ISSN 0140 3818
SSSU 14

UNIVERSITY OF SOUTHAMPTON

DEPARTMENT OF SHIP SCIENCE

FACULTY OF ENGINEERING AND APPLIED SCIENCE

SORACOS V3.2
A suite of programs for the simulation of Roll and Capsize of ships
VOLUME 2 : MAINTENANCE MANUAL

David G. Jones October 1982

SORACOS V3.2:

A suite of programs for the simulation of Roll and Capsize of Ships

Volume 2 MAINTENANCE MANUAL

David G. Jones October 1982

OVERVIEW

This manual describes the detailed operation of a suite of programs capable of simulating the large angle rolling of ships.

The suite, SORACOSV3·2. consists of four programs viz:

GZPHASE

PERTPHASE

TIMEPHASE

GLOBALS

It should be noted at this point, that each of the first three programs use routines that are contained in GLOBALS. Therefore GZPHASE...TIMEPHASE are always executed with GLOBALS.

CONTENTS

Overview (i)

Operating Environment

- Software Requirements 2
- Hardware Requirements 3

Gzphase

- Program Description 5
- Program Structure 6
- Module Interelationship 7
- Labelled Block/Module Interelationship 8
- Common Block Maps 9
- Operating Environment 15
- File Usage 16
- File Descriptions 17
- Procedure Descriptions
  - EXECUTIVE CONTROLLER 18
  - GZPMNU 19
  - GZDFNE 20
  - GZREAD 22
  - GZPRNT 24
  - GZPLOT 25

Pertphase

- Program Description 28
- Labelled block/Module Interelationship 29
- Common Block Maps 30
- Operating Environment 50
- File Usage 51
- File Descriptions 52
- Procedure Descriptions
  - EXECUTIVE CONTROLLER 53
  - CPXDET 54
  - ECEHLL 56
  - EQTN1 59
  - EQTN2 61
  - EQTN3 63
  - FINDLS 65
  - PRHMSS 67
  - PRPLOT 69

PRRDEX 71
PRRSLT 72
PRMNU 74
PRDTTE 76
PRWRTE 77
PSRDDT 78
RESID1 79
RESID2 81

Timephase

Program Description 84
Labelled Block/Module Interelationship 85
Common Block Maps 86
Operating Environment 102
File Usage 103
File Descriptions 104
Procedure Descriptions
EXECUTIVE CONTROLLER 105
DF 106
DFEQN1 111
DFEQN2 113
RKMRTS 116
FRCFTN 120
TMHMSS 122
TMMNU 124
TMPLOT 126
TMRDDT 128
TMRDEX 129
TMRSLT 130
TMDTTE 131
TRPZDL 132
TMWRTE 135
XYFCN 136

Globals

Program Description 140
Program Structure 141
Module Interelationship 142
Common Block/Module Interelationship 143
Common Block Maps 143
Operating Environment 146
Procedure Description
GRID 147
GRSOO1 149
GRSELT 151
SPLINT 153

DRAAXI 156
XYPLTG 158
DEVSEL 162
WINDUP 163
WNDDWN 164
CLRSCN 165
BEEP 166
PLTC 167

Amendment List 169

Copy Control 170

CHAPTER 1

OPERATIONAL ENVIRONMENT

CHAPTER 1

OPERATIONAL ENVIRONMENT

This chapter outlines the software and hardware requirements for the successful execution and use of the SORACOS suite of programs.

Details of actual channel assignments, disk formats required etc. can be found under the appropriate program description (chps 2...5).

1.1 Software Requirements

The SORACOS suite is written in an extended form of ANSI X3.9 1966 FORTRAN IV, and is not upward compatable with FORTRAN 77. The extensions used are as follows:

(i) Use of END, ERR in a READ statement.
i.e. READ (NIN, 900, END=100, ERR=200)A

(ii) Use of list directed READ statement
i.e. READ (NIN,*)A
This allows free format input.

(iii) Use of quotes in format statements
i.e. 905 FORMAT (1H, 2X,'**** ERROR 10')

SORACOS also uses the GIND-F graphics software library MK2.6.

1.2 Hardware Requirements

SORACOS was designed to be used on a PRIME 550 computer system. However, it is relatively machine independent and can be executed on any comparible system capable of the software support outlined in the previous section and the hardware support outlined in this section.

In addition to the computer, a lineprinter is required for the printing of results. Some form of file storage, usually disk is also needed.

The user interacts with the program via a tektronix type graphics terminal and perhaps a CALCOMP 81 flatbed plotter. Alternatively the user could interact with SORACOS via a normal VDU and CALCOMP 81 plotter.

CHAPTER 2

GZPHASE

CHAPTER 2

GZPHASE

This chapter describes the detailed operation of the program 'GZPHASE' together with its relationship with other programs in the suite.

Procedural descriptions are in the order found in 'GZPHASE'.

2.1 Program Description

GZPHASE is a program within the suite SORACOSV3.2.

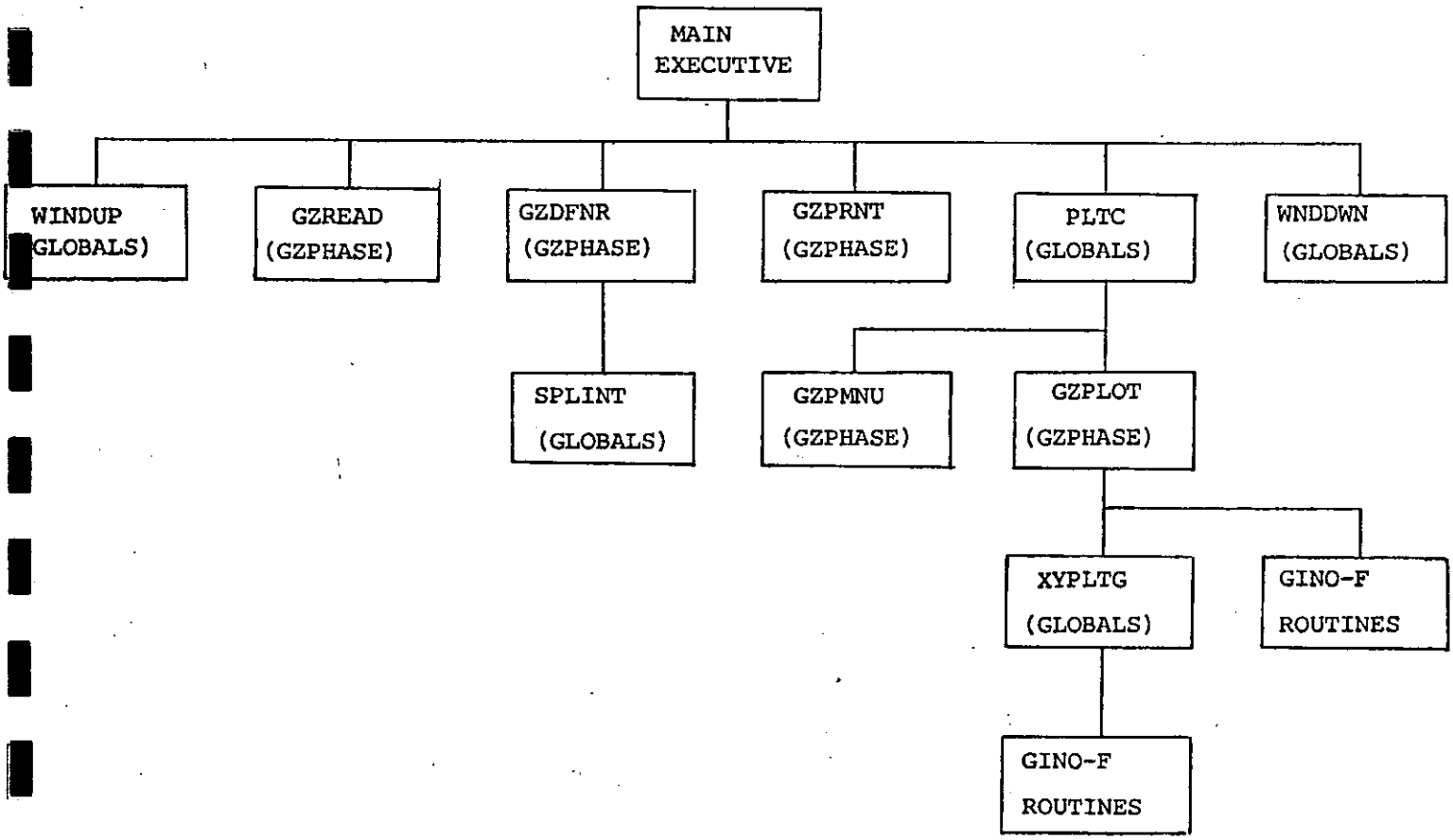
Its function is to calculate the odd order polynomial coefficients of the GZ curve, as required by PERTPHASE and TIMEPHASE.

Options are provided to allow the fitted and original curves to be plotted, together with a list of users data points.

The language used is the same as specified for the whole suite, as is the peripheral requirements.

2.2 Program Structure

The procedural structure of GZPHASE is illustrated below.



2.2.1 Module Interelationship

CALLED BY / ROUTINE CALLED	GZPHASE-	EXECUTIVE	GZPMNU	GZREAD	GZDFNE	GZPRNT	GZPLOT	GLOBALS-	GRID	GRSOO1	GRSELT	SPLINT	DRAAXI	XYPLTG	DEVSEL	WINDUP	WNDDWN	CLRSCN	BEEP	PLTC
GZPHASE-																				
GZPMNU																				X
GZREAD		X																		
GZDFNE		X																		
GZPRNT		X																		
GZPLOT																				X
GLOBALS-																				
GRID														X						
GRSOO1											X									
GRSELT														X						
SPLINT					X															
DRAAXI														X						
XYPLTG							X													
DEVSEL																				X
WINDUP		X																		
WNDDWN		X																		
CLRSCN																				X
BEEP																				X
PLTC		X																		
GINO-F																				
T4O1O															X					
CC81															X					
CHAHOL													X	X						
CHASIZ													X	X						
CHAANG													X	X						
CHAARR														X						
CHAFIX													X							
CHAINT													X							
CHASWI															X					
DEVEND																				
DEVICE															X					
LINTO2													X	X						
LINBY2									X				X							
MOVTO2							X		X				X	X						
SYMBOL							X							X						
SCALE2															X					
PICCLE														X						
M/C EXIT		X																		
T1OU																			X	X

2.2.2 Labelled Block/Module Interelationship

COMMON BLOCK \ MODULE	GZPHASE -	EXECUTIVE	GZPMNU	GZREAD	GZDFNE	GZPRNT	GZPLOT	GLOBALS -	GRID	GRSOOl	GRSELT	SPLINT	DRAAXI	XYPLTG	DEVSEL	WINDUP	WNDDWN	CLRSON	BEEP	PLTC
DATA 1		X		X	X	X	X							X						
GZDATA				X	X	X	X													
INPOUT		X	X	X		X	X								X	X				X
PLMODE			X											X						
WHERE		X		X											X	X		X	X	

2.2.2 Labelled Common Block Variable Map

COMMON/DATA1 /IPASS, NOERR, ITTL

IPASS - Integer
Not used in this version.

NOERR - Integer
Represents the number of errors encountered on input of data.

ITTL - Integer array ITTL (40)

Represents the title of the current run as specified by the user. Characters are stored in A2 format (i.e. 2 characters per element) with a maximum of 80 characters.

2.2.2.2 Labelled Common Block Variable Map

COMMON/GZDATA/GZVNSH, GZCOEF, GZORDR, GZWGHT, GZAMP, GZANG, GZNPNT, GZFLAG, GZCURV

GZVNSH - Real

GZVNSH represents the vanishing angle of the users GZ curve.

Supplied by user.

GZCOEF - Real array GZCOEF(15)

GZCOEF represents the GZ coefficients of the fitted polynomial, given that the GZ curve can be represented as -

$$GZ(\theta) = \frac{w_o^2}{r_1} \left(r_1 + r_2\theta^3 + r_3\theta^5 \ldots \frac{r_{n+1}}{2}\theta^n\right)$$

then

$$GZCOEF(1) = r_1$$
$$GZCOEF(3) = r_2$$
$$\vdots$$
$$GZCOEF(N) = \frac{r_{N+1}}{2}$$

It should be noted that N must be 3..15 and that only the odd terms of GZCOEF are used.

GZORDR - Integer

GZORDR represents the order of the GZ curve. It will have an odd value in the range 3...15.

GZWGHT - Integer

GZWGHT is a flag that represents the weighting used by the polynomial fitting routine that is required by the user.

GZWGHT = 1 ; more accurate at small angles of heel.

GZWGHT = 2 ; more accurate at large angles of heel.

GZAMP - Real array GZAMP(25)

GZAMP contains the GZ (metres) for points supplied from the users GZ curve.

GZANG - Real array GZANG(25)

GZANG contains the angle of heel (radians) for each of the corresponding points in GZAMP i.e. GZANG(I) is the angle of heel that produces a GZ of GZAMP(I).

GZNPNT - Integer

GZNPNT represents the number of points supplied by the user to define the GZ curve. GZNPNT is also the number of elements used in GZAMP and GZANG.

If GZNPNT = 0 then it means that the user is going to supply the GZ coefficients direct to the program.

GZPLAG - Integer

Not used in this version

GZCURV - Real array GZCURV(52)

The users points are expanded out to 52 pairs of points using cubic-spline interpolation. As these points are equidistant GZCURV represents the GZ of each of these expanded points.

2.2.2.3 Labelled Common Block Variable Map

COMMON/INPOUT/NIN, NOUT, NINM, NOUTP3, NOUTP4,
NOUTP2, NOUTR, NOUTG, NOUTM, NOUTP1

NIN - Integer

NIN represents the fortran logical channel number of the input data file

Set in WINDUP (GLOBALS)

NOUT - Integer

NOUT represents the fortran logical channel number of the output file which is laterlisted on a line pointer.

NINM - Integer

NINM represent the fortran logical channel number of console input. (Terminal).

NOUTP3 - Integer

Not used in this version.

NOUTP4 - Integer

Not used in this version.

NOUTP2 - Integer

NOUTP2 represents the fortran logical channel number of the flat bed plotter for plotting output.

NOUTR - Integer

NOUTR represents the fortran logical channel number of the scratch data file, which is deleted when not required.

NOUTG - Integer

Not used in this version.

NOUTM - Integer

NOUTG represents the fortran logical channel number of the console output (Terminal).

NOUTP1 - Integer

NOUTP1 represents the fortran logical channel number of the tektronix terminal for plotting.

2.2.2.4 Labelled Common Block Variable Map

COMMON/PLMODE/MODE

MODE - Integer

MODE represents the plotting mode required, and can have two values

MODE = 1; plot lines
MODE = 2; plot symbols (+)

2.2.2.5 Labelled Common Block Variable Map

COMMON/WHERE/ATAMTE

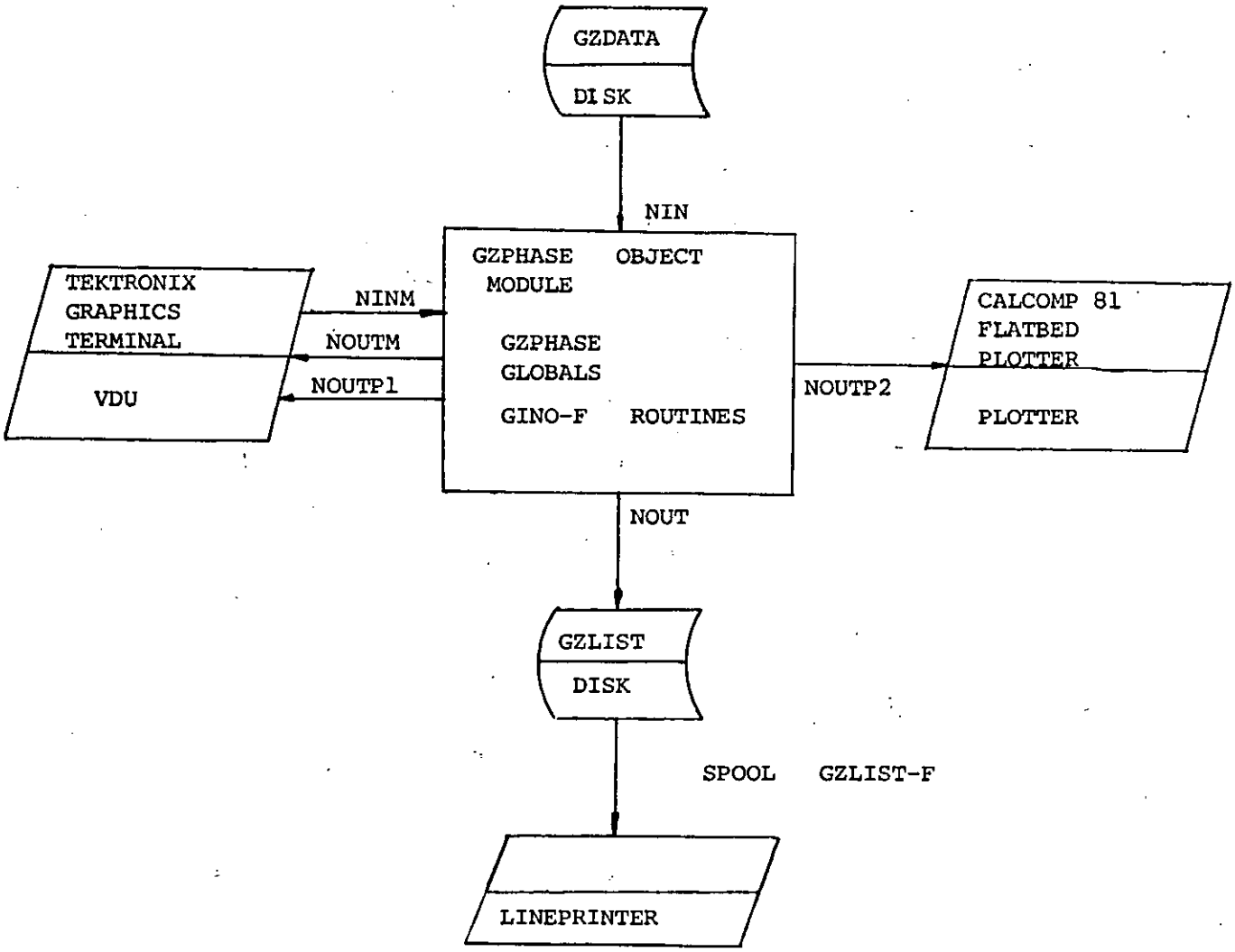
ATAMTE - Logical

Represents the location of the program suite. When set to true, indicates that SORACOS is situated at AMTE(H).

Set in WINDUP (GLOBALS).

2.3 Operating Environment

The relationship between data files and peripherals, together with the programs required in the GZPHASE object module is illustrated below.



2.3.1 File Usage

PROGRAM VARIABLE	LOGICAL CHANNEL NUMBER	PRIMOS FUNIT NUMBER	FILE NAME	R/W	COMMENTS
NOUTM	1	-	-	W	Output to users terminal
NINM	1	-	-	R	Input from users terminal
NIN	8	4	GZDATA	R	Contains data for program
NOUT	9	5	GZLIST	W	Contains results for later printing.(uses fortran format effectors)
NOUTP1	1	-	-	W	Used by GINO for graphic output. (terminal).
NOUTP2	1	-	-	W	Used by GINO for graphic output. (Flatbed plotter).

2.3.2 File Descriptions

FILENAME	TYPE	FORMAT	MEDIA
GZDATA	STANDARD SERIAL MAGNETIC	MINIMUM 128 BYTES/RECORD	DISK
GZLIST	STANDARD SERIAL MAGNETIC	MINIMUM 128 BYTES/RECORD	DISK

Description

The main segment of GZPHASE that controls the running of the program.

Specification

None.

Theory

None.

Description

The structure of the main segment is straightforward and self-explanatory.

Input Parameters

None.

Error Indicators

None

Ancillary Routines

WINDUP (GLOBALS)	GZREAD (GZPHASE)	EXIT (M/C DEP.)
WNDDWN (GLOBALS)	GZDFNE (GZPHASE)	
PLTC (GLOBALS)	GZPRNT (GZPHASE)	

Timing

Not applicable

Storage

120 integer elements

Accuracy

Not applicable

References

None

2.4.1 GZPMNU

Description

Subroutine to present the menu of plot to the user, and accept and validate the users selection.

Specification

SUBROUTINE GZPMNU (IX, IY)

Description of parameters

IX Integer

On exit IX specifies the user requirement for graphic output.

IX = 0 : Terminate program

IX = 1 : Plot users GZ curve

IX = 2 : Plot fitted GZ curve

IY Integer

Not used

Theory

None

Description

The structure of GZPMNU is self explanatory and straightforward.

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable

Storage

There are no internally declared arrays.

Accuracy

Not applicable

References

None

Description

Routine to determine the coefficients of an odd order polynomial that gives the best fit to the GZ curve.

Specification

SUBROUTINE GZDFNE

Theory

The theoretical basis of the procedure is formulated in appendix A2 of (1).
The fit is optimised (among polynomials of specified degree) with respect to a weighted least squares criterion, and a choice of two weight functions is available. One gives greatest weight to error at large roll angles (near the vanishing angle), the other to errors at small roll angles.

Description

The user supplied data is expanded into 52 points using cubic spline interpolation. Two of the 52 points i.e. 'GZCURV(51)'and 'GZCURV(52)' are beyond the vanishing angle as to obtain continuity in the curve fit through the vanishing angle.

The Tchebycheff polynomials are generated next using the recursion formula A2.11 or A2.22 in (1), depending on the weight function required.

The coefficients of the polynomial are then obtained using A2.15, A2.16, A2.18 from (1) or A2.26, A2.27, A2.28 from (1).

Finally correction is made for the vanishing angle from A2.29 in (1).

Error Indicators

None

Auxillary Routines

SPLINT (GLOBALS)

Timing

Timing is related to the order GZORDR.

Storage

THere are 40 integer elements and real elements declared internally.

Accuracy

The routine can produce coefficients that are accurate to 5 significant figures using single precision 32 bit arithmetic.

References

(1) 'Ship Roll Response in regular and non-regular seas'
J.G. Wright
Southampton University report (1978) p.106-111.

2.4.3 GZREAD

Description

To read in and validate data relating to the GZ curve.
Errors are printed at the users terminal (NOUTM).

Specification

SUBROUTINE GZREAD (IOPT).

Description of Parameters

IOPT Integer
On exit, IOPT specifies whether the user wants the GZ coefficients calculated, or if he is supplying them.
IOPT = 0 : user supplies GZ coefficients
IOPT = 1 : calculate GZ coefficients.

Theory

None

Description

The function of global variables is described elsewhere.

NOERR is a count of errors that occur on input. This is to allow the program to proceed as far as possible before aborting. Should NOERR $\neq 0$ then GZPHASE aborts. The structure of GZREAD is by and large self-explanatory.

Error Indicators

None.

Auxillary Routines

EXIT (M/C DEPENDENT)

Timing

Not applicable.

Storage

140 integer elements and 117 real elements.

Accuracy

Not applicable

References

None.

2.4.4 GZPRNT

Description

Utility routine for printing the GZDATA supplied by the user or GZ coefficients determined by GZDFNE.

Specification

SUBROUTINE GZPRNT (CURVE).

Description of Parameters

CURVE - Integer
On entry CURVE has a value which indicates the output required.

CURVE = 1 : GZ curve data as supplied by the user

CURVE = 2 : GZ coefficients are to be printed.

Unchanged on exit.

Theory

None

Description

Definition of global variables is described elsewhere.
The structure and operation of this routine is self-explanatory.

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable

Storage

40 integer elements and 119 real elements.

Accuracy

Not applicable.

References

None

2.4.5 GZPLOT

Description

Utility routine for plotting the GZ curve either

1) From the user supplied data
2) From the fitted coefficients.

Specification

```
SUBROUTINE GZPLOT (CURVE, IY)
INTEGER CURVE, IY
```

Description of Parameters

CURVE - Integer

On entry CURVE specifies which output is required.

CURVE = 1 : a plot of the GZ curve from the users data points.

CURVE = 2 : a plot of the fitted GZ curve using the polynomial coefficients.

Unchanged on exit.

IY - Integer

Not used.

Theory

None

Description

Definitions of common block variables is described elsewhere. The structure and function of this routine is self explanatory except the function of DPTS.

DPTS (I,J) stores the points to be plotted.
DPTS (1,J) stores the J^{th} x-ordinate.
DPTS (2,J) stores the J^{th} y-ordinate corresponding to the J^{th} x-ordinate.

Error Indicators

None.

Auxillary Routines

XYPLTG (GLOBALS) MOVTO2 (GINOF)
SYMBOL (GINOF)

Timing

Not applicable.

Storage

64 integer elements and 219 real elements.

Accuracy

Not applicable.

References

None.

CHAPTER 3

PERTPHASE

CHAPTER 3

PERTPHASE

This chapter describes the detailed operation of the program 'PERTPHASE' together with its relationship with other programs in the suite.

Procedural descriptions are in the order found in 'PERTPHASE'.

3.1 Program Description

PERTPHASE is a program within the suite SORACOS V3·2. Its function is to calculate a frequency response and a measure of stability of a ship rolling in beam seas with zero forward speed.

Two mathematical models are available, both can take into account heave and one of the models can cope with coupled sway as well.

Options are available for plotting the different responses and providing the results in tabular form.

The language used is the same as specified for the whole suite, as is the peripheral requirements.

3.2.2 Labelled Common Block/Module Interelationship

COMMON BLOCK / MODULE	PERTPHASE	EXECUTIVE	CPXDET	ECHELL	EQTN1	EQTN2	EQTN3	FINDLS	PRHMSS	PRPLOT	PRREDX	PRRSLT	PRMNU	PRDTTE	PRWRTE	PSRDDT	RESID1	RESID2	RESID3	RE33A	RTFNDR	RTFND2	ROMIN	GLOBALS-	GRID	GRSOO1	GRSELT	SPLINT	DRAAXI	XYPLTG	DEVSEL	WINDUP	WNDDOWN	CLRSON	BEEP	PLTC
DATA1		X							X	X	X	X		X	X	X														X						
DATA2										X	X			X																						
DATA5					X	X									X	X																				
DATA6					X	X	X					X			X	X	X	X		X																
DATA7					X	X	X		X			X	X		X	X		X																		
DWBPRT							X		X											X																
DWBTME							X		X						X	X				X																
GZDATA					X	X	X		X							X																				
INPOUT		X							X	X	X	X	X	X	X	X	X	X	X				X							X	X	X		X		X
PLMODE													X																	X						
PRHMDT					X	X	X	X	X						X	X	X	X		X																
PRHMRS					X	X	X		X										X	X		X														
RSDE1									X								X	X		X																
WHERE											X																						X	X		

Labelled Common Block Variable Map

COMMON/DATA1 /IPASS, NOERR, ITTL

IPASS - Integer

Not used in this version.

NOERR - Integer

Represents the number of errors encountered on input of data.

ITTL - Integer array ITTL (40)

Represents the title of the current run as specified by the user. Characters are stored in A2 format (i.e. 2 characters per element) with a maximum of 80 characters.

Labelled Common Block Variable Map

COMMON/DATAZ/XEXP, YEXP, NOEXP, IXOPT, IYOPT

XEXP - real array of DIMENSION (25)
XEXP(I) specifies the I^{th} x-coordinate of the users experimental points in users data units.

YEXP - real array of DIMENSION (25)
YEXP(I) specifies the I^{th} y-coordinate of the users experimental points in users data units.

NOEXP - integer
represents number of pairs of points in XEXP and YEXP. If NOEXP = 0 then no points are plotted.

IXOPT - integer
specifies the users option for the x-axis of the graph onto which the experimental points will be plotted. Uses the same xonvention as 'PRMNU'.

IYOPT - Integer
as IXOPT but for the y-axis.

Note The experimental points are only plotted onto a graph under the following conditions

(i) O NOEXP = 25
(ii) IXOPT and IYOPT correspond to the users selection for the x and y-axis of his graph as specified from the plotting menu.

Labelled Common Block Variable Map

COMMON/DATA 5/BARK1

BARK1 - Real

Defines the angle dependent damping $\tilde{k}_1$

Labelled common block variable map

COMMON/DATA6/PAFREQ, PAPZ, PADZ, PAP, PADR, NOPAPZ
(variables relating to parametric)

PAFREQ - real array of DIMENSION (25)
If the parametric is defined as a function of frequency the PAFREQ(I) contains the wave encounter frequency for the I^{th} point.

PAPZ - real array of DIMENSION (25)
PAPZ(I) represents the parametric amplitude p_Z at a wave encounter frequency of PAFREQ(I) and waveslope α_e (ALPHAE)

PADZ - real array of DIMENSION (25)
PADZ(I) represents the parametric phase δ_Z at a wave encounter frequency of PAFREQ(I) and waveslope α_e (ALPHAE)

Note The above terms must be obtained (if required) using a linear ship motion program (or similar) and supplied to the program by the user.

PAP
PADR - real arrays of DIMENSION (25)
calculated in the following way-

$$PAP(I) = (\alpha m^2 + 2\alpha_m p_Z \cos(\delta_e) + p_Z^2)$$

$$PADR(I) = \tan^{-1}(\alpha_m + p_Z\cos(\delta_Z)/ - p_Z\sin(\delta_Z))$$

where

$$\delta_Z = PADZ(I)$$

and $$p_Z = PAPZ(I)$$

NOPAPZ - Integer
Number of terms in sets of points to define parametric
If NOPAPZ = 0 then parametric is independent of frequency, and is set by PEE and DELTAR.

Labelled common block variable map

COMMON/DATA7/SWPREQ, NOSWAY, PAS1, PAS2, PAVD, PAYO,
PADY, F1, F2, D1, D2, S1, S2, VD, YO, DY

(Variables relating to sway motion)

SWFREQ - real array of DIMENSION(25)
If sway is to be considered, the SWFREQ(I) specifies the frequency of the I^{th} sway coefficient.

NOSWAY - integer
represents the number of sway coefficients to be described by the user. If NOSWAY=0 then sway is not included into the basic model. (Model 1 only)

PAS1 - real array of DIMENSION(25)
PASL(I) specifies the sway coupling S1 given a wave encounter frequency of SWFREQ(I).

PAS2 - real array of DIMENSION(25)
PAS2(I) specifies the sway coupling S2 given a wave encounter frequency of SWFREQ(I).

PAVD - real array of DIMENSION(25)
PAVD(I) specifies the sway motion amplitude of the ship (metres) given a wave encounter frequency of SWFREQ(I) and waveslope α_e (ALPHAE)

PADY - real array of DIMENSION (25)
PADY(I) specifies the phase of the sway motion (rads) given a wave encounter frequency of SWFREQ(I) and waveslope α_e (ALPHAE)

Note The above terms must be obtained (if required) using a linear ship motion program (or similar) and supplied to the program by the user.

The terms F1...D2 relate to the motion in the following way.

Sway can be considered thus

$$\ddot{\theta} + D(\dot{\theta}) + R(\theta,t) = B_e + f_1\cos(\omega t+\delta_1) + f_2\cos(\omega t +\delta_2)$$

from here then

$$F1 = f_1$$

$$D1 = \delta_1$$

$$F2 = f_2$$

$$D2 = \delta_2$$

The terms S1...DY correspond to the current values calculated from PAS1...PADY using cubic spline interpolation. (All real).

Labelled common block variable map

COMMON/DWBPRT/P

P - real array of DIMENSION (5)

Given some non-linear function of θ such as

$$F_N(\theta) = F_3\theta^3 + F_5\theta^5 \ldots F_N\theta^N$$

$$F_N(a_o + a_1\sin\omega t) \approx P_o + P_1\sin\omega t + P_2\cos 2\omega t + P_3\sin 3\omega t$$

P represents the P_N terms when $F_N(\theta)$ corresponds to the non-linear part of $H(\theta)$ such that

$P(1) = P_o$

$P(2) = P_1$

etc.

Note A more detailed explanation of $H(\theta)$ can be found in the theory section of Vol. 1.

A more detailed explanation of the method used to calculate P can be found in the description of 'FINDLS'.

Labelled Common Block Variable Map

COMMON/DWBTME/IEQN, GZCOEG, PSGZCG, ZO, DELTAY

IEQN - Integer

represents mathematical model to use as selected by the user

IEQN = 1 ; model 1 (University of Southampton)

IEQN = 2 ; model 2 (University of Newfoundland)

GZCOEG - real array of DIMENSION (15)

GZCOEG represents polynomial coefficients that describe the displaced GZ curve H(θ) given

$$H(\theta) = \frac{\omega_o^2}{h_1} \left(h_1 + h_2\theta^3 + h_3\theta^5 \ldots \frac{h_{n+1}}{2}\theta^n\right)$$

then

$GZCOEG(1) = h_1$

$GZCOEG(3) = h_2$

.

.

.

$GZCOEG(N) = \frac{h_{n+1}}{2}$

N must be in range 3...15 and only the odd terms of GZCOEG are used.

PSGZCG - real array of DIMENSION(15)

PSGZCG represents the function H(θ) such that

$$H(\theta) = \frac{\omega_o^2}{h_1} \left(h_1\theta + h_2\theta^3 + h_3\theta^5 \ldots \frac{h_{n+1}}{2}\theta^n\right)$$

then

$$\text{PSGZCG (1)} = \omega_o^2$$

$$\text{PSGZCG (3)} = \frac{\omega_o^2 . h^2}{h_1}$$

.

.

.

$$\text{PSGZCG(N)} = \frac{\omega_o^2}{h_1} . \frac{h_{n+1}}{2}$$

N must be same as N for GZCOEG (=GZORDR) and only the odd terms of PSGZCG is used.

ZO - Real

represents relative heave amplitude (see theory Vol. 1)

DELTAY - real

represents relative heave phase such that

$$\delta_Z = \delta + \text{DELTAY}$$

(see theory Vol. 1)

2 Labelled Common Block Variable Map

COMMON/GZDATA/GZVNSH, GZCOEF, GZORDR, GZWGHT, GZAMP, GZANG, GZNPNT, GZFLAG, GZCURV

GZVNSH - Real

GZVNSH represents the vanishing angle of the users GZ curve.

Supplied by user.

GZCOEF - Real array GZCOEF(15)

GZCOEF represents the GZ coefficients of the fitted polynomial, given that the GZ curve can be represented as -

$$GZ(\theta) = \frac{w_o^2}{r_1} \left(r_1 + r_2\theta^3 + r_3\theta^5 \ldots \frac{r_{n+1}}{2}\theta^n\right)$$

then

GZCOEF(1) = r_1

GZCOEF(3) = r_2

"

"

"

GZCOEF(N) = $\frac{r_{N+1}}{2}$

It should be noted that N must be ≥ 15 and that only the odd terms of GZCOEF are used.

GZORDR - Integer

GZORDR represents the order of the GZ curve. It will have an odd value in the range 3...15.

GZWGHT - Integer

GZWGHT is a flag that represents the weighting used by the polynomial fitting routine that is required by the user.

GZWGHT = 1 ; more accurate at small angles of heel.

GZWGHT = 2 ; more accurate at large angles of heel.

GZAMP - Real array GZAMP(25)

GZAMP contains the GZ (metres) for points supplied from the users GZ curve.

GZANG - Real array GZANG(25)

GZANG contains the angle of heel (radians) for each of the corresponding points in GZAMP i.e. GZANG(I) is the angle of heel that produces a GZ of GZAMP(I).

GZNPNT - Integer

GZNPNT represents the number of points supplied by the user to define the GZ curve. GZNPNT is also the number of elements used in GZAMP and GZANG.

If GZNPNT = 0 then it means that the user is going to supply the GZ coefficients direct to the program.

GZPLAG - Integer

Not used in this version

GZCURV - Real array GZCURV(52)

The users points are expanded out to 52 pairs of points using cubic-spline interpolation. As these points are equidistant GZCURV represents the GZ of each of these expanded points.

Labelled Common Block Variable Map

COMMON/INPOUT/NIN, NOUT, NINM, NOUTP3, NOUTP4,
NOUTP2, NOUTR, NOUTG, NOUTM, NOUTP1

NIN - Integer

NIN represents the fortran logical channel number of the input data file

Set in WINDUP (GLOBALS)

NOUT - Integer

NOUT represents the fortran logical channel number of the output file which is laterlisted on a line pointer.

NINM - Integer

NINM represent the fortran logical channel number of console input. (Terminal).

NOUTP3 - Integer

Not used in this version.

NOUTP4 - Integer

Not used in this version.

NOUTP2 - Integer

NOUTP2 represents the fortran logical channel number of the flat bed plotter for plotting output.

NOUTR - Integer

NOUTR represents the fortran logical channel number of the scratch data file, which is deleted when not required.

NOUTG - Integer

Not used in this version.

NOUTM - Integer

NOUTG represents the fortran logical channel number of the console output (Terminal).

NOUTP1 - Integer

NOUTP1 represents the fortran logical channel number of the tektronix terminal for plotting.

Labelled Common Block Variable Map

COMMON/PLMODE/MODE

MODE - Integer

MODE represents the plotting mode required, and can have two values

MODE = 1; plot lines
MODE = 2; plot symbols (+)

Labelled common block variable map

COMMON/PRHMDT/K1, K3, PEE, DELTAR, THETAS, PSGZCF,
ASTRT, AFIN, OMEGAO, OMGMIN, OMGMAX,
L, ALPHAM, ALPHAE, NOSTEP, NMAX, NDPLCS

(user supplied coefficients)

K1 - real
linear damping term K_1

K3 - real
non-linear(cubic) damping term K_3

PEE - real
current value of parametric (if NOPAPZ 0) or
value of parametric amplitude (if NOPAPZ=0)

DELTAR - real
current parametric phase (rads) if NOPAPZ 0 or
value of parametric phase if NOPAPZ = 0.
DELTAR = δp

THETAS - real
still water bias angle (rads). See Vol 1 for conventions.

PSGZCF - real array of DIMENSION (25)
PSGZCF represents the function GZ(θ) such that if

$$GZ(\theta) = \frac{\omega_o^2}{r_1}\left(r_1\theta + r_2\theta^3 + r_3\theta^5 \ldots r_{\frac{n+1}{2}}\theta^n\right)$$

then

$$PSGZCF(1) = \omega_o^2$$

$$PSGZCF(3) = \frac{\omega_o^2}{r_1}.r_2$$

.
.
.

$$PSGZCF(N) = \frac{\omega_o^2}{r_1}.\ r_{\frac{n+1}{2}}$$

N must be same as N for GZCOEF (=GZORDR)
and only the odd terms of PSGZCF are used.

ASTRT - real
minimum relative roll amplitude (rads) to use in the frequency response.

AFIN - real
maximum relative roll amplitude (rads) to use in the frequency response.

OMEGAO - real
natural roll frequency ω_o (rad/sec) of ship.

OMGMIN - real
minimum frequency in frequency response (rad/sec).
Must be greater than zero.

OMGMAX - real
maximum frequency in frequency response (rad/sec)

L - real array of DIMENSION (5)
Given some non-linear function of θ such as

$$F_N(\theta) = F_3\theta^3 + F_5\theta^5 \ldots F_N\theta^N$$

it can be shown that

$$F_N(a_o + a_1\sin\omega t) \approx L_o + L_1\sin\omega t + L_2\cos 2\omega t + L_3\sin 3\omega t \ldots$$

L represents the L_N terms when $F_N(\theta)$ corresponds to the non-linear part of the GZ curve such that

$$L(1) = L_o$$

$$L(2) = L_1$$

etc.

Note A more detailed explanation of the method used to calculate L can be found in the description of 'FINDLS'.

ALPHAM - real

maximum waveslope α_m (in rads)

ALPHAE - real

effective waveslope α_e (in rads).

NOSTEP - integer

number of steps for search.

NMAX - integer

Not used.

NDPLCS - integer

accuracy of solution required in number of decimal places.

Labelled common block variable map

COMMON/PRHMRS/RELPHS, DELTAP, RELAO, ABSAO, RELAZ,
RELA3, RELOS2, RELOS3, ABSA1, ABSPHS,
GAMMA

(calculated results of pertubation analysis)
(at current wave frequency ω (OMEGA))

RELPHS - real
relative phase of roll motion (rads)

DELTAP - real
parametric phase (rads) δp

RELAO - real
a_o term in relative coordinates (rads)

ABSAO - real
a_o term in absolute coordinates (rads)

RELA2 - real
amplitude of second harmonic of roll motion (rads)
in relative coordinates (a_2)

RELA3 - real
amplitude of third harmonic of roll motion (rads)
in relative coordinates (a_3)

RELPS2 - real
phase of second harmonic of roll motion (rads)
in relative coordinates.

RELPS3 - real
phase of third harmonic of roll motion (rads)
in relative coordinates

ABSA1 - real
amplitude of fundamental of roll motion (rads)
in absolute coordinates (a_1)

ABSPHS - real
phase of fundamental of roll motion (rads)
in absolute coordinates.

GAMMA - Complex
GAMMA represents the stability of the roll motion at the current frequency. For better description see theory section of Vol. 1.

Labelled common block variable map

COMMON/RSDE1/RELA1

RELA1 - real
represents the current relative roll amplitude
of the ship in radians

5 Labelled Common Block Variable Map

COMMON/WHERE/ATAMTE

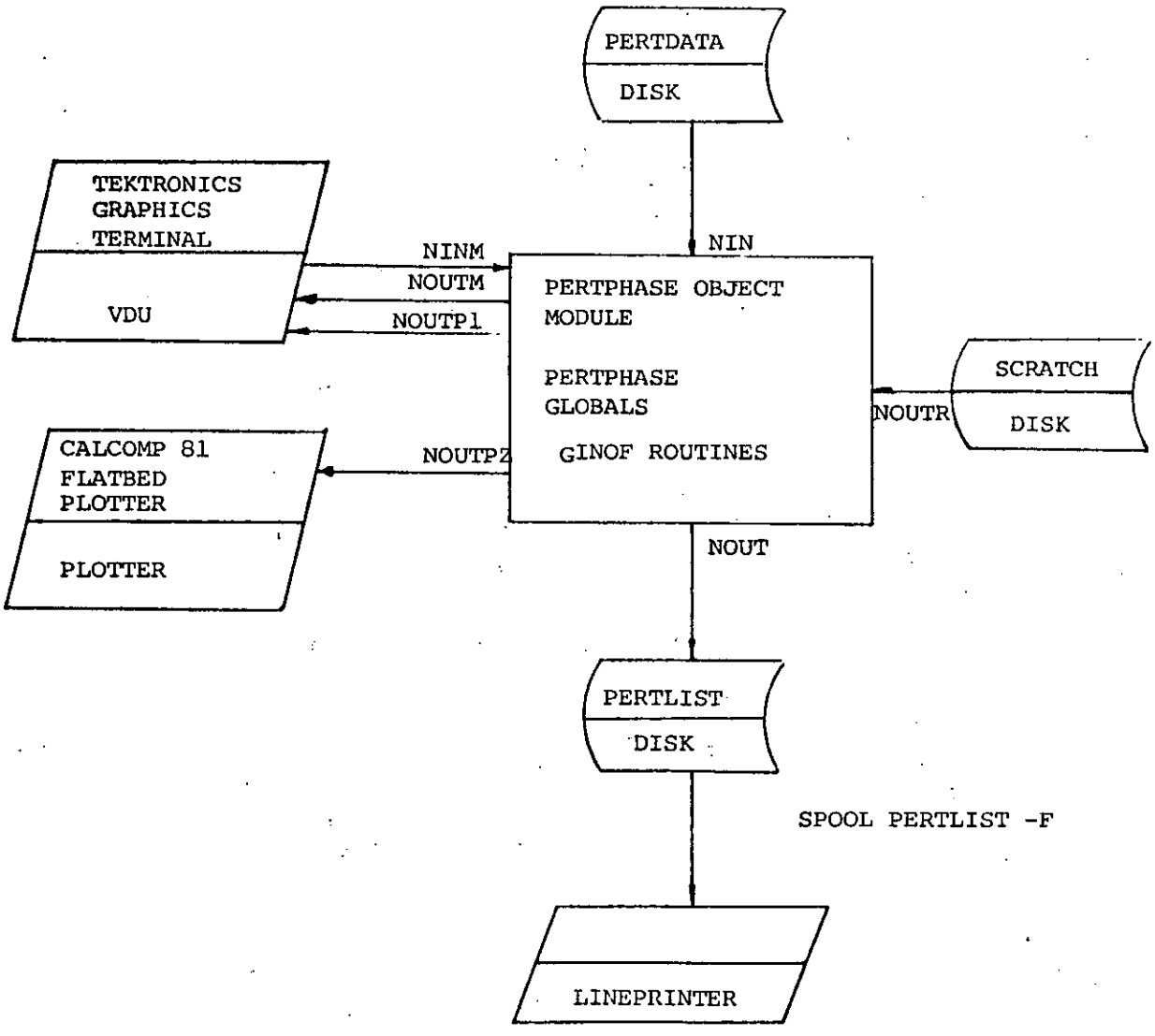
ATAMTE - Logical

Represents the location of the program suite. When set to true, indicates that SORACOS is situated at AMTE(H).

Set in WINDUP (GLOBALS).

3.3 Operating Environment

The relationship between data files and peripherals, together with the programs required in the PERTPHASE object module is illustrated below.



3.3.1 File Usage

PROGRAM VARIABLE	LOGICAL CHANNEL NUMBER	PRIMOS FUNIT NUMBER	FILE NAME	R/W	COMMENTS
NOUTM	1	-	-	W	Output to users terminal
NINM	1	-	-	R	Input from users terminal
NIN	8	4	PERTDATA	R	Contains data for program
NOUT	9	5	PERTLIST	W	Contains results for later printing (uses fortran format effecters).
NOUTP1	1	-	-	W	Used by GINO for graphic output (terminal).
NOUTP2	1	-	-	W	Used by GINO for graphic output. (flatbed plotter)
NOUTR	11	7	-	R/W	Scratch file for program.

3.3.2 File Descriptions

FILENAME	TYPE	FORMAT	MEDIA
PERTDATA	STANDARD SERIAL MAGNETIC	MINIMUM 128 BYTES/RECORD	DISK
PERTLIST	STANDARD SERIAL MAGNETIC	MINIMUM 128 BYTES/RECORD	DISK
SCRATCH	STANDARD SERIAL MAGNETIC	MINIMUM 128 BYTES/RECORD	DISK

Description

The main segment of PERTPHASE that controls the running of the program.

Specification

None

Theory

None

Description

The structure of the main segment is straightforward and self-explanatory.

Error Indicators

None

Auxillary Routines

WINDUP (GLOBALS)	PSRDDT (PERTPHASE)	PRDTTE (PERTPHASE)
WNDOWN (GLOBALS)	PRRDEX (PERTPHASE)	PRHMSS (PERTPHASE)
PLTC (GLOBALS)	PRWRTE (PERTPHASE)	PRRSLT (PERTPHASE)

Timing

Not applicable

Storage

120 integer elements are declared internally.

Accuracy

Not applicable

References

None

Purpose

Subroutine to determine the determinant of a complex square matrix by Crout Reduction.

Specification

SUBROUTINE CPXDET (A, ND, DETR, DETI, IFAIL)

Description of Parameters

A - Complex array of DIMENSION (ND, ND)
On entry A contains the matrix for which the determinant is to be found.
On exit A contains the L and U factors such that

$$\underline{A} = \underline{L} * \underline{U}$$

ND - Integer
On entry ND specifies the size of A.
Unchanged on exit.

DETR - Real
On exit DETR specifies the real part of the determinant.

DETI - Real
On exit DETI specifies the imaginary part of the determinant.

IFAIL - Integer
On exit IFAIL indicates if an error occurred in the routine

IFAIL = 0 : No error

IFAIL = 1 : Zero on leading diagonal

Theory

The method of Crout reduction is described in the reference.

Description

The routine is straight forward and self explanatory.

Error Indicators

When the routine returns IFAIL signals if an error occurred.

If IFAIL = 0 the routine has been successful. If IFAIL = 0 then a zero occurred on the leading diagonal (<1.00E - 30)

Auxillary Routines

None

Timing

The time taken to calculate a determinant is proportional to N^3.

Storage

There are no internally declared arrays.

Accuracy

References

(1) HILDEERAND F.B.
'Introduction to Numerical Analysis'
McGraw-Hill, (1956)

Purpose

Subroutine to determine the characteristic exponent of the Hill Equation. The Hill equation being of the form

$$\frac{d^2\phi}{d\tau^2} + Q(t)\phi = 0$$

where $\tau = \frac{\omega}{2} t$ to produce time scaling.

Specification

SUBROUTINE ECEHIL (Q, M, N, LAMBDA, MU, WKSP, IFAIL)

Description of Parameters

Q - Complex array of DIMENSION N
On entry Q(K) contains the complex fourier coefficients of $Q(\tau)$ such that

$$Q(J) = \frac{1}{\pi} \int Q(\tau) e^{(-2jJ\tau)} d\tau$$

evaluated between $-\pi/2$ and $\pi/2$
As fortran does not allow negative indices

$Q(N-1)/2 + 1) = Q_o$

and $Q(1) = Q_{-(N-1)/2}$

etc...
Unchanged on exit.

N - Integer
On entry N represents the range of coefficients contained in Q i.e. If

$Q_{-1o} \cdots Q_{1o}$ then N=21.

Unchanged on exit.

M - Integer

On entry M indicates the dimensions of WKSP.

M must equal (N + 1)/2

Unchanged on exit.

WAMBDA - Complex

On entry WAMBDA is set equal to Q_o

Unchanged on exit.

MU - Complex

On exit MU contains the characteristic exponent of the Hill Equation based on Floquets theorm that the Hill Equation has two linearly independent solutions. These solutions having MU and -MU as the exponents.

WKPS - Complex array of DIMENSION M, M

Workspace.

IFAIL - Integer

On exit IFAIL indicates if an error has occurred within the routine.

IFAIL = 0 ; no error

IFAIL = 1 ; zero occurred on leading diagonal.

Theory

The theory is detailed in (3).

Description

The "DO 1" loop and "DO 2" loop forms a matrix in WKSP such that

$$D(o) = \text{DET}\ \delta_{nm} + \frac{Q_{A^-M}}{\lambda + (N + 2jn)^2}$$

MU is then found from

$$\cosh(\pi\mu) = 1 - D(o)\ (1 - \cos(\pi\sqrt{\lambda}))$$

based on a number of identities (3).

Error Indicators

When the routine returns, IFAIL signals if an error occurred.

If IFAIL = 0 the routine has been successful. If IFAIL = 1 then a zero occurred on the leading diagonal of WKSP when finding the determinant D(o).

Auxillary Routines

CPXDET (PERTPHASE)

Timing

The time taken is proportional to N^3.

Storage

There are no internally declared arrays.

Accuracy

References

(1) MABINUS W, WINKLER S.
'Hills Equation'
Interscience 1966

(2) ARSCOTT F.M.,
'Periodic Differential Equations'
Perganon Press 1964

(3) WRIGHT J.G.,
'Ship roll response and capsize in regular and non regular seas'
Southampton University Report, 1978.

Purpose

Utility routine to calculate other variables required for the complete pertubation series solution for mathematical model 1 (with no sway).

Specification

SUBROUTINE EQTN1 (OMEGA, RELA1)

Description of Parameters

OMEGA - Real
On entry OMEGA specifies the frequency in rads/sec of the waves, to produce a roll amplitude of RELA1.
Unchanged on exit.

RELA1 - Real
On entry RELA1 specifies the relative ship roll amplitude in radians.
Unchanged on exit.

Theory

Details of the theory can be found in the references.

Description

As this model does not include sway, the sway terms (S1...DY) are set to zero.

A test is conducted to see if the parametric has been defined as a function of frequency. If this is so then p and δp (PEE, DELTAR) are calculated using cubic spline interpolation.

The various phases and amplitudes are calculated next.

To calculate the stability parameter GAMMA, a time history is produced (105 points) from $t = -\pi/\omega$ to $t = \pi/\omega$, of the variational equation (see 1 and 2).

Having done this, a fourier transform is conducted via complex numerical integration to find the fourier coefficients Q(I).

Finally KAPPA is calculated (1) so that GAMMA may be found.

Error Indicators

None

Auxillary Routines

ECEHIL (PERTPHASE)

Timing

The time taken is proportional to N.

Storage

There are 223 complex elements and 502 real elements.

Accuracy

Not applicable.

References

(1) WRIGHT J.G.,
'Ship roll response and capsize in regular and non-regular seas', Southampton University-AMTE(H) report. 1978.

(2) FÉAT G.R., JONES, D.G.,
'Parametric Excitation and the stability of a ship subjected to a steady heeling moment'. ISP Nov. 1981.

Purpose

Utility routine to calculate other variables required for the complete pertubation series solution for mathematical model 1 including sway.

Specification

```
SUBROUTINE EQTN2 (OMEGA, RELA1)
```

Description of Parameters

OMEGA - Real
On entry OMEGA specifies the frequency in rads/sec of the waves, to produce a roll amplitude of RELA1.
Unchanged on exit.

RELA1 - Real
On entry RELA1 specifies the relative ship roll amplitude in radius.
Unchanged on exit.

Theory

Details of the theory can be found in the reference.

Description

The structure is similar to EQTN1, with the addition of terms relating to the calculation of sway.

Error Indicators

None.

Auxillary Routines

ECEHIL (PERTPHASE)

Timing

The time taken is proportional to N.

Storage

There are 223 complex elements and 502 real elements declared internally.

Accuracy

Not applicable.

References

(1) FÉAT G.R.,
End of contract report on 'Mechanism of Capsize in Beam Waves'
March 1983

Purpose

Utility routine to calculate other variables required for the complete pertubation series solution for mathematical model 2. (does not include sway).

Specification

SUBROUTINE EQTN3 (OMEGA, RELA1)

Description of Parameters

OMEGA - Real
On entry OMEGA specifies the frequency in rads/sec of the waves, to produce a roll amplitude of RELA1.
Unchanged on exit.

RELA1 - Real
On entry RELA1 specifies the relative ship roll amplitude in radians.
Unchanged on exit.

Theory

Details of the theory can be found in the reference.

Description

The structure is similar to EQTN1.

Error Indicators

None

Auxillary Routines

ECEHIL (PERTPHASE)

Timing

The time taken is proportional to N.

Storage

There are 223 complex elements and 502 real elements.

Accuracy

Not applicable.

References

(1) BASS, D.W.,
'On the response of biased ships in large amplitude waves'.
ISP Jan. 1983

Purpose

Subroutine to determine the following series to represent the nonlinear part of the GZ curve, such that

$$F_N\ (a_o + a_1 \sin\omega t\ ) = L_o + L_1 \sin\omega t + L_2 \cos\omega t \ldots$$

Specification

SUBROUTINE FINDLS (RELA1, FNWϕ, L)

Description of Parameters

RELA1 - Real
On entry RELA1 specifies the relative ship roll in radians (a, sinωt)
Unchanged on exit.

FNWϕ - Real array of DIMENSION (15)
On entry FNWϕ(I) is a function of the GZ curve such that

FNWϕ(I) = OMEGAϕ++2*GZCOEF(I)/GZCOEF(1)

Unchanged on exit.

L - Real array of DIMENSION (5)
On exit L(I) specifies the coefficients as specified above.

Theory

A detailed explanation of the theory and method can be found in (1).

Description

The code follows by and large the details given on P112-113 of (1). It should be noted that BINCFF contains binomial coefficients for generation of the series. Also

$$A\phi N(I) = \theta_S^{(I-1)}$$

and $$AlN(I) = a_1^{(I-1)}$$

and $$L(1) = L_o$$

$$L(5) \quad L_4$$

Error Indicators

None

Auxillary Routines

None

Timing

Proportional to N

Storage

There are 649 real elements declared internally.

Accuracy

Not applicable

References

(1) WRIGHT, J.G.,
'Ship roll response and capsize in regular and non-regular seas'
Southampton University report 1978, ppl12-113.

Purpose

Subroutine to determine the frequency response of the system using the required mathematical model.

Specification

SUBROUTINE PRHMSS

Theory

None

Description

Initially, a flag, EQTN is set to the type of model required.

EQTN = 1 ; model 1 (Wright and Feat)
EQTN = 2 ; model 1 including sway (Feat)
EQTN = 3 ; model 2 (Bass)

Titles for use with the graphs is next written to NOUTR, the end of these titles being marked by the constant ENDFLE.

The rootfinding is initialised and then the approximate rootfinder is called depending on the model being used. Basically RELA1 is set at some value and the values of frequency that produce that roll amplitude are gound via the root finding.

If roots were found for the current value of RELA1, then the other variables are determined, phases etc, and conversion is then made from radians and radians/sec to degrees and hertz.

These variables are then written into the scratch file for later accessing.

RELA1 is incremented and the process repeated.

When the scratch is complete the scratch file is marked with ENDFLE and control returns to the calling program.

** WARNING **

The input/output to the scratch file has to be matched through a number of subroutines. Changes to formats or I/O statements regarding NOUTR must be undertaken with utmost care to avoid errors occurring. (This program assumes the scratch fiel to have 128 bytes/record).

Error Indicators

None

Auxillary Routines

RTFNDR	(PERTPHASE)	EQTN1	(PERTPHASE)	RESID1	(PERTPHASE)
RTFND2	(PERTPHASE)	EQTN2	(PERTPHASE)	RESID2	(PERTPHASE)
		EQTN3	(PERTPHASE)	RESID3	(PERTPHASE)

Timing

The timing is dependant on the model being used, GZORDR and NOSTEP.

Storage

There are 40 integer elements and 332 real elements declared internally.

Accuracy

See 'RTFNDR' or 'RTFND2'

References

None

Purpose

Subroutine to read in the results of the frequency analysis from the scratch file and plot out the appropriate options as dictated by the arguments of the routine.

Specification

SUBROUTINE TMPLOT (NX, NY)

Description of Parameters

NX - Integer
On entry NX specifies the option for the x-axis.
Its value is the same as returned by TMMNU.
Unchanged on exit.

NY - Integer
As NX but for the y-axis.

Theory

None

Description

The scratch file is initially reset and the titles for the graph are read into 'ITITLS'. If the title does not correspond to the appropriate option then the array is overwritten when reading in the next line.

If the 'ENDFLE' marker is encountered then the next step of the routine is executed.

Lines of results are now read in from the scratch file (NOUTR) and the values of the selected options are stored in DPTS. If the end of the file is encountered by the routine, by finding the ENDFLE marker, or DPTS is full, then a call is made to XYPLTG (GLOBALS) to plot the results on the previously nominated plot device.

The next section of the routine plots the experimental points onto the graph just drawn, should this be appropriate. Control is then returned to the calling routine.

** WARNING **

** WARNING **

The input/output to the scratch file has to be matched through a number of subroutines. Changes to formats or I/O statements regarding NOUTR must be undertaken with utmost care to avoid errors occurring.

(This program assumes the scratch file to have a minimum of 128 bytes/record)

Error Indicators

None

Auxillary Routines

XYPLTG (GLOBALS)
MOVTOZ (GINO-F)
SYMBOL (GINO-F)

Timing

Not applicable

Storage

There are 64 integer elements and 520 real elements declared internally.

Accuracy

Not applicable

References

None

Purpose

Subroutine for reading in data that represents experimental points, for plotting on the computed results graphs.

Specification

CALL PRRDEX

Description of Parameters

None

Theory

None

Description

The structure is self-explanatory

Auxillary Routines

None

Timing

Not applicable

Storage

There are 40 integer elements and 100 real elements declared internally.

Accuracy

Not applicable

References

None

Purpose

Subroutine to print the results from the frequency response.

Specification

SUBROUTINE PRRSLT

Description of Parameters

None

Theory

None

Description

The routine is quite straight forward, it simply reads the titles for the graphs until the ENDFLE marker is found.

The routine then progresses to read in the results and point them out until the next ENDFLE marker is read.

If the parametric terms were functions of frequency or sway was included then the process is repeated as there is a second set of results to print out.

Control is then restored to the calling program.

** WARNING **

The input/output to the scratch file has to be matched through a number of subroutines. Changes to formats or I/O statements regarding NOUTR must be undertaken with utmost care to avoid errors occurring.

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable

Storage

There are 40 integer and 245 real elements declared internally.

Accuracy

Not applicable

References

None

Purpose

Subroutine to establish the plot to be made, by prompting the user.

Specification

SUBROUTINE PRMNU (IX, IY)

Description of Parameters

IX - Integer
On exit IX represents the option required by the user for plotting on the x-axis. IX can be one of the following:

IX = 0 ; Terminate program
IX = 1 ; frequency (RAD/SEC)
IX = 2 ; frequency (HERTZ)
IX = 3 ; absolute roll amplitude
IX = 4 ; absolute roll phase
IX = 5 ; relative roll amplitude
IX = 6 ; relative roll phase
IX = 7 ; 1st harmonic amplitude
IX = 8 ; 1st harmonic phase
IX = 9 ; 2nd harmonic amplitude
IX = 10 ; 2nd harmonic phase
IX = 11 ; absolute a_o
IX = 12 ; relative a_o
IX = 13 ; real part of gamma

If the sway is included then the following additional options are available for plotting:-

IX = 14 ; parametric amplitude
IX = 15 ; parametric phase
IX = 16 ; sway coupling S1
IX = 17 ; sway coupling S2
IX = 18 ; drift velocity
IX = 19 ; sway amplitude
IX = 20 ; sway phase.

IY - Integer
Same as IX but for the y-axis

Theory

None

Description

The operation of the routine is self-explanatory.

Error Indicators

If the user inputs an invalid option then the user is queried again.

Auxillary Routines

None

Timing

Not applicable

Storage

There are 150 real elements declared internally.

Accuracy

Not applicable

References

None

Purpose

Utility routine to print out the experimental points as supplied by user.

Specification

SUBROUTINE PRDTTE

Description of Parameters

None

Theory

None

Description

The operation of the routine is self-explanatory.

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable

Storage

There are 40 integer and 100 real elements declared internally.

Accuracy

Not applicable

References

None

Purpose

Utility routine to point out tables of constants as supplied by the user.

Specification

SUBROUTINE PRWRTE

Description of Parameters

None

Theory

None

Description

The operation of the routine is self-explanatory

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable

Storage

There are 45 integer elements and 320 real elements declared internally.

Accuracy

Not applicable

References

None

Purpose

Subroutine to read in and validate,data for the frequency response.

Specification

SUBROUTINE PSRDDT

Description of Parameters

None

Theory

None

Description

The routine reads in various coefficients etc. relating to the frequency analysis, options being selected by various flags. Should a variable be out of range then the flag NOERR is incremented and an error message is written to the users terminal. Execution continues until a call to PRDEX is made. If NOERR is zero, there, then execution is aborted.

Error Indicators

See above.

Auxillary Routines

None

Timing

Not applicable

Storage

There are 45 integer elements and 320 real elements declared internally.

Accuracy

Not applicable

References

None

3.4.15 RESID1

Purpose

Subroutine to define the residue of the function for determining the frequency response of ship roll from mathematical model 1 (no sway).

Specification

SUBROUTINE RESID1 (OMEGA, RSDUE)

Description of Parameters

OMEGA - Real
On entry OMEGA specifies the current estimate of the root for which the residue is to be found.
Unchanged on exit.

RESDUE - Real
On exit RSDUE specifies the residue of the function at the frequency OMEGA.

Theory

From the petubation expansion (1) and (2) an expression can be found such that

$$F(\omega, a_i) = 0$$

Given a_1, is varied until the above identity is satisfied. (Using RTFNDR). This routine specifies the above identity for mathematical model 1 (no sway).

Description

The routine is self-explanatory.

Error Indicators

None

Auxillary Routines

None

Timing

Time taken is proportional to NOPAPZ and GZORDR.

Storage

There are 195 real elements declared internally.

Accuracy

Not applicable.

References

(1) WRIGHT, J.G.,
'Ship roll response and capsize in regular and non-regular seas' Southampton University - AMTE(H) Report 1978.

(2) FEAT G.R., JONES, D.G.,
'Parametric Excitation and the Stability of a Ship Subjected to a steady heeling moment'.
ISP Nov. 1981.

Purpose

Subroutine to define the residue of the function for determining the frequency response of ship roll from mathematical model 1 (with sway).

Specification

```
SUBROUTINE RESID2 (OMEGA, RSDUE)
```

Description of Parameters

OMEGA - Real
On entry OMEGA specifies the current estimate of the root for which the residue is then to be found.
Unchanged on exit.

RSDUE - Real
On exit RSDUE specifies the residue of the function at the frequency OMEGA.

Theory

From the pertubatic expansion (1) an expression can be found such that:-

$$F(w,a_1) = 0$$

given a_1, w is varied until the above identity is satisfied, (Using RTFNDR). This routine specifies the above identity for mathematical model 1 (with sway).

Description

The routine is self-explanatory.

Error Indicators

None

Auxillary Routines

None

Timing

Time take is proportional to NOPAPZ, GZORDR and NOSWAY.

Storage

There are 195 real elements declared internally.

Accuracy

Not applicable

References

(1) FEAT, G.R.,
End of contract report on 'Mechanism of Capsize in Beam Waves'.
March 1983

CHAPTER 4

TIMEPHASE

CHAPTER 4

TIMEPHASE

This chapter describes the detailed operation of the program 'TIMEPHASE', together with its relationships with other programs in the suite.

Procedural descriptions are in the order found in 'TIMEPHASE'.

4.1 Program Description

TIMEPHASE is a program within the suite SORACOS V3.2.

Its function is to provide a time domain simulation of a boat or ship rolling in beam waves with zero forward speed.

Options are provided for initial conditions, different forcing functions, including a user supplied forcing function, and various plotting options including phase portraits.

The language used is the same as specified for the whole suite as is the peripheral requirements.

4.2.2 Labelled Block/Module Interelationship

COMMON BLOCK / MODULE	TIMEPHASE	EXECUTIVE	DF	DFEQTN1	DFEQTN2	RKMRTS	FRCPTN	TMHMSS	TMMNU	TMPLOT	TMRDDT	TMRDEX	TMRSLT	TMDTTE	TRPZDL	TMWRTE	XXFCN	GLOBALS-	GRID	GRSOOL	GRSELT	SPLINT	DRAAXI	XYPLTG	DEVSEL	WINDUP	WINDDWN	CLRSCN	BEEP	PLTC
DATA1		X						X		X	X	X	X	X		X								X						
DATA6				X	X						X					X														
DATA10				X	X			X		X	X	X				X														
DWBTME					X			X			X					X														
GZDATA				X	X			X			X																			
INPOUT		X							X	X	X	X	X	X		X								X	X					X
PLMODE									X														X							
PRHMDT				X	X			X			X					X														
TMHMDT				X	X			X			X					X														
WHERE												X													X			X	X	

Labelled Common Block Variable Map

COMMON/DATA1 /IPASS, NOERR, ITTL

IPASS - Integer

Not used in this version.

NOERR - Integer

Represents the number of errors encountered on input of data.

ITTL - Integer array ITTL (40)

Represents the title of the current run as specified by the user. Characters are stored in A2 format (i.e. 2 characters per element) with a maximum of 80 characters.

Labelled Common Block Variable Map

COMMON/DATA6/BARK1

BARK1 - Real

Defines the angle dependent damping $\tilde{k}_1$

Labelled Common Block Variable Map

COMMON/DATA10/SWVST, SWVS, SFT, NOWVS, XEXP, YEXP, NOEXP,
WVSLPE, FORCE, IXOPT, IYOPT.

SWVST - Real array of DIMENSION (250)
If the user is specifying his own forcing function, then SWVST(I) contains the time in seconds of the I^{th} points.

SWVS - Real array of DIMENSION (250)
If the user is specifying his own forcing function, then SWVS(I) contains the waveslope in radians of the I^{th} point.

SFT - Real array of DIMENSION (250)
If the user is specifying his own forcing function then SFT(I) contains the forcing function $(\beta + \ddot{\alpha})$ of the I^{th} point.

NOWVS - Integer
Number of points in user defined forcing function. If NOWVS = 0 then user requires a regular time domain response.

XEXP - Real array of DIMENSION (250)
If the user is specifying experimental points for plotting onto the corresponding graph, then XEXP(I) contains the x-coordinate of the I^{th} point in user data units.

YEXP - Real array of DIMENSION (250)
If the user is specifying experimental points for plotting onto the corresponding graph, then YEXP(I) contains the y-coordinate of the I^{th} point in user data units.

NOEXP - Integer
represents number of pairs of points in XEXP(I)
and YEXP(I)
If NOEXP = 0 then no experimental points to be plotted.

WVSLPE - real
instantaneous waveslope at present point in calculation
in radians.

FORCE - real
instantaneous value of forcing function at present point
in calculation.

IXOPT - integer
represents the option on the x-axis for the experimental
point graph. Has the same values as 'TMMNU' etc.
i.e. IXOPT = 1 signifies time in seconds.

IYOPT - integer
as IXOPT but for the y-axis.

Labelled Common Block Variable Map

COMMON/DWBTME/IEQN, GZCOEG, PSGZCG, ZO, DELTAY

IEQN - Integer
represents mathematical model to use as selected by the user

IEQN = 1 ; model 1 (University of Southampton)

IEQN = 2 ; model 2 (University of Newfoundland)

GZCOEG - real array of DIMENSION (15)
GZCOEG represents polynomial coefficients that describe the displaced GZ curve H(θ) given

$$H(\theta) = \frac{\omega_o^2}{h_1} \left(h_1 + h_2\theta^3 + h_3\theta^5 \ldots \frac{h_{n+1}}{2}\theta^n\right)$$

then

GZCOEG(1) = h_1

GZCOEG(3) = h_2

.

.

.

GZCOEG(N) = $\frac{h_{n+1}}{2}$

N must be in range 3...15 and only the odd terms of GZCOEG are used.

PSGZCG - real array of DIMENSION(15)
PSGZCG represents the function H(θ) such that

$$H(\theta) = \frac{\omega_o^2}{h_1} \left(h_1\theta + h_2\theta^3 + h_3\theta^5 \ldots h_{\frac{n+1}{2}}\theta^n\right)$$

then

$$\text{PSGZCG (1)} = \omega_o^2$$

$$\text{PSGZCG (3)} = \frac{\omega_o^2 . h^2}{h_1}$$

.

.

.

$$\text{PSGZCG(N)} = \frac{\omega_o^2}{h_1} \cdot \frac{h_{n+1}}{2}$$

N must be same as N for GZCOEG (=GZORDR) and only the odd terms of PSGZCG is used.

ZO - Real

represents relative heave amplitude (see theory Vol. 1)

DELTAY - real

represents relative heave phase such that

$$\delta_Z = \delta + \text{DELTAY}$$

(see theory Vol. 1)

? Labelled Common Block Variable Map

COMMON/GZDATA/GZVNSH, GZCOEF, GZORDR, GZWGHT, GZAMP, GZANG, GZNPNT, GZFLAG, GZCURV

GZVNSH - Real

GZVNSH represents the vanishing angle of the users GZ curve.

Supplied by user.

GZCOEF - Real array GZCOEF(15)

GZCOEF represents the GZ coefficients of the fitted polynomial, given that the GZ curve can be represented as -

$$GZ(\theta) = \frac{w_o^2}{r_1} \left(r_1 + r_2\theta^3 + r_3\theta^5 \ldots \frac{r_{n+1}}{2}\theta^n\right)$$

then

GZCOEF(1) = r_1

GZCOEF(3) = r_2

"

"

"

GZCOEF(N) = $\frac{r_{N+1}}{2}$

It should be noted that N must be 3..15 and that only the odd terms of GZCOEF are used.

GZORDR - Integer

GZORDR represents the order of the GZ curve. It will have an odd value in the range 3...15.

GZWGHT - Integer

GZWGHT is a flag that represents the weighting used by the polynomial fitting routine that is required by the user.

GZWGHT = 1 ; more accurate at small angles of heel.

GZWGHT = 2 ; more accurate at large angles of heel.

GZAMP - Real array GZAMP(25)

GZAMP contains the GZ (metres) for points supplied from the users GZ curve.

GZANG - Real array GZANG(25)

GZANG contains the angle of heel (radians) for each of the corresponding points in GZAMP i.e. GZANG(I) is the angle of heel that produces a GZ of GZAMP(I).

GZNPNT - Integer

GZNPNT represents the number of points supplied by the user to define the GZ curve. GZNPNT is also the number of elements used in GZAMP and GZANG.

If GZNPNT = 0 then it means that the user is going to supply the GZ coefficients direct to the program.

GZPLAG - Integer

Not used in this version

GZCURV - Real array GZCURV(52)

The users points are expanded out to 52 pairs of points using cubic-spline interpolation. As these points are equidistant GZCURV represents the GZ of each of these expanded points.

Labelled Common Block Variable Map

COMMON/INPOUT/NIN, NOUT, NINM, NOUTP3, NOUTP4,
NOUTP2, NOUTR, NOUTG, NOUTM, NOUTP1

NIN - Integer

NIN represents the fortran logical channel number of the input data file

Set in WINDUP (GLOBALS)

NOUT - Integer

NOUT represents the fortran logical channel number of the output file which is laterlisted on a line pointer.

NINM - Integer

NINM represent the fortran logical channel number of console input. (Terminal).

NOUTP3 - Integer

Not used in this version.

NOUTP4 - Integer

Not used in this version.

NOUTP2 - Integer

NOUTP2 represents the fortran logical channel number of the flat bed plotter for plotting output.

NOUTR - Integer

NOUTR represents the fortran logical channel number of the scratch data file, which is deleted when not required.

NOUTG - Integer

Not used in this version.

NOUTM - Integer

NOUTG represents the fortran logical channel number of the console output (Terminal).

NOUTP1 - Integer

NOUTP1 represents the fortran logical channel number of the tektronix terminal for plotting.

Labelled Common Block Variable Map

COMMON/PLMODE/MODE

MODE - Integer

MODE represents the plotting mode required, and can have two values

MODE = 1; plot lines
MODE = 2; plot symbols (+)

Labelled common block variable map

```
COMMON/PRHMDT/K1, K3, PEE, DELTAR, THETAS, PSGZCF,
              ASTRT, AFIN, OMEGAO, OMGMIN, OMGMAX,
              L, ALPHAM, ALPHAE, NOSTEP, NMAX, NDPLCS
```

(user supplied coefficients)

K1 - real
linear damping term K_1

K3 - real
non-linear(cubic) damping term K_3

PEE - real
current value of parametric (if NOPAPZ 0) or
value of parametric amplitude (if NOPAPZ=0)

DELTAR - real
current parametric phase (rads) if NOPAPZ 0 or
value of parametric phase if NOPAPZ = 0.
DELTAR = δp

THETAS - real
still water bias angle (rads). See Vol 1 for conventions.

PSGZCF - real array of DIMENSION (25)
PSGZCF represents the function GZ(θ) such that if

$$GZ(\theta) = \frac{\omega_o^2}{r_1}\left(r_1\theta + r_2\theta^3 + r_3\theta^5 \ldots r_{\frac{n+1}{2}}\theta^n\right)$$

then

$$PSGZCF(1) = \omega_o^2$$

$$PSGZCF(3) = \frac{\omega_o^2}{r_1}.r_2$$

.

.

.

$$PSGZCF(N) = \frac{\omega_o^2}{r_1}.\ r_{\frac{n+1}{2}}$$

N must be same as N for GZCOEF (=GZORDR)
and only the odd terms of PSGZCF are used.

ASTRT - real
minimum relative roll amplitude (rads) to use in the frequency response.

AFIN - real
maximum relative roll amplitude (rads) to use in the frequency response.

OMEGAO - real
natural roll frequency ω_o (rad/sec) of ship.

OMGMIN - real
minimum frequency in frequency response (rad/sec).
Must be greater than zero.

OMGMAX - real
maximum frequency in frequency response (rad/sec)

L - real array of DIMENSION (5)
Given some non-linear function of θ such as

$$F_N(\theta) = F_3\theta^3 + F_5\theta^5 \ldots F_N\theta^N$$

it can be shown that

$$F_N(a_o + a_1\sin\omega t) \simeq L_o + L_1\sin\omega t + L_2\cos 2\omega t + L_3\sin 3\omega t \ldots$$

L represents the L_N terms when $F_N(\theta)$ corresponds to the non-linear part of the GZ curve such that

$L(1) = L_o$

$L(2) = L_1$

etc.

Note A more detailed explanation of the method used to calculate L can be found in the description of 'FINDLS'.

ALPHAM - real

maximum waveslope α_m (in rads)

ALPHAE - real

effective waveslope α_e (in rads).

NOSTEP - integer

number of steps for search.

NMAX - integer

Not used.

NDPLCS - integer

accuracy of solution required in number of decimal places.

Labelled common block variable map

```
COMMON/TMHMDT/AM, AE, TS, ROLL, OMEGA, TSTOP,
              DELTA, ISOLN
```

(user supplied variably)

AM - real array of DIMENSION (8)
A number of variables AM...TS are functions specified in the time domain with the following waveshape:-

AM(I) defines the maximum waveslope α_m such that

AM(1)	=	LEVEL1	(rads)
AM(2)	=	LEVEL2	(rads)
AM(3)	=	LEVEL3	(rads)
AM(4)	=	T1	(secs)
AM(5)	=	T2	(secs)
AM(6)	=	T3	(secs)
AM(7)	=	T4	(secs)
AM(8)	=	T5	(secs)

AE - real array of DIMENSION(8)
Same as AM but for the effective waveslope α_e.

TS - real array of DIMENSION(8)
Same as AM but for the still water bias θS.

ROLL - real array of DIMENSION(2)
ROLL(1) represents θ at the current point in the solution.
ROLL(2) represents $\dot{\theta}$ at the current point in the solution.

OMEGA - real
frequency of wave encounter ω (rad/sec)

TSTOP - real
end point of solution (secs)

DELTA - real

phase of waves δ (rads)

ISOLN - integer

Solution method to use

ISOLN = 1: Runga-Kutta-Mertson

ISOLN = 2: Trapezoidal with error control.

5 Labelled Common Block Variable Map

COMMON/WHERE/ATAMTE

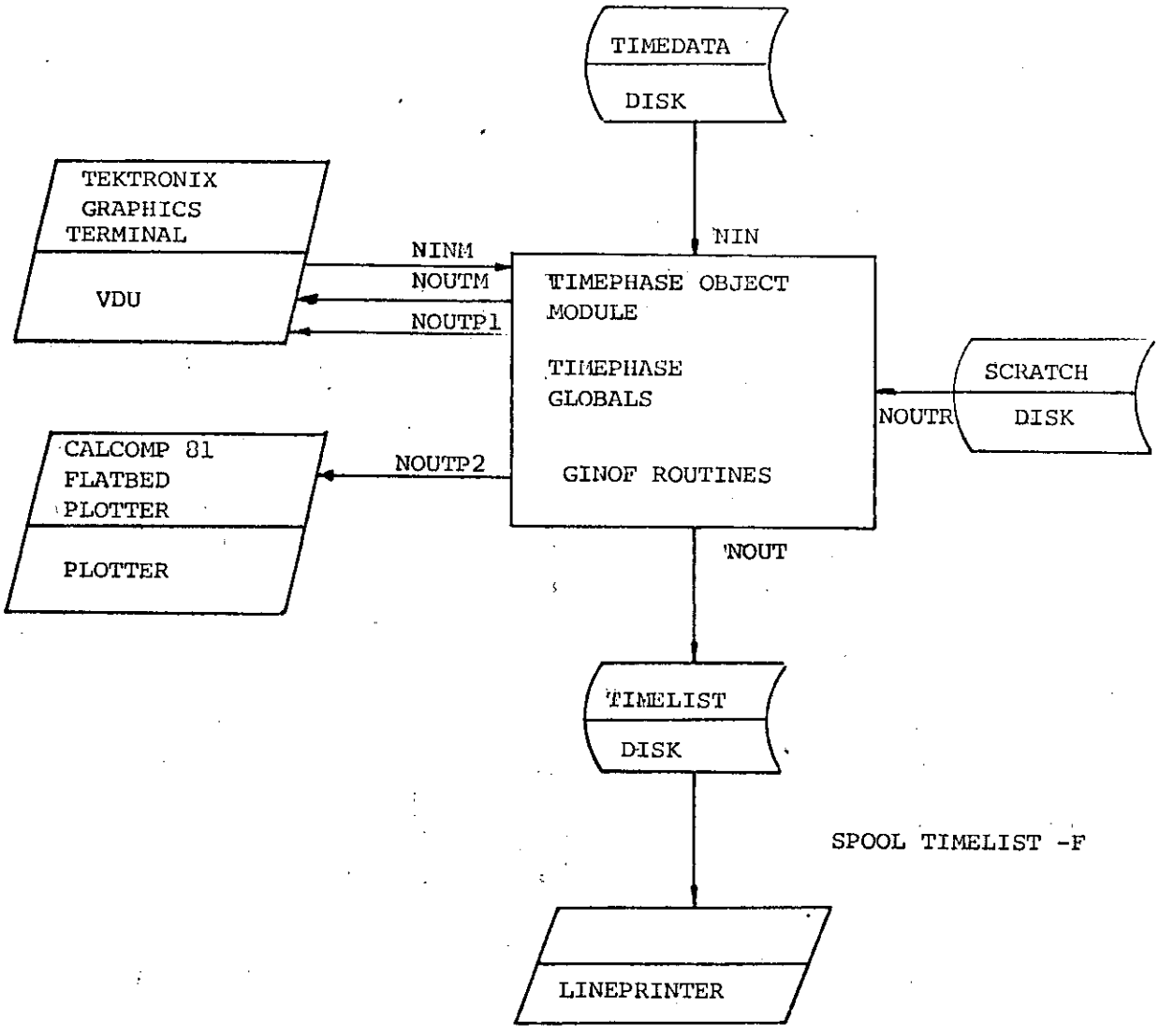
ATAMTE - Logical

Represents the location of the program suite. When set to true, indicates that SORACOS is situated at AMTE(H).

Set in WINDUP (GLOBALS).

4.3 Operating Environment

The relationship between data files and peripherals, together with the programs required in the TIMEPHASE object module is illustrated below.



4.3.1 File Usage

PROGRAM VARIABLE	LOGICAL CHANNEL NUMBER	PRIMOS FUNIT NUMBER	FILENAME	R/W	COMMENTS
NOUTM	1	-	-	W	Output to users terminal
NINM	1	-	-	R	input from users terminal
NIN	8	4	TIMEDATA	R	contains data for program
NOUT	9	5	TIMELIST	W	contains results for later printing (uses fortran format effectors)
NOUTP1	1	-	-	W	Used by GINO for graphic output (terminal)
NOUTP2	1	-	-	W	Used by GINO for graphic output (flatbed plotter)
NOUTR	11	7	SCRATCH	R/W	Scratch file used by program

4.3.2 File Descriptions

FILENAME	TYPE	FORMAT	MEDIA
TIMEDATA	STANDARD SERIAL MAGNETIC	MINIMUM 128 BYTES/RECORD	DISK
TIMELIST	STANDARD SERIAL MAGNETIC	MINIMUM 128 BYTES/RECORD	DISK
SCRATCH	STANDARD SERIAL MAGNETIC	MINIMUM 128 BYTES/RECORD	DISK

Purpose

The main segment of TIMEPHASE that controls the running of the program.

Specification

None

Description of Parameters

Not applicable

Theory

None

Description

The structure is straightforward and self-explanatory.

Error Indicators

None

Auxillary Routines

WINDUP (GLOBALS)	TMRDDT (TIMEPHASE)
WNDDWN (GLOBALS)	TMRDEX (TIMEPHASE)
PLTC (GLOBALS)	TMWRTE (TIMEPHASE)
	TMDTTE (TIMEPHASE)
	TMHMSS (TIMEPHASE)
	TMRSLT (TIMEPHASE)

Timing

Not applicable

Storage

120 Integer elements

Accuracy

Not applicable

References

None

Purpose

This subroutine integrates a system of ordinary differential equations over a range using Mersons method or the Trapezoidal Rule.

The routine advances the solution of a system of ordinary differential equations

$$\frac{dy_i}{dx} = f_i\ (x,y_1,y_2,\ldots,y_N),\ i = 1,2,\ldots,N.$$

from x to x + ho, using a number of steps of Merson's form of the Runge-Rutta method (RKN) or with the trapizoidal method and error control (TEC).

Specification

```
SUBROUTINE DF(X,Y,G,T,N,IFAIL,HO,H,AUX, YO,E,
              A,B,C,D,ISOLN)
```

Description of Parameters

X - Real
Before entry, X must be set to the initial value of the independant variable x.
On exit it will contain x + hO, unless an error has occured, when it will contain the current value.

Y - Real array of DIMENSION at least (N)
Before entry, Y must be set to the intial values of $y_1, y_2, \cdots y_N$.
On exit it will contain the computed values at x + hO, unless an error has occured, when it will contain the current values.

G - Real array, DIMENSION at least (N)
Before entry, G must be set to error bounds specified by the user for each component of the solution.
Unchanged on exit.

T - Integer

On entry, T must contain 1,2, or 3, to define the type of error test to be used. If the local error in Y(I) is estimated as E(I) then

T = 1 gives a mixed test: $|E(I)| < G(I)*(1 + |Y(I)|)$,

T = 2 gives an absolute test : $|E(I)| < G(I)$,

T = 3 gives a relative test : $|E(I)| < G(I)*|Y(I)|$

For most cases T = 1 is recommended.

T is unchanged on exit.

N - Integer

On entry, N specifies the number of differential equations.

Unchanged on exit.

IFAIL - Integer

On exit IFAIL specifies if an error has occurred.

(see section 'Error Indicators')

Hϕ - Real

On entry, Hϕ specifies the interval hϕ over which integration is required.

Unchanged on exit.

H - Real

On entry, H specifies the estimate of the step length needed for integration. (The routine will modify this if necessary to maintain local accuracy). If H is zero on entry it will be set initially to Hϕ.

On exit, H specifies the final step size.

AUX - External

Subroutine supplied by user of the form

```
SUBROUTINE AUX (F,Y,X)
REAL F(n), Y(n)
```

where n is the numerical value of N.

AUX evaluates the derivatives of Y(1), Y(2),...Y(N) at a general point X, and places them in F(1),F(2),..., F(N).

Yϕ - Real array of DIMENSION(N)
Workspace

E - Real array of DIMENSION(N)
Work space

A - Real array of DIMENSION(N)
Work space

B - Real array of DIMENSION (N)
Work space

C - Real array of DIMENSION (N)
Work space

D - Real array of DIMENSION (N)
Work space

ISOLN - Integer
On entry it specifies the method of solution required
ISOLN = 0 ; Runga-Kutta with Mertson's error control (RKM)
ISOLN = 1 ; Trapezoidal with error control (TEC)

RKM is best suited to lightly non-linear non-shift systems
TEC is suited to the same kinds of systems as RKM, but
gives a more accurate result when the system is close
to its limit of stability.

Theory

The theory used can be found in section 4.4. (RKMRTS)
or 4.4 (TRPZDL) as required.

Descriptions

The operation of this routine is fairly self-explanatory.

Error Indicators

Errors detected by the routine:

IFAIL = 1 This indicates that the step length has been halved repeatedly until it is less than 10^{-9} (initial step length). The values of X and Y(I) are those at the current stage of integrations.

IFAIL = 2 This indicates that the routine has been entered with T not equal to 1, 2 or 3. The values of X and Y(I) are the initial values.

IFAIL = 3 This indicates that the number of steps required exceeds the largest integer representable in the machine.

Auxillary Routines

RKMRTS (TIMEPHASE)
TRPZDL (TIMEPHASE)

Timing

This depends on the complexity of the system. The routine AUX is called five times for each integration step of length H.

Storage

There are no internally declared arrays.

Accuracy

The error per step is of order H^5 for RKMRTS with small H, and H^3 for TRPZDL and small step.

The error estimates obtained from these two methods are not strict bounds, but they are fairly reliable over one step. Note that only the local error is controlled by varying the step-length H. Over a number of steps the errors may accumulate in various ways, depending on the system, and there is no guarantee that the overall error will be less than the bound specified.

The user can check the results by repeating the calculation with a different set of error bounds G(I).

The user can chose the type of error test to be applied by specifying the appropriate value of T. The mixed test(given by T = 1) is effectively an absolute test for $|Y(1)| \ll 1$, and a relative test for $|Y(1)| \gg 1$, and so it meets most cases. The absolute test (T=2) may be more appropriate if the solution oscillates, but only a fixed number of decimal places are required. The relative test (T=3) may be appropriate if the solution increases rapidly over the range provided no component passes through zero.

If the routine terminates with IFAIL = 1 (step-length too small) the program can be tried again with a smaller initial value of H.

However if the failure persists, or if computing time is getting too large, the user should consider there is a more fundamental difficulty. For example,

(i) In the region of a singularity (infinite value) of the solution, the routine will usually stop with H too small, unless overflow occurs first. Numerical integration cannot be continued through a singularity, and analytical treatment should be considered.

(ii) For 'stiff' equations, where the solution contains rapidly decaying components, the routine will require a very small H in order to preserve stability (TRPZDL does this automatically) and this may make the computing time excessively long. If H is small when the system solution is varying slowly then the system is probably stiff and another method of needs to be sought (Gears Method).

References

See appropriate routine.

Purpose

Subroutine to define the ship roll in the time domain by supplying the differentials. Model used was developed at Southampton University.

Specification

SUBROUTINE DFEQN1 (F,Y, TIME)

Description of Parameters

F - Real array of DIMENSION (2)
On exit the array F(I) contains the derivatives of Y(I) evaluated at the point TIME.

Y - Real array of DIMENSION (2)
On entry Y(I) contains the value of the Ith function at the point TIME.
Unchanged on exit.

TIME - Real
On entry TIME specifies the value of the independant variable at which the derivatives are required.
Unchanged on exit.

Theory

Consider a ship rolling in a long crested wave system, incident on the beam. The basic frame of reference is a fixed, right hand coordinate system (x,y,z) origin as the still water surface, with the ship moving forwards along the x-axis, with y-axis positive to starborad. Suppose for simplicity that the c. of g. lies on the load waterplane. Let the angle of roll be denoted by ϕ . Suppose the waves are long in comparison to the ships beam so that the wave slope as experienced at the ship is approximately that on the x-axis. If we assume that $\theta=\phi - \alpha$ and given that the waves have a maximum waveslope α_m then relative roll is given by:

$$+ \tilde{k}_1\theta^2\dot{\theta}$$

$$\ddot{\theta} + k_1\dot{\theta} + k_3\dot{\theta}^3 + (1 - p\cos(wt + \delta p))\, R(\theta) = \underbrace{B + f\cos(wt+\delta)}_{F(t)}$$

where

$f = \alpha_e\omega^2$, $\delta p = \delta + \delta p$, $\tilde{k}_1$ = angle dependent damping -

(further details of nomenclature can be found in 1).

if $F(2) = \frac{d(\dot{\theta})}{dt} = \ddot{\theta}$

$F(1) = \frac{d(\theta)}{dt} = \dot{\theta}$

then we have two simultaneous, first order, differential equations viz

$$\frac{d(\theta)}{dt} = \dot{\theta}$$

$$\frac{d(\dot{\theta})}{dt} = F(t) - k_1\dot{\theta} - k3\dot{\theta}^3 - (1 - p\cos(wt+\delta p))R(\theta) - \tilde{k}_1\theta^2\dot{\theta}$$

This corresponds to the last few lines in the routine. If the user supplies his own F(t), i.e. non-periodic then p must be zero.

Description

The routine is fairly straight forward, and self explanatory. If the user is supplying F(t), (the forcing function), then piecewise linear interpolation is used between parts.

Purpose

Subroutine to define the ship roll in the time domain by supplying the differentials.

Model used was developed by University of Newfoundland.

Specification

SUBROUTINE DPEQN2 (F, Y, TIME)

Description of Parameters

F - Real array of DIMENSION (2)
On exit the array F(I) contains the derivitives of Y(I) evaluated at the point TIME.

Y - Real array of DIMENSION (2)
On entry Y(I) contains the value of the Ith function at the point TIME.
Unchanged on exit.

TIME - Real
On entry TIME specifies the value of the independent variable at which the derivatives are required.
Unchanged on exit.

Theory

Consider a ship rolling in a long crested wave system, incident in the beam. The basic frame of reference is a fixed, right hand coordinate system (x,y,z), origin on the still water surface, with the ship moving forwards along the x-axis, with y-axis positive to starboard. Suppose for simplicity that the c. of g. lies on the load waterplane. Let the angle of roll be donated by ϕ. Suppose the waves are long in comparison to the ships beam so that the waveslope experienced at the ship is approximately that on the x-axis. If we assume that $\theta = \phi - \alpha$ and given that the waves have a maximum waveslope α_m then relative roll is given by:

$$\ddot{\theta} + k_1\dot{\theta} + k_3\dot{\theta}^3 + (1 - p\cos(\omega t+\delta p))R(\theta,\theta_s, H_o) = f\cos(\omega t +\delta)$$

$f(t)$

where

$f = \alpha_e\omega^2$, $\delta p = \delta+\delta p$

(further details of nomenclature can be found in 1)

If $F(2) = \frac{d(\dot{\theta})}{dt} = \ddot{\theta}$

and $F(1) = \frac{d(\theta)}{dt} = \dot{\theta}$

then we have two simultaneous, first order differential equations viz

$$\frac{d(\theta)}{dt} = \dot{\theta}$$

$$\frac{d(\dot{\theta})}{dt} = F(t) - k_1\theta - k_3\theta^3 - (1 - p\cos(\omega t + \delta p))R(\theta,\theta_s, H_o)$$

This corresponds to the last few lines of the routine.

--WARNING--

The user cannot supply his own F(t) with this model.

Error Indicators

None

Auxillary Routines

FRCPTN (TIMEPHASE)

XYFCN (TIMEPHASE)

Timing

Not applicable.

Storage

Accuracy

Not applicable

References

2 Bass, D.W.
'On the Response of Biased Ships in Large Amplitude Waves'
Int. Shipbuilding Progress Vol 30, January 1983.

Purpose

The routine advances the solution of a system of first-order ordinary differential equations

$$\frac{dy_i}{dx} = f_i(x,y_1,y_2, \ldots, y_N),\ i = 1,2,\ldots,N$$

from x to x + h, using a single step of Mersons form of the Runge-Kutta method.

The routine also gives estimates of the truncation error in each of the variables.

Specification

SUBROUTINE RKMRTS (Y,E,X,H,N,AUX,S,B,C,D)

Description of Parameters

Y - Real array of DIMENSION at least (N).
Before entry, Y must be set to the initial values of $Y_1, Y_2, \ldots, Y_N$.
On exit, it contains the computed values at the end of the step h.

E - Real array of DIMENSION at least (N).
On exit, E(1), E(2), ..., E(N) contain estimates of the magnitudes of the truncation errors (Y(1), Y(2),...,Y(N).

X - Real
On entry, X is set to the initial value of the independent variable x.
On exit, it contains x + h.

H - Real
On entry, H must specify the step-length h.
Unchanged on exit.

N - Integer
On entry, N must specify the number of differential equations.
Unchanged on exit.

AUX - External

Subroutine supplied by user, with specification

```
SUBROUTINE AUX (F,Y,X)
REAL F(n), Y(n), X
```

where n is the numerical value of N.

AUX must evaluate the derivatives of Y(1), Y(2),...,Y(N) at a general point X, and place them in F(1), F(2),..., F(N).

A - Real array of DIMENSION (N)

Used as work space.

B - Real array of DIMENSION (N)

Used as work space.

C - Real array of DIMENSION (N)

Used as work space.

D - Real array of DIMENSION (N)

Used as work space.

Theory

The theory is better detailed in the references but briefly, given a set of N differential equations, at a point x the derivatives are

$$\frac{dy_i}{dx} = F\ (x,\ y_1,\ y_2 \cdots y_N)$$

and the following terms can be found:-

$$K_1 = \frac{h}{3} \cdot F(x, y_1,\ y_2 \cdots y_N)$$

$$K_2 = \frac{h}{3} \cdot F(x + \frac{h}{3}, y_1 + k_1,\ y_2 + k_1, \ldots, y_N + k_1)$$

$$K_3 = \frac{h}{3} \cdot F(x + \frac{h}{3},\ y_1 + \frac{K_1}{2} + \frac{K_2}{2},\ y_2 + \frac{K_1}{2} + \frac{K_2}{2},\ \ldots)$$

$$K_4 = \frac{h}{3} \cdot F(x + \frac{h}{2},\ y_1 +$$

$$K_5 = \frac{h}{3}. F(x + h, y_1 + \frac{3}{2} k_1 - \frac{9}{2} K_3 + 6K_4, \ldots)$$

The finite-difference formula-

$$y_i(x+h) = y_i(x) + F(x + h, y_1 + \frac{3}{2} K_1 - \frac{9}{2} K_3 + 6K_4, \ldots)$$

$$\text{error} = 0.2 (K1 - \frac{9}{2} K_3 + 4K_4 - \frac{K_5}{2})$$

Description

The structure is straight forward and self-explanatory.

Error Indicators

Estimates of the local truncation error are given in E(1), E(2), ..., E(N) on exit. The user should test these quantities, and reduce the step-length if they are too large.

Auxillary Routines

AUX (DFEQN1 or DFEQN2)

Timing

This depends on the complexity of the system. The routine AUX is called five times for each step.

Storage

There are no internally declared arrays.

Accuracy

The truncation error per step is of order h^5 for sufficiently small h. The error estimates are not strict bounds, but they are fairly reliable over one step. Over a large number of steps the errors may accumulate in various ways depending on the system. A simple check on the accuracy over a range is to repeat the whole calculation with the step-length halved.

References

(1) LAMBERT, J.D.
'Computational Methods in Ordinary Differential Equations'
Wiley 1973 pp130-135.

(2) MAYERS, D.F.
'Numerical Solution of Ordinary and Partial Differential Equations' Fox, L(ed), Pergamon 1962 pp16-27.

Purpose

Function to find instantaneous values of the time domain function defined by H1...H3 and T1...T5.

Specification

REAL FUNCTION FRCFTN (H1, H2, H3, T1, T2, T3, T4, T5, X)

Description of Parameters

H1 - Real
Oneentry H1 specifies the initial level of the function that exist between T2...T2
Unchanged on exit.

H2 - Real
On entry H2 specifies the level of the forcing function between the times T3...T4
Unchanged on exit.

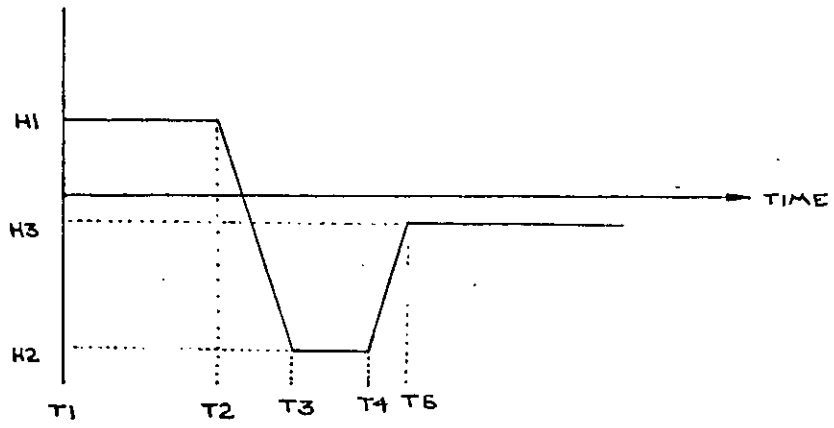
H3 - Real
On entry H3 specifies the level of the forcing function after the time T5.

T1...T5 Real
On entry T1...T5 specify the times at which the forcing functions level changes.
Unchanged on exit.

X - Real
On entry X specifies the time at which the value of the function is required.
Unchanged on exit.

Theory

The time domain function is assumed to behave in the following manner.



From the diagram it can be easily seen what H1...T5 represents.

Description

The routine simply tests for the interval that X lies in and then performs linear interpolation.

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable

Storage

There are no internally declared arrays.

Accuracy

The accuracy is dependent on how close X is to one of the points T1...T5.

References

None

4.4.6 TMHMSS

Description

Subroutine to determine the time domain response of the system from the differential equations.

Specification

SUBROUTINE TMHMSS

Description of Parameters

None

Theory

None

Description

The scratch file (NOUTR) is reset and the titles for use on the graphs are written to A, followed by a marker, which will indicate the end of the titles. The marker is the number - 10000.0.

The next section of code is fairly self-explanatory, and initalises the time domain solution.

The solution is conducted in the DO 30 loop. The appropriate model being selected via IF statements.

Should the solution be terminated early then two markers are written to the scratch file and control returned to the calling program.

-- WARNING --

The input/output to the scratch file has to be matched through a number of subroutines. Changes to formats or I/O statements regarding NOUTR must be undertaken with utmost care to avoid errors occurring. (This program assumes the scratch file to have 128 bytes/record).

Error Indicators

None

Auxillary Routines

DF (TIMEPHASE)

DFEQN1 (TIMEPHASE)

DFEQN2 (TIMEPHASE)

Timing

The timing is dependent upon GZORDR and NOSTEP.

Storage

There are 40 integer elements and 1459 real elements declared internally.

Accuracy

See 'DF'.

References

None.

Description

Subroutine to establish which plot to make.

Specification

SUBROUTINE TMMNU (IX, IY)

Description of Parameters

IX Integer

On exit IX specifies the users option for plotting on the X-axis. The options available are

IX = 0	;	Terminate program
IX = 1	;	time
IX = 2	;	frequency
IX = 3	;	maximum wave slope
IX = 4	;	effective wave slope
IX = 5	;	still water bias angle
IX = 6	;	relative roll angle
IX = 7	;	relative velocity of roll
IX = 8	;	instantaneous wave slope
IX = 9	;	absolute roll angle
IX = 10	;	forcing function

IY Integer

On exit IY specifies the users option for plotting. The options are as above.

Theory

None

Description

The structure of this routine is self-explanatory.

Error Indicators

None

Auxillary Routine

None

Timing

Not applicable

Storage

There are no internally declared arrays.

Accuracy

Not applicable

References

None

Purpose

Subroutine to read in the results of the time domain computation from the scratch file and plot out the appropriate options as dictated by the arguments of the routine.

Specification

SUBROUTINE TMPLOT (NX, NY)

Description of parameters

NX - Integer
On entry NX specifies the option for the x-axis. Its value is the same as returned by TMMNU. Unchanged on exit.

NY - Integer
As NX but for the y-axis.

Theory

None

Description

The scratch file is initially reset and the titles for the graph are read into 'ITITLS'. If the title does not correspond to the appropriate option then the array is overwritten when reading the next line.

If the 'ENDFLE' marker is encountered (see TMHMSS) then the next step of the routine is executed. Lines of results are now read in from the scratch file (NOUTR) and the values of the selected options are stored in DPTS. If the end of the file is encountered by the routine finding the ENDFLE marker, or DPTS is full then a call is made to XYPLTG (GLOBALS) to plot the results on the previously nominated plot device.

The next section of the routine plots experimental points onto the graph just drawn should thus be appropriate. Control is then returned to the calling routine.

****WARNING****

The input/output to the scratch file has to be matched through a number of subroutines. Changes to formats or I/O statements regarding NOUTR must be undertaken with utmost care to avoid errors occurring.
(This program assumes the scratch file to have a minimum of 128 bytes/record.)

Error Indicators

None

Auxillary Routines

XYPLTG (GLOBALS)	MOVTO2	(GINO-F)
	SYMBOL	(GINO-F)

Timing

Not applicable

Storage

There are 64 integer elements and 1462 real elements declared internally.

Accuracy

Not applicable

References

None

4.4.9 TMRDDT

Description

Subroutine to read in the data relating to the time domain simulation.

Specification

SUBROUTINE TMRDDT

Description of Parameters

None

Theory

None

Description

The structure is self-explanatory

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable

Storage

There are 40 integer elements and 1433 real elements.

Accuracy

Not applicable

References

None

Purpose

Subroutine for reading in data that represents experimental points, for plotting on the computed results graphs.

Specification

SUBROUTINE TMRDEX

Description of parameters

None

Theory

None

Description

The structure is self explanatory

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable

Storage

There are 40 integer elements and 1250 real elements declared internally.

Accuracy

Not applicable

References

None

Purpose

Subroutine to print results from a time domain analysis.

Specification

SUBROUTINE TMRSLT

Description of Parameters

None

Theory

None

Description

The routine is quite straight forward, it simply reads the titles for the graphs until the ENDFLE marker is found.

The routine then progresses to read in the results and print them out until the next ENDFLE marker is read in. Control then returns to the calling program.

WARNING

The input/output to the scratch file has to be matched through a number of subroutines. Changes to formats or I/O statements regarding NOUTR must be undertaken with utmost care to avoid errors occurring.

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable

Storage

There are 40 integer and 10 real elements declared internally.

Accuracy

Not applicable

References

None

Purpose

Subroutine to print out the user defined experimental points.

Specification

SUBROUTINE TMDTTE

Description of Parameters

None

Theory

None

Description

The structure of this routine is self-explanatory

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable

Storage

There are 40 integer elements and 1250 real elements declared internally.

Accuracy

Not applicable

References

None

Purpose

The routine advances the solution of a system of first-order ordinary differential equations

$$\frac{dy_i}{dx} = f_i\ (x, y_1, y_2, \ldots, y_N)\ ,\ i = 1, 2, \ldots N$$

from x to x + h, using a single step of trapezoidal integration.

The routine also gives estimates of the truncation error in each of the variables.

Specification

SUBROUTINE TRPXDL (Y, E, X, H, N, AUX, A, B, C, D)

Description of Parameters

Y - Real array of DIMENSION at least (N).
Before entry, Y must be set to the initial values of $Y_1, Y_2, \ldots Y_N$.
On exit it contains the computed values at the end of the step h.

E - Real array of DIMENSION at least (N).
On exit, E(1), E(2), ..., E(N) contain estimates of the magnitudes of the truncation errors in Y(1), Y(2), ..., Y(N).

X - Real
On entry, X is set to the initial value of the independent variable x.
On exit, it contains x + h.

H - Real
On entry H must specify the step-length h.
Unchanged on exit.

N - Integer

On entry, N must specify the number of differential equations.

Unchanged on exit.

AUX - External

Subroutine supplied by user with specification,

```
SUBROUTINE AUX (F, Y, X)
REAL  F(n), Y(n), X
```

where n is the numerical value of N.

AUX must evaluate the derivatives of Y(1), Y(2),...,Y(N), at a general point X, and place them in F(1), F(2), ...,F(N).

A - Real array of DIMENSION (H)

Used as workspace.

B - Real array of DIMENSION (N)

Used as workspace

C - Real array of DIMENSION (H)

Used as workspace

D - Real array of DIMENSION (N)

Used as workspace

Theory

The theory is better detailed in the references, but briefly, given a set of N differential equations, at a point x the derivatives are -

$$\frac{dy_i}{dx} = F(x, y_1, y_2, \ldots y_N)$$

The integration is then conducted over the step length h to give

$$y_i(x + h) = y_i(x) + h^* (F(x) + F(x + h))$$

The error is calculated by repeating the calculation in two steps, and comparing the two solutions by subtraction.

Description

The structure of the routine is self-explanatory.

Error indicators

Estimates of the local truncation error are given in E(1), E(2), ..., E(N) on exit. The user should test these quantities and reduce the step-length if they are too large.

Auxillary Routines

AUX (DFEQN1 OR DFEQN2)

Timing

This depends on the complexity of the system. The routine AUX is called six times for each step.

Storage

There are no internally declared arrays.

Accuracy

The truncation error per step is of order h^3 for sufficiently small h. The error estimates are not strict bounds, but they are fairly reliable over one step. Over a large number of steps the errors may accumulate in various ways depending on the system. A simple check on the accuracy over the range is to repeat the whole calculation with the step-length halved.

References

1 BOWN G.C.
"TRAN:- A program for transient analysis of circuits"
Proc. IEE, vol. 122, no. 3, Mar. 1975, pp253-258.

Purpose

Subroutine to print coefficients of the time domain analysis.

Specification

SUBROUTINE TMWRTE

Description of Parameters

None

Theory

None

Description

The structure of this routine is self-explanatory.

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable

Storage

There are integer elements and real elements declared internally.

Accuracy

Not applicable

References

None

Purpose

Subroutine to construct a piecewise-linear approximation for a function Y(X) from a set of N points, and to use this approximation to give a numerical value for the function Y for a given value of X.

Specification

SUBROUTINE XYFCN (XDATA, YDATA, N, X, Y, IFAIL)

Description of Parameters

XDATA - Real array of DIMENSION (250)
On entry XDATA(I) represents, in ascending order, the values of the independant variable X.
Unchanged on exit.

YDATA - Real array of DIMENSION (250)
On entry YDATA(I) represents the values of the dependent variable Y(I) corresponding to X(I).
Unchanged on exit.

N - Integer
On entry N specifies the number of points in XDATA and YDATA. Unchanged on exit.

X - Real
On entry X specifies the value of the independant variable for which the approximate value of Y is desired.
Unchanged on exit.

Y - Real
On exit, Y is the estimate of the dependent variable.

IFAIL - Integer
On exit IFAIL specifies whether an error has occurred in the routine.

IFAIL = 0 ; no error
IFAIL = 1 ; N>250 or N<2
IFAIL = 2 ; X> XDATA(N)OR X< XDATA(1)

CHAPTER 5

GLOBALS

CHAPTER 5

GLOBALS

This chapter describes the detailed operation of the library routines that constitute 'GLOBALS' together with the relationships with other programs in the suite.

Procedural descriptions are in the order found in 'GLOBALS'.

5.1 Program Description

GLOBALS is a program within the suite SORACOS V3.2.
It consists of a set of routines that are called by the programs GZPHASE...TIMEPHASE.

GLOBALS constitutes a set of library routines, and is always executed in conjunction with whichever program is being run.

The language used is the same as specified for the whole suite, as is the peripheral requirements.

5.2 Program Structure

'GLOBALS' has no overall procedural structure, as it only contains library routines called by other programs.

5.2.1 Module Interelationship

CALLED BY / ROUTINE CALLED	GLOBALS -	GRID	GRSOO1	GRSELT	SPLINT	DRAAXI	XYPLTG	DEVSEL	WINDUP	WNDDWN	CLRSCN	BEEP	PLTC
GLOBALS-													
GRID							X						
GRSOO1				X									
GRSELT							X						
SPLINT													
DRAAXI							X						
XYPLTG													
DEVSEL													X
WINDUP													
WNDDWN													
CLRSCN													X
BEEP													X
PLTC													
GINO-F-													
T4OlO								X					
CC81								X					
CHAHOL						X	X						
CHASIZ						X	X						
CHAANG						X	X						
CHAARR							X						
CHAFIX						X							
CHAINT						X							
CHASWI								X					
DEVEND													
DEVICE								X					
LINTO2						X	X						
LINBY2		X				X							
MOVTO2		X				X	X						
SYMBOL							X						
SCALE2								X					
PICCLE							X						
M/C DEP.													
EXIT													
TlOU											X	X	

5.2.2 Labelled Common Block/Module Interrelationships

COMMON BLOCK \ MODULE	GLOBALS :	GRID	GRSOO1	GRSELT	SPLINT	DRAAXI	XYPLTG	DEVSEL	WINDUP	WNDDWN	CLRSLN	BEEP	PLTC
DATA1							X						
INPUT								X	X				X
PLMODE							X						
WHERE								X	X		X	X	

Labelled Common Block Variable Map

COMMON/INPOUT/NIN, NOUT, NINM, NOUTP3, NOUTP4,
NOUTP2, NOUTR, NOUTG, NOUTM, NOUTP1

NIN - Integer

NIN represents the fortran logical channel number of the input data file

Set in WINDUP (GLOBALS)

NOUT - Integer

NOUT represents the fortran logical channel number of the output file which is laterlisted on a line pointer.

NINM - Integer

NINM represent the fortran logical channel number of console input. (Terminal).

NOUTP3 - Integer

Not used in this version.

NOUTP4 - Integer

Not used in this version.

NOUTP2 - Integer

NOUTP2 represents the fortran logical channel number of the flat bed plotter for plotting output.

NOUTR - Integer

NOUTR represents the fortran logical channel number of the scratch data file, which is deleted when not required.

NOUTG - Integer

Not used in this version.

NOUTM - Integer

NOUTG represents the fortran logical channel number of the console output (Terminal).

NOUTP1 - Integer

NOUTP1 represents the fortran logical channel number of the tektronix terminal for plotting.

Labelled Common Block Variable Map

COMMON/PLMODE/MODE

MODE - Integer

MODE represents the plotting mode required, and can have two values

MODE = 1; plot lines
MODE = 2; plot symbols (+)

5 Labelled Common Block Variable Map

COMMON/WHERE/ATAMTE

ATAMTE - Logical

Represents the location of the program suite. When set to true, indicates that SORACOS is situated at AMTE(H).

Set in WINDUP (GLOBALS).

5.3 Operating Environment

The relationship between data files and peripherals, together with the programs required in the object module is dictated under the appropriate section for the main program being executed i.e. GZPHASE...TIMEPHASE.

Description

Subroutine for use with XYPLTG to draw a grid on the current graph.

Specification

SUBROUTINE GRID (XORGIN, YORGIN, XLEN, YLEN, XNINT, YNINT)

Description of Parameters

XORGIN - Real
On entry XORGIN specifies the x-coordinate of the graph origin in current device units (usually mm).
Unchanged on exit.

YORGIN - Real
On entry YORGIN specifies the y-coordinate of the graph origin in current device units (usually mm).
Unchanged on exit.

XLEN - Real
On entry XLEN specifies the length of the x-axis in current device units (usually mm).
Unchanged on exit.

YLEN - Real
On entry YLEN specifies the length of the y-axis in current device units, (usually mm).
Unchanged on exit.

XNINT - Integer
On entry XNINT specifies the number of intervals on the x-axis.
Unchanged on exit.

YNINT - Integer
On entry YNINT specifies the number of intervals on the y-axis.
Unchanged on exit.

Theory

None.

Description

The structure of GRID is self explanatory and straight forward.

Error Indicators

None

Auxillary Routines

MOVTO2 (GIND-F)
LINBY2 (GIND-F)

Timing

Not applicable.

Storage

There are no internally declared arrays.

Accuracy

Not applicable.

References

None.

Purpose

Subroutine for setting ranges or intervals on graphs to a set series i.e. 0.1, 0.15, 0.2, 0.25, 0.4, 0.5, 1.0.

Specification'

```
SUBROUTINE GRSOOl (VALUE, RNDDIR)
```

Description of Parameters

VALUE - Real

On entry VALUE is set to the variable to be rounded up or down.

On exit VALUE is set to its nearest discrete value (rounded up or down) that is of the order

0.1, 0.15, 0.2, 0.25, 0.4, 0.5, 1.0

i.e.

If VALUE = 64.375 on entry, on exit

VALUE = 50.0 ; RNDDIR = 0

VALUE = 100.0 ; RNDDIR = 1

RNDDIR - Integer

On entry RNDDIR specifies the rounding direction

RNDDIR = 0 round down

RNDDIR = 1 round up

Unchanged on exit.

Theory

None

Description

The structure of GRSOOl is fairly straightforward, suffice to say that 'BASE' is 'VALUE' but normalised.

Error Indicators

None.

Auxillary Routine

None.

Timing

Not applicable.

Storage

There are no internally declared arrays.

Accuracy

Not applicable.

References

None.

Purpose

Subroutines to set the range on a graph to tidy intervals.

Specification

SUBROUTINE GRSELT (XMIN, XMAX, NINT)

Description of Parameters

XMIN - Real
On entry XMIN specifies the minimum point of the graph axis.

On exit XMIN is round up or down to the order of 0, 1, 1.5, 2, 2.5, 4, 5, 10.

XMAX - Real
On entry XMAX specifies the maximum point of the graph axis.
On exit XMAX is set such that each interval is integer powers of

0.1, 0.15, 0.2, 0.25, 0.4, 0.5, 1.0

NINT - Integer
On entry NINT specifies the number of intervals required.
Unchanged on exit.

Theory

None.

Description

XMIN is rounded up or down depending on whether XMIN is positive or negative.

The interval is rounded up and XMAX is calculated from XMIN, NINT and the round up interval.

Error Indicators

None.

Auxillary Routines

GRSOOl (GLOBALS)

Timing

Not applicable.

Storage

There are no internally declared arrays.

Accuracy

Not applicable.

References

None.

Purpose

Subroutine to interpolate at a given point x=A from a table of function values Y_I evaluated at points X_I (I = 1, 2, ... N+1) e.g. fitting cubic spline functions.

Specification

SUBROUTINE SPLINT (N, A, X, Y, W, D, IG, VAL)

Description of Parameters

N - Integer
On entry, N must specify the number of intervals along the x axis; that is, there are N+1 data points along the x axis.
Unchanged on exit.

A - Real
On entry, A must specify the x co-ordinate of the point of interpolation.
Unchanged on exit.

X - Real array of DIMENSION at least (IG).
Before entry, X(I) must be set to X_I, the x co-ordinate of the I^{th} data point for I = 1, 2, 111, N+1.
The X(I) must be supplied in strictly ascending order, i.e.

X(1) <X(2)<... <X(N+1)

On exit, the contents of X are unchanged.

Y - Real array of DIMENSION at least (IG).
Before entry, Y(I) must be set to Y_I, the y co-ordinate of the Ith data point, for I = 1, 2, ..., N + 1.
On exit, the contents of Y are unchanged.

W — Real array of DIMENSION at least (IG)

D — Used as workspace.

IG - Integer

On entry, IG must specify a value not less than N+1 and not greater than 25.

Unchanged on exit.

VAL - Real

On exit, VAL contains the interpolated function value at the points x=A.

Theory

For a description of the theory see the references.

Description

None

Error Indicators

None

Auxillary Routines

None

Timing

The computation time is proportional to N.

Storage

There are no internally declared arrays.

Accuracy

The accuracy of the result depends on the choice of data values X_I and Y_I and the position of the interpolation points relative to these points.

References

(1) HAYES, J.G. (Ed.)
'Numerical Approximation to Functions and Data'
Athlone Press, 1970.

(2) HANDSCOMB, D.C. (Ed.)
'Methods of Numerical Approximation'
Pergamen Press, 1966.

Purpose

Utility routine for use with XYPLTG to plot out the axis, for a graph.

Specification

SUBROUTINE DRAAXI (XORGIN, YORGIN, RLENGTH, NINT, VBEG, VEND, IXORY)

Description of Parameters

XORGIN - Real
On entry XORGIN specifies the x-coordinate of the graph origin in current device units (usually mm)
Unchanged on exit.

YORGIN - Real
On entry YORGIN specifies the y-coordinate of the graph origin in current device units (usually mm)
Unchanged on exit.

RLENGTH - Real
On entry RLENGTH specifies the length of the axis to be drawn in current device units
Unchanged on exit.

NINT - Integer
On entry NINT specifies the number of intervals to be made on the axis.
Unchanged on exit.

VBEG - Real
On entry VBEG specifies the value of the origin in the users data units.
Unchanged on exit.

VEND - Real

On entry VEND specifies the value of the end of the axis in the users data units.

Unchanged on exit.

IXORY - Integer

On entry IXORY specifies the axis to be drawn.

IXORY = 0 ; y-axis

IXORY = 1 ; x-axis

Unchanged on exit.

Theory

None

Description

Initially the basic axis line is drawn

The 'DO 10' loop then draws on the tie marks.
The 'DO 30' loop moves the drawing point below each tie mark and writes the value of that point (scaled by a factor 'SIZE' .)

Finally the value of the scaling i.e. 'SIZE' is written at the end of the axis.

Error Indicators

None

Auxillary Routines

MOVTO2	(GINOF)	CHASIZ	(GINOF)
LINTO2	(GINOF)	CHAFIX	(GINOF)
LINBY2	(GINOF)	CHAINT	(GINOF)
CHAANG	(GINOF)	CHAHOL	(GINOF)

Timing

Not applicable.

Storage

There are no internally declared arrays.

Accuracy

Not applicable.

References

None.

Purpose

Subroutine to plot a graph of A4 size using the GIND-F subroutine library.

Specification

SUBROUTINE XYPLTG (NODPTS, DPTS, ITITLS, XMIN, XMAX, YMIN, YMAX, IFILE, IAUTO)

Description of Parameters

NODPTS - Integer
On entry NODPTS specifies the number of points to be plotted.
Unchanged on exit.

DPTS - Real array of DIMENSION (2,101)
On entry DPTS contains the x and y coordinates in user data units for plotting.
DPTS (1,I) is the Ith x co-ordinate and DPTS (2,J) is the Jth y co-ordinate.
Unchanged on exit.

ITITLS - Integer array of DIMENSION 2,12)
On entry ITITLS contains the titles for the x and y axis in 12A2 format.
ITITLS (1,I) is the Ith pair of characters for the x axis.
ITITLS (1,J) is the Jth pair of characters for the y axis.

If an element contains the characters 'C' then the rest of the title is stripped, when the message

PLOT OF AGAINST

is written. This allows the title to contain the units when written against the appropriate axis, but the units are removed for writing the graphs title.

XMIN - Real

On entry XMIN specifies the minimum value to be plotted on the x-axis in user data units.

Not used if autoscaling is selected.
Unchanged on exit, unless autoscaling was selected, then XMIN is set to the minimum used for the graph.

XMAX - Real

On entry XMAX specifies the maximum value to be plotted on the x-axis in user data units.
Not used if autoscaling is selected.
Unchanged on exit, unless autoscaling was selected, then XMAX is set to the maximum used for the graph.

YMIN - Real

As XMIN but for the y-axis.

YMAX - Real

As XMAX but for the y-axis.

IFILE - Not used in this version.

IAUTO - Integer

On entry IAUTO specifies if autoscaling is required. If it is the data is scaled so as to fit all the data on the graph, and for the axis to have 'neat' intervals.

IAUTO = 0; not selected
IAUTO = 1; selected

Unchanged on exit.

Theory

None.

Description

The routine assumes that the drawing device has already been nominated using GINOF.

After clearing the drawing area, a test is made for autoscaling, if this has been selected then the smallest and largest values of the function to be plotted are found.

These are then altered via GRSELT to obtain neat tie marks.

Having done this, the titles are transfered into the individual arrays 'IXTITL' and 'IYTITL'. Any characters after and including '(' are stripped away and replaced with blanks.

The x and y axis are drawn using DRAAXI and a grid to put on the graph using GRID.

The next block of code is self explanatory, where the titles are placed at the head of the graph.

The titles are then transferred, again, into 'IXTITLS' and 'IYTITLS', this time, retaining characters after and including '('. These titles are then written out next to the appropriate axis.

Finally the points are plotted out, depending on the value of 'MODE'.

Error Indicators'

None

Auxillary Routines

DRAAXI (GLOBALS)	CHAHOL (GINOF)	SYMBOL (GINOF)
GRID (GLOBALS)	CHASIZ (GINOF)	PIZLLE (GINOF)
	CHAANG (GINOF)	
	CHAARR (GINOF)	
	LINTO2 (GINOF)	
	MOVTO2 (GINOF)	

Timing

Not applicable.

Storage

There are 64 integer elements declared internally.

Accuracy

Not applicable.

References

None.

5.4.7 DEVSEL

Purpose

Subroutine to query user about plot device required and to set up as appropriate.

Specification

SUBROUTINE DEVSEL (IDEV)

Description of Parameters

IDEV - Integer

Theory

None

Description

The routine is self explanatory.

Error Indicators

None

Auxillary Routines

T4010	(GINOF)	DEVIZE	(GINOF)
CC81	(GINOF)	SCALEZ	(GINOF)
CHASWI	(GINOF)		

Timing

Not applicable.

Storage

There are no internally declared arrays.

Accuracy

Not applicable.

References

None.

Purpose

Subroutine for the setting up of certain global parameters used throughout the suite.

Specification

SUBROUTINE WINDUP

Description of Parameters

None.

Theory

None.

Description

The operation of this routine is self-explanatory.

Error Indicators

None

Auxillary Routines

None

Timing

Not applicable.

Storage

There are no internally declared arrays.

Accuracy

Not applicable.

References

None.

5.4.9 WNDDWN

Purpose

Subroutine to shut down the program environment. However, it is not used in this version.

Purpose

Machine dependent routine to clear the VDU screen, for use with a PRIME 550 system.

Specification

SUBROUTINE CLRSCN

Description of Parameters

None

Theory

None

Description

The structure of the routine is self-explanatory except that TlOU outputs an arcis character and the DO loop forms a delay for the screen to 'settle' after clearing.

Error Indicators

None.

Auxillary Routines

TlOU (M/C DEP).

Timing

Not applicable.

Storage

There are no internally declared arrays.

Accuracy

Not applicable.

References

None.

Purpose

Machine dependent routine to ping the 'bell' of the VDU, for use with a PRIME 550 system.

Specification

SUBROUTINE BEEP

Description of Parameters

None

Theory

None

Description

The structure of this routine is self explanatory except that T1OU outputs an ascis character.

Error Indicators

None

Auxillary Routines

T1OU (M/C DEP.)

Timing

Not applicable.

Storage

There are no internally declared arrays.

Accuracy

Not applicable.

References

None.

Purpose

Subroutine to control the plotting options.

Specification

SUBROUTINE PLTC (MENU, PLOT)

Description of Parameters

MENU - External
A user supplied subroutine of the format.

```
SUBROUTINE MENU (IX, IY)
```

This routine querys the user about the options for the x and y axis and sets IX and IY accordingly. If IX = 0 then its assumed that the program is then to be terminated.

PLOT - External
A user supplied subroutine of the format

```
SUBROUTINE PLOT (IX, IY)
```

This routine plots IX versus IY on the appropriate device.

Theory

None.

Description

On entry to PLTC, 'IDEV' is set to zero so the use will be queried over the plot device they require.

The rest of the routine is self explanatory.

Error Indicators

None

Auxillary Routines

CLRSON	(GLOBALS)	MENU	(MAIN MODULE)
DEVSEL	(GLOBALS)	PLOT	(MAIN MODULE)
BEEP	(GLOBALS)		
DEVEND	(GINOF)		

Timing

Not applicable.

Storage

There are no internally declared arrays.

Accuracy

Not applicable.

References

None.

AMENDMENT LIST

SECTION NUMBER	TITLE	DATE OF ISSUE	ISSUED BY

COPY CONTROL

NAME	LOCATION	
G. FÉAT	SHIP SCIENCE, UNIVERSITY OF SOUTHAMPTON	MASTER COPY
W.B. MARSHFIELD	AMTE(H), GOSPORT, HANTS	COPY
LIBRARY	SHIP SCIENCE, UNIVERSITY OF SOUTHAMPTON	COPY