The determination of non-stationary random vibration response characteristics by numerical simulation techniques.

John McLean Lovell

Ph.D Thesis

Department of Mechanical Engineering

Faculty of Engineering

University of Glasgow

Submitted November 1981

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

ACKNOWLEDGMENTS

The author gratefully acknowledges the invaluable technical contribution so willingly given by Professor J.D. Robson and Dr D.B. Macvean throughout his period of research at the University of Glasgow. Credit is also extended to the Department of Computer Services for the use of their facilities and expertise and also to the Department of Mechanical Engineering under whose auspices this project was conducted.

The editing and typing services afforded by the author's mother were also much appreciated.

CONTENTS

SUMMARY

The technique of sample averaging is considered for application to the non-stationary vibration problem associated with road vehicle ride. Time history realisations of the vehicle response are achieved by a discretised lumped parameter model idealisation simulated on a digital computer. Sets of realisation histories are collated to obtain the overal statistical response characteristics. The road vehicle ride problem is the result of random road roughness exciting the vehicle as it traverses the surface. This dynamic excitation may be considered as a stationary function of time, provided the vehicle traverse velocity does not vary. Under variable velocity conditions the excitation is a non-stationary function of time. It is the solution of this non-stationary accelerating vehicle problem which is the subject of this study.

An alternative method of solution for the non-stationary vehicle problem has already been achieved. This alternative, like sample averaging, places heavy emphasis on the use of numerical methods on a digital computer for the evaluation of results. Unlike sample averaging, it is not normally applicable to road vehicles which possess significant non-linear dynamic characteristics in their suspension configuration.

Ultimately the objective of this thesis is to make a comparative appraisal of the viability of sample averaging as a general means of determining the non-stationary response characteristics of road vehicles. To permit full justification of the technique and thereby ensure flexibility of application, it is imperative that all methods of digital simulation are scrutinised prior to implementation.

In essence the simulator consists of two distinct numerical modules. One module is concerned with the generation of a large sample of statistically independent road surface profile realisations, while the other applies itself to analysing vehicle response. The additional problems encountered when interfacing the two modules are also fully investigated. Upon implementation, the simulator proves itself a flexible and viable tool for the solution of the non-stationary problem while providing some surprisingly new observations.

LIST OF FIGURES

LIST OF FIGURES (contd)

LIST OF FIGURES (contd)

LIST OF FIGURES (contd)

LIST OF FIGURES (contd)

LIST OF TABLES

NOTATION

General

CDF	cumulative density function, $\int_{-\infty}^{V} Pr(v)\,dv$
C	damping coefficient
E	true expected value, $\int_{-\infty}^{\infty} x(t)Pr(x)dx$
e_n	truncation error at increment n
h	incremental step length
k	stiffness coefficient
L	spatial lag
m	mass coefficient
P(t)	force acting on dynamic system at time, t
$Pr(V:V<V_i)$	probability of event V being less than V_i
$R_q(L)$	Auto - correlation q.q at lag L
$R_{pq}(L)$	Cross - correlation p.q at lag L
$O(h^P)$	Truncation error of order P
$S_q(f)$	f dependant power spectral density of quantity q.
t	time
Var(r)	variance of r
x	spatial or traverse distance
$\dot{x}$	traverse velocity

NOTATION (contd)

General

y	response displacement
z	input displacement
$\|\alpha(\omega)\|^2$	square of complex receptance
ω	angular frequency
ω_n	natural angular frequency
ζ	damping ratio
$\langle x(t) \rangle$	approximate expected value, $\underset{T \Rightarrow \infty}{\text{Limit}} \frac{1}{2T}\int_{-T}^{T} x(t)dt$
$\binom{k}{e}$	$\frac{k!}{e!(k-e)!}$
$\underset{\sim}{\underset{\sim}{\ }}$	a matrix
$\underset{\sim}{\ }$	a vector
$\dot{\ }$	differential w.r.t. time, t eg. $\dot{y} = \dot{y}(t) = \frac{dy}{dt}$
$'$	differential w.r.t. traverse distance, x eg. $\acute{z} = \acute{z}(x) = \frac{dz}{dx}$

LOCAL

Chapter I

$H(t)$	impulsive receptance of dynamic system

Chapter II

$\underline{A}$	coefficient matrix defined in eqn. (6.6)
$\underset{\sim}{a}$	coefficient vector defined in eqn. (4.4)
$\overline{\underset{\sim}{a}}$	coefficient vector defined in eqn. (6.6)
a_i	weighting coefficient
$\underline{B}$	weighting matrix defined in eqn. (5.4)
$\overline{\underline{B}}$	weighting matrix defined in eqn. (5.5)
$\underset{\sim}{b}$	weighting vector defined in eqn. (5.4)
B_w^j	discrete j^{th} backward shift operator
b_i	weighting coefficient
C_i	discretised convariance at lag increment i
$\underline{D}$	coefficient matrix defined in eqn. (5.5)
f	time dependant frequency
$F(S)$	Laplace transform of $f(t)$
$f(t)$	time dependant function
F_w^j	discrete j^{th} forward shift operator
ε_i	complex roots of characteristics polynomial
h_i	complex roots of polynomial eqn. (3.12)
P	scalar quantity defined in eqn. (6.19)

LOCAL

Chapter II(contd)

Q	scaling factor defined in eqn. (2.1)
$\underline{R}$	correlation matrix defined in eqn. (4.4)
R_k	discretised correlation value at lag increment k
$\overline{R}_n$	summation series defined by eqn. (3.5)
$\overline{R}(S)$	Laplace transform of $\overline{R}_n$
$\underset{\sim}{r}$	correlation vector defined by eqn. (4.4)
$\underset{\sim}{\overline{r}}$	correlation vector defined by eqn. (5.4)
$\underset{\sim}{\overline{r}}(p)$	correlation vector defined by eqn. (6.6)
r_n	random input at increment number n
S	Laplace operator
$TF(S)$	transfer function in Laplace domain
W	complex operator defined by eqn. (3.13)
λ	complex operator defined by $\lambda = \exp(S)$

Chapter III

$\underline{A}$	elastic stiffness matrix
a_i	weighting coefficient
b_j	weighting coefficient
$\underset{\sim}{E}$	elastic energy vector
$\underset{\sim}{0}$	null vector

LOCAL

Chapter III (Contd)

M	constant primitive root multiplier
r_{RNG}	output from pseudo random number generator
r_S	random variable defined by eqn. (4.1)
r_G	Gaussian random variable obtained from (4.2) or (4.5)
r_R	random variable defined by (4.3)
r_P	random variable defined by (4.4)
$r_{G,n}$	Gaussian random variable at increment number n
S_N	RNG seed (suffix '0' for old, 'N' for new)
SM	product $S_o.M$
W	constant - a function of computer word length
z	input displacement vector

Chapter IV

B	scalar constant
c	scalar constant
d	scalar constant
N_{max}	maximum lag increment number of AR filter
N	lag increment number
M	sample size

LOCAL

Chapter V

a_i	weighting coefficient
b_i	weighting coefficient
c_i	weighting coefficient
$F(\ )$	the differential equation
$f(\ )$ or f	"function" of differential equation
$\overline{f}(\ )$	finite difference approximation to y_n
f^i	Jacobian of "function" ie $\frac{\partial f}{\partial i}$
$f^{(p)}$	p^{th} estimate of "function"
$G(\ )$	a discretised function
K	Lipschitz constant
q	scalar value
r_i	roots of characteristic equation (7.1) or (7.2)
y_m	approximate solution at increment m, step length h
$\overline{y}_m$	approximate solution at increment m, step length $h/2$
$y(t_m)$	true solution at time t_m
$y_m^{(5)}$	5th order approximation to solution at increment m
$y_m^{(p)}$	Predictor approximate solution at increment m
$y_m^{(c)}$	Corrector approximate solution at increment m

LOCAL

Chapter V (contd)

$\rho(r)$	a_i coefficient polynomial from characteristic equation, defined by (7.1) and (7.2)
$\acute{\rho}(r)$	derivative of $\rho(r)$ w.r.t. r

Chapter VI

$A_{i,j}(m)$	scalar defined by (4.6)
$\underset{\sim}{a}_m^{(j)}$	vector defined by (4.9)
$B_i(m)$	scaler defined by (4.5)
$\underset{\sim}{c}$	vector defined by (4.10)
$E_i(m)$	scalar defined by (4.8)
$e_n^{(p)}$	predictor truncation error at step n
$e_n^{(c)}$	corrector truncation error at step n
$F_i(m)$	scalar defined by (4.4)
$G_i(m)$	scalar defined by (4.3)
$\underline{P}$	Pascal matrix defined by (4.9)
Δf	scalar defined by (4.10)
$\nabla^k y_m$	k^{th} order backward difference operator defined in section 3.

Chapter VII

A	scalar constant defined in section 2
a_m	scalar variable used by Nordsieck

Chapter VII (Contd)

$\underline{B}$	matrix utilised in (5.3)
B	scalar constant defined in section 2
b_m	scalar variable used by Nordsieck
C	scalar constant defined in section 2
c_m	scalar variable used by Nordsieck
D	scalar constant defined in section 2
d_m	scalar variable used by Nordsieck
D_f	difference between j^{th} and $(j+1)^{th}$ "function" evaluation
$\underset{\sim}{E}$	constant vector defined by (2.12)
$\underset{\sim}{e}_m$	constant vector defined by (2.11)
F_m	scalar $(f_{m+1} - f_m)$
$\underset{\sim}{g}$	constant vector defined by (2.11)
$G(\underset{\sim}{y}_m^{(i)})$	the corrector function defined by (5.1)
$\underline{I}$	unit matrix
K	scaling factor
$\underline{M}$	constant matrix defined by (2.12)
$\underline{0}$	null matrix
q	scalar approximation to Jacobian $h\,\partial f/\partial y$

Chapter VII(Contd)

Q	(superscript) power of non linearity
r_k	constant used in stability analysis
$\underline{S}_m$	stability matrix defined by (5.4)
$\underset{\sim}{T}_i$	vector defined by (5.2)
$\underline{T}$	matrix defined by (5.5)
$\underline{U}$	constant matrix defined by (2.11)
$\underset{\sim}{V}$	constant vector defined by (2.11)
$\underset{\sim}{v}_m$	variable vector defined by (2.11)
Y	scalar constant defined in section 2
λ	variable of characteristic equation.

Chapter VIII

$\lvert E_a \rvert$	modulus absolute error
H	step length ratio
T	natural period
$y^{(p)}$	p^{th} estimate of y

Chapter IX

h_{int}	interpolated step length
INT(X)	nearest integer value to X
K	spatial angular wave number

LOCAL

Chapter IX (Contd)

N_{int}	number of interpolations
z_{int}	interpolated input displacement
λ	spatial wavelength

Chapter X

B	scalar constant
K	spatial angular wave number
K_o	lowest recorded K
N	total number of realisations in sample
T	time lag
λ	spatial wavelength

ABBREVIATIONS

AR	Autoregressive
ARMA	Autoregressive moving average
Averaged MS	Sample averaged mean square value for all increments and all sampled realisations
CDF	Cumulative density function
DE	Ordinary differential equation (initial value problem)
FFT	Fast Fourier Transform
LSDE	Linear stochastic difference equation
MA	Moving average
MS	Mean square
NS	Non-stationary
PC	Predictor Corrector
PSD	Power spectral density
RCI	Response convolution integral method
RK	Runge Kutta
RNG	(pseudo) Random number generator
SA	Sample average at a fixed instant
SAMS	Sample average mean square at a fixed instant.

CHAPTER I

THE NON-STATIONARY VEHICLE PROBLEM - AN OVERVIEW

1. Problem justification and objectives

In the field of road vehicle engineering, vibrational response is an important consideration. Vibrational response is of fundamental concern to passenger comfort and to component stressing and fatigue. The main source of vertical excitation is caused by road surface roughness transmitted through the vehicle suspension system to the vehicle itself. Optimal design of the suspension system is consequently of considerable importance. The dynamic parameters of both vehicle and suspension influence the optimisation, as does the nature of excitation caused by road surface roughness.

Much work has been done in the statistical description of road surface roughness. Strong evidence exists to support the claim that vertical undulations on a road surface profile can be considered stochastically stationary with respect to traverse (or spatial) distance along this profile. Should the road vehicle traverse this profile at constant velocity, the resultant vertical input excitation to the base of the vehicle suspension can be considered stochastically stationary with respect to both time and spatial distance. (Stationarity with respect to both time and space follows automatically from the linear relationship, at constant traverse velocity, between both time and traverse distance.) Thus at constant velocity, a linear suspension system can be analysed using the standard simplifications of stationary stochastic dynamics (see eg Ref. 1). However, under variable traverse velocity conditions the input excitation ceases to be stationary with respect to time. Since the suspension system has a vibrational response which is time dependant, then non-stationarity with respect to time adds considerably to the complexity of any stochastic dynamic analysis. It is the analysis of such non-stationary suspension response that is of concern in this thesis.

Several interesting solutions of displacement response characteristics of road vehicles subject to varying velocity have been presented before (Ref. 2 and 3). Their solutions are specific to particular

road profile correlations and rely on the evaluation of complicated double integrals with infinite bounds. Evaluation requires specialist knowledge and cannot therefore be readily applied by the vehicle design engineer to all practical road profile correlations and design configurations. It is the objective of this thesis to investigate the practicability of sample averaging at fixed time instants, a set of computer simulated realisation histories (henceforth referred to as the sample average (SA) technique) as an alternative design tool for the non-specialist design engineer.

This SA technique has certain inherent advantages over the evaluation of complicated correlation double integrals. All information concerning each realisation is available at little extra effort for further analysis and interpretation. Thus, not only can information concerning the mean square displacement response be obtained, but also information concerning vertical velocity and acceleration response characteristics and their associated probability distributions. As the analysis is done by determining transient response realisations, it should be possible to cater for non-linear suspension systems. The non-definitive nature of statistical information from a sample of deterministic realisations is the major weakness in this approach. However, this aspect is also fully investigated.

Before launching into the detailed arguments for the computer simulation approach employed, a brief justification must be given for choosing to simulate on a computer rather than by physical model tests. Two arguments apply. The first concerns flexibility; a computer simulated model is readily amenable to variation in the physical parameters and obviates the need to have such "variation" components manufactured. Secondly, computer simulation is far more cost effective.

A digital rather than a hybrid computer was chosen for two main reasons. The type of hybrid which would lend itself to this type of problem was simply not available at that time. Secondly, specialist hybrid machines are far more difficult to justify than general purpose digital machines. Consequently, digital machines are far more widely

distributed and correspondingly interest in digital solution techniques is also greater.

The objectives of this thesis are therefore to investigate the viability of sample averaging as an alternative approach to the non-stationary vehicle problem. As a result of the additional statistical information made available by this SA technique, existing knowledge on the non-stationary behavioural characteristics of road vehicles is extended.

2. Formation of the simplified vehicle model

From the point of view of dynamic excitation a road vehicle can be regarded as an elastic frame or monocoque structure mounted on several heavily damped spring suspension units. The relative stiffness and mass ratios of the frame compared with the suspension is such that the frame can be considered a rigid mass when the suspension system is analysed. A lumped parameter model of the vehicle suspension can therefore be considered. In this idealised system all stiffness and damping parameters are lumped on the suspension while most inertia forces are attributed to the rigid frame.

Several suspension units support the frame. In a full dynamic analysis of the vehicle suspension all relevant degrees of freedom and the interaction (or cross coupling) between them must all be modelled. However in establishing the viability of sample averaging simulated time realisations, the majority of the fundamental problems which require proving are exhibited in a one degree of freedom lumped parameter model. (A one degree of freedom model is also desirable for comparison with the aforementioned existing solutions.) Analysis of a one degree of freedom model is therefore considered sufficient as vindication of the viability of the sample averaging technique.

The description of road surface roughness is an important aspect of the vehicle suspension model as it is the source of excitation. Normally only vertical input excitations are included in the model simulation. There are two reasons for this. Firstly, suspension units are designed to be more or less rigid with respect to all

lateral movement. Secondly, the vehicle configuration ensures that any lateral input excitation along the line of vehicle traverse, would either be absorbed as rotational energy to the wheel alone, with negligible transmission to the suspension unit, or damped out by the tyre itself.

A considerable number of investigations have been undertaken to establish the statistical characteristics of vertical displacement road roughness. There is strong evidence to suggest that the vertical displacement road surface roughness adheres to a Gaussian probability distribution. However the spatial displacement auto correlations (or equivalent spectral wave number characteristics) vary depending on the class of road under consideration. It is usual therefore to define a road profile in terms of either the spatial displacement auto correlation or wave number spectrum and assume the probability distribution is Gaussian.

A mathmetical description of the one degree of freedom vehicle suspension model can now be formulated (See fig 1.1). (In future for brevity this model will be referred to as the vehicle model.) The vertical equation of motion of the vehicle model takes the form

$$m\ddot{y}(t) + c\dot{y}(t) + ky(t) = c\dot{z}(t) + kz(t) \qquad (2.1)$$

where
- m = lumped inertial mass coefficient of the vehicle
- c = viscous damping coefficient of suspension
- k = stiffness coefficient of suspension
- $y(t)$ = time dependant vertical displacement response
- $z(t)$ = time dependant vertical displacement input excitation.

The single dot above the variables $y(t)$ and $z(t)$ denotes the derivative with respect to time

eg. $\dot{z}(t) = \frac{dz(t)}{dt}$

while the double dot denotes the second order time derivative

eg. $\ddot{y}(t) = \frac{d^2y(t)}{dt^2}$

The input excitation displacement and velocity, normally described in terms of the spatial road profile, are thus defined in terms of

the horizontal traverse distance, x. Consequently vertical displacement and velocity (or displacement gradient) are denoted respectively by

$$z(x) \text{ and } \acute{z}(x) = \frac{dz(x)}{dx}$$

These quantities are related to their time based counterparts, $z(t)$ and $\dot{z}(t)$, by the instantaneous traverse velocity $\dot{x}(t)$, in accordance with the following simple formulae

$$z(t) = z(x(t)) \tag{2.2}$$

$$\dot{z}(t) = \frac{dz(t)}{dt} = \frac{dz(x)}{dx} \cdot \frac{dx}{dt}$$

$$= \acute{z}(x) \cdot \dot{x}(t) \tag{2.3}$$

Note Henceforth, the superscript dot ($\dot{}$) will refer to the derivative with respect to time, while the superscript prime ($'$) will refer to the derivative with respect to the spatial variable, x.

It is also convenient to use the term spatially stationary to refer to quantities which are stationary with respect to 'x', while the term stationary refers to quantities which are stationary with respect to 't'.

3. Methods of analysing the non-stationary vehicle problem

A brief outline of the possible methods of tackling the non-stationary vehicle problem are now given to facilitate more detailed discussion.

The alternative method to sample averaging used by Ref. 2 and 3 relies on the evaluation of complicated double integral formulae to determine the mean square displacement response characteristics. These double integral formulae are obtained from the convolution integral relation

$$y(t) = \int_0^t H(T)z(t-T)dT \tag{3.1}$$

where $y(t)$ = displacement response at time, t

$z(t)$ = input excitation displacement at time, t

$H(t)$ = impulsive receptance of the dynamic system at time, t

which determines the displacement response at time t of a linear

dynamic system initially at rest, and from the relation which defines the response autocorrelation function $R_y(t_1,t_2)$

ie $$R_y(t_1,t_2) = \langle y(t_1).y(t_2) \rangle \qquad (3.2)$$

where $\langle y(t_1).y(t_2) \rangle$ denotes the expected value of the product $y(t_1).y(t_2)$

Substitute (3.1) in (3.2) to obtain the mean square response convolution integral formula

$$R_y(t_1,t_2) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} H(T_1)H(T_2)R_z(t_1-T_1)R_z(t_2-T_2)dT_1dT_2. \qquad (3.3)$$

Normally, the relationship between the elapse time, t, and the traverse distance, x, is explicitly defined, and it is possible to reformulate the right hand side of (3.3) in terms of the spatial variable, x. It is this approach that forms the basis of the investigations of Ref. 2 and 3. As this approach is based on the convolution integral, it is consequently restricted to linear suspension configurations.

The evaluation of the resultant integral formula is an extremely difficult numerical problem. In Ref. 3 the authors achieved simplification by the careful selection of road profile autocorrelation which had a particularly simple analytic form. Their technique required specialist knowledge and is not readily applicable to all road profile correlations. The results obtained upon evaluating these integrals are, ignoring numerical error, definitive on the expected value. This is a quality which sample averaging does not display because of the statistical scatter which must be present in a finite sample. It is the definitive nature of the solution which made this method so attractive to the authors of Ref. 2 and 3.

As yet the only results available by this alternate method concern displacement response values. Should a derivative response correlations be required it would be necessary to repeat the evaluation with a different mean square response convolution integral approximately doubling the work involved. Not only is this approach restricted to linear suspension configurations, but no information at all is available on the probability distribution of the response characteristics.

The technique investigated in this project is that of sample averaging. It relies on the assimilation of statistical information obtained from a set of random time histories simulating vehicle behavioural response. Each input excitation realisation is a generated sample member of a stochastic population which possesses the statistical characteristics (road profile correlation, probability distribution and time variant traverse velocity) for the case in question. Conceptually, the approach is very simple.

Since this approach is based on simulating the transient response histories, inherent advantages result. These advantages are listed below

1) Vertical displacement, velocity, and acceleration response histories are all present in each transient simulation. It is therefore possible to establish all statistical response characteristics (ie mean square values and probability distributions). Such characteristics are important for passenger comfort and component fatigue considerations.

2) As transient solutions are being evaluated, careful design of the numerical factors in the vehicle simulator should allow for the analysis of non-linear vehicle systems.

3) The method of generating realisations of the road profile allows both realistic and elegant analytical correlations to be simulated with equal ease.

Again it must be emphasised that, with this technique of sample averaging, statistical scatter must be anticipated. Consequently the results will be non-definitive. Ultimately it is the magnitude of the tolerance achieved by a physically manageable sample size which will decide the viability of sample averaging as the method of solving the non-stationary vehicle problem.

4. Segmentation of the problem

When extracting non-stationary response characteristics by sample averaging the analysis breaks naturally into two main segments. The first of

these is concerned with the design of the overall road vehicle simulator on a digital computer. The other segment is concerned with the statistical interpretation of this sample of simulated realisations.

Investigation of the numerical techniques required to implement the road vehicle simulator comprises the major part of this thesis. The extensive nature of the numerical investigation does not mean inflexibility of application. Once established, the simulator can be readily applied to any design configuration and road profile correlation. It does not therefore affect the technique's usefulness as a design tool.

The problems associated with the road simulator design can be further subdivided as follows:

1) The Road Profile Generator is concerned with the problems associated with the generation of a family of statistically independant random road profile realisations

2) The Dynamic Vehicle Integrator is concerned with the problems of digitally simulating the vehicle response

3) The Road Vehicle Simulator is concerned with the problems of interfacing a spatially stationary road profile generator to a time based dynamic vehicle integrator while including the effects of non-stationarity.

In the sections which follow, the problems associated with simulation and statistical interpretation will be briefly discussed.

5. The Road Profile Generator

Both displacement, $z(t)$, and velocity, $\dot{z}(t)$, are required as input excitations to the vehicle model equation of motion (2.1). The road profile generator is required to generate these input realisations. It is essential that these realisations manifest the correct statistical input characteristics. It is also essential that each realisation is independant of all others in the sampled population to ensure

correct statistical representation.

This requirement for a large number of independant realisations of the road profile prompts the consideration of linear stochastic difference equations (LSDE) as a means of road profile generation. However, by using such difference equations, it is only practical to consider stationary profile realisations. If these generated profiles are made stationary with respect to time it is only possible to consider constant traverse velocity cases which is contrary to the objectives of this thesis. However, if the generators are made spatially stationary then it is possible to include the effects of variable velocity in the equation of motion of the vehicle model. It is this latter option which was selected.

Choosing and implementing a linear stochastic difference equation as a means of generating road profile realisations at spatially equal increments is considered in some depth in the ensuing chapters.

6. The Dynamic Vehicle Integrator

The dynamic vehicle integrator is required to simulate the vehicular response when subjected to prescribed vertical displacement and velocity excitations. The equation of motion of the time based vehicle model is given by (2.1). However, the prescribed excitation inputs are described in terms of spatial increments, x, and so, as mentioned in the previous section, it is necessary to introduce the effects of variable traverse velocity through the equation of motion.

The transformation of (2.1) from a time base to a spatial base (or vice versa) is easily accomplished provided the instantaneous traverse velocity is explicitly defined. In order to simulate the transformed equation of motion it is necessary to consider a numerical integration technique which can accommodate variation of the dynamic parameters of differential equation (2.1). It is also desirable to choose a method which can accommodate non-linear suspension configurations. A search for a suitable integration technique was required.

7. The Road Vehicle Simulator

The problems encountered by having a spatially based road profile generator and a time based dynamic vehicle integrator were briefly outlined in the previous section. It is necessary to consider how these segments can best be interfaced. A spatially based formulation is eventually chosen. The reasons for this decision will be discussed.

8. Interpretation of the Statistical Results

The ability to assess the statistical performance of any arbitrary road surface and vehicle suspension configuration is the ultimate objective of this sample averaging technique. The method's viability rests on the ability to attain a reasonably definitive solution from physically manageable sample sizes. Consequently, the quality of all statistical information obtained by this method must be studied in detail. The problems associated with extracting meaningful answers from manageable simulation sample sizes must also be considered before reaching a conclusion on the method's viability.

Non-stationary results available from Convolution Integral approach provide a means of checking this sampling technique.

9. Conclusion

Consideration of vehicle suspension design is of crucial importance in the reduction of the road vehicle's vibrational response. The primary source of excitation is caused by road surface roughness as the vehicle traverses the road profile. Under variable traverse velocity conditions, the standard simplifications of stationary stochastic dynamics no longer apply and other methods of analysis must be sought.

Unlike the Convolution Integral approach, this sample averaging technique can readily be applied to non linear suspension

configurations. The Convolution Integral approach proves rather unwieldy for the solution of new road profile correlation problems, in particular those profiles with no elegant mathematical form. The sample averaging approach is totally correlation profile independant.

Sample averaging also reveals all additional statistical information (derivative mean square response characteristics, probability distributions) as a natural by-product of the solution method. However, it is the degree of convergence obtained from a limited sample of realisation records that will ultimately determine the viability of the technique.

It was proposed to consider a one degree of freedom vehicle suspension model. A one degree of freedom model displays all the characteristics necessary to ascertain the feasibility of sample averaging for solving the non-stationary vehicle problem.

To ensure proper design of the road vehicle simulator, it is necessary to carefully consider the numerical properties of the various digital simulation aspects. The complexity of designing a road vehicle simulation should not influence the method's ultimate viability, for the design of this numerical tool is a once only operation.

Comparison with existing non-stationary mean square displacement response results provides valuable insight into the nature of statistical scatter present in averaging a set of sample realisations, as well as ascertaining the method's ability to cope with non-stationary problems.

REFERENCES - CHAPTER I

Ref. 1 Robson, J.D., Macvean, D.B., Paling, V.R., Dodds, C.J. (1971) "Random Vibrations" Int. Centre. Mech. Sciences, Course and Lectures. 115, Udine. pub. Springer.

Ref. 2 Virchis, V.J., and Robson, J.D. (1971) "Response of an accelerating vehicle to random road undulation" J. Sound and Vibration. 18, 423 - 427.

Ref. 3 Sobczyk, K., and Macvean, D.B. (1976) "Non-stationary random vibration of road vehicles with variable velocity" Symp. Stochastic. Problems in Dynamics, Int. Union of Theoretical and Appl. Mech., Uni. Southampton, 21.1 - 21.6.

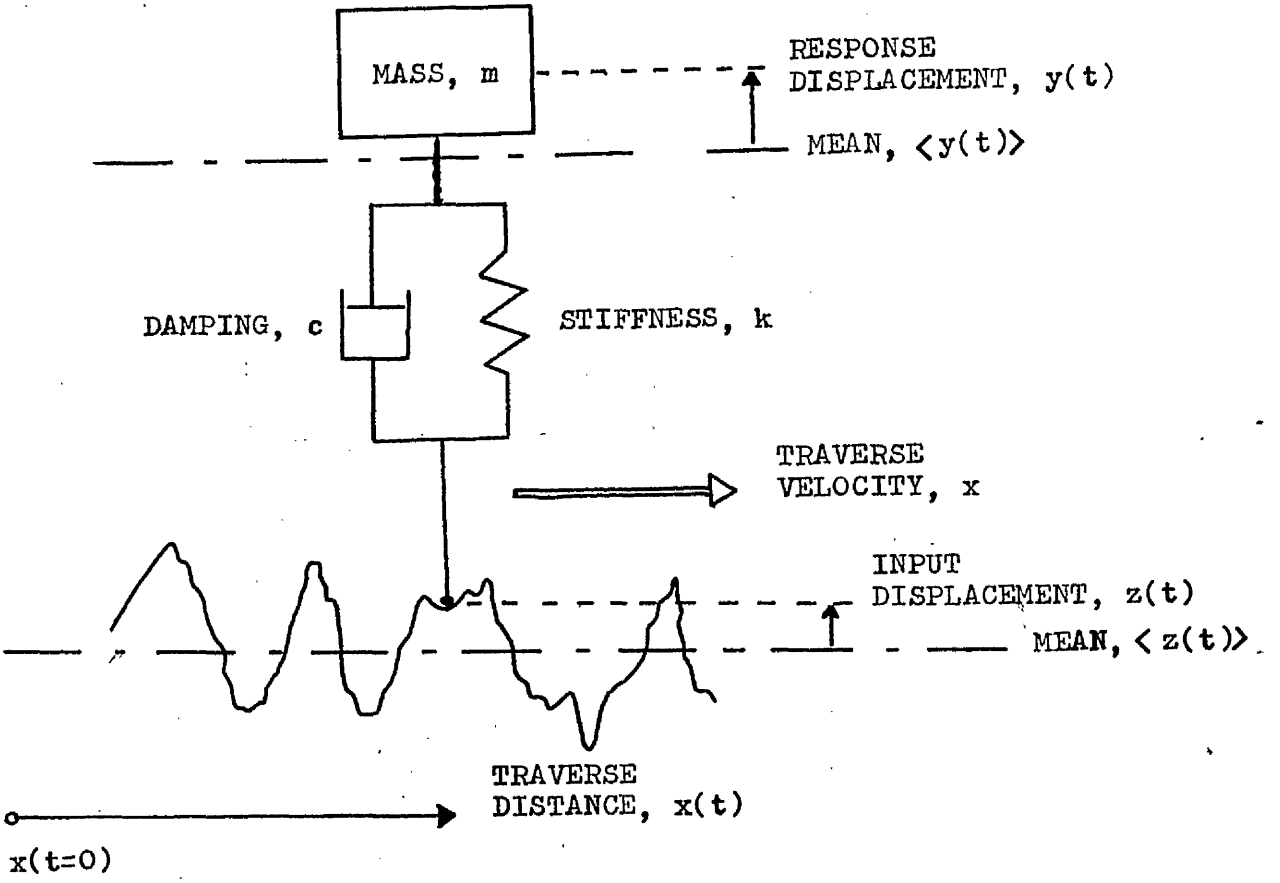


FIG 1.1 Schematic one degree of freedom lumped parameter model of a road vehicle suspension system.

CHAPTER II

USING THE LINEAR STOCHASTIC DIFFERENCE EQUATIONS (LSDE)

1. Introduction

The type of second order ordinary differential equation that requires solution in problems such as the vehicle problem is given by

$$m\ddot{y}(t) + c\dot{y}(t) + ky(t) = c\dot{z}(t) + kz(t) \qquad (1.1)$$

where $y(t)$ is the displacement response at time, t,

$z(t)$ is the random input displacement at time, t,

and $\dot{z}(t)$ is the derivative of $z(t)$ with respect to time.

An equation of the form of (1.1) requires both displacement and the velocity (or slope) road profiles, $z(t)$ and $\dot{z}(t)$ respectively, to be input. Each road profile realisation in the sample must be statistically independant and must adhere to the correct auto- and cross-correlation curves, while at the same time exhibit Gaussian probability distribution. It is proposed to generate independant Gaussian probability distributions by means of a Gaussian random number generator and to use LSDEs to obtain the correct correlations.

It is the selection of the most suitable type of LSDE which is considered in this chapter. Considerations concerning the LSDE are easily the most complex part of the road profile generator. For general reference on the subject consult for example ref. 1 or 2. Discussion on how to obtain independant and correct probability distributions for the profile generator is left until chapter III. As the generator is required to generate both displacements $z(t)$ and velocities $\dot{z}(t)$, the requirement for cross-correlation between these two random signals suggests that multivariate LSDEs must be considered. However, the special relationships between all auto- and cross-correlations (displacement, velocity, displacement - velocity, velocity - displacement) permits a monovariate LSDE to fulfil all generator requirements (see chapter III).

The object of this chapter is therefore to consider the theoretical aspects of the various types of LSDE and assess which can most effectively generate road profile surfaces. It is also desirable to readily establish the filter coefficients when new road correlations are considered.

2. Types available

The road profile is assumed to be of a stationary nature and as a consequence of this only three types of LSDE need be considered.

These types are Autoregressive (AR), Moving Average (MA), and Mixed Autoregressive Moving Average (ARMA) filters

All three types are covered by the general difference equation

$$\sum_{i=0}^{p} a_i z_{n-i} = \frac{1}{Q} \sum_{j=0}^{q} b_j r_{n-j} \qquad (2.1)$$

If $p > 0$ and $q = 0$, then it is an AR of order p.

If $p = 0$ and $q > 0$, then it is an MA of order q.

If $p > 0$ and $q > 0$, then it is an ARMA of order p.q.

The a_i's and b_i's are weighting coefficients and the ratio $1/Q$ is the scaling factor. The suffix 'n' is the current discretised sequence increment, and r_j is a sequence of mutually uncorrelated random variables having a zero mean and a variance, Var(r).

Clearly, the AR and MA models are particular cases of the more general ARMA model. It is however useful to consider the properties of all three cases.

A relationship exists between the "a_i" weighting coefficients of the simple AR filter and the "b_j" coefficients of the simple MA filter. (Refer Appendix A)

This relationship is important in two respects. It demonstrates that a finite MA filter can be replaced by an infinite AR filter

and vice versa, and that the "b_j" weighting coefficients of an MA filter can be directly determined from known equivalent "a_i" coefficients. The converse is also true.

The determination of the "b_j"s from equivalent "a_i"s is crucial in establishing sensible MA and ARMA models.

3. Stability and Invertibility requirements of LSDEs

Stability and/or invertibility are prerequisites for all LSDEs. Consequently these criteria are considered before any detailed discussion on the particular types of LSDEs available. Both requirements are considered together as they are related.

Consider equation (2.1) with the scale factor $1/Q$ set to unity

$$\sum_{i=0}^{p} a_i \, z_{n-i} = \sum_{y=0}^{q} b_j \, r_{n-j} \qquad (3.1)$$

By definition the Laplace Transform, F(s), of any time dependant function, f(t), is given by

$$F(s) = \int_0^{\infty} f(t). \exp(-st)dt \qquad (3.2)$$

Multiply eqn. (3.2) by exp (-khs) to give

$$\exp(-khs)\ F(s) = \int_0^{\infty} f(t) \exp\left[-(t + kh)s\right] dt \qquad (3.3)$$

Let $t = T - kh$ and substitute in (3.3) to obtain

$$\exp(-khs)\ F(s) = \int_0^{\infty} f(T - kh) \exp(-Ts)\, dT \qquad (3.4)$$

Consequently, multiplying the Laplace transform, F(s), by exp(-khs) is equivalent to delaying the time dependant function, f(t), by "k" increments of length h, and vice versa.

The right hand side of (3.1) possesses a non-recursive formulation (ie no feedback properties). Consequently, for the purpose of

stability analysis, it can be regarded as a single input quantity, $\overline{R}_n$ such that

$$\overline{R}_n = \sum_{j=0}^{q} b_j \, r_{n-j} \tag{3.5}$$

Thus, $\overline{R}_n$ can be regarded as the total stable filter input.

Equation (3.1) can now be written in the form

$$\sum_{i=0}^{p} a_i \, z_{n-i} = \overline{R}_n \tag{3.6}$$

Take the Laplace Transform to obtain

$$\sum_{i=0}^{p} a_i \exp(-ihS) \, z(S) = \overline{R}(S) \tag{3.7}$$

From control theory the transfer function, TF(S), in the complex S - plane is given by

$$TF(S) = \frac{z(S)}{\overline{R}(S)} = \frac{1}{\sum_{i=0}^{p} a_i \exp(-ihS)} \tag{3.8}$$

For stability the roots of the denominator in (3.8) must all lie in the negative half of the complex S - plane. If the denominator

$$\sum_{i=0}^{p} a_i \exp(-ihS) = \prod_{i=0}^{p} (1 - g_i^{-1} S) = 0 \tag{3.9}$$

where g_i, for i = 1, 2, -- p, are the roots, then for stability $g_i \leq 0$ for all i. Unfortunately, the form of the denominator makes the factorisation into roots very difficult.

As the filter (3.6) is discrete, time increments, h, elapse between sampling instants. An upper band sampling frequency, $\omega_{s\ max}$, is effectively imposed, such that

$$\omega_{s\ max} = \frac{2\pi}{h} \tag{3.10}$$

All frequencies greater than $\omega_{s\ max}/2$ cannot be processed by the filter.

Thus the effective stable region of the complex S - plane is the region bounded by the lines

$$\left.\begin{aligned} -\infty \leq \mathrm{Re}\,(g_i) &\leq 0 \\ -\,i\frac{\omega_{s\,max}}{2} \leq \mathrm{Im}\,(g_i) &\leq i\frac{\omega_{s\,max}}{2} \end{aligned}\right\} \qquad (3.11)$$

The stable region defined by (3.11) maps into the unit circle in the complex λ-plane if the complex transformation $\lambda = \exp(s)$ is implemented. Thus in the λ- plane the stable region is given by

$$|\lambda| \leq 1$$

Employ this transformation $\lambda = \exp(hs)$ to the denominator in the S - plane (3.9),

hence,

$$\sum_{i=0}^{p} a_i \exp(-ihs) = 0$$

in the S - plane

becomes,

$$\sum_{i=0}^{p} a_i \lambda^i = \prod_{i=0}^{p} (1-h_i^{-1}\lambda) = 0 \qquad (3.12)$$

in the λ- plane, h_i, for all i, are the roots

$$\left.\begin{aligned} &\text{If } ihs = i \ln\lambda = \ln\lambda^i \\ &\text{then } \exp(ihs) = \lambda^i \end{aligned}\right)$$

(3.12) is the characteristic equation.

For stability all roots h_i must lie within the unit circle, ie

$$|h_i| \leq 1$$

Consequently to check for stability it is necessary to solve the characteristic polynomial (3.12) and evaluate the roots g_i.

To solve for roots of high order characteristic polynomials, by numerical means, proves difficult. The polynomials are very

frequently ill conditioned. Roots near the stability bound may therefore be misinterpreted.

To overcome this problem another complex transformation from the λ to the W - plane is implemented by means of the relation

$$\lambda = \frac{W + 1}{W - 1} \tag{3.13}$$

The stable region is transformed from within the unit circle of the λ - plane, to the negative half of the complex W - plane bounded for stability by

$$\text{Re } (W) \quad \leq \quad 0 \tag{3.14}$$

In the W - plane the transformation of the characteristic equation takes the form

$$\sum_{i=0}^{p} a'_i \, w^i = 0 \tag{3.15}$$

The solution of this polynomial is no more easily accomplished than the solution of the characteristic equation (3.12). However, as the stability is Re $(W) \leq 0$, it is ammenable to the implementation of the Routh Hurwitz stability test (see chapter III). By this means the problem of ill conditioning is overcome.

For the stability considerations attention is concentrated on the recursive AR filter part (left hand side) of (3.1) and the inputs r_{n-j} grouped in the single input variable, $\overline{R}_n$, defined in (3.5). However, another criterion must be considered to ensure a sensible filter is obtained. This criterion is known as invertibility.

The relevance of invertibility is demonstrated by focusing attention on the non-recursive MA right hand side of (3.1) so that a filter of form

$$\overline{z}_n \quad = \quad \sum_{j=0}^{q} b_j \, r_{n-j} \tag{3.16}$$

is obtained.

Transform into the λ - plane, to yield

$$\bar{z}(\lambda) = \sum_{j=0}^{q} b_j \lambda^{n-j} . r(\lambda)$$

$$= \prod_{j=0}^{q} (1-g_j^{-1}\lambda)\, r(\lambda) \qquad (3.17)$$

where, g_j^{-1}, for all j, are the roots of the characteristic equation.

(3.17) can be reformulated as

$$\left(\prod_{j=0}^{q} (1-g_j^{-1}\lambda_j) \right)^{-1} \bar{R}(\lambda) = r(\lambda) \qquad (3.18)$$

Expand the left hand side in terms of partial fractions to obtain:-

$$\sum_{j=0}^{q} \frac{k_j}{(1-g_j^{-1}\lambda_j)} \bar{R}(\lambda) = r(\lambda) \qquad (3.19)$$

where K_j is a scalar constant.

The factor $\frac{1}{(1-g_j^{-1}\lambda_j)}$ of (3.19) can be expanded such that

$$\frac{1}{(1-g_j^{-1}\lambda_j)} = 1+(g_j^{-1}\lambda_j) + (g_j^{-1}\lambda_j)^2+\ldots$$

The series is convergent if $|g_j^{-1}|<1$

ie if $|g_j|>1$.

Consequently, by consideration of this constraint imposed on the equivalent AR filter, it becomes obvious that the roots of the characteristic equation, g_j, of an MA filter must satisfy the condition $|g_j|>1$, for j = 0, 1, 2 ... q, to ensure a sensible filter.

4. The Autoregressive (AR) Process

4.1 Derivation of Yule-Walker Relations

The monovariate AR process takes the form

$$z_n = \sum_{i=1}^{p} a_i \, z_{n-i} + r_n \qquad (4.1)$$

where the a_i's are the weighting coefficients and r_n is a series of mutually uncorrelated discretised random inputs, having zero mean and variance

$$\text{Var}(r) = \langle r_n \, r_m \rangle \quad \text{"p" is the filter order.}$$

Pre multiply by z_{n-k}, where $k = 1, 2, 3, \cdots p$, to obtain

$$z_{n-k} \; z_n = \sum_{i=1}^{p} a_i \, z_{n-k} \; z_{n-i} + z_{n-k} \; r_n$$

Taking expected values yields the following relationship

$$C_k = \sum_{i=1}^{p} a_i \; C_{k-i} \qquad (4.2)$$

where Autocovariance, $C_k = \langle z_{n-k} \, z_n \rangle$

and $\langle z_{n-k} \; r_n \rangle = 0$, since z_{n-k}

is only influenced by inputs r_i up until time increment $i = (n-k)$.

The autocorrelations, R_i, can then be obtained by dividing throughout by C_o, so that

$$R_k = \sum_{i=1}^{p} a_i \; R_{k-i} \qquad (4.3)$$

where R_k is the Autocorrelation at lag k and $R_o = 1$.

Expressing this in matrix form, (4.3) becomes

$$\begin{bmatrix} R_1 \\ R_2 \\ \cdot \\ \cdot \\ \cdot \\ R_p \end{bmatrix} = \begin{bmatrix} 1 & R_{-1} & R_{-2} & \cdots & R_{-p+1} \\ R_1 & 1 & R_{-1} & \cdots & R_{-p+2} \\ \cdot & & & & \\ \cdot & & & & \\ \cdot & & & & \\ R_{p-1} & R_{p-2} & R_{p-3} & \cdots & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \cdot \\ \cdot \\ \cdot \\ a_p \end{bmatrix}$$

Since the autocorrelation function is symmetrical about the zero lag point, then

$$R_{-k} = R_k \quad \text{where} \quad k = 1, 2, 3 \ldots p$$

so that the relationship becomes

$$\underset{\sim}{r} = \underline{R} \cdot \underset{\sim}{a} \tag{4.4}$$

where $\underset{\sim}{r}^T = \begin{bmatrix} R_1 & R_2 & \ldots & R_p \end{bmatrix}$

$\underset{\sim}{a}^T = \begin{bmatrix} a_1 & a_2 & \ldots & a_p \end{bmatrix}$

and $$\underline{R} = \begin{bmatrix} 1 & R_1 & R_2 & & R_{p-1} \\ R_1 & 1 & R_1 & & \cdot \\ R_2 & R_1 & 1 & & \cdot \\ \cdot & & & & \cdot \\ \cdot & & & & \cdot \\ \cdot & & & & \cdot \\ R_{p-1} & \cdot & \cdot & \cdot & \cdot \; 1 \end{bmatrix}$$

(4.4) is the Yule-Walker relationship.

The weighting coefficient vector, $\underset{\sim}{a}$, can be easily and uniquely determined from (4.4) as follows:-

$$\underset{\sim}{a} = \underline{R}^{-1} \cdot \underset{\sim}{r} \tag{4.5}$$

Unfortunately the efficiency of this method of coefficient determination decreases as the square of process order, p, due to the necessity of inverting the aquare matrix, $\underline{R}$.

The filter output variance, Var (z), can be determined by taking expected values as in (4.2), but this time with, k = 0, so that

$$C_o = \sum_{i=1}^{p} a_i \, C_{-i} + \mathrm{Var}(r) \qquad (4.6)$$

Since, $\langle z_n \cdot r_n \rangle = \langle r_n \cdot r_n \rangle$

Divide (4.6) by C_o , =Var(z), to obtain

$$1 = \sum_{i=1}^{p} a_i \, R_{-i} + \mathrm{Var}(r)/\mathrm{Var}(z) \qquad (4.7)$$

Recall that $R_{-k} = R_k$ and rearranging (4.7) yields

$$\mathrm{Var}(z) = \frac{\mathrm{Var}(r)}{1 - \sum_{i=1}^{p} a_i \, R_i} \qquad (4.8)$$

4. 2 Least Squares Asymptotic Estimation

When using the Yule-Walker relations to establish the weighting coefficients, "a_i", an inverse square law exists between efficiency of coefficient evaluation and filter order, p. It becomes increasingly desirable with higher order filters to establish a more expedient means of solution.

This can be done by employing a pivotal reduction technique used in regression analysis.

To establish a filter of order, p, it is first necessary to determine the weighting coefficients $a_i^{(i)}$ for i = 1, 2, ---p .
The superscript (i) refers to the coefficients present in the ith order Yule-Walker representation.

Recall the Yule-Walker relation for order p is given by (4.4)
For a first order relation, i = 1, $a_1^{(1)} = R_1$

From the second order case, i = 2, by eliminating $a_1^{(2)}$,

$$a_2^{(2)} = \frac{\text{Det}\begin{vmatrix} 1 & R_1 \\ R_1 & R_2 \end{vmatrix}}{\text{Det}\begin{vmatrix} 1 & R_1 \\ R_1 & 1 \end{vmatrix}}$$

and similarly, for i = 3

$$a_3^{(3)} = \frac{\text{Det}\begin{vmatrix} 1 & R_1 & R_2 \\ R_1 & 1 & R_1 \\ R_2 & R_1 & R_3 \end{vmatrix}}{\text{Det}\begin{vmatrix} 1 & R_1 & R_2 \\ R_1 & 1 & R_1 \\ R_2 & R_1 & 1 \end{vmatrix}}$$

. . . etc. (4.9)

and for i = p

$$a_p^{(p)} = \frac{\text{Det}\begin{vmatrix} 1 & R_1 & \cdots & R_{p-1} \\ R_1 & 1 & \cdots & R_{p-2} \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ R_{p-1} & & & R_p \end{vmatrix}}{\text{Det}\begin{vmatrix} 1 & R_1 & \cdots & R_{p-1} \\ R_1 & 1 & \cdots & R_{p-2} \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ R_{p-1} & & & 1 \end{vmatrix}}$$

Thus, the coefficients $a_1^{(1)}$, $a_2^{(2)}$, $a_3^{(3)}$, . . . $a_p^{(p)}$ are established.

The $a_j^{(i)}$ coefficients, for $j \leq i$, can be determined recursively as follows:-

The second order Yule Walker relation, yields

$$\begin{bmatrix} 1 & R_1 \\ R_1 & 1 \end{bmatrix} \begin{bmatrix} a_1^{(2)} \\ a_2^{(2)} \end{bmatrix} = \begin{bmatrix} R_2 \\ R_1 \end{bmatrix}$$

or in matrix notation

$$\underline{R}_2 \cdot \underset{\sim}{a}_2 = \underset{\sim}{r}_2 \qquad (4.10)$$

Imbedded in the third order relations are the equations

$$a_1^{(3)} + R_1 a_2^{(3)} + R_2 a_3^{(3)} = R_1$$

$$R_1 a_1^{(3)} + a_2^{(3)} + R_1 a_3^{(3)} = R_2$$

which can be rearranged, such that

$$\underline{R}_2 \begin{bmatrix} a_1^{(3)} \\ a_2^{(3)} \end{bmatrix} = \begin{bmatrix} R_1 - R_2 & a_3^{(3)} \\ R_2 - R_1 & a_3^{(3)} \end{bmatrix}$$

ie

$$\begin{bmatrix} a_1^{(3)} \\ a_2^{(3)} \end{bmatrix} = R_2^{-1} \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} - a_3^{(3)} \underline{R}_2^{-1} \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} \qquad (4.11)$$

and from (4.10)

$$\underline{R}_2^{-1} \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} = \begin{bmatrix} a_1^{(2)} \\ a_2^{(2)} \end{bmatrix}$$

substitute in (4.11) to obtain

$$\begin{bmatrix} a_1^{(3)} \\ a_2^{(3)} \end{bmatrix} = \begin{bmatrix} a_1^{(2)} \\ a_2^{(2)} \end{bmatrix} - a_3^{(3)} \begin{bmatrix} a_2^{(2)} \\ a_1^{(2)} \end{bmatrix}$$

. . . etc.

These general recursive relations, due to Durbin (3), are given by

$$\left.\begin{aligned} a_{i+1}^{(i+1)} &= \frac{R_{i+1} - \sum_{j=1}^{i} a_j^{(i)} R_{i+1-j}}{1 - \sum_{j=1}^{i} a_j^{(i)} R_j} \\ a_j^{(i+1)} &= a_j^{(i)} - a_{i+1}^{(i+1)} \cdot a_{i-j+1}^{(i)} \quad \text{for } j = i, i-1, \ldots .1 \end{aligned}\right\} \quad (4.12)$$

For high order AR filters this method of establishing the weighting coefficients is far more efficient.

4. 3 Roots of the Characteristic Equation and the Effect on Performance.

The AR filter can be expressed as

$$\sum_{i=0}^{p} a_i \, z_{n-i} = r_n \qquad (4.13)$$

which has a characteristic equation

$$\sum_{i=0}^{p} a_i \, \lambda^{n-i} = 0 \qquad (4.14)$$

On factorisation this becomes

$$\prod_{i=1}^{p} (1 - g_i^{-1} \lambda) = 0 \qquad (4.15)$$

where g_i, $i = 1, 2 \ldots p$, are the roots of (4.14). Each root can be either real or complex.

The AR filter must satisfy the constraint of stability (see section 3).

The AR filter can be regarded as a digital filter, which has a transfer function, TF(S), in the complex Laplace domain (or S - domain) such that

$$TF(S) = \frac{z_n^{(S)}}{r_n^{(S)}} = \frac{1}{\sum_{i=0}^{p} S_i \exp(g_i h S)} \qquad (4.16)$$

where S_i is a constant.

Thus the filter response, z_n, for $0 \leq n \leq N$, can be regarded as being composed of a sum of damped exponentials (assuming stability is met). If the root g_i is complex, the damped response is oscillatory, and non-oscillatory if g_i real.

Since the input r_n is of a random nature then these exponential decays, between one time increment and the next, must have arbitrary random phase.

As a consequence of the AR filter's recursive nature resulting in damped exponential response characteristics, an efficient order of filter parameter idealisation can be achieved for a comparatively high order process correlation, provided phase is unimportant.

4. 4 Stability

AR filters are by nature iterative and as such are subject to numerical instability problems. (In stochastic time series analysis literature stability is often referred to as stationarity.) The stability criterion namely $|h_i| < 1$ where h_i are the roots of the characteristic equation was established by (3.12).

There is no guaranteed method of establishing weighting coefficients which are stable. The best that can be managed is to test their stability before the filter is implemented

A scheme to check for stability is described in Chapter III - Implementation of a mono-road profile.

How severe a restriction stability may prove to be, will depend on the correlations of interest. However, it was empirically observed in the correlations tested that stability was always achieved (see Chapter IV).

4. 5 Power Spectral Density

The power spectral density can be derived directly from the a_i weighting coefficients in the following manner.

From Appendix B, the power spectral density, S(f), is given by

$$S(f) = 2.\mathrm{Var}(r) \left| b \,.\, \exp(-i 2\pi f) \right|^2 \quad \text{for } 0 \leq f \leq \tfrac{1}{2} \qquad (B.7)$$

and from Appendix A,

$$b(B) = a^{-1}(B) \qquad (A.4)$$

where $a(B) = \sum_{i=0}^{p} a_i B^i$

and B^i is the i^{th} Backward shift operator,

and hence the power spectrum is given by

$$S(f) = \frac{2\,\mathrm{Var}(r)}{\left|\sum_{i=0}^{p} a_i \exp(-i2\pi f)\right|} \qquad 0 \le f \le \tfrac{1}{2} \qquad (4.17)$$

By comparing this spectrum with the original road spectrum, an easy check of the filter model's validity is provided.

5. The Moving Average (MA) Process

5.1 Derivation of the MA Filter Coefficients

The monovariate MA Process takes the form

$$Z_n = \sum_{j=0}^{q} b_j r_{n-j} \qquad (5.1)$$

where the "b_j"s are the weighting coefficients and r_n, for $0 \le n \le N$, is a series of mutually uncorrelated random inputs with zero mean and variance, Var(r). "q" is the filter order.

Pre-multiply by z_{n-k}, for $k = 0, 1, 2, \cdots q$, to obtain

$$z_{n-k} \cdot z_n = \sum_{j=0}^{q} b_j r_{n-k-j} \cdot \sum_{j=0}^{q} b_j r_{n-j}$$

Take expected values, so that

$$C_k = \mathrm{Var}(r) \sum_{j=0}^{q-k} b_j b_{j+k} \qquad (5.2)$$

where autocovariance, $C_k = \langle y_{n-k} y_n \rangle$

Since $r_n(t)$ is mutually uncorrelated, for $0 \le n \le N$,

then $\langle r_{n-k} r_n \rangle = \mathrm{Var}(r)$ for $k = 0$

$= 0$ for $k \neq 0$

To obtain the autocorrelations divide by C_o, thus

$$R_k = \sum_{j=0}^{q-k} b_j \, b_{j+k} \cdot \mathrm{Var}(r) \qquad (5.3)$$

where $R_k = C_k / C_0$ is the autocorrelation of $y(t)$ at lag k.

Express (5.3) in the matrix form

$$\begin{bmatrix} R_0 \\ R_1 \\ R_2 \\ \vdots \\ R_q \end{bmatrix} = \mathrm{Var}(r) \begin{bmatrix} b_0 & b_1 & b_2 & \cdots & b_q \\ 0 & b_0 & b_1 & & b_{q-1} \\ 0 & 0 & b_0 & \cdots & b_{q-2} \\ \vdots & & & & \\ 0 & 0 & \cdots & \cdots & b_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_q \end{bmatrix}$$

or more concisely

$$\underset{\sim}{\bar{r}} = \mathrm{Var}(r) \, \underline{B} \, \underset{\sim}{.b} \qquad (5.4)$$

Unfortunately, these equations possess no unique solution as there are $q!$ cross products "$b_i b_j$", where $j = 0, 1, 2, --, q$ and $i = 0, 1, 2, ---, q-j$, and only q equations to establish their values.

The "b_j" weighting coefficient can in fact be shown to be unique, once the constraint of invertible stationarity is considered. This is proved in subsection 5.4 for the general ARMA model of which the MA model is a subspecies.

To meet the invertible stationarity constraints the "b_j" weighting coefficients must be arrived at by an iterative numerical technique.

The output variance, $\mathrm{Var}(z)$, is given by (5.2) when $k = 0$, so that $\mathrm{Var}(z) = C_o$.

5.2 Efficient Calculation of MA Filter Coefficients

In Appendix A the relationship

$b(B) = a^{-1}(B)$ is established (A.4)

This relationship is used by Durbin (3) to establish the coefficients, "b_i", from a set of established AR filter coefficients, "a_i". The goodness of fit to a fixed order MA process is dependant on the order at which the equivalent infinite AR process is truncated.

This means of numerical estimation ensures that the invertibility constraint is met (see section 3). Unfortunately, it proves to be intractible for high order MA filters.

Another means of estimating the "b_i" coefficients is to use a Newton Raphson iterative technique to make an estimate of the "b_i" coefficients.

For the $(i+1)^{th}$ iteration the relationship is

$$\underset{\sim}{b}^{i+1} = \underset{\sim}{b}^{i} - (\underline{D}^{i})^{-1} \underset{\sim}{f}^{i} \qquad (5.5)$$

where $\underset{\sim}{f}_i = \underline{B}^i \underset{\sim}{b}^i$

$\underline{D} = \overline{\underline{B}} + \underline{B}$

$$\overline{\underline{B}} = \begin{bmatrix} b_0 & b_1 & \cdots & b_{q-2} & b_{q-1} & b_q \\ b_1 & b_2 & & b_{q-1} & b_q & 0 \\ b_2 & b_3 & & b_q & 0 & 0 \\ \vdots & & & & & \\ b_q & 0 & & 0 & 0 & 0 \end{bmatrix}$$

and $\underline{B}$ is as defined in subsection 5.1.

Unfortunately, this method takes no account of the invertible stationarity constraint imposed on the "b_i" coefficients. Once this constraint is imposed this method is also intractible for high order filters.

5.3 Roots of the Characteristic Equation and the Effect on Performance

The MA process takes the form

$$z_n = \sum_{j=0}^{q} b_j r_{n-j} \tag{5.6}$$

This has the characteristic equation

$$\sum_{j=0}^{q} b_j \lambda^{(n-j)} = 0 \tag{5.7}$$

which after factorisation becomes

$$\prod_{j=0}^{q} (1 - g_j^{-1} \lambda) = 0 \tag{5.8}$$

where g_j, $j = 0, 1, 2, .., q$, are the roots of the characteristic equation, and can be either real or complex.

The MA filter is not subject to the stability constraint but instead is subject to the constraint of invertibility (see section 3).

The MA filter can be regarded as a digital filter, possessing the digital transfer function (T.F(S))

$$TF(S) = z_n(S)/r_n(S) = \sum_{j=0}^{q} T_j \exp(g_j hs) \tag{5.9}$$

where T_j is a constant and S refers to the complex Laplace Domain. Thus the filter comprises of a finite series of delay operators applied directly to the input. If the roots, g_j, are complex the response will be oscillatory.

Since, TF(S) is finite and non-recursive the correlation will exhibit a sharp cut-off after lag, $j = q$.

Since, each delay operator deals with only one single random input at any specific time increment, h, excellent phase characteristics will be portrayed by the filter up to time lags of magnitude, qh, at the expense of inefficient fitting for a prescribed correlation order.

5.4 Stability and Invertibility

MA filters are of a non-recursive nature and as such are intrinsically stable. Such filters are however subject to the constraint of invertibility, namely, $|g_j| > 1$ where, g_j, are the roots of the characteristic equation, in order to ensure a sensible filter.

It is demonstrated in section 6.4 that the invertibility constraint is necessary to ensure uniqueness of the filter weighting coefficients, "b_j". Unfortunately, there is no sure way of arriving at the unique, "b_j", coefficients short of trial and error tests using the invertibility constraint. For high order filters this may prove a wearisome occupation.

The invertibility constraint imposed on an MA filter does however score in one respect over the stability constraint of an AR filter, for it does have one unique invertible solution. The AR filter has no such guarantee of stability. In fact in the empirical tests on the AR filters described in the following chapters, instability is never encountered.

5.5 Power Spectral Density

In an analogous manner to the AR process, the MA power spectrum, S(f), is obtained directly from the, "b_j", weighting coefficients.

From Appendix B, the power spectral density is given by

$$S(f) = 2\,\mathrm{Var}(r)\,\left|b\exp(-i2\pi f)\right|^2 \quad \text{for } 0 \le f \le \tfrac{1}{2} \qquad (B.7)$$

$$= 2\,\mathrm{Var}(r)\,\left|\sum_{j=0}^{q} b_j \exp(-i2\pi f)\right|^2 \qquad (B.10)$$

Once again an easy check of the filter model's validity is provided on comparison with the original spectrum.

6. The Autoregressive Moving Average (ARMA) Process

6.1 General

As mentioned in the introduction, the ARMA model takes the form

$$1/a_o \sum_{i=o}^{p} a_i z_{n-i} = 1/a_o \sum_{j=o}^{q} b_i r_{n-j} \qquad (2.1)$$

This is the general form of which the AR and MA models are special cases:

AR if $p > 0$ and $q = 0$, MA if $p = 0$ and $q > 0$

Most of the discussion on the AR and MA filters, with the exception of the weighting coefficient determination (done in subsection 6.2), applies to the mixed ARMA model.

In Appendix A the relationship between the MA and AR processes is derived. It demonstrates that a finite MA filter can be equivalenced by an infinite AR filter and vice versa. However, a process which is essentially AR in nature cannot be efficiently modelled by an MA filter. The same is also true for an AR model of an MA process. A logical extension of this efficient modelling criterion comes when the process to be modelled contains properties of both types. Under such conditions it is essential to incorporate both AR and MA characteristics into the model, resulting in an ARMA filter.

6. 2 Efficient Calculation of ARMA filter coefficients.

Determination of the AR filter coefficients can quite simply be done either by the Yule-Walker relations or by the least square approximation technique. Determination of the MA terms is less straight forward because of the non-linear nature of the filter and the necessity to consider invertibility to establish uniqueness. It can reasonably be expected, therefore, that coefficient determination of the mixed ARMA model is yet more complex. This is indeed the case as the following description will show

The ARMA filter takes the form

$$z_n = \sum_{i=1}^{p} a_i z_{n-i} + \sum_{j=0}^{q} b_j r_{n-j} \qquad (6.1)$$

In an analagous manner to that used in the derivation of the Yule-Walker relations, pre-multiply throughout by, z_{n-k}, to obtain

$$C_k = \sum_{i=1}^{p} a_i C_{k-i} + \sum_{j=0}^{q} b_j \langle z_{n-k} \; r_{n-j} \rangle \qquad (6.2)$$

Since, as before, r_n, is mutually uncorrelated, n = 0, 1, 2, ..,N, and since, z_{n-k}, is only influenced by random inputs prior to time instant, n-k, it follows that

$$\langle z_{n-k} \; r_{n-j} \rangle = 0, \text{ for } k-j > 0$$

$$\neq 0, \text{ for } k-j \leq 0$$

and hence (6.2) reduces to a Yule-Walker type relation, for $(k-j) > 0$,

$$\left.\begin{aligned} R_k &= \sum_{i=1}^{p} a_i R_{k-i} \\ \text{which in matrix notation is} & \\ \underset{\sim}{r} &= \underline{R} \; \underset{\sim}{a} \end{aligned}\right\} \qquad (6.3)$$

Since the restriction, $(k-j) > 0$, is imposed, this implies that, $k > q$, and since, $k = 0, 1, 2, \ldots, p$, it follows that, $p > q$. Thus the ARMA filter, (6.1), considered in the following is subject to the constraint, $p > q$.

The ARMA filter of (6.1) can be written in the form

$$\sum_{j=0}^{q} b_j \, r_{n-j} = \sum_{i=0}^{p} a_i \, z_{n-i} \qquad (6.4)$$

$$\text{where } a_0 = -1$$

By considering an MA filter of the form

$$\overline{P}_n = \sum_{i=0}^{p} a_i \, z_{n-i} \qquad (6.5)$$

where, $\overline{P}_n$, is the output variable and where the "a_i" coefficients have already been determined from (6.3). The influence these "a_i" coefficients have on the total ARMA filter can be determined.

This is achieved in an analogous manner to that used in establishing the MA relationship:-

Premultiply, (6.5) by, $\overline{P}_{n-k}$, so that

$$\overline{P}_{n-k}\ \overline{P}_n = \sum_{i=0}^{p} a_i\ z_{n-k-i} \cdot \sum_{i=0}^{p} a_i\ z_{n-i}$$

Take expected values, to obtain

$$\left.\begin{aligned} R_k(p) &= \sum_{i=0}^{p} (a_i\ a_{i+k})\ \mathrm{Var}(z) \\ \text{where}\quad R_k(p) &= \langle \overline{P}_{n-k}\ \overline{P}_n \rangle \\ \text{or in matrix form} \\ \underset{\sim}{r}(p) &= \mathrm{Var}(z)\ \underline{A}\ \underset{\sim}{a} \end{aligned}\right\} \qquad (6.6)$$

$$\text{where}\quad \underline{A} = \begin{bmatrix} a_0 & a_1 & \cdots & a_p \\ 0 & a_0 & \cdots & a_{p-1} \\ 0 & 0 & \cdots & a_{p-2} \\ \vdots & & & \\ 0 & 0 & & a_0 \end{bmatrix}$$

$$\underset{\sim}{\bar{a}}^{T} = \begin{bmatrix} a_0 & a_1 & \cdots & a_p \end{bmatrix}$$

$$\text{and}\quad \underset{\sim}{r}(p)^{T} = \begin{bmatrix} R(p)_0 & R(p)_1 & \cdots & R(p)_p \end{bmatrix}$$

As in the determination of the pure AR filter coefficients, the method described above for the ARMA filter does not guarantee that the constraint of invertibility of the, "b_j", weighting coefficients will be met. Consequently, a trial and error check must be implemented. Again, for high order filters this proves intractable. The effect of this pure AR process can then be removed by subtracting the, $\underset{\sim}{r}(p)$, vector from the total correlation.

As with the simple MA filter, the Newton Raphson iterative technique can then be employed to determine the, "b_j", weighting coefficients of this "residual" MA process. For the, '$(i)^{th}$', iteration this would lead to the relations

$$\underset{\sim}{b}^{(i)} = \underset{\sim}{b}^{(i-1)} - \underline{D}^{(i-1)} \; \underset{\sim}{f}^{(i-1)} \tag{6.7}$$

where $\underset{\sim}{f}^{(i)} = \underline{B}^{(i)} \; \underset{\sim}{b}^{(i)} - \underset{\sim}{r}(p)$

$$\underline{D}^{(i)} = \overline{\underline{B}}^{(i)} + \underline{B}^{(i)},$$

$$\overline{\underline{B}}^{(i)} = \begin{bmatrix} b_0^{(i)} & b_1^{(i)} & \cdots & b_q^{(i)} \\ b_1^{(i)} & b_2^{(i)} & \cdots & 0 \\ \vdots & & & \\ b_q^{(i)} & 0 & & 0 \end{bmatrix},$$

$$\underline{B}^{(i)} = \begin{bmatrix} b_0^{(i)} & b_1^{(i)} & \cdots & b_q^{(i)} \\ 0 & b_0^{(i)} & & b_{q-1}^{(i)} \\ \vdots & & & \\ 0 & 0 & & b_0^{(i)} \end{bmatrix}.$$

6.3 Roots of the Characteristic Equation and the effects on performance.

The ARMA filter is the general model of which the AR and MA processes are special cases.

From (2.1) the ARMA filter takes the form

$$\sum_{i=0}^{p} a_i \, z_{n-i} = \sum_{j=0}^{q} b_j \, r_{n-j} \qquad (6.8)$$

assuming $a_0 = 1$. (Remember a_0 is simply a scaling factor)

(6.8) has the characteristic equations

$$\left.\begin{aligned} \sum_{i=0}^{p} a_i \lambda^{n-i} &= 0 \\ \text{and} \quad \sum_{j=0}^{q} b_j \lambda^{n-j} &= 0 \end{aligned}\right\} \qquad (6.9)$$

which on factorisation become respectively

$$\left.\begin{aligned} \prod_{i=1}^{p} (1-g_i^{-1}\lambda_i) &= 0 \\ \text{and} \quad \prod_{j=1}^{q} (1-\bar{g}_j^{-1}\lambda_j) &= 0 \end{aligned}\right\} \qquad (6.10)$$

where g_i, $i = 0, 1, \ldots, p$, and $\bar{g}_j$, $j = 0, 1, 2 \ldots, q$, are the respective roots.

The stability and invertibility constraints both apply to this mixed filter.

The ARMA filter can be regarded as the digital filter which in the complex, S - domain has the transfer function TF(S) such that

$$TF(S) = \frac{z_n(S)}{r_n(S)} = \frac{\sum_{j=0}^{q} T_j \exp(\bar{g}_j h S)}{\sum_{i=0}^{p} S_i \exp(g_i h S)} \qquad (6.11)$$

where T_j and S_i are constants.

The ARMA filter incorporates the damped exponential response characteristics of the AR process together with the good phase characteristics, up to time lags qh, of the MA process (sections 4.3 and 5.3). To some extent the mixing of both types of characteristics "muffles" the virtues of the simple AR or MA filter. In any event, good phase lag characteristics are superfluous to the requirements of a road profile generator.

6.4 Stability and Invertibility

Since the ARMA model incorporates a recursive AR model with a non-recursive MA model, it is subject to the constraints of stability and invertibility. These concepts are developed in Section 3 and their relevance to AR and MA filters are discussed in Section 4.4 and 5.4 respectively.

To establish the unique solution of the MA part requires the same tedisome trial and error approach to establish the invertible, "b_j", parameters. As with the AR process there is no guarantee that stability will be achieved.

6.5 Power Spectral Density

In a similar manner to that implemented in Sections 4.5 and 5.5 for the AR and MA filters, the power spectral density is easily seen to be given by

$$S(f) = 2\,\mathrm{Var}(r)\,\frac{\left|\sum_{i=0}^{q} b_i \exp(-i2\pi f)\right|^2}{\left|\sum_{j=0}^{p} a_j \exp(-i2\pi f)\right|^2} \qquad (6.12)$$

for $0 \le f \le \frac{1}{2}$

6.6 Uniqueness of the general ARMA model

The uniqueness of the general ARMA model can be demonstrated in the following manner.

Consider the ARMA filter

$$\sum_{i=0}^{p} a_i\, z_{n-i} = \sum_{j=0}^{q} b_j\, r_{n-j} \qquad (6.13)$$

which can be expressed in the form

$$\prod_{i=0}^{p}(1-g_i B_w)z(t) = \prod_{j=0}^{q}(1-\bar{g}_j B_w)r(t) \qquad (6.14)$$

where, g_i and $\bar{g}_j$, for $i = 0, 1, .., p$ and $j = 0, 1, .., q$, are the roots of the characteristic equations of the left hand and right hand sides of (6.13) respectively.

From Appendix B, the covariance generating function is given by

$$C(B_w) = \mathrm{Var}(r)\; b(B_w)\; b(F_w) \qquad (B.2)$$

where $b(B_w) = \sum_{j=0}^{q} b_j\, B_w^{\,j}$ and $b(F_w) = \sum_{j=0}^{q} b_j\, F_w^{\,j}$

and $B_w^{\,j}$ is the j^{th} backward shift operator

and $F_w^{\,j}$ is the j^{th} forward shift operator.

However, in the case of (6.14), $b(B_w)$ is the series given by

$$b(B_w) = \prod_{i=0}^{p} (1-g_i B_w)^{-1} \prod_{j=0}^{q} (1-\bar{g}_j B_w) \qquad (6.15)$$

and $b(F_w)$ has a similar form.

Substitute in (B.6) to obtain

$$C(B_w) = \mathrm{Var}(r) \prod_{i=0}^{p} \{(1-g_i B_w)^{-1} (1-g_i . F_w)^{-1}\} \times \prod_{j=0}^{q} \{(1-\bar{g}_j B_w) (1-\bar{g}_j F_w)\} \qquad (6.16)$$

Consider the root pair product

$$(1-\bar{g}_j B_w) (1-g_j F_w) = 1-\bar{g}_j B_w-\bar{g}_j F_w+\bar{g}_j^2 B_w F_w$$

$$= 1-\bar{g}_j B_w-\bar{g}_j F_w+\bar{g}_j^2 \qquad (6.17)$$

since $F_w = B_w^{-1}$

Similarly, the product

$$\bar{g}_j^2(1-\bar{g}_j^{-1} B_w) (1-\bar{g}_j^{-1} F_w) = \bar{g}_j^2 -\bar{g}_j B_w-\bar{g}_j F_w+ B F_w$$

$$= \bar{g}_j^2 -\bar{g}_j B_w-\bar{g}_j F_w+ 1$$

and hence

$$(1-\bar{g}_j B_w) (1-\bar{g}_j F_w) = \bar{g}_j^2(1-\bar{g}_j^{-1} B_w) (1-\bar{g}_j^{-1} F_w) \qquad (6.18)$$

From (6.16) and (6.18) it follows that (6.15) can be rewritten in the form

$$\prod_{i=0}^{p} (1-g_i B_w)z(t) = P \prod_{j=0}^{q}(1-\bar{g}_j^{\pm 1} B_w)r(t) \qquad (6.19)$$

where $P = \prod_j \bar{g}_j^2$ for any combination of products of j for $0 \leq j \leq q$.

Thus, there are a multiplicity of ARMA filters (j! combinations) which satisfy the covariance generating function.

It follows that if any complex root, $\bar{g}_j$, lies outside the unit circle, then, $\bar{g}_j^{-1}$, lies within, and hence only one ARMA filter satisfies the criterion of invertibility.

7. Conclusion

From theoretical considerations of the behaviour of the various types of linear stochastic difference equation, it becomes apparent that the AR filter best meets the design requirements of generating a monovariate road profile, with the added bonus of easy, "a_i", weighting coefficient determination. Theoretically, the most significant shortcoming was the inability to guarantee stability. However, empirically, as is shown in Chapter IV, this proves to be of little hindrance, as no unstable filters were uncovered using the methods outlined in Section 4.

To qualify this conclusion, it is necessary to mention both the coefficient determination characteristics and the performance characteristics of each of the linear stochastic difference equations in turn and discuss there relevance in meeting the design requirements of a mono road profile. This is done in the following paragraphs.

Filter coefficient determination characteristics

With AR filters irrespective of order, the weighting coefficients are readily determined from the Yule-Walker relations. Unfortunately, the resultant filters are not guaranteed stable. Despite this all filters so determined were observed to never violate the stability constraint. (See Chapter IV).

There is a multiplicity of MA filter coefficients which fit the prescribed correlation. However, only one unique set is meaningful and satisfies the invertibility constraint. The determination of

this set becomes progressively more intractable as the order of the filter increases, and high order MA filters are definitely not recommended for this reason.

The determination of the ARMA filter coefficients requires the use of both the AR and MA determination techniques with the drawbacks of both.

All filters, irrespective of type, are easily validated by evaluating their Spectra.

Filter Performance Characteristics

AR filters, by virtue of their recursive nature, give a parsimonious filter fit to a high order correlation, since the filter has an envelope response comprising of damped exponentials, some or all of which may be oscillatory. This is achieved at the expense of poor phase lag characteristics.

Non-recursive MA filters realise good phase relations with a sharp cut off. The non-recursive nature also results in non-parsimonious filter fitting for a prescribed correlation. (If the required filter order is too great then determination of the filter coefficients proves intractable.)

Combining both characteristics in an ARMA model does to some extent muffle the desirable properties of the simple MA and AR filters. However, an ARMA model will yield a parsimonious filter with reasonable phase characteristics. As with the simple MA process, if the order of the MA part of the filter proves to be too high, then coefficient determination will prove intractable. Since no guarantee of a suitably low order can be given in advance, then attempts at using this method should be approached with caution.

The phase aspect of the MA part of the ARMA filter becomes important in cases where there is a requirement for prescribed cross-correlations between different AR filters generating in parallel. Under such circumstances multivariate rather than monovariate LSDEs must be considered.

In multivariate processes, ARMA filters must be considered as they are the only type which maintain the necessary phase characteristics between parallel generated processes together with parsimonious generation of the autocorrelations of individual processes. Clearly, as a result of the difficulties of determining the coefficients of ARMA filters, they are best avoided unless extensions into multivariate processes are contemplated.

No mention is made of the probability distribution of the filter output. It is assumed Gaussian. These properties are mainly dependant on the probability distribution of the filter input. The generation of such Gaussian inputs, together with the pseudo-periodic properties exhibited, are discussed in Chapter III.

REFERENCES - CHAPTER II

Ref. 1 Box, G.E.P. and Jenkins, G.M. (1970 or rev. ed 1976) "Time series analysis: forecasting and control" pub. Wiley.

Ref. 2 Anderson, T.W. (1971) "The statistical analysis of time series" pub. Wiley.

Ref. 3 Durbin, J. (1960) "The fitting of time series models" Revue Inst. De Stat. 28, 233-243.

APPENDIX A

Equivalence of the coefficients of AR and MA filters

Consider the MA filter

$$z_n = \sum_{j=0}^{q} b_j \, r_{n-j} \qquad (A.1)$$

where, as usual, 'b_j' are weighting coefficients of the mutually uncorrelated inputs, r_n of variance Var (r).

Define the backward shift operator, $B_w^{\,k}$ on, z_n as the, k^{th} back value, z_{n-k}, such that

$$B_w^{\,k} z_n = z_{n-k} \qquad (A.2)$$

and the backward shift series as

$$b(B_w)_q = \sum_{j=0}^{q} b_j \, B_w^{\,j} \qquad (A.3)$$

Thus in abbreviated form (A1) is expressed as

$$z_n = b(B_w)_q \, r_n \qquad (A.4)$$

In a similar manner, the abbreviated form of the AR filter

$$\sum_{j=0}^{q} a_j \, z_{n-j} = r_n \qquad (A.5)$$

is $a(B_w)_q \, z_n = r_n \qquad (A.6)$

where $a(B_w)_q = \sum_{j=0}^{q} a_j \, B_w^{\,j} \qquad (A.7)$

Multiply (A6) by $b(B_w)_q$ to obtain

$$b(B_w)_q \; a(B_w)_q \; z_n = b(B_w)_q \, r_n$$

Substitute (A4) in right hand side of above equation so that

$$b(B_w)_q \; a(B_w)q \; z_n = z_n$$

and hence, $b(B_w)_q = \left(a(B_w)_q\right)^{-1} \qquad (A.8)$

On expansion the right hand side of (A8) yields an infinite series. (A8) demonstrates that any finite MA filter can be replaced by an infinite AR filter. The converse is also true.

APPENDIX B

Determination of the Spectrum of a digital filter

Consider the infinite MA process

$$z_n = \sum_{j=0}^{\infty} b_j \, r_{n-j} \qquad (B.1)$$

with notation as before.

In a similar manner to (5.2), the output covariances, C_k, for all lags $k = 0, 1, \ldots, \infty$ are given by

$$C_k = \mathrm{Var}(r) \sum_{j=0}^{\infty} b_j \, b_{j+k} \qquad (5.2)$$

For convenience, define the autocovariance generating function, $C(B_w)$, (no suffix for ∞ series) as

$$C(B_w) = \sum_{k=-\infty}^{\infty} C_k \, B_w^{\,k} \qquad (B.2)$$

Substitute for, C_k using (5.2), to give

$$C(B_w) = \mathrm{Var}(r) \sum_{k=-\infty}^{\infty} \sum_{j=0}^{\infty} b_j \, b_{j+k} \, B_w^{\,k}$$

$$= \mathrm{Var}(r) \sum_{j=0}^{\infty} \sum_{k=-j}^{\infty} b_j \, b_{j+k} \, B_w^{\,k}$$

since, $b_j = 0 \quad \text{for} \quad j < 0$

Make the substitution $l = j+k$, so that

$$C(B_w) = \mathrm{Var}(r) \sum_{j=0}^{\infty} \sum_{l=0}^{\infty} b_j \, b_l \, B_w^{\,(l-j)}$$

$$= \mathrm{Var}(r) \sum_{l=0}^{\infty} b_l \, B_w^{\,l} \sum_{j=0}^{\infty} b_j \, B_w^{\,-j}$$

$$= \mathrm{Var}(r) \; b(B_w) \; b(B_w)^{-1} \qquad (B.3)$$

In an analagous manner to (A.2) and (A.3) define the "k^{th}" forward shift operator "F_w^{k}", of the series $z_{n-k}(t)$, to be

$$F_w^{k} z_n(t) = z_{n+k}(t) = B_w^{-k} z_n(t) \qquad (B.4)$$

and the forward shift series "$b(F_w)_q$" on weighting coefficients "b_j" to be defined by

$$b(F_w)_q = \sum_{j=0}^{q} b_j F_w^{j} \qquad (B.5)$$

(no suffices for an infinite series).

Hence, (B.3) becomes

$$C(Bw) = Var(r)\ b(B_w)\ b(F_w) \qquad (B.6)$$

and the Power Spectrum, S(f) is easily obtained by assigning

$$(B_w) = \exp(-i 2\pi f)$$

and $$(F_w) = (B_w)^{-1} = \exp(i 2\pi f)$$

in (B.6) to obtain

$$S(f) = 2\ Var(r)\ b \exp(-i 2\pi f)\ b \exp(i 2\pi f)$$

$$= 2\ Var(r) \left| b \exp(-i 2\pi f) \right|^2 \qquad (B.7)$$

for $0 \le f \le \frac{1}{2}$

CHAPTER III

IMPLEMENTATION OF THE ROAD PROFILE GENERATOR

1. Introduction

It is clearly undesirable to store large quantities of random road surface data to obtain a sample set of independant road profiles with the correct statistical characteristics. An alternative approach is to use a road profile generator to produce an infinite number of independant surface realisations, each with the correct statistical characteristics. The only prescribed information the generator requires is the road profile's displacement autocorrelation. By far the most complex part of road profile generator design concerns the use of AR filters to enforce the correct correlation characteristics. (The theory was considered in Chapter II). However, the generation of independant Gaussian random realisations is now discussed. The various constituent parts of the road profile generator are then brought together and the practical implementation of the road profile generator considered.

2. An overview of the road profile generator

A brief functional description of the components in the road profile generator is now given for the purpose of clarification.

Uniformly distributed random data is generated by means of a pseudo random number generator (RNG). Although RNGs exhibit periodic characteristics, the period for repetition is normally extremely long and, since there is an extremely large range of RNG multiplier coefficients to choose from, an almost infinite supply of independantly generated realisation sources is available for selection.

As road profile realisations normally have a Gaussian distribution, the uniformly distributed output from the RNG is passed through a Gaussian filter. The Gaussian filter outputs uncorrelated random data with a Gaussian probability distribution.

The correct displacement autocorrelation is achieved by means of an autoregressive (AR) filter. The AR filter processes the Gaussian random data such that the correct displacement autocorrelation is obtained. It was qualitatively anticipated that the AR filter would have little influence on the probability distribution. The empirical evidence of Chapter IV confirms this presumption.

As the vehicle model requires both displacement and velocity excitation input, some means of generating the velocity road profile was also required. Realisations of the velocity road profile were obtained by numerically differentiating the displacement realisations. A fifth order central difference formula was selected and used in the velocity filter for this purpose. To ensure both displacement and velocity generations are output in phase, a simple delay routine (Hold filter) was placed in the path of the displacement generation.

A schematic representation of the displacement road profile generator is shown in figure 3.1.

3. The Pseudo random number generator (RNG)

A very large number of pseudo random number generators are available for selection. Virtually all are acceptable and produce uniformly distributed random data. The description of the simple RNG algorithm adopted for this project follows.

A simple integer seed value, S_o, is accepted by the RNG and multiplied by an integer, M, held constant for the duration of the realisation record. It is essential that, M, has the properties of a primitive root - otherwise maximum sequence length is not obtained.

Only the, $2^{N/2}$ least significant digits of the resultant integer product are retained, where, 2^N is the largest binary digit the computer can handle. The retained quantity, S_N can be more succinctly defined as follows,

$$S_N = SM - (SM \,\%\, W) \,.\, W \qquad (3.1)$$

where $SM = S_O \,.\, M$

$$W = 2^{N/2} + 1$$

(The symbol % represents an integer divide operation)

To obtain a newly generated random number, S_N substitute the old value of, S_N for S_O and repeat the calculation of (3.1).

The algorithm (3.1) produces random data uniformly distributed within the integer range, 0 to W.

A normalised real output, r_{RNG}, with zero mean value is obtained in the following manner,

$$r_{RNG} = 2(S_{N/W} - 0.5) \qquad (3.2)$$

The pseudo periodic properties exhibited by the RNG cause the generated random data to be repeated after every $2^{N/2}$ generated data points. Danger of repetition is not a problem as the realisation records are normally much shorter than $2^{N/2}$ and a very large supply of alternative primitive root multipliers, M, exists.

Several independant, r_{RNG}, realisations are generated in parallel, to be synthesised when input to the Gaussian filter.

4. The Gaussian Filter

To process the uniformly distributed random data, r_{RNG}, to obtain uncorrelated random output with a Gaussian distribution, a Gaussian digital filter was introduced. Three alternative filters were considered; a Central Limit method, Teichroew's method, and the Polar method.

The central limit method directly embodied the concept of the Central Limit theorem. A total of twelve independant, r_{RNG}, realisations were summed. The resultant probability distribution proved so rich in extreme values as a consequence of truncating after twelve components that it was clearly unworkable. A much larger number of independant, r_{RNG}, records must be summed. This is clearly impractical and this method was considered no further.

Teichroew's method (ref. 1) is a development of the basic Central Limit theorem. Again twelve independant, r_{RNG}, were summed

$$r_S = \left(\sum_{i=1}^{12} r_{RNG}(i) - 6\right)/4 \qquad (4.1)$$

This quantity, r_S, is filtered to produce the Gaussian output, as follows

$$r_G = ((((C5 \cdot r_S + C4) r_S^2 + C3) r_S^2 + C2) r_S^2 + C1) r_S \qquad (4.2)$$

where $C1 = 3.949846138$,

$C2 = 0.252408784$,

$C3 = 0.076542912$,

$C4 = 0.008355968$,

$C5 = 0.029899776$.

The Polar method (ref. 2) calculates the product of a Rayliegh distributed random variable, $r_{RNG}(1)$, and the cosine Polar coordinate of two independant uniformly distributed random variables, $r_{RNG}(1)$ and, $r_{RNG}(2)$. The digital approximation to the Rayliegh distributed variable, r_R, is obtained by the formula

$$r_R = \sqrt{-2 \ln (r_{RNG}(1))} \qquad (4.3)$$

where $r_{RNG}(1)$ is a uniformly distributed random variable in the range $-1 \leq r_{RNG}(1) \leq 1$.

(The cumulative density function of a Rayliegh probability distribution takes the form of a truncated log series.)

The cosine Polar random variable, r_P, is obtained from

$$r_P = \frac{r_{RNG}^2(1) - r_{RNG}^2(2)}{r_{RNG}^2(1) + r_{RNG}^2(2)} \qquad (4.4)$$

(4.4) is obtained from the trigonometric relation

$$\cos 2\theta = \frac{A^2 - B^2}{A^2 + B^2}$$

Should the denominator of (4.4) lie outside the unit circle, the attempted evaluation of, r_P, must be discarded and a new value sought by iteration. A certain degree of redundancy exists in this method. However, once a successful sum-of-square denominator of (4.4) is found, a Gaussian distributed random variable is obtained from the relation

$$r_G = r_R \, r_P \qquad (4.5)$$

Both Teichroew and the Polar methods have a standard deviation of unity.

The Teichrow and Polar methods were compared, with their relative numerical complexity being considered first, followed by their

ability to generate Gaussian data.

A comparison Table 3.1 of the number of arithmetic operations required to generate Gaussian random data by both methods was drawn up. The Polar method appears at first computationally less tedious (by as much as a factor of ten) ignoring the calls to two computer library functions, SQRT( ) and LOG( ). This conclusion is the result of ignoring the inherent redundancy present in the Polar method. These results are shown as non bracketed in Table 3.1.

To estimate the degree of redundancy present in the Polar method, six independant tests of 1000 realised points were computed. The results are shown in Table 3.2. The average degree of redundancy, normalised with respect to unity, was observed to be 0.272 (25% approximately). As each redundant loop requires two additional RNG calls, the calculation of the sum of squares denominator of (4.4), and one further arithmetic IF test. Thus a 25% additional weighting must be placed on these redundant arithmetic operations in the Polar method. The bracketed values in Table 3.1 show the net effect of redundancy.

Since, SQRT( ), LOG( ), and arithmetic IF are not present in the Teichroew method, this comparison is not wholly valid and a computer implemented comparison is necessary. On an IBM 370/158 computer the c.p.u. time ratio of the Polar and Teichroew method was approximately 1:2·2 in favour of the Polar technique.

Although this evidence was insufficient to permit a firm conclusion to be drawn, it does give justification regarding numerical efficiency for implementing the Polar method. When it is borne in mind that the Gaussian filter is implemented at every generated point, this consideration is very significant.

An empirical comparison of the resultant probability distribution for both Gaussian filter methods was also undertaken. The comparisons were drawn from six independant tests of 1001 points with both methods. The cumulative density functions (CDF) were compared with the theoretical Gaussian distribution. In the tests each realised value was categorised into any one of eight quantile segments.

The quantities were arranged to be symmetrically disposed about the maximum probability value of the Gaussian distribution (in this case the mean value is zero). Each quantile was designed to accommodate realised values within a range width of one standard deviation. Any realised value of r_G, can thus be accounted for.

Some empirical performance tests were conducted. Normalised histograms for the average of six tests of Teichroew and Polar methods are shown plotted against the Gaussian distribution in fig. 3.2 and 3.3, respectively. The CDF of the test results is only accurately known at the end of each quantile segment. Both methods perform well.

Both the Teichroew and Polar methods adequately generate random data which conforms to a Gaussian distribution. The Polar method was chosen on the grounds of its superior numerical efficiency. (A more detailed study of the probability distribution is conducted in Chapter IV when the entire road profile generator is tested.)

5. The Autoregressive (AR) Filter

The AR filter segment is by far the most complex part of the road profile generator. The complexities arise from the amount of work necessary in ascertaining the filter coefficients and from the amount of prior checking required to ensure stable and physically sensible displacement road profile outputs.

Use of the AR filter is straightforward. The form of the monovariate AR filter was described in Chapter II - (4.1). With uncorrelated Gaussian random input, r_G, the equation becomes,

$$z_n = \sum_{i=1}^{p} a_i \, z_{n-i} + r_{G,n} \qquad (5.1)$$

where z_n = is the displacement road value at time increment, t_n,

$r_{G,n}$ = uncorrelated Gaussian random input value at time increment, t_n,

a_i = weighting coefficients of the AR filter,

p = AR filter order.

In use the AR filter is a simple digital filter.

Determination of the weighting coefficients, 'a_i', to meet the needs of a prescribed road profile displacement autocorrelation is much more involved. The process for determining the 'a_i' coefficients is shown schematically in fig. 3.4. The description follows.

Given a prescribed displacement autocorrelation curve, it is first necessary to verify its physical realisability. As the generated road profiles adhere to this correlation curve and as the profiles are used as physical inputs to an elastic system (the vehicle model), it is apparent that these input displacements govern the total elastic strain energy stored in the model. The elastic strain energy (E) must always be greater or equal to zero in order to be physically meaningful. In matrix notation this inequality can be expressed as,

$$\underset{\sim}{E} = \tfrac{1}{2}\ \underset{\sim}{Z}^T\ \underline{A}\ \underset{\sim}{Z} \geq \underset{\sim}{0} \qquad (5.2)$$

where $\underset{\sim}{Z}$ = input displacement vector,

$\underline{A}$ = elastic stiffness matrix,

$\underset{\sim}{0}$ = null vector,

and $\underset{\sim}{E}$ = elastic energy vector.

(5.2) is the definition of positive definite quadratic form, and $\underline{A}$ is said to be positive definite provided it is symmetric. A necessary and sufficient test for positive definiteness is to check that all main diagonal sub-determinants in $\underline{A}$ are greater than zero, ie. check if

$$\text{Det } A_i = \begin{vmatrix} a_{11} & a_{12} & \cdots \\ a_{21} & a_{22} & \cdots \\ \cdot & & \\ \cdots\cdots & & a_{ii} \end{vmatrix} > 0 \quad \text{for all i.}$$

The Yule-Walker correlation matrix $\underline{R}$ of Chapter II - (4.4) has the displacements expressed in this quadratic form. Consequently, all that need be done to ensure physically realisable correlation curves, is apply the positive definite determinant test, described for the matrix, $\underline{A}$, to the displacement correlation matrix, $\underline{R}$.

Having established the autocorrelation as positive definite, work can commence in ascertaining the AR filter weighting coefficients, 'a_i'. In Chapter II, two methods of establishing the 'a_i' coefficients were outlined (section 4.1 and 4.2). As determination of the filter coefficients need only be accomplished once per road profile correlation, the amount of numerical effort required to do this is not a prime consideration. The direct evaluation of the Yule-Walker relations was therefore undertaken purely for ease of implementation of the computer program.

No guarantee can be given that the resultant AR filter will be stable. It is therefore necessary to empirically test the numerical stability. As explained in Chapter II - section 3, instability can be determined by checking if the roots of the characteristic equation have any moduli greater than unity. If the order, p of the characteristic equation is high, the direct determination of the roots of this polynomial is prone to numerical ill conditioning. To overcome this problem, the characteristic equation is forced to undergo the bi-linear transformation, $\lambda = \frac{W+1}{W-1}$, mapping from the complex λ -plane onto the complex W-plane.

The 'a_i' coefficients are transformed into 'm_j' coefficients by the relation

$$m_j = \sum_{k=0}^{n} a_k \sum_{e=0}^{j} \left\{ \binom{k}{e} \binom{n-k}{j-e} (-1)^e \right\} \qquad (5.3)$$

where $\binom{k}{e} = \frac{k!}{e!\,(k-e)!}$,

while the transformed characteristic equation is given by

$$m_0 W^n + m_1 W^{n-1} + \ldots\ldots + m_p = 0 \qquad (5.4)$$

(5.3) lends itself well to recursive computation. (See Jury Ref. 3).

The test for instability in the AR filter is now simply a matter of checking if any roots of (5.4) are greater than zero. The Routh stability criterion can be used for this purpose. The Routh

technique is not prone to ill-conditioning. This stability test can be found in any standard text book on Control Theory and will not therefore be described.

However, should the Routh test indicate instability, the complexity of the AR filter determination together with the indirect nature of the stability test makes it impossible to decide how to alter the filter coefficients to ensure stability while approximately adhering to the original prescribed correlation curve.

In an attempt to alleviate this instability situation an empirical relooping procedure was implemented in the determination process. (See fig. 3.4). This relooping procedure was designed to reduce the order of the prescribed correlation curve by one and new AR filter coefficients determined by means of the Yule-Walker relation. This relooping procedure can be repeated as many times as there are prescribed correlation points, however, care should be taken to ensure the prescribed correlation curve is still being generated by the reduced filter.

In all the AR filters determined from physically realisable (positive definite) road profile correlations, no instability was ever encountered. It may be that the Routh test is redundant, however, such a possibility is left for others to consider.

6. Velocity (or displacement gradient) Profile Filter

So far the description has only included the design of the displacement profile generator, however, the second order DE of the vehicle model also requires a velocity excitation input. If the displacement road profile autocorrelation is defined, then the derivative correlations of displacement - velocity, velocity - displacement, and velocity are implicitly defined.

The displacement autocorrelation

$$R_z(L) = \langle z(x)\, z(x + L) \rangle \qquad (6.1)$$

$$\text{or} = \langle z(x - L)\, z(x) \rangle$$

Where $\langle\ \rangle$ denotes the approximate expected value, can be used to determine the displacement - velocity correlation, $R_{z\dot{z}}(L)$, by differentiating with respect to, L, as follows,

$$\frac{d}{dL} R_z(L) = \lim_{\delta L \to 0} \left(\langle z(x)\, z(x + L + \delta L) \rangle - \langle z(x)\, z(x + L) \rangle \right)$$

$$= \lim_{\delta L \to 0} \left\langle z(x) \frac{z(x + L + \delta L) - z(x + L)}{\delta L} \right\rangle$$

$$= \left\langle z(x) \frac{d}{dL} z(x + L) \right\rangle$$

$$= R_{z\dot{z}}(L) \qquad (6.2)$$

In a similar manner the other derivative correlations can also be established from the displacement autocorrelation, to yield the the following relations,

$$\frac{d}{dL}\left(R_z(L)\right) = -R_{\dot{z}z}(L) = R_{z\dot{z}}(L) \qquad (6.3)$$

$$\frac{d^2}{dL^2}\left(R_z(L)\right) = -\langle \ddot{z}(x)\, \dot{z}(x + L) \rangle$$

$$= -R_{\dot{z}}(L) \qquad (6.4)$$

These cross correlations which exist between the displacement and the velocity profiles make the use of another independant monovariate AR filter to generate velocity realisations invalid. Either multivariate ARMA filters can be considered or the generated displacement realisation can be numerically differentiated. Differentiation was chosen because of the ease of implementation.

The central difference differentiation formula

$$\dot{z}_n = (z_{n+3} - z_{n-3} + 9(z_{n-2} - z_{n+2}) + 45(z_{n+1} - z_{n-1}))/60h + O(h^6) \qquad (6.5)$$

where h = incremental step length,

and truncation error, $O(h^6) = -\frac{1}{140} h^6 z_n^{vii}$

was used.

All symmetric finite difference formulae take the form

$$\acute{z}_n = \sum_{j=-q}^{q} b_j \, z_{n+j} \qquad (6.6)$$

where b_j are weighting coefficients such that

$$b_j = -b_{-j} \quad \text{and} \quad b_o = 0$$

Equation (6.6) is similar in form to a MA filter (see Chapter II - section 5.2). There are however, three essential differences.

1) The summation bounds in (6.6) are $-q < j < q$ compared with $0 < j < q$ (for positive q) in a MA filter.

2) The coefficients of (6.6) are antisymmetric ie. $b_j = -b_{-j}$

3) The filter inputs z_{n+j}, for all j, in (6.6) are correlated in accordance with the characteristics of the displacement AR filter.

Consider the effect of pre-multiplying (6.6) by z_{n-k}, such that

$$z_{n-k} \, \acute{z}_n = z_{n-k} \sum_{j=-q}^{q} b_j \, z_{n+j}$$

Take expected values and normalise with respect to $\langle z_n \, z_n \rangle$ to obtain the relation

$$R_{z\acute{z}}(L) = \sum_{j=-q}^{q} b_j \, R_z(L+j)$$

$$= \frac{d}{dL}\left(R_z(L)\right) + 0\,(h^6) \qquad (6.7)$$

Similarly,

$$R_{\acute{z}z}(L) \simeq -\frac{d}{dL}\left(R_z(L)\right) + 0\,(h^6) \qquad (6.8)$$

$R_{z\acute{z}}(L)$ and $R_{\acute{z}z}(L)$ are correct to the order of the numerical differentiation. It follows that $R_{\acute{z}}(L)$ is also correctly defined, since

$$R_{\acute{z}}(L) = -\frac{d}{dL}\left(R_{zz}(L)\right) \frac{d}{dL}\left(R_{zz}(L)\right) \qquad (6.8)$$

Thus by numerically differentiating the displacement road profile, a velocity profile is generated with the correct derivative correlation characteristics. As this differentiating filter is non recursive it is therefore intrinsically stable.

As the velocity (or differentiating) filter is based on a central difference formula, it follows that this velocity filter is generating output $\dot{z}_n$ at discrete increment number 'n', while the AR filter is outputting displacement z_{n+q} at increment number 'n+q'. A hold routine is introduced after the displacement generator to ensure synchronisation.

7. Conclusion

A monovariate displacement road profile generator can be effectively implemented. The schematic representation is shown in fig. 3.1. A pseudo random number generator is used as a source of uncorrelated uniformly distributed random data. This random data is passed through a Gaussian filter to achieve the correct probability distribution. Adherance to a prescribed displacement correlation is obtained by an AR filter. A velocity filter is used to numerically differentiate the displacement realisation to produce a velocity realisation record. A hold routine is implemented in the path of the displacement realisation to ensure synchronisation between displacement and velocity records. Both displacement and velocity realisations are required as parallel inputs to the vehicle model.

The determination of the weighting coefficients of the AR filter is the most complex part of implementation. Both physical realisability of the displacement autocorrelation and numerical stability of the AR filter must be ensured.

Operation Method	Summutation "+" or "-"	Multiplication "X"	Division "÷"	Integer Division "%"	Square Root "√"	Logarithm "LN"	Arithmetic "IF"
Teichroew							
-Basic Routine	16	6	1	-	-	-	-
-RNG * Calls(12)	12	36	-	12	-	-	-
-Total	28	42	1	12	-	-	-
Polar							
-Basic Routine	2 ($2\frac{1}{4}$)	4 ($4\frac{1}{2}$)	1	-	1	1	1 ($1\frac{1}{4}$)
-RNG * Calls (2)	2 ($2\frac{1}{2}$)	6 ($7\frac{1}{2}$)	-	2 ($2\frac{1}{2}$)	-	-	-
-Total	4 ($4\frac{3}{4}$)	10 (12)	1	2 ($2\frac{1}{2}$)	1	1	1 ($1\frac{1}{4}$)
Difference in total number of operations (Teichroew-Polar)	+24 ($+23\frac{1}{4}$)	+32 (+30)	0	+10 ($+9\frac{1}{2}$)	-1	-1	-1 ($-1\frac{1}{4}$)
* NB Each Call of RNG requires	1	3	-	1	-	-	-

Table 3.1 Comparison of the number of arithmetic operations to generate one value of r_G, using both Teichroew and Polar methods.

Test Number	1	2	3	4	5	6	Average
Degree of Redundancy	0.269	0.279	0.293	0.250	0.265	0.278	0.272

Table 3.2 Empirical observations on the degree of redundancy present when using the Polar-Gaussian Filter.

REFERENCES - CHAPTER III

Ref. 1 Teichroew, D. (1953) "Distribution sampling with high speed computers" Diss., Uni. North Carolina.

Ref. 2 Box, G.E.P., and Muller, M.E. (1958) "A note on the generation of random normal deviates" Ann. Math. Stat. 29,610 - 611.

Ref. 3 Jury, E.I. (1958) "Sampled data control systems" pub. Wiley.

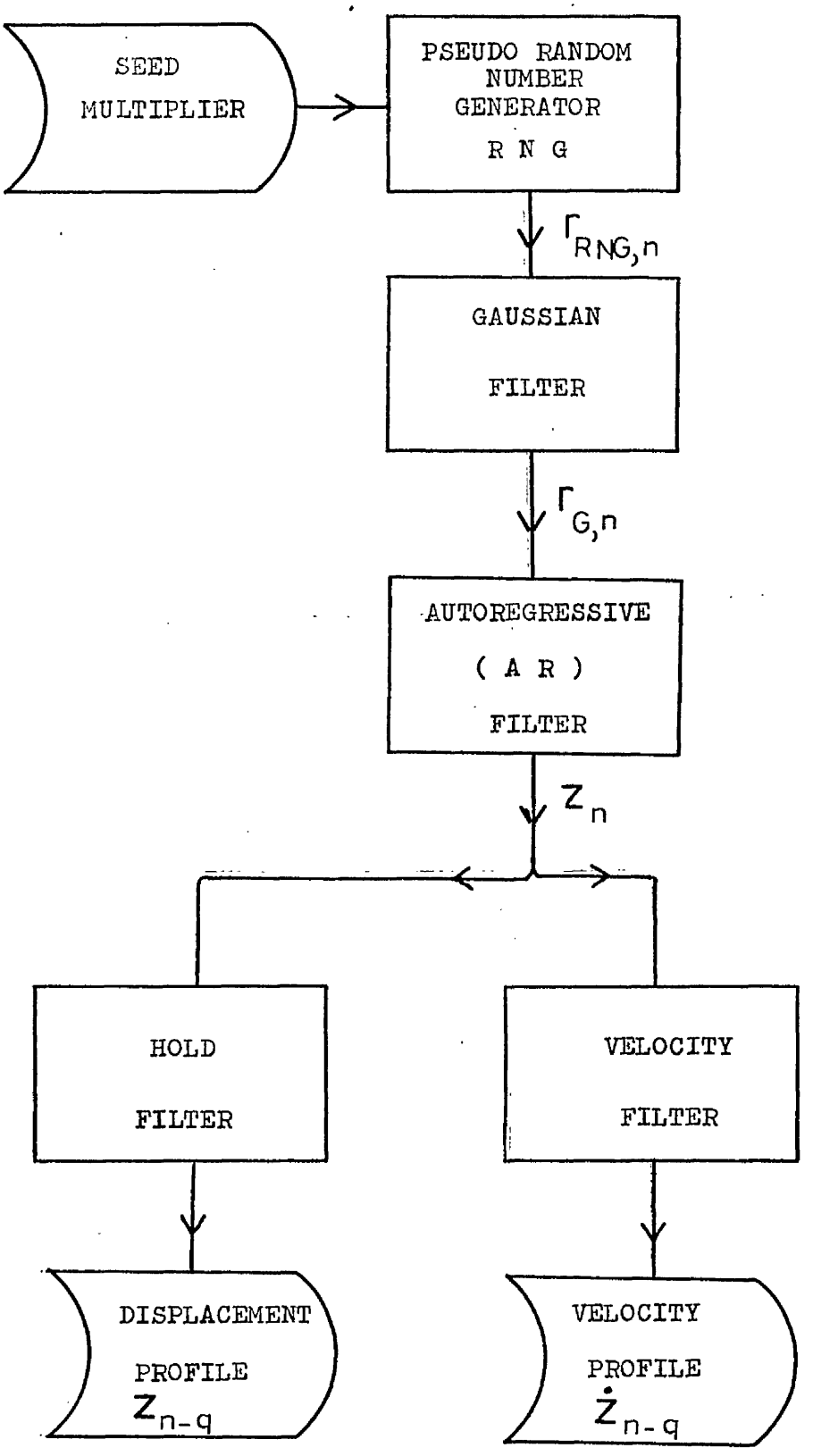


FIG 3.1 Program flow logistics of the Road Profile Generator.

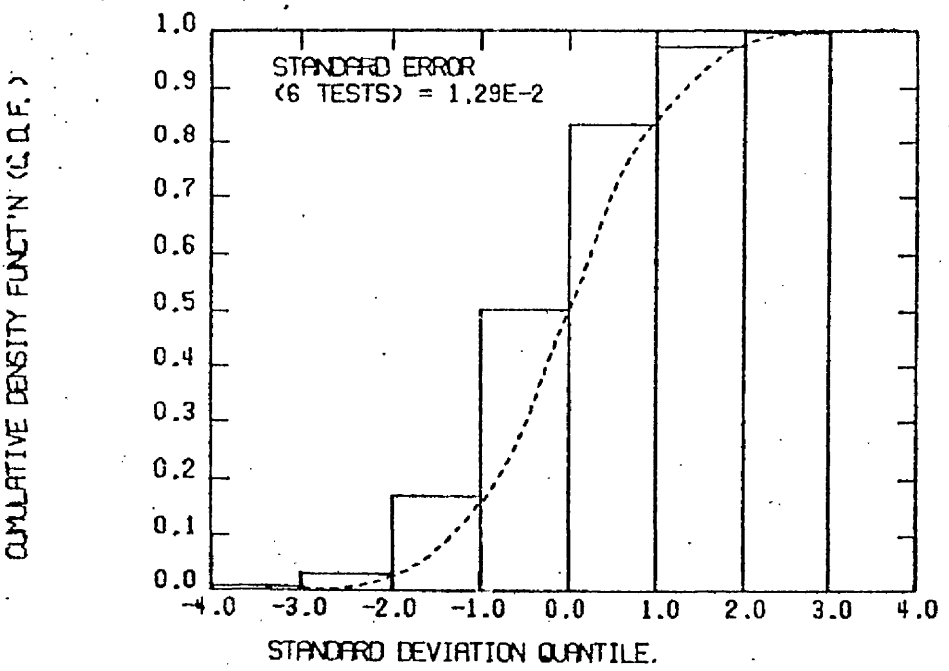


FIG. 3.2 C.D.F. AVERAGE TEICHROEW VS. GAUSSIAN DISTRBN.

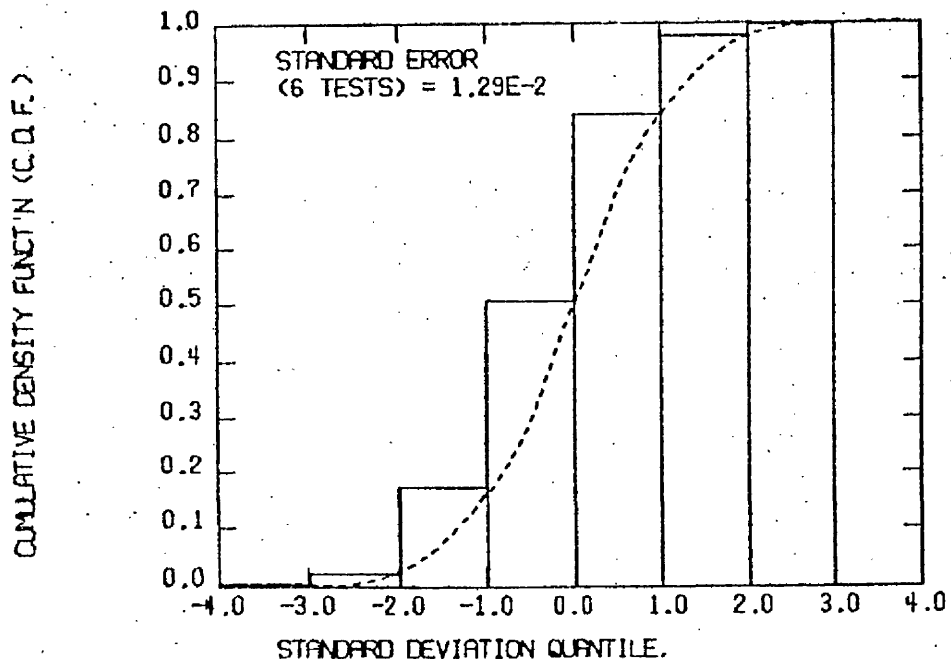


FIG. 3.3 C.D.F. AVERAGE POLAR VS. GAUSSIAN DISTRBN.

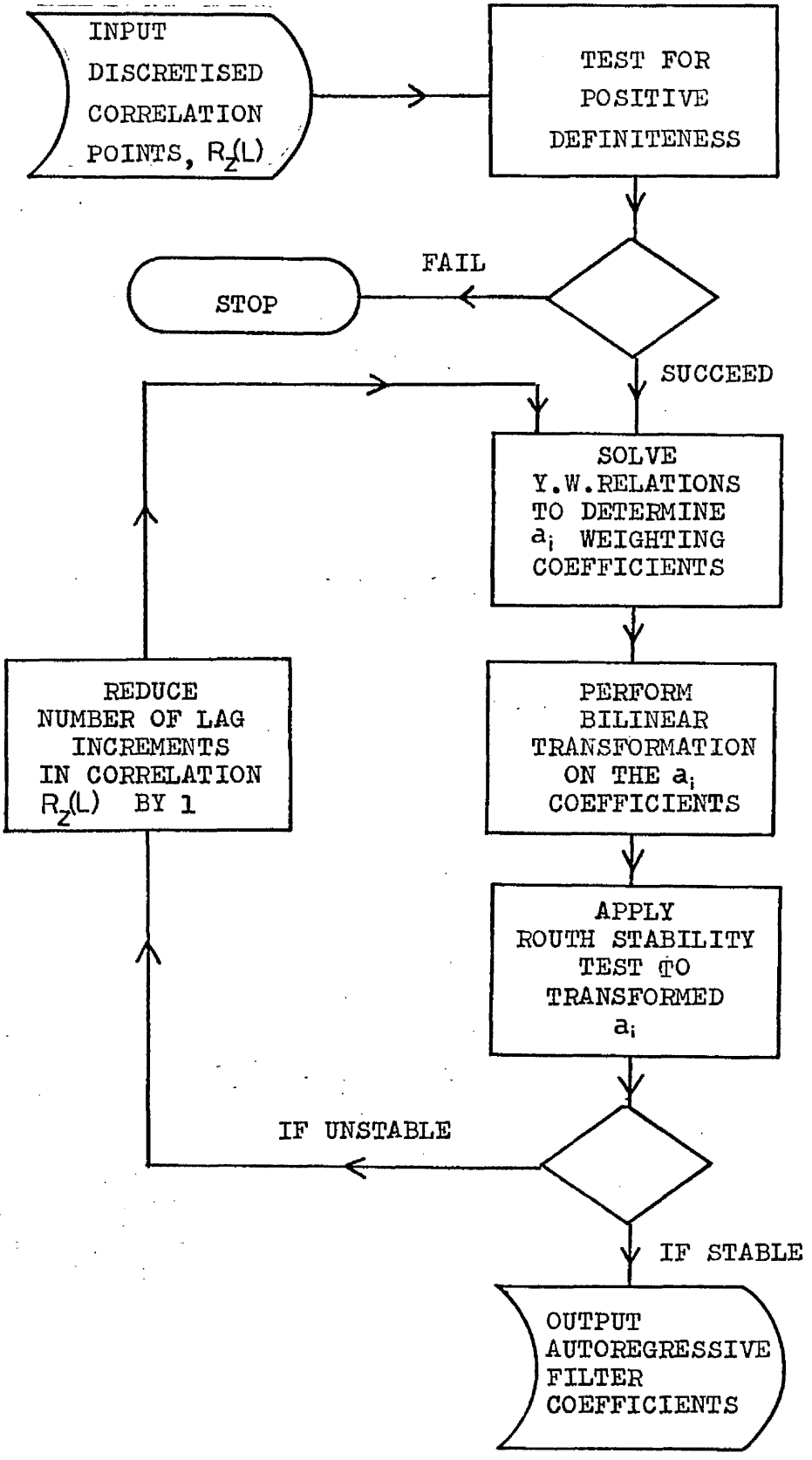


FIG 3.4 Program flow logistics of the AR Filter coefficient determination procedure.

CHAPTER IV

TESTING THE ROAD PROFILE GENERATOR

1. Introduction and objectives

A road profile generator is required to generate realistic combinations of displacement and velocity road profiles for input into the second order DE vehicle problem. Such profile realisations must meet prescribed requirements concerning their statistical behaviour. The simplest and most satisfactory method of verifying the profile generator is by the implementation of empirical tests on generated realisations.

The two prescribed statistical requirements which require verification are as follows.

1) To ensure the statistical adequacy of the road profile generator in achieving the correct auto-and cross-correlations.

2) To ensure that Gaussian realisations are adequately reproduced.

All of the following prescribed road profile correlations were tested for physical realisability and their numerical stability ensured before these statistical tests were undertaken.

2. The example road profile correlations tested

Three example road profile correlations were selected for testing purposes. These correlations are described in the following subsections.

The nature of the road profile generator is such that it requires a discretised version of the displacement auto-correlation, $R_z(L)$.

In the process of discretisation, the substitution

$$n\,h(x) = L \qquad (2.1)$$

where $h(x)$ = discretised spatial lag increment

and $n = 0, 1, 2, \ldots, N$

must be made, to obtain the discretised correlation $R_z(n\,h(x))$. The derivative correlations can be similarly discretised.

2.1 The Virchis Correlation

This approximate algebraic description of an actual road profile correlation was proposed in Ref. 1. This idealised correlation was described by the equation

$$R_z(L) = 0.6 \exp(-c\,L^2) + 0.4 \exp(-c\,L) \cos d \qquad (2.2)$$

where $c = 0.001$, $d = \frac{\pi L}{129}$.

From Chapter III (6.3), the relation

$$R_{z\dot{z}}(L) = -R_{\dot{z}z}(L) = \frac{d}{DL}(R_z(L)) \qquad (2.3)$$

was established.

Differentiate (2.2) to obtain

$$R_{z\dot{z}}(L) = -4\,10^{-4}\left\{3\,L \exp(-c\,L^2) + \exp(-c\,L)\left[\cos d + \frac{\pi\,10^3}{129} \sin d\right]\right\} \qquad (2.4)$$

At $L = 0$, the theoretical relation (2.3) yields the condition

$$R_{z\dot{z}}(0) = R_{\dot{z}z}(0) = 0 \qquad (2.5)$$

while in (2.4)

$$R_{z\dot{z}}(0) = -4 \times 10^{-4} \neq 0 \qquad (2.6)$$

This continuous derivative correlation does not satisfy the theoretical constraint. (The authors of Ref. 1 could ignore this problem as only the displacement correlation was considered.) In a discretised empirical approach considered, the value of $R_{z\dot{z}}(0)$ in (2.6) is sufficiently small to be ignored.

2.2 The Exponential Correlation

This idealised correlation takes the form

$$R_z(L) = \exp(-B|L|) \tag{2.7}$$

It was the correlation analysed in Ref. 2.

The derivative correlation $R_{\dot{z}z}(L)$ is given by the equation

$$R_{\dot{z}z}(L) = -B \exp(-B|L|) \tag{2.8}$$

where B is a constant.

As with Virchis, at $L = 0$

$$R_{\dot{z}z}(0) \neq 0.$$

The fact that the derivative correlation (2.8) achieves maximum amplitude at $L = 0$, where it also exhibits a large discontinuity, makes this derivative process meaningless. Uncorrelated realisations of the derivative process must therefore be anticipated.

2.3 The Modified Exponential Correlation

The modified exponential correlation has the form

$$R_z(L) = (1+B|L|) \exp(-B|L|) \tag{2.9}$$

where B is a constant.

The differential correlation

$$R_{\dot{z}z}(L) = -B^2|L| \exp(-B|L|) \tag{2.10}$$

satisfies the zero value constraint at $L = 0$.

3. The auto- and cross-correlation test results

Displacement, velocity, and displacement - velocity auto- and cross-correlations were calculated from realisations of the road profile generator. (The velocity - displacement cross-correlation can reasonably be expected to be the mirror image of the displacement - velocity correlation.) Each realisation test comprised of 1156 generated data increments. An initial period of transience was allowed to elapse before any correlation measurements were recorded. Correlations were calculated up to a maximum lag increment, N_{max}, of 100 (less than one tenth of the realisation record length). The number of lag increment coefficients, N, present in the various AR filters tested was always very much less than N_{max}. Comparisons are, however, drawn between theoretical and generated correlations up to $N_{max} = 100$.

With all observed results up to lag N, the realisations closely adhered to the theoretical correlation curves, with maximum variation occurring, as can reasonably be expected, in regions of high gradient. The variation between any realised correlation value and the theoretical curve was never more than 10% of the auto-correlation value at zero lag (ie the standard deviation). Clearly this variation is dependant on the length of the realisation record. The standard error (SE) on such realisation records is known to be inversely proportional to $\sqrt{M}$, where M is the number of sample points in the record, thus if M = 1156, then $1/\sqrt{M}$ = 2.94 E -2. It is interesting to observe that this has the same order of magnitude as all normalised variations mentioned above (ie less than 10%).

With lag increments greater than N, the observed realisations may or may not adhere to the theoretical correlation curve. The degree of adherence depends overwhelmingly on the nature of the correlation curve.

Reference is now made to the individual correlations tested. A considerable number of graphs are included. In each graph two sample realisations are shown plotted with their theoretical equivalent correlation.

3.1 The Virchis Correlation

A sample Virchis correlation was tested with N_{max} set to 20, and h(x) = 12 ft. The coefficients of this 20 pt filter are shown in Table 4.1. Samples of these correlation test results are shown in Figs. 4.1.a - 4.1.c. Lag increments as high as N = 20, demonstrates faithful reproduction of the theoretical correlation. Above N = 20 significant residual correlation is evident. From the theoretical nature of the AR filter this result can be anticipated. An AR filter generates to a prescribed correlation curve by the use of a series of damped exponential decay functions (both real and complex) each of random amplitude. Consequently at lags greater than N sharp cut off of the correlation is impossible to achieve. In the case of this Virchis example, at N = 20 the model

correlation value has significant magnitude (about 0.3 of maximum), it is hardly surprising therefore that a large residual correlation is present.

As the derivative correlations are obtained from realisations which are generated by numerically differentiating the displacement realisation, it is hardly surprising that these correlations also have significant residual values.

3.2 The Exponential Correlation

The discretised exponential correlation had the following parameter values,

$B = 1.246$

$N_{max} = 4$

$h(x) = 1$ ft

The coefficients of the 4 pt AR filter are shown in Table 4.2. (Clearly, by virtue of the fact that the AR filter generates the required correlation by means of damped exponential impulses, it should be possible to generate this simple exponential displacement correlation by a two point filter.) However, the 4 pt filter tested also achieves good displacement correlation results up until $N = 20$ (see fig 4.2.a). For $N > 20$, the degree of correlation is minimal and the realised results are mainly the result of random noise. The derivative correlations in this case do not exist.

3.3 The Modified Exponential Correlation

The parameters of the discretised modified exponential correlation were as follows,

$B = 2.181$, $N_{max} = 4$

$h(x) = 1$ ft.

This correlation with the above values is a reasonable facsimile of the exponential correlation in the previous subsection. The filter coefficients are given in Table 4.3. Good auto- and cross-correlation results are obtained up until lag $N = 20$ (see fig 4.3a, b and c). With the low degree of correlation recorded after $N = 20$, random noise starts to become significant.

4. Optimum order for the AR Filter

Apart from correlations which have extremely simple forms (like the exponential correlation where an order of $N = 2$ would suffice), it is a difficult task to estimate an optimum filter order. In the vehicle simulation problem the choice of value for the spatial lag increment, $h(x)$, is of vital importance to ensure viability in the final solution. The value of $h(x)$ must be selected in accordance with the range of traverse velocities experienced by the vehicle. Many realistic road profile correlations take the form of piecewise continuous functions. Each continuous segment must be adequately represented in the AR filter. The combined effect of a predetermined step length $h(x)$, and a piecewise continuous correlation curve will be to ensure that the optimum AR filter order is approximately that of the maximum lag increment on the correlation curve. Thus with realistic correlation curves there is little point in considering AR filter order optimisation. For this reason, optimisation is not considered in the test correlations either.

5. Probability distribution of the Road Profile Generator realisations

The statistical description of a road profile is normally defined in terms of the displacement auto-correlation $R_z(L)$ (or by the power spectral density - the Fourier transformed equivalent). Most measured road surfaces are described in this manner.

Implicit in the displacement auto-correlation description are the displacement and velocity profile variances, which are defined respectively as

$$\left.\begin{aligned} \text{Var}(z) &= \langle z_n \; z_n \rangle = R_z(0) \\ \text{Var}(\dot{z}) &= \langle \dot{z}_n \; \dot{z}_n \rangle = R_{\dot{z}}(0) = -\frac{d^2}{dL^2}\left(R_z(0)\right) \end{aligned}\right\} \quad (4.1)$$

If both profiles are Gaussian random variables, quantifying these variances are enough to fully define the probability distributions. Exacting tests are now undertaken to ensure the adherence of the realisations to the correct Gaussian distribution.

In the tests which follow, the cumulative density function (CDF) of all quantities output by the road profile generator were compared with a Gaussian distribution. The CDF is defined as follows,

$$\text{CDF} = \int_{-\infty}^{V} \text{Pr}\,(v)dv \qquad (4.2)$$

where $\text{Pr}\,(v)$ refers to the probability density function. Each CDF realisation plot is sampled over 1100 data points. The mean (Mn) and the variance (Var) of the sample were obtained. Bias was removed from any realised record of v by the formula

$$v_{norm} = \frac{v - M_n\,(v)}{\sqrt{\text{Var}\,(v)}} \qquad (4.3)$$

The values of v_{norm} were then arranged in ascending order and the CDF distribution curve of v_{norm} determined.

Results specific to the various output quantities of the road profile generator are now discussed in turn.

(a) Output (r_G) from the Polar filter.

As mentioned in Chapter III, r_G is the output of the intermediate stage in the road profile generator prior to processing by the AR filter. It is important to compare the probability distribution of r_G with the distribution of the quantities eventually output by the road generator, in order to assess the degree of degradation in the distribution caused by the AR and velocity filtering stages.

A sample normalised CDF (r_G) is shown plotted against the theoretical distribution in fig. 4.4.

The maximum divergence, Max div (r_G), from the theoretical Gaussian distribution was 2.9 E - 2

The non-zero mean value, $M_n\,(r_G)$ was 6.82 E - 2

The variance, Var (r_G) was 1.01

The number of events exceeding a 3 x Var (r_G) limit was 3.

(b) Output (z) from the AR filter

Output z is the displacement road profile realisation. The displacement realisations for the various correlations mentioned in

the previous sections were tested. The results are in Table 4.4. References to the various graphs are also included in the Table.

(c) Output ($\dot{z}$) from the Velocity filter

Sample realisations of the velocity road profile generator were measured for probability distribution. The results with reference to appropriate graphs are shown in Table 4.5.

In all three cases (a, b, and c) the realised probability distributions are of a very high standard. The realisations were noted to be slightly rich in extreme events (ie greater than 3 x Var). With linear suspension configurations this effect is of minimal concern, however, such effects should be closely monitored when non-linear analysis is attempted.

6. Empirical Evidence as to the stable nature of AR filters

Quite apart from the AR filters specifically described in the previous sections, other AR filters were tested with many parametral variations while still keeping to the three aforementioned road profile correlations. Such variations are not specifically mentioned as they offer no additional information as to the potential of the road profile generator. However, with none of these other AR filters extracted and tested was instability ever recorded. There appears to be considerable empirical evidence to indicate that physically realisable road profile correlations are always stable.

7. Conclusion

The road profile generator has been shown to work well with all the road profile correlations tested. Both probability distribution and correlation requirements have been satisfied.

The order of the AR filter used to generate a prescribed correlation need not be as great as the maximum lag increment for the prescribed correlation. However, the criterion which should be

used to optimise the filter order will in fact be overruled by other physical constraints imposed by the model. These constraints concern the nature of the correlation curve, which in many real cases is piecewise continuous, and of the incremental step length optimised to best suit the range of traverse velocities under consideration.

The quality of the probability distribution of the output is exceptional. It is only slightly rich in a few extreme values.

In the many AR filters tested, for any physically realisable road profile correlation, no instability was ever encountered. It is suggested from this empirical evidence that AR filters obtained from physically realisable road profile correlations might always be stable.

Filter Order, N_{max} = 20; Incremental step length, h(x) = 12ft			
Lag Increment, i	AR Filter Coefficient, a_i	Lag Increment, i	AR Filter Coefficient, a_i
1	1.9683135 E 0	11	-1.9278726 E-2
2	-1.7022791 E 0	12	-5.3024532 E-4
3	6.9039996 E-1	13	9.8034381 E-4
4	8.4133426 E-2	14	-2.7388283 E-3
5	-2.6660150 E-1	15	4.3232381 E-3
6	9.8588392 E-2	16	9.2711116 E-4
7	3.2534771 E-2	17	3.6037029 E-3
8	-6.8717380 E-2	18	1.7752569 E-2
9	1.0552446 E-2	19	-1.7733661 E-2
10	5.8651331 E-3	20	2.6920412 E-2

Table 4.1 AR filter coefficients for a 20 point Virchis Correlation.

Filter Order, N_{max} = 4; Incremental step length, h(x) = 1ft			
Lag Increment, i	AR Filter Coefficient, a_i	Lag Increment, i	AR Filter Coefficient, a_i
1	8.7952211 E-1	3	-2.0468783 E-13
2	-1.5161713 E-13	4	2.3509536 E-13

Table 4.2 AR filter coefficients for a 4 point Exponential Correlation

Filter Order, N_{max} = 4; Incremental step length, h(x) - 1ft			
Lag Increment, i	AR Filter Coefficient, a_i	Lag Increment, i	AR Filter Coefficient, a_i
1	1.8632284 E 0	3	2.8576759 E-1
2	-1.1304446 E 0	4	-5.0868043 E-2

Table 4.3 AR filter coefficients for a 4 point Modified Exponential Correlation.

Correlation Curve	Number of events exceeding 3 standard deviations	Maximum divergence from Normal Distribution	Average Value	Variance	Consult graph
Virchis	3	4.2 E-2	6.04 E-1	1.80 E 1	Fig 4.5a
Virchis	5	2.8 E-2	-4.33 E-1	2.20 E 1	Fig 4.5b
Exp.	4	2.6 E-2	5.66 E-1	4.57	Fig 4.5c
Mod. Exp.	5	2.9 E-2	2.13	5.58 E 1	Fig 4.5d

Table 4.4 Displacement statistics of sample realisations from the Road Profile Generator. sample size, N = 1156

Correlation Curve	Number of events exceeding 3 standard deviations	Maximum divergence from Normal Distribution	Average Value	Variance	Consult graph
Virchis	4	2.1 E-2	2.30 E-4	2.53 E-2	Fig 4.6a
Mod. Exp.	3	4.4 E-2	-1.74 E-3	2.42	Fig 4.6c

Table 4.5 Velocity statistics of sample realisations from the Road Profile Generator. sample size, N = 1156

REFERENCES - CHAPTER IV

Ref. 1 Virchis and Robson (see Chapter I Ref. 2)

Ref. 2 Sobczyk and Macvean (see Chapter I Ref. 3)

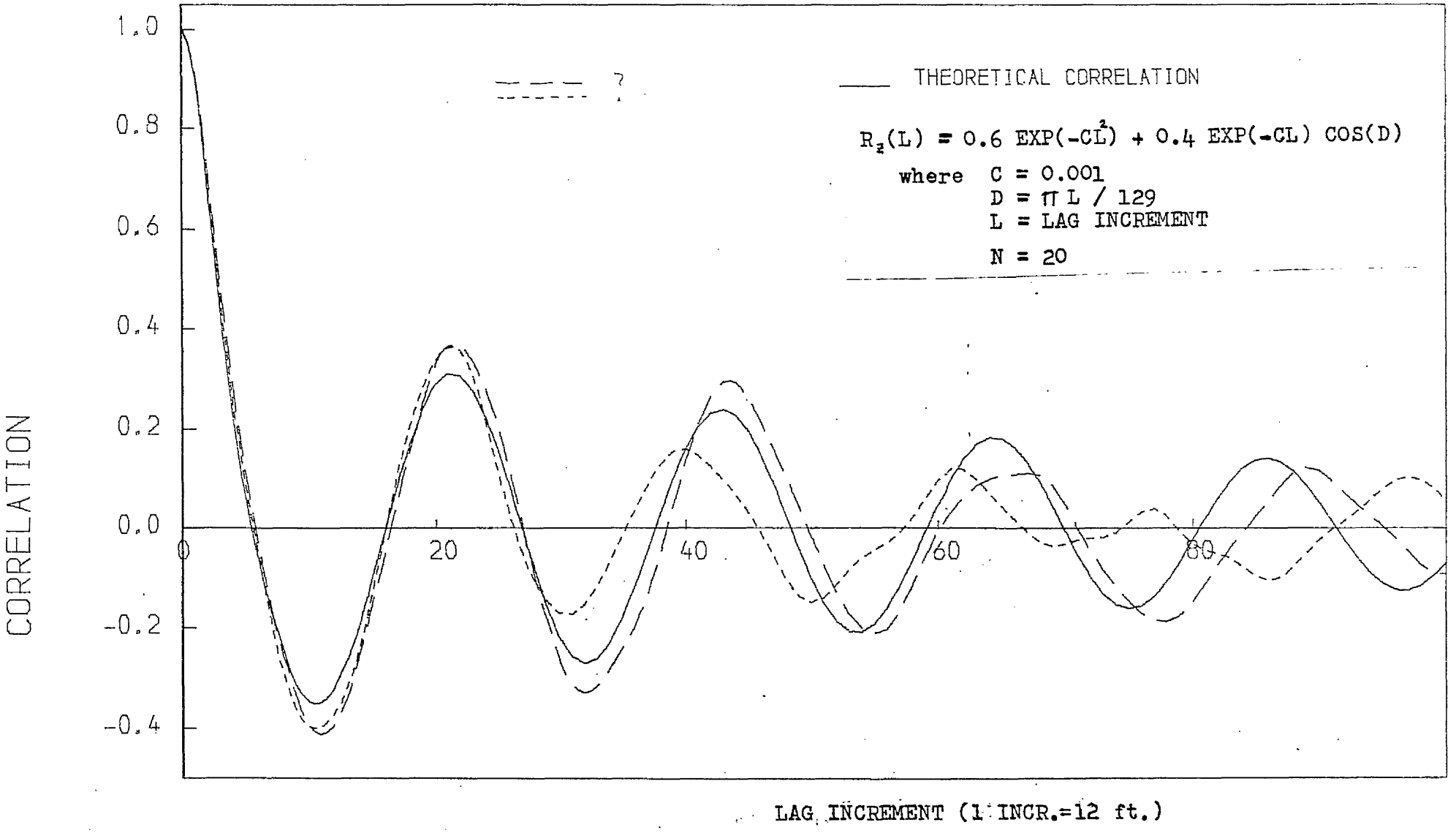


FIG. 4.1a GENERATED REALISATIONS C.F. THEORETICAL VIRCHIS CORRELATION - (DISPLACEMENT-DISPLACEMENT CORRELATION)

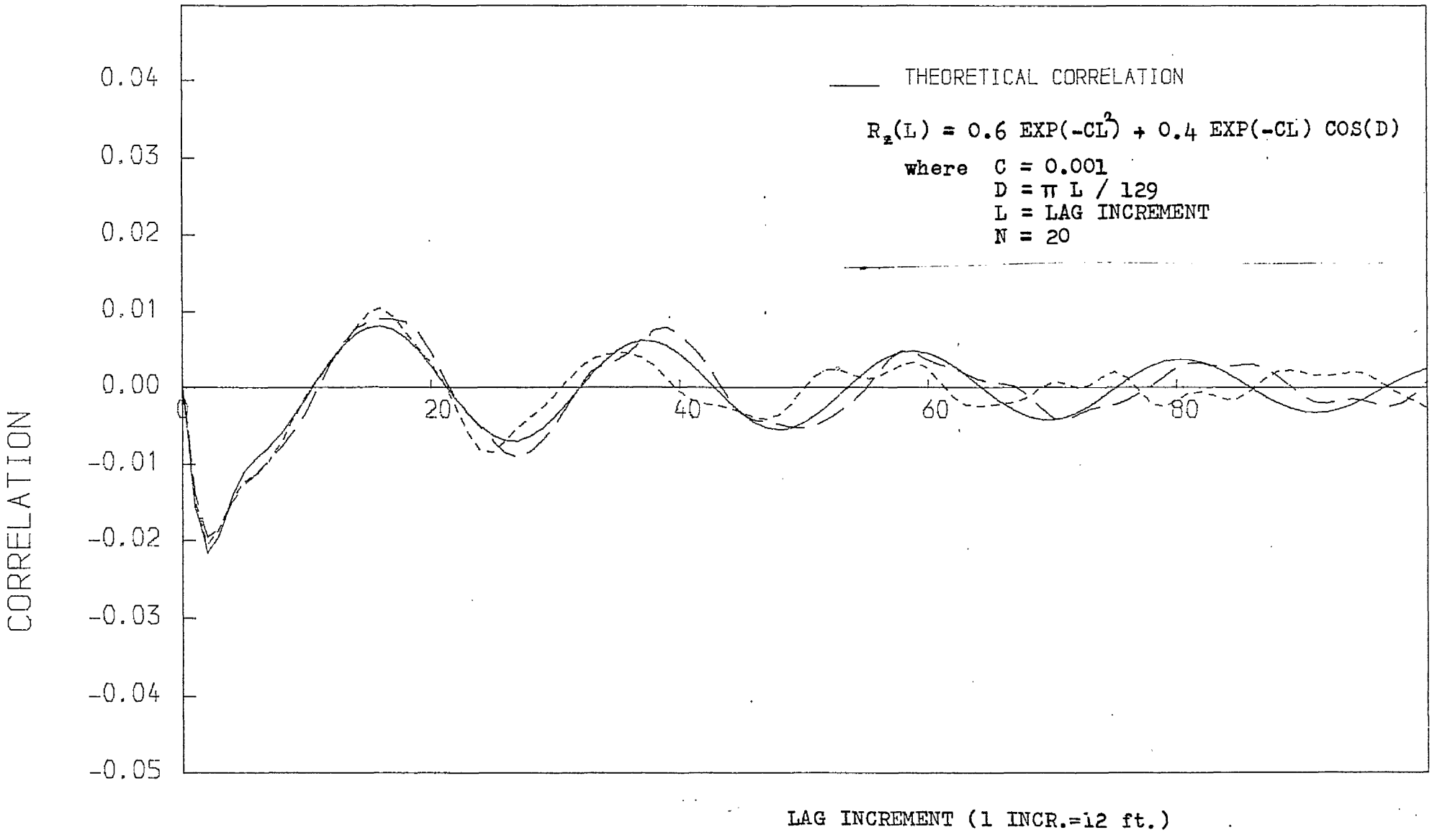


FIG. 4.16 GENERATED REALISATIONS C.F. THEORETICAL VIRCHIS CORRELATION - (DISPLACEMENT-VELOCITY CORRELATION)

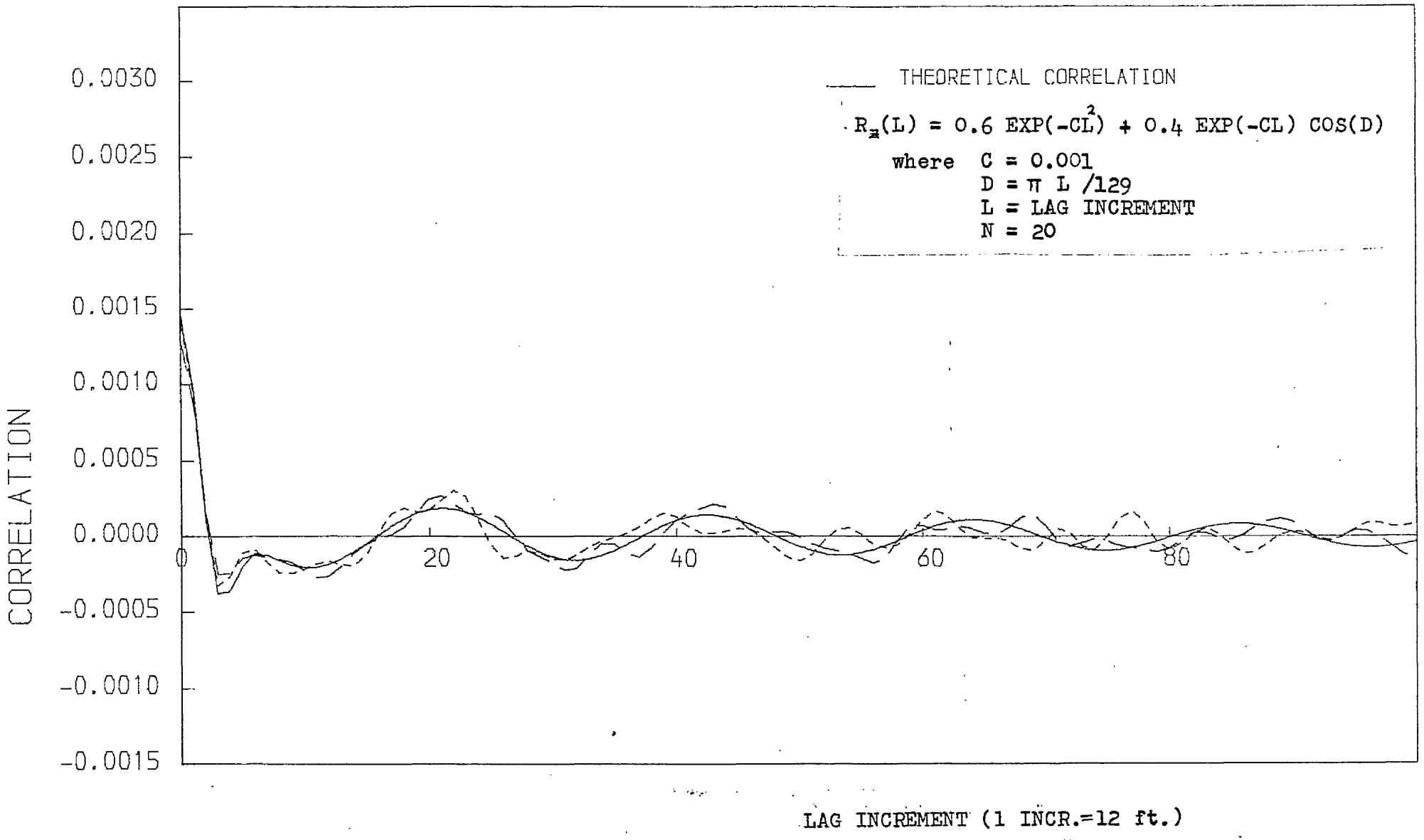


FIG. 4.1c GENERATED REALISATIONS C.F. THEORETICAL VIRCHIS CORRELATION - (VELOCITY-VELOCITY CORRELATION)

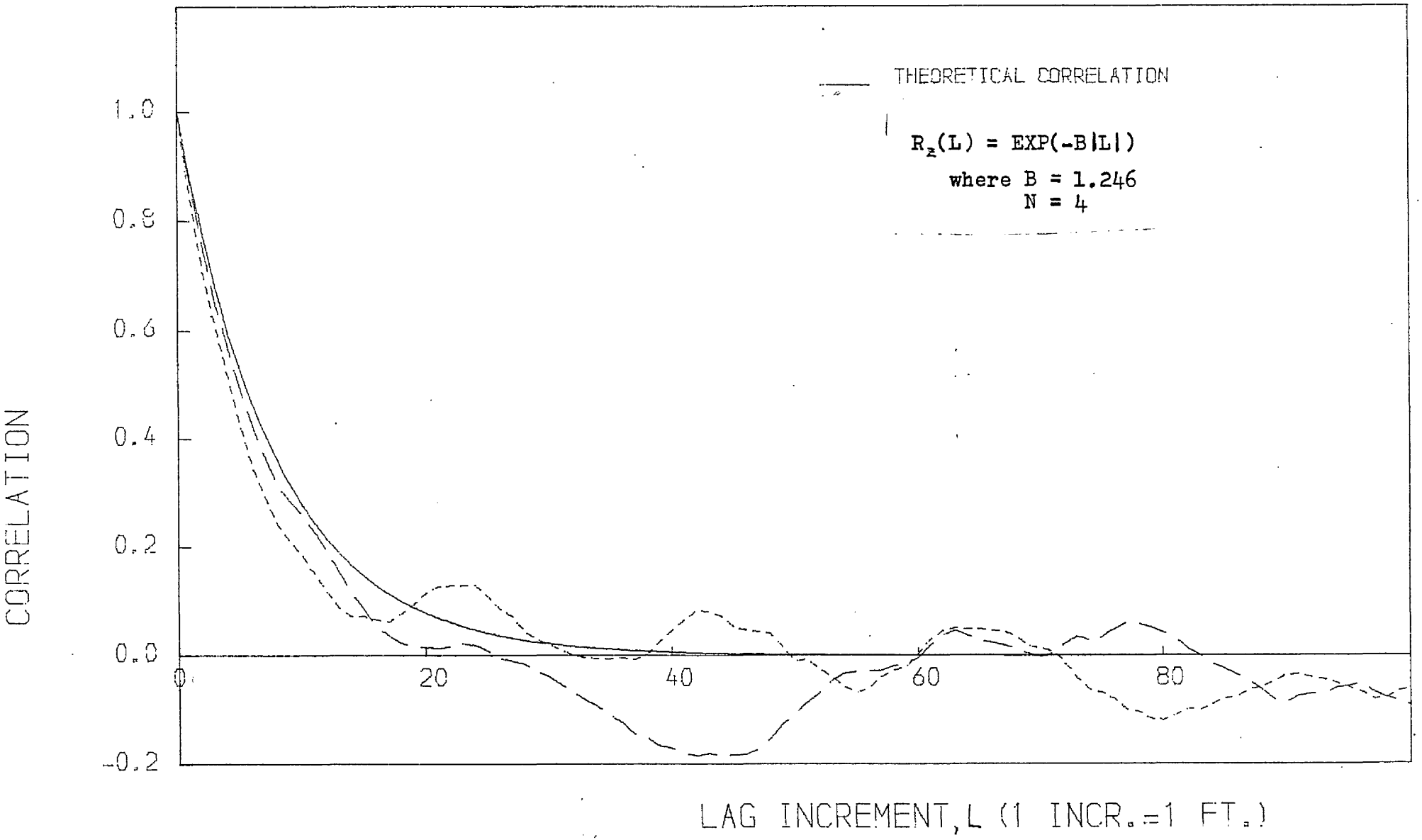


FIG. 4.2 CORRELATIONS - GENERATED C.F. THEORETICAL , - EXP. CORRELN. - (DISPLACEMENT-DISPLACEMENT CORRELATION)

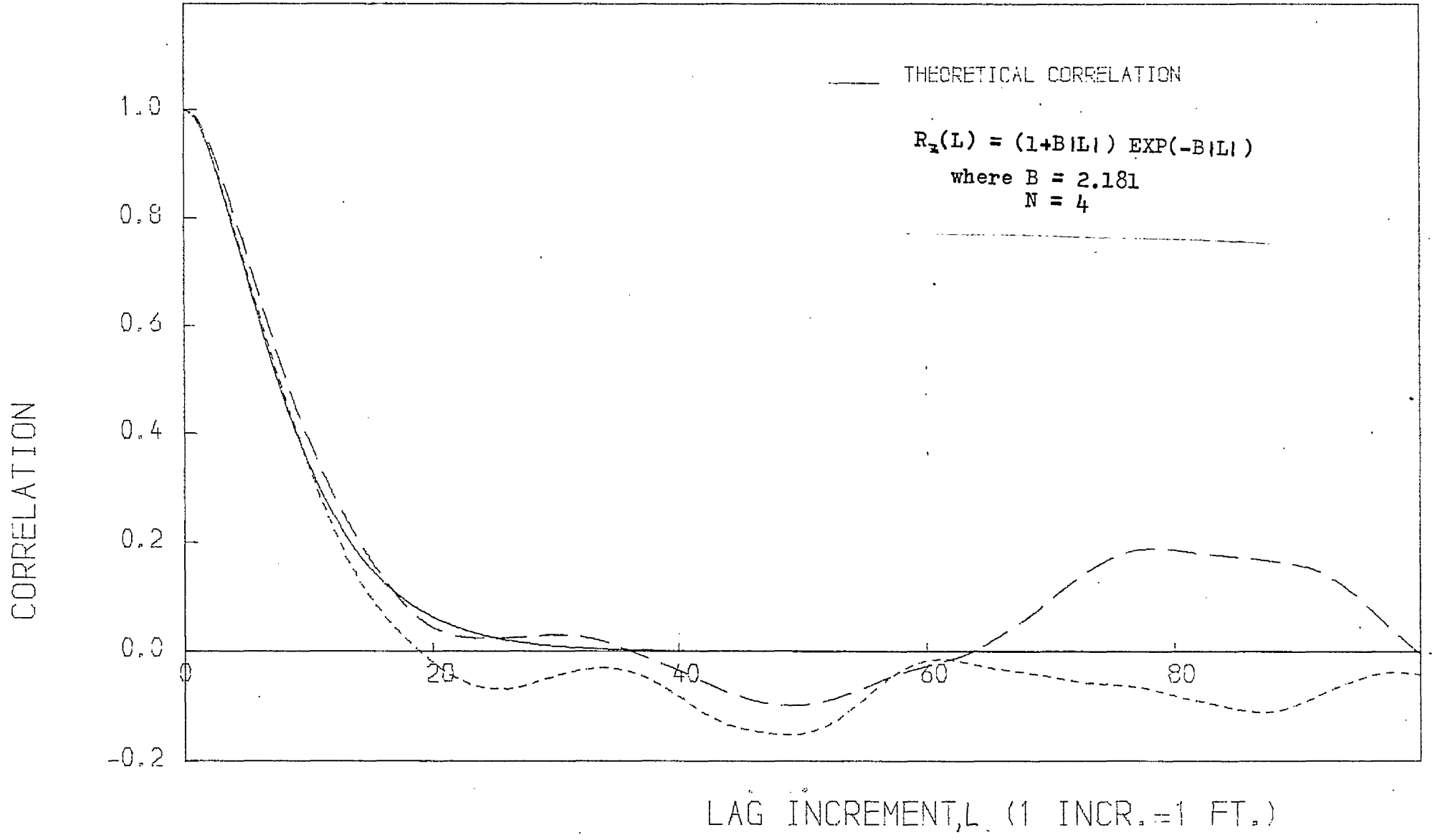


FIG. 4.3a CORRELATIONS - GENERATED C.F. THEORETICAL - MOD. EXP. CORRELN. -(DISPLACEMENT-DISPLACEMENT CORRELATION)

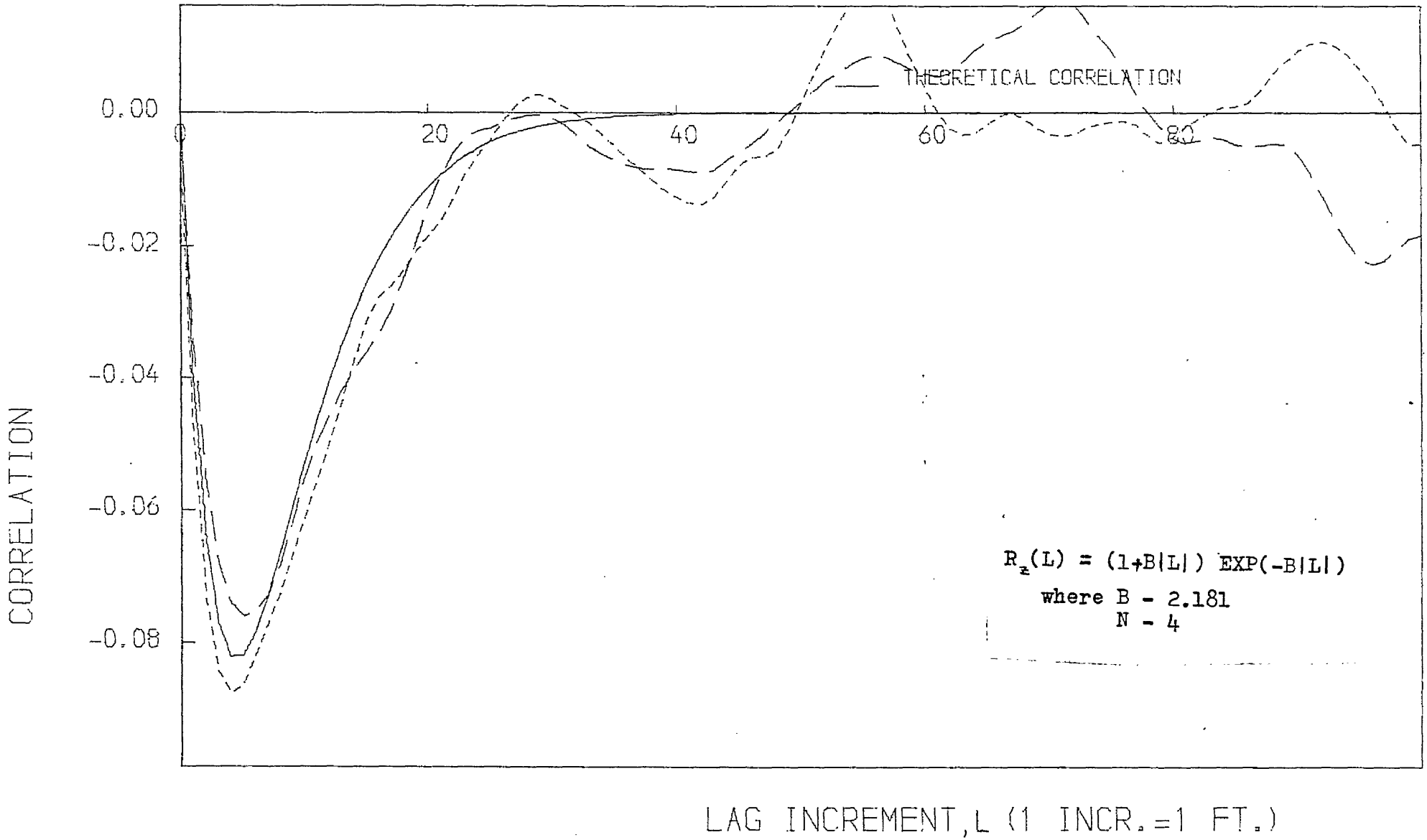


FIG. 4.3b CORRELATIONS - GENERATED C.F. THEORETICAL - MOD. EXP. CORRELN. - (DISPLACEMENT-VELOCITY CORRELATION)

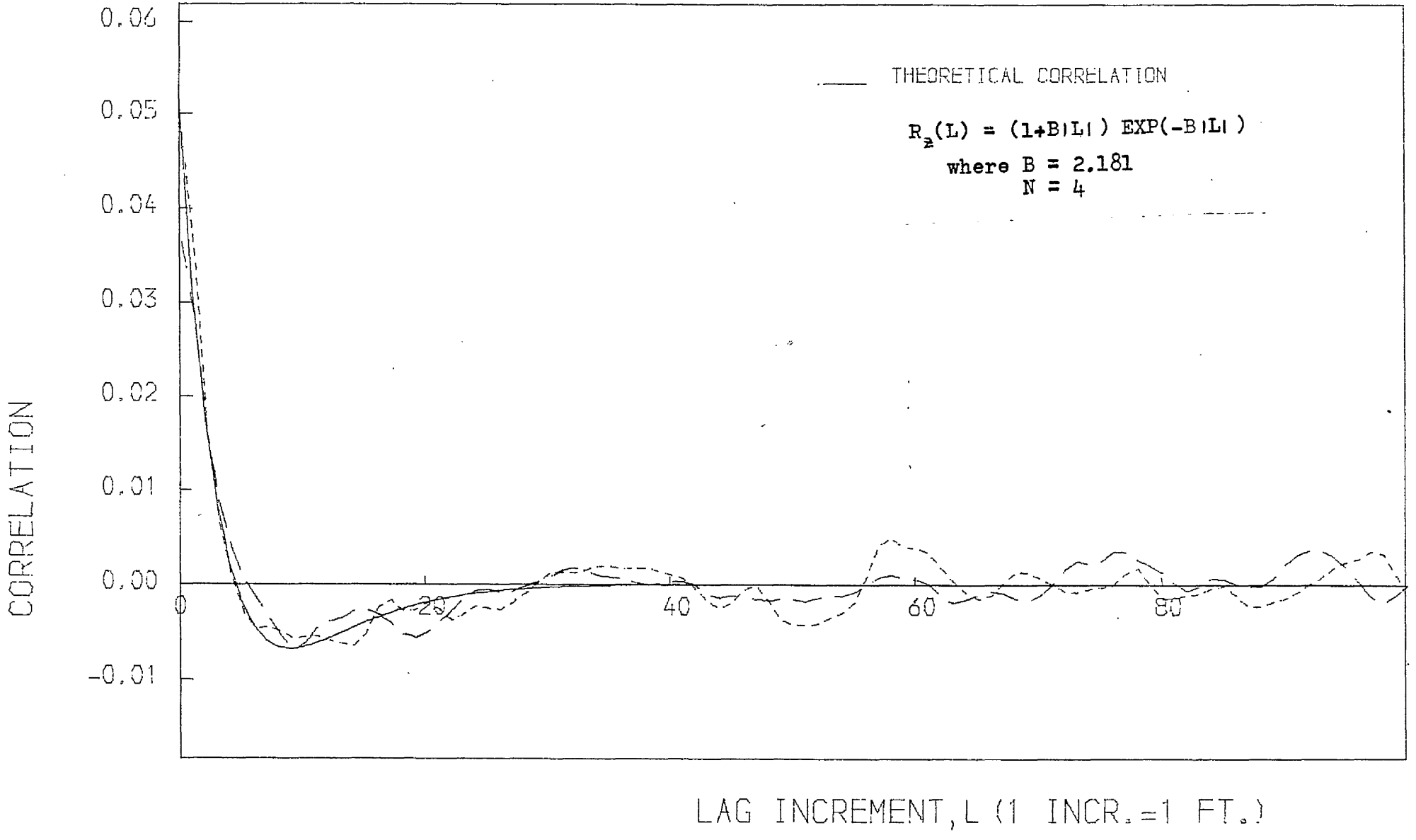


FIG. 4.3c CORRELATIONS - GENERATED C.F. THEORETICAL - MOD. EXP. CORRELN. - (VELOCITY-VELOCITY CORRELATION)

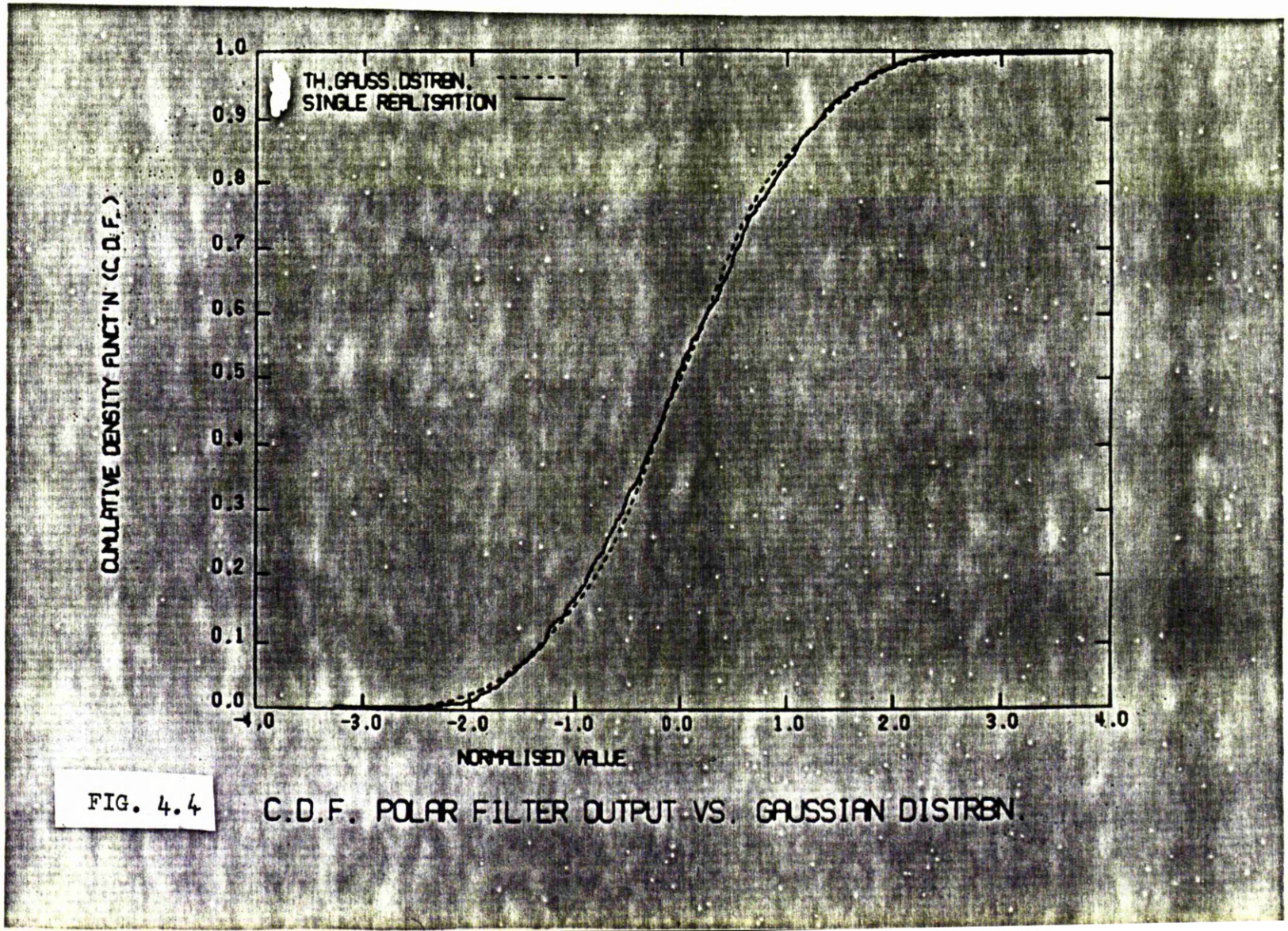


FIG. 4.4 C.D.F. POLAR FILTER OUTPUT VS. GAUSSIAN DISTRBN.

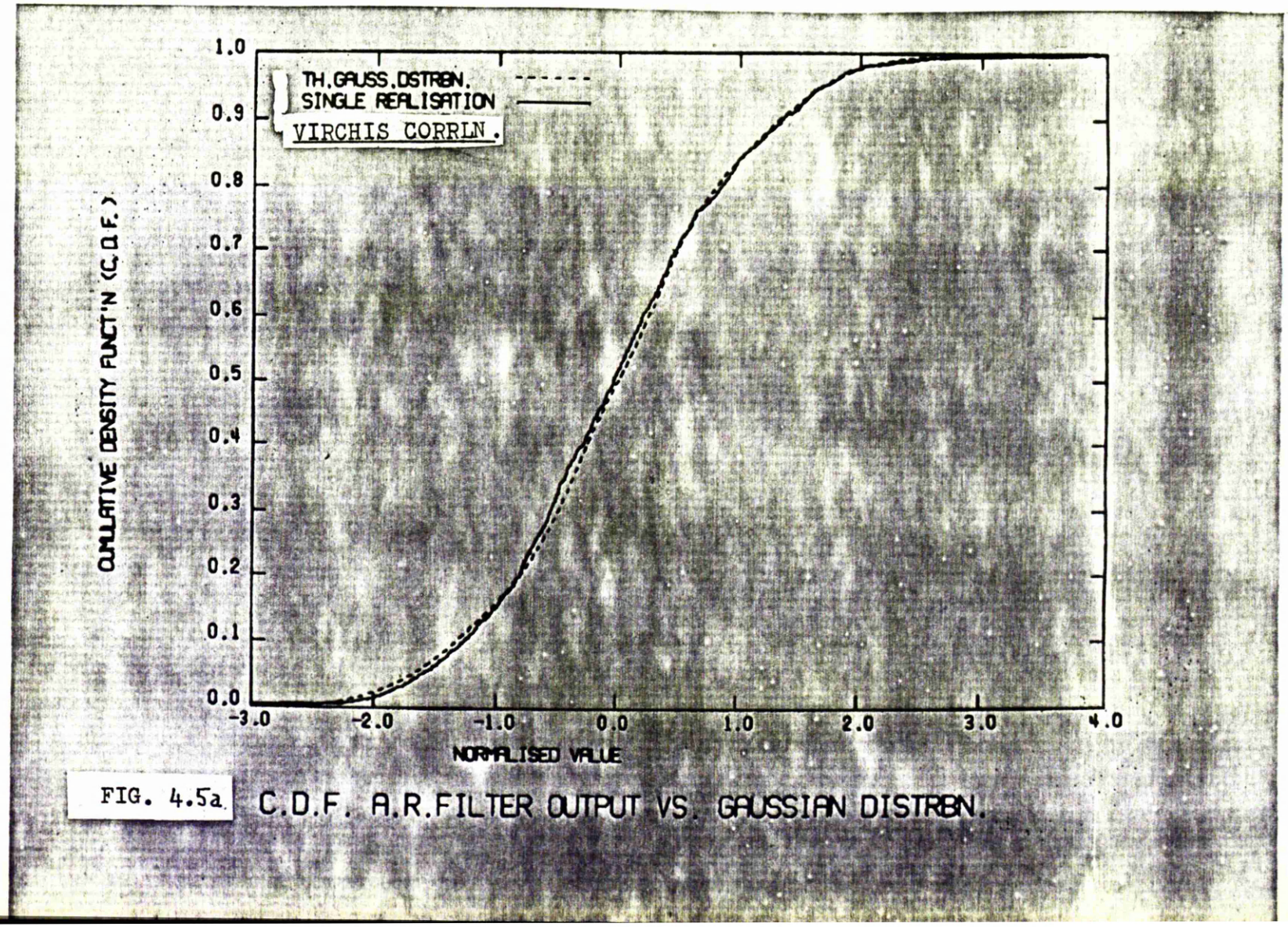


FIG. 4.5a C.D.F. A.R.FILTER OUTPUT VS. GAUSSIAN DISTRBN.

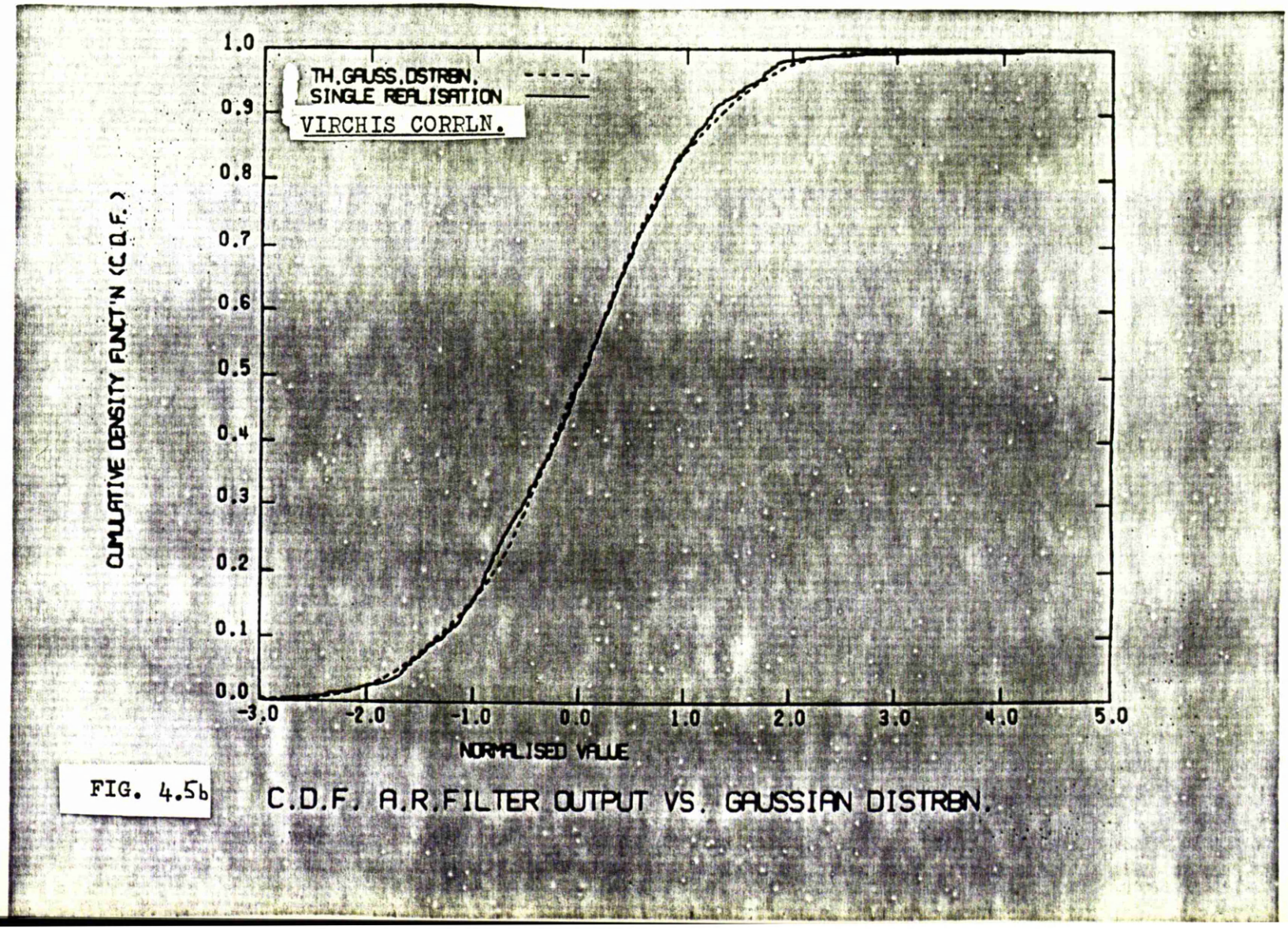


FIG. 4.5b C.D.F. A.R.FILTER OUTPUT VS. GAUSSIAN DISTRBN.

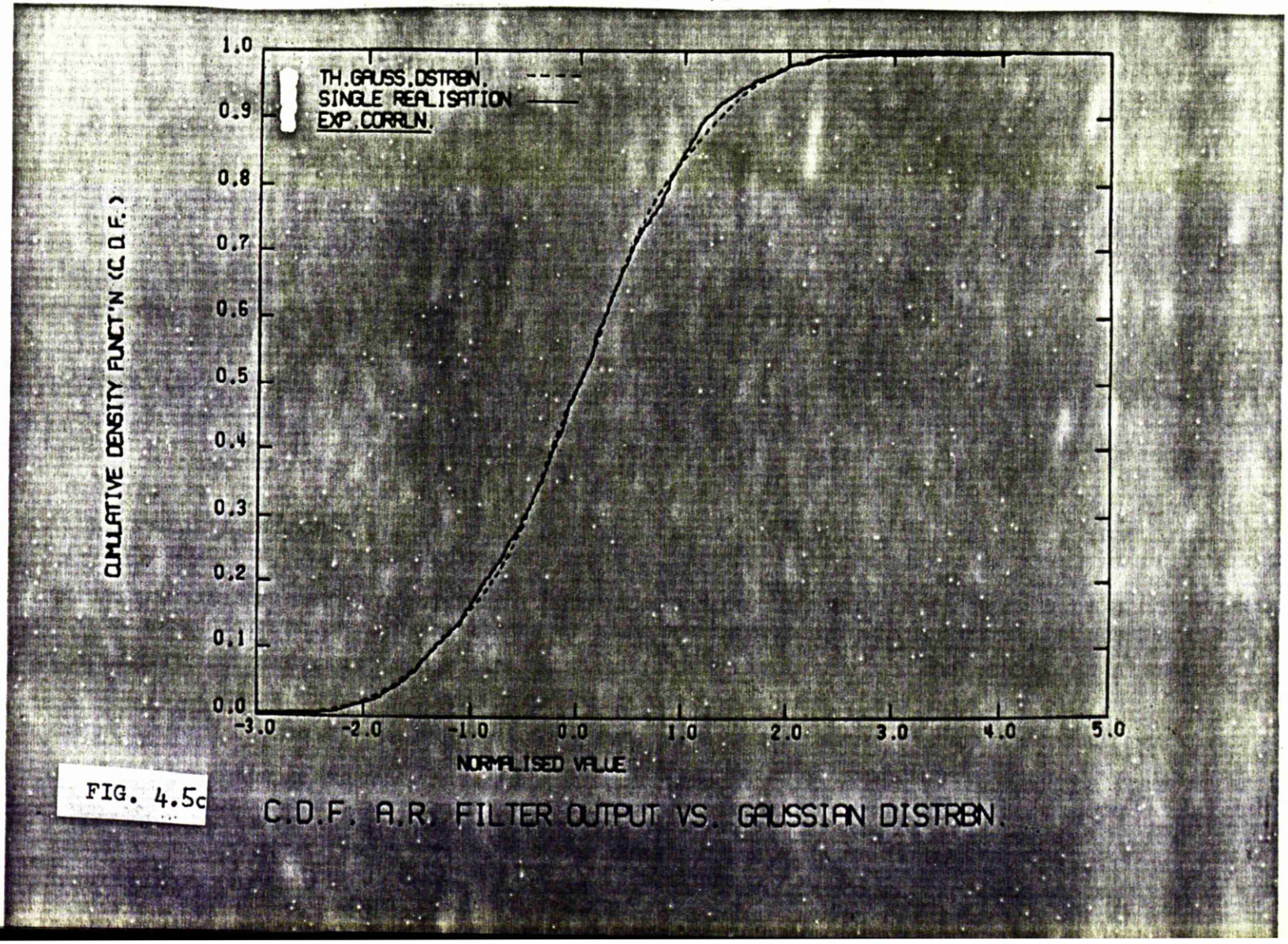


FIG. 4.5c C.D.F. A.R. FILTER OUTPUT VS. GAUSSIAN DISTRBN.

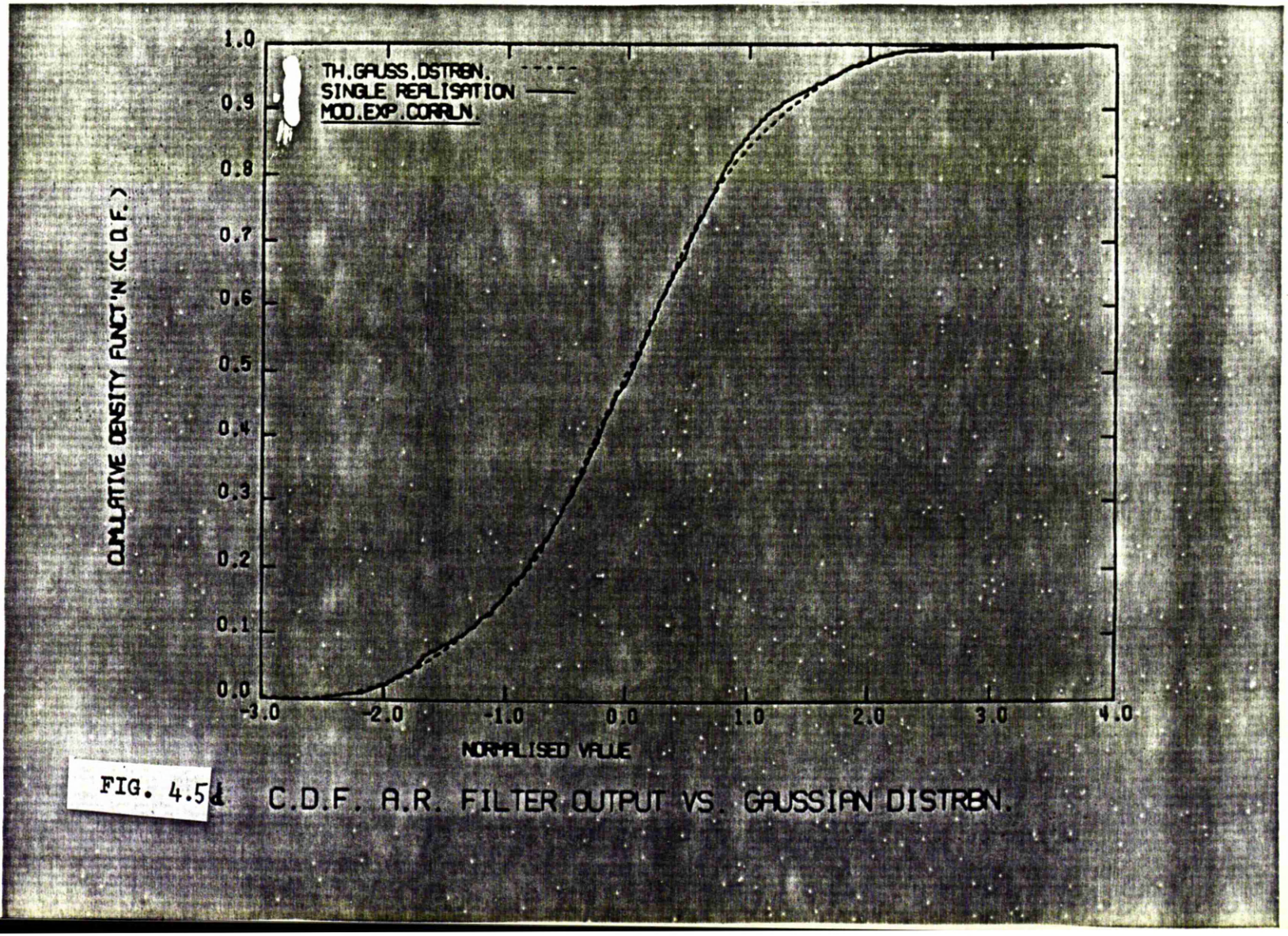


FIG. 4.5 C.D.F. A.R. FILTER OUTPUT VS. GAUSSIAN DISTRBN.

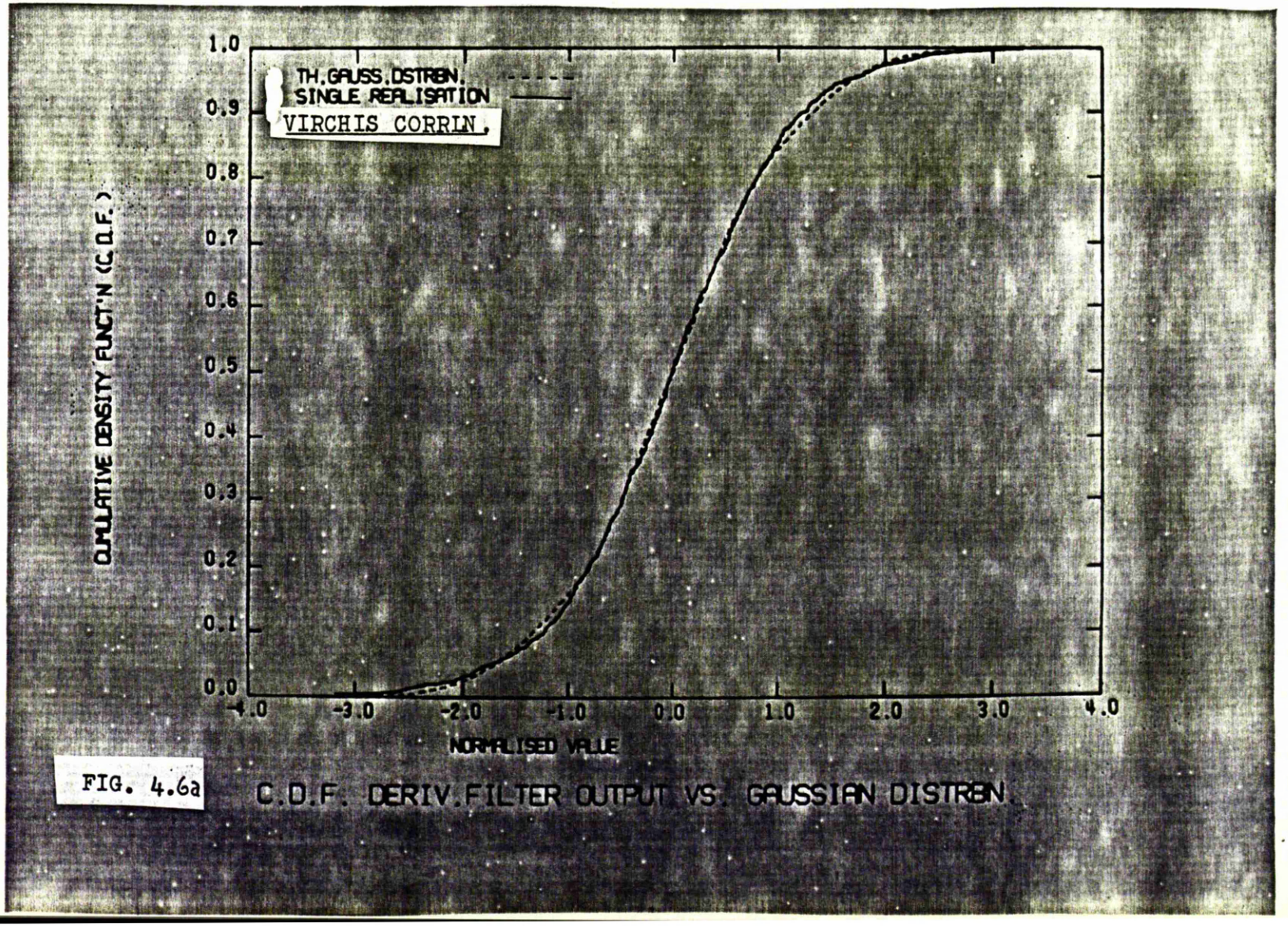


FIG. 4.6a C.D.F. DERIV. FILTER OUTPUT VS. GAUSSIAN DISTRBN.

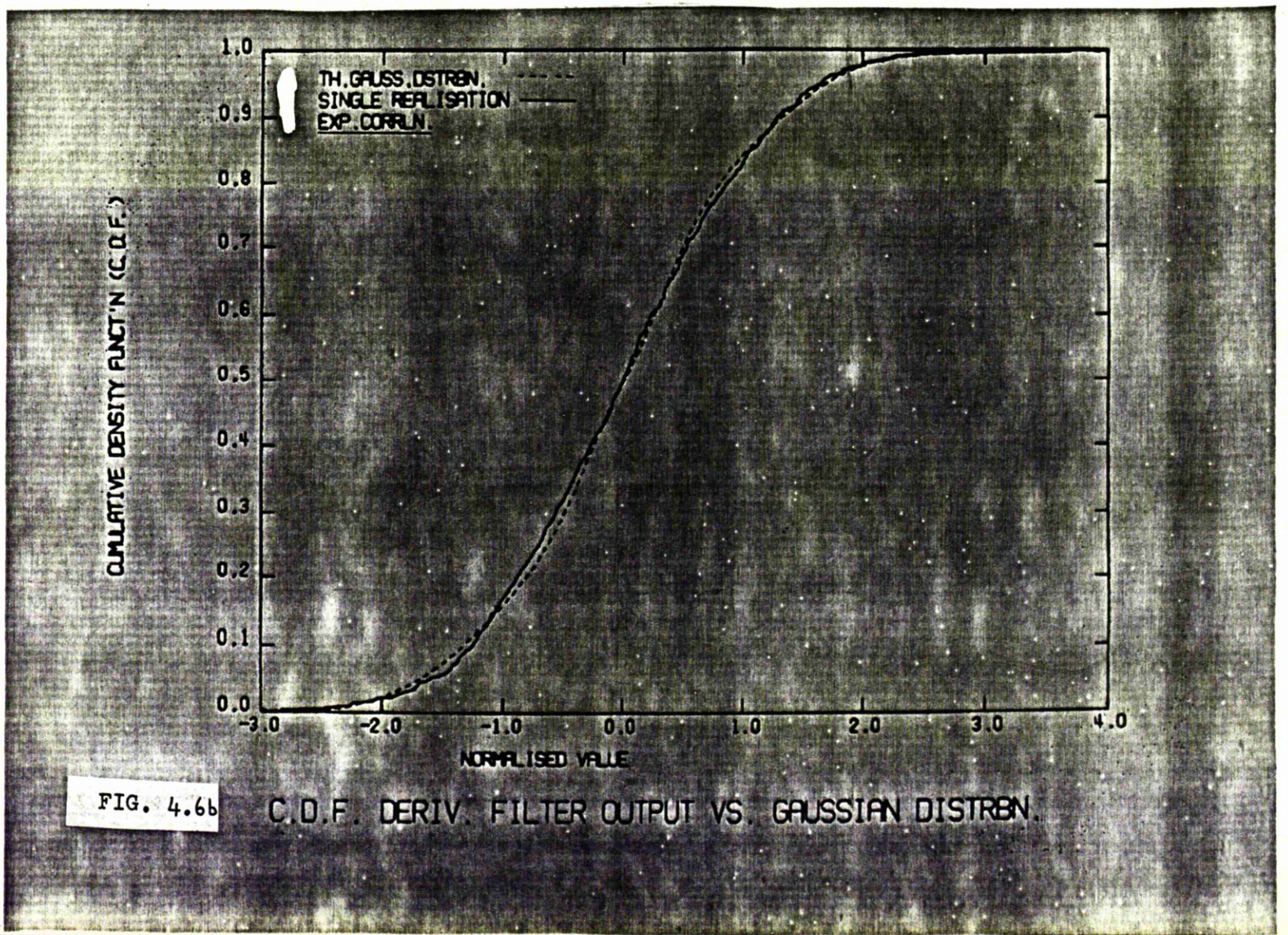


FIG. 4.6b C.D.F. DERIV. FILTER OUTPUT VS. GAUSSIAN DISTRBN.

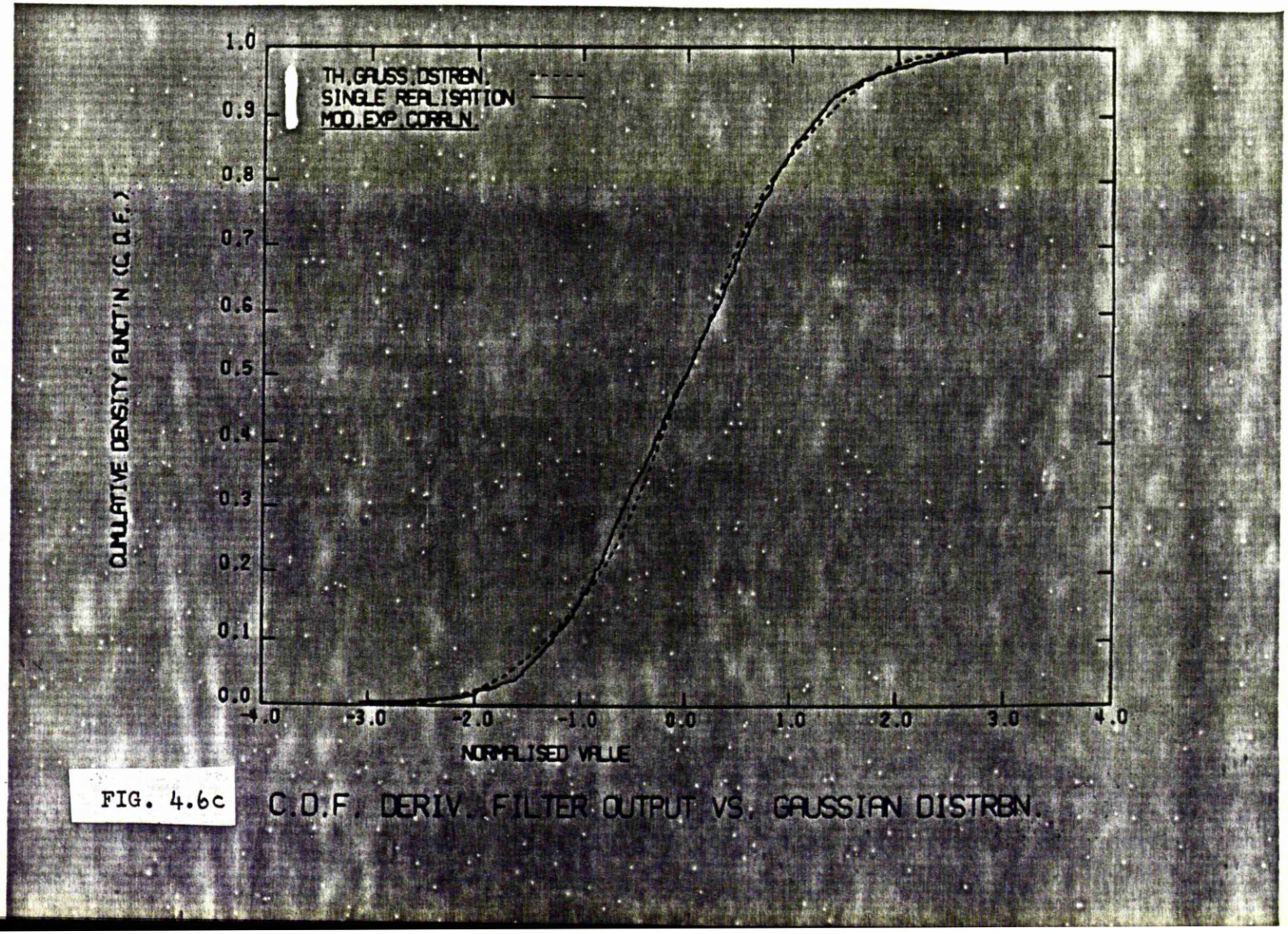


FIG. 4.6c C.D.F. DERIV. FILTER OUTPUT VS. GAUSSIAN DISTRBN.

CHAPTER V

BASIC CONCEPTS IN NUMERICAL INTEGRATION

1. Introduction

This chapter is concerned with the basic problems encountered when implementing a numerical integration scheme on a digital computer. The types of integration method discussed, are those relevant to the solution of dynamic lumped parameter vehicle models. More precisely, this involves investigation into the numerical methods for solving initial value problems in ordinary differential equation systems. The chapter covers the appropriate ground work as a necessary prerequisite to detailed discussion on specific solution methods.

2. The differential equation problem

A differential equation of order, p, takes the form

$$F(y,\dot{y},\ldots, y^{(p)};t) = 0 \qquad (2.1)$$

where $y = y(t)$, $\dot{y} = \frac{d}{dt} y(t)$, etc

In general a whole family of solutions to this problem exists. One method to uniquely fix the solution is to specify, for some arbitrary value of the independant variable, t (=a, say), the values $y(a)$, $\dot{y}(a)$, ---, $y^{(p)}(a)$. Solutions for all other instances of, t, are now uniquely defined. This is known as an initial value problem in ordinary differential equation. It is this problem which is of specific interest in this project. Henceforth in the interests of brevity the term "DE" will be used to refer to this specific initial value problem.

Normally for evaluation purposes, the DE, (2.1), is expressed in the form

$$y^{(p)} = f(y, \dot{y}, --y^{(p-1)}; t). \qquad (2.2)$$

f is frequently referred to simply as the "function".

Existence of a solution to the above problem can be established under fairly general conditions, which are not terribly meaningful for most practical applications. In practice, numerical values of solution, y, are required for some specified range of values of independant variable, t. For some special classes of problem closed form solutions are available. Many more have none. In such cases it is essential to resort to numerical methods to obtain an appropriate answer. Discrete variable methods are excellent for this purpose.

Discrete variable (or finite difference, or numerical integration) methods owe their great strength to the fact they are almost universally applicable to all DE systems. (NB In the text discrete variable methods are also referred to as integrators.) Discretisation methods do not attempt a continuous approximation to the exact solution. Instead, approximate point solutions, y_1, y_2, ---y_n, are sought at discrete increments of the independant variable, t_1, t_2,---, t_n. These incremental changes to the independant variable are often referred to as the step length, h. The step length need not be constant. Increment (or step) number, n, refers to the point solution at time instant, t_n.

Discrete variable methods use a finite serialised approximation to the true solution. In general, they can be written in the form

$$y_n = \bar{f}(y_{n-j}, ---, y_{n-i}, h(f_{n-j}, ---, f_{n-i})) \qquad (2.3)$$

where $\bar{f} \neq f$

and i and j are both integers ≥ 0.

The difference between the true solution of the DE, $y(t_n)$, and the discretised approximate solution, y_n, is known as the truncation (or discretisation) error, e_n. It is defined by

$$e_n = y(t_n) - y_n. \qquad (2.4)$$

The magnitude of this error term can be greatly reduced by re-evaluating the "function" at the current step number. Many repetitive calculations may be required, but then digital computers perform this task well.

Broadly speaking, discrete variable methods fall into two general categories; one step methods, and multistep methods. One step methods only require information from one incremental step number in order to re-evaluate at a new step. Thus, for example to solve for y_n at constant t_n, information at instant t_{n-1}, is all that is required. This is not so for multistep methods where information at instants t_{n-2}, t_{n-3}, etc. may also be required.

All truncation errors used by discrete variable methods can be assigned an order.

In one step methods, this definition is achieved by algebraically manipulating the method formula into the form of a Taylor's Series expansion, thus

$$y_n = y_{n-1} + \frac{h}{1!}\dot{y}_{n-1} + .. + \frac{h^p}{p!}y_{n-1}^{(p)} + O(h^{p+1}) \qquad (2.5)$$

By ascertaining the power, $p + 1$, at which it ceases to be a Taylor's Series, determines the order of the error. Hence, in (2.5) the order of the truncation error is said to be $p + 1$.

In multistep methods, this order is defined by expanding each term of the discretisation method as a Taylor's Series. The coefficients of the various powers of h, are then summed and the lowest power of h, at which such summutations cease to equate to zero, established. If coefficient summutations up to power p, equate to zero, then the order of truncation error is said to be $p + 1$.

With either one or multistep methods, if the truncation error is of order $p + 1$, then the discrete variable method order is said to be, p.

Digital computers calculate to a fixed number of significant figures. The size of these numbers is established in binary arithmetic and is called the word length of the computer. In computational work loss of these less significant decimal digits results in another type of error called round off.

Both discretisation and round off error can be either global or local. Local error is the error penalty incurred over one single

incremental step length, h. Global error arises from error built up over the entire step by step process.

Integration methods can be either explicit or implicit. With an explicit integration method, the quantity to be evaluated, y_n, is expressed solely in terms of known quantities at previous incremental step numbers. For explicit methods, (2.3) can be rewritten as,

$$y_n = \bar{f}(y_{n-1},--, y_{n-i}, h(f_{n-1},--, f_{n-i})) \quad (2.6)$$

A method is implicit if the quantity to be evaluated, $y_n^{(p)}$, is partly based on the previous estimate, $y_n^{(p-1)}$, such that (2.3) takes the form

$$y_n^{(p)} = \bar{f}(y_n^{(p-1)}, y_{n-1},--, y_{n-i}, h(f_n^{(p-1)}, f_{n-1},--f_{n-i})) \quad (2.7)$$

All discrete variable methods are recursive. From a set of estimated values, y_n, y_{n-1},--, y_{n-i}, another set, y_{n+k},--, y_{n-i+k}, advanced by, k steps is also estimated. In turn this set, k steps advanced, is used as the basis for estimating a second set of values, 2k steps advanced. The process is continually repeated, or recursively applied.

Recursive formulae suffer from inherent instability problems. At any step, n, the approximate solution, y_n $(=y(t_n) - e_n)$, contains a truncation error term, e_n. If e_n is recursively amplified by the discretisation process, the result is a numerical explosion known as instability. Instability totally invalidates any numerical solution.

As far as this project is concerned, there are two types of instability. There is relative instability which arises from the nature of the DE itself. However, as spring mass systems are intrinsically stable, this criterion is of no great concern. The other type is called conditional (or partial or weak) instability, although in the text it is simply referred to as instability. This instability is a function of step length, h. (There is another stability criterion, known as absolute stability, where the integrators are stable for all values of, h. However, as shown by Dahlquist (Ref. 1), this criterion is very restrictive on the

solution method accuracy and will therefore be discussed no further.)

Ideally the approximated solution, y_n, must, by repetitive calculation, tend towards the true solution $y(t_n)$ as, h, tends to zero. This property is known as convergence. It can be shown that necessary and sufficient conditions for convergence are stability and consistency. Stability, or rather instability, has been defined above. The condition of consistency, for any discrete variable method, is simply the order of the method must be greater than one. In practice it is an exceptionally easy condition to satisfy.

3. Order of integrator for vehicle dynamics

The first order DE takes the form

$$\dot{y} = f(y, t) \tag{3.1}$$

where t is the independent time variable and $\dot{y} = \frac{dy}{dt}$.

A second order DE has the general form

$$\ddot{y} = f(\dot{y}, y, t) \tag{3.2}$$

where $\ddot{y} = \frac{d^2y}{dt^2}$

In both cases the "function" f can be a linear or non-linear combination of y, $\dot{y}$, and t.

The second order formula (3.2) can be reformulated as two coupled first order DEs :-

$$\begin{aligned} \dot{y} &= v \\ \dot{v} &= f(v, y, t) \end{aligned} \tag{3.3}$$

where $v = \frac{dy}{dt}$

In vehicle dynamics, valuable information has been gained using models described by second order DEs.

A typical example is

$$m\ddot{y}(t) + c\dot{y}(t) + ky(t) = P(t). \qquad (3.4)$$

where m = mass, c = damping coefficient, k = stiffness coefficient, y(t) = displacement, and P(t) = excitation function. Rearranging this becomes,

$$\ddot{y}(t) = (P(t) - c\dot{y}(t) - ky(t)) \frac{1}{m} \qquad (3.5)$$

and this in turn can be reformulated as the two coupled first order DEs.

$$\left.\begin{aligned} \dot{y}(t) &= v(t) \\ \dot{v}(t) &= (P(t) - cv(t) - ky(t)) \frac{1}{m} \end{aligned}\right\} \qquad (3.6)$$

For higher order DEs, two schools of thought exist on how to treat them. Should the second order be treated directly, or should the DEs be reduced to a set of coupled first order formulae?

The direct approach to the solution of second order DEs is the one favoured by the structural engineers. Influencing them in this choice are two design criteria which take precedence over all others and which are superfluous to the objectives set by this project. These criteria are observation of symmetry in a matrix system of simiItaneous equations to ensure ease in numerical manipulation and the ability to handle stiff systems of differential equations without encountering instability. Hence, criteria like accuracy of solution, ability to change step length, ability to cope with non-linear system parameters fill a subserviant role.

The approach currently favoured by many numerical analysts is that which reduces the higher order systems to a set of first order DEs. In application a system of first order DEs is much more flexible. In his book, Henrici (Ref. 2) pp. 109, favours the first order approach. He argues with both theoretical and empirical validation, Chapters 4 and 6, that reduction to a first order system substantially reduces round off error, while there is little to separate the two approaches as far as truncation error is concerned. It is also true that this first order approach, when applied to a second order DE dynamic system, automatically supplies both displacement and velocity results, while

the direct approach supplies only displacements. In mechanical system design, velocity responses are also important as a source of many dynamic forces. (Velocity responses can be obtained from the direct approach, but only with the extra computational effort of subsequent numerical differentiation.)

In conclusion, because of the arguments presented, a decision was made to opt for the reduction to first order approach. Consequently in all following discussion on DEs, interest is totally concentrated on first order solution methods.

4. Factors affecting choice of numerical method

In this section desirable numerical integrator design criteria are listed and discussed. At this initial phase, selection of method type, the assessment criterion is done on a purely qualitative basis. To attempt anything more is difficult because of the highly variable nature of the "function", $f(y, t)$, where $\dot{y} = f(y, t)$, and the tremendous wealth of methods contained within any particular type.

The design criteria for the numerical integration process are as follows.

1. Computational efficiency
    - (a) in computer operations
    - (b) in computer storage
2. Accuracy
3. Computational ease of local error estimation
4. Stable numerical properties
5. Variable step length capability
6. Variable order capability
7. Self start up capability
8. Uniqueness of numerical solution

4. 1 Computational Efficiency

(a) in computer operations (computational effort)

The number of numerical operations the method performs, for a prescribed accuracy, per unit of simulation time governs the method's

numerical efficiency. Henceforth, it is often referred to as computational effort.

The four factors which influence computational effort are listed below,

i the number of incremental time steps per unit of simulation time,

ii the order of the integration method,

iii the number of "function" evaluations per incremental time step,

iv the complexity of the "function" evaluations.

Factors i, ii and iii are a function of the numerical integration method, while factor iv is simply a function of the DE system to be integrated.

Clearly, for large systems of coupled DEs "function" evaluation is a major source of numerical inefficiency. For a single DE this may not be the case.

(b) in computer storage

This aspect of efficiency is simply defined as the amount of extra information the computer must store at each time increment to permit the method to work. This is dependant on the method of integration and the size of the system of DEs.

4. 2 Accuracy

This is a prerequisite for any numerical solution. It should always be considered in conjunction with efficiency since it is always possible to achieve greater accuracy at the expense of greatly increased computational effort. Primarily, it is a function of the incremental step length and the order of the integration formula.

4. 3 Computational ease of local error estimation

The numerical methods used for solving problems in DEs are processes regressing from sets of initial conditions and influenced by

sets of forcing functions. As such it is an easy matter for the numerical solutions to deviate significantly from true. A means of checking for this and thereby enabling corrective treatment to be implemented, is supplied by the local error estimate.

Clearly, it is desirable for this error estimation to be achieved with minimal computational effort, as estimates of error should be obtained at every time increment.

4. 4 Stable numerical properties

Stability is a constraint which all recursive type difference equations must meet to ensure a meaningful and bounded solution. Instability manifests itself in what could be described as a numerical explosion after a relatively few number of incremental time steps. (For fuller explanation see Section 6)

4. 5 Variable step length capability

This feature is often included in a numerical integrator. Should the local error fail to meet prescribed tolerance limits as the integrator time marches, it is usually desirable to make running adjustments to the incremental step length, h, to reduce the magnitude of this error. However passive error monitoring is adequate for the objectives of this project.

This ease of step length alteration during the time marching process is important in the non-stationary problem for quite a different reason. With the non-stationary problem, the marching process is accomplished in the space domain and not the time domain. (See Chapter IX). Under such circumstances the velocity terms of the second order DEs are broken into the product $\frac{dy}{dx} \cdot \frac{dx}{dt}$ where y is vertical displacement, x is the distance travelled along the road profile, and, t is the independant time variable. The $\frac{dx}{dt}$ profile velocity terms are grouped with the physical parameters of the system of dynamic equations. This separation is equivalent to altering the step length relative to the time constants of the differential equation system. Ability to cope with step length changes during the marching process consequently becomes important

because of this time constant alteration.

This step length alteration ability can also prove a useful self starting device. (See subsection 4. 7)

4. 6 Variable order capability

As with step length alteration this ability to change order can be used as a method of error control during the time marching process. As this ability is superfluous to requirements the usefulness of this device comes as a self starting mechanism. (see subsection 4. 7)

4. 7 Self start up capability

At the start of the time marching process it is necessary to define the initial conditions that exist in the physical system. With some numerical integrators this is all the information required. Such methods are truly self starting. However by taking into account the variable step length or variable order ability of some integrator methods, it is possible to greatly increase the range of integration methods which can be self starting. Such pseudo self start methods either begin with a very small step length to keep initial errors down at low magnitude or, alternatively commence with a low integration order and then increase the order to establish all necessary information for the higher order integrator formula.

4. 8 Uniqueness of the numerical solution

The uniqueness of the numerical solution is necessary to ensure convergence. This is a particular problem in non-linear DEs. Although in the non-stationary problem (because of the space domain approach) the parameters are changing as the process time marches, the DEs are in fact quasi linear because of the parameters independence of the system's responses.

5. Types of solution methods

Attempts at classifying methods into either one step or multistep methods become somewhat clouded when particular integration

method types are considered. There are many instances of method types which span the boundary between those two classifications, with some methods residing firmly in the multistep class, while others of the same type are entrenched in the one step method class.

In the following subsections the various method types are briefly discussed.

5. 1 Taylor's Series Solution

The Taylor's Series method is not a serious contender in the choice of integration formula. However, it is of great theoretical importance. Consequently it is an excellent starting point in the consideration of method class.

The Taylor's Series solution takes the form

$$y_m = y_{m-1} + \frac{h}{1!}\dot{y}_{m-1} + \frac{h^2}{2!}\ddot{y}_{m-1} + \frac{h^3}{3!}\dddot{y}_{m-1} \tag{5.1}$$

where 'y^N_{m-i}' is the 'Nth' derivative with respect to time at time 't_{m-i}', and 'h' is the time step increment $(t_m - t_{m-1})$.

Consider the, by now familiar, DE

$$\dot{y} = f(y, t) \tag{5.2}$$

Differentiate to obtain

$$\ddot{y} = f^t + f\, f^y \tag{5.3}$$

where $f^i = \frac{\delta f}{\delta i}$.

At time instant, t_{m-1}, (5.2) and (5.3) yield

$$\dot{y}_{m-1} = f_{m-1} \tag{5.4}$$

$$\ddot{y}_{m-1} = f^t_{m-1} + f_{m-1}\, f^y_{m-1} \tag{5.5}$$

respectively.

Substitute in (5.1) to obtain

$$y_m = y_{m-1} + h\left\{ f_{m-1} + \frac{h}{2}\left(f^t_{m-1} + f_{m-1}\, f^y_{m-1}\right)\right\} + O(h^3) \tag{5.6}$$

where $O(h^3)$ = truncation error term of highest order.

Consequently it is possible to obtain a solution from the Taylor's Series

provided the higher derivatives of the original DE (5.2) can be determined. If 'n' higher derivatives had been considered the truncation error would have been, $O(h^{n+1})$. Such solution methods are of little practical value because of the computational difficulty of evaluating these higher derivatives.

The real importance of the Taylor's Series is in establishing the truncation error in other more readily usable integration methods.

5.2 Runge Kutta (RK) Methods

The essence of Runge Kutta (RK) methods are easily summarised. Such methods comprise of the following two basic operations.

(1) Successive "function" evaluations are used to iteratively estimate the slope, in the following manner,

$$\begin{aligned} f_m^{(o)} &= f(y_{m-1},\ t_{m-1}) \\ f_m^{(1)} &= f(y_{m-1} + a_1\, h\, f_m^{o},\ t_{m-1} + a_2\, h) \\ &\vdots \\ f_m^{(n)} &= f(y_{m-1} + a_{2n-1}\, h\, f_m^{n-1},\ t_{m-1} + a_n h) \end{aligned} \qquad (5.7)$$

(2) The 'n' function evaluations are then "averaged" and y_m evaluated, to yield

$$\begin{aligned} y_m &= y_{m-1} + b\, h\, (b_o\, f_m^{(o)} + b_1\, f_m^{(1)} + b_2\, f_m^{(2)} + \cdots + b_n\, f_m^{(n-1)}) \\ &= y_{m-1} + h\, F(y_m,\ t_m) \end{aligned} \qquad (5.8)$$

It is obvious that the general RK method requires no information other than, y_{m-1}, to proceed time marching. It is a true self starting method.

Changing incremental step length, h, in the RK method is a trivial operation.

The main disadvantage of RK methods lies in their low computational efficiency, especially with higher order formulae. To reduce the truncation error by one order of magnitude it is also necessary to increase the number of "function" evaluations by one. During a time marching process many incremental time steps are calculated and the extra computational effort of such additional "function" evaluations becomes very significant.

5.3 Predictor-Corrector (PC) Methods

All linear multistep methods take the form

$$\sum_{i=0}^{k} a_i y_n = h \sum_{i=0}^{k} b_i f_{n-i} \qquad (5.9)$$

where a_i and b_i are constant coefficients.

(5.9) is a k - step method. The term linear follows the fact that the "f"s are entered as a linear series and not, as might be supposed, because the method is restricted to linear "f"s.

Most predictor corrector (PC) methods come under the classification of linear multistep methods. There are, however, important exceptions to this. PC methods do exist which are not in the true sense multistep but can in fact be shown to be equivalent to and to exhibit many properties of multistep methods.

Linear multistep methods can be established as follows,
Take the DE

$$\dot{y} = f(y, t) \qquad (5.10)$$

and integrate between t_{m-1} and t_m, so that

$$y_m - y_{m-1} = \int_{t_{m-1}}^{t_m} f(y, t)\, dt \qquad (5.11)$$

while replacing $f(y, t)$ by an interpolating polynomial. Such methods are integration based.

(It is possible to replace $\dot{y}$ on the left hand side of (5.10) by an interpolating polynomial of y which is then differentiated. These methods are in general unstable.)

Predictor corrector formulae owe their basis to linear multistep methods. In practice it is normally convenient to have an explicit Predictor as an initial estimate of y_m which is then improved upon by an implicit Corrector.

In essence, Predictor Corrector methods comprise of the following operations.

(1) The P step

An estimation is made using the explicit Predictor formula as to the value of 'y' at 't_m'.

$$y_m^{(0)} = \sum_{i=1}^{p} \left(a_i \, y_{m-i} + h \, b_i \, \dot{y}_{m-i} \right) \qquad (5.12)$$

where a_i and b_i are constants.

Neither y_{m-i} nor f_{m-i} are calculated at this stage.

They are in fact stored and carried forward from previous time step increments.

(2) The E step

A function evaluation is next performed by making use of (5.10) and the last estimated value of $y_m^{(j-1)}$.

$$\dot{y}_m^{(j)} = f(y_m^{(j-1)}, t_m) \qquad (5.13)$$

(3) The C step

By means of the 'j^{th}' function evaluation the estimated value of y is now updated, by means of the Corrector formula, to yield

$$y_m^{(j+1)} = \sum_{i=1}^{q} \left(a_i \, y_{m-i} + h \, b_i \, \dot{y}_{m-i} \right) + a_o \, y_m^{(j)} + h \, b_o \, \dot{y}_m^{(j)} \qquad (5.14)$$

It should be noted that the order of both Predictor and Corrector stages is quite independant of the number of function evaluations.

The E step and the C step can be iterated until the desired degree of convergence is achieved, although clearly it is undesirable for efficiency reasons to have too many iterations.

Predictor Correctors have two possible modes of operation.

(1) the $P\,(E\,C)^k$ mode

(2) the $P\,(E\,C)^k\,E$

where $(E\,C)^k$ is an evaluation (E) step followed by a correction (C) step, with the sequence being repeated 'k' times. The stability properties of the two modes are quite different (see Chase, P.E., Ref. 3). This aspect is discussed in some detail later.

PC methods require a considerable amount of information to proceed time marching. They are consequently not self-starting. However once the process is started maximum usage is made of this previously computed information to make such PC methods computationally efficient, and unlike RK methods the order of truncation error is independant of the number of function evaluations per time step.

In the previously derived form PC methods are not amenable to step length change. Fortunately PC methods do exist (discussed in detail later) whereby this back value information is not stored in this back value (multistep) form but in terms of higher derivatives. In this form, it lends itself to step length changes. This also means that PC formulae are amenable to pseudo self start procedures. The initial step length, h_I, is so reduced to keep the initial truncation error, $O(h_I^{p+1})$, well below the error, $O(h^{p+1})$ of a typical step, h.

If this latter type of transformed PC method is adopted, the method becomes exceedingly desirable provided the self starting mechanism is not too frequently implemented.

5.4 Other less common types

Less well known types are considered briefly. Some can be quickly eliminated as being unsuitable for the purpose in hand. Others are dropped because of scanty knowledge of their numerical behaviour, there being no apparent overriding advantage for their implementation.

5.4.1 Block type methods

These methods can be either explicit or implicit and take the general form

$$\sum_{i=e}^{p} a_i \, y_{m+i} = \sum_{j=o}^{q} b_j \, y_{m-j} + h \sum_{k=-p}^{q} c_k \, f_{m-k}$$

where $e = 1, 2, 3, \text{---} p$, and a_i, b_i, c_i, are scaler coefficients. Next step the method advances 'p' increments so that the values y_{m+p+i}, for $i = 1, 2, \text{---} p$, are calculated. The f_{m-k} terms can be replaced by an interpolating polynomial. Clearly it is impossible to alter the step length at every step length increment, h. This can only be done in blocks of 'p'. This class must therefore be ruled out.

5.4.2 Hybrid type methods

Such methods combine the features of PC and RK methods. They take the form

$$y_{m+1} = \sum_{i=o}^{p} \{ a_i \, y_{m-i} + h \, b_i \, f_{m-i} \} + h \, F(y_m, t_m)$$

The terms within $\{ \}$ are the explicit Predictor part and F is the explicit RK function evaluation part.

Unfortunately such methods are not self starting, do not lend themselves to step length change, and the RK portion requires many function evaluations. They may have future potential especially if a step length alteration capability is developed. At the present state of development this class must be discounted.

5.4.3 Multistep Multiderivative type methods

Multistep multiderivative methods have much in common with predictor corrector methods. Unlike predictor corrector methods which are based on (4.3.2)

$$y_m - y_{m-1} = \int_{t_{m-1}}^{t_m} f(y, t) \, dt$$

the multiderivative method has its basis extended to include higher derivatives of $f(y,t)$.

The method takes the general form

$$y_m = \sum_{i=o}^{p} a_i \, y_{m-i} + \sum_{k=1}^{r} h^k \sum_{i=o}^{p} b_{ki} \frac{d^k}{dt^k}(y_{m-i})$$

where t is the independant variable
and r is the order of higher derivatives included.

These methods can be used in a predictor corrector form. It is not included in Section 4.3.It is at a rudimentary phase of development, although a few applications with stiff systems, not relevant to this project have been attempted - Liniger and Willoughby Ref. 4 and Enright Ref. 5.

Efficient application of this method requires easy evaluation of the higher derivatives, $\frac{d^k (y_{m-i})}{dt^k}$. Such derivative evaluations can often prove difficult.

Since this class has no apparent overriding advantages, it was thought best to reject this method at the preliminary phase for the reasons discussed in the last two paragraphs.

5.4.4. Recursive Convolution Integral type methods

Such methods are based on the Convolution integral

$$y(t) = \int_0^t P(t) \; H(t-T) \, dT$$

This formula is often used in dynamics, where $H(t)$ is the impulsive receptance of the system,

$P(t)$ is the input disturbance
$y(t)$ is the system response.

To increase the computational efficiency the Convolution integral is discretised and expressed in a recursive form. (See Trauboth Ref. 6).

This type of solution method is based on the superposition principle and as such is only applicable to linear elastic systems.

Although frequently used in structural dynamics, this restriction to linear systems makes it rather inflexible for use as a general method of handling the non-stationary (and frequently non-linear) vehicle suspension problem. It must therefore be ruled out.

6. Existence and Uniqueness of the Solution

Most types of numerical solution to the initial value problem (3.1)

$$\dot{y} = f(y, t)$$

can be classed under the general p step method which takes the form

$$\sum_{i=o}^{p} a_i \, y_{m-i}^{(t)} + h \, G(y_m^{(t)}, y_{m-1}^{(t)}, .., y_{m-p}^{(t)}, t, h) = 0 \qquad (6.1)$$

It is necessary to establish if a unique solution to the numerical problem exists. To do this it is best to break the problem into two halves and determine, in turn, if there is a unique solution to the differential equation (3.1) and having established this determine if a unique numerical solution to the difference problem (6.1) exists.

That a unique solution to the initial value problem (3.1) exists is proved by Henrici (Ref. 2) pp 15 Section 1.2. The requirements are merely stated in the following paragraphs.

Conditions for uniqueness of (3.1)

For an initial value problem, (3.1), if the conditions:

(1) that $f(y, t)$ is defined and continuous in the region $a \leq t \leq b$, and $-\infty < y < \infty$ for any finite a, b

(2) that a constant, K, exists within the bound $a \leq t \leq b$ such that, for any y and y*

$$|f(y^*,t) - f(y,t)| \leq K \, |y^* - y|$$

— a Lipschitz condition

then there exists only one function $y^{(t)}$ such that

(1) $y(t)$ is continuous and differential for $a \leq t \leq b$

(2) $\dot{y}(t) = f(y(t), t)$, for $a \leq t \leq b$

(3) $y(a) = q$, where q is a constant.

As for the numerical integration method (6.1), a unique solution obviously exists if the integration formula is explicit or if the function to be evaluated has a linear nature. For an implicit formula, (6.1) can be written in the form

$$y_m = F(y_{m-i}) \tag{6.2}$$

or rewriting in the iterative form

$$y_m^{(p+1)} = F(y_m^{(p)}) \tag{6.3}$$

for $i = 0, 1, 2$ ---, and where the superscript (p) refers to the p^{th} iterative approximation.

The conditions for the existence of a unique solution are stated in the following paragraph. (For proof see Henrici (Ref. 2) pp 216 - 217).

Conditions for uniqueness of the implicit method (6.3)

For a numerical initial value problem of the form described by (6.3), if the conditions that

(1) $F(y_{m-i})$ is defined and continuous within the bound $-\infty < y < \infty$

(2) a constant K exists such that for any y and y*

$$\left|F(y^*_{m-i}) - F(y_{m-i})\right| \leq K \left|y^*_m - y_m\right|$$

where $0 \leq K \leq 1$

then there exists a unique solution y_{m-i} which for any arbitrary $y^{(0)}$ converges to y and for which the following inequality holds -

$$\left|y - y^{(e)}\right| \leq \frac{K}{1-K} \left|y^{(e)} - y^{(e-1)}\right| \leq \frac{K^e}{1-K} \left| y^{(1)} - y^{(0)}\right|$$

The uniqueness of the integration method is dependant on the conditions for the uniqueness of the DE being satisfied. RK methods which are normally explicit possess a unique solution, providing the DE solution is unique. PC methods, because of their implicit nature, must also satisfy the conditions of a unique integration method.

7. Convergence, consistency and stability

Clearly in order to obtain meaningful answers from any recursive integration method, it is vital that at any incremental step number, n, the approximated solution, y_n, is a good estimate of the true solution, $y(t_n)$. To achieve this aim, the integration method must meet the condition of convergence.

In future chapters, interest is concentrated on linear multistep methods and consequently the following discussion is so orientated.

From the linear multistep method (5.9), the characteristic polynomial equation

$$\sum_{i=0}^{k} (a_i - hb_i)\, r^i = 0 \qquad (7.1)$$

can be obtained.

For convenience, (7.1) can be rewritten as

$$\rho(r) - h\ \sigma(r) = 0 \qquad (7.2)$$

where r_i are the k distinct roots of the polynomial.

Necessary conditions for convergence are stability and consistency (for proof see Henrici (Ref. 2) pp 218-220 and pp 224-225 respectively). Translating into terms more immediately relevant to linear multistep methods, the method is stable when all the roots, r_i for $i = 0, 1, \ldots, k$ of the characteristic equation satisfy the condition

$$|r_i| \leq 1 \quad \text{for all} \quad i.$$

Consistency is satisfied if the order of the method is greater than one, ie

$$\begin{aligned} \text{if} \quad & \rho(1) = 0 \\ \text{and if} \quad & \acute{\rho}(1) = \sigma(1) \\ \text{where} \quad & \acute{\rho}(r) = \frac{d}{dr}\rho(r). \end{aligned}$$

It can be demonstrated (see Henrici (Ref. 2) pp 244-246) that stability and consistency are also sufficient conditions for convergence.

These criteria for convergence are in part dependant on the coefficients of the linear multistep method, and partly on the step

length, h. This reliance on the coefficients of individual methods effectively means that the convergence criteria can only be considered when individual methods are considered, It cannot therefore be used as a general decision criterion for selecting a method type.

8. Methods of Local Truncation Error Estimation

Although the order, p, of error term, $O(h)^p$, can be established for any numerical method of solution in the initial value DE problem, some means of estimating the actual magnitude of the local truncation error is required to check for convergence as the integration proceeds in time. There are various methods of calculating this error. The methods are dependant on the type of integration method used.

In the following subsections the main types are dealt with in turn.

8. 1 Error Estimation in Runge Kutta Methods

These estimation techniques are also applicable to other solution methods. Normally with multistep methods, estimation routines, with considerably increased computational efficiency can be chosen in preference. Unfortunately, this is not true of RK methods.

There are several methods which can be used. All require two independant estimations of the solution at the current time of interest, t_m. These two independant solutions can be achieved in either of two ways: (a) by the implementation of the integration formula using two different step lengths, and (b) by the implementation of two integration formulae of different type or order. (In case (b) the different order integration formulae need not necessarily be of the same type.)

As a result of this requirement of at least two independant evaluations per step, error checks incur a severe penalty in computational efficiency.

Method (a)

Consider for example, a 4^{th} order method.

Let $y(t_m)$ be the true solution at $t = t_o+mh$ and let y_m be

the approximated solution with a step length of h,
then

$$y(t_m) = y_m + O(h)^5 + ... \qquad (8.1)$$

where $O(h)^5$ is the most significant error term.

Recompute the solution with a step length of h/2, where $\bar{y}_m$ is not the approximated solution, to obtain

$$y(t_m) = \bar{y}_m + O(h/2)^5 + ...$$

$$= \bar{y}_m + O/32(h)^5 \qquad (8.2)$$

Eliminate $y(t_m)$ from (8.1) and (8.2), and an estimate of the most significant local truncation error term can be obtained, thus

$$O(h)^5 = 32/31\ (\bar{y}_m - y_m) \qquad (8.3)$$

To work well, this error estimate (8.3) must assume a reasonably constant fifth derivative.

This method of error estimation can be implemented in two ways; with the h/2 as the basic step length and error estimates made every alternate h/2 step, or with the h regarded as the basic step and two extra formula implementations of step length h/2 required to estimate the error at every step, h.

Method (b)

Consider for example, 4^{th} and 5^{th} order methods, not necessarily the same method, or even another RK method.

Once again, let $y(t_m)$ be the true solution at $t = t_0 + mh$, and $y_m^{(4)}$ be the approximated fourth order solution such that

$$y(t_m) = y_m^{(4)} + O(h)^5 + O(h)^6 + ... \qquad (8.4)$$

where $O(h)^5$ is the most significant error term,
$O(h)^6$ the second most significant, etc.

For the 5th order method, the error is given by

$$y(t_m) = y_m^{(5)} + O(h)^6 + ... \qquad (8.5)$$

where $O(h)^6$ is the most significant error term.

Eliminate $y(t_m)$ from (8.4) and (8.5), and hence

$$O(h)^5 = y_m^{(5)} - y_m^{(4)} \tag{8.6}$$

An estimate of the most significant error term in the 4^{th} order method is obtained.

8.2 Error estimation in Predictor Corrector Methods

For methods of the PC type, one method of error estimation is almost universally accepted because of its simplicity and efficiency. It relies on the Predictor and the Corrector being of the same order, q.

Let $y(t_m)$ be the exact solution at time, $t = t_0 + mh$ (where t_0 is the initial time), and let $y_m^{(p)}$ and $y_m^{(c)}$ be the Predictor and Corrector estimates of $y(t_m)$, respectively.

Then, $$y(t_m) = y_m^{(p)} + ah^q \frac{d^q f}{dt^q}(T_1) \tag{8.7}$$

and $$y(t_m) = y_m^{(c)} + bh^q \frac{d^q f}{dt^q}(T_2) \tag{8.8}$$

where $t_{m-q} \leq T_1 \leq t_{m-1}$

and $t_{m-q+k} \leq T_2 \leq t_{m-1+k}$.

$y(t_m)$ is eliminated to yield the approximate relation

$$y_m^{(c)} - y_m^{(p)} = (a - b)\, h^q \frac{d^q f}{dt^q}(T) \tag{8.9}$$

where $t_{m-q} \leq T \leq t_{m-1+k}$

(It is assumed $\frac{d^q f}{dt^q}(T)$ is approximately constant for the range of t defined)

(8.9) can be rewritten as

$$h^q \frac{d^q f}{dt^q}(T) = \frac{y_m^{(c)} - y_m^{(p)}}{(a - b)} \tag{8.10}$$

Both (a) and (b) are known constants. They were established when Predictor and Corrector were in turn equivalenced with the Taylor's Series.

From (8.10) the Predictor and Corrector error can be approximated respectively as follows.

$$\text{Predictor error} \simeq a\, h^q \frac{d^q f}{dt^q}(T) \simeq \frac{a}{a-b}\left(y_m^{(c)} - y_m^{(p)}\right) \qquad (8.11)$$

$$\text{Corrector error} \simeq b\, h^q \frac{d^q f}{dt^q}(T) \simeq \frac{b}{a-b}\left(y_m^{(c)} - y_m^{(p)}\right) \qquad (8.12)$$

9. Variable Order Numerical Integration Methods

Any method of the RK or the PC type can incorporate the added flexibility of dynamic order selection. This is a desirable characteristic if higher derivative discontinuities exist. Under such circumstances if the numerical integration order is not reduced, the meaningless higher order differences introduce erratic behaviour into the solution.

The computational complexity in implementing variable order is considerable. Both RK and PC methods require comparisons to be drawn between two formulae of different order, comparing errors of the P^{th} and the $(P + 1)^{th}$ order. If the order of the numerical method becomes too high, such that discontinuities in the higher derivatives of the function to be integrated are encountered, then the erratic behaviour of the numerical solution would render the comparison between the P^{th} and $(P + 1)^{th}$ order formulae meaningless. Although meaningless the comparison test would be implemented and found wanting. This in turn would influence the decision process to opt for a yet higher order, and so on ad infinitum. Some means of checking this order escalation is necessary. No totally satisfactory method of doing so is available at present. Those that exist are computationally inefficient.

High order formulae have a narrow stability margin compared with low order formulae. Consequently every time the order of the method is increased, the stability margin is reduced. Reduction in the stability bound should be avoided especially in problems where the conventional eigenvalue stability analysis is inapplicable.

10. Conclusion

Predictor Corrector methods appear to exhibit the best overall properties. They are however subject to certain constraints which must be considered when individual methods are dealt with.

For various reasons (see subsection 4.4) the less common methods were dropped at an early stage in the decision process. The comparisons drawn below are between Runge Kutta (RK) and the Predictor Corrector (PC) type. The degree to which both methods meet the design criteria is discussed in turn.

1) Computational Efficiency

(a) in arithmetic operations (computational effort)

The order of explicit RK methods is dependant on the number of function evaluations. It realises a poor efficiency for a desired accuracy. This is particularly true in cases where function evaluations are complex.

Implicit RK methods suffer from added arithmetic complexity because of their iterative nature.

Predictor Corrector methods require only a few function evaluations. The number of evaluations required is independant of the order of the process. Increasing the order by one, with a corresponding increase in accuracy, simply involves a few extra arithmetic operations.

(b) in computer storage

RK methods require a minimal amount of storage as back value information. Other than y_{m-1} for a current time t_m, no additional information is needed by the method.

PC methods require information to be stored about their back values, whether stored as back values or in a higher derivative form.

2) Accuracy

Both RK and PC methods can attain, for a given order n, a truncation error of magnitude $O(h^{n+1})$. It is only when qualified by computational

efficiency that the superiority of PC methods becomes evident.

3) Computational ease of local error estimation

In RK methods, all the techniques of error estimation, whether through step length halving or through a secondary integration scheme, approximately doubles the computational complexity. With the PC method, such error estimation is trivial provided the predictor and the corrector are of identical order.

4) Stable mumerical properties

All methods irrespective of class exhibit stability properties which are dependant on the parameters of the integration routine, the step length, h, and the function to be integrated. No overall class properties exist, and this criterion cannot influence the choice of method type.

5) Variable step length capability

This is a trivial operation with RK methods as information about prior time steps is not required, and the alterations are simply achieved by altering the value of h in the formula.

In the case of PCs, step length alteration proves difficult if the additional information required by the method is stored in back value (multistep) format, since error results when h is changed. Some PC formulae store the additional information as combinations of higher derivatives. In this form it is amenable to changes in step length, and as with RK methods this is achieved simply by altering h in the formula.

6) Variable order capability

In both classes of method, the decision processes involved in automatic optimisation of the integration order add considerably to the algorithm's complexity. PCs have the added difficulty that the

additional information may not be in a form condusive to order change. This would be the case if the additional information was stored in a "combination of higher derivatives" form.

Since the order of the process to be integrated is anticipated not to vary, and also because of the complexities of order determination, a variable order capability was considered irrelevant to the needs of this project.

7) Self start up capability

RK methods do not require information about back values. They are therefore self-starting.

This is not true of PC methods where pseudo start up can be implemented in one of 3 ways, as follows:-

i. by use of a very small initial step length, effectively keeping the truncation error below that of a typical step length,

ii. by starting with a low order PC method and building the order up as the steps progress and more information becomes available until the desired order is reached,

iii. by initial use of an RK method to establish sufficient additional information to allow the PC method to proceed.

(i) is only practicable in methods amenable to step length change, and (ii) is only of use in those amenable to order change. Both (ii) and (iii) are difficult to program.

For PC methods start up is a relatively complicated operation and limits the viability of PC methods to cases where infrequent restarts are required during the period of the integration process.

8) Uniqueness of the numerical solution

Before considering the uniqueness of the integration method it is necessary to establish the uniqueness of the function to be evaluated.

Once accomplished the integration methods can then be considered.

If the integration method is explicit as in conventional RK formulae, then the uniqueness of the method presents no problem. However if it is implicit, as PC methods are, then it can also possess a unique solution, subject to some additional constraints.

REFERENCES - CHAPTER V

Ref. 1 Dahlquist, G.G. (1963) "A special stability problem for linear multistep methods". BIT. 3, 27 - 43.

Ref. 2 Henrici, P. (1962) "Discrete variable methods in ordinary differential equations". pub. Wiley.

Ref. 3 Chase, P.E. (1962) "Stability properties of predictor-corrector methods for ordinary differential equations". J. Assoc. Comput. Mach. 9, 457 - 468.

Ref. 4 Liniger, W., and Willoughby, R.A. (1970) "Efficient integration methods for stiff systems of ordinary differential equations". SIAM. J. Numer. Anal. 7, 47 - 66.

Ref. 5 Enright, W.H. (1974) "Second derivative multistep methods for stiff ordinary differential equations". SIAM. J. Numer. Anal. 11, 321 - 331.

Ref. 6 Trauboth, H.H. (1969) "Recursive formulae for the evaluation of the Convolution Integral". J. Assoc. Comput. Mach. 16, 63 - 72.

Additional General References

Lambert, J.D. (1973) "Computational methods in ordinary differential equations". pub. Wiley.

Collatz, L. (1960) "Numerical treatment of differential equations". 3rd ed. pub. Springer.

CHAPTER VI

CHOICE OF A PREDICTOR CORRECTOR (PC) METHOD

1. Introduction

In the previous chapter the justifications were presented for opting for the PC type method. This chapter attempts to do likewise for the chosen PC method.

As the range of PC methods is extensive some means had to be found to reduce the range to a few best methods, and then by a closer examination decide on the one which best suits the vehicle simulation design requirements.

The chapter opens by discussing the literature available on PC methods. From that together with the results of the few comparative tests available the chapter forms a short list. The equivalent Adams - Bashford - Moulton linear multistep basis of the short listed methods is derived. The methods are then briefly stated and a discussion follows on their relative capabilities in meeting the vehicle simulator requirements.

2. Discussion on and preparation of a Short List

In the previous chapter, the conclusion was drawn that methods based on linear multistep methods of the PC class best met the desired integrator's design requirements provided formulations are chosen which overcome the inherent shortcomings of linear multistep methods. By surveillance of the literature the selection of an optimum method is attempted.

Because of the extensive nature of the literature it was decided to limit the choice to methods which have been fully developed and numerically tested. Furthermore it was decided to concentrate interest on methods equivalent to linear multistep methods with correctors of the Adams - Moulton form. Adams - Moulton correctors were found by Hull and Newbury (Ref. 1), to exhibit error tolerances of about 1/5th the magnitude of any of the others tested. A result which

held true for up to eight corrector iterations. The predictor is simply a tool for making a good initial estimate.

Articles containing results of comparative tests on the various methods are few. Most are concerned with comparing the author's own with one other method. The paper by Hull, Enright, Fellen, Sedgwick (Ref. 2) is the only true comparative test of non-stiff integration methods. This paper achieves two main objectives. It defines a general quantitative assessment scheme for empirically testing integration schemes along very similar lines to Hull (Ref. 3). More importantly from the interests of this thesis it empirically tests implicit and explicit RK methods (the 4th Order and the 6th Order by Butcher, and of the 8th by Shanks), the extrapolation method of Bulirsch and Stoer (Ref 4), and the Adams based methods of Gear (Ref. 5) (based on Nordsieck Ref. 6) and of Krogh (Ref. 7) on a wide variety of problems. All their PC tests were implemented in an automated variable integration order, variable incremental step length form - features which are superfluous to the simulator design requirements. It was assumed that the results would hold true for a fixed order, fixed step length form.

This article concluded that the method of Bulirsch and Stoer was best for differential equations with simple function evaluations (up to 25 arithmetic operations), while Krogh was best for more complex functions with Gear also performing well. In the comparison between Gear and Krogh, the better performance of Krogh was attributed to its more flexible implementation with respect to order change. For this reason both Gear and Krogh had at this stage to be considered of equal merit. The inflexible size limitation on the function evaluations when using the Bulirsch - Stoer caused this method to be rejected. RK methods were found in test to be non-competitive - a result consistent with the conclusions of the previous chapter.

In the preface to the book of Shampine and Gordon (Ref. 8) it is stated that Gear and Krogh are considered to be two of the best methods available with the present state of the art. The fact that they have both been adopted by the NAG system library in recent years

adds further weight to the body of opinion backing Gear and Krogh as good general methods.

Gear's method is essentially the Nordsieck formula except that it contains certain sophistications in program implementation and philosophy. Since the interest of this thesis does not include such sophistications, the Nordsieck basis is considered and henceforth referred to by this name.

3. The Adams - Bashford - Moulton Linear Multistep PC Formulation

This linear multistep method forms the basis of the short listed methods of Krogh and Nordsieck. For this reason its derivation is included here.

As mentioned in Section 4.3 of previous chapter

$$\dot{y} = f(y, t) \tag{3.1}$$

can be integrated between t_{m-1} and t_m to obtain

$$y_m - y_{m-1} = h \int_{t_{m-1}}^{t_m} f(y, t)\, dt \tag{3.2}$$

If the function $f(y, t)$ is replaced by an interpolating polynomial, the basis of the linear multistep method is formed. Depending on the polynomial chosen the resulting method can either be an explicit prediction or an implicit corrector. The Adams - Bashford predictor, and the Adams - Moulton corrector are examples of each. These methods are the linear multistep basis for the short listed methods of Krogh and Nordsieck.

Before proceeding with the derivation, the standard backward difference operators must be defined, thus

$$\nabla y_m = y_m - y_{m-1}$$

$$\nabla^2 y_m = \nabla y_m - \nabla y_{m-1} = y_m - 2y_{m-1} + y_{m-2}$$

$$\vdots$$

$$\nabla^k y_m = \nabla^{k-1} y_m - \nabla^{k-1} y_{m-1} = \sum_{r=0}^{k} (-1)^r \binom{k}{r} y_{m-r}$$

and hence by rearrangement and recursive substitution

$$y_{m-1} = y_m - \nabla y_m = (1 - \nabla)\, y_m$$

$$y_{m-2} = y_m - 2\nabla y_m + \nabla^2 y_m = (1 - \nabla)^2 y_m$$

$$\vdots$$

$$y_{m-k} = y_m - \binom{k}{1} \nabla y_m + \binom{k}{2} \nabla^2 y_m - \ldots\ldots (-1)^r \binom{k}{r} \nabla^r y_m + \ldots. \tag{3.3}$$

Make the substitution in (3.3) of $T = -k$, such that

$$y_{m+T} = y_m + \frac{T}{1!} \nabla y_m + \frac{T(T+1)}{2!} \nabla^2 y_m + \ldots\ldots + \frac{1}{r!} \prod_{i=0}^{r} (T + i)\, \nabla^r y_m + \ldots \tag{3.4}$$

- Newton's backward difference interpolation formula.

(3.1) and (3.2) gives the relationship

$$y_m = y_{m-1} + h \int_{t_{m-1}}^{t_m} \dot{y}(t)\, dt \tag{3.5}$$

Replace t by $t_{m-1} + hT$ to obtain

$$y_m = y_{m-1} + h \int_0^1 \dot{y}\,(t_{m-1} + hT)\, dT \tag{3.6}$$

$$= y_{m-1} + h \int_0^1 \Big(\dot{y}_{m-1} + \frac{T}{1!} \nabla \dot{y}_{m-1} + \frac{T(T+1)}{2!} \nabla^2 \dot{y}_{m-1} + \ldots\ldots + \frac{1}{r!} \prod_{i=0}^{r} (T+i)\, \nabla^r \dot{y}_{m-1} + \ldots \Big) dT$$

(by (3.4))

and after integration, this equation becomes

$$y_m = y_{m-1} + h\Big(\dot{y}_{m-1} + \frac{1}{2} \nabla \dot{y}_{m-1} + \frac{5}{12} \nabla^2 \dot{y}_{m-1} + \ldots\ldots\Big) \tag{3.7}$$

which is the Adams - Bashford Explicit Predictor formula.

The Adams - Moulton Implicit Corrector formula is obtained in the following manner:-

Replace t by $t_m - hk$ in (3.5), so that

$$y_m = y_{m-1} + h \int_o^1 \dot{y}(t_m - hk)\, dk \qquad (3.8)$$

$$= y_{m-1} + h \int_o^1 \left(\dot{y}_m - \frac{k}{1!} \nabla \dot{y}_m + \frac{k(k-1)}{2!} \nabla^2 \dot{y}_m - \ldots + (-1)^r \frac{k!}{r!(k-r)!} \nabla^r \dot{y}_m - \ldots \right) dk$$

(by (3.3).)

which on integrating gives

$$y_m = y_{m-1} + h(\dot{y}_m - \frac{1}{2} \nabla \dot{y}_m - \frac{1}{12} \nabla^2 \dot{y}_m - \ldots.) \qquad (3.9)$$

- the Adams - Moulton Implicit Corrector.

It is obvious the Adams - Bashford - Moulton PC method is not amenable to step length change. If a change of step length is contemplated, the spacing of the back values $\dot{y}_{m-i}$ is not correct. This could only be rectified at the expense of considerable numerical complexity and the method must be rejected.

4. Characteristics of Methods

For the purposes of the argument to finalise the choice of PC method, it is necessary to outline the two methods and discuss their characteristics. A detailed derivation of both is considered superfluous, although the Nordsieck method finally chosen is derived in the next chapter. Both methods have a variable step length capacity.

Outline of Krogh's Method (Krogh Ref. 7)

The predictor formula in Krogh's method, which is capable of handling any order differential equation system directly, simplified for a single component of a first order DE system takes the form at m^{th} time instant -

$$y_m^{(0)} = y_{m-1} + h_{m-1} \sum_{i=0}^{p-1} G_i(m-1)\, F_i(m-1) \tag{4.1}$$

while the corrector is given by

$$y_m^{(1)} = y_m^{(0)} + h_{m+1}\, G_p(m-1)\, F_p(m)/B_p(m) \tag{4.2}$$

where

$$G_i(m) = \begin{cases} 1 & \text{for } i = 0 \\ \sum_{j=1}^{i} \dfrac{A_{ij}(m)}{j+1} & \text{for } i = 1, 2, \ldots. \end{cases} \tag{4.3}$$

$$F_i(m) = \begin{cases} f(t_m) & \text{for } i = 0 \\ \prod_{j=0}^{i-1} E_j(m)\, f(t_m, t_{m-1}, \ldots., t_{m-i}) & \\ & \text{for } i = 1, 2, \ldots. \end{cases} \tag{4.4}$$

$$B_i(m) = \begin{cases} 1 & \text{for } i = 0 \\ \prod_{j=0}^{i-1} E_j(m+1)/E_j(m) & \\ & \text{for } i = 1, 2, \ldots. \end{cases} \tag{4.5}$$

The $A_{ij}(m)$ are firstly calculated recursively in the following manner,

$$A_{i+1,j+1}(m) = \begin{cases} h_{m+1}/E_i(m) & \text{for } j = 0 \text{ and } i = 0, 1, 2, \ldots p \\ E_i(m) \sum_{k=j}^{i} A_{k,j}(m) & \text{for } j = 1, 2, \ldots p-1 \text{ and } i = j, j+1, \ldots p-1 \end{cases} \tag{4.6}$$

The other variables are defined as follows,

$$E_i(m) = \sum_{k=0}^{i} h_{m-k} \tag{4.8}$$

h_m = incremental step length at the "m^{th}" step.

The derivation of these formulae for the general case of a system of "d^{th}" order is given by Krogh (Ref. 7). The notation used here

is consistent with the rest of this section. Krogh's notation differs with respect to G, F, B, E which are replaced by their Greek equivalents, and with respect to the single suffix on G - a simplification resulting from 1^{st} order DEs only being considered here.

Krogh's method essentially is a multistep, multiderivative method, with an infinitely variable incremental step length. In (4.1) and (4.2) it is simplified in form by omitting the multiderivative capability and considering only the 1^{st} order DE. Even in this form it is numerically complex.

On comparison with Nordsieck's method it becomes obvious that even in the simplified 1^{st} derivative form given in (4.1) and (4.2), Krogh's method is considerably more arithmetically complex. Since Krogh recommends only gradual changes in step length then no significant saving in computational effort can be anticipated from the infinitely variable step length capability. Any such computational savings which could result from the method can therefore reasonably be expected to come from the variable order facility. This is borne out by Hull et al. (Ref. 2).

It should be noted that in (4.1) and (4.2), the predictor formula is of order "p" while the corrector has an order "p + 1", This makes error estimation more difficult than it need be (see previous chapter, section 7.2).

If the infinitely variable step length facility is removed then with 1^{st} order DEs, the method simplifies to form an Adams - Bashford predictor of order "p" and an Adams - Moulton corrector of order "p + 1" which as mentioned in section 3 is inflexible with respect to step length change.

Outline of Nordsieck's Method. (Nordsieck Ref. 6)

At its inception, the Nordsieck method was geared fundamentally to overcome the difficulty of changing incremental step length, while maintaining the desirable features of the linear multistep form of other PC methods. Step length change is easily accomplished by

altering 'h' in the Nordsieck predictor - corrector formulae. This step length flexibility is accomplished by storing the solution history in a vector of additional quantities based on higher derivatives rather than as a series of back values. (A detailed derivation is given in the following chapter.)

The p^{th} order predictor vector formula is

$$\underset{\sim}{a}_m^{(0)} = \underline{P}\, \underset{\sim}{a}_{m-1} \tag{4.9}$$

where

$$\underline{P} = \begin{bmatrix} 1 & 1 & 1 & & & & 1 \\ & 1 & 2 & 3 & . & . & p-1 \\ & & 1 & 3 & & & . \\ & & & & & & . \\ & & & & & & . \\ & 0 & & & & 1 & p-1 \\ & & & & & & 1 \end{bmatrix}$$

(The Pascal triangular matrix)

and $[\underset{\sim}{a}_m^{(j)}]^T = \left| y_m^{(j)} \; f_m^{(j)} \; a_{2,m}^{(j)} \; . \; . \; . \; . \; a_{p,m}^{(j)} \right|$

for the j^{th} correction.

The p^{th} order corrector vector at the $(j+1)^{th}$ correction is given by

$$\underset{\sim}{a}_m^{(j+1)} = \underset{\sim}{a}_m^{(j)} + h \quad \Delta f \quad \underset{\sim}{c} \tag{4.10}$$

where $\Delta f = f_m^{(j)} - f_m^{(0)}$

and $\underset{\sim}{c}^T = \left| c_0 \; c_1 \; c_2 \; . \; . \; . \; . \; c_p \right|$

where $c_1 = 0$, always

A single superficial glance reveals that the Nordsieck method is computationally much less complex than that of Krogh, despite the requirement for the calculation of a vector of quantities, $\underset{\sim}{a}_m^{(j+1)}$.

In fact the computational efficiency can be shown to be greater than that of Adams - Bashford - Moulton PC method, which Krogh reduces to if the flexible step length capability is removed. Gear (Ref. 9) demonstrated that as the predictor matrix, $\underline{P}$, was of Pascal triangular form great savings in computational effort could be achieved as shown in the accompanying table.

Method	Multiplications	Additions	Example comparative time*
Nordsieck	$p + j - 2$	$p(p + 1)/2 + 2j - 2$	27 units
Adams	$2p + j - 2$	$2(p + j) - 3$	31 units

where p = integrator order, j = no. of corrector iterations.

* The example comparative time is taken for $p = 5$, $j = 2$ where the conservative assumption that the time ratio of 1 multiplication : 1 addition = 2 : 1 .

In the brief outline of the methods no mention was made to the truncation error order, or the stability properties. On these points bearing in mind that the order of the integrator is fixed there is little to choose between them. However for completeness they will now be briefly mentioned.

It is difficult to estimate the algebraic value of the truncation error of the Krogh algorithm, owing to its continuously varying step length. However, as step length changes occur gradually, then it is reasonable to approximate the error by that of the Adams - Bashford Predictor and Adams - Moulton Corrector with some mean step length $\bar{h}$. Krogh's method reduces to that of Adams - Bashford - Moulton if the variable step length capability is omitted for the case of a first order DE. Thus, the predictor error, $e_n^{(p)}$, can be approximated by

$$e_n^{(p)} = K \bar{h}^{p} y_n^{(p)}$$

for a predictor of order $p - 1$.

The corrector error, $e_n^{(c)}$, is given by

$$e_n^{(c)} = K \bar{h}^{p+1} y_n^{(p+1)}$$

where the corrector is of order p.

It is the lower order formula which governs the order of truncation error for the method. (This is easily shown if the predictor formula with its error term is expanded in the corrector.)

The Nordsieck Method can be shown to be directly equivalent to the Adams - Bashford - Moulton linear multistep method, with identical truncation error. The predictor and the corrector are of the same order in this method.

As was mentioned in the previous chapter, stability is very much dependant on the differential equation to be integrated. On a variety of test problems done by Hull et al, there appears to be no significant difference in stability bounds. (In these variable order variable step length forms a limited stability bound would have been demonstrated by poor computational efficiency, which neither method exhibits.) Although Krogh (Ref. 10) does claim better stability properties as a result of the infinitely variable step length capability. However, the improvement appears marginal.

5. Conclusion

The literature survey narrowed the choice in the PC class to two methods, namely Krogh and Nordsieck. Of these two methods it was reasoned that Nordsieck best met the numerical integrators design requirements.

The variable order capability was considered superfluous to the design requirements of the integrator. It is only with the inclusion of this facility that the Krogh Method can prove computationally competitive. Without this facility, on a step by step basis, the infinitely variable step length capability in the Krogh Method proves too cumbersome to use. It is considerably more complex than the fixed step length Adams - Bashford - Moulton PC Method which in turn requires more computational effort per step than does a variable step length Nordsieck Method.

Related to the design requirement of computational efficiency is the requirement of efficient error estimation. Once again, because of the rejection of the variable order facility, error estimation in the

Krogh Method proves difficult, the predictor formula being of a different order from the corrector. If the predictor and the corrector possess the same order, as is the case with the Nordsieck Method, error estimation is a trivial operation (see section 7.2, previous chapter).

Start up can be easily accomplished by making use of the variable step length capability, where an artificially small step length is chosen initially. Although with the Krogh method it is computationally very inefficient. (Krogh in fact advocates use of the variable order facility for start up.)

On the other design requirements of accuracy and stability, neither Krogh nor Nordsieck possessed any overwhelming advantage.

In short, the Krogh Method relies heavily on its variable order capability. Without this facility the method is not competitive. However as the order of the process to be integrated is anticipated not to vary then order change capabilities appear superfluous to design requirements. Consequently, Krogh's Method must be rejected in favour of Nordsieck.

REFERENCES - CHAPTER VI

Ref. 1 Hull, T.E., and Newbury, A.C. (1962) "Corrector formulas for multistep integration methods." J. Soc. Indust. Appl. Math. 10, 351 - 369.

Ref. 2 Hull, T.E., Enright, W.H., Fellen, B.M., Sedgwick, A.E. (1972) "Comparing methods for ordinary differential equations." SIAM. J. Numer. Anal. 9, 603 - 637.

Ref. 3 Hull, T.E. (1967) "A search for optimum methods for the numerical integration of ordinary differential equations." SIAM. Rev. 9, 647 - 654.

Ref. 4 Bulirsch, R., Stoer, J. (1966) "Numerical treatment of ordinary differential equations by extrapolation methods". Num. Math. 8, 1 - 13.

Ref. 5 Gear, C.W. (1971) "The automatic integration of ordinary differential equations. "Comm. Assoc. Comp. Mach. 14, 176 - 179.

Ref. 6 Nordsieck, A. (1962) "On the numerical integration of ordinary differential equations." Math. Comp. 16, 22 - 49.

Ref. 7 Krogh, F.T. (1969) "A variable step, variable order multistep method for the numerical solution of ordinary differential equations." Information Processing 68 (Proc. IFIP Congress 1968) pub. North Holland.

Ref. 8 Shampine, L.F. and Gordon, M.K. (1975) "Computer solution of ordinary differential equations." pub. Freeman.

Ref. 9 Gear, C.W. (1967) "The numerical integration of ordinary differential equations." Math Comp. 21, 146 - 156.

Ref.10 Krogh, F.T. (1973) "Algorithms for changing the step size." SIAM. J. Numer. Anal. 10, 949 - 965.

CHAPTER VII

MODE OF OPERATION OF A NORDSIECK BASED INTEGRATOR ALGORITHM

1. Introduction

A mode of operation of a Nordsieck based algorithm is arrived at.

An initial preamble through the derivation of the Nordsieck method and its equivalence with linear multistep methods, is used as a basis to describe the operation sequence of the Nordsieck based algorithm which was finally implemented.

From this basis the salient features of the implementation are discussed with reasons given for their adoption.

2. Derivation of the Nordsieck Method

In Nordsieck (Ref. 1) it was the author's intention to merge the efficient aspects of linear multistep PC methods with an easily implemented step length change capability. (The step length change capability also facilitates integration start up.) To prevent obscuring Nordsieck's objectives it is his original approach which is stated in this section rather than the more elegant approach of Osborne 1966 where the equivalence with linear multistep PC methods was demonstrated.

In the solution of the standard first order DE,

$$\dot{y} = f(y, t) \qquad (2.1)$$

the approach commences by opting to save at the current time, t_m, the quantity, y_m, and its, 'p', higher derivatives (for a 'p^{th}' order integrator). Since all stored information applies to the current time step, t_m, (unlike linear multistep PC methods) no complicated interpolation is required for step length changes. At this point the method is modified to reduce the insensitivity and consequent inaccuracy caused by wild fluctuations in higher derivative magnitudes.

Thus, in the basic Taylor's Series integrator,

$$y_{m+1} = y_m + h \left(f(y_m, t_m) + \frac{h}{2!} \ddot{y}_m + \frac{h^2}{3!} \dddot{y}_m + \cdots + \frac{h^{p-1}}{p!} y_m^p \right) \quad (2.2)$$

The following substitutions are implemented in, for example the 5 point vector of higher derivatives,

$$\left.\begin{aligned} a_m &= \frac{h}{2!} \ddot{y}_m \quad , \quad b_m = \frac{h^2}{3!} \dddot{y}_m \quad , \\ c_m &= \frac{h^3}{4!} y_m^{iv} \quad , \quad d_m = \frac{h^4}{5!} y_m^{v} \end{aligned}\right\} \quad (2.3)$$

It is reasonable to assume that the scaling effect of h^{i-1} causes the effect of wild fluctuations in 'y_m^i' to be reduced in proportion to the order 'i' in the 'i^{th}' coefficient.

Now the integrator is fully defined by the following set of Taylor's series.

$$\left.\begin{aligned} y_{m+1} &= y_m + h f_m + \frac{h^2}{2!} \ddot{y}_m + \frac{h^3}{3!} \dddot{y} + \frac{h^4}{4!} y_m^{iv} + \frac{h^5}{5!} y_m^{v} + \frac{h^6}{6!} y_m^{vi} \\ f_{m+1} &= f_m + h \ddot{y}_m + \frac{h^2}{2!} \dddot{y}_m + \frac{h^3}{3!} y_m^{iv} + \frac{h^4}{4!} y_m^{v} + \frac{h^5}{5!} y_m^{vi} \\ &\vdots \\ y_{m+1}^{v} &= y_m^{v} + h y_m^{vi} \end{aligned}\right\} \quad (2.4)$$

$$\text{Let } e_m = \frac{h^5}{6!} y_m^{vi} \quad (2.5)$$

Then from (2.3) the following equations are obtained

$$\left.\begin{aligned} y_{m+1} &= y_m + h \left(f_m + a_m + b_m + c_m + d_m + e_m \right) \\ f_{m+1} &= f_m + 2a_m + 3b_m + 4c_m + 5d_m + 6e_m \\ a_{m+1} &= a_m + 3b_m + 6c_m + 10d_m + 15e_m \\ b_{m+1} &= b_m + 4c_m + 10d_m + 20e_m \\ c_{m+1} &= c_m + 5d_m + 15e_m \\ d_{m+1} &= d_m + 6e_m \end{aligned}\right\} \quad (2.6)$$

As an example in equation set (2.6), d_{m+1}, is derived from (2.3), (2.4) and (2.5) in the following manner,

$$y^{v}_{m+1} = y^{v}_{m} + h\, y^{vi}_{m} \qquad (2.4)$$

and $$d_{m+1} = \frac{h^4}{5!}\, y^{v}_{m+1}$$

hence, $$d_{m+1} = \frac{h^4}{5!}\,(y^{v}_{m} + h\, y^{vi}_{m})$$

but $$e_m = \frac{h^5}{6!}\, y^{vi}_{m} \qquad (2.5)$$

and so $$d_{m+1} = d_m + 6e_m \qquad (2.6)$$

If the predicted value of $f^{(0)}_{m+1}$ is defined by

$$f^{0}_{m+1} = f_m + 2a_m + 3b_m + 4c_m + 5d_m \qquad (2.7)$$

then by use of the function evaluation of the differential equation

$$f(y(t_{m+1}),\ t_{m+1}) \quad \text{in} \quad (2.6)$$

$$e_m = 1/6\left(f(y(t_{m+1}),\ t_{m+1}) - f^{(0)}_{m+1}\right) \qquad (2.8)$$

$$= 1/6\ D_f.$$

Eliminate e_m from (2.6) to obtain

$$\left.\begin{aligned}
y_{m+1} &= y_m + h\,(f_m + a_m + b_m + c_m + d_m + 1/6\ D_f) \\
f_{m+1} &= f_m + 2a_m + 3b_m + 4c_m + 5d_m + D_f \\
a_{m+1} &= a_m + 3b_m + 6c_m + 10d_m + 15/6\ D_f \\
b_{m+1} &= b_m + 4c_m + 10d_m + 20/6\ D_f \\
c_{m+1} &= c_m + 5d_m + 15/6\ D_f \\
d_{m+1} &= d_m + D_f
\end{aligned}\right\} \qquad (2.9)$$

The second equation of set (2.9) is redundant as f_{m+1} is never implicitly required. It can be replaced by $f(y(t_{m+1}), t_{m+1})$. This is not true of the predicted value, where

$$f_{m+1}^{(0)} = f_m + 2a_m + 3b_m + 4c_m + 5d_m \qquad (2.10)$$

Consequently, equation (2.10) replaces equation (2.9.ii).

Equation (2.9) with the aforementioned substitution is very unstable. To improve stability, it is necessary to alter the coefficients of Df throughout the equation set. Consequently, the coefficients 1/6, 15/6, 20/6, 15/6, 1 are replaced by the as yet unknown coefficients Y, A, B, C, D, respectively, to yield the result

$$y_{m+1} = y_m + hf_m + \underset{\sim}{e}^T \underset{\sim}{v}_m + Yh \; D_f \qquad (2.11.i)$$

$$hf_{m+1} = hf_m + \underset{\sim}{g}^T \underset{\sim}{v}_m \qquad (2.11.ii)$$

$$\underset{\sim}{v}_{m+1} = \underline{U}\,\underset{\sim}{v}_m + h \; Df \; \underset{\sim}{V} \qquad (2.11.iii)$$

where vectors

$$\underset{\sim}{v}_m^T = h\begin{bmatrix} a_m & b_m & c_m & d_m \end{bmatrix}$$

$$\underset{\sim}{e}_m^T = \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\underset{\sim}{g}^T = \begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix}$$

$$\underset{\sim}{V}^T = \begin{bmatrix} A & B & C & D \end{bmatrix}$$

and matrix

$$\underline{U} = \begin{bmatrix} 1 & 3 & 6 & 10 \\ . & 1 & 4 & 10 \\ . & & 1 & 5 \\ 0 & . & . & 1 \end{bmatrix}$$

NB The relations (2.11) hold for any degree of Nordsieck type method.

Recall from equation (2.8) that

$$D_f = f(y(t_{m+1}), t_{m+1}) - f_{m+1}^{(0)}$$

Eliminate $f_{m+1}^{(0)}$ from equations (2.11.i) and (2.11.iii) by use of (2.11.ii) to obtain

$$y_{m+1} = y_m + h\, f_m + \underset{\sim}{E}^T \underset{\sim}{v}_m + Y\, h\, F_m \qquad (2.12.i)$$

$$\underset{\sim}{v}_{m+1} = \underline{M}\, \underset{\sim}{v}_m + h\, F_m \underset{\sim}{V} \qquad (2.12.ii)$$

where matrix $\underline{M} = \underline{U} - \underset{\sim}{V}\, \underset{\sim}{g}^T$,

vector $\underset{\sim}{E} = (\underset{\sim}{e} - Y\underset{\sim}{g})$,

and scaler $F_m = f_{m+1} - f_m$.

Equations (2.12) yield the variational equations

$$\left.\begin{aligned} \delta y_{m+1} &= \delta y_m + h\, \delta f_m + \underset{\sim}{E}^T \delta \underset{\sim}{v}_m + Yh\, (\delta f_{m+1} - \delta f_m) \\ \delta \underset{\sim}{v}_{m+1} &= \underline{M}\, \delta \underset{\sim}{v}_m + h\, (\delta f_{m+1} - \delta f_m)\, \underset{\sim}{V} \end{aligned}\right\} \qquad (2.13)$$

For notational convenience, the scalar substitution $q = h\frac{\partial f}{\partial y}$ is implemented, despite the fact $\frac{\partial f}{\partial y}$ is a Jacobian matrix.

'q' is now substituted into (2.13) and the equations are rearranged to give

$$\left.\begin{aligned} (1 - Yq)\, \delta y_{m+1} &= (1 + q - Yq)\, \delta y_m + \underset{\sim}{E}^T \delta \underset{\sim}{v}_m \\ -q\, \delta y_{m+1} \underset{\sim}{V} + \delta \underset{\sim}{v}_{m+1} &= \underline{M}\, \delta \underset{\sim}{v}_m - q\, \delta y_m \underset{\sim}{V} \end{aligned}\right\} \qquad (2.14)$$

If (2.14.i) is substituted in (2.14.ii), after rearrangement the result obtained is

$$\begin{aligned} \delta \underset{\sim}{v}_{m+1} &= \underline{M}\, \delta \underset{\sim}{v}_m + \frac{q}{1-Yq}\, (q\, \delta y_m + \underset{\sim}{E}^T \delta \underset{\sim}{v}_m)\, \underset{\sim}{V} \\ &= \frac{1}{1-Yq} \left(q^2\, \delta y_m \underset{\sim}{V} + \underline{Q}\, \delta \underset{\sim}{v}_m\right) \end{aligned} \qquad (2.15)$$

where matrix

$$\underline{Q} = (1-Yq)\, \underline{M} + q(\underset{\sim}{V}\, \underset{\sim}{E}^T).$$

Equations (2.14.i) and (2.15) yield the general partitioned matrix relation

$$\begin{bmatrix} \delta y_{m+1} \\ \hline \delta \underset{\sim}{v}_{m+1} \end{bmatrix} = \frac{1}{1-Yq} \left[\begin{array}{c|c} (1 + q - Yq) & \underset{\sim}{E}^T \\ \hline q^2 \underset{\sim}{V} & \underline{Q} \end{array}\right] \begin{bmatrix} \delta y_m \\ \hline \delta \underset{\sim}{v}_m \end{bmatrix} \qquad (2.16)$$

The next step in the stability improvement is to establish the characteristic equation. This is straight forwardly, if somewhat tediously obtained from (2.16) by setting

$$\delta y_i = r_y \lambda^i$$

and $$\delta \underset{\sim}{v}_i^T = \lambda^i \begin{bmatrix} r_a & r_b & r_c & r_d \end{bmatrix}$$

for all i

where r_k = constant for all k.

(See for example Rutishauser (Ref. 7)).

The solution of the eigenvalue problem of the resulting determinant is next established, to yield the characteristic equation

$$\begin{aligned} &(1 - Yq)(\lambda - 1)^5 \\ &+ [2A + 3B + 4C + 5D - (1 + A + B + C + D)q](\lambda - 1)^4 \\ &+ [6B + 24C + 70D - (2A + 6B + 14C + 30D)q](\lambda - 1)^3 \\ &+ [24C + 180D - (6B + 36C + 150D)q](\lambda - 1)^2 \\ &+ [120D - (24C + 240D)q](\lambda - 1) \\ &+ (120D)q = 0 \end{aligned} \qquad (2.17)$$

Clearly there is no unique choice of Y, A, B, C, D, and q which satisfies equation (2.17). However, by means of the method of undetermined coefficients, these roots - as functions of Y, A, B, C, D and q - can be determined.

The function

$(1 - Yq) \prod_{n=1}^{5} (\lambda - \lambda_n)$ is set equal to the left hand side (L.H.S.) of the

characteristic equation (2.17), where

$$\lambda_n = u_{0,n} + u_{1,n}\, q + u_{2,n}\, q^2 + \ldots\ldots$$

for n = 1, 2, . ., 5

where $u_{j,n}$ are the, as yet, undetermined coefficients.

On expansion, powers of q are equated to establish the coefficients $u_{j,n}$.

The essential root, λ_0, can be shown to agree with an exponential series up to an error of $0(q^6)$, with

$$\lambda_0 = \exp(q) - \frac{1}{D}\left[(3 - 6Y - A + 1/5C)\, q^6/6! + (49 - 105Y - 14A - 7/5B + 14/5C + D)\, q^7/7!\right] + 0(q^8) \qquad (2.18)$$

The essential root, λ_0, as discussed more generally in an earlier chapter, merely determines the stability of the differential equation system.

In order to optimise the stability of the method it is necessary to examine the non-essential roots. Since, 'q' is the Jacobian $h \frac{\partial f_i}{\partial y_j}$ for all i, j, it too is highly dependant on the differential equation system, as well as on h which can be so adjusted to act as a stability controller. 'q' is thus of no great significance when looking at the method dependant non-essential roots. Hence, as far as the non-essential roots are concerned, the analysis can be simplified by setting $q = 0$ in equation (2.17) to obtain

$$(\lambda - 1)\left[\lambda^4 + (2A + 3B + 4C + 5D - 4)\,\lambda^3 + (-6A - 3B + 12C + 55D + 6)\,\lambda^2 + (6A - 3B - 12C + 55D - 4)\,\lambda + (-2A + 3B - 4C + 5D + 1)\right] = 0 \qquad (2.19)$$

To optimise the method's stability it is necessary to adjust A, B, C, D such that all non-essential roots lie on the origin. Whence, equation (2.19) must take the form

$$(\lambda - 1)\,\lambda^4 = 0 \qquad (2.20)$$

Equate powers in λ^i, for i = 0, 1, 2, 3, to uniquely specify A, B, C, D. Hence coefficient values for optimum method stability are

$$A = 25/24\ ,\quad B = 35/72\ ,\quad C = 5/48\ ,\quad D = 1/120.$$

It is now a simple matter to optimise the method's accuracy, by choosing Y such that the essential root, λ_0, agrees with an exponential series through terms in q^6. Thus, to eliminate the term in q^6 in equation (2.18), the value Y = 95/288 must be adopted. The error order or degree, $O(h^7)$, is now minimised for a 5th order method. This is in keeping with stability constraint established by Dahlquist (Ref. 2) section 2.6, where it is proven that a method of order k , where k is odd, can never be stable if the degree of the error term is greater than k + 2 . (N.B. This is different from Dahlquist's definition of degree. He defines it as the largest degree of h for which the method holds. If degree is so defined, then by his definition for stability, degree is never greater than k + 1 for odd k).

Should an even order of k be chosen then, again from Dahlquist, the degree of error could never be greater than k + 1 . Clearly it is advantageous computationally to choose an odd k value of method, assuming maximum degree of error is achieved (as indeed it is with Nordsieck.

3. Operation sequence of the Nordsieck based algorithm

In this section, the basic operational phases of prediction (P), function evaluation (E), and correction (C) are outlined for the Nordsieck Algorithm. These operational steps are then related to the set of two coupled first order DEs

$$\dot{y}(t) = v(t) \qquad (3.1)$$
$$\dot{v}(t) = (P(t) - c\,v(t) - k\,y(t))\,1/m$$

which constitute the vehicle dynamics problem (Refer to Chap. IV sect 2)

Thus predicting the 6 point vector of the DE

$$\dot{y} = f(y, t) \tag{3.2}$$

at the 'm^{th}' step from those at the '$(m-1)^{th}$' step, proceed as follows

$$\left.\begin{aligned}
y_m^{(0)} &= y_{m-1} + h\left[f_{m-1} + a_{m-1} + b_{m-1} + c_{m-1} + d_{m-1}\right] \\
f_m^{(0)} &= f_{m-1} + 2a_{m-1} + 3b_{m-1} + 4c_{m-1} + 5d_{m-1} \\
a_m^{(0)} &= a_{m-1} + 3b_{m-1} + 6c_{m-1} + 10d_{m-1} \\
b_m^{(0)} &= b_{m-1} + 4c_{m-1} + 10d_{m-1} \\
c_m^{(0)} &= c_{m-1} + 5d_{m-1} \\
d_m^{(0)} &= d_{m-1}
\end{aligned}\right\} \tag{3.3}$$

As a consequence of the Pascal triangular form of the coefficients of the y, f, a, b ...etc. on the right hand side of (3.3), prediction can be accomplished by the very efficient algorithm (Gear 1967)

```
For  I  :=  0 Step  1  until  k - 2  do
For  J  :=  k - 1 Step  - 1  until  I + 1  do
     A(J - 1)  :=  A(J - 1) + A(J)
```

where A(J) is the 'J^{th}' component of the vector A such that

$$A^T = \left[y_{/h} \quad f \quad a \quad b \quad c \quad d \right]$$

- (See Chapter V, Section 4 for fuller discussion)

This algorithm requires no multiplication. - This is the P phase.

N.B. The computer storage locations used by y, f, a, b, c, d at the $(m-1)^{th}$ step, can immediately be reallocated to store these quantities at the m^{th} step. Reversal of integration direction easily restores the $(m-1)^{th}$ value set if required for start up or error control.

Function Evaluation (E) is the next phase to be performed. It is achieved by substituting the predicted value $y_m^{(0)}$ in the original DE such that

$$\dot{y}_m^{(1)} = f(y_m^{(0)}, t_m) \tag{3.4}$$

Correction (C) is the third operation phase. The difference, D_f, between the prediction, $f_p^{(0)}$, and the function evaluation, $\dot{y}_m^{(1)}$, is calculated to obtain

$$D_{f_m}^{(1)} = \dot{y}_m^{(1)} - f_m^{(0)}$$

The general form is given by

$$D_f^{(i)} = \dot{y}^{(i)} - \dot{y}^{(i-1)} \quad \text{in subsequent iterations}$$

The predicted vector is then corrected in the following manner.

$$\left.\begin{aligned} y_m^{(1)} &= y_m^{(0)} + 95/288\ h\ D_f \\ f_m^{(1)} &= f_m^{(0)} \\ a_m^{(1)} &= a_m^{(0)} + 25/24\ h\ D_f \\ b_m^{(1)} &= b_m^{(0)} + 35/72\ h\ D_f \\ c_m^{(1)} &= c_m^{(0)} + 5/48\ h\ D_f \\ d_m^{(1)} &= d_m^{(0)} + 1/120\ h\ D_f \end{aligned}\right\} \qquad (3.5)$$

Further 'E' and 'C' operations can then be performed until the desired accuracy is obtained.

Extension to two coupled first order DEs (2.11) is easily accomplished. Two separate vectors are held in store; a displacement vector $\underset{\sim}{Y}$ and a velocity vector $\dot{\underset{\sim}{Y}}$. Prediction for both $\underset{\sim}{Y}$ and $\dot{\underset{\sim}{Y}}$ is first obtained, followed by similtaneous function evaluations from which the differences

$$\left.\begin{aligned} D_{f_{y_m}} &= v_m^{(0)} - f_{y_m}^{(0)} \\ \text{and} \quad D_{f_{v_m}} &= (P_m - cv_m^{(0)} - ky_m^{(0)})\ 1/m - f_{v_m}^{(0)} \end{aligned}\right\} \qquad (3.6)$$

are obtained. Thirdly, the $\underset{\sim}{Y}$ and $\dot{\underset{\sim}{Y}}$ vectors are corrected using $D_{f_{y_m}}$ and $D_{f_{v_m}}$ respectively.

Incremental step length changes can be easily accomplished. A change of step length from h to Kh, where K is a scaling factor,

is implemented by altering the quantities

$$h,\ a,\ b,\ c,\ d$$

such that they become

$$Kh,\ Ka,\ K^2b,\ K^3c,\ K^4d \quad \text{respectively.}$$

Reversal of the integration direction, from for example "t + h" to 't' can be achieved by altering quantities of odd power in h, namely

$$h,\ a,\ c$$

such that they become

$$-h,\ -a,\ -c$$

Both step length change and integration direction reversal follow automatically from equations (2.2) and (2.3). Both are important for error control and implementation of program start up, discussed later in this chapter.

4. The equivalence of Nordsieck and linear multistep methods

The equivalencing of Nordsieck with linear multistep methods was demonstrated by Osborne (Ref. 3).

By recursive substitution of (2.12.ii) the relationship

$$\underset{\sim}{v}_{m+p+1} = \underline{M}^{p+1}\,\underset{\sim}{v}_m + h\,L_{m+p+1,\,m}(\underline{M})\ \underset{\sim}{V} \qquad (4.1)$$

is obtained, where matrix

$$L_{m+p+1,\,m}(\underline{M}) = \sum_{i=0}^{p} F_{m+p+1-i}\,\underline{M}^i$$

and $\underline{M}^0 = \underline{I}$, the unit matrix

At the $(m + s + 1)^{th}$ step, (2.12.i) becomes

$$\begin{aligned} y_{m+s+1} &= y_{m+s} + h\,f_{m+s} + \underset{\sim}{E}^T \underset{\sim}{v}_{m+s} + Yh\,F_{m+s} \\ &= y_{m+s} + h\,f_{m+s} + Yh\,F_{m+s} \\ &\quad + \underset{\sim}{E}^T(\underline{M}^s\,\underset{\sim}{v}_m + h\,L_{m+s,\,m}(\underline{M})\ \underset{\sim}{V}) \end{aligned} \qquad (4.2)$$

by use of equation (4.1).

Multiply y_{m+s+1} by the scalar quantities

R_{s+1}, for $s = 0, 1, \ldots P$, and sum to obtain

$$\sum_{s=1}^{p+1} R_s\, y_{m+s} = \sum_{s=0}^{p} R_{s+1}\left(y_{m+s} + h\, f_{m+s} + Yh\, F_{m+s}\right) + \underset{\sim}{E}^{t}\left[\left(\sum_{s=0}^{p} R_{s+1}\, \underline{M}^{s}\right)\underset{\sim}{v}_m + h\left(\sum_{s=0}^{p} R_{s+1}\, L_{m+s,m}(\underline{M})\right)\underline{V}\right]. \tag{4.3}$$

If

$$\sum_{s=0}^{p} R_{s+1}\, \underline{M}^{s} = \underline{0} \tag{4.4}$$

then the term in $\underset{\sim}{v}_m$ vanishes from the equation and (4.3) reduces to a linear multistep method.

It should be noted that (4.3) is a linear combination of the Nordsieck method. This fact is important when the question of an optimum predictor - corrector sequence is considered.

5. Optimal Predictor-Corrector Sequence.

A decision must be reached on an optimal sequence of prediction (P), function evaluation (E), and correction (C). Should the predictor - corrector step end with a 'C' or an 'E' operation? How many iterative 'C' operations should there be? In other words, should sequence of operation be $P(EC)^I$ or $PE(CE)^I$ and what value of 'I' is optimum?

The first analytic attempt at making such decisions was given by Hull and Creemer (Ref. 4), where an Adams multistep $P(EC)^I$ was considered. It was found that stability was the overriding consideration. This is generally true. Their method of analysing the stability of $P(EC)^I$ Adams model holds good for all $P(EC)^I$ or $P(EC)^I E$ multistep methods. Their method considers the '$i+1^{th}$' application, for $i = 1, 2, \ldots I$, of the corrector forumla at the 'm^{th}' step

$$\underset{\sim}{y}_m^{(i+1)} = \underset{\sim}{y}_m^{(i)} + \underset{\sim}{c}\, G(\underset{\sim}{y}_m^{(i)}) \tag{5.1}$$

with global error, $\underset{\sim}{e}_m$, included (as before, $\underset{\sim}{y}$, $\underset{\sim}{c}$ are column vectors

and $G(\underset{\sim}{y})$ is a correction function). The corrector is then considered with the inclusion of local error, d_m. On subtraction an error equation is obtained. It takes the form

$$\underset{\sim}{e}_m^{(i+1)} = \underset{\sim}{e}_m^{(i)} + c\,\frac{\partial G(\underset{\sim}{T}_i)}{\partial y}\,\underset{\sim}{e}_m^{(i)} \qquad (5.2)$$

where $\underset{\sim}{e}_m^{i}$ is the global error vector after 'i' iterations at the 'm^{th}' step, and $\underset{\sim}{y}(t_m) + \underset{\sim}{d}_m < \underset{\sim}{T}_i < \underset{\sim}{y}(t_m) + \underset{\sim}{e}_m$ and $\underset{\sim}{y}(t_m)$ is the true value.

By recursive substitution of (5.2) for i=0,1,..I, an expression for the final error $e_m^{(I)}$ after 'I' iterations is obtained, once the effect of the prediction formula

$$\underset{\sim}{y}_m^{(0)} = \underline{B}\,\underset{\sim}{y}_{m-1}\,, \qquad (5.3)$$

with $\underline{B}$ a square matrix, is included. Error growth is determined from a stability matrix,

$$\underline{S}_m = \prod_{i=0}^{I} \left(\underline{I} + c\,\frac{\partial G(\underset{\sim}{T}_i)}{\partial \underset{\sim}{y}}\right) \qquad (5.4)$$

contained within the resultant expression.

(5.4) is the stability matrix. A full derivation and manipulation to yield an eigen type solution of the roots for the generalised multistep form can be found in Gear (Ref. 5).pp. 136-142.

Furthermore, Gear (Ref. 5) chapter IX section 2 demonstrated that the stability of any PECE . . . sequence used in a multistep formula possesses identical stability characteristics to the same PECE. . . sequence used in its multiderivative equivalent or vice versa. This analysis ignores numerical round off. The equivalence holds provided the multiderivative vector $\underset{\sim}{a}_m^{(i)}$ is a linear combination of the multistep vector $\underset{\sim}{y}_m^{(i)}$, thus

$$\underset{\sim}{a}_m^{(i)} = \underline{T}\,\underset{\sim}{y}_m^{(i)} \qquad (5.5)$$

where $\underline{T}$ is a square matrix, dimensions k x k
where $\underset{\sim}{y}_m$ is a 'k' step vector.

On substitution of (5.5) in the multistep formulae a similar derivation to that yielding (5.4) gives a stability matrix.

$$\underline{S}'_m = \underline{T}\,\underline{S}_m\,\underline{T}^{-1} \qquad (5.6)$$

for the multiderivative sequence.

This result is general. It proves that the stability of a PECE . . . sequence of an Adams - Bashford - Moulton multistep method is identical to a similar sequence in the Nordsieck multiderivative formula, one method being a linear combination of the other. Consequently if an optimal sequence for the Adams multistep method can be found, it is also optimal for the Nordsieck method.

Hall (Ref. 6) demonstrated experimentally that for multistep methods of order less than 11, the optimum sequence is $P(EC)^2E$ with a predictor and corrector of the same order. The 5th order formula of Nordsieck is equivalent to a 5th order Adams method. Consequently the optimum PECE . . . sequence for the chosen Nordsieck method is $P(EC)^2E$ and not $P(EC)^2$, as originally advocated by Nordsieck.

6. Step length flexibility

No automated step length alteration procedure, based on error control logic is implemented in this program. However efficient change of step length properties were essential for two reasons. Namely, to permit easy establishment of the higher order unknown quantities at start up, and secondly as a necessary condition for the accomplishment of a space domain analysis of the variable velocity road vehicle problem.

Since the initial displacement, y, and velocity, v are known, start up is achieved by setting all unknown additional quantities, (a, b, c, d), to zero and time marching an appropriate number of steps to allow an initial estimate of these additional quantities (for the 5th order Nordsieck method used the required number of steps is 4). After the 4th step, the direction of the integration method is reversed (as described in section 3) to permit a return to time step zero. The initial values of y and v are reassigned, while the new self-generated values of a, b, c, d are retained.

The 4 time steps forward followed by the 4 step reversal procedure is repeated to achieve an improved estimate of a, b, c, d at the initial start off time. The improved estimates of a, b, c, d are now good enough for the integration to commence. Pseudo start up has been achieved.

The step length used during start up is a fractional value of that used during normal step lengths. By keeping the step length at start up small it is possible to keep any initial error in the additional quantities (a, b, c, d) below the computer round off error obtained during normal running step length increments. The method of arriving at the fractional step length value is left until Chapter VIII.

A full explanation for choosing a space-domain approach is discussed in Chapter IX. However given that it is desirable for the numerical integrator of the non-stationary vehicle simulator to be done in the space-domain, then the velocity term, dy/dt, in the differential equation is broken into the product dy/dx . dx/dt, where y is (as always) vertical displacement, x is distance travelled along road profile, and t is time.

The road profile velocity, dx/dt, is grouped with the dynamic parameters of the differential equation. It is equivalent to altering the step length relative to the decay constants of the two first order DEs (4.1). Consequently as the road profile velocity changes, so also does the effective step length.

7. Monitoring the Integrators convergence.

The local truncation error has a magnitude given by $863/(12\times7!)\ h^7\ y^{vii}$ The equivalence with linear multistep methods already being established, this is most easily determined by comparing this equivalent linear multistep method with the Taylor's Series expansion (described in general terms in Chapter V). It should be borne in mind that the Adams linear multistep and Nordsieck methods are equivalent in all aspects save error, due to computer round off. This aspect will not be considered here.

The test advocated by Nordsieck for local error estimation is essentially that described in chapter V section 8.2. In Nordsieck's case error estimation, at predictor stage is carried out on the predicted value of 'f'. Correction stage error estimation is also carried out on 'f' but this is done during the evaluation part of the cycle which follows correction. Thus from the point of view of passive error

monitoring, since the method of Chapter V.8.2 is effectively used, all that is of concern is local error growth between time steps. Consequently at every time step the differences $\left|f^{(3)} - f^{(0)}\right|$ for both y_m and v_m are recorded to observe if growth occurs as the system time marches.

The approximate stability test described by Nordsieck is adopted here, due to the simplicity of its implementation and its minimal computational effort.

The stability criterion involves checking the relative magnitude difference between successive iterations of the responses y, and v. The tests are

$$\left|y^{(2)} - y^{(1)}\right| \leq 1/8 \left|y^{(1)} - y^{(0)}\right|$$

and $$\left|v^{(2)} - v^{(1)}\right| \leq 1/8 \left|v^{(1)} - v^{(0)}\right|$$

These relations are easily arrived at by comparing successive iterations of equation (4.5.i), namely

$y_m^{(i)} = y_m^{(i-1)} + Y\ h\ D_f$ (for the ith iteration and where as before $Y = 95/288$), to obtain

$$\left|y^{(2)} - y^{(1)}\right| = Y\ q \left|y^{(1)} - y^{(0)}\right|$$

where the 'q' is as specified in Section 2.
The inequality and the factor 1/8 stem from stability considerations (Nordseick Ref. 1 page 30) - A 'Y q' product of 1/8 ensures all non-essential roots lie well within unit circle of the complex λ-plane.

No automatic step length control logic is used in the program. Both tests are used in a passive manner, as checks on the validity of the solution. Vindication of their value is borne out in chapter VIII where tests on the integrator algorithm are discussed.

The conditions of consistency of order 7 and stability are thus met by this implementation of the Nordsieck algorithm. These two conditions (as discussed in chapter V section 7) are necessary and sufficient for the solution to converge.

8. Extension to non-linear DEs

The general form of the 'N^{th}' power non-linear first order DE is given by

$$\sum_{i=1}^{N} a_i(\dot{y})^i = f(y^Q, y^{Q-1}, \ldots, y, t) \qquad (8.1)$$

where Q is the power of y non-linearity.

In section 3 the operational sequence of this Nordsieck based integrator was described. The extension of this integrator to include such non-linear DEs is trivial although subject to qualification.

Equations (3.3) of the predictor (P) remain unchanged. As do equations (3.5) of the corrector (C). It is the function evaluation stage (E) of equation (3.4) which requires slight modification. The non-linear equivalent of equation (3.4) is given by

$$\dot{y}_m^{(P)} = \frac{1}{a_N}\left[f^{(p-1)}\left(y_m^Q, y_m^{Q-1}, \ldots . y_m, t_m\right) - \sum_{i=1}^{N-1} a_i \left(\dot{y}_m^{(P-1)}\right)^i \right]^{1/N} \qquad (8.2)$$

where m is the increment number,
and (P) is the correction number.
Equation (8.2) is easily extended to include systems of coupled first order DEs.

The method of monitoring convergence as the integrator time marched was discussed in section 7. Again the extension to non-linear systems is trivial.

The error monitoring technique is still applicable provided it is assumed (see chapter V section 8.2) that $\frac{d^q f}{dt^q}$ can be assumed reasonably constant over incremental time step h, where q is the integrator order. Such an assumption is reasonable for small h in cases where the function f is continuous. A category into which most dynamic systems fall.

The stability monitoring technique also applies in non-linear DEs.

Effectively, this technique monitors the convergence pattern of the solution at the current time instant alone, and is therefore relatively independant of the non-linear variation in the function evaluation as time proceeds. This technique is subject to a qualification. It is only valid if the prediction (P) is sufficiently accurate (hence a 7^{th} order formula). This qualification follows from the fact that the function evaluation (E) of a non-linear DE is non unique. Consequently, it is possible to predict the wrong solution. However the high order integrator formula can be reasonably expected to eliminate such an eventuality under most circumstances.

9. Conclusion

From the discussion on the implementation the following conclusions can be drawn.

1) If the coefficients of the method are chosen as described by Nordsieck then the stability is optimised.

2) Dahlquist's stability theorems demonstrate that it is computationally advantageous to choose a method where the order, k, is odd, if as in this case the error is of maximum degree.

3) The method is easily extended to include second order ordinary differential equations.

4) Gear's efficient predictor algorithm should be included.

5) The equivalence with linear multistep methods demonstrates that the optimum predict - evaluate - correct sequence for methods of order less than eleven is $P(EC)^2E$.

6) Step length and reversal of integration direction facilities must be included to overcome start up difficulties and problems incurred in the variable velocity road vehicle simulator.

7) The error and stability tests of Nordsieck are adopted, but only for passive monitoring of the behaviour of the integrator.

8) The extension of the integrator implimentation to include non-linear systems of coupled first order DEs is trivial and can be reasonably expected to yield valid results unless under the most extreme non-linear conditions.

REFERENCES - CHAPTER VII

Ref. 1 Nordsieck, A. (1962) (see ch. 6 Ref. 6)

Ref. 2 Dahlquist, G.G. (1956) "Convergence and stability in the numerical integration of ordinary differential equations." Math. Scand. 4, 33 - 35.

Ref. 3 Osborne, M.R. (1966) "On Nordsieck's method for the numerical solution of ordinary differential equations." BIT. 6, 51 - 57.

Ref. 4 Hull, T.E., and Creemer, A.L. (1963) "Efficiency of predictor - corrector procedures." J. Ass. Comput. Mach. 10, 291 - 301.

Ref. 5 Gear, C.W. (1971) "Numerical initial value problems in ordinary differential equations." pub. Prentice.

Ref. 6 Hall, G.H. (1974) "Stability analysis of predictor - corrector algorithms of Adams type." SIAM. J. Numer. Anal. 11, 494 - 505.

Ref. 7 Rutishauser, H. (1952) "Über die instabilität von methoden zur integration gewöhnlicher differential gleichungen." ZAMP. 3, 65 - 74.

CHAPTER VIII

TESTING THE DYNAMIC VEHICLE INTEGRATOR

1. Introduction and Objectives

The dynamic vehicle integrator, based on Nordsieck's integration algorithm, is used to model the response of a road vehicle when subjected to deterministic vertical road excitation.

The deterministic excitations can be classified into one of two categories. The first category involves inputting, to the vehicle model, excitations with simple shape functions eg a sinusoid or a step. The second category involves the input of random data. For brevity the former excitation category will be referred to as the deterministic tests, and the latter as statistical tests.

Only deterministic testing is considered in this chapter. Statistical testing is left until tests on the overall road vehicle simulator are implemented.

The deterministic tests on the integrator were designed to ascertain the following objectives.

a) Establish an optimum step length ratio, H(=h(t)/T) for the second order DE of the vehicle model.

b) Establish the effectiveness of the integrator in dealing with large sudden discontinuities at start up. (A sudden large discontinuity is the condition which exists when the current input acceleration due to the externally applied input excitation is large in comparison to the response acceleration of the previous time step).

c) Establish effectiveness of integration in dealing with subsequent* large discontinuities without further start ups.

* The term subsequent refers to discontinuities after time t = 0, where no further restart ups were implemented.

d) Establish effectiveness of stability test in monitoring the performance of the integrator so that the viability of solutions, where no closed form comparisons exist, can in future be ascertained.

2. Design of the Deterministic Tests

The deterministic tests were carried out on the simple 2nd Order DE

$$\ddot{y}(t) + 2\zeta\omega_n \dot{y}(t) + \omega_n^2 y(t) = P(t) \qquad (2.1)$$

where ζ = damping ratio, and ω_n = natural angular frequency.

In all tests the realistic damping ratio for a motor vehicle of $\zeta = 0.32$ was used. The frequency was normalised to unity (ie $\omega_n = 2\pi$). The system is initially at rest at time $t = 0$.

An input excitation load, $P(t)$, is applied. There are two design load cases of $P(t)$,

a) $P(t) = P \cos \omega t$ for $t \geq 0$
$= 0$ for $t < 0$

This input load case exhibits sudden large discontinuity in input excitation at time, $t = 0$, without subsequent discontinuities in input excitation. This load case proves most useful in establishing all deterministic test objectives save testing subsequent discontinuities.

b) $P(t) = P$ for $0 \leq t \leq T$ where T = period
$= 0$ elsewhere.

Sudden large discontinuities to input excitation at times $t = 0$ and $t = T$ are exhibited by this load case. It is primarily designed to test the integrator's effectiveness in dealing with subsequent large discontinuities, without further restart ups. This feature is important especially when random input excitations are encountered, as such subsequent large input discontinuity events will almost certainly occur from time to time.

3. Optimum Step length ratio, H

Decision on an optimum step length ratio, H, was arrived at after comparing the integrator results with the closed form solution. The step length ratio, H, is defined as follows,

$$H = h(t)/T$$

where $h(t)$ is the incremental step length used by the integrator formula, and T is the decay constant of the DE. In the case of a second order DE used for the vehicle model, it is sufficient to approximate T by the undamped natural period of the dynamic system ($= 2\pi/\omega$). In all the tests which follow it was this approximation to T which was used.

The absolute error magnitude, E_a

where $|E_a| = \left|\frac{y_n - y(t_n)}{y(t_n)}\right|$

y_n = displacement recorded by integrator solution,

and $y(t_n)$ = displacement of theoretical solution

for input excitation $P \cos \omega t$ and various H, is plotted against time increment number, N, in the graphs (figs. 8.1 - 8.4). The periodic nature of the DE's response causes the absolute error, $|E_a|$, to also undergo cyclic variation. Consequently, it was decided to measure $|E_a|$ at equivalent points every half cycle. For the purpose of these graphs the peak value of $y(t_n)$ was chosen.

To permit a controlled comparison. it was decided to exclude any step length ratio reduction at start up, during the variation of 'H' tests, provided acceptable results are obtained. (See following sub section.)

From these empirical tests, a step length ratio, H = 0.01, was considered a reasonable compromise between efficiency and accuracy. It was this value of H which was used in all subsequent road vehicle simulation studies.

4. Effectiveness in dealing with sudden large discontinuities at start up.

Reference to the graphs (figs. 8.1 - 8.4) demonstrates that the integrator can produce acceptable results for step length ratio $H \leq 0.05$. Although the start up procedure is implemented no initial step length ratio reduction was introduced and correspondingly in these graphs a reduction by at least one order of magnitude was observed in the absolute error magnitude $|E_a|$ after only several cycles. This was true for all H tested.

The integrator can thus deal with sudden discontinuities at start up without initial step length ratio reduction, although implementation of this feature is clearly desirable for the minimisation of initial error and subsequent error accumulation.

5. Effectiveness in dealing with subsequent large discontinuities.

The ability of the integrator to deal with subsequent large discontinuities without further start ups is demonstrated by the step input excitation load case (b). Here the subsequent discontinuity occurs at time, $t = T$.

It was observed, for all H considered, the absolute error increased at $t = T$ by no more than one order of magnitude. In general after only several time steps, the error fell in magnitude to its previous before discontinuity value.

For the compromise step length ratio, $H = 0.01$, the absolute error one time increment before this subsequent discontinuity is given by

$$|E_a|_{N=99} = 2.0370 \times 10^{-7}$$

while at one time increment after this discontinuity is given by

$$|E_a|_{N=101} = 4.7812 \times 10^{-7}$$

- the absolute error had almost completely recovered from this subsequent shock.

The nature of random road excitation, in a truly uncorrelated form,

makes it very desirable to have an integrator capable of handling large subsequent discontinuities in input excitation, without the use of any step length reduction.

6. Effectiveness of stability test for performance monitoring.

Observation of the frequency of failure in the stability test over 1000 time increments, yielded the results of the graph fig. 8.5.

In this graph, stability failure occurred when $\| y^{(2)} - y^{(1)} \| > \| y^{(1)} - y^{(0)} \|$. This is a relaxation on the test advocated by Nordsieck. The factor 1/8 was dropped from the right hand side of the inequality to ensure that only truly unstable events were recorded. (Recall from Chapter Vll only passive monitoring is advocated.)

In fig. 8.5, a marked increase in stability failure frequency is observed with increasing step length ratio, H. The graph demonstrates that a truly unstable $H(=0.1)$ has a marked increase in failure frequency of the stability test.

7. Conclusion

Deterministic excitations with simple shape functions were input into the dynamic integrator as the first stage of testing the method's viability. A step length ratio of $H = 0.01$ was decided upon as being a realistic compromise between accuracy and computational effort. The ratio was established by measuring the absolute error at equivalent time instants every half cycle. The ability of the integrator in effectively dealing with large discontinuities at start up and at subsequent time instants was demonstrated. At start up an initial step length reduction is recommended. Unfortunately such reductions are impractical at subsequent time instants in the case of random input excitations. However these step length reductions are not essential for the successful implementation of the integrator. The viability of the stability test was demonstrated by correlating frequency of stability test failure with absolute error information.

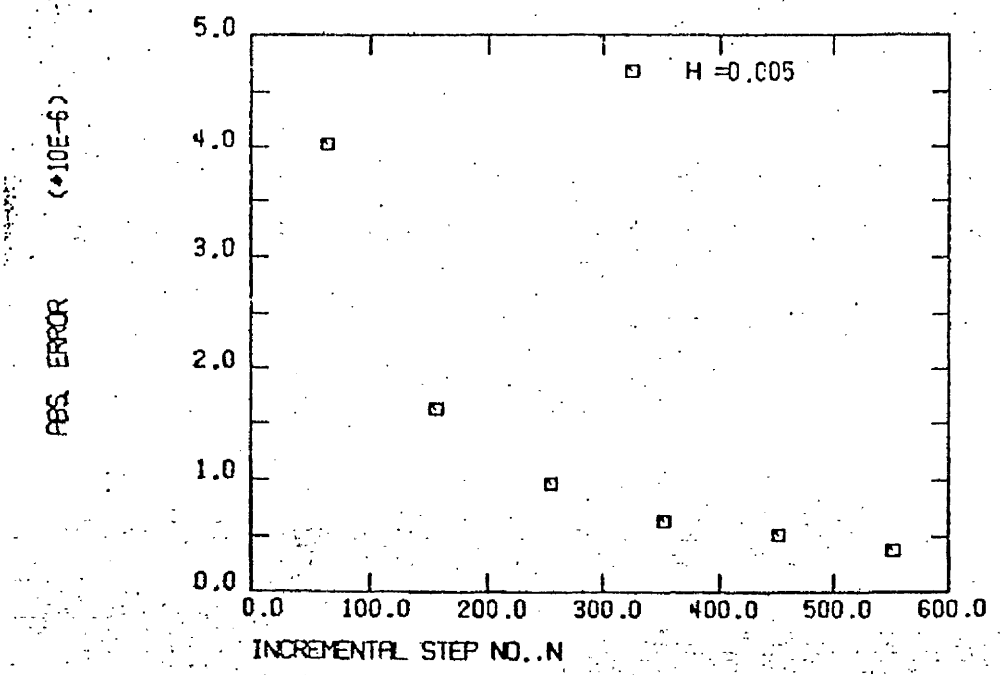


FIG. 8.1 ABS. ERROR VS. INCREMENTAL STEP NO..N

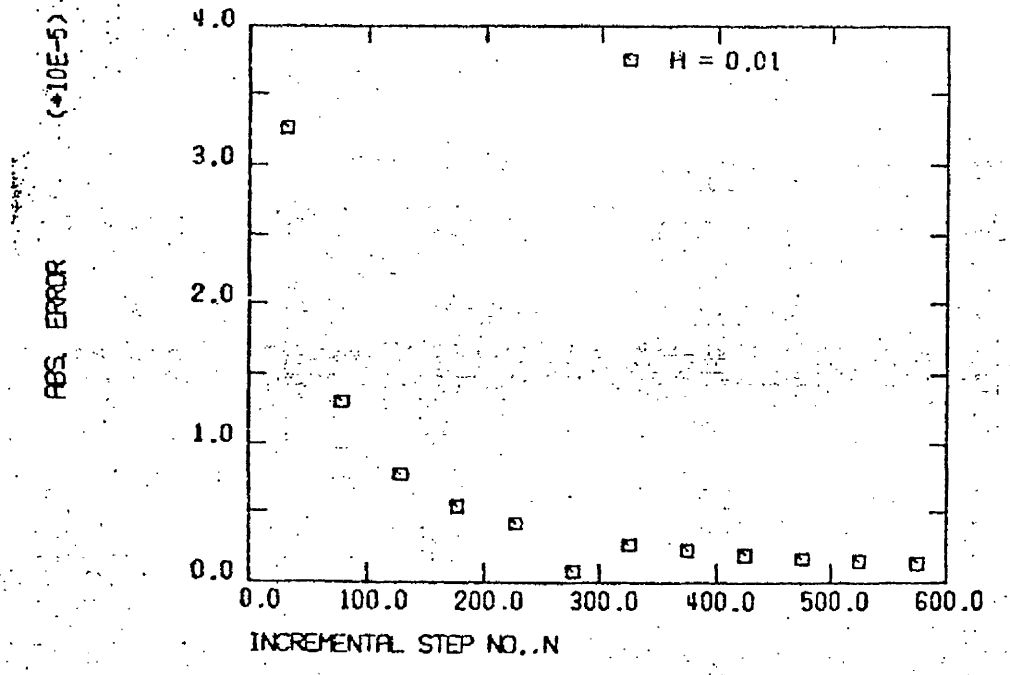


FIG. 8.2 ABS. ERROR VS. INCREMENTAL STEP NO..N

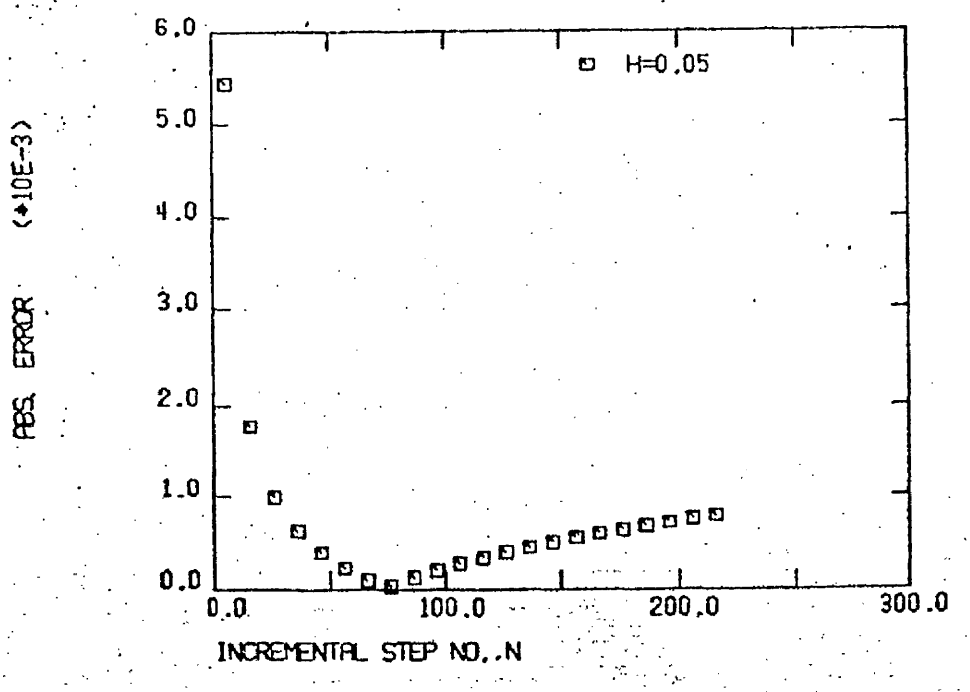


FIG. 8.3 ABS. ERROR VS. INCREMENTAL STEP NO..N

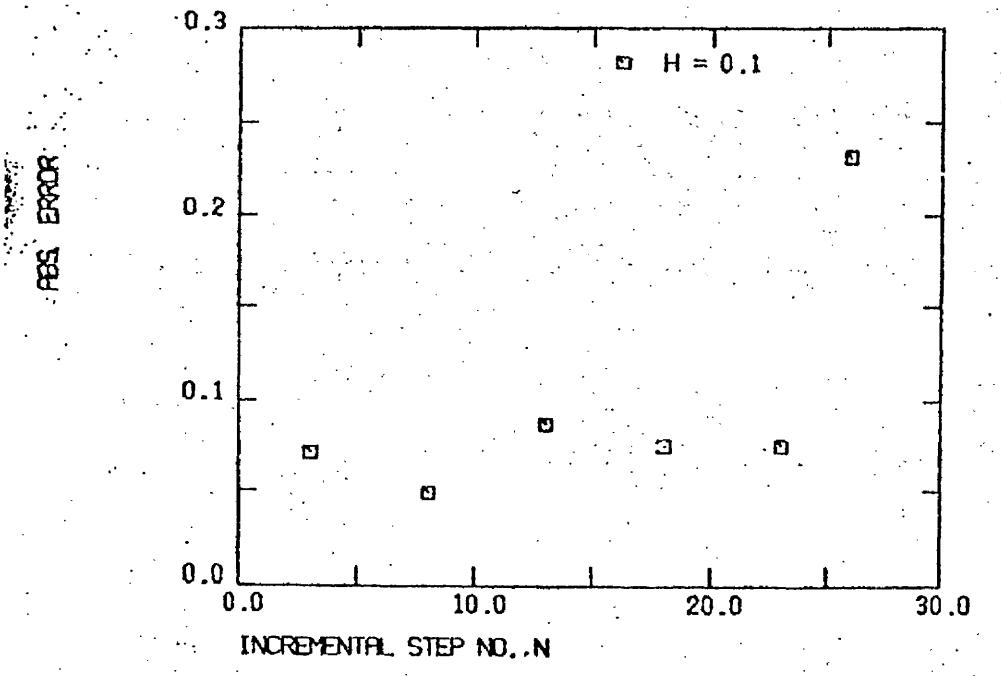


FIG. 8.4 ABS. ERROR VS. INCREMENTAL STEP NO..N

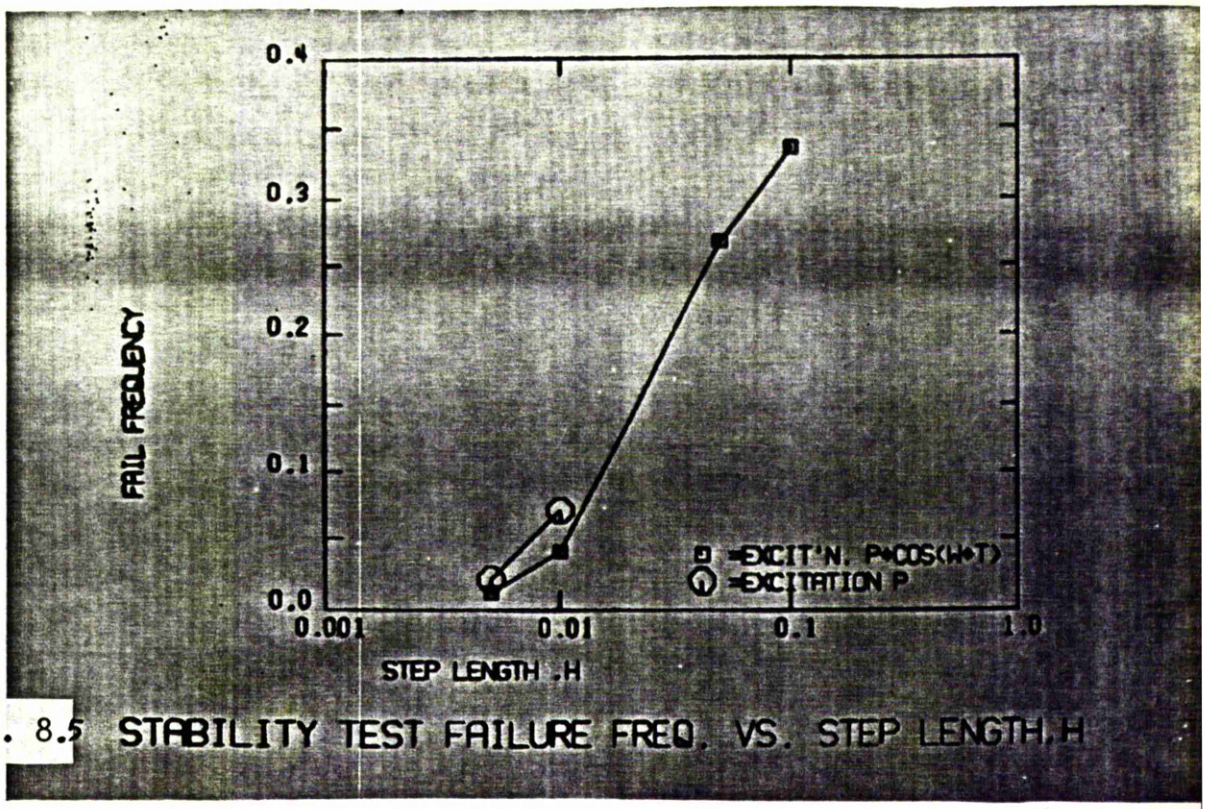


FIG. 8.5 STABILITY TEST FAILURE FREQ. VS. STEP LENGTH H

CHAPTER IX

THE ROAD VEHICLE SIMULATOR

1. Introduction

So far the road profile generator and dynamic vehicle integrator have been treated as two quite separate entities. The road profile generator is used to obtain discretised realisations of road surface excitation at spatially equal increments. The dynamic vehicle integrator is essentially orientated to modelling the response of the vehicle while using time as the independant variable. This chapter is devoted to the marriage of these two model segments. This marriage is not without difficulty. These difficulties are discussed together with the techniques employed in overcoming them. The chapter ends with a brief description of the end product - a road vehicle simulator program.

2. Interfacing the integrator and generator

The road profile generator, described in Chapter III outputs a displacement $Z(x)$ and a velocity $z'(x)$ are generated at discretised distance (or spatial) lag increments $h(x)$, $h(x)$ being held constant throughout the entire realisation process. The dynamic integrator models a time based differential equation idealisation of the dynamic system, and takes the form

$$m\ddot{y}(t) + c\dot{y}(t) + ky(t) = c\dot{z}(t) + kz(t). \qquad (2.1)$$

Unless a constant traverse velocity is maintained by the road vehicle throughout the entire realisation an incompatibility exists between the time and spatial based segments of the simulator. For compatibility either a common spatial or time base must be adopted.

2.1. A time based simulator

To enforce a common time base throughout both segments of the simulator, either the generator must, in some way, be transformed to ensure discretised profile points are generated at regular time increments,

or alternatively, the spatially based profile points must be interpolated in accordance with the instantaneous vehicle traverse velocity to ensure constant time increments are input to the integrator.

To transform the generator in such a way so as to ensure constant time increments is clearly impossible when the requirements of the AR filter are recalled. As the AR filter is obtained from a spatial road profile correlation, the realisation points generated at constant increments must also be spatially based. Only under conditions of constant traverse velocity can both spatial lag increments $h(x)$ and time lag increments $h(t)$ be held constant, since the relationship between them is given by

$$h(x) = \dot{x}(t)\, h(t) + \tfrac{1}{2} \ddot{x}(t)\, h^2(t) \qquad (2.2)$$

Consequently, as variable traverse velocity are of prime concern in this project this approach must be discounted.

The alternative time based approach makes use of interpolation. However as the intention would be to interpolate between random data points a least squares technique would have to be employed. As this technique is cumbersome in use an alternative was sought.

2.2. A spatially based simulator

The alternative to a time based simulator is to focus attention on the dynamic integrator and transform into a spatially based form.

To achieve this transformation, the differentials $\dot{y}(t)$ and $\ddot{y}(t)$ of equation (2.1) can be expressed in the following manner.

$$\left.\begin{aligned}
\dot{y}(t) &= \frac{dy(t)}{dt} = \frac{dy}{dx}\,\frac{dx}{dt} = y'(x)\,\dot{x}(t) \\
\text{and}\quad \ddot{y}(t) &= \frac{d}{dt}\left(\frac{dy(t)}{dt}\right) = \frac{d}{dt}\left(\frac{dy}{dx}\,\frac{dx}{dt}\right) \\
&= \frac{d}{dt}\left(\frac{dy}{dx}\right)\frac{dx}{dt} + \frac{dy}{dx}\,\frac{d^2x}{dt^2} \\
&= \frac{d}{dx}\,\frac{dx}{dt}\left(\frac{dy}{dx}\right)\frac{dx}{dt} + \frac{dy}{dx}\,\frac{d^2x}{dt^2} \\
&= \left(\frac{dx}{dt}\right)^2 \frac{d^2y}{dx^2} + \frac{d^2x}{dt^2}\,\frac{dy}{dx} \\
&= \dot{x}^2\, y''(x) + \ddot{x}\, y'(x)
\end{aligned}\right\} \qquad (2.3)$$

where $\dot{y}(x) = \frac{dy}{dx}$

$\dot{x} = \frac{dx}{dt}$ - the traverse velocity

and $\ddot{x} = \frac{d^2x}{dt^2}$ - the traverse acceleration

In an identical manner

$$\dot{z}(t) = \dot{x}(t)\, \dot{z}(x) \qquad (2.4)$$

Substitute (2.3) and (2.4) in (2.1) to obtain

$$\left[m\dot{x}^2\right] \ddot{y}(x) + \left[m\ddot{x} + c\dot{x}\right] \dot{y}(x) + k\, y(x) = \left[c\dot{x}\right] \dot{z}(x) + k\, z(x) \qquad (2.5)$$

Henceforth, the square bracketed parameters of (2.5) shall be referred to as the "pseudo dynamic parameters" of the spatially based DE.

Equation (2.5) is the spatially based formulation of the DE (2.1). The input excitations $z(x)$ and $\dot{z}(x)$ can be generated at constant spatial increments by the profile generator. The dynamic integrator can accommodate such input excitation provided the pseudo dynamic parameters of (2.5) are continually updated in accordance with the vehicle traverse velocity and acceleration.

Updating the pseudo parameters in accordance with $\dot{x}$ and $\ddot{x}$, has the effect of altering the decay constants of the DE (2.5). As far as the integrator algorithm is concerned, altering these dynamic parameters is equivalent to altering the incremental step length, h (see Chapter VII). The chosen integrator is amenable to such step length changes.

The spatial formulation was the technique adopted for the vehicle simulator, the dynamic parameters being continually updated as the realisation process proceeds. The difficulties which arise are now discussed.

3. Singularity points during simulation

Since the pseudo dynamic parameters of the spatial equation (2.5) are dependant on traverse velocity $\dot{x}$ and acceleration $\ddot{x}$, then

under certain specific conditions some of these coefficients might disappear and alter the order of the DE. Such singularity points together with their methods of solution are now discussed.

When both $\dot{x}$ and $\ddot{x}$ are zero, the problem is reduced to one of static equilibrium and is of no further concern in this project.

When $\dot{x} = 0$ and $\ddot{x} \neq 0$, equation is reduced to the first order DE, (2.5)

$$m\ddot{x}\, \acute{y}(x) + k\, y(x) = k\, z(x) \tag{3.1}$$

Typically, this situation is encountered initially when a vehicle accelerates from rest, or subsequently under variable acceleration/deceleration conditions when the velocity is momentarily zero.

Recall from chapter VII, the integrator treats a second order DE as pair of coupled first order equations, but under the singularity condition a single order DE requires solution. This dramatic transition contravenes the conditions for reasonable continuity required by the integrator. Special techniques must be employed to take account of these odd singularity points. It is impossible to effect a restart as restarts require continuity over several points. The singularity exists for isolated points only irrespective of the incremental step length. Consequently several alternatives were considered.

The simplest alternative is to replace the zero traverse velocity with a small positive value, and effectively banish the singularity from the problem. Reasonable results can be expected provided the inequality

$$\dot{x}_0 \ll \omega_n / K_{min}\,, \tag{3.2}$$

where $\dot{x}_0$ = replacement positive traverse velocity,

K_{min} = shortest duration angular wave number present, in the road profile,

ω_n = natural angular frequency of the road vehicle,

holds. Under such conditions, the frequency of the excitation energy is far removed from the vehicle's resonant frequency and should therefore be a reasonable approximation to the singularity condition.

A more rigorous alternative is to implement a Taylor Series extrapolation from the point of singularity to the next discrete time increment where the second order DE is again in force. Instability is no problem as the Taylor extrapolation is used only once and not recursively. The higher derivative response quantities required by the Taylor extrapolation can be obtained from the singularity equation (3.1).

It was this latter Taylor Series approach which was implemented.

4. Updating the parameters of the spatial based DE

It is assumed that the pattern of traverse acceleration $\ddot{x}(t)$ is explicitly defined. From $\ddot{x}(t)$, the spatial distance, $x(t)$ ($= n\,h(x)$), travelled along the road profile can be determined. The time varying quantities in the spatial DE can thus be determined at any spatial increment number, n. Their determination is accomplished in the following manner.

The incremental time step is given by

$$h_n(t) = \left[-\dot{x}_{n-1}(t) + \sqrt{\dot{x}^2_{n-1}(t) + 2\,\ddot{x}_n(t)\;h(x)} \right] \Big/ \ddot{x}_n(t) \qquad (4.1)$$

while the traverse velocity is given by

$$\dot{x}_n(t) = \dot{x}_{n-1}(t) + \ddot{x}_n(t)\;h_n(t) \qquad (4.2)$$

5. Interpolative matching of space and time

In Chapter VIII a step length ratio $h(t)/T$ of 0.01 was empirically observed to be nearly optimal. This observation was made with fixed dynamic parameters in the DE. As discussed in the previous sections, conditions of variable traverse velocity cause these parameters to continually change. As constant spatial increments $h(x)$ are output by the profile generator, then at low traverse velocities the step length ratio $h(t)/T$ is in danger of exceeding the 0.01 bound. To overcome this difficulty the following interpolation procedure was implemented to establish the need to interpolate

and the number of interpolated points required.

a) Interpolate if ratio $h(t)/T > 0.005$. The 0.005 was chosen to ensure the interpolated step length ratio never exceeded 0.01.

b) Calculate number of interpolated points
$N_{int} = INT\left(h(t)/0.005T \right)$
where INT (x) is the nearest integer value to x.

c) Calculate interpolated step length
$h_{Int}(t) = h(t)/N_{int}$

d) Calculate interpolated vertical displacement and velocity excitations as follows
$z_{int,\,k}(x) = \left(k\, z_{n+1}(x) + (N_i - k)\, z_n(x) \right)/N_i$
for the k^{th} interpolated point between the n^{th} and $(n+1)^{th}$ generated road profile points. Do likewise for velocity $\dot{z}(x)$

e) Calculate interpolated road traverse velocity
$\dot{x}_{int,\,k}(t) = \dot{x}_{int,\,k-1}(t) + \ddot{x}_{int,\,k}(t)\; h_{int}(t)$
(Recall road traverse acceleration $\ddot{x}(t)$ is explicitly and externally defined.)

This interpolation procedure should only be required when $\dot{x}(t)$ approaches zero. Under such circumstances

$$h(t) \simeq \sqrt{\left(2\, h(x)/\ddot{x}(t)\right)} \qquad (5.1)$$

and unless traverse acceleration $\ddot{x}(t)$ is large, the step length ratio can easily exceed the 0.01 bound. (With uniform acceleration this is only applicable when the vehicle starts from rest.)

6. Random response conditions at start up

Irrespective of whether the road vehicle commences the non-stationary traverse from rest or from a uniform traverse velocity, both are cases of steady state response initial conditions. As the ultimate objective is to converge on the mean square response condition,

there can be little objection to using the root mean square value of these steady state initial conditions as replacements for random initial conditions.

7. Extension to non linear vehicle problems

In Chapter VII the ability of the dynamic integrator in handling time dependant DEs, coupled or otherwise, was demonstrated. The extension to the spatially based non linear DE pair will now be demonstrated.

The non linear time based equivalent of Chapter VII (8.2) is given by

$$\dot{y}_m^{(P)} = \frac{1}{a_N} \left\{ f^{(p-1)} \left(y_m^{Q}, y_m^{Q-1}, \ldots . y_m, t_m \right) - \sum_{i=1}^{N-1} a_i \left(\dot{y}^{(p-1)} \right)^i \right\}^{1/N} \qquad (7.1)$$

where m is the step increment number,
p is the correction number,
N is the power of the non linearity in $\dot{y}$,
and Q is the power of the non linearity in y.

In the two coupled first order DEs of the vehicle model, (7.1) becomes

$$\dot{y}_m^{(P)} = v_m$$

$$\dot{v}^{(P)} = \frac{1}{a_N} \left\{ f(v_m^{R}, v_m^{R-1}, \ldots, v_m, y_m^{Q}, y_m^{Q-1}, \ldots, y_m, t_m) - \sum_{i=1}^{N-1} a_i \left(\dot{v}^{(p-1)} \right)^i \right\}^{1/N} \qquad (7.2)$$

where, additionally in this case, R is the power of the non linearity in velocity, v.

To obtain a spatially based formulation, it is simply a matter of

substituting relations (2.3) and (2.4) into the above equation (7.2). The extension to non linear spatially dependant DEs becomes trivial.

8. An overview of the road vehicle simulator program

Having presented the various design aspects of the digital road vehicle simulator, a general overview of the flow logistics of the simulator program are now presented.

The overall flow logistics of the simulator are shown schematically in fig. 9.1. The road profile generator remains unchanged from the description given in Chapter III. The start up vehicle response conditions simply use the initial steady state expected values described in section 6. Evaluation of the current time based incremental step length, $h_n(t)$, is obtained from (4.1). A decision is made, based on the value of, $h_n(t)$, as to whether interpolation is necessary (section 5). The pseudo dynamic parameters of the spatially based DE (2.5) are next evaluated ready for use by the integrator module. This module makes use of both interpolated and non interpolated values in an identical manner, however if interpolation is required the module is re-employed "N_{int}" times between the $(n-1)^{th}$ and n^{th} points on the generated road profile.

The flow logistics of the spatial integrator module are shown in more detail in fig. 9.2. Most values input to this module refer to the new increment, n, however the old response values of $y(x)$, $y'(x)$ and $y''(x)$ at increment (n-1) are also input. A check for the occurrence of the singularity condition (see section 3) in the spatial DE is implemented. The occurrence of this singularity activates a single step Taylor's Series extrapolation followed by the integrator start up procedure. Otherwise an ordinary integrator step is implemented. The implementation of the integrator step and start up procedures have already been described in Chapter VII, sections 3 and 6.

9. Conclusion

Time dependant second order DEs of the vehicle model can be reformulated as spatially dependant DEs with "pseudo dynamic parameters".

These parameters are dependant on the instantaneous traverse velocity and acceleration. Formulating the DE on a spatial basis ensures compatability with the road profile generator, which generates road profile points at constant spatial increments. This spatial approach is not without difficulty, tests being required for singularity points and for the decision of interpolating between generated profile points. It is recommended that the random response initial conditions be replaced by the steady state expected values.

The ease with which this spatially based road vehicle simulator is extended to include non linear suspension configurations is demonstrated.

The various design considerations can be assembled to form the road vehicle simulator program, the flow logistics of which are explained in this chapter.

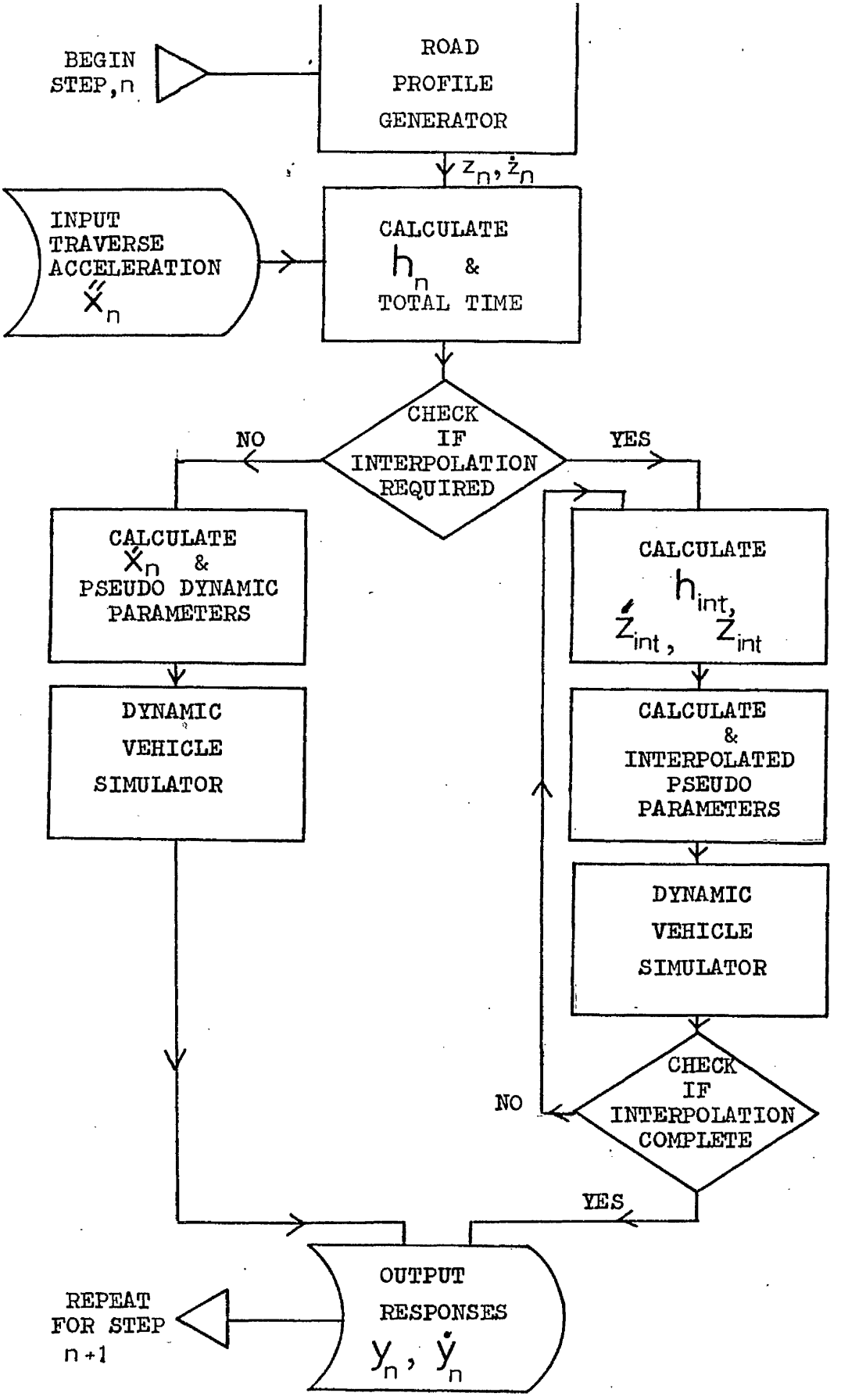


FIG 9.1 Program flow logistics of Dynamic Vehicle Simulator

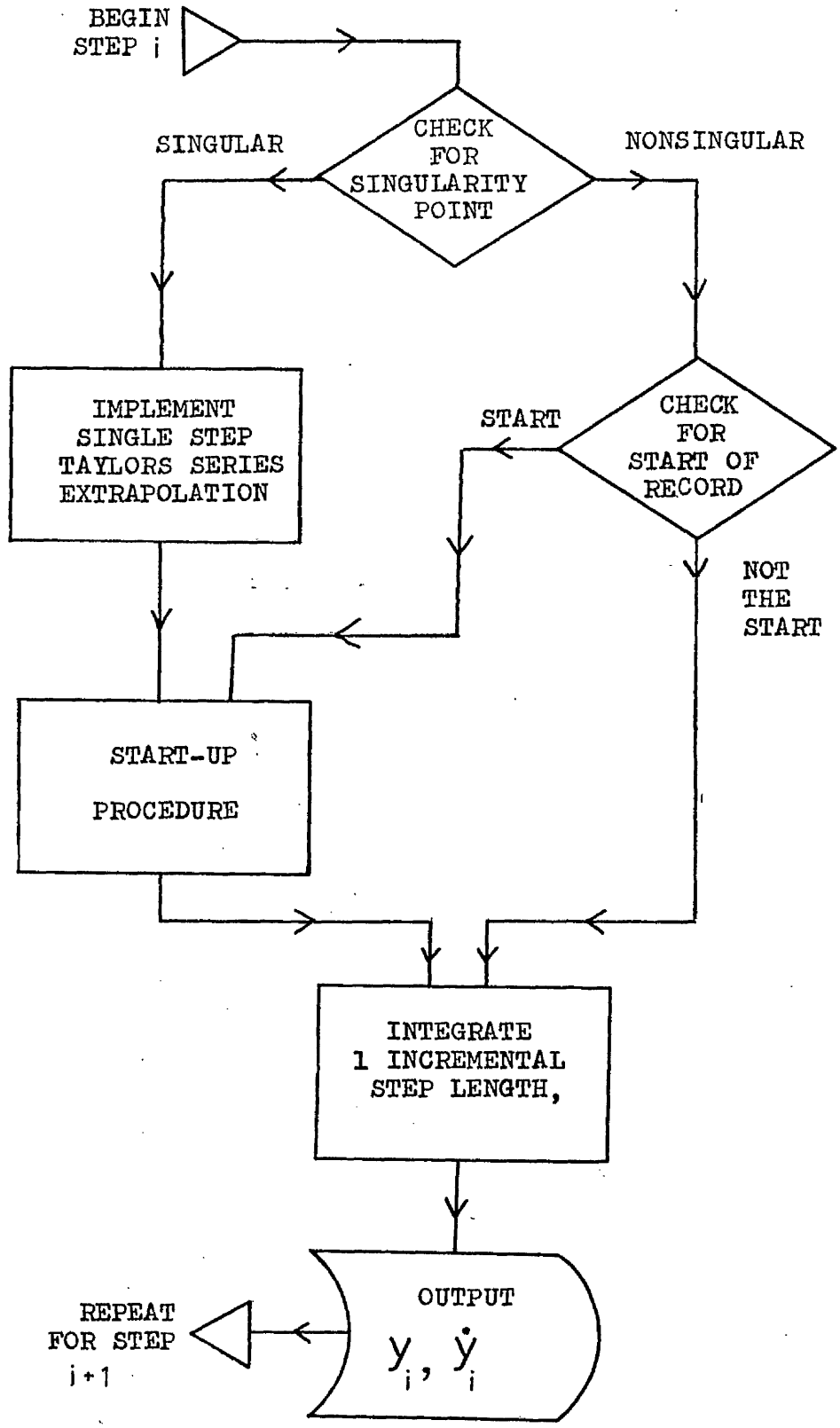


FIG 9.2 Program flow logistics of the spatially based Dynamic Vehicle Integrator.

CHAPTER X

THE ROAD VEHICLE SIMULATOR IN USE

1. Introduction

The objectives of this chapter are twofold. Firstly the ability of the road vehicle simulator in dealing with random excitation is considered. The other concerns the establishing of new non-stationary results.

The simulator comprises of two basic modules, a road profile generator and a dynamic vehicle integrator. Both modules have already been tested separately (Chapter III and Chapter VIII respectively). Combining the modules into a complete vehicle simulator requires testing of a statistical nature. For this purpose both stationary (Stat) and non-stationary (N-Stat) tests are considered separately.

The first phase of statistical testing concerned the Stat (or constant traverse velocity) case. Such testing was included to ensure the complete simulator could adequately handle random excitation. The road profile generators' statistical simulation capabilities had already been tested and consequently effort was concentrated on the dynamic simulator. As both excitation and response spectra can be obtained from the simulation, it is a straight forward matter to check the results using classical random vibration theory.

Phase two of statistical testing concerned the variable traverse velocity (N-Stat) case. It is the problem of ultimate concern to this project. Undertaking tests on N-Stat random processes is much more difficult. No closed form solutions exist for the purpose of vindication. Reliance must be placed on comparisons with other numerical evaluations of N-Stat characteristics. (Ref. 1 and 2 - Both solutions were established by numerically evaluating the displacement response correlation integrals).

The approach adopted in this project concerns the statistical assimilation on non-stationary response realisations, and consequently no deterministic answer can be expected. The inapplicability of the ergodicity principle serves to worsen this problem.

Instead, the technique of sample averaging at single instants is applied to a large sample of independant realisations.

As a natural by-product, this approach yields additional information as to the velocity and acceleration response characteristics of the vehicle. No alternative solutions exist for comparison. However by the nature of the technique applied, it is sufficient to verify the displacement characteristics to ensure viable derivative responses.

Information on the probability distribution of the responses is also available. Once again this information is not available by the other approach.

Complications arising from this method of sample averaging are discussed, including problems associated with sample size and statistical scatter.

The chapter closes with a short discussion on the usefulness of the N-Stat information obtained and suggests avenues worthy of more detailed examination.

2. Stationary Random Testing

Stationary tests were introduced to vet the dynamic simulator's ability to handle random type excitation from the profile generator. To this end it was sufficient to compare the simulation response spectra with classical vibration theory.

The road profile is a stationary random process with respect to traverse distance (x). The stationarity or non-stationarity with respect to time (t) of the vehicle's input excitation is dependant on the vehicle's traverse velocity ($\dot{x}$) profile. At any constant $\dot{x}$, this vertical input excitation is a stationary random process. This fact is easily demonstrated as follows.

The auto-correlation function $R_{z(x)}(L)$ of the road profile vertical displacement $z(x)$ as a function of traverse distance x is defined by

$$R_{z(x)}(L) = \langle z(x) \; z(x + L) \rangle \qquad (2.1)$$

where L is the spatial lag.

The corresponding spatial power spectral density, $S_{z(x)}(K)$, as a function of angular wave number, K, is given by

$$S_{z(x)}(K) = \frac{1}{2\pi}\int_{-\infty}^{\infty} R_{z(x)}(L) \exp(-i K L)\, dL \qquad (2.2)$$

where by definition $K = 1/\lambda$, λ is the spatial wave length, and i is the imaginary operator.

In contrast, the road vehicle's response is a function of time based angular frequency, ω. At constant traverse velocity, $\dot{x}$, the relationship between traverse distance, x, and time, t, is simply

$$x = \dot{x}\, t \qquad (2.3)$$

It follows that the relation between time lag, T, and spatial lag, L, is simply

$$\left.\begin{aligned} L &= \dot{x}\, T \\ \text{and } dL &= \dot{x}\, d\, T \end{aligned}\right\} \qquad (2.4)$$

The relationship between ω and $\dot{x}$ is given by

$$\dot{x} = \lambda\, \omega \qquad (2.5)$$

It follows therefore that

$$\omega = \dot{x}\, K \qquad (2.6)$$

Combining formulae (2.1) through (2.6) yields, at constant $\dot{x}$

$$S_{z(x)}(K) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \langle z(t)\ z(t+T) \rangle \exp(-i\ \omega\ T)\ \dot{x}\, dT$$

$$\text{or} \quad S_{z(x)}(K) = \dot{x}\, S_{z(t)}(\omega) \qquad (2.7)$$

Equation (2.7) demonstrates the simple relationship which exists at constant $\dot{x}$, between spatial and frequency spectra.

Making use of the classic relation between input, $S_{z(t)}(\omega)$, and response $S_{y(t)}(\omega)$ spectra

$$\text{ie.} \quad S_{y(t)}(\omega) = |\alpha(\omega)|^2\, S_{z(t)}(\omega) \qquad (2.8)$$

where $\alpha(\omega)$ is the receptance,

[For a one degree of freedom system,

$$|\alpha(\omega)|^2 = \frac{\omega_n^4 + 4(\zeta\omega\ \omega_n)^2}{(\omega_n^2 - \omega^2)^2 + 4(\zeta\omega\ \omega_n)^2}$$

- usual notation applies.]

yields the relation

$$S_{y(t)}(\omega) = \frac{1}{\dot{x}} |\alpha(\omega)|^2 S_{z(x)}(K) \qquad (2.9)$$

It is equation (2.9) which is used to verify the validity of the stationary simulations.

Three correlations were tested; Virchis, Exp, and Mod Exp (see chapter IV for their definition). Fig. 10.1 shows typical results of plotting the spectral ratio $\dot{x}\, S_{y(t)}(\omega) / S_{z(x)}(K)$ against wave number ratio K/K_o where K_o is the lowest recorded wave number. Ideally this spectral ratio should adhere to $|\alpha(\omega)|^2$ - also plotted in Fig 10.1. The plotted example is of the Exp correlation, with $h(x) = 0.5$, $\dot{x} = 22_{f.p.s.}$, and $B = 0.1284 f_{t.}^{-1}$ The spectral ratios for a single realisation and over an average of 6 independant realisations are shown.

As with Exp correlation of Fig. 10.1, good results were achieved with the other above mentioned correlations.

All spectral results were obtained using the technique of Fast Fourier Transform (F.F.T.). Each realisation record comprised 1024 (ie 2^{10}) data points. To eliminate the effects of transience only the latter 512 (ie 2^9) data points were considered. Consequently, the lowest recorded angular wave number, K_o is given by

$$K_o = \frac{1}{\lambda_o} = \frac{1}{512\ h(x)} \qquad (2.10)$$

where $h(x)$ is the incremental spatial increment.

3. Non-Stationary Random Testing

3.1 Introduction

Effectiveness of the simulator in handling stationary random data, does not automatically guarantee similar ability in non-stationary simulation. Recall non-stationarity is introduced into the simulation by varying the integrator's dynamic parameters with changes in traverse velocity, $\dot{x}$. Consequently further testing is essential.

The concept of sample averaging a series of time realisations is not new. Normally the ergodicity principle overcomes the need for such a course of action. The practical limitations of sample averaging must therefore be learned as the tests proceed.

3.2 The Test Correlations

Effort was concentrated on the following two road profile correlations.

a) The Exponential (Exp) Correlation.

This correlation is expressed as

$$R(L) = \exp(-B|L|) \qquad (3.1)$$

where $B = 0.1284$.

The Exp correlation is the one considered in ref. 2. B is a dimensional constant which in this case is expressed in feet (cf. Sobczyk/Macvean 1976 where the unit of B is metres).

b) The Modified Exponential (Mod Exp) Correlation.

The definition of this correlation is given by

$$R(L) = (1 + B|L|)\exp(-B|L|) \qquad (3.2)$$

where $B = 0.2247$ - also expressed in feet.

In both cases the spatial lag, L, is given by

$$L = n\,h(x)$$

where $n = 0, 1, \ldots, 4$

and $h(x) = 1.0$ feet.

Initially the vehicle starts from rest ($\dot{x} = 0$) and accelerates uniformly such that $\ddot{x} = 0.2g$.

3.3 The sample averaging technique

At this point it is necessary to define what is meant by sample averaging.

A specific time lapse, t_r, after the commencement of the i^{th} realisation history, an instantaneous or point value $y_i(t_r)$ is recorded. On another independant $(i + 1)^{th}$ realisation, the value $y_{i+1}(t_r)$ is recorded. By recording the realised point value $y_i(t_r)$ across a large number of realisations - the sample - it is possible to establish the average value after elapse instant, t_r. As it is mean square values which are of interest in this study, then let the sample average (SA) mean square (MS) value (SAMS) of the quantity $y(t_r)$ be defined as follows.

$$\text{SAMS}\left(y(t_r)\right) = \frac{1}{N} \sum_{i=1}^{N} y_i^{\,2}(t_r) \qquad (3.3)$$

where N is the total number of realisations in the sample.

The SAMS values are kept for all recorded time instants, t_r.

3.4 The Sample Size Tested

At the outset it was impossible to predetermine the sample size which would reveal meaningfully accurate statistical results. Irrespective of the probability distribution under consideration, confidence level is normally inversely proportional to $\sqrt{N}$. At this stage with this particular non-stationary problem it is very difficult to be more specific.

Constraints were placed on the amount of sample data which could be stored for subsequent interpretation. Initially, it was also desirable to keep the realisation data in an unprocessed form. Use of a data condensation technique could cause the loss of information which on hind-sight might prove meaningful.

Each realisation comprised of 1025 sampled data points. At each one, the values of inputs $z(x)$, $\acute{z}(x)$ and responses $y(x)$, $\acute{y}(x)$, $\acute{\acute{y}}(x)$ require storage. For 50 independant realisations, then the required

computer storage is in excess of 0.25×10^6 words ($\frac{1}{2}$ megabyte). Computer restrictions dictated a compromise. The somewhat arbitrary compromise decision is described below.

Five adjacent data points at 20 point intervals were saved on every realisation. A 20 point interval was considered reasonable bearing in mind an order in the AR filter of 4 and a 5^{th} order integrator formula. The adjacent points were stored to empirically monitor the effects on sample convergence by smearing the mean square results over 5 values. (In fact the notion of smearing proved totally fruitless and will therefore not be discussed further.)

The curtailment of data as described enabled a five fold reduction to be made.

3.5 Statistical characteristics of the sampled input

Both input displacement and velocity excitations are stationary random processes with respect to space. SAMS input data is sampled at fixed spatial increments. To achieve this, every 25^{th} data point is sampled.

In Fig. 10.2 and 10.3, the SAMS displacement inputs for both correlations are shown plotted against traverse velocity, $\dot{x}$. Similar plots are shown in Fig. 10.4 and 10.5 of the SAMS velocities. It is worth emphasising that input velocities (ie displacement gradients) are measured with respect to the space domain. If these quantities were measured with respect to time, their SAMS values would increase with $\dot{x}$. All plots (Fig. 10.2 - 10.5) have been normalised by dividing by the "averaged mean" input value; displacement by "averaged mean" displacement, and velocity by velocity. (In future the term "averaged MS" value should be interpreted as the mean square value averaged throughout each entire realisation record, and across all 50 members of the realisation sample. This concept is only applicable to stationary phenomena ie spatial input excitations.) In each of the plots considerable scatter is observed about the "averaged MS" value line of unity. The ratios, velocity/displacement of "averaged MS" input values for both correlations are shown in Table 10.1.

Also note from Fig. 10.2 - 10.5, the SAMS value can deviate by as much as 40% from the "averaged MS" value.

3.6 Determination of equivalent stationary MS response integrals

Preliminary to estimating the N Stat MS response characteristics, the Stat counterparts should be evaluated for varying traverse velocity, $\dot{x}$. The Stat case will act as datum.

Recall from (2.7) the time based input spectrum can be expressed in terms of the spatial road profile spectrum $S_{z(x)}(K)$ and $\dot{x}$, where $S_{z(x)}(K)$ is given by (2.2)

First, evaluate $S_{z(x)}(K)$.

For the Exp Correlation,

$$S_{z(x)}(K) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \exp(-B\,L)\ \exp(-iK\,L)\ dL$$

$$= -Re\left[\exp\left(-(B + iK)\,L\right)/\pi(B + iK)\right]_{L=0}^{\infty}$$

$$= B\,/\,\pi(B^2 + K^2)$$

$$= B\,\dot{x}^2\,/\,\left((\dot{x}B)^2 + \omega^2\right) \qquad (3.4)$$

since $K = \omega/\dot{x}$

For the Mod Exp Correlation

$$S_{z(x)}(K) = \frac{1}{2\pi}\int_{-\infty}^{\infty}(1 + B\,L)\exp(-B\,L)\exp(-iK\,L)\,dL$$

$$= -Re\left[\frac{\exp\left(-(B + iK)\,L\right)}{\pi(B + iK)}\left(1 + B\,L + \frac{B}{B + iK}\right)\right]_{L=0}^{\infty}$$

$$= 2B^3/\pi(B^2 + K^2)$$

$$= \frac{2\dot{x}^4 B^3}{\pi\left((\dot{x}B)^2 + \omega^2\right)} \qquad (3.5)$$

Using (2.7) in conjunction with

spectra $S_{z(x)}(K)$, the stationary MS responses, $\langle y^2 \rangle$ etc., can be evaluated by means of the relationship

$$\langle y^2 \rangle = \int_0^\infty \|\alpha(\omega)\|^2 \, S_{z(x)}(K) \, dK \qquad (3.6)$$

Thus for the Exp Correlation the response MS displacement is given by

$$\langle y^2 \rangle = \frac{1}{\pi} \int_0^\infty \|\alpha(\omega)\|^2 \, \frac{B\dot{x}}{(B\dot{x})^2 + \omega^2} \, d\omega \qquad (3.7)$$

While for the Mod Exp Correlation, the responses are

$$\langle y^2 \rangle = \frac{1}{\pi} \int_0^\infty \|\alpha(\omega)\|^2 \, \frac{2(B\dot{x})^3}{\left((B\dot{x})^2 + \omega^2\right)^2} \, d\omega \qquad (3.8)$$

$$\langle \dot{y}^2 \rangle = \frac{1}{\pi} \int_0^\infty \|\alpha(\omega)\|^2 \, \frac{2(B\dot{x})^3 \omega^2}{\left((B\dot{x})^2 + \omega^2\right)^2} \, d\omega \qquad (3.9)$$

$$\langle \ddot{y}^2 \rangle = \frac{1}{\pi} \int_0^\infty \|\alpha(\omega)\|^2 \, \frac{2(B\dot{x})^3 \omega^4}{\left((B\dot{x})^2 + \omega^2\right)^2} \, d\omega \qquad (3.10)$$

In the case of the Exp Correlation, no derivative response (ie $\langle \dot{y}^2 \rangle$, and $\langle \ddot{y}^2 \rangle$) exists. This follows from the fact that the Exp Correlation is not physically realisable, being non-differentiable at zero lag.

The evaluation of these even infinite fractional integrals (3.7 - 3.10) is tedious and is achieved in the following manner.

Express the integral, I_n, in the complex form

$$I_n = \frac{1}{2\pi} \int_0^\infty \frac{g_n(i\omega)}{h_n(i\omega) \cdot h_n(-i\omega)} \, d\omega$$

where $h_n(x) = a_0 x^n + a_1 x^{n-1} + \ldots\ldots + a_n$

and $g_n(x) = b_0 x^{2n-2} + b_1 x^{2n-4} + \ldots\ldots + b_{n-1}$

Once coefficients a_j and b_j are determined for all j, I_n can be evaluated as follows:

$$I_n = \frac{(-1)^{(n+1)} N_n}{2a_0 \quad D_n}$$

where

$$\text{determinant} \quad D_n = \begin{vmatrix} d_{11} & d_{12} & & d_{1n} \\ d_{21} & & & \cdot \\ & & & \cdot \\ & & & \cdot \\ d_{n1} & \cdot & \cdot & \cdot \end{vmatrix}$$

$d_{mr} = a_{2m - r}$, and $a_s = 0$ for $s < 0$ and $s > n$.

Determinant, N_n, is obtained from D_n by replacing the entries of column 1 (d_{11} d_{12} . . . d_{1n}) by (b_0 b_1 . . . b_{n-1}) respectively (ref. 3).

The evaluation of these stationary MS integrals is shown graphically in fig. 10.6 - 10.9. In fig. 10.6 and 10.7 the results are shown plotted beside their non-stationary counterparts.

The mean square response expressions of equations (3.9) and (3.10) are differentials with respect to time. The differential MS responses recorded from the sample averaged results are spatially based. To allow a comparison (3.9) and (3.10) can be readily transformed to a spatial (x) base by utilising equations (2.3) of Chapter IX.

From the relation $\dot{y} = \dot{x} y'$

the transformation $\langle y'^2 \rangle = \frac{1}{\dot{x}^2} \langle \dot{y}^2 \rangle$

is obtained, while $\ddot{y} = \dot{x}^2 y'' + \ddot{x} y'$

yields $\langle y''^2 \rangle = \frac{1}{\dot{x}^4} \langle \ddot{y}^2 \rangle$

under stationary conditions where $\ddot{x} = 0$.

In fig. 10.10 and 10.11 these spatially based derivative responses are shown plotted beside their non-stationary counterparts.

3.7 Non-Stationary MS Response Characteristics

The non-stationary results obtained by sampling the realisations are now discussed. All results are normalised with respect to the "averaged MS" input displacement.

In fig. 10.6 and 10.7 the variation in MS displacement response is shown plotted against $\dot{x}$, for both Exp and Mod Exp Correlations. The stationary counterparts are included for comparison. It is difficult to establish a criterion for quantitative assessment of the N-Stat results, however, the range of scatter observed is only marginally greater than that observed with the N-Stat input displacement (see Table 10.2). If the true N-Stat responses lie above the unity line of "averaged MS" displacement, increased scatter must be expected.

It was demonstrated in ref. 2 that for the Exp Correlation with typical traverse acceleration of 0.2g, only marginal deviation (1 - 2%) below the stationary result was recorded. Clearly such minute variation is beyond the resolution of this sampling technique unless for extremely large sample sizes. The N-Stat results of Fig. 10.6 do nothing to contradict their conclusion for the Exp Correlation. The Mod Exp displacement results of Fig. 10.7 also agree within the bounds of resolution with their stationary counterparts. However, in this case no alternative solutions exist for comparison.

Derivative MS responses ($\langle \dot{y}^2 \rangle$, $\langle \ddot{y}^2 \rangle$) for the Mod Exp Correlation are shown plotted in Fig. 10.10 and 10.11 respectively. The results are in reasonable agreement with their stationary counterparts. All tend to zero asymptotically.

In Fig. 10.12 and 10.13, the N-Stat SAMS responses for the Exp Correlation are recorded. The plots are similar in form to the Mod Exp results. While the Exp Correlation is not physically realisable, it does serve to demonstrate that this simulation sampling technique can yield satisfactory results, despite making crude mathematical approximations to the displacement road profile correlation at the outset of analysis.

Note: Neither the correlation exponential decay constant, B, nor the traverse velocity, $\dot{x}$, act as independant variable parameters. The introduction of the product property $B\dot{x}$, would have reduced the number of the physical parameters by one. However, product $B\dot{x}$ was not introduced due to the very special nature of both correlations. Product $B\dot{x}$ would not have been generally applicable to other correlations, nor would it be possible to achieve a 'feel' for physically realistic parameters.

3.8 Probability Distribution

The probability distributions of the SAMS values at constant arbitrary time instants were established. Ideally for a large enough sample the input MS distribution should be Chi-Square with one degree of freedom. Both input and output instantaneous values from 50 independant simulated realisations were compared with this ideal.

In Fig 10.14 and 10.15, the Exp Correlation input and responses MS displacements respectively are shown compared with the ideal Chi-Square distribution. Fig. 10.16 and 10.17 do likewise for the Mod Exp Correlation.

Input and response MS velocity distributions for the Exp Correlation are plotted in Fig. 10.18 and 10.19, while Fig. 10.20 and 10.21 show the Mod Exp counterparts.

In Table 10.3, the maximum deviation data of all above mentioned MS probability distributions from Chi-Square are summarised for sample point 1000.

Given the limited sample size, the results compare closely to the Chi-Square distribution. It would appear the deviation of response MS values, as compared with the input deviation magnitudes are not significantly different. Therefore, it must be tentatively concluded that the output MS displacement and velocity conform to a Chi-Square distribution of degree 1, and hence output displacement and velocity

are normally distributed. By implication in a linear dynamic system the acceleration response must also adhere to a normal form.

Similar Chi-Square deviations were observed at other arbitrary instantaneous data points.

It is reasonable to presume that the quality of a Chi-Square distribution extracted from a 50 realisation point sample is at least as good as that obtained from a 50 point ergodic sample along a single realisation record. The justification for this hypothesis lies in the fact that when applying the ergodicity principle along a single time history, none of the sample data points are statistically independant.

In order to utilise this hypothesis to further verify the reasonable Chi-Square nature of the 50 point MS sample, probability distributions from independant ergodic samples were assembled from stationary input displacement realisations. The sample sizes tested were 50, 100, 200, 400, and 800 data points, each with the same sampling interval. The resultant distribution plots are shown compared with the theoretical Chi-Square ideal in fig. 10.22 - 26. The maximum deviations are recorded in Table 10.4. On comparing the ergodic results of 10.4 with the SA counterparts in Table 10.3, the latter (SA) distribution exhibits lower Chi-Square deviation than the ergodic sample of equivalent size. This result adds further weight to the belief that the non-stationary MS responses exhibit a Chi-Square distribution.

3.9 Statistical F - Tests

To further vindicate the reasonable nature of the sample averaged results an F - test was applied at arbitrarily chosen sample data points on both correlations. The 50 ensembled mean square points were divided into two independant samples (S_1 and S_2) each of size 25. The F - ratio (S_1/S_2) was calculated and a search made for values which exceeded the 5 and 95 percentile limits. To exceed these limits for two independant samples, each of size 25, the F - ratio must lie outside the range

$$0.5051 < \text{F - ratio} < 1.980.$$

Example results for sampled data point 125 are shown in Table 10.5. Only two F - ratio results (marked *) exceeded these limits - a result not beyond the bounds of statistical probability.

3.10 Implications derived from a single non-stationary realisation record

It is not always practicable to sample average a large number of independant realisations in order to establish non-stationary dynamic characteristics. Consequently it is useful to reflect on the amount of information which can be obtained from a single time history simulation.

Single non-stationary input and response realisations for both the Mod Exp and Exp Correlations are shown plotted against time in Fig10.27 and 10.28 respectively. In both cases, the maximum responses (shown dotted) can be seen peaking marginally above the input displacements. Any attempt at establishing the mean square values would require the realisations to be averaged over time, and would require the application of the ergodicity principle where stationarity is assumed. Even if time averaging was applied only in a piecewise manner, the quality of the non-stationary information would be impaired.

Information of probability distribution is also impossible, unless a time averaging technique is applied here also. However the results of section 3.8 (Probability Distribution) indicate that the response realisations adhere to a normal distribution within statistical limits. It is reasonable to assume this to be true of other linear systems undergoing similar non-stationary excursions. No such assumptions could be applied to non-linear systems where sample averaging is essential to yield this information.

Single velocity response realisations v time are shown plotted in Fig. 10.29 and 10.30 for the Mod Exp and Exp Correlations respectively. A marked decrease in response amplitude with increasing time is evident. The same trend occurs in the acceleration responses, Fig. 10.31 and 10.32. With acceleration, this decay is much more pronounced. All velocity and acceleration response realisations

bear strong resemblance to the decay characteristics of the SAMS results.

Clearly the next step is to attempt a least squares fit based on the MS values over the entire length of a single realisation history. It is straightforward in hind-sight to assume this curve should adopt a quadratic form. (This assumption follows naturally from the shape of the sample averaged MS response curves). Unfortunately, attempts at fitting curves to individual realisation records proved totally unsatisfactory. This was due solely to the extremely ill conditioned nature of the realisation data. However a more detailed study of the problems involved in curve fitting is required before the concept is dismissed outright. Had it proved possible, in the time available, to overcome these numerical problems, then it would have been necessary to establish a criterion for generalising the order of the least squares fit to other correlations and other traverse acceleration profiles. At present all that can be attempted on a single record is a crude envelope curve covering the peaks in the mean square response. This will certainly achieve a worst case estimate.

4. Conclusion

4.1 Stationary Random Tests

Testing the simulator's capability in handling stationary (constant traverse velocity) excitation proved straightforward. Response spectra were obtained which agreed well with classical vibration theory.

4.2 Non-Stationary Random Tests

(a) General -

Analysis was concentrated on the Exp and Mod Exp road profile correlations. The non-stationarity was induced by introducing constant acceleration into the problem.

The technique of sample averaging was applied in order to assimilate

the statistical data. At the outset it was difficult to ascertain which information could most usefully be retained from each simulation history for subsequent statistical interpretation. As a result of constraints on computer storage, a decision was reached to retain samples of all generated information. In hind-sight this was not the optimum decision. It would have been more beneficial to retain only the running total of sample averaged MS information at every point and discard the raw data from each realisation record. Information on probability distribution should also be retained at regular traverse velocity increments, but certainly not at all generated data points.

The averaged results of the MS input realisations were used to ascertain the statistical scatter which could be anticipated in the SAMS responses. These input samples appeared reasonably stationary, however any sampled point could vary by as much as 40% from the "averaged mean" value given a sample of only 50 realisations. This upper error bound could reasonably be expected to decrease as the square of the sample size.

To act as datum it was necessary to establish how the stationary MS responses varied with traverse velocity. Evaluation of these stationary response integrals is a tedious business. The time based MS response velocity and acceleration exhibit a marked upward trend with increasing traverse velocity. Such trends, although surprising at first, can be attributed to the fact that input velocity (displacement gradient) measured with respect to a time base increases linearly with traverse velocity.

(b) Sample averaged MS Responses -

Given typical vehicle traverse accelerations, existing papers indicate that non-stationary MS displacement responses show only marginal deviation (1 - 2%) from their stationary equivalents. For practical sample sizes such effects lie beyond the resolution of this sample averaging (SA) technique; however the SA results do not contradict the findings of these already existing studies. It would appear that the effect of non-stationarity on MS displacement response is

marginal for typical road vehicle accelerations.

There are no existing solutions with which to compare the non-stationary MS derivative (velocity and acceleration) response results obtained in this project. However if the displacement results are in agreement with existing solutions, then confidence can also be placed in the derivative responses afforded by this SA approach. The derivative results are a natural by-product of the simulation.

Both non-stationary displacement and velocity MS responses are in good agreement with their stationary counterparts. When these derivative quantities are expressed in a spatially based form they demonstrate a pronounced asymptotic decay with increase in traverse velocity. This is due solely to the dominating effect of traverse velocity as indicated in section 3.6.

The crudeness of the mathematical road description defined by the Exp correlation, meant the derivative velocity input correlation is not physically realisable. However, the similarity of the conclusions obtained from both Exp and Mod Exp non-stationary responses simply serve to demonstrate the flexibility of the technique in handling very approximate road profile descriptions. This occurrence is frequently encountered in practice where correlations are often defined in a piecewise continuous manner.

(c) Further Statistical Testing -

By sampling response values at consistent arbitrary traverse velocities across all realisations in the sample, it was demonstrated that the response characteristics adhere, within statistical limits, to a normal probability distribution. It is proposed that this might be generally true of linear systems undergoing non-stationary excitation.

Further vindication of the reasonable statistical nature of the results was achieved by the introduction of an F - test.

(d) Inferences from a single realisation record -

Short of assembling a batch of independant realisations it is not possible to extract information on the probability distribution of non-stationary response results. However based on the results of the SA information, it would appear reasonable to assume a normal probability distribution, provided the dynamic system is linear.

Upon careful scrutiny of the displacement realisation records, the response amplitudes appear to peak marginally above their input counterparts. No great inference can be achieved from this other than displacement response might be marginally greater than input excitation. When a single derivative (velocity or acceleration) response record is considered decay characteristics strongly similar to those of the SA results are observed. Based on the strength of these observations, attempts were made at a least squares fit on a single realisation of displacement, velocity and acceleration. Unfortunately the data proved too ill conditioned to be viable. At present a crude envelope curve covering the peaks of the MS responses will certainly give a worst case estimate of non-stationary effects. However a more detailed study of methods of curve fitting applicable to this type of random data is required. A more detailed investigation into the generalised criteria for establishing the order of the curve fit on other input correlations and to other road traverse acceleration profiles would also be required. Both aspects of the problem should be adequately addressed before rejecting outright this notion of curve fitting.

Correlation	"True mean" input ratio (velocity/displacement)
Exp	0.196
Mod Exp	0.0469

Table 10.1 "True mean value" magnitude ratio for Exp and Mod Exp Correlations.

Correlation	Input			Response		
	Min.	Max.	Range	Min.	Max.	Range
Exp	0.69	1.42	0.73	0.63	1.58	0.95
Mod Exp	0.61	1.60	0.99	0.77	2.09	1.32

Table 10.2 Scatter range of sampled MS displacement values.

Correlation	Maximum Divergence			
	Input		Response	
	Displacement	Velocity	Displacement	Velocity
Exp	0.18	0.16	0.20	0.25
Mod Exp	0.25	0.22	0.22	0.13

Table 10.3 Maximum divergence of MS probability distribution from theoretical Chi-Square with one degree of freedom.

Ergodic Sample Size	50	100	200	400	800
Maximum Divergence	0.34	0.23	0.22	0.18	0.14

Table 10.4 Maximum divergence of ergodic MS sample from theoretical Chi-Square with one degree of freedom.

Correlation		Input		Response		
		Displacement	Velocity	Displacement	Velocity	Acceleration
Exp	S_1	1.286	1.377	1.021	4.189	6.424
	S_2	1.608	1.428	1.563	4.790	6.883
	F	0.799	0.964	0.653	0.875	0.933
Mod Exp	S_1	1.189	5.035	1.518	2.255	4.508
	S_2	1.189	3.217	1.872	4.980	2.107
	F	1.000	1.565	0.811	0.453*	2.140*

Table 10.5 Sample F - ratio tests for data point 125 - both F - fractions, S_1, and S_2, comprise a sample size of 25 values.

REFERENCES - CHAPTER X

Ref. 1 Virchis, V.J., and Robson, J.D. (1971) (see Chapter 1, ref. 2).

Ref. 2 Sobczyk, K, and Macvean, D.B. (1976) (see Chapter 1, ref. 3)

Ref. 3 Pugachev (1965) "Theory of random functions and its application to control problems". Revised and translated by O.M. Blunn, Oxford Pergamon Press.

Vehicle characteristics : $f_n = 1.584$ g, $\zeta = 0.35$

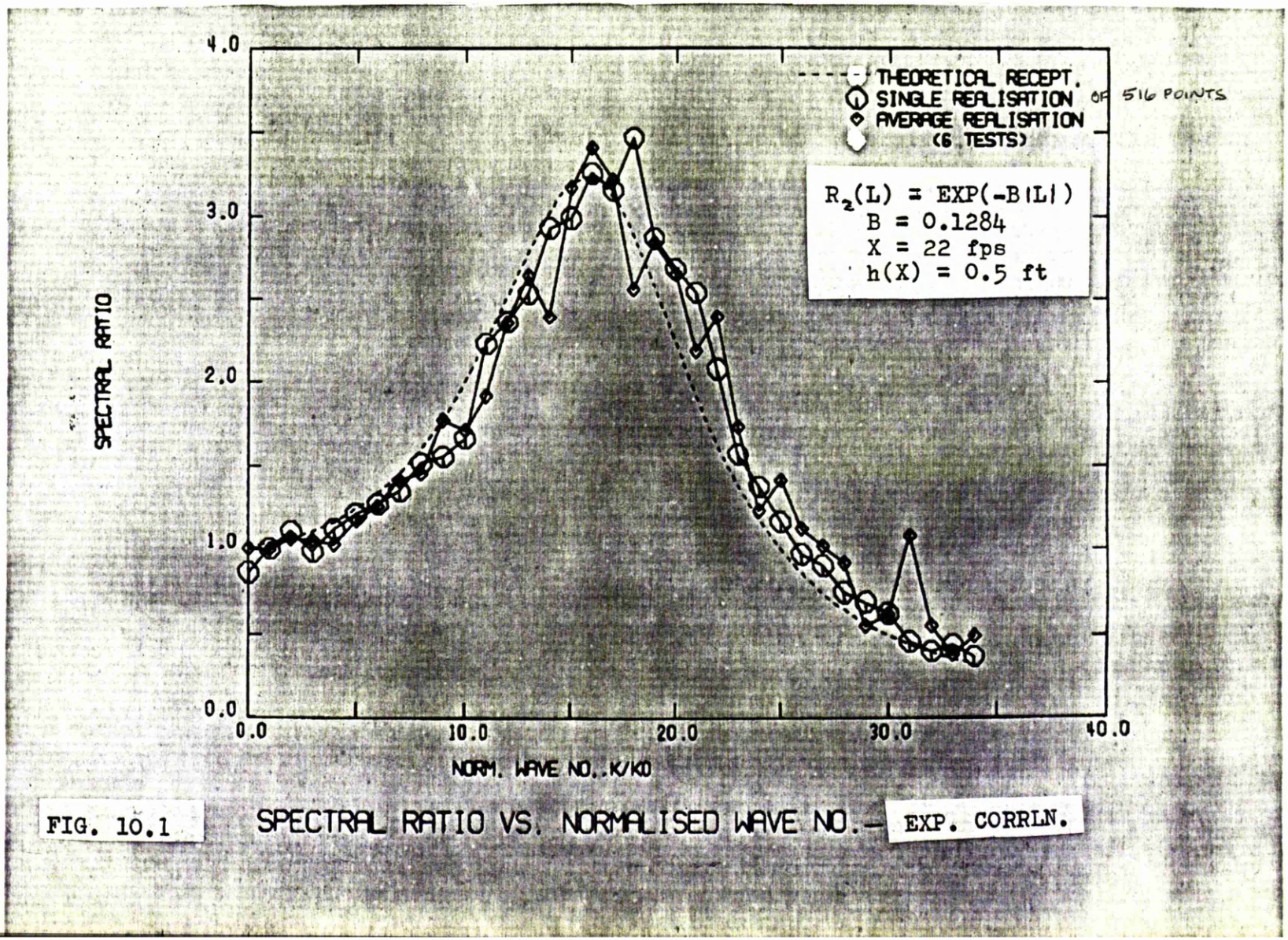


FIG. 10.1 SPECTRAL RATIO VS. NORMALISED WAVE NO.— EXP. CORRLN.

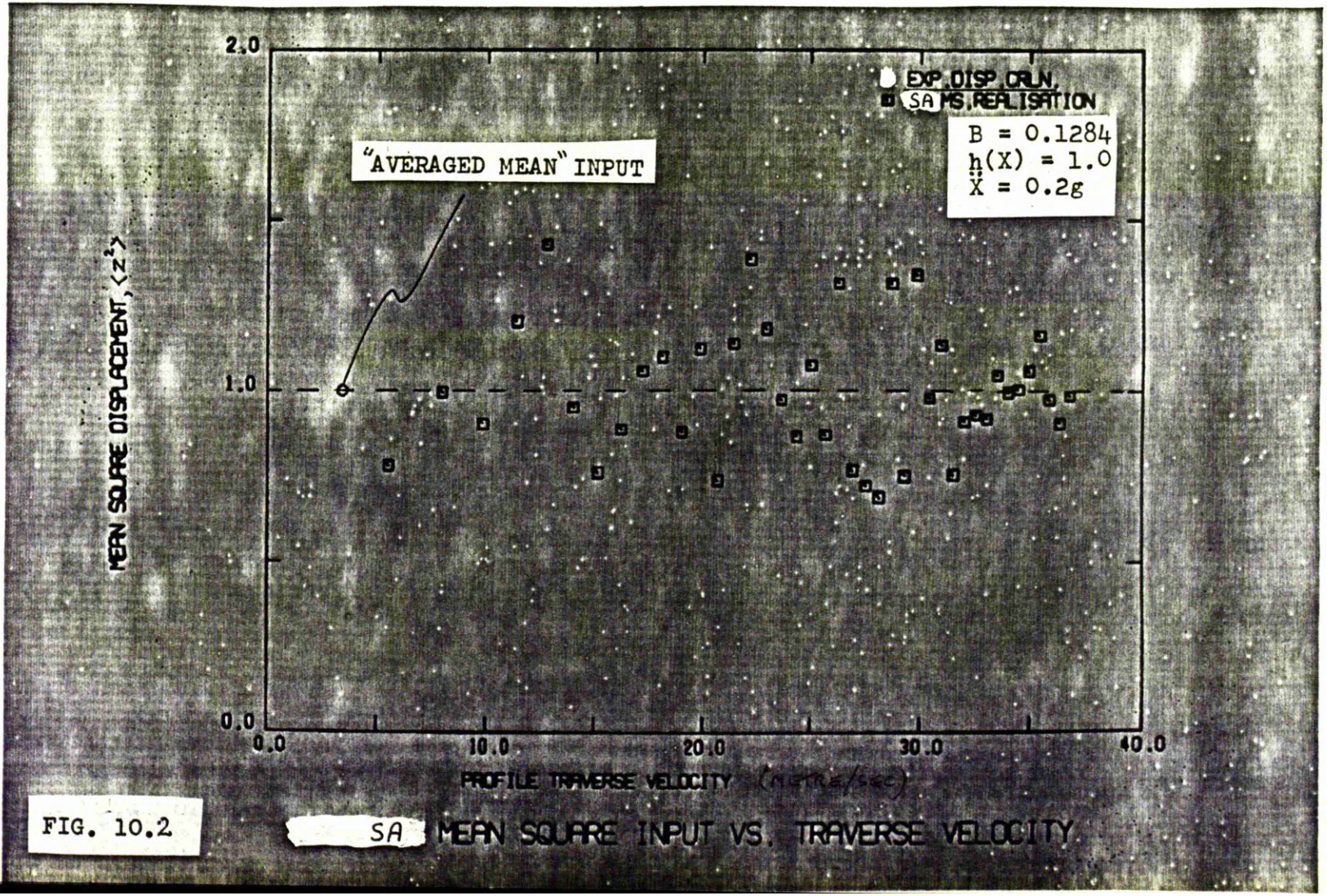


FIG. 10.2 SA MEAN SQUARE INPUT VS. TRAVERSE VELOCITY

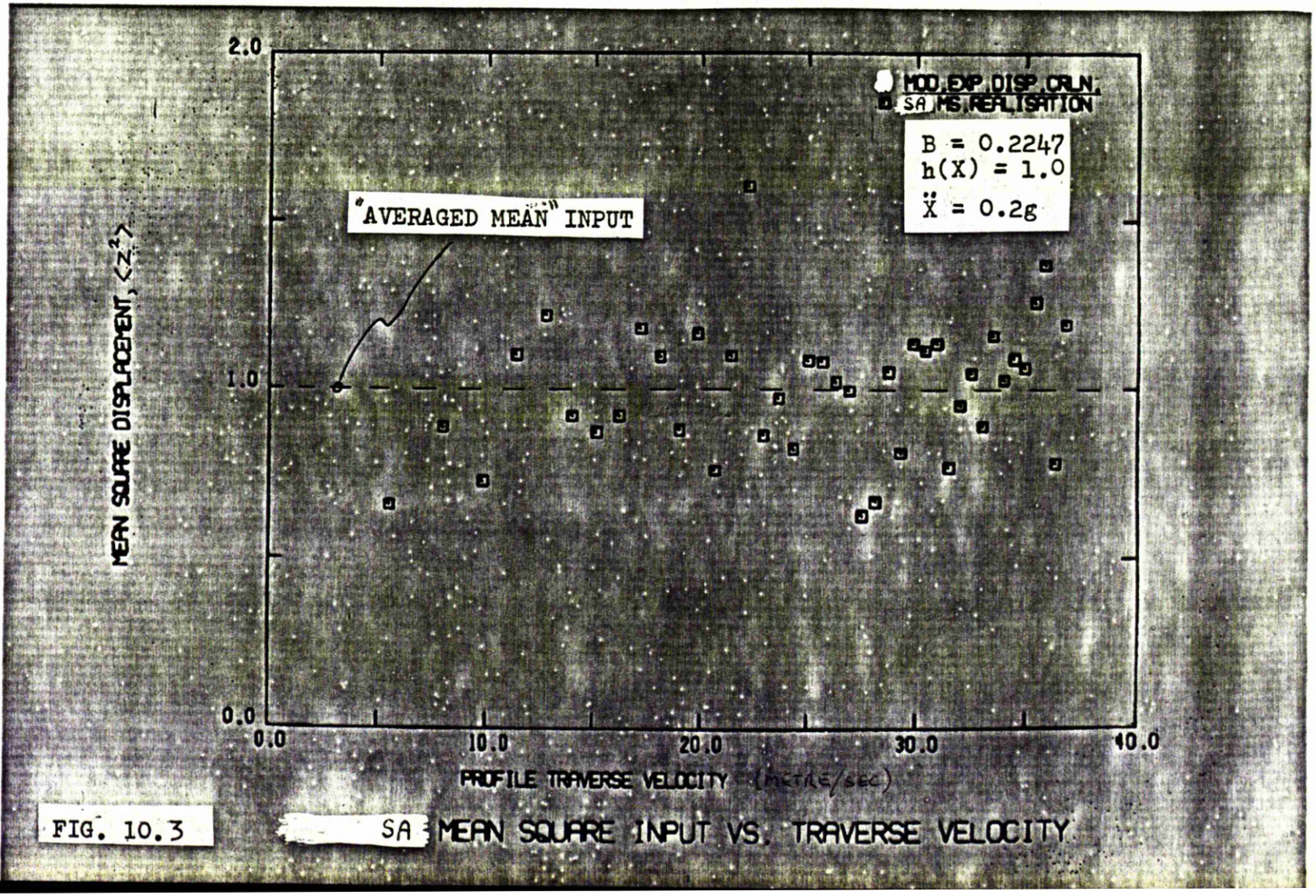


FIG. 10.3 SA MEAN SQUARE INPUT VS. TRAVERSE VELOCITY

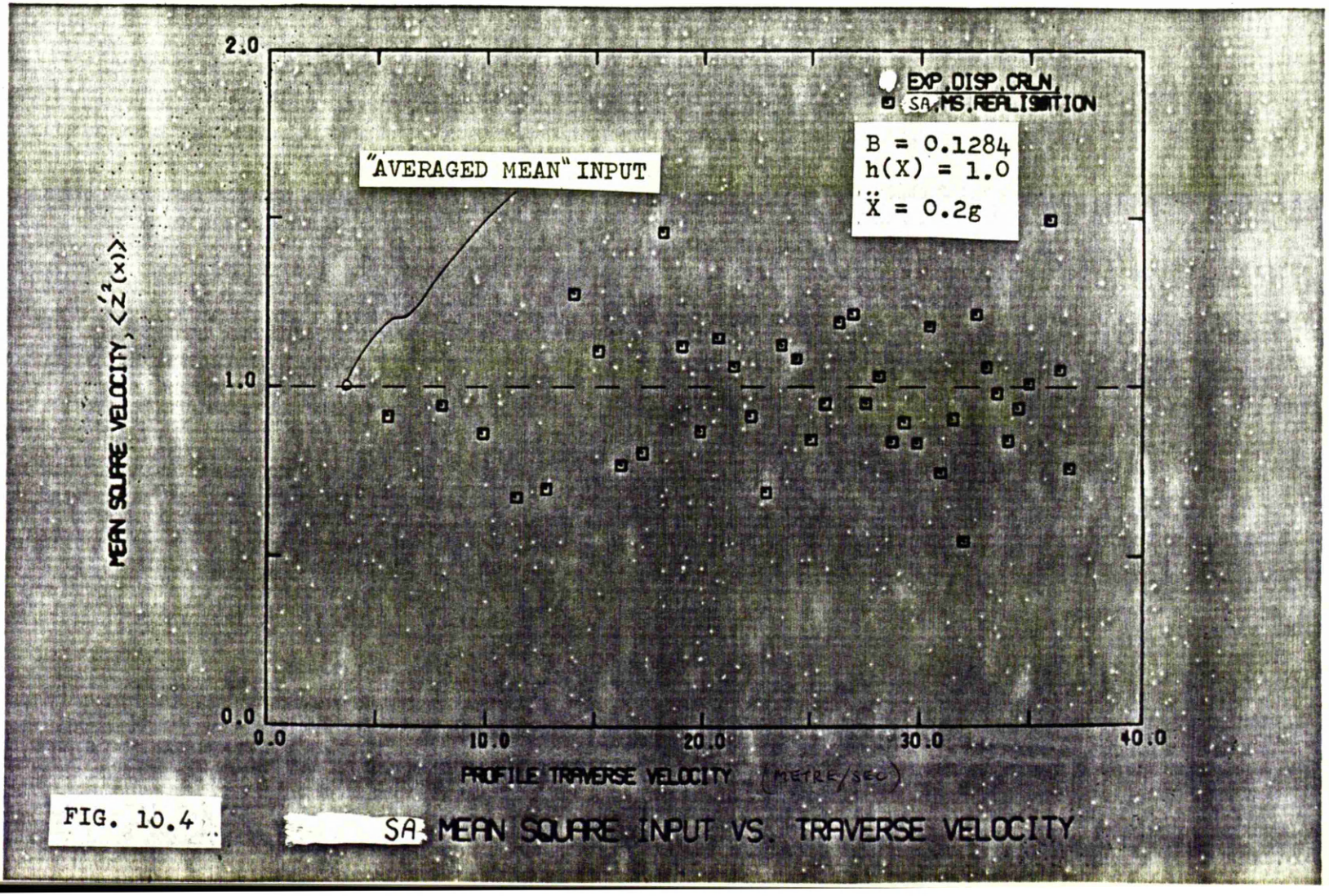


FIG. 10.4 SA MEAN SQUARE INPUT VS. TRAVERSE VELOCITY

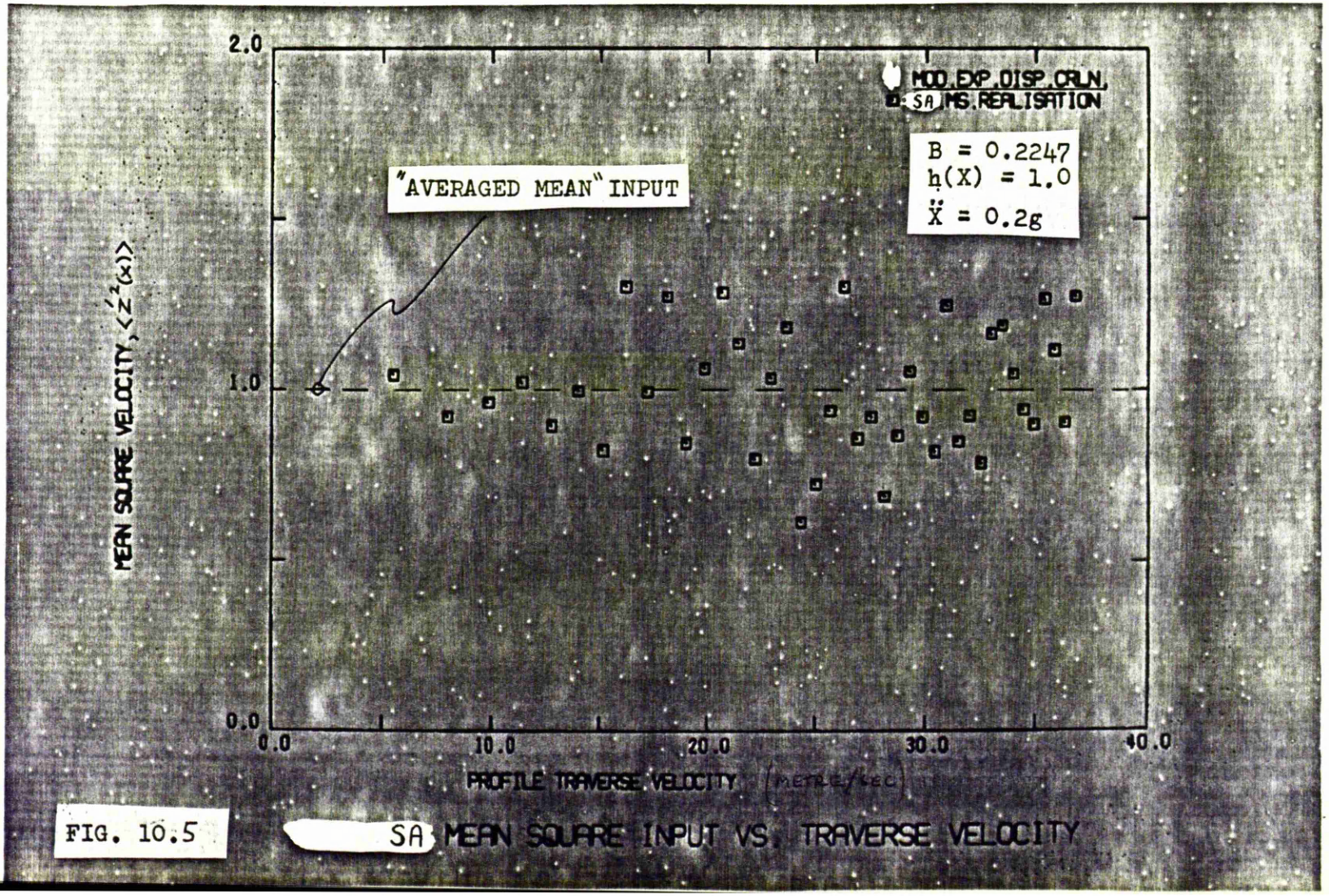


FIG. 10.5 SA MEAN SQUARE INPUT VS. TRAVERSE VELOCITY

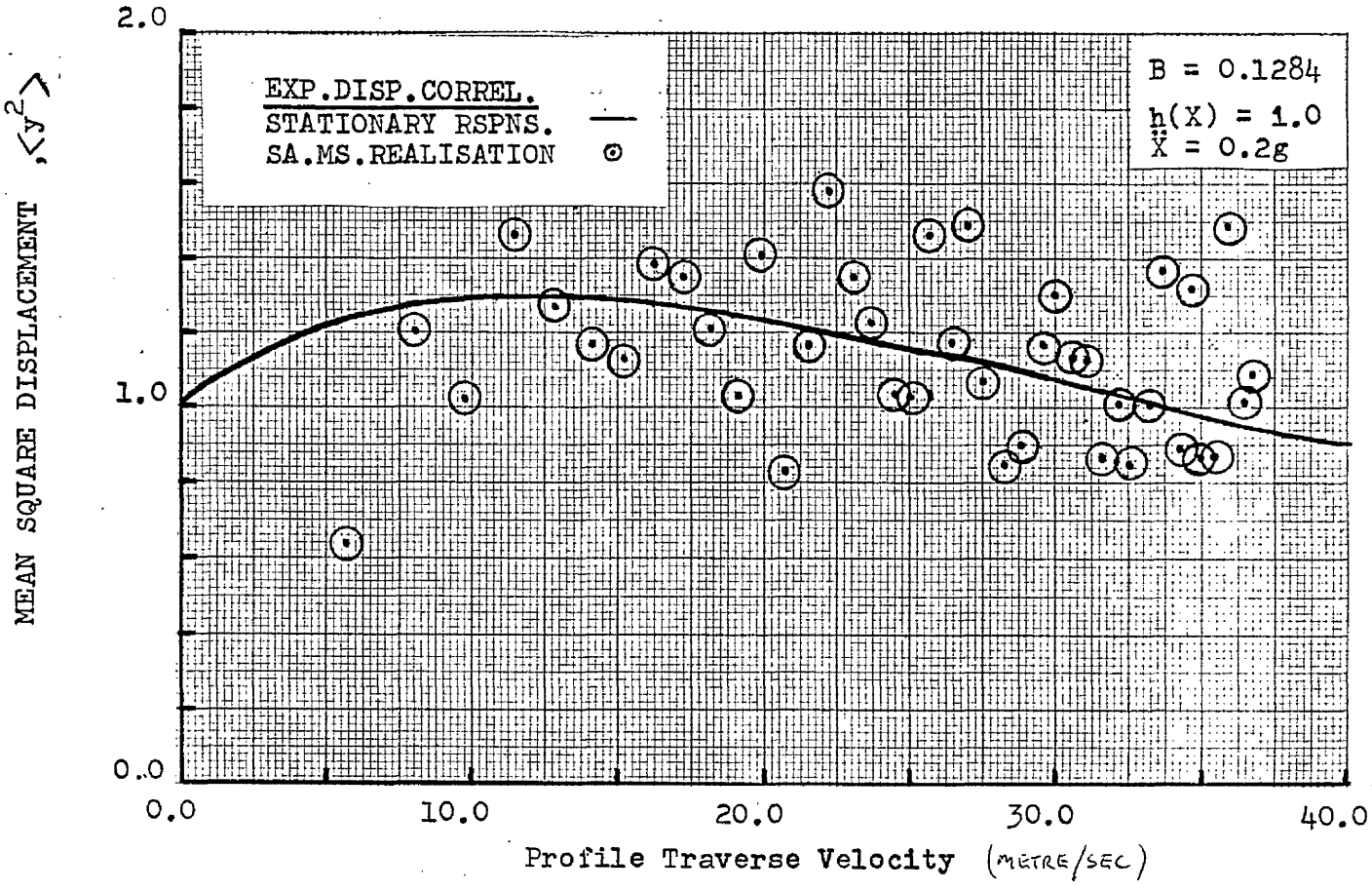


FIG. 10.6 SA MEAN SQUARE RESPONSE VS. TRAVERSE VELOCITY

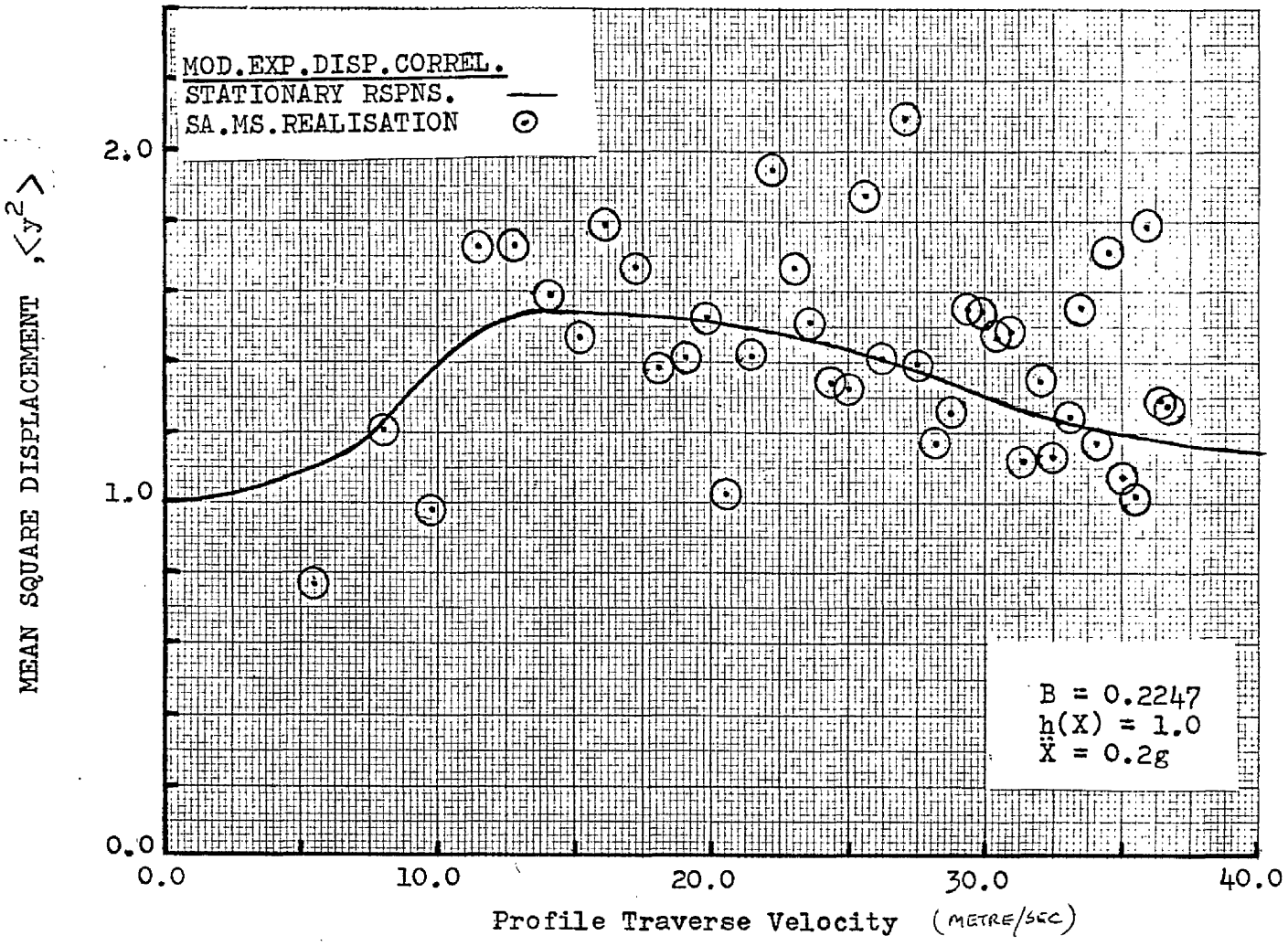


FIG. 10.7 SA MEAN SQUARE RESPONSE VS. TRAVERSE VELOCITY

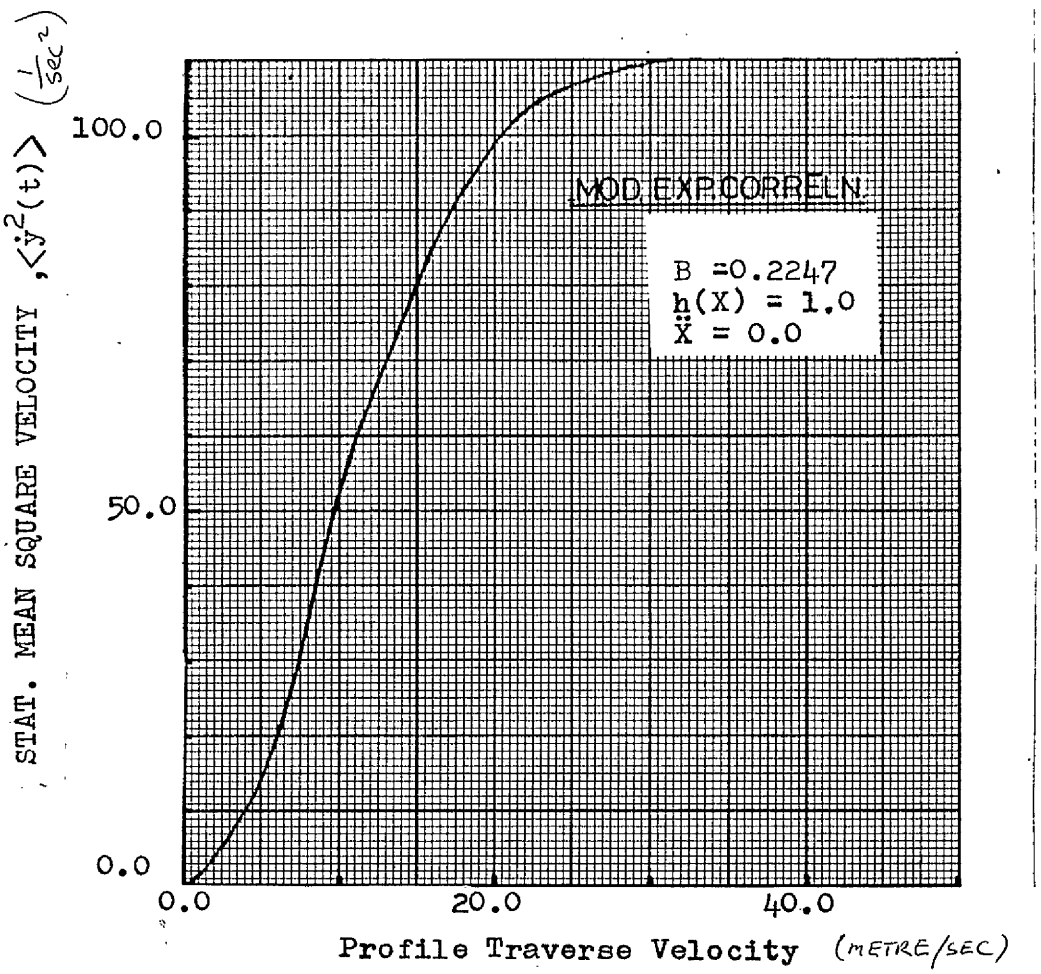


FIG. 10.8 STAT. MEAN SQUARE RESPONSE VS. TRAVERSE VELOCITY

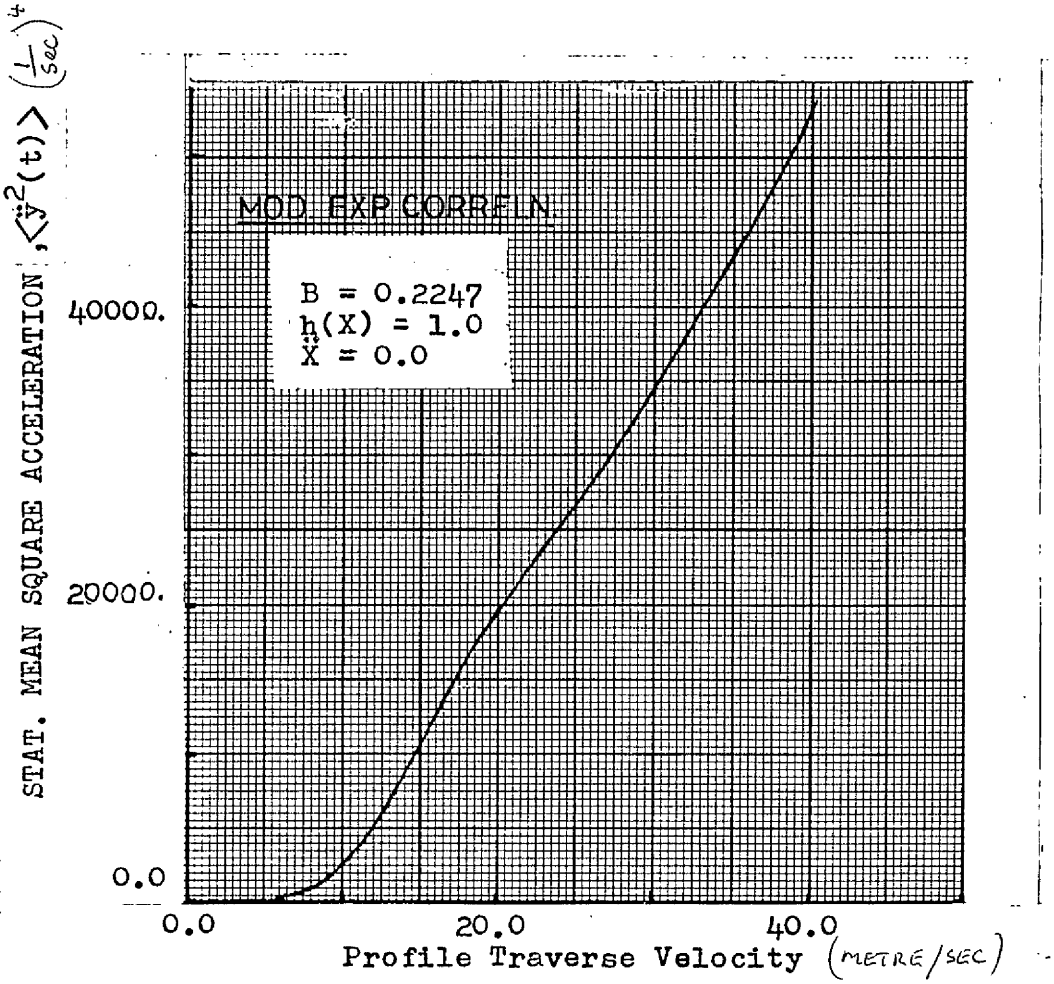


FIG. 10.9 STAT. MEAN SQUARE RESPONSE VS. TRAVERSE VELOCITY

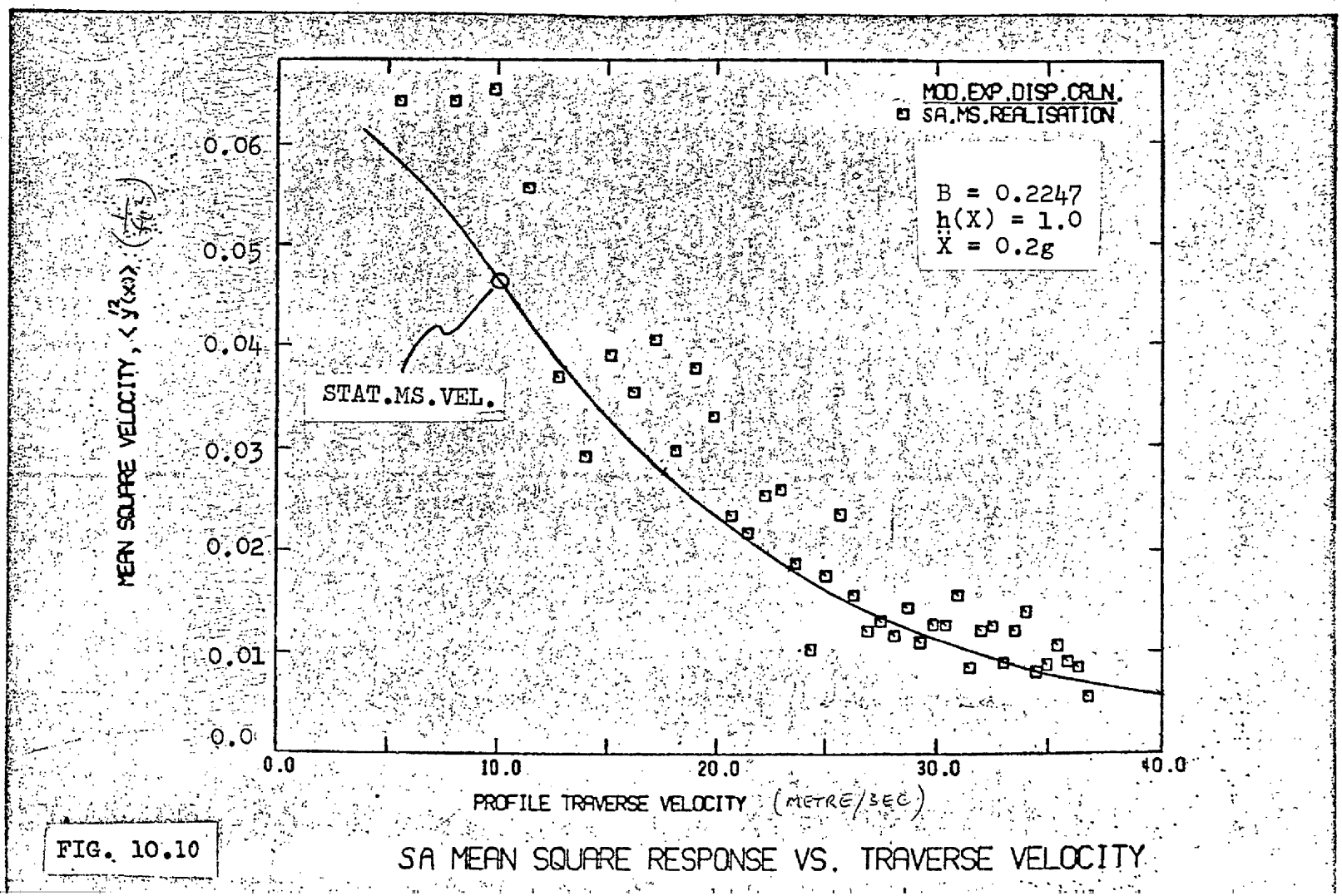


FIG. 10.10

SA MEAN SQUARE RESPONSE VS. TRAVERSE VELOCITY

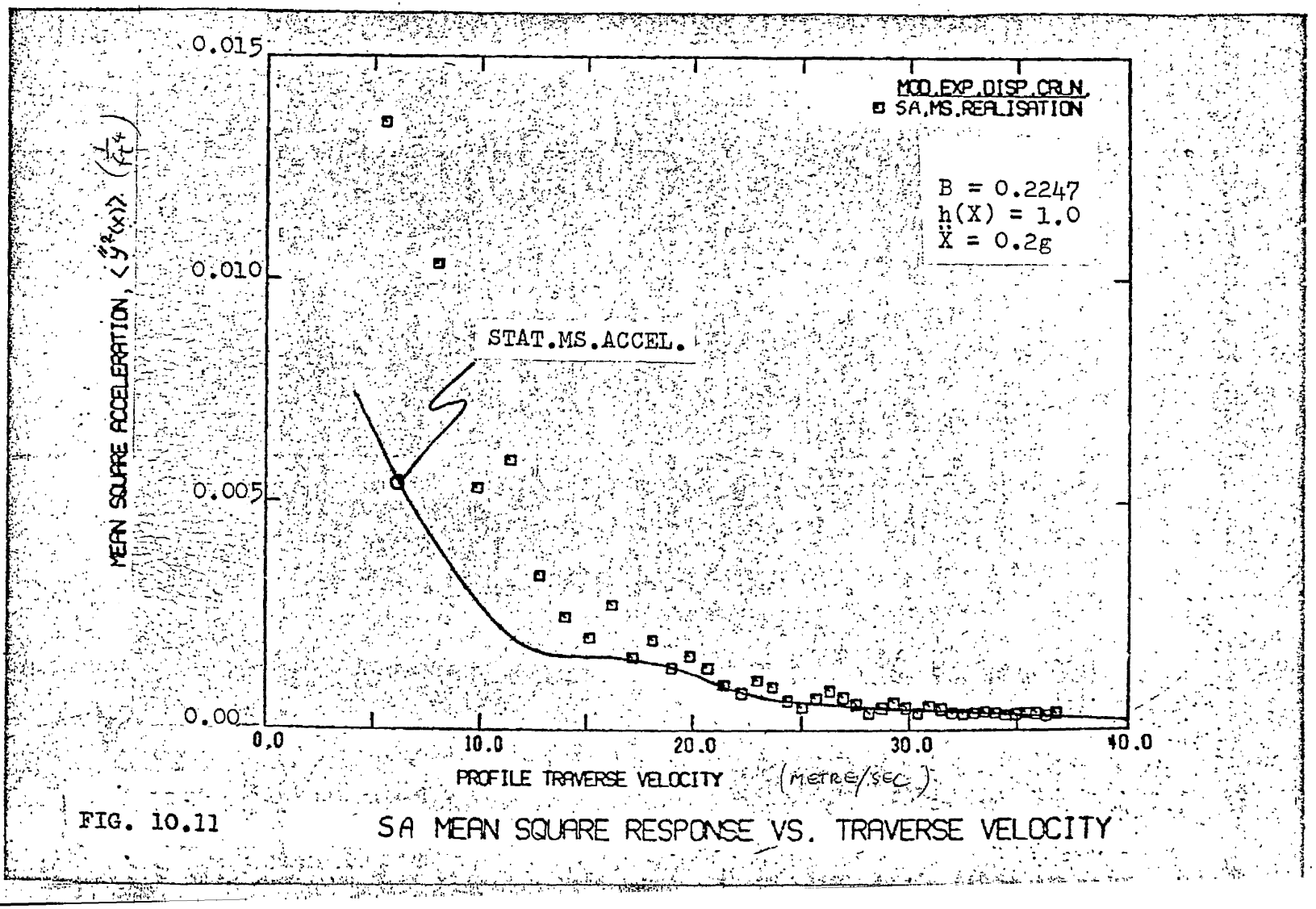


FIG. 10.11 SA MEAN SQUARE RESPONSE VS. TRAVERSE VELOCITY

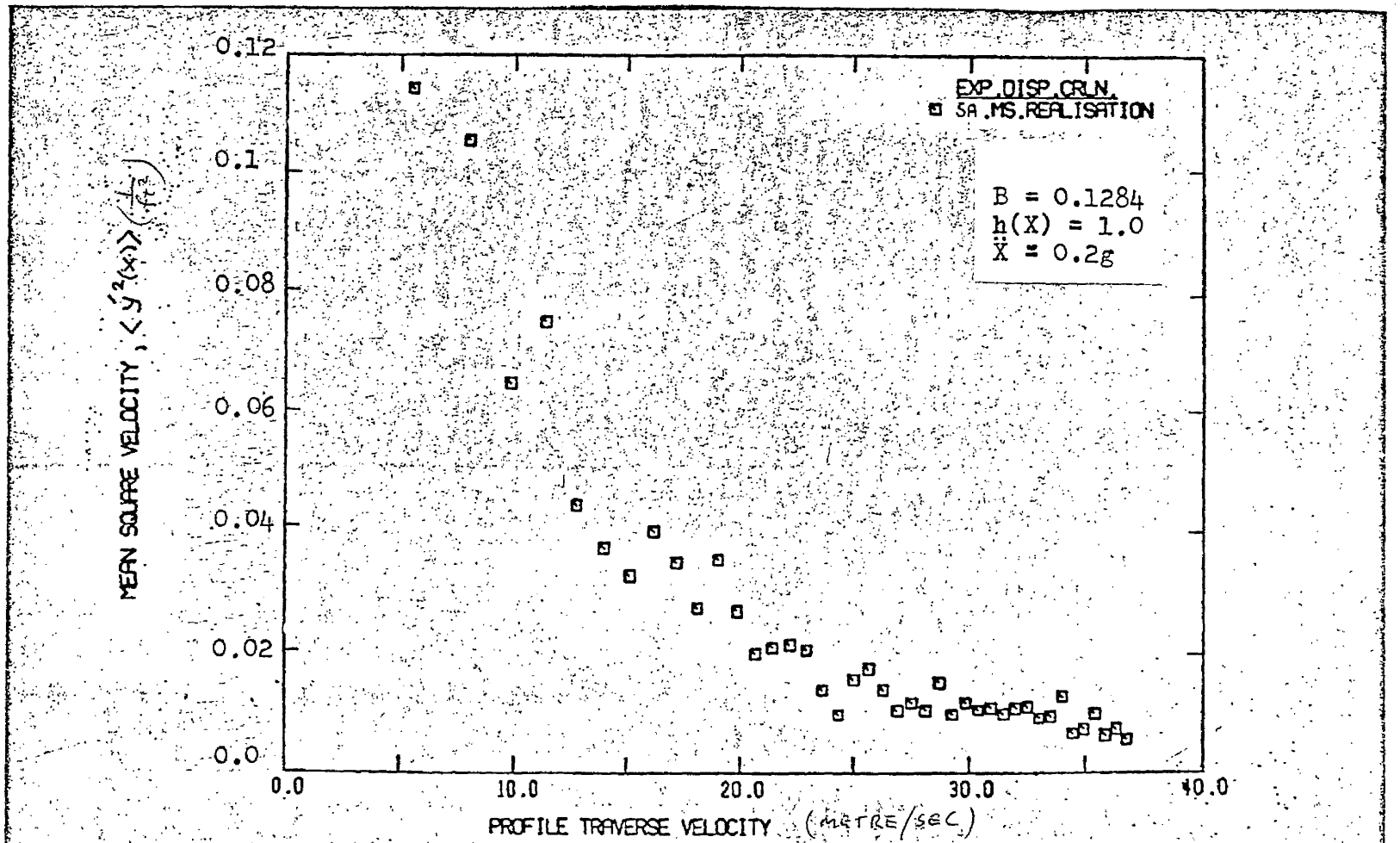


FIG. 10.12 SA MEAN SQUARE RESPONSE VS. TRAVERSE VELOCITY

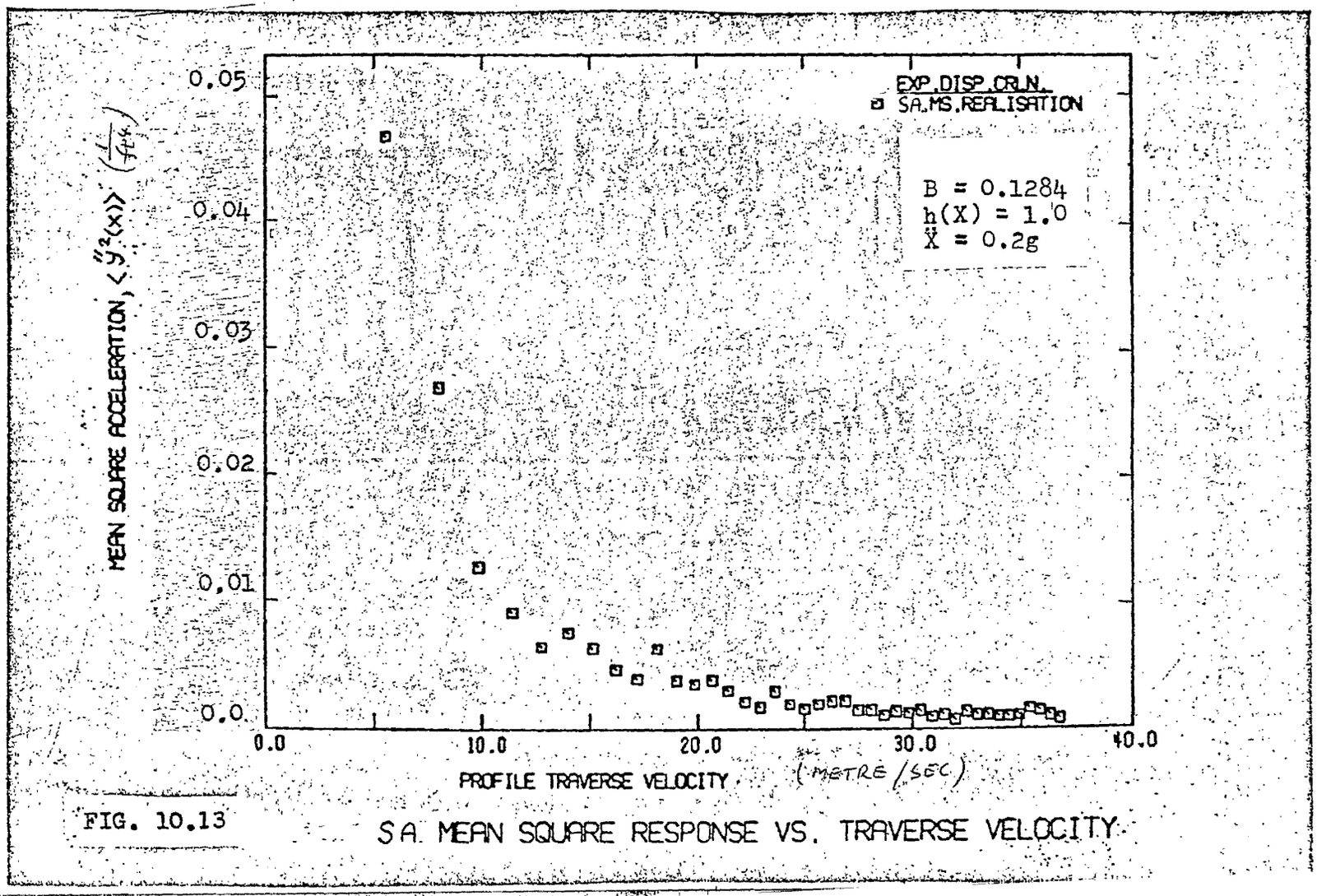


FIG. 10.13

S.A. MEAN SQUARE RESPONSE VS. TRAVERSE VELOCITY

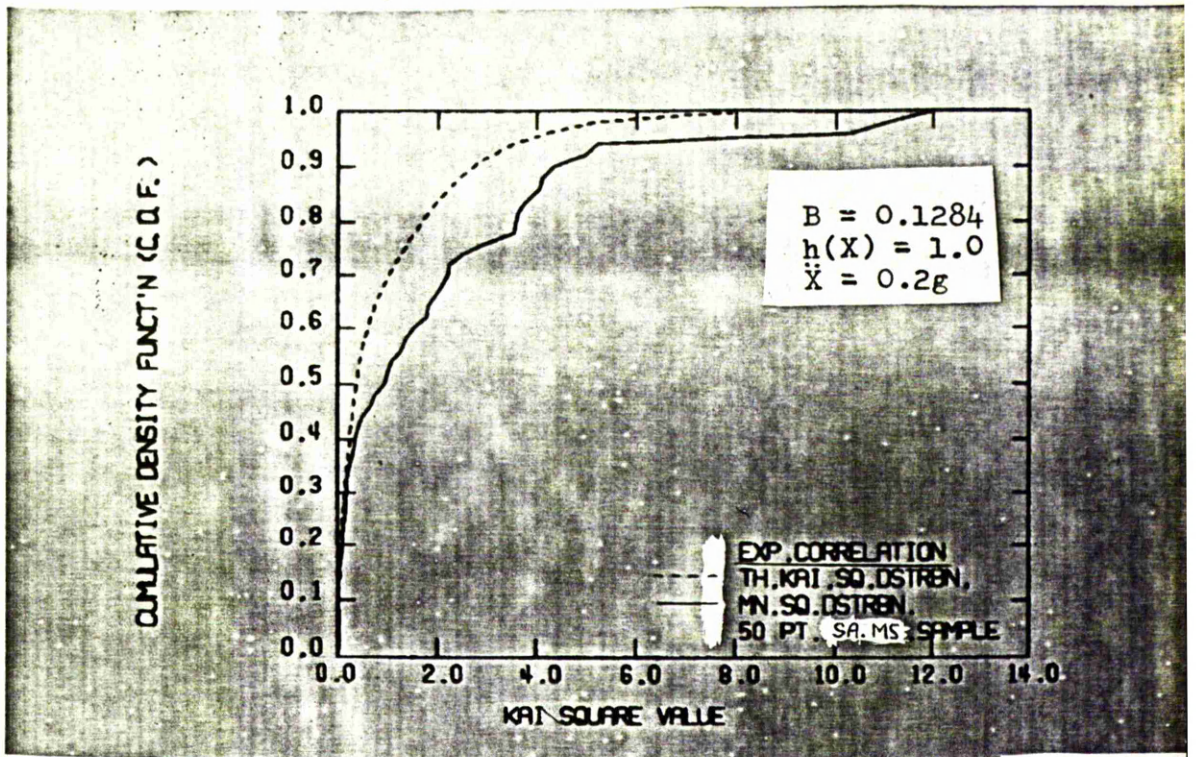


FIG. 10.14 DISTRIBUTION OF MEAN SQUARE INPUT DISP. -NONSTNRY.

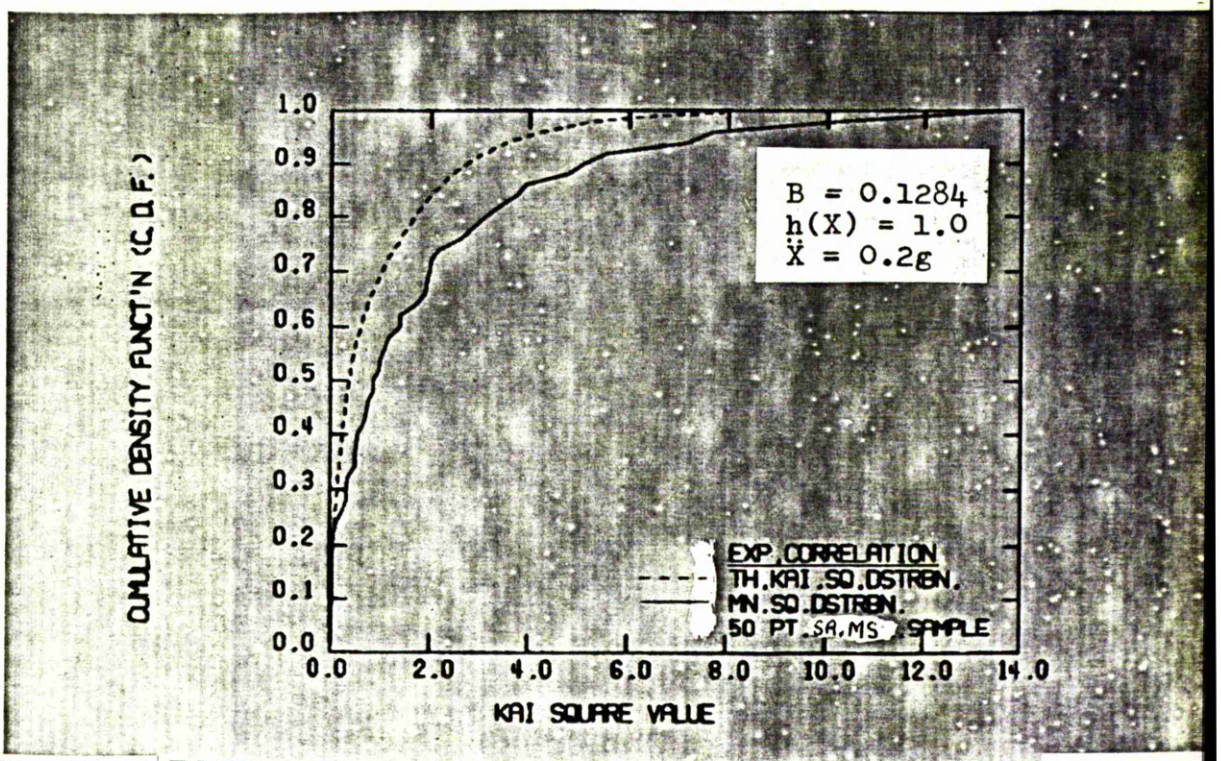


FIG. 10.15 DISTRIBUTION OF MEAN SQUARE RESPONSE DISP. -NONSTNRY.

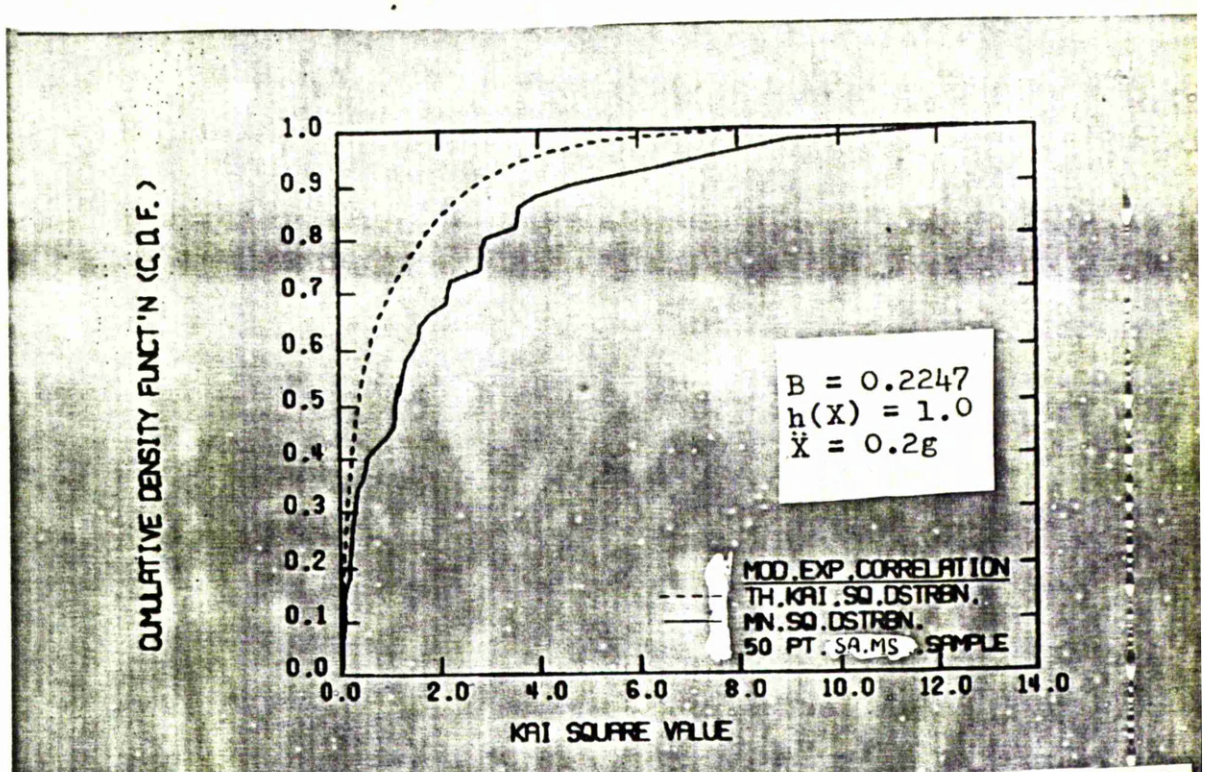


FIG. 10.16 DISTRIBUTION OF MEAN SQUARE INPUT DISP. -NONSTNRY.

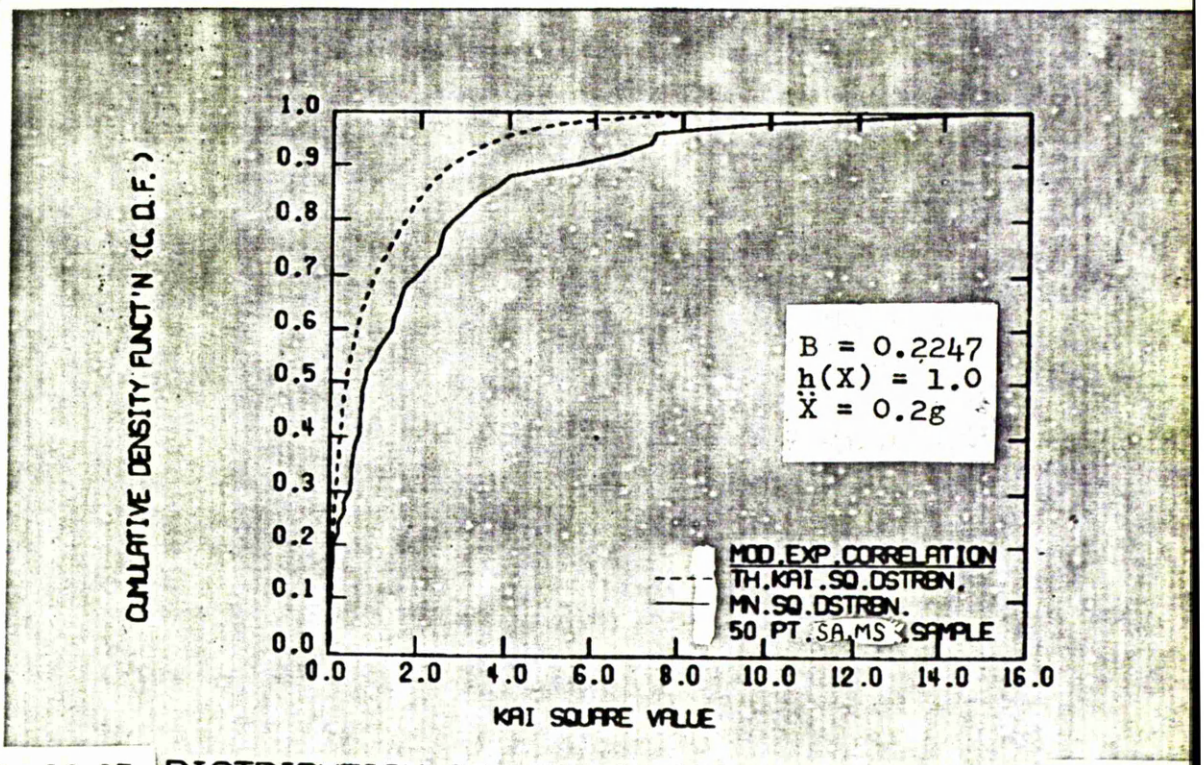


FIG. 10.17 DISTRIBUTION OF MEAN SQUARE RESPONSE DISP.-NONSTNRY

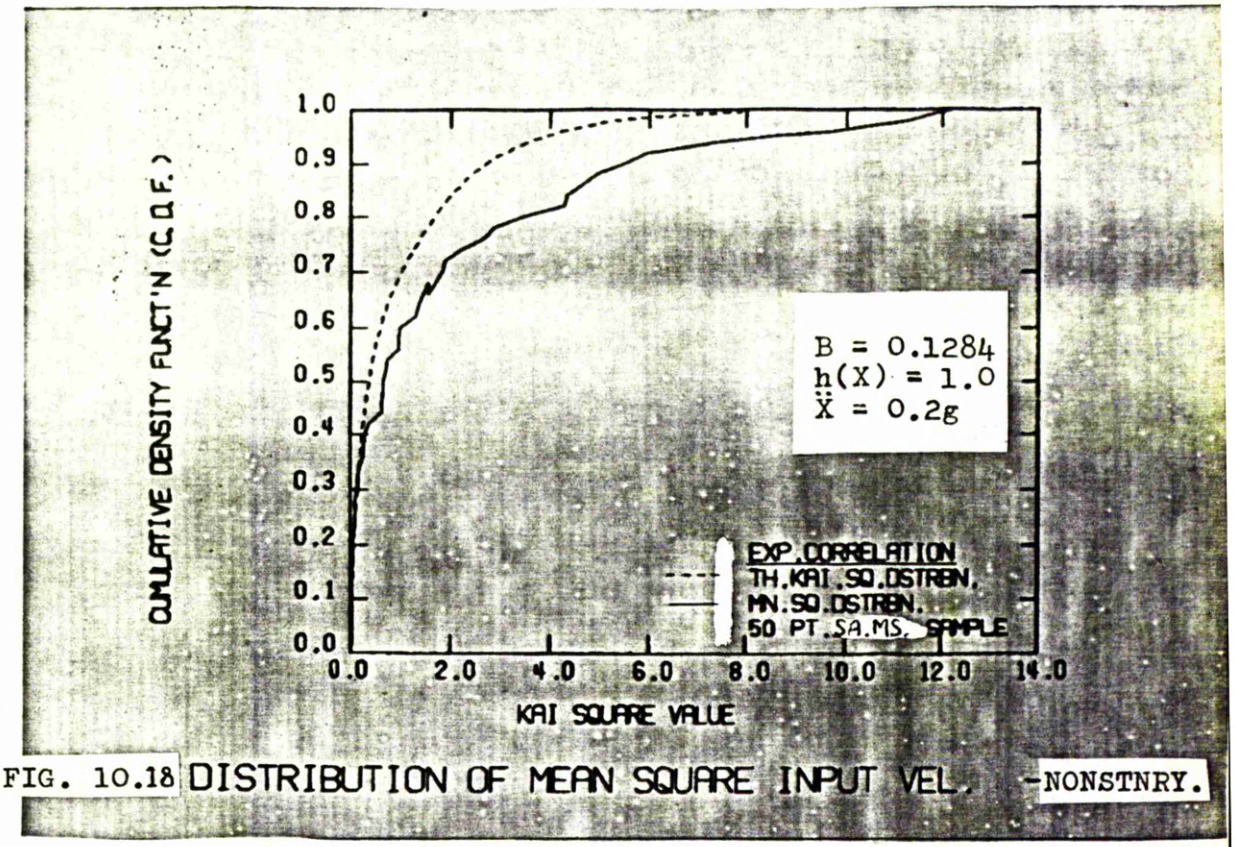


FIG. 10.18 DISTRIBUTION OF MEAN SQUARE INPUT VEL. -NONSTNRY.

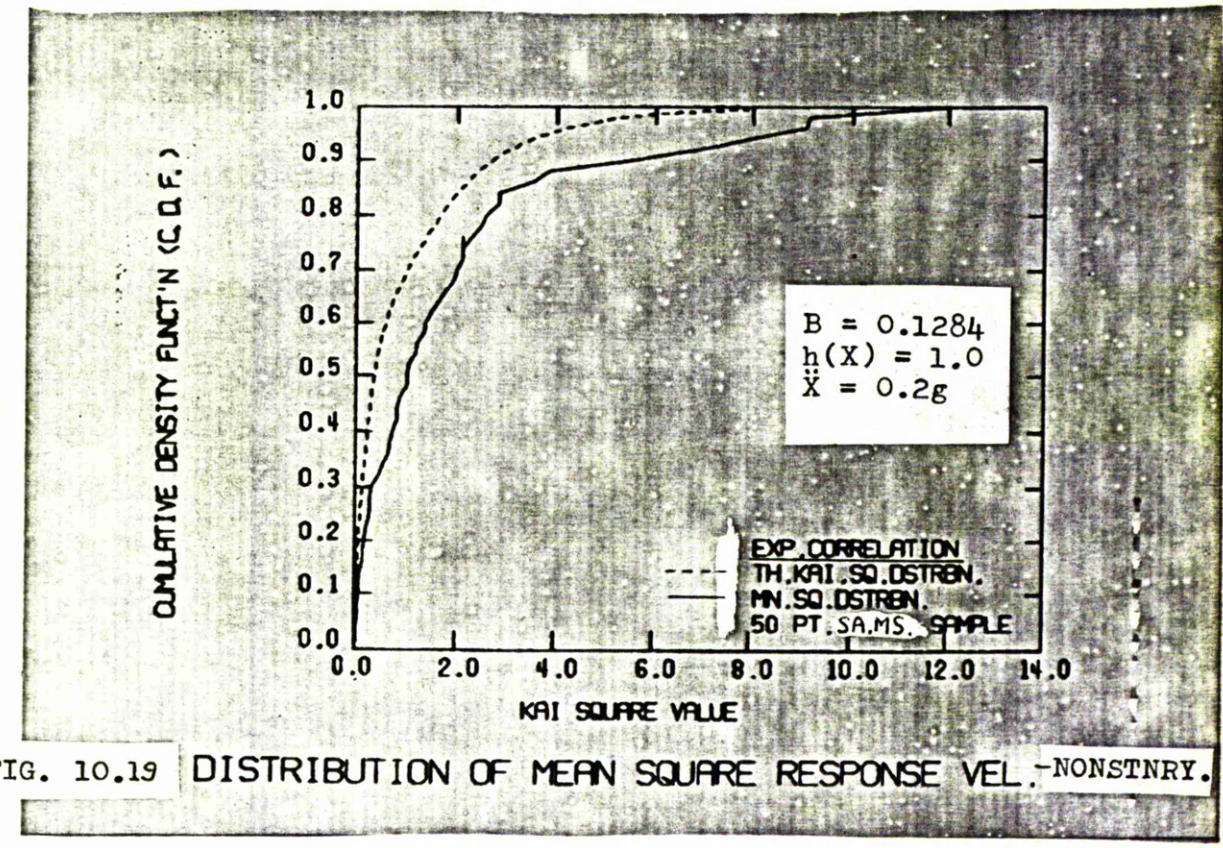


FIG. 10.19 DISTRIBUTION OF MEAN SQUARE RESPONSE VEL. -NONSTNRY.

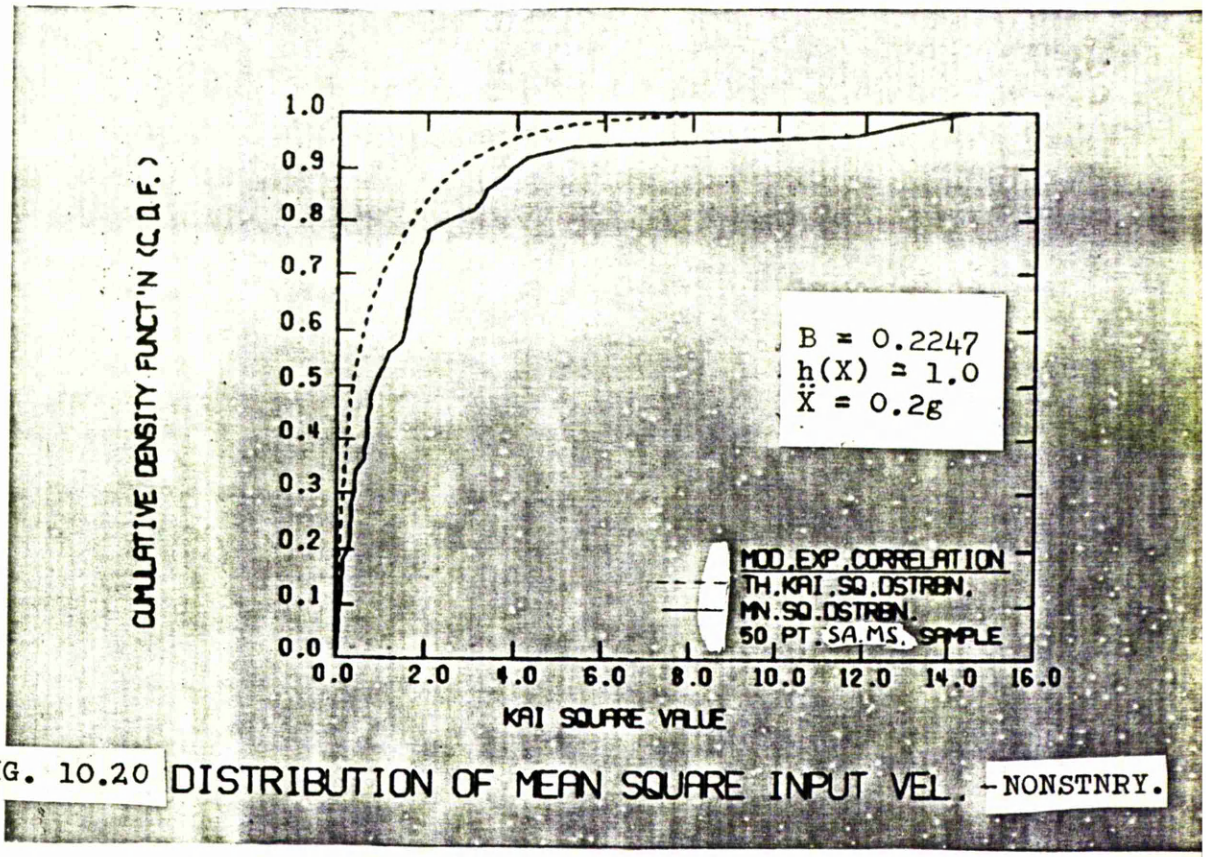


FIG. 10.20 DISTRIBUTION OF MEAN SQUARE INPUT VEL. - NONSTNRY.

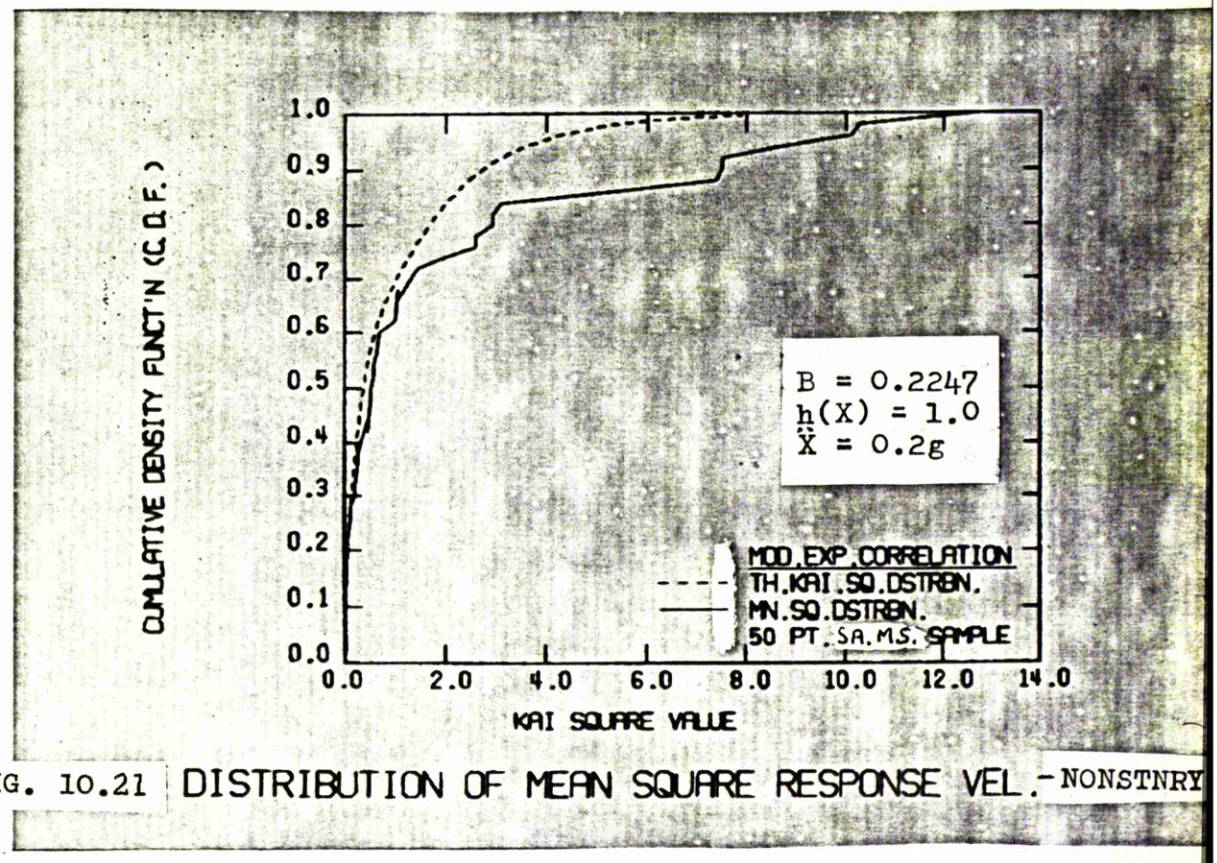


FIG. 10.21 DISTRIBUTION OF MEAN SQUARE RESPONSE VEL. - NONSTNRY

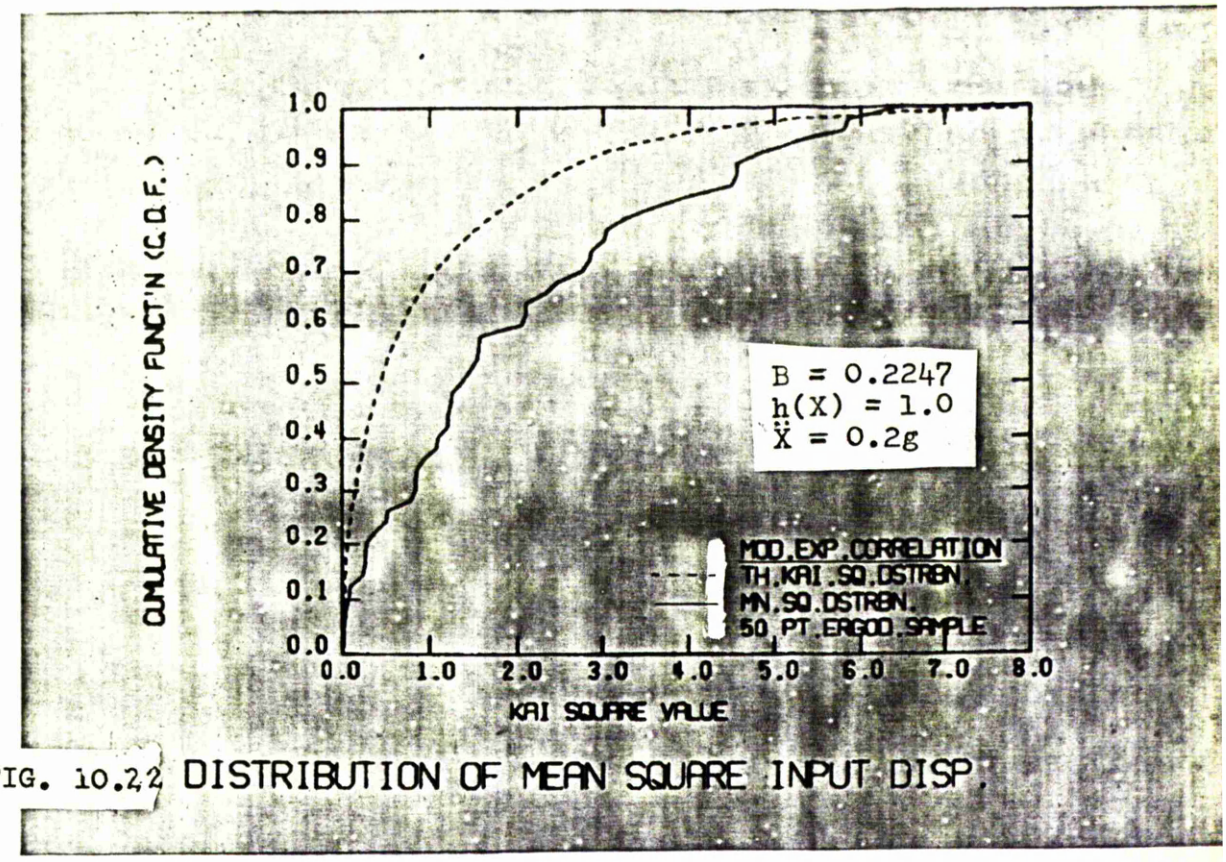


FIG. 10.22 DISTRIBUTION OF MEAN SQUARE INPUT DISP.

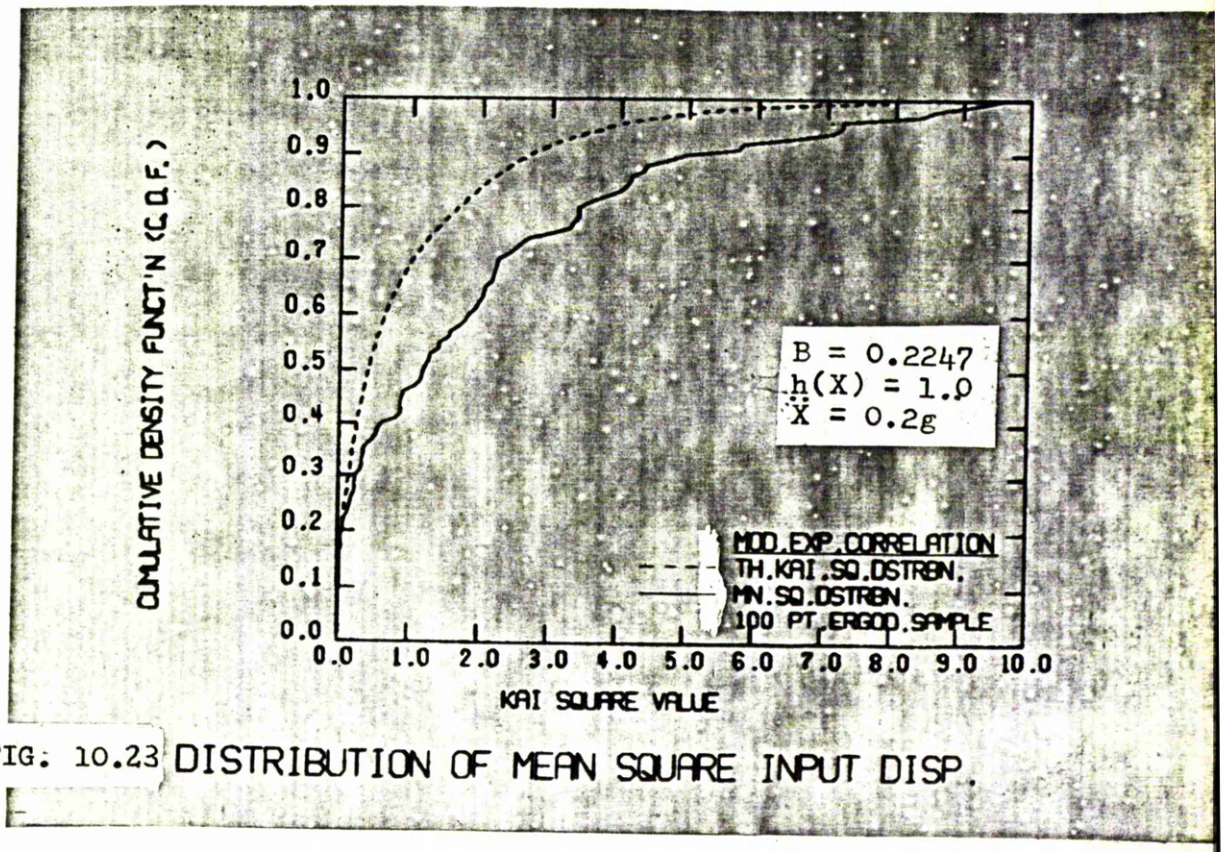


FIG. 10.23 DISTRIBUTION OF MEAN SQUARE INPUT DISP.

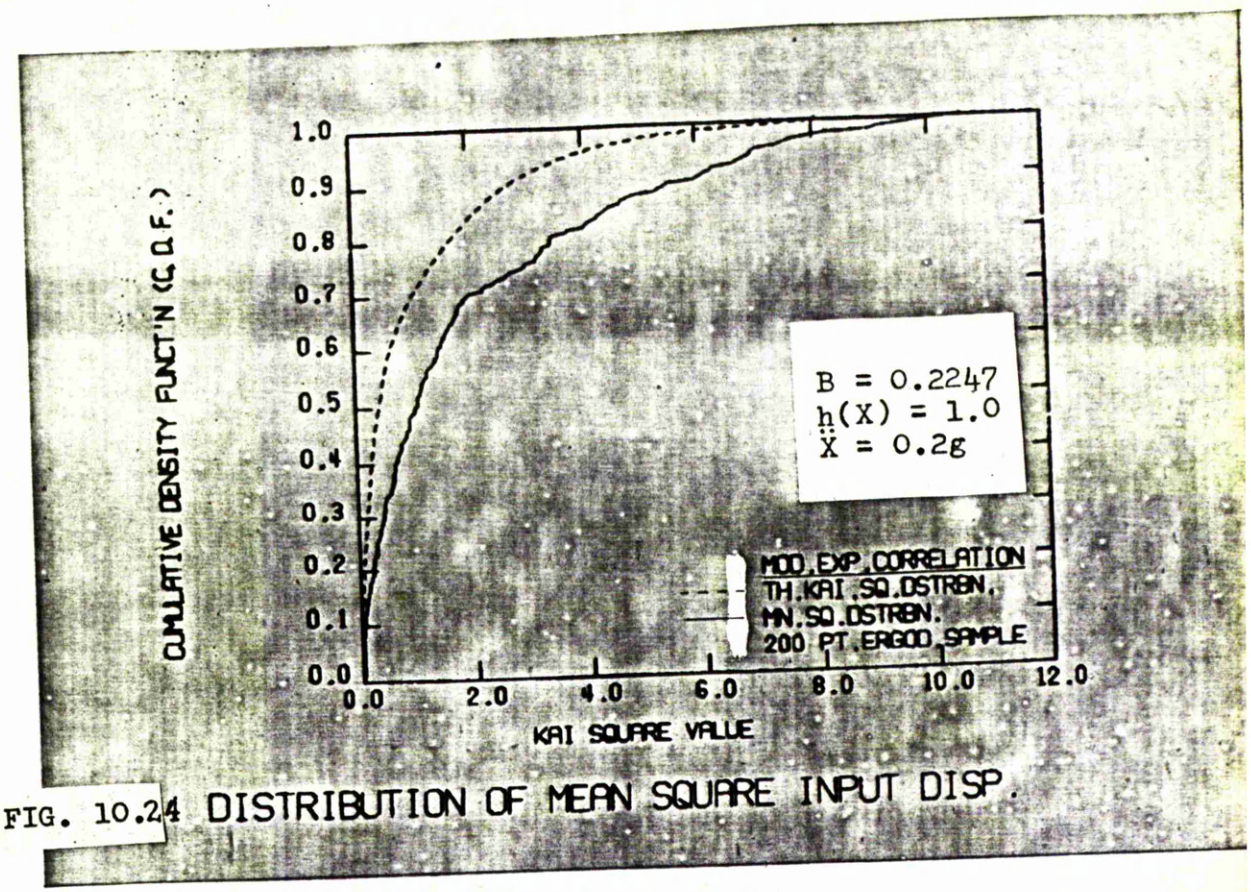


FIG. 10.24 DISTRIBUTION OF MEAN SQUARE INPUT DISP.

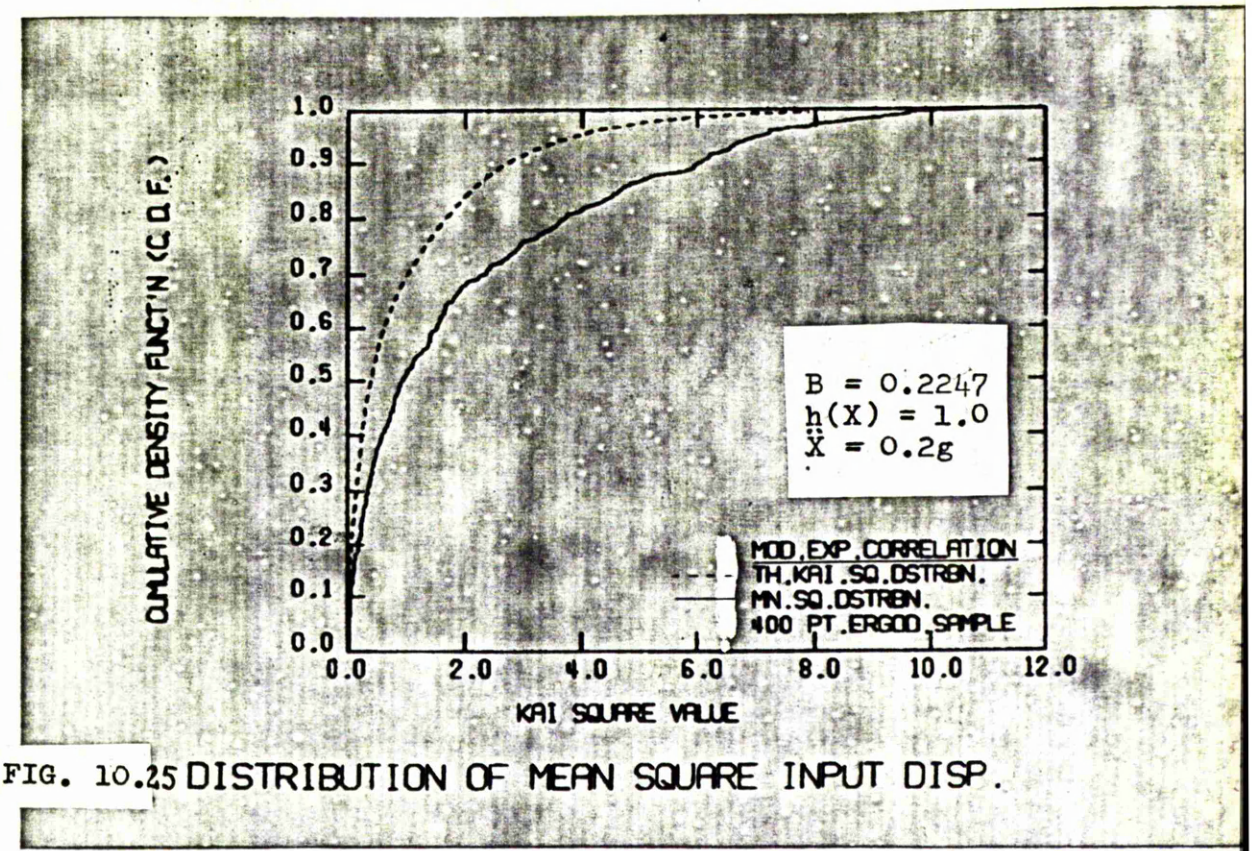


FIG. 10.25 DISTRIBUTION OF MEAN SQUARE INPUT DISP.

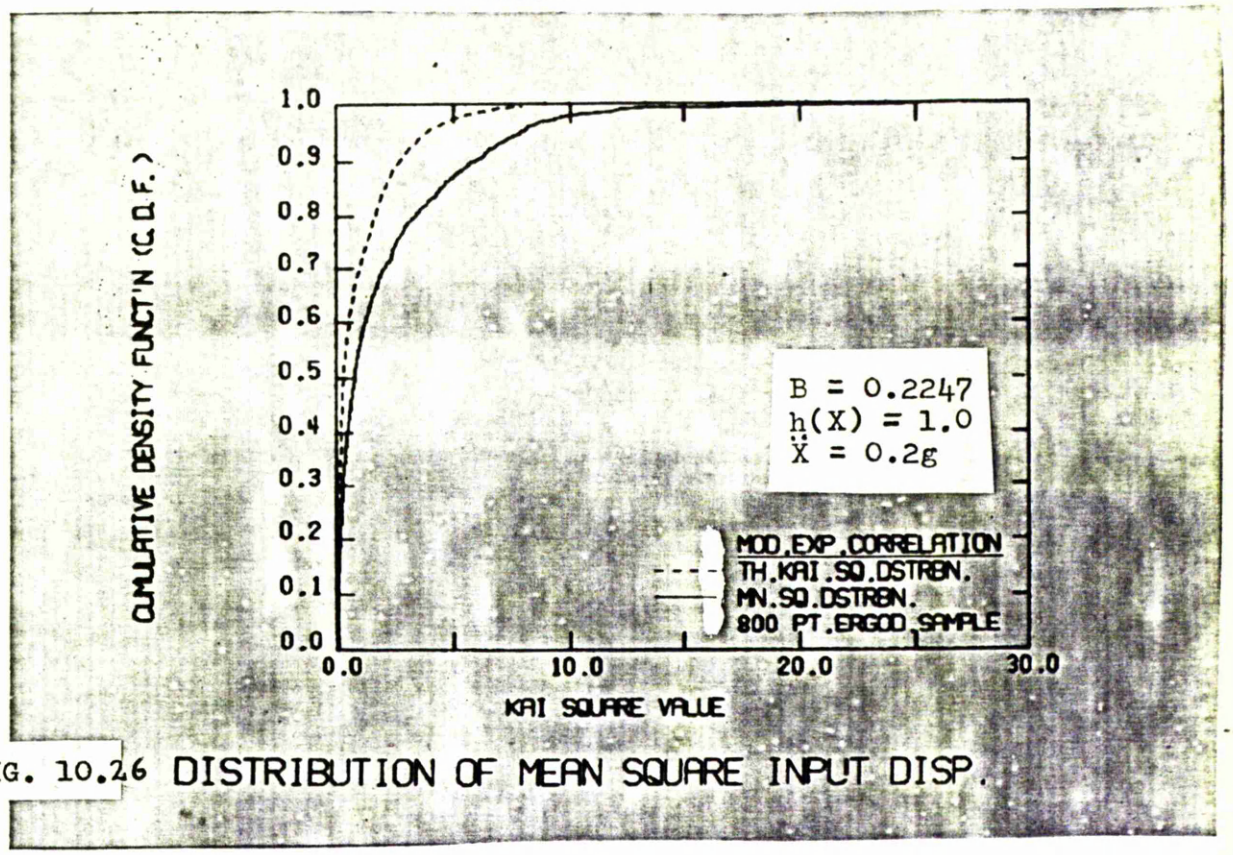


FIG. 10.26 DISTRIBUTION OF MEAN SQUARE INPUT DISP.

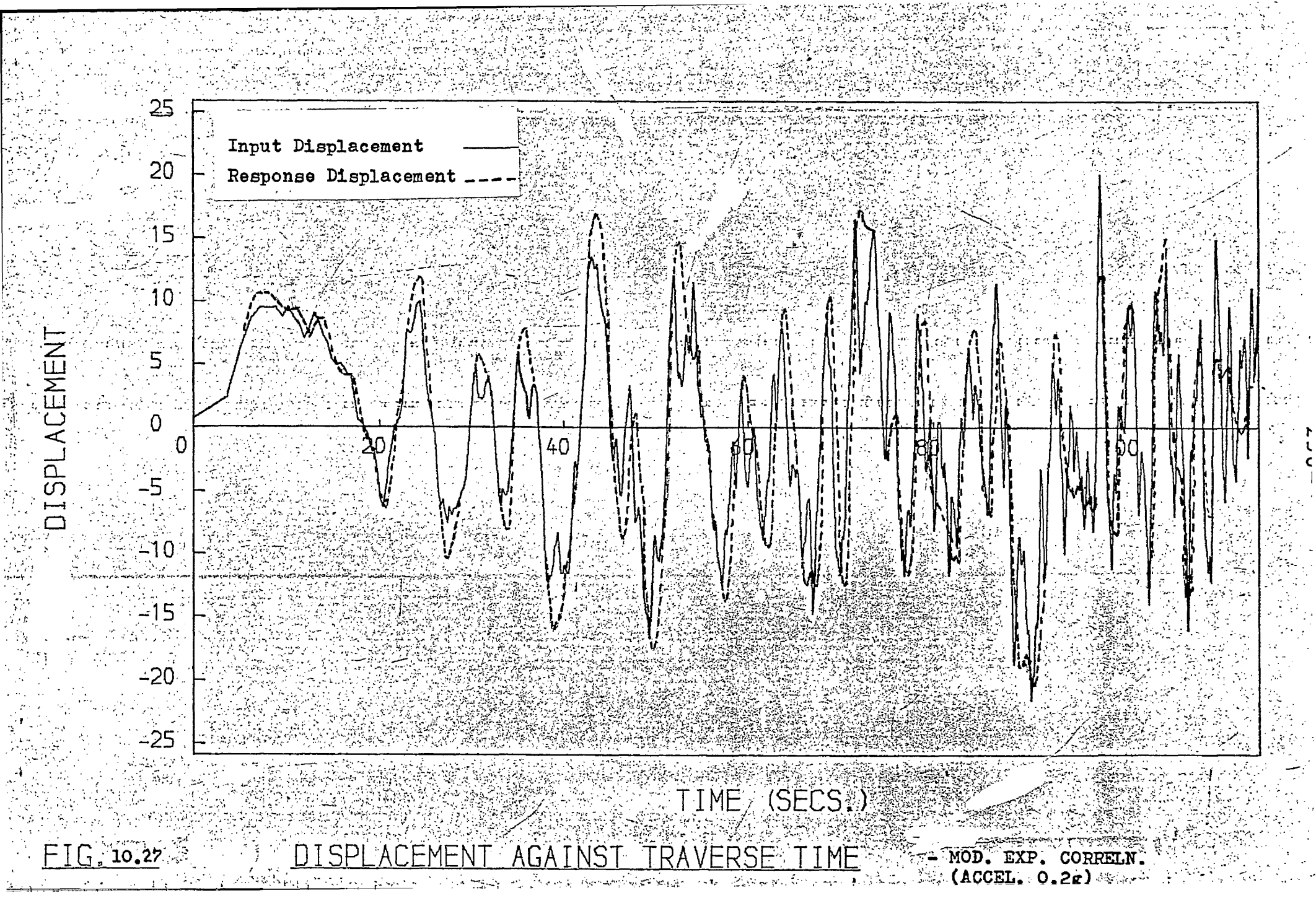


FIG. 10.27 DISPLACEMENT AGAINST TRAVERSE TIME - MOD. EXP. CORRELN. (ACCEL. 0.2g)

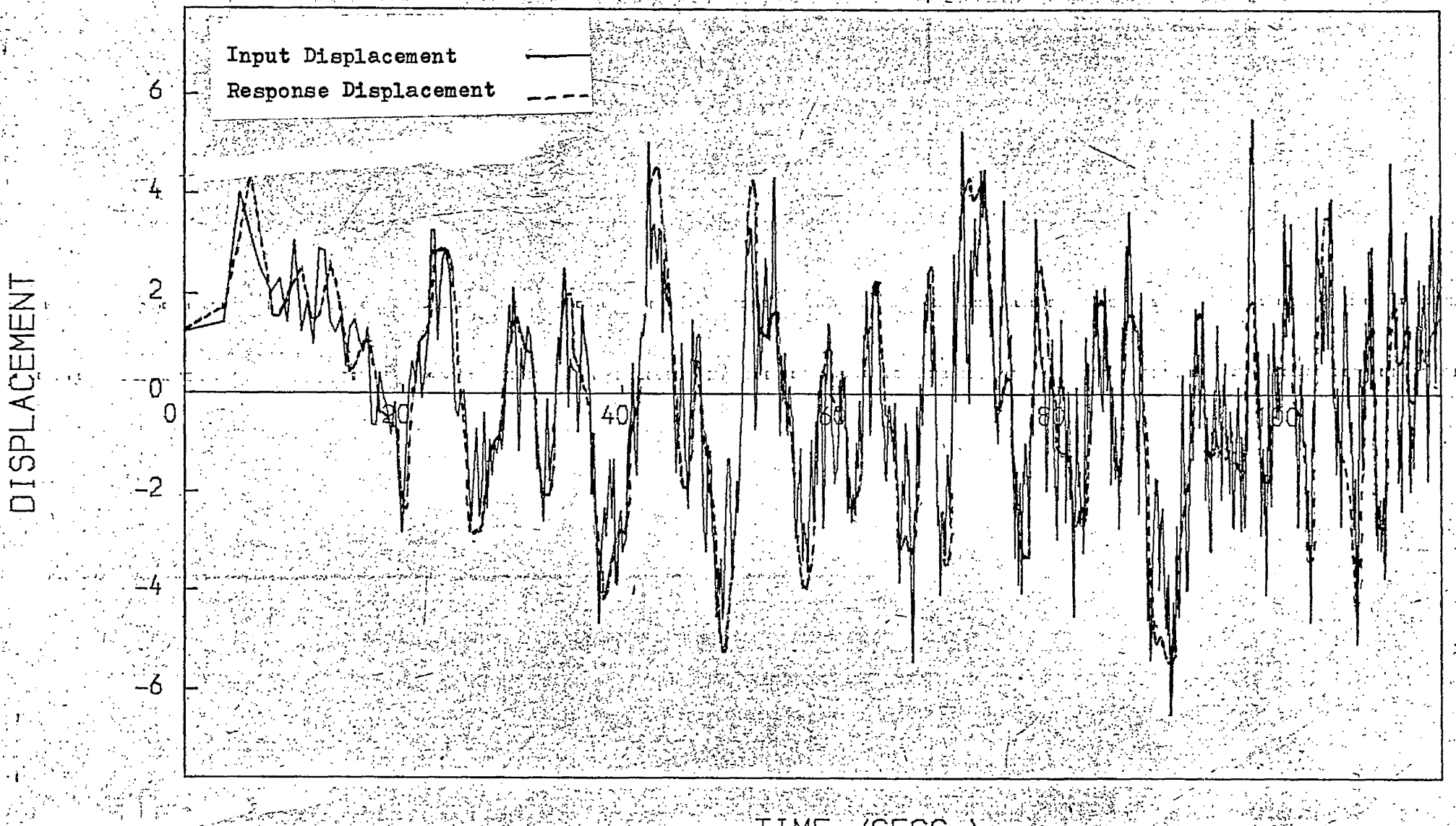


FIG. 10.28 DISPLACEMENT AGAINST TRAVERSE TIME

- EXP. CORRELN. (ACCEL. 0.2g)

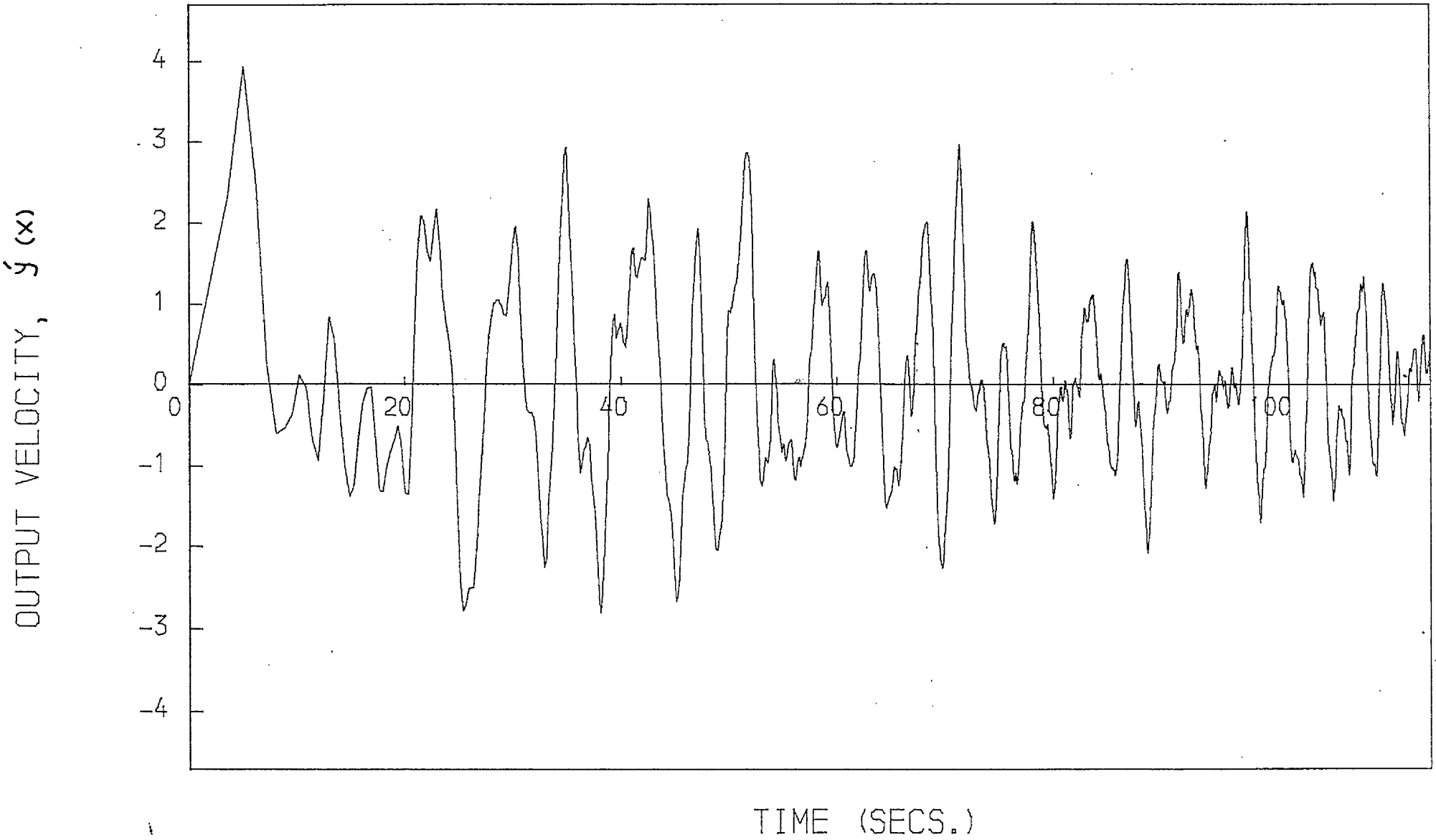


FIG. 10.29 OUTPUT VELOCITY AGAINST TRAVERSE TIME - MOD. EXP. CORRELN. (ACCEL. 0.2g)

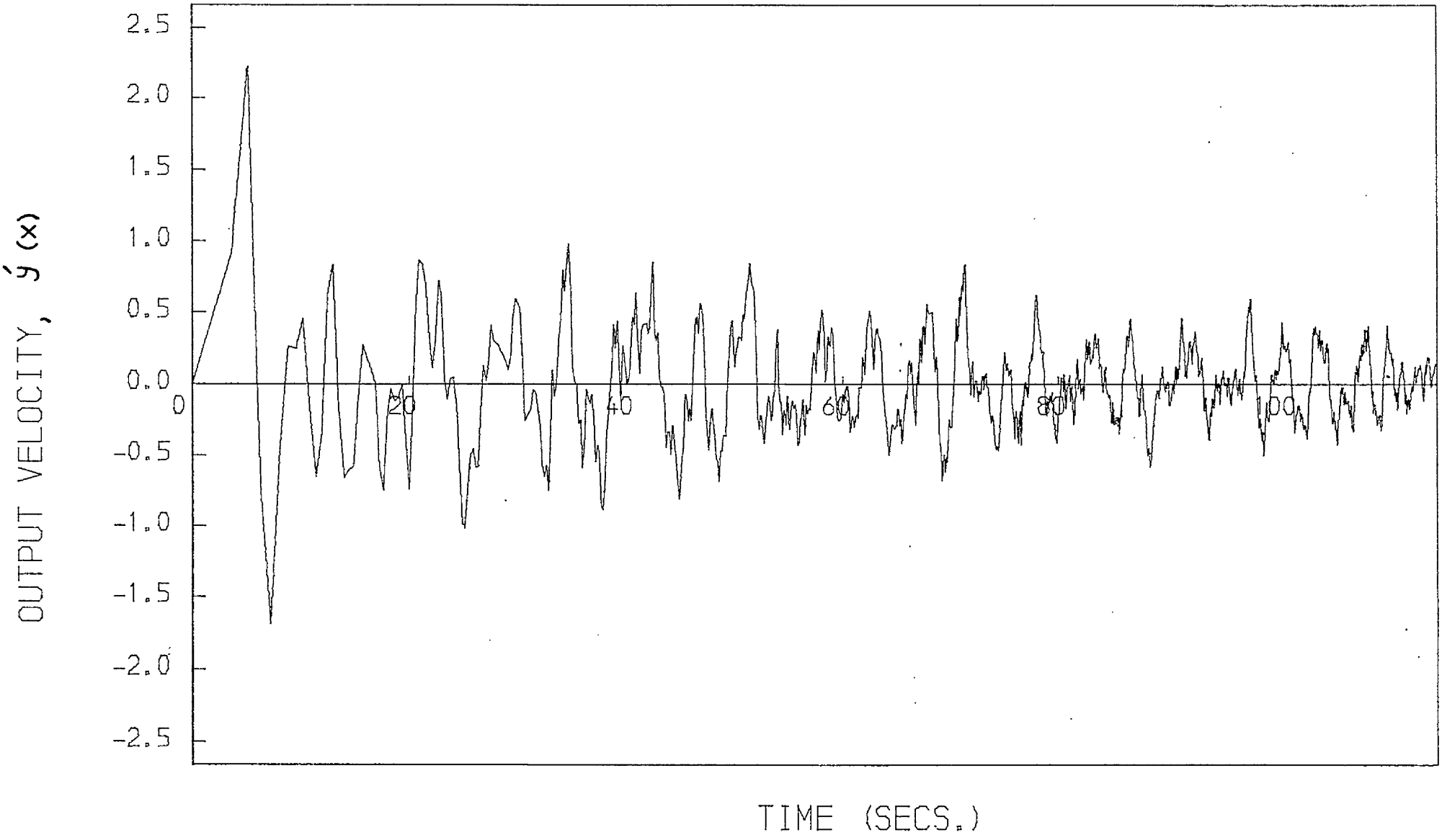


FIG. 10.30 OUTPUT VELOCITY AGAINST TRAVERSE TIME

- EXP. CORRELN. (ACCEL. 0.2g)

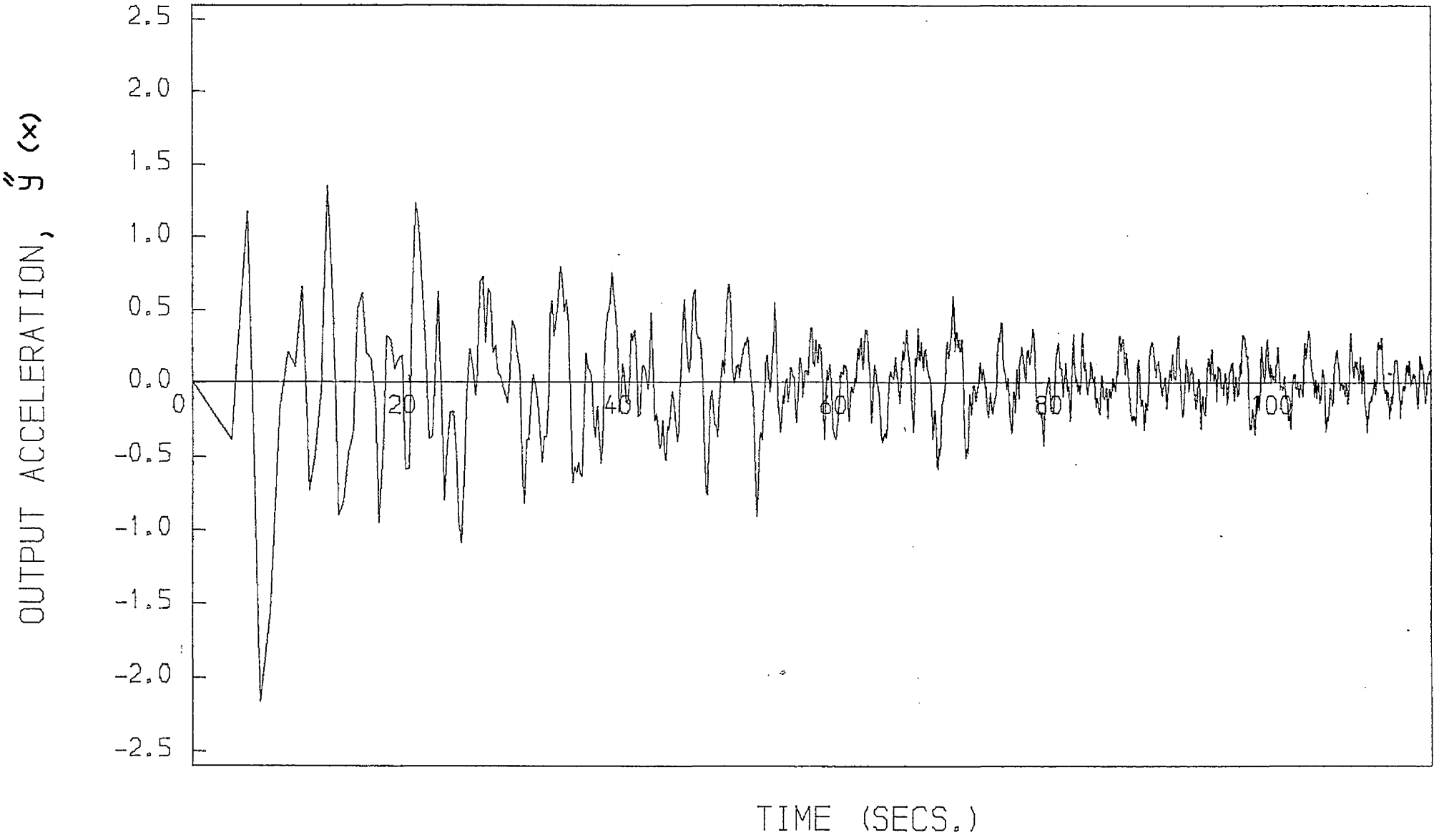


FIG. 10.31 OUTPUT ACCELERATION AGAINST TRAVERSE TIME

MOD. EXP. CORRELN. (ACCEL. 0.2g)

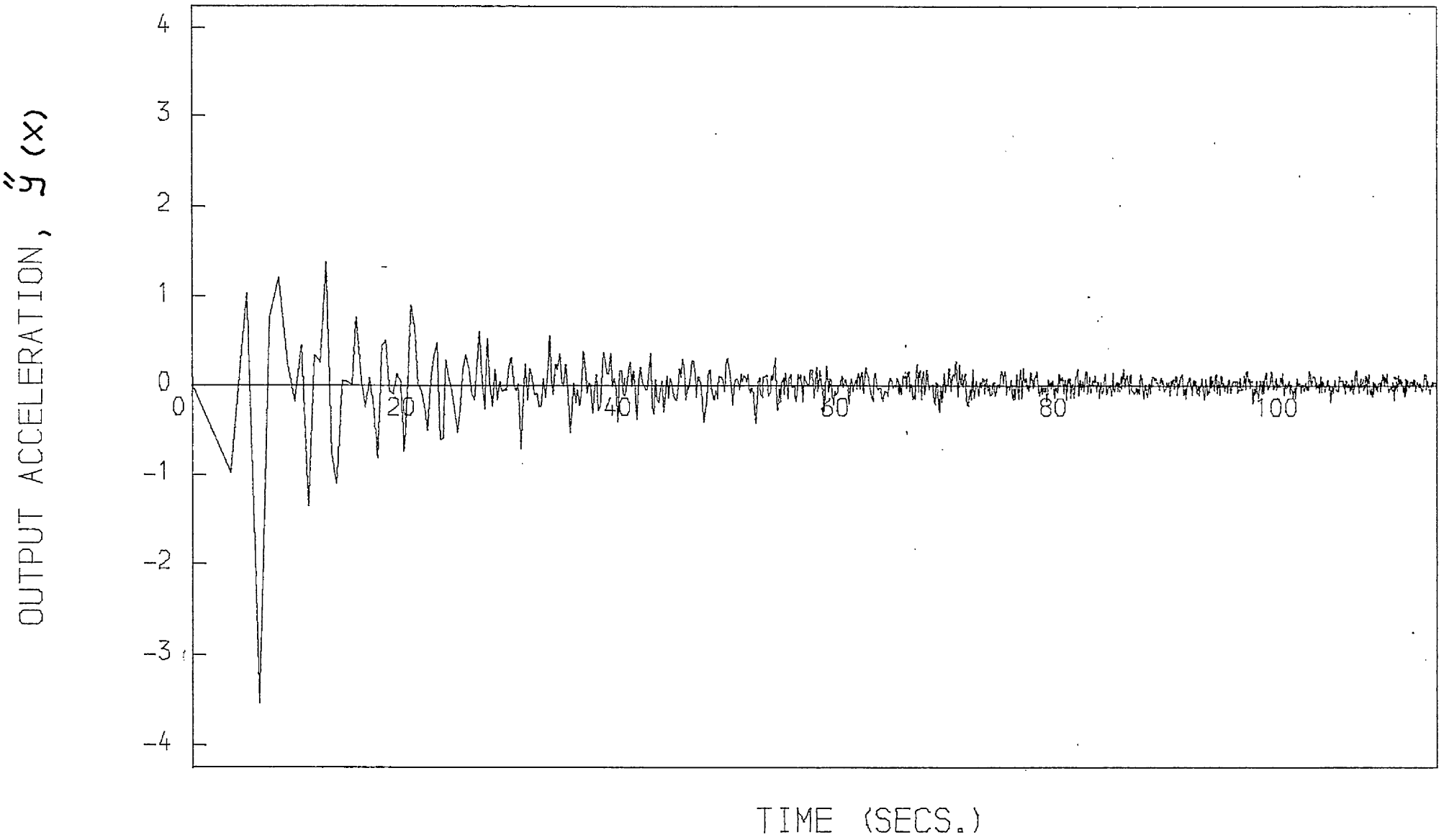


FIG. 10.32 OUTPUT ACCELERATION AGAINST TRAVERSE TIME - EXP. CORRELN. (ACCEL. 0.2g)

CHAPTER XI

A COMPARATIVE APPRAISAL OF SAMPLE AVERAGING

1. Introduction

The ultimate objective of this project has been to determine the viability of averaging a sample of simulated digital computer realisations as a general means of solving the non-stationary vehicle problem. It is therefore necessary to reflect and make a comparative assessment of the strengths, weaknesses and possible potential of the sample averaging approach. The idealised lumped parameter vehicle suspension model was the one used. Random road surface excitations conforming to any road profile correlation can be used.

Clearly such an appraisal is useless if it does not compare the results with alternative approaches. Comparisons with the alternative response convolution integral method are made. The basis of both approaches are summarised in Chapter I, Section 3.

2. A lumped parameter digital computer simulation

The characteristics of vehicle dynamics are such as to enable conceptual acceptance of the lumped parameter model idealisation to be easily accomplished. The relative stiffness and mass ratios of the car body compared with that of the suspension system allows for the assumption that all inertia effects can be attributed to the car body while stiffness and damping effects stem solely from the suspension system.

Road vehicles ride on several suspension units, however only the simulation of a single suspension unit has been undertaken. A great deal of valuable information can be obtained from considering a single unit, besides which, the majority of fundamental problems,

pertinent to establishing the viability of sample averaging, are present in this simple model. It is also true that the only non-stationary solutions available for comparison consider only a one degree model.

For simulation purposes, use of a computer is easily the most viable approach. Model tests lack flexibility when variations in physical model are required. Such models are also much less cost effective. A digital (as opposed to analogue) computer was chosen due to its much greater accuracy and problem independancy properties, together with its universal availability.

Once a decision in favour of digital simulation was made, then it was essential to consider the numerical techniques required. There is no alternative to finite difference formulae for digitally approximating differential equation models.

3. Digital simulation of the road profile

It is normally the vertical component of road surface displacement which is important as a source of vehicle excitation. Suspension geometry can normally be regarded as rigid with regard to all lateral movement, besides which, lateral input excitation contra to the line of vehicle traverse, would either be absorbed in rotational energy to the wheel or damped out by the tyre itself.

There are strong indications that displacement profiles along any particular road are spatially stationary and that probability distributions are normal (as indeed is the derivative profile surface gradient). This spatial stationary quality allows for roads to be simply described by a spatial lag auto-correlation function, $R(L)$, or by the fourier equivalent power spectral density, $S(K)$. The probability distribution is assumed normal.

Interest is normally centred on vehicle response, given a road profile described in this concise auto-correlation form. As this sample averaging technique requires a batch of independant road realisations, it is necessary to generate road profiles with appropriate charac-

teristics. As a large statistically independant sample is required, then the only means capable of achieving this is to use linear stochastic difference equations (LSDE) as profile generators. The number of potential independant realisations is very large indeed, being a function of the word length of the digital computer and of the number of primitive roots available for use in the pseudo random number generator, RNG. Normality of distribution can readily be achieved by any one of many Gaussian digital filters.

4. Emphasis on numerical methods

4.1 General

In the sample averaging approach to this non-stationary problem, heavy emphasis has been placed on numerical considerations. For this there can be no apology. Careful design of these factors is crucial for ensuring the method's flexibility in handling all facets of the non-stationary vehicle problem. It enables the road profile generator to mimic all types of road correlation, whether they are of a physically realistic piecewise continuous type, or of an elegant mathematical form. Careful numerical design also allows the dynamic integrator module to simulate all types of suspension configuration whether linear or not. Upon implementation this vehicle simulator can be applied with equal ease to new road profile/vehicle suspension configurations. This flexibility and ease of use is of major advantage when strict timescales require to be met.

The alternative approach, the response convolution integral method, uses convoluted double response integrals. Their evaluation also relies heavily on numerical methods. However the solution characteristics are such that it is necessary to first of all simplify the integral function and then reassess the validity of the numerical approach to the new response integral. Evaluation requires specialist knowledge which may not always be available.

4.2 Road profile generator

LSDEs were used on the strength of their ability to generate a large

batch of independant road profile realisations. However in a general assessment of LSDE techniques, the exacting constraints of numerical stability and invertibility must be carefully considered prior to making a decision. Autoregressive (AR) filters were chosen because of their ability to match the properties required by the road profile generator.

AR filters can readily mimic high lag value profile correlations without the need to resort to excessively high filter orders. The filter coefficients are readily calculable given the road profile displacement correlation. Although no assurances of numerically stable filter coefficients can be given, stability is easily checked. The checking procedure is readily automated on a computer. It was empirically observed that of all the AR filter coefficients determined, for a selection of profile correlations, instability was never encountered.

While AR filter generators possess poor phase lag properties, this feature is normally regarded as superfluous in the description of a road profile. (In the description of road profiles such information is generally not available.)

AR filters generate discretised profile realisation values at constant spatial increments. In short they exhibit spatially stationary characteristics. It is the intention to simulate the non-stationary response characteristics induced by variable traverse velocity. The non-stationarity can, however, be introduced into the simulation via the dynamic integrator (see subsection 4.3).

The criterion for optimising the filter order was not considered. Other constraints dictate what this order should be. The nature of the displacement correlation curve, whether mathematically elegant or of a piecewise continuous form, influences the decision, as does the incremental step length, $h(x)$, which is optimised for the range of traverse velocities under consideration.

Usually AR filters assume the probability distribution properties of the input signal. Normal distributions can readily be achieved by the introduction of a Gaussian filter. This filtering should be

undertaken immediately following the RNG segment, which generates white noise random data, and prior to processing in the AR filter. Several Gaussian filters were considered, A Polar filter was selected on the basis of its superior numerical properties. Empirical tests carried out on the road profile generator output (ie after AR filtering) proved the results to possess good normal distribution properties.

4.3 Dynamic vehicle integrator

The lumped parameter approach enabled the dynamic system to be idealised mathematically by a set of second order ordinary differential equations (DEs). Thus restricted to solving these initial value DE problems on a digital computer, there was no recourse but to apply finite difference methods.

It is a straightforward matter to reduce a set of N second order DEs to an equivalent first order DE system comprising of 2N similtaneous equations. A decision was made to pursue the first order approach, primarily because of the superior properties in handling round off error. The approach also automatically supplies information on velocity responses without the need to resort to numerical differentiation. Velocity response is very important in mechanical design.

Careful consideration of the numerical factors must be entertained if the desired flexibility of application is to be incorporated into the vehicle integrator. Error build-up and limits, stability, convergence, consistency and arithmetic effort must all be considered. The majority of integrator method types could be rejected outright. This included a method frequently favoured in linear structural dynamics, based on recursive application of the convolution integral, because of the inability to handle non-linear suspension systems. Two types of integration method, namely Runge Kutta (RK) and Predictor Corrector (PC) methods were worthy of closer scrutiny.

Methods of the PC type proved to exhibit superior numerical behaviour. Several key factors formed the basis of this conclusion. The arithmetic effort involved in advancing one incremental step, h, is

is considerably less for the same order of accuracy with PC type methods. This operation is applied repeatedly over a large number of incremental steps; it is therefore not surprising that this criterion is crucial to the choice of method. When both the predictor and corrector formulae are of indentical order, error evaluation is a trivial by-product of the integration process. It is only with substantial additional arithmetic effort that error information can be extorted from RK methods.

Another important consideration when arriving at a decision on method type concerns the ability to change step length continually throughout the simulation process. This property is important when inducing non-stationarity into the dynamic simulator. Non-stationarity is introduced into the model by repeated updates to the "pseudo dynamic parameters", where the traverse velocity may change after every incremental step, $h(x)$. Such variation in these parameters effectively changes the decay constants, T, of the dynamic system and hence for a constant, $h(x)$, the step length ratio, $H(= h(x)/T)$, has effectively changed. Fortunately special forms of PC type methods exist which are viable when step length changes are enacted. Consequently the variable step length criterion does not sway the choice against PC methods.

The issues of stability and convergence prove to be functions of individual integration methods rather than method types. Their behaviour is entirely dependant on the coefficients of the method formula. This is easily demonstrated by performing the stability analysis using the characteristic equation.

Objective comparisons of numerical integration methods are very few in number. Only one study (Hull et al. ref. 6.2) has been conducted on the relevant class of DE problem. Of the PC type, with variable step length facility tested, two methods performed well, namely Krogh (ref. 6.7) and Gear (ref. 6.5). Krogh has the added refinement of a variable integration order facility. This refinement is totally superfluous to requirement and only serves to make the integrator more cumbersome in use. Gear's method better fits the bill, although here too the automatic step length change feature is not required. Consequently a decision was made to implement the basic Nordsieck (ref. 6.6) method on which Gear is founded. However the efficient

predictor algorithm of Gear (ref. 6.11) should be maintained.

At this stage a conclusion in favour of a Nordsieck based integrator algorithm had been reached. Decisions now centred round the refinement of the algorithm to best meet the needs of the dynamic simulator. To do so it was essential to consider the detailed derivation of the Nordsieck formula together with the criteria for optimising stability. A description of the proposed operation sequence for a Nordsieck based algorithm was required. Implementation comprised essentially of three operations; predict (P) by estimating value at next incremental step, evaluate (E) the DE function based on the most up to date value estimate, and correct (c) by comparing the new estimate of the function with its predecessor. The extension to coupled first order DEs was also explained. Equivalence of Nordsieck's algorithm with the Adams - Bashford - Moulton linear multistep method (Osborne ref. 7.3) is crucial to the argument for choice of predict - evaluate - correct sequence. In his book (ref. 7.5), Gear demonstrated that the stability characteristics of any PECE .... sequence is identical in both the Nordsieck form or in its multistep equivalent, while Hull (ref. 7.6) empirically observed that for Adams - Bashford - Moulton linear multistep methods of order less than eleven, $P(EC)^2E$ is in fact the optimum algorithmic sequence. It was proposed to implement only a passive form of error monitoring, the automatic step length change feature not being required.

5. The road vehicle simulator

Detailed design of the final form of the road vehicle simulator was considered. It was necessary to ensure all features of both profile generator and vehicle integrator were correctly incorporated and interfaced.

Most problems stemmed from the difficulties of interfacing a time based integrator to a spatially based generator. It was concluded that a spatial formulation was the only feasible way of dealing with the problems. Consequently non-stationarity was induced into the DEs of the integrator via the "pseudo dynamic parameters." The values of

these parameters change with instantaneous traverse velocity, $\dot{x}$. Updates to the current incremental step length, h(x) and, $\dot{x}$ are obtained from the traverse acceleration profile and the previous value of $\dot{x}$. Normally, it can be assumed that the traverse acceleration profile is explicitly defined.

Singularity points occur momentarily, in this spatial approach, when the traverse velocity, $\dot{x}$, falls to zero. This condition must be specifically tested for and a special routine, based on a Taylor's series expansion incurred for one incremental step only. Afterwards the start-up evaluation algorithm must be reinvoked.

The additional problems caused by interpolative matching of space and time can also be effectively overcome. A solution is achieved by over estimating the number of interpolated profile points used by the integrator. This problem is encountered when traverse velocities, $\dot{x}$, are very low - but non zero.

Ideally, the start-up initial conditions should be random. However the objective is to establish the mean square response characteristics. It was proposed therefore to substitute for these random conditions the stationary mean square response values at the start-up traverse velocity. By so doing it is assumed that the vehicle response is stationary up until the instant of start up. The variation in stationary mean square response with traverse velocity was evaluated for the systems tested, while the calculation procedure for alternative stationary profile correlation/vehicle suspension configurations is outlined.

6. Empirical testing

6.1 General

Empirical tests were required to validate the numerical procedures adopted. Three phases of testing were necessary. One part was to verify the road profile generator, while the second was to do likewise with the dynamic vehicle integrator module. The final phase of testing concerned the statistical behaviour of the entire dynamic vehicle simulator.

6.2 Road profile generator

Road profiles adhering to a selection of displacement auto-correlations, were generated and tested. Two criteria require to be met in order to validate this module. It is essential to ensure generated displacement and velocity profiles adhere closely to the theoretical correlations. Good agreement was obtained for most correlations. Exceptions were observed in the generated derivative correlations where the theoretical equivalent was non-continuous at zero lag (ie not physically realisable). The second criterion concerns probability distribution. Both the generated displacement and velocity profiles adhere closely to the desired theoretical normal condition.

6.3 Dynamic vehicle integrator

The vehicle integrator tests were of a deterministic nature. They were designed to vet the salient features of the integrator module while providing a theoretical solution with which to compare the simulated results.

For a realistic vehicle configuration, it was observed that a step length ratio, H, of 0.01 was approximately optimal.

The result of sudden large discontinuity tests also proved favourable. Despite the simulated response being thrown off true it quickly recovered, requiring only several incremental time steps. Such discontinuity tests are important to validate the integrator's ability to cope with rapid large fluctuation in vehicle input excitation. Such fluctuations are almost certain to occur occasionally when the simulator is excited by random road data. It is sudden large discontinuity events which tax the integrator to its limits.

The availability of theoretical comparisons proved invaluable for establishing the empirical error monitoring criterion which would provide best insight into the simulator's performance. This turned out to be the failure frequency of the stability test.

6.4 Road vehicle simulator

Upon the marriage of the two segments of profile generator and vehicle simulator, additional statistical testing was required on the complete simulator. In this way the solutions to the spatial/time base interface problems could be vetted.

Two phases of testing were necessary. The first concerned stationary tests. For this purpose response spectral tests were sufficient (probability distribution tests on the road profile generator having already been evaluated). The answers were in good agreement with those obtained from classical random vibration theory.

Non-stationary tests comprised the other part of statistical validation. General ability to determine non-stationary response characteristics is after all the ultimate aim of this project. This phase of evaluation is much more complex. The various aspects are dealt with in their own right later in the chapter.

7. Quality of the non-stationary results

7.1 General

Simulations were conducted using generated road data which adhered to one of two test correlations, either Exp or Mod Exp correlation. In both cases the road vehicle started from rest and was subjected to uniform acceleration. The size of the resultant non-stationary realisation sample was 50. This limited sample size was due primarily to constraints imposed by the computer system.

In non-stationary analysis, the information of interest comprises of the sample averaged mean square response values of displacement, velocity, and acceleration, together with instantaneous probability distributions across the sample of realised records.

Very few non-stationary results are available for comparison (Virchis/Robson ref. 1.2, Sobczyk/Macvean ref. 1.3). The conclusions reached concern only mean square (MS) displacement response.

Both concur that for typical vehicle traverse acceleration, the effects of non-stationarity on mean square displacement response are marginal (only a few per cent different from stationary equivalent). No results are available concerning MS velocity or acceleration responses. Neither is there any on probability distribution. The amount of comparison checking which can be attempted is therefore very limited. Fortunately however viable displacement responses obtained by this simulation technique also mean viable derivative responses. The velocity response must be correct to enable correct evaluation of displacement (this fact follows automatically from the coupled nature of the first order DE pair).

7.2 Sample averaged mean square results

Upon sample averaging the MS input displacement and velocity excitations, considerable scatter about the "averaged mean square" value was observed. Surveillance of input scatter permits a better feel for the response scatter which can reasonably be expected.

Datum values of stationary MS response displacement, velocity and acceleration for varying traverse velocity were evaluated.

The alternative solution non-stationary MS displacement results demonstrated that the non-stationary deviation is minimal and clearly beyond the resolution of this technique unless for very large sample sizes. Consequently if the stationary MS displacement response curves are assumed to be the "averaged mean square" response curves, the scatter present in the sample averaged results appears fairly evenly distributed either side of these curves. The scatter amplitude is only marginally greater than observed in the input results. It must be concluded, therefore, that the sample average mean square (SAMS) displacement responses do not contradict the results obtained by the alternative solution method.

No alternative non-stationary MS velocity response results exist. Here the SAMS responses drop off sharply with increasing traverse velocity. The absolute magnitude of statistical scatter is very much reduced enabling more accurate bounds on the estimate of "averaged MS" velocity response. The difference between these solutions and

their stationary counterparts are minimal.

Non-stationary MS acceleration results are also in good agreement with their stationary counterparts.

The crudeness of the mathematical road description defined by the Exp correlation profile meant the derivative velocity profile was not physically realisable. However the similarity in the response conclusions reached for both the Exp and Mod Exp correlations demonstrated the flexibility of the sample averaging technique in dealing with very crude idealisations of road profile description. Such occurences are often encountered in the real world.

7.3 Probability distribution

The sample averaging technique readily yields information on the probability distribution of all response quantities. Data was sampled across the sample at arbitrary fixed instants on the realisation record. By comparing MS values with a theoretical Chi-Square distribution of order one, it was easy to verify normality of the response distributions.

Further vindication of the reasonable statistical nature of the sample averaged results was obtained from the implementation of an F - test.

8. Suggested sample averaging technique

In hind-sight it is easy to see that the decisions made for retaining data was not optimal. The decision had been made to ensure a selection of all data was retained to enable every possible type of subsequent post-processing interpretation to be accomplished. However it would have proven more beneficial to retain only a running total of sample averaged MS information at every point and discard the raw realisation data of each simulation record. A little care in the design of such a sample averaging processor would ensure the elimination of possible numerical ill-conditioning problems.

Probability distribution information should also be retained at regular traverse velocity increments but certainly not at all generated data points.

9. Inferences from a single non-stationary realisation

The similarity between the results of a single realisation record and those obtained by sample averaging is very strong. Naturally enough this leads to the question concerning the amount of information which can be extracted from a single realisation record.

Non-stationary MS displacement responses peak marginally above the input displacements. It is tentatively suggested (based on the conclusions of alternative solutions and those obtained by sample averaging) that this indicates a response at traverse velocities greater than zero, which is only marginally higher than input displacement. It is difficult to be more precise. In any event the non-stationary deviations are marginal.

A strong similarity exists between the sample averaged MS velocity response and the results observed from a single realisation record. Reasonably accurate inferences as to the non-stationary behaviour can be obtained from such a single record. A crude envelope curve fit on a single MS response realisation would certainly indicate non-stationary behaviour and also yield a worst case estimate of MS response.

Once again with acceleration responses, a marked similarity is exhibited between MS ensembled results and those of a single realisation. The argument for a crude envelope fit applies here too.

Attempts to produce more accurate inferences from a single realisation by means of least squares fitting failed to produce meaningful results. This was due to the ill-conditioned nature of the random realisation data. However this aspect was not exhaustively studied and is probably worthy of closer investigation.

As far as probability distribution is concerned no analytic information can be extracted from a single realisation. However the

probability distribution for the sample realisations for the system configurations tested, appeared normal. It is likely that normality of distribution can be assumed for all linear suspension systems.

10. Inclusion of non-linearity

Extending the integrator to include non-linear effects is a trivial matter. As demonstrated in Chap. IX sect. 7, it is merely a matter of altering the function evaluation (E) stage of the integration algorithm. This is a simple arithmetic exercise even for pairs of coupled first order DEs.

A non-linear capability greatly enhances the flexibility and appeal of this simulation approach. There are numerous examples of dynamic road vehicles possessing significant non-linear properties. Automobiles for example normally have dampers which are far from linear. Aircraft suspension systems during take-off or landing exhibit physical characteristics which although not truly non-linear are time or traverse distance dependant. This simulation technique is applicable to both these non-linear non-stationary problems.

In cases where non-stationarity is not applicable, non-linear problems require solving in the time, as opposed to frequency, domain. As far as this simulation technique is concerned stationarity is simply a sub set of non-stationarity and is therefore readily solvable. However in stationary problems the ergodicity principle is applicable. Consequently there is no need to sample average a set of realisations - a single realisation will suffice.

11. Extension to multi freedom systems

Extension of the dynamic integrator to handle multi degree of freedom (multi freedom) systems is straightforward. Instead of solving for a single pair of coupled first order DEs, a solution for a multi membered set of coupled first order DEs is required. The extension to multi freedom simply adds to the complexity of the function evaluation (or E step) of the DEs, but as no matrix inversion is required

there is no essential difference.

To extend the road profile generator to deal with multi freedom input excitation is also trivial, provided the restriction is placed that all suspension units traverse the same road profile track. Differences in input excitation to the various units are then readily introduced by incorporating phase lag at the appropriate suspension units. Unfortunately extending the generator to handle multi track simulations is not nearly so straightforward. Cross correlation relationships between tracks would require incorporation into the model. A closer look at multivariate linear stochastic difference equations would be necessary. ARMA type filters are most likely to fulfil this requirement.

12. A comparative assessment of non-stationary solution methods

The objective of this project is to assess the viability of the sample averaging (SA) method as a means of solving non-stationary vehicle suspension dynamics. To make an effective appraisal it is essential to compare this approach with the alternative response convolution integral (RCI) method.

12.1 Accuracy of MS response results

Consider first the quality of the MS results made available by both approaches. Only displacement MS responses are available from the RCI method. It did however produce definitive displacement responses (accurate to within the limits of the numerical techniques applied to integral evaluation). The SA method produced displacement results with significant statistical scatter. The non-stationary deviation, for typical vehicle accelerations, is very marginal and well beyond the resolution of the SA method unless for very large sample sizes. It can be argued, with reasonable justification, that failure of the SA method to detect non-stationary deviation indicates that non-stationary considerations are insignificant (remember the scatter results were fairly evenly distributed either side of the stationary MS response curves).

For a uniformly accelerating road vehicle with a linear suspension configuration, the effects of non-stationarity are marginal and beyond the resolution of the SA technique for any practical sample size. This may not be generally true for a vehicle experiencing variable traverse acceleration or possessing non-linear suspension characteristics. However as far as a definitive solution is concerned the RCI method is preferable although the SA approach can readily produce information of sufficient quality for design purposes. It is the ease of obtaining a variety of design parameters (discussed below) which proves to be the most salient aspects of the SA technique.

12.2 Probability distribution

Only the SA method can yield useful information on the non-stationary probability distribution of all response quantities; displacement, velocity, and acceleration. Probability distribution is important in fatigue analysis.

12.3 Ease of solution

Once the dynamic vehicle simulator has been developed it is a relatively easy matter to obtain results using the SA method, for any road profile/vehicle suspension configuration. All statistical information concerning displacement velocity and acceleration responses is readily available. The RCI method requires the complex double integral to be first simplified by hand and then numerically evaluated. The MS displacement response is thus obtained. If an MS velocity response is required, the entire process must be repeated and similarly for the MS acceleration. Should the road displacement correlation be approximate such that no analytic derivative correlation exists, the RCI method cannot be used to achieve derivative MS response answers. The SA method requires no such analytic niceties to achieve reasonable approximate solutions.

12.4 Non Linear capability

Non linearities are easily included in the SA method. The RCI method which invokes the convolution integral for the solution of the responses relies on the assumption of linear superposition. Consequently it is not applicable to non linear suspension configurations.

12.5 Extension to multi freedom systems

With every additional degree of freedom incorporated into the vehicle model, the RCI method becomes more and more intractible. (Every additional freedom requires an extra set of auto correlation integral expressions together with an extra set of cross correlation integral expressions). The SA method is readily applicable to multi freedom simulations of the single road track excitation type. Extension to multitrack simulations requires a more detailed study of multivariate LSDEs.

12.6 Summary

To conclude, the RCI method produces highly difinitive answers, while the SA method does not. However SA technique can readily yield information of sufficient quality for design purposes.

The SA method is easy to use and capable of being applied to most road profile/vehicle suspension configurations. The RCI method is rather inflexible. Flexibility and ease of application are major assets when used in an industrial environment where strict timescales normally require to be met.

All statistical information is readily available via the SA method. With the RCI method each MS response requires a separate integral evaluation. Generally displacement velocity and acceleration responses are important to the design engineer. Internal mechanical forces in the vehicle are dependant on all three factors.

Probability distribution information is only available using the SA method. This factor is vital in fatigue analysis.

13. A statistical simulator - a third alternative?

The simulation philosophy applied in this SA technique relied on the development of two distinct numerical modules; a road profile generator and a dynamic vehicle integrator. Both modules are discretised approximations to the real thing, and are in essence digital filters. The question as to whether the processes effected by each module could be combined into one module, must be raised. This which in effect is a statistical simulator, would take the form

$$\sum a_i y_i = \sum b_j r_j$$

where "y_i" is the displacement response at incremental step number "i", "r_j" is the uncorrelated random input at increment "j", and coefficients "a_i" and "b_j" are combined functions of the road profile correlation, the dynamic parameters of the system and the traverse velocity profile.

Such an approach might possibly reduce the amount of numerical processing. However this must be weighed against the fact that neither velocity nor acceleration response information is directly available. This information would have to be obtained from another statistical filter or through differentiating the output process. No work has ever been attempted in this approach and consequently it will be considered no further.

14. In conclusion

It is now necessary to stop and take stock of the advantages and weaknesses of this sample averaging technique. It is also time to reflect on the method's future potential as the result of further developments in the simulator and also because of application to new areas.

As discussed in the text of this thesis, the development of the numerical techniques to solve the non-stationary problem by sample averaging was not without difficulty. In fact it comprised the major part of this project. However development is a once only operation and when completed a flexible analysis tool is available for ready application to most road profile/vehicle suspension configurations. It is the flexibility and ease of application which comprise two of the three major assets of the sample averaging technique. The other major advantage of the method stems from the fact that all statistical information is readily available.

This latter fact is important for full appreciation of design problems. In mechanical systems all internal forces in the vehicle are dependant on either displacement, velocity or acceleration responses. In vehicle design this information is relevant to both stress analysis and ride behaviour.

The SA approach is not an unqualified success. This fact was evident by the amount of statistical scatter present in the sample averaged MS displacement responses. The marginal non-stationary effects for linear suspension system undergoing uniform acceleration lie beyond the resolution of the SA method. This may not be generally true. However the information is of sufficient quality for design purposes. The alternative RCI method produced definitive MS displacement response results, but as yet that is all it has produced.

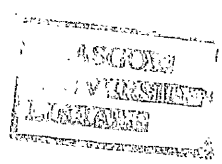
Information on the response probability distributions is well defined by this SA method. No such information is available from the RCI method. Response probability distributions are of crucial importance when estimating fatigue life of vehicle components.

MS response results from a single realisation can yield crude design information as to worst case estimates in situations where sample averaging is not always practicable. The strong similarity which exists between single realisation records and sample averaged MS results indicates that the potential information available from a single record has not been fully exploited.

Extension of the road profile generator to include the multitrack model should be considered. This would make the simulator applicable to all multi degree of freedom suspension configurations.

Other than the automobile industry, the simulator can be applied to any non stiff dynamic system subject to similar non stationary behaviour. (A non stiff dynamic system is one in which all the idealised natural frequencies of the model lie within a narrow frequency band.) Aircraft suspensions during take-off or landing are ideal examples.

In conclusion therefore the sample averaging approach to the non-stationary vehicle suspension is not an outright winner. Whether or not this approach should be adopted depends very much on which criteria sway the balance. The RCI method produces highly definitive results. The SA method is easy and flexible in application and all relevant statistical design parameters are instantly available.



LOVELL, J. M.

Ph. D. Thesis 1981

COMPUTER PROGRAM DOCUMENTION

Character Set used in Program Listings

The attached computer listings are non standard and utilise a subset of the full Algol 60 character set. While in most cases the differences are obvious clarification is required for many of the non alpha numeric characters - such as " > ", " £ ", etc. A list of the more obscure symbol definitions follows.

Arithmetic

£ >	Assignment	(:=)
:/:	Integer division	
<	Addition	(+)
–	Subtraction	(–)
*	Multiplication	(×)
* *	Power	(↑)

Logicals

>	Equal to	(=)
\	Greater than	(>)
)	Less than	(<)
^	Not	(¬)
^ >	Not equal to	(≠)
\ >	Greater or equal to	(≥)
) >	Less or equal to	(≤)
+	And	(∩)
:OR:	Or	(∪)

Brackets and Miscellaneous

[/ ..... /]	Brackets around array identifiers
[- - - - -]	Brackets around procedure parameters
[:[: ... :]:]	Brackets around a write text format
£	Separates array bounds (:)eg ARRAY A[/1 £ 5/]

Program ARFILT

PURPOSE:

This program is used to determine the coefficients of an autoregressive filter with prescribed autocorrelation characteristics. A discretised description of the correlation profile is input at regular lag increments, up to and including the highest lag value of interest. In the process of establishing the filter coefficients the necessary tests are undertaken to ensure that the input correlation is positive definite and that the recursive autoregressive filter obtained is numerically stable. (For a fuller description of the program refer to Ph.D. Thesis Chapter III section 5. The flow logistics are shown schematically on the attached figure 3.4).

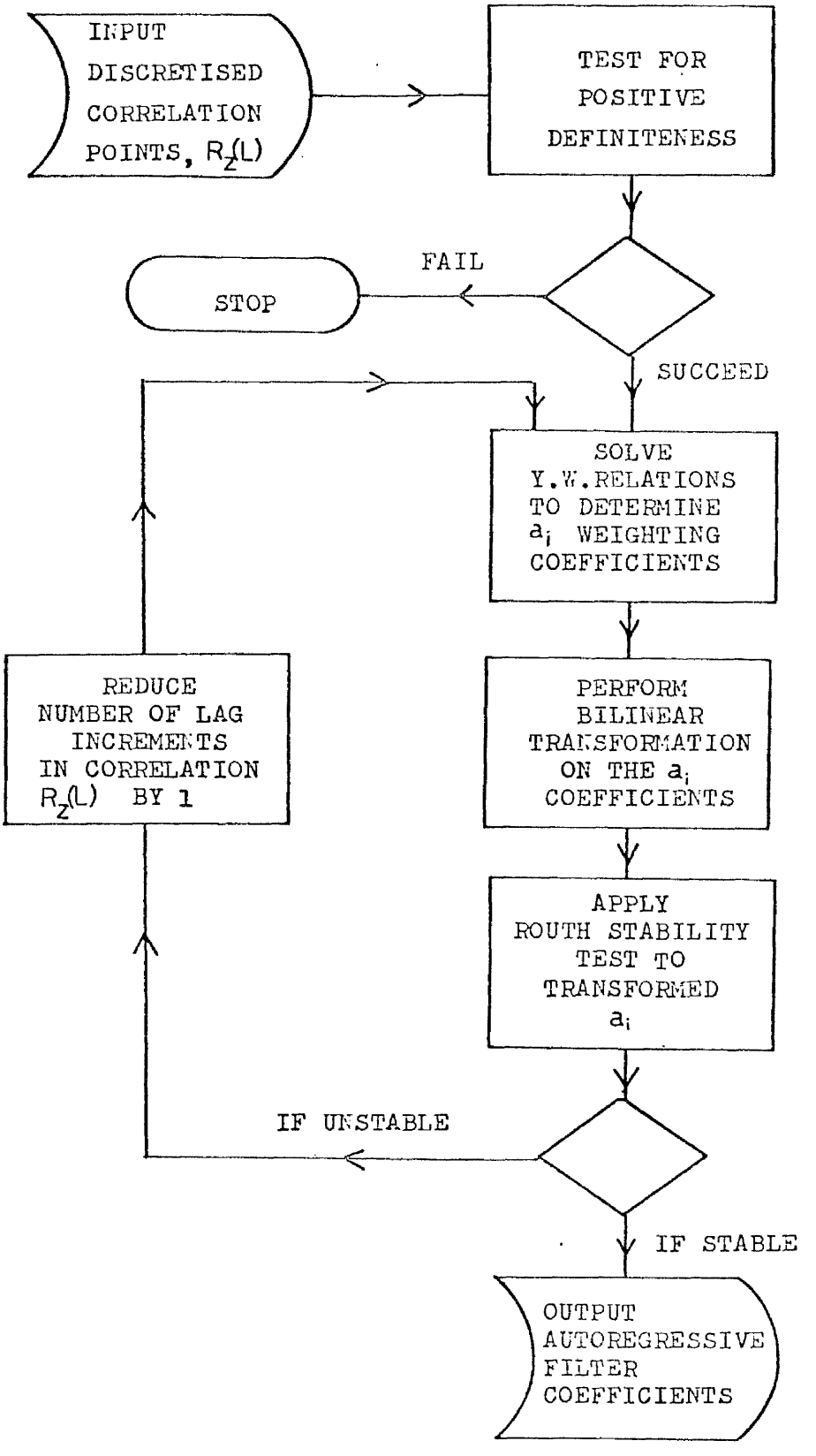


FIG 3.4 Program flow logistics of the AR Filter coefficient determination procedure.

PROGRAM ARFILT

```
Z63JL*UNIFIL(1).ARFILTDELFT
      1       :BEGIN:                                                                  00000000
      2                                                                                00000001
      3                                                                                00000002
      4                                                                                00000003
      5       :COMMENT: PROGRAM TO DETERMINE COEFFICIENTS OF  MULTIVARIABLE A.R.       00000004
      6        FILTERS WITH PRESCRIBED AUTO + CROSS CORRELATIONS IN TURN FOR EACH      00000005
      7        FILTER , THE PROGRAM TESTS FOR&-                                        00000006
      8        1) THE PHYSICAL REALISABILITY BY A TEST FOR POSITIVE DEFINITNES         00000007
      9        2) THE STABILITY BY USING A SUITABLE BILINEAR TRANSFORMATIONON THE      00000008
     10           COEFFICIENTS + APPLYING THE ROUTH STABILITY CRITERION.               00000009
     11        SHOULD THE COEFFICIENTS FAIL TO MEET THE TEST CRITERIA , THE ORDER OF   00000010
     12        ALL THE FILTERS IS REDUCED AN APPROPRIATE AMOUNT + THE PROGRAM IS       00000011
     13        RESTARTED AT THIS REDUCED FLTORD      ;                                 00000012
     14                                                                                00000013
     15                                                                                00000014
     16                                                                                00000015
     17       :INTEGER:FLTORD,NOFILT,TOTORD,COLUMN,Z,ZZ,ROW,KP,JP,IP,INCRP,            00000016
     18                ORIG;                                                           00000017
     19       :REAL:DET;                                                               00000018
     20       :BOOLEAN:POSDEF,PUT;                                                     00000019
     21                                                                                00000020
     22       :COMMENT:   LIST OF PROCEDURES                                           00000021
     23                   ********************;                                       00000022
     24                                                                                00000023
     25       :REAL::PROCEDURE:INNERP(U,V,K,S,F);                                      00000024
     26                                                                                00000025
     27       :VALUE:S,F;                                                              00000026
     28       :INTEGER:K,S,F;                                                          00000027
     29       :REAL:U,V;                                                               00000028
     30       :BEGIN:                                                                  00000029
     31           :REAL:H;                                                             00000030
     32            H&>0;                                                               00000031
     33           :FOR:K&>S:STEP:1:UNTIL:F:DO:                                         00000032
     34                H&>H<U*V;                                                       00000033
     35            INNERP&>H;                                                          00000034
     36       :END:   INNERP;                                                          00000035
     37                                                                                00000036
     38                                                                                00000037
     39                                                                                00000038
     40       :PROCEDURE:CROUT2(A,B,M,N,Y,DET,POSDEF,OUT,PUT);                         00000039
     41                                                                                00000040
     42       :COMMENT: THIS PROCEDURE SOLVES A SET LINEAR SIMILTANEOUS EQUATIONS      00000041
     43            OF THE FORM -  @A@*:Y:>:B:                                          00000042
     44                 WHERE @A@ IS AN [M-N]*[M-N] DIMENSIONAL MATRIX                00000043
     45                       :B: IS AN [M-N] DIMENSIONAL VECTOR                      00000044
     46                       :Y: IS AN [M-N] DIMENSIONAL VECTOR .. THE REQUIRED      00000045
     47                        SOLUTION.                                              00000046
     48            THIS ALGORITHM IS BASED  ON CROUT2 AS DESCRIBED IN C.A.C.M. 1961   00000047
     49            ALGORITHM NO. 43                                                    00000048
     50            THE ALTERATIONS ARE MINOR + IN ESSENCE CONSIST OF -                 00000049
     51                 INSERTING THE @OOLEAN POSDEF @HICH IF &>:TRUE: MERELY          00000050
     52                 EVALUATES THE :DETERMINANTS:                                   00000051
     53                                                                                00000052
     54       :VALUE:M,N,A,B;                                                          00000053
     55       :REAL:DET;                                                               00000054
     56       :INTEGER:M,N;                                                            00000055
```

```
57      :ARRAY:A,B,Y;                                                                    00000056
58      :BOOLEAN:POSDEF,PUT;                                                             00000057
59      :LABEL:OUT;                                                                      00000058
60      :BEGIN:                                                                          00000059
61          :INTEGER::ARRAY:PIVOTE/M&N/3;                                                00000060
62          :INTEGER:K,I,J,IMAX,P,INCR,DIM3C;                                            00000061
63          :REAL:TEMP,QUOT;                                                             00000062
64          :IF:^POSDEF + PUT:THEN:                                                      00000063
65          :BEGIN:                                                                      00000064
66                %RITETEXTE:E:B VECTOR:3:3;                                             00000065
67                NEWLINEE1;                                                           00000066
68                OUTPUTEBE/I/3,DIM3C,DIM3C,1,M,N,INCR,3,8;                            00000067
69                NEWLINEE1;                                                           00000068
70                %RITETEXTE:E:A MATRIX:3:3;                                           00000069
71                NEWLINEE1;                                                           00000070
72              :FOR:J&>M:STEP:1:UNTIL:N:DO:                                           00000071
73              :BEGIN:                                                                00000072
74                    PRINTEJ,3,0;                                                     00000073
75                    NEWLINEE1;                                                       00000074
76                    OUTPUTEAE/J,I/3,DIM3C,J,1,M,N,INCR,0,8;                          00000075
77              :END::                                                                 00000076
78               NEWLINEE1;                                                          00000077
79          :END::                                                                     00000078
80           DET&>1;                                                                   00000079
81          :FOR:K&>M:STEP:1:UNTIL:N:DO:                                               00000080
82          :BEGIN:                                                                    00000081
83                TEMP&>0;                                                             00000082
84              :FOR:I&>K:STEP:1:UNTIL:N:DO:                                           00000083
85              :BEGIN:                                                                00000084
86                    AE/I,K/3&>AE/I,K/3-INNERPEAE/I,P/3,AE/P,K/3,P,M,K-1;              00000085
87                  :IF:ABSEAE/I,K/33\TEMP:THEN:                                       00000086
88                  :BEGIN:                                                            00000087
89                        TEMP&>ABSEAE/I,K/33;                                         00000088
90                        IMAX&>I;                                                     00000089
91                  :END::                                                             00000090
92              :END::                                                                 00000091
93               PIVOTE/K/3&>IMAX;                                                     00000092
94              :IF:IMAX^>K:THEN:                                                      00000093
95              :BEGIN:                                                                00000094
96                    DET&>-DET;                                                       00000095
97                  :FOR:J&>M:STEP:1:UNTIL:N:DO:                                       00000096
98                  :BEGIN:                                                            00000097
99                        TEMP&>AE/K,J/3;                                              00000098
100                       AE/K,J/3&>AE/IMAX,J/3;                                       00000099
101                       AE/IMAX,J/3&>TEMP;                                           00000100
102                 :END::                                                           00000101
103                 :IF:^POSDEF:THEN:                                                00000102
104                 :BEGIN:                                                          00000103
105                       TEMP&>BE/K/3;                                              00000104
106                       BE/K/3&>BE/IMAX/3;                                         00000105
107                       BE/IMAX/3&>TEMP;                                           00000106
108                 :END::                                                           00000107
109             :END::                                                               00000108
110             :IF:AE/K,K/3>0:THEN:                                                 00000109
111             :BEGIN:                                                              00000110
112                   %RITETEXTL:E:SINGULAR:3:3;                                     00000111
113                   NEWLINEE2;                                                     00000112
```

```
                    :GOTO:OUT:                                                  00000113
                :END:;                                                          00000114
                 QUOT&>1/AE/K,K/J;                                              00000115
                :FOR:I&>M<K:STEP:1:UNTIL:N:DO:                                  00000116
                      AE/I,K/J&>QUOT*AE/I,K/J;                                  00000117
                :FOR:J&>M<K:STEP:1:UNTIL:N:DO:                                  00000118
                      AE/K,J/J&>AE/K,J/J-INNERPE AE/K,P/J,AE/P,J/J,P,M,K-1J;    00000119
                :IF:^POSDEF:THEN:                                               00000120
                      BE/K/J&>BE/K/J-INNERPE AE/K,P/J,BE/P/J,P,M,K-1J;          00000121
           :END:;                                                               00000122
            #RITETEXTE:E:VALUE OF DETERMINANT IS > :J:J;                        00000123
           :FOR:K&>N:STEP:-1:UNTIL:M:DO:                                        00000124
                 DET&>AE/K,K/J*DET;                                             00000125
           :IF:^POSDEF:THEN:                                                    00000126
           :FOR:K&>N:STEP:-1:UNTIL:M:DO:                                        00000127
                YE/K/J&>BE/K/J-INNERPE AE/K,P/J,YE/P/J,P,K<I,NJJ/AE/K,K/J;      00000128
            PRINTEDET,0,12J;                                                    00000129
           NEWLINEEJ;                                                           00000130
           :IF:^POSDEF:THEN:                                                    00000131
           :BEGIN:                                                              00000132
                 #RITETEXTE:E: Y VECTOR:J:J;                                    00000133
                 NEWLINEEJ;                                                     00000134
                 OUTPUTEYE/I/J,DIM3E,DIM3E,I,M,N,INCR,0,8J;                     00000135
                 NEWLINEEJ;                                                     00000136
           :END:;                                                               00000137
      :END: OF CROUTZ ;                                                         00000138
                                                                                00000139
                                                                                00000140
      :PROCEDURE:TRNSFMEFI,#FI,Q,NJ;                                            00000141
                                                                                00000142
      :COMMENT: THIS PROCEDURE TRANSFORMS THE COEFFICIENTS OF A CHARACTERISTIC  00000143
          POLYNOMIAL EQUATION IN L INTO A FORM WHICH SATISFIES THE ROUTH         00000144
          CRITERION. THE TRANSFORMATION L>EW<IJ/EW-1J IS USED...                00000145
          I.E.  FIEOJ*L**N<FIE1J*L**EN-1J<....<FIENJ>0                         00000146
          BECOMES                                                               00000147
                #FIEOJ*#**N<#FIE1J*L**EN-1J<...<#FIENJ>0                        00000148
          ....SEE JURY  PP.34   ;                                               00000149
                                                                                00000150
      :VALUE:Q,N;                                                               00000151
      :ARRAY:FI,#FI;                                                            00000152
      :INTEGER:N,Q;                                                             00000153
      :BEGIN:                                                                   00000154
          :INTEGER:J,K,L,IA,IIA,IB,INCR,QJ,NQ,SIGN;                             00000155
          :REAL:FACTA,FACTB,PROFCT;                                             00000156
           NQ&>N-Q;                                                             00000157
          :FOR:J&>0:STEP:1:UNTIL:NQ:DO:                                         00000158
          :BEGIN:                                                               00000159
                QJ&>Q<J;                                                        00000160
                #FIE/QJ/J&>0;                                                   00000161
                PROFCT&>0;                                                      00000162
               :FOR:K&>0:STEP:1:UNTIL:NQ:DO:                                    00000163
               :BEGIN:                                                          00000164
                     IIA&>N-K-J;                                                00000165
                     FACTA&>NCHEN-K,JJ;                                         00000166
                     FACTB&>1;                                                  00000167
                     PROFCT&>FACTA;                                             00000168
                SIGN&>1;                                                        00000169
```

```
171                     :FOR:L&>1:STEP:1:UNTIL:J:DO:                                    00000170
172                     :BEGIN:                                                         00000171
173                           SIGN&>-SIGN;                                              00000172
174                           IA&>J-L<1;                                                00000173
175                           IIA&>IIA<1;                                               00000174
176                           IB&>K-L<1;                                                00000175
177                          :IF:IIA>0:THEN:   FACTA&>1                                 00000176
178                          :ELSE::IF:IIA\0:THEN:   FACTA&>FACTA*IA/IIA;               00000177
179                           FACTB&>FACTB*IB;                                          00000178
180                           FACTB&>FACTB/L;                                           00000179
181                           PRDFCT&>FACTA*FACTB*SIGN<PRDFCT;                          00000180
182                     :END:;                                                          00000181
183                      WFIC/QJ/J&>WFIC/QJ/J<PRDFCT*FIC/K<Q/J;                         00000182
184               :END:;                                                              00000183
185          :END:;                                                                   00000184
186           NEWLINE(1);                                                             00000185
187           WRITETEXT(:E:FI COEFFICIENTS:);                                         00000186
188           NEWLINE(1);                                                             00000187
189           OUTPUTFIC/J/I,K,K,J,Q,N,INCR,0,8);                                      00000188
190           NEWLINE(1);                                                             00000189
191           WRITETEXT(:E:TRANSFORMED WFI COEFFICIENTS:);                            00000190
192           NEWLINE(1);                                                             00000191
193           OUTPUTCWFIC/J/I,K,K,J,Q,N,INCR,0,8);                                    00000192
194           NEWLINE(1);                                                             00000193
195      :END:  OF  TRNSFM ;                                                          00000194
196                                                                                   00000195
197                                                                                   00000196
198      :REAL::PROCEDURE:NCR(N,R);                                                   00000197
199                                                                                   00000198
200      :COMMENT:  THIS PROCEDURE DETERMINES THE VALUE OF THE FOLLOWING              00000199
201          FACTORIAL EXPRESSION &-                                                  00000200
202               FACT(N)/(FACT(R)*FACT(N-R))  ;                                      00000201
203      :VALUE:N,R;                                                                  00000202
204      :INTEGER:N,R;                                                                00000203
205      :BEGIN:                                                                      00000204
206          :INTEGER:M;                                                              00000205
207          :REAL:NUM,DENOM;                                                         00000206
208          :IF:N)0:OR:R)0:OR:R\N:THEN:NCR&>0                                        00000207
209          :ELSE::IF:N>R:OR:N>0:OR:R>0:THEN:   NCR&>1                               00000208
210          :ELSE:                                                                   00000209
211          :BEGIN:                                                                  00000210
212                NUM&>1;                                                           00000211
213                DENOM&>1;                                                         00000212
214               :FOR:M&>N-R<1:STEP:1:UNTIL:N:DO:                                   00000213
215                     NUM&>NUM*M;                                                  00000214
216               :FOR:M&>1:STEP:1:UNTIL:R:DO:                                       00000215
217                     DENOM&>DENOM*M;                                              00000216
218                NCR&>NUM/DENOM;                                                   00000217
219          :END:;                                                                   00000218
220      :END:  OF NCR;                                                               00000219
221                                                                                   00000220
222                                                                                   00000221
223                                                                                   00000222
224      :PROCEDURE:ROUTHCFI,Q,N,REINIT);                                             00000223
225                                                                                   00000224
226      :COMMENT:  THIS PROCEDURE TESTS FOR STABILITY IN THE CHARACTERISTIC          00000225
227          EQUATION ...                                                             00000226
```

```
228          FIC0]#L##N<FIC1]#L##(N-1]<....<FICN]>0                                    00000227
229          #HERE N > ORDER OF POLYNOMIAL                                             00000228
230               TEMPC/I/] > TEMPORARY STORAGE FOR THE ROW TO BEDETERMINED BY         00000229
231                         ROUTH'S METHOD :                                           00000230
232                                                                                    00000231
233     :VALUE:N,Q;                                                                    00000232
234     :INTEGER:N,Q;                                                                  00000233
235     :ARRAY:FI;                                                                     00000234
236     :LABEL:REINIT;                                                                 00000235
237     :BEGIN:                                                                        00000236
238          :INTEGER:J,I,HALFN,HLFNP1;                                                00000237
239          :ARRAY:TEMPC/0&N:/:2/3,RFIC/0&N<3/3;                                      00000238
240          :FOR:I&>0:STEP:1:UNTIL:N:DO:                                              00000239
241               RFIC/I/3&>FIC/I/3;                                                   00000240
242           RFIC/N<1/3&>0;                                                           00000241
243           RFIC/N<2/3&>0;                                                           00000242
244           RFIC/N<3/3&>0;                                                           00000243
245           HALFN&>N:/:2;                                                            00000244
246           HLFNP1&>HALFN<1;                                                         00000245
247          :FOR:I&>0:STEP:1:UNTIL:HALFN:DO:                                          00000246
248               TEMPC/I/3&>0;                                                       00000247
249           NEWLINEC5];                                                              00000248
250           #WRITETEXTC:C:ROUTH STABILITY TEST ... CHECK FOR CHANGE OF SIGN:]:];      00000249
251           NEWLINEC2];                                                              00000250
252           #WRITETEXTC:C:VALUE OF ROW 0:]:];                                        00000251
253           NEWLINEC2];                                                              00000252
254           :FOR:I&>0:STEP:1:UNTIL:HLFNP1:DO:                                        00000253
255               PRINTERFIC/2#I/],0,6];                                               00000254
256           NEWLINEC2];                                                              00000255
257          :FOR:J&>1:STEP:1:UNTIL:N:DO:                                              00000256
258          :BEGIN:                                                                   00000257
259               NEWLINEC1];                                                          00000258
260               #RITETEXTC:C:VALUE OF ROW  !:]:];                                    00000259
261               PRINTEJ<1,2,0];                                                      00000260
262               NEWLINEC2];                                                          00000261
263                                                                                    00000262
264               :COMMENT: THIS LOOP ESTABLISHES NEXT ROW ;                           00000263
265                                                                                    00000264
266               :FOR:I&>1:STEP:1:UNTIL:HLFNP1:DO:                                    00000265
267               :BEGIN:                                                              00000266
268                    TEMPC/I-1/3&>CRFIC/1/3#RFIC/2#I/3-RFIC/0/3#RFIC/2#I<1/3]/00000267
269                                                                  RFIC/1/3];        00000268
270                    :IF:TEMPC/0/3/RFIC/0/3)0:THEN:                                  00000269
271                    :BEGIN:                                                         00000270
272                         #RITETEXTC:C:UNSTABLE ...BY ROUTH STABILITY CRITERIO00000271
273     N       FILTER NO. >:]:];                                                      00000272
274                         :GOTO:REINIT;                                              00000273
275                    :END:;                                                          00000274
276                     PRINTCTEMPC/I-1/3,0,6];                                        00000275
277               :END:;                                                           00000276
278                                                                                    00000277
279               :COMMENT: THIS LOOP SHIFTS ROWS UP BY DELETING TOP ROW;          00000278
280                                                                                    00000279
281               :FOR:I&>0:STEP:1:UNTIL:HALFN:DO:                                     00000280
282               :BEGIN:                                                              00000281
283                    RFIC/2#I/3&>RFIC/2#I<1/3;                                       00000282
284                    RFIC/2#I<1/3&>TEMPC/I/3;                                        00000283
```

```
285                 :END::                                                  00000284
286             :END:;                                                      00000285
287         :END:  OF ROUTH :                                               00000286
288                                                                         00000287
289                                                                         00000288
290                                                                         00000289
291         :PROCEDURE:VECROWEROWVEC,RDEL,NOFILT,FLTORD,COLUMN];            00000290
292                                                                         00000291
293         :COMMENT:  THIS PROCEDURE LOADS THE CURRENT COLUMN OF THE CORRELATION  00000292
294             MATRIX , REFERRED TO IN THE THEORY BY THE GREEK LETTER ROW , INTO  00000293
295             THE VECTOR ROWVEC , READY FOR MANIPULATION BY CROUT2  ;     00000294
296                                                                         00000295
297         :VALUE:NOFILT,FLTORD,COLUMN:                                    00000296
298         :INTEGER:NOFILT,FLTORD,COLUMN:                                  00000297
299         :ARRAY:RDEL,ROWVEC:                                             00000298
300         :BEGIN:                                                         00000299
301             :INTEGER:I,J,K:                                             00000300
302              K&>0:                                                      00000301
303             :FOR:J&>I:STEP:1:UNTIL:NOFILT:DO:                           00000302
304             :FOR:I&>1:STEP:1:UNTIL:FLTORD:DO:                           00000303
305             :BEGIN:                                                     00000304
306                   K&>K<1:                                               00000305
307                   ROWVEC[K/3]&>RDEL[J,COLUMN,I/3:                       00000306
308             :END:;                                                      00000307
309         :END:  OF VECROW :                                              00000308
310                                                                         00000309
311                                                                         00000310
312                                                                         00000311
313         :PROCEDURE:RDCOPERDEL,NOFILT,FLTOPD];                           00000312
314                                                                         00000313
315         :COMMENT:  THIS PROCEDURE READS IN AUTO + CROSS CORRELATION VECTORS IN  00000314
316             THE FOLLOWING MANNER ...                                    00000315
317             RDEL[1,1,I/3   RDEL[1,2,I/3   RDEL[1,3,I/3  ...RDEL[1,NOFILT,I/3]00000316
318                            RDEL[2,2,I/3   RDEL[2,3,I/3  ...RDEL[2,NOFILT,I/3]00000317
319                                                          @              00000318
320                                                          @              00000319
321                                                  RDEL[NOFILT,NOFILT,I/3]00000320
322             AS THE CROSS CORRELATIONS ARE ANTI SYMMETRIC                00000321
323             RDEL[J,K,I/3 &>-RDEL[K,J,I/3 ...WHERE K^>J                  00000322
324                                                 I > CORRELATION LAG VALUE ;  00000323
325                                                                         00000324
326         :VALUE:NOFILT FLTORD:                                           00000325
327         :INTEGER:NOFILT,FLTORD:                                         00000326
328         :ARRAY:PDEL:                                                    00000327
329         :BEGIN:                                                         00000328
330             :INTEGER:I,J,K:                                             00000329
331             :FOR:K&>1:STEP:1:UNTIL:NOFILT:DO:                           00000330
332             :BEGIN:                                                     00000331
333                   :FOR:I&>0:STEP:1:UNTIL:FLTORD:DO:                     00000332
334                         RDEL[K,K,I/3&>READ:                             00000333
335                   :FOR:J&>K<1:STEP:1:UNTIL:NOFILT:DO:                   00000334
336                   :FOR:I&>1:STEP:1:UNTIL:FLTORD:DO:                     00000335
337                   :BEGIN:                                               00000336
338                         RDEL[K,J,I/3&>READ:                             00000337
339                         RDEL[J,K,I/3&>-RDEL[K,J,I/3:                    00000338
340                   :END::                                                00000339
341             :END:;                                                      00000340
```

```
342      :END:  OF RDCOR ;                                                    00000341
343                                                                           00000342
344                                                                           00000343
345                                                                           00000344
346      :PROCEDURE:KEEPFI(DUMMY,FI,NOFILT,COLUMN,FLTORD);                    00000345
347                                                                           00000346
348      :COMMENT:  THIS PROCEDURE STORES THE FI COEFFICIENTS OF THE CURRENT  00000347
349          FILTER , COLUMN;                                                 00000348
350                                                                           00000349
351      :VALUE:COLUMN,NOFILT,FLTORD;                                         00000350
352      :INTEGER:COLUMN,NOFILT,FLTORD;                                       00000351
353      :ARRAY:DUMMY,FI;                                                     00000352
354      :BEGIN:                                                              00000353
355          :INTEGER:J,I;                                                    00000354
356          :FOR:J:=1:STEP:1:UNTIL:NOFILT:DO:                                00000355
357          :FOR:I:=1:STEP:1:UNTIL:FLTORD:DO:                                00000356
358              FI(J,COLUMN,I):=DUMMY(J-1)*FLTORD+I;                         00000357
359      :END:  OF KEEPFI ;                                                   00000358
360                                                                           00000359
361                                                                           00000360
362                                                                           00000361
363      :PROCEDURE:LOADMX(MLTMTX,RDEL,FLTORD,TOTORD);                        00000362
364                                                                           00000363
365      :COMMENT:  THIS PROCEDURE LOADS THE LARGE MLTMTX MATRIX FROM THE     00000364
366          CORRELATION RDEL(I,J,K) , WHERE INDEX VARIABLES I + J REFER TO   00000365
367          CROSS CORRELATION + K REFERS TO THE LAG.                         00000366
368          MLTMTX IS THE MATRIX ...                                         00000367
369        ****                                                    ****       00000368
370        *  R(1,1)  R(1,1)  R(1,2)  ...   R(1,2)  R(2,1)  R(2,2)  ... *    00000369
371        *  R(1,1)  R(1,1)  R(1,1)  ...   R(1,2)  R(1,2)  R(2,1)  ... *    00000370
372        *  R(1,2)  .....                 R(1,2)  .....               *    00000371
373        *    .                             .                         *    00000372
374        *    .                             .                         *    00000373
375        *    .                             .                         *    00000374
376        *    .                             .                         *    00000375
377        *    .                             .                         *    00000376
378        *  R(2,1)  R(1,2)  R(1,2)  .....                             *    00000377
379        *  R(2,1)  R(2,1)  R(1,2)  .....                             *    00000378
380        *    .                             .                         *    00000379
381        *    .                             .                         *    00000380
382        *    .                             .                         *    00000381
383        ****                                                    ****;      00000382
384                                                                           00000383
385      :VALUE:FLTORD,TOTORD;                                                00000384
386      :INTEGER:FLTORD,TOTORD;                                              00000385
387      :ARRAY:MLTMTX,RDEL;                                                  00000386
388      :BEGIN:                                                              00000387
389          :INTEGER:I,J,K,L,X,Y;                                            00000388
390          :FOR:L:=1:STEP:FLTORD:UNTIL:TOTORD:DO:                           00000389
391          :FOR:K:=1:STEP:FLTORD:UNTIL:TOTORD:DO:                           00000390
392          :FOR:J:=0:STEP:1:UNTIL:FLTORD-1:DO:                              00000391
393          :FOR:I:=J:STEP:1:UNTIL:FLTORD-1:DO:                              00000392
394          :BEGIN:                                                          00000393
395                X:=I-K;                                                    00000394
396                X:=J-L;                                                    00000395
397                MLTMTX(X,Y/J):=MLTMTX(Y,X/J):=RDEL(K:/:FLTORD<1,L:/:FLTORD<1,00000396
398                                                                 I-J/J); 00000397
```

```
399         ;END;;                                                          00000398
400       ;END;  OF LOADMX ;                                                00000399
401                                                                         00000400
402                                                                         00000401
403                                                                         00000402
404       ;PROCEDURE;OUTPUT(MEMBER,DIM3,DIM2,DIM1,INIT,FIN,INCR,M,N);       00000403
405                                                                         00000404
406       ;COMMENT;  THIS IS MERELY AN OUTPUT PROCEDURE .                   00000405
407                 MEMBER > VALUE TO BE PRINTED                            00000406
408                 INIT > INITIAL INDEX VALUE                              00000407
409                 FIN  > FINAL INDEX VALUE                                00000408
410                 INCR > NUMBER OF VALUES PER LINE  ;                     00000409
411                                                                         00000410
412       ;VALUE;INIT,FIN,M,N,DIM3,DIM2;                                    00000411
413       ;INTEGER;INIT,FIN,INCR,M,N,DIM3,DIM2,DIM1;                        00000412
414       ;REAL;MEMBER;                                                     00000413
415       ;BEGIN;                                                           00000414
416           ;INTEGER;COUNT1,COUNT2;                                       00000415
417            COUNT1&>0;                                                   00000416
418            COUNT2&>INIT;                                                00000417
419            PRINT(COUNT2,3,0);                                           00000418
420           ;FOR;DIM1&>INIT;STEP;1;UNTIL;FIN;DO;                          00000419
421           ;BEGIN;                                                       00000420
422                ;IF;DIM1>INIT;THEN;                                      00000421
423                 ;BEGIN;                                                 00000422
424                     ;IF;M>0;THEN;    INCR&>N<9                          00000423
425                     ;ELSE;;IF;N>0;THEN;    INCR&>M<3                   00000424
426                     ;ELSE;;IF;M^>0 + N^>0;THEN;    INCR&>M<N<9;        00000425
427                      INCR&>114;/;INCR;                                  00000426
428                ;END;;                                                   00000427
429                 COUNT1&>COUNT1<1;                                       00000428
430                 PRINT(MEMBER,M,N);                                      00000429
431                ;IF;COUNT1>INCR + COUNT1)FIN;THEN;                       00000430
432                ;BEGIN;                                                  00000431
433                      NEWLINE(1);                                        00000432
434                      COUNT2&>COUNT2<INCR;                               00000433
435                      PRINT(COUNT2,3,0);                                 00000434
436                      COUNT1&>0;                                         00000435
437                ;END;;                                                   00000436
438           ;END;;                                                        00000437
439            NEWLINE(1);                                                  00000438
440       ;END;  OF OUTPUT ;                                                00000439
441                                                                         00000440
442                                                                         00000441
443                                                                         00000442
444                                                                         00000443
445       ;COMMENT;   START OF PROGRAM                                      00000444
446                   ***************;                                     00000445
447                                                                         00000446
448        POSDEF&>;TRUE;;                                                  00000447
449        FLTORD&>READ;                                                    00000448
450        NOFILT&>READ;                                                    00000449
451        TOTORD&>ORIG&>FLTORD=NOFILT;                                     00000450
452       ;BEGIN;                                                           00000451
453           ;ARRAY;ROW(1&NOFILT,1&NOFILT,2&FLTORD/3,ROWVEC(1&TOTORD/3,    00000452
454                  MLTMTX(1&TOTORD,1&TOTORD/3,DUMMY,WDUMMY(0&TOTORD/3,    00000453
455                  FIC/1&NOFILT,1&NOFILT,1&FLTORD/3;                      00000454
```

```
456           SELINE[3];                                                    00000455
457           PDCORRDEL,NOFILT,FLTORD];                                     00000456
458           FREEIN;                                                       00000457
459          :GOTO:STRT;                                                    00000458
460      REINIT&                                                            00000459
461          :IF:FLTORD)>1:THEN:                                            00000460
462          :BEGIN:                                                        00000461
463                WRITETEXT[C:FAILURE TO FIND FILTER NO. :3:];             00000462
464                PRINTCOLUMN,3,0];                                        00000463
465                NEWLINE[2];                                              00000464
466               :GOTO:OUT;                                                00000465
467          :END:;                                                         00000466
468           FLTORD&>FLTORD-1;                                             00000467
469      STRT&                                                              00000468
470           TOTORD&>FLTORD*NOFILT;                                        00000469
471           LOADMXEMLTMTX,RDEL,FLTORD,TOTORD];                            00000470
472           COLUMN&>1;                                                    00000471
473          :IF:^POSDEF:THEN::GOTO:NXTCOL;                                 00000472
474          :FOR:Z&>1:STEP:FLTORD:UNTIL:TOTORD:DO:                         00000473
475          :BEGIN:                                                        00000474
476           PRINTZ,3,0];                                                  00000475
477           NEWLINE[1];                                                   00000476
478           PUT&>:FALSE::                                                 00000477
479          :FOR:ZZ&>Z:STEP:1:UNTIL:FLTORD<Z-1:DO:                         00000478
480          :BEGIN:                                                        00000479
481                CROUTZEMLTMTX,ROWVEC,Z,ZZ,DUMMY,DET,POSDEF,OUT,PUT];     00000480
482               :IF:DET)0:THEN:                                           00000481
483               :BEGIN:                                                   00000482
484                     WRITETEXT[C:LARGE CORRELATION MATRIX IS NOT POSITIVE 00000483
485      DEFINITE:3:];                                                      00000484
486                     NEWLINE[1];                                         00000485
487                     WRITETEXT[C:  FLTORD  COLUMN:3:];                   00000486
488                     NEWLINE[1];                                         00000487
489                     PRINTEFLTORD,6,0];                                  00000488
490                     PRINTECOLUMN,6,0];                                  00000489
491                     NEWLINE[1];                                         00000490
492                     FLTORD&>ZZ-Z<1:                                     00000491
493                    :GOTO:REINIT;                                        00000492
494               :END::                                                    00000493
495          :END:;                                                         00000494
496          :END:;                                                         00000495
497          POSDEF&>:FALSE::                                               00000496
498      NXTCOL&                                                            00000497
499           VECROWEROWVEC,RDEL,NOFILT,FLTORD,COLUMN];                     00000498
500           NEWLINE[2];                                                   00000499
501          :IF:TOTORD^>ORIG:THEN:   PUT&>:FALSE:                          00000500
502          :ELSE:PUT&> :TRUE::                                            00000501
503           CROUTZEMLTMTX,ROWVEC,1,TOTORD,DUMMY,DET,POSDEF,OUT,PUT];      00000502
504           KEEPFIEDUMMY,FI,NOFILT,COLUMN,FLTORD];                        00000503
505          :IF:FLTORD>TOTORD:THEN:                                        00000504
506          :BEGIN:                                                        00000505
507                DUMMYE/0/]&>-RDELE/1,1,0/];                              00000506
508                TRNSFMEDUMMY,#DUMMY,0,FLTORD];                           00000507
509               :FOR:ZZ&>0:STEP:1:UNTIL:FLTORD:DO:                        00000508
510               :IF:#DUMMYE/ZZ/]/#DUMMYE/1/]:0:THEN:                      00000509
511               :BEGIN:                                                   00000510
512                     WRITETEXT[C:INSTABILITY ENCOUNTERED IN COEFFICIENT FI:]:00000511
```

```
513       ];                                                                    00000512
514                 PRINTE22,3,03;                                              00000513
515                 NEWLINE[2];                                                 00000514
516                 !GOTO!REINIT;                                               00000515
517             !END!;                                                          00000516
518              ROUTHEWDUMMY,0,FLTORD,REINIT];                                 00000517
519         !END!;                                                              00000518
520             !IF!COLUMN!NOFILT!THEN!                                         00000519
521             !BEGIN!                                                         00000520
522                   COLUMN&>COLUMN<1;                                         00000521
523                 !GOTO!NXTCOL;                                               00000522
524             !END!;                                                          00000523
525              WRITETEXTE:E:FILTER ORDER  > :3:];                             00000524
526              PRINTEFLTORD,3,03;                                             00000525
527              NEWLINE[2];                                                    00000526
528             !FOR!KP&>1!STEP!1!UNTIL!NOFILT!DO!                              00000527
529             !BEGIN!                                                         00000528
530                  WRITETEXTE:E:FILTER COEFFICIENT > :3:];                    00000529
531                  PRINTEKP,3,03;                                             00000530
532                  NEWLINE[2];                                                00000531
533                 !FOR!JP&>1!STEP!1!UNTIL!NOFILT!DO!                          00000532
534                 !BEGIN!                                                     00000533
535                       PRINTEJP,3,03;                                        00000534
536                       PRINTEKP,3,03;                                        00000535
537                       NEWLINE[1];                                           00000536
538                       WRITETEXTE:E:  ***  ***:3:];                          00000537
539                       NEWLINE[2];                                           00000538
540                       OUTPUTEFIE/JP,KP,IP/3,JP,KP,IP,1,FLTORD,INCRP,0,12];00000539
541                       NEWLINE[2];                                           00000540
542                 !END!;                                                      00000541
543             !END!;                                                          00000542
544              SELOUTE1];                                                     00000543
545              !FOR!KP&>1!STEP!1!UNTIL!NOFILT!DO!                             00000544
546              !FOR!JP&>1!STEP!1!UNTIL!NOFILT!DO!                             00000545
547              !FOR!IP&>1!STEP!1!UNTIL!FLTORD!DO!                             00000546
548                   PRINTEFIE/JP,KP,IP/3,0,12];                               00000547
549              FREEOUT;                                                       00000548
550       !END!;                                                                00000549
551       OUT&                                                                  00000550
552       !END!                                                                 00000551

$RT UNIFIL.TALT
```

OUTPUT FILTER COEFFICIENTS TO FILE (FOR SUBSEQUENT PROCESSING)

ARFILT - Program Input

FLTORD (Integer)
Defines the order of the autoregressive (AR) filter coefficients.

NOFILT (Integer)
Defines the number of AR filters. For the monovariate case this should be assigned the value "1".

RDEL [1:NOFILT,1;NOFILT,0:FLTORD] (Array)
Defines the auto and cross correlation vectors from lag increment 0 to FLTORD.
These vectors are input in the following manner.

RDEL[1,1,I] RDEL[1,2,I] RDEL[1,3,I] ....[1,NOFILT,I]

RDEL[2,2,I] RDEL[2,3,I] ....[2,NOFILT,I]

RDEL[NOFILT,NOFILT,I]

And this is repeated for I = 0,1,...,FLTORD. (The program will utilise the antisymmetry properties to fully define remaining auto and cross correlations.)

In the monovariate case, RDEL effectively reduces to a one dimensional array.

OUTPUT FROM ARFILT — SAMPLE LISTING

```
1
VALUE OF DETERMINANT IS =  1.000000000000&  0
VALUE OF DETERMINANT IS =  2.264408609646&- 1
VALUE OF DETERMINANT IS =  5.127545898556&- 2
VALUE OF DETERMINANT IS =  1.161085856629&- 2
VALUE OF DETERMINANT IS =  2.629172694183&- 3
VALUE OF DETERMINANT IS =  5.953521022028&- 4
VALUE OF DETERMINANT IS =  1.348120366463&- 4
VALUE OF DETERMINANT IS =  3.052695229847&- 5
VALUE OF DETERMINANT IS =  6.912549055819&- 6
VALUE OF DETERMINANT IS =  1.565283490534&- 6
VALUE OF DETERMINANT IS =  3.544441255973&- 7
VALUE OF DETERMINANT IS =  8.026562941965&- 8
VALUE OF DETERMINANT IS =  1.817428522474&- 8
VALUE OF DETERMINANT IS =  4.115400611962&- 9
VALUE OF DETERMINANT IS =  9.318948166328&-10
VALUE OF DETERMINANT IS =  2.110198552876&-10
VALUE OF DETERMINANT IS =  4.778333449111&-11
VALUE OF DETERMINANT IS =  1.082009891458&-11
VALUE OF DETERMINANT IS =  2.450112485738&-12
VALUE OF DETERMINANT IS =  5.548055381142&-13
```

CHECK FOR POSITIVE DEFINITENESS — EACH SUCCESSIVE ORDER SUB DETERMINANT IS CHECKED AND THE VALUE OUTPUT IN TURN.

B VECTOR — IE. AUTOCORRELATION VECTOR OF YULE WALKER RELATION

```
 1   8.795221148&- 1   7.735591498&- 1   6.803623788&- 1   5.983937578&- 1   5.263005428&- 1   4.628929658&- 1
 7   4.071246008&- 1   3.580750888&- 1   3.149345598&- 1   2.769922618&- 1   2.436208198&- 1   2.142698988&- 1
13   1.884551138&- 1   1.657504408&- 1   1.457811778&- 1   1.282177698&- 1   1.127703638&- 1   9.918402838&- 2
19   8.723454628&- 2   7.672471258&- 2
```

A MATRIX — IE. AUTOCORRELATION MATRIX OF YULE WALKER RELATION

LAG INCREMENT OF 1ST VARIABLE PER LINE

```
(1)
 1   1.000000008&  0   8.795221148&- 1   7.735591498&- 1   6.803623788&- 1   5.983937578&- 1   5.263005428&- 1
 7   4.628929658&- 1   4.071246008&- 1   3.580750888&- 1   3.149345598&- 1   2.769922618&- 1   2.436208198&- 1
13   2.142698988&- 1   1.884551138&- 1   1.657504408&- 1   1.457811778&- 1   1.282177698&- 1   1.127703638&- 1
19   9.918402838&- 2   8.723454628&- 2

(2)   ROW NUMBER
 1   8.795221148&- 1   1.000000008&  0   8.795221148&- 1   7.735591498&- 1   6.803623788&- 1   5.983937578&- 1
 7   5.263005428&- 1   4.628929658&- 1   4.071246008&- 1   3.580750888&- 1   3.149345598&- 1   2.769922618&- 1
13   2.436208198&- 1   2.142698988&- 1   1.884551138&- 1   1.657504408&- 1   1.457811778&- 1   1.282177698&- 1
19   1.127703638&- 1   9.918402838&- 2

(3)
 1   7.735591498&- 1   8.795221148&- 1   1.000000008&  0   8.795221148&- 1   7.735591498&- 1   6.803623788&- 1
 7   5.983937578&- 1   5.263005428&- 1   4.628929658&- 1   4.071246008&- 1   3.580750888&- 1   3.149345598&- 1
13   2.769922618&- 1   2.436208198&- 1   2.142698988&- 1   1.884551138&- 1   1.657504408&- 1   1.457811778&- 1
19   1.282177698&- 1   1.127703638&- 1

(4)
 1   6.803623788&- 1   7.735591498&- 1   8.795221148&- 1   1.000000008&  0   8.795221148&- 1   7.735591498&- 1
 7   6.803623788&- 1   5.983937578&- 1   5.263005428&- 1   4.628929658&- 1   4.071246008&- 1   3.580750888&- 1
13   3.149345598&- 1   2.769922618&- 1   2.436208198&- 1   2.142698988&- 1   1.884551138&- 1   1.657504408&- 1
19   1.457811778&- 1   1.282177698&- 1
 5
 1   5.983937578&- 1   6.803623788&- 1   7.735591498&- 1   8.795221148&- 1   1.000000008&  0   8.795221148&- 1
 7   7.735591498&- 1   6.803623788&- 1   5.983937578&- 1   5.263005428&- 1   4.628929658&- 1   4.071246008&- 1
13   3.580750888&- 1   3.149345598&- 1   2.769922618&- 1   2.436208198&- 1   2.142698988&- 1   1.884551138&- 1
19   1.657504408&- 1   1.457811778&- 1
 6
 1   5.263005428&- 1   5.983937578&- 1   6.803623788&- 1   7.735591498&- 1   8.795221148&- 1   1.000000008&  0
 7   8.795221148&- 1   7.735591498&- 1   6.803623788&- 1   5.983937578&- 1   5.263005428&- 1   4.628929658&- 1
13   4.071246008&- 1   3.580750888&- 1   3.149345598&- 1   2.769922618&- 1   2.436208198&- 1   2.142698988&- 1
19   1.884551138&- 1   1.657504408&- 1
 7
```

7	1.000000000 0	8.795221148- 1	7.735591498- 1	6.803623788- 1	5.983937578- 1	5.263005428- 1
13	4.628929658- 1	4.071246008- 1	3.580750888- 1	3.149349598- 1	2.769922618- 1	2.436208198- 1
19	2.142698988- 1	1.884551138- 1				
8						
1	4.071246008- 1	4.628929658- 1	5.263005428- 1	5.983937578- 1	6.803623788- 1	7.735591498- 1
7	8.795221148- 1	1.000000000 0	8.795221148- 1	7.735591498- 1	6.803623788- 1	5.983937578- 1
13	5.263005428- 1	4.628929658- 1	4.071246008- 1	3.580750888- 1	3.149349598- 1	2.769922618- 1
19	2.436208198- 1	2.142698988- 1				
9						
1	3.580750888- 1	4.071246008- 1	4.628929658- 1	5.263005428- 1	5.983937578- 1	6.803623788- 1
7	7.735591498- 1	8.795221148- 1	1.000000000 0	8.795221148- 1	7.735591498- 1	6.803623788- 1
13	5.983937578- 1	5.263005428- 1	4.628929658- 1	4.071246008- 1	3.580750888- 1	3.149349598- 1
19	2.769922618- 1	2.436208198- 1				
10						
1	3.149349598- 1	3.580750888- 1	4.071246008- 1	4.628929658- 1	5.263005428- 1	5.983937578- 1
7	6.803623788- 1	7.735591498- 1	8.795221148- 1	1.000000000 0	8.795221148- 1	7.735591498- 1
13	6.803623788- 1	5.983937578- 1	5.263005428- 1	4.628929658- 1	4.071246008- 1	3.580750888- 1
19	3.149349598- 1	2.769922618- 1				
11						
1	2.769922618- 1	3.149349598- 1	3.580750888- 1	4.071246008- 1	4.628929658- 1	5.263005428- 1
7	5.983937578- 1	6.803623788- 1	7.735591498- 1	8.795221148- 1	1.000000000 0	8.795221148- 1
13	7.735591498- 1	6.803623788- 1	5.983937578- 1	5.263005428- 1	4.628929658- 1	4.071246008- 1
19	3.580750888- 1	3.149349598- 1				
12						
1	2.436208198- 1	2.769922618- 1	3.149349598- 1	3.580750888- 1	4.071246008- 1	4.628929658- 1
7	5.263005428- 1	5.983937578- 1	6.803623788- 1	7.735591498- 1	8.795221148- 1	1.000000000 0
13	8.795221148- 1	7.735591498- 1	6.803623788- 1	5.983937578- 1	5.263005428- 1	4.628929658- 1
19	4.071246008- 1	3.580750888- 1				
13						
1	2.142698988- 1	2.436208198- 1	2.769922618- 1	3.149349598- 1	3.580750888- 1	4.071246008- 1
7	4.628929658- 1	5.263005428- 1	5.983937578- 1	6.803623788- 1	7.735591498- 1	8.795221148- 1
13	1.000000000 0	8.795221148- 1	7.735591498- 1	6.803623788- 1	5.983937578- 1	5.263005428- 1
19	4.628929658- 1	4.071246008- 1				
14						
1	1.684551138- 1	2.142698988- 1	2.436208198- 1	2.769922618- 1	3.149349598- 1	3.580750888- 1
7	4.071246008- 1	4.628929658- 1	5.263005428- 1	5.983937578- 1	6.803623788- 1	7.735591498- 1
13	8.795221148- 1	1.000000000 0	8.795221148- 1	7.735591498- 1	6.803623788- 1	5.983937578- 1
19	5.263005428- 1	4.628929658- 1				
15						
1	1.657504408- 1	1.884551138- 1	2.142698988- 1	2.436208198- 1	2.769922618- 1	3.149349598- 1
7	3.580750888- 1	4.071246008- 1	4.628929658- 1	5.263005428- 1	5.983937578- 1	6.803623788- 1
13	7.735591498- 1	8.795221148- 1	1.000000000 0	8.795221148- 1	7.735591498- 1	6.803623788- 1
19	5.983937578- 1	5.263005428- 1				
16						
1	1.457811778- 1	1.657504408- 1	1.884551138- 1	2.142698988- 1	2.436208198- 1	2.769922618- 1
7	3.149349598- 1	3.580750888- 1	4.071246008- 1	4.628929658- 1	5.263005428- 1	5.983937578- 1
13	6.803623788- 1	7.735591498- 1	8.795221148- 1	1.000000000 0	8.795221148- 1	7.735591498- 1
19	6.803623788- 1	5.983937578- 1				
17						
1	1.282177698- 1	1.457811778- 1	1.657504408- 1	1.884551138- 1	2.142698988- 1	2.436208198- 1
7	2.769922618- 1	3.149349598- 1	3.580750888- 1	4.071246008- 1	4.628929658- 1	5.263005428- 1
13	5.983937578- 1	6.803623788- 1	7.735591498- 1	8.795221148- 1	1.000000000 0	8.795221148- 1
19	7.735591498- 1	6.803623788- 1				
18						
1	1.127703638- 1	1.282177698- 1	1.457811778- 1	1.657504408- 1	1.884551138- 1	2.142698988- 1
7	2.436208198- 1	2.769922618- 1	3.149349598- 1	3.580750888- 1	4.071246008- 1	4.628929658- 1
13	5.263005428- 1	5.983937578- 1	6.803623788- 1	7.735591498- 1	8.795221148- 1	1.000000000 0
19	8.795221148- 1	7.735591498- 1				
19						
1	9.918402838- 2	1.127703638- 1	1.282177698- 1	1.457811778- 1	1.657504408- 1	1.884551138- 1
7	2.142698988- 1	2.436208198- 1	2.769922618- 1	3.149349598- 1	3.580750888- 1	4.071246008- 1
13	4.628929658- 1	5.263005428- 1	5.983937578- 1	6.803623788- 1	7.735591498- 1	8.795221148- 1
19	1.000000000 0	8.795221148- 1				
20						
1	8.723454628- 2	9.918402838- 2	1.127703638- 1	1.282177698- 1	1.457811778- 1	1.657504408- 1

13	4.071246608&- 1	4.628929658&- 1	5.263005428&- 1	5.983937578&- 1	6.803623788&- 1	7.735591498&- 1
19	8.795221148&- 1	1.000000008& 0				

VALUE OF DETERMINANT IS = 5.548055381142&-13

Y VECTOR

1	8.795221148&- 1	-1.516171278&-13	-2.046878268&-13	6.732182328&-13	-8.518483208&-13	3.233026508&-13
7	4.555552498&-13	-4.933101558&-13	-2.882495498&-13	9.452874968&-13	-8.287266098&-13	2.577846228&-13
13	2.651498408&-14	-3.148737548&-14	1.361963048&-13	-2.268265608&-13	1.554124148&-14	2.270860678&-13
19	-1.555425218&-13	2.647568738&-14				

AR FILTER COEFFICIENTS DETERMINED FROM YULE WALKER RELATIONS

FI COEFFICIENTS

0	-1.000000008& 0	8.795221148&- 1	-1.516171278&-13	-2.046878268&-13	6.732182328&-13	-8.518483208&-13
6	3.233026508&-13	4.555552498&-13	-4.933101558&-13	-2.882495498&-13	9.452874968&-13	-8.287266098&-13
12	2.577846228&-13	2.651498408&-14	-3.148737548&-14	1.361963048&-13	-2.268265608&-13	1.554124148&-14
18	2.270860678&-13	-1.555425218&-13	2.647568738&-14			

AR FILTER COEFFICIENTS FROM LAG 0 — READY FOR BILINEAR TRANSFORMATION

TRANSFORMED MFI COEFFICIENTS

0	-1.204778868&- 1	-4.168601958& 0	-5.631263878& 1	-4.381413538& 2	-2.288229218& 3	-8.685944578& 3
6	-2.512388918& 4	-5.706583378& 4	-1.038113208& 5	-1.531875478& 5	-1.847560008& 5	-1.827324538& 5
12	-1.481286808& 5	-9.797416638& 4	-5.239611098& 4	-2.232265548& 4	-7.401778798& 3	-1.841858658& 3
18	-3.236873618& 2	-3.583139818& 1	-1.879522118& 0			

TRANSFORMATION $Z = \frac{W+1}{W-1}$ APPLIED TO AR FILTER COEFFICIENTS

ROUTH STABILITY TEST ... CHECK FOR CHANGE OF SIGN

APPLICATION OF ROUTH STABILITY TEST

VALUE OF ROW 0

-1.204779&- 1 -5.631264& 1 -2.288229& 3 -2.512389& 4 -1.038113& 5 -1.847560& 5 -1.481287& 5 -5.239611& 4
-7.401771& 3 -3.236874& 2 -1.879522& 0 0.000000& 0

VALUE OF ROW 1 2

-4.364980& 1 -2.037194& 3 -2.347461& 4 -9.938401& 4 -1.794748& 5 -1.452971& 5 -5.175098& 4 -7.348539& 3
-3.226518& 2 -1.879522& 0 0.000000& 0

VALUE OF ROW 1 3

-2.435871& 2 -6.444094& 3 -4.757456& 4 -1.368475& 5 -1.688564& 5 -9.303189& 4 -2.152626& 4 -1.811345& 3
-3.565190& 1 0.000000& 0 0.000000& 0

VALUE OF ROW 1 4

-8.824396& 2 -1.494945& 4 -7.500466& 4 -1.492164& 5 -1.286262& 5 -4.787671& 4 -7.024007& 3 -3.162631& 2
-1.879522& 0 0.000000& 0 0.000000& 0

VALUE OF ROW 1 5

-2.317473& 3 -2.687034& 4 -9.485306& 4 -1.333507& 5 -7.981609& 4 -1.968137& 4 -1.723744& 3 -3.513308& 1
0.000000& 0 0.000000& 0 0.000000& 0

VALUE OF ROW 1 6

-4.717852& 3 -3.888512& 4 -9.943960& 4 -9.823407& 4 -4.038251& 4 -6.367645& 3 -3.028853& 2 -1.879522& 0
0.000000& 0 0.000000& 0 0.000000& 0

VALUE OF ROW 1 7

-7.769446& 3 -4.650320& 4 -8.509677& 4 -5.997965& 4 -1.655349& 4 -1.574963& 3 -3.420863& 1 -1.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0

VALUE OF ROW 1 8

-1.064691& 4 -4.676617& 4 -6.181254& 4 -3.033071& 4 -5.411278& 3 -2.821120& 2 -1.879522& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0

VALUE OF ROW 1 9

-1.237620& 4 -3.998988& 4 -3.784622& 4 -1.262468& 4 -1.369095& 3 -3.283828& 1 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 10

-1.236394& 4 -2.925445& 4 -1.948723& 4 -4.233482& 3 -2.538621& 2 -1.879522& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 11

-1.234642& 4 -1.833966& 4 -8.367305& 3 -1.114982& 3 -3.885681& 1 0.000000& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 12

-8.875532& 3 -9.824890& 3 -2.945884& 3 -2.181126& 2 -1.879522& 0 0.000000& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 13

-5.313963& 3 -4.461394& 3 -8.258112& 2 -2.946505& 1 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 14

-3.044992& 3 -1.690914& 3 -1.748548& 2 -1.879522& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 15

-1.510497& 3 -5.206636& 2 -2.518500& 1 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 16

-6.413151& 2 -1.240846& 2 -1.879522& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 17

-2.284056& 2 -2.075814& 1 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 18

-6.580012& 1 -1.879522& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 19

-1.423394& 1 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 20

-1.879522& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0
VALUE OF ROW 1 21

0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0 0.000000& 0
0.000000& 0 0.000000& 0 0.000000& 0 FILTER ORDER = 20

FILTER COEFFICIENT = 1

FINAL STABLE AR FILTER COEFFICIENTS.

LAG	1 ***	1 ***			
1	8.795221143117&-1	-1.516171272954&-13	-2.046878284725&-13	6.732182319069&-13	-8.518483200247&-13
6	3.233426495178&-13	4.555552494525&-13	-4.933101552374&-13	-2.882495487510&-13	9.452874959271&-13
11	-8.287265893448&-13	2.577846222631&-13	2.651498300411&-14	-3.148737538851&-14	1.361963035968&-13
16	-2.268265977378&-13	1.554124139886&-14	2.270061074249&-13	-1.555425212726&-13	2.647580729112&-14
21					

Program DSYS2

PURPOSE:

This program is used to simulate response realisations of a vehicle traversing a road profile with variable velocity.

The program firstly generates discretised random road data at regular spatial lag increments in accordance with prescribed correlation characteristics and a Gaussian probability distribution. Both displacement and gradient random data points are produced at every realised point on the record. These values are then fed into a dynamic vehicle simulator which models in the space domain the behavioural response of the vehicle. Response displacement, together with the first and second spatial derivatives are output. The entire process of road generation and response simulation is repeated on a point by point basis throughout the entire simulated realisation.

The overall flow logistics of the program are shown schematically on Fig 9.1, while the breakdown of the segments concerned with road profile generation and dynamic vehicle integration are shown in Fig. 3.1 and Fig. 9.2 respectively.

For a more detailed description of the mode of operation of the program refer to Ph.D. thesis Chapter IX Section 8 and Chapter III Section 2.

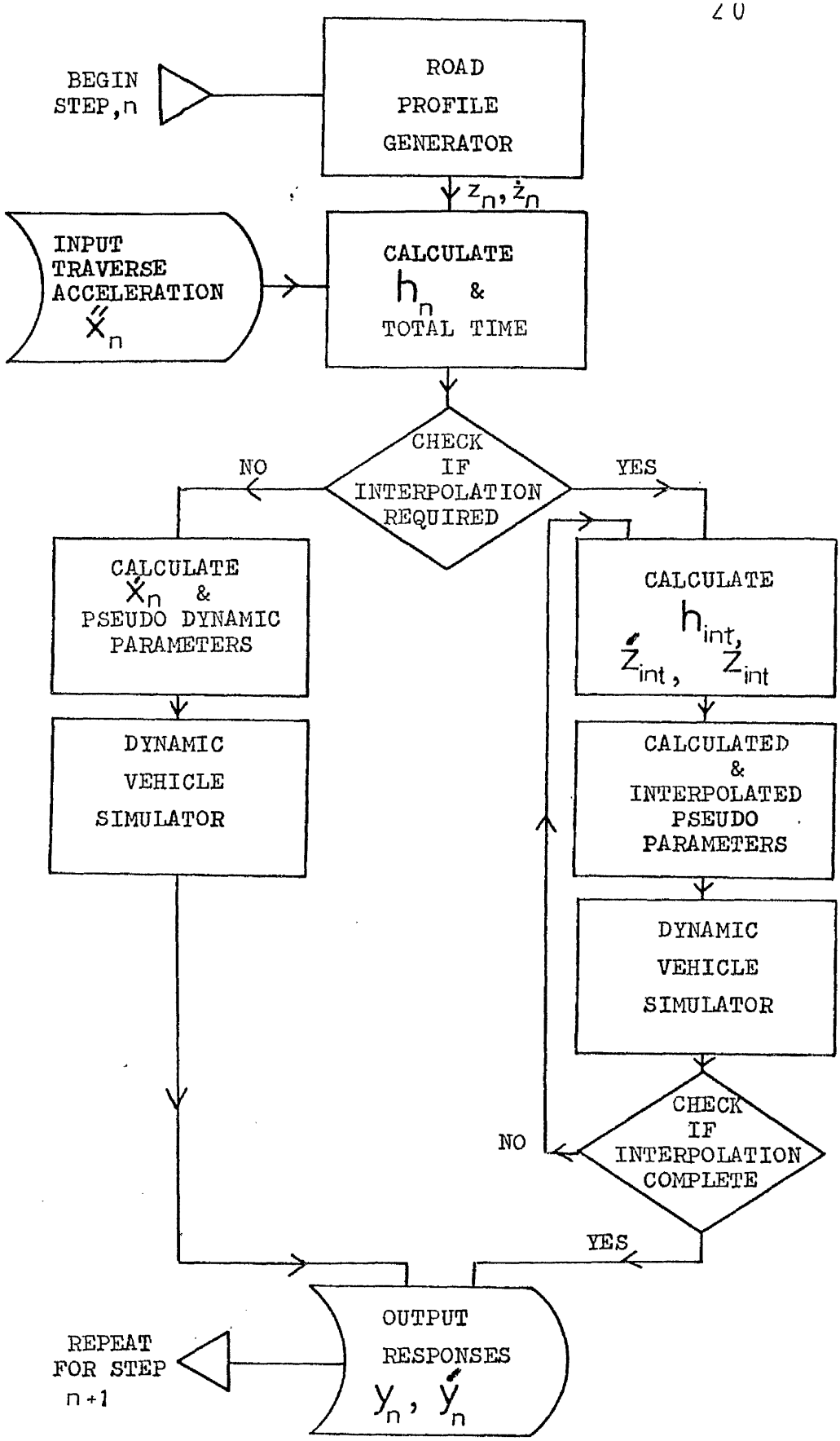


FIG 9.1 Program flow logistics of Dynamic Vehicle Simulator

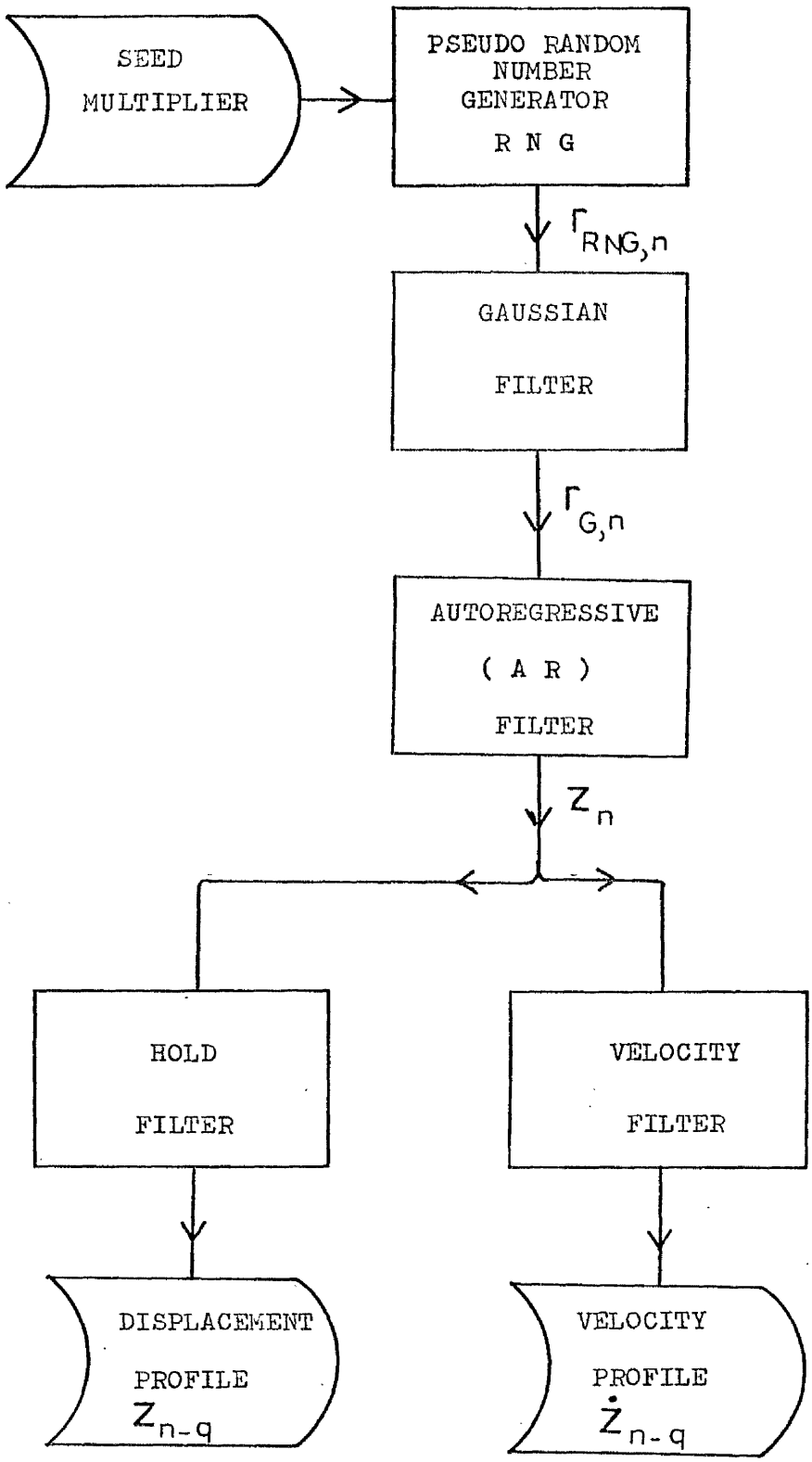


FIG 3.1 Program flow logistics of the Road Profile Generator.

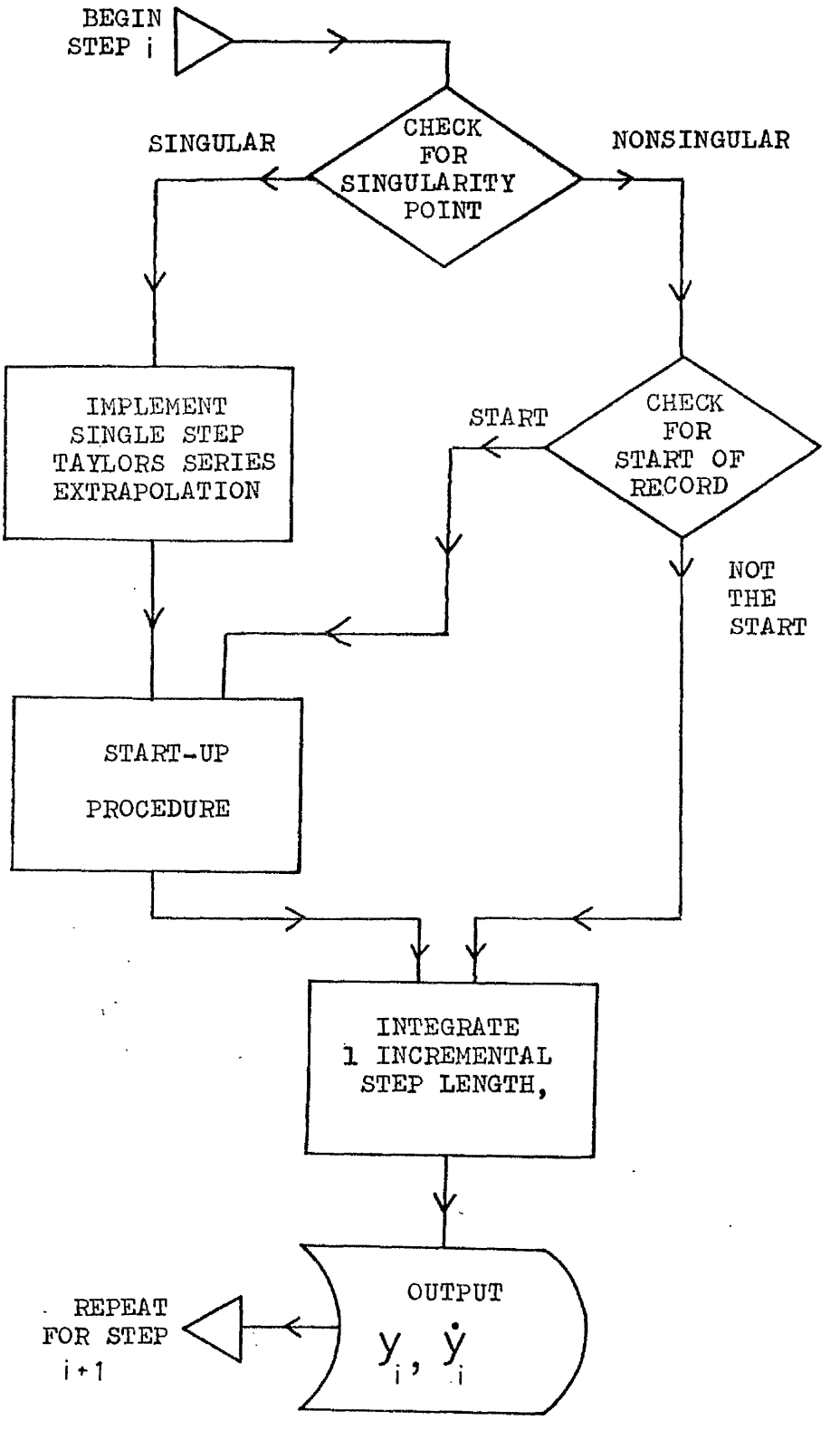


FIG 9.2 Program flow logistics of the spatially based Dynamic Vehicle Integrator.

PROGRAM DSYS2

```
?63UL=UNIFIL(1).DSYS2
      1       :BEGIN;                                                                  00000000
      2       :INTEGER:NOTEST,NORNG,NOFILT,FLTORD,INITO,NOPTS,INTORD,NOTERP,DUOFLT,    00000001
      3                TRNSNT,LAG,TOTPTS,TRNS1,FLTNO,PTNO,TST,J,L,I,DFLTNO,DFNOM1,     00000002
      4                HNO,COUNT,NPP,TERP,TOTPT1,SNOTRP,TRNM1P,FLTST,SDNOM1,BLOCK,     00000003
      5                 TRNS2,MEMBER;                                                  00000004
      6       :REAL:ZETA,NFREQ,WN,TOTTIM,C1,NORMRT,MAXFRQ,DXTNW,D2XTNW,DXT,D2XT,DT,    00000005
      7             STPLTH,DTMAX,DXT1,D2XT1,DX,REDUND,YT1,YTO1,YTO2,DYT1,DYTO1,DYTO2,  00000006
      8             FY1,FDY1,                                                          00000007
      9            DIST,D2XTG,DTMX2C,SCX,FACT,RDUMMY,DXTO,DTTERP,BLKTIM,NWBTIM,SDFLTN;00000008
     10       :BOOLEAN:NRMLSE,COLATE;                                                  00000009
     11        NOTEST&>READ;                                                           00000010
     12        NORNG&>READ;                                                            00000011
     13        NOFILT&>READ;                                                           00000012
     14       FLTORD&>READ;                                                            00000013
     15        INITO&>3;                                                               00000014
     16        NOPTS&>READ;                                                            00000015
     17        INTORD&>READ;                                                           00000016
     18        STPLTH&>READ;                                                           00000017
     19        HNO&>READ;                                                              00000018
     20        BLOCK&>READ;                                                            00000019
     21        MEMBER&>READ;                                                           00000020
     22        DUOFLT&>2*NOFILT;                                                       00000021
     23        TRNM1&>5*FLTORD;                                                        00000022
     24        TRNSNT&>TRNM1&<18;                                                      00000023
     25        LAG&>NOPTS/2;IO;                                                        00000024
     26        TOTPTS&>NOPTS<TRNSNT;                                                   00000025
     27        TOTPT1&>TOTPTS<1;                                                       00000026
     28        TRNS1&>TRNSNT<1;                                                        00000027
     29        TRNS2&>TRNSNT<2;                                                        00000028
     30        NPP&>25;/;NOFILT;                                                       00000029
     31        NRMLSE&>:FALSE:;                                                        00000030
     32        COLATE&>:TRUE:;                                                         00000031
     33        WRITETEXT(0: NOTEST     NORNG     NOFILT     FLTORD     TRNSNT     NOPTS 00000032
     34             LAG     TOTPTS:3:3;                                                00000033
     35        NEALINE(1);                                                             00000034
     36        PRINTENOTEST,7,0);                                                      00000035
     37        PRINTENORNG,7,0);                                                       00000036
     38        PRINTENOFILT,7,0);                                                      00000037
     39        PRINTEFLTORD,7,0);                                                      00000038
     40        PRINTETRNSNT,7,0);                                                      00000039
     41        PRINTENOPTS,7,0);                                                       00000040
     42        PRINTELAG,7,0);                                                         00000041
     43        PRINTETOTPTS,7,0);                                                      00000042
     44        NEWLINE(0);                                                             00000043
     45       :BEGIN;                                                                  00000044
     46       :ARRAY:Y,DYC/1&NOFILT,3&INTORD/3,                                        00000045
     47              ZC/1&DUOFLT,1&FLTORD<INITO<1/3,                                   00000046
     48              ZI,DAMZ,SDYZC/1&DUOFLT/3,                                         00000047
     49              RORTA&C/1&NOFILT,1&FLTORD/3,                                      00000048
     50              RAMJC/1&DUOFLT,3&LAG/3,                                           00000049
     51              RTA&ZC/1&DUOFLT,1&DUOFLT,3&LAG/3,                                 00000050
     52              TY,TDYC/1&NOFILT,1&3/3,                                           00000051
     53              SPRMAY,SPRMAZC/1&NOFILT,1&INTORD/3,                               00000052
     54              FLTRIN,PARMA,PARMAY,PARMAZ,PARMB,FLTDEVC/1&NOFILT/3,              00000053
     55              MCCI,MCYI,BIGDZY,                                                 00000054
     56              BIGY,BIGDY,BIGZ,BIGDZ,MSY,MSDY,MSYZC/1&NOFILT,TRNSNT&TOTPT1/3,    00000055
```

```
          MSZIX,MSDZIX,MSZIX,MSDZIX,MSYT,MSDYT,MSYZT,MSZIT,MSDZITE/1&NOFILT/3,
           FIE/1&NOFILT,1&NOFILT,1&FLTORD/3,
           TLZE/1&NOFILT,3&6,-18&18/3,TLYE/1&NOFILT,0&6/3,
           TLPANDE/1&DUOFLT,1&FLTORD<INITO<1/3,
           BLKDTE/3&BLOCK/3,
           VALY,VALDY,VALYZ,VALZI,VALDZIE/1&NOFILT,3&BLOCK/3,
           LXE/-3&3/3,
           SPARME/1&INTORD-1/3,
           CORCOFE/3&INTORD/3:
      :INTEGER::ARRAY:MULT,SEEDE/1&NOFILT,1&NORNG/3,
                      MULTD,SEEDDE/1&NOFILT,1&NORNG/3,
                        TLSDE/1&NOFILT,1&NORNG/3,
                        PQNTIL,AQNTILE/1&NOFILT,-HNO&HNO-1/3:
      :REAL:C1,NORMRT,STDEV,PARM,MS:


      :COMMENT:   LIST OF PROCEDURE DECLARATIONS
                  ******************************:

      :PROCEDURE:DEVATN(STDEV,FLTDEV,ROWTAW,FI,FLTORD,FLTNO):

      :COMMENT:   CALCULATES THE THEORETICAL STANDARD DEVIATION FLTDEV OF AN
          A.R. FILTER , WITH A RANDOM INPUT HAVING STANDARD DEVIATION STDEV.
          ARRAY FI CONTAINS THE FILTER COEFFICIENTS + ARRAY ROWTAW THE THEOR-
          ETICAL CORRELATIONS:

      :VALUE:STDEV,FLTORD,FLTNO:
      :REAL:STDEV,FLTDEV:
      :ARRAY:ROWTAW,FI:
      :INTEGER:FLTORD,FLTNO:
      :BEGIN:
          :INTEGER:I:
           FLTDEV&>1:
          :FOR:I&>I:STEP:1:UNTIL:FLTORD:DO:
                FLTDEV&>FLTDEV-ROWTAWE/FLTNO,I/3*FIE/FLTNO,I,I/3:
           FLTDEV&>STDEV/SQRT(FLTDEV):
      :END:   OF  DEVATN:


      :PROCEDURE:STEPEY,DY,ZI,TY,TDY,CORCOF,PARMAY,PARMAZ,PARMB,YT1,YTO1,YTO2,
                 DYT1,DYTO1,DYTO2,FLTNO,INTORD,STPLTH,PARM,FY1,FDY1):


      :COMMENT:   IMPLEMENTS ONE R.F.C.F.C.F. STEP OF A DYNAMICALLY VARYING
          2ND. ORDER D.E. WHERE YE/FLTNO,0/3 + DYE/FLTNO,0/3 ARE THE SOLN. OF
          VARIABLE + THE 1ST. DERIVATIVE RESP.  ZI IS AN ARRAY CONTAINING THE
          DYNAMICALLY VARYING INPUTS TO THE D.E.  THE INTEGRATION METHOD IS OF
          THE ONE STEP TYPE.  PARMAY,PARMAZ,PARMB ARE THE DYNAMICALLY VARYING
          DYNAMICAL PARAMETERS OF THE SYSTEM, STPLTH IS THE INTEGRATION STEP
          THE ORDER OF  THE INTEGTATION.  FLTNOIS AN INDEX VARIABLE.  TY,TDY,
          YT1,YTO1,YTO2,DYT1,DYTO1,DYTO2 GIVE INFORMATION ON THE CONVERGENCE
          OF THE ROUTINE:


      :VALUE:FLTNO,INTORD,STPLTH,PARMAY,PARMAZ,PARMB,PARM:
      :INTEGER:FLTNO,INTORD:
      :REAL:PARMAY,PARMAZ,PARMB,YT1,YTO1,YTO2,DYT1,DYTO1,DYTO2,STPLTH,PARM,FY1
            ,FDY1:
```

```
:ARRAY:Y,DY,ZI,TY,TDY,CORCOF;                                                  00000113
:BEGIN:                                                                        00000114
    :INTEGER:I;                                                                00000115
     PRDICTEY,INTORD,TY,FLTNO];                                                00000116
     PRDICTEDY,INTORD,TDY,FLTNO];                                              00000117
    :FOR:I&>2:STEP:1:UNTIL:3:DO:                                               00000118
    :BEGIN:                                                                    00000119
          FUNCTEFY1,FDY1,YE/FLTNO,0/3,DYE/FLTNO,0/3,ZIE/DFNOM1/3,              00000120
               ZIE/DFLTNO/3,STPLTH,PARMAY,PARMAZ,PARMB,FLTNO,PARM3;            00000121
          CORECTEDY,TDY,FDY1,CORCOF,INTORD,FLTNO,I3;                           00000122
          CORECTEY,TY,FY1,CORCOF,INTORD,FLTNO,I3;                              00000123
    :END:;                                                                     00000124
     FUNCTEFY1,FDY1,YE/FLTNO,0/3,DYE/FLTNO,0/3,ZIE/DFNOM1/3,ZIE/DFLTNO/3       00000125
          ,STPLTH,PARMAY,PARMAZ,PARMB,FLTNO,PARM3;                             00000126
     TESTEY,TY,FY1,YT1,YT01,YT02,FLTNO];                                       00000127
     TESTEDY,TDY,FDY1,DYT1,DYT01,DYT02,FLTNO];                                 00000128
:END:  OF  STEP;                                                               00000129
                                                                               00000130
:PROCEDURE:STARTEY,DY,ZI,TY,TDY,CORCOF,SPRMAY,SPRMAZ,PARMB,YT1,YT01,YT02       00000131
               ,DYT1,DYT01,DYT02,FLTNO,INTORD,STPLTH,Z,FLTORD,NOTERP,          00000132
               PARM,FY1,FDY1];                                                 00000133
                                                                               00000134
:COMMENT:   IMPLEMENTS A PROCEDURE TO ESTABLISH HIGHER INDEX VALUES OF         00000135
    THE ARRAYS Y,DY.  THESE ARE INITIALLY UNDEFINED AND ARE NECESSARY          00000136
    FOR A SOLN. TO THE ONE STEP PREDICTOR CORRECTOR USED IN THE                00000137
    PROCEDURE STEP.  THE METHOD OPERATES AS FOLLOWS &-                         00000138
      INITIALISE YE/FLTNO,0/3,+ DYE/FLTNO,0/3-FLTNO IS AN EXTERNALLY           00000139
    DEFINED  INDEX NO. - + ASSIGN THE HIGHER INDEXED VALUES Y,DYE/FLTNO,       00000140
    I/3,FOR I\3,TO ZERO.   IMPLEMENT THE PROCEDURE STEP SEVERAL TIMES,         00000141
    THE NUMBER OF TIMES BEING GOVERNED BY INTORD, SO THAT ALL HIGHER           00000142
    INDEXED VALUES OF Y,DY ARE EVALUATED.  REVERSE THE DIRECTION OF            00000143
    INTEGRATION + USING STEP RETURN TO STARTING POINT.   REINITIALISE Y,       00000144
    DYE/FLTNO,0/3 #HILE RETAINING THE HIGHER INDEXED VALUES OF THESE           00000145
    ARRAYS. REPEAT SEVERAL TIMES UNTIL THE DESIRED ACCURACY IS OBTAINED;       00000146
                                                                               00000147
                                                                               00000148
:VALUE:FLTNO,INTORD,STPLTH,PARMB,FLTORD,NOTERP,PARM;                           00000149
:INTEGER:FLTNO,INTORD,FLTORD,NOTERP;                                           00000150
:REAL:YT1,YT01,YT02,DYT1,DYT01,DYT02,STPLTH,FY1,FDY1;                          00000151
:ARRAY:Y,DY,ZI,TY,TDY,CORCOF,Z,PARM,SPRMAY,SPRMAZ,PARMB;                       00000152
:BEGIN:                                                                        00000153
    :ARRAY:YTIMD,DYTIMDE/1&NOFILT/3;                                           00000154
    :INTEGER:SIGN,REITER,J,I,0,SFLTNO;                                         00000155
    :FOR:SFLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                                     00000156
    :BEGIN:                                                                    00000157
          YTIMDE/SFLTNO/3&>YE/FLTNO,0/3;                                       00000158
          DYTIMDE/SFLTNO/3&>DYE/FLTNO,0/3;                                     00000159
         :FOR:J&>1:STEP:1:UNTIL:INTORD:DO:                                     00000160
               YE/SFLTNO,J/3&>DYE/SFLTNO,J/3&>0:                               00000161
    :END:;                                                                     00000162
     SIGN&>-1;                                                                 00000163
    :FOR:REITER&>1:STEP:1:UNTIL:4:DO:                                          00000164
    :BEGIN:                                                                    00000165
          SIGN&>-SIGN;                                                         00000166
         :FOR:J&>1:STEP:1:UNTIL:INTORD-I:DO:                                   00000167
         :BEGIN:                                                               00000168
              :IF:SIGN)0:THEN:   0&>INTORD-J-I                                 00000169
```

```
                    :ELSE:                                                          00000170
                    :BEGIN:                                                         00000171
                          G&>J;                                                     00000172
                         :IF:J>1:THEN:                                              00000173
                         :FOR:SFLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                     00000174
                         :BEGIN:                                                    00000175
                               ZIC/2*SFLTNO-1/3&>ZC/2*SFLTNO-1,FLTORD-1/3;          00000176
                               ZIC/2*SFLTNO/3&>ZC/2*SFLTNO,FLTORD-1/3;              00000177
                         :END:;                                                     00000178
                    :END:;                                                          00000179
                    :FOR:SFLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                          00000180
                    :BEGIN:                                                         00000181
                     INTERPC ZC/2*SFLTNO-1,FLTORD-1/3,ZC/2*SFLTNO-1,FLTORD/3,       00000182
                           ZIC/2*SFLTNO-1/3,0,NOTERP*SIGN];                         00000183
                     INTERPC ZC/2*SFLTNO,FLTORD-1/3,ZC/2*SFLTNO,FLTORD/3,           00000184
                           ZIC/2*SFLTNO/3,0,NOTERP*SIGN];                           00000185
                     STEPCY,DY,ZI,TY,TDY,CORCOF,SPRMAYC/SFLTNO,J/3,SPRMAZE/         00000186
                        SFLTNO,J/3,PARMBC/FLTNO/3,YT1,YT01,                         00000187
                        YT02,DYT1,DYT01,DYT02,FLTNO,INTORD,STPLTH,PARMC/J/3         00000188
                        ,FY1,FDY1];                                                 00000189
                    :END:;                                                          00000190
                    :IF:SIGN)0 + 0>0:THEN:                                          00000191
                    :FOR:SFLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                          00000192
                    :BEGIN:                                                         00000193
                          YC/SFLTNO,0/3&>YTIMOC/SFLTNO/3;                           00000194
                          DYC/SFLTNO,0/3&>DYTIMOC/SFLTNO/3;                         00000195
                    :END:;                                                          00000196
               :END:;                                                               00000197
                STPLTH&>-STPLTH;                                                    00000198
               :FOR:SFLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                               00000199
               :FOR:I&>1:STEP:1:UNTIL:INTORD:DO:                                    00000200
               :BEGIN:                                                              00000201
                     YC/SFLTNO,I/3&>-YC/SFLTNO,I/3;                                 00000202
                     DYC/SFLTNO,I/3&>-DYC/SFLTNO,I/3;                               00000203
               :END:;                                                               00000204
          :END:;                                                                    00000205
     :END:   OF  START;                                                             00000206
                                                                                    00000207
                                                                                    00000208
     :PROCEDURE:GRADCERN,GWDTH,QUNTLE,HNO,L,TRNSNT,PTNO];                           00000209
                                                                                    00000210
     :COMMENT:   USED TO NOTE THE QUANTILE,QUNTLEC/L,TEMP/3,INTO WHICH THE          00000211
               RANDOM NUMBER FALLS. L IS AN EXTERNALLY EVALUATED INDEX *            00000212
               -HNO)>TEMP)>HNO-1.                                                   00000213
               HNO IS AN INTEGER EQUAL TO HALF NUMBER OF QUANTILES REQUIRED.        00000214
               THESE QUANTILES ARE SYMMETRIC ABOUT THE MEAN VALUE OF RN WHICH       00000215
               IS ASSUMED TO BE ZERO.                                               00000216
               THE QUANTILE BANDWIDTH IS GIVEN BY GWDTH.                            00000217
               PTNO IS THE INDEX OF THE CURRENT RANDOM NUMBER,RN,* TRNSNT           00000218
               INDICATES AT WHICH PTNO THE PROCEDURE SHOULD START OPERATING;        00000219
                                                                                    00000220
     :VALUE:RN,GWDTH,HNO,L,TRNSNT,PTNO;                                             00000221
     :REAL:RN,GWDTH;                                                                00000222
     :INTEGER:::ARRAY:QUNTLE;                                                       00000223
     :INTEGER:HNO,L,TRNSNT,PTNO;                                                    00000224
     :BEGIN:                                                                        00000225
          :INTEGER:TEMP,I;                                                          00000226
```

```
228         :IF:PTNO)>TRNSNT:THEN:                                          00000227
229         :FOR:I&>-HNO:STEP:1:UNTIL:HNO-1:DO:                             00000228
230                 QUNTLEC/L,I/J&>0;                                       00000229
231          TEMP&>ENTIERCRN/G#DTH3;                                        00000230
232         :IF:TEMP)-HNO:THEN:   TEMP&>-HNO                                00000231
233         :ELSE::IF:TEMP\>HNO:THEN:   TEMP&>HNO-1;                        00000232
234          QUNTLEC/L,TEMP/J&>QUNTLEC/L,TEMP/J<1;                          00000233
235     :END:   OF  GRADQ;                                                  00000234
236                                                                         00000235
237                                                                         00000236
238     :PROCEDURE:PREDICTEY,K,TY,FLTNO];                                   00000237
239                                                                         00000238
240     :COMMENT:   IMPLEMENTS A SINGLE NUMERICAL PREDICTION OF ONE STEP METHOD.00000239
241         IT ESTIMATES THE NEW VALUE OF THE Y ARRAYCI.E. THE ESTIMATED SOLN.  00000240
242         YE/0/1 PLUS COMBINATIONS OF HIGHER ORDER DERIVATIVES].  K IS THE    00000241
243         ORDER OF THE PREDICTION PROCEDURE.  TY IS A SAVE STORE FOR THE SOLN.00000242
244         YE/0/1.  FLTNO IS AN INDEX VARIABLE;                                00000243
245                                                                         00000244
246     :VALUE:K,FLTNO;                                                     00000245
247     :INTEGER:K,FLTNO;                                                   00000246
248     :ARRAY:Y,TY;                                                        00000247
249     :BEGIN:                                                             00000248
250         :INTEGER:II,JJ;                                                 00000249
251         :FOR:JJ&>0:STEP:1:UNTIL:K-1:DO:                                 00000250
252         :FOR:II&>K:STEP:-1:UNTIL:JJ<1:DO:                               00000251
253                 YE/FLTNO,II-1/J&>YE/FLTNO,II-1/J<YE/FLTNO,II/J;         00000252
254          TYE/FLTNO,I/J&>YE/FLTNO,0/J;                                   00000253
255     :END:   OF  PREDICT;                                                00000254
256                                                                         00000255
257                                                                         00000256
258     :PROCEDURE:FUNCTEFYI,FDYI,Y,DY,Z,DZ,H,AY,AZ,B,FLTNO,PARM];         00000257
259                                                                         00000258
260     :COMMENT:   PERFORMS A FUNCTION EVALUATION OF A 2ND. ORDER D.E.  THE 00000259
261         APPROPRIATE IS INDEXED BY FLTNO.  FYI IS THE EVALUATION         00000260
262         OF THE 1ST, DERIVATIVE OF Y+ FDYI IS THE 1ST. OF DY.  Z,DZ ARE INPUT00000261
263         VARIABLES.  H IS THE INTEGRATION STEP LENGTH.  THE OTHER PARAMETERS 00000262
264         ARE THE DYNAMICAL PARAMETERS OF THE D.E. :                      00000263
265                                                                         00000264
266     :VALUE:Y,DY,Z,DZ,AY,AZ,B,H,FLTNO,PARM;                              00000265
267     :REAL:FYI,FDYI,Y,DY,Z,DZ,AY,AZ,B,H,PARM;                            00000266
268     :INTEGER:FLTNO;                                                     00000267
269     :BEGIN:                                                             00000268
270          FYI&>DY#H;                                                     00000269
271         :COMMENT:CHOOSE FDYI ACCORDING TO FLTNO:                        00000270
272          FDYI&>CAZ#DZ-AY#DY<B#CZ-Y]]#PARM;                              00000271
273     :END:   OF  FUNCT;                                                  00000272
274                                                                         00000273
275                                                                         00000274
276     :PROCEDURE:CORECTEY,TY,FYI,C,INTORD,FLTNO,I];                       00000275
277                                                                         00000276
278     :COMMENT:   IMPLEMENTS A SINGLE NUMERICAL INTEGRATION CORRECTION OF ONE 00000277
279         STEP METHOD.  IT CORRECTS THE VALUE OF YE/I/J,WHERE I>0,....INTORD, 00000278
280         BY COMPARING THE OLD VALUE YE/I/J WITH  THAT OBTAINED INTHE FUNCTION00000279
281         EVALUATION,FYI, + CORRECTING ACCORDINGLY.  THE ARRAY C CONTAINS THE 00000280
282         APPROPRIATE MULTIPLICANDS.  TYE/L/J IS A SAVE STORE FOR THE CURRENT 00000281
283         CORRECTION OF YE/0/J.  FLTNO+ I ARE INDEX VARIABLES:               00000282
284                                                                         00000283
```

```
:VALUE:FY1,INTORD,FLTNO,I;                                                    00000284
:REAL:FY1;                                                                    00000285
:INTEGER:INTORD,FLTNO,I;                                                      00000286
:ARRAY:Y,TY,C;                                                                00000287
:BEGIN:                                                                       00000288
    :REAL:DIFF;                                                               00000289
    :INTEGER:JJ;                                                              00000290
     DIFF&>FY1-YC/FLTNO,1/3;                                                  00000291
    :FOR:JJ&>0:STEP:1:UNTIL:INTORD:DO:                                        00000292
          YC/FLTNO,JJ/3&>YC/FLTNO,JJ/3<CC/JJ/3*DIFF;                          00000293
     TYC/FLTNO,I/3&>YC/FLTNO,0/3;                                             00000294
:END:   OF  CORECT;                                                           00000295
                                                                              00000296
                                                                              00000297
:PROCEDURE:TESTCY,TY,FY1,YT1,YTO1,YTO2,FLTNO3;                                00000298
                                                                              00000299
:COMMENT:   TESTS FOR THE CONVERGENCE OF THE FUNCT-CORECT ITERATIONS.         00000300
    IT COMPARES THE DIFFERENCE IN VALUES OF TYCI.E. THE VALUE OF YI           00000301
    OBTAINED IN SUCCESSIVE ITERATIONS.   THESE ARE OUTPUT AS YTO1 + YTO2      00000302
    IT ALSO COMPARES THE FIRST DERIVATIVE,YC/I/3, WITH THE FUNCTION           00000303
   EVALUATION FY1 + OUTPUTS THE DIFFERENCE YT1 ;                              00000304
                                                                              00000305
:VALUE:FY1,FLTNO;                                                             00000306
:REAL:YT1,YTO1,YTO2,FY1;                                                      00000307
:ARRAY:Y,TY;                                                                  00000308
:INTEGER:FLTNO;                                                               00000309
:BEGIN:                                                                       00000310
     YTO2&>ABSETYC/FLTNO,3/3-TYC/FLTNO,2/33;                                  00000311
     YTO1&>ABSETYC/FLTNO,2/3-TYC/FLTNO,1/33/8;                                00000312
     YT1&>ABSEFY1-YC/FLTNO,1/33;                                              00000313
:END:   OF  TEST;                                                             00000314
                                                                              00000315
                                                                              00000316
:PROCEDURE:TIMSUMCSTPLTH,DXT,D2XT,DT3;                                        00000317
                                                                              00000318
:COMMENT:   CALCULATES THE TIME INCREMENT DT FROM A KNOWLEDGE OF THE          00000319
    DISTANCE INCREMENT STPLTH, THE TRAVERSE VELOCITY DXT + ACCELERATION       00000320
    D2XT .  DXT + D2XT ARE ASSUMED CONSTANT OVER THE INCREMENT;               00000321
                                                                              00000322
:VALUE:STPLTH,DXT,D2XT;                                                       00000323
:REAL:STPLTH,DXT,D2XT,DT;                                                     00000324
:BEGIN:                                                                       00000325
    :IF:D2XT>0:THEN:   DT&>STPLTH/DXT                                         00000326
    :ELSE:   DT&>C-DXT<SQRTCDXT*DXT<2*D2XT*STPLTH33/D2XT;                     00000327
:END:   OF  TIMSUM;                                                           00000328
                                                                              00000329
                                                                              00000330
:PROCEDURE:DTSHFTCDT,NEWDT,BLOCK,INITPT,PTNO,BLKTIM,NWBTIM3;                  00000331
                                                                              00000332
:COMMENT:   MOVES MEMBERS OF THE DT ARRAY ALONG BY 1+ INPUTS NEWDT AT         00000333
     DTC/BLOCK/3. IT CALCULATES THE NEW BLOCK SUM NWBTIM FROM THE OLD         00000334
     BLOCK SUM BLKTIM;                                                        00000335
                                                                              00000336
:VALUE:NEWDT,BLOCK,INITPT,PTNO;                                               00000337
:INTEGER:BLOCK,INITPT,PTNO;                                                   00000338
:REAL:NEWDT,BLKTIM,NWBTIM;                                                    00000339
:ARRAY:DT;                                                                    00000340
```

```
342     :BEGIN:                                                                 00000341
343         :INTEGER:II;                                                        00000342
344         :IF:PTNO)>INITPT:THEN:                                              00000343
345         :BEGIN:                                                             00000344
346               BLKTIM&>0;                                                    00000345
347               NWBTIM&>0;                                                    00000346
348              :FOR:II&>0:STEP:I:UNTIL:BLOCK:DO:                              00000347
349                    DTC/II/J&>0;                                             00000348
350         :END:;                                                              00000349
351         :FOR:II&>0:STEP:I:UNTIL:BLOCK-1:DO:                                 00000350
352               DTC/II/J&>DTC/II<1/J;                                         00000351
353          DTC/BLOCK/J&>NEWDT;                                                00000352
354          BLKTIM&>NWBTIM;                                                    00000353
355          NWBTIM&>BLKTIM<DTC/BLOCK/J-DTC/0/J;                                00000354
356     :END:   OF DTSHIFT;                                                     00000355
357                                                                             00000356
358                                                                             00000357
359     :PROCEDURE:BKMNSQ(INPUT,DT,MEAN,VAL,BLKTIM,NWBTIM,BLOCK,INITPT,PTNO,J); 00000358
360                                                                             00000359
361     :COMMENT:   CALCULATES CONTINUOUSLY THE MEAN SQUARE OF A BLOCK OF LENGTH00000360
362          BLOCK;                                                             00000361
363                                                                             00000362
364     :VALUE:PTNO,INITPT,BLOCK,DT,INPUT,BLKTIM,NWBTIM,J;                      00000363
365     :INTEGER:PTNO,INITPT,BLOCK,J;                                           00000364
366     :REAL:MEAN,DT,INPUT,BLKTIM,NWBTIM;                                      00000365
367     :ARRAY:VAL;                                                             00000366
368     :BEGIN:                                                                 00000367
369         :INTEGER:II;                                                        00000368
370         :IF:PTNO)>INITPT:THEN:                                              00000369
371         :BEGIN:                                                             00000370
372               MEAN&>0;                                                      00000371
373              :FOR:II&>0:STEP:I:UNTIL:BLOCK:DO:                              00000372
374                    VALC/J,II/J&>0;                                          00000373
375         :END:;                                                              00000374
376         :FOR:II&>0:STEP:I:UNTIL:BLOCK-1:DO:                                 00000375
377               VALC/J,II/J&>VALC/J,II<1/J;                                   00000376
378          VALC/J,BLOCK/J&>INPUT*INPUT*DT;                                    00000377
379         :IF:NWBTIM>0:THEN:    MEAN&>0                                       00000378
380         :ELSE:MEAN&>(MEAN*BLKTIMJ-VALC/J,0/J<VALC/J,BLOCK/JJ/NWBTIM;        00000379
381     :END:   OF BKMNSQ;                                                      00000380
382     :PROCEDURE:MNSQREGY,DT,MS,TJ;                                           00000381
383                                                                             00000382
384     :COMMENT:   CALCULATES THE MEAN SQUARE RESPONSE MS OF Y BY INTEGRATING  00000383
385         Y**2(TJW.R.T. TIME IN DISCRETE TIME INCREMENTS DT, THUS THE MEAN    00000384
386         SQUARE RESPONSE OVER INCREMENT DT IS FOUND;                         00000385
387                                                                             00000386
388     :VALUE:DT,Y,T;                                                          00000387
389     :REAL:MS,DT,Y,T;                                                        00000388
390          MS&>MS<(Y*Y-MSJ*DT/T;                                              00000389
391                                                                             00000390
392                                                                             00000391
393     :PROCEDURE:INTERP(X1,X2,XI,TERP,NOTERPJ;                                00000392
394                                                                             00000393
395     :COMMENT:   INTERPOLATES LINEARLY BETWEEN ZPTS.  AT DISTANCE RATIO      00000394
396         TERP/NOTERP FROM X1 ;                                               00000395
397     :VALUE:TERP,NOTERP;                                                     00000396
398     :INTEGER:TERP,NOTERP;                                                   00000397
```

```
399     :REAL:X1,X2,X1:                                                    00000398
400     :BEGIN:                                                            00000399
401         :IF:TERP>7:THEN:   X1&>X1                                      00000400
402         :ELSE:   X1&>X1<(X2-X1)/NOTERP:                                00000401
403     :END:   OF INTERP:                                                 00000402
404                                                                        00000403
405                                                                        00000404
406     :PROCEDURE:POLARESEED,MULT,L,PTNO,PQNTIL,FAIL,FLTRIN,TOTPTS,REDUND,00000405
407                    TRNSNT,HNO,STDEV):                                  00000406
408                                                                        00000407
409     :COMMENT:   GENERATES THE LTH. GAUSSIAN RANDOM OUTPUT FLTRIN BY MEANS OF00000408
410         A POLAR GENERATOR WHICH REQUIRES 3 INDEPENDANT PSEUDO RANDOM NUMBERS00000409
411         SEEDE/L,I/), WHERE I>1,2,3. THE CURRENT VALUE OF SEEDE/L,I/) IS 00000410
412         GENERATED USING AGLOBAL RANDOM NUMBER GENERATOR , FUNCTIO RNG. 00000411
413         SEEDE/L,I/) IS THEN NORMALISED + GIVEN A ZERO MEAN + OUTPUT AS 00000412
414         FLTRIN. MULTE/L,I/) IS THE CURRENTLY INDEXED PRIMITIVE ROOT    00000413
415         REQUIRED BY FUNCTION RNG FOR SEEDE/L,I/). REDUND NOTES THE FRACTION00000414
416         OR THE DEGREE OF REDUNDANCY IN THE POLAR GENERATOR. PTNO IS THE 00000415
417         CURRENT TIME INCREMENT. TOTPTS IS THE FINAL TIME STEP. FAIL IS THE00000416
418         EXIT LABEL SHOULD THE FUNCTION RNG BREAKDOWN:                  00000417
419                                                                        00000418
420     :VALUE:L,PTNO,TOTPTS,TRNSNT,HNO:                                   00000419
421     :INTEGER:L,PTNO,TOTPTS,TRNSNT,HNO:                                 00000420
422     :REAL:FLTRIN,STDEV,REDUND:                                         00000421
423     :INTEGER::ARRAY:SEED,MULT,PQNTIL:                                  00000422
424     :LABEL:FAIL:                                                       00000423
425     :BEGIN:                                                            00000424
426         :INTEGER:I,TEMP,INT:                                           00000425
427         :REAL:SUM,DIFF,RAD:                                            00000426
428         :ARRAY:RN,SQRNE/1&3/):                                         00000427
429         :IF:PTNO>0:THEN:   REDUND&>0:                                  00000428
430          STDEV&>1:                                                     00000429
431         RELOOP&                                                        00000430
432         :FOR:I&>1:STEP:1:UNTIL:3:DO:                                   00000431
433         :BEGIN:                                                        00000432
434               SEEDE/L,I/)&>RNGESEEDE/L,I/),MULTE/L,I/)):               00000433
435              :IF:SEEDE/L,I/))0:THEN:                                   00000434
436              :BEGIN:                                                   00000435
437                    #RITETEXTE:E:RNG FAILURE:):):                       00000436
438                    NEWLINEE1):                                         00000437
439                   :GOTO:FAIL:                                          00000438
440              :END::                                                    00000439
441               RNE/I/)&>SEEDE/L,I/)/65537:                              00000440
442         :END::                                                         00000441
443          RNE/1/)&>(RNE/1/)-0.5)*2:                                     00000442
444          SQRNE/1/)&>RNE/1/)*RNE/1/):                                   00000443
445          SQRNE/2/)&>RNE/2/)*RNE/2/):                                   00000444
446          SUM&>SQRNE/1/)<SQRNE/2/):                                     00000445
447         :IF:SUM\1:THEN:                                                00000446
448         :BEGIN:                                                        00000447
449               REDUND&>REDUND<1:                                        00000448
450              :GOTO:RELOOP:                                             00000449
451         :END::                                                         00000450
452          DIFF&>SQRNE/2/)-SQRNE/1/):                                    00000451
453          RAD&>SQRTE-2*LNERNE/3/))):                                    00000452
454          FLTRIN&>RAD*DIFF/SUM:                                         00000453
455          GRADGEFLTRIN,STDEV,PQNTIL,HNO,L,TRNSNT,PTNO):                 00000454
```

```
:END:   OF POLAR :                                                              00000455
                                                                                00000456
                                                                                00000457
:INTEGER::PROCEDURE: RNG(XN,KAY);                                               00000458
                                                                                00000459
:COMMENT:   A PSEUDO RANDOM NUMBER GENERATOR;                                   00000460
                                                                                00000461
:VALUE:XN,KAY;                                                                  00000462
:INTEGER:XN,KAY;                                                                00000463
:BEGIN:                                                                         00000464
     XN&>KAY*XN;                                                                00000465
     RNG&>XN-(XN:/:65537)*65537;                                                00000466
:END:   OF RNG :                                                                00000467
                                                                                00000468
                                                                                00000469
:PROCEDURE:ARFILT(FLTRIN,RAND,FI,FLTORD,L,PTNO,AQNTIL,FLTDEV,TOTPTS,            00000470
                  NOFILT,INITO,TRNSNT,HNO);                                     00000471
                                                                                00000472
:COMMENT:   GENERATES A SINGLE OUTPUT RAND(L,FLTORD) OF A RANDOM                00000473
    REALISATION ACCORDING TO A PRESCRIBED L:TH CORRELATION CURVE, BY            00000474
    MEANS OF A LINEAR AUTOREGRESSIVE FILTER& - MONOVARIABLE IF NOFILT>1         00000475
                                          MULTIVARIABLE IF NOFILT\1.            00000476
    THE INPUT TO THE FILTER IS THE GAUSSIAN RANDOM NUMBER FLTRIN.               00000477
    FI(L,J,I) ARE THE COEFFICIENTS OF THE L:TH AUTOREGRESSIVE FILTER.           00000478
    FLTORD IS THE ORDER OF THE FILTER IN TERMS OF THE BACK OR LAG VALUES        00000479
    + IS ASSUMED THE SAME FOR ALL NOFILT FILTERS.   PTNO IS THE CURRENT         00000480
    TIME INCREMENT.   TOTPTS IS THE FINAL TIME STEP.   FLTDEV IS THE            00000481
    THEORETICAL STANDARD DEVIATION OF RAND(L,FLTORD).                           00000482
    INITO>3 IF MONOVARIABLE FILTERS ARE REQUIRED + ARE THEN                     00000483
    DIFFERENTIATED NUMERICALLY.                                                 00000484
        >0 IF MULTIVARIABLE ARE REQUIRED.                                       00000485
    NOFILT IS THE TOTAL NO. OF FILTERS.   AQNTIL(L,I) NOTES THE TOTAL           00000486
    NUMBER OF TIMES RAND(L,I) FALLS INTO THE VARIOUS QUANTILES EACH OF          00000487
    ONE STANDARD DEVIATION IN WIDTH :                                           00000488
                                                                                00000489
:VALUE:FLTORD,L,PTNO,TOTPTS,NOFILT,INITO,FLTRIN,TRNSNT,HNO;                     00000490
:INTEGER:FLTORD,L,PTNO,TOTPTS,NOFILT,INITO,TRNSNT,HNO;                          00000491
:ARRAY:RAND,FI;                                                                 00000492
:INTEGER::ARRAY:AQNTIL;                                                         00000493
:REAL:FLTRIN,FLTDEV;                                                            00000494
:BEGIN:                                                                         00000495
    :INTEGER:I,TEMP;                                                            00000496
    :IF:PTNO>0:THEN:                                                            00000497
    :BEGIN:                                                                     00000498
        :IF:L>1:THEN:                                                           00000499
        :FOR:J&>1:STEP:1:UNTIL:NOFILT:DO:                                       00000500
        :FOR:I&>1:STEP:1:UNTIL:FLTORD<INITO<1:DO:                               00000501
            RAND(J,I/3&>0;                                                      00000502
    :END:;                                                                      00000503
     RAND(L,FLTORD<INITO<1/3&>FLTRIN;                                           00000504
    :FOR:J&>1:STEP:1:UNTIL:NOFILT:DO:                                           00000505
    :FOR:I&>1:STEP:1:UNTIL:FLTORD:DO:                                           00000506
        RAND(L,FLTORD<INITO<1/3&>RAND(L,FLTORD<INITO<1/3<FI(J,L,I/3             00000507
                                    *RAND(J,FLTORD<INITO<1-1/3;                 00000508
    :FOR:I&>1:STEP:1:UNTIL:FLTORD<INITO:DO:                                     00000509
        RAND(L,I/3&>RAND(L,I<1/3;                                               00000510
     GRADW(RAND(L,FLTORD/3,FLTDEV,AQNTIL,HNO,L,TRNSNT,PTNO);                    00000511
```

```
:END:   OF ARFILT;
:PROCEDURE:DERIVE(Z3,DZD,X,CI,PTNO,INCR);
:COMMENT:   ESTIMATES THE DERIVATIVE DZD AT THE POINT 3 LAGS BEHIND THE
    INPUT VALUE Z3.   USING A 6 ORDER SYMMETRIC FINITE DIFFERENCE
    EQUATION.   THE PROCEDURE MUST BE CALLED 7 TIMES IN ORDER TO FILL
    THE  X(/-3&3/) ARRAY ONCE.   THEREAFTER EACH SUCCESSIVE CALL PERMITS
    THE EVALUATION OF DERIVATIVE,DZD,3 TIME INCREMENTS BEHIND CURRENT
    INPUT Z3.   X(+ CI ARE EVALUATED INTERNALLY + THE EXTERNAL
    DECLARATION PREVENTS THE THE NECESSITY FOR THEIR RE EVALUATION.
    PTNO IS THE CURRENT TIME INCREMENT;
:VALUE:Z3,PTNO,INCR;
:REAL:Z3,DZD,CI,INCR;
:ARRAY:X;
:INTEGER:PTNO;
:BEGIN:
    :INTEGER:I;
    :IF:PTNO>0:THEN:   CI&>1/(60*INCR);
    :FOR:I&>-3:STEP:1:UNTIL:2:DO:
          X(/I/)&>X(/I<1/);
     X(/3/)&>Z3;
    :IF:PTNO\6:THEN:
          DZD&>CI*(X(/3/)-X(/-3/))<9*(X(/-2/)-X(/2/))<45*(X(/1/)-X(/-1/))
    :ELSE:DZD&>0;
:END:  OF DERIV;
:PROCEDURE:CORRELCY,RTAW,RAND,SUMY,BARY,L,PTNO,TRNSNT,FLTORD,TOTPTS,LAG,
                NOFILT,NRMLSE,NORMRT,TRNSI);
:COMMENT:   CALCULATES THE AUTO + CROSS CORRELATIONS OF AN INPUT RAND.
    IT TAKES THE RANDOM NUMBER RAND,ASSIGNS IT TO Y(/L,LAG/).   THE
    PRODUCTS Y(/L,LAG/)*Y(/I,J/) FOR ALL L,I,J ARE CALCULATED + ARE
    ADDED TO A CUMULATIVE STORE.   THE CORRELATIONS ARE THEN EVALUATED
    WHEN PTNO >TOTPTS + ARE OUTPUT IN THE ARRAY RTAW(/L,J,I/), WHERE L
    J ARE THE CORRELATION INDICES (J)>L), + I THE LAG INDEX.
    OTHER NON-MENTIONED EXTERNALLY DECLARED PARAMETERS ARE SO DECLARED
    TO PERMIT THEIR VALUES TO BE RECALLED ON THE NEXT RE-ENTRY INTO THE
    PROCEDURE :
:VALUE:L,PTNO,TRNSNT,FLTORD,RAND,TRNSI;
:INTEGER:L,PTNO,TRNSNT,FLTORD,TOTPTS,LAG,NOFILT,TRNSI;
:REAL:NORMRT,RAND;
:ARRAY:Y,RTAW,BARY,SUMY;
:BOOLEAN:NRMLSE;
:BEGIN:
    :INTEGER:J,I,NOPTS,CORPTS;
    :ARRAY:BARXY(/1&NOFILT,1&NOFILT/);
    :IF:PTNO)>TRNSI:THEN:
    :BEGIN:
          SUMY(/L/)&>0;
         :FOR:I&>0:STEP:1:UNTIL:LAG:DO:
         :BEGIN:
```

```
                    YE/L,I/3&>0;                                                00000569
                   :FOR:J&>1:STEP:1:UNTIL:NOFILT:DO:                            00000570
                        RTAWE/L,J,I/3&>0:                                       00000571
              :END::                                                            00000572
         :END::                                                                 00000573
         :FOR:I&>0:STEP:1:UNTIL:LAG-1:DO:                                       00000574
              YE/L,I/3&>YE/L,I<1/3:                                             00000575
          YE/L,LAG/3&>RAND:                                                     00000576
          SUMYE/L/3&>SUMYE/L/3<YE/L,LAG/3:                                      00000577
         :FOR:J&>L:STEP:-1:UNTIL:1:DO:                                          00000578
         :FOR:I&>0:STEP:1:UNTIL:LAG:DO:                                         00000579
              RTAWE/L,J,I/3&>RTAWE/L,J,I/3<YE/L,0/3*YE/J,I/3:                   00000580
         :IF:PTNO>TOTPTS:THEN:                                                  00000581
         :BEGIN:                                                                00000582
              NOPTS&>TOTPTS-TRNSNT:                                             00000583
              CORPTS&>NOPTS-LAG:                                                00000584
              BARYE/L/3&>SUMYE/L/3/NOPTS:                                       00000585
              :FOR: J&>L :STEP: -1 :UNTIL: 1 :DO:                               00000586
             :BEGIN:                                                            00000587
                  BARXYE/L,J/3&>BARYE/L/3*BARYE/J/3:                            00000588
                 :FOR:I&>0:STEP:1:UNTIL:LAG:DO:                                 00000589
                      RTAWE/L,J,I/3&>RTAWE/L,J,I/3/CORPTS-BARXYE/L,J/3:         00000590
                 :IF:^NRMLSE + L>1 :THEN:   NORMRT&>RTAWE/1,1,0/3               00000591
                 :ELSE::IF:NRMLSE:THEN:   NORMRT&>RTAWE/L,L,0/3:                00000592
                      :FOR:I&>0:STEP:1:UNTIL:LAG:DO:                            00000593
                           RTAWE/L,J,I/3&>RTAWE/L,J,I/3/NORMRT:                 00000594
              :END::                                                            00000595
         :END::                                                                 00000596
    :END:   OF CORREL :                                                         00000597
                                                                                00000598
                                                                                00000599
    :COMMENT:   START   OF   PROGRAM                                            00000600
                ******************;                                            00000601
                                                                                00000602
                                                                                00000603
         WRITETEXTE:E:AUTOREGEESSIVE FILTER COEFFICIENTS:3:3:                   00000604
         NEWLINEE 2:                                                            00000605
         SELINE1:                                                               00000606
        :FOR:L&>1:STEP:1:UNTIL:NOFILT:DO:                                       00000607
        :FOR:J&>1:STEP:1:UNTIL:NOFILT:DO:                                       00000608
        :FOR:I&>1:STEP:1:UNTIL:FLTORD:DO:                                       00000609
             FIE/J,L,I/3&>READ:                                                 00000610
     FREEIN:                                                                    00000611
         SELINE2:                                                               00000612
        :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                                   00000613
        :FOR:I&>1:STEP:1:UNTIL:FLTORD:DO:                                       00000614
             ROWTAWE/FLTNO,I/3&>READ:                                           00000615
         FREEIN:                                                                00000616
         STDEV&>1:                                                              00000617
        :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                                   00000618
             DEVATWESTDEV,FLTDEVE/FLTNO/3,ROWTAW,FI,FLTORD,FLTNO3:              00000619
        :FOR:L&>1:STEP:1:UNTIL:NOFILT:DO:                                       00000620
        :BEGIN:                                                                 00000621
             WRITETEXTE:E:FILTER NO.:3:3:                                       00000622
             PRINTEL,3,JJ:                                                      00000623
             NEWLINEE 2:                                                        00000624
            :FOR:J&>1:STEP:1:UNTIL:NOFILT:DO:                                   00000625
```

```
627             :BEGIN:                                                    00000626
628                   WRITETEXTC:E:COEFFICIENT SET NO.:3:3;                00000627
629                   PRINTEJ,3,03;                                        00000628
630                   NEWLINEC13;                                          00000629
631                  :FOR:I&>1:STEP:1:UNTIL:FLTORD:DO;                     00000630
632                  :BEGIN:                                               00000631
633                        PRINTEFIE/J,L,I/3,0,12];                        00000632
634                        NEWLINEC13;                                     00000633
635                  :END:;                                                00000634
636             :END:;                                                     00000635
637          :END:;                                                        00000636
638       NEWLINEC43;                                                      00000637
639       WRITETEXTC:E:FLTNO   R.NO.   MULTIPLIER        SEED:E:2C:3:::3:3; 00000638
640      :FOR:J&>1:STEP:1:UNTIL:NOFILT:DO;                                 00000639
641      :FOR:I&>1:STEP:1:UNTIL:NORNG:DO;                                  00000640
642      :BEGIN:                                                           00000641
643           MULTE/J,I/J&>MULTOE/J,I/J&>READ;                             00000642
644           SEEDE/J,I/J&>SEEDOE/J,I/J&>READ;                             00000643
645           PRINTEJ,4,01;                                                00000644
646           PRINTCI,4,03;                                                00000645
647           PRINTEMULTE/J,I/3,9,03;                                      00000646
648           PRINTESEEDE/J,I/3,9,03;                                      00000647
649           NEWLINEC13;                                                  00000648
650      :END:;                                                            00000649
651       CORCOFE/1/J&>1;                                                  00000650
652      :IF:INTORD>5:THEN:                                                00000651
653      :BEGIN:                                                           00000652
654           CORCOFE/0/J&>95/288;                                         00000653
655           CORCOFE/2/J&>25/24;                                          00000654
656           CORCOFE/3/J&>35/72;                                          00000655
657           CORCOFE/4/J&>5/48;                                           00000656
658           CORCOFE/5/J&>1/120;                                          00000657
659      :END:                                                             00000658
660      :ELSE::IF:INTORD>4:THEN:                                          00000659
661      :BEGIN:                                                           00000660
662           CORCOFE/0/J&>251/720;                                        00000661
663           CORCOFE/2/J&>11/12;                                          00000662
664           CORCOFE/3/J&>1/3;                                            00000663
665           CORCOFE/4/J&>1/24;                                           00000664
666      :END:                                                             00000665
667      :ELSE::IF:INTORD>6:THEN:                                          00000666
668      :BEGIN:                                                           00000667
669           CORCOFE/0/J&>19087/60480;                                    00000668
670           CORCOFE/2/J&>137/120;                                        00000669
671           CORCOFE/3/J&>5/8;                                            00000670
672           CORCOFE/4/J&>17/96;                                          00000671
673           CORCOFE/5/J&>1/140;                                          00000672
674           CORCOFE/6/J&>1/720;                                          00000673
675      :END:;                                                            00000674
676       MAXFRQ&>0;                                                       00000675
677      :FOR:I&>1:STEP:1:UNTIL:NOFILT:DO:                                 00000676
678      :BEGIN:                                                           00000677
679           NFREQ&>READ;                                                 00000678
680           ZETA&>READ;                                                  00000679
681           WN&>NFREQ*8*ARCTANC13;                                       00000680
682           PARMBE/I/J&>WN*WN;                                           00000681
683           PARMAE/I/J&>2*ZETA*WN;                                       00000682
```

ESTABLISH INTEGRATION METHOD PREDICTOR COEFFICIENTS

ESTABLISH PHYSICAL PARAMETERS OF MODEL

```
684         :IF:MAXFRQ)NFREQ:THEN:    MAXFRQ&>NFREQ;                                00000683
685       :END:;                                                                    00000684
686       :COMMENT:    1/2 MINIMUM DT BECAUSE OF NOTERP;                            00000685
687        DTMAX&>0.025/MAXFRQ;                                                     00000686
688        DTMXZ0&>20*DTMAX;                                                        00000687
689       :FOR:TST&>1:STEP:1:UNTIL:NOTEST:DO:                                       00000688
690       :BEGIN:                                                                   00000689
691            TOTTIM&>0;                                                          00000690
692            DIST&>0;                                                            00000691
693       :COMMENT:    INPUT INITIAL TRAVERSE VELOCITY,DXT + ACCELERATION,D2XTG,   00000692
694            IN UNITS OF F.P.S. + G RESP.  :                                     00000693
695            DXT&>DXTI&>DXTO&>READ;                                              00000694
696            D2XTG&>READ;                                                        00000695
697            D2XTN#&>D2XTG*32.17535465;                                          00000696
698            D2XT&>D2XTI&>D2XTN#;                                                00000697
699            NEWLINE(4);                                                         00000698
700            WRITETEXT(:C: INITIAL VELOCITY      INITIAL ACCELATION:E:C 85:I:F   00000699
701       *P.S.          G    OR    F.P.S*S.:C:2C:I::I:);                         00000700
702       PRINT(DXT,0,12);                                                         00000701
703            PRINT(D2XTG,2,6);                                                   00000702
704            PRINT(D2XTN#,0,12);                                                 00000703
705            PAPERTHROW;                                                         00000704
706            COUNT&>0;                                                           00000705
707                 :FOR:PTNO&>0:STEP:1:UNTIL:TOTPTI:DO:                          00000706
708                :BEGIN:                                                       00000707
709                 :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                        00000708
710                 :BEGIN:                                                      00000709
711                      DFLTNO&>2*FLTNO;                                        00000710
712                      DFNOM1&>DFLTNO-1;                                       00000711
713                      POLARSEED,MULT,FLTNO,PTNO,PQNTIL,FAIL,FLTRINC/FLTNO/1,  00000712
714                           TOTPTS,REDUND,TRNSNT,HNO,STDEV);                   00000713
715                      ARFILT(FLTRINC/FLTNO/1,Z,FI,FLTORD,DFNOM1,PTNO,AQNTIL,  00000714
716                            FLTDEVC/FLTNO/1,TOTPTS,NOFILT,INITO,TRNSNT,HNO);  00000715
717                      ZC/DFLTNO,FLTORD-1/1&>ZC/DFLTNO,FLTORD/1;                00000716
718                      DERIVCZC/DFNOM1,FLTORD<INITO/1,ZC/DFLTNO,FLTORD/1,LX,C1, 00000717
719                           PTNO,STPLTH);                                      00000718
720                 :END:   OF FLTNO;                                            00000719
721                 :IF:PTNO)TRNSNT:THEN:                                        00000720
722                 :BEGIN:                                                      00000721
723                      :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                   00000722
724                       MSZXC/FLTNO/1&>MSDZXC/FLTNO/1&>MSZIXC/FLTNO/1&>        00000723
725                       MSDZIXC/FLTNO/1&>MSYTC/FLTNO/1&>MSDYTC/FLTNO/1&>MSYZT  00000724
726                       C/FLTNO/1&>MSYC/FLTNO,TRNS1/1&>MSDYC/FLTNO,TRNS1/1&>MSYZ 00000725
727                       C/FLTNO,TRNS1/1&>MSYC/FLTNO,TRNSNT/1&>MSDYC/FLTNO,TRNSNT/00000726
728                       1&>MSYZC/FLTNO,TRNSNT/1&>YTO1&>YTO2&>YT1&>DYTO1&>DYTO2&> 00000727
729                       DYT1&>0;                                               00000728
730                      :GOTO:LAB1;                                             00000729
731                 :END:;                                                       00000730
732                 :IF:COLATE:THEN:                                             00000731
733                 :FOR:L&>DFNOM1:STEP:1:UNTIL:DFLTNO:DO:                       00000732
734                       CORRELCRAND,RTAWZ,ZC/L,FLTORD/1,SUMZ,BARZ,L,PTNO,TRNSNT, 00000733
735                            FLTORD,TOTPTI,LAG,DUOFLT,NRMLSE,NORMRT,TRNS1);     00000734
736       :COMMENT:    INPUT ACCELARATION D2XTN# EVERY STPLTH :                    00000735
737                  D2XTN#&>D2XTN#;                                              00000736
738                  TIMSUM(STPLTH,DXTI,D2XTN#,DT);                               00000737
739                 :IF:DT\DTMAX:THEN:    NOTERP&>ENTIER(DT/DTMAX);              00000738
740                 :ELSE:    NOTERP&>1;                                         00000739
```

GENERATE ROAD PROFILE DISPLACEMENT & GRADIENT AT CURRENT POINT NUMBER

INITIALISE PARAMETERS

ESTABLISH GENERATED ROAD PROFILE CORRELATIONS

ESTABLISH INCREMENTAL TIME STEP & WHETHER INTERPOLATION OF ROAD PROFILE IS REQUIRED.

```
741          DX&>STPLTH/NOTERP;                                                    00000740
742         :IF:DXTI>0 + D2XTN#>0:THEN:   :GOTO:SKIP;                              00000741
743         :FOR:TERP&>0:STEP:1:UNTIL:NOTERP-1:DO:                                 00000742
744         :BEGIN:                                                                00000743
745              :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                             00000744
746              :BEGIN:                                                           00000745
747                   DFLTNO&>2#FLTNO:                                             00000746
748                   DFNOMI&>DFLTNO-1:                                            00000747
749                   PARMAZE/FLTNO/J&>PARMAE/FLTNO/J#DXTI:                        00000748
750                   PARMAYE/FLTNO/J&>PARMAZE/FLTNO/J<D2XTI:                      00000749
751                  :IF:PTNO>TRNSNT:THEN:                                         00000750
752                  :BEGIN:                                                       00000751
753                       ZIE/DFNOMI/J&>ZE/DFNOMI,FLTORD/J:                        00000752
754                       ZIE/DFLTNO/J&>ZE/DFLTNO,FLTORD/J:                        00000753
755                       YE/FLTNO,0/J&>ZE/DFNOMI,FLTORD/J:                        00000754
756                       DYE/FLTNO,0/J&>0:                                        00000755
757                       DYE/FLTNO,0/J&>0:                                        00000756
758                       :GOTO:PRNTI:                                             00000757
759                  :END:                                                         00000758
760                  :ELSE::IF: PTNO>TRNSI + TERP>0 + FLTNO>1 + DXTO>0             00000759
761                                                              :THEN:            00000760
762                  :BEGIN:                                                       00000761
763                       :FOR:FLTST&>1:STEP:1:UNTIL:NOFILI:DO:                    00000762
764                       :BEGIN:                                                  00000763
765                            SDFLTN&>2#FLTST:                                    00000764
766                            SDNOMI&>SDFLTN-1:                                   00000765
767                           :FOR:J&>1:STEP:1:UNTIL:2:DO:                         00000766
768                                TLYE/FLTST,J/J&>PARMBE/FLTST/J#                 00000767
769                                   EZIE/SDFLTN-2<J/J-TLYE/FLTST,J-1/J           00000768
770                                        J/D2XTI:                                00000769
771                            FACT&>1:                                            00000770
772                           :FOR:J&>1:STEP:1:UNTIL:2:DO:                         00000771
773                           :BEGIN:                                              00000772
774                                FACT&>FACT#DX/J:                                00000773
775                                YE/FLTST,0/J&>TLYE/FLTST,J/J#FACT<              00000774
776                                                        YE/FLTST,0/J:           00000775
777                           :END::                                               00000776
778                            DYE/FLTST,0/J&>DYE/FLTST,0/J<DX#TLYE/FLTST          00000777
779                                                              ,2/J:              00000778
780                       :END:   OF   FLTST;                                      00000779
781                        DXTI&>SQRTE2#D2XTI#DXJ:                                 00000780
782                        DT&>SQRTE2#DX/D2XTIJ:                                   00000781
783                        TOTTIM&>DT:                                             00000782
784                        DTTERP&>DT:                                             00000783
785                        DIST&>DX:                                               00000784
786                  :GOTO:LMNSQ:                                                  00000785
787                  :END:                                                         00000786
788                  :ELSE::IF:PTNO>TRNSI+ETERP>1+DXTO>0:OR:TERP>0+DXTO^>0         00000787
789                                                          J:THEN:               00000788
790                  :BEGIN:                                                       00000789
791                       :IF:FLTNO^>1:THEN:   :GOTO:PRNTI:                        00000790
792                       :IF:DXTO^>0:THEN:                                        00000791
793                       :BEGIN:                                                  00000792
794                            TOTTIM&>0:                                          00000793
795                            DIST&>0:                                            00000794
796                       :END:                                                    00000795
797                       :ELSE:                                                   00000796
```

TEST FOR SINGULARITY

ADVANCE REALISATION BY ONE SPATIAL INCREMENT FOR EACH GENERATED FLTNO

```
798                 :BEGIN:                                                 00000797
799                       TOTTIM&>DT:                                       00000798
800                       DIST&>DX:                                         00000799
801                 :END::                                                  00000800
802                 :IF:INTORD-1\NOTERP:THEN:                               00000801
803                       SNOTRP&>INTORD-1:                                 00000802
804                 :ELSE:    SNOTRP&>NOTERP:                               00000803
805                  SDX&>STPLTH/SNOTRP:                                    00000804
806                  D2XTI&>D2XT;                                           00000805
807                 :FOR:J&>1:STEP:1:UNTIL:INTORD-1:DO:                     00000806
808                 :BEGIN:                                                 00000807
809                       INTERP[D2XT,D2XTNW,D2XTI,J,SNOTRP]:               00000808
810                       DXTI&>DXTI+DXTI<Z+D2XTI+SDX:                      00000809
811                       SPARM[J/]&>SDX/DXTI:                              00000810
812                       DXTI&>SQRT[DXTI]:                                 00000811
813                      :FOR:FLTST&>1:STEP:1:UNTIL:NOFILT:DO:              00000812
814                      :BEGIN:                                            00000813
815                            SPRMAZ[/FLTST,J/]&>PARMA[/FLTST/]*           00000814
816                                                          DXTI:00000815
817                          SPRMAY[/FLTST,J/]&>SPRMAZ[/FLTST,J/]<00000816
818                                                          DZXTI:00000817
819                       :END::                                            00000818
820                 :END::                                                  00000819
821                  STARTEY,DY,ZI,TY,TDY,CORCOF,SPRMAY,SPRMAZ,             00000820
822                        PARMB,YT1,YTO1,                                  00000821
823                        YTO2,DYT1,DYTO1,DYTO2,FLTNO,INTORD,SDX,Z,00000822
824                        FLTORD,SNOTRP,SPARM,FY1,FDY1]:                   00000823
825                  D2XTI&>D2XT;                                           00000824
826                 :IF:DXTO>0:THEN:    DXTI&>D2XTI+DT                      00000825
827                 :ELSE:    DXTI&>DXTO:                                  00000826
828                 :GOTO:LMNSQ;                                           00000827
829           :END::                                                        00000828
830           :IF:D2XT>D2XTNW:THEN:    D2XTI&>D2XTNW                        00000829
831           :ELSE:    INTERP[D2XT,D2XTNW,D2XTI,TERP,NOTERP]:             00000830
832           :COMMENT:    BY LINEARLY INTERPOLATING THE ACCELERAT-         00000831
833            ION,DISCONTINUITIES IN VELOCITY + ACCELERATION ARE           00000832
834           MINIMISED:                                                    00000833
835            DXTI&>SQRT[DXTI+DXTI<2+D2XTI+DX]:                            00000834
836            TIMSUM[DX,DXTI,D2XTI,DT]:                                    00000835
837            PARM&>DX/[DXTI+DXTI]:                                        00000836
838            TOTTIM&>TOTTIM<DT:                                           00000837
839            DIST&>DIST<DX:                                               00000838
840           :IF:TERP>0:THEN:                                              00000839
841           :BEGIN:                                                       00000840
842                 MNSQREE Z[/DFNOM1,FLTORD-1/],STPLTH,MSZX[/FLTNO/00000841
843                                                    ],DIST]:      00000842
844                 MNSQREE Z[/DFLTNO,FLTORD-1/],STPLTH,MSDZX[/FLTNO00000843
845                                                   /],DIST]:      00000844
846           :END::                                                        00000845
847            MNSQREE Z I[/DFNOM1/],DX,MSZIX[/FLTNO/],DIST]:               00000846
848            MNSQREE Z I[/DFLTNO/],DX,MSDZIX[/FLTNO/],DIST]:              00000847
849           :IF:PTNO>TRNS1 + TERP>1:THEN:                                 00000848
850           :BEGIN:                                                       00000849
851                 ZI[/DFNOM1/]&>ZI[/DFNOM1,FLTORD-1/];                    00000850
852                 ZI[/DFLTNO/]&>ZI[/DFLTNO,FLTORD-1/];                    00000851
853                 YI[/FLTNO,0/]&>ZI[/DFNOM1,FLTORD-1/];                   00000852
854                 DYI[/FLTNO,0/]&>0;                                      00000853
```

```
855                         :END:                                                00000854
856                         :ELSE:                                               00000855
857                         :BEGIN:                                              00000856
858                          INTERPCZE/DFNOM1,FLTORD-1/3,ZE/DFNOM1,FLTORD/3,     00000857
859                               ZIE/DFNOM1/3,TERP,NOTERP3:                     00000858
860                          INTERPCZE/DFLTNO,FLTORD-1/3,ZE/DFLTNO,FLTORD/3,     00000859
861                               ZIE/DFLTNO/3,TERP,NOTERP3:                     00000860
862                         :END::                                               00000861
863                          DTTERP&>DTTERP<DT:                                  00000862
864                          STEPLY,DY,ZI,TY,TDY,CORCOF,PARMAYE/FLTNO/3,         00000863
865                               PARMAZE/FLTNO/3,PARMBE/FLTNO/3,YT1,YT01,YT02,  00000864
866                               DYT1,DYT01,DYT02,FLTNO,INTORD,DX,PARM,FY1,FDY1 00000865
867                                                                           3;00000866
868                          FY1&>FY1/DX:                                        00000867
869                          FDY1&>FDY1/DX:                                      00000868
870       LMNSQ6                                                                 00000869
871                         :IF:TERP>0:THEN:                                     00000870
872                         :BEGIN:                                              00000871
873                               BIGYE/FLTNO,PTNO/3&>YE/FLTNO,0/3:              00000872
874                               BIGDYE/FLTNO,PTNO/3&>DYE/FLTNO,0/3:            00000873
875                               BIGD2YE/FLTNO,PTNO/3&>FDY1:                    00000874
876                               BIGZE/FLTNO,PTNO/3&>ZIE/DFNOM1/3:              00000875
877                               BIGDZE/FLTNO,PTNO/3&>ZIE/DFLTNO/3:             00000876
878                               DTSHFTCBLKDT,DTTERP,BLOCK,TRNS2,PTNO,BLKTIM,   00000877
879                                    NWBTIM3:                                  00000878
880                               BKMNSQCYE/FLTNO,0/3,DTTERP,MSYTE/FLTNO/3,VALY, 00000879
881                                    BLKTIM,NWBTIM,BLOCK,TRNS2,PTNO,FLTNO3:    00000880
882                               BKMNSQCDYE/FLTNO,0/3,DTTERP,MSDYTE/FLTNO/3,    00000881
883                                    VALDY,BLKTIM,NWBTIM,BLOCK,TRNS2,PTNO,     00000882
884                                    FLTNO3:                                   00000883
885                               BKMNSQCYE/FLTNO,0/3-ZIE/DFNOM1/3,DTTERP,MSYZT  00000884
886                                    E/FLTNO/3,VALYZ,BLKTIM,NWBTIM,BLOCK,      00000885
887                                    TRNS2,PTNO,FLTNO3:                        00000886
888                               BKMNSQCZIE/DFNOM1/3,DTTERP,MSZITE/FLTNO/3,     00000887
889                                    VALZI,BLKTIM,NWBTIM,BLOCK,TRNS2,PTNO,     00000888
890                                    FLTNO3:                                   00000889
891                               BKMNSQCZIE/DFLTNO/3,DTTERP,MSDZITE/FLTNO/3,    00000890
892                                    VALDZI,BLKTIM,NWBTIM,BLOCK,TRNS2,PTNO,    00000891
893                                    FLTNO3:                                   00000892
894                         :END::                                               00000893
895                          MSYE/FLTNO,PTNO/3&>MSYTE/FLTNO/3:                   00000894
896                          MSDYE/FLTNO,PTNO/3&>MSDYTE/FLTNO/3:                 00000895
897                          MSYZE/FLTNO,PTNO/3&>MSYZTE/FLTNO/3:                 00000896
898                          MSZIE/FLTNO,PTNO/3&>MSZITE/FLTNO/3:                 00000897
899                          MSDZIE/FLTNO,PTNO/3&>MSDZITE/FLTNO/3:               00000898
900       PRNT16                                                                 00000899
901                         :IF:TERP>0 + PTNO\>TRNSNT:OR:PTNO>TRNS1 + TERP>1     00000900
902                                                                   :THEN:     00000901
903               :BEGIN:                                                        00000902
904                         :IF:PTNO\>TRNSNT+PTNO)TRNSNT<25 :OR: PTNO\>TOTPT1-25 00000903
905                                                                   :THEN:     00000904
906                    :BEGIN:                                                   00000905
907                          DTTERP&>0:                                          00000906
908                          COUNT&>COUNT<1:                                     00000907
909                         :IF:COUNT>1:THEN:                                    00000908
910                         :BEGIN:                                              00000909
911                               WRITETEXTE:E:  INCR FILT CURRENT TIME     DISPL00000910
```

```
ACEMENT    VELOCITY        DISPLACEMENT    VELOCITY            TRAVERSE 00000911
   DISTANCE    :C:C 3S:3:NO.  NO.     ESECS3        INPUT          INPUT 00000912
          OUTPUT          OUTPUT          VELCF/S3    CFEET3:3:3: 00000913
                    :END::                                              00000914
                     NEWLINEC13:                                        00000915
                     PRINTCPTNO-TRNSNT,5,03:                            00000916
                     PRINTCFLTNO,2,03:                                  00000917
                     PRINTCTOTTIM,6,33:                                 00000918
                     PRINTCZIC/DFNOM1/3,0,73:                           00000919
                     PRINTCZIC/DFLTNO/3,0,73:                           00000920
                     PRINTCYC/FLTNO,0/3,0,73:                           00000921
                     PRINTCDYC/FLTNO,0/3,0,73:                          00000922
                     PRINTCDXTI,7,53:                                   00000923
                     PRINTCDIST,7,23:                                   00000924
                     NEWLINEC13:                                        00000925
                     PRINTCFDYF,0,73:                                   00000926
                     PRINTCMSYC/FLTNO,PTNO/3,0,23:                      00000927
                     PRINTCMSDYC/FLTNO,PTNO/3,0,23:                     00000928
                     PRINTCMSZIC/FLTNO,PTNO/3,0,23:                     00000929
                     PRINTCMSDZIC/FLTNO,PTNO/3,0,23:                    00000930
                     PRINTCYTO1,0,13:                                   00000931
                     PRINTCYTO2,0,13:                                   00000932
                     PRINTCYT1,0,13:                                    00000933
                     PRINTCDYTO1,0,13:                                  00000934
                     PRINTCDYTO2,0,13:                                  00000935
                     PRINTCDYT1,0,13:                                   00000936
                    :IF:COUNT>NPP :OR: PTNO>TOTPTS<1+FLTNO>NOFILT:THEN: 00000937
                    :BEGIN:                                             00000938
                     NEWLINEC23:                                        00000939
                     WRITETEXTC:C: OUTPUT          MSY         MSDY     00000940
     MSZI       MSDZI     YTO1      YTO2      YT1      DYTO1     DYTO2  00000941
    DYT1:C:2C:3: ACCELERATION *   INSTANTANEOUS MEAN SQUARE RESPONSES  * 00000942
    CONVERGENCE   TESTS   ON   NUMERICAL   INTEGRATION:3:3:              00000943
                          PAPERTHROW:                                   00000944
                          COUNT&>0:                                     00000945
                    :END::                                              00000946
                :END:   OF PRINTING:                                    00000947
                :END:   OF FLTNO :                                      00000948
          :END::                                                        00000949
          :END:   OF TERP :                                             00000950
SKIP&                                                                   00000951
           D2XT&>D2XTNW:                                                00000952
LAB1&                                                                   00000953
      :END:   OF PTNO ;                                                 00000954
       WRITETEXTC:C::C:33S:3:AVERAGE  MEAN  SQUARE  VALUES  OVER  TOTAL R00000955
ECORD  LENGTH:C:C 2:S:3:INPUT  MEAN  SQUARES  W.R.T.  DISTANCE          00000956
*    RESPONSE  MEAN  SQUARES  W.R.T.  TIME:C:C:3:FLTNO  DISP. INPUT     00000957
 VEL. INPUT     DISP. INPUT      VEL. INPUT  * DISP.RESPONSE   VEL. RESP00000958
ONSE  DISP. RESPONSE:C:C 3S:3:C INTERPOLATED3  INTERPOLATED3 *:C:3S:3:C00000959
RELATIVE3:C:C 1CS:3:MSZX          MSDZX          MSZIX          MSDZIX00000960
        *     MSYT          MSDYT          MSYZT:3:3:                 00000961
       NEWLINEC23:                                                      00000962
      :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                             00000963
      :BEGIN:                                                           00000964
           PRINTCFLTNO,2,03:                                            00000965
           PRINTCMSZXC/FLTNO/3,0,73:                                    00000966
           PRINTCMSDZXC/FLTNO/3,0,73:                                   00000967
```

END OF FLTNO Loop

```
                PRINTCMSZIXC/FLTNO/3,0,7];                                      00000968
                PRINTCMSDZIXC/FLTNO/3,0,7];                                     00000969
                WRITETEXTC:C:*:3:3];                                            00000970
                PRINTCMSYTC/FLTNO/3,0,7];                                       00000971
                PRINTCMSDYTC/FLTNO/3,0,7];                                      00000972
                PRINTCMSYZTC/FLTNO/3,0,7];                                      00000973
                NEWLINEC1];                                                     00000974
          :END:;                                                                00000975
           REDUND&>REDUND/TOTPTS;                                               00000976
           WRITETEXTC:C:POLAR GENERATOR - DEGREE OF REDUNDANCY >:3:3];          00000977
           PRINTCREDUND,0,12];                                                  00000978
           WRITETEXTC:C:STANDARD DEVIATION >:3:3];                              00000979
           PRINTESTDEV,0,12];                                                   00000980
           NEWLINEC1];                                                          00000981
           WRITETEXTC:C:FREQUENCY OF OCCURENCE - IN VARIOUS CLASS INTERVALS     00000982
      :C:2C:3:*  QUANTILE:E:C:3:  * CATEGORY - IN STANDARD DEVIATION INCREMENT00000983
      S:C:C 4S:3:*:C:C 6S:3:*   :3:3];                                         00000984
          :FOR:I&>-HNO:STEP:1:UNTIL:HNO:DO:                                     00000985
               :IF:I^>0:THEN:   PRINTCI,5,0];                                   00000986
           NEWLINEC1];                                                          00000987
           WRITETEXTC:C:FILTER  *:C:C:3:NUMBER  *:3:3];                         00000988
          :FOR:I&>1-HNO:STEP:1:UNTIL:HNO:DO:                                    00000989
                WRITETEXTC:C:     ***:3:3];                                     00000990
           NEWLINEC1];                                                          00000991
          :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                                 00000992
          :BEGIN:                                                               00000993
                PRINTCFLTNO,3,0];                                               00000994
                WRITETEXTC:C:  #  :3:3];                                        00000995
               :FOR:I&>-HNO:STEP:1:UNTIL:HNO-1:DO:                              00000996
                     PRINTCPQNTILC/FLTNO,I/3,5,0];                              00000997
                NEWLINEC1];                                                     00000998
          :END:;                                                                00000999
           NEWLINEC2];                                                          00001000
           WRITETEXTC:C: AUTOREGRESSIVE FILTERS - STANDARD DEVIATIONS:3:3];     00001001
           NEWLINEC1];                                                          00001002
          :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                                 00001003
          :BEGIN:                                                               00001004
                PRINTCFLTNO,3,0];                                               00001005
                PRINTCFLTDEVC/FLTNO/3,0,12];                                    00001006
                NEWLINEC1];                                                     00001007
          :END:;                                                                00001008
           NEWLINEC1];                                                          00001009
           WRITETEXTC:C:FREQUENCY OF OCCURENCE - IN VARIOUS CLASS INTERVALS     00001010
      :C:2C:3:*  QUANTILE:E:C:3:  * CATEGORY - IN STANDARD DEVIATION INCREMENT00001011
      S:C:C 4S:3:*:C:C 6S:3:*   :3:3];                                         00001012
          :FOR:I&>-HNO:STEP:1:UNTIL:HNO:DO:                                     00001013
               :IF:I^>0:THEN:   PRINTCI,5,0];                                   00001014
           NEWLINEC1];                                                          00001015
           WRITETEXTC:C:FILTER  *:C:C:3:NUMBER  *:3:3];                         00001016
          :FOR:I&>1-HNO:STEP:1:UNTIL:HNO:DO:                                    00001017
                WRITETEXTC:C:     ***:3:3];                                     00001018
           NEWLINEC1];                                                          00001019
          :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                                 00001020
          :BEGIN:                                                               00001021
                PRINTCFLTNO,3,0];                                               00001022
                WRITETEXTC:C:  #  :3:3];                                        00001023
               :FOR:I&>-HNO:STEP:1:UNTIL:HNO-1:DO:                              00001024
```

```
                    PRINTEAQNTILE/FLTNO,I/3,5,03;                                   00001025
                NEWLINEC13;                                                         00001026
            :END:;                                                                  00001027
             NEWLINEC23;                                                            00001028
             #RITETEXTC:E: CURRENT MULTIPLIERS + SEEDS ARE:C:2C 9S:3:J       I      00001029
          MULT     SEED:3:3;                                                        00001030
             NEWLINEC13;                                                            00001031
            :FOR:J&>1:STEP:1:UNTIL:NOFILT:DO:                                       00001032
            :FOR:I&>1:STEP:1:UNTIL:NORNG:DO:                                        00001033
            :BEGIN:                                                                 00001034
                PRINTCJ,5,03;                                                       00001035
                PRINTCI,5,03;                                                       00001036
                PRINTCMULTE/J,I/3,6,03;                                             00001037
                PRINTCSEEDC/J,I/3,6,03;                                             00001038
                NEWLINEC13;                                                         00001039
            :END:;                                                                  00001040
             #RITETEXTC:C:REALISED INPUT CORRELATIONS - CNORMALISING FACTOR, NO00001041
      RMRT > :3:3:                                                                  00001042
             PRINTCNORMRT,0,12];                                                    00001043
             #RITETEXTC:C:3:E:C 3S:3:LAG:3:3;                                       00001044
             PAPERTHROW;                                                            00001045
            :IF:COLATE:THEN:                                                        00001046
            :BEGIN:                                                                 00001047
             #RITETEXTC:C:INPUT CORRELATIONS ARE:C:C 3S:3:LAG:3:3;                  00001048
            :FOR:J&>1:STEP:1:UNTIL:DUOFLT:DO:                                       00001049
            :FOR:I&>1:STEP:1:UNTIL:J:DO:                                            00001050
            :BEGIN:                                                                 00001051
                PRINTCJ,3,03;                                                       00001052
                PRINTCI,3,03;                                                       00001053
                SPACEC93;                                                           00001054
            :END:;                                                                  00001055
             NEWLINEC13;                                                            00001056
             #RITETEXTC:C:   ***:3:3;                                               00001057
            :FOR:J&>1:STEP:1:UNTIL:DUOFLT:DO:                                       00001058
            :FOR:I&>1:STEP:1:UNTIL:J:DO:                                            00001059
                #RITETEXTC:C:   ***   ***          :3:3;                          00001060
             NEWLINEC23;                                                            00001061
            :FOR:I&>0:STEP:1:UNTIL:LAG:DO:                                          00001062
             :BEGIN:                                                                00001063
                PRINTCI,3,03;                                                       00001064
               :FOR:L&>1:STEP:1:UNTIL:DUOFLT:DO:                                    00001065
               :FOR:J&>1:STEP:1:UNTIL:L:DO:                                         00001066
                    PRINTCRTA#ZC/L,J,I/3,0,12];                                     00001067
                NEWLINEC13;                                                         00001068
            :END:;                                                                  00001069
             #RITETEXTC:C:CORRELATION MEAN INPUT VALUES -CDEVIATION FROM ZERO ME00001070
    AN3:C:C:3: FLTNO       BARZCJ3            BARZCJ<13:C:C:3:3:3:3;               00001071
          L&>0:                                                                     00001072
           I&>1:                                                                    00001073
          :FOR:J&>1:STEP:2:UNTIL:DUOFLT:DO:                                         00001074
          :BEGIN:                                                                   00001075
               L&>L<1;                                                              00001076
               I&>I<1;                                                              00001077
              :IF:I>2:THEN:                                                         00001078
              :BEGIN:                                                               00001079
                   NEWLINEC13;                                                      00001080
                   PRINTCL,5,03;                                                    00001081
```

```
1083                    I&>0;                                          00001082
1084                :END:;                                             00001083
1085                 PRINTCBARZE/J/3,0,12];                            00001084
1086                 PRINTCBARZE/J<I/3,0,12];                          00001085
1087           :END:;                                                  00001086
1088            NEWLINEC1];                                            00001087
1089           :END:   OF COLATE;                                      00001088
1090           :COMMENT: SELOUT REALISATION STORES;                    00001089
1091            SELOUTC2];                                             00001090
1092            PRINTCDXTG,0,6];                                       00001091
1093            PRINTCD2XTG,0,6];                                      00001092
1094           :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                   00001093
1095           :FOR:I&>1:STEP:1:UNTIL:NORNG:DO:                        00001094
1096           :BEGIN:                                                 00001095
1097                 PRINTCMULTDC/FLTNO,I/3,0,6];                      00001096
1098                 PRINTCSEEDOC/FLTNO,I/3,0,6];                      00001097
1099           :END:;                                                  00001098
1100           :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                   00001099
1101           :FOR:PTNO&>TRNS1:STEP:1:UNTIL:TOTPT1:DO:                00001100
1102                 PRINTCBIGZE/FLTNO,PTNO/3,0,6];                    00001101
1103           :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                   00001102
1104           :FOR:PTNO&>TRNS1:STEP:1:UNTIL:TOTPT1:DO:                00001103
1105                 PRINTCBIGDZC/FLTNO,PTNO/3,0,6];                   00001104
1106           :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                   00001105
1107           :FOR:PTNO&>TRNS1:STEP:1:UNTIL:TOTPT1:DO:                00001106
1108                 PRINTCBIGYC/FLTNO,PTNO/3,0,6];                    00001107
1109           :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                   00001108
1110           :FOR:PTNO&>TRNS1:STEP:1:UNTIL:TOTPT1:DO:                00001109
1111                 PRINTCBIGDYC/FLTNO,PTNO/3,0,6];                   00001110
1112           :FOR:FLTNO&>1:STEP:1:UNTIL:NOFILT:DO:                   00001111
1113           :FOR:PTNO&>TRNS1:STEP:1:UNTIL:TOTPT1:DO:                00001112
1114                 PRINTCBIGDZYC/FLTNO,PTNO/3,0,6];                  00001113
1115            PRINTC9.999999:70,0,6];                                00001114
1116            FREEOUT;                                               00001115
1117      :END:   OF TST;                                              00001116
1118      :END:;                                                       00001117
1119      FAIL&                                                        00001118
1120      :END:                                                        00001119
```

OUTPUT TO DISK FILE (FOR SUBSEQUENT PROCESSING)

DISP. / GRAD. — ROAD PROFILE DISPLACEMENT & GRADIENT

DISP. / DERIV. / 2ND DERIV. — VEHICLE RESPONSE DISPLACEMENT, 1ST + 2ND DERIVATIVE

END OF RECORD MARKER

```
$PRT UNIFIL.MULTIVAR
```

DSYS2 - Program Input

NOTEST (Integer)
Defines the number of generated independent realisations required (N.B. MULT and SEED still only need be input for a single test. Each realisation assumes the next consecutive SEED values, while maintaining original MULT values.)

NORNG (Integer)
Defines the number of independent random number generators required. (For the polar generator used by the program this should be set to "3".)

NOFILT (Integer)
Defines the number of autoregressive filters required. (For single track realisations this should be set to "1".)

FLTORD (Integer)
Defines the order (i.e. highest lag value) of the autoregressive filter.

NOPTS (Integer)
Defines the number of points on each realisation record, excluding those used during the initial transient of the autoregressive filter.

INTORD (Integer)
Defines the order fo the Predictor explicit integration formula. It is recommended, but not essential that this variable be set to "5", although "4" or "6" are also permissable.

STPLTH (Real)
Defines the incremental spatial step length, h(x), which should be consistent with the lag increments of the autoregressive filter.

HNO (Integer)
Defines the number of quantile segments either side of the mean value thus enabling a measure of quality control on the assessment of probability distribution - normally set to "4".

BLOCK (Integer)
Defines the number of consecutive realisation points along the record over which the mean square values should be averaged.

BLOCK (Integer) - continued
As this smoothing technique proved of little value for non stationary realisations, it is recommended that BLOCK be set to "1".

MEMBER (Integer)
Defines a now redundant variable and should therefore be deleted from the program.

FI[1:NOFILT,1:NOFILT,1:FLTORD] (Array)
Defines the autoregressive filter coefficients used by the road generator.
FI[J,L,I] is input in a nested format, with "I" the innermost loop, followed by "L", which in turn has "J" as the outermost loop variable. More concisely this can be expressed as follows
(J=1:NOFILT(L=1:NOFILT(I=1:FLTORD))).
If NOFILT = 1, FI is effectively reduced to a one dimensional array and the coefficients of a single AR filter should be input.
If NOFILT = 1, FI becomes a tri-dimensional array. Consequently, to obtain more than one independent autoregressive filter, the FI[J,L,I] input should be assigned to zero when $J \neq L$.

ROWTAW[1:NOFILT,1:FLTORD] (Array)
Defines the theoretical displacement road profile correlations.
ROWTAW[FLTNO,I] is input according to the nesting format
(FLTNO=1:NOFLT(I=1;FLTORD)).
For a single autoregressive filter
FLTNO = 1.
For more than one autoregressive filter
FLTNO > 1.

MULT[1:NOFLT,1:NORNG],SEED[1:NOFLT,1:NORNG] (Integer Arrays)
Defines the primative root values of the variables MULT and the starting values of SEED. These quantities are used by the random number generators, and are input as a MULT, SEED pair such the nesting format for MULT[J,I], SEED[J,I] takes the form
(J=1;NOFILT(I=1:NORNG))
N.B. For each autoregressive filter J, NORNG should always assume the value of "3".

NFREQ,ZETA (Real)

Defines the natural frequency and damping ratio respectively for each of the NOFILT degrees of reedom. NFREQ and ZETA are input in pairs for each of the NOFILT degrees of freedom of the model.

DXT,D2XTG (Real)

Defines the initial traverse velocity and acceleration in units of feet per second and "g" respectively. These quantities are input as pairs for each of the NOTEST independent tests.

OUTPUT FROM DSYS2 — SAMPLE LISTING

NOTEST	NORNG	NCFILT	FLTORD	TRNSNT	NOPTS	LAG	TOTPTS
1	3	1	28	118	1024	102	1142

AUTOREGRESSIVE FILTER COEFFICIENTS

FILTER NO. 1

COEFFICIENT SET NO. 1

```
 1.863970186071&  0
-1.134611897728&  0
 3.022527435261&- 1
-8.051803542410&- 2
 2.144944644157&- 2
-5.713983871934&- 3
 1.522165745433&- 3
-4.054944561398&- 4
 1.080209683954&- 4
-2.877609583635&- 5
 7.665802040450&- 6
-2.042145753125&- 6
 5.440161056296&- 7
-1.449056993240&- 7
 3.859158375747&- 8
-1.028948827626&- 8
 2.752297016628&- 9
-7.297457352203&-10
 1.782406465938&-10
-2.966108758663&-11
```

INPUT PARAMETERS

FLTNO	R.NO.	MULTIPLIER	SEED
1	1	147	1
1	2	182	1
1	3	190	1

INITIAL VELOCITY F.P.S.		INITIAL ACCELATION G OR F.P.S*S.	
0.000000000000&	0	0.200000	6.435070930068& 0

SAMPLE SPATIAL INCREMENT REALISATION POINTS — 1ST & LAST 25 POINTS

(COMPLETE REALISATION INFORMATION OUTPUT TO FILE IN DIFFERENT FORMAT)

INCR NO.	FILT NO.	CURRENT TIME (SECS)	DISPLACEMENT INPUT (Z)	VELOCITY INPUT (DZ)	DISPLACEMENT OUTPUT (Y)	VELOCITY OUTPUT (DY)	TRAVERSE VEL (F/S)	TRAVERSE DISTANCE (FEET)
0	1	0.000	7.2248222&- 1	1.6577607& 0	7.2248222&- 1	0.0000000& 0	0.00000	0.00
0.0000610& 0	1	0.000& 0	0.000& 0	-1.1&- 24 -5.50& 57	0.0& 0	0.0& 0 0.0& 0	0.0& 0 0.0& 0	0.0& 0
1	1	0.094	7.2248222&- 1	1.6577607& 0	9.7919254&- 1	-4.1600894& 0	0.60640	0.03
0.0000610& 0	1	9.59&- 1	1.73& 1	5.22&- 1 2.43& 0	0.0& 0	0.0& 0 0.0& 0	0.0& 0 0.0& 0	0.0& 0
1	1	0.094	7.2248222&- 1	1.6577607& 0	9.7919254&- 1	-4.1600894& 0	0.60640	0.03
4.7914391&- 2	1	9.59&- 1	1.73& 1	5.22&- 1 2.43& 0	2.9&- 3	4.2&- 2 9.9&- 3	5.5&- 1 3.5&- 1	1.5&- 1
2	1	0.538	2.4349792& 0	2.8851651& 0	2.4764912& 0	2.3227290& 0	3.66357	1.04
-3.8878641&- 1	1	6.13& 0	5.40& 0	5.93& 0 8.32& 0	4.1&- 3	4.9&- 5 4.1&- 4	2.6&- 4 5.8&- 3	9.3&- 4
3	1	0.763	6.2648695& 0	3.5679846& 0	5.5623390& 0	3.9332109& 0	5.15195	2.06
1.1732951& 0	1	1.45& 1	8.79& 0	1.72& 1 9.81& 0	3.2&- 3	2.4&- 4 2.9&- 4	9.9&- 4 3.2&- 3	3.7&- 4
4	1	0.938	8.7226963& 0	1.5467521& 0	9.0405171& 0	2.4487789& 0	6.29863	3.08
-2.1638182& 0	1	2.84& 1	8.21& 0	2.94& 1 8.27& 0	2.0&- 3	5.1&- 4 9.4&- 5	1.7&- 3 8.5&- 4	2.6&- 4
5	1	1.086	9.5431516& 0	1.9083069&- 1	1.0331945& 1	2.4239279&- 1	7.26097	4.10
-1.5526877& 0	1	4.01& 1	7.00& 0	3.85& 1 7.05& 0	8.9&- 5	2.8&- 4 3.1&- 5	8.5&- 4 2.5&- 4	3.3&- 5
6	1	1.216	9.4989995& 0	7.2299996&- 2	1.0036226& 1	-5.9364541&- 1	8.11305	5.11
-1.8395566&- 1	1	4.71& 1	6.23& 0	4.46& 1 6.23& 0	4.2&- 4	6.9&- 5 4.3&- 5	1.8&- 4 3.0&- 4	2.6&- 5
7	1	1.335	9.5810951& 0	-5.2992747&- 1	9.4403485& 0	-5.2061849&- 1	8.88809	6.14
2.1081801&- 1	1	5.12& 1	5.66& 0	4.91& 1 5.66& 0	4.5&- 4	9.4&- 5 2.8&- 5	2.1&- 4 1.7&- 4	4.3&- 5
8	1	1.442	8.7972068& 0	8.3890361&- 2	9.0095376& 0	-3.5524497&- 1	9.58480	7.14
1.1174534&- 1	1	5.35& 1	5.22& 0	5.13& 1 5.21& 0	1.9&- 4	4.5&- 5 9.2&- 6	1.0&- 4 5.5&- 5	1.5&- 5
9	1	1.542	9.5660074& 0	1.0625963&- 1	8.8459096& 0	1.2033448&- 1	10.23418	8.14
6.6072604&- 1	1	5.52& 1	4.86& 0	5.41& 1 4.85& 0	4.8&- 5	1.3&- 4 1.2&- 5	2.9&- 4 7.3&- 5	1.1&- 5
10	1	1.639	8.7134256& 0	-8.0566767&- 1	9.0075082& 0	-3.9943869&- 2	10.86451	9.17
-7.2776846&- 1	1	5.69& 1	4.56& 0	5.55& 1 4.59& 0	1.4&- 4	2.9&- 4 5.2&- 5	5.5&- 4 2.6&- 4	1.2&- 5
11	1	1.728	8.1853531& 0	-9.7547979&- 1	8.6270264& 0	-6.7434893&- 1	11.44149	10.17
-4.8045502&- 1	1	5.79& 1	4.33& 0	5.61& 1 4.39& 0	3.3&- 4	4.1&- 4 2.7&- 5	7.7&- 4 1.3&- 4	4.4&- 5
12	1	1.812	7.0830499& 0	-2.4899258&- 1	7.7885761& 0	-9.3015310&- 1	11.99074	11.17
-5.0763662&- 2	1	5.80& 1	4.16& 0	5.68& 1 4.16& 0	2.5&- 5	1.5&- 5 8.6&- 8	2.9&- 5 4.3&- 7	2.0&- 6
13	1	1.897	7.9430670& 0	1.4078690& 0	7.1312549& 0	-1.2130108&- 1	12.54159	12.22
1.3522045& 0	1	5.76& 1	3.97& 0	5.62& 1 4.07& 0	8.2&- 4	2.4&- 3 2.7&- 4	3.6&- 3 1.1&- 3	2.2&- 4
14	1	1.974	9.0288629& 0	2.6635273&- 1	7.5622033& 0	8.2968508&- 1	13.04460	13.22
5.2082956&- 1	1	5.76& 1	3.83& 0	5.72& 1 3.91& 0	4.7&- 3	5.9&- 3 3.8&- 4	8.9&- 3 1.5&- 3	6.4&- 4
15	1	2.049	8.3682656& 0	-1.1424325&- 1	8.3756160& 0	5.5335100&- 1	13.52892	14.22
-8.9715779&- 1	1	5.81& 1	3.70& 0	5.77& 1 3.81& 0	1.9&- 3	3.3&- 3 2.8&- 4	5.0&- 3 1.1&- 3	2.9&- 4
16	1	2.121	7.2695446& 0	-7.2572416&- 1	8.4752821& 0	-3.3318320&- 1	13.99650	15.22
-7.9924125&- 1	1	5.86& 1	3.57& 0	5.75& 1 3.70& 0	1.0&- 3	1.6&- 3 1.3&- 4	2.4&- 3 5.0&- 4	1.4&- 4
17	1	2.181	6.7119740& 0	-9.8386894&- 1	7.7017414& 0	-9.7869200&- 1	14.44895	16.22
-4.7211006&- 1	1	5.86& 1	3.48& 0	5.71& 1 3.60& 0	4.2&- 5	3.0&- 5 5.9&- 7	4.6&- 5 2.4&- 6	3.3&- 6
18	1	2.259	5.3978851& 0	-1.0217041& 0	6.5981771& 0	-1.3851155& 0	14.88765	17.22
-3.2510273&- 1	1	5.82& 1	3.43& 0	5.62& 1 3.52& 0	4.4&- 4	6.9&- 4 5.3&- 5	1.1&- 3 2.1&- 4	5.1&- 5
19	1	2.324	4.9901292& 0	-2.3084406&- 3	5.2073989& 0	-1.2527200& 0	15.31380	18.22
5.0631144&- 1	1	5.73& 1	3.38& 0	5.53& 1 3.42& 0	1.4&- 3	2.2&- 3 1.6&- 4	3.3&- 3 6.5&- 4	1.5&- 4
20	1	2.394	5.0692140& 0	-4.3377259&- 2	4.2248836& 0	-6.4595315&- 1	15.76246	19.30
6.1500447&- 1	1	5.61& 1	3.29& 0	5.44& 1 3.32& 0	2.7&- 3	2.6&- 3 9.0&- 5	3.0&- 3 2.7&- 4	3.0&- 4
21	1	2.456	4.7452340& 0	-5.3544306&- 1	3.8123688& 0	-2.2766548&- 1	16.16555	20.30
2.0045820&- 1	1	5.50& 1	3.20& 0	5.38& 1 3.24& 0	1.7&- 2	1.4&- 2 1.7&- 4	1.5&- 2 5.1&- 4	1.6&- 3
22	1	2.516	4.1845340& 0	-7.9153054&- 2	3.6801846& 0	-4.7247152&- 2	16.55884	21.30
1.6910297&- 1	1	5.39& 1	3.12& 0	5.27& 1 3.16& 0	3.1&- 3	6.6&- 3 6.3&- 4	7.5&- 3 1.9&- 3	5.1&- 4
23	1	2.576	4.2092402& 0	-7.7974607&- 1	3.6668885& 0	-2.7042934&- 2	16.94301	22.30
-1.1716072&- 1	1	5.30& 1	3.05& 0	5.14& 1 3.10& 0	1.6&- 3	4.8&- 3 5.2&- 4	5.4&- 3 1.6&- 3	3.2&- 4

THESE HEADINGS REFER TO LINES 1,3,5,7 etc.

OUTPUT	MSY	MSDY	MSZI	YT01	YT02	YT1	DYT01	DYT02	DYT1
ACCELERATION	INSTANTANEOUS MEAN			CONVERGENCE TESTS ON NUMERICAL INTEGRATION					

THIS ANNOTATION REFERS TO LINES 2,4,6,8 etc.

INCR NO.	FILT NO.	CURRENT TIME (SECS)	DISPLACEMENT INPUT	VELOCITY INPUT	DISPLACEMENT OUTPUT	VELOCITY OUTPUT	TRAVERSE VEL(F/S)	DISTANCE (FEET)
INCREMENT 24	1	2.633	2.43053428 0	-2.2045994& 0	3.42211998 0	-6.0280057&- 1	17.31865	23.30
-9.5100340&- 1		5.208 1	2.498 0	5.088 1 3.148 0	4.08- 3	1.38- 2 1.48- 3	1.48- 2 4.18- 3	8.08- 4
INCREMENT 1000	1000 1	17.537	6.88816178 0	-1.6302968&- 1	2.1820076& 0	4.3826001&- 1	113.44503	999.97
-9.9469140&- 4		1.338 1	1.978- 1	2.788 1 2.228 0	7.98- 3	2.38- 2 1.38- 3	8.88- 3 1.38- 3	9.98- 5
	1001 1	17.546	5.6364073& 0	-1.0840195& 0	2.6073495& 0	4.1404783&- 1	113.50174	1000.97
-6.1058129&- 2		1.358 1	1.788- 1	2.638 1 2.258 0	7.48- 3	1.48- 2 7.18- 4	5.38- 3 7.18- 4	6.38- 5
	1002 1	17.555	4.69665378 0	-2.0699627& 0	2.9818078& 0	3.1470818&- 1	113.55842	1001.97
-1.3290834&- 1		1.378 1	1.608- 1	2.498 1 2.278 0	4.98- 3	8.88- 3 4.58- 4	3.38- 3 4.58- 4	4.08- 5
	1003 1	17.564	2.65773328 0	-2.5592482& 0	3.2197961& 0	1.5780069&- 1	113.61507	1002.97
-1.7047387&- 1		1.388 1	1.468- 1	2.408 1 2.268 0	3.78- 3	1.38- 3 6.08- 6	5.08- 4 6.08- 6	9.68- 6
	1004 1	17.573	-1.9733244&- 1	-2.3856922& 0	3.2903705& 0	-1.3906057&- 2	113.67170	1003.97
-1.7151922&- 1		1.388 1	1.358- 1	2.358 1 2.308 0	2.98- 4	7.88- 3 4.78- 4	3.08- 3 4.78- 4	3.08- 5
	1005 1	17.581	-1.7979270& 0	-1.3304791& 0	3.1992385& 0	-1.6262831&- 1	113.72829	1004.97
-1.0932023&- 1		1.368 1	1.258- 1	2.318 1 2.338 0	2.58- 3	7.58- 3 4.18- 4	2.88- 3 4.18- 4	3.28- 5
	1006 1	17.590	-2.3463653& 0	1.3405525& 0	3.0030133& 0	-2.0048341&- 1	113.78486	1005.97
5.3448768&- 2		1.338 1	1.208- 1	2.328 1 2.178 0	2.38- 3	1.58- 3 4.88- 5	5.88- 4 4.88- 5	8.78- 6
	1007 1	17.599	6.12412328- 1	3.2407030& 0	2.8543154& 0	-6.5491375&- 2	113.84140	1006.97
1.8470313&- 1		1.288 1	1.208- 1	2.378 1 2.048 0	3.48- 4	2.08- 2 1.28- 3	7.68- 3 1.28- 3	7.68- 5
	1008 1	17.608	3.1664946& 0	2.0719802& 0	2.8858789& 0	1.0888379&- 1	113.89792	1007.97
1.2179193&- 1		1.228 1	1.238- 1	2.428 1 2.038 0	8.98- 3	1.58- 2 7.68- 4	5.88- 3 7.68- 4	7.08- 5
	1009 1	17.617	4.9000585& 0	1.4501118& 0	3.0456661& 0	1.8523296&- 1	113.95440	1008.97
9.1079076&- 2		1.168 1	1.268- 1	2.478 1 2.008 0	1.78- 3	4.78- 2 2.98- 3	1.88- 2 2.98- 3	1.88- 4
	1010 1	17.625	6.4480222& 0	2.0172042& 0	3.2690562& 0	3.0370790&- 1	114.01086	1009.97
1.2834195&- 1		1.108 1	1.288- 1	2.498 1 2.048 0	2.08- 2	6.48- 2 3.58- 3	2.48- 2 3.58- 3	2.78- 4
	1011 1	17.634	8.7262969& 0	1.8656394& 0	3.6486399& 0	4.4091066&- 1	114.06729	1010.97
1.2499678&- 1		1.048 1	1.308- 1	2.518 1 2.088 0	1.78- 2	5.28- 3 2.88- 5	2.08- 3 2.88- 5	4.18- 5
	1012 1	17.643	1.0016565& 1	1.2564018& 0	4.1471560& 0	5.4214959&- 1	114.12369	1011.97
8.7616453&- 2		9.948 0	1.298- 1	2.498 1 2.098 0	4.68- 4	2.88- 2 1.78- 3	1.18- 2 1.78- 3	1.18- 4
	1013 1	17.652	1.1114570& 1	3.1636925&- 1	4.7207859& 0	6.0403680&- 1	114.18006	1012.97
3.0536375&- 2		9.538 0	1.268- 1	2.478 1 2.128 0	5.58- 3	1.28- 2 5.88- 4	4.48- 3 5.88- 4	5.28- 5
	1014 1	17.660	1.0451027& 1	-1.2944566& 0	5.3264557& 0	5.8859850&- 1	114.23640	1013.97
-7.6410637&- 2		9.248 0	1.248- 1	2.498 1 2.128 0	2.68- 3	5.98- 3 4.28- 4	2.28- 3 4.28- 4	1.98- 5
	1015 1	17.669	8.8389243& 0	-1.8542234& 0	5.8661870& 0	4.7339596&- 1	114.29272	1014.97
-1.1925971&- 1		9.018 0	1.238- 1	2.538 1 2.088 0	1.28- 3	2.18- 2 1.38- 3	7.98- 3 1.38- 3	8.08- 5
	1016 1	17.678	6.8930450& 0	-2.0141240& 0	6.2715605& 0	3.5365850&- 1	114.34901	1015.97
-1.3931344&- 1		8.868 0	1.258- 1	2.618 1 1.928 0	8.48- 3	3.38- 2 1.98- 3	1.38- 2 1.98- 3	1.48- 4
	1017 1	17.687	5.0272690& 0	-1.5401733& 0	6.5630338& 0	2.1494815&- 1	114.40527	1016.97
-1.1822676&- 1		8.748 0	1.288- 1	2.678 1 1.798 0	1.18- 2	2.48- 2 1.28- 3	9.18- 3 1.28- 3	1.08- 4
	1018 1	17.695	4.0866977& 0	-2.0558913&- 1	6.7251890& 0	1.3170109&- 1	114.46151	1017.97
-4.0379960&- 2		8.618 0	1.328- 1	2.698 1 1.678 0	7.58- 3	7.28- 3 2.98- 4	2.78- 3 2.98- 4	3.68- 5
	1019 1	17.704	4.5123569& 0	8.2219554&- 1	6.8504843& 0	1.3131243&- 1	114.51771	1018.97
2.4284401&- 2		8.458 0	1.358- 1	2.618 1 1.558 0	2.98- 3	8.98- 3 6.08- 4	3.48- 3 6.08- 4	3.08- 5
	1020 1	17.713	5.3971546& 0	7.5240124&- 1	6.9993497& 0	1.6054232&- 1	114.57389	1019.97
2.3697735&- 2		8.298 0	1.358- 1	2.438 1 1.538 0	3.78- 3	1.18- 2 6.18- 4	4.38- 3 6.18- 4	4.78- 5
	1021 1	17.722	6.1216255& 0	1.0[illegible]& 0	7.1721049& 0	1.8368446&- 1	114.63004	1020.97
4.5412598&- 2		7.838 0	1.348- 1	2.208 1 1.558 0	2.58- 3	1.48- 2 8.88- 4	5.18- 3 8.88- 4	4.88- 5
	1022 1	17.730	7.5533519& 0	1.4650[illegible]& 0	7.3799881& 0	2.5093756&- 1	114.68617	1021.97
7.4707842&- 2		7.318 0	1.348- 1	2.008 1 1.818 0	6.38- 3	2.78- 2 1.58- 3	1.08- 2 1.58- 3	1.18- 4
	1023 1	17.739	8.9541447& 0	1.5973357& 0	7.6753634& 0	3.3232785&- 1	114.74227	1022.97
8.5985976&- 2		6.638 0	1.368- 1	1.858 1 1.448 0	7.88- 3	1.08- 2 4.78- 4	3.88- 3 4.78- 4	4.88- 5

OUTPUT	MSY	MSDY	MSZI	MSDZI	YT01	YT02	YT1	DYT01	DYT02	DYT1

ACCELERATION * INSTANTANEOUS MEAN SQUARE RESPONSES * CONVERGENCE TESTS ON NUMERICAL INTEGRATION

INCR NO.	FILT NO.	CURRENT TIME (SECS)	DISPLACEMENT INPUT	VELOCITY INPUT	DISPLACEMENT OUTPUT	VELOCITY OUTPUT	TRAVERSE VEL(F/S)	DISTANCE (FEET)
1024	1	17.748	1.0443&38& 1	8.41605[illegible]&- 1	8.0449763& 0	4.03703198&- 1	114.79833	1023.97
4.43868058&- 2		5.61& 0	1.41&- 1	1.00& 1	2.68&- 3	2.78&- 5	5.38&- 5	1.06&- 5	5.38&- 5	3.06&- 6
1025	1	17.756	1.0442735& 1	8.42[illegible]&- 2	8.4632629& 0	4.18609698&- 1	114.85438	1024.97
-4.1427861&- 3		4.95& 0	1.46&- 1	1.42& 1	[illegible]&- 1	1.48&- 3	1.08&- 2	5.98&- 4	3.98&- 3	5.98&- 4	4.08&- 5

OUTPUT	MSY	MSDY	MSZI	MSDZI	YT01	YT02	YT1	DYT01	DYT02	DYT1

ACCELERATION * INSTANTANEOUS MEAN SQUARE RESPONSES * CONVERGENCE TESTS ON NUMERICAL INTEGRATION

DISTANCE

AVERAGE MEAN SQUARE VALUES OVER TOTAL RECORD LENGTH

FLTNO	DISP. INPUT MSZX	VEL. INPUT MSUZX	DISP. INPUT (INTERPOLATED) MSZIX	VEL. INPUT (INTERPOLATED) MSUZIX	*	DISP. RESPONSE MSYT	VEL. RESPONSE MSOYT	DISP. RESPONSE (RELATIVE) MSYZT
1	5.47777195 1	2.36335898 0	5.46445758 1	2.35238486 0	*	4.94793226 0	1.46042766- 1	2.67228118 1

POLAR GENERATOR - DEGREE OF REDUNDANCY = 2.64446336252226- 1 STANDARD DEVIATION = 1.00000000000068 0

FREQUENCY OF OCCURENCE - IN VARIOUS CLASS INTERVALS

STATISTICS ON POLAR GENERATOR OUTPUT

* QUANTILE * CATEGORY - IN STANDARD DEVIATION INCREMENTS *	-4	-3	-2	-1	1	2	3	4
FILTER NUMBER *	***	***	***	***	***	***	***	***
1 *	1	18	160	339	358	130	19	1

AUTOREGRESSIVE FILTERS - STANDARD DEVIATIONS

1 6.779637364455& 0

STATISTICS ON AUTO REGRESSIVE FILTER OUTPUT

FREQUENCY OF OCCURENCE - IN VARIOUS CLASS INTERVALS

* QUANTILE * CATEGORY - IN STANDARD DEVIATION INCREMENTS *	-4	-3	-2	-1	1	2	3	4
FILTER NUMBER *	***	***	***	***	***	***	***	***
1 *	4	34	199	342	279	159	19	0

CURRENT MULTIPLIERS & SEEDS ARE

NEW RANDOM NUMBER SEEDS AT THE TERMINATION OF THE REALISATION RECORD FOR THE VARIOUS PRIMITIVE ROOT MULTIPLIERS USED BY THE RANDOM NUMBER GENERATOR.

J	I	MULT	SEED
1	1	187	9442
1	2	188	42728
1	3	190	62277

REALISED INPUT CORRELATIONS (NORMALISING FACTOR, NORMRT = 5.4350407094198 1)

LAG

REALISED INPUT ROAD PROFILE CORRELATIONS

D2?

INP CORRELATIONS ARE

L	1 1 *** *** DISPLACEMENT AUTOCORRELATION	2 1 *** *** DERIV. – DISP CROSS CORRELATION	2 2 *** *** DERIVATIVE AUTOCORRELATION
0	1.000000000000& 0	5.107475927339&- 4	4.280585193358&- 2
1	9.790471818999&- 1	4.008325434590&- 2	3.270525726613&- 2
2	9.269250052415&- 1	6.411531554945&- 2	1.748946957059&- 2
3	8.566515433651&- 1	7.721667314728&- 2	9.096253754089&- 3
4	7.769929258022&- 1	8.330732540508&- 2	3.327433851692&- 3
5	6.938371985169&- 1	8.460373282014&- 2	-2.814942883517&- 4
6	6.109189573919&- 1	8.342940447112&- 2	-1.786684462173&- 3
7	5.296579752577&- 1	8.124196933180&- 2	-2.689667024033&- 3
8	4.514292913802&- 1	7.804173480566&- 2	-3.628102843026&- 3
9	3.766560376172&- 1	7.400461586549&- 2	-4.501696547489&- 3
10	3.061743318439&- 1	6.909835518499&- 2	-5.224956684705&- 3
11	2.410798686660&- 1	6.371539351542&- 2	-5.443767266865&- 3
12	1.814902934650&- 1	5.814088107292&- 2	-5.891013907297&- 3
13	1.279003442073&- 1	5.175185927628&- 2	-6.902647413193&- 3
14	8.101728034741&- 2	4.432719340218&- 2	-7.932494155968&- 3
15	4.166083390076&- 2	3.606233501187&- 2	-8.468976583389&- 3
16	1.023112918498&- 2	2.777789791258&- 2	-7.837990399274&- 3
17	-1.374164423799&- 2	2.062745648120&- 2	-6.468951097483&- 3
18	-3.155536684688&- 2	1.470092681405&- 2	-5.580310993439&- 3
19	-4.422775012222&- 2	9.505538774898&- 3	-4.589557192242&- 3
20	-5.254009031701&- 2	5.742629722496&- 3	-2.882770382500&- 3
21	-5.830611986545&- 2	3.512527346454&- 3	-1.896858258993&- 3
22	-6.253134185069&- 2	1.593442153643&- 3	-2.014968546716&- 3
23	-6.482407445805&- 2	-5.206452643823&- 4	-2.117068973256&- 3
24	-6.499541954678&- 2	-2.532815612493&- 3	-1.900228343053&- 3
25	-6.304784585463&- 2	-4.278845687107&- 3	-1.584541916977&- 3
26	-5.936507689648&- 2	-5.622297630623&- 3	-1.053329179288&- 3
27	-5.448746570815&- 2	-6.223943404308&- 3	-2.698849158961&- 5
28	-4.943740488018&- 2	-5.681196440268&- 3	9.772001080598&- 4
29	-4.521255103600&- 2	-4.430957329403&- 3	1.448435347939&- 3
30	-4.217421713511&- 2	-2.770714586316&- 3	1.984597801369&- 3
31	-4.097334177619&- 2	-5.524963073429&- 4	2.224831787909&- 3
32	-4.171664790186&- 2	1.364232558811&- 3	1.487471261682&- 3
33	-4.348775695282&- 2	2.22[illegible]5450356&- 3	3.328363764001&- 4
34	-4.550293799451&- 2	2.[illegible]56651041530&- 3	-1.307711475278&- 4
35	-4.708723308228&- 2	2.057359533335&- 3	-2.279881020436&- 4
36	-4.839839200171&- 2	1.664307709634&- 3	-5.866297570546&- 4
37	-4.899118891566&- 2	[illegible].155855930761&- 4	-7.277594197523&- 4
38	-4.868612200437&- 2	3.145161824092&- 4	-4.033020802417&- 4
39	-4.791336261384&- 2	-3.471944305263&- 4	-1.204711286542&- 3
40	-4.619275556812&- 2	-2.234412225079&- 3	-2.457549777115&- 3
41	-4.234653853769&- 2	-[illegible]7747[illegible]3295&- 3	-2.329809231047&- 3
42	-3.657411407767&- 2	-[illegible]&- 3	-7.265016829722&- 4
43	-3.042060461809&- 2	-[illegible].282152198293&- 3	7.742186023883&- 4
44	-2.532196139661&- 2	-[illegible].15[illegible]2022854&- 3	1.347420291272&- 3
45	-2.181007957795&- 2	-3.[illegible]526014572&- 3	1.277872804758&- 3
46	-1.983865228154&- 2	-2.714394454873&- 3	8.669513151458&- 4
47	-1.874896504366&- 2	-2.112946512795&- 3	3.298505691931&- 4
48	-1.782536119914&- 2	-2.162792721110&- 3	-3.747805511170&- 4
49	-1.634031849121&- 2	-2.816124187359&- 3	-9.404089007340&- 4
50	-1.342505621114&- 2	-3.[illegible]73183971506&- 3	-1.222631679298&- 3
51	-9.027574954285&- 3	-[illegible].105449469697&- 3	-1.092811178756&- 3
52	-3.407247781370&- 3	-[illegible].[illegible]4185821059&- 3	2.678945113014&- 4
53	2.282027782077&- 3	-4.437978596826&- 3	2.029508184559&- 3
54	6.175124918546&- 3	-2.029150187576&- 3	2.507371024199&- 3
55	7.777372535795&- 3	6.303996584222&- 5	1.359660615637&- 3
56	7.980572041706&- 3	5.633029576865&- 4	-1.532841802012&- 4

51

57	8.5765843952548- 3	-1.2569546092878- 4	-1.2399950690128- 3
58	1.0282089297358- 2	-1.8384233283788- 3	-2.1433715490328- 3
59	1.4186979389568- 2	-4.2254246032548- 3	-2.4663468616898- 3
60	2.0270046858598- 2	-6.6473837090608- 3	-2.5085887896668- 3
61	2.8669334762888- 2	-9.0476527039908- 3	-1.9993909025498- 3
62	3.8760868169868- 2	-1.0311937932348- 2	-4.0581421275958- 4
63	4.9164802667958- 2	-9.8024287521668- 3	1.3622798123798- 3
64	5.8183415474798- 2	-7.8899350134108- 3	2.1957442426288- 3
65	6.4823306247178- 2	-5.6927726345318- 3	2.1562728201128- 3
66	6.9241445638998- 2	-3.6262159247118- 3	1.6544041229798- 3
67	7.1413260163988- 2	-1.9637465403748- 3	1.5310607797748- 3
68	7.2197481307398- 2	-4.0051983531308- 4	1.7065087402258- 3
69	7.1175996213668- 2	1.3894122374708- 3	1.4718596014888- 3
70	6.8611136614588- 2	2.6588395663438- 3	5.7124621043828- 4
71	6.5288809865668- 2	2.3579551636618- 3	-1.1634403518948- 3
72	6.2998322143478- 2	8.9484609857798- 4	-1.1577420769128- 3
73	6.1801018395408- 2	4.7983668052068- 4	1.7655627615858- 4
74	6.0300400900198- 2	1.0505539560628- 3	8.1499883173278- 4
75	5.8022703609768- 2	2.0728177397468- 3	1.3826334692378- 3
76	5.4358589206948- 2	3.7782946020198- 3	1.8379458858088- 3
77	4.8868204274938- 2	5.6247428016488- 3	1.8600909720348- 3
78	4.1768386141728- 2	7.5086491429928- 3	1.8558870983098- 3
79	3.3102594176768- 2	9.1427875939338- 3	1.2531898403598- 3
80	2.3637470039888- 2	9.7197554902238- 3	-2.6947424239048- 4
81	1.4640976460598- 2	8.6871448475158- 3	-1.5822489461468- 3
82	7.2759473316088- 3	7.1314866791548- 3	-1.1366569715318- 3
83	7.1531559194368- 4	6.6657185406298- 3	7.0069642163778- 5
84	-6.1695498586648- 3	6.8521562607468- 3	-3.7355078794558- 5
85	-1.3082703917568- 2	6.1991289148458- 3	-1.3165191958638- 3
86	-1.8909735436388- 2	4.2016283225608- 3	-2.6342955089768- 3
87	-2.2432232293828- 2	1.7361693895008- 3	-3.2374205167418- 3
88	-2.2967437244118- 2	-1.9148534011768- 3	-2.7928896562188- 3
89	-2.0965574149658- 2	-4.0470911718058- 3	-1.2744325375918- 3
90	-1.7673203698418- 2	-4.2674090872488- 3	8.3625168007728- 4
91	-1.5060302598838- 2	-2.5637532942938- 3	2.3688535633208- 3
92	-1.4507122758498- 2	6.4318709045168- 5	2.6643127763628- 3
93	-1.6397795385848- 2	2.5110431156968- 3	2.2035579121358- 3
94	-2.0370850014158- 2	4.4117870208408- 3	1.5547503751468- 3
95	-2.6058381058388- 2	5.6585708012878- 3	1.0086282456838- 3
96	-3.2731072205298- 2	6.4108773384868- 3	4.1448365801328- 4
97	-3.9943727218278- 2	6.3015198960288- 3	-7.7813655558248- 4
98	-4.6473757177928- 2	4.8372090471268- 3	-2.0164463553338- 3
99	-5.1192830745598- 2	2.3761324379008- 3	-2.9339591556258- 3
100	-5.3209072844738- 2	-9.0330960522858- 4	-3.4972537121588- 3
101	-5.1740956114798- 2	-4.4134885281848- 3	-3.4501233450168- 3
102	-4.6915524853208- 2	-7.6384395927848- 3	-2.9249951875008- 3

CORRELATION MEAN INPUT VALUES -(DEVIATION FROM ZERO MEAN)

FLTNO	BARZ(J)	BARZ(J+1)
1	-8.5235700205498- 1	9.5160817234698- 3

CALCULATED MEAN INPUT VALUES
ALONG GENERATED ROAD PROFILE REALISATION