (",12200* 20(1H+))15," LATTICE POINTS HAVE BEEN EVALUATED (",12200* 20(1H+))10 N SEARCH =",E10.4," AT')122306720FORMAT(// " THE SMALLEST WEIGHTED RESIDUAL SUM OF SQUARES VALUE"/12260* FOUND IN SEARCH =",E10.4," AT')12260* FOUND IN THE SEARCH =",E10.4," AT')12260* * FOUND IN THE SMALLEST RÉSIDUAL SUM OF SQUARES VALUE",12260* * FOUND IN THE SMALLEST RÉSIDUAL SUM OF SQUARES VALUE",12260* * FOUND IN THE SARCH =",E10.4," AT')12260* * FOUND IN THE SEARCH =",E10.4," AT')122706731FORMAT(// " HHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED",123006735FORMAT(/ " WHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND",12310* BY THE MINIMIZATION /" PROCESS.THE MINIMIZATION IS THEREFORE",12320* * BY THE MINIMIZATION WITH REGION"/" USING THE PARAMETER ESTIMATES'/)123006740FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY',123506741FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED",123506741FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED",12350	664 FURMATC ENTER -BALK- FOR LORRELTION OR -OK- FOR ACTIVATION OF",	
6700 FORMAT('/" (19) D0 YOU MANT A ",11,"-DIMENSIONAL SEARCH" //) 12170 6705 FORMAT(' HOH MANY FUNCTION EVALUATIONS D0 YOU ALLOW IN THE", 12180 * "SEARCH (DEFAULT=",14,")" //) 12190 6710 FORMAT('/20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12200 * 20(1H*) 12," POINTS PÉR PARAMETER ") 12210 6715 FORMAT(// IS," LATTICE POINTS HAVE BEEN EVALUATED (", 12220 * 12," POINTS PÉR PARAMETER ")" 12230 6720 FORMAT(// " THE SMALLEST WEICHTED RESIDUAL SUM OF SQUARES VALUE"/ 12250 6720 FORMAT(// " THE SMALLEST RÉSIDUAL SUM OF SQUARES VALUE"/ 12250 6720 FORMAT(// " THE SMALLEST RÉSIDUAL SUM OF SQUARES VALUE", 12250 6720 FORMAT(// " THE SMALLEST RÉSIDUAL SUM OF SQUARES VALUE", 12260 * FOUND IN SEARCH =",E10.4, "AT" /) 12270 6731 FORMAT(/" WHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12200 * " SUING THE FOLLOWING'" INITIAL PARAMETER ESTIMATES' /) 12300 6731 FORMAT(/" WHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND ", 12300 6745 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND ", 12300 6740 FORMAT(/" WHICH I		
6705 FORMAT(' HOW MANY FUNCTION EVALUATIONS DO YOU ALLOW IN THE", 12180 * SEARCH (OEFAULT=",I4,")" //) 12190 6710 FORMAT(//20(1H*),I2,"-D I M E N S I O N A L S E A R C H ", 12200 *20(1H*) 1210 12210 6715 FORMAT(// I5," LATTICE POINTS HAVE BEEN EVALUATED (", 12210 6720 FORMAT(// THE SMALLEST WEICHTED RESIDUAL SUM OF SQUARES VALUE'/ 12230 6720 FORMAT(// " THE SMALLEST WEICHTED RESIDUAL SUM OF SQUARES VALUE'/ 12250 6725 FORMAT(// " THE SMALLEST RESIDUAL SUM OF SQUARES VALUE", 12260 * FOUND IN SEARCH = ", E10.4," ATT /) 12250 6730 FORMAT(SX," PAR.NO. ", I1," = ", E11.4, SX, 40A1) 12280 6731 FORMAT('M HICH IS SMALLER THAN THE VALUE (", E10.4,") OBTAINED", 12280 6735 FORMAT(/ " HHICH IS SMALLER THAN THE VALUE (", E10.4,") FOUND", 12300 * " BY THE MINIMIZATION" " PROCESS. THE MINIMIZATION IS THEREFORE", 12320 * " BY THE MINIMIZATION THE VALUE (", E10.4,") FOUND", 12330 6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (", E10.4,") FOUND BY", 12350 6741 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (", E10.4,") FOUND BY", 12350 6741 <td></td> <td></td>		
<pre>* "SEARCH (DEFAULT=",14,")" //) 6710 FORMAT(//201H+),12,"-D I M E N S I O N A L S E A R C H ", 12200 *20(1H+)) 6715 FORMAT(// I5," LATTICE POINTS HAVE BEEN EVALUATED (", 12210 6715 FORMAT(// T THE SMALLEST NEIGHTED RESIDUAL SUM OF SQUARES VALUE"/ 12230 6720 FORMAT(// " THE SMALLEST NEIGHTED RESIDUAL SUM OF SQUARES VALUE"/ 12240 * FOUND IN SEARCH =",E10.4, "AT' /) 6725 FORMAT(// " THE SMALLEST RÉSIDUAL SUM OF SQUARES VALUE", 12260 * FOUND IN THE SEARCH ==,E10.4, "AT' /) 6730 FORMAT(S," PAR.NO. ",11," ==,E11.4, SX.40A1) 6731 FORMAT(// " HHI S SMALLEST RÉSIDUAL SUM OF SQUARES VALUE", 12260 * FOUND IN THE SEARCH ==,E10.4, "AT' /) 6731 FORMAT(// " HHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12200 6735 FORMAT(/ " HHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND", 12310 * BY THE HINIMIZATION"/" PROCESS.THE MINIMIZATION IS THEREFORE", 12320 * THE MINIMIZATION /" PROCESS.THE MINIMIZATION IS THEREFORE", 12330 6740 FORMAT(/ " HHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12340 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES"/) 6741 FORMAT(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES"/) 6741 FORMAT(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES"/) 6741 FORMAT(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES"/) 6743 FORMAT(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 6741 FORMAT(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES"/) 6743 FORMAT(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 6741 FORMAT(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES"/) 12370 </pre>	6705 FORMATC" HOW MANY FUNCTION EVALUATIONS DO YOU ALLOW IN THE	
6710 FORMAT(//20(1H*),12,"-D I M E N S I O N A L S E A R C H ", 12200 *20(1H*)) 12210 6715 FORMAT(// 15," LATTICE POINTS HAVE BEEN EVALUATED (", 12220 *12," POINTS PER PARAMETER) 12230 6720 FORMAT(// "THE SMALLEST HEIGHTED RESIDUAL SUM OF SQUARES VALUE"/ 12250 6725 FORMAT(// "THE SMALLEST RESIDUAL SUM OF SQUARES VALUE", 12260 * "FOUND IN THE SEARCH =",E10.4," AT'/) 12270 6730 FORMAT(// "HIE SMALLEST RESIDUAL SUM OF SQUARES VALUE", 12260 * "JOIND THE SEARCH = ",E10.4," AT'/) 12270 6730 FORMAT(// "HIE SMALLEST RESIDUAL SUM OF SQUARES VALUE", 12280 6731 FORMAT(// "HICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12290 * USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES'/) 12300 6735 FORMAT(/ " WHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND", 12310 * RESTARTED IN NEW MINIMUM REGION'/ USING THE PARAMETERS ABOVE"/) 12330 6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12340 * THE MINIMIZATION PROCESS.THE MINIMIZATION BY", 12350 6741 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12350 6741 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 6741 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12350 6741 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") TOUND BY", 12350 6741 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") TOUND BY", 12350 6741 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 6741 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 6741 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 * USING THE FOLLOWING'" INITIAL PARAMETER ESTIMATES"/) 12350 6741 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * USING THE FOLLOWING'" INITIAL PARAMETER ESTIMATES"/) 12370 6745 FORMAT(/ TO HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * USING THE FOLLOWING'" INITIAL PARAMETER ESTIMATES"/) 12370	• " SEARCH (DEFAULT=", 14, ")" //)	
<pre>*20(1H*)) format(// I5," LATTICE POINTS HAVE BEEN EVALUATED (", 12220 *12," POINTS PÉR PARAMETER)") format(// THE SMALLEST WEIGHTED RESIDUAL SUM OF SQUARES VALUE'/ 12230 format(// THE SMALLEST WEIGHTED RESIDUAL SUM OF SQUARES VALUE'/ 12250 format(// THE SMALLEST RÉSIDUAL SUM OF SQUARES VALUE", 12260 *" FOUND IN THE SMALLEST RÉSIDUAL SUM OF SQUARES VALUE", 12260 format(Sx," PAR. NO. ",11," = ,e11.4, 5X, 40A1) 12280 format(// "HHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12290 *" BU SING THE FOLLONING/" INITIAL PARAMETER ESTIMATES') 12300 format(/ " HHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND", 12310 *" BY THE HINIMIZATION"/" PROCESS.THE HINIMIZATION IS THEREFORE", 12320 *" BY THE HINIMIZATION"/" PROCESS.THE HINIMIZATION IS THEREFORE", 12320 *" HE MINIMIZATION PROCESS.THE MINIMIZATION BY LETERS ABOVE"/) 12330 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12350 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") THEREFORE", 12320 *" BY THE HINIMIZATION PROCESS.THE HINIMIZATION BY", 12350 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12350 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") THE MINIMIZATION BY LE330 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") THE MINIMIZATION BY LE330 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") THE MINIMIZATION BY LE330 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") THE MINIMIZATION BY LE330 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") THE MINIMIZATION BY LE330 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") THE MINIMIZATION BY LE330 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") THE MINIMIZATION BY LE330 format(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") THE MINIMIZATION BY LE330 format(/" HICH IS LARGER THAN THE VALUE (",E10.4,") THE MINIMIZATION BY LE330 format(/" HICH IS LARGER THAN THE VALUE (",E10.4,") THE MINIMIZATION BY LE330 fo</pre>	6710 FORMAT(//20(1H*), I2, "-D I MENSIONAL SEARCH",	
<pre>*12, "POINTS PER PARAMETER)") 6720 FORMAT(// THE SMALLEST MEIGHTED RESIDUAL SUM OF SQUARES VALUE'/ 12240 * "FOUND IN SEARCH =",E10.4," AT') 6725 FORMAT(// THE SMALLEST RESIDUAL SUM OF SQUARES VALUE', 12260 6730 FORMAT(5X," PAR.MO. ",I1," =",E11.4,5X,40A1) 6731 FORMAT(/ WHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12290 * USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES') 6735 FORMAT(/ " WHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND", 12310 * "BY THE MINIMIZATION"/" PROCESS.THE MINIMIZATION IS THEREFORE", 12320 * "RESTARTED IN NEW MINIMUM REGION'/" USING THE PARAMETERS ABOVE"/) 12350 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12350 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES") 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES") 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES") 6743 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 6741 FORMAT(//" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES") 6745 FORMAT(// THICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 6741 FORMAT(//" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES") 6745 FORMAT(//" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES") 6745 FORMAT(// THICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 ** USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES" /) 12370 6745 FORMAT(// 77((1++))) </pre>	*20(1H*))	
6720 FORMAT(// "THE SMALLEST MEIGHTED RESIDUAL SUM OF SQUARES VALUE"/ 12260 * "FOUND IN SEARCH = ",E10.4," AT'/) 12250 6725 FORMAT(// "THE SMALLEST RESIDUAL SUM OF SQUARES VALUE", 12260 * "FOUND IN THE SEARCH = ",E10.4," AT'/) 12270 6730 FORMAT (S," PAR. NO. ",11," = ",E11.4, SX.40A1) 12280 6731 FORMAT (S," PAR. NO. ",11," = ",E11.4, SX.40A1) 12280 * "USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES"/) 12300 6735 FORMAT (/" WHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12300 * "USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES"/) 12300 * "BY THE MINIMIZATION"/" PROCESS.THE MINIMIZATION IS THEREFORE", 12320 * "BY THE MINIMIZATION"/" PROCESS.THE MINIMIZATION IS THEREFORE", 12320 6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12340 * THE MINIMIZATION PROCESS." AT") 12370 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * "USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES"/) 12370 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * "USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES"/) 12370	6/15 FURMAIC // 15, " LATTICE POINTS HAVE BEEN EVALUATED (",	
<pre>* FOUND IN SEARCH =",E10.4," AT'/) 12250 6725 FORMAT(//" THE SMALLEST RESIDUAL SUM OF SOUARES VALUE", 12260 * FOUND IN THE SEARCH =",E10.4," AT'/) 12270 6730 FORMAT(5X," PAR.NO. ",11," = ,E11.4,5X,40A1) 12280 6731 FORMAT('' WHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12290 * USING THE FOLLOWING"/" INITIAL PARAMETER ESTIMATES') 12300 6735 FORMAT(/" WHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12300 6735 FORMAT('' WHICH IS SMALLER THAN THE VALUE (',E10.4,") FOUND", 12310 * "BY THE MINIMIZATION"/" PROCESS.THE MINIMIZATION IS THEREFORE", 12320 * "RESTARTED IN NEW MINIMUM REGION"/" USING THE PARAMETERS ABOVE'/) 12330 6740 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12340 * "THE MINIMIZATION PROCESS."/ AT') 12350 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * " USING THE FOLLOWING"/" INITIAL PARAMETER ESTIMATES"/) 12370 6745 FORMAT(//" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * " USING THE FOLLOWING"/" INITIAL PARAMETER ESTIMATES"/) 12370 6745 FORMAT(//" COLUMING"/" INITIAL PARAMETER ESTIMATES"/) 12370 6746 FORMAT(//" TOTIH')) 12380</pre>	ATTO EDDWAT(// " THE CWALLEST HEICHTED DECIDIAL CUM OF COUNDED VALUE"	
6725 FORMAT(// "THE SMALLEST RÉSIDUAL SUM OF SQUARES VALUE", 12260 *" FOUND IN THE SEARCH =",E10.4, "AT" /) 12270 6730 FORMAT(5X, "PAR.NO. ",11," =",E11.4, 5X.40A1) 12280 6731 FORMAT(5X, "PAR.NO. ",11," =",E11.4, 5X.40A1) 12280 6731 FORMAT(15, "MALLER THAN THE VALUE (",E10.4,") OBTAINED", 12290 12300 6735 FORMAT(/ " HHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND", 12310 12300 6735 FORMAT(/ " HHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND", 12310 12300 * BY THE HINIMIZATION"" PROCESS.THE HINIMIZATION IS THEREFORE", 12320 12300 * BY THE HINIMIZATION "T USING THE PARAMETER FAS ABOVE"/) 12330 12300 6740 FORMAT(/ " HHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12340 12350 6741 FORMAT(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 12370 6741 FORMAT(/" HHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 12370 6741 FORMAT(/ 7(1HHI S LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 12370 6741 FORMAT(/ T HICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 12370 6745 FORMAT(/ 7(1HHI S LARGER THAN THE VALUE (",E10.4,") DBTAINED", 12360 12370	* FOUND IN SEARCH =" EIG 4 " AT" /)	
<pre>* FOUND IN THE SEARCH = ",E10.4, "AT" /) 12270 6730 FORMAT(5X, "PAR.NO. ",11," = ",E11.4,5X,40A1) 12280 6731 FORMAT(/" WHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12290 * USING THE FOLLOWING " INITIAL PARAMETER ESTIMATES' /) 12300 6735 FORMAT(/" WHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND", 12310 * BY THE MINIMIZATION " PROCESS.THE MINIMIZATION IS THEREFORE", 12320 * RESTARTED IN NEW MINIMUM REGION " USING THE PARAMETERS ABOVE /) 12330 6740 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY, 12340 * THE MINIMIZATION PROCESS.T AT") 12350 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * "USING THE FOLLOWING" " INITIAL PARAMETER ESTIMATES" /) 12370 6745 FORMAT(// 77(1H+)) 12380 </pre>		
6/31 FORMAT(/" WHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12290 * USING THE FOLLOWING/" INITIAL PARAMETER ESTIMATES") 12300 6735 FORMAT(/ " WHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND", 12310 * BY THE MINIMIZATION"" PROCESS.THE MINIMIZATION IS THEREFORE", 12320 * RESTATED IN NEW MINIMUR REGION"/" USING THE PARAMETER SABOVE"/) 12330 6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12340 * THE MINIMIZATION PROCESS'/ AT " 12350 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * USING THE FOLLOWING'/ INITIAL PARAMETER ESTIMATES"/) 12370 6745 FORMAT(// TY (THE)) 12370	* FOUND IN THE SEARCH = ",E10.4, " AT" /)	12270
6/31 FORMAT(/" WHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED", 12290 * USING THE FOLLOWING/" INITIAL PARAMETER ESTIMATES") 12300 6735 FORMAT(/ " WHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND", 12310 * BY THE MINIMIZATION"" PROCESS.THE MINIMIZATION IS THEREFORE", 12320 * RESTATED IN NEW MINIMUR REGION"/" USING THE PARAMETER SABOVE"/) 12330 6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12340 * THE MINIMIZATION PROCESS'/ AT " 12350 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * USING THE FOLLOWING'/ INITIAL PARAMETER ESTIMATES"/) 12370 6745 FORMAT(// TY (THE)) 12370	6730 FORMAT(5X, PAR.NO. ,11, = ,E11.4,5X,40A1)	
 USING THE FOLLOWING /" INITIAL PARAMETER ESTIMATES'/) 6735 FORMAT(/" HHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND,", 2310 " BY THE MINIMIZATION"/" PROCESS.THE MINIMIZATION IS THEREFORE", 2320 " RESTARTED IN NEW MINIMUM REGION"/" USING THE PARAMETERS ABOVE"/) 12330 6740 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12350 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12350 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 *" USING THE FOLLOWING"/" INITIAL PARAMETER ESTIMATES"/) 12370 6745 FORMAT(// 77(1H+)) 	6731 FORMAI(/ WHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED".	12290
 BY THE MINIMIZATION / PROCESS.THE MINIMIZATION IS THEREFORE", 12320 "RESTARTED IN NEW MINIMUM REGION"/ USING THE PARAMETERS ABOVE/) 12330 6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12340 "THE MINIMIZATION PROCESS"/ AT") 12350 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * USING THE FOLLDWING / INITIAL PARAMETER ESTIMATES" /) 12370 6745 FORMAT(/ 77(1H+)) 12380 	* USING THE FOLLOWING / "INITIAL PARAMETER ESTIMATES"/)	12300
 * RESTARTED IN NEW MINIMUM REGION"/" USING THE PARAMETERS ABOVE"/) 12330 6740 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY, 12340 * THE MINIMIZATION PROCESS'/ AT") 12350 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * USING THE FOLLOWING"" INITIAL PARAMETER ESTIMATES"/) 12370 6745 FORMAT(// 77(1H+)) 12380 	6735 FORMAT(/ WHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND",	
6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", 12340 * "THE MININIZATION PROCESS"/ AT" 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 * "USING THE FOLLDWING/" INITIAL PARAMETER ESTIMATES"/) 12370 6745 FORMAT(// 77(1H+)) 12380	BE THE DINITIZATION / PROLESS THE DINITIZATION IS THEREFORE,	
* THE MINIMIZATION PROCESS'/ AT) 12350 6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", 12360 *" USING THE FOLLOWING"/" INITIAL PARAMETER ÉSTIMATES"/) 12370 6745 FORMAT(// 77(1H+)) 12380	A740 EOPMAT(/ " UNICH IS LARCED THAN THE VALUE (" E10 (") EOUMO DY"	
6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",EI0.4,") OBTAINED", 12360 *" USING THE FOLLOWING"/" INITIAL PARAMETER ESTIMATES"/) 12370 6745 FORMAT(// 77(1H+)) 12380	* THE MINIMIZATION PROCESS / AT)	
* USING THE FOLLOWING"/" INITIAL PARAMETER ESTIMATES"/) 12370 6745 FORMAT(// 77(1H*)) 12380	6741 FORMAT(/ WHICH IS LARGER THAN THE VALUE (". FIG 4 ") ORTAINED"	
6745 FORMAT(// 77(1H+)) 12380		
6747 FORMAT(/ " DO YOU WANT TO MAKE ANY ADDITIONAL CORRECTIONS BEFORE", 12390	6745 FORMAT(// 77(1H+))	
	6747 FORMAT(/ " DO YOU WANT TO MAKE ANY ADDITIONAL CORRECTIONS BEFORE",	

F02

<pre>* THE RESTART //) 6748 FORMAT(// DO YOU WANT TO START THE MINIMIZATION PROCESS / * USING THE PARAMETER VALUES FOUND IN THE SEARCH ABOVE //) 675 FORMAT(// (20) DD YOU WANT TO REPEAT THE COMPUTATIONS ABOVE BUT '/ * WITH SOME CHANGE IN THE INPUTS '/) 680 FORMAT(" DO YOU WANT ANOTHER RUN WITH A NEW SET OF OBSERVATIONS // *) 685 FORMAT(" I HOPE YOU HAVE BEEN PLEASED WITH THE PROGRAM. '/ * " IF YOU HAVE ANY SUGGESTIONS FOR ALTERATIONS PLEASE CONTACT ME '// * " P.VENG PEDERSEN //) 690 FORMAT((21) DO YOU WANT TO USE THE FINAL COMPUTED PARAMETERS '/ * " P.VENG PEDERSEN //) 690 FORMAT((21) DO YOU WANT TO USE THE FINAL COMPUTED PARAMETERS '/ * " FROM THE RUN ABOVE AS STARTING PARAMETERS FOR THE NEXT RUN. '//) 692 FORMAT((22) DO YOU WANT TO MAKE MORE ALTERATIONS '//) 694 FORMAT((22) DO YOU WANT TO MAKE MORE ALTERATIONS '//)</pre>	12400 12410 12420 12430 12440 12450 12460 12460 12470 12480 12500 12510 12510 12520 12530

	20840	
IF (NROW.LE.O) GO TO 100 IFAULT=0	20850	
CALL CHOL (A NOOL C NULLTY TEAULT)	20860	
CALL CHOL(A, NROW, C, NULLTY, IFAULT) IF (IFAULT.NE.0) GO TO 100	20870 20880	
NN=(NROW*(NROW+1))/2	20880	
IROW=NROW	20900	
NDIAGENN	20910	
16 IF(C(NDIAG).EQ.0D0) GO TO 11	20920	
L=NDIAG	20930	
DO 10 I=IROW, NROW	20940	
W(I)=C(L)	20950	
	20960	
10 CONTINUE	20970	
I COL =NROW	20980	
JCOL=NN	20990	
MDIAG=NN	21000	
15 L=JCOL	21010	
X=0D0	21020	
IF(ICOL.EQ.IROW)X=1D0/W(IROW)	21030	
K=NROW	21040	
13 IF (K.EQ. IROW) GO TO 12	21050	
X=X-W(K)+C(L)	21060	
K=K-1	21070	

E03

	L=L-1	21080
	IF (L.GT.MDIAG) L=L-K+1	21090
	GO TO 13	21100
	2 C(L)=X/W(IROW)	21110
	IF (ICOL.EQ.IROW) GO TO 14	21120
	MDIAG=MDIAG-ICOL	21130
	ICOL=ICOL-1	21140
	JCOL=JCOL-1	21150
	GO TO 15	21160
	1 L=NDIAG	21170
	DO 17 J=IROW, NROW	21180
	C(L)=0D0	21190
	L=L+J 7 CONTINUE	21200
	4 NDIAG=NDIAG-IRON	21210 21220
	IROW=IROW-1	21230
	IF (IROW, NE. 0) GO TO 16	21240
10	O RETURN	21250
	END	21260
С		21270
2	SUBROUTINE LSQ(F,FUNC)	21280
	DIMENSION F(20), F1(20), F2(20), Y(100), WYN(100), XX(9, 100), A(9),	21290
	*LABEL(100)	21300
	DOUBLE PRECISION FUNC, YOBS, YCAL, W	21310
	COMMON /DATA/ XX.Y.WYN.NOBS	21320
	COMMON /PARLIM/ F1,F2	21330
	COMMON /CONSTR/LIMITS	21340
	COMMON /FUNNUM/ITHFUN	21350
	COMMON /B4/ NIND	21360
	COMMON /B6/ NOP	21370
	COMMON /B10/LABEL	21380
	COMMON /B14/NVIOL	21390
	S=0.	21400
	FUNC=0D0	21410
	IF(LIMITS.EQ.0) GO TO 6	21420
	DO 5 I=1,NOP	21430
	IF(F(I).GE.F1(I).AND.F(I).LE.F2(I).OR.F1(I).EQ.F2(I)) GO TO 5	21440
	B=ABS(F1(I)-F(I)) B2=ABS(F2(I)-F(I))	21450
		21460
	B=1.E50*(1.+AMIN1(B,B2)/AMAX1(B,B2)) IF(B.GT.S)S=B	21470
	5 CONTINUE	21480 21490
	IF(S.GE. 1E50) GO TO 20	21500
	6 CONTINUE	21510
	DO 15 I=1,NOBS	21520
	DO 10 J=1,NIND	21530
1	0 A(J)=XX(J,I)	21540
	ITHFUN=LABEL(I)	21550
	S=Y(1)	21560
	IF(NIND.EG.1) CALL MODEL(S,A(1),F,0)	21570
	IF(NIND.GT.1) CALL MODEL(S,A,F,O)	21580
	IF(S.GE.1E50) GO TO 20	21590
	YCAL=S	21600
	YOBS=Y(1)	21610
	W=WYN(I)	21620
1	5 FUNC=FUNC+W*(YOBS-YCAL)*(YOBS-YCAL)	21630
-	RETURN	21640
4	0 FUNC=S+1.000000001	21650
	NVIOL=NVIOL+1	21660
	RETURN	21670
~	END	21680
		21690

F03

	and the second	
SUBROUTINE READ(N, AR)	21700	
DIMENSION A(78), AR(12), AL(10)	21710	
COMMON/B5/NG,NS,IGO	21720	
COMMON /B15/SKIP	21730	
LOGICAL SKIP	21740	
DATA YES, ANO, BACK, AO, AK, BLANK, PLUS, AMINUS, COMMA, DOT, E	21750	
*/1HY,1HN,1HB,1H0,1HK,1H ,1H+,1H-,1H,,1H.,1HE/	21760	
DATA AL/1H0, 1H1, 1H2, 1H3, 1H4, 1H5, 1H6, 1H7, 1H8, 1H9 /	21770	
5 IG0=1	21780	
SKIP=.FALSE.	21790	
DO 6 L=1.78	21800	
6 A(L)=BLANK	21810	
D0 7 L=1,12	21820	
7 AR(L)=0.0	21830	
READ(5,200)A	21850	
1F(EOF(5))8,9.8		
8 WRITE(6,225)	21850	
6 WRITE(0.22)	21860	

90 NO=14	2940
WRITE(6,620)	2950
92 CALL READ(1, AR)	2960
GO TO (1003,1001)IGO	2970
91 NIND=IFIX(ANS+.1)	2980
IF(NIND.LT.1.OR.NIND.GT.9) GO TO 93	2990
GO TO 94	3000
93 WRITE(6, 1024)	3010
GO TO 92	3020
94 IF(NS.GT.NQ) GO TO 1006	3030
95 NQ=15	3040
IF(NIND.GT.1) GO TO 97	3050
WRITE(6,625)	3060
CALL REAU(1, AR)	3070
GO TC (1000,1001) IGO	3080
96 PLT=ANS	3090

F01

97 IF(NS.GT.NQ) GO TO 1006	3100
100 NG=15	3110
WRITE(6,635) CALL READ(1,AR)	3120 3130
GO TO (1000,1001) IGO	3140
101 ANS16=ANS	3150
INGHT=0 IF(ANS16.EQ.2HNO) GO TO 102	3160 3170
WRITE(6,636)	3180
CALL READ(1, AR)	3190
IF(1G0.EQ.2) GO TO 1001 IF(ANS.EQ.4HBACK) GO TO 1004	3200 3210
INGHT=IFIX(ANS+.1)	3220
IF(INGHT.LT.1.OR.INGHT.GT.4) GO TO 1007	3230
IF(INGHT.EQ.1) GO TO 102 WRITE(6,637)INGHT	3240 3250
CALL READ(1,AR)	3260
IF(IGO.EQ.2) GO TO 1001	3270
IF(ANS.EQ.4HBACK) GO TO 1004 ANS16B=ANS	3280
102 IF(NS.GT.NO) GO TO 1006	3290 5300
105 NQ=17	3310
NSAVE=1 J=0	3320
NOBS2=NCBS+5	3330 3340
IF(ANS16.EQ.2HNO.OR.IWGHT.GT.1) GO TO 120	3350
106 NN=NIND+3	3360
WRITE(6,640)NOBS,NN,NIND IW=1	3370 3380
DO 109 1=1,NOBS2	3390
J=J+1	3400
N=LABEL(J) IF(N.EQ.NSAVE+1) WRITE(6,641)ITHSET(N),N,J	3410 3420
NSAVE=N	3430
1060 CALL READ(NN, AR)	3440
IF(IGO.EG.2) GO TO 1001 IF(ANS.EG.4HBACK) GO TO 1004	3450 3460
IF(AR(3).LT.0.) WRITE(6,1033)	3470
IF(AR(3).LT.0.) GO TO 1060	3480
J=IFIX(ANS+.1) IF(J.LT.1) GO TO 1007	3490 3500
IF(J.GT.NOBS) GO TO 107	3510
GO TO 108	3520
107 NAGR=8 GO TO 1008	3530 3540
108 CONTINUE	3550
DO 1080 L=1,NIND	3560
1080 XX(L,J)=AR(L+3) INDEX(J)=J	3570
Y(J)=AR(2)	3580 3590
WY(J)=AR(3)	3600
IF(J.EQ.NOBS)GO TO 125 109 CONTINUE	3610
WRITE(6,642)NOBS	3620 3630
GO TO 105	3640
20 NN=NIND+2 WRITE(6,655)NOBS,NN,NIND	3650
IW=2	3660 3670
DO 123 1=1,NOBS2	3680
	3690
N=LABEL(J) IF(N.EQ.NSAVE+1) WRITE(6,641)ITHSET(N),N,J	3700 3710
	57.10

G01

3720	
3730	
3740	
3750	
3760	
3770	
3770	
3/80	
3/40	
3800	
3810	
3820	
3830	
3840	
3850	
3860	- 1
3870	
3880	
3890	
3900	
3910	
3920	
1910	
3940	
3950	- 1
3060	
	3720 3730 3740 3750 3760 3770 3780 3810 3810 3810 3810 3820 3830 3840 3850 3860 3860 3860 3870 3860 3990 3990 3990 3990

6720 FORMAT(// " THE SMALLEST WEIGHTED RESIDUAL SUM OF SQUARES VALUE"/ *" FOUND IN SEARCH =",E10.4," AT'/)	12240
6725 FORMAT(// " THE SMALLEST RESIDUAL SUM OF SQUARES VALUE", * FOUND IN THE SEARCH = ",E10.4," AT" /)	12260 12270
6730 FORMAT(5X, " PAR.NO. ",11," =",E11.4,5X,40A1) 6731 FORMAT(/" WHICH IS SMALLER THAN THE VALUE (",E10.4,") OBTAINED",	12280 12290
* USING THE FOLLOWING"/" INITIAL PARAMETER ESTIMATES"/) 6735 FORMAT(/ "WHICH IS SMALLER THAN THE VALUE (",E10.4,") FOUND",	12300 12310
*" BY THE MINIMIZATION"/" PROCESS.THE MINIMIZATION IS THEREFORE", *" RESTARTED IN NEW MINIMUM REGION"/" USING THE PARAMETERS ABOVE"/)	12320
6740 FORMAT(/ " WHICH IS LARGER THAN THE VALUE (",E10.4,") FOUND BY", " THE MINIMIZATION PROCESS"/" AT")	12340 12350
6741 FORMAT(/" WHICH IS LARGER THAN THE VALUE (",E10.4,") OBTAINED", *" USING THE FOLLOWING"/" INITIAL PARAMETER ESTIMATES"/)	12360 12370
6745 FORMAT(// 77(1H*)) 6747 FORMAT(/ " DO YOU WANT TO MAKE ANY ADDITIONAL CORRECTIONS BEFORE",	12380 12390

FOZ

*" THE RESTART"//)	12400
6748 FORMAT(//" DO YOU WANT TO START THE MINIMIZATION PROCESS"/	12410
* USING THE PARAMETER VALUES FOUND IN THE SEARCH ABOVE" //)	12420
675 FORMAT(//" (20) DO YOU WANT TO REPEAT THE COMPUTATIONS ABOVE BUT /	12430
* WITH SOME CHANGE IN THE INPUTS"//)	12440
680 FORMAT(" DO YOU WANT ANOTHER RUN WITH A NEW SET OF OBSERVATIONS"//	12450
•)	12460
685 FORMAT(" I HOPE YOU HAVE BEEN PLEASED WITH THE PROGRAM."/	12470
*" IF YOU HAVE ANY SUGGESTIONS FOR ALTERATIONS PLEASE CONTACT ME"//	12480
*" REGARDS"//	12490
*" P.VENG PEDERSEN"//)	12500
690 FORMAT(" (21) DO YOU WANT TO USE THE FINAL COMPUTED PARAMETERS"/	12510
*" FROM THE RUN ABOVE AS STARTING PARAMETERS FOR THE NEXT RUN."//)	12520
692 FORMATC YOU CAN NOW MAKE ALTERATIONS IN PREVIOUS INPUTS ",	12530
* FOR THE NEXT RUN.")	12540
694 FORMAT(" (22) DO YOU WANT TO MAKE MORE ALTERATIONS"//)	12550
750 FORMAT(5(/),1X,77(1H*)/) 752 FORMAT((1X 20(1H*)) T A P U T D A T A " 20(1H*)///	12560
752 FORMAT(/1X,29(1H*), " I N P U T D A T A ",28(1H*)/// *" NUMBER UF VARIABLE PARAMETERS =",14/	12570
*" NUMBER OF PARAMETERS HELD CONSTANT =",14/	12590
*" NUMBER C" OBSERVATIONS =", 14/	12600
* NUMBER C OBSERVATIONS = 14/ * NUMBER OF DUMMY OBSERVATIONS = 14/	12610
NUMBER OF RESPONSE SYSTEMS = 14/ NUMBER OF INDEPENDENT VARIABLES = 14///	12620
* NUMBER OF INDEPENDENT VARIABLES = ,14///	12630
* PAR.NO. ,7X, ALLOWED RANGE ,9X, INITIAL ESTIMATE , * INITIAL STEP SIZE /1X,77(1H-))	12640
* INITIAL STEP SIZE"/1X.77(1H-))	12650
754 FURMAT(14,4(6X,E10.4))	12660
756 FORMAT(1X,77(1H-)/// OBS.NO. X1",12X," X2",10X," RESPONSE",7X,	12670
*" WEIGHT", 4X," NORM.WEIGHT" /1X,77(1H-))	12680
758 FORMAT(14,4X,E13.4,15X,E13.4)	12690
760 FORMAT(14,4X,E13,4,15X,3(E13,4,1X))	12700
762 FORMAT(14,4X,3(E13.4,1X))	12710
764 FORMAT(14,4X,5(E13.4,1X))	12720
765 FORMAT(/9X,	12730
* PLOT OF RESPONSE VERSUS INDEPENDENT VARIABLE X1 (ARB.UNITS)") 766 FORMAT(//	12740
* MAXIMUM NUMBER OF FUNCTION EVALUATIONS ALLOWED =",14/	12750
*" STOPPING CRITERION FOR CONVERGENCE =",E10.4)	12760
767 FORMAT(///4X,	12780
*" PLOT OF RESPONSE VERSUS SECOND INDEPENDENT VARIABLE X2",	12790
* (ARB.UNITS))	12800
768 FORMAT(STATISTICS ON THE PARAMETERS HAVE BEEN REQUESTED"/	12810
 CRITERION FOR EXPANSION OF SIMPLEX BEFORE QUADRATIC FITTING =", 	12820
*E10.4)	12830
770 FORMAT(" ANALYSIS OF RESIDUALS HAS BEEN REQUESTED")	12840
772 FORMAT(" RESIDUAL PLOTS HAVE BEEN REQUESTED")	12850
774 FORMAT(12860
* NO OUTPUT FROM THE FUNCTION MINIMIZATION HAS BEEN REQUESTED)	12870
776 FORMAT(" PARTIAL PROGRESS REPORT OF THE FUNCTION MINIMIZATION",	12880
*" HAS BEEN REQUESTED")	12890
778 FORMAT(" FULL PROGRESS REPORT OF THE FUNCTION MINIMIZATION",	12900
*" HAS BEEN REQUESTED") 780 EORMAT(" DI OT DE THE EITTED CHOVE AND DATA HAS BEEN DEGHESTED")	12910
780 FORMAT(" PLOT OF THE FITTED CURVE AND DATA HAS BEEN REQUESTED") 781 FORMAT(" WEIGHTING HAS BEEN DONE ACCORDING TO SCHEME NO.", 12,	12920
(WITH R=",E10.4,")")	12930
789 FORMAT(/1X, 16(1H+), " SUM-OF-SQUARES FUNCTION MINIMIZATION REPORT "	12950
*,16(1H+)//)	12950
790 FORMAT(/1X,77(1H+)//)	12970
791 FORMAT(/ " IT IS NECESSARY TO MAKE CORRECTION IN YOUR INPUT",	12980
*" TC REACH CONVERGENCE. ")	12990
792 FORMAT(" PROBABLY BECAUSE OF POSSIBLE CONSTRAIN(S) AS SEEN",	13000
*" FRCM")	13010
	·

GOZ

793 FORMAT(/ "_THE REASON_CAN ALSO BE UNSUITABLE VALUE(S) OF)	13020
794 FORMAT(/ TRY NEW INITIAL ESTIMATES OR RESCALE THE PARAMETERS ,	13030
*" OR CHANGE THE VALUE OF")	13040
795 FORMAT(/" OR CHANGE THE LIMITS OF ONE OR MORE OF THE PARAMETERS") 800 FORMAT(/1X,31(1H*)," R E S U L T S ",31(1H*)//	13050 13060
1X,77(1H)////	13070
* PAR.NO. ALLOWED RANGE", 5X, " INITIAL VALUE FINAL VALUE STD."	13080
*,"DEV.",4X," C.V.(PCT)"/1X,77(1H-))	13090
8010 FORMAT(/1X,31(1H*)," R E S U L T S ",31(1H*)//	13100
+1X,77(1H+)////	13110
*" PAR.NO. ALLOWED RANGE", 5X, " INITIAL VALUE FINAL VALUE"/	13120
*1X,57(1H-)) 8011 FORMAT(14,3X,4(E11.5,1X))	13130 13140
8012 FORMAT(1X, 57(1H-))	13150
802 FORMAT(14,3X,6(E11.5,1X))	13160
803 FORMAT(1X,77(1H-)///	13170
* PAR.NO. APPROX. 95 PCT. CONF.LIMITS ,/1X,38(1H-))	13180
8030 FORMAT(/ " THE MINIMUM FOUND BY THE MINIMIZATION",	13190
* PROCESS APPEARS / TO BE CONSTRAINED BY THE ,A5, * LIMIT OF PARAMETER NUMBER ,12)	13200
8031 FORMAT(/	13210 13220
* *** ENTER BIO OR BIT IF YOU WANT A NEW RUN WITH DIFFERENT"/	13230
**** PARAMETER LIMITS OR DIFFERENT PARAMETER ESTIMATES /	13240
* *** OTHERWISE TYPE -OK- TO CONTINUE"//)	13250
804 FORMAT(//	13260

		61340
	ITHFUN=LABEL(1)	21550
	S=Y(1)	21560
	IF(NIND.EQ.1) CALL MODEL(S,A(1),F,0)	21570
	Tranta Contraction Contraction Contraction	21370
	IF(NIND.GT.1) CALL MODEL(S,A,F,0)	21580
	IF(S.GE.1E50) GO TO 20	21590
	YCAL=S	21600
	YOBS=Y(1)	
		21610
	W=WYN(I)	21620
15	5 FUNC=FUNC+W+(YOBS-YEAL)+(YOBS-YEAL)	21630
	RETURN	21640
20	FUNC=S+1.000000001	
21		21650
	NVIOL=NVIOL+1	21660
	RETURN	21670
	END	21680
~		21080
1	***************************************	21690

F03

SUBROUTINE READ(N, AR) DIMENSION A(78), AR(12), AL(10)	21700 21710
COMMON/B5/NO,NS,IGO COMMON /B15/SKIP	21720
LOGICAL SKIP	21730 21740
DATA YES, AND, BACK, AO, AK, BLANK, PLUS, AMINUS, COMMA, DOT, F	21750
/1HY, 1HN, 1HB, 1HO, 1HK, 1H', 1H+, 1H-, 1H, , 1H, , 1HE/ DATA AL/1HO, 1H1, 1H2, 1H3, 1H4, 1H5, 1H6, 1H7, 1H8, 1H9 /	21760 21770
IGO=1	21780
SKIP=.FALSE.	21790
DO 6 L=1,78 A(L)=BLANK	21800 21810
D0 7 L=1,12	21820
AR(L)=0.0 READ(5,200)A	21830 21840
IF(EOF(5))8,9,8	21850
WRITE(6,225)	21860
GO TO 5 CONTINUE	21870 21880
D0 11 I=1,78	21890
IF(A(I).EQ.1HR) GO TO 139	21900
IF(A(I).EQ.BACK) GO TO 140 CONTINUE	21910 21920
IF(A(1).EQ.YES) GO TO 130	21930
IF(A(1).EQ.AND) GO TO 135 IF(A(1).EQ.AD.AND.A(2).EQ.AK) GO TO 145	21940 21950
1=0	21960
J=1	21970
K=0 FAC=1.	21980 21990
GO TO 15	22000
FAC=-1. I=I+1	22010 22020
IF(1.EQ.79) GO TO 125	22030
D0 20 L=1,10	22040
IF(A(I).EQ.AL(L)) GO TO 50 CONTINUE	22050 22060
IF(K.EQ.0) GO TO 30	22070
IF(A(1).E0.BLANK.OR.A(1).E0.COMMA.OR.A(1).E0.DOT.OR.A(1).E0.E) GO TO 55	22080 22090
WRITE(6,205)A(1)	22100
GO TO 5	22110
IF(A(I).EQ.BLANK) GO TO 15 IF(J.GT.1.AND.A(I).EQ.COMMA) GO TO 15	22120 22130
IF(A(I).EQ.AMINUS) GO TO 10	22140
IF(A(I).E0.DOT) GO TO 71 GO TO 25	22150
IF(K.EQ.0) IFLAG=1	22160 22170
AR(J)=AR(J)+FLOAT(L-1)*(10.**(-K))	22180
K=K+1 GO TO 15	22190 22200
AR(J)=FAC+AR(J)+(10.++(I-IFLAG-1))	22210
IF(A(1).EQ.DOT.AND.A(1+1).EQ.COMMA.OR.A(1+1).EQ.BLANK) GO TO 60 IF(A(1).EQ.DOT) GO TO 70	22220
IF(A(I).EQ.E) GO TO 90	22230 22240
J=J+1	22250
IF(J.GT.N) GO TO 65 FAC=1.	22260 22270
K=0	22280
GO TO 15	22290
I=I+1 IF(1.E0.79) RETURN	22300 22310

G03

IF(A(I).EQ.BLANK) GO TO 65	22320
WRITE(6,215)N GO TO 5	22330
70 K=0	22340 22350
71 Î=I+1	22360
K=K+1	22370
D0 75 L=1,10	22380
IF(A(1).EQ.AL(L)) GO TO 85	22390
75 CONTINUE	22400
IF(A(I).EQ.BLANK.OR.A(I).EQ.COMMA) GO TO 60	22410
IF(A(1).EQ.E) GO TO 90	22420
GO TO 25 85 AR(J)=AR(J)+FAC+FLOAT(L-1)*(10.**(-K))	22430
GO TO 71	22440 22450
90 KK=0	22450
NEXP=0	22470
NFAC=1	22480
NCOUNT=0	22490
GO TO 100	22500
95 NFAC=-1	22510
100 1=1+1 IF(1.EQ.79) GO TO 126	22520
NCOUNT=NCOUNT+1	22530 22540
IF(NCOUNT.EQ.5) GO TO 126	22550
IF(NCOUNT.GT.1) GO TO 105	22560
TETATI ED BLANK OD ATT ED DILLES CO TO 100	22670

109 CUNIINUE HRITE(6,642)NOBS G0 T0 105 720 NN=NIND+2 HRITE(6,655)NOBS,NN,NIND IW=2 D0 123 I=1,NOBS2 J=J+1 N=LABEL(J) IF(N.E0.NSAVE+1) WRITE(6,641)ITHSET(N),N,J

G01

MSAVE:H 770 TALL BEDGUELAND 970 TELMEND: UNDER DO 1004 970 TELMEND: UNDER DO 1007 980 TELMEND: UNDER DO 1005 980 TELMEND: UNDER DO 1001 980 TELMEND: UNDER DO 1004 980 TELMEND: UNDER DO 1004 980 TELMEND: UNDER DO 1004 980 </th <th></th> <th></th> <th></th>			
CALL READOM.ADD. 3730 FF00E CD.2 GD TO 1004 3730 JEFFIXCAMS.10 D0 1007 FF01E CD.2 GD TO 1007 3730 IFO.1 TO 1008 3730 ZO TO 1008 3800 THOSC.017-1 3800 THOSC.017-1 3800 THOSC.017-1 3800 TO TO 120 12.5 ZO TOT 120 3800 TO 120 TO 120 3800 TO 12			
CALL BEADOM ARD - 1001 3730 IF 102 EC 27 BOYL 0D 10 1004 3730 J = FF 1X CMMS - 11 / 60 T0 1007 3730 IF 0.1 T1 / 60 T0 1007 3730 IF 0.1 T1 / 60 T0 1007 3730 IC 10 T0 122 ES1 00 T0 121 3730 IC 10 T0 122 ES1 00 T0 121 3700 IC 10 T0 1008 3800 IC 00 T0 200 T1 008 3800 IC 00 T202 II NND 3800 IC 10 T0 1008 3800 IC 10 T1 008 3800 IC 10 T1 008 3800 IC 10 T0 1008 3800 IC 10 T1 100 125 IC 10 T1 100 3800 IC 10 T1 100 3800 IC 10 T1 100 125 IC 10 T1 100 126 IC 10 T1 100 126<		NSAVE=N	3720
I FCAMS.ED. 4480.KC) 60 T0 1004 3750 I FCL GT. NOBS) 60 T0 121 3760 Q TO T22 3760 121 MAGES 3800 22 CONTINUE 3800 20 OT 120 3800 20 OT 120 3800 20 OT 120 3800 20 OT 120 141 MAGES 20 OT 180E 3800 20 OT 220 L=1, NIND 3800 20 OT 220 L=1, NIND 3800 20 OT 220 L=1, NIND 3800 20 T 25, OWT NUE 3800 21 F CMER (ABOES) GO TO 125 3800 22 (DUT 180E 3900 21 F CMER (ABOE) ARD 3900 22 (DUT 180E 3900 23 (DUT 180E (AL 2900) 3900 24 (DUT 180E (AL 2900) 3900 25 (DUT 180E (AL 2900) 3900 26 (DUT 180E (AL 2900) 3900		CALL READ(NN, AR)	3730
J=F1KXAMS-10 3760 IFG.J.T.F.I.B.CO TO 1007 3770 IG. TO 122 3770 IMARAS 3700 IMARAS 3800			
IF C4_UT_13 G0 T0 1007, 3770 IF C4_UT_13 M005 G0 T0 121 3780 IF MAREA 3800 IC TO Too 3800 IZ CONTINUE 3900 IZ CONTINUE 1000 IZ CONTINUE </td <td></td> <td></td> <td></td>			
IF CJ, GT, MOES, DG TO 121 3780 121 MACH 22 3510 122 CONTINUE 1800 122 CONTINUE 1800 122 CONTINUE 1800 123 CONTINUE 1800 124 CONTINUE 1800 125 CONTINUE 1800 126 CONTINUE 1800 127 CONTINUE 1800 128 FGL ECONST 3500 129 FGL ECONST 3500 120 CONTINUE 1800 121 FGL ECONST 3500 122 CONTINUE 1800 123 FGL ECONST 3500 124 FGL ECONST 3500 125 FGL ECONST 3500 126 FFAINDOST 3500 127 CONTINUE 3500 128 FGL ECONST 3500 129 FGL ECONST 3500 120 FGL ECONST 3500 121 FGL ECONST 3500 122 FGL ECONST 3500 123 FGL ECONST 3500 124 FGL ECONST 3500 125 FGL ECONST 3500 126 FGL ECONST 3500 127 FGL ECONST <t< td=""><td></td><td>J-1F1A(ARST.1) IE(1 T 1) CO TO 1007</td><td></td></t<>		J-1F1A(ARST.1) IE(1 T 1) CO TO 1007	
12 00, T0, 122 3790 12 CONTINUE 3800 13 CONTINUE 3800 14 CONTINUE 3800 15 CONTINUE 3800 16 CONTINUE 3800 17 CONTINUE 1000 18 CONTINUE 1000 19 CONTINUE 10000 19 <		IF(J.GT.NORS) GO TO 121	
121 MACRES 300 122 CD TO TUGOS 3310 122 CD TO TUGOS 3310 122 CD TIZQ 11, 1400 123 COLTARCLES) 3340 1100EX(2)=21, 1400 3350 123 CONTINUE 3360 124 CONTINUE 3360 125 CONTINUE 3360 126 CONTINUE 3360 127 CONTINUE 3360 126 CONTINUE 3360 127 CONTINUE 3360 128 CONTINUE 3360 129 CONTINUE 3360 120 CONTINUE 3360 121 CONTINUE 3360 120 CONTINUE 3360 121 CONTINUE 3360 121 CONTINUE 3360 122 CONTINUE 6000 121 FIGNINGUE, COLOTION 3960 121 CONTINUE 6000 3960 121 CONTINUE 6000 1310			
122 CONTINUE 320 120 CONTINUE 330 122 CONTINUE 330 123 CONTINUE 330 124 CONTINUE 330 125 CONTINUE 330 125 CONTINUE 330 126 CONTINUE 330 127 CONTINUE 330 128 CONTINUE 330 129 CONTINUE 330 120 CONTINUE 3900 121 CONTINUE 3900 122 CONTINUE 3900 120 CALL RACTINNO CONTINUE 3900 121 CONTINUE 3900 122 CONTINUE 3900 124 FERINECONTON 3900 125 CONTINUE 3900 126 CONTINUE 3900 127 CONTINUE 3900 126 CONTINUE 3900 127 CONTINUE 3900 128 CONTINUE 3900 129 CONTINUE <	121	NAGR=8	
D0 1220 L=1, NHO 3340 INDEX(J)=J 3340 3340 IND		GO TO 1008	
1220 XX(L,J)=AR(L+2) 3560 140 MURCK(J)=J 3560 123 CONTINUE 3560 123 CONTINUE 3560 124 CONTINUE 3560 125 CONTINUE 3560 126 CONTINUE 3560 127 CONTINUE 3560 128 CONTINUE 3560 129 CONTINUE 3560 120 CONTINUE 3560 121 CONTINUE 3560 122 CONTINUE 3560 120 CONTINUE 3560 121 CONTINUE 3560 122 CONTINUE 3560 123 CONTINUE 3560 124 CONTINUE 3560 125 CONTINUE 3560 126 F(NICRA) CONTINUE 3600 127 CONTINUE 100 3560 128 CONTINUE 100 131 129 CONTINUE 4010 4010 120 CONTINUE	122		
INDEX.(J)=J 3350 IV IF(J):EUE 3360 IV IF(J):EUE 3370 IV IF(J):EUE 3370 IV IF(J):EUE 3370 IV IF(J):EUE 3370 IV IF(E):EUE IV 3370 IV IF(E):EUE IV 3370 IV IF(E):EUE IV 3370 IV IF(E):EUE IV 100 IV IF(E):EUE IV 100 100 IV IF(E):EUE IV IV IV IV IF(E):EUE IV IV <tr< td=""><td>1220</td><td>UU 1220 L=1, NINU YY(1 1)=AD(1+2)</td><td></td></tr<>	1220	UU 1220 L=1, NINU YY(1 1)=AD(1+2)	
Y(J)=AR(2) 360 IPS(J)=E0, MOBS to TO 125 3870 MIT EC, GAC2NOBS 3880 MIT EC, GAC2NOBS 3890 MIT EC, GAC2NOBS 3990 D0 125 CONTINUE 3990 D0 126 L=1, MOBS 3990 XLD=XX(1, 1) 3990 MIT EC, GAC2NOBS 3990 YLD=XX(1, 1) 3990 TFCKE, GT, MOJ GO TO 1006 3990 TFCKE, GT, MOJ GO TO 1006 3990 TFCKE, GE, 2000 TO 1004 3990 TFCKE, GE, 2000 TO 1004 3990 TFCKE, GE, 2000 TO 1004 4000 MIT EC, GE, 2000 TO 1004 4000 TFCKE, GE, 2000 TO 1004 4000 MIT EC, GE, 2000 TO 1010 4000 MIT EC, MOBS 4000 MOB TE, GE, 2000 TO 1010 4000 MIT EC, MOBS 4000 TFCKE, MOBS 10 TO 104 TFCKE, MOBS 10 TO 104 TFCKE, MOBS 10 TO 104 TFCKE, MOBS	1220	INDEX(J)=J	3040
IF4.1E0.M085: G0 T0 125 3370 123 CONTINUE 3390 120 T0 120 3900 120 T0 120 3900 120 T0 120 3900 120 T0 120 3900 121 C0 T12W 3900 122 C0WT12W 3900 123 C0WT12W 3900 124 C12W 3900 125 C11 T0 120 3900 126 IF4(IND.Gf.1) 2(1)=xx(2,1) 3940 126 IF4(IND.Gf.1) 2(1)=xx(2,1) 3940 126 IF4(IND.Gf.1) 2(1)=xx(2,1) 3940 127 C00 T0 1006 3950 128 IF1E4.6600 3960 129 IF4(IND.Gf.1) 2(1)=xx(2,1) 3940 120 C00T IF4.600 C0 T0 1004 4000 121 IF4(IND.Ec.) -1 GR.NOP.E01.0R.NOP2.E01.0R.NOB52.E01)60T0 1013 4000 120 C00T INUE 4000 4000 121 IF4(IND.EC.1) GO T0 1012 4000 120 C00T INUE 4100 4100			
HATTE (6, 642) NOBES 3800 GO TO 120 3900 DO T322 ET 100S 3900 DO T325 ET 100S 3900 126 (F (AIRO, GT, 1), 2(1)=XX(2,1) 3940 126 (F (AIRO, GT, 1), 2(1)=XX(2,1) 3940 126 (F (AIRO, GT, 1), 2(1)=XX(2,1) 3940 126 (F (AIRO, GT, 10, 2G) TO 1006 3990 HTTE (6, 660) 3990 CALL READ(1, AR) 3990 IF (AIRO, GL, 2) GD TO 1001 3990 IF (AIRO, E0, 2-1), OR, NOP, E0, -1, OR, NOP2, E0, -1), GOTO 1013 4040 VO 133 I =1, MOBS 4000 IF (AIRO, E.G, 1027) 4000 VO 133 I =1, MOBS 4000 IF (AIRO, E.G, 10, OR, NOP, E0, -1, OR, NOP2, E0, -1), GOTO 1013 4040 VO 133 I =1, MOBS 4000 IF (AIRO, E.G, 10, OR TO 1310 4040 VO 133 I =1, MOBS 4100 IF (AIRO, C.G, 1), OR TO 133 4100 IF (AIRO, C.G, 1), OR TO 133 4100 VO 132 J =1, MOBS 4100 JO 100, T, MAX, AMO, MAX, NE, 1) GO TO 1010 4100 JO 104, I =1, MOP 4100 JO 105, I M 4100		IF(J.EQ.NOBS) GO TO 125	
12 GOT 120 3900 12 CONTINUE 3920 126 TCINUE 3920 127 TCIN 3920 128 TCIN 3920 129 TCIN 3920 120 NG-18 3920 121 TCIN 3930 121 TCIN 3930 121 TCIN 3930 121 TCIN TCIN 120 TCIN TCIN 121 TCIN TCIN 122 TCIN TCIN TCIN 123 TCIN TCIN TCIN 124 TCIN TCIN TCIN 125 TCIN TCIN TCIN TCIN 126 TCIN TCIN TCIN TCIN 127 TCIN TCIN TCIN	123	CONTINUE	
125 CONTINUE 3910 00 126 Fit (NDS 3920 Xt125Xx(1;1) 3930 3940 126 Fit (NDG, GT, 12(1)=XX(2;1)) 3940 130 MGT (Fit (6, 660)) 3990 147 Fit (NG, 660) 3990 157 Fit (NG, 660) 3990 167 Fit (NG, 600) 3990 171 Fit (NG, 600 T0 1004 4000 180 Fit (NG, 600 T0 1004 4000 191 Fit (NG, 60 T0 131) 4000 191 Fit (NG, 60 T0 1331 4000 191 Fit (NG, 60 T0 1331) 4000 191 Fit (NG, 60 T0 130) 4000 191 Fit (NG, 60 T0 1310) 4000 191 Fit (NG, 60 T0 1310) 4000 191 Fit (NG, 70 T0 1012) 4000 190 T31 Fit (NG, 805 C0 T0 1310) 4000 191 Fit (NG, 70 T0 1012) 4000 192 CONTINUE 4100 193 Fit (NG, 70 T) 1012 4100 193 CONTINUE 4100		WRITE(6,642)NOBS	
D0 126 1=1, NOBS 3920 X1)=5xX(1,1) 3930 126 1F(MIND GT, 1) Z(1)=xX(2,1) 3940 FF(MEG,T,MO) GT 10 1006 3950 MRTE(6,660) 3970 CALL READ(1,AR) 3970 FF(ME,G.2)+B(CX,00 TO 1001 3970 FF(ME,G.2)+B(CX,00 TO 1004 3970 FF(ME,G.2)+B(CX,00 TO 1004 3970 FF(ME,G.2)+B(CX,00 TO 1004 4000 HRITE(6,1027) 6010 GO TO 130 4000 HRITE(6,1027) 600 TO 131 131 15(HEMC,EQ1,0R.MOP.EQ1.OR.MOP2.EQ1.OR.NOBS2.EQ1)60TO 1013 4000 DO 135 1=1, NOBS 4000 F(AMS16.ME.3HYES) GO TO 1310 4000 IF (HEMT, GT, 1) WY(1)=HEIGHT(Y(1), INGHT, ANS16B) 4000 DO 132 1=1, NOBS2 4100 IF (INDEXCJ).EQ.1:D GO TO 1012 4000 DO 133 1=1, NOBS2 4100 IF (INDEXCJ).EQ.1:D GO TO 133 4110 IS10 CONTINUE 4100 IF (INDEXCJ).EQ.1:D GO TO 1012 4100 IS10 CONTINUE 4100 IF (INDEXCJ).EQ.1:D GO TO 143 4200 IS10 CONTINUE 4100 <td>176</td> <td></td> <td></td>	176		
X(1)=XX(1,1) Z(1)=XX(2,1) Z(1)=XX(2,1) Z(1)=XX(2,1) Z(1)=XX(2,1) Z(1)=XX(2,1) Z(1)=XX(2,1) Z(1)=XX(1,1) Z(125		
126 IF(MIND.GT.1) 2(1)=XX(2,1) 3940 130 MQ=18 3960 130 MQ=16 3970 CALL READ(5, GL, AR) 3970 FC(16, GL, 22, GD TO 1001 3990 IF(CMS, E0, HBACK) GO TO 1004 3990 IF(CMS, E0, HBACK) GO TO 1004 4000 WRITE (6, 1027) GO TO 131 4000 GO TO 130 IF(MIND, GT.1) OR DT 0.131 4000 IF(INE) CE, 0.2, 10, NOP, E0, -1.0R. NOP2, E0, -1.0R. NOBS2, E0, -1)GOTO 1013 4050 J1 IF(WHU, E, D, -1, OR, NOP, E0, -1.0R. NOBS2, E0, -1)GOTO 1013 4050 J1 IF(MIND, GT, 1) W(1)=HEIGHT(Y(1), INGHT, ANS16B) 4050 J2 J-1, NOBS2 4000 D0 132 J-1, NOBS2 4100 I52 LONTINUE 4100 4100 I52 LONTINUE 4100 4100		X(1)=XX(1,1)	
IF(NS.GT.ND) E0 T0 1006 3950 MRITE (6,600) 3970 CALL READ(1,AR) 3980 IF(160.E0.22) E0 T0 1001 3990 IF(160.E0.22) E0 T0 1004 4000 IF(ARS.E0.4HBARX) G0 T0 1004 4000 MRITE (6,0027) 4010 MRITE (6,0027) 4010 MRITE (6,0027) 4020 JO (153) I=1,0085 4010 IF(ARS.E0.2HBARX) G0 T0 1310 4000 IF(ARS16.ME.3HYES) G0 T0 1310 4050 IF(ARS16.ME.3HYES) G0 T0 1012 4080 IF(ARS16.ME.3HYES) G0 T0 133 4110 ISO CONTINUE 4080 IF(ARS16.ME.3HYES) G0 T0 133 4110 ISO CONTINUE 4120 J-0 4120 J-10 CONTINUE 4120 J-20 4130 MSAVE=1 4120 MSAVE=1 4130 MCITE(6,1022) I H 4130 G0 T0 (106,120) IH 4130 IF(ARS11.E0,2HNOL) GO	126	IF(NIND.GT.1) Z(1)=XX(2.1)	
WRITE(6,600) 3970 CALL READ(1, AR) 3980 IF(IGO.E0.2) GO TO 1001 3990 IF(ANS.E0.4HBACK) GO TO 1004 4010 WRITE(6,1027) 4020 GO TO 150 4030 131 IF(NEWC.E01.0R.NOP.E01.0R.NOP2.E01.0R.NOBS2.E01)GOTO 1013 4040 00 T33 I=1, MOBS 4050 IF(ANS16.ME.3HYES) GO TO 1310 4050 IF(INGHT.GT.1) WY(1)=WEIGHT(Y(1), INGHT, ANS16B) 4050 131 OCONTINUE 4070 IF(INGEX.GL.1). GO TO 1012 4080 1310 CONTINUE 4010 IF(INDEX(J).E0.1) GO TO 103 4070 IF(INDEX(J).E0.1) GO TO 133 4110 J=0 4130 NSAVE=1 4130 WRITE(6,1032)1 4130 IF(INDEX(J).E0.1) IH 4150 00 TO 140,120 1H 4150 132 CONTINUE 4170 IF(NIDP.GT.HAX.AND.'NAX.NE.1) GO TO 1010 4180 00 TO 143 4210 IF(K).E0.1) GO TO 143 4200 00 TO 143 4210 IF(K).E0.1) GO TO 143 4200 00 TO 143 4200		IF(NS.GT.NQ) GO TO 1006	
CALL READ(1, AR) IF (IBC) E0.2.2 (50 T0 1001 IF (ANS E0. 4HBACK) E0 T0 1004 IF (ANS E0. 4HBACK) E0 T0 1031 WRITE(6, 1027) GU T0 130 131 IF (NFUNC.E01.0R.NOP2.E01.0R.NOBS2.E01)GUT0 1013 4040 00 133 I=1, M0BS IF (ANS 16. ME. 3HYES) GU T0 1310 IF (ANS 16. ME. 3HYES) GU T0 1012 16 (MV(1).LT.0.) GU T0 1012 17 (INCHT.GT.1) WV(1)=HE(BE(TY(1), INGHT, ANS 16B) 1310 CUNTINUE 100 132 J=1, M0BS2 IF (ANS 16. ME. 3HYES) GU T0 133 1310 CUNTINUE 00 132 J=1, M0BS2 CUNTINUE 4070 132 CUNTINUE 4100 133 CUNTINUE 4100 143 CUNTINUE 4100 141 J=1, M0P 144 J=1, M0P 145 (MOP, GT, MAX.AND.NAX.NE.1) GU T0 1010 4100 145 (J) GU T0 143 4100 141 J=1, M11 K=MF(J) 145 (K) E0. I) GU T0 143 4210 141 CUNTINUE 4210 141 CUNTINUE 4210 4	130		
IF(1G0.E0.2) G0 T0 1001 3990 IF(ANS.E0.4HBACK) G0 T0 101 4000 IF(ANS.E0.2HOK) G0 T0 131 4010 WRITE(6,1027) 4020 G0 T0 130 4020 J1 IF(NEWC.E01.0R.NOP.E01.OR.NOP2.E01.OR.NOBS2.E01)G0T0 1013 4040 D0 133 I=1, NOBS 4050 IF(ANS16.MC.SHYES) G0 T0 1310 4050 IF(INGHT.GT.1) WY(1)=WEIGHT(Y(1), IMGHT, ANS16B) 4060 IF(INGHT.GT.1) WY(1)=WEIGHT(Y(1), IMGHT, ANS16B) 4070 D0 133 J=1, NOBS2 4090 D0 133 J=1, NOBS2 4090 D0 133 J=1, NOBS2 4100 IF(INDEX(J).E0.I) G0 T0 1013 4100 J=0 4130 NAVE=1 4130 MRITE(6, 1032) I 4150 UT (100, GT, MAX, AND.MAX.NE.1) G0 T0 1010 4180 D0 144 I=1, NOP 4160 J=1, NOI 4120 D0 144 I=1, NOP 4200 IF(K.E0.I) G0 T0 143 4200 D0 144 I=1, NOP 4200 IF(FS(K) I.E.F2		WRITE(6,660)	
IF (AMS : E0. 4HBACK) GO TO 130 4000 IF (AMS : E0. 2HBAC & GO TO 131 4010 MRI TE (6, 1027) 4020 GO TO 130 4030 131 IF (NFUNC : E01. OR. NOP : E01. OR. NOBS2 : E01) GOTO 1013 4040 00 133 I = 1, NOBS 4050 IF (AMS1 6. HC. 3HYES) GO TO 1310 4060 IF (AMS1 6. HC. 3HYES) GO TO 1012 4080 101 COMTINUE 4080 102 OT 132 J = 1, NOBS2 4090 1031 COMTINUE 4080 1040 TO 132 J = 1, NOBS2 4100 1151 COMTINUE 4080 120 COMTINUE 4090 121 COMTINUE 4080 120 COMTINUE 4100 1310 COMTINUE 4100 132 J = 1, NOBS2 4100 133 COMTINUE 4100 134 COMTINUE 4100 135 COMTINUE 4100 136 OWTINUE 4100 137 COMTINUE 4100 138 COMTINUE 4100 139 COMTINUE 4100 140 CONTINUE 4100 157 COT 143 4200 158 COT 144 4100<		CALL READ(1, AR)	
IF(AMS.E0.2HOK) G0 T0 131 4010 WRITE(6, 1027) 4020 G0 T0 130 4030 131 IF(FUNC.E01.0R.NOP.E01.0R.NOB2.E01.0GTO 1013 4040 D0 133 I=1.NOBS 4050 IF(AMS16.ME_3MSES) G0 T0 1310 4060 IF(IMENT.GT.1) WY(I)=WEIGHT(Y(I),IWGHT,ANS16B) 4060 IF(IMENT.GT.1) WY(I)=WEIGHT(Y(I),IWGHT,ANS16B) 4070 IF(IMEXT.GT.1) WY(I)=WEIGHT(Y(I),IWGHT,ANS16B) 4070 131 IF(FUNC.E0.) G0 TO 1012 4080 1310 CONTINUE 4090 D0 132 J=1.N0BS2 4000 IF(INDEXCJ).E0.I) G0 TO 133 4110 132 CONTINUE 4120 J=0 4130 NSAVE=1 4160 WRITE(6, 1032) IW 4160 131 GONTINUE 4160 J=0 4160 VIA 4 I=1, NOP 4170 IF(NLODP,GT.MAX.AND.MX.NE.1) GO TO 1010 4180 D0 144 I=1, NOP 4100 IF(ANS11.E0.2HON GO TO 143 4200 IF(ANS11.E0.2HON GO TO 143 4200 IF(ANS11.E0.2HON GO TO 143 4200 IF(K.E0.1) GO TO 142 4230 <t< td=""><td></td><td>IF(IGU.EU.2) GU IU IUUI IE(ANS ED (UDACK) CO TO 100/</td><td></td></t<>		IF(IGU.EU.2) GU IU IUUI IE(ANS ED (UDACK) CO TO 100/	
HRITE (6, 1027) 4020 GO TO 130 4030 131 IF (NFUNC.EQ1. OR. NOP.EQ1. OR. NOBS2.EQ1)GOTO 1013 4040 00 133 I=1, NOBS 4050 1F (ANS16. N.E. 3HYES) GO TO 1310 4060 1F (INGHT.GT. I) W(1)=HEIGHT(Y(1), INGHT, ANS16B) 4070 1F (MY(1), LT.0.) GO TO 1012 4080 131 CONTINUE 4090 00 132 J=1, NOBS2 4110 132 CONTINUE 4090 07 IF (INPEX(J).ED.I) GO TO 133 4110 132 CONTINUE 4130 MSAVE=1 4130 WRITE (6, 1032)I 4160 133 CONTINUE 4150 00 TO (106,120) IH 4160 133 CONTINUE 4160 134 CONTINUE 4160 135 CONTINUE 4160 136 CONTINUE 4160 137 CONTINUE 4160 138 CONTINUE 4160 144 I=1, NOP 4190 0 TO (144 I=1, NOP 4190 0 TO (142 4230 141 CONTINUE 4210 142 IF(F(K)_EQ.I) GO TO 142 4230 142 IF(K) L, E, F2(K)) GO TO		IF (ANS ED 2HOK) GO TO 131	
GO TO 130 4030 131 IF(HFUNC.EG1.0R.NOP.EG1.0R.NOP2.EG1.0GTO 1013 4040 00 133 1-1,MOBS 4050 IF(ANS16.ME.SHYES) GO TO 1310 4050 4050 IF(IMGHT.GT.1) HY(1)=HEIGHT(Y(1),IMGHT,ANS16B) 4050 1310 CONTINUE 4070 IF(IMGHT.GT.1). MY(1)=HEIGHT(Y(1),IMGHT,ANS16B) 4070 1310 CONTINUE 4050 1310 CONTINUE 4070 132 J=1,NOBS2 4000 132 L=1,NOBS2 4000 132 L=1,NOBS2 4000 132 L=1,NOBS2 4100 132 L=1,NOBS2 4100 132 CONTINUE 4120 J=0 MATE(6,1032)1 4130 MRITE(6,1032)1 4150 4160 GO TO (106,120) 1W 4150 4160 131 DO 144, 1=1, MOP 4160 147 IF(KLODP,GT,MAX,AND,MAX,NE,1) GO TO 1010 4180 00 T0 (44, 1=1, MOP 4200 141 CONTINUE 4200 <td< td=""><td></td><td>WRITE(6,1027)</td><td></td></td<>		WRITE(6,1027)	
D0 133 I=1,NOBS 4050 IF (ANS16.ME.SHYES) G0 T0 1310 4060 IF (IMGHT.GT.1) HY(1)=HEIGHT(Y(1), INGHT, ANS16B) 4070 IF (IMGHT.GT.1) AY(1)=HEIGHT(Y(1), INGHT, ANS16B) 4070 1310 CONTINUE 4090 D0 132 J=1, NOBS2 4090 IF (IMDEX(J).E0.I) G0 T0 133 4110 132 CONTINUE 4120 J=0 4130 MSAVE=1 4120 MRITE (6, 1032) I 4150 G0 T0 (106,120) IH 4150 133 CONTINUE 4160 134 CONTINUE 4160 135 CONTINUE 4160 136 CONTINUE 4160 137 CONTINUE 4160 140 D0 144 I=1, NOP 4190 157 CAMS11.E0.2HNOD.GO TO 143 4200 00 144 I=1, NOP 4190 16 (K.E0.I) GO TO 142 4230 17 IF (K.E0.I) GO TO 142 4230 184 CONTINUE 4200 197 IF (K.E0.I) GO TO 143 4200 198 CONTINUE 4200 199 IF (K.E0.I) GO TO 143 4200 191 IF (K.E0.I) GO TO 1443 4200		GO TO 130	
IF(INCHT.GT.1) HY(I)=HEIGHT(Y(I), INGHT, ANS16B) 4070 IF(INCHT.GT.1) HY(I)=HEIGHT(Y(I), INGHT, ANS16B) 4080 1310 CONTINUE 4090 D0 132 J=1, NOBS2 4100 IF(INDEX.J).E0.1) GO TO 133 4110 J=0 4120 NSAVE=1 4120 HRITE(6, 1032)1 4160 GO TO (106, 120) IW 4160 133 CONTINUE 4170 IF(INDEX.J).E0.1) GO TO 1010 4180 D0 144 I=1, NOP 4190 IF(INDOP.GT.MAX.AND.NAX.NE.1) GO TO 1010 4180 D0 144 I=1, NOP 4200 D0 141 J=1, N11 4210 K=NF(J) 4200 IF(K.E0.I) GO TO 142 4230 IF(K.E0.I) GO TO 142 4230 IF(K.E0.I) GO TO 143 4200 IF(K.E0.I) GO TO 143 4200 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4270 IF(K)=F5(K).T.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4280 IF(K)=F5(K) 4280 4290 GO TO 144 4300 4300 I420 HYITUE 4300 4300 I420 HYITUE 4300 <	131	IF(NFUNC.E01.OR.NOP.E01.OR.NOP2.E01.OR.NOBS2.E01)GOTO 1013	
IF(INCHT.GT.1) HY(I)=HEIGHT(Y(I), INGHT, ANS16B) 4070 IF(INCHT.GT.1) HY(I)=HEIGHT(Y(I), INGHT, ANS16B) 4080 1310 CONTINUE 4090 D0 132 J=1, NOBS2 4100 IF(INDEX.J).E0.1) GO TO 133 4110 J=0 4120 NSAVE=1 4120 HRITE(6, 1032)1 4160 GO TO (106, 120) IW 4160 133 CONTINUE 4170 IF(INDEX.J).E0.1) GO TO 1010 4180 D0 144 I=1, NOP 4190 IF(INDOP.GT.MAX.AND.NAX.NE.1) GO TO 1010 4180 D0 144 I=1, NOP 4200 D0 141 J=1, N11 4210 K=NF(J) 4200 IF(K.E0.I) GO TO 142 4230 IF(K.E0.I) GO TO 142 4230 IF(K.E0.I) GO TO 143 4200 IF(K.E0.I) GO TO 143 4200 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4270 IF(K)=F5(K).T.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4280 IF(K)=F5(K) 4280 4290 GO TO 144 4300 4300 I420 HYITUE 4300 4300 I420 HYITUE 4300 <		DO 133 I=1,NOBS	
IF (HY (1), LT, 0.) GO TO 1012 4080 1310 CONTINUE 4090 D0 132 J=1, NOBS2 4100 IF (INDEX(J), E0, I) GO TO 133 4110 32 CONTINUE 4120 J=0 4130 NSAVE=1 4140 WRITE(6,1032)I 4150 GO TO (106,120) IW 4150 133 CONTINUE 4160 134 CONTINUE 4160 MRITE(6,1032)I 4160 WRITE(6,1032)I W 4160 135 CONTINUE 4160 136 OT 0 (106,120) IW 4160 137 CONTINUE 4160 138 CONTINUE 4160 140 DD 144 I=1, NOP 4170 141 CONTINUE 4200 141 CONTINUE 4210 141 CONTINUE 4220 142 IF(F ⁺ (K), E0, F2(K)) GO TO 143 4250 142 IF(F ⁺ (K), E0, F2(K)) GO TO 143 4250 142 ONTINUE 4280 142 ONTINUE 4280 142 ONTINUE 4250 142 IF(F ⁺ (K), E0, F2(K)) GO TO 143 4250 142 ONTINUE 4280 142 IF(F ⁺ (K)		IF(ANSIG.NE.SHTES) GU IU ISIU IE(INCHT CT 1) HY(I)-HEICHT(Y(I) INCHT ANGIAD)	
1310 CONTINUE 4090 D0 132 J=1,N0BS2 4100 132 CONTINUE 4110 J=0 4130 4130 NSAVE=1 4130 4130 HRTE(6,1032)1 4150 4160 GD T0 (106,120) IW 4160 4160 133 CONTINUE 4190 F(NLODP, GT, MAX, AND, MAX, NE, 1) GO TO 1010 4180 DD 144 1=1, M0P 4190 IF(K, ISO, I) GO TO 143 4200 IF(K, E0, I) GO TO 142 4230 IF(K, E0, I) GO TO 142 4230 IF(K, E0, I) GO TO 143 4250 IF(FS(K), LT, F1(K), OR, FS(K), GT, F2(K)) GO TO 1420 4270 F(K) = FS(K) F(K) 4200 IF(F(K) = C, F2(K)) GO TO 143 4260 IF(FS(K), LT, F1(K), OR, FS(K), GT, F2(K)) GO TO 1420 4270 FSAVE(K) = F(K) 4280 GO TO 144 4300 I420 HITE(6, 1030)K 4310 GO TO 144 4320		IF(WY(I) IT 0) GO TO 1012	
D0 132 J=1,N0B52 4100 IF(INDEX(J).E0.I) GO TO 133 4110 132 CONTINUE 4120 J=0 4130 NSAVE=1 4140 HRITE(6,1032)I 4150 GO TO (106,120) IW 4150 133 CONTINUE 4160 134 LET.NOP.GT.MAX.AND.*AX.NE.1) GO TO 1010 4180 D0 144 I=1,NOP 4190 IF(ANST1.E0.2HNO) GO TO 143 4200 D0 141 J=1,N11 4210 K=NF(J) 4220 IF(K.E0.I) GO TO 142 4230 IF(K.E0.I) GO TO 142 4230 IF(S(K).LT.F1((K).OR.FS(K).GT.F2(K)) GO TO 1420 4260 IF(FS(K).LT.F1((K).OR.FS(K).GT.F2(K)) GO TO 1420 4280 FSAVE(K)=F(K) 4280 I420 WRITE(6, 1030)K 4300 I420 WRITE(6, 1030)K 4310 I420 WRITE(6, 1030)K 4310	1310	CONTINUE	
IF(INDEX(J).E0.I) G0 T0 133 4110 132 CONTINUE 4120 J=0 4130 NSAVE=1 4140 MRTE(6,1032)I 4160 GD T0 (106,120) IW 4160 133 CONTINUE 4170 IF(NLODP.GT.MAX.AND.MAX.NE.1) G0 T0 1010 4180 D0 144 I=1,N0P 4170 IF(ANS11.E0.2HNO) G0 T0 143 4200 D0 141 J=1,N11 4210 K=MF(J) 4220 IF(K.E0,I) G0 T0 142 4230 141 CONTINUE 4250 142 IF(F*(K).E0.F2(K)) G0 T0 143 4260 IF(S(K).LT.F1(K).OR.FS(K).GT.F2(K)) G0 T0 1420 4270 F(X)=FS(K) 60 T0 143 4260 IF(S(K).EL.F1(K).OR.FS(K).GT.F2(K)) G0 T0 1420 4280 IF(A)=FS(K) 4280 4280 IF(K)=FS(K) 4280 4280 I420 WRITE(6, 1030)K 4310 4200 I420 WRITE(6, 1030)K 4310 4310 I420 WRITE(6, 1030)K 4310 4320	10.0	D0 132 J=1,N0BS2	
J=0 4130 NSAVE=1 4140 WRTE(6,1032)I 4140 GD TO (106,120) IW 4160 133 CONTINUE 4170 IF(NLODP.GT.MAX.AND.MAX.NE.1) GO TO 1010 4180 D0 144 I=1.NOP 4190 IF(ANS11.E0.2HND) GO TO 143 4200 D0 144 J=1,N11 4210 K=MF(J) 4220 IF(K.E0.I) GO TO 142 4230 141 CONTINUE 4250 142 IF(F*(K).E0.F2(K)) GO TO 143 4250 IF(SK).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4270 F(X)=FS(K) FSAVE(K)=F(K) GO TO 144 4300 1420 WRITE(6,1030)K 4310 CALL READ(1, AR) 4320		IF(INDEX(J).EQ.I) GO TO 133	
NSAVE=1 WRITE(6, 1032) I G0 T0 (106, 120) IW 4140 4150 4150 133 CONTINUE 4160 135 CONTINUE 4170 IF(NLDOP.GT.MAX.AND.MAX.NE.1) GO TO 1010 4180 4190 D0 144 I=1, NOP 4190 IF(ANST1.E0.2HNO) GO TO 143 4200 D0 141 J=1, N11 4210 K=MF(J) 4220 IF(K.E0.1) GO TO 142 4230 141 CONTINUE 4250 142 IF(F5'(K).E0.F2(K)) GO TO 143 4250 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4270 FSAVE(K)=F(K) 4280 GO TO 144 4300 142 O WRITE(6, 1030)K 4300 142 O WRITE(6, 1030)K 4310 CALL READ(1, AR) 4320	132		
HRITE(6, 1032)1 4150 G0 T0 (106, 120) IW 4160 133 CONTINUE 4170 IF(NLOOP.GT.MAX.AND.MAX.NE.1) GO TO 1010 4180 D0 144 I=1, NOP 4190 IF(ANST1.E0.2HNO) GO TO 143 4200 D0 141 J=1,N11 4210 K=NF(J) 4220 IF(K.E0.I) GO TO 142 4230 141 CONTINUE 4250 GO TO 143 4250 IF(K.E0.I) GO TO 142 4230 IF(K.E0.I) GO TO 143 4250 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4270 FSAVE(K)=FS(K) 60 TO 1420 4280 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4280 IVO TO 144 4300 4300 1420 WRITE(6, 1030)K 4310 4320			
133 CONTINUE 4170 IF(NLDOP.GT.MAX.AND.MAX.NE.1) GO TO 1010 4180 D0 144 I=1,NOP 4190 IF(ANST1.E0.2HNO) GO TO 143 4200 D0 141 J=1,N11 4210 K=MF(J) 60 TO 142 IF(K.E0.I) GO TO 142 4230 141 CONTINUE 4250 142 IF(F'(K).E0.F2(K)) GO TO 143 4250 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4270 F(K)=F5(K) 4280 FSAVE(K)=F(K) 4290 GO TO 144 4300 1420 WRITE(6,1030)K 4310 CALL READ(1,AR) 4320		NDAVE-1 UPITE(6 1032)1	
133 CONTINUE 4170 IF(NLDOP.GT.MAX.AND.MAX.NE.1) GO TO 1010 4180 D0 144 I=1,NOP 4190 IF(ANST1.E0.2HNO) GO TO 143 4200 D0 141 J=1,N11 4210 K=MF(J) 60 TO 142 IF(K.E0.I) GO TO 142 4230 141 CONTINUE 4250 142 IF(F'(K).E0.F2(K)) GO TO 143 4250 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4270 F(K)=F5(K) 4280 FSAVE(K)=F(K) 4290 GO TO 144 4300 1420 WRITE(6,1030)K 4310 CALL READ(1,AR) 4320		GO TO (106,120) IN	
IF(NL00P.GT.MAX.AND.MAX.NE.1) GO TO 1010 4180 D0 144 I=1,N0P 4190 IF(ANS11.E0.ZHND) GO TO 143 4200 D0 141 J=1,N11 4210 K=MF(J) 4220 IF(K.E0.I) GO TO 142 4230 141 CONTINUE 4250 GO TO 143 4250 IF(K.E0.I) GO TO 142 4230 141 CONTINUE 4250 GO TO 143 4250 IF(SK).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4270 F(X)=F5(K) 4280 FSAVE(K)=F(K) 4280 GO TO 144 4300 1420 WRITE(6, 1030)K 4310 CALL READ(1, AR) 4320	133	CONTINUE	
IF(ANS11.E0.2HN0) G0 T0 143 4200 D0 141 J=1,N11 4210 K=NF(J) 4220 IF(K.E0.I) G0 T0 142 4230 Yator TiNUE 4220 G0 T0 143 4250 142 IF(F'(K).E0.F2(K)) G0 T0 143 4260 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) G0 T0 1420 4270 FSAVE(K)=F5(K) 4280 G0 T0 144 4300 1420 WRITE(6, 1030)K 4310 CALL READ(1,AR) 4320		IF(NLOOP.GT.MAX.AND.MAX.NE.1) GO TO 1010	
D0 141 J=1,N11 4210 K=NF(J) 4220 IF(K, EO, I) GO TO 142 4230 141 CONTINUE 4240 GO TO 143 4250 142 IF(F*(K).EO.F2(K)) GO TO 143 4260 IF(F*(K).EO.F2(K)).GT.F2(K)) GO TO 1420 4270 F(K)=FS(K) 4280 FSAVE(K)=F(K) 4290 GO TO 144 4300 1420 WRITE(6, 1030)K 4310 CALL READ(1,AR) 4320		DO 144 I=1,NOP	
K=MF(J) 4220 IF(K,EQ,I) GO TO 142 4230 141 CONTINUE 4240 GO TO 143 4250 142 IF(F'(K).EQ.F2(K)) GO TO 143 4260 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4270 F(X)=FS(K) 4280 GO TO 144 4300 1420 WRITE(6, 1030)K 4310 CALL_READ(1,AR) 4320			
IF(K.EQ.1) GO TO 142 4230 141 CONTINUE 4240 GO TO 143 4250 142 IF(F'(K).E0.F2(K)) GO TO 143 4260 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420 4270 FSAVE(K)=F(K) 4290 GO TO 144 4300 1420 WRITE(6,1030)K 4310 CALL READ(1,AR) 4320		UU 141 J=1,N11	
141 CONTINUE 4240 G0 T0 143 4250 142 IF(F*(K).E0.F2(K)) G0 T0 143 4260 IF(F\$(K).LT.F1(K).OR.FS(K).GT.F2(K)) G0 T0 1420 4270 F(K)=FS(K) 4290 G0 T0 144 4300 1420 WRITE(6,1030)K 4310 CALL READ(1,AR) 4320			4220
G0 T0 143 4250 142 IF(F'(K).E0.F2(K)) G0 T0 143 4260 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) G0 T0 1420 4270 F(X)=FS(K) 4280 G0 T0 144 4290 G0 T0 144 4300 1420 WRITE(6,1030)K 4310 CALL READ(1,AR) 4320	141		
142 IF(F ¹ (K).E0.F2(K)) G0 T0 143 4260 IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) G0 T0 1420 4270 F(K)=FS(K) 4280 FSAVE(K)=F(K) 4290 G0 T0 144 4300 1420 HTE(6,1030)K CALL READ(1,AR) 4320		GO TO 143	
IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) G0 T0 1420 4270 F(K)=FS(K) 4280 FSAVE(K)=F(K) 4290 G0 T0 144 4300 1420 WRITE(6,1030)K 4310 CALL READ(1,AR) 4320	142	IF(F'(K).EQ.F2(K)) GO TO 143	
F(k)=FS(k) 4280 FSAVE(k)=F(k) 4290 G0 T0 144 4300 1420 WRITE(6, 1030)k 4310 CALL_READ(1,AR) 4320		IF(FS(K).LT.F1(K).OR.FS(K).GT.F2(K)) GO TO 1420	4270
GO TO 144 4300 1420 HRITE(6,1030)k 4310 CALL_READ(1,AR) 4320			4280
1420 WRITE(6,1030)k 4310 CALL READ(1,AR) 4320			
CALL READ(1, AR) 4320	1/20	UD 10 144	
	1420		
1000		1F(1G0,E0,2) G0 T0 1001	

H01

	IF(ANS.ED.4HBACK) GO TO 1004	4340
	IF(ANS.EQ.2HOK) GO TO 143	4350
	WRITE(6,1027)	4360
	GO TO 1420	4370
14	3 F(1)=F1(1)+0.5*(F2(1)-F1(1))	4380
1.1	FSAVE(I)=F(I)	4390
14	4 CONTINUE	4390
1.4		4400
	NEWNOP=NOP	4410
	DO 150 I=1,NOP	4420
	IF(ANS12.E0.2HNO) GO TO 148	4430
	D0_145_J=1,N12	4440
	K=NT(J)	4450
	IF(K.EQ.1) GO TO 146	4460
14	5 CONTINUE	4470
1000	GO TO 148	4480
14	6 IF((STPS(1)+F(1)).LT.F1(1).OR.(STPS(1)+F(1)).GT.F2(1)) GO TO 147	4490
	STEP(1)=STPS(1) G0 T0 149	4500
	GO TO 149	4510
14	7 WRITE(6,1031)1 CALL READ(1,AR) IF(100.60.2) GO TO 1001 IF(ANS.ED.4HBACK) GO TO 1004	4520
	CALL READ(1, AR)	4530
	IF(1G0,E0,2) GO TO 1001	4540
	IF(ANS.EQ.4HBACK) GO TO 1004	4550
	IF (ANS.EQ.2HOK) GO TO 148	4560
	WRITE(6,1027)	45/0
	GO TO 147	4580
14	8 DEL1=F(1)-F1(1)	4590
	DE(2=E2(1)-E(1))	4600
	IE(DEL1 GT DEL2) STEP(L)=-(0 3+0 1/ELOAT(L))+DEL1	4610
	IF(DEL1 LE DEL2) STEP(1)=(0 1+0 1/EL DAT(1))+DEL2	4620
	IF(DEL1.GT.DEL2) STEP(I)=-(0.3+0.1/FLOAT(I))+DEL1 IF(DEL1.LE.DEL2) STEP(I)=(0.3+0.1/FLOAT(I))+DEL2 STPS(I)=STEP(I)	4630
14	9 STEPS(1)=STEP(1)	4030
	TELEVILLE CONTINUE CONTINUE	4640

781 FORMAT("WEIGHTING HAS BEEN DONE ACCORDING TO SCHEME NO.",12, " (WITH R=",E10.4.")")	
789 FORMAT(/1X,16(1H+), SUM-OF-SQUARES FUNCTION MINIMIZATION REPORT " +,16(1H+)//)	ti)
790 FORMAT(/1X,77(1H+)//)	
791 FORMAT(/ "IT IS NECESSARY TO MAKE CORRECTION IN YOUR INPUT", * TO REACH CONVERGENCE	
792 FORMAT(" PROBABLY BECAUSE OF POSSIBLE CONSTRAIN(S) AS SEEN", *" FRCH")	

GOZ

93 FORMAT(/ "THE REASON CAN ALSO BE UNSUITABLE VALUE(S) OF) 94 FORMAT(/ TRY NEW INITIAL ESTIMATES OR RESCALE THE PARAMETERS", • OR CHANGE THE VALUE OF)	
S ENDMAT(/" OR CHANCE THE LIMITS OF ONE OR MORE OF THE DADAMETERS"	
95 FORMAT(/" OR CHANGE THE LIMITS OF ONE OR MORE OF THE PARAMETERS 00 FORMAT(/1X,31(1H+)," R E S U L T S ",31(1H+)// +1X.77(1H+)///	1
1X,77(1H)//// * PAR.NO. ALLOWED RANGE",5X," INITIAL VALUE FINAL VALUE STD." * DEV: "4X," C.V.(PCT)"/1X,77(1H-)) 10 FORMAT(/1X,31(1H*)," R E S U L T S ",31(1H*)//	
, DEV. ,4X, C.V. (PCT) //X,77(1H-)) 10 FORMAT(/1X,31(1H), R E S U L T S ",31(1H*)//	
10 FORMAT(/1X,31(1H*), " R E S U L T S ",31(1H*)//	
1X,//(1H)////	
* PAR. NO. ALLOWED RANGE, JA, INITIAL VALUE FINAL VALUE /	
11 FORMAT(14, 3X, 4(E11, 5, 1X))	
12 FORMAT(1X, 57(1H-))	
D2 FORMAT(14,3X,6(E11.5,1X))	
- DAD NO ADDDOV OS DET CONE LINITE: (1V 70/14-)	
TO FORMAT(/ " THE MINIMUM FOUND BY THE MINIMIZATION"	
* PROCESS APPEARS / TO BE CONSTRAINED BY THE .AS.	
*, DEV, *, &, C.V. (PCT) /1X, //(1H-)) 10 FORMAT(/1X, 31(1H+), " R E S U L T S ",31(1H+)// *1X, 77(1H+)//// * PAR.NO. ALLOWED RANGE",5X, " INITIAL VALUE FINAL VALUE"/ *1X, 57(1H-) 11 FORMAT(14, 3X, 4(E11.5, 1X)) 12 FORMAT(14, 3X, 4(E11.5, 1X)) 12 FORMAT(14, 3X, 6(E11.5, 1X)) 13 FORMAT(14, 3X, 6(E11.5, 1X)) 14 FORMAT(14, 3X, 6(E11.5, 1X)) 15 FORMAT(14, 3X, 6(E11.5, 1X)) 15 FORMAT(14, 3X, 6(E11.5, 1X)) 15 FORMAT(14, 3X, 6(E11.5, 1X)) 16 FORMAT(14, 3X, 6(E11.5, 1X)) 17 FORMAT(14, 3X, 6(E11.5, 1X)) 18 FORMAT(14, 3X, 6(E11.5, 1X)) 19 FORMAT(14, 3X, 6(E11.5, 1X)) 19 FORMAT(14, 3X, 6(E11.5, 1X)) 10 FORMAT(14, 3X, 6(E11.5, 1X)) 10 FORMAT(14, 3X, 6(E11.5, 1X)) 10 FORMAT(14, 3X, 6(E11.5, 1X)) 10 FORMAT(14, 3X, 6(E11.5, 1X)) 12 FORMAT(14, 3X, 6(E11.5, 1X)) 13 FORMAT(14, 3X, 6(E11.5, 1X)) 14 FORMAT(14, 3X, 6(E11.5, 1X)) 15 FORMAT(14, 3X, 6(E11.5, 1X)) 15 FORMAT(14, 3X, 6(E11.5, 1X)) 16 FORMAT(14, 3X, 6(E11.5, 1X)) 17 FORMAT(14, 3X, 6(E11.5, 1X)) 18 FORMAT(14, 3X, 6(E11.5, 1X)) 19 FORMAT(14, 3X, 6(E11.5, 1X)) 19 FORMAT(14, 3X, 6(E11.5, 1X)) 10 FORMAT(14, 3X, 6(E11.5, 1X)) 10 FORMAT(14, 3X, 6(E11.5, 1X)) 11 FORMAT(14, 3X, 6(E11.5, 1X)) 12 FORMAT(14, 3X, 6(E11.5, 1X)) 13 FORMAT(14, 3X, 6(E11.5, 1X)) 14 FORMAT(14, 3X, 6(E11.5, 1X)) 15 FORMAT(14, 3X, 6(E11.5, 1X)) 15 FORMAT(14, 3X, 6(E11.5, 1X)) 15 FORMAT(14, 3X, 6(E11.5, 1X)) 15 FORMAT(14, 3X, 6(E11.5, 1X)) 16 FORMAT(14, 3X, 6(E11.5, 1X)) 17 FORMAT(14, 3X, 6(E11.5, 1X)) 17 FORMAT(14, 3X, 6(E11.5, 1X)) 18 FORMAT(14, 3X, 6(E11.5, 1X)) 19 FORMAT(14,	
ST FORMAT(/	
**** ENTER BIO UK BIT IF YOU WANT A NEW RUN WITH DIFFERENT"/	
t" ttt OTHERWISE TYPE -OK- TO CONTINUE"//)	
0 FORMAT(/ " THE MINIMUM FOUND BY THE MINIMIZATION", *" PROCESS APPEARS"/" TO BE CONSTRAINED BY THE ",A5, *" LIMIT OF PARAMETER NUMBER ",I2) 1 FORMAT(/ * *** ENTER B10 OR B11 IF YOU WANT A NEW RUN WITH DIFFERENT"/ *" *** ENTER LIMITS OR DIFFERENT PARAMETER ESTIMATES"/ *" *** OTHERWISE TYPE -OK- TO CONTINUE"//) 4 FORMAT(// *" RES. SUM OF SQUARES",4X,E12.6,5X," REG. SUM OF SQUARES",5X, *E12.6/	
<pre>FORMAT(// " RES. SUM OF SQUARES",4X,E12.6,5X," REG. SUM OF SQUARES",5X, *E12.6/ * TOT. SUM OF SQUARES",4X,E12.6,5X," SUM OF SQUARED RESPONSE ", *E12.6/ * HEAN OF RESPONSE 7,7X,E12.6,5X," RES. MEAN SQUARE 7,7X,E12.6/ * HEAN OF RESPONSE 7,7X,E12.6,5X," RES. MEAN SQUARE 7,7X,E12.6/</pre>	
TOT SUM OF COMMENT OF FAR & EV T CUM OF COMMENT PERSONNEL	
E12 6/	
* MEAN OF RESPONSE",7X,E12.6,5X," RES. MEAN SQUARE ",7X,E12.6/ * "REG. MEAN SQUARE",7X,E12.6,5X," MEAN OF RESIDUALS",7X,E12.6/ * STD. DEV OF RESIDUALS ",E12.6,5X," PCT. RESPONSE EXPLAINED ", *E12.6/" CORR. CZEFFICIENT R ",2X,E12.6/ FORMATCI4.6X,2E13.4	
*" REG. MEAN SQUARE", 7X, E12.6, 5X, " MEAN OF RESIDUALS", 7X, E12.6/	
* STD. DEV. OF RESIDUALS ",E12.6,SX," PCT. RESPONSE EXPLAINED ",	10 m
*E12.6/ CORK. COEFFICIENT R ",2X,E12.6/)
FORMAT(1Y 38(1H-))	
1 FORMAT(/ " RELATIVE POSITION OF THE CALCULATED PARAMETER".	
* IN ITS ALLOWED RANGE. /)	
5 FORMAT(14,6X,2E13.4) 0 FORMAT(1X,38/1H-)) 1 FORMAT(/ "RELATIVE POSITION OF THE CALCULATED PARAMETER", * IN ITS ALLOWED RANGE."/) 2 FORMAT(/ "RELATIVE POSITIONS OF THE CALCULATED PARAMETERS", * IN THEIR ALLOWED RANGE."/) 3 FORMAT(/ "RELATIVE POSITIONS OF THE CALCULATED PARAMETERS", 4 FORMAT(/ "RELATIVE POSITIONS OF THE CALCULATED PARAMETERS", 4 FORMAT(/ "RESIDUAL SUM OF SOURCES = ",E12.6/) 5 FORMAT(/ "RESIDUAL SUM OF SOURCES = ",E12.6/) 6 FORMAT(4 FORMAT() 5 FORMAT()	
3 FORMAT(PAR.NO. , 12, 3X, 65A1)	
4 FURMAT(/ RESIDUAL SUM OF SULARES = ,E12.6/)	
6 FORMAT(
S FORMAT() WEIGHTED RESIDUAL SUM OF SQUARES =",E12.6/) 6 FORMAT(* MGT, RES. SUM OF SQRES ",E12.6,5X," NGT. RES. MEAN SQUARE",3X,	3
*" MGT. RES. SUM OF SQRES ",E12.6,5X," MGT. RES. MEAN SQUARE",3X, *E12.6/ *" MEAN OF WGT. RESIDUALS ",E12.6,5X," STD. DEV. OF WGT. RES.",22 *Li2.6/) 7 FORMAT("THE RESIDUAL SUM OF SQUARES FOR RESPONSE SYSTEM NO." * 12" IS ",E12.6) * ECOMMAT("THE RESIDUAL SUM OF SQUARES FOR RESPONSE SYSTEM NO."	
MEAN OF WGT. RESIDUALS ",E12.6,5X," STD. DEV. OF WGT. RES.",2)	(,
7 FORMAT(" THE RESIDUAL SUM OF SOUARES FOR RESOURCE EVETEN NO. "	
*, I2, " IS ", E12.6)	
<pre>/ FORMAT(1HE RESIDUAL SUM OF SQUARES FOR RESPONSE SYSTEM NO." *,I2, 15 ",EI2.6) 8 FORMAT(" THE WEIGHTED RESIDUAL SUM OF SQUARES FOR RESPONSE", * SYSTEM NO.", EI2." IS', EI2.6) 0 FORMAT("THE REGRESSION IS NOT SIGNIFICANT") 2 FORMAT(/THE REGRESSION IS NOT SIGNIFICANT") 2 FORMAT(/- OBS.NO. X1',9%, "X2 RESPONSE CALC.RESPONSE DIFF." * DIFF(PCT) NORN.DEVIATE" /1X,77(1H-)) 3 FORMAT(/28X," RESPONSE SYSTEM NO.",I2/) 4 FORMAT(14.3X,EI0.4.10X,5EI0.4.)</pre>	
FORMAT(" THE REGRESSION IS NOT SIGNIFICANT")	
FORMAT(/" OBS.NO. X1",9X, "X2 RESPONSE CALC.RESPONSE DIFF."	
* DIFF(PCT) NORM.DEVIATE /1X,77(1H-))	
5 FORMAT(/28X, RESPONSE SYSTEM NO. , 12/)	
FORMAT(14,3X,E10.4,10X,5E10.4) FORMAT(14, 3X,7E10.4)	
<pre>* DIF(PL) NOR.DEVIALE /1X,//(H-)) \$ FORMAT(28X, "RESPONSE SYSTEM NO.",12/) \$ FORMAT(14,3X,E10.4,10X,5E10.4) \$ FORMAT(14,3X,7E10.4) \$ FORMAT(14,3X,7E10.4) \$ FORMAT(1,000,7(" X",I1,4X)/1X,77(1H-)) \$ FORMAT(1X,77(1H-)) \$ FORMAT(1X,77(1H-)) \$ FORMAT(1X,77(1H-)) </pre>	
FORMAT(/ OBS.NO",7(" X",11,4X)/1X,77(1H-)) FORMAT(1X,77(1H-)/) FORMAT(15,2X,7E10.4)	
FORMAT(15,2X,7E10.4) FORMAT(ASSUMING A NORMAL DISTRIBUTION OF ERRORS,THE UNIT NORMA	
LINNAL ASSUMING A NORMAL DISTRIPTION OF EDDODE THE HALT NORMA	

819 FORMAT(15,2X,7E10.4)
820 FORMAT(" ASSUMING A NORMAL DISTRIBUTION OF ERRORS,THE UNIT NORMAL" 13620
*," DEVIATE FORM"/" OF THE RESIDUALS TOGETHER WITH THE T-DIS", 13630

HOZ

<pre>* "TRIBUTION WITH",13," DEGREES OF FREEDOM"/" INDICATES THAT") 822 FORMAT(20X," OBSERVATION NUMBER ",13)</pre>	13640
824 FORMAT(/ " IS AN OUTLIER (P.LT.0.05) "/)	13660
8240 FORMAT(/21X," RESPONSE SYSTEM NO.", IZ/)	13670
8241 FORMAT(/" OBS.NO. WEIGHT NORM.WEIGHT WEIGHTED RESIDUAL",	13680
* UNIT NORMAL DEVIATE /47X. (CRIT.VALUES=+-".F4.2.") //	13690
*1X.67(1H-))	13700
8242 FORMAT(14,A2,2E12,4,2E15,4)	13710
8243 FORMAT(14,2X,2E12,4,2E15,4)	13720
8244 FORMAT(1X,67(1H-)/)	13730
\$245 FORMAT(THE ORSERVATION MARKED + IN THE TABLE AROVE"/	13740
* APPEARS TO BE AN OUTLIER "/)	13750
<pre>824 FORMAT(/ ' IS AN OUTLIER (P.LT.0.05) '/) 8240 FORMAT(/21X, " RESPONSE SYSTEM NO.",IZ/) 8241 FORMAT(/' OBS.NO. WEIGHT NORM.WEIGHT WEIGHTED RESIDUAL", * "UNIT NORMAL DEVIATE '/47X," (CRIT.VALUES=+-",F4.2,")"/ *1X,67(1H-) 8242 FORMAT(14,A2,2E12.4,2E15.4) 8244 FORMAT(14,2X,2E12.4,2E15.4) 8245 FORMAT(14,67(1H-)/) 8245 FORMAT(* THE OBSERVATION MAXKED * IN THE TABLE ABOVE"/ * "APPEARS TO BE AN OUTLIER."/) 8246 FORMAT(* THE OBSERVATIONS MARKED * IN THE TABLE ABOVE"/ * "APPEART TO BE OUTLIERS."/) 8266 FORMAT(* ARE OUTLIERS."/) 8266 FORMAT(* ARE OUTLIERS."/)</pre>	13760
*" ADDEAD TO BE OUTLIEDS "/)	17770
826 FORMAT(/" ARE OUTLIERS (P.LT.0.05)"/)	13780
\$761 EODWAT THE EITTED CUDVE CAN BE DI OTTED EDOW FOLLOUTING 51 DOTA	TC" 17700
1/2/1 DT NO Y DECOMPETATIVE SUTHEND FOLLOWING ST FOR	13 13/90
*/72(" PT.NO. X", 8X," RESPONSE)/1X,58(1H-)) 8262 FORMAT(2(14,3X,2E11.4)) 8263 FORMAT(1X,58(1H-)) 8264 FORMAT(/ /18X," RESPONSE SYSTEM NO.",12/) 827 FORMAT(/ " VARIANCE-COVARIANCE MATRIX OF THE PARAMETERS." /) 8270 FORMAT(/ " CORRELATION MATRIX "/) 8200 FORMAT(/ CORRELATION MATRIX "/)	13800
	13010
1203 FURNATION, JOURT / DECROMES CVETCH NO T 12/)	13820
204 FURNAL (/ 10A, RESPUNSE STATER NU	13830
227 FORMAT(/ VARIANCE-LUVARIANCE HAIRIX OF THE PARAMETERS. /)	13840
62/0 FORMAT(/ LORRELATION MATRIX /)	13850
629 FURMAT(1X,0E13.6)	13860
<pre>829 FORMAT(1X,6E13.6) 830 FORMAT(/" DURBIN WATSON STATISTIC FOR SERIAL CORRELATION OF", " RESIDUALS = ",E10.4/" (REF. J.DURBIN AND G.S.WATSON," " BIOMETRICA,PP 157-178,1951)"/) 832 FORMAT(" THE PROBABILITY OF RANDOMNESS OF RUNS =",E12.4/ " (REF. F.S.SWED ANC C.EISENHART, ANN. OF MATH. STATS.,14,", " "DEAL \$2" 453" (14)"</pre>	13870
RESIDUALS = ,E10.47 (REF. J.DURBIN AND G.S.WATSON,	13880
* BIUMETRICA, PP 157-178, 1951) /)	13890
832 FORMATC THE PROBABILITY OF RANDOMNESS OF RUNS = ",E12.4/	13900
* (REF. F.S.SWED AND C.EISENHART, ANN. OF MATH. STATS., 14, ",	13910
8320 FORMAT(ASSUMING EQUAL PROBABILITY (0.5) OF GETTING NEGATIVE A	ND 13930
*, PUSITIVE / RESIDUALS, THE PROBABILITY OF GETTING .12.	13940
*" OR LESS RESIDUALS OF SAME SIGN"/" OF A TOTAL OF .13.	13950

",4X,E12.6,5X," REG. SUM O' SQUARES",5X,	13270
",4X,E12.6,5X," SUM UF SQUARED RESPONSE ",	13280
X,E12.6,5X," RES. MEAN SOUARE ",7X,E12.6/ X,E12.6,5X," MEAN OF RESIDUALS",7X,E12.6/ ALS ",E12.6,5X," PCT. RESPONSE EXPLAINED ", IENT R ",2X,E12.6/)	13300 13310 13320 13330 13340
OSITION OF THE CALCULATED PARAMETER", E."/)	13350 13360 13370 13380
OSITIONS OF THE CALCULATED PARAMETERS", NGE."/) ,3X,65A1) M OF SQUARES =",E12.6/)	13390 13400 13410 13420
SIDUAL SUM OF SQUARES =",E12.6/)	13430
RES ",E12.6,5X," HGT. RES. MEAN SQUARE",3X,	13440 13450
ALS ",E12.6,5X," STD. DEV. OF MGT. RES.",2X,	13460 13470
L SUM OF SQUARES FOR RESPONSE SYSTEM NO."	13480 13490
	13500

22230	
22250	
22260	
22280	
22290	
22310	
	22230 22240 22250 22260 22270 22280 22290 22300 22310

G03

IF(A(I).EQ.BLANK) GO TO 65 WRITE(6,215)N GO TO 5	22320 22330 22340
70 K=0	22350
71 Î=I+1	22360
K=K+1	22370
D0 75 L=1,10	22380
IF(A(I).EQ.AL(L)) GO TO 85	22390
75 CONTINUE	22400
IF(A(I).EQ.BLANK.OR.A(I).EQ.COMMA) GO TO 60	22410
IF(A(1).E0.E) GO TO 90	22420
	22430
85 AR(J)=AR(J)+FAC*FLOAT(L-1)*(10.**(-K)) G0 T0 71	22440 22450
90 KK=0	22450
NEXP=0	22470
NFAC=1	22480
NCOUNT=0	22490
GO TO 100	22500
95 NFAC=-1	22510
00 I=I+1	22520
IF(1.EQ.79) GO TO 126	22530
NCOUNT=NCOUNT+1	22540
IF(NCOUNT.EQ.5) GO TO 126	22550
IF(NCOUNT.GT.1) GO TO 105	22560
IF(A(I).EQ.BLANK.OR.A(I).EQ.PLUS) GO TO 100	22570
IF(A(I).EQ.AMINUS) GO TO 95 05 CONTINUE	22580 22590
DO 110 L=1,10	22600
IF(A(1).E0.AL(L)) GO TO 115	22610
10 CONTINUE	22620
IF(KK.EQ.0) GO TO 25	22630
IF(A(1).EQ.BLANK.OR.A(1).EQ.COMMA) GO TO 120	22640
GO TO 25	22650
15 IF(KK.EQ.1) NEXP-10*NEXP	22660
NEXP=NEXP+NFAC*(L-1)	22670
KK=KK+1	22680
GO TO 100	22690
20 AR(J)=AR(J)*(10.**(NEXP)) Gũ TO 60	22700
25 WRITE(6,210)N	22710 22720
GO TO 5	22730
26 WRITE(6,220)	22740
GO TO 5	22750
30 AR(1)=3HYES	22760
RETURN	22770
35 AR(1)=2HNO	22780
RETURN	22790
39 SKIP=.TRUE.	22800
40 AR(1)=4HBACK J1=I+1	22810
J2=1+1 J2=1+2	22820
IF(A(I+3).NE.1HK) GO TO 141	22830 22840
J1=I+4	22150
J2=1+5	22860
41 N1=11	22870
N2=11	22880
00 142 I=1,10	22890
IF(A(J1).EQ.AL(I)) N1=I-1	22900
42 IF(A(J2).E0.AL(I)) N2=I-1	22910
IF(N1.E0.11.OR.N2.E0.11.AND.A(J2).NE.BLANK.OR.	22920
*N1.EQ.O.AND.N2.EQ.O) RETURN	22920 22930

H03

	NC-NO	
	NS=NO	22940
	NQ=10+N1+N2	22950
	IF(A(J2).EQ.BLANK) NQ=N1	22960
	1G0=2	22970
	IF(SKIP.AND.NO.GE.1.AND.NQ.LE.22) RETURN	22980
	IF(NO.GE.1.AND.NO.LE.NS) RETURN	22990
	NQ=NS	23000
	HRITE(6,230)	23010
	GO TO 5	23020
14	5 AR(1)=ZHOK	23030
	RETURN	23040
20	0 FORMAT(78A1)	
		23050
24	D5 FORMAT(/ INPUT ERROR. UNRECOGNIZED CHARACTER (,A1,) /	23060
	* REENTER LAST LINE IN CORRECT FORM. "//)	23070
21	IO FORMAT(/ INPUT ERROR. TOO FEW NUMBERS ON LINE ABOVE. ,12,	23080
	*" ARE EXPECTED. "/" REENTER LAST LINE IN CORRECT FORM ACCORDING",	23090
	*" TO REQUEST."//)	23100
21	15 FORMAT(/" INPUT ERROR. TOO MANY NUMBERS ON LINE ABOVE.", 12,	23110
	* EXPECTED. / REENTER LAST LINE IN CORRECT FORM ACCORDING",	23120
	• TO REQUEST."//)	23130
22	20 FORMAT(/" INPUT ERRORWRONG EXPONENT-"/	23140
	*" REENTER LAST LINE IN CORRECT FORM ACCORDING TO REQUEST."//)	23150
22	5 FORMAT(/" INPUT ERROR, YOU FORGOT TO ENTER DATATRY AGAIN-" /)	23160
23	O FORMAT(/" INPUT ERROR, ILLIGAL TRANSFER COMMAND - TRY AGAIN -"/)	23170
576.5	END	23180
C	***************************************	23190
-	SUBROUTINE PLACE(F1, F2, F, NP, NOP, A)	23200
	DIMENSION A(650), F1(20), F2(20), F(20)	23210
	DATA DOT, X, BLANK /1H-, 1HX, 1H /	23220
	DO 20 J=1,NOP	23230
	N1=NP+(J-1)+1	
	N2=N1+NP-1	23240
		23250

H01

IF(ANS.EQ.4HBACK) GO TO 1004	4340
IF(ANS.EQ.2HOK) GO TO 143	4350
WRITE(6,1027) GO TO 1420	4360 4370
143 F(1)=F1(1)+0.5*(F2(1)-F1(1))	4370
FSAVE(I)=F(I)	4390
144 CONTINUE NEWNOP=NOP	4400
DO 150 I=1.NOP	4410 4420
IF(ANS12.E0.2HNO) GO TO 148	4430
D0 145 J=1,N12 K=NT(J)	4440 4450
IF(K.EQ.1) GO TO 146	4460
45 CONTINUE	4470
GO TO 143 46 IF((STPS(I)+F(I)),LT,F1(I),OR.(STPS(I)+F(I)),GT,F2(I)) GO TO 147	4480 4490
STEP(1)=STPS(1)	4500
GO TO 149	4510
47 WRITE(6,1031)I CALL READ(1,AR)	4520 4530
IF(IGO.EQ.2) GO TO 1001	4540
IF(ANS.EQ.4HBACK) GO TO 1004	4550
IF(ANS.EQ.2HOK) GO TO 148 WRITE(6,1027)	4560 4570
GO TO 147	4580
48 DEL1=F(I)-F1(I)	4590
DEL2=F2(I)-F(I) IF(DEL1.GT.DEL2) STEP(I)=-(0.3+0.1/FLOAT(I))+DEL1	4600 4610
IF(DEL1.LE.DEL2) STEP(1)=(0.3+0.1/FLOAT(1))+DEL2	4620
STPS(1)=STEP(1)	4630
49 STEPS(J)=STEP(1) 17(F1(1).EQ.F2(1)) STEP(1)=0.	4640 4650
IF(F1(I).EQ.F2(I)) NEWNOP=NEWNOP-1	4660
50 CONTINUE	4670
IF(NEWNUP.GT.NOBS) GO TO 1011 NOWGHT=0	4680 4690
IF(ANS16.NE.3HYES) GO TO 1525	4700
IF(IWGHT.NE.1) GO TO 1510 N=0	4710
N-U NDUMMY=0	4720 4730
DO 1500 I=1,NOBS	4740
IF(WY(I).E0.0.) NDUMMY=NDUMMY+1 00 IF(WY(I).E0.0OR.WY(I).E0.1.) N=N+1	4750
IF(NDUMMY.EQ.O.OR.N.NE.NOBS) GO TO 1510	4760 4770
NOWGHT=1	4780
DO 1505 I=1,NOBS 05 WYN(I)=WY(I)	4790 4800
GO TO 155	4810
10 SUMW=0.	4820
DO 1515 I=1,NOBS 15 SUMW=SUMW+WY(I)	4830 4840
FACTOR=FLOAT(NOBS-NDUMMY)/SUMW	4850
DO 1520 I=1,NOBS	4860
20 WYN(I)=FACTÓR+WY(I) GO TO 155	4870 4880
25 DO 1530 I=1,NOBS	4890
WY(I)=1.	4900
30 WYN(1)=1. 55 CONTINUE	4910 4920
57 WRITE(6,662)	4930
CALL READ(1, AR)	4940
IF(IGO.EG.2) GO TO 1001	4950

	IF(ANS.EQ.4HBACK) GO TO 1004 IF(ANS.EO.3HYES) GO TO 300 IF(ANS.EO.2HNO) GO TO 159 WRITE(6,1020) GO TO 157 IWRITE(6,664) CALL BEAC1 AB	4960	
	IF (ANS ED 3HYES) GO TO 300	4970	
	IF (ANS ED 2HND) GO TO 159	4980	
	HRITE(6, 1020)	4990	
	G0 T0 157	5000	
158	HRITE(6 664)	5010	
	CALL READ(1,AR) IF(IGO.EO.2) GO TO 1001 IF(ANS.EO.4HBACK) GO TO 1004 IF(ANS.EO.2HOK) GO TO 159 WRITE(6,1027) GO TO 158	5020	
	LE(160 E0 2) 60 TO 1001	5030	
	LE CANS ED CHRACK) SO TO 1004	5040	
	IE (ANS EO 2HOK) CO TO 150	5050	
	HRITE(A 1027)	5060	
	GO TO 158	5070	
150	IF(NRUN_EQ.0.AND.MAX.NE.1) WRITE(6,670)	5080	
1.54	GO TO 350	5090	
160	NO=19	5100	and the second
100	NEWNOP=NOP	5110	
		5120	
1400	16/60/1-1,NOP-1	5130	
1000		5140	
	IF (NEW PO 1) CO TO 142	5150	
		5160	
		5170	
		5180	
161		5190	
167		5700	
102		5200 5210	
163	CALL DEAD(1 AD)	5270	
105	16/160 60 20 00 1001	5220 5230	
	16/14/9 E9.27 40 10 10 10 1004	5250	
	NOCHT-1 VIANCA 10 10 1004	5240	
	D0 1600 1=1,NOP 1F(F1(1):E0.F2(1)) NEWNOP=NEWNOP-1 IF(NEWNOP.E0.0.0R.NOP.GT.7) G0 T0 :666 IF(MAX.E0.1) G0 T0 162 MRITE(6,6700) NEWNOP CALL READ(1,AR) G0 T0 (1000,1001) IG0 IF(ANS.E0.2HNO) G0 T0 1666 IF(NEWNOP.LT.3) G0 T0 1666 IF(NEWNOP.LT.3) G0 T0 1665 MRITE(6,6705) NRECOM(NEWNOP) CALL READ(1,AR) F(16).E0.2) G0 T0 1001 IF(ANS.E0.4HBACK) G0 T0 1004 NPO[NT=1,IX(ANS+1) IF(NPO!NT=1,IX(ANS+1) IF(NPO	5250	
	GO TO 165	5260	
164	UPITE (A 11-1-1	5270 5280	The second contract of the
104	GO TO 163	5290	
	do 10 195	3270	

814 FORMAT(14,3X,E10.4,10X,5E10.4)
816 FORMAT(14, 3X,7E10.4)
817 FORMAT(/" 085.NO",7(" X",11,4X)/1X,77(1H-))
818 FORMAT(1X,77(1H-)/)
819 FORMAT(15,2X,7E10,4)
820 FORMAT(" ASSUMING A NORMAL DISTRIBUTION OF ERRORS, THE UNIT NORMAL"
*," DEVIATE FORM / OF THE RESIDUALS TOGETHER WITH THE T-DIS".
· · · · · · · · · · · · · · · · · · ·

HOZ

	State 12
* TRIBUTION WITH", 13," DEGREES OF FREEDOM"/" INDICATES THAT")	13640
822 FORMAT(20X," OBSERVATION NUMBER ".13)	13650
824 EODWAT(/ " IS AN OUTLIED (D IT 0 05) "/)	17440
BEA FORMAT (/24Y = DECOUNCE CVCTER NO = 12/)	13000
<pre>* TRIBUTION WITH",13, DEGREES OF FREEDOM"/" INDICATES THAT") 822 FORMAT(20X, "OBSERVATION NUMBER ",13) 824 FORMAT(/ IS AN OUTLIER (P.LT.0.05)"/) 8240 FORMAT(/21X, "RESPONSE SYSTEM NO.",12/) 8241 FORMAT(/21X, "RESPONSE SYSTEM NO.",12/) 8241 FORMAT(4.2,212,4,215,4) 8245 FORMAT(4.2,2212,4,2215,4) 8245 FORMAT(4.2,2212,4,2215,4) 8244 FORMAT(14,22,2212,4,2215,4) 8245 FORMAT(14,22,2212,4,2215,4) 8246 FORMAT(7, THE OBSERVATION MARKED * IN THE TABLE ABOVE"/ * "APPEARS TO BE AN OUTLIER.'/) 8266 FORMAT(- THE OBSERVATION MARKED * IN THE TABLE ABOVE"/ * "APPEAR TO BE OUTLIERS.') 8264 FORMAT(7, ARE OUTLIERS.') 8265 FORMAT(2(14,3X,2211,4)) 8265 FORMAT(2(14,3X,2211,4)) 8265 FORMAT(7, S8(1H-3)) 8264 FORMAT(7, S8(1H-3)) 8264 FORMAT(7, S8(1H-3)) 8265 FORMAT(7, ARE OUTLIERS.') 8270 FORMAT(7, CORRELATION MATRIX '/) 8270 FORMAT(7, CORRELATION MATRIX '/) 8270 FORMAT(7, CORRELATION MATRIX '/) 8270 FORMAT(7, THE POBBABILITY OF RANDOMAESS OF RUNS =",E12,4/ * "RESIDUALS =",E10,4/" (REF. J,OURBIN AND G.S.WATSON," * BIOMETRICA,PP 157-178,19510"/) 832 FORMAT(7 THE PROBABILITY OF RANDOMAESS OF RUNS =",E12,4/ * "RESIDUALS = ",E10,4/" (REF. J,OURBIN AND G.S.WATSON," * BIOMETRICA,PP 157-178,19510"/) 832 FORMAT(7 THE PROBABILITY OF RANDOMAESS OF RUNS =",E12,4/ * "REF. S.SWED AND C.EISENHART, ANN. OF MATH. STATS.,14,", * "P66-87,19430" // * 330 FORMAT(7 THE PROBABILITY OF RANDOMAESS OF RUNS =",E12,4/ * "REF. S.SWED AND C.EISENHART, ANN. OF MATH. STATS.,14,", * "P66-87,19430" // * 330 FORMAT(7 THE PROBABILITY OF RANDOMAESS OF RUNS =",E12,4/ * "RESIDUALS IS =",E10,4/" (REF. J,OURBIN AND G.S.WATSON," * BIOMETRICA,PP 157-178,19510"/) 832 FORMAT(7 THE PROBABILITY OF RANDOMAESS OF RUNS =",E12,4/ * "RESIDUALS IS =",E10,4/" (REF. J,OURBIN AND G.S.WATSON," * BIOMETRICA,PP 157-178,19510"/) 832 FORMAT(7 THE PROBABILITY OF RANDOMAESS OF RUNS =",E12,4/ * "RE</pre>	130/0
8241 FURMATC / UBS.NU. WEIGHT NORM.WEIGHT WEIGHTED RESIDUAL,	13680
*" UNIT NORMAL DEVIATE"/47X." (CRIT.VALUES=+-",F4.2.")"/	13690
+1X.67(1H-))	13700
\$242 EODWAT(14 A2 2E12 4 2E15 4)	17710
	13/10
6243 FURHAI(14,2X,2E12.4,2E13.4)	13/20
8244 FORMAT(1X,67(1H-)/)	13730
8245 FORMAT(" THE OBSERVATION MANKED * IN THE TABLE ABOVE"/	13740
*" ADDEADS TO BE AN OUTLIED "/)	13750
12/4 EOPATY "TUE OPERVATIONS HADKED + IN THE TADLE ADOVE"/	17760
6246 FURNALL THE UBSERVATIONS HARKED TIN THE TABLE ABOVE /	13/60
* APPEAR TO BE OUTLIERS. 7)	15770
826 FORMAT(/" ARE OUTLIERS (P.LT.0.05)"/)	13780
\$261 FORMAT(" THE FITTED CURVE CAN BE PLOTTED FROM FOLLOWING S1 POINTS"	13790
·//2/" DT NO V" V " DECONNEE")/1V 50/14-1)	17900
2/2 FI.NU. A , 04, RESPONSE // 14, 36(IF-7)	13800
8262 FURHA: (2(14, 3X, 2E11.4))	15810
8263 FORMAT(1X,58(1H-))	13820
8264 FORMAT(/18X." RESPONSE SYSTEM NO.", 12/)	13830
\$27 FORMAT(/ " VARIANCE-COVARIANCE MATRIX OF THE DADAMETERS " /)	13840
127 FORMATIVE CONCLUSION ATTACK BATATA OF THE FARMETERS. //	13040
6270 FURHATCY LURRELATION HATRIX /)	13850
829 FORMAT(1X,6E13.6)	13860
830 FORMAT(/" DURBIN WATSON STATISTIC FOR SERIAL CORRELATION OF".	13870
*" RESIDUALS =" E10 4/" (REE DURRIN AND C S HATSON "	13880
- DIDUCTO - DO 167 17 1001 - DO NOT AND G.S. MATSON,	13860
BIUHEIRICA, PP 13/-1/8, 1931) /)	13890
832 FURMATC THE PROBABILITY OF RANDOMNESS OF RUNS = ",E12.4/	13900
*" (REF. S.SHED AND C.EISENHART, ANN, OF MATH, STATS., 14,",	13910
*"PP66-87 1943)" /)	13920
\$720 FORMATC" ACCUMUNC FOUNI DOODADILITY (A E) OF CETTING NECATIVE AND	17070
6320 FORMATC ASSUMING EQUAL PROBABILITY (0.57 OF GETTING NEGATIVE AND	13430
*, POSITIVE / RESIDUALS, THE PROBABILITY OF GETTING ",12,	13940
*" OR LESS RESIDUALS OF SAME SIGN"/" OF A TOTAL OF". 13.	13950
*" RESIDIALS IS =" E11 4 /)	13960
\$77 FORMATION THIS INDICATES THAT THE FORODS ADD NOT DANDON AT THE	17070
633 FURNALL INIS INUICA S THAT THE ERRORS ARE NUT RANDUM AT THE ,	139/0
* MINIMUM FOUND /)	13980
834 FORMAT(" THE RATIO OF " E DEGREES OF FREEDOM OF RESIDUALS TO THE".	13990
*" NUMBER"/" OF RESIDUA S =" F10 4 " A SIGNIFICANT CORRELATION"	14000
. " DETUECH THE DECIDIALS" /" IS THEDEFODE EXDECTED VIOLATING THE	1/010
-, BETWEEN THE RESIDUALS / IS THEREFORE EXPECTED, VIULATING THE	14010
* ASSUMPTION BEHIND THE STATISTICAL / EVALUATION OF THE ,	14020
* PARAMETERS.)	14030
\$36 FORMAT(//8X	14040
*" DIOT OF RECIDIALS VERSUS INDEDENDENT VARIABLE VI (COL UNITO)")	1/050
FLUT OF RESIDENCES VERSUS INDEPENDENT VARIABLE AT (SLL.UNITS))	14030
8360 FURMAT(7/2X, PLUT UF WEIGHTED RESIDUALS VERSUS ,	14060
* INDEPENDENT VARIABLE X1 (SCL.UNITS))	14070
838 FORMAT(//11X.	14080
+" PLOT OF RESIDIALS VERSUS RESPONSE ESTIMATE (SCI INVITE)")	14000
The Fourt of Residence rendered broke Estimate (Ste. Units) /	14070
8380 FURHAT(7/3X, PLUT UF WEIGHTED RESILUALS ,	14100
* VERSUS RESPONSE ESTIMATE (SCL.UNIS)")	14110
840 FORMAT(//5X," PLOT OF RESIDUALS VERSUS 1ST, INDEPENDENT".	14120
* VARIARIE X1 (SCI INITS)")	14130
\$400 EOPWAT(//2X " DI OT OF HEICHTED DESIDIALS VERSUS"	1/1/0
BYOU FORMATCHIZA, FLOT OF WEIGHTED RESIDUALS VERSUS ,	14140
* IST. INDEPENDENT VARIABLE X1 (SLL.UNITS))	14150
842 FORMAT(//5X," PLOT OF RESIDUALS VERSUS 2ND. INDEPENDENT VARIABLE".	14160
*" X2 (SCL.UNITS) ")	14170
\$420 FORMAT(//2Y " PLOT OF HEIGHTED RESIDING S VERSIS"	1/180
" THO INCOMPANY AND	14180
2 CHU. HUCP'ENDENI VARIABLE X2 'SLL.UNIIS))	14190
8401 FURMAT(," CONVERGENCE CRITERION ",E10.4)	14200
8402 FORMAT(" EXPANSION CRITERION ".E12.4)	14210
\$403 FORMAT(" MINIMUM OF EITTED QUADRATIC SURFACE IS " EIS #/	14220
- CONDADE LITU MINIME COUND BY ITED TION " FIS	1/270
- CONFINE WITH HIMINUN FULLY BI TIERATION ,E13.0/	14230
<pre>833 FORMAT(" THE INDEPENDENT VARIABLE X1 (SCL.UNITS)") 834 FORMAT(" THE RATIO OF): DEGREES OF FREEDOM OF RESIDUALS TO THE", * "NUMBER"/" OF RESIDUALS =".FIO.4,". A SIGNIFICANT CORRELATION" * "BETWEEN THE RESIDUALS =".FIO.4,". A SIGNIFICANT CORRELATION" * "PLOT OF RESIDUALS VERSUS INDEPENDENT VARIABLE X1 (SCL.UNITS)") 836 FORMAT(//AX, * "PLOT OF RESIDUALS VERSUS INDEPENDENT VARIABLE X1 (SCL.UNITS)") 838 FORMAT(//11X, * "INDEPENDENT VARIABLE X1 (SCL.UNITS)") 838 FORMAT(//11X, * "PLOT OF RESIDUALS VERSUS RESPONSE ESTIMATE (SCL.UNITS)") 830 FORMAT(//7X, "PLOT OF WEIGHTED RESIDUALS VERSUS", * "VERSUS RESPONSE ESTIMATE (SCL.UNITS)") 8400 FORMAT(//5X," PLOT OF RESIDUALS VERSUS IST. INDEPENDENT, * "VARIABLE X1 (SCL.UNITS)") 8400 FORMAT(//2X, "PLOT OF RESIDUALS VERSUS SIST. INDEPENDENT VARIABLE X1 (SCL.UNITS)") 8400 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "IST. INDEPENDENT VARIABLE X1 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT('/2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "ZD. INDEPENDENT VARIABLE X2 (SCL.UNITS)") 8420 FORMAT('/2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "ZD. INDEPENDENT VARIABLE X2 (SCL.UNITS)") 8420 FORMAT('/2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "ZD. INDEP</pre>	14240
A DADAMETERS TO"/" THACCUDATE AND A NEU DUN ULTU A DIFFERRENT"	
* PARATELERS IS / INALLUKALE AND A NEW RUN WITH A DIFFERENT .	14250

	1/2/0
* STOPPING CRITERION, EXPANSION"/" CRITERION OR DIFFERENT RANGE", * OF THE PARAMETERS IS RECOMMENDED.")	14260 14270
8404 FORMAT(/ TYPE BACK (B**) IF YOU WANT TO MAKE CORRECTIONS OR .	14280
*" OK IF YOU WANT"/" TO CONTINUE BUT WITHOUT STATISTICAL",	14290
*" EVALUATION OF THE PARAMETERS"//)	14300
8422 FORMAT(" NO. OF SS EVALUATIONS ",14,17X, * NO. OF CONSTR. VIOLATIONS ",14)	14310
* NO. OF CONSTR. VIOLATIONS ",14)	14320
8424 FORMAT(" THE QUADRATIC SURFACE FITTING REQUIRED A FURTHER", 14,	14330
*" SS EVALUATIONS")	14340
8425 FORMAT(/1H)	14350
844 FORMAT(14360
//1X,27(1H)," END OF FITTING NO.",12,2X,26(1H*) //) 1000 IF(ANS.EQ.3HYES.OR.ANS.EQ.2HNO) GO TO 1002	14370 14380
IF(ANS.EQ.4HBACK) GO TO 1004	14390
WRITE(6,1020)	14400
1001 LE(SKIP) NS=NO	14410
GO TO (5,10,15,20,25,30,35,40,45,50,55,70,85,90,95,100,105,	14420
G0 T0 (5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 70, 85, 90, 95, 100, 105, *130, 160, 170, 175, 185 1002 G0 T0 (6, 11, 16, 21, 26, 31, 36, 41, 46, 51, 56, 71, 86, 91, 96, 101, 105, *130, 161, 171, 175, 185, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10	14430
1002 GO TO (6,11,16,21,26,31,36,41,46,51,56,71,86,91,96,101,105,	14440
*130,161,171,175,185)NO	14450
1003 IF (ANS.EG. 40BACK) GU 10 1004	14460
GO TO 1002 1004 NS=NQ	14470
WRITE(6,1021)	14480 14490
SKIPS=SKIP	14500
CALL READ(1,AR)	14510
IF(ANS.EQ.4HBACK) GO TO 1004	14520
SK IP=SK IPS	14530
NO=IFIX(ANS+.1)	14540
IF(SKIP.AND.NO.GE.1.AND.NO.LE.22) GO TO 1001	14550
D0 1005 I=1,NS	14560
IF(I.EO.NO) GO TO 1001	14570
1005 CONTINUE WRITE(6,1022)	14580
MITTERO, 10227	14590

814 FORMAT(14,3X,E10.4,10X,5E10.4)
816 FORMAT(14, 3X,7E10.4)
817 FORMAT(/" 085.NO",7(" X",11,4X)/1X,77(1H-))
818 FORMAT(1X,77(1H-)/)
819 FORMAT(15,2X,7E10,4)
820 FORMAT(" ASSUMING A NORMAL DISTRIBUTION OF ERRORS, THE UNIT NORMAL"
*," DEVIATE FORM / OF THE RESIDUALS TOGETHER WITH THE T-DIS".
· · · · · · · · · · · · · · · · · · ·

HOZ

	State 12
* TRIBUTION WITH", 13," DEGREES OF FREEDOM"/" INDICATES THAT")	13640
822 FORMAT(20X," OBSERVATION NUMBER ".13)	13650
824 EODWAT(/ " IS AN OUTLIED (D IT 0 05) "/)	17440
BEA FORMAT (/24Y = DECOUNCE CVCTER NO = 12/)	13000
<pre>* TRIBUTION WITH",13, DEGREES OF FREEDOM"/" INDICATES THAT") 822 FORMAT(20X, "OBSERVATION NUMBER ",13) 824 FORMAT(/ 'IS AN OUTLIER (P.LT.0.05)"/) 8240 FORMAT(/21X, "RESPONSE SYSTEM NO.",12/) 8241 FORMAT(/21X, "RESPONSE SYSTEM NO.",12/) 8241 FORMAT(4.2,212.4,215.4) 8245 FORMAT(4.2,2212.4,2215.4) 8245 FORMAT(4.2,2212.4,2215.4) 8244 FORMAT(14.42,2212.4,2215.4) 8244 FORMAT(14.42,2212.4,2215.4) 8245 FORMAT(14.42,2212.4,2215.4) 8244 FORMAT(14.42,212.4,2215.4) 8244 FORMAT(14.67(1H-7)) 8246 FORMAT(-THE OBSERVATION MARKED * IN THE TABLE ABOVE"/ * "APPEARS TO BE AN OUTLIER.'/) 8264 FORMAT(-THE OBSERVATION MARKED * IN THE TABLE ABOVE"/ * "APPEAR TO BE OUTLIERS.') 8264 FORMAT(-THE OBSERVATION MARKED * IN THE TABLE ABOVE"/ * "APPEAR TO BE OUTLIERS.') 8264 FORMAT(-THE OBSERVATION MARKED * IN THE TABLE ABOVE"/ */2(" PT.NO. X'.8X," RESPONSE")/1X,58(1H-)) 8265 FORMAT(2(14,3X,2211.4)) 8265 FORMAT(/14X,58(1H-)) 8264 FORMAT(/14X,58(1H-)) 8265 FORMAT(/ 18X, "RESPONSE SYSTEM NO.",12/) 8276 FORMAT(/14X,58(1H-)) 8264 FORMAT(/ 18X, "RESPONSE SYSTEM NO.",12/) 8276 FORMAT(/ CORRELATION MATRIX ')) 8276 FORMAT(/ 18X, "RESPONSE SYSTEM NO.",12/) 8276 FORMAT(/ CORRELATION MATRIX ')) 8276 FORMAT(/ THE PROBABILITY OF RANDOMNESS OF RUNS =",E12.4/ * "RESIDUALS = ",E10.4/" (REF. J.OURBIN AND G.S. MATSON," * BIOMETRICA,PP 157-178,19510'') 832 FORMAT(- THE PROBABILITY OF RANDOMNESS OF RUNS = ',E12.4/ * "REF. S.SWED AND C.EISENHART, ANN. OF MATH. STATS.,14,", * "P66-87,1943'') 833 FORMAT(- THE ROBABILITY OF RANDOMNESS OF RUNS = ',E12.4/ * "OR LESS RESIDUALS OF SAME SIGN'' OF A TOTAL OF",13, * "RESIDUALS IS = ,E11.4 /) 833 FORMAT(- THE ROBABILITY OF ANDONNESS OF RUNS = ',E12.4/ * "OR LESS RESIDUALS OF SAME SIGN'' OF A TOTAL OF",13, * "RESIDUALS IS = ,E11.4 /) 833 FORMAT(- THE ROBABILITY OF CANDONNESS O</pre>	130/0
8241 FURMATC / UBS.NU. WEIGHT NORM.WEIGHT WEIGHTED RESIDUAL,	13680
*" UNIT NORMAL DEVIATE"/47X." (CRIT.VALUES=+-",F4.2.")"/	13690
+1X.67(1H-))	13700
\$242 EODWAT(14 A2 2E12 4 2E15 4)	17710
	13/10
6243 FURHAI(14,2X,2E12.4,2E13.4)	13/20
8244 FORMAT(1X,67(1H-)/)	13730
8245 FORMAT(" THE OBSERVATION MANKED * IN THE TABLE ABOVE"/	13740
*" ADDEADS TO BE AN OUTLIED "/)	13750
12/4 EOPATY "TUE OPERVATIONS HADKED + IN THE TADLE ADOVE"/	17760
6246 FURNALL THE UBSERVATIONS HARKED TIN THE TABLE ABOVE /	13/60
* APPEAR TO BE OUTLIERS. 7)	15770
826 FORMAT(/" ARE OUTLIERS (P.LT.0.05)"/)	13780
\$261 FORMAT(" THE FITTED CURVE CAN BE PLOTTED FROM FOLLOWING S1 POINTS"	13790
·//2/" DT NO V" V " DECONNEE")/1V 50/14-1)	17900
2/2 FI.NU. A , 04, RESPONSE // 14, 36(IF-7)	13800
8262 FURHA: (2(14, 3X, 2E11.4))	15810
8263 FORMAT(1X,58(1H-))	13820
8264 FORMAT(/18X." RESPONSE SYSTEM NO.", 12/)	13830
\$27 FORMAT(/ " VARIANCE-COVARIANCE MATRIX OF THE DADAMETERS " /)	13840
227 FORMATIVE CONCLUSION ATTACK BATATA OF THE FARMETERS. //	13040
6270 FURHATCY LURRELATION HATRIX /)	13850
829 FORMAT(1X,6E13.6)	13860
830 FORMAT(/" DURBIN WATSON STATISTIC FOR SERIAL CORRELATION OF".	13870
*" RESIDUALS =" E10 4/" (REE DURRIN AND C S HATSON "	13880
- DIDUCTO - DO 167 17 1001 - DO NOT AND G.S. MATSON,	13860
BIUHEIRICA, PP 13/-1/8, 1931) /)	13890
832 FURMATC THE PROBABILITY OF RANDOMNESS OF RUNS = ",E12.4/	13900
*" (REF. S.SHED AND C.EISENHART, ANN, OF MATH, STATS., 14,",	13910
*"PP66-87 1943)" /)	13920
\$720 FORMATC" ACCUMUNC FOUNI DOODADILITY (A E) OF CETTING NECATIVE AND	17070
6320 FORMATC ASSUMING EQUAL PROBABILITY (0.57 OF GETTING NEGATIVE AND	13430
*, POSITIVE / RESIDUALS, THE PROBABILITY OF GETTING ",12,	13940
*" OR LESS RESIDUALS OF SAME SIGN"/" OF A TOTAL OF". 13.	13950
*" RESIDIALS IS =" E11 4 /)	13960
\$77 FORMATION THIS INDICATES THAT THE EDDODE ADD NOT DANDON AT THE	17070
633 FURNALL INIS INUICA S THAT THE ERRORS ARE NUT RANDUM AT THE ,	139/0
* MINIMUM FOUND /)	13980
834 FORMAT(" THE RATIO OF " E DEGREES OF FREEDOM OF RESIDUALS TO THE".	13990
*" NUMBER"/" OF RESIDUA S =" F10 4 " A SIGNIFICANT CORRELATION"	14000
· " DETUECH THE DECIDIALS" /" IS THEDEFODE EXDECTED VIOLATING THE	1/010
-, BETWEEN THE RESIDUALS / IS THEREFORE EXPECTED, VIULATING THE	14010
* ASSUMPTION BEHIND THE STATISTICAL / EVALUATION OF THE ,	14020
* PARAMETERS.)	14030
\$36 FORMAT(//8X	14040
*" DIOT OF RECIDIALS VERSUS INDEDENDENT VARIABLE VI (COL UNITO)")	1/050
FLUT OF RESIDENCES VERSUS INDEPENDENT VARIABLE AT (SLL.UNITS))	14030
8360 FURMAT(7/2X, PLUT UF WEIGHTED RESIDUALS VERSUS ,	14060
* INDEPENDENT VARIABLE X1 (SCL.UNITS))	14070
838 FORMAT(//11X.	14080
+" PLOT OF RESIDIALS VERSUS RESPONSE ESTIMATE (SCI INVITE)")	14000
The Fourt of Residence rendered broke Estimate (Ste. Units) /	14070
8380 FURHAT(7/3X, PLUT UF WEIGHTED RESILUALS ,	14100
* VERSUS RESPONSE ESTIMATE (SCL.UNIS)")	14110
840 FORMAT(//5X," PLOT OF RESIDUALS VERSUS 1ST, INDEPENDENT".	14120
* VARIARIE X1 (SCI INITS)")	14130
\$400 EOPWAT(//2X " DI OT OF HEICHTED DESIDIALS VERSUS"	1/1/0
BYOU FORMATCHIZA, FLOT OF WEIGHTED RESIDUALS VERSUS ,	14140
* IST. INDEPENDENT VARIABLE X1 (SLL.UNITS))	14150
842 FORMAT(//5X," PLOT OF RESIDUALS VERSUS 2ND. INDEPENDENT VARIABLE".	14160
*" X2 (SCL.UNITS) ")	14170
\$420 FORMAT(//2Y " PLOT OF HEIGHTED RESIDING S VERSIS"	1/180
" THO INCOMPANY AND	14180
2 CHU. HUCP'ENDENI VARIABLE X2 'SLL.UNIIS))	14190
8401 FURMAT(," CONVERGENCE CRITERION ",E10.4)	14200
8402 FORMAT(" EXPANSION CRITERION ".E12.4)	14210
\$403 FORMAT(" MINIMUM OF EITTED QUADRATIC SURFACE IS " EIS #/	14220
- CONDADE LITU MINIME COUND BY ITED TION " FIS	1/270
- CONFINE WITH HIMINUN FULLY BI TIERATION ,E13.0/	14230
<pre>833 FORMAT(" THE INDEPENDENT VARIABLE X1 (SCL.UNITS)") 834 FORMAT(" THE RATIO OF): DEGREES OF FREEDOM OF RESIDUALS TO THE", * "NUMBER"/" OF RESIDUALS =".FIO.4,". A SIGNIFICANT CORRELATION" * "BETWEEN THE RESIDUALS =".FIO.4,". A SIGNIFICANT CORRELATION" * "PLOT OF RESIDUALS VERSUS INDEPENDENT VARIABLE X1 (SCL.UNITS)") 836 FORMAT(//AX, * "PLOT OF RESIDUALS VERSUS INDEPENDENT VARIABLE X1 (SCL.UNITS)") 838 FORMAT(//11X, * "INDEPENDENT VARIABLE X1 (SCL.UNITS)") 838 FORMAT(//11X, * "PLOT OF RESIDUALS VERSUS RESPONSE ESTIMATE (SCL.UNITS)") 830 FORMAT(//7X, "PLOT OF WEIGHTED RESIDUALS VERSUS", * "VERSUS RESPONSE ESTIMATE (SCL.UNITS)") 8400 FORMAT(//5X," PLOT OF RESIDUALS VERSUS IST. INDEPENDENT, * "VARIABLE X1 (SCL.UNITS)") 8400 FORMAT(//2X, "PLOT OF RESIDUALS VERSUS SIST. INDEPENDENT VARIABLE X1 (SCL.UNITS)") 8400 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "IST. INDEPENDENT VARIABLE X1 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "X2 (SCL.UNITS)") 8420 FORMAT('/2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "ZD. INDEPENDENT VARIABLE X2 (SCL.UNITS)") 8420 FORMAT('/2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "ZD. INDEPENDENT VARIABLE X2 (SCL.UNITS)") 8420 FORMAT('/2X," PLOT OF WEIGHTED RESIDUALS VERSUS", * "ZD. INDEP</pre>	14240
A DADAMETERS TO"/" THACCUDATE AND A NEU DUN ULTU A DIFFERRENT"	
* PARATELERS IS / INALLUKALE AND A NEW RUN WITH A DIFFERENT .	14250

	1/2/0
* STOPPING CRITERION, EXPANSION"/" CRITERION OR DIFFERENT RANGE", * OF THE PARAMETERS IS RECOMMENDED.")	14260 14270
8404 FORMAT(/ TYPE BACK (B**) IF YOU WANT TO MAKE CORRECTIONS OR .	14280
*" OK IF YOU WANT"/" TO CONTINUE BUT WITHOUT STATISTICAL",	14290
*" EVALUATION OF THE PARAMETERS"//)	14300
8422 FORMAT(" NO. OF SS EVALUATIONS ",14,17X, * NO. OF CONSTR. VIOLATIONS ",14)	14310
* NO. OF CONSTR. VIOLATIONS ",14)	14320
8424 FORMAT(" THE QUADRATIC SURFACE FITTING REQUIRED A FURTHER", 14,	14330
*" SS EVALUATIONS")	14340
8425 FORMAT(/1H)	14350
844 FORMAT(14360
//1X,27(1H)," END OF FITTING NO.",12,2X,26(1H*) //) 1000 IF(ANS.EQ.3HYES.OR.ANS.EQ.2HNO) GO TO 1002	14370 14380
IF(ANS.EQ.4HBACK) GO TO 1004	14390
WRITE(6,1020)	14400
1001 LE(SKIP) NS=NO	14410
GO TO (5,10,15,20,25,30,35,40,45,50,55,70,85,90,95,100,105,	14420
G0 T0 (5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 70, 85, 90, 95, 100, 105, *130, 160, 170, 175, 185 1002 G0 T0 (6, 11, 16, 21, 26, 31, 36, 41, 46, 51, 56, 71, 86, 91, 96, 101, 105, *130, 161, 171, 175, 185, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10	14430
1002 GO TO (6,11,16,21,26,31,36,41,46,51,56,71,86,91,96,101,105,	14440
*130,161,171,175,185)NO	14450
1003 IF(ANS.EG.48BACK) GU 10 1004	14460
GO TO 1002 1004 NS=NQ	14470
WRITE(6,1021)	14480 14490
SKIPS=SKIP	14500
CALL READ(1,AR)	14510
IF(ANS.EQ.4HBACK) GO TO 1004	14520
SK IP=SK IPS	14530
NO=IFIX(ANS+.1)	14540
IF(SKIP.AND.NO.GE.1.AND.NO.LE.22) GO TO 1001	14550
D0 1005 I=1,NS	14560
IF(I.EO.NO) GO TO 1001	14570
1005 CONTINUE WRITE(6,1022)	14580
MITTERO, 10227	14590

NZ=11 D0 142 I=1,10 IF(A(J1).E0.AL(I)) N1=I-1 142 IF(A(J2).E0.AL(I)) NZ=I-1 IF(N1.E0.11.0R.NZ.E0.11.AND.A(J2).NE.BLANK.OR. +N1.E0.0.AND.NZ.E0.0) RETURN

H03

NS=NO	22940
NQ=10+N1+N2	22950
IF(A(JZ).EQ.BLANK) NQ=N1	22960
1G0=2	22970
IF(SKIP.AND.NQ.GE.1.AND.NQ.LE.22) RETURN	22980
IF(NQ.GE.1.AND.NQ.LE.NS) RETURN	22990
NG=NS	23000
WRITE(6,230)	23010
GO TO S	23020
145 AR(1)=2H0K	23030
RETURN	23040
200 FORMAT(78A1)	23050
205 FORMAT(/" INPUT ERROR. UNRECOGNIZED CHARACTER (",A1,")"/	23060
* REENTER LAST LINE IN CORRECT FORM. //)	23070
210 FORMAT(/" INPUT ERROR. TOO FEW NUMBERS ON LINE ABOVE. , 12,	23080
*" ARE EXPECTED. "/" REENTER LAST LINE IN CORRECT FORM ACCORDING",	23090
*" TO REQUEST."//)	23100
215 FORMAT //" INDUT EDDOD TOO MANY NUMBERS ON I THE ADOVE " 12	23110
215 FORMAT(// INPUT ERROR. TOO MANY NUMBERS ON LINE ABOVE	23120
* TO REQUEST. "//)	
220 FORMAT(/" INPUT ERRORWRONG EXPONENT-"/	23130 23140
*" REENTER LAST LINE IN CORRECT FORM ACCORDING TO REQUEST."//)	
REENTER LAST LINE IN LURRELT FURH ALLURDING TU REQUEST. (7)	23150
225 FORMAT(/" INPUT ERROR, YOU FORGOT TO ENTER DATATRY AGAIN-" /)	23160
230 FORMAT(/" INPUT ERROR, ILLIGAL TRANSFER COMMAND - TRY AGAIN -"/)	23170
END	23180
C	23190
SUBROUTINE PLACE(F1,F2,F,NP,NOP,A)	23200
DIMENSION A(650), F1(20), F2(20), F(20)	23210
DATA DOT, X, BLANK /1H-, 1HX, 1H /	23220
DO 20 J=1,NOP	23230
N1=NP*(J-1)+1	23240
N2=N1+NP-1	23250
IF(F1(J).EQ.F2(J)) GO TO 10	23260
DO 5 I=N1.N2	23270
5 A(1)=DOT	23280
NX=N1+IFIX((NP-1)*(F(J)-F1(J))/(F2(J)-F1(J))+.5)	23290
IF(NX.GT.N2.DR.NX.LT.N1) GO TO 20	23300
A(NX)=X	23310
GO TO ZO	23320
10 CONTINUE	23330
DO 15 I=N1.N2	23340
15 A(I)=BLANK	23350
20 CONTINUE	
RETURN	23360
END	23370
	23380
	23390
FUNCTION WEIGHT(Y, INGHT, R)	23400
I=IWGHT-1	23410
GO TO (5,10,15)I	23420
5 IF(Y.LT.O. OR.Y.EQ.O. AND.R.LT.O.) GO TO 20	23430
WEIGHT=Y**R	23440
RETURN	23450
10 IF(R+Y.LT.1) GO TO 20	23460
WEIGHT=ALOG(R+Y)	23470
RETURN	23480
15 WEIGHT=EXP(R*Y)	23490
RETURN	23500
20 WEIGHT=-1.	23510
RETURN	23520
END	23530
C	23540
SUBROUTINE XYPLOT(F)	23550
SUBJOUTINE ATTENTION	23330

DIMENSION X(100), Y(100), WYN(100), LX(100), LY(100), LIM1(10), LIM2(10	23560
* LLKF(75)_LYF(75);XF(75),YF(75),F(20),A(75),XX(9,100) COMMON /DATA/ XX,Y,WYN,NOBS COMMON /FUNNUM/ITHFUN	23570
CONTON /DATA/ XX, T, MTN, NUBS	23580
COMPON / FUNNOR/ LIFEON	23590
COMMON /LXLY/LX,LY	23600
COMMON /B1/ X COMMON /B4/ NIND	23610
COMMON /B6/ NOP	23620
COMMON /B8/NFUNC	23630
COMMON /B9/LIM1,LIM2	23640
DATA BLANK, POINT1, POINT2, DOT, STAR /1H ,1H+,1H0,1H.,1HX/	23650 23660
DO 45 11-1 NEUNC	23670
DO 45 II=1,NFUNC N1=LIM1(II)	23680
NZ=N1+1	23690
N3=LIM2(II)	23700
NN=N3-1	23710
XMIN=X(N1)	23720
XMAX=X(N1)	23730
YMIN=Y(N1)	23740
YMAX=Y(N1)	23750
IF(11.EQ.1) WRITE(6,100)	23760
IF(II.GT.1) WRITE(6.101)[]	23770
DO 5 1=N2,N3	23780
IF(X(I),GT,XMAX) XMAX=X(I)	23790
IF(XCI).IT.XMIND XMIN=XCI) IF(YCI).GT.YMAXX YMAX=YCI) IF(YCI).IT.YMINDYMIN=YCI)	23800
IF(Y(I).GT.YMAX) YMAX=Y(I)	23810
i IF(Y(I).LT.YMIN)YMIN=Y(I)	23820
SPANX=XMAX-XMIN	23830
IF(SPANX.EQ.0.)RETURN	23840
DELX=SPANX/74.	23850
00 10 1=1,75	23860
XF(1)=XMIN+FLOAT(1-1)+DELX	23870
DO 7 LENIAN	23880
7 IF(XF(1),GT,X(L),AND,XF(1),LE,X(L+1)) +VE(1)=V(L)+(V(L+1)-Y(L))+(VE(L)-Y(L))/(V(L+1)-V(L))	23890

FACTOR=FLOAT(NOBS-NDUMMY)/SUMM	4850	
DO 1520 I=1,NOBS 1520 WYN(I)=FACTOR+WY(I)	4860 4870	
GO TO 155	4820	
1525 DO 1530 1=1,NOBS	4880 4890	
WY(1)=1. 1530 WYN(1)=1.	4900	
155 CONTINUE	4910	
157 WRITE(6.662)	4910 4920 4930 4940	
CALL READ(1, AR)	4940	
IF(IGO.EO.2) GO TO 1001	4950	

	IF (ANS.ED.4HBACK) GO TO 1004	4960 4970
	IF(ANS.E0.3HYES) GO TO 300 IF(ANS.E0.2HNO) GO TO 159	4980
3	WRITE(6,1020)	4990
	GO TO 157	5000
130 1	WRITE(6,664) CALL READ(1 AR)	5010 5020
. 3	CALL READ(1,AR) IF(IGD.EQ.2) GO TO 1001	5030
	IF(ANS.EQ.4HBACK) GO TO 1004	5040
2	IF(ANS.EQ.2HOK) GO TO 159	5050
	WRITE(6,1027) GO TO 158	5060 5070
159	IF(NRUN.EQ.O.AND.MAX.NE.1) WRITE(6,670)	5080
	GO TO 350	5090
	NQ=19	5100
	NEWNOP=NOP D0 1600 I=1,NOP	5110 5120
600	IE(E1(I), EQ, E2(I)) NEWNOP=NEWNOP-1	5130
	IF(NEWNOP.EQ.0.0R.NOP.GT.7) GO TO 1666	5140
	IF(MAX.EQ.1) GO TO 162	5150
1	WRITE(6,6700)NEWNOP CALL READ(1,AR)	5160 5170
. 3	GO TO (1000, 1001) IGO	5180
161	GO TO (1000,1001) IGO IF(ANS.EQ.2HND) GO TO 1666	5190
162	IF(NEWNOP.LT.3) GO TO 165	5200
147	WRITE(6,6705) NRECOM(NEWNOP) CALL READ(1,AR)	5210
105	IF(IG0.E0.2) GO TO 1001	5220 5230
	IF (ANS.EC. 4HBACK) GO TO 1004	5240
5	NPOINT=I+IX(ANS+.1)	5250
	IF(NPOINT T.NCRIT(NEWNOP).OR.NPOINT.GT.2187) GO TO 164	5260
	GO TO 165 WRITE(6,10/1)	5270 5280
	GO TO 163	5290
165 1	RTTE(6,671)NEWNOP IF(MAX.EQ.1) CALL LSQ(F,FUNC1)	5300
3	IF(MAX.EQ.1) CALL LSQ(F,FUNC1)	5310
1	IF(MAX.EQ.1) FUNC=FUNC1 CALL SEARCH(FNEW,STPNEW,SSMIN,ANS16,NPOINT,NEVAL,NEWNOP,STEP,NP)	5320 5330
3	IF(NEWNOP.GT.2) WRITE(6,6715)NEVAL,NP	5340
	IF(ANS16.EQ.3HYES) WRITE(6,6720)SSMIN	5350
3	IF(ANS16.E0.2HNO) WRITE(6.6725)SSMIN	5360
	CALL PLACE(F1,F2,FNEW,40,NPR,A) D0 1650 J=1,NPR	5370 5380
	IF(F1(J).E0.F2(J)) GO TO 1650	5390
1	N1=40*(J-1)+1	5400
	N2=N1+39	5410
650	WRITE(6,6730)J,FNEW(J),(A(I),I=N1,N2) CONTINUE	5420 5430
	IF(SSMIN.GE.FUNC) GO TO 1653	5440
	IF(MAX.EQ.1) GO TO 1652	5450
1	WRITE(6,6735) FUNC	5460
	DO 1651 I=1,NOP F(I)=FNEW(I)	5470 5480
	FSAVE(I)=F(I)	5480
1	IF(F(I)+STPNEW(I).GT.F2(I)) STPNEW(I)=-STPNEW(I)	5500
651 9	STEP(I)=STPNEW(I)	5510
457	GO TO 1664 HRITE(6,6731) FUNC	5520
1 200	GO TO 1655	5530 5540
653	IF(MAX.EQ.1) GO TO 1654	5550
	WRITE(6,6740) FUNC	5560
2	CALL PLACE(F1,F2,FMIN,40,NPR,A)	5570

J01

GO TO 1656 1654 WRITE(6,6741) FUNC 1655 CALL PLACE(F1,F2,F,40,NPR,A)	5580 5590
1655 CALL PLACE(F1,F2,F,40,NPR,A)	5600
1656 CONTINUE D0 1657 J=1, NPR	5610 5620
IF(F1(J).E0.F2(J)) G0 T0 1657	5630
N1=40*(J-1)+1	5640
N2=N1+39 IF(MAX.EQ.1) WRITE(6,6730):,F(J),(A(I),I=N1,N2)	5650 5660
IF(MAX.NE.1) WRITE(6,6730) J, FMIN(J), (A(1), I=N1, N2)	5670
1657 CONTINUE	5680
WRITE(6,6745)	5690
1660 WRITE(6,6748) CALL GEAD(1,AR) IF(IGJ.E0.2) GO TO 1001 IF(ANS.E0.4HBACK) GO TO 1004 IF(ANS.E0.3HYES) GO TO 1661 IF(ANS.E0.2HNO.AND.MAX.E0.1) GO TO 1663 F(ANS.E0.2HNO.AND.MAX.E0.1) GO TO 1663	5700 5710
1F(1GJ.EG.2) GO TO 1001	5720
IF(ANS.ED.4HBACK) GO TO 1004	5730
IF (ANS EQ 2HNO AND MAX EQ 1) GO TO 1663	5740 5750
F(ANS.EQ.2HNO) GO TO 1666	5760
WRITE(6,1020) G0 T0 1660	5770
1661 ANS11=3HYES	5780 5790
ANS12=3HYES	5800
N11=NOP	5810
N12=NOP D0 1662 I=1,NOP	5820 5830
NF(1)=1	5840
NT(I)=I	5850
F(1)=FNEW(1) FS(1)=F(1)	5860

* 1SI. INDEPENDENT VARIABLE X1 (SCL.UNITS)")	14150
<pre>842 FORMAT(//5X," PLOT OF RESIDUALS VERSUS 2ND. INDEPENDENT VARIABLE", * X2 (SCL.UNITS) ")</pre>	14160
8420 FORMAT(//2X," PLOT OF WEIGHTED RESIDUALS VERSUS",	14170 14180
*" 2ND. INDEPENDENT VARIABLE X2 (SCL.UNITS)")	14190
8401 FORMAT(, CONVERGENCE CRITERION , E10.4)	14200
8402 FORMAT(" EXPANSION CRITERION ",E12.4) 8403 FORMAT(" MINIMUM OF FITTED QUADRATIC SURFACE IS ",E15.8/	14210
*" COMPARE WITH MINIMUM FOUND BY ITERATION ",E15.8/	14210 14220 14230
*" IF DIFFERENCE IS LARGE THE STATISTICAL EVALUATION OF THE",	14240
*" PARAMETERS IS"/" INACCURATE AND A NEW RUN WITH A DIFFERENT",	14250

*" STOPPING CRITERION, EXPANSION"/" CRITERION OR DIFFERENT RANGE",	14260
*" OF THE PARAMETERS IS RECOMMENDED.")	14270
04 FORMAT(/ TYPE BACK (B**) IF YOU WANT TO MAKE CORRECTIONS OR", *" OK IE YOU WANT"/" TO CONTINUE BUT WITHOUT STATISTICAL"	14280
<pre>*" OK IF YOU WANT"/" TO CONTINUE BUT WITHOUT STATISTICAL", *" EVALUATION OF THE PARAMETERS"// ></pre>	14290 14300
22 FORMAT(" NO. OF SS EVALUATIONS ",14,17%,	14310
*" NO. OF CONSTR. VIOLATIONS ",14)	14320
24 FORMAT(" THE QUADRATIC SURFACE FITTING REQUIRED A FURTHER", 14,	14330
* SS EVALUATIONS") 25 FORMAT(/1H)	14340 14350
44 FORMAT(14360
//1X,27(1H)," END OF FITTING NO.",12,2X,26(1H*) //)	14370
00 IF(ANS.ED.3HYES.OR.ANS.ED.2HNO) GO TO 1002	14380
IF(ANS.EQ.4HBACK) GO TO 1004 WRITE(6,1020)	14390 14400
01 IF(SKIP) NS=NQ	14410
GO TO (5 10 15 20 25 30 35 40 45 50 55 70 \$5 90 95 100 105	14420
*130,160,170,175,185 N0 02 G0 T0 (6,11,16,21,26,31,36,41,46,51,56,71,86,91,96,101,105,	14430
*130,161,171,175,185 NQ	1440 14450
03 IF(ANS.EQ.4HBACK) GO TO 1004	14460
GO TO 1002	14470
04 NS=NO	14480
WRITE(6,1021) SKIPS=SKIP	14490 14500
CALL READ(1, AR)	14510
CALL READ(1,AR) IF(ANS.EQ.4HBACK) GO TO 1004	14520
SKIP=SKIPS NQ=IFIX(ANS+.1)	14530
IF(SKIP.AND.NO.GE.1.AND.NO.LE.22) GO TO 1001	14540
DO 1005 I=1,NS	14560
IF(I.EQ.NO) GO TO 1001	14570
05 CONTINUE WRITE(6,1022)	14580 14590
NQ=NS	14600
GO TO 1004	14610
06 NCORR=NCORR+1	14620
IF(NCORR.EQ.1.AND.NS.NE.NG.AND.NS.NE.22) WRITE(6,1023)NS NG=NS	14630
GO TO 1001	14650
07 WRITE(6,1024)	14660
GO TO 1001 08 WRITE(6,1025) NAGR	14670
GO TO 1001	14680
09 WRITE(6,1026)	14700
GO TO 1001	14710
10 WRITE(6,1028) GO TO 1004	14720
11 WRITE(6, 1029)	14730 14740
GO TO 1004	14750
12 WRITE(6,1034)1,Y(1),IWGHT	14760
GO TO 1004 13 WRITE(6,1035)	14770 14780
GO TO 1004	14780
20 FORMAT()," INPUT ERROR, YOUR ANSWER MUST BE ONE OF THE FOLLOWING"	/ 14800
*" FOUR, Y N B** R** - TRY AGAIN -"/)	14810
21 FORMAT(/" WHAT REQUEST NUMBER DO YOU WANT TO GO TO"//) 35 FORMAT(" YOU FORGOT AN ESSENTIAL REQUEST (L.E. & 9.10 OR 17)")	14820
35 FORMAT(" YOU FORGOT AN ESSENTIAL REQUEST (I.E. 8,9,10 OR 17)") 22 FORMAT(/," INPUT ERROR, THE ENTERED NUMBER IS NOT AMONG"/	14840
*" THE REQUEST NUMBERS ABOVE TRY AGAIN -"/)	14850
23 FORMAT(" CONTINUE NOW AFTER CORRECTION FROM REQUEST NO."13/) 24 FORMAT(/," INPUT ERROR, THE ENTERED NUMBER IS NOT",	14860 14870
et ioninity, inforentiation, ine entered number 13 nul ,	170/0

JOZ

		C	
	*" IN THE ALLOWED RANGE TRY AGAIN -"/)	14880	
	1025 FORMAT(/, " INPUT ERROR, YOUR INPUT DOES NOT AGREE WITH YOUR "/	14890	
	* INPUT ABOVE UNDER REQUEST NUMBER , 13, - TRY AGAIN - /)	14900	
	1026 FORMAT(/, INPUT ERROR, THE INPUT IS NOT IN THE RIGHT ORDER	14910	
	· - TRY AGAIN - /)	14920	
	1027 FORMAT(/ " INPUT ERROR, YOUR ANSWER MUST BE OK B OR R",	14930	
	* - TRY AGAIN - /)	14940	
	1028 FORMAT(" INPUT ERROR, YOUR INPUT UNDER REQUEST NUMBER 6 DOES"/	14950	
	* NOT AGREE WITH THE INPUT UNDER REQUEST NUMBER 5")	14960	
	1029 FORMAT(INPUT ERROR, YOUR INPUT UNDER REQUEST NUMBER 9 DOES /	14970	
	* NOT AGREE WITH THE INPUT UNDER REQUEST NUMBER &)	14980	
	1030 FORMAT(/ INPUT ERROR, THE INITIAL ESTIMATE OF PARAMETER NO	14990	
	*12," IS NOT WITHIN"/" ITS LIMITS. ENTER -BACK- IF YOU WANT TO",	15000	
	* MAKE CORRECTION OR -OK- IF YOU / WANT THE PROGRAM TO CHOOSE .	15010	
	* AN ACCEPTABLE INITIAL PARAMETER ESTIMATE (/)	15020	
	1031 FORMAT(/" INPUT ERROR, THE STEP SIZE FOR PARAMETER NO.", 12,	15030	
	* DOES NOT AGREE WITH / THE LIMITS FOR THIS PARAMETER. ENTER .	15040	
	BACK- IF YOU WANT TO MAKE CORRECTION / OR -OK- IF YOU WANT .	15050	
	* THE PROGRAM TO CHOOSE AN ACCEPTABLE STEP SIZE //)	15060	
	1032 FORMAT(" INPUT ERROR, YOU FORGOT TO ENTER OBSERVATION NUMBER", 13/	15070	
	* REENTER YOUR DATA STRICTLY ACCORDING TO FOLLOWING REQUEST "/)	15080	
	1033 FORMAT(" INPUT ERROR, THE WEIGHT FOR THE OBSERVATION ",	15090	
	*"IS NEGATIVE."/" REENTER THE LAST LINE IN CORRECT FORM."//>	15100	
	1034 FORMAT(/" INPUT ERROR, THE WEIGHT FOR OBSERVATION NO.", 13.	15110	
	* (Y=",E10.4,") "/" IS NOT DEFINED BY THE WEIGHTING SCHEME NO.",	15120	
	*12)	15130	
	END	15140	
3	C ************************************	15150	
	SUBROUTINE NELDR(F, STEP, NOP, FUNC, MAX, IPRINT, STOPCR, NLOOP, IQUAD,	15160	
	 SIMP, VAR, FUNCTN, IFAULT) 	15170	
.0	SUBROUTINE NELDR(F,STEP,NOP,FUNC,MAX,IPRINT,STOPCR,NLOOP,IQUAD, *SIMP,VAR,FUNCTN,IFAULT)		15160

*" IN THE ALLOWED RANGE TRY AGAIN -"/)	14880
FORMAT(/, " INPUT ERROR, YOUR INPUT DOES NOT AGREE WITH YOUR "/	14890
* INPUT ABOVE UNDER REQUEST NUMBER", 13," - TRY AGAIN - "/)	14900
FORMAT(/, INPUT ERROR, THE INPUT IS NOT IN THE RIGHT ORDER. ",	14910
• - TRY AGAIN - /)	14920
FORMAT(/ INPUT ERROR, YOUR ANSWER MUST BE OK B OR R",	14930
*"- TRY AGAIN -"/)	14940
FORMAT(" INPUT ERROR, YOUR INPUT UNDER REQUEST NUMBER 6 DOES"/	14950
*" NOT AGREE WITH THE INPUT UNDER REQUEST NUMBER 5")	14960
FORMAT(" INPUT ERROR, YOUR INPUT UNDER REQUEST NUMBER 9 DOES"/	
*" NOT AGREE WITH THE INPUT UNDER REQUEST NUMBER 8")	14970
EODMATCH INDIA EDDO THE INFUL ONDER REQUEST NUMBER 0 7	14980
FORMAT(/" INPUT ERROR, THE INITIAL ESTIMATE OF PARAMETER NO.",	14990
<pre>*12, IS NOT WITHIN"/" ITS LIMITS. ENTER -BACK- IF YOU WANT TO", * MAKE CORRECTION OR -OK- IF YOU"/" WANT THE PROGRAM TO CHOOSE",</pre>	15000
TAKE CORRECTION OR -UK- IF TOU / MANIT THE PROGRAM TO CHOUSE ,	15010
* AN ACCEPTABLE INITIAL PARAMETER ESTIMATE //)	15020
FORMAT(/ INPUT ERROR, THE STEP SIZE FOR PARAMETER NO. 12 " DOES NOT AGREE WITH // THE LIMITS FOR THIS PARAMETER. ENTER, " BACK_ IF YOU WANT TO MAKE CORRECTION" OR -OK- IF YOU WANT,	15030
DUES NOT AGREE HITH / THE LIMITS FOR THIS PARAMETER. ENTER,	15040
-BALK- IF YOU MANT TU MAKE CURRELITON / UR -UK- IF YOU MANT ,	15050
* THE PROGRAM TO CHOOSE AN ACCEPTABLE STEP SIZE //)	15060
FORMAT(" INPUT ERROR, YOU FORGOT TO ENTER OBSERVATION NUMBER", 13/ *" REENTER YOUR DATA STRICTLY ACCORDING TO FOLLOWING REQUEST "/)	15070
* REENTER YOUR DATA STRICTLY ACCORDING TO FOLLOWING REQUEST "/)	15080
FORMAT(" INPUT ERROR, THE WEIGHT FOR THE OBSERVATION ",	15090
*"IS NEGATIVE."/" REENTER THE LAST LINE IN CORRECT FORM."//>	15100
FORMAT(/" INPUT ERROR, THE WEIGHT FOR OBSERVATION NO.", 13,	15110
*" (Y=",E10,4,") "/" IS NOT DEFINED BY THE WEIGHTING SCHEME NO "	15120
*12.)	15130
END	15140
***************************************	15150
CURROUTINE NELOD/E CTED NOD FUNC MAY IDDINT CTODED NI COD TOURD	15175

23480
23490 23500
23510 23520
23530 23540 23550

DIMENSION X(100), Y(100), WYN(100), LX(100), LY(100), LIM1(10), LIM2(10)	23560
*,LXF(75),LYF(75),XF(75),YF(75),F(20),A(75),XX(9,100) COMMON /DATA/ XX,Y,WYN,NOBS	23570 23580
COMMON /FUNNUM/ITHFUN	23590
COMMON /LXLY/LX,LY Common /B1/ X	23600 23610
COMMON /B4/ NIND	23620
COMMON /B6/ NOP Common /B8/NFUNC	23630 23640
COMMON /B9/LIM1,LIM2	23650
DATA BLANK, POINT1, POINT2, DOT, STAR /1H ,1H*,1H0,1H.,1HX/ DO 45 II=1, NFUNC	23660 23670
N1=LIM1(II)	23680
N2=N1+1 N3=LIM2(11)	23690 23700
NN=N3-1	23710
XMIN=X(N1) XMAX=X(N1)	23720 23730
YMIN=Y(N1) YMAX=Y(N1)	23740
IF(II.EQ.1) WRITE(6,100)	23750 23760
IF(II.GT.1) WRITE(6,101)II DO 5 1=N2,N3	23770 23780
IF(X(1),GT,XMAX) XMAX=X(1)	23790
IF(X(I).LT.XMIN) XMIN=X(I) IF(Y(I).GT.YMAX) YMAX=Y(I)	23800 23810
5 IF(Y(I).LT.YMIN)YMIN=Y(I)	23820
SPANX=XMAX-XMIN IF(SPANX.EQ.0.)RETURN	23830 23840
DELX=SPANX/74.	23850
D0 10 1=1,75 XF(I)=XMIN+FLOAT(I-1)+DELX	23860 23870
D0 7 L=N1,NN 7 IF(XF(1).GT.X(L).AND.XF(1).LE.X(L+1))	23880
YF(1)=Y(L)+(Y(L+1)-Y(L))(XF(1)-X(L))/(X(L+1)-X(L))	23890 23900
ITHFUN=II CALL MODEL(YF(I),XF(I),F,0)	23910 23920
IF(YF(I).GT.YMAX) YMAX=YF(I)	23930
10 IF(YF(I).LT.YMIN) YMIN=YF(I) SPANY=YMAX-YMIN	23940 23950
IF(SPANY.EQ.0.)RETURN	23960
DO 15 I=N1,N3 IF(WYN(I).EQ.0.) GO TO 12	23970 23980
LY(I)=IFIX(49.*(YMAX-Y(I))/SPANY+.5) +1	23990
LX(I)=IFIX(74.*(X(I)-XMIN)/SPANX*.5) +1 GO TO 15	24000 24010
12 LY(1)=0 LX(1)=0	24020 24030
15 CONTINUE	24040
D0 20 I=1,75 LYF(I)=IFIX(49.*(YMAX-YF(I))/SPANY+.5) +1	24050 24060
20 LXF(I)=IFIX(74.*(XF(I)-XMIN)/SPANX+.5) +1	24070
NZERO=IFIX(49.*YMAX/SPANY+.5) +1 IF(NZERO.EO.50.AND.YMIN.GT.0.)NZERO=51	24080 24090
IF(NZERO.EO.1.AND.YMAX.LT.O.) NZERO=0	24100
D0 40 I=1,50 D0 35 J=1,75	24110 24120
A(J)=BLANK	24130
IF(I.EQ.NZERO)A(J)=DOT N=0	24140 24150
DO 25 K=1,75 25 IF(LXF(K).EQ.J.AND.LYF(K).EQ.I) A(J)=POINT1	24160 24170
	24170

J03

	IF(A(J).EQ.POINT1)N=1	24180
	DO 30 L=N1,N3	24190
30	IF(LX(L).EQ.J.AND.LY(L).EQ.I) A(J)=POINT2	24200
35	IF(N.EQ.1.AND.A(J).EQ.POINT2)A(J)=STAR	24210
40	WRITE(6,105)A	24220
45	WRITE(6,110) CONTINUE	24230 24240
	WRITE(6,115)	24250
100	FORMAT(//15X, " PLOT OF FITTED CURVE AND DATA POINTS (SCL.UNITS)"//	24260
	*1X,77(1H-))	24270
101	FORMAT(/26X " RESPONSE SYSTEM NO " 12//1X 77(1H-))	24280
105	FORMAT(2H 1,75A1,1H1) FORMAT(2H 1,77(1H-)/ 0 = OBSERVED POINTS + = CALCULATED POINTS",	24290
110	FORMAT (1X,77(1H-)/" 0 =OBSERVED POINTS + =CALCULATED POINTS",	24300
	A -LALLULATED AND UBSERVED PUINTS ()	24310
115	FORMAT("THE ACCURACY OF THE ABOVE PLOT(S) IS +-1/2 CHARACTER", " AND LINE DISTANCE."//)	24320
	RETURN	24330 24340
	END	24350
C		24360
	SUBROUTINE PLOT(X,Y,NPOINT,LINES)	24370
	DIMENSION X(100), Y(* 0), A(75), LX(100), LY(100), LIM1(10), LIM2(10)	24380
	COMMON /LXLY/LX,LY	24390
	COMMON /B7/ ITABLE	24400
	COMMON /B8/NFUNC	24410
	COMMON /B9/LIM1,LIM2	24420
	COMMON /B13/IGNORE DATA ?LANK,POINT,DOT/1H ,1H0,1H./	24430
	N1=1	24440 24450
	N2=2	24460
	N3=NPOINT	24470
	WZ-1	2//00

F(1)=FNEW(1)	5480	
FSAVE(I)=F(I)	5490	
IF(F(I)+STPNEW(I).GT.F2(I)) STPNEW(I)=-STPNEW(I)	5500	
651 STEP(1)=STPNEW(1)	5510	
GO TO 1664	5520	
652 WRITE(6,6731) FUNC	5530	
GO TO 1655	5540	
653 IF(MAX.EQ.1) GO TO 1654	5550	
WRITE(6,6740) FUNC	5560	
CALL PLACE(F1, F2, FMIN, 40, NPR, A)	5570	

J01

GO TO 1656	5580
1654 WRITE(6,6741) FUNC 1655 CALL PLACE(F1,F2,F,40,NPR,A)	5590 5600
1656 CONTINUE DO 1657 J=1, NPR	5610 5620
IF(F1(J).EQ.F2(J)) GO TO 1657	5630
N1=40*(J-1)+1	5640
N2=N1+39	5650
IF(MAX.E0.1) WRITE(6,6730) J,F(J),(A(I),I=N1,N2) IF(MAX.NE.1) WRITE(6,6730) J,FMIN(J),(A(I),I=N1,N2)	5660 5670
1657 CONTINUE	5680
WRITE(6,6745)	5690
1660 WRITE(6,6748) CALL READ(1,AR)	5700
IF(IGJ.EQ.2) GO TO 1001	5710 5720
IF(ANS.EQ.4HBACK) GO TO 1004	5730
IF(ANS.EQ. 3HYES) GO TO 1661	5740
LF(ANS.EQ.2HNO.AND.MAX.EQ.1) GO TO 1663 .F(ANS.EQ.2HNO) GO TO 1666	5750 5760
WRITE(6, 1020)	5770
GD TO 1660	5780
1661 ANS11=3HYES	5790
ANS12=3HYES N11=NOP	5800 5810
N12=NOP	5820
DO 1662 I=1,NOP	5830
NF(1)=1	5840
NT(I)=I F(I)=FNEW(I)	5850 5860
FS(1)=F(1)	5870
FSAVE(1)=F(1)	5880
IF(F(1)+STPNEW(1).GT.F2(1)) STPNEW(1)=-STPNEW(1)	5890
STEP(I)=STPNEW(I) 1/(F1(I).E0.F2(I)) STEP(I)=0.	5900 5910
1662 STPS(1)=STEP(1)	5920
IF(MAX.NE.1) GO TO 1664	5930
1563 NS=18	5940
GO TO 25 1664 WRITE(6,6747)	5950 5960
CALL READ(1, AR)	5970
IF(IGO.EO.2) GO TO 1001	5980
IF(ANS.EQ.4HBACK) GO TO 1004	5990
IF(ANS.EQ.2HNO) GO TO 350 IF(ANS.NE.3HYES) GO TO 1665	6000 6010
NO=22 G0 T0 1004	6020
GO TO 1004	6030
1665 WRITE(6,1029) GO TO 1664	6040
1666 IF(NS.GT.NQ) GO TO 1906	6050 6060
170 NQ=20	6070
WRITE(6,675)	6080
CALL READ(1,AR) IF(IGO.EO.2) GO TO 1001	6090 6100
IF(ANS.EG.4HBALK) GO TO 1004	6110
IF(ANS.EQ.2HNO) GO TO 171	6120
IF(ANS.EQ.3HYES) GO TO 173	6130
WRITE(6,1020) GO TO 170	6140 6150
171 WRITE(6.680)	6160
CALL READ(1, AR)	6170
IF(IGO.EO.2) GO TO 1001	6180
IF(ANS.ED.4HBACK) GO TO 1004	6190

K01

IF (ANS.E0.34YES) NS=8 6200 IF (ANS.E0.34YES) GO TO 9 6210 IF (ANS.E0.24HNO) GO TO 172 6220	
IF(ANS.EQ.3HYES) GO TO 9 6210	
IF(ANS.EQ.3HYES) GO TO 9 6210	
1E(ANS.ED.2HND) GO TO 172 6220	
WRITE(6,1020) 6230	
GO TO 171 6240	
172 WRITE(6,685) 6250	
STOP 6260	
173 JF(NS, GT.NQ) GO TO 1006 6270	
175 NO=21 6280 WRITE(6,690) 6290	
CALL READ(1,AR) 0300	
IF(IG0.E0.2) G0 TO 1001 6310	
IE(ANS, ED, 4HRACK) GO TO 1004 6320	
IF(ANS.EQ.3HYES) GO TO 178 6330	
IF(ANS.EQ.2HNO) GO TO 176 6340	
WRITE(6,1020) 6350	
G0 T0 175 6360	
176 CONTINUE 6370	
ANS11=3HYES 6380 ANS12=2HNO 6390	
N11=NOP 6400	
DO 177 I=1,NDP 6410	
NF(1)=1 6420	
177 FS(1)=FSAVE(1) 6430	
GO TO 179 6440	
178 ANS12=2HND 6450	
ANS11=3HVES 6460	
N11=NOP 6470 D0 1780 I=1,NOP 6480	
DO 1780 I=1,NOP 6480 WF(I)=I 6490	
1780 FS(1)=F(1) 6500	

GO TO 1004	14790
1020 FORMAT()," INPUT ERROR, YOUR ANSWER MUST BE ONE OF THE FOLLOWING"/	14800
*" FOUR, Y N B** R** - TRY AGAIN -"/)	14810
1021 FORMAT(/" WHAT REQUEST NUMBER DO YOU WANT TO GO TO"//)	14820
1035 FORMAT(" YOU FORGOT AN ESSENTIAL REQUEST (I.E. 8,9,10 OR 17)")	14830
1022 FORMAT(/," INPUT ERROR, THE ENTERED NUMBER IS NOT AMONG"/	14840
*" THE REQUEST NUMBERS ABOVE TRY AGAIN -"/)	14850
1023 FORMAT(" CONTINUE NOW AFTER CORRECTION FROM REQUEST NO."13/)	14860
1024 FORMAT(/," INPUT ERROR, THE ENTERED NUMBER IS NOT",	14870

_			_
	*" IN THE ALLOWED RANGE TRY AGAIN -"/)	14880	
	1025 FORMAT(7, INPUT ERROR, YOUR INPUT DOES NOT AGREE WITH YOUR"/ * INPUT ABOVE UNDER REQUEST NUMBER", I3, - TRY AGAIN - "/)	14890	
	*" INPUT ABOVE UNDER REQUEST NUMBER", 13, " - TRY AGAIN -"/) 1026 FORMAT(/," INPUT ERROR, THE INPUT IS NOT IN THE RIGHT ORDER. ",	14900	
	* - TRY AGAIN - /)	14920	
	1027 FORMAT(/ " INPUT ERROR, YOUR ANSWER MUST BE OK B OR R" ,	14930	
	* - TRY AGAIN - "/)	14940	
	1028 FORMAT(" INPUT ERROR, YOUR INPUT UNDER REQUEST NUMBER 6 DOES"/ *" NOT AGREE WITH THE INPUT UNDER REQUEST NUMBER 5")	14950	
	1029 FORMAT(" INPUT ERROR, YOUR INPUT UNDER REQUEST NUMBER 9 DOES"/	14970	
	" NOT ACCES UTTU THE INDUT UNDER DECHECT NUMPER ""	14980	
	1030 FORMAT(/ INPUT ERER, THE INITIAL ESTIMATE OF PARAMETER NO., *12," IS NOT WITHIN"/" ITS LIMITS. ENTER -BACK- IF YOU WANT TO", *" MAKE CORRECTION OR -OK- IF YOU"/" WANT THE PROGRAM TO CHOOSE",	14990	
	*12, IS NUT WITHIN / ITS LIMITS, ENTER -BACK- IF YOU HANT TO,	15000	
	* AN ACCEPTABLE INITIAL PARAMETER ESTIMATE" //)	15010 15020	
	 AN ACCEPTABLE INITIAL PARAMETER ESTIMATE" //) 1031 FORMAT(/" INPUT ERROR, THE STEP SIZE FOR PARAMETER NO.", IZ, * DOES NOT AGREE WITH'/ THE LIMITS FOR THIS PARAMETER. ENTER", 	15030	
	* DOES NOT AGREE WITH / THE LIMITS FOR THIS PARAMETER. ENTER ,	15040	
		15050	
	 THE PROGRAM TO CHOOSE AN ACCEPTABLE STEP SIZE //) 1032 FORMAT(" INPUT ERROR, YOU FORGOT TO ENTER OBSERVATION NUMBER", 13/ * REENTER YOUR DATA STRICTLY ACCORDING TO FOLLOWING REQUEST '/) 	15060 15070	
	* REENTER YOUR DATA STRICTLY ACCORDING TO FOLLOWING REQUEST ()	15080	
	1033 FORMAT(" INPUT ERROR, THE WEIGHT FOR THE OBSERVATION ", *"IS NEGATIVE."/" REENTER THE LAST LINE IN CORRECT FORM."//)	15090	
	* IS NEGATIVE."/" REENTER THE LAST LINE IN CORRECT FORM."//)	15100	
	1034 FORMAT(/ IMPUT ERROR, THE WEIGHT FOR OBSERVATION NO.", 12, (Y=",E10.4,") "/" IS NOT DEFINED BY THE WEIGHTING SCHEME NO.",	15110	
	*12)	15130	
	END	15140	
	C SUBROUTINE NELDR(F, STEP, NOP, FUNC, MAX, IPRINT, STOPCR, NLOOP, IQUAD,		
	* SIMP, VAR, FUNCTN, IFAULT)	15160	
	DOUBLE PRECISION DMEAN, DMEANS, DHCV, FUNC, FUNCS, H, HMIN, HMAX.	15180	
	+HSTAR,HSTST,YMIN,AO,AVAL,BMAT,VC,TEMP,OMIN,PBAR DIMENSION F(20),STEP(20),AR(12),F1(20),F2(20)	15190	
	DIMENSION F4203, STEP(203, AR(12), F1(203, F2(20)) DIMENSION G(11, 203, H(11), PBAR(203, PSTAR(203, PSTST(203, VARCOV(55)) DIMENSION AVAL(203, PMAT(55), PMIN(203, VARCOV, 103, TEMP(203))	15200	
	DIMENSION AVAL(20), RMAT(55), PMIN(20), V(55), VAR(10), TEMP(20)	15210	
	DIMENSION AVAL(20), BMAT(55), PMIN(20), VC(55), VAR(10), TÉMP(20) EQUIVALENCE(PMIN, PSTAR), (AVAL, PBAR)	15230	
	COMMON /PARLIM/F1,F2	15240	
		15250	
	COMMON /B4/ NIND	15260	
	COMMON /B5/NG, NS, 1GO	15280	
	COMMON /B11/AR	15290	
		15300 15310	
	COMMON /B15/ SKIP	15320	
	LOGICAL SKIP	15330	
	A=1.	15340	
	B=0.5 C=2 5	15350	
	LIMITS=1	15370	
	CALL FUNCTN(F,FUNC)	15380	
	DIMENSION AVAL(20), BHAT(55), PMIN(20), VC(35), VAR(10), TEMP(20) EQUIVALENCE(PMIN, PSTAR, (AVAL, PBAR) COMMON /PARLIM/FI,F2 COMMON /CONSTR/LIM/TS COMMON /B3/VARCOV, VC, NEVAL1, NEVAL2, AMINO COMMON /B4/ NINO COMMON /B1/AR COMMON /B1/AR COMMON /B1/AR COMMON /B1/AR COMMON /B1/NVIOL COMMON /B15/ SKIP LOGICAL SKIP A=1. B=0.5 C=2.5 LIMITS=1 CALL FUNCTN(F,FUNC) NEVAL=1 IF(FUNC,GT.1D50) WRITE(6,1000)	15390	
	IF(FUNC.GT.1D50) WRITE(6,1000) 1000 FORMAT(/ ONE OF MORE OF THE INITIAL PARAMETERS DOES NOT LIE"/	15400	
	* WITHIN THE SPECIFIED CONSTRAINED PARAMETER SPACE. ()	15420	
	IF(FUNC.GT.1D50) WRITE(6,1000) 1000 FORMAT(/ ONE OR MORE OF THE INITIAL PARAMETERS COES NOT LIE"/ * WITHIN THE SPECIFIED CONSTRAINED PARAMETER SPACE."/) IM=6 IM=6	15430	
		15440	
	55 WRITE(IW,100)IPRINT 100 FORMAT("PROGRESS REPORT EVERY",14," FUNCTION EVALUATIONS."//	15450 15460	
	100 FORMAT(" PROGRESS REPORT EVERY", 14," FUNCTION EVALUATIONS."// *" EVAL.NO. FUNC.VALUE PARAMETERS" /) 5 IFAULT = 0	15470	
	IF(NOP.LE.O) IFAULT=3	15490	

IF(NLOOP.LE.O) IFAULT=4	15500
IF (IFAULT.NE.O) RETURN	15510
NAP=0	15520
LOOP=0	15530
IFLAG=0	15540
DO 1 1=1,NOP	15550
IF(STEP(I).NE.0.0) NAP=NAP+1	15560
1 CONTINUE	15570
IF(NAP.EQ.0)RETURN	15580
MORE=0.20*((FLOAT(NAP+1))**2.5)	15590
MORE=MINO(50, MORE)	15600
MORE=MAX0(20, MORE)	15610
DO 6 I=1,NOP	15620
6 G(1,1)=F(1)	15630
IROW=2	15640
DO 7 I=1,NOP	15650
IF(STEP(1).EQ.0.) GO TO 7	15660
DO 9 J = 1, NOP	15670
9 G(IROW, J)=F(J)	15680
G(IROW, I)=G(IROW, I)+STEP(I)	15690
IROW=IROW+1	15700
7 CONTINUE	15710
NP1=NAP+1 NEVAL=0	15720
D0 10 I=1,NP1	15730
DO 11 J=1,NOP	15740 15750
11 F(J)=G(I,J)	15760
CALL FUNCTN(F,H(I))	15770
NEVAL=NEVAL+1	15780
IF(IPRINT)10,10,12	15790
12 IF(NPR.LE.5)WRITE(IW, 1010)NEVAL, H(I), (F(J), J=1, NPR)	15800
THE PREMATE FOR THE PREME	12000

IF(NZERO.EO.50.AND.YMIN.GT.0.)NZERO=51	24090	
IF(NZERO.EQ.1.AND.YMAX.LT.O.) NZERO=0	24100	
DO 40 I=1,50	24110	
DO 35 J=1,75	24120	
A(J)=BLANK	24130	
IF(1.EQ.NZERO)A(J)=DOT	24140	
N=O	24150	
D0 25 K=1,75	24160	
25 IF(LXF(K).EQ.J.AND.LYF(K).EQ.1) A(J)=POINT1	24170	

J03

TELAL IN ED DOINTAIN-1	2/140
IF(A(J).ED.POINT1)N=1 D0 30 L=N1.N3	24180 24190
30 IF(LX(L).EQ.J.AND.LY(L).EQ.I) A(J)=POINTZ	24200
35 IF(N.EQ.1.AND.A(J).EQ.POINT2)A(J)=STAR	24210
40 WRITE(6,105)A	24220
WRITE(6,110)	24230
45 CONTINUÉ	24240
WRITE(6,115)	24250
100 FORMAT(//15x," PLOT OF FITTED CURVE AND DATA POINTS (SCL.UNITS)"//	24260
•1X,77(1H-))	24270
101 FORMAT(/26X," RESPONSE SYSTEM NO.",12//1X,77(1H-)) 105 FORMAT(2H 1,75A1,1H1)	24280
105 FURMAT(2H 1,75A1,1HI)	24290
110 FORMAT (1X,77(1H-)/" 0 =OBSERVED POINTS + =CALCULATED POINTS",	24300
*" X =CALCULATED AND OBSERVED POINTS"/)	24310
115 FORMAT(" THE ACCURACY OF THE ABOVE PLOT(S) IS +-1/2 CHARACTER", * AND LINE DISTANCE."//)	24320 24330
RETURN	24340
END	24350
C	24360
SUBROUTINE PLOT(X,Y,NPOINT,LINES)	24370
SUBROUTINE PLOT(X,Y,NPOINT,LINES) DIMENSION X(100),Y(' 0),A(75),LX(100),LY(100),LIM1(10),LIM2(10)	24380
COMMON /LXLY/LX,LY	24390
COMMON /B7/ ITABLE	24400
COMMON /B8/NFUNC	24410
COMMON /B9/LIM1,LIM2	24420
COMMON /B13/IGNORE	24430
DATA PLANK, POINT, DOT/1H , 1HO, 1H./	24440
N2=2	24450 24460
NSENPOINT	24470
N4=1	24480
IF(ITABLE.EQ.0) N4=NFUNC	24490
DO 25 11=1,N4	24500
IF(ITABLE.EQ.1) GO TO 1	24510
IF(11.GT.1) WRITE(6,125)11	24520
N1=LIM1(II)	24530
N2=N1+1	24540
N3=LIM2(II) (XMIN=X(N1)	24550
XMAX=X(N1)	24560 24570
YMIN=Y(N1)	24580
YMAX=Y(N1)	24590
IF(IGNORE.EQ.0) GO TO 3	24600
DO 2 1=N2,N3	24610
IF(Y(1).EQ.0.) GO TO 2	24620
XMIN=X(1)	24630
XMAX=X(I)	24640
YMIN=Y(I)	24650
YMAX=Y(I) 2 CONTINUE	24660
3 CONTINUE	24670
D0 5 1=N2,N3	24680 24690
IF(Y(1).E0.0 AND. IGNORE.NE.0) GO TO 5	24700
IF(X(1),GT,XMAX) XMAX=X(1)	24710
IF(X(I),LT,XMIN) XMIN=X(I)	24720
IF(Y(I).GT.YMAX) YMAX=Y(I)	24730
IF(Y(I).LT.YMIN) YMIN=Y(I)	24740
5 CONTINUE	24750
SPANX=XMAX-XMIN	24760
SPANY=YMAX-YMIN	24770
IF(SPANX.EQ.OOR.SPANY.EQ.O.)RETURN	24780
B=FLOAT(LINES-1)	24790

K03

DO 10 I=N1.N3	24800
IF(Y(1).EQ.0AND.IGNORE.NE.0) GO TO 9	24810
LY(I)=IFIX(B*(YMAX-Y(I))/SPANY+.5)+1	24820
LX(I)=IFIX(74.*(X(I)-XMIN)/SPANX+.5)+1	24830
GO TO 10	24840
9 LY(I)=0	24850
LX(I)=0	24860
10 CONTINUE	24870
NZERO=IFIX(B*YMAX/SPANY+.5)+1	24880
IF(NZERO.EQ.LINES.AND.YMIN.GT.O.)NZERO=LINES+1	24890
IF(VZERO.EQ.1.AND.YMAX.LT.0.) NZERO=0	24900
WRITE(6,110)	24910
DO 20 I=1,LINES	24920
DO 15 J=1,75	24930
A(J)=BLANK	24940
IF(I.EQ.NZERO)A(J)=DOT	24950
DO 15 K=N1,N3	24960
15 IF(LX(K).EQ.J.AND.LY(K).EQ.I) A(J)=POINT	24970
20 WRITE(6,105)A	24980
WRITE(6,100)	24990
WRITE(6,115)YMAX, YMIN, XMAX, XMIN	25000
100 FORMAT(1H ,77(1H-))	25010
105 FORMAT(2H 1,75A1,1H1)	25020
110 FORMAT(/1X,77(1H-))	25030
115 FORMAT(" YMAX=",E11.4," YMIN=",E11.4," XMAX=",E11.4,	25040
* XMIN=",E11.4) IF(ITABLE.E0.0) GO TO 25	25050
IE(ITABLE E0.0) GO TO 25	25060
HRITE(6, 120) NPOINT (X(1), Y(1), 1=N1, N3)	25070
WRITE(6,120) NPOINT (X(1),Y(1),I=N1,N3) 120 FORMAT(/" THE COORDINATES OF THE",I3," POINTS ARE"// *(3(" (",E9.3,",",E9.3,")",2X)))	25080
+(3(" (" F9 3 " F9 3 ")" 2x)))	25090
125 FORMAT(/ 28X, " RESPONSE SYSTEM NO.", 12)	25100
25 CONTINUE	25110

NU=22 GD T0 1004 1665 WRITE(6,1022)) GD T0 1664 1666 IF(NS.GT.NO) GO T0 1906 170 NO=20 WRITE(6,675) CALL READ(1,AR) IF(IGO.EO.2) GO T0 1001 IF(ANS.EO.4HBACK) GO T0 1004 IF(ANS.EO.3HYES) GO T0 173 WRITE(6,1020) GD T0 170 171 WRITE(6,680) CALL READ(1,AR) IF(IGO.EO.2) GO T0 1001 IF(ANS.EO.4HBACK) GO T0 1004	6020 6030 6040 6050 6060 6070 6080 6090 6110 6110 6120 6130 6140 6150 6160 6170 6180 6190	
	K01	
IF(ANS.EQ.3HYES) NS=8 IF(ANS.EQ.2HNOS) GO TO 9 IF(ANS.EQ.2HNOS) GO TO 172 WRITE(6,1020) GO TO 171 172 WRITE(6,685) STOP 173 IF(NS.GT.NO) GO TO 1006 175 NG=21 WRITE(6,690) CALL READ(1,AR) IF(IG.EQ.2) GO TO 1001 IF(ANS.EQ.4HBACK) GO TO 1004	6200 6210 6220 6230 6240 6250 6250 6250 6270 6280 6270 6280 6290 5290 5310 6310 6320	
IF(ANS.E0.3HYES) GO TO 178 IF(ANS.E0.2HNO) GO TO 176 WRITE(6,1020) GO TO 175 176 CONTINUE ANS11=3HYES ANS12=2HNO N11=NOP DO 177 I=1,NOP NF(I)=I 177 FS(I)=FSAVE(I) GO TO 179 178 ANS12=2HNO ANS11=3HYES N11=NOP DO 1780 I=1,NOP NF(I)=I 1780 FS(I)=F(I)	6330 6340 6350 6350 6360 6390 6400 6410 6420 6440 6440 6440 6440 6440 6440 644	
179 IF(NS.GT.NQ) GO TO 1006 HRITE(6,692) NO=22 GO TO 1004 185 NO=22 WRITE(6,694) CALL READ(1,AR) IF(IGO.EO.2) GO TO 1001 IF(ANS.EO.3HYES.OR.AMS.EO.4HBACK) GO TO 1004 IF(ANS.EO.2HNO) GO TO 186 WRITE(6,1020) GO TO 185	6510 6520 6530 6540 6550 6560 6570 6580 6590 6690 6600 6610	
186 N0=18 G0 T0 131 300 WRITE(6,663) CALL READ(1,AR) IF(IGD.ED.2) G0 T0 1001 IF(ANS.ED.4HBACK) G0 T0 1004 IF(ANS.EL.3HYES.AND.ANS.NE.2HNO) G0 T0 301 PLOTDP=ANS	6620 6630 6640 6650 6660 6670 6680 6690 6700	
GO TO 302 301 WRITE(6,1020)	6710 6720	
GO TO 300 302 NCON=NOP-NEWNOP WRITE(6,750) WRITE(6,752) WRITE(7,750)	6730 6740 6750 6760 6770	
D0 305 1=1,NOP 305 WRITE(6,754)1,F1(1).F2(1),F(1),STEP(1) WRITE(6,756) WSAVE=1	6780 6790 6800 6810	

L01

	DO 310 1=1,NOBS	6820	
	N=LABEL(1)	6830	
	IF(N.EQ.NSAVE+1) WRITE(6,813)N	6840	
	NSAVE=N	6 50	
	IF(NIND.GT.1) GO TO 308	6860	
	IF(ANS16.EQ. 3HYES) GO TO 307	6870	
	WRITE(6,758)1,X(1),Y(1)	6880	
	GO TO 310	6890	
30	7 WRITE(6,760)1,X(1),Y(1),WY(1),WYN(1)	6900	
	GO TO 310	6910	
30	8 IF(ANS16.EQ. 3HYES) GO TO 309	6920	
	WRITE(6,762)1,X(1),Z(1),Y(1)	6930	
	GO TO 310	6940	
30	9 WRITE(6,764)1,X(1),Z(1),Y(1),WY(1),WYN(1)	6950	
31	O CONTINUÉ	6960	
	WRITE(6,818)	6970	
	IF(NIND.LT.3) GO TO 312	6980	
	WRITE(6,817)(1,1=3,9)	6990	
144	DO 311 I=1,NOBS 1 HRITE(6,819)I,(XX(J,I),J=3,NIND)	7000	
31	1 WRITE(6,819)1,(XX(J,1),J=3,NIND)	7010	
	WRITE(6,818) 2 IF(PLOTOP.EO.2HNO) GO TO 313	7020	
51.	2 TECELOTOP.E0.2HNO) GO TO 513	7030	
	WRITE(6,765)	7040	

COMMON /B12/NPR	15300	
COMMON /B14/NVIOL	15310	
COMMON /B15/ SKIP	15320	
LOGICAL SKIP	15330	
A=1.	15340	
8=0.5	15350	
C=2.5	15360	
LIMITS=1	15370	
CALL FUNCTN(F, FUNC)	15570	
NEVAL=1	15380	
	15390	
IF(FUNC.GT.1D50) WRITE(6,1000)	15400	
1000 FORMAT(/ ONE OR MORE OF THE INITIAL PARAMETERS COES NOT LIE /	15410	
* WITHIN THE SPECIFIED CONSTRAINED PARAMETER SPACE. /)	15420	
IW=6	15430	
IF(IPRINT)5,5,55	15440	
55 WRITE(IW, 100) IPRINT	15450	
100 FORMAT(" PROGRESS REPORT EVERY", 14, " FUNCTION EVALUATIONS."//	15460	
EVAL.NO. FUNC.VALUE PARAMETERS /)	15470	
5 IFAULT = 0	15480	
IF(NOP.LE.O) IFAULT=3	15490	

K02

	IF(NLOOP.LE.O) IFAULT=4	15500
	IF (IFAULT.NE.O) RETURN	15510
	NAP=0	15520
	LOOP=0	
		15530
	IFLAG=0	15540
	DO 1 I=1,NOP	15550
	IF(STEP(I).NE.0.0) NAP=NAP+1	15560
1	CONTINUE	15570
	IF(NAP.EQ.0)RETURN	15580
	MORE=0.20*((FLOAT(NAP+1))**2.5)	15590
	MORE=MINO(50, MORE)	15600
	MORE=MAX0(20, MORE)	15610
	DO 6 I=1,NOP	15620
0	G(1,I)=F(I)	15630
	IROW=2	15640
	DO 7 I=1,NOP	15650
	IF(STEP(1).EQ.0.) GO TO 7	15660
	DO 9 J = 1, NOP	
		15670
9	G(IROW, J)=F(J)	15680
	G(IROW, I)=G(IROW, I)+STEP(I)	15690
	IROW=IROW+1	15700
7	CONTINUE	15710
	NP1=NAP+1	15720
	NEVAL=0	15730
	DO 10 I=1,NP1	15740
	DO 11 J=1,NOP	15750
11	F(J)=G(1,J)	
		15760
	CALL FUNCTN(F,H(I))	15770
	NEVAL=NEVAL+1	15780
	IF(IPRINT)10,10,12	15790
12	IF(NPR.LE.5)WRITE(IW, 1010)NEVAL, H(I), (F(J), J=1, NPR)	15800
1010	FORMAT(1X, 14, D16.8, 1X, 5E11.5)	
1010	FURHAI (1A, 14, 010.0, 1A, 3E11.37	15810
444	IF(NPR.GT.5) WRITE(IW, 101)NEVAL, H(I), (F(J), J=1, NPR)	15820
101	FORMAT(1X, 14, D16.8, 1X, 5E11.5/(22X, 5E11.5))	15830
10	CONTINUE	15840
	L00P=L00P+1	15850
	IMAX=1	
		15860
	IMIN=1	15870
	DO 13 I=2,NP1	15880
	IF(H(1)-H(IMAX))15,15,14	15890
14	IMAX=1	
17		15900
13	IF(H(1)-H(IMIN))16,13,13	15910
16	IMIN = I	15920
13	CONTINUE	15930
	HMAX = H(IMAX)	15940
	HMIN = H(IMIN)	15950
	DO 17 I=1,NOP	15960
17	PBAR(1)=0.0	15970
	DO 18 I=1,NP1	15980
	IF(I-IMAX)19,18,19	15990
10	DO 20 J=1,NOP	
		16000
	PBAR(J) = PBAR(J)+G(I,J)	16010
18	CONTINUE	16020
	D0 602 J=1,NOP	16030
602	PBAR(J) = PBAR(J)/NAP	16040
002		
	DO 21 I=1,NOP	16050
21	PSTAR(I)=A*(PBAR(I)-G(IMAX,I))+PBAR(I)	16060
	CALL FUNCTN (PSTAR, HSTAR)	16070
	NEVAL=NEVAL+1	16080
	IF(IPRINT)57,57,56	16090
		10070
	IE WOD IE E AND MODINEVAL IDDINES FO AS UDITE THE ASAAS	1/100
56	IF(NPR.LE.5.AND.MOD(NEVAL, IPRINT).EQ.0) WRITE(IW, 1010) *NEVAL, HSTAR, (PSTAR(J), J=1, NPR)	16100 16110

LOZ

IF(NPR.GT.5.AND.MOD(NEVAL, IPRINT).EQ.0) WRITE(
*NEVAL, HSTAR, (PSTAR(J), J=1, NPR)	16130	
57 IF (HSTAR-HMIN)22,23,23	16140	
22 DO 24 I=1,NOP	16150	
24 PSTST(1)=Č*(PSTAR(1)-PBAR(1))+PBAR(1) CALL FUNCTN (PSTST,HSTST)	16160 16170	
NEVAL=NEVAL+1	16180	
IF(IPRINT)60,60,59	16190	
59 IF(NPR.LE.5.AND.MOD(NEVAL, IPRINT).EQ.0) WRITE(IW, 1010) 16200	
<pre>*NEVAL,HSTST,(PSTST(J),J=1,NPR)</pre>	16210	
IF(NPR.GT.5.AND.MOD(NEVAL, IPRINT).EQ.0) WRITE(IW, 101) 16220	
*NEVAL, HSTST, (PSTST(J), J=1, NPR)	16230	
60 1F(HSTST-HMIN)25,26,26	16240	
25 DO 27 I=1.NOP IF(STEP(I).NE.0.0) G(IMAX,I)=PSTST(I)	16250	
27 CONTINUE	16260 16270	
H(IMAX)=HSTST	16280	
GO TO 41	16290	
23 DO 28 I=1,NP1	16300	
IF(1-1MAX)29,28,29	16310	
29 IF(HSTAR-H(1))26,28,28	16320	

D0 2 1 = N2, N3 24610 IF(Y(1), E0.0.) G0 TO 2 24620 XMIN=X(1) 24630 XMAX=X(1) 24640 YMIN=Y(1) 24660 YMAX=Y(1) 24660 2 CONTINUE 24660 2 CONTINUE 24680 D0 5 I=N2,N3 24680 D0 5 I=N2,N3 24680 IF(Y(1),E0.0., AND, IGNORE.NE.0) GO TO 5 24690 IF(Y(1),E0.0., AND, IGNORE.NE.0) GO TO 5 24690 IF(Y(1),E0.0., ANAX_X(1) 2470 IF(X(1),IT,XMAX) XMAX=X(1) 24720 IF(Y(1),E1,YMAX) YMAX=Y(1) 24730 IF(Y(1),LT,YMIN) XMIN=X(1) 24730 SPANX=XMAX-XMIN 24760 SPANX=XMAX-XMIN 24760 SPANX=XMAX-YMIN 24760 B=FLOAT(LINES-1) 24780	1F(1GNUKE.EU.0) GU 10 5	24600	
IF(Y(1).E0.0.) G0 T0 2 24620 XMIN=X(1) 24630 XMAX=X(1) 24630 YMIN=Y(1) 24650 YMAX=Y(1) 24660 2 CONTINUE 24660 2 CONTINUE 24660 3 CONTINUE 24660 1F(Y(1).E0.0AND.IGNORE.NE.0) GO TO 5 24690 1F(X(1).ET.XMAX) XMAX=X(1) 24670 1F(X(1).ET.XMAX) XMAX=X(1) 24710 1F(X(1).ET.XMIN) XMIN=X(1) 24720 1F(Y(1).ET.YMIN) XMIN=X(1) 24730 1F(Y(1).LT.YMIN) YMIN=Y(1) 24740 SPANX=XMAX-XMIN 24760 SPANY=YMAX-YMIN 24770 IF(SPANX.E0.0OR.SPANY.E0.0.)RETURN 24770	DO 2 I=N2.N3	24610	
XMIN=X(1) 24630 XMAX=X(1) 24640 YMAX=X(1) 24640 YMAX=Y(1) 24650 YMAX=Y(1) 24660 2 CONTINUE 24660 3 CONTINUE 24680 D0 5 1=N2,N3 24680 IF(Y(1).E0.0.,AND.IGNORE.NE.0) GO TO 5 24690 IF(X(1).GT.XMAX) XMAX=X(1) 24700 IF(X(1).GT.YMAX) XMAX=X(1) 24710 IF(X(1).GT.YMAX) XMAX=X(1) 24720 IF(Y(1).EC.YMAX) YMAX=Y(1) 24720 IF(Y(1).LT.YMIN) XMIN=X(1) 24730 IF(Y(1).LT.YMIN) YMIN=Y(1) 24740 S CONTINUE 24750 SPANX=XMAX-XMIN 24760 SPANX=XMAX-XMIN 24760 SPANY=YMAX-YMIN 24770 IF(SPANX.E0.0., OR.SPANY,E0.0.) RETURN 24770		24620	
XHAX=X(1) 24640 YHIN=Y(1) 24650 YHAX=Y(1) 24650 Z CONTINUE 24660 2 CONTINUE 24670 3 CONTINUE 24660 D0 5 I=N2.N3 24670 IF(Y(1).E0.0AND.IGNORE.NE.0) GO TO 5 24690 IF(Y(1).E1.XHIN) XHIN=X(1) 24710 IF(Y(1).E1.XHIN) XHIN=X(1) 24720 IF(Y(1).L1.YHAX) YHAX=Y(1) 24720 IF(Y(1).L1.YHAX) YHAX=Y(1) 24730 IF(Y(1).L1.YHIN) YHIN=Y(1) 24750 SDANX=XMAX-XHIN 24750 SPANX=XMAX-XHIN 24770 IF(SPANX.E0.0OR.SPANY.E0.0.)RETURN 24770			
YMIN=Y(1) 24650 YMAX=Y(1) 24660 2 CONTINUE 24670 3 CONTINUE 24680 DD 5 I=N2,N3 24690 IF(Y(1).E0.0AND.IGNORE.NE.0) GO TO 5 24700 IF(X(1).GT.XMAX) XMAX=X(1) 24710 IF(X(1).GT.XMAX) XMAX=Y(1) 24720 IF(Y(1).GT.YMAX) YMAX=Y(1) 24730 IF(Y(1).LT.YMIN) XMIN=X(1) 24740 SPANX=XMAX-XMIN 24760 SPANX=XMAX-XMIN 24760 SPANY=YMAX-YMIN 24770 IF(SPANX.E0.0OR.SPANY.E0.0.)RETURN 24770			
YHÁX=Y(Î) 24660 2 CONTINUE 24670 3 CONTINUE 24680 DD 5 I=N2,N3 24690 IF(Y(I).60.0AND.IGNORE.NE.0) GO TO 5 24700 IF(X(I).GT.XMAX) XMAX=X(I) 24710 IF(X(I).LT.XMIN) XMIN=X(I) 24720 IF(Y(I).CT.YMAX) MAX=Y(I) 24730 IF(Y(I).LT.YMIN) YMIN=Y(I) 24740 5 CONTINUE 24750 SPANX=XMAX-XMIN 24760 SPANX=XMAX-XMIN 24760 IF(SPANX.E0.0OR.SPANY,E0.0.)RETURN 24770		24040	
2 CONTINUE 24670 3 CONTINUE 24680 00 5 I=N2.N3 24690 IF(Y(I).E0.0AND.IGNORE.NE.0) GO TO 5 24700 IF(X(I).GT.XMAX) XMAX=X(I) 24710 IF(X(I).GT.YMAX) YMAX=Y(I) 24720 IF(Y(I).GT.YMAX) YMAX=Y(I) 24730 IF(Y(I).LT.YMIN) YMIN=Y(I) 24740 5 CONTINUE 24750 SPANX=XMAX-XMIN 24770 IF(SPANX.E0.0OR.SPANY.E0.0.)RETURN 24770			
3 CONTINUE 24680 DD 5 I=N2,N3 24690 IF(Y(I).E0.0AND.IGNORE.NE.0) GD TO 5 24700 IF(X(I).GT.XMAX) XMAX=X(I) 24710 IF(X(I).GT.XMAX) XMAX=X(I) 24720 IF(Y(I).GT.YMAX) YMAX=Y(I) 24720 IF(Y(I).GT.YMAX) YMAX=Y(I) 24730 IF(Y(I).LT.YMIN) YMIN=Y(I) 24740 SPANX=XMAX-XMIN 24760 SPANX=YMAX-YMIN 24770 IF(SPANX.E0.0OR.SPANY.E0.0.)RETURN 24770		24660	
3 CONTINUE 24680 DD 5 I=N2,N3 24690 IF(Y(I).E0.0AND.IGNORE.NE.0) GO TO 5 24700 IF(X(I).GT.XMAX) XMAX=X(I) 24710 IF(X(I).GT.YMAX) YMAX=Y(I) 24720 IF(Y(I).GT.YMAX) YMAX=Y(I) 24730 IF(Y(I).LT.YMIN) XMIN=Y(I) 24740 SPANX=XMAX-XMIN 24750 SPANX=YMAX-YMIN 24760 SPANY=YMAX-YMIN 24770 IF(SPANX.E0.0OR.SPANY.E0.0.)RETURN 24770	2 CONTINUE	24670	
D0 5 1=N2,N3 24690 1F(Y(1),E0.0.,AND.1GNORE.NE.0) GO TO 5 24700 1F(X(1),GT.XMAX) XMAX=X(1) 24710 1F(X(1),LT.XMIN) XMIN=X(1) 24720 1F(Y(1),GT.YMAX) YMAX=Y(1) 24720 1F(Y(1),LT.YMIN) YMIN=Y(1) 24730 5 CONTINUE 24750 SPANX=XMAX-XMIN 24750 SPANX=YMAX-YMIN 24760 SPANX=YMAX-YMIN 24770 1F(SPANX.E0.0.,OR.SPANY,E0.0.)RETURN 24770	3 CONTINUE		
IF(Y(1).E0.0.AND.IGNORE.NE.0) GO TO 5 24700 IF(X(1).GT.XMAX) XMAX=X(I) 24710 IF(X(1).GT.XMAX) XMAX=X(I) 24720 IF(Y(1).GT.YMAX) YMAX=Y(I) 24730 IF(Y(1).LT.YMIN) YMIN=Y(I) 24740 SCONTINUE 24750 SPANX=XMAX-XMIN 24760 SPANX=YMAX-YMIN 24760 IF(SPANX_E0.0)RETURN 24770		24690	
IF(X(1).GT.XMAX) XMAX=X(1) 24710 IF(X(1).LT.XMIN) XMIN=X(1) 24720 IF(Y(1).GT.YMAX) YMAX=Y(1) 24730 IF(Y(1).LT.YMIN) YMIN=Y(1) 24740 SDANX=XMAX-XMIN 24750 SPANX=XMAX-XMIN 24760 SPANX=XMAX-XMIN 24770 IF(SPANX.E0.0.)CR.SPANY.E0.0.)RETURN 24780			
IF(X(1).LT.XHIN) XHIN=X(1) 24720 IF(Y(1).GT.YHAX) YHAX=Y(1) 24730 IF(Y(1).LT.YHIN) YHIN=Y(1) 24740 SCONTINUE 24750 SPANX=XMAX-XHIN 24760 SPANX=XMAX-XHIN 24760 SPANY=YMAX-YHIN 24760 IF(SPANX.EQ.0)RETURN 24770 IF(SPANX.EQ.0)RETURN 24780		24700	
IF(Y(1).GT.YHAX) YHAX=Y(1) 24730 IF(Y(1).LT.YHIN) YHIN=Y(1) 24740 SCONTINUE 24750 SPANX=XMAX-XHIN 24760 SPANY=YHAX-YHIN 24770 IF(SPANX.E0.0OR.SPANY.E0.0.)RETURN 24780			
IF(Y(1),LT.YHIN) YHIN=Y(1) 24740 5 CONTINUE 24750 SPANX=XMAX-XHIN 24760 SPANY=YMAX-YHIN 24770 IF(SPANX.E0.0., OR.SPANY,E0.0.) RETURN 24780			
5 CONTINUE 24750 SPANX=XMAX-XMIN 24760 SPANY=YMAX-YMIN 24770 IF(SPANX.EQ.0OR.SPANY.EQ.0.)RETURN 24770			
5 CONTINUE 24750 SPANX=XMAX-XMIN 24760 SPANY=YMAX-YMIN 24770 IF(SPANX.EQ.0OR.SPANY.EQ.0.)RETURN 24770	IF(Y(I).LT.YMIN) YMIN=Y(I)	24740	
SPANX=XMX-XHIN 24760 SPANY=YMAX-YMIN 24770 IF(SPANX.E0.0., OR.SPANY.E0.0.) RETURN 24780	5 CONTINUE	24750	
SPANY=YMAX-YMIN 24770 1F(SPANX.EQ.0OR.SPANY.EQ.0.)RETURN 24780	SPANX=XMAX-XMIN	24760	
IF(SPANX.EQ.0OR.SPANY.EQ.0.)RETURN 24780			
8=FEUAT(LINES-1) 24/90		24780	
	B-FLUAI(LINES-1)	24/90	

K03

DO 10 I=N1,N3	24800
IF(Y(1).EQ.0 AND. IGNORE.NE.0) GO TO 9	24810
LY(I)=IFIX(B*(YMAX-Y(I))/SPANY+,5)+1	24820
LX(I)=IFIX(74,*(X(I)-XMIN)/SPANX+,5)+1	24830
GO TO 10	24840
9 LY(1)=0	24850
LX(I)=0	
10 CONTINUE	24860 24870
NZERO=IFIX(B+YMAX/SPANY+.5)+1	24880
IF (NZERO.EQ.LINES.AND.YMIN.GT.O.)NZERO=LINES+1	24890
IF (NZERO.EO.1.AND.YMAX.LT.O.) NZERO=0	24900
WRITE(6,110)	24910
DO 20 I=1.LINES	24920
D0 15_J=1.75	24930
A(J)=BLANK	24940
IF(I.EQ.NZERO)A(J)=DOT	24950
DO 15 K=N1,N3	24960
15 IF(LX(K).EQ.J.AND.LY(K).EQ.I) A(J)=POINT	24970
20 WRITE(6,105)A	24980
WRITE(6,100)	24990
WRITE(6,115)YMAX, YMIN, XMAX, XMIN	25000
100 FORMAT(1H .77(1H-))	25010
105 FORMAT(2H 1.75A1.1HI)	25020
100 FORMAT(1H 27(1H-)) 105 FORMAT(2H 1,75A1,1HI) 110 FORMAT(/HX,77(1H-))	25030
115 FORMAT(" YMAX=",E11.4," YMIN=",E11.4," XMAX=",E11.4,	25040
* XMIN=" F11 4)	25050
* XMIN= ,E11.4) IF(ITABLE.E0.0) GO TO 25	25060
WRITE(6,120) NPOINT, (X(I), Y(I), I=N1, N3)	
120 CODMIT / THE COODDINATES OF THE 12 " DOINTS ADE"//	25070
120 FORMAT(/ THE COORDINATES OF THE ,13, POINTS ARE // *(3(" (",E9.3,",",E9.3,")",2X))) 125 FORMAT(/ 28X," RESPONSE SYSTEM NO.",12)	25080
	25090
123 FURMAI(7 28X, RESPONSE STSTEM NU. ,12)	25100
25 CONTINUE	25110
RETURN	25120
END	25130
	25140
SUBROUTINE KOLMIR(Z, N, DN, CRIT, IPRINT)	25150
DIMENSION X(100),Z(100),DNCRIT(30)	25160
DIMENSION X(100),Z(100),DNCRIT(30) DATA DNCRIT /00.,0.331,337,319,300,285,271,258,249, 222,232,227,220,313,232,231,337,319,300,285,271,258,249,	25170
*.C7C,.CJ7,.CC/,.CCV,.CIJ,.CVV,.CVV,.I7J,.I7V,.100,100,.104,.10C.	25180
*.180,.176,.172,.168,.164,.161 /	25190
*.180,.176,.172,.168,.164,.161 / IF(N.LT.31) CRIT=DNCRIT(N)	25200
IF(N.GT.30) CRIT=.886/SORT(FLOAT(N))	25210
SX:0.	25220
5xx=0.	25230
DO 1 I=1,N	25240
X(1)=Z(1)	25250
SX=SX+X(1)	25260
1 SXX=SXX + X(1)+X(1)	25280
AN=N	25280
BAR=SX/AN	
	25290
SIGMA=SORT((SXX-AN+BAR)/FLOAT(N-1))	25300
IF(SIGMA.EQ.O.)RETURN	25310
D0 6 1=2,N	25320
IF(X(1)-X(1-1))2,6,6	25330
2 TEMP=X(1)	25340
1H=1-1	25350
DO 4 J=1, IM	25360
L=I-J	25370
IF(TEMP-X(L))3,5,5	25380
3 X(L+1)=X(L)	25390
4 CONTINUE	25400
X(1)=TEMP	25410

L03

GO TO 6	25420
5 X(L+1)=TEMP	25430
6 CONTINUE	25440
Ad=N	25450
DN=0.	25460
D0 7 I=1,N	25470
XT=(X(I)-BAR)/SIGMA	25480
FSTAR=SDF(XT)	25480
D1=ABS(FSTAR-FLOAT(I-1)/AN)	25500
D2=ABS(FSTAR-FLOAT(I)/AN)	25510
7 DN=AMAX(ON,D1,D2)	25510
IF(IPRINT.E0.0) RETURN	25510
WRITE(6,100) DN,CRIT	25520
IF(DN.GT.CRIT) WRITE(6,101)	25530
IF(IPRINT.E0.0) RETURN	25530
WRITE(6,100) DN.CRIT	25540
IF(DN.GT.CRIT) WRITE(6,101)	25550
100 FORMAT(" KOLMOGOROV-SMIRNOV STATISTIC FOR TEST OF NORMALITY",	25560
* OF THE RESIDUALS =",F.14" (CRIT.VALUE = ",F6.3,	25570
" ,LT.0.05-REF. ",F.14" (CRIT.VALUE = ",F6.3,	25580
101 FORMAT(" THE STATISTIC INDICATES THAT THE RESIDUALS ARE NOT",	25590
* NORMALLY DISTRIBUTED."/)	25600
RETURN	25610
END	25620

IF(IGO.EO.2) GO TO 1001	6670
IF(ANS.EO.4HBACK) GO TO 1004	6680
IF(ANS.NE.3HYES.AND.ANS.NE.2HNO) GO TO 301	6690
PLOTDP=ANS	6700
GO TO 302	6710
301 WRITE(6,1020)	6720
GO TO 300	6730
302 NCON=NOP-NEWNOP	6740
WRITE(6,750)	6750
WRITE(6,525)	6760
WRITE(6,752) NEWNOP,NCON,NOBS,NDUMMY,NFUNC,NIND	6770
DO 305 I=1,NOP	6780
305 WRITE(6,754)I,F1(1),F2(I),F(I),STEP(I)	6790
WRITE(6,756)	6800
NSAVE=1	6810

L01

-		
	D0 310 1=1,NOBS	6820
	N=LABEL(I)	6830
	IF(N.EO.NSAVE+1) WRITE(6,813)N NSAVE=N	£ 40 #850
	IF(NIND.GT.1) GO TO 308	6850 6860
	IF(ANS16.EQ.3HYES) GO TO 307	6870
	WRITE(6,758)1,X(1),Y(1)	6880
07	GO TO 310 WRITE(6,760)I,×(I),Y(I),WY(I),WYN(I)	6890 6900
	GO TO 310	6910
8(IF(ANS16.EQ. 3HYES) GO TO 309	6920
	WRITE(6,762)1,X(1),Z(1),Y(1) GO TO 310	6930 6940
)9	WRITE(6,764)1,X(1),Z(1),Y(1),WY(1),WYN(1)	6950
10	CONTINUE	6960
	WRITE(6,818) IF(NIND.LT.3) GO TO 312	6970 6980
	WRITE(6,817)(1,1=3,9)	6990
-	DO 311 I=1,NOBS	7000
11	WRITE(6,819)I,(XX(J,I),J=3,NIND)	7010
12	WRITE(6,818) IF(PLOTDP.EQ.2HNO) GO TO 313	7020 7030
1	WRITE(6,765)	7040
	CALL PLOT(X, Y, NUBS, 50)	7050
	IF(NIND.GT.1) WRITE(6,767) IF(NIND.GT.1)CALL PLOT(Z,Y,NOBS,50)	7060 7070
13	WRITE(6,766)MAX, STOPCR	7080
	IF(IQUAD.EQ.1) WRITE(6,768)SIMP	7090
	IF(ANS4.E0.3HYES) WRITE(6,770) IF(PLOTRS.E0.3HYES) WRITE(6,772)	7100 7110
	IF(IPRINT.EQ1) WRITE(6,774)	7120
	IF(IPRINT.EQ.0) WRITE(6,776)	7130
	IF(IPRINT.GT.0) WRITE(6,778) IF(PLT.EQ.3HYES)WRITE(6,780)	7140
	IF(ANS16.EQ.3HYES.AND.IWGHT.GT.1) WRITE(6,781)IWGHT,ANS16B	7150 7160
	WRITE(6,790)	7170
-	GO TO 158	7180
0	NPR=NOP D0 3500 I=1,NOP	7190 7200
	N=NOP+1-I	7210
	IF(STEP(N).NE.0.) GO TO 3501	7220
	NPR=NPR-1 IF(MAX.EQ.1) GO TO 160	7230 7240
	NRUN=NRUN+1	7250
	NVIOL=0	7260
	IF(IPRINT.E01) GO TO 351 WRITE(6,790)	7270
	WRITE(6,525)	7280 7290
	WRITE(6,789)	7300
	CALL NELDR(F, STEP, NOP, FUNC, MAX, IPRINT, STOPCR, NLOOP, IQUAD, SIMP,	7310
1	VAR, LSO, IFAULT) SKIP=.FALSE.	7320 7330
	IF(IFAULT.EQ.5.AND.IGO.EQ.2) GO TO 1001	7340
	IF(IFAULT.EQ.5.AND.AR(1).EQ.4HBACK) GD TO 1004	7350
	LIMITS=0 D0 3510 I=1,NOP	7360 7370
10	FMIN(1)=F(1)	7380
	1F(1FAULT.EQ.3) GO TO 3514	7390
	IF(IFAULT.NE.1.AND.IFAULT.NE.2) GO TO 352 ANS12=2HNO	7400 7410
	IF(IFAULT.E0.2) GO TO 3511	7420
	WRITE(6,791)	7430

M01

NO-18	
NO=18 GO TO 1004	7440 7450
3511 IFLAG=1	7460
00 3512 I=1 NOP	7470
D0 3512 I=1,NOP IF(F2(I).E0.F1(I)) GD TO 3512 DEL=(F2(I)-F1(I))+0.001	7480
DEL = (F2(1) - F1(1)) + 0.001	7490
IF(F(1)-F1(1).GT.DEL.AND.F2(1)-F(1).GT.DEL) GO TO 3512	7500
IFLAG=2	7510
GO TO 3601	7520
3512 CONTINUE	7530
G) TO 3602	7540
3513 IF(IFLAG.ED.2) WRITE(6,793)	7550
IF(IFLAG.EQ.1) WRITE(6,794)	7560
WRITE(6,8401)STOPCR WRITE(6,8402)SIMP	7570
IE (IE AC 60 1) UPITE (4 205)	7580 7590
IF(IFLAG.EQ.1) WRITE(6,795) IF(IFAULT.NE.2) GO TO 3514	7600
3520 WRITE(6,8404)	7610
3520 WRITE(6,8404) CALL READ(1,AR) IF(IGD.ED.2) GD 1001	7620
IF(IGO.EQ.2) GO TO 1001	7630
IF(ANS.EQ.4HBACK.OR.ANS.EQ.2HNO) GO TO 3514	7640
IF (ANS.EQ.2HOK.OR.ANS.EQ.3HYES) GO TO 3521	7650
WRITE(6,1020) GO TO 3520	7660
GO TO 3520	7670
3521 IQUAD=0	7680
GO TO 352	7690

D0 17 I=1,NOP	15960
17 PBAR(1)=0.0	15970
DO 18 I=1,NP1	15980
IF(I-IMAX)19,18,19	15990
19 DO 20 J=1,NOP	16000
20 PBAR(J) = PBAR(J)+G(I,J)	16010
18 CONTINUE	16020
D0 602 J=1,NOP	16030
602 PBAR(J) = PBAR(J)/NAP	16040
DO 21 I=1,NOP	16050
21 PSTAR(I)=A+(PBAR(I)-G(IMAX,I))+PBAR(I)	16060
CALL FUNCTN (PSTAR, HSTAR)	16070
NEVAL=NEVAL+1	16080
IF(IPRINT)57,57,56	16090
56 IF(NPR.LE.5.AND.MOD(NEVAL, IPRINT).E0.0) WRITE(IW, 1010)	16100
*NEVAL, HSTAR, (PSTAR(J), J=1, NPR)	16110

LOZ

IF(NPR.GT.5.AND.MOD(NEVAL, IPRINT).E0.0) WRITE(IW, 101)	16120
•NEVAL, HSTAR, (PSTAR(J), J=1, NPR)	16130
57 IF(HSTAR-HMIN)22,23,23	16140
22 D0 24 I=1,NOP 24 PSTST(I)=C+(PSTAR(I)-PBAR(I))+PBAR(I)	16150 16160
CALL FUNCTN (PSTST, HSTST)	16170
NEVAL=NEVAL+1	16180
IF(IPRINT)60,60,59	16190
59 IF(NPR.LE.5.AND.MOD(NEVAL, IPRINT).EQ.0) WRITE(IW, 1010)	16200
<pre>*NEVAL,HSTST,(PSTST(J),J=1,NPR)</pre>	16210
IF(NPR.GT.5.AND.MOD(NEVAL, IPRINT).EQ.0) WRITE(IW, 101)	16220
<pre>*NEVAL,HSTST,(PSTST(J),J=1,NPR) 60 1F(HSTST-HMIN)25,26,26</pre>	16230 16240
25 DO 27 I=1,NOP	16250
IF(STEP(1).NE.0.0) G(IMAX, I)=PSTST(1)	16260
27 CONTINUE	16270
H(IMAX)=HSTST	16280
GO TO 41	16290
23 DO 28 I=1,NP1	16300
IF(I-IMAX)29,28,29 29 IF(HSTAR-H(I))26,28,28	16310 16320
28 CONTINUE	16330
IF (HSTAR-HMAX) 30, 30, 31	16340
30 DO 32 I=1,NOP	16350
IF(STEP(1).NE.0.0) G(IMAX, I)=PSTAR(1)	16360
32 CONTINUE	16370
HMAX=HSTAR	16380
H(IMAX)=HSTAR 31 DO 33 I=1,NOP	16390
33 PSTST(1)=B+G(IMAX,1)+(1.0-B)+PBAR(1)	16400 16410
CALL FUNCTN (PSTST, HSTST)	16420
NEVAL=NEVAL+1	16430
IF(IPRINT) 63,63,62	16440
62 IF(NPR.LE.5.AND.MOD(NEVAL, IPRINT).EQ.0)WRITE(IH, 1010)	16450
<pre>*NEVAL,HSTST,(PSTST(J),J=1,NPR)</pre>	16460
IF(NPR.GT.5.AND.MOD(NEVAL, IPRINT).E0.0)WRITE(IW, 101) *NEVAL, HSTST, (PSTST(J), J=1, NPR)	16470
63 IF(HSTST-HMAX)35,35,34	16480
35 DO 36 I=1,NOP	16500
IF(STEP(1).NE.0.0) G(1MAX, 1)=PSTST(1)	16510
36 CONTINUE	16520
H(IMAX)=HSTST	16530
GO TO 41	16540
34 DO 38 I=1,NP1 IF(I.EQ.IMIN) GO TO 38	16550
DO 39 J=1,NOP	16560 16570
IF(STEP(J).NE.0.0) G(I,J)=(G(I,J)+G(IMIN,J))/2.0	16580
39 F(J) = G(1, J)	16590
CALL FUNCTN (F,H(1))	16600
NEVAL=NEVAL+1	16610
1. (IPRINT) 38, 38, 65	16620
65 IF(NPR.LE.5.AND.MOD(NEVAL, IPRINT).E0.0)WRITE(IW, 1010)	16630
<pre>*NEVAL,H(I),(F(J),J=1,NPR) IF(NPR.GT.5.AND.MOD(NEVAL,IPRINT).E0.0)WRITE(IW,101)</pre>	16640
*NEVAL,H(1), (F(J), J=1,NPR)	16660
38 CONTINUE	16670
GO TO 41	16680
26 D0 40 I=1,NOP	16690
IF(STEP(1).NE.O.O) G(IMAX, 1)=PSTAR(1)	16700
40 CONTINUE H(IMAX)=HSTAR	16710
41 IF(NEVAL-MAX.GT.0) GO TO 46	16720 16730
	10/30

MOZ

IF(IFLAG.NE.1) GO TO 4100	16740	
IF(LOOP-MAX0(NLOOP, MORE))45,46,45	16750	
100 IF(LOOP-NLOOP)45,46,45	16760	
46 DHCV=0D0	16770	
DMEAN=0D0	16780	
DO 42 I=1,NP1	16790	
IF(H(I).GE.1050) GO TO 45	16800	
42 DMEAN=DMEAN+H(I)	16810	
DMEAN=DMEAN/FLOAT(NP1)	16820	
DO 601 I=1,NP1	16830	
601 DHCV=DHCV+(H(I)-DMEAN)+(H(I)-DMEAN)	16840	
DHCV=1D2+DSQRT(DHCV/FLOAT(NP1))/DMEAN	16850	
DO 53 1=1,NOP	16860	
IF(STEP(1),EQ.0.0) GO TO 53	16870	
F(1)=0.0	16880	
D0 54 J=1,NP1	16890	
54 F(1)=F(1)+G(J,1)	16900	
F(1)=F(1)/NP1	16910	
53 CONTINUE	16920	
CALL FUNCTN (F,FUNC)	16930	
NEVAL =NEVAL+1	16940	
IF(IPRIALLE.0) GO TO 700		
IFTIFRITILE.U. GO TO	16950	
IF(NPR.LE.5.AND.MOD(NEVAL, IPRINT).E0.0)WRITE(IW, 1010)	16960	
+NEVAL, FUNC, (F(J), J=1, NPR)	16970	
IF(NPR.GT.5.AND.MOD(NEVAL, IPRINT).EQ.0)WRITE(IW, 101)	16980	

SX=SX+X(I) 1 SXX=SXX + X(I)+X(I) AN=N BAR=SX/AN	25260 25270 25280 25290	
SIGMA=SQRT((SXX-AN+B\R+BAR)/FLQAT(N-1)) IF(SIGMA.EQ.0.)RETURN D0 6 I=2,N IF(X(I)-X(I-1))2,6,6	25300 25310 25320 25330	
2 TEMP=X(1) IM=1-1 D0 4 J=1,IM	25330 25350 25360	
L=1-J IF(TEMP-X(L))3,5,5 3 X(L+1)=X(L)	25370 25380	
4 CONTINUE X(1)=TEMP	25390 25400 25410	

L03

GO T		25420
5 X(L+	1)=TEMP	25430
6 CONT		25440 25450
DN=0		25460
D0 7	I=1,N	25470
XT=C	X(I)-BAR)/SIGMA	25480
FSTA	R=SDF(XT) BS(FSTAR-FLOAT(I-1)/AN)	25490
	BS(FSTAR-FLOAT(1)/AN)	25500 25510
7 DN=A	MAX1(DN,D1,D2)	25520
IF(I	PRINT.EQ.0) RETURN	25530
	E(6,100) DN,CRIT N.GT.CRIT) WRITE(6,101)	25540 25550
O FORM	AT(" KOLMOGOROV-SMIRNOV STATISTIC FOR TEST OF NORMALITY".	25560
* OF	ATC KOLMOGOROV-SMIRNOV STATISTIC FOR TEST OF NORMALITY", THE RESIDUALS = ",F7.4/" (CRIT.VALUE = ",F6.3,	25570
* ,P	.L1.0.05REF. H.W.LILLIEFORS, J.A.S.A., 62, P399, 1967) /)	25580
* NO	AT(" THE STATISTIC INDICATES THAT THE RESIDUALS ARE NOT", RMALLY DISTRIBUTED."/)	25590 25600
RETU		25610
END		25620
	TION SEQPRO(E, NE)	25630 256+0
	NSION E(100)	25650
N=0		25660
NU=1		25670
NFU=	0 (1).GT.0.)N=1	25680 25690
00 5	J=2,NE	25700
IFCE	(J).GT.0.)N=N+1	25710
5 IF(E	(J).LT.OAND.E(J-1).GT.O.	25720
IE(N	E(J).GT.OAND.E(J-1).LT.O.)NU=NU+1 U.EQ.1) GO TO 20	25730 25740
	LT.NE-N)N=NE-N	25750
M=NE		25760
MM=M NN=N		25770 25780
	E.GT.25) GO TO 25	25790
DO 1	5 I=2,NU DD(1,2).EQ.0) GO TO 10	25800
IFCM	OD(1,2).EQ.0) GO TO 10	25810
	1+1)/2-1 NFU+NBC(MM,KK)+NBC(NN,KK-1)+NBC(MM,KK-1)+NBC(NN,KK)	25820 25830
GO T		25840
0 KK=1		25850
	NFU+2*NBC(MM,KK)*NBC(NN,KK)	25860
5 CONT	INDE RO=FLOAT(NFU)/FLOAT(NBC(NE,M))	25870 25880
RETU	RN	25890
SEOP		25900
RETUR		25910 25920
AN=N		25930
BAR=2	2. *AM*AN/(AM+AN)+1.	25940
SIGH	A=(BAR-1.)*(BAR-2.)/(AM+AN-1.)	25950
SEOP	FLOAT(NU)+.5-BAR)/SIGMA RO=SDF(Z)	25960 25970
RETUR		25980
END		25990
	TION_NBC(N,M)	26000
DOUR	LE PRECISION NC,ND,NE,NF	26010 26020
IE (M	.GT.N) GO TO 6	26030

M03

IF W FO W FO AL FO AL FO TO F	24040
IF(N.EQ.M.OR.M.EQ.O) GO TO 5	26040
IF(M.EQ.1.OR.N-M.EQ.1) GO TO 4	26050
IF(2+M-N.GT.0) GO TO 1	26060
NC=N-M	26070
ND=NC+1	26080
K=M-1	26090
GO TO 2	26100
1 NC=M	26110
ND=M+1	26120
K=N-M-1	26130
2 NE=1	26140
00 3 J=1,K	20140
UU 3 J=1,K	26150
NF=J+1	26160
_ ND=ND*(NC+NF)	26170
3 NE=NE+NF	26180
NBC=ND/NE	26190
RETURN	26200
4 NBC=N	26210
RETURN	26220
5 NBC=1	26230
RETURN	26240
6 NBC=0	26250
RETURN	26260
END	26270
	26280
	20200

IF(IFAULT.E0.5.AND.IGO.E0.2) GO TO 1001 IF(IFAULT.E0.5.AND.AR(1).E0.4HBACK) GO TO 1004 LIMITS=0 DO 3510 I=1,NOP 3510 FMIN(I)=F(I) IF(IFAULT.E0.3) GO TO 3514 IF(IFAULT.RE.1.AND.IFAULT.NE.2) GO TO 352 ANS12=2HNO IF(IFAULT.E0.2) GO TO 3511 WRITE(6,791)

M01

	NQ=18	7440	
	GO TO 1004	7450	
3511	IFLAG=1 D0 3512 I=1.NOP	7460 7470	
	IF(F2(I).E0.F1(I)) GO TO 3512	7480	
	IF(F(1)-F1(1).GT.DEL.AND.F2(1)-F(1).GT.DEL) GO TO 3512	7500	
	IFLAG=2	7510	
3512		7520 7530	
33.6	G) TO 3602	7540	
3513	IF(IFLAG.EQ.2) WRITE(6,793)	7550	
	IF(IFLAG.EQ.1) WRITE(6,794)	7560	
		7570 7580	
	DEL=(F2(1)-F1(1))*0.001 IF(F(1)-F1(1).GT.DEL.AND.F2(1)-F(1).GT.DEL) GO TO 3512 IF(AG=2 GO TO 3601 CONTINUE GO TO 3602 IF(IFLAG.E0.2) WRITE(6,793) IF(IFLAG.E0.1) WRITE(6,794) WRITE(6,8402)STOPCR WRITE(6,8402)STOPCR HRITE(6,8402)STOPCR IF(IFLAG.E0.1) WRITE(6,795) IF(IFLAG.E0.1) WRITE(6,795) IF(IFLAG.E0.2) GO TO 3514 WRITE(6,8404)	7590	
	IF(IFAULT.NE.2) GO TO 3514	7600	
3520	WRITE(6,8404)	7610	
	CALL READ(1, AR)	7620	
	IF(100.E0.27 60 10 1001 IF(ANS E0 4HRACK OR ANS E0 2HND) CO TO 3514	7630 7640	
	IF (ANS.EQ. 2HOK.OR.ANS.EQ. 3HYES) GO TO 3521	7650	
	WRITE(6,1020)	7660	
75.74	GO TO 3520	7670	
3521	IQUAD=0 CO TO 352	7680 7690	
3514	WRITE(6,8401)STOPCR WRITE(6,8402)SIMP IF(IFLAG.E0.1) WRITE(6,795) IF(IFLAG.E0.1) WRITE(6,795) IF(IFAULT.NE.2) GO TO 3514 WRITE(6,8404) CALL READ(1,AR) IF(IGD.E0.24)BACK.OR.ANS.E0.2HNO) GO TO 3514 IF(ANS.E0.2HBACK.OR.ANS.E0.3HYES) GO TO 3521 WRITE(6,1020) GO TO 3520 IGUAD=0 GO TO 3522 NO=18 GO TO 1004 SSRES=0. SUMRES=0. SSY=0.	7700	
	GO TO 1004	7710	
352	SSRES=0.	7720	
	SURRES=0.	7730 7740	
	SSY=0.	7750	
	SY=0.	7760	
	SSYC=0.	7770	
	SYC=0. SYCY=0.	7780 7790	
	IFLAG=0	7800	
	DO 355 LI-1 NEUNC	7810	
	RSS(11)=0.	7820	
	WRSS(II)=0.	7830	
	N2=11M2(11)	7840 7850	
	DO 355 1=N1,N2	7860	
	DO 353 J=1,NIND	7870	
555	G(J)=XX((J, I)	7880	
	THEINETT	7890 7900	
	IF(NIND.EQ.1) CALL MODEL(B,G(1),F,0)	7910	
	IF(NIND.GT.1) CALL MODEL(B,G,F,0)	7920	
	YEST(1)=B REG(1)=Y(1)=YEST(1)	7930 7940	
	1F(WYN(1) FO 0)RES(1)=0	7950	
	WRES(I)=RES(I)+SORT(WYN(I))	7960	
	SUMRES=SUMRES+RES(1)	7970	
	WSUMRS=WSUMRS+WRES(1)	7980	
	SSRES:SSRES+RESO	7990 8000	
	RSS(11)=RSS(11)+RESO	8010	
	SSS(1):=0. WRSS(1):=0. WRSS(1):=0. N1=LIM(1): N2=LIM2(1): D0 353 J=1.N1ND G(J)=XX(J,1): B=Y(1): ITHFUN=11: IF(WIND.GT.1) CALL MODEL(B,G(1),F,0): IF(WIND.GT.1) CALL MODEL(B,G(F,0): YEST(1)=B RES(1)=YEST(1): H(WN(1).E0.0.).RES(1)=0. WRES(1)=RES(1): SUMRES=SUMRES+RES(1): SUMRES=SUMRES+RES(1): SUMRES=SUMRES+RES(1): SSRES=SSRES+RES(1): SSRES=SSRES+RES(1): RES(1)=RES(1): SSRES=SSRES+RES(1): RES(1)=RES(1): SSRES=SSRES=RES(1): RES(1)=RES(1): RES(1)=RES(1): F(ABS(YEST(1)).LT.1E=50) DIFPCT(1)=9.999E+98 IF(ABS(YEST(1)).LT.1E=50) DIFPCT(1)=100.*RES(1)/YEST(1): IF(WN(1).E0.0.) G0 TO 355	8020	
	16(ABS(YEST(1)).LT.1E-50) DIFPCT(1)=9.999E+98	8030	
	1F(WYW(1),FQ.0.) GO TO 355	8040 8050	
		0070	

N01

		1
	SSY=SSY+Y(1)+Y(1)	
	SY=SY+Y(1)	
	SSYC=SSYC+8+8	
	SYC=SYC+B	
	SYCY=SYCY+B+Y(1)	
355		
	NOBSS=NOBS	
	NOBS=NOBS-NDUMMY	
	YBAR=SY/FLOAT(NOBS)	
	SSTOT=SSY-SY+SY/FLOAT(NOBS)	
	BENORS	
	R=ABS(SYCY-SY+SYC/B)/SORT(ABS((SSY-SY+SY/B)+(SSYC-SYC+SYC/B)))	
	RSD=100+R+R	
	SSREG=SSTOT-SSRES	
	RESBAR=SUMRES/FLOAT(NOBS)	
	WRSBAR=WSUMRS/FLOAT(NOBS)	
	NRGDF = NEWNOP - 1	
	IF(WRGDF.LE.0) WRGDF=1	
	NRSDF=NOBS-NEWNOP	
	NTODE = NOBS - 1	
	REGMSQ=SSREG/FLOAT(NRGDF)	
	RESMSQ=SSRES/FLOAT(NRSDF)	
	RSMSG=FUNC/FLOAT(NRSDF)	
	SOVRES=SORT(RESMSO)	
	3DVRS=SCT(hRSMSD)	
	OVERF=REGMS0/F+ MS0	
	T=1.95	
	IF (NRSOF.LE. 30.	
	WRITE(6,790)	
	WRITE(6,525)	
	NOBS=NOBSS	

IF(NPR.GT.5.AND.MOD(NEVAL, IPRINT).EQ.0)WRITE(IW, 101)	16650	
<pre>*NEVAL,H(I),(F(J),J=1,NPR)</pre>	16660	
38 CONTINUE	16670	
GO TO 41	16680	
26 DO 40 I=1.NOP	16690	
IF(STEP(1).NE.0.0) G(IMAX, I)=PSTAR(1)	16700	
40 CONTINUE	16710	
H(IMAX)=HSTAR	16720	
41 IF(NEVAL-MAX.GT.0) GO TO 46	16730	

MOZ

	IF(IFLAG.NE.1) GO TO 4100 IF(LOOP-MAXO(NLOOP,MORE))45,46,45	16740 16750
46	IF(LOOP-NLOOP)45,46,45 DHCV=0D0 DHEAN=0D0	16760 16770 16780
	D0 42 I=1,NP1 IF(H(I).GE.1050) G0 T0 45	16790 16800
4Z	DMEAN=DMEAN+H(I) DMEAN=DMEAN/FLOAT(NP1)	16810 16820
01	DO 601 I=1,NP1	16830 16840
	DHCV=1D2+DSQRT(DHCV/FLOAT(NP1))/DMEAN D0 53 I=1,NOP	16850 16860
	F(STEP(1).E0.0.0) GO TO 53 F(1)=0.0	16870 16880
54	D0 54 J=1,NP1 F(I)=F(I)+G(J,I)	16890
53	F(I)=F(I)/NP1 CONTINUE	16910 16920
	CALL FUNCTN (F,FUNC) NEVAL=NEVAL+1	16930 16940
	IF(IPRIA:LE.0) GO TO 700 IF(NPR.LE.5.AND.MOD(NEVAL, IPRINT).EQ.0)WRITE(IW,1010)	16950
	<pre>*NEVAL,FUNC,(F(J),J=1,NPR) IF(NPR.GT.5.AND.MODCNEVAL,IPRINT).EQ.0)WRITE(IW,101) MPVALENTE(IW,101)</pre>	16970 16980
700	•NEVAL_FUNC,(F(J),J=1,NPR) IF(NEVAL-MAX) 44,44,43 G0 T0 67	16990 17000 17010
67	WRITE(IN, 102)MAX FORMAT(40H NUMBER OF FUNCTION EVALUATIONS EXCEEDS ,14)	17020 17030
	IF(NVIOL.GT.10) WRITE(IW,8031)NVIOL WRITE(IW.103)DHCV	17040 17050
03	FORMAT(" COEFFICIENT OF VARIATION (PCT) OF FUNCTION VALUES", •" OF LAST SIMPLEX" ,D10.4)	17060 17070
940	IF(NPR.LE.5.)WRITE(IW,1040)(F(I),I=1,NPR) FORMAT("CENTROID ",5E12.6)	17080 17090
04	IF(NPR.GT.5)WRITE(IW,104)(F(I),I=1,NPR) FORMAT(CENTROID ",5E12.6/(12X,5E12.6))	17100
	HRITE(IH,103)FUNC FORMAT(FUNCTION VALUE AT CENTROID ,D15.8 /) IFAULT = 1	17120 17130
	RETURN IF(DHCV-DBLE(STOPCR))72,48,48	17140 17150 17160
	IFLAG=0	17170 17180
72	GO TO 45 IF(IPRINT)47,70,70	17190 17200
70	WRITE(IW, 106)DHCV FORMAT(" + INITIAL EVIDENCE OF CONVERGENCE COEFF. OF VARIATION",	17210 17220
	<pre>* (PCT) ",D10.4) IF(NPR.LE.5) WRITE(IW,1010)NEVAL,FUNC,(F(I),I=1,NPR)</pre>	17230 17240
	IF(NPR.GT.5) WRITE(IW,101)NEVAL,FUNC,(F(I),1=1,NPR) IF(IFLAG)49,50,49	17250 17260
	JFLAG=1 DMEANS=DMEAN FUNCS=FUNC	17270 17280
	LOOP=0 GO TO 45	17290 17300 17310
49	TF(OABS(1D1*(DMEANS-DMEAN)/DMEANS).GT.DBLE(STOPCR).OR. *DABS(1D2*(FUNC-FUNCS)/FUNCS).GT.DBLE(STOPCR)) GO TO 51	17320 17330
	NEVAL 1=NEVAL IF(ICUAD.NE.0.0R.H(IMIN).GT.FUNC) GO TO 4905	17340 17350

NOZ

FUNC=H(IMIN) D0 4900 J=1,NOP	17360 17370
900 F(1)=G(1MIN.J)	17380
905 CONTINUE	17390
IF(IPRINT)74,73,73	17400
73 IF(NVIOL.GT.0) WRITE(IW, 107)NEVAL, NVIOL	17410
107 FORMAT(/ PROCESS CONVERGES AFTER ,14, "-",13," FUNCTION",	17420
* EVALUATIONS /)	17430
IF(NPR.LE.5)WRITE(IW.1080)(F(I).I=1.NPR)	17440
1080 FORMAT(MINIMUM AT ,SE12.6)	17450
1080 FORMAT(MINIMUM AT ,5E12.6) IF(NPR.GT.5) WRITE(IW,108)(F(I),I=1,NPR)	17460
108 FORMAT(MINIMUM AT ,5E12.6/(12X,5E12.6))	17470
WRITE(IW, 109)FUNC	17480
WRITE(IW, 103)DHCV	17490
109 FORMAT(/ 26H MINIMUM FUNCTION VALUE ,D15.8/)	17500
74 CONTINUE	17510
IF(IPRINT.GT.O.AND.NVIOL.GT.NPR) WRITE(IW,8035)	17520
8035 FORMAT((FUNC.VALUES GREATER THAN 1D+50 INDICATE VIOLATION ,	17530
• OF PARAMETER CONSTRAINS))	17540
IF(IPRINT.GE.O.AND.NVIOL.GT.1) WRITE(IW,8031)NVIOL	17550
<pre>S031 FORMAT(* NO. OF VIOLATIONS OF PARAMETER CONSTRAINS, * DURING MINIMIZATION ,14)</pre>	17560
· DURING HINIHIZATION ,14)	17570
IF(IQUAD)200,75,200	17580
75 RETURN	17590
200 LIMITS=0 ALOW=5HLOWER	17600 17610
ALUH=SHLUHER AUPP=SHUPPER	
ICNSTR=0	17620 17630
DO 806 I=1,NOP	17640
IF(F1(1).EQ.F2(1)) GO TO 806	17650
DEL=(F2(1)-F1(1))+0.001	17660
TELECTIVESTICE OF TO .	17670

	SIGMA=(BAR-1.)*(BAR-2.)/(AM+AN-1.)	25950
	Z= (FLOAT(NU)+.5-BAR)/SIGMA SEOPRO=SDF(Z)	25960
	RETURN	25980
	END	25990
C		26000
	FUNCTION NBC(N,M) Double Precision NC,ND,NE,NF	26010
	IF(M.GT.N) GO TO 6	26030

_		
	IF(N.EQ.M.OR.M.EQ.0) GO TO 5	26040
	IF(M.EQ.1.0R.N-M.EQ.1) GO TO 4	26050
	IF(2+M-N.GT.0) GO TO 1	26060
	NC=N-M	26070
	ND=NC+1 K=M-1	26080 26090
	GO TO 2	26100
	1 NC=M	26110
	ND=M+1	26120
	K=N-M-1	26130
	2 NE=1	26140
	DO 3 J=1,K NF=J+1	26150 26160
	ND=ND+(NC+NF)	26170
	3 NE=NE+NF	26180
	NBC=ND/NE	26190
	RETURN	26200
	4 NBC=N RETURN	26210 26220
	5 NBC=1	26230
	RETURN	26240
	6 NBC=0	26250
	RETURN	26260
С	END	26270
L	FUNCTION SDF(X)	26280 26290
	AX=ABS(X)	26300
	T=1,/(1.+.2316419*AX) D=.3989423*EXP(-X*X/2.)	26310
	D=.3989423+EXP(-X+X/2.)	26320
	SDF=1D+T+((((1.330274+T-1.821256)+T+1.781478)+T-	26330
	*.3565638)*T*.3193815) IF(X)1,2,2	26340
	1 SDF=1SDF	26350 26360
	2 RETURN	26370
	END	26380
С	***************************************	26390
	SUBROUTINE PROBICE,NE,M,P) DIMENSION E(100)	26400
	NS=0	26410 26420
	N=0	26430
	DO 5 J=1,NE	26440
	5 IF(E(J).GT.O.) N=N+1	26450
	IF(N.LT.NE-N) N=NE-N M=NE-N	26460
		26470 26480
	IF(NE.GT.25) GO TO 15	26490
	DO 10 J=1,MM	26500
	10 NS=NS+NBC(ME, J-1)	26510
	P=FLOAT(NS)*(2.**(-NE))	26520
	RETURN 15 Z=FLOAT(2+M+1-NE)/SQRT(FLOAT(NE))	26530
	P=SDF(Z)	26540 26550
	RETURN	26560
-	END	26570
С		26580
	SUBROUTINE SEARCH(FNEW, STPNEW, SSMIN, ANS16, NPOINT, NEVAL, •NEWNOP, STEP, NP)	26590
	DIMENSION FNEW(7), F1(20), F2(20), STPNEW(7), P1(75), P2(75), XX(9, 100),	26600 26610
	•SS(75), SSS(10,10), PDEV(10,20), FF(7), Y(100), LABEL(100).	26620
	*WYN(100), DEL(7), PP(7, 12), IS(7), STEP(20), A(10), AL(9), CON(10, 10)	26630
	COMMON /DATA/ XX.Y.WYN.NOBS	26640
	COMMON /CONSTR/LIMITS	26650

N03

	COMMON /PARLIM/ F1.F2 COMMON /FUNNUM/ITHFUN COMMON /84/ NIND COMMON /86/ NOP COMMON /86/ NOP	26660 26670 26680 26690 26700
	DATA AL/3H1 , 3H2 , 3H3 , 3H4 , 3H5 , 3H6 , 3H7 , 3H8 , 3H9 /	26710
	LIMITS=0	26720
	IF(NEWNOP-2) 5,50,135 CONTINUE	26730 26740
,	DO 10 I=1 NOP	26750
	FNEW(1)=F1(1)	26760
	STPNEW(1)=STEP(1)	26770
10	IF(F1(1).NE.F2(1)) IP=1	26780
	DEL(IP)=(F2(IP)-F1(IP))/74. D0 20 1=1,75	26790
	FNEW(IP)=F1(IP)+FLOAT(I-1)+DEL(IP)	26800 26810
	CALL LSQ(FNEW, SS(1))	26820
20	P1(I)=FNEW(IP)	26830
	IF(ANS16.EQ.2HNO) WRITE(6,1000) IP, F1(IP), F2(IP)	26840
	IF(ANS16.EQ. 3HYES) WRITE(6, 1010) IP, F1(IP), F2(IP)	26850
	CALL PLOT(P1,SS,75,50) IF(ANS16.EQ.2HNO) WRITE(6,1020)	26860 26870
	IF(ANS16.EQ. 3HYES) WRITE(6, 1025)	26880
	D0 30 1=1,37	26890
	J=1+37	26900
30	WRITE(6, 1026)1, SS(1), P1(1), J, SS(J), P1(J)	26910
	HRITE(6,1027) ISAVE=1	26920 26930
	SSMIN=SS(1)	26940
	00 40 1=2,75	26950
	IF(SS(I).GE.SSMIN) GO TO 40	26960
	SSMIN=SS(1)	26970

YESI(1)=B RES(1)=Y(1)-YEST(1)	
IF(WYN(1).EQ.0.)RES(1)=0.	
WRES(1)=RES(1)+SORT(WYN(1)) SUMRES=SUMRES+RES(1)	
WSUMRS=WSUMRS+WRES(1)	
RESO=RES(1) • RES(1)	
SSRES=SSRES+RESO RSS(11)=RSS(11)+RESO	
WRSS(11)=WRSS(11)+WYN(1)*RESO	
IF(ABS(YEST(1)).LT.1E-50) DIFPCT(1)=9.999E+98 IF(ABS(YEST(1)).GE.1E-50) DIFPCT(1)=100.*RES(1)/YEST(1)	
IF(WYN(I).EQ.O.) GO TO 355	

N01

and the second	
GO TO 3612	8680
3610 WRITE(6,8010)	8690
DO 3611 I=1,NOP	8700
3611 WRITE(6,8011)1,F1(1),F2(1),FSAVE(1),F(1)	8710
WRITE(6,8012)	8720
GO TO 3600	8730
3612 CONTINUE	8740
ALOW= SHLOWER	8750
AUPP=SHUPPER	8760
ICNSTR=0	8770
00 3621 1=1,NOP	8780
D0 3621 I=1,NDP IF(F1(I),E0,F2(I)) G0 T0 3621	8790
DEL=(F2(1)-F1(1))+0.001	8800
1F(F(1)-F1(1).GT.DEL) GO TO 3620	8810
ICNSTR=ICNSTR+1	8820
WRITE(6,8030) ALOW, I	8830
3620 IF(F2(1)-F(1).GT.DEL) GO TO 3621	8840
ICNSTR=ICNSTR+1	8850
WRITE(6,8030) AUPP,1	8860
3621 CONTINUÉ	8870
IF(ICNSTR.EQ.0) GO TO 3622	8880
WRITE(6,8031)	8890
CALL READ(1, AR)	8900
IF(IGO.EO.2.OR.ANS.EO.4HBACK) WRITE(6,844) NRUN IF(IGO.EO.2) GO TO 1001	8910
IF(1G0.EQ.2) GO TO 1001	8920
IF(ANS.EQ.4HBACK) GO TO 1004	8930
3622 CONTINUE	8940
IF(IFLAG.NE.0) GO TO 3513	8950
UNITE A STATATE CARTA CAL VALA BEAMAA BEAMAA BEAMAA	1474

IF(NPR.LE.5) WRITE(IW, 1010)NEVAL, FUNC, (F(I), I=1, NPR) IF(NPR.GT.5) WRITE(IW, 101)NEVAL, FUNC, (F(I), I=1, NPR)	
47 IF(IFLAG)49,50,49	
50 IFLAG=1	
51 DMEANS=DMEAN	
FUNCS=FUNC	
LOOP=0	
GO TO 45	
49 IF(DABS(1D1+(DMEANS-DMEAN)/DMEANS).GT.DBLE(STOPCR).OR.	
DABS(1D2(FUNC-FUNCS)/FUNCS).GT.DBLE(STOPCR)) GO TO 51	
NEVAL 1=NEVAL	
IF(ICUAD.NE.O.OR.H(IMIN).GT.FUNC) GO TO 4905	

HOZ

	FUNC=H(IMIN)	17360
1000	D0 4900 J=1,NOP F(1)=G(IMIN,J)	17370
	CONTINUE	17390
	IF(IPRINT)74,73,73	17400
73	IF(NVIOL.GT.O) WRITE(IW, 107)NEVAL, NVIOL	17410
107	FORMAT(/ PROCESS CONVERGES AFTER ,14, - ,13, FUNCTION ,	17420
	• EVALUATIONS /)	17430
	IF(NPR.LE.5)WRITE(IW, 1080)(F(I), I=1, NPR)	17440
1080	FORMAT(MININUM AT , SE12.6)	17450
10.	IF(NPR.GT.5) WRITE(IW,108)(F(I),I=1,NPR) FORMAT(MINIMUM AT .5E12.6/(12X.5E12.6))	17460
100	FORMAT(MINIMUM AT ",5E12.6/(12X,5E12.6)) WRITE(IW,109)FUNC	17480
	WRITE(IW, 103)DHCV	17490
109	FORMAT(/ 26H MINIMUM FUNCTION VALUE , D15.8/)	17500
74	CONTINUE	17510
	IF(IPRINT.GT.O.AND.NVIOL.GT.NPR) WRITE(IW,8035)	17520
8035	FORMATC (FUNC. VALUES GREATER THAN 10+50 INDICATE VIOLATION",	17530
	• OF PARAMETER CONSTRAINS))	17540
	IF(IPRINT.GE.O.AND.NVIOL.GT.1) WRITE(IW,8031)NVIOL	17550
8031	FORMAT(" + NO. OF VIOLATIONS OF PARAMETER CONSTRAINS", • DURING MINIMIZATION", 14)	17560
	IF(10UAD)200,75,200	17570 17580
75	RETURN	17590
	LIMITS=0	17600
	ALOW=5HLOWER	17610
	AUPP=SHUPPER	17620
	ICNSTR=0	17630
	DO 806 I=1,NOP	17640
	IF(F1(1).EQ.F2(1)) GO TO 806	17650
	DEL=(F2(I)-F1(I))*0.001	17660
	IF(F(I)-F1(I).GT.DEL) GO TO 6. ICNSTR=ICNSTR+1	17670
	IF(ICNSTR.NE.1) GO TO 800	17680 17690
	WRITE(6,8000) FUNC, (F(L), L=1, NPR)	17700
	WRITE(6,8005)	17710
800	WRITE(6,8010) ALOW, F1(1), I, F(1)	17720
80Z	IF(F2(I)-F(I).GT.DEL) GO TO 306	17730
	ICNSTR=ICNSTR+1	17740
	IF(1CNSTR.NE.1) GO TO 804	17750
	WRITE(6,8000) FUNC,(F(L),L=1,NPR)	17760
	WRITE(6,8005) WRITE(6,8010) AUPP,F2(1),1,F(1)	17770
	CONTINUE	17790
000	IF(ICNSTR.EQ.0) GO TO 814	17800
	IF(IPRINT.EQ1.AND.NVIOL.GT.') WRITE(IW,8031)NVIOL	17810
	WRITE(6,8015)	17820
808	CALL READ(1, AR)	17830
	IF(AR(1).E0.4HBACK.OR.IGO.E0.2) GO TO 810	17840
	IF(AR(1).EQ.2HNO) GO TO 812	17850
	IF(AR(1).EQ.3HYES) GO TO 814 WRITE(6,8030)	17860
	GO TO 808	17870 17880
810	IFAULT=5	17890
	WRITE(6,8020)	17900
	RETURN	17910
812	IQUAD=0	17920
	WRITE(6,8025)	17930
	RETURN	17940
814	CONTINUE FORMATC • THE RESIDUAL SUM OF SQUARES MINIMUM (",D12.6,	17950
	FURNALL F OF RESIDUAL SUB OF SUBARES FURIFUE (117.6.	17960

8005 FORMAT(" *"/" * SEEM TO BE CONSTRAINED BY THE"/ *") 8010 FORMAT(" *",5X,A5," LIMIT (",E11.5,") OF PAR.NO.",12,	17980
* (",E11.5,")") 8015 FORMAT(" *"/" * A STATISTICAL EVALUATION OF THE PARAMETERS UND	ER", 18000
 THESE CONDITIONS MAY FAIL."/ ENTER BIO OR BII IF YOU WANT A NEW RUN WITH DIFFERENT"/ PARAMETER LIMITS OR DIFFERENT INITIAL PARAMETER ESTIMATES" 	18020 18030 / 18040
*" * ENTER -YES- IF YOU WANT TO CONTINUE OR"/ *" * ENTER -NO- IF YOU WANT TO CONTINUE BUT WITHOUT A STATISTIC	AL", 18050
<pre>* EVALUATION'//) 8020 FORMAT(/1X,77(1H-)//) 8025 FORMAT(/'* REMEMBER YOUR INPUT UNDER REQUEST NO.3 HAS NOW",</pre>	18070 18080 18090
* BEEN CHANGED TO -NO-"/) 8030 FORMAT(/" INPUT ERROR, YOUR ANSWER MUST BE ONE OF THE FOLLOWIN	G"/ 18100
• FOUR, Y N B R - TRY AGAIN - 7) IF(IPRINT)233,232,232 232 WRITE(IN,301)	18120 18130 18140
301 FORMAT(/1X,13(1H+), " FITTING OF QUADRATIC SURFACE IN REGION OF *" MINIMUM ",13(1H+) /)	, 18150 18160
233 NEVAL=0 NFIX=0 B=SIMP/100.	18170 18180 18190
SIMP2=FUNC+B SIMP2=AMAX1(B,SIMP2)	18200 18210
DO 201 [=1,MP1 204 [F(H(1),GE,1D50) GO TO 2030 TEST-DABS(H(1)-FUNC)	18220 18230 18240
IF(TEST-SIMP2)202,201,201 202 D0 203 J=1,NOP	18250

C

N03

-						
1	COMMON /PARLIM/ F1,F2			26660		
1	COMMON /FUNNUM/ITHFUN			26670		
	COMMON /B4/ NIND			26680		
	COMMON /B6/ NOP			26690		
	CONNON /B10/LABEL			26700		
	DATA AL/3H1 , 3H2 , 3H3 , 3H4 , 3H5 , 3H6 , 3H7 , 3H LIMITS=0	8 , 3H9	1	26710		
	IF(NEWNOP-2) 5,50,135			26720 26730		
	CONTINUE			26740		
	DO 10 [=1 NOP			26750		
	FNEW(1)=FI(1)			26760		
	STPNEN(I)=STEP(I)			26770		
	IF(F1(1).WE.F2(1)) IP=1			26780		
	DEL(IP)=(F2(IP)-F1(IP))/74.			26790		
	DO 20 I=1,75			26800		
	FNEW(IP)=F1(IP)+FLOAT(I-1)+DEL(IP)			26810		
	CALL LSQ(FNEW,SS(I)) P1(I)=FNEW(IP)			26820 26830		
	IF(ANS16.E0.2HNO) WRITE(6,1000) IP, F1(IP), F2(IP)			26840		
1	IF(ANS16.EQ. 3HYES) WRITE(6, 1010) IP, F1(IP), F2(IP)			26850		
1	CALL PLOT(P1, SS, 75, 50)			26860		
1	IF(ANS16.EQ.2HNO) WRITE(6,1020)			26870		
	IF(ANS16.EQ. 3HYES) WRITE(6, 1025)			26880		
1	DO 30 I=1,37			26890		
				26900		
	WRITE(6, 1026)1, SS(1), P1(1), J, SS(J), P1(J)			26910		
	WRITE(6,1027) ISAVE=1			26920 26930		
	SSMIN=SS(1)			26940		
	DO 40 1=2,75			26950		
1	IF(SS(1).GE.SSMIN) GO TO 40			26960		
\$	SSMIN=SS(1)			26970		
	ISAVE=1			26980		
	CONTINUE			26990		
	FNEW(IP)=P1(ISAVE)			27000		
	STPNEH(1P)=DEL(1P)/2. RETURN			27010		
	IP1=7			27020 27030		
	0 55 I=1,NOP			27040		
ì	FNEW(1)=F1(1)			27050		
	STPNEW(1)=STEP(1)			27060		
1	IF(1.GT.IP1) GO TO 55			27070		
	IF(F1(1).NE.F2(1)) IP1=1			27080		
	IF(F1(1).NE.F2(1)) IP2=1			27090		
	DEL(IP1)=(F2(IP1)-F1(IP1))/9.			27100		
	DEL(IP2)=(F2(IP2)-F1(IP2))/9. ISAVE=1			27110 27120		
	JSAVE=1					
	SSNE-1 SSNIN=1.E99			27130 27140		
	SSMAX=0.			27150		
	DO 60 J=1,10			27160		
1	FNEW(IP1)=F1(IP1)+FLOAT(J-1)+DEL(IP1)			27170		
	P1(J)=FNEW(IP1)			27180		
ļ	DO 60 1=1,10			27190		
2	FNEW(IP2)=F1(IP2)+FLOAT(I-1)+DEL(IP2)			27200		
	P2(I)=FNEW(IP2) CALL LSQ(FNEW,SSS(I,J))			27210		
1	IF(SSS(I,J).GE.SSMIN) GO TO 60			27220 27230		
é	SSMIN=SSS(1,J)			27240		
	ISAVE=1			27250		
- 12						
	JSAVE=J			27260		

FNEW(IP1)=P1(JSAVE) 27280	
FNEN(IP2)=P2(ISAVE) 27290	
STPNEH([P1)=DEL([P1)/2. 27300	
STPNEN(IP2)=DEL(IP2)/2. 27310	
D=(SSMAX-SSMIN)/9. 27320	
D0 65 I=1,10 27330	
65 A(I)=SSMIN+(I-1)+D 27340	
A(10)=A(10)+0.01+D 27350	
WRITE(6,1030)IP1,(I,P1(I),I=1,10) 27360	
WRITE(6,1040)IP2,(I,P2(I),I=1,10) 27370 IF(ANS16.EQ.2HNO) WRITE(6,1050) 27380	
IF(ANS16.EQ.2HNO) WRITE(6,1050) 27380	
IF(ANS16.EQ.3HYES) HRITE(6,1060) 27390	
HRITE(6,1070)1P1,1P2 27400	
HRITE(6,1075)(1,1=1,10) 27410	and the second
D0 67 [=1,10 27420	
HRITE(6, 1080) (SSS(1, J), 1=1, 9, 2) 27430	
67 WRITE(6,1085)(SSS(1,J),J=2,10,2) D0 120 J=1,10 D0 120 J=1,10 IF(J.E0.1) G0 T0 70 Z7460 Z7470	
D0 120 1=1,10 27460	
IF(J.EQ.1) GO TO 70 27770	
IF(J.EQ.10) GO TO 80 27480	
PDEV(I, J)=SIGNOF(SSS(I, J+1)-SSS(I, J-1)) 27490	
GO TO 90 27500	
70 PDEV(1,J)=SIGNOF(4.+SSS(1,J+1)-3.+SSS(1,J)-SSS(1,J+2)) 27510	
GO TO 90 27520	
80 PDEV(1, J)=SIGNOF(-4.*SSS(1, J-1)+3.*SSS(1, J)+SSS(1, J-2)) 27530	
90 K=J+10 27540	
IF(1.E0.1) G0_T0_100 27550	
IF(1,E0,10) G0 T0 110 27560	

IF(SDV.EQ.0.) CV=0.	8440
360 WRITE(6,802)1,F1(1),F2(1),FSAVE(1),F(1),SDV,CV	8450
WRITE(6,803)	8460
DO 361 I=1,NOP	8470
E=T+FAC+SORT(VAR(1))	8480
E1=F(1)-E	8490
E2=F(1)+E	8500
361 [F(F1(1).NE.F2(1)) WRITE(6.805)1,E1,E2	8510
WRITE(6,8050)	8520
SOO CONTINUE	8530
IFLAG=0	8540
GO TO 3602	8550
5601 WRITE(6,792)	8560
602 CALL PLACE(F1,F2,F,65,NPR,A) IF(NPR.EQ.1) WRITE(6,8051) IF(NPR.GT.1) WRITE(6,8052)	8570
IF(NPR.EQ.1) WRITE(6,8051)	8580
IF(NPR.GT.1) WRITE(6,8052)	8590
DO 3605 J=1, NPR	8600
IF(F1(J).EQ.F2(J)) GO TO 3605	8610
N1=65+(J-1)+1	8620
N2=N1+64	8630
WRITE(6,8053) J,(A(1),1=N1,N2)	8640
GOS CONTINUE	8650
IF(IFLAG.NE.O.AND.ANS16.EQ.2HNO) WRITE(6,8054)FUNC	8660
IF(IFLAG.NE.O.AND.ANS16.NE.2HNO) WRITE(6,8055)FUNC	8670

5610 MiTTEr Ca BOTO 5600 5611 District Tail Nop 5600 5611 District Tail Nop 5600 5611 District Tail Nop 5700 5612 District Tail Nop 5700 5611 District Tail Nop 5700 5612 District Tail Nop 5800 5612 District Distric District District District District District Distric		
5610 MailTEG, 4010) 860 5611 MailTEG, 4010) 870 5611 MailTEG, 4010) 870 5611 MailTEG, 4010) 870 5612 Gotto Science 870 5613 Gotto Science 870 5614 Gotto Science 870 5615 Gotto Science 870 5616 Gotto Science 870 5617 Gotto Science 870 5618 Gotto Science 870 5619 Gotto Science 870 5619 Gotto Science 870 5620 Gotto Science 870 5621 Gotto Science 850 5621		
5610 MailTEG, 4010) 860 5611 MailTEG, 4010) 870 5611 MailTEG, 4010) 870 5611 MailTEG, 4010) 870 5612 Gotto Science 870 5613 Gotto Science 870 5614 Gotto Science 870 5615 Gotto Science 870 5616 Gotto Science 870 5617 Gotto Science 870 5618 Gotto Science 870 5619 Gotto Science 870 5619 Gotto Science 870 5620 Gotto Science 870 5621 Gotto Science 850 5621		
10 10 15 14 14 14 14 14 14 15<		
5611 MRITEC 6, 400131, Fr(1), F2(1), FSAVE(1), F(1) 8710 MRITEC 6, 40013 8720 5612 CONTINUE 8720 ALQUESH GUER 8750 ALQUESH GUER 8760 ALQUESH GUER 8760 ALQUESH GUER 8760 ID 8760		
GO TD 3600 6730 ALDE-SHUDEE 5730 ALDE-SHUDEE 5730 ALDE-SHUDEE 5730 DO 3621 LE1, ADF 5730 DO 3621 LE1, ADF 5730 DO 3621 LE1, ADF 5730 DE 3620 LECTO SECONDAUM. 5700 DE 3620 LECTO SECONDAUM. 5800 DE 201 FEIL, ADF SECONDAUM. 5800 BE20 LECTO SECONDAUM. 5800 BE20 LECTO SECONDAUM. 5800 BE21 FEIL, ADF 5800 BE21 FEIL 5800 BE21 FEIL, ADF 5900 BE21 FEIL, ADF <td></td> <td></td>		
1612 CONTINUE 8740 A (Destrato) 8770 DD (Soft) 1::1, ADD 8770 DE (Soft) 1::1, ADD 8700 DE (Soft) 1::1, ADD 8800 Itwitter (Destrato) 8900 Itwitter (Desto		
A LOB-SHOWER ALDE-SHOWER TO TREE TO TREE TO TREE TO TREE TO TREE TO TREE TO TREE ALDE-SHOWER HITECS, 60307 ALOW,I SCOTT ALCO, TABLES TO TO 3621 TREE TO TREE TO TREE		
AuPD=:HuPPER TeleFite-0 DF 26:1 1:0.10 ^D DF 26:1 1:0.10 ^D TeleFiteFiteFiteFiteFiteFiteFiteFiteFiteFit		
CHRITR-0 8770 DD 5627 11:1, MDP 8770 TECHT27:10:17(1):10:001 T0 3621 8770 TECHT27:10:17(1):10:001 T0 3621 8770 TEKET27:10:17(1):10:001 T0 3621 8770 TEKET27:10:17(1):10:001 T0 3621 8870 TEKET27:10:17(1):10:001 T0 3621 8870 WHITEC6, 8030 ALON, 1 8870 Yourse, 8030 ALON, 1		
D0 3621 LE1 MOP TF(FL) EG F2(1); GU TO 3621 DEL:(F2):DF(1):D0,D01 TOUSTRET:MSTR-1 WEITE(6,8030 ALOW,I SECONTRET:MSTR-1 WEITE(6,8030 ALOW,I SECONTRE:C		
IFCF1(1), 60, 62(1)), 60, 10, 3621 870 DELC(75)-F1(1)(17), 00, 10, 10, 20 880 ICMSTR-ITMSTR-1 880 IFC7(1), 67, DEL), 60, 10, 3621 880 IFC7(1), 67, DEL), 60, 10, 3622 880 IFC7(1), 67, DEL, 00, 00, 10, 3622 880 IFC7(1), 60, 10, 30, 00, 10, 3622 880 IFC7(1), 60, 00, 10, 3622 880 IFC7(1), 60, 00, 10, 364 890 IFC7(1), 60, 00, 10, 3513 880 IFC7(1), 60, 00, 10, 3513 880 IFC7(1), 60, 00, 10, 1004 890 IFC7(1), 60, 00, 10, 1004 890 IFC7(1), 60, 00, 10, 3513 890 IFC7(1), 60, 10, 00, 10, 351 890 IFC7(1), 60, 00, 10, 1513 890 IFC7(1), 60, 10, 00, 10, 351 890 IFC7(1), 60, 10, 30, 10, 10, 10, 10, 10, 10, 10, 10, 10, 1	D0 3621 1=1,NDP	
IF(F(1)-F(1), GT, DEL) GD TO 3620 8810 3620 INERTIFE (6, 0000 ALDH, 1 8820 3620 INERTIFE (6, 0000 ALDH, 1 8820 METTE (6, 0000 ALDH, 1 8850 9620 INERTIFE (6, 0000 ALDH, 1 8850 METTE (6, 0000 ALDH, 1 8850 9621 CONTINUE 8850 METTE (6, 0000 ALDH, 1 8850 9621 CONTINUE 8850 METTE (6, 0000 ALDH, 1 8850 9621 CONTINUE 8850 F(10KSTR, E0, 0) E0 T0 501 8850 1F(10KSTR, E0, 0) E0 T0 5313 8950 1F(10KLE, 60, 0) E0 T0 5313 8950 METTE (6, 0000 ANDH, NE, 1) NETTE (6, 0000 ANDH, 1) 8950 1F(10KLE, 60, 0) E0 T0 3513 8950 METTE (6, 0000 ANDH, 1) NETTE (6, 0000 ANDH, 1) 8950 1F(10KLE, 60, 1) METTE (6, 0000 ANDH, 1) 8950 1F(10KLE, 60, 1) METTE (6, 0000 ANDH, 1) 8950 1F(10KLE, 60, 1) NETTE (6, 0000 ANDH, 1) 8950 1F(10KLE, 60, 1) NETTE (6, 0000 ANDH, 1) 8950 1F(1	IF(F1(1).E0.F2(1)) GO TO 3621	
LCMSTR=1CMSTR=1 LCMSTR=1CMSTR=1 B400 IF(F2C1)=FC10, CT.DEL / CD TO 3621 LF(F2C1)=FC10, CT.DEL / CD TO 3621 LF(F2C1)=FC10, CD TO 5622 LF(F2C1)=FC10, CD TO 5622 HF(F1C0)=FC1, CD TO 5622 HF(F1C0)=FC1, CD TO 5622 HF(F1C0)=FC1, CD TO 100 IF(F1C0)=FC1, CD TO 100 IF(F1C0)=FC1, CD TO 100 IF(F1C0)=FC1, CD TO 100 HF(F1C0)=FC1, FC1, MF(F1C0)=FC1, CD TO 100 HF(F1C0)=FC1, MF(F1C0)=FC1, MF(F1C0)=FC1, FC1, FC1, MF(F1C0)=FC1, FC1, FC1, MF(F1C0)=FC1, FC1, FC1, MF(F1C0)=FC1, FC1, FC1, FC1, FC1, FC1, FC1, FC1,		
wRITE(6,030) ALOW,1 8830 b20 F(F2(2)-FC(),2(1)-FC(),2(1)-FC() 8840 tCMSTR-ICMSTR-1 8850 wRITE(6,000) ALOP,1 8840 b21 F(F(1)ME 8840 wRITE(6,000) ALOP,1 8940 wRITE(6,000) ALOP,1 8940 if (IG,000) ALOP,1 8940 wRITE(6,000) ALOP,1 8940		
5420 JF(F2(1)-F(1), GT.DEL) GD TO 3621 8840 INSTRUCTION 8840 WRITE(6,130) AUPP,1 8840 5621 JF(1100000000000000000000000000000000000		
ICMSTR-1(MSTR-1 metrics, 8030) AUPP,1 8850 8850 8670 8670 8670 8670 8670 8670 8670 867		
wRITE(6,8030) AUPP,I 8860 621 CONTINUE 8870 IF(ICNSTR.E0.0) GD TO 3622 8880 wRITE(6,8037) 8890 CFL REACT AR ANS.ED.4HBACK) WRITE(6,844) NRUN 8900 IF(ICNSTR.E0.0) TO 1001 8900 IF(ICNSTR.E0.0) TO 1001 8900 IF(ICNSTR.E0.0) TO 1001 8900 IF(ICNSTR.E0.0) TO 1004 8900 IF(ICNSTR.E0.0) TO 1034 8900 OD 351 II:1, NUME 8900 IF(ICNSTRES.ASD.R.E0.0) TO 344 8900 OD 351 II:1, NUME 9000 WRITE(6,807) II.RSS(II) 9000 ISI II:1, NUME 9000 WRITE(6,807) II.RSS(II) 9000 WRITE(6,807) II.SS 9000 WRITE(6,807) II.SS 9000 WRITE(6,807) II.SS 9000 WRITE(6,807) II.SS 9000 WRITE(6,8101) II.SS <td< td=""><td></td><td></td></td<>		
IFC[[LMSTR_E0.0] G0 T0 3622 8880 RTTEC6.8031 8900 CALL READ(7, ARC) 8900 IFC1[GD.E0.2] C0 RAMS.E0.4HBACK) WRITE(6,844) NRUN 8910 IFC1[GD.E0.2] C0 TO 1001 8920 IFC1[GD.E0.2] C0 TO 1001 8920 IFC1[GD.E0.2] C0 TO 1004 8930 S022 COMTINUE 8940 IFC1[FLAC.NE.0] D0 TO 3513 8940 IFC1[FLAC.NE.0] D0 TO 3513 8940 IFC1[FLAC.NE.0] D0 TO 354 8940 IFC1[FLAC.NE.0] D0 TO 354 8940 IFC1[FLAC.NE.0] D0 TO 364 9900 VIEVES.AVD.NOWCHT.NE.1) WRITE(6,800)IL,WRSN50, 8960 VIEVES.60.01 (FLASS) 9910 VIEVES.60.01 (FLASS) 9900 JESS (FLASS) 9910 VIEVES.60.01 (FLASS) 9910 VIEVES.60.01 (FLASS) 9910 VIEVES.60.01 (FLASS) 9900	WRITE(6,8030) AUPP, I	
##1TE(6,803*) 8890 CALL READ(1,AR) 8000 FC(EG).E0.2; 0R,AMS.E0.4HBACK) HRITE(6,844) NRUN 8910 FC(EG).E0.2; 0C,AMBACK) GD TO 1004 8930 B622 GMITINEK 800 HITE(6,80.2); 0C TO 3513 8900 HITE(6,80.2); 0C TO 3513 8900 HITE(6,80.2); SEREG, SSTOT,SSY,YBAR,RESMSO,REGMSO,RESBAR, 8900 HITE(6,80.3); NEES,AND.NONCHT.WE.1) HRITE(6,806); FUNC, HRSNSO, 8990 HITE(6,80.3); DG TO 3540 9000 Q0 353 [1:1, HPUNC 9010 HHITE(6,80.7); IRSS(11) 9010 HITE(6,80.7); IRSS(11) 9010 HITE(6,80.7); IRSS(11) 9010 HITE(6,80.7); IRSS(11) 9020 J65 FC(ANSIG, C, STOT, SSY, HAR, RESMSO, REGNSO, RESBAR, 9000 J7 HITE(6,80.7); IRSS(11) 9010 J80 HITE(6,80.7); IRSS(11) 9010 J90 J161, HPUNC 9010 J90 J161, HPUNC 9010 J90 J90 9030 J90 J90 9030 J90 J90 9030 J90 9030 <		
CALL READ(1,AR) IF<[160.E0.2.2) GO AMS.E0.4HBACK) HRITE(6,844) NRUN FG (160.E0.2.2) GO TO 1001 FG (160.E0.2.2) GO TO 1004 8020 FG (1F (1F (AS) E0.4HBACK) GO TO 1004 8030 8040 FG (1F (1F (AS) E0.4HBACK) GO TO 1004 8040 FG (1F (AS) E0.4HBACKS CONTO, SSY, YBAR, RESHSO, REGNSO, RESBAR, WRITE(6,807) FL, GO TO 3513 WRITE(6,807) FL, GO TO 364 9010 901		
IF(IG0.50.2) CR.AMS.E0. 4HBACK> HRITE(6,844) NRUN 8910 IF(IG0.50.2) CO TO 1001 8920 IF(AMS.E0. 4HBACK> GO TO 1004 8930 S022 CONTINUE 8940 IF(IFLAG.HC.) CO TO 10513 8950 WITTE(6, 800/SBRES, SSREG, SSTOT, SSY, YBAR, RESMSD, REGMSD, RESBAR, 8950 *SDVRES, RSD, R 8970 IF(AMS.E0. 4HVES.AND.NONCHT.NE.1) WRITE(6, 806)FUNC, HRSMSD, 8980 *WISBAR, MSOVRS 8970 IF(AMS.E0. 1) GO TO 364 9000 OO 363 II:1, HFUNC 9010 00 363 II:1, HEND 9010 00 363 II:1, HES, AND NONCHT.NE.1) WRITE(6, 808)II, WRSS(II) 9030 9040 9030 9051 WILLEG, 8071, LRSS(II) 9030 9051 9070 9030 9054 WILLEG, 8071, LRSS(II) 9030 9055 9070 9030 9056 WILLEG, 8071, LRSS(II) 9030 9056 9040 9040 9057 9070 9030 9058 9070 9030 9059 9040 9040 9040 9040 9040 </td <td></td> <td></td>		
IF (CIGD.ED.2) GD T0 1001 8920 5622 CONTINUE 8930 5622 CONTINUE 8940 IF (IF.AG. ME. 0) GD T0 3513 8950 MRITE (6, 80) SSRES, SSREG, SSTOT, SSY, YBAR, RESMSD, REGMSD, RESBAR, 8950 8950 IF (IF.AG. ME. 0) GD T0 3513 8950 MRITE (6, 80) SSRES, SSREG, SSTOT, SSY, YBAR, RESMSD, REGMSD, RESBAR, 8950 8970 IF (ARSI6, ED. 3WYES, AND. NOWCHT. WE.1) HRITE(6, 806)FUNC, HRSMSD, 8980 8970 IF (MFUNC, ED.1) GD T0 364 9000 DO 363 II:1, MURC 9010 HRITE (6, 807)II, RSS(II) 9020 364 IF (ARSI6, ED. 3WYES, AND. NOWCHT. WE.1) HRITE(6, 808)II, HRSS(II) 9030 J65 IF (RASI6, ED. 3WYES, AND. NOWCHT. WE.1) HRITE(6, 808)II, HRSS(II) 9030 J66 IF (RASI6, ED. 3WYES, AND. NOWCHT. WE.1) HRITE(6, 808)II, HRSS(II) 9030 MULL=0 9040 9040 HARE(1) 9010 9040 MUL=0 9040 9040 MOUL=0 9040 9040	IF(IGO EO 2 OR ANS EO 4HRACK) HRITE(6 844) NRIN	
IFCAMS.EG. 448ACK) GD TO 1004 8930 622 CONTINUE 8940 IFC(IFLAG, WE.0) GD TO 3513 8940 WITE (6, 800, SERS, SSREG, SSTOT, SSY, YBAR, RESMSD, REGMSD, RESBAR, 8960 8950 SDVRES, RSD, R 8970 IFC(ARS16, EC, 3WYES, AND.NOWCHT.WE.1) WRITE(6, 806)FUNC, WRSMSD, 8980 8970 WIRSBAR, MSDVRS 8970 IFC(MFLE, EC, 1) GD TO 364 9000 OD 353 11:1, WFUNC 9000 OD 353 11:1, WFUNC 9000 MOUT: FC(A, 207) I, MAS(IT) 9000 MOUT: FC(A, 207) I, MAITE(6, 810) 9000 WOUT: FC(A, 10, 10, MAITE(6, 810) 9000 WOUT: FC(A, 10, 10, MAITE(6, 813) 9000 WOUT: FC(A, 10, 10, 10, MAITE(6, 813) 9000 MOUT: FC(A, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10	IF(IGO.EQ.2) GO TO 1001	
IF(1FLG, ME.O) GO TO 3513 3950 WITE (6, 80, 580ES, SSTOT, SSY, YBAR, RESMSO, REGMSO, RESBAR, 8960 3960 *SDVMES, RSO, R 5960 *SDVMES, RSO, R 3960 *IF(AMS16, EC, 1) HELS, MOL, NOMGHT, NE. 1) HRITE(6, 806)FUNC, HRSMSO, 8960 3960 *IF(AMS16, EC, 1) FUES, AND, NOMGHT, NE. 1) HRITE(6, 806)FUNC, HRSMSO, 8960 9000 *D0 363 TI=1, NFUNC 9010 WHITE (6, 807)T, RSS(11) 9030 \$64 TE(RSG, LT, SO, 1) HRITE(6, 810) 9030 MUTTE (6, 812) 9030 NOUL; =0 9030 NSAVE =1 9030 NGAVE =1 9030 NGAVE =1 9030 NGAVE =1 9030 NSAVE =1 9030 NGAVE =1 9030 NGAVE =1 9100 NGAVE =1 9100 </td <td>IF(ANS.EG. 4HBACK) GO TO 1004</td> <td></td>	IF(ANS.EG. 4HBACK) GO TO 1004	
WRITE(6,804)SSRES, SSREG, SSTOT, SSY, YBAR, RESMSD, REGMSD, RESBAR, 8960 *SDVRES, R60,R 1F(AMS16,E0, 3HYES, AMD, NOMCHT.NE.1) WRITE(6,806)FUNC, WRSMSD, 8990 *URSBAR, MSDVRS 9000 00 351 1F1, HEUNE WRITE(6,807)IL, RSS(II) 9010 964 9010 965 1F(AMS16,E0, 3HYES, AMD, NOMCHT.NE.1) WRITE(6,808)IL, WRSS(II) 9020 964 9010 964 9010 965 1F(AMS16,E0,3HYES, AMD, NOMCHT.NE.1) WRITE(6,808)IL, WRSS(II) 9030 966 9040 9040 WRITE(6,807)IL, RSS(II) 9050 9050 966 9040 9050 967 9050 9050 967 9050 9050 967 9050 9050 967 9050 9050 967 9050 9050 967 9050 9050 967 9050 9050 967 9050 9050 967 9050 9050 967 9050 9050 967 9070 <		
*SDVRES, RSD,R *SDVRES, RSD,R *RSBAR, MSVRS *MRSBAR, MSVRS 1:1, MFUNC (EG.) GD TO 364 00 363 [1:1, MFUNC (WRITE (6, 807)], RSS(11) WRITE (6, 812) *GO *GO *GO *GO *GO *GO *GO *GO	IF(IFLAG.NE.0) GO TO 3513	8950
IF (AM S16, E0, 3HYES, AMD, NOHCHT, NE, 1) HRITE(6, 806)FUNC, HRSMSD, 8980 HRSMA, MSOVRS 8990 IF (NFUNC, E0, 1) GD TO 364 9000 DO 363 II (=1, NFUNC HRITE(6, 807)II, HRSU(1) 9010 WRITE(6, 807)II, HRSU(1) HRITE(6, 808)II, HRSS(II) 9030 S64 IF (RSD, IT. 50.) HRITE(6, 810) 9040 HRITE(6, 812) 9050 NOULLED 9000 WSAVE =1 9000 NSAVE = 1 90000 NSA	WRITE (0, 804) SSRES, SSREG, SSTUT, SSY, YBAR, RESHSU, REGHSU, RESBAR,	8960
-HRSBAR, MSDVRS 8990 IF<(FUNC.E.C.) GO TO 364		
IF(NFUNC.E0.1) G0 T0 364 9000 00 363 11:1.NEUNC 9010 WRITE(6,807)11.RES(11) 9020 363 IF(ANS16.E0.3HYES.AMD.NOWCHT.NE.1) WRITE(6,808)11,WRSS(11) 9030 364 IF(RSL,LT.50.) WRITE(6,810) 9040 WRITE(6,812) 9050 WOUL=0 9060 NSAVE=1 9070 00 365 1:1.NOBS 9080 W=LABEL(1) 9090 IF(NED.NSAVE+1) WRITE(6,813)N 9090 NSAVE=N 9110 ANROV:RES(1)/SOVRES 9120 IF(NIND.GC.1)WRITE(6,814)I,X(1),Y(1),YEST(1),RES(1),DIFPCT(1) 9130 *AMROV 9150 *IF(NIND.GT.1)WRITE(6,816)I,X(1),Z(1),Y(1),YEST(1),RES(1), 9150 *IF(RIMD.GC.1)WRITE(6,816)I,X(1),Z(1),Y(1),YEST(1),RES(1), 9150 *IF(NIND.GT.1)WRITE(6,816)I,X(1),Z(1),Y(1),YEST(1),RES(1), 9150 *IF(NIND.GT.1)WRITE(6,816)I,X(1),Z(1),Y(1),YEST(1),RES(1), 9150 *IF(NIND.GT.1)WRITE(6,816)I,X(1),Z(1),Y(1),YEST(1),RES(1), 9150 *IF(ADUTL=1 9180 *OUTL=40UTL=1 9180 *OUTL=40UTL=1 9180 *OUTL=10UTL=1 9180 *OUTL=10UTL=1 <t< td=""><td>WRSHAR, MSDVRS</td><td></td></t<>	WRSHAR, MSDVRS	
D0 363 11:1, MFUNC 9010 WRITE(6,807)ILRSS(11) 9020 364 IF(ANSI6,E0,3HYES,AYD,NOMGHT,NE.1) WRITE(6,808)IL,WRSS(11) 9030 364 IF(RSG,LT,S) WRITE(6,810) 9050 W011:=0 9050 W011:=0 9060 W0365 9080 W=LABEL(1) 9010 IF(N,E0,MSAVE+1) WRITE(6,813)N NSAVE=N 9110 AMROV-RES(1)/SOVRES 9120 IF(N,E0,MSAVE+1)/WRITE(6,813)N 9100 NSAVE=N 9110 AMROV-RES(1)/SOVRES 9120 IF(N,E0,MSAVE+1)/SOVRES 9120 IF(N,E0,MSAVE+1)/SOVRES 9130 .AMROV 9150 IF(IND.EC,1)/WRITE(6,816)I,X(1),Z(1),Y(1),YEST(1),RES(1), 9150 IF(RIND.EC,1)/SOVRES 9170 NOUT==0007L1=1 9180 IDUTI(NOUT).IT.1) GO TO 365 9170 NOUT==0007L1=1 9180 IDUTI(NOUT).IT.3) GO TO 367 9220 WRITE(6,817).IL,3,9) 9230 J66 IF(RAS,ATO).JRS 9230 J66 WRITE(6,819).J,J=3,HIND) 9250 J67 1920 J77 9220 W011=#0005 9240 J66 WRITE(6,819		
363 1F (AMS16,E0, 3MYES, ARD, NONGHT, NE, 1) WRITE(6,808) 11, WRSS(11) 9030 364 1F (RSQ, 1T, 50,) WRITE(6,810) 9040 WRITE(6,812) 9050 NOUTL=0 9060 00.365 1=1, NOBS 9030 WILLABEL(1) 9040 NSAVE=1 9040 NSAVE=1 9040 NSAVE=1 9040 NGUT_E0_NSAVE+1) WRITE(6,813)N NSAVE=1 9100 NSAVE=1 9100 NSAVE=1 9100 NSAVE=1 9100 NARDV=RES(1)/SDVRES 9100 1F(N.E0.NSAVE+1) WRITE(6,814)I,X(1),Y(1),YEST(1),RES(1),DIFPCT(1) 9130 9100 NARDV=RES(1)/SDVRES 9100 1F(N.E0.C1.1)WRITE(6,816)I,X(1),Z(1),Y(1),YEST(1),RES(1),DIFPCT(1) 9130 *JAMRDV 1F(ABS(ANRDV),LT,T) GO TO 365 9170 NOUTL=MOUTL+1 9180 9100 1D(1, MOUTL=1) 9180 9100 VDIT_L(MOUTL+2) 9180 9100 VDIT_L(MOUTL+1) 9180 9100 VDIT_L(MOUTL+1) 9180 9100 <td>00 363 11=1,NFUNC</td> <td></td>	00 363 11=1,NFUNC	
364 IF (R\$G, LT, 50.) WRITE(6, 812) 9050 MOUT_=0 9060 MSAVE=1 9070 DD 365 1=1, NOBS 9080 W=LABEL(1) 9090 IF (N, E0, NSAVE+1) WRITE(6, 813)N 9100 NSAVE=N 9110 AMROV=RES(1)/SOVRES 9120 IF (N ND, E0, I)WRITE(6, 814)I,X(I),Y(I), YEST(I), RES(I), DIFPCT(I) 9130 *,AMROV 9150 IF (N ND, GT, I)WRITE(6, 816)I,X(I),Z(I),Y(I),YEST(I), RES(I), 9150 *OIFPCT(I), AMROV 9150 *IF (ABS(AMROV), LT, T) GO TO 365 9170 MUL_=NOUTL+1 9180 IOUT (NOUTL)=i 9190 365 CONTINUE 9200 SC CONTINUE 9200 SK IFE (6, 818) 9200 SK IFE (6, 818) 9200 SK IFE (6, 818) 9200 SK IFE (6, 816) 9200 SK IFE (6, 816) <td>WRITE(6,807)[1,RSS(1])</td> <td></td>	WRITE(6,807)[1,RSS(1])	
WRITE(6,812) 9050 NOUL:0 9060 NSAVE:1 9070 D0 365 1:1,NOBS 9080 WitABEL(1) 9090 F(N.E0.NSAVE+1) WRITE(6,813)N 9090 NSAVE:W 9110 AMRDV:RES(1)'SOURES 9120 IF(N.HO.E0.1)WRITE(6,814)I,X(I),Y(I),YEST(I),RES(I),DIFPCT(I) 9130 *,AMRDV 9140 F(NIHO.E0.1)WRITE(6,816)I,X(I),Z(I),Y(I),YEST(I),RES(I), 9150 *DIFPCT(1),AMRDV 9160 *DIT(NOUT.1)WRITE(6,816)I,X(I),Z(I),Y(I),YEST(I),RES(I), 9150 *DIT(NOUT.1) 9180 *SOUT(NUE 9200 *SOUT(NUE 9200 *SOUT(NUE 9200 *SOUT.1) 9210 *SOUT.1) 9220 *SOUT.1) 9220 *SOUT.1)	565 IF (ANS16.EU. SHYES.AWD. NOWGHI.NE.1) WRITE(6,808) II, WRSS(II)	
NQU1:=0 9060 NSAVE=1 9070 D0 365 1=1,N0BS 9080 N=LABEL(1) 9090 IF(N,E0.NSAVE+1) WRITE(6,813)N 9100 NSAVE=N 9110 ANRDV:RES(1)/SDVRES 9120 IF(NIND.E0.I)WRITE(6,814)I,X(1),Y(1),YEST(1),RES(1),DIFPCT(1) 9130 *,AMROV 9140 *,AMROV 9140 *,AMROV 9140 *,AMROV 9140 *,AMROV 9150 *DIFPCT(1),ANROV 9160 IF(ABS(AMRDV),LT,T) GO TO 365 9170 NOUT:=NOUTL+1 9180 IOUTI:NOUTL>1 9180 YRITE(6,818) 9200 VRITE(6,818) 9210 IF(CABS(ARROV),LT,T) GO TO 365 9210 VRITE(6,818) 9200 VRITE(6,818) 9210 IF(0,10,11,1,13,9) 9230 D0 366 I=1,NOBS 9240 56 WRITE(6,818) 9260 56 WRITE(6,818) 9260 56 IF(6,818) 9260 56 IF(6,00,005 9270 56 IF(6,00,005 9270		
NSAVE:1 9070 D0 365 1:1,N085 9080 N:LABEL(1) 9090 IF(N.E0.NSAVE+1) WRITE(6,813)N 9100 NSAVE:W 9110 ANRDV:RES(1)/SDVRES 9120 IF(N.H0.E0.1)WRITE(6,814)I,X(1),Y(1),YEST(1),RES(1),DIFPCT(1) 9130 *,ANRDV 9140 IF(N.H0.E0.1)WRITE(6,816)I,X(1),Z(1),Y(1),YEST(1),RES(1),DIFPCT(1) 9130 *DIFPCT(1),ANRDV 9140 IF(N.H0.GT.1)WRITE(6,816)I,X(1),Z(1),Y(1),YEST(1),RES(1), 9150 *DIFPCT(1),ANRDV 9160 IF(ABS(ANRDV),LT.T) GO TO 365 9170 NOUTL:NOUTL+1 9180 IOUTL(NOUTL+2i 9130 \$455 CONTINUE 9210 \$476 (A 18) 9210 IF((AB(AT))(1, T.3), GO TO 367 9220 \$476 WRITE(6,818) 9240 \$400 MRITE(6,818) 9240 \$410 MRITE(6,818) 9240 \$410 MRITE(6,818) 9240 \$410 MRITE(6,010,1),J=3,NIND) 9250 \$410 MRITE(6,010,1),GOV(J,1),J=3,NIND) 9250 \$410 MRITE(6,010,10,000 TO 3701 9270	NOUTLED	
D0 365 1=1,NOBS 9080 N=LABEL(1) 9090 IF(N.EQ.NSAVE+1) WRITE(6,813)N 9100 NSAVE=N 9110 ANADV:RES(1)/SDVRES 9120 IF(NIND.EQ.1)WRITE(6,814)I,X(1),Y(1),YEST(1),RES(1),DIFPCT(1) 9130 *,AMROV 9150 *,DIFPCT(1),AMROV 9150 *,DIFPCT(1),AMROV 9150 *,DIFPCT(1),AMROV 9160 IF(ABS(AMROV).LT.T) GO TO 365 9170 NOUT.=NOUTL+1 9180 IOUTL(NOUTL+1 9120 VRITE(6,818) 9210 IF(ANIND.LT.3) GO TO 367 9220 WRITE(6,819) 9230 D0 366 I=1,NOBS 9240 56 WRITE(6,818) 56 IF(COLUTL-EQ.0) GO TO 3701 #ITE(6,818) 9260 56 IF(CNUUTL-EQ.0) SGDFF	NSAVE = 1	
iF(N_E0,NSAVE+1) HRITE(6,813)N 9100 NSAVE=N 9110 ANRDV: PS(1)/SDVRES 9120 iF(N_H0.E0.1) HITE(6,814)I,X(1),Y(1),YEST(1),RES(1),DIFPCT(1) 9130 *,ANRDV PI00 9100 *,ANRDV PI00 9140 *,ANRDV PI10 9130 *,ANRDV PI10 9140 *,ANRDV PI10 9150 *,DIFPCT(1),ANRDV PI00 9170 *DIFPCT(1),ANRDV 9180 9170 NOUTL=NOUTL+1 9180 9190 10UTL(NOUTL)=1 9190 9210 VRITE(6,A18) 9220 9220 VRITE(6,A18) 9220 9230 D0 366 150* 9240 96* HTE(6,A18) 9260	00 365 1=1,NOBS	
NSAVE=N 9110 ANADV:RES(1)/SDVRES 9120 IF(NIND.E0.1)WRITE(6,814)I,X(1),Y(1),YEST(1),RES(1),DIFPCT(1) 9130 *,AMRDV 91+0 iF(NIND.GT.1)WRITE(6,816)I,X(1),Z(1),Y(1),YEST(1),RES(1), 9150 DIFPCT(1),ANRDV 9160 IF(ABS(ANRDV).LT.T) GO TO 365 9170 NOUT:NOUTL+1 9180 IOUTL:NOUTL+1 9180 IOUTL:NOUTL+1 9190 VRITE(6,818) 9200 VRITE(6,818) 9210 IF(ABS(ANRD(T),1),1=3,9) 9220 DO 366 1=1,NOBS 9220 56 WRITE(6,818) 9220 56 WRITE(6,0) WRSDF 9220		
AMADV-RES(1)/SDVAES 9120 IF(NIND.EG.1)WRITE(6,814)1,X(1),Y(1),YEST(1),RES(1),DIFPCT(1) 9130 *,AMRDV 91+0 IF(NIND.GT.1)WRITE(6,816)1,X(1),Z(1),Y(1),YEST(1),RES(1), 9150 *0[FPCT(1),XMRDV 9140 IF(ABS(ANRDV),LT.T) GO TO 365 9170 NOUTL=NOUTL+1 9180 IOUTL(NOUTL+2i 9190 365 CONTINUE 9200 STE(6,818) 9210 IF(NIND.LT.3) GO TO 367 9220 WRITE(6,818) 9220 JE(0.15,000) 9220 S66 La1,NOB5 9220 J66 WRITE(6,818) 9220 J76 TE(6,000,000) 9220 J76 TE(6,000,000) 9220 J76 TE(6,000,000) 9220 J76 WRITE(6,818) 9220 J76 WRITE(6,818) 9220 J76 TE(6,000,000) 9220		
1F(NIND.EG.1)WRITE(6,814)1,X(1),Y(1),YEST(1),RES(1),DIFPCT(1) 9130 *,ANROV 91+0 1F(NIND.GT.1)WRITE(6,816)1,X(1),Z(1),Y(1),YEST(1),RES(1), 9150 *DIFPCT(1),ANROV 9150 *DIFPCT(1),ANROV 9160 1F(ABS(ANROV),LT.T) GO TO 365 9170 NOUTL=NOUTL+1 9180 1OUTL(NOUTL)=1 9190 365 CONTINUE 9200 VRITE(6,818) 9210 1F(NIND.LT.3) GO TO 367 9220 VRITE(6,818) 9220 00 366 1=1,NOB5 9240 566 WRITE(6,819)1,(XX(J,1),J=3,NIND) 9250 961 9220 566 WRITE(6,818) 9260 566 WRITE(6,810) 9220 566 WRITE(6,810) 9220 566 WRITE(6,810) 9220 566 WRITE(6,810) 9220 566 WRITE(6,010) 9220 566 WRITE(6,010) 9220 567 9220 568 9220 569 9220 560 9220 561 9220 562 9220 563		
*,AMROV 91+0 iF(NIMO_GT.1)WRITE(6,816)1,X(I),Z(I),Y(I),YEST(I),RES(I), 9150 *DIFPCT(I),AMROV 9160 IF(ABS(AMROV).LI.T) GO TO 365 9170 NOUTL:NOUTL+1 9180 10UTL(NOUTL)=i 9190 365 CONTINUE 9200 WRITE(6,818) 9210 IF(NIMO.LT.3) GO TO 367 9220 WRITE(6,818) 9210 1F(CNIMO.T),I=3,9) 9220 DO 366 I=1,NORS 9220 566 WRITE(6,819)I,(XX(J,I),J=3,NIND) 9250 WRITE(6,818) 9260 566 WRITE(6,818) 9270 WRITE(6,818) 9270 WRITE(6,818) 9270		
IF (N1ND.GT. 1) #RITE(6,816)1,X(1),Z(1),Y(1),YEST(1),RES(1), 9150 *DIFPCT(1),ANRDV 9160 *DIFPCT(1),ANRDV 9160 *DIFPCT(1),ANRDV 9160 *IF(CABS(ANRDV),LT.T) GO TO 365 9170 NOUTL:NOUTL:NOUTL:1 9180 10UTL(NOUTL)=i 9190 365 CONTINUE 9200 \$SRITE(6,818) 9210 IF(NIND.LT.3) GO TO 367 9220 #RITE(6,817)(1,1=3,9) 9230 DO 366 1=1,NOB5 9240 \$66 HITE(6,819)1,(XX(J,1),J=3,NIND) 9250 #RITE(6,818) 9260 \$66 HITE(6,0) GO TO 3701 9270 #RITE(6,0) SGD FD 9270	ANROV	
*DIFPCT(1), ANROV IF(ABS(ANROV), LT, T) GO TO 365 NOUTL=NOUTL+T IOUTL(NOUTL)=1 9180 10UTL(NOUTL)=1 9565 CONTINUE 9781FE(6,818) 1F(N1ND, LT, 3) GO TO 367 WRITE(6,819) 00 366 1=1,NOB5 966 WRITE(6,819) 90 366 1=1,NOB5 966 WRITE(6,819) 9750		
NOUTL:NOUTL: 9180 IOUTL(NOUTL): 9190 365 CONTINUE 9200 VRITE(6,818) 9210 IF(NINDLT.3) GO TO 367 9220 WRITE(6,817)(I,I=3,9) 9230 DO 366 I=1,NOB5 9240 566 WRITE(6,819)I, (XX(J,I),J=3,NIND) 9250 WRITE(6,818) 9260 567 If (NDUTL,E0,0) GO TO 3701 9270 WRITE(6,0) SG TO 9270	+DIFPCT(I),ANRDV	
IOUTL (NOUTL)=i 9190 365 CONTINUE 9200 WRITE(6,818) 9210 If (NIND.LT.3) GO TO 367 9220 MRITE(6,77)(I,1=3,9) 9230 D0 366 1=1,NOR5 9240 566 WRITE(6,819)I, (XX(J,1),J=3,NIND) 9250 WRITE(6,818) 9260 567 If (NOUTL,E0,0) GO TO 3701 9270 WRITE(6,620) NR5DF 9270		
365 CONTINUE 9200 VRITE(6,818) 9210 16 (NIN0.17.3) GO TO 367 9220 WRITE(6,817)(1,1=3,9) 9230 DO 366 1=1,0085 9240 566 WRITE(6,818) 9250 WRITE(6,818) 9260 367 16(000000000000000000000000000000000000		
VRITE(6,818) 9210 IF(NINDLT.3) GO TO 367 9220 WRITE(6,817)(I,1=3,9) 9230 DO 366 I=1,NOR5 9240 566 WRITE(6,819)I, (XX(J,1),J=3,NIND) 9250 WRITE(6,819)I, GO TO 3701 9250 WRITE(6,00) GO TO 3701 9270 WRITE(6,00) SGD FF 9270		
16 (N1N0.17.3) GO TO 367 9220 WR1TE(6, A17):(1, 1-3, 9) 9230 DO 366 1=1,NOB5 9240 566 WR1TE(6, A19)1, (OX(J, 1), J=3, NIND) 9250 WR1TE(6, B18) 9260 567 16 (NDUTL.EG.0) GO TO 3701 9270 WR1TE(6, CO) NR50F 9270		
D0 366 1=1,NOB5 9240 366 WRITE(6,819)I,(XX(J,I),J=3,NIND) 9250 WRITE(6,818) 9260 367 1F(NDUTL.E0.0) GD TO 3701 9270 WRITE(6,820) NR5DF 9280	IF (NIND.LT.3) GO TO 367	
D0 366 1=1,NOB5 9240 366 WRITE(6,819)I,(XX(J,I),J=3,NIND) 9250 WRITE(6,818) 9260 367 1F(NDUTL.E0.0) GD TO 3701 9270 WRITE(6,820) NR5DF 9280	WEITE(6,817)(1,1=3,9)	9230
WRITE(6,818) 367 IF (NOUTL.EG.0) GO TO 3701 WRITE(6,820) NRSDF 9270 9270	DO SAA IST NORS	
367 1F (NOUTL.EQ.0) GO TO 3701 9270 WR ITE (6.820) NRSDF 9280	566 WHITE(6,819)1,(30((J,1),J=5,WIND)	
WRITE(6,820) NRSDF 9280	147 1E(NOUT) ED 01 CO TO 3701	
00 370 1-1, NOUTL 9290	WRITE(6,820) NRSDF	
	00 370 1=1, NOUTL	
		MONA-

P01

	IF(ICNSTR.NE.1) GO TO 804	17750	
	WRITE(6,8000) FUNC, (F(L), L=1, NPR)	17760	
112025	WRITE(6,8005)	17770	
	WRITE(6,8010) AUPP,F2(1),1,F(1)	17780	
806	5 CONTINUE	17790	
	IF(ICNSTR.EQ.0) GO TO 814	17800	
	IF(IPRINT.ED1.AND.NVIOL.GT.1) WRITE(IW, 8031)NVIOL	17810	
	WRITE(6,8015)	17820	
808	CALL READ(1, AR)	17830	
	IF(AR(1).E0.4HBACK.OR.IGO.E0.2) GO TO 810 IF(AR(1).E0.2HNO) GO TO 812	17840	
	IF(AR(1).EQ.3HYES) GO TO 812	17850 17860	
	WRITE(6,8030)	17870	
	GO TO 808	17880	
810	IFAULT=5	17890	
	WRITE(6,8020)	17900	
	RETURN	17910	the second second
812	2 IQUAD=0	17920	
	WRITE(6,8025)	17930	
	RETURN	17940	
	CONTINUE	17950	
8000	FORMAT(* THE RESIDUAL SUM OF SQUARES MINIMUM (,D12.6,	17960	
	*) AT THE PARAMETER VALUES / * * /(* * ,7E10.4))	17970	

8005 FORMAT(" */~ * SEEM TO BE CONSTRAINED BY THE"/ * .) 8010 FORMAT(" */~ * SEEM TO BE CONSTRAINED BY THE"/ * .) 8010 FORMAT(" */ * A STATISTICAL EVALUATION OF THE PARAMETERS UNDER", * THESE CONDITIONS MAY FAIL."/ * * ENTER BIO OR BIT IF YOU WANT A NEW RUN WITH DIFFERENT // * * ENTER HON OR BIT IF YOU WANT TO CONTINUE DATA PARAMETER ESTIMATES"/ * * ENTER - YES- IF YOU WANT TO CONTINUE BUT WITHOUT A STATISTICAL", * ENTER - NO- IF YOU WANT TO CONTINUE BUT WITHOUT A STATISTICAL", * ENTER - NO- IF YOU WANT TO CONTINUE BUT HITHOUT A STATISTICAL", * ENTER - NO- IF YOU WANT TO CONTINUE BUT HITHOUT A STATISTICAL", * ENTER - NO- IF YOU WANT TO CONTINUE BUT HITHOUT A STATISTICAL", * BEEN CHANGED TO - MO- /) 8020 FORMAT(/1 * NPUT ERROR, YOUR ANSHER MUST BE ONE OF THE FOLLOWING"/ * FOUR, Y M B R - TRY AGAIN - /) IF(IPRINT)233, 232, 232 232 WRITE(IW, 301) 301 FORMAT(/1 *, 13(1H+), * FITTING OF QUADRATIC SURFACE IN REGION OF", * THENMAT ', 13(1H+) / ` FITTING OF QUADRATIC SURFACE IN REGION OF", * THENMAT ', 13(1H+) / ` 233 NEYAL=0 NFIX=0 BESIMP/100. SIMP2-FONC+B SIMP2-ANAXI(B, SIMP2) DO 201 I=1.NP1 204 IF(H(I).GE 1D50) GO TO 2030 TESTEDABS(H(I)-FUNC) IF(TESTE-SIMP2)202,201,201 202 DD 203 J=1, NOP IF(STEP(J).NE.0.0) G(I,J)=G(I,J)-F(J))+G(I,J) 2031 PST(J)=G(I,J) CALL FUNCTW(PSTST,H(I)) NEYAL=NEYAL+1 GO TO 204 203 DD 2051 J=1, NOP IF(STEP(J).ED.0.) GO TO 2031 IF(G(I,J).IT,F(LJ)) G(I,J)=FI(J) 2031 PST(J)=G(I,J) CALL FUNCTW(PSTST,H(I)) NEYAL=NEYAL+1 NEIX=NEIX+1 201 CONTINUE A0 = H(1) DO 205 J=1, NOP II=1.1 DO 206 J=1, NOP DI =1.1 DO 206 J=1, NOP DI =1.1 HITHOUSE ANAXI(ENT) NEYAL=NEYAL+1 NEIX=NEIX+1 201 CONTINUE A0 = H(1) DO 205 J=1, NOP DI =1.1 A0 = H(1) DO 205 J=1, NOP DI =1.1 A0 = H(1) DO 205 J=1, NOP DI =1.1 A0 = H(1) DO 205 J=1, NOP DI =1.1 A0 = H(1) DO 205 J=1, NOP DI =1.1 A0 = H(1) DO 205 J=1, NOP DI =1.1 A0 = H(1) DO 205 J=1, NOP DI =1.1 A0 = H(1) DO 205 J=1, NOP DI =1.1 A0 = H(1) DO 205 J=1, NOP DI =1.1 A0 = H(1) A0 = A0 = H(17990 18000 18010 18030 18040 \$210 11=1+1 D0 206 J=1,NOP 206 PSTAR(J)=(G(1,J)+G(11,J))/2.0 CALL FUNCTN(PSTAR,AVAL(1)) IF(AVAL(1).GT.(H(1)+H(11))/2D0)AVAL(!)=H(1)+H(11)-AVAL(1) D0 207 I=1,NAP I1=1-1 I2=1+1 12=1+1 IF(I1.LT.1)GOTO207 DO 208 J=1,I1 DU 208 3-1,11 J1=J=1 D0 209 k=1,NOP PSTST(k)=(G(12,k)+G(J1,k))/2.0 CALL FUNCTN(PSTST,HSTST) IF(HSTST,GT.(H(12)+H(J1))/2D0)HSTST=H(12)+H(J1)-HSTST NEVAL=NEVAL+1

POZ

FNEW(I)=F1(I) STPNEW(I)=STEP(I)	27050 27060	
IF(1.GT.IP1) GO TO 55 IF(F1(1).NE.F2(1)) IP1=1	27070 27080	
55 IF(F1(I).NE.F2(I)) IP2=1 DEL(IP1)=(F2(IP1)-F1(IP1))/9.	27090 27100	
DEL(IP2)=(F2(IP2)-F1(IP2))/9. ISAVE=1	27110 27120	
JSAVE=1 SSMIN=1.E99	27130 27140	
SSMAX=0. D0_60_J=1,10	27150 27160	
FNEW(IP1)=F1(IP1)+FLOAT(J-1)*DEL(IP1) P1(J)=FNEW(IP1) P1(J)=FNEW(IP1)	27170 27180	
D0 60 I=1,10 FNEW(IP2)=F1(IP2)+FLOAT(I-1)+DEL(IP2) P2(1)=FNEW(IP2)	27190 27200 27210	
CALL LSQ(FNEW,SSS(1,J)) IF(SSS(1,J),GE.SSMIN) GO TO 60	27220 27230	
SSMIN-SSS(1,J) ISAVE=1	27240 27250	
JSAVE=J 60 IF(SSS(1,J).GT.SSMAX) SSMAX=SSS(1,J)	27260 27270	

FHEXIL(D):=P:ICEANE) 27280 STREEXIL(D):SEC(12):22: 27300 STREEXIL(D):SEC(12):22: 27300 STREEXIL(D):SEC(12):22: 27300 STREEXIL(D):SEC(12):22: 27300 STREEXIL(D):SEC(12):22: 27300 STREEXIL(D):SEC(12):21:100 27300 HETEG: (D00):17:(1):17:00 27400 D0 120:11:10 27500 D0 120:10			
FHEML(1P2)=P2(1SWE) 27290 STTEMEN(1P2)=D2(1SWE) 27290 Der CSWAR-SSRIB/A 27200 Der CSWAR-SSRIB/A 27290 Der CSWAR-SSRIB			
FHEML(1P2)=P2(1SWE) 27290 STTEMEN(1P2)=D2(1SWE) 27290 Der CSWAR-SSRIB/A 27200 Der CSWAR-SSRIB/A 27290 Der CSWAR-SSRIB		FNEW(IP1)=P1(JSAVE)	27280
STRVENL(P2)=DE(1(P2)/2. C) GSM/151501 C) GSM/151501 (F) GSM/151501 METTEG6, 10301(P), (1, P1(1), [-1, 10) METTEG6, 10301(P), (1, P1(2), [-1, 10) METTEG6, 10301(P), (1, P2), (2, 2) METTEG6, 10301(P), (1, 1-2, 2) METTEG6, 10301(P), (1, 1-3, 2) METTEG6, 10301(P), (1, 1, 1-3, 2) METTEG6, 10301(P), (1, 1, 1-3, 2) METTEG6, 10301(P), (1, 2) METTEG6, (1, 2) MET		FNEH(IP2)=P2(ISAVE)	27290
D - CS9MA-S9118/7. D - CS9MA-S9118/7. D - CS9MA-S9118/7. D - CS9MA-S9118/7. D - CS9MA-S9118/7. D - CS9MA-S9118/7. H T E (6, 1050) IP ; (1, P2(1), [=1, 10) H T E (6, 1050) IP ; (1, P2(1), [=1, 10) H T E (6, 1050) IP ; (1, P2(1), [=1, 10) D - CS9MA-S9118/7. H T E (6, 1070) IP ; (1, P2(1), [=1, 10) D - CS9MA-S9118/7. H T E (6, 1070) IP ; (1, P2(1), [=1, 10) D - CS9MA-S9118/7. D - CS9MA-S9118/		STPNEW(IP1)=DEL(IP1)/2.	
00 65 1:1:10 2730 64 (1)3:7:4:100 2730 65 (1)3:7:4:100 2730 66 (1)3:7:4:100 2730 67 (1)3:7:4:100 2730 67 (1)3:7:4:100 2730 67 (1)3:7:4:100 2730 67 (1)3:7:4:110 2730 67 (1)3:7:4:110 2730 67 (1)3:7:4:110 2740 67 (1)3:1:110 2740 68 (1)3:1:110 2740 60 (1)3:1:110 2740 60 (1)3:1:110 2740 70 (1)3:1:100 2740 71 (1)3:1:100 2740 72 (1)3:1:100 2740 73 (1)3:1:100 2740 74 (1)3:1:100 2740 75 (1)3:1:100 2740 76 (1)3:1:100 2740 77 (1)3:1:100 2740 78 (1)3:1:100 2740 79 (1)3:1:1000:1:100 2740			
65 A(1)=55M(H*(-1)-10 A(10)-7(0)-0)(-0)(1)[-1](0) P(-1)(-1)(-1)(-1)(-1)(-1)(-1)(-1)(-1)(-1)			
A (10) = A (10) = 0, 0 = 0 W Te (A (10) = 0, 0 = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0, 0 = 0) = 0 W Te (A (10) = 0) =	-		27330
HTTEC6.10501Pf.(1,P3C1),E1,100 2760 HTTEC6.10201Pf.(1,P3C1),E1,100 2770 HTTEC6.10201Pf.(1,P3C1),E1,100 2770 HTTEC6.10201Pf.(1,P3C1),E1,100 2770 HTTEC6.10201Pf.(1,P3C1),E1,100 2770 HTTEC6.10201Pf.(1,P3C1),E1,100 2770 HTTEC6.10201Pf.(1,P3C1),E1,100 2740 HTTEC6.10201Pf.(1,P3C1),E1,10,20 2740 HTTEC6.10201Pf.(1,P3C1),E1,10,100 2740 HTTEC6.10201Pf.(1,P3C1),E1,10,100 2740 HTTEC6.10201Pf.(1,P3C1),E1,10,10,20 2750 HTTEC6.10201Pf.(1,P3C1),E1,10,10,20 2750 HTTEC6.10201Pf.(1,P3C1),E1,10,10,20 2750 HTTEC6.10201Pf.(1,P3C1),E1,10,10,20 2750 HTTEC6.10201Pf.(1,P3C1),E1,10,10,20 2750 HTTEC6.10201Pf.(1,P3C1),J1,10,11 2760 HTTEC6.10201Pf.(1,P3C1),J1,20,11,100 2760 HTTEC6.112001Pf.(1,	65		27340
WHTEC6, 1060/102, (1, P2C1), 1=1, 10) 2730 TFCAMSD, E.G. AMD, WHTEC6, 1060) 2730 WHTEC6, 1073)(1, 1=1, 10) 2740 00 67 [=1,10] 2740 00 67 [=1,10] 2740 00 67 [=1,10] 2740 00 77 [=1,10] 2740 00 77 [=1,10] 2740 00 77 [=1,10] 2740 00 78 [=1,10] 2740 00 78 [=1,10] 2740 00 78 [=1,10] 2740 00 78 [=1,10] 2740 00 78 [=1,10] 2740 10 78 [=1,10] 2740 10 78 [=1,10] 2740 10 78 [=1,10] 2740 10 78 [=1,10] 2740 11 4 25 [EMDFC4.stSS(1,1)-13.stSS(1,1)-25S(1,1,2)) 2750 11 5 [EMDFC4.stSS(1,1,1)-3.stSS(1,1,2)-25S(1,2)) 2750 11 6 [=1,10] 2740 11 7 [=1,10] 2740 11 7 [=1,10] 2740 11 7 [=1,10] 2740 11 7 [=1,10] 2740 11 7 [=1,10] 2740 11 7 [=1,10] 2740 11 7 [=1,10] 2740			
IF CAMSIG. E0. 2HMD1 WHITE(6, 1050) 2780 IF CAMSIG. E0. SHYES WHITE(6, 1050) 2780 WHITE(C, 107) IF(1)E2 27.00 WO G (7 1-10) 27.00 WO TE(C, 107) IF(1)E2 27.00 WI TE(C, 107) IF(1)E2 27.00 WI TE(C, 107) IF(1)E2 27.00 WI TE(C, 100) CSSS(1, 1), 1=7, 10 27.00 DO TE(D, 1-1, 10) 27.00 DO TE(D, 1-1, 10) 27.00 DO TE(D, 1-1, 10) 27.00 F(1, EE, 10, 00 TO 70 27.00 F(2, EE, 10) CO TO 10 27.00 DO TE(1, 1)=51GNOF(-4, *SSS(1, 1, 1)-3, *SSS(1, 1, 2)) 27.10 DO TE(1, 1)=51GNOF(-4, *SSS(1, 1, 1)-3, *SSS(1, 2, 1)) 27.00 F(1, EE, 10) CO TO 10 10 27.00 DO TE(1, L)=51GNOF(-4, *SSS(1, 1, 1)-3, *SSS(1, 2, 1)) 27.00 DO TE(1, L)=51GNOF(-4, *SSS(1, 1, 1)-3, *SSS(1, 2, 1)) 27.00 DO TE			
IF CAUSIG: EG. 3HYESD. HATTECG, 1060) 27300 HTTECG: 1070 JPT PZ; 27400 HTTEGG: 1000 JCSSCI, JJ, J=2; 10; 20 27400 D0 120 J=1; 10 D0 70 D7 120 J=1; 10 27400 D0 170 J=1; 10 27500 D0 170 J=1; 10 27500 D0 170 J=1; 10 100 D0 170 J=1; 10 100 D1 00 27500 Ff(1:E0:10) 50 DT 10 27500 D0 DFEV(1, L)=SIGMOF(-4, *SSS(1, J)-SSS(1, J)-SSS(1, J)) 27500 D0 DFEV(1, L, S=SIGMOF(-4, *SSS(1, J), SSS(1, J), SSS(1, J)) 27500 D0 DFEV(1, L, S=SIGMOF(-4, *SSS(1, J), SSS(1, J), SSS(1, J)) 27500 D0 DFEV(1, L, S=SIGMOF(-4, *SSS(1, J), SSS(1, J), SSS(1, J)) 27500 D0 DFEV			
HITEG. 107301P1_1P2 27400 HITEG. 107301.1=1.100 27410 OF 67.107301.1=1.00 27400 OF 67.107301.1=1.00 27400 OF 67.107301.1=1.00 27400 OF 100 1200 J=1.10 27400 OF 100 100 00 27400 OF 000 100 00 27500 OF 000 100 100 27500 OF 000 100 100 27500 OF 000 1100 27500 OF 000 1100 27500 OF 000 1100 100 27500 OF 000 1100 100 100 27500 OF 000 1100 100 100 27500 OF 000 1100 100 27500 OF			
wHTEC6,1075)(1,1=1,10) 2740 00 67 WHTEC6,1075)(1555(1,1),1,4=1,9,2) 2740 00 100 100 100 100 00 100 100 100 100 01 100 100 100 100 100 00 100 100 100 100 100 00 100 100 100 100 100 00 100 100 100 100 100 00 100 100 100 100 100 00 100 100 100 100 100 00 100 100 100 100 100 100 00 100		UPITE// 1070/101 102	
D0 67 1=1,10 27420 FW TETE(4,1080)(SSS(1,J),J=2,10,2) 27430 D0 120 1=1,10 27440 D0 120 1=1,10 27440 D1 120 (1-1,10) 27440 PEV(1,J)=SEGMOF(5SS(1,J+1)-SSS(1,J-1)) 27440 POEV(1,J)=SEGMOF(54,*SSS(1,J+1)-SSS(1,J-2)) 27500 POEV(1,J)=SEGMOF(54,*SSS(1,J-1)*3,*SSS(1,J)-SSS(1,J-2)) 27500 POEV(1,J)=SEGMOF(-4,*SSS(1,J-1)*3,*SSS(1,J)-SSS(1,J-2)) 27500 POEV(1,J)=SEGMOF(-4,*SSS(1,J-1)*3,*SSS(1,J)-SSS(1,J-2)) 27500 POEV(1,J)=SEGMOF(-4,*SSS(1,J-1)*3,*SSS(1,J)-SSS(1,J-2)) 27500 POEV(1,J)=SEGMOF(-4,*SSS(1,J)-SSS(1,J)-SSS(1,J-2)) 27500 POEV(1,L)=SEGMOF(-4,*SSS(1,J)-SSS(1,J)-SSS(1,Z,J)) 27500 POEV(1,L)=SEGMOF(-4,*SSS(1,J)-SSS(1,J)-SSS(1,Z,J)) 27500 POEV(1,L)=SEGMOF(-4,*SSS(1,J)-SSS(1,J)-SSS(1,Z,J)) 27600 POEV(1,L)=SEGMOF(-4,*SSS(1,J)-SSS(1,Z,J)) 27600 POEV(1,L)=SEGMOF(-4,*SSS(1,J)-SSS(1,J)-SSS(1,Z,J)) 27600 POEV(1,L)=SEGMOF(-4,*SSS(1,J)-SSS(1,J)-SSS(1,Z,J)) 27600 POEV(1,L)=SEGMOF(-4,*SSS(1,J)-SSS(1,J)-SSS(1,Z,J)) 27600 POEV(1,L)=SEGMOF(-4,*SSS(1,J)-SSS(1,J)-SSS(1,Z,J)) 27600 POEV(1,L)=SEGMOF(-4,*SSSS(1,J)-SSS(1,J)-SSS(1,Z,J)) <td< th=""><th></th><td></td><td></td></td<>			
HHTE(6, 1080) (SSS(1, J), J=1, 9, 2) 2740 00 170 J=1, 10 27400 00 170 0 1551000F(<-4, \$SS(1, J)-153, \$SS(1, J)-553(1, J)-2) 27500 00 170 0 10 27400 10 170 10 27500 27500 10 170 100 27500 27500 10 171 100 101 27500 27500 10 175 1000F(-4, *SSS(1, J,)-SSS(1, J,)-SSS(1, J,)-SSS(1, J,)-SSS(1, Z, J)) 27500 10 175 1000F(-4, *SSS(1, J), +3, *SSS(1, J), +SSS(1, Z, J)) 27500 10 175 1000F(-4, *SSS(1, J), +1, 2, SSS(1, J), +SSS(1, Z, J)) 27600 10 175 1000F(-4, *SSS(1, J), +1, 2, SSS(1, Z, J), +SSS(1, Z, J)) 27600 10 175 1000F(-4, *SSS(1, J), +1, 2, SSS(1, Z, J), +1, 2, SSS(1, Z, Z, J)) 27600		D0 67 1=1 10	
67 HATTE (6, 1085) (SSS(1, J), J=2, 10, 2) D (2) (J=1, 10 D (2) (J=1, 10) D		WRITE(6, 1080)(SSS(1, 1), 1=1, 9, 2)	
D0 120 J=1;10 D0 120 J=1;10 P0 FUC(LD, 1)=0 DT 70 FF(J=E0, 1):0 DT 70 P0 FUC(L, 1)=S1600F(x=SSS(1,J=1)-SSS(1,J=2)) P0 FUC(L, 1)=S1600F(x=SSS(1,J=1)-SSS(1,J=2)) P0 FUC(L, 1)=S1600F(x=SSS(1,J=1)+3,=SSS(1,J=2)) P0 FUC(L, 1)=S1600F(x=SSS(1,J=1)+3,=SSS(1,J=2)) P0 FUC(L, 1)=S1600F(x=SSS(1+1,J=1)+3,=SSS(1,J=2)) P0 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+2,J)) P0 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+2,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+2,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+2,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+2,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+2,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+1,J)-SSS(1+2,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+1,J)-SSS(1+2,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+1,J)-SSS(1+2,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+2,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+2,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+1,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+1,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+1,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+1,J)) P1 FUC(L, 1)=S1600F(x=SSS(1+1,J)-SSS(1+1,J	67	WRITE(6, 1085)(SSS(1, J), J=2, 10, 2)	
00 120 1-1,10 27460 F(J,EQ,10) GD TO 70 77 F(J,EQ,10) SGD TO 70 77 P(J,EQ,10) SGD TO 70 77 POEV(1,D) =SIGNOF(4.+SSS(1,J+1)-3,+SSS(1,J)-SSS(1,J+2)) 77 70 POEV(1,D) =SIGNOF(-4.+SSS(1,J-1)+3,+SSS(1,J)-SSS(1,J-2)) 77 70 FOEV(1,J) =SIGNOF(-4.+SSS(1,J-1)+3,+SSS(1,J-2)) 77 70 FOEV(1,J) =SIGNOF(-4.+SSS(1,J)-SSS(1,J-2)) 77 70 FOEV(1,K) =SIGNOF(-4.+SSS(1,J)-SSS(1,J)-SSS(1,J-2)) 77 70 FOEV(1,K) =SIGNOF(-4.+SSS(1,J)-SSS(1,J)-SSS(1,J-2)) 77 70 FOEV(1,K) =SIGNOF(-4.+SSS(1,J)-SSS(1,J)-SSS(1,J)) 77 70 FOEV(1,K) =SIGNOF(-4.+SSS(1,J)-SSS(1,J)-SSS(1,J)-SSS(1,J) 77 70 FOEV(1,K) =SIGNOF(-4.+SSS(1,J)-SSS(1,J)-SSS(1,J)-SSS(1,J)-SSS(1,J) 77 70 FOEV(1,K) =SIGNOF(-4.+SSS(1,J)-SSS(1,J)-SSS(1,J)-SSS(1,J)-SSS(1,J) 77 70 FOEV(1,K) =SIGNOF(-4.+SSS(1,J)-SSS(1,J		DO 120 J=1.10	
IF(J.E0.1) G0 T0 70 27470 IF(J.E0.10) G0 T0 80 27480 PDEV(I, J)=SIGNOF(4.+SSS(I, J+1)-3.+SSS(I, J)-SSS(I, J)-SSSS(I, J)-SSSS(I, J)-SSSS(I, J)-SSS(I, J)-SSS(I, J)-SSSS(I, J)-SSS(DO 120 I=1,10	
PDEV(1,J)=516MDF(SSS(1,J+1)-5SS(1,J-1)) GU TO 90 FOEV(1,J)=516MDF(4.+SSS(1,J+1)-3.+SSS(1,J)-SSS(1,J+2)) FOEV(1,J)=516MDF(-4.+SSS(1,J-1)+3.+SSS(1,J)-SSS(1,J-2)) FOEV(1,J)=516MDF(-4.+SSS(1,J-1)+3.+SSS(1,J)+SSS(1,J-2)) FOEV(1,J)=516MDF(-4.+SSS(1+,J)-3.+SSS(1,J)+SSS(1,J-2)) FOEV(1,K)=516MDF(SSS(1+,J)-SSS(1-1,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)-SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)-SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)-SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)-SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1+,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1+,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1+,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1+,J)+SSS(1-2,J)) FOEV(1,K)=516MDF(-4.+SSS(1+,J)+3.+SSS(1+,J)+SSS(1+,J		IF(J.EQ.1) GO TO 70	
G0 10 90 27500 G0 10 90		IF(J.EQ.10) GO TO 80	27480
G0 10 90 27500 G0 10 90		PDEV(I, J)=SIGNOF(SSS(I, J+1)-SSS(I, J-1))	27490
GD TD 90 K=1+10 27520 K=1+10 90 PDEV(1, j)=SIGNOF(-4,*SSS(1, j-1)+3.*SSS(1, j-2)) 27530 Z7540 91 F(1, E0, 1) GD TD 100 27540 1 F(1, E0, 1) GD TD 110 27540 92 DEV(1, K)=SIGNOF(-4,*SSS(1+1, j)-SSS(1-1, j)) 27570 93 GD 70 115 27570 94 DD 9DEV(1, K)=SIGNOF(-4,*SSS(1+1, j)-3.*SSS(1, j)-SSS(1+2, j)) 27580 95 GD 70 115 27570 96 DD 70 115 27570 97 GD 70 115 27570 98 DD 111 27570 99 DO V(1, K)=SIGNOF(-4,*SSS(1-1, j)*3.*SSS(1, j)*SSS(1-2, j)) 27610 99 DO V(1, K)=SIGNOF(-4,*SSS(1, j).LE.A(L+1)) CDN(1, j)=AL(L) 27620 99 DO V(1, K)=SIGNOF(-4,*SSS(1, j).J=1,20),1=1,100 27650 90 HI TE(6, 1100) (VOEV(1, j), j=1,20),1=1,100 27680 90 HI TE(6, 1130) 1, A(1), A(1+1) 27710 90 NET INVE 27730 90 NEVAL-UE 21877 GD TO 150 27730 91 NEVAL UE 21877 GD TO 150 27780 92 NF0-1 27800 93 NF1ET IXCMP01N***(1, /FLOAT(NEHNOP))*.001) 27780 94 NEVAL-UE 21877 GD TO 170 27810 95 NF1ET IXCMP01K*** 27800 96 NF1ET IXCM		GO TO 90	
80 PDEV(1,J):SIGNOF(-4.*SSS(1,J-1)*3.*SSS(1,J)*SSS(1,J-2)) 2750 91 F4:1-0 2750 1F(1:E0.1) GO TO 100 2750 1F(1:E0.1) GO TO 110 2750 90 F2:1E0:10) GO TO 110 2750 90 750 2750 90 750 2750 90 750 2750 90 750 2750 90 750 2750 90 750 2750 90 750 2750 90 750 2750 90 750 2750 90 750 2750 90 750 2750 90 750 2760 90 750 2760 90 750 2760 90 750 2760 90 750 2760 90 750 2760 90 151 2760 90 151 2760 90 151 2770 90 151 2770 <th>70</th> <td>PDEV(I,J)=SIGNOF(4.*SSS(I,J+1)-3.*SSS(I,J)-SSS(I,J+2))</td> <td></td>	70	PDEV(I,J)=SIGNOF(4.*SSS(I,J+1)-3.*SSS(I,J)-SSS(I,J+2))	
90 K=1+10 27540 IF(1,E0,10) G0 T0 110 27500 OPDEV(1,K)=51GMPF(4,*5SS(1+1,J)-5SS(1-J,J)) 27500 G0 T0 115 100 PDEV(1,K)=51GMPF(4,*5SS(1+1,J)-3,*5SS(1,J)-SSS(1+2,J)) 27500 G0 T0 115 110 PDEV(1,K)=51GMPF(4,*SSS(1,J)+SSS(1,J)+SSS(1-2,J)) 27610 115 C0MT1MUE 27660 WR ITE(6,1100) (CPDEV(1,J),J=1,20),I=1,10) 27660 WR ITE(6,1100) (CPDEV(1,J),J=1,20),I=1,10) 27660 WR ITE(6,1100) (CPDEV(1,J),J=1,20),I=1,10) 27660 WR ITE(6,1100) (CPDEV(1,J),J=1,10) 27660 WR ITE(6,1100) (CPDEV(1,J),J=1,10) 27660 WR ITE(6,1100) (CPDEV(1,J),J=1,10) 27660 WR ITE(6,1100,4(1),A(1+1)) 27700 IF(ISS) G1,J,G,A(1,A(1+1)) 27700 IF(ISS) G1,J,G,A(1,A(1+1)) 27700 IF(ISS) G1,J,G,A(1,A(1+1)) 27700 IF(ISS) G1,J,G,A(1,A(1+1)) 27700 IF(ISS) G1,J,G,G,G) G0 T0 150 27700 IF(ISS) G1,J,G,G) G0 T0 150 27700 IF(ISS) G1,J,G,G) G0 T0 150 27700 IF(ISS) G1,J,G,G) G0 T0 170 27700 IF(ISS) G1,J,G,G) G0 T0 170 K=00F+1 C, 100 G0 C0			
IF(1.E0.10) G0 T0 100 27550 IF(1.E0.10) G0 T0 110 27560 PDEV(1,K)=S1GNDF(SSS(1+1,J)-SSS(1-1,J)) 27580 100 PDEV(1,K)=S1GNDF(-4.*SSS(1+,J)-3.*SSS(1,J)-SSS(1+2,J)) 27590 100 PDEV(1,K)=S1GNDF(-4.*SSS(1-1,J)+3.*SSS(1,J)+SSS(1-2,J)) 27600 115 CONTINUE 27620 120 IF(SSS(1,J).GE.A(L).AND.SSS(1,J).LE.A(L+1)) CON(1,J)=AL(L) 27640 120 IF(SSS(1,J).GE.A(L).AND.SSS(1,J).LE.A(L+1)) CON(1,J)=AL(L) 27650 WRITE(6,1100)(PDEV(1,J),J=1,20).I=1,10) 27660 WRITE(6,1100)(PDEV(1,J),J=1,20).I=1,10) 27660 WRITE(6,1100)(CON(1,J),J=1,20).I=1,10) 27660 WRITE(6,1100)(CON(1,J),J=1,20).I=1,10) 27660 WRITE(6,1100)(CON(1,J),J=1,20).I=1,10) 27660 WRITE(6,1100)(CON(1,J),J=1,10) 27660 IF(I.E0.10) EO TO 150 27700 WRITE(6,1130)1.A(1).A(1+1) 27710 S0 OD TO 140 27750 IF(MEWALLE.2187) GO TO 150 27760 MP=MP-1 27700 IF(NPE E0.7) GO TO 170 27800 K=400P+1 27800 IF(NPE E0.7) GO TO 170 27800 K=400P+1 27800 IF(NPE E0.7) GO TO	80	PDEV(1, J)=SIGNOF(-4.*SSS(1, J-1)+3.*SSS(1, J)+SSS(1, J-2))	
IF(I_E0.10) 60 T0 110 27560 PDEV(I,k)=516MPC(4,*SSS(I+1,J)-SSS(I,J)) 27580 00 PDEV(I,k)=516MPC(4,*SSS(I+1,J)-3.*SSS(I,J)*SSS(I+2,J)) 27600 10 PDEV(I,k)=516MPC(-4,*SSS(I-1,J)*3.*SSS(I,J)*SSS(I-2,J)) 27600 10 PDEV(I,k)=516MPC(-4,*SSS(I,J).+3.*SSS(I,J)*SSS(I-2,J)) 27600 10 D12 L=1,9 27620 00 12 L=1,9 27660 HRITE(6,100)(CPDEV(I,J),L=1,20),I=1,10) 27660 HRITE(6,100)(CPDEV(I,J),J=1,20),I=1,10) 27660 HRITE(6,1100)(CPDEV(I,J),J=1,20),I=1,10) 27660 HRITE(6,1100)(CPDEV(I,J),J=1,20),I=1,10) 27690 HRITE(6,1130)I,A(I),A(I+1) 27700 HON FWALLELEXIB/7 GO TO 150 27760 HF(MEMPOHT**(1,FEDAT(NEHNOP))*,001) 27760 HON FACLENP**NEHNDP 27780 HF(MEMAPOHT***(1,FEDAT(NEHNOP))*,001) 27760 <t< th=""><th>90</th><td></td><td></td></t<>	90		
PDEV(T,k)=SIGMOF(SSS(1+1,J)-SSS(1-1,J)) GD TO 115 100 PDEV(T,k)=SIGMOF(4.*SSS(1+,J)-3.*SSS(1,J)-SSS(1+2,J)) 27500 110 PDEV(T,k)=SIGMOF(-4.*SSS(1-1,J)+3.*SSS(1,J)+SSS(1-2,J)) 115 CONTINUE 115 CONTINUE 116 CONTINUE 117 CONTINUE 117 CONTINUE 118 CONTINUE 119 CONTINUE 119 CONTINUE 119 CONTINUE 119 CONTINUE 119 CONTINUE 119 CONTINUE 110 CONTINUE 110 CONTINUE 110 CONTINUE 110 CONTINUE 111 CONTINUE 111 CONTINUE 111 CONTINUE 112 CONTINUE 114 CONTINUE 115 CONTINUE 115 CONTINUE 116 CONTINUE 117 CONTINUE 118 CONTINUE 119 CONTINUE 119 CONTINUE 119 CONTINUE 119 CONTINUE 119 CONTINUE 110 CONTINUE 110 CONTINUE 110 CONTINUE 110 CONTINUE 111 CONTINUE 111 CONTINUE 111 CONTINUE 112 CONTINUE 113 CONTINUE 114 CONTINUE 115 CONTINUE 115 CONTINUE 116 CONTINUE 117 CONTINUE 118 CONTINUE 119 CONTINUE 119 CONTINUE 119 CONTINUE 119 CONTINUE 119 CONTINUE 119 CONTINUE 110 CONTINUE 110 CONTINUE 110 CONTINUE 110 CONTINUE 110 CONTINUE 111 CONTINUE 111 CONTINUE 111 CONTINUE 112 CONTINUE 115 CONTINUE 115 CONTINUE 115 CONTINUE 115 CONTINUE 115 CONTINUE 116 CONTINUE 117 CONTINUE 118 CONTINUE 118 CONTINUE 119 CONTINUE 119 CONTINUE 110 CONTINUE 110 CONTINUE 110 CONTINUE 110 CONTINUE 111 CONTINUE 111 CONTINUE 111 CONTINUE 112 C			
G0 T0 115 27580 00 PDEVC(I,k)=SIGNPE(-4.+SSS(I+1,J)-3.+SSS(I,J)-SSS(I+2,J)) 27600 10 PDEVC(I,k)=SIGNDF(-4.+SSS(I-1,J)+3.+SSS(I,J)+SSS(I-2,J)) 27610 115 CONTINUE 27620 00 120 L=1,9 27630 120 F(SSS(I,J),GE,A(L),AND.SSS(I,J).LE.A(L+1)) CON(I,J)=AL(L) 27640 PRITE(6,100)(PDEV(I,J),J=1,20),I=1,10) 27660 PRITE(6,1100)(CPDEV(I,J),J=1,20),I=1,10) 27660 PRITE(6,1100)(CD0(I,J),J=1,20),I=1,10) 27660 PRITE(6,1100)(CD0(I,J),J=1,20),I=1,10) 27660 PRITE(6,1100)(CD0(I,J),J=1,20),I=1,10) 27660 PRITE(6,1100)(CD0(I,J),J=1,20),I=1,10) 27660 PRITE(6,1120)(CD0(I,J),J=1,20),I=1,10) 27660 PRITE(6,1120)(CD0(I,J),J=1,20),I=1,10) 27660 PRITE(6,1120)(CD0(I,J),J=1,20),I=1,10) 27760 PRITE(6,1120)(CD0(I,J),J=1,20),I=1,10) 27760 PRITE(6,1120)(CD0(I,J),J=1,20),I=1,20) 27760 PRITE(6,1120)(CD0(I,J),J=1,20),I=1,20) 27760 PRITE(6,1120)(CD0(I,J),A(I+1)) 27720 PRITE(6,120)(CD0(I,J),A(I),A(I+1) 27730 PRITE(A,TACL,FERCHARDEN))>.001) 27760 PRITE(A,TACL,FERCHARDEN 27780			
100 PDEV(T,K)=SIGNCF(4.*SSS(I+1,J)-3.*SSS(I,J)-SSS(I+2,J)) 27500 100 PDEV(T,K)=SIGNOF(-4.*SSS(I-1,J)+3.*SSS(I,J)+SSS(I-2,J)) 27610 115 CONTINUE 27630 120 IF(SSS(I,J),GE,A(L),AND,SSS(I,J),LE,A(L+1)) CON(I,J)=AL(L) 27640 HRITE(6,1000)IP1,IP2 27650 27660 HRITE(6,1100)(CPDEV(I,J),J=1,20),I=1,10) 27660 27660 HRITE(6,1100) 27660 27660 D0 130 I=1,10 27660 HRITE(6,1100) 27660 27760 D0 130 I=1,10 27660 HRITE(6,1101) 27660 27700 D0 130 I=1,10 27660 27700 HRITE(6,1130),A(I),A(I+1) 27700 IF(I,E0.10) GO TO 130 27700 RETURN 27730 135 HP=IFIX(MPOINT**(1, /FLOAT(NENOP))*.001) 27760 IF(MEVAL,LE,2187) GO TO 150 27780 IF(MEVAL,LE,2187) GO TO 170 27800 K=HOP+1 27800 IF(MCP,E0,7) GO TO 170 27800 K=HOP+1 27800 IF(MCP,E0,7) GO TO 170 27800			
G0 T0 115 27600 10 PDEV(1,K)=SIGNOF(-4.+SSS(1-1,J)+3.+SSS(1,J)+SSS(1-2,J)) 27610 115 CONTINUE 27630 120 L1=1,9 27630 120 HF(SSS(1,J).GE.A(L)AND.SSS(1,J).LE.A(L+1)) CON(1,J)=AL(L) 27640 HRITE(6,100)(CPDEV(1,J),J=1,20),I=1,10) 27660 HRITE(6,1100)(CPDEV(1,J),J=1,20),I=1,10) 27660 HRITE(6,1120)(CON(1,J),J=1,10) 27660 HRITE(6,1120)(CON(1,J),J=1,10) 27660 HRITE(6,1120)(CON(1,J),J=1,0) 27690 IF(1,E0.10) GO TO 130 27700 HRITE(6,1130)1,A(1),A(1+1) 27710 130 CONTINUE 27730 HETURN 27730 HETURN 27730 HETURN 27730 IS WP=IFIX(NPOINT+*(1./FLOAT(NEWNOP))+.001) 27730 IS WP=IP-1 27730 GO TO 140 27730 IF(KCP_E0.7) GO TO 170 27800 IF(KCP_E0.7) GO TO 170 27800 IF(KCP=10.7) GO TO 170 27800 IF(I)=0. 27	100		
110 PDEV(1,k)=SIGNOF(-4,*SSS(1-1,J)*3.*SSS(1,J)*SSS(1-2,J)) 27610 115 CONTINUE 27620 00 120 L=1,9 27630 120 TF(SSS(1,J),GE,A(L),AND.SSS(1,J).LE.A(L+1)) CON(1,J)=AL(L) 27640 WRITE(6,1000)(P),IP2 27660 27660 WRITE(6,1100)(CPDEV(1,J),J=1,20),I=1,10) 27660 WRITE(6,1100)(CON(1,J),J=1,10) 27680 HRTE(6,1100)(CON(1,J),J=1,10) 27680 WRITE(6,1130)I,A(1),A(1+1) 27700 130 CONTINUE 27700 RETURN 27720 RETURN 27750 140 27750 151 CONTINUE 00 TO 140 27760 153 WP=FIX(AN POINT**(1,/FLOAT(NEWNOP))*.001) 27760 154 CONTINUE 27780 155 CONTINUE 27800 155 CONTINUE 27800 156 CONTINUE 27800 157 CONTINUE 27800 158 CONTINUE 27800 150 CONTINUE 27800 150 C		CO TO 115	
115 CONTINUE 27620 120 L20 L20 L20 115 L20 L20 L20 L20 115 L20 L20 L20 L20 116 L20 L20 L20 L20 L20 116 L20 L20 L20 L20 L20 L20 116 L20	110		
D0 120 L=1,9 27630 120 IF(SSS(I,J),GE,A(L),AND,SSS(I,J),LE,A(L+1)) CON(I,J)=AL(L) 27640 WRITE(6,1100)(COPEV(I,J),J=1,20),I=1,10) 27660 WRITE(6,1100)(COPEV(I,J),J=1,20),I=1,10) 27670 D0 130 I=1,10 WRITE(6,1100)(CON(I,J),J=1,10) 27690 IF(I,E0,10) GO TO 130 27700 WRITE(6,1130)I,A(I),A(I+1) 27720 RETURN 27720 RETURN 27720 RETURN 27770 IS NP=IFIX(NPOINT**(1./FLOAT(NEWNOP))*.001) 27760 NP=NP-1 27780 IF(NEVALLE.2187) GO TO 150 27780 NP=NP-1 27780 ISO CONTINUE 27780 IF(NEVALLE.2187) GO TO 150 27780 NP=NP-1 27780 ISO CONTINUE 27800 IF(NEVALLE.2187) GO TO 170 27800 K=MOP+1 27800 IF(NEVALLE.2187) GO TO 170 27800 K=MOP+1 27800 IF(NEVALLE.2187) 27800 IF(NEVALLE.2187) 27800 IF(I)=0. 27800 <	115	CONTINUE	
120 IF(SSS(I, J), GE,A(L),AMD,SSS(I, J),LE,A(L+1)) 27640 WRITE(6,1100) IP2 27650 WRITE(6,1100) (CPDEV(I, J), J=1, 20), I=1, 10) 27660 J0 130 1=1, 10 27690 WRITE(6,1120) GO TO 130 27700 WRITE(6,1120) GO TO 130 27700 WRITE(6,1120) GO TO 130 27700 WRITE(6,1130)1,A(I),A(I+1) 27720 RETURN 27720 RETURN 27750 130 CONTINUE 27760 RETURN 27760 140 WPAL=NP**NENDOP 140 Z7780 150 CONTINUE 760 27780 160 TO 170 K=MOP+1 27780 150 Z7810 160 Z7800 171 Z7810 172 Z7810 173 Z7810 174 Z7810 175 Z7810 176 Z7810 177 Z7810 170 Z7810		D0 120 L=1,9	
HRITE(6,1090)[F],[P2 27650 HRITE(6,1100)((FDEV(I,J),J=1,20),I=1,10) 27660 D0 130 [=1,10 27680 JWRITE(6,1120)(CON(I,J),J=1,10) 27680 JRITE(6,1130)[A(I),J=1,10) 27680 JRITE(6,1130)[A(I),J=1,10) 27690 JF(I,E0,10) GO TO 130 27700 WRITE(6,1130)[A(I),A(I+1) 27720 RETURN 27720 RETURN 27730 135 NP=IFIX(NPOINT++(1,/FLOAT(NENNOP))+.001) 27740 140 NEVAL=NP+*NEHNOP 27750 IF(NEVAL LE, 2187) GO TO 150 27770 NP=HP-1 27780 I50 CONTINUE 27780 150 CONTINUE 27780 150 CONTINUE 27800 IF(NOP-E0.7) GO TO 170 27800 K=HOP+1 27800 D0 160 1=k,7 27800 IF(NOP-E0.7) GO TO 170 27800 IF(I)=0. 27800 160 F2(1)=0. 27800 170 CONTINUE 27800 170 CONTINUE 27800 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27800 SSMIN=1.698 27800	120	IF(SSS(I,J).GE.A(L).AND.SSS(I,J).LE.A(L+1)) CON(I,J)=AL(L)	
WRITE(6,1110) 27670 D0 130 1=1,10 27680 WRITE(6,1120)(CON(1,J),J=1,10) 27690 IF(1,E0.10) G0 TD 130 27700 WRITE(6,1130)1,A(1),A(1+1) 27710 130 CONTINUE 27720 RETURN 27730 135 WP=IFIX(NPDINT**(1./FLOAT(NEWNOP))*.001) 27730 140 NEVAL=HP**WEWNOP 27730 140 NEVAL=L2:187) G0 TO 150 27760 NP=HP-1 27770 G0 TO 140 27780 150 CONTINUE 27780 IF(NOP.E0.7) G0 TO 170 27800 K=HOP*1 27800 D0 160 1=k,7 27800 F1(1)=0. 27800 160 F2(1)=0. 27800 170 CONTINUE 27800 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27800 190 DEL(1)=F262(1)-F1(1))/FLOAT(NP-1) 27800 180 DEL(1)=F2636 27800		WRITE(6,1090) IP1, IP2	
D0 130 1=1,10 27680 WRITE(6,1120)(CON(1,J),J=1,10) 27690 IF(1,E0,10) G0 T0 130 27700 WRITE(6,1130)1,A(1),A(1+1) 27710 130 CONTINUE 27720 RETURN 27730 135 NP=IFIX(NPOINT+*(1./FLOAT(NEWNOP))+.001) 27740 140 NEVAL=NP+*NEUMOP 27750 IF(MEVAL.LE.2187) G0 T0 150 27760 NP=NP-1 27770 G0 T0 140 27780 150 CONTINUE 27790 IF(NCP-E0.7) G0 T0 170 27800 K=HOP+1 27900 K=HOP+1 27900 K=HOP+1 27800 160 F2(1)=0. 27820 160 F2(1)=0. 27830 160 F2(1)=0. 27830 170 CONTINUE 27830 170 CONT		HRITE(6,1100)((PDEV(1,J),J=1,20),I=1,10)	
WRITE(6,1120)(CON(1,J),J=1,10) 27600 IF(1.E0.10) GO TO 130 27700 WRITE(6,1130)1,A(1),A(1+1) 27710 130 CONTINUE 27720 RETURN 27730 135 NP=1F1X(NPDINT**(1./FLOAT(NEWNOP))*.001) 27740 140 NEVAL=NP**NEWNOP 27750 150 CONTINUE 27760 NP=NP-1 27770 GO TO 140 27780 150 CONTINUE 27780 IF(NEVAL.LE.2187) GO TO 150 27760 NP=NP-1 27770 GO TO 140 27780 150 CONTINUE 27780 IF(NEVAL.KE,7 27800 K=NOP+1 27810 DO 160 1=k,7 27820 F1(1)=0. 27820 160 F2(1)=0. 27820 DO 160 1=k,7 27820 F1(1)=0. 27820 170 CONTINUE 27820 D0 180 I=1,7 27860 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27820 SSMIN=1.e98 27880			
IF(I.E0.10) G0 T0 130 27700 WRITE(6,1130)I,A(I),A(I+1) 27710 130 CONTINUE 27720 RETURN 27730 15 WP=IFIX(NPOINT**(1./FLOAT(NEWNOP))*.001) 27730 15 WP=IFIX(NPOINT**(1./FLOAT(NEWNOP))*.001) 27750 140 NEVAL=NP**NEWNOP 27750 140 NEVAL=NP**NEWNOP 27760 NP=NP-1 27770 G0 T0 140 27780 150 CONTINUE 27780 IF(NEVAL.LE.2187) GO TO 170 27800 K=HOP+1 27780 D0 160 1=k,7 27810 F1(1)=0. 27830 160 F2(I)=0. 27830 170 CONTINUE 27830 D0 180 I=1,7 27800 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27800 SMIN=1.E98 27880			
WRITE(6,1130)1,A(1),A(1+1) 27710 130 CONTINUE 27720 RETURN 27730 135 NP=1F1X(NPOINT**(1./FLOAT(NEWNOP))*.001) 27740 140 NEVAL=NP**NEWNOP 27750 176 NP=NP-1 27760 NP=NP-1 27770 150 C0NTINUE 27780 150 C0NTINUE 27780 150 C0NTINUE 27780 150 C0NTINUE 27780 150 CONTINUE 27800 16(NSP=E0.7) GO TO 170 27800 17(1)=0. 27810 160 F2(1)=0. 27820 160 F2(1)=0. 27820 170 CONTINUE 27820 160 F2(1)=0. 27820 170 CONTINUE 27820 170 D0 160 1=k,7 27820 170 CONTINUE 27820 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27820 180 DEL(1)=(F26) 278		WRITE(6,1120)(LUN(1,J),J=1,10)	27690
130 CONTINUE 27720 RETURN 27730 135 NP=IF1X(NPOINT**(1./FLOAT(NEWNOP))*.001) 27730 140 NEVAL=HP**WEWNOP 27750 140 NEVAL=HP**WEWNOP 27760 NP=NP-1 27770 G0 T0 150 150 CONTINUE 27780 150 CONTINUE 27780 150 CONTINUE 27780 150 CONTINUE 27800 160 1=k,7 27820 160 1=k,7 27820 160 F2(1)=0. 27830 170 CONTINUE 27830 160 F2(1)=0. 27830 170 CONTINUE 27830 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27820 SMIN=1.E98 27880			
RETURN 27730 135 NP=IFIX(NPOINT+*(1./FLOAT(NEWNOP))*.001) 27740 140 NEVAL=MP**MEWNOP 27750 IF(MEVAL=LE.2187) GD TO 150 27760 NP=NP-1 27770 GO TO 140 27780 150 CONTINUE 27780 150 CONTINUE 27800 K=HOP+1 27810 D0 160 1=k,7 27820 F1(1)=0. 27820 160 F2(1)=0. 27820 170 CONTINUE 27820 D0 180 I=1,7 27860 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27820 SMIN=1.E98 27820	170		
135 NP=1F1X(NP01NT+*(1./FLOAT(NEWNOP))+.001) 27740 140 NEVAL=MP**NEWNOP 27750 IF(NEVAL.LE.2187) GO TO 150 27760 NP=MP-1 27770 27770 GO TO 140 27780 150 CONTINUE 27780 IF(NOP.E0.7) GO TO 170 27810 K=HOP+1 27810 27810 D0 160 1ek,7 27820 F1(1)=0. 27830 27830 160 F2(1)=0. 27840 170 CONTINUE 27840 170 27840 27840 170 27840 27840 170 27860 27860 180 DE1(1)=(F1(1))/FLOAT(NP-1) 27870 SSMIN=1.E98 27880 27880			
140 NEVAL=HP*+MEMOP 27750 IF(MEVAL.LE.2187) G0 T0 150 27760 NP=NP-1 27770 G0 T0 140 27780 150 CONTINUE IF(NOP.E0.7) G0 T0 170 27800 k=AOP+1 27810 D0 160 1=k,7 27820 F1(1)=0. 27830 160 F2(1)=0. 27830 170 27820 160 F2(1)=0. 27830 170 CONTINUE 27830 160 F2(1)=0. 27840 170 CONTINUE 27850 D0 180 I=1.7 27860 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27870 SMIN=1.E98 27880	135		
IF(MEVALLE.2187) G0 T0 150 27760 MP=NP-1 27770 150 C0NTINUE 27780 IF(NSP_E0.7) G0 T0 170 27800 k=NOP+1 27810 D0 160 1=k,7 27820 F1(1)=0. 27830 160 F2(1)=0. 27830 D0 180 [=1,7 27850 D0 180 [=1,7 27860 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27820 SSMIN=1.E98 27880	140	NEVAL =NP++NEWNOP	
MP=MP-1 2770 G0 T0 140 27780 150 CONTINUE 27700 IF(NOP:E0.7) G0 T0 170 27800 k=NOP+1 27810 D0 160 1=K,7 27820 F1(1)=0. 27830 160 F2(1)=0. 27840 170 CONTINUE 27850 D0 180 I=1,7 27860 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27870 SSMIN=1.E98 27880	10.000	IF(NEVAL.LE.2187) GO TO 150	
G0 T0 140 27280 150 CONTINUE 27790 IF(NGP_E0.7) G0 T0 170 27800 k=HOP+1 27810 D0 160 1=k,7 27820 F1(1)=0. 27830 160 F2(1)=0. 27830 170 CONTINUE 27830 D0 180 I=1,7 27850 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27870 SMIN=1.E98 27800		NP=NP-1	
150 CONTINUE 27790 IF(NOP:E0.7) GO TO 170 27800 K=HOP+1 27810 D0 160 1=k,7 27820 F1(1)=0. 27830 160 F2(1)=0. 27830 170 CONTINUE 27830 D0 180 I=1,7 27860 180 DE(1)=(F1(1))/FLOAT(NP-1) 27870 SSMIN=1.E98 27880		GO TO 140	
IF(NOP:E0.7) G0 T0 170 27800 k=NOP+1 27810 D0 160 1=K,7 27820 F1(1)=0. 27830 160 F2(1)=0. 27840 170 CONTINUE 27850 D0 180 I=1,7 27860 180 DE(1)=(F1(1))/FLOAT(NP-1) 27870 SSMIN=1.E98 27880	150		27790
D0 160 1=k,7 27820 F1(1)=0. 27830 160 F2(1)=0. 27830 170 CONTINUE 27850 D0 180 I=1,7 27860 180 DE(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27870 SSMIN=1.E98 27880			
F1(1)=0. 27830 160 F2(1)=0. 27840 170 CONTINUE 27850 D0 180 I=1.7 27860 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27870 SMIN=1.E98 27880			
160 F2(1)=0. 27840 170 CONTINUE 27850 D0 180 E1.7 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27860 SSM1N=1.E98 27880			
170 CONTINUE 27850 D0 180 I=1,7 27860 180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27870 SSM1N=1.E98 27880	140		
D0 180 T=1,7 180 DEL(J)=(F2(I)-F1(I))/FLOAT(NP-1) SSMIN=1.E98 27860	170	CONTINUE	
180 DEL(1)=(F2(1)-F1(1))/FLOAT(NP-1) 27870 SSMIN=1.E98 27880	170		
SSMIN=1.E98 27880	180		
	100		
LIUTY			

P03

FNEW(I)=F1(I) STPNEW(I)=STEP(I)	27050 27060	
IF(1.GT.IP1) GO TO 55 IF(F1(1).NE.F2(1)) IP1=1	27070 27080	
55 IF(F1(I).NE.F2(I)) IP2=1 DEL(IP1)=(F2(IP1)-F1(IP1))/9.	27090 27100	
DEL(IP2)=(F2(IP2)-F1(IP2))/9. ISAVE=1	27110 27120	
JSAVE=1 SSMIN=1.E99	27130 27140	
SSMAX=0. D0_60_J=1,10	27150 27160	
FNEW(IP1)=F1(IP1)+FLOAT(J-1)*DEL(IP1) P1(J)=FNEW(IP1) P1(J)=FNEW(IP1)	27170 27180	
D0 60 I=1,10 FNEW(IP2)=F1(IP2)+FLOAT(I-1)+DEL(IP2) P2(1)=FNEW(IP2)	27190 27200 27210	
CALL LSQ(FNEW,SSS(1,J)) IF(SSS(1,J),GE.SSMIN) GO TO 60	27220 27230	
SSMIN-SSS(1,J) ISAVE=1	27240 27250	
JSAVE=J 60 IF(SSS(1,J).GT.SSMAX) SSMAX=SSS(1,J)	27260 27270	

	FNEW(IP1)=P1(JSAVE)	27280
	FNEH(IP2)=P2(ISAVE)	27290
	STPNEW(IP1)=DEL(IP1)/2.	27300
	STPNEW(IP2)=DEL(IP2)/2.	27310
	D=(SSMAX-SSMIN)/9.	27320
-	D0 65 I=1,10	27330
65	A(I)=SSMIN+(I-1)+D	27340
	A(10)=A(10)+0.01+D	27350
	WRITE(6, 1030) IP1, (I, P1(I), I=1, 10)	27360
	WRITE(6, 1040) IP2, (I, P2(I), I=1, 10)	27370
	IF(ANS16.EQ.2HNO) WRITE(6,1050)	27380
	IF(ANS16.EQ.3HYES) WRITE(6,1060) WRITE(6,1070)IP1,IP2	27390
	WRITE(6, 1075)(1, I=1, 10)	27400 27410
	D0 67 I=1,10	27420
	WRITE(6,1080)(SSS(1,J),J=1,9,2)	27430
67	WRITE(6, 1085)(SSS(1, J), J=2, 10, 2)	27440
	DO 120 J=1,10	27450
	D0 120 1=1,10	27460
	IF(J.EQ.1) GO TO 70	27470
	IF(J.EQ.10) GO TO 80	27480
	PDEV(1, J)=SIGNOF(SSS(1, J+1)-SSS(1, J-1))	27490
	GO TO 90	27500
70	PDEV(1, J)=SIGNOF(4.*SSS(1, J+1)-3.*SSS(1, J)-SSS(1, J+2))	27510
	GO TO 90	27520
80	PDEV(1, J)=SIGNOF(-4.*SSS(1, J-1)+3.*SSS(1, J)+SSS(1, J-2))	27530
90	K=J+10	27540
	IF(I.EQ.1) GO TO 100	27550
	IF(I.EQ.10) GO TO 110	27560
	PDEV(1,K)=SIGNOF(SSS(1+1,J)-SSS(1-1,J)) G0 T0 115	27570
100	PDEV(1,K)=SIGNCF(4.*SSS(1+1,J)-3.*SSS(1,J)-SSS(1+2,J))	27580 27590
	GO TO 115	27600
110	PDEV(1,K)=SIGNOF(-4.*SSS(1-1,J)+3.*SSS(1,J)+SSS(1-2,J))	27610
115	CONTINUE	27620
10.02	D0 120 L=1,9	27630
120	IF(SSS(I, J).GE.A(L).AND.SSS(I, J).LE.A(L+1)) CON(I, J)=AL(L)	27640
	WRITE(6, 1090) IP1, IP2	27650
	HRITE(6,1100)((PDEV(1,J),J=1,20),I=1,10)	27660
	WRITE(6,1110)	27670
	D0 130 1=1,10	27680
	WRITE(6,1120)(CON(1,J),J=1,10)	27690
	IF(1.E0.10) GO TO 130	27700
170	WRITE(6,1130)1,A(1),A(1+1) CONTINUE	27710
130	RETURN	27720 27730
135	NP=IFIX(NPOINT**(1./FLOAT(NEWNOP))+.001)	27740
140	NEVAL=NP**NEWNOP	27750
	IF(NEVAL.LE.2187) GO TO 150	27760
	NP=NP-1	27770
	GO TO 140	27780
150	CONTINUE	27790
	IF(NOP.E0.7) GO TO 170	27800
	K=NOP+1	27810
	D0 160 1=K,7	27820
	F1(1)=0.	27830
100	F2(1)=0.	27840
170	CONTINUE	27850
180	D0 180 I=1,7 DEL(I)=(F2(I)-F1(I))/FLOAT(NP-1)	27860
180	SSMIN=1.E98	27870
	NEVAL=0	27880 27890
		2/070

P03