AD A116589

TN NO: N-1630

(12)

TITLE: SEADYN USER'S MANUAL

AUTHOR: R. L. Webster* and P. A. Palo

*Consulting Engineer
Brigham City, UT 84302

DATE: April 1982

SPONSOR: Naval Facilities Engineering Command

PROGRAM NO: YF59.556.091.01.402

TECHNICAL NOTE

NAVAL CIVIL ENGINEERING LABORATORY
PORT HUENEME, CALIFORNIA 93043

Approved for public release; distribution unlimited.

METRIC CONVERSION FACTORS

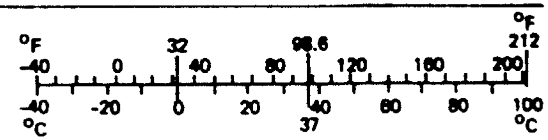
Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	*2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2,000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

*1 in = 2.54 (exactly). For other exact conversions and more detailed tables, see NBS Misc. Publ. 286, Units of Weights and Measures, Price $2.25, SD Catalog No. C13.10:286.

Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1,000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F



Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER TN-1630	2. GOVT ACCESSION NO. A116587 DN787011	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) SEADYN USER'S MANUAL		5. TYPE OF REPORT & PERIOD COVERED Final; Oct 77 -- Sep 81
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) R. L. Webster* and P. A. Palo		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS NAVAL CIVIL ENGINEERING LABORATORY Port Hueneme, California 93043		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62759N: YF59.556.091.01.402
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Facilities Engineering Command Alexandria, Virginia 22332		12. REPORT DATE April 1982
		13. NUMBER OF PAGES 170
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

*Consulting Engineer
Brigham City, UT 84302

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer program, cable dynamics, finite-element, nonlinear, mooring analysis.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the input structure and general use of the nonlinear cable dynamics computer model SEADYN. This program performs static, time and frequency domain dynamic, and modal analyses for arbitrarily configured cable-truss structures. Significant capabilities include: multi-materials, bottom interaction, nonlinear material properties, material damping, payout reel-in, strumming effects, spatially varying current

continued

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

20. Continued

fields, imposed motions, and random wave loading. Results can be saved and reused during the current execution or at a later date. A free-field input reader is used.

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	

Library Card

Naval Civil Engineering Laboratory
SEADYN USER'S MANUAL (Final), by R. L. Webster and P. A. Palo
TN-1630 170 pp illus April 1982 Unclassified

1. Computer program 2. Cable dynamics I. YF59.556.091.01.402

This report describes the input structure and general use of the nonlinear cable dynamics computer model SEADYN. This program performs static, time and frequency domain dynamic, and modal analyses for arbitrarily configured cable-truss structures. Significant capabilities include: multi-materials, bottom interaction, nonlinear material properties, material damping, payout reel-in, strumming effects, spatially varying current fields, imposed motions, and random wave loading. Results can be saved and reused during the current execution or at a later date. A free-field input reader is used.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

REVISION RECORD

Revision Set	Date	Description*	Pages

*Additional comments or descriptions can be made on NOTE pages in back of report.

FOREWORD

The process of learning to use a complex computer program is most often frustrating. Regardless of the care taken to document and explain the various features of the program, a new user is easily confused by unfamiliar terminology and descriptions of the methods employed. Usually, the developers have become so conversant with the problems addressed by the program and the way that they have interpreted and solved them, that it is difficult to communicate with the uninitiated. They simply forget how far they have come from the beginning. Often the appropriate beginning point can be found and quick learning obtained through dialogue between the developer or an experienced user and the novice. Unfortunately, the written page does not provide for such feedback. One should keep in mind that it is at least as difficult to write a manual that adequately describes a program as it is to learn to use the program.

PREFACE

The SEADYN cable dynamics computer program was developed over a 6-year period primarily by the Naval Civil Engineering Laboratory (NCEL) under the sponsorship of the Naval Facilities Engineering Command (NAVFAC). SEADYN is the largest of several cable analysis programs developed under the Large Displacement Cable Dynamics project. These programs have been validated by laboratory and at-sea experiments and allow for the confident analysis of arbitrarily configured cable structures subject to a wide variety of environmental and system loads.

This report is one in a series of five used to document the SEADYN computer program. This set of reports consists of:

(1) SEADYN User's Manual: Describes the program input format and external file requirements, and briefly discusses use of the program. Example inputs are presented.

(2) SEADYN Theoretical Models: Describes the finite-element formulation, implementation of particular submodels (strumming model, linearizations in the frequency domain solution, etc.), and numerical solution techniques (Ref 1).

(3) SEADYN Programmer's Reference: Describes SEADYN coding structure, logic, memory usage, and required system routines (Ref 2).

(4) SEADYN Test Cases Report: Includes detailed input and corresponding output for use in confirming the program's operation (Ref 3).

(5) SEAPLOT User's Manual: Describes input format for the graphics program that complements SEADYN's tabular output; requires the CDC-Cybernet® UNIPLOT graphics support package (Ref 4).

References 5 through 7 provide the following information and data on SEADYN-related topics: Data on cables, chain, and other SEADYN input parameters (Ref 5); a summary of the comparisons between SEADYN predictions and measured data (Ref 6); and a bibliography of NCEL reports published on cable dynamics (Ref 7).

The user is cautioned that only the cable portion of the program has been satisfactorily validated. The mooring-related options are still under development and are subject to change. Therefore, the use of SEADYN for problems involving vessel dynamics is not recommended at this time.

CONTENTS

CONTENTS (Cont'd)

CONTENTS (Cont'd)

1.0 INTRODUCTION

The SEADYN computer program simulates the responses of cable and truss type systems in an ocean environment. Such structures often appear deceptively simple. As a result the uninitiated is surprised that careful modeling is required to obtain a solution. Cable structures, even a simple catenary span, are highly nonlinear. Adding the offshore environment greatly compounds the problem. This computer program provides extensive capability for dealing with these problems and includes considerable flexibility and variety in the solution methods. But capability, flexibility, and variety have not been achieved without a price -- an increase in complexity.

SEADYN is a finite element program that employs simple line elements. These line elements can be used repetitively to describe long mooring lines, intricate truss structures, or complex cable systems. SEADYN includes lumped body models to simulate buoys, anchors, weights, etc. in the structure. These lumped masses are treated as point loads that represent weight/buoyancy and fluid drag effects. SEADYN also models dynamic rigid bodies that represent ships, barges, platforms, and mooring buoys. These are distinct from the previously mentioned lumped bodies since they provide for multiple line attachments and full six degree-of-freedom body response calculations. Two additional unique features of the program are the ability to represent variable length lines (payout/reel-in) and the component adequacy checks (tests buoys, anchors, and lines for potential underdesign).

The complicated nature of the nonlinear problems dealt with by SEADYN and the variety of numerical methods built into the program make it tempting to expound on the theories involved. Indeed, it is felt that a program such as SEADYN can be used intelligently only when the user has some fundamental knowledge of nonlinear mechanics and numerical solution methods. Attempts to treat such a program as a magic black box will inevitably lead to costly piles of paper full of useless numbers. Unfortunately, those who succumb to the black box syndrome are often unaware of erroneous or meaningless results since they seldom question what the computer has done. To such a user, a successful execution of the program with no diagnostic messages means correct answers. A successful solution of nonlinear equations requires the selection of appropriate solution procedures and the specification of incrementing and or iteration parameters. Default values can be provided when the user knows little else to do, but often the appropriate method and parameters are problem dependent. It is unrealistic to expect a single manual to tell all that is known on the subject. In the first place that is simply impossible; even if the body of knowledge is static much of the know how is obtained through experience and cannot be given as a set of rules or instructions. Secondly, the bulk and detail would be overwhelming to the novice, and much that is important would be diluted.

The approach taken here is to briefly outline the nature of the nonlinear problem, describe in simple terms the pertinent features of the nonlinear solution methods, present the input form for SEADYN, and illustrate the modeling process with some simple problems. Exposition of the governing equations and theoretical details is contained in Reference 1.

2.0 CAPABILITY OVERVIEW

The SEADYN program is designed primarily to analyze the static and dynamic responses of underwater cable and truss structures. The major features of the program are summarized in Table 2-1. The structural load paths are defined by straight lines between node points. The elements represented by these lines are assumed to have negligible bending resistance, and torsional effects are ignored. Thus the only load in each element is tension or compression. An option is provided to have compression treated as negligible. Should any element in the structure reach a displaced state representing a shortening of that element beyond its unstretched length, the stiffness is set to zero. The stiffness is restored when a stretched state is obtained. The physical limitations imposed by the water surface or the seafloor can be approximated by setting limits on specific nodes that the user expects to be involved. The positions of the nodes are continually monitored in nonlinear solution steps to activate or deactivate these tension-only element and limit conditions.

Two basic material models are provided to represent static load behavior. One uses a load-strain tabular format for specifying general nonlinear behavior. The other is a two-parameter model that presumes curve fitting of empirical data. The assumed form is $T = ae^b$. Material damping effects can be included through a one parameter (Kelvin damping) or a two-parameter (Reid-NOAA damping) viscoelastic model. A proportional damping option is also provided.

The fluid dissipative and loading effects on the structure can also be modeled. Reynolds number-dependent drag and constant added mass coefficients are provided for the line elements and lumped bodies. These program coefficients can be selected or the user can input his own via a subroutine. Dynamic fluid-solid interaction on rigid bodies is treated using drag and added mass coefficients, built-in potential theory solutions for mooring buoys in frequency domain analyses, and externally supplied data files for ships and platforms; only drag is considered in static analyses. An approximate static loading function is also provided for ships.

The solution formats in SEADYN can be assembled in varying order. Using the basic assumption that large deformations occur, the behavior at each motion state is dependent on its current configuration and possibly on the previous history. A staged or sequenced solution form can be used to apply loads and perform calculations. The configuration at the beginning of one stage (called a subanalysis) is automatically defined as the configuration reached at the end of the previous stage.

A typical sequence might be as follows:

Stage 1: Establish a structurally stable reference state. (Usually this represents the static state subjected only to gravity loads.)

Stage 2: Apply steady-state or static loadings and special boundary conditions (e.g., impose wind and current loads).

Stage 3: Apply dynamic loads and/or impose movement.

Stage 4: Recover a previously calculated state (RESTART option) and impose a new static loading.

Stage 5: Apply a new dynamic load sequence.

Stage 6: Calculate mode shapes and natural frequencies.

Stage 7: Continue dynamic analysis of Stage 5, etc.

The input data structure of SEADYN is segregated into a problem description set and a sequence of subanalysis data sets. Each subanalysis (Stages 1 through 7) is identified by a subanalysis option (SAO) flag. A list of the SAO flags and a brief description of their purpose follow:

DEAD - Nonlinear static analysis to apply gravity/buoyancy and specific point loadings.

LIVE - Nonlinear static analysis to apply fluid loads and various other load states.

DYN - Nonlinear dynamic analysis (time domain). Allows time variations in loads, currents, motions (towing, etc.), line lengths (payout/reel-in), and lumped body impact.

TSSS - Time Sequenced Static Solutions. Approximates nonlinear dynamics by neglecting accelerations and solves for configuration changes due to payout/reel-in and/or imposed boundary motion as a sequence of static nonlinear solutions. Generates a LIVE solution for each time step requested.

MODE - Calculates vibration mode shapes and natural frequencies for a linearized representation of the current state. This provides information only. The mode data are not used by any other subanalysis now defined in SEADYN.

FREQ - Frequency domain dynamic response using linearized frequency dependent solutions. Calculates regular wave response or response spectra at any node for specified wave spectra imposed on ships and mooring buoys.

CHEK - Evaluates the adequacy of various mooring components in the currently defined state.

The LIVE, TSSS, and DYN SAO sets allow for adjustment of drag loading to approximate the effects of line strumming. This is done in the form of response-dependent drag amplification factors that are periodically calculated for user-identified strings of elements.

The FREQ SAO set can use an implied LIVE SAO to approximate the change in static position due to wave-induced steady drift effects.

A restart capability is provided to allow the saving and subsequent use of any of the calculated states. A restart can be in the current run (as in the earlier example) or in some subsequent computer run. The restart files are also used as a data base for a separate post-processor graphics program SEAPLT, Reference 4.

SEADYN provides a minimal set of drag coefficients, time variation functions, current profiles, and ship-load functions. When these prove inadequate, alternate definitions can be provided through user subroutines and data files. Frequency domain dynamic equations are provided for mooring buoys and other bodies that can be approximated by a simple sphere immersed to the equator. More complex body dynamics can be described through an externally generated ship motion data file.

Table 2-1. SEADYN Capability Summary

General

- 3D large displacement response of cable and truss-type structures using the finite element method
- Can include lumped bodies, six degree-of-freedom rigid bodies (ships, platforms, etc.) and fluid-solid interactions
- Staged format, sequential analysis for statics and dynamics
- Treats nonlinear materials with internal damping, nonconservative loads, and nonlinear constraints

Special Features

- Variable length lines to simulate payout/reel-in dynamics
- Automatic estimation of drag coefficient amplification to approximate strumming
- Restart options
- Wave spectrum analysis with approximation for steady drift forces
- Component adequacy checks using design rules
- Plotting interface
- Free-field input format
- Catenary element for treating bottom interaction

Load/Boundary Conditions

- Gravity/buoyancy loads in water and air
- Arbitrary point loads and flow fields
- Wind and surface current loads on rigid bodies
- Built-in or user-supplied drag functions with flow/response-dependent amplification
- Arbitrary time variations (built-in or user-supplied)
- Moved boundaries
- Conditional constraints for surface and bottom limits

Static Solution Methods

- Sequence of linear increments (Euler's method)
- Residual feedback method (incremental self-correcting)
- Modified Newton-Raphson (various forms)
- Viscous relaxation method

Dynamic Solution Methods

- Nonlinear Transient
  - Sequence of Linear Increments
  - Newmark's β (residual feedback form)
  - Direct Integration Method (a multi-parameter predictor/corrector)
  - Time sequenced static solutions (quasi-static)
- Frequency Domain
  - (linearized with respect to the static state)
  - Mode shapes and frequencies
  - Response to wave spectra - Superposition of frequency-dependent steady-state solutions, fully coupled ship, buoy and line responses

3.0 THEORY OVERVIEW

The SEADYN program seeks to represent a general spatial arrangement of cable and truss components as a collection of simple elements. The present version of SEADYN allows two element forms: a straight line between two points (nodes), and a bottom-limited catenary. The straight line element is straight before and after deformation of the structure. It assumes negligible bending resistance and uses the instantaneous distance between the defining nodes along with the unloaded (unstretched) length of the element to estimate the strain. A single value for the element tension is then obtained from the tension/strain data given for that element. Since straight lines and constant tensions are low order approximations to the actual behavior of flexible lines, it is important that the analyst carefully consider how the structure is modeled. Regions where large curvature and/or highly variable tensions are expected should be approximated with more elements than in those regions where tensions vary slightly and lines are more nearly straight. In general, more elements mean more accuracy and more cost. Good modeling practice leads to a rational compromise between accuracy and cost. Achieving this compromise requires engineering judgment that comes from experience.

The bottom-limited catenary element uses classical catenary equations in an iterative procedure to develop stiffness and force relationships for a line which must interact with the bottom. Two nodes are used in the element definition, and adjustments are made in the equations to deal with line stretch and the amount of line lying along the bottom. Only gravity loads are applied to this element (current loads are neglected). Only approximate mass relations are provided for treating the dynamics of line pick-up and lay-down, so this element should be used with caution in transient dynamic solutions.

Since SEADYN deals with large deflection effects, the position and velocity of all nodes and the unstretched lengths of all elements must be considered in each step of the solution. This is referred to as a geometric nonlinearity, and it poses some problems not encountered in small displacement analyses. In static analyses, where inertia effects are ignored, it is possible to have a set of unstretched lengths and nodal positions that represent an unstable structural configuration. Unless the specified loads and solution procedures produce appropriate movements to modify the position of the nodes and reorient the elements, the structure represents a mechanism. This means that it is not capable of providing a static load path between the points where loads are applied and where they are supported. Another form of instability can also occur. This is the more familiar buckling instability in which an apparently stable structural configuration will suddenly deform to a radically different shape with only minor changes in loading.

The majority of numerical solution techniques used in SEADYN can be classified as initial value methods. This means that a solution step proceeds from a state where all pertinent data are presumed to be known to a state where estimates are made of the effects of loading changes using some sort of a predictor. When this estimate is within certain reasonable bounds of accuracy it can be improved by iterative corrections. In those situations where the initial state is not accurately described or where it represents an unstable state, the predictor is usually very inaccurate, if not undefined. SEADYN provides various means of dealing with this problem, but unfortunately there is no all powerful method that works every time with little or no special input. It is in dealing with this problem that the new user (and often the experienced user) will encounter the most frustration.

The various solution methods are discussed in the following paragraphs. The user will readily find the methods that work effectively on well-posed problems. In these cases more than one approach may be available, and the choice of which one to use can be based on economics, accuracy, or user experience. The discussion focuses on difficulties that can be encountered in highly nonlinear or poorly posed problems in the hope that the user can identify the difficulty and see how to adjust.

The first step in any analysis is to describe the initial geometric state. As noted above, this requires the specification of the length (stretched or unstretched) of every element and the spatial position of each node. The ideal is to have a consistent set of nodal positions and element lengths that describe an equilibrium state under a known loading. The natural way of describing such a state is to give the spatial coordinates of each node and the element tensions. Unstretched element lengths are then easily calculated from the strains (obtained from the tensions and material properties) and the stretched lengths (obtained from the nodal positions). When such a configuration description is available, the solution process can start simply by specifying a loading sequence that begins with the loads represented by the initial equilibrium state.

Most often, the analyst does not have an exact equilibrium state from which to start. Initial unstretched element lengths are known and the general form of the structural layout can be defined, but the correct nodal positions and tensions for a particular loading are not known. For simple structural forms it may be possible to guess a set of nodal positions that approximates the equilibrium state with the distance between the nodes representing the unstretched rather than the stretched lengths (catenary response, for example). Generally, even this approximate shape will be awkward to specify. It may be more convenient to list the unstretched element lengths independent of the positions of the nodes in the guessed configuration. In either of these situations the element tension data are extraneous because the correct tensions can be obtained only after the correct nodal positions are obtained. In the first case, the nodal positions imply unstretched lengths, which in turn means all the element tensions are zero. In the second case, the nodal positions imply guesses of the stretched lengths which, when taken with the specified unstretched lengths, imply a nonequilibrium set of tensions.

Neither of these approaches will give a stable state from which to proceed directly through an analysis sequence. An equilibrium state for the structural configuration must be found for some initial loading state. When the starting point is not an equilibrium state, the initial value solution methods fail; the predictor becomes inaccurate or undefined. This is true in static analyses because the predictor is expressed in the form of a tangent stiffness matrix that is singular unless the structure is stable. However, it is possible to construct a very approximate predictor for unstable systems using a set of artifical tensions. SEADYN provides for input of artificial tensions and/or a pseudo-tension generator called numerical damping. The predictors constructed in this manner can be very inaccurate, and the appropriate choices for the set of tensions and numerical damping parameters are highly problem dependent. Some trial solutions may be required to find the appropriate choices for a specific structure.

Various numerical solution schemes are provided that attempt to converge on the correct equilibrium state when the structure is initially unstable or nearly so. All of them rely on some combination of iteration and incrementing (applying the load in steps). The multiplicity of methods is maintained in SEADYN because no universally effective method has been found. Most problems can be handled quite well once the analyst has experimented with the solution parameters and the specific problem to identify and adjust to any sensitivities. One should plan to spend a little time becoming familiar with the initial responses in highly non-linear problems. The discussions in Sections 8 and 9 will be helpful.

Static Solution Methods

A brief description of the static solution methods follows. More detailed descriptions are given in References 8 and 9.*

Residual Feedback Method (RFB). This is an incremental, self-correcting procedure that presumes loads or imposed displacements are applied in a sequence of steps. The first step is a simple linear one. Each step after the first adjusts the load increment by an estimate of the equilibrium error from the previous step. The incremental stiffness matrix is recalculated at each step and reflects the nonlinearities apparent at the last completed step. This method is the least expensive to apply, the most numerically stable, and the least accurate. It requires explicit listing of the steps and performs only that number of steps with no iteration. It requires a stable state to start from, which can be artifically produced by fictitious tensions or initial numerical damping. The RFB method can fail if the incremental stiffness matrix becomes singular or very ill-conditioned after the first step. The final state can have large equilibrium errors since a full iteration to convergence is never done. More accurate solutions can be obtained by increasing the number of steps.

Modified Newton-Raphson Method (MNR). This method can be used in a fully iterative (single-step) form or an incremental-iterative form. The method iteratively evaluates the difference between the external nodal forces and the internal reactions to search for the displaced state that satisifies equilibrium. The tangential stiffness matrix is used to estimate the displacement changes for each iteration. This matrix can be recalculated at each step or at user-specified intervals. When the incremental form is used, the user must estimate the number of steps to be used. Depending on the rate of convergence at each step, the step size can be increased or decreased automatically by the program. Various schemes are used on the iterations to accelerate convergence.

*Another solution form is also in SEADYN. It is called a sequence of linear increments (SLI). This is essentially Euler's method and it requires very small steps to get accuracy. It is of no value in getting initial configurations, and its use is not recommended.

The first iteration at each load step can extrapolate from the converged state in the previous step. Fictitious tensions can be used in the starting state, and numerical damping can be employed on each step. The MNR method is conditionally stable. It can diverge if the predictor and convegence accelerators are inefficient or inappropriate. The convergent behavior can be obtained very slowly with excessive numerical damping. Convergence of the iterations is determined from simultaneous occurrences of small displacement changes and small values for the force residual (difference between external and internal forces). The user can specify the convergence tolerance, the frequency of tangent stiffness matrix calculations, the limits on iterations, and the characteristics of the extrapolations and convergence accelerators. Default values are taken when no selection is made.

Viscous Relaxation Method (VRR). This is a generalized form of the Newton-Raphson method, which can automatically adapt the characteristics of the solution steps to the behavior being sensed. An artificial time parameter is used to produce load steps or iterations. When applying the load in steps, a series of numerically damped load steps are produced that have a form similar to the RFB method.* When the full-load level is reached, the VRR method iterates while adjusting the damping level and pseudo-time step to move to the equilibrium state. As convergence is achieved, the method degenerates to the Newton-Raphson method with a stiffness matrix evaluation at each iteration. This is the most robust and the most expensive method. It is often capable of getting convergent solutions when the others fail. The VRR method can fail if there are excessive numbers of iterations or an inappropriate selection of control parameters. The methods used to adjust the solution characteristics are heuristically defined and can work well on some problems and be ineffective on others. The user can select the initial damping level, the initial step size, the number of starting increments, and the convergence tolerances. Default values are provided, but the appropriate choice of parameters is problem dependent. Inappropriate selections of damping can lead to very slow convergence or wildly oscillating behavior (cyclic calculations without convergence). Convergence is signaled by a very low value of the force residual or by the simultaneous occurrence of low force residual and small pseudo-velocities at all of the nodes.

Even when a stable structural state has been obtained, it is possible to develop solution instabilities while subsequently applying additional loads and/or movement. These problems result from physical instabilities in the structure (e.g., buckling) or from numerical instabilities in the solution. Often both situations occur together. In general one does not encounter the classical bifurcation buckling behavior exemplified by an Euler column. An instantaneous shift in deformation behavior is a small displacement abstraction of a large displacement nonlinear phenomenon. Incremental/iterative solutions would show a buckling instability as a large change in deflection for a small change in load. Graphically this would appear as a strongly curved load/deflection plot with a slope tending to zero. Analytically this means a portion of the stiffness matrix contains small coefficients relative to the rest of the matrix. Numerically this means the stiffness matrix is ill-conditioned (nearly singular).

---

*A parallel form which employs the SLI format is referred to as the VRS method.

In modeling cable structures (moorings in particular), one often encounters portions of the structure with stiffnesses much lower than the rest of the system. Lines carrying very low tension have very little stiffness in lateral motion even though their stiffness in stretching can be high. The angular motion components on moored vessels and buoys can have much lower stiffnesses than the linear motion components. This disparity in stiffness may not always signal a buckling-type problem, but it does produce numerical problems. The low initial tension problem is not a buckling problem unless compressive loads are applied. Loads that produce tension will stabilize the structure. (This is the initial state problem discussed earlier.) On the other hand, the buoy angular response problem can well be a buckling problem. There can be loading conditions that drive the buoy angular response to give large angle changes for small changes in load magnitude and/or direction. Or there can be limit states (which represent a sort of buckling) where the buoy position is unstable and tends to move into a state representing a twisting of the mooring lines. SEADYN makes no attempt to detect line crossings or related phenomena.

Attempts to converge on an equilibrium position when very low stiffness components are in the system can be costly (requiring many iterations and small steps). The buoy angular response and low tension systems are particularly troublesome. In both cases, the iterations can produce large changes in position, and convergence will be hard to obtain. The problem is compounded by numerical errors associated with solving the stiffness equations. A mixture of stiff and soft components leads to numerical ill-conditioning errors, which can be compounded by the sequence of the equation processing. This sequence is determined by the node numbering scheme in SEADYN. The ideal sequencing causes the soft components to be processed before the stiff ones. Unfortunately, one cannot afford the luxury of optimum ordering in this sense because it can greatly increase the equation bandwidth, thereby increasing the solution costs. Consult Reference 10 for more on equation ordering and solution errors.

Rigid bodies (mooring buoys, ships, platforms, etc.) are interfaced to cable elements using multiple attachment points. Nodes, which are used on the cable elements, have three degrees-of-freedom (one for each displacement component). The rigid bodies require six degrees-of-freedom (three displacements and three angular components). Slave/master constraints are imposed that imply the rigid body kinematics. A pair of nodes is used to define the six degrees-of-freedom for a rigid body and all other nodes on the body are required to move as though rigid links exist between them and the body node pair. The body node pair is the master, and all other nodes on the body are rigid link slaves.

Lumped bodies are assumed to have insignificant spatial dimensions so that their effect on the kinematics of the line is negligible. They are simply lumped at nodes where they produce mass and drag load effects.

Dynamic Solution Methods

The dynamic solution procedures in SEADYN fall into two categories: time domain and frequency domain. Time domain analyses are fully nonlinear. Two basic solution methods are available for numerical integration of the nonlinear time domain equations. Both solutions are based on a generalized form of the Newmark difference equations (Ref 8). At present there are no time domain rigid body equations available in SEADYN so that nonlinear time domain solutions for ships, platforms, and/or buoys are not possible. Master/slave relations are not functional in the time domain. A brief description of time domain solution methods follows.

Implicit RFB Method. This is an implicit integration scheme that follows the more traditional Newmark format of solving a set of simultaneous algebraic equations at each time step. Payout/reel-in and moving boundary options have not been implemented in this method. Specification of three integration parameters and the time step size is required. The method is strongly stable but can be inaccurate for large time steps.

Direct Integration Method (DIM). This is a predictor-corrector technique that does not require the formation of a stiffness matrix. Specification of three integration parameters and iteration controls is required. Time step size can be specified or calculated by the solution routine. The iterative corrector is conditionally convergent and requires strict upper bounds on the time step.

There are two types of frequency domain analyses in SEADYN. One of the frequency domain options is simply the estimation of vibration mode shapes and natural frequencies. It uses the well known Jacobi method. This option gives information about the dynamic character of the equations. The information is made available on a data file but is not used elsewhere by SEADYN. In the present version of the program, only a diagonal mass matrix is used, and no correction is made for the lack of a tangential added mass on the cable elements. All other dynamic options in SEADYN make this adjustment.

The other frequency domain option is a quasi-linear procedure that solves a set of frequency-dependent linearized equations for steady-state harmonic responses. These responses represent the fully coupled behavior of mooring components and a single-moored vessel subjected to wave forces defined through a wave-height spectrum. Spectral superposition techniques are used to estimate response spectrum data. The methods used assume that the motions involved are small amplitude perturbations about a nonlinear static reference state. The steady effects of wave-induced drift forces can be approximated by an iterative procedure that automatically generates a LIVE subanalysis. The LIVE subanalysis adds the drift forces onto the static reference state after a frequency spectrum pass has been made. The user has the option of limiting this procedure to a single pass or repeating it until no significant changes occur in drift forces and wave responses. See References 11 and 12 for theoretical details. SEADYN has frequency domain equations for the response of spherical mooring buoys. These equations assume that the water line is at the equator. It is assumed that the frequency domain equations for the ship or platform are available in a user-provided file that gives motion coefficients and loading functions versus wave frequency and heading. The format of this file is described in Appendix A.

4.0 INPUT DATA STRUCTURE

The SEADYN program uses a free-field input format. Information describing the problem and subanalyses is organized into blocks of data called records, terminated by column 80 or a terminating character (see Section 5). Examples of records are punched cards or lines separated by a return. General rules for generating SEADYN data records are given in Section 5, followed by specific descriptions. The data deck always begins with at least one title record. An unlimited number of title records can be used at the beginning of the deck. Later in the deck structure when the logic calls for a new title record, only one is to be given. Immediately following the title record set, SEADYN expects to find either a RESTART or PROBLEM data set. These provide a basic description of the problem either from a saved file or new data input. The PROBLEM data set consists of a variable set of records defining the geometry, materials, and various tables to be used in the problem solution stages. The data sets are identified by flag records or keywords. These keywords are limited to 10 characters in length, but only the first four are required.

The various solution stages are described in SAO data sets under the headings described previously. A termination of a string of solution stages is indicated by an END or NEW record. END terminates the run. NEW returns to the title record set (only one record this time) and expects to find RESTART or new PROBLEM directives. The basic program flow is represented in Figure 4-1. Representative input decks are included later in the examples. These decks provide useful references for the input instructions section.

The free-form input allows the data records for a ship loading file to be read in the fixed-field format. When required, this data can be included anywhere in the data deck after the first title record set. The input format is described in Appendix B.

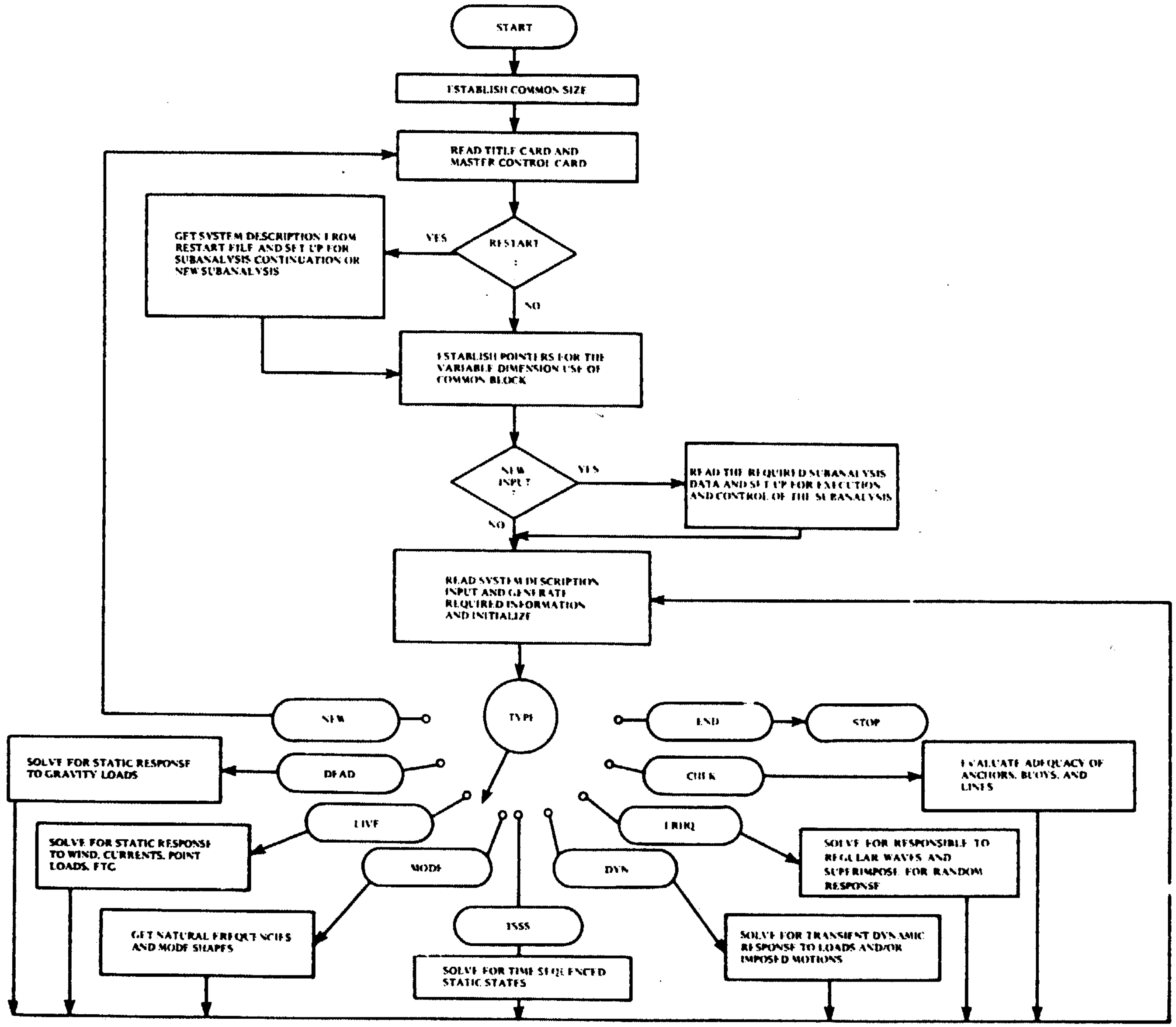


Figure 4-1. Macro flow chart of SEADYN.

5.0 FREE-FIELD INPUT RULES

The following special characters are recognized by the subroutine FREINP:

$	Record Terminator Flag Signals no more data to be read for the record image being processed. Multiple record images (records) can appear on a single record separated by record terminators. Double-record terminators signal the end of a record, and any data following this are treated as a comment. Comments will be listed as part of the record but will not be transmitted to the data file.
;	Alternate Record Terminator Flag Performs same function as $
COLUMN80	Default Record Terminator Unless a prior termination or continuation is signaled, the end of record (COLUMN80) is taken as a record termination.
,	Word Delimitor (Separator) Separates sequences of data entries in a record image. Repeated delimitors produce zeros in the words. An initial comma produces a zero in the first word of the record. All words not explicitly defined are assumed to be 0.0. A comma can be used to signal multiple records (continuation) when only blanks occur between the last comma and the end of record.
BLANK	Separator/Delimitor Leading blanks are ignored. Once the beginning of a word is detected, a blank will terminate the word. Any blanks following a delimitor are treated as leading blanks for the next word. The following are equivalent: xx yy xx , yy xx,yy
.	Continuation Flag Signals a word termination with the next word to be read from the next record. See "," for alternate continuation.
W	Word Position Flag Used to override the word sequencing and shift to a new word in the record. Input then follows in sequence from the new word location. The new word number is given immediately following the W and before the next "," or blank.

The W can be used as a delimitor of the previous word on all but the first word in a record. The combination ",W" is the same as W alone. The first word of a record is not checked for the W flag so ",W" must be used to skip to a new sequence from the first word position. Any W after the first word and before the record terminator will be interpreted as a position flag.

* **Comment Record Flag**
This character anywhere on a record will terminate it and the remainder of the record is treated as a comment. * in column 1 produces no record.

' **Hollerith (Alpha-Numeric) Field Delimitor**
This character is the apostrophe (11-8-5 on the 026 keypunch and 8-5 on the 029 keypunch). It signals the start and end of a character string. The string can be any length up to a maximum of ten-character "words" allowed by the application. This is usually 8, but it can be more or less as the use dictates. Any legal character can be used in the string except the apostrophe. Character strings can continue past column 80 to the next record since the string automatically signals continuation until the terminal apostrophe is found. All blanks in the string are counted as characters.

(**Fixed Format Initiator Flag**
This in column 1 of any record after the title record signals that the records up to the next ")" record are in fixed format. These records are written on a special data file in BCD format.

) **Fixed Format Terminator Flag**
This in column 1 signals the end of a sequence of rigid format records.

Any record with a "$", ";", or "*" in column 1 will be treated as a comment record. It will be listed but will not produce a data record.

Each free-field input deck is presumed to begin with one or more title records. Title records are read and listed until a specific record terminator is detected ($ or ; but not COLUMN80). The record on which the terminator is detected will be used as a page heading for the run.

The FREINP subroutine processes the entire input deck and translates it into a series of data records. As noted above, a data record can span more than one line or there can be one or more records on a line. After the title line, the data are assumed to be arranged in blocks headed by a flag record. Each flag record is limited to ten alpha-numeric characters. Only the first four characters of the flag record have meaning. For example, the flag record ELEMENTS could be shortened to ELEM to produce the same result. In those instances where the flag has only three characters, there must be a blank or word terminator after the third character. The specific data order applicable to the flag record is assumed until the next flag record is detected. Flag records must have the flag word in the first word position.

Data records are assumed to be in floating point form unless a character is detected that is inconsistent for a floating point number. In this case, the word will be treated as a Hollerith word. All floating point words assume a decimal at the end of the word if none is given. Words actually intended to be integers are converted to a fixed-point form at the time they are used by the program. The maximum length of a data record is determined by the program using the free-field subroutine.

The beginning and end delimitors for fixed format records must appear in column 1 of a distinct data record (individual record). This specifically requires that the previous record must have been appropriately terminated (no continuation). The ( ) delimitors are the only things read on that record, and the next data processed are assumed to be on the next record. Only one rigid format data set is allowed in any run. A rigid format data set cannot be input before the initial title record set is completed.

6.0 PROBLEM DEFINITION DATA

There are two ways of providing problem definition data: RESTART or a new PROBLEM data set.

RESTART presumes a previously generated restart file is available that contains all of the problem description data and the results of the SAO calculations. RESTART can be used to continue the previous SAO with some of the options and parameters changed, if desired. At the completion of this SAO, any other appropriate SAO can be requested. Alternatively, RESTART can be used to establish a starting state for a new SAO. The appropriate form for RESTART input is:

Title Record Set

RESTART

(Restart data)

SAO flag

(SAO data as needed)

END or NEW

The PROBLEM data set provides a complete description of all of the nodes, elements, bodies, etc., that describe the problem at hand. The order of the data records is:

Title Record Set

PROBLEM

(Problem data sets)

SAO Flag

(SAO data as needed)

END or NEW

The order of the problem data sets is not rigidly specified. Use the data sets as needed and follow logical sequences. For example: LINE requires start and end node data that must be defined by NODE and/or other LINE data. The data set descriptions are listed alphabetically by the flag record following the title, REST and PROB descriptions. A summary of flag records (four characters) follows:

Title

REST	Restart data
PROB	New problem definition
BLOC	Body locations
BODY	Define lumped body table
ELEM	Line or cable element definitions
FLOW	Flow-field library definitions
FLUI	Fluid media definitions
INVE	Modify component inventory
LIMI	Limit set definitions
LINE	Generate lines of nodes
LLOC	Limit locations
MATE	Material table definitions
NODE	Node point definitions
SHIP	Ship data definitions
STRU	Strum string definitions
TENS	Initial tension input
TFUN	Time function library definitions

The next sections present detailed instructions for inputting this information to SEADYN, including expected units where applicable (F = force, L = length, T = time).

6.1 Title Record Set

Any number of title records can be used to begin the data set. At least one is required. The last title record must be terminated with a record terminator (, or $). These record terminator characters cannot appear anywhere else in the title set.

Subsequent requirements for a title record set (e.g., following a NEW flag) must be limited to a single title record using the character string delimitors.

6.2 RESTART - Restart data record - Must immediately follow title records when it is used.

Word	Variable Name	Description
1	TYPE	Solution type being restarted (DEAD, LIVE, DYN, NEW). (Job will abort if DEAD, LIVE or DYN is given and does not match restart data)
2	NTAPE	File code number for the data file (1, 2, 3 or 4)
3	NFILE	The number of the restart record to be read from the data file. (Default is the last one written)
4	IRST	Restart file flag. (Same function as in SAO data, see Section 7.1.12 SAVE)
5	IDCHK	Identification check flag 1 - Read the title record from the file and compare the first 10 characters with the 10 characters given in CHKWRD (Word 15); abort if they do not match 0 - No identification check is made

(The data words beyond this point are not used if Word (1) is NEW)

Word	Variable Name	Description
6	NIPR	New value for number of steps between printing. If this is >0 and not equal to the value given previously, the output interval will be changed by the ratio NIPR (new)/NIPR (old)
7	IBG	New value for optional debug output flag
8	DTT	New step size
9	TMAX	New maximum time (default is old one)
10	GAMNEW	New integrator parameters (default is old values)
11	BETNEW	
12	ALPNEW	
13	DTOUT	New output time interval

Word	Variable Name	Description
14	DTRSRT	New restart save time interval
15	CHKWRD	Label check word (see Word 5), 10 characters

NOTES:

1. The data file referred to by NTAPE and the restart record referred to by NFILE must have been created on a previous SAO using the SAVE option.

2. TYPE (Word (1)) is used to signal the type of restart to be executed. NEW means the data on the file describes the beginning point of a new SAO set, and an SAO flag record is expected on the next input record. The DEAD, LIVE, and DYN types signal that the SAO data on the file is the same as the SAO form indicated, and that the previous SAO is to be continued using the indicated parameter changes. Subsequent SAO sets can follow after the completion of the restarted SAO. A restart to continue a TSSS SAO should have TYPE = "LIVE".

6.3 PROBLEM - New problem definition - Must immediately follow title records when it is used.

Word	Variable Name	Description
1	NN	Number of nodes in model
2	NE	Number of line elements in model
3	IDIR	Gravity direction code (see Note 1) 0 - no gravity loading ±1 - for X direction ±2 - for Y direction ±3 - for Z direction
4	JDYN	Dynamic solution option flag ≧0 - mass and acceleration storage will be allocated -1 - no dynamics terms calculated and storage truncated (DYN, MODE, FREQ not allowed)
5	IPRO	Input print option flag 0 - echo all input data with interpretation -1 - suppress input echo 1 - echo input plus print slopes and approximate transit times for each element
6	INDRAG	Drag model override flag (see Note 2) 0 - use default drag model regardless of flags on individual components ≠0 - user-defined drag models will be used as called for
7	GRAV	Gravitational acceleration (LT^{-2}) Defaults to 32.17 ft/sec^2 if IDIR ≠ 0 and GRAV not given
8	NSFILE	Ship load data file flag 0 - no data file >0 - ship load data for /NSFILE/ ships are provided as rigid format input in this run <0 - data file available with /NSFILE/ ships on it

NOTES:

1. It is assumed that the global axes are selected such that one of them coincides with the direction of gravity. Thus, IDIR = -2 means that gravity acts in the negative Y direction (i.e., +Y is up).

2. INDRAG can be used to select user-specified drag coefficients, since it is passed to the DRAGCO subroutine whenever it is called. If it is zero it overrides all other drag coefficient specifications and the program will use the built-in or default values described in Appendix C.

6.4 BLOC - Lumped/rigid body location specifications

Word	Variable Name	Description
1	LBODN (I)	Body number (see BODY table)
2	LBEG	Beginning node number for body location(s)
3	LEND	Ending node number for body location(s)
4	LINC	Node number increment
5	LIMSET (J)	Limit set number (same as in LLOC record)

NOTES:

1. Cylindrical buoys use the attached elements to define the orientation of the cylindrical axis. The first two connecting elements are used. The slopes of these two elements will be averaged to get the orientation of the cylinder. If only one element is connected to the body, its direction will be used.

2. Mooring buoys (rigid bodies) require a minimum of four nodes in their definition. The buoy location must be a two-node pair to give position and angle (six degrees-of-freedom). Two additional slave nodes must be tied to the buoy. These two nodes are automatically defined from the first two nodes (in numerical order) slaved to the buoy location node. Recall that slave nodes need not have lines attached to them. The location node for a rigid body is the first of the two node pair (the one that gives the position coordinates). When less than three lines are attached to a mooring buoy, the roll motion about the line between the first two slave nodes is assumed to be fixed to avoid equation singularity problems.

3. Input for Words (3) and (4) is not required if only one body is present.

4. No more than 10 lines can be connected to any body location with limits imposed on it with an LLOC or BLOC data set.

5. The present version of SEADYN does not allow rigid bodies (using slave and angle nodes) in the time domain solutions (DYN).

6.5 BODY - Defines lumped body table (A catalog of bodies available to the problem. See BLOC record to identify where the bodies are used.)

Word	Variable Name	Description
1	I	Body table number
2	IDRB (I)	Drag coefficient number 0 - use default drag (see Appendix C). ≠0 - use DRAGCO subroutine unless INDRAG = 0
3	ADM (I)	Body buoyancy (negative means weight) (F)
4	DBU (I)	Body diameter (L)
5	BLEN (I)	Body length (L) 0 - sphere or lump >0 - in-line cylindrical body
6	BAMC (I)	Added mass coefficient (Default for sphere = 0.5, for cylinder = 1.0)
7	BWND (I)	Wind drag coefficient for surface buoy (L^2) (C_D * A_s)
8	BSCD (I)	Surface current drag coefficient (L^2) (C_D * A_s)
9	BMOM (I)	Mass moment of inertia (FLT^2)
10	MEDMB (I)	Fluid medium in which ADM(I) is defined. See FLUID record (Default = 1).

NOTES:

1. Lumped bodies can be spherical or an in-line cylinder. These bodies provide only loads/mass to the nodes where they are located. They do not have stiffness terms, nor do they have wave-induced loads in the FREQ subanalysis.

2. Bodies used as mooring buoys cannot be cylindrical. They are treated as rigid bodies in static analyses and are assumed to be spherical bodies with the water line at the equator in FREQ subanalyses (see Note 2 on BLOC record).

3. Environmental loads on the body are determined from the relative velocity between the fluid and node, the dimensions and form of the body, and the drag coefficients (default or user defined).

6.6 ELEMENT - Line or cable element definitions

Word	Variable Name	Description
1	I	Element number
2	IT (1, I)	Node number for first end of element
3	IT (2, I)	Node number for second end of element
4		Not used at present but the word position must be accounted for.
5	MAT (I)	Material number
6	KOMP (I)	Type code 0 - element can not support compression 1 - element can support compression -1 - element is a catenary (see Note 4)
7	KSKP	Generation code (see Note 1)
8	ISIGO	Initial state flag (give on first element record only; see Note 2) 0 - equilibrium initial state with valid stretched lengths and tensions. Word (10) not used. 1 - compatible initial state defining unstretched lengths with dummy tensions given in Word (9). 2 - guessed deformed state. Unstretched lengths given in Word (10). Dummy tensions given in Word (9).
9	SIG (I)	Initial tension (see Note 3)
10	DSO (I)	Unstretched length (required for catenary element, see Note 4)
11	MEDIUM (I)	Fluid medium flag (the element is assumed to remain in the original medium at all times). See FLUID record (default is fluid 1).

NOTES:

1. Elements must be in ascending element number order, and the first and last ones must be input. Any omitted elements will be generated using the KSKP value on the element card following the omitted elements. Elements are incremented by 1; node numbers are incremented by KSKP. The element preceding the omitted elements is used to begin the increments.

For example: if element 1, defined by nodes 6 and 12, preceded the omitted elements and KSKP = 4, then the next element generated would be element 2 defined by nodes 10 and 16. The generated elements will be assumed to have the same properties as given by the element previous to the omitted ones.

2. The initial state flag (ISIGO) has three basic options:

ISIGO = 0: The preloads are presumed to represent an equilibrium state. The element lengths obtained from the nodal positions are used with these preloads and the constitutive relations to calculate the unstretched lengths. DSO(I), Word (10), is not required.

ISIGO = 1: The configuration represented by the nodal positions is an unstretched compatible arrangement of cables. The preloads (SIG(I)) are merely estimates. The unstretched lengths are obtained from the nodal positions.

ISIGO = 2: The configuration represented by the nodal positions is a compatible estimate of the deformed configuration. The values of the unstretched lengths are given on the element records. Element loads are calculated from the unstretched lengths, stretched lengths represented by the nodal positions, and the material constitutive relations.

3. When a string of elements lies along a catenary curve that has been generated by a node generation record (LINE record), the initial tension for each element can be obtained internally from the catenary generation using the TENS record. The initial tension will be taken to be the catenary tension half-way between the two nodes defining the element. SIG (I) is not input if the TENS record is used.

4. A catenary element is signaled by KOMP(I) < 0. The unstretched length input is required in this case. The catenary element uses the positions of the two nodes and the unstretched length to estimate stiffnesses and internal reactions. This is done by iteratively solving the classical catenary equations with adjustments for line stretch. Only the weight of the catenary is used in this iteration (current loading is neglected). The lowest node on the element is assumed to be at a bottom-limited condition. The line is assumed to lie on the bottom rather than to sag when the line length is sufficient. Note that the catenary line generation of the LINE record (Section 6.11) is a node generator and has no function relative to defining catenary elements. The unstretched length may be obtained from the LINE input data by entering a minus times the catenary number from the LINE record (see Note 4 of Section 6.11) for Word 10. Note: The present algorithm fails when the top node of a catenary element is on the bottom.

6.7 FLOW - Flow-field library definitions

Word	Variable Name	Description
1	I	Flow-field number (10 max)
2	IFLCOD (I)	Flow-field code (see Note 1)
3	FLPAR (1, I)	Optional flow-field parameters (see Notes)
4	FLPAR (2, I)	
	.	
	.	
	.	
12	FLPAR (10, I)	

NOTES:

1. Flow-field codes select a particular form of built-in function or signal a call to the subroutine CURUSR. Positive field codes refer to built-in functions, while negative codes refer to user subroutines. A request for a built-in function beyond those currently defined will cause an abort of the run. The parameters are used as appropriate for the built-in functions, and they are provided in the calling sequence to the user subroutines.

2. The user subroutine calling sequence is:

```
CALL CURUSR (T, N, NN, X, V, FLPAR (1, I))
```

where
- T = current time
- N = ABS (IFLCOD (I))
- NN = number of nodes
- X = nodal position vector
- V = nodal flow velocity vector

The nodal vector storage format is X (I, J) and V (I, J), where I = 1, 2, 3 for X, Y, and Z components, respectively; and J = node number.

3. The only built-in flow field at present is:

IFLCOD

1 - uniform velocity field

FLPAR

1 - X component of flow velocity (LT^{-1})
2 - Y component of flow velocity (LT^{-1})
3 - Z component of flow velocity (LT^{-1})

6 8 FLUID - Fluid media definitions (used only if IDIR ≠ 0 on PROB record). Must precede MATE and BODY records.

Word	Variable Name	Description
1	FDEPTH (I)	Coordinate at fluid surface
2	--	Property code
		0 - read properties from Words (3) and (4)
		1 - 1.30×10^{-5} ft^2/sec, 64.0 lb/ft^3 (seawater)
		2 - 1.68×10^{-4} ft^2/sec, 0.0765 lb/ft^3 (air)
3	FVISC (I)	Kinematic viscosity (L^2T^{-1})
4	FGAM (I)	Specific weight (FL^{-3})

NOTES:

1. Provide one record for each fluid in ascending order. The first fluid given is assumed to extend infinitely below its surface.

2. The FDEPTH for the highest fluid need not be given. It is assumed to extend infinitely above the previous fluid.

3. Pressure effects from the accumulated depths are ignored.

4. The present version of the program only allows two fluids.

6.9 <u>INVENTORY</u> - Units conversion for component inventory (see Appendix D for the inventory contents).

Word	Variable Name	Description
1	FRCVY	Conversion factor for weights and strengths (Default = 1.0)
2	FLNVY	Conversion factor for buoy diameters and lengths (Default = 1.0)
3	FDIVY	Conversion factor for line diameters (Default = 1.0)
4	INVPNT	Inventory Print Flag 0 - do not print inventory 1 - print all inventory entries

NOTES:

1. The value obtained from the inventory is multiplied by the corresponding factor to get it into the units implied by the problem input data.

2. Inventory data will not be used if this record is not input.

6.10 LIMIT - Limit set definitions (used only if IDIR ≠ 0 on PROB record)

Word	Variable Name	Description
1	LIMIT (I)	Limit set number
2	CORDLM (I)	Vertical coordinate of limit (L)
3	TOLIM (I)	Limit tolerance (L)
4	RELFAC (I)	Release factor (default = 1.001)
5	JANCR (I)	Fixity code 0 - buoyant limit (surface bound) with limit imposed only on vertical component 1 - weight limit (bottom bound) with limit imposed only on vertical component 3 - weight limit with all three components held when limit is imposed

NOTES:

1. Repeat records for as many limits as needed up to 50.

2. This data set defines a table of limit conditions. These are assigned to specific nodes using LLOC, BLOC, or possibly LINE records (see Note 3 of LINE record).

3. RELFAC gives the ratio of the sum of the vertical components of the line tensions to the external loads at the limited node that must be exceeded to release a limit in a transient (DYN) analysis. External loads include weight/buoyancy of the lines, bodies at the node, and any current and point loads. The release factor is always 1.0 in any static analysis.

6.11 LINE - Generates lines of nodes

Word	Variable Name	Description
1	NUMN	Number of nodes to be generated
2	NOLD	Beginning reference node (lowest node on a catenary)
3	I	Ending reference node (highest node on a catenary)
4	KSKP	Node numbering increment (default = 1)
5	NBC	Boundary condition flag 0 - do not copy constraint codes. Set all constraints to zero. 1 - copy constraint codes from node NOLD 2 - copy constraint codes from node I
6	NFIRST	Number of first generated node (default = NOLD + KSKP)
7	NCAT	Generation code 0 - straight line 1 - catenary with sag laid flat at limit (default = vertical coordinate of NOLD) 2 - catenary with unconstrained sag 3 - catenary as in 1 except no nodes placed on bottom
8	GCATW	Catenary weight per unit length (FL^{-1})
9	GCATH	Catenary horizontal component of tension (F)
10	LIMCAT	Limit set number for NCAT = 1 (see Word (1) of LIMIT record) may be used in place of the LLOC data for nodes generated on the bottom by the NCAT = 1 option (see Note 3)

NOTES:

1. Both the beginning and ending nodes must have been previously defined by NODE or LINE records. The beginning and ending nodes need have no numerical relation to the generated node numbers; only the spatial relation of the generated nodes is significant.

2. The value of KSKP can be positive or negative. Incrementing starts from NFIRST.

3. All generated nodes are evenly spaced on the generated straight line. Nodes generated on a catenary are placed to give uniform angle change, except for NCAT = 1 with line on bottom. If sag below the bottom is detected with NCAT=1, the tangent point is calculated. If the distance between NOLD and the tangent point is greater than 0.75 times the uniform spacing distance, a node is placed at the tangent point. If the distance is greater than 1.50 times the uniform spacing distance, NFIRST is placed half-way between the tangent and NOLD. The node at the tangent is then NFIRST + KSKP, otherwise the tangent is NFIRST. The remaining generated nodes are placed on a new uniform angle spacing from the tangent point to the last node. All nodes placed on the bottom are given the limit set number LIMCAT. If LIMCAT is not given, the LLOC (or BLOC) records should be used to set limits.

4. The catenary options provided here are simply for the purpose of generating nodes. They have no direct function relative to the generation of catenary line elements (see Section 6.6). Information needed to calculate the unstretched length of a catenary element can be passed through this node generation input. This is done by using NCAT = 3 and following Note 4 of Section 6.6. This information pass-through can be done even when no nodes are to be generated by setting NUMN = 0. This particular situation arises when a single element leg is desired and the horizontal component of pretension is known instead of the line length.

6.12 LLOC - Limit location specification

Word	Variable Name	Description
1	LIMSET	Limit set number (see Word (1) on LIMIT record)
2	LBEG	First node where limit is applied
3	LEND	Last node where limit is applied
4	LINC	Node number increment

NOTES:

1. The maximum number of limit locations is 50.

2. No more than 10 line elements can be connected to a limited node.

3. Generation of nodes on a catenary can cause generation of up to two limited nodes per catenary. This situation is described in Note 3 of the LINE record.

4. Body location data can also be used to locate node limits (see BLOC record).

6.13 MATERIAL - Material table definitions

Word	Variable Name	Description
1	I	Material number
2	IDRG (I)	Drag coefficient code 0 - use default drag coefficients (see Appendix C) ≠ 0 and INDRAG ≠ 0 - use DRAGCO subroutine
3	DIAM (I)	Cable or line diameter (L). Used only for fluid load computation.
4	G3 (I)	Weight per unit length (negative means buoyant) (FL^{-1})
5	MED (I)	Reference medium code. See FLUID record (default is fluid 1).
6	CAMC (I)	Added mass coefficient (default = 1.0)
7	TENULT (I)	Ultimate tension capability (F) (see Note 3 for use)
8	IE (I)	Option flag for tension (T)/strain (ε) properties 0 - use exponent form $T = a\varepsilon^b$ n - n points in tabular form (maximum n is 20) (see Notes 1 and 2)
9	TT (1, I)	First tension in table or a (F)
10	STR (1, I)	First strain in table or b
11	TTD (I)	Damping parameter CA_1 (FT) (see Note 4)
12	TTK (I)	Damping parameter EA_1 (F) (see Note 4)
13	TT (2, I)	Second tension in table (F)
14	STR (2, I)	Second strain in table
		(repeat pairs for all table points)

NOTES:

1. When point input is used and data are required outside of the given values, the slope of the last (first if only one) segment is assumed to be continued.

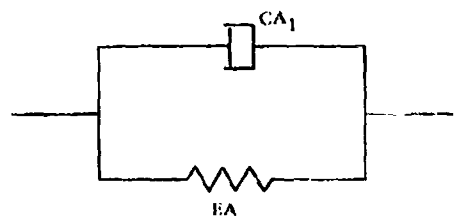
2. Input must be in increasing tension order. Thus, materials with compression stiffness must begin by listing the largest compressive load first and progressing to the largest tensile load.

3. The ultimate tension input is used only by the frequency domain dynamic solution. When TENULT (I) is greater than zero, the random estimates for tensions will be factored and compared to the ultimate tension.

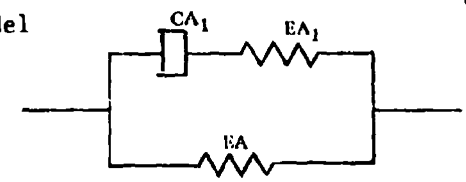
4. There are two material damping models provided:

KELVIN Model

EA is provided in the tension/strain data
CA_1 is the dashpot coefficient



NOAA-REID Model



EA_1 is an additional stiffness parameter. Additional information regarding the applicability and use of these models can be found in Reference 13.

6.14 NODE - Node point definitions

Word	Variable Name	Description
1	I	Node number
2	KSKP	Node generation
3	XO (3*I-2)	X } Global coordinates of node I (L for position, dimensionless or degrees for angles)
4	XO (3*I-1)	Y
5	XO (3*I)	Z
6	NODFIX (3*I-2)	X } Displacement constraint codes (see Note 2)
7	NODFIX (3*I-1)	Y
8	NODFIX (3*I)	Z
9	DFLAG	Angular coordinate flag -- "DEGREE" - means the nodal coordinates of angle nodes are input in degrees. Otherwise angles are in radians (see Note 1).

NOTES:

1. Each node is assumed to require three components to describe its position relative to a global right-handed cartesian system. Rigid bodies requiring six degrees-of-freedom (ships and mooring buoys) use two consecutive nodes. The first gives position and the second gives angular coordinates. All angular computations are done in radians, and angular output is in radians. Input of initial angular positions can be either in radians or degrees as indicated by DFLAG. No equations are available for rigid bodies in the time domain solutions (DYN).

2. Constraint codes define the active and constrained components of movement.

- 0 - no constraint (free)
- 1 - no movement (fixed), subject to release by limit set conditions
- -N - means the node is slaved to node N
- 2 - identifies a component reserved for payout that will be unconstrained (free) when activated. (A flag of 1 remains in effect on a node activated by payout.) Cannot be freed by a limit condition.
- 3 - identifies a component that is fixed and cannot be freed by a limit set condition.

3. Slave nodes must be numbered last. This means the node numbers of slave nodes must be larger than any of the active and master nodes. The input routine counts the slave nodes and deducts that number from the total nodes (NN) to get the number of active nodes. Slave nodes need not have lines connected to them. Slave nodes cannot be used in time domain solutions (DYN). Master nodes are assumed to be a node pair (see Note 1) since the slave nodes implies angular responses.

4. A node with any or all constraint codes set to 1, 2, or 3 is still considered an active node even though it cannot move. It is assumed that any of the constrained components can be modified by FREE, MOVE, or PAYOUT. LIMIT checks will affect only components with a constraint code of 1 or 0.

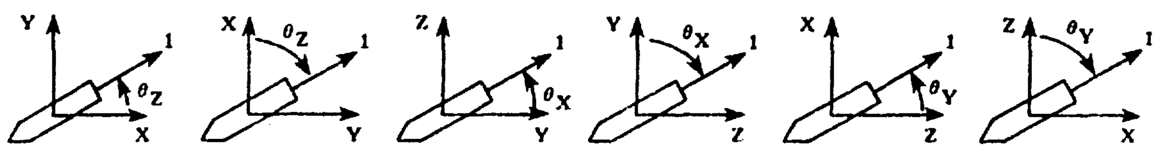
5. Input of nodes need not be in numerical order. Nodes can be entered more than once, and the last entry will be the one used in the solution. Omitted nodes can be generated using a straight line with uniform spacing with the aid of the KSKP parameter. If KSKP ≠ 0 on a node record, then that node is designated as the last node on the line, and the one input preceding it is designated as the first node on the line. Nodes are generated evenly along the line between these two points with node numbers incremented by KSKP from the first to the last node. The difference between the node numbers at the ends of the line must be an integer multiple of KSKP, and KSKP cannot be negative. The generated nodes will have the same constraint codes as the first node on the line. More general node generation schemes are available using LINE.

6. All of the NN nodes must be accounted for in the combined specifications for nodes and generation schemes.

7. Slave nodes need not have elements connected to them. This allows the investigation of the response of specific points on a ship at locations other than the attachments of mooring and working lines.

8. All points of attachment of lines to rigid bodies, such as ships or mooring buoys, must be slaved to the primary node defining the body. The program imposes no limit to the number or relative locations of these attachments, except that the frequency domain solution for mooring buoys uses the first two attachments to define the local coordinate system for the motion equations. These two attachments are also used for reference in the CHEK option (see Note 2 on BLOC record).

9. The initial orientation of a ship relative to the global coordinate system must be input. The angle required in this definition can be seen in the following sketch. (The global axes are X, Y, Z and the local ships axes are 1, 2, 3, representing aft, starboard, and up, respectively.)



6.15 SHIP - Ship data definitions

Word	Variable Name	Description
1	ISHIP (I)	Node where ship is located
2	LSHP (I)	Load function option -1 - search file for equivalent ship 0 - use analytical load functions n>0 - use n^{th} ship from loading file
3	CPROP (I)	Propeller resistance coefficient (default = 1.00)
4	CR (I)	Longitudinal resistance coefficient for hull to be used only for analytical functions (default = program calculates one, see Note 5)
5	CS (I)	Hull wetted surface coefficient to be used only for analytical functions (default = 2.70)
6	CMS	Amidships coefficient to be used to calculate C_R for analytical functions (default = 0.98)
7	SLT (I)	Total length of ship (L)
8	SAE (I)	End projected wind area (L^2)
9	SAS (I)	Side projected wind area (L^2)
10	SLWL (I)	Water-line length (L)
11	SBEAM (I)	Beam at midships (L)
12	SDRFT (I)	Draft at midships (L)
13	SDSP (I)	Volume displacement (L^3)
14	APROP (I)	Propeller projected area (L^2)
15	FSFRW (I)	Load table wind force conversion factor (default = 1.0)
16	FSFRC (I)	Load table current force conversion factor (default = 1.0)
17	FSLEN (I)	Load table length conversion factor (default = 1.0)
18	FSVEL (I)	Load table velocity conversion factor (default = 1.0)

Word	Variable Name	Description
19	SHIPK (1,I)	Heave restoring coefficient (default = 10^{22})
20	SHIPK (2,I)	Roll restoring coefficient (default = 10^{22})
21	SHIPK (3,I)	Pitch restoring coefficient (default = 10^{22})
22	SHIPK (4,I)	Heave/pitch restoring coefficient (default = 0)
23	WDEPTH (I)	Water depth at ship location (default = 10 times draft) (see Note 6)

NOTES:

1. Linearized ship restoring coefficients can be input if desired. Otherwise the ship will be assumed fixed in heave, roll, and pitch during static analyses. During frequency domain analyses, the restoring matrix is obtained from the ship motion file.

2. Words (7) through (14) may be omitted if LSHP (I) = n and the moored ship is the same as on file n of the ship loading file (i.e., no similarity scaling required).

3. The conversions factors, Words (15) through (18), multiply the values of force, length, and velocity from the ship loading file to get values consistent with the units implied by the rest of the input data.

4. The local coordinate system for ships assumes the 1, 2, and 3 directions are aft, starboard, and up, respectively. This is consistent with the loading conventions of Appendixes B, F, and G.

5. When analytical loading functions are used, the longitudinal resistance coefficient can be input or calculated using a table look up. The total coefficient, CR (I), consists of three parts:

C_r = residuary coefficient

C_f = frictional resistance coefficient

ΔC_f = fouling resistance coefficient

The value for C_f is always calculated in the program using:

$$C_f = \frac{0.456}{(\log_{10} R_e)^{2.58}} - \frac{1700}{R_e} \qquad R_e \geq 5 \times 10^5$$

where R_e is Reynold's Number based on the longitudinal component of the velocity; or

$$C_f = 0.002 \qquad R_e < 5 \times 10^5$$

When CR (I) is input, it is taken to be $C_r + \Delta C_f$. When it is zero or no input is given, then C_r is obtained from a table and ΔC_f is assumed to be 0.0005.

6. The water depth is used only in the similarity scaling from tabular ship load data (see Appendix F).

6.16 STRUM - Strum string definitions

Word	Variable Name	Description
1	ISTRNG (I)	Number of elements in string (max 20)
2	KSTRNG (1, I)	List of string elements
3	KSTRNG (2, I)	Must be listed in sequence from one end of the string to the other. It does not matter which end is listed first.
	.	
	.	
	.	
31	KSTRNS (30,I)	

NOTES:

1. A string consists of adjacent (contiguous) elements to be considered in estimating drag amplification factors for strumming analyses. An element can be contained in more than one string. The drag amplification factor for each element listed in a string is the largest value it has in any string.

2. The maximum number of strings is 30. The maximum number of elements per string is 20.

3. Drag amplification estimates are made on LIVE, DYN, and TSSS subanalyses with nonzero relative fluid velocities whenever these strum strings are defined. The procedure assumes that the strings are supported at each end and an auxiliary vibration mode analysis is conducted on each string to estimate its possible involvement with vortex shedding-induced strumming. The general approach of Skop, Griffin and Remberg (Ref 14) is used. The occurrence of the drag amplification estimates is controlled by the changes in relative velocity and the CEPS parameter input on the SOLU record (Section 7.1.14) associated with the SAO data set.

6.17 TENSION - Initial tension input

Word	Variable Name	Description
1	LBEG	Beginning element number
2	LEND	Ending element number
3	LINC	Element increment
4	LCODE	Option code 0 - assigned initial tension, the value in Word (5) N - estimate initial tension from the N^{th} catenary defined on the LINE records (see Note 2)
5	SIG	Tension to be assigned to elements (F).

NOTES:

1. Catenary line tensions (LCODE = N) for the N^{th} catenary are calculated for each element based on the position on the catenary at the midpoint of each element (see LINE record and ELEMENT record, Note 3).

2. The TENSION record must not precede the LINE record which generates the catenary this data record refers to.

6.18 TFUNCTION - Time-function library definitions

Word	Variable Name	Description
1	I	Time-function number (20 max)
2	ITFCOD (I)	Time-function-type code (see Notes)
3	TPARM (1, I)	Optional time-function parameters
4	TPARM (2, I)	
	.	
	.	
	.	
22	TPARM (20, I)	

NOTES:

1. Time-function-type codes select a particular form of built-in function or signal a call to the subroutine TFNUSR. Positive-function codes refer to built-in functions, while negative codes refer to the user subroutine. A request for a built-in function beyond those currently defined will cause an abort of the run. The parameters are used as appropriate for the built-in functions, and they are provided in the calling sequence to the user subroutine.

2. The user subroutine calling sequence is:

```
CALL TFNUSR (T, F, N, TPARM (1, T))
```

where T = current time
F = the returned value of the time function
N = ABS (ITFCOD (I))

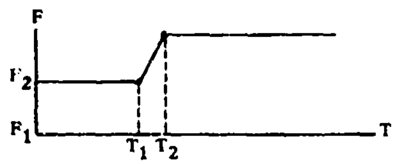
(See Appendix H.)

3. The built-in functions presently available:

<u>ITFCOD</u>

ITFCOD	Function	TPARM INPUT 1	2	3	4	5	6
1	Ramp Build-up/Decay	T_1	F_1	T_2	F_2	0	0
2	Shifted Sine	A	T_0	T_s	B	0	0
3	Ramped Sine	A	T_0	T_s	B	T_R	0
4	Decayed Sine	A	T_0	T_s	B	T_R	T_E

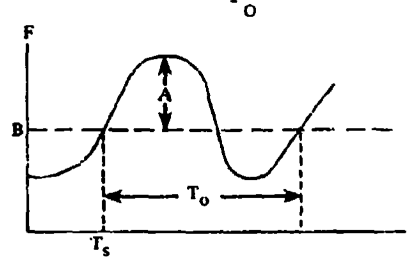
1 Ramp Build-up/Decay



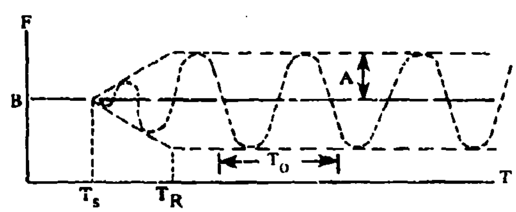
2 Shifted Sine

$$F = A \sin [\omega (T-T_s)] + B$$

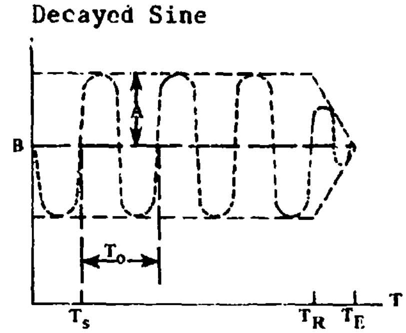
$$\omega = \frac{2\pi}{T_o}$$



3 Ramped Sine



4 Decayed Sine



ITFCOD

3 Tabular Input (max of 9 points plus end values)

1	F_o	Function value for times less than T_1
2	T_1	First time point
3	F_1	First time function value
4	T_2	etc.
5	F_2	
.	.	
.	.	
.	.	
18	T_9	
19	F_9	
20	F_B	Function value for times greater than last time

ITFCOD

4 Random File Input

TPARM

1	NOTAPES		Number of excitations (5 max)
2	TTZERO		Starting time on excitation tapes
3	SCAFACT	(1)	Multiplier
4		(2)	
5		(3)	
6		(4)	
7		(5)	
8	SHIPVEL	(1)	Velocities of excitation points
9		(2)	
10		(3)	
11		(4)	
12		(5)	

7.0 SUBANALYSIS OPTION (SAO) DATA

Each subanalysis is headed by an option flag record:

DEAD	static analysis with gravity and point loads
LIVE	static analysis with arbitrary combined loads
TSSS	time-sequenced static solutions (approximate dynamic analysis using LIVE that allows moved nodes, payout, and other time-varying loads but neglects inertia effects)
DYN	transient nonlinear dynamic analysis (time domain)
FREQ	frequency domain dynamic solution (response spectrum analysis) for wave excitation
MODE	determination of natural frequencies and mode shapes for current position
CHEK	component adequacy checks
END	run termination
NEW	define new problem -- title card read next

An SAO flag record has the flag name as Word (1) with no other data in the record. SAO data are grouped following the SAO flag to identify solution characteristics, loading, boundary conditions, and output requests. Appropriate default values are assumed when no data are given. Unless explicity stated, the data records have no required order, since each data set is identified by a keyword.

7.1 DEAD, LIVE, TSSS, DYN Data Set

The data set description for DEAD, LIVE, TSSS, and DYN are given in alphabetical order of the keyword:

Keyword	
CURR	flow-field specification
FIX	applies temporary fixed conditions to nodes
FREE	releases previously imposed fixity
IMPA	impacting body input
INIT	describes dynamic initial conditions
KEEP	retains the data defined in the preceding SAO for this SAO
LOAD	specifies load conditions
LVAR	specifies load variation codes
MOVE	specifies displacement/velocity/acceleration at nodes
OUTP	selects output data
PAYO	defines payout/reel-in of lines
SAVE	defines restart file save intervals
SBUO	specifies solution option characteristics
SOLU	solution option characteristics
STEP	solution step size data
SURF	surface current data
TIME	time step data
WIND	surface wind data

7.1.1 CURRENT - Flow-field specification

Word	Variable Name	Description
1	OPTION	"CURR"
2	NFLUID	Flow-field number from library defined in FLOW record (see Notes)
3	CURMUL	Flow-field scale factor (multiplier for the values obtained from the flow library; default is 1.0)
4	NFLVRY	Flow-field variation code For LIVE SAO 1 - increase flow field incrementally 0 - hold flow field at full amplitude -1 - decrease flow field incrementally For DYN and TSSS SAO 0 - hold flow field amplitude constant or use the time variation implied by the CURUSR subroutine >0 - use n^{th} time variation function from TFUN record

NOTES:

1. When NFLUID ≧ 0, the flow field will be evaluated each time the fluid loads are calculated. The flow velocity obtained from the requested flow field (see FLOW library, Section 6.7) will be multiplied by CURMUL and the appropriate time variation code for DYN and TSSS. Note that for LIVE subanalyses, the variation code multiplies the fluid loads instead of the fluid velocity.

2. When NFLUID < 0, the flow field will be evaluated only once during the SAO.

7.1.2 FIX - Applies fixed conditions to nodes

Word	Variable Name	Description
1	OPTION	"FIX"
2	ICODE	Constraint code (default = 1, see Note 2)
3		List of node component codes CODE = NODE*10+I where I = 1, 2, or 3 for X, Y or Z direction, respectively
4 (etc.)		Maximum list length is 98 (use additional FIX records to get more)

NOTES:

1. FIX causes the selected code to be placed in the constraint code array (NODFIX) for the indicated component. The component will remain fixed until released by a FREE record or a predefined LIMIT record.

2. Allowable constraint codes:

   1 - component fixed unless limit condition overrides.
   2 - component fixed and reserved for payout. It will become free when activated by a payout mitosis* (see PAYO record, Section 7.1.11). It cannot be released by a limit condition.
   3 - component fixed and cannot be released by a limit condition.

3. The MOVE record will override any constraint code.

CAUTION: Do not FIX any components of a slave node.

*Mitosis refers to the halving of an element into two new separate elements; the original element length is replaced by two elements (see Reference 1 for further details).

7.1.3 FREE - Removes fixed conditions for nodes

Word	Variable Name	Description
1	OPTION	"FREE"
2		List of node component codes CODE = NODE*10+I where I = 1, 2 or 3 for X, Y, or Z direction, respectively
3 (etc.)		Maximum list length is 99 (use additional FREE records to get more).

NOTES:

1. FREE causes a 0 to be placed in the constraint code array (NODFIX) for the indicated component. The component will remain free until reset by a FIX record or a predefined LIMIT record.

CAUTION: Do not FREE any components of a slave node.

7.1.4 IMPACT - Impacting body

Word	Variable Name	Description
1	OPTION	"IMPA"
2	IMPNOD	Node where impact occurs
3	IMPBOD	Body number (see BODY data records)
4	VIB (1)	V_x } Components of body velocity at impact
5	VIB (2)	V_y } Components of body velocity at impact
6	VIB (3)	V_z } Components of body velocity at impact
7	IOPT	Weight option 0 - floating body, no weight added, only adds inertias (e.g., iceberg or snagged vessel) 1 - new lumped body added to the system

7.1.5 INITIAL - Describes dynamic initial conditions

Word	Variable Name	Description
1	OPTION	"INIT"
2	VI (1)	V_x } Components of nodal velocity
3	VI (2)	V_y } Components of nodal velocity
4	VI (3)	V_z } Components of nodal velocity
5	NBEG	Beginning node number
6	NEND	Ending node number
7	NINC	Node number increment

NOTES:

1. If Word (5) is zero or is not given, the velocity components are assumed to be on all nodes in the system. In this case, the last INIT record encountered is the one used.

2. When Word (5) through Word (7) are not zero, the velocity components are assigned to the individual nodes. Repeat the INIT records as many times as needed to define all nodal velocities.

7.1.6 KEEP - Retains the data from the preceeding SAO

Word	Variable Name	Description
1	OPTION	"KEEP"

NOTES:

1. This flag causes the data initialization to be bypassed, thereby allowing the preceding SAO data to remain in effect. Only those data to be changed need be entered.

2. KEEP must be the first flag encountered in the SAO data set if it is used.

7.1.7 LOAD - Specifies load conditions

Word	Variable Name	Description
1	OPTION	"LOAD"
2	LSET	Load set number (3 max, default = 1)
3	FP (LSET,I)	F_x } Point load components (F or FL)
4	FP (LSET, I+1)	F_y } Point load components (F or FL)
5	FP (LSET, I+2)	F_z } Point load components (F or FL)
6	NBEG	Begin node number
7	NEND	End node number
8	NINC	Node number increment (default = 1)

NOTES:

1. Loads are assigned as members of up to three load sets according to LSET. The LVARY record file is then used to identify the characteristics of each load set.

2. Load components are placed in the point load array FP by load set number and node numbers. Word (6) must always be given. The same loads can be applied to a sequence of nodes by giving appropriate values for Word (7) and Word (8).

3. If the node is used for the angular position of a rigid body, the load is assumed to be a moment about the specified axis (units FL). Otherwise the loads are point forces (units F).

7.1.8 LVARY - Specifies load variation codes

Word	Variable Name	Description
1	OPTION	"LVAR"
2	ILF (1)	Variation code for load set 1
3	ILF (2)	Variation code for load set 2
4	ILF (3)	Variation code for load set 3
5	JDLD	Gravity load variation code for LIVE subanalyses (see Note 2)

NOTES:

1. Variation Code:

For DEAD and LIVE

1 - increment the load set factor from 0 to 1.0 (apply load)
0 - hold the load set factor at 1.0 (steady load)
-1 - increment the load set factor from 1.0 to 0 (remove load)

For DYN

0 - hold the load set factor at 1.0 (steady load)
I>0 - use the Ith time variation function from the TFUN record set to get the load set factor

2. The only loads used in a DEAD SAO are the internally calculated gravity loads and point loads defined in LOAD record data sets. The gravity loads are assumed to have a +1 variation code in the DEAD SAO. The LIVE SAO assumes a variation code of 0 for gravity loads. When a LIVE SAO has not been preceded by a DEAD SAO it may be advantageous to increment the gravity loads along with the LIVE loads. Setting JDLD to +1 will accomplish this.

7.1.9 MOVE - Specifies node point motion

Word	Variable Name	Description
1	OPTION	"MOVE"
2	INOD	Node number to be moved
3	MC (1)	X motion code
4	MVC (1)	X motion variation code
5	AMP (1)	X motion amplitude
6	MC (2)	Y motion code
7	MVC (2)	Y motion variation code
8	AMP (2)	Y motion amplitude
9	MC (3)	Z motion code
10	MVC (3)	Z motion variation code
11	AMP (3)	Z motion amplitude

NOTES:

1. Motion codes:

   0 - no motion specified (see Note 4)
   1 - displacement
   2 - velocity (DYN and TSSS only)
   3 - acceleration (DYN and TSSS only)

2. Motion variation codes:

   For DEAD and LIVE

   1 - increment motion factor from 0 to 1.0 (apply deflection)
   0 - hold the motion factor at 1.0 (hold deflected position)

The motion factor is the multiplier of the motion amplitude for the increment level (load factor) at the current step.

   For DYN and TSSS

   0 - hold the motion factor at 1.0
   I>0 - use the I^{th} time variation function from the TFUN record set to get the motion factor

3. The motion amplitude will be an angular displacement, velocity, or acceleration on nodes used to define angular motion of rigid bodies.

4. DYN and TSSS require all three motion components at a node to be defined. Individual components can be defined in DEAD and LIVE. A zero motion variation code in DYN and TSSS means a fixed component that remains at its initial position.

5. The maximum number of components (X, Y, Z) of imposed displacements in DEAD, LIVE, and MODE is 30. The maximum number of moved nodes in DYN and TSSS is 5.

6. The constraint code array (NODFIX) entry is superseded by a MOVE instruction. The constraint code is restored at the completion of the SAO.

7.1.10 OUTPUT - Specifies output data

Word	Variable Name	Description
1	OPTION	"OUTP"
2	NIPR	Number of solution steps between output prints
3	DTOUT	Output time interval (T) (DYN and TSSS only) or Output angle interval (degrees) (LIVE with heading change)
4	IBG	Debug output parameters (see Appendix I)
5	IFXOUT	Output current list of node component fixity codes if this is nonzero.

NOTES:

1. NIPR is used to calculate the output interval on DEAD and LIVE by multiplying the first step size by NIPR. The output interval remains the same even though the step size is changed unless NIPR = 1. When NIPR = 1, the output is at every step, regardless of step size.

2. DYN and TSSS can use NIPR or DTOUT to determine the output interval. If both are given, DTOUT has precedence.

3. Output is given at the beginning and end of each subanalysis. Output is also given when the load or time parameter exceeds the value of the parameter at the last output plus the output interval. Extra output records can be produced when a restart save request does not correspond with an output interval (see SAVE record).

4. Output for LIVE SAO with heading changes will occur at angle intervals selected by DTOUT. If DTOUT is not given, the results will be output at every angle calculated. NIPR has no effect on heading change output.

7.1.11 PAYOUT - Payout/reel-in data

Word	Variable Name	Description
1	OPTION	"PAYO"
2	JOP (I)	Initial node where payout/reel-in occurs
3	JPELT (I)	Initial element number for payout
4	PAYV (I)	Payout rate (LT^{-1}) + for payout - for reel-in
5	AMAXL (I)	Mitosis length (see Notes) (L)
6	NGROW (I)	Number of elements available for growth
7	NSHRINK (I)	Number of elements available for reel-in
8	NELPOI (I)	Element number increment (default = 1)
9	NPOVRY (I)	Number of time variation function 0 - constant rate n>0 - use n^{th} function defined in TFUN record

NOTES:

1. Payout/reel-in is approximated by incrementally changing the unstrained element length. When the unstrained length exceeds AMAXL (I) plus a reference length during payout and NGROW (I) > 0, then a mitosis operation will be performed. The payout element will be divided into two elements. The original payout node will be assigned to the division point, and the new node introduced in the new element will become the payout node. The new element is obtained by adding NELPOI (I) and JPELT (I). During reel-in the reverse process will occur when the unstrained length is less than AMAXL (I) and NSHRINK (I) > 0. In this case the new element is identified by subtracting NELPOI (I) from JPELT (I). AMAXL (I) = 0.0 causes the mitosis check to be ignored. The reference length for payout mitosis is one of the following:

A - The initial unstretched length of the payout element (JPELT (I)) if the next element to be activated has a different material number or if this is the first mitosis for that payout end.

B - AMAXL (I) for all other situations.

2. The elements available for payout are inactive until a mitosis activates them by assigning a length to them. These elements must be included in the total number of elements, and their nodes, material, etc., must be defined in the usual manner. The nodes for these inactive elements must also be defined in the usual manner with their coordinates defined to be the same as the initial position of the payout node (causes the element length to be zero). These nodes must be given constraint codes for all degrees-of-freedom. The appropriate codes are:

1 - if the component is to remain fixed and subject to limit checks (if any) after mitosis
2 - if the component is to be free after mitosis but possibly subject to subsequent limit checks
3 - if the component is to remain fixed and not subject to limit checks after mitosis

3. The elements available for reel-in must be active portions of the structure and sequentially connected to the reel-in point (no branches). The nodal constraint codes on the reel-in node will be assigned to the new reel-in node, and the deactivated nodes will be appropriately constrained automatically.

4. Payout/reel-in can occur only at fixed nodes or nodes defined in MOVE data sets. In either case, the displacements of all three components must be defined.

7.1.12 SAVE - Defines restart file save intervals

Word	Variable Name	Description
1	OPTION	"SAVE"
2	IRST	Restart file flag 0 - do not save records -n - rewind file and save every n^{th} output record n≥1 - extend present file and save every n^{th} output record
3	DTRSRT	Restart save time interval (T) (DYN and TSSS SAO only)
4	NIXPRN	Output suppress flag (see Note 3)

NOTES:

1. Restart files are:

   01 - DEAD
   02 - LIVE (TSSS)
   03 - DYN

2. When DTRSRT is given for DYN or TSSS, IRST is used only to signal file rewinds. The restart data is then written when the current time is greater than or equal to the time of the last save plus the save time interval.

3. If the step/time on the SAVE record does not coincide with the OUTPUT record request, then an extra output record will be produced unless NIXPRN ≠ 0.

7.1.13 SBUOY - Specifies vertical motion for a body on the surface

Word	Variable Name	Description
1	OPTION	"SBUO"
2	NODSUR	Node where body is located
3	NMOTN (I)	Motion code 1 - displacement 2 - velocity 3 - acceleration
4	IBS (I)	Time function number (see TFUN record, Section 6.18)
5	SBAMP (I)	Surface motion amplitude

NOTES:

1. The motion indicated by these parameters will be imposed on the vertical component whenever the body is held to a limit. This means the body node must appear on the LLOC record. When the limit restraint is exceeded, the body is released from the motion and the limit condition.

7.1.14 SOLUTION - Specifies solution option characteristics

Word	Variable Name	Description
1	OPTION	"SOLU"
2	SOPTN	Solution method: MNR (default for DEAD, LIVE, TSSS, FREQ) SLI RFB VRS } static only VRR } DIM (default for DYN) dynamic only
3	DMU	Numerical damping factor (see Note 1)
4	RERR	Residual norm error bound (default = 0.001) (see Note 2)
5	DERR	Displacement and pseudo-velocity norm error bound (default = 0.001) (see Note 2)
6	DALPHA	Proportional damping multiplier of mass matrix (DYN) or Alpha integration parameter (VR solutions - default = 1.0) or Ship angle damping for MNR
7	DBETA	Proportional damping multiplier of stiffness matrix (DYN) or Initial pseudo-time step (VR solutions - default = 1.0) or Buoy angle damping for MNR
8	SRCHFC	1D search factor for MNR method =0.0 - no 1D search of alternating estimates <0.0 - no alternating estimate checks >0.0 - the 1D search initial guess factor (see Note 4)

Word	Variable Name	Description
9	PARMT	MNR method extrapolation parameter (default = 0.5) (see Note 5)
10	CEPS	Strum update parameter (default = 0.0001) 0.0 - no drag amplifications for this SAO <0.0 - calculate drag amplifications only once per SAO >0.0 - calculate drag amplification at the beginning of this SAO, and each time the relative velocity norm changes more than CEPS (see Note 6).
11	NUP	Update option flag (let default)
12	JMPDT	Step-size control number (default = 2)
13	LMITER	Iteration limit (default = 200)
14	KONVRT	Number of trials before divergence abort (default = 3)
15	NRUP	Newton-Raphson update interval. Used on MNR method to signal how often to recalculate the tangent stiffness matrix, K_T. This is a packed word: • Units and tens digits give number of steps before new K_T. • Hundreds and above digits give the number of alternating sign trials on a given step before new K_T (default = 205, i.e., two alternating tries and five steps)

NOTES:

1. The numerical damping factor, DMU, is used to avoid problems with singular or ill-conditioned stiffness matrices. It has no influence on the DIM method. It should be used with caution with the SLI and RFB methods since it alters the stiffness matrix, hence the equilibrium equations, in these two cases. Possible values are:

$DMU \geq 1.0$	very heavy
$1.0 \geq DMU \geq 0.1$	heavy
$0.1 \geq DMU \geq 0.01$	moderate
$0.01 \geq DMU \geq 0.001$	light
$0.001 \geq DMU$	very light

If DMU < 10^8 and a singularity is encountered, then DMU is set to 0.001, and the step is tried again. On a repeated singularity, DMU is increased to 0.1, and the step is tried once more. If the singularity persists, the calculation is aborted.

2. The error bounds are used to test for convergence of the iterations. RERR is used only in the MNR and VRR methods. DERR is used in MNR, VRR, VRS and DIM methods.

The convergence criteria are:

$$\text{RNORM} \leq /\text{RERR}/$$
$$\text{DNORM} \leq /\text{DERR}/$$

where RNORM and DNORM represent norms of the nodal force residual and displacement increments, respectively. Both of these inequalities must be satisfied for convergence of the MNR iterations, while only the displacements are checked in the DIM method. The VRR solution checks the residual (RERR) as well as displacements and velocities (DERR). The residual is not used in the VRS solution.

Some flexibility in the form of these norms is available. If the value of RERR is greater than zero, then:

$$\text{RNORM} = \left(\sum_{i=1}^{N} R_i^2/N \right)^{1/2} / T_{max}$$

where: R_i = the i^{th} component of the nodal residual

N = the total number of nodal degrees-of-freedom

T_{max} = the maximum element tension

If the value of RERR is negative, then:

$$\text{RNORM} = \left(\frac{\sum_{i=1}^{N} R_i^2}{\sum_{i=1}^{M} R_i^2} \right)^{1/2}$$

where: M = the total number of nodal components including the fixed nodes; i.e., the reactions are included in the denominator (MNR method only).

If the value of DERR is greater than zero on static analyses, then:

$$\text{DNORM} = \left(\sum_{i=1}^{N} \Delta(\Delta q_i)^2/N \right)^{1/2}$$

where: $\Delta(\Delta q_i)$ = the i^{th} component of the change in the displacement increment.

If DERR is negative on static analyses (except VR methods), then:

$$\text{DNORM} = \left(\frac{\sum_{i=1}^{N} \Delta(\Delta q_i)^2}{\sum_{i=1}^{N} \Delta q_i^{\,2}} \right)^{1/2}$$

In the DYN subanalysis, the norm selection is the opposite of that selection used in static analyses. See the discussion of norms in Reference 8.

3. The terms DALPHA and DBETA can be used to specify internal damping proportional to the mass and/or stiffness matrix. The assumed form of the damping matrix is:

$$[C] = \alpha\ [M] + \beta\ [K]$$

This procedure can be used in DYN subanalysis.

4. The 1D search factor, SRCHFC, can be used with the MNR method in an attempt to enhance a poor initial configuration. See the discussion of the MNR method in Reference 8.

5. PARMT is an extrapolation parameter used in the incremental MNR method. The starting estimate for the displacement on all but the first step is the accumulated displacements from the previous step plus PARMT times the change in displacement calculated from the preceding step.

6. When strum strings are defined in the STRUM record (Section 6.16) and CEPS is not zero for LIVE, DYN or TSSS SAO's, drag amplification factors will be computed for the strings. This computation will be done at the beginning of the subanalysis and whenever the relative velocity between the fluid and cable elements changes significantly. A significant change is indicated by:

$$\frac{\left|\text{VNORM}_i - \text{VNORM}_R\right|}{\text{VNORM}_R} \geq \text{CEPS}$$

where $$\text{VNORM} = \left(\frac{\sum_{i=1}^{N} [\text{VF(I)} - \text{UD(I)}]^2}{N} \right)^{1/2}$$

R = time when drag amplifications were last calculated
i = present time
VF = fluid velocity components at each node
UD = nodal velocity components (DYN only)

7.1.15 STEP - Solution step-size data (static solutions)

Word	Variable Name	Description
1	OPTION	"STEP"
2	JSTEPR	Number of steps for load incrementing (default = 1)
3	NSTUP	Start up option n>1 - divide first step into n substeps in an arithmetic progression
4	HEDINC	Surface load heading increment (degrees)
5	HEDEND	Surface load total heading change (degree)

NOTES:

1. JSTEPR indicates the number of load/movement steps requested. During this phase of the SAO, the load and movement magnitudes are varied. When HEDINC is not zero, a sequence of static configurations will be generated following the convergence or completion of the magnitude incrementing. This additional loading sequence moves the surface wind and current headings through the excursion defined by HEDEND and the heading variation codes.

2. See Note 9 of Section 6.14 (NODE record) for global heading conventions.

3. See Section 7.1.16 (SURF record) and Section 7.1.18 (WIND record) for initial headings and variation codes.

7.1.16 SURFACE - Surface current data

Word	Variable Name	Description
1	OPTION	"SURF"
2	CURNT	Surface current velocity (LT^{-1})
3	CAD	Initial current heading (degrees)
4	KODEC (1)	Variation code for fixed heading (load varies)
5	KODEC (2)	Variation code for variable heading (load constant)

NOTES:

1. See Note 9 for Section 6.14 (NODE record) for global heading convention.

2. See Section 7.1.15 (STEP record).

3. Variation codes:

   0 - hold constant at value given
   -1 - increment down (not for heading variation)
   1 - increment up

4. Ships and buoys located at the surface limit the use this of these data and the auxiliary loading data to calculate static loads. Ships will obtain loading coefficients from a data file or built-in functions (see SHIP record, Section 6.15, and Appendixes B, F, and G). Buoys will use the BWND and BSCD parameters (see BODY record, Section 6.5). If the buoy is pulled away from the limit, its loads will be calculated from the CURR record (Section 7.1.1) defined flow field and the body drag functions (built-in or user defined). See Word (2) on BODY record, Section 6.5.

7.1.17 TIME - Time step data

Word	Variable Name	Description
1	OPTION	"TIME"
2	DT	Time step
3	TMAX	Final time
4	T	Initial time (defaults to 0)
5	DTU	Update time (defaults to DT); always let default for DIM solutions
6	ALPNEW	α integration parameter (default = 0)
7	BETNEW	β integration parameter (default = 1/12)
8	GAMNEW	γ integration parameter (default = 1/2)

NOTES:

1. If DT $\leq$ 0.0, the time step will be internally estimated. If DT = -A, then the estimated value will be multiplied by A. If DTU = -B, then DTU is set to B times the DT estimate.

2. The integration parameters are those of the generalized Newmark difference equations (Ref 4).

3. A nonzero initial time can be used for TSSS or DYN SAO to adjust to time in the TFUN records.

7.1.18 WIND - Surface wind data

Word	Variable Name	Description
1	OPTION	"WIND"
2	WIND	Wind velocity (LT^{-1})
3	WAD	Initial wind heading (degrees)
4	KODEW (1)	Variation code for fixed heading (load varies)
5	KODEW (2)	Variation code for variable heading (load constant)

NOTES:

1. See Note 9 of Section 6.14 (NODE record) for global heading convention.

2. See Section 7.1.15 (STEP record)

3. Variation codes:

   0 - hold constant at value given

   -1 - increment down (not for heading variation)

   1 - increment up

7.2 MODE - Mode shape calculation

The calculation of natural frequencies and mode shapes has only four optional data records: MSOL, FIX, FREE and MOVE. The FIX, FREE, and MOVE records are the same as for DEAD, LIVE, TSSS, FREQ, and DYN. They allow imposition or release of constraints prior to mode calculations. MSOL selects solution options. An MSOL record is not needed if the defaults are acceptable.

7.2.1 MSOL - Specifies solution format parameters

Word	Variable Name	Description
1	OPTION	"MSOL"
2	MODEI1	Mode shape order flag 0 - list mode shapes in order of increasing frequency 1 - list in order calculated
3	MODEI2	Mode shape output flag 0 - print all mode shapes n - print n mode shapes -n - print n mode shapes and write them on logical unit 20

7.2.2 FIX - Applies fixed conditions to nodes

This record is identical to FIX in Section 7.1.2.

7.2.3 FREE - Removes fixed conditions for nodes

This record is identical to FREE in Section 7.1.3.

7.2.4 MOVE - Specifies node point motion

This record is identical to MOVE in Section 7.1.9.

7.3 FREQ - Response spectrum calculation

The frequency domain subanalysis data set is composed of three levels of data following the FREQ flag record. The first level describes the setup of the implied LIVE solution for drift force updates. The second level specifies the setup of the frequency domain solution. The third level gives the data to be processed for each wave heading considered.

The characteristics of the implied LIVE step to provide drift force updates can be specified by the appropriate records:

OUTP	See Section 7.1
SAVE	
SOLU	
STEP	

These must appear between the FREQ keyword and any of the FREQ setup cards.

The data records for the FREQ solution setup are:

- FSOL - selects frequency domain options
- SPECTRUM - specifies spectral characteristics (required)
- EXTERNAL - ship motion file conversion factors
- RESULTS - optional output requests

Each of these is optional except for SPEC, and can be given in any sequence in advance of the first HEAD record.

The data records for the wave headings are grouped by HEAD flag records. The HEAD record sets contain one or more RAND or REGU record sets.

7.3.1 FSOL - Specifies frequency domain options

Word	Variable Name	Description
1	OPTION	"FSOL"
2	IDRITR	Drift force iteration flag (see Note 1) 0 - no drift force iterations 1 - perform drift forces iterations to convergence -1 - do a single drift force update with no repeat of frequency solution ±2 - same as ±1 except use only surge force (for single-point moorings)
3	IFRQUP	Iteration option flag for drift force updates 0 - start from original static reference 1 - start from last configuration found
4	ICMCHF	Component check file flag 0 - save response file 04 for use by CHEK record (see Note 2)
5	ICONMS	Mass matrix format 0 - lumped mass 1 - consistent mass
6	IROLIT	Roll damping iteration flag 0 - do not iterate n - iterate no more than n times to estimate ship's roll damping
7	ALPHA	Proportional damping multiplier of mass matrix
8	BETA	Proportional damping multiplier of stiffness matrix

NOTES:

1. If the static reference state is to be modified for the effects of wave drift forces, then the accumulated drift forces for a pass through the wave spectrum must be statically applied. A single update simply assumes the dynamic solution is unchanged by the new static reference state. The iterative solution repeats the frequency pass until negligible changes in position due to drift forces are calculated. This convergence is checked using the following measure:

$$DN = \left(\sum_{i=1}^{N} X_i^2 / N \right)^{1/2}$$

$$\frac{/DN' - DN/}{DN} \leq DERR$$

where: X = nodal point coordinates

N = total number of coordinates (3 times the number of active nodes)

DERR = displacement error tolerance (see SOLU record, Section 7.1.14)

The prime indicates the next value of DN.

2. Dynamic data for the CHEK record must be passed on file 04 or the CHEK results will include only the static reference state.

3. Roll iterations at each frequency seek to converge on a roll angle estimate by adjusting the roll damping. Convergence is assumed if a change of less than 1 degree is found on two successive cycles or when the number of cycles equal IROLIT.

4. Internal damping proportional to the mass and/or stiffness matrix is used when ALPHA and/or BETA are non-zero. The form is:

$$[C] = \alpha [M] + \beta [K]$$

7.3.2 SPECTRUM - Specifies spectral characteristics

Word	Variable Name	Description
1	OPTION	"SPEC"
2	SPECA	A coefficient (L^2 T^{-4})
3	SPECB	B coefficient (T^{-4})
4	DOMG	$\Delta\omega$, frequency increment (T^{-1})
5	OMGMN	ω_{min}, lower bound on frequency (T^{-1})
6	OMGMX	ω_{max}, upper bound on frequency (T^{-1})
7	AMPMN	Cut-off amplitude for waves (L) (default = 0.0001, see Note 2)

NOTES:

1. The wave spectrum is assumed to have the form:

$$S(\omega) = A/\omega^5 e^{-B/\omega^4}$$

where $S(\omega)$ is based on twice the square of the wave height. Any spectrum having this general form can be input. Values for common spectra are:

SPECTRUM	A	B
Pierson-Moskowitz	135.0	$9.7 \times 10^4/V_k{}^4$
Bretschneider	$4200\ H_s{}^2/T_s{}^4$	$1050/T_s{}^4$
I.S.S.C.	$2760\ H_s{}^2/T_s{}^4$	$690/T_s{}^4$

where V_k = wind speed (knots)

H_s = significant wave height (ft)

T_s = significant wave period (sec)

ω = circular frequency (radians/sec)

$S(\omega)$ = spectral ordinate (ft^2-sec)

2. Regular wave responses are calculated for waves with frequencies between ω_{min} and ω_{max}, which have wave amplitudes (based on the spectrum) greater than AMPMN. Incrementing will not proceed beyond ω_{max} regardless of the wave amplitude.

7.3.3 EXTERNAL - Specifies external ship motion file conversion factors

Word	Variable Name	Description
1	OPTION	"EXTE"
2	FRCFAC	Force conversion factor
3	FACLEN	Length conversion factor
4	TIMFAC	Time conversion factor
5	NFILEF	File format code (default = 0) 1 - NSRDC ship motion file 0 - NCEL ship motion file

NOTES:

1. The conversion factors are used as multipliers of the data on the external coefficient file (ship motion file). Each of them has a default value of 1.0.

2. The file formats are described in Appendix A.

7.3.4 RESULTS - Specifies optional output requests

Word	Variable Name	Description
1	OPTION	"RESU"
2	IUNRES	Unrestrained buoy and ship motion outputs 0 - no 1 - yes
3	IBGF	Debug output flag (see Appendix I)

NOTES:

1. The IBGF flag produces extra output for the frequency response solution only. The IBG flag (Section 7.1.10, OUTPUT record) will produce extra output only for the implied LIVE solution used for drift force updates.

7.3.5 HEADING - Wave heading record

Word	Variable Name	Description
1	OPTION	"HEAD"
2	GHED	Wave heading in global system (degrees)

NOTES:

1. HEAD data sets consist of the record followed by a series of records of RAND and/or REGU response requests. Each HEAD data set can be terminated by a DONE record, another HEAD record, or a valid SAO flag.

7.3.6 RANDOM - Random response data requests

Word	Variable Name	Description
1	OPTION	"RAND"

This flag record is followed by a string of records having the following form:

Word	Variable Name	Description
1	VTYPE	"NODE", "SHIP", "TENS", "DONE" (see Notes)
2	NUMC	Node or element number
3	NDRT	If VTYPE = "NODE" gives the global component direction 1 = X 2 = Y 3 = Z
4	SFSTAT	Static load factor ("TENS" only)
5	SF3	(1/3) load factor ("TENS" only)
6	SF10	(1/10) load factor ("TENS" only)
7	ST100	(1/100) load factor ("TENS" only)

NOTES:

1. The spectral response of the ship, element tensions, and any of the nodal displacement components can be calculated. VTYPE signals the one that is desired. Any number of these cards can be provided. The input is terminated by a card with VTYPE = "DONE". Termination can also be accomplished with a REGU, HEAD, or any valid SAO flag. The response data calculated represent the average of the 1/3, 1/10, 1/100 highest responses.

2. VTYPE = "SHIP" requires no other entries on the card.

3. VTYPE = "TENS" requires the specification of the element number in NUMC. If values are given for SFSTAT through SF100 and if the element material has the ultimate tension specified, then estimates of the factored maximum loads will be printed.

4. VTYPE = "NODE" requires the specification of the node number in NUMC and the component direction in NDRT.

5. A TENS record must be included for each element that will be involved in a subsequent component adequacy check. For example, if a check of an anchor capacity is to be made including dynamic effects, every element that connects to the anchor (or fixed node where the anchor is) must be called out on a TENS record.

7.3.7 REGULAR - Regular response data requests

Word	Variable Name	Description
1	OPTION	"REGU"

This flag record is followed by a string of records having the following form:

Word	Variable Name	Description
1	NODE	Node for which response is requested.
2	LOCAL	0 - produces output in global system 1 - produces output in ship's local system
3	WRFQ	Circular frequency of response (T^{-1})
4	WAMP	Wave amplitude for response (L) (default = 1.0)
5	IOCODE	0 - output is displacements 1 - output is position
6	NUM	Number of divisions per cycle (default = 30)

NOTES:

1. These records will generate the displacement or position versus time for all three nodal components through one cycle of motion.

2. The local coordinate system will not be used if position output is requested.

3. There is no limit to the number of requests that can be made.

4. A DONE, RAND, HEAD or any valid SAO flag record will terminate this option.

7.3.8 DONE - Optional terminator flag

Done	Variable Name	Description
1	OPTION	"DONE"

NOTES:

1. Can be used to terminate REGU and RAND record strings and HEAD data sets. Optional terminations are taken from the flag records if a DONE record is not given. Acceptable terminators are RAND, REGU, HEAD, or any of the SAO record flags.

7.4.1 CONF - Wave heading configuration

Use only if previous SAO is FREQ with more than one wave heading.

Word	Variable Name	Description
1	Type	"CONF"
2	--	Wave heading set number from previous FREQ. (default = 1)

NOTES:

1. These data are required to get solution data for wave headings other than the first one given in the foregoing FREQ solution (see Section 7.3).

7.4.2 ANCH, BUOY, LINE - Component requests

Word	Variable Name	Description
1	CTYPE	"ANCH", "BUOY", "LINE", "DONE"
2	NELD	Node or element number
3	ICODE	Component code
4	JCODE	Bottom factor code
5	CAPY	Component capacity (default = use inventory with SAFAC)
6	CMPID	Component Identifier for Inventory Anchor weight Chain size Line diameter Buoy O.D. (input in inventory units)
7	SAFAC	Safety factor (default = 1.0)

NOTES:

1. CTYPE identifies the type of component to be checked. Only anchors, buoys, or lines are allowed. Any number of cards can be given. Input is terminated by CTYPE = "DONE".

2. Negative ICODE with CTYPE = ANCH means the vertical load will be used in the capacity check.

3. The CHEK SAO can be initiated following any other subanalysis. If the previous one was not FREQ, the present state is evaluated. If it was FREQ and the dynamic response file was generated, the dynamic tensions will be included in the check for those elements of the file. (See Note 2 of Section 7.3.1, FSOL record)

4. If CTYPE is "ANCH" or "BUOY", then NELD is the node where the anchor or buoy is located. Anchor checks can be made for fixed nodes as well as for active nodes where limits have been defined.

5. No more than ten lines can be attached to any anchor or buoy at an active node with limits (see Note 3, Section 6.12, LLOC record). Anchor checks at fixed nodes allow up to twenty lines to be attached to the fixed node, which has no limit conditions specified.

6. Component codes:

ANCHORS

1. Navy standard stockless with stabilizers
2. NAVSHIPS lightweight
3. NAVFAC STATO
4. Imbedment (no inventory)
5. Stock (Admiralty) (no inventory)
6. Mushroom (no inventory)

BUOYS

1. Bar riser chain
2. Spherical or other (no inventory)

LINES

0. Chain
1. Samson 2-in-1® braided nylon
2. Samson 2-in-1® power braid
3. Samson 2-in-1® stable braid
4. Samson Blue Streak®
5. Other (no inventory)

7. Bottom factor codes for anchors:

1. Compacted sand
2. Stiff dense clay
3. Sticky clay of medium density (cohesive)
4. Soft mud (fluid), loose coarse sand, gravel
5. Hard bottom (rock, shale, boulders)

8. Contents of the inventory are presented in Appendix D.

7.5 END, NEW - Problem termination

The END flag record signals the completion of the SAO sequence with no more data to be read. The run is terminated.

The NEW flag record signals the completion of the SAO sequence. The next data record is a title card to begin a new problem. This title card must use the character string delimitors, and only one title record is allowed.

8.0 EXAMPLE PROBLEMS

This section contains four example problems and the description of some optional approaches to solving them. Although not exhaustive, these problems demonstrate many of the features of SEADYN. The models have purposely been kept small and geometerically simple to make them convenient to handle. The user should be aware that SEADYN is capable of modeling much more complicated geometries.

8.1 Towed Body Example

A spherical body towed at the end of a 280-foot line will be used to demonstrate various methods for starting the problem, the effects of grid coarseness, and changing a quasi-static solution to a dynamic solution. The pertinent problem data are:

Tow Cable Data

Unstretched length	280 ft = 3,360 in.
Drag diameter	0.35 in.
Cable weight (in seawater)	0.169 lb/ft = 0.014 lb/in.
EA	1.92×10^5 lb
Normal drag coefficient, C_N	$1.5 + 4.0\ (R_e)^{-1/2}$
Tangential Drag Coefficient, C_T	$0.2\ C_N$

Body Data

Weight (in seawater)	580.9 lb
Effective diameter	1.0 ft = 12 in.

The first part of the problem is to determine the position of the cable and body when towed in a straight line at a constant velocity of 10.5 knots. A convenient model for this is to assume the tow point is fixed and the system is subjected to a uniform current equal to the tow speed. This allows the static solution form to be used.

Although the general shape of the system at the tow speed can be guessed, the exact positions of the nodes and the tensions in the various portions of the cable are not easy to compute. Three approaches to calculating the steady-state towing shape are outlined.

APPROACH 1: Begin by assuming the line is hanging straight down. Apply the gravity load with no current to get the suspended tension distribution. Incrementally apply the current to deploy the cable in an approximate solution (RFB method). Iterate on the deployed state to satisfy equilibrium (MNR method). The input for accomplishing this is shown in Table 8-1. It should be noted that the first step requires artifical tensions to get started. Numerical damping and the MNR solution work well here. Either VR method could also be used.

APPROACH 2: Begin by assuming the line is deployed horizontally in an unstretched state. Apply the gravity and current simultaneously in increments to get an approximate solution (RFB method). Dummy tensions are assumed. Iterate on the deployed state to satisfy equilibrium (MNR method). The input is shown in Table 8-2.

Table 8-1. TOWED BODY APPROACH 1 INPUT

```
TOWED BODY DEMONSTRATION PROBLEM

APPROACH 1 --- START FROM VERTICAL;
PROB
11,10, 2,,1,1,386
FLUI
0,0,.00252,.037
BODY
1,,-580.9,12
MATE
1,1,.35,.014112,1,W8,0,1.92E5,1
NODE
1,,,3360                 *BODY NODE
11,1,W6,1,1,1            *TOWED NODE       IMPLIED GENERATION
ELEM
1,1,2,,1,W8,1            *ASSUME UNSTRETCHED LENGTHS
10 10 11,,1
BLOC;1,1                 *LOCATE BODY AT NODE 1
FLOW
1,1, 212.8
DEAD                     *DEPLOY LINE STRAIGHT DOWN WITH MNR + DAMPING
     SOLU,,.001
LIVE                     *MOVE TO APPROX TOWED SHAPE WITH 10 STEP RFB
     CURR,1,1,,1
     SOLU,RFB
     STEP,10
LIVE                     *ITERATE TO CORRECT STATE WITH MNR
     CURR,1
END
```

Table 8-2. TOWED BODY APPROACH 2

```
TOWED BODY DEMONSTRATION PROBLEM

APPROACH 2 --- START FROM HORIZONTAL;
PROB;11,10,2,,1,1,386
FLUI;0,0,.00252,.037
BODY;1,0,-580.9,12
BLOC;1,1
MATE
1,1,.35,.014112,1,W8,0,1.92E5,1
NODE
1,,3360
11,W6,1,1,1
LINE;9,1,11
ELEM
1,1,2,,1,W8,1            *ASSUME UNSTRETCHED LENGTHS
10,10,11,,1
TENS;1,10,1,,500         *APPLY DUMMY TENSIONS
FLOW;1,1, 212.8
LIVE                     *MOVE TO APPROX STATE WITH RFB
     CURR,1 ,1
     SOLU,RFB
     STEP,100,10
     LVAR,W5,1           *SET GRAVITY LOAD FLAG
LIVE
     SOLU,VRR
     CURR,1
END
```

APPROACH 3: Begin by guessing an angle that approximates the deployed state. Deploy the unstretched cable at that angle. Iterate on this guessed state to satisfy equilibrium. Input for this approach is given in Table 8-3. The VRR method is used in this case since the quality of the initial guess is very low and divergent MNR iterations can be expected. The VRR method automatically compensates for the lack of initial tension.

These three approaches are not the only possible options. In each approach, the beginning state was described by the positions of the nodes in an unstretched state (ISIGO = 1). Other approaches can be devised quite easily.

The number of elements for these tests was chosen somewhat arbitrarily. The results of changing the number of elements are summarized in Figure 8-1. The figure shows quite accurate results even for very crude models. These models have assumed uniform spacing of the nodes for input convenience. The equilibrium state suggests it would be more appropriate to put shorter elements near the towed body where the curvature is the greatest.

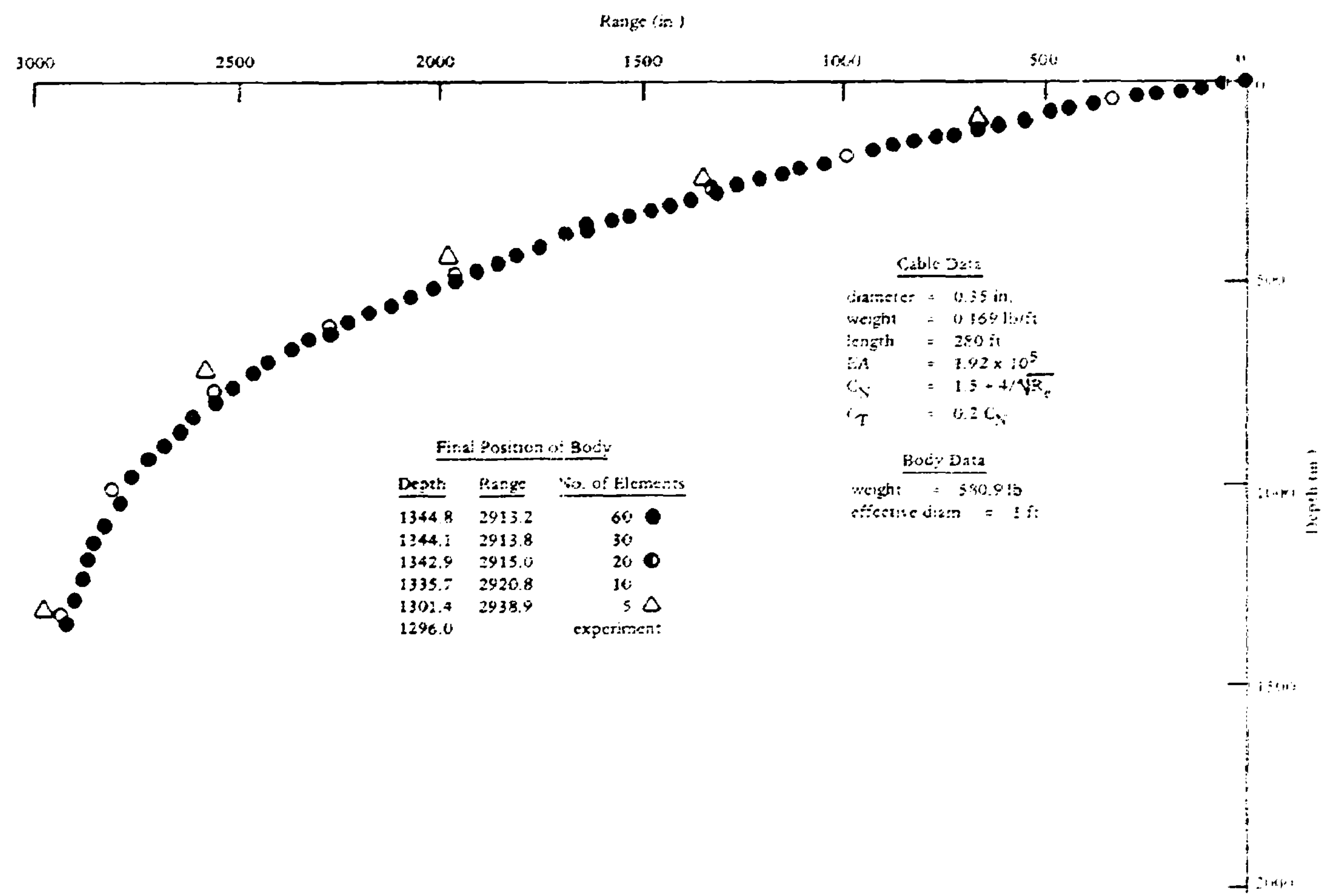


Figure 8-1. Towed body test case

The steady-state towing configuration can be used as the starting point for a dynamic analysis in which the tow velocity and direction are changed. Table 8-4 shows the additional input required to continue from any of the previous converged static states to a dynamic solution where the tow point is moved. The key here is the change in flow-field velocity to zero and assigning the initial towing velocity to all of the nodes. The movement function given defines a sudden change of direction of the towing at the same velocity. Arbitrary movement can be defined.

Table 8-3. TOWED BODY APPROACH 3

```
TOWED BODY DEMONSTRATION PROBLEM

APPROACH 3 --- START FROM SLOPED LINE;
PROB;11,10, 2,,1,1,386
FLUI;0,0,.00252,.037
BODY;1,0,-580.9,12
BLOC;1,1
MATE
1,1,.35,.014112,1,W8,0,1.92E5,1
NODE
1,,3195.55,1038.297
11,W6,1,1,1
LINE;9,1,11
ELEM
1,1,2,,1,W8,1            *ASSUME UNSTRETCHED LENGTHS
10,10,11,,1
FLOW;1,1, 212.8
LIVE                     *ITERATE ON GUESS TO CONVERGENCE WITH VR
     SOLU,VRR
     CURR,1
END
```

Table 8-4. TOWED BODY DYNAMIC INPUT

```
TFUN
     1,3,1,1,1,1,.866,90,.866
     2,3,0,1,0,1,.5,90,.5

DYN
     INIT,-212.8
     MOVE,11,2,1,-212.8,W9,2,2,212.8
     TIME,,30              *FIND OWN TIME STEP
     OUTP,W3,1
```

One general comment is needed on this problem. Care must be taken to assure all data are provided in consistent units. The choice of units is arbitrary since SEADYN does not presume any units (except for certain identified defaults). The choice made here is:

Length - inches
Force - pounds
Time - seconds

Note that lengths, diameters, forces, velocities, fluid properties, and gravitational acceleration all use these units. In all of the approaches, the tow cable drag coefficients are given by the DRAGCO subroutine shown in Table 8-5.

Table 8-5. DRAGCO SUBROUTINE

```
      SUBROUTINE DRAGCO(INDRAG,ITYPE,NUM,REN,RET,CN,CT)
      CN=0.
      CT=0.
      IF (REN.LT.100.) GO TO 500
      GO TO (100,200,300).ITYPE
C         SPHERICAL BUOY
  100 CN=.47
      GO TO 500
C         CYLINDRICAL BUOY
  200 GO TO 300
C         CABLE
  300 CN=1.5+4./SQRT(REN)
      CT=0.02*CN
  500 RETURN
      END
```

8.2 Buoy Relaxation Example

Consider a buoy restrained by a single line that is anchored to the seafloor. Assume that the buoy is snagged and hauled over to a certain position and held there. Then, after some time, the buoy and line are released to move dynamically to the original gravity loaded state. A SEADYN analysis of this scenario is presented in three stages:

STAGE 1 - Establish the gravity-loaded initial state.
STAGE 2 - Move the buoy over to the restrained (snagged) position.
STAGE 3 - Release the restraint and follow the dynamics back to the initial state.

The actual execution of these stages is dependent on how the snagged position is defined. The problem statement implies the dynamics of the snagging operation are not important and only the calculation of the final static state in the restrained position is needed. Two different definitions of the snagged state are pursued here. The first presumes the snagged position of the buoy is known. The second presumes the magnitude and direction of the snagging load is specified.

Recalling the previous example problem, there are various approaches that can be used to get the initial static state. In both cases a gravity-load vertical state is used as a starting point. Loads or displacements are then applied, and iterations are carried to convergence at the final state. The dynamic phase is initiated simply by releasing the displacement constraint or force. Input for these examples is presented in Table 8-6. The second case is assumed as a restart from the first to demonstrate the RESTART section. The viscous relaxation method was chosen for both of these since it is most likely to remain stable and obtain convergence.

The dynamic solution used the direct iteration method since this has proven to be the most reliable. The algorithm was left to choose the time step needed since no external loading with its own time characteristics was present.

Table 8-6. BUOY RELAXATION INPUT

```
BUOY RELAXATION DEMONSTRATION PROBLEM -- INITIAL POSITION GIVEN;

PROB;11,10,-2,,1
FLUI;0,1
BODY;1, ,.129,2
BLOC;1,1
MATE;1,,.15,.00055,W9,2.36,1
NODE
1,,,5.551
11,1,0,.104,W6,1,1,1
ELEM
1,1,2,,1                *ISIGO=0 WITH NO LOADS SAME AS ISIGO=1
10,10,11,,1
DEAD                    *GET VERTICAL STATIC STATE
     SOLU,,.01          *SET NUMERICAL DAMPING
     SAVE,-1            *SAVE DEAD RESULTS
LIVE                    *MOVE TO SNAGGED POSITION
     SOLU,VRR
     MOVE,1,1,1,4,1,1,-2
DYN                     *RELEASE FOR RELAXATION
     TIME,,10
     OUTP,,1
NEW
"BUOY RELAXATION DEMONSTRATION PROBLEM -- FORCES SPECIFIED"
REST;NEW,1              *GET THE LAST DATA SAVED FROM PREVIOUS DEAD
LIVE                    *APPLY SNAG LOAD
     SOLU,VRR
     LOAD,1,.115,-.0265,,1     *LVAR DEFA
DYN                     *RELEASE FOR REL     ON
     TIME,,10
     OUTP,,1
END
```

8.3 Anchor Last Example

An anchor last deployment scheme for tethered buoys is readily modeled in SEADYN. The starting state before the release of the anchor requires some careful consideration, however, when special situations are modeled. Static analysis problems can occur when floating lines with low initial tensions are involved. This presents a problem because slight changes in tension distribution can cause large position changes. In such situations it is best to make a reasonable guess on the initial state with no tensions and go immediately to the dynamic analysis following the release of the anchor.

This is a relatively simple problem intended to demonstrate the mechanics of getting the static and dynamic solutions accomplished. The problem presented here is sufficiently well-conditioned to allow the initial static state to be calculated. The problem characteristics are given in Figure 8-2. The input data are given in Table 8-7. The input assumes the anchor is released at time zero, and the top buoy is held for 10 seconds before it is released. The line between the two buoys is assumed to be neutrally buoyant. The initial state holds node 1 fixed and applies 1,000 pounds horizontal force to node 10. This stretches the neutrally buoyant line out straight and produces a catenary configuration in the heavy line. The guessed initial input uses unstretched lengths (ISICO = 1) and the catenary line generator. The initial distance between nodes 4 and 10 is selected to give the desired line length for the catenary.

Table 8-7. ANCHOR LAST DEPLOYMENT INPUT

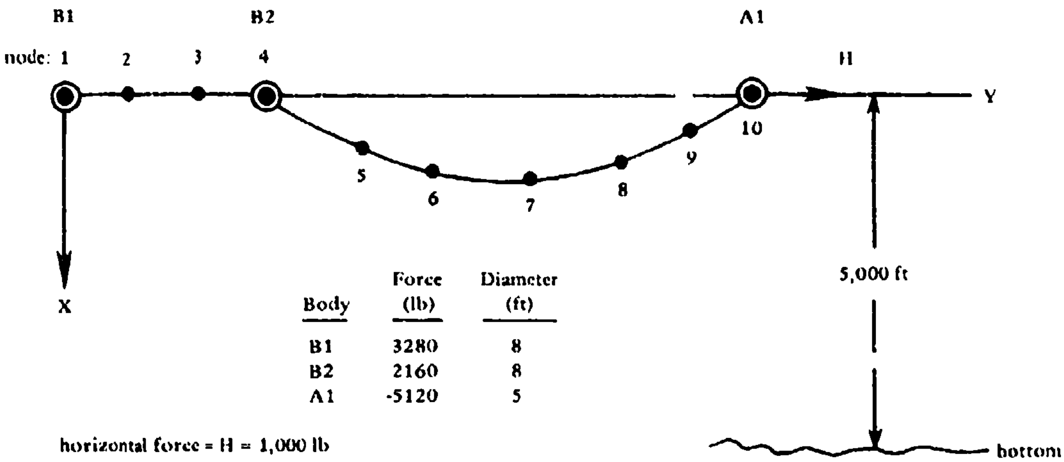
```
DOUBLE BUOY DEPLOYMENT USING ANCHOR LAST METHOD;

PROB;10,9,1,1,1
FLUI;0,1
LIMI
     1,,1                  *SURFACE
     2,5000,1,,1           *BOTTOM
BLOC
  1,1,W5,1                 *BUOY 1 AT NODE 1
  2,4,W5,1                 *BUOY 2 AT NODE 4
  3,10,W5,2                *ANCHOR AT NODE 10
NODE
1,W6,3,3,3                 *HOLD TOP BUOY
4,,,1200,,3                *VERTICAL HOLD ON BUOY
10,,,2400,,3,,1            *HOLD ANCHOR, LEAVE HORIZONTAL FREE
LINE;5,4,10,W7,2,.5,1000              *GENERATE CATENARY AND NODES 2,3
2,1,4
ELEM
1,1,2,,1,W8,1,1000
4,4,5,,2
9,9,10,,2
MATE
1,,.3,W9,1.E4,1
2,,.2,.5,W9,1.E4,1
BODY
1,,2000,8,W8,20            *BUOY 1 WITH SURFACE DRAG
2,,1000,6,W8,15            *BUOY 2 WITH SURFACE DRAG
3,,-5000,5                 *ANCHOR
DEAD                       *SINGLE STEP VRR SOLUTION
     SOLU,VRR
     LOAD,1,,1000,,10
DYN                        *DROP ANCHOR
     FIX,1,41              *CHANGE CODE TO ALLOW BUOY PULL-UNDER
     FREE,101
     TIME,,10.
     OUTP,,2
DYN                        *RELEASE BUOY 1
     FREE,11,12
     TIME,,1000
     OUTP,,100
END
```

8.4 Payout From a Moving Ship Example

The deployment of a cable system by paying out line from a moving ship is demonstrated in this example. The problem selected is the final stages of deployment of a trapezoidal array system (Ref 15). Figure 8-3 shows the starting configuration. The system contains two clump anchors and two suspension buoys. In the sequence modeled, the first clump anchor is in place and the final anchor is being lowered by a line paying out from a cable ship. The payout rate and ship velocity are chosen so as to place the anchor at the desired point. The demonstration problem uses a straight line ship velocity (away from the cable system) of 1 ft/sec and a constant payout rate of 2 ft/sec.

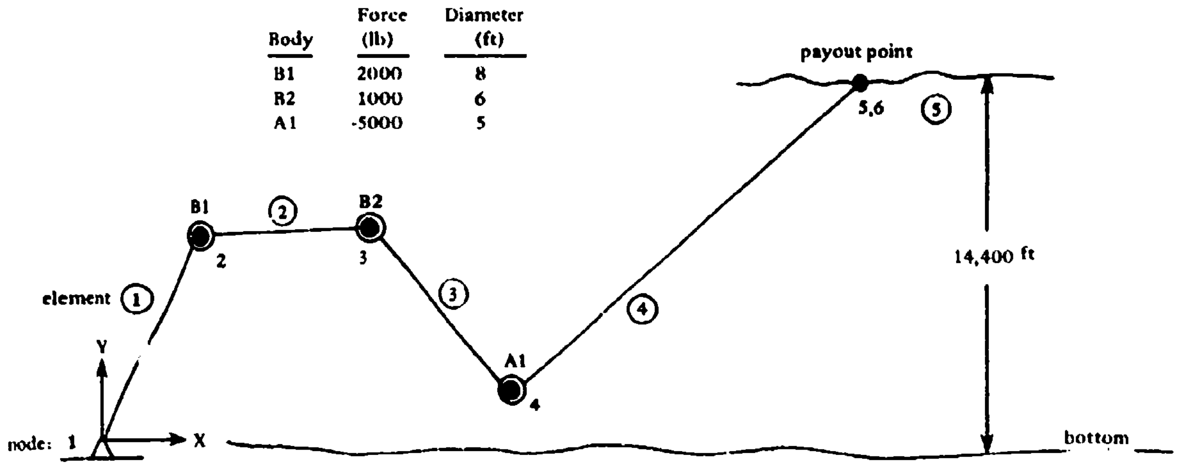
The element model used is the simplest possible: only one element per line. The payout line element is assumed to grow to a maximum length of 10,000 foot and then be divided into two elements (mitosis). Up to this point element 5 and node 6 are not involved in the solution. When mitosis occurs, node 5 is placed at the dividing point, node 6 takes the position of the moving payout point, and element 5 enters the system as the upper half of the payout line with an unstretched length equal to the mitosis length. Element 5 is then the payout element, and element 4 maintains a constant unstretched length equal to the unstretched length just before mitosis minus the mitosis length. When the anchor (node 4) reaches the bottom, it is held fixed in three components.

Table 8-8 lists the input for a DYN analysis using the DIM method. Table 8-9 presents input for a TSSS analysis of the same time sequence. The VRR form was used, but the MNR method could also have been selected.



Body	Force (lb)	Diameter (ft)
B1	3280	8
B2	2160	8
A1	-5120	5

Figure 8-2. Anchor last model.



Body	Force (lb)	Diameter (ft)
B1	2000	8
B2	1000	6
A1	-5000	5

Figure 8-3. Payout model.

Table 8-8. PAYOUT FROM MOVING SHIP INPUT

```
PAYOUT FROM A MOVING SHIP;

PROB;6,5,2,1,1
FLUI;14400,1
BODY
1,,3280,8
2,,2160,8
3,,5120,5
BLOC
1,2
2,3
3,4,W5,1                 *STOP ANCHOR AT BOTTOM
LIMI;1,0,5,,1
MATE;1,,.1667,W9,1.E6,1
NODE
1,W6,1,1,1               *GLOBALLY FIX ANCHOR NODE
2,,7500,13300
3,,10400,14200
4,,23000,5700
5,,29739,14400,W6,2,2,2 *ORIGINAL PAYOUT NODE
6,,29739,14400,W6,1,1,1 *NEW PAYOUT NODE
ELEM
1,1,2,,1
5,5,6,,1
DEAD
     SOLU,VRR
DYN
     MOVE,5,2,,1,1,W9,1
     PAYO,5,4,2,5000,1,1
     SOLU,W12,-1         *RESTRICT TIME STEP GROWTH
     TIME,.5,3600
     OUTP,,100
END
```

Table 8-9. TSSS INPUT FOR PAYOUT PROBLEM

```
TSSS
     SOLU,VRR
     TIME,100,4000
     MOVE,5,2,,1,1,W9,1
     PAYO,5,4,2,5000,1,1
     OUTP,,100
END
```

9.0 HELPFUL HINTS FOR MODELING AND TROUBLESHOOTING

The most common problem encountered by a new user of a complicated computer program is the mastery of the input syntax. It is hoped that the example problems will alleviate a large number of these problems. An attempt to formulate SEADYN input for similar problems should be made before going on to the user's own problems. A variety of small problems should be tried until the syntax is relatively familiar.

Beyond the syntax problems, one encounters the modeling errors and the need for understanding the quirks of the solution methods. Finite element modeling requires that certain abstractions and approximations be made to get from the real world to the numerical model. One must decide how many nodes to use, where to place them, how to number them, what to lump where, etc. Since SEADYN uses straight-line elements, there will always be a disparity between the actual length and the modeled length on curved lines. Also there will always be the problem of step changes in tension from one element to the next. These approximations are inherent in the modeling process, and the analyst needs to become familiar with the consequences of approximating for his particular situation. Answers to questions of how many nodes and where to place them are strongly problem-dependent. In general, enough nodes should be provided to model geometry and tension variations to the level that is important to the problem. Each element assumes a uniform material so each material must be represented by at least one element. Highly curved regions, and regions where tensions can vary significantly require shorter elements. How short is short enough can be answered from grid convergence studies on a particular problem. The result presented in Figure 8-1 is an example of such a convergence study.

The solution of simultaneous equations in SEADYN follows a Gaussian elimination procedure that attempts to minimize computer storage by taking advantage of equation symmetry and bandedness. This means that the amount of storage required and the number of matrix operations are sensitive to the way the nodes are numbered. As a general rule, the largest difference between node numbers defining any element determines the bandwidth. The smaller the difference the smaller the bandwidth and the smaller the cost. In calculating the maximum node number difference, the master node number or the master node number plus one (since node pairs are used to get rotation) must be used in place of the slave nodes. Another general rule for node numbering (mentioned in Section 3.0) is to number such that the softer (lower stiffness) components are encountered first. This reduces the numerical error propagation in the analysis. Often it is not obvious what components are soft relative to others and/or requirements for meeting this condition are in conflict with the optimum bandwidth numbering. Judgment and experience on the part of the analyst are needed in making compromises in this area. It should be noted that all solution methods in SEADYN, except the DIM method, require the solution of simultaneous equations and are bandwidth sensitive. The MODE analysis uses the Jacobi iteration method and is also bandwidth sensitive.

Errors that can occur because of node numbering can show up as a singular system of equations. The message printed would be "EQUATION DECOMPOSITION FAILED, etc." (see Section 10.3). This message also results from problems with input errors and for unstable structures, so one should not be hasty in concluding it is the result of accumulated numerical error. A specific situation where this can occur is in a multileg mooring of a ship. Certain pretensions and positions of the legs can cause the ship's responses to be stiffer than the lower portions of the mooring legs. If the ship's nodes are numbered before the mooring legs, numerical error propagation can cause a singular matrix error at the lower ends of the lines. Renumbering the nodes so that the ship is last will remove the problem.

A very common error leading to a singular stiffness matrix is an incomplete set of boundary conditions or an inconsistent set of pretensions that causes a portion of the structure to go slack. Boundary conditions must be given to fully restrict rigid body motion of the structure. This means six degrees-of-freedom must be constrained. Since SEADYN treats geometric nonlinearities, the rotation degrees-of-freedom represented by elements displacing one end laterally relative to the other can be restrained by tension in the element. For example, a single element held fixed at one end (three constraints), with a tensile load at the other, is a stable system if the stretch in the element is in balance with the load. The tension in the element resists the pivoting of the element about the fixed end. There is no twist degree-of-freedom in the element, so no constraint is required for it. If the end load is not applied in line with the element, a rotation must occur until the element aligns with the load.

Section 3.0 discussed the problem of obtaining a stable initial configuration of the finite element model to be used as a starting point. The analyst should always keep in mind that the finite element model is an approximation of the actual structure. Even if exact data describing an initial static state of the actual structure were available, use of that data in a finite element model would not represent a converged equilibrium state. Some minor adjustments in position of the nodes and element tensions would inevitably be calculated by the numerical iterations.

Two procedures for getting stable initial configurations have been used with some success. The examples demonstrate both of them. The first approach is to use a numerically damped MNR method or a combination of a few RFB steps followed by MNR iterations. The second approach is to use the VRR method with or without increments. In either of these approaches it may be necessary to adjust the analysis parameters to get convergence. Two general forms of behavior can lead to a lack of convergence, very slow convergence, or wildly oscillating behavior.

If damping is too large for the particular situation, movement will be sluggish, and the force residual will not change significantly in any one iteration. This can occur in the VRR solution if the structure is stiff (e.g., heavily loaded, high EA lines). The VRR solution will signal this situation by repeated output of the SLOW CONVERGENCE message with neither of the INCREASING NORM messages. The appropriate action in this case is to try the analysis again with a much smaller initial numerical damping (Word (3) on the SOLU record). The default value is 0.001, but in some situations it may be necessary to make it much smaller. Alternatively, the MNR method could be tried with default parameters.

The very slow convergence behavior of the VRR method suggests that the algorithm wants to adapt near-zero damping, which is the Newton-Raphson method. Sluggish behavior of the MNR method also means numerical damping is too high.

The more common situation is the divergent behavior evidenced by very large (often increasing) oscillating results from the iterations. This usually means large damping should be used or strategy modifications made which are suggested by the pattern of the unstable behavior. Divergence is most often encountered using the MNR method when a poor starting point is given that requires a significant angular movement. The VRR approach will generally work much better than the MNR approach in these cases since it can more readily adapt to the problem by damping and step-size adjustments. The adaptive features of the VRR method have been developed somewhat empirically by following a particular divergent pattern and then constructing a strategy to detect and react to it. This means it is possible that the action taken is inappropriate in some new situations or that a nonconvergent behavior with a different pattern is undetected. These situations usually produce copious messages with a mixture of INCREASING NORM and SLOW CONVERGENCE forms, and no clear pattern of reduction of the force residual is obtained. The solution will then fail by running to the limit number of iterations (Word (13) on SOLU record) or the job limits (time or output lines). When the VRR method runs to the limit iterations without converging, the pattern of the messages and the values of the residual and velocity norms should be studied before attempting a rerun. Sluggish behavior will be indicated by small velocity norms, and large residual norms with small position changes. In this case, the actions noted above should be considered. If the pattern shows large velocity norms and no definite residual norm pattern, an increase in damping could be effective. Applying the loading in increments could also be considered.

If very erratic norm changes occur and the values are very large, one should review the problem data to see if anything is wrong there. Finding nothing wrong, a different strategy should be considered. This can involve a change in the initial state input and/or a different mix of analysis parameters. Very abrupt load/displacement changes should be introduced incrementally. The physical possibility of unstable states should be investigated (it can be that the load level requested produces a physical instability).

One mistake that is sometimes made in dealing with the staging of subanalyses is the sudden omission of loads or constraints imposed during the preceding stage. The load and displacement data defined at the beginning of each stage must be consistent with the total values obtained at the end of the preceding stage. For example, assume Stage A applied load A to the structure with displacement constraints A. At the beginning of Stage B, the loads and constraints defined must be loads A and displacement constraints A unless it is an initiation of a dynamic sequence with suddenly released conditions. If Stage B is intended to begin from load A and impose additional loads B, it is necessary to redefine loads A with a zero variation code on the LVAR record, and define loads B with the +1 variation code. Stage B will then maintain loads A and incrementally add loads B. If loads A were omitted, the initial loading would not be in balance with the nodal positions and element tensions. This would then lead to an abrupt change in position and possibly a divergent solution step.

MODE analyses do not require the specification of loads, but the displacement constraints in force at the end of the preceding stage must remain in force. If these displacements were imposed with a MOVE record, then a FIX record or MOVE record would be required for these components in the MODE SAO set.

Dynamic analyses present a new set of problems. The most effective form for time domain dynamics is the DIM method. Generally, the default parameters will be appropriate. The choice of a time step is the main concern. The DIM method will select its own time step if none is given; however, the selection algorithm is not completely reliable. One problem that occurs for strongly nonlinear dynamics is that the appropriate time step may vary with time. A new upper limit on the time step is calculated only when there is a signal that the analysis is not converging. The signals used are:

a. A displacement change norm exceeding 1.E12

b. A large number of iterations without convergence (LMITER on the SOLU card)

c. A persistently increasing displacement change norm

In these situations the time step upper bound will be recalculated and the step size reduced. It is possible that the problem will not be detected early enough to avoid fatal growth of errors in the analysis. Time domain analyses which make repeated time step changes, should be rerun with the step forced to remain below the range of the changes. This is done by specifying DT (Word (2) on the TIME record) and setting JMPDT to -1 (Word (12) on the SOLU record).

Time domain analyses without material damping tend to be sensitive to step size and will produce spurious oscillations if the step size and/or convergence tolerance are not appropriate. Even with damping, erroneous oscillations can be induced by setting the convergence tolerance (DERR) too high. The correct value of DERR is somewhat problem-dependent. The default value of 0.001 is a middle-of-the-road choice. One should not set it much higher without some verification that the results are acceptable. It may be necessary to set it lower. One example of this is when the time step is desired to be much smaller than the upper bound, and strong nonlinearities are present (e.g., payout or bottom interaction). There is a trade-off between step size, DERR, and the number of iterations to convergence. Although the optimum mix of steps and iterations is hard to determine, experience has indicated that DT and DERR values, which give between three and about seven or eight iterations per step, are reasonable for strongly nonlinear problems.

10.0 SUMMARY OF MAJOR DIAGNOSTIC MESSAGES

The SEADYN program makes some checks of the input data and attempts to aid the user in finding his errors by printing various messages. No attempt has been made to be comprehensive in this feature since it is very difficult to foresee and/or detect many of the possible errors. The input routines that process the PROB and SAO data sets produce various diagnostics that evaluate errors detected in the input. These messages are generally self-explanatory. They deal mainly with program restrictions, such as the maximum number of items allowed, or the completeness and consistency of the data provided. The user should have little difficulty interpreting the problem detected, and should be able to make appropriate corrections with the aid of this manual.

During the processing of the subanalyses, checks are made on the validity of the requests and the convergence of the analysis procedures. The messages that can be printed are listed below with a brief description of the probable cause and/or cure. The action taken after the error detection is indicated by the following codes:

(F) Fatal, run aborted.

(N) Abort analysis case and seek a new problem definition by searching the deck for a NEW SAO record.

(O) Abort present SAO activity and go to the next SAO record.

(S) Skip this request and go to the next card.

(C) Continue calculation with action as indicated.

10.1 System Description Checks

(F) DEPTH CORRECTION ERROR -- SHIP DRAFT EXCEEDS WATER DEPTH

Gross input error. Check for dropped deck!

(N) ELEMENT GENERATION ERROR

First element was not input, or element cards are not in increasing element number order, or last element was not input.

(F) ERROR IN (WIND/CURRENT) LOAD TABLE ON SHIP XXX
HEADING = XXX
LAST TABLE ENTRY = XXX
SYMMETRY FLAG = XXX

Heading requested exceeds the largest value in the table. Check ship load input table.

(N) IMPROPERLY DEFINED MOORING BUOY AT NODE XXX NO. OF SLAVES FOUND = XXX

Mooring buoys require at least two slaves.

(N) NODES OMITTED IN GENERATION

Message is followed by a list of ones and zeros (ten per line) corresponding to the nodes in the system. The zeros indicate which nodes were not defined either by input or generation sequence. All nodes must be accounted for. Following the integers, the nodal coordinates are printed to aid in finding the error. The problem usually comes from improper node generation input.

(N) SHIP DATA INPUT ERROR ON XXX (Ship No.)
NO. OF SHIPS ON FILE = XXX
LOAD FUNCTION OPTION = XXX

Blank record for moored ship data requested definition of ship from load file with no load file defined.

OR

Attempted ship scaling with no load file defined.

(N) TAPE POSITIONING OR FORMAT ERROR
UNABLE TO FIND SHIP DATA
ITEM XXX LAST READ IS FOR HEADING XXXX
AND WAVELENGTH XXXX.
WANTED HEADING XXXX WITH WAVE LENGTH XXXX.

The ship motion file is not formulated properly or other input or equipment malfunction has made reading the file impossible.

(C) WARNING--UNITS DO NOT APPEAR TO BE CONSISTENT
GRAVITATIONAL ACCELERATION FROM TAPE IS XXX
UNIT LABELS FROM TAPE ARE XXX XXX XXX

The ship motions file conversion factors do not properly convert the GRAV on the file to the GRAV specified in this run. Calculation still proceeds.

10.2 Subanalysis Option Errors

(S) ANCHOR NOT ON BOTTOM AT NODE XXX
SKIP REQUEST FOR XXXX

Anchor weight is not sufficient to hold the lines at this node and it has been lifted off the bottom or has not reached the bottom.

(F) BLANK COMMON ON TAPE LARGER THAN SPACE AVAILABLE

The data saved on the file requires more storage than is presently available. The two numbers printed are the required and current values of NCOM.

(S) CAPACITY AND COMPONENT ID BOTH ZERO
SKIP REQUEST FOR XXXX (CTYPE)

Check input card.

(C) DATA NOT AVAILABLE FOR XXXX
SKIP THIS REQUEST

Regular wave response data were requested for a frequency outside of the range that was generated on the Ship Motion file.

(F) INCONSISTENT ANALYSIS REQUEST ON RESTART

Attempted a restart of DEAD, LIVE, or DYN, and the file was not from that type of subanalysis.

(N) INSUFFICIENT STORAGE TO PROCEED
COMMON SIZE = XXX
STORAGE NEEDED = XXX
HALF BANDWIDTH = XXX
DEGREES OF FREEDOM = XXX
BASE SIZE = XXX

Subanalysis request (DEAD, LIVE, DYN) cannot be processed due to storage limitations. Problem must be reformulated or NCOM increased. (See Appendix J.)

(N) INVALID CALCULATION OPTION = XXXX
CASE TERMINATED

The frequency domain calculation option was not RAND, REGU, or DONE. Records are out of sequence or record mispunched.

(N) INVALID COMPONENT TYPE XXXX
ASSUMED OF INPUT

Occurs when random response requests are being processed and a card is encountered which does not have SHIP, NODE, TENS or DONE keyword. The items to this point are processed, and the case is aborted with the additional message.

(S) INVALID COMPONENT TYPE
SKIP REQUEST FOR XXXX

The component number is not one recognized by the component inventory.

(C) INVALID NODE NUMBER = XXX
SKIP THIS REQUEST

Regular wave response data were requested for a node number, which is less than 1 or greater than the number of nodes in the model.

(S) MORE THAN TWENTY LINES ON ANCHOR AT NODE XXX
SKIP REQUEST FOR XXXX

The fixed node where anchor capacity check is requested has too many elements connected to it to make the check.

(S) NO DYNAMIC TENSION PROVIDED ON ELEMENT XXXX
IGNORE DYNAMICS

The random response data for this element was not requested by a TENS record for this wave heading in the FREQ SAO.

(S) NO LINES CONNECTED TO NODE XXX
SKIP REQUEST FOR XXXX

Check node number.

(N) NOT ENOUGH STORAGE FOR FREQUENCY SOLUTION
NEED XXXX, WITH XXXX AVAILABLE

Storage inadequate for FREQ analysis. Increase NCOM. (See Appendix J.)

(O) NOT ENOUGH STORAGE FOR MODE SHAPES
NEED XXXX, WITH XXXX AVAILABLE

Storage inadequate for MODE analysis. Increase NCOM. (See Appendix J.)

(N) NUMBER OF SHIPS = XXXX
DYNAMIC SOLUTION PRESENTLY LIMITED TO ONE SHIP

Frequency domain analysis requested with more than one ship.

(N) POSSIBLE SEQUENCE ERROR -- CASE TERMINATED

(See previous explanation)
The dynamic response file is not written.

(0) REQUESTED DYNAMIC EFFECTS WITH NO FREQUENCY DOMAIN FILE PROVIDED
ABORT ADEQUACY CHECK

Dynamic response file was not saved in the previous FREQ SAO (see Word (3) on FSOL record).

(N) SHIP MOTION DATA EXCEEDS LIMIT

	NOB	NOH	NOK	NRV
LIMITS	5	30	30	8
VALUES READ	XXX	XXX	XXX	XXX

The ship motion file has arrays larger than the dimensions in SEADYN/DSSM.

NOB = number of Froude Numbers
NOH = number of wave headings
NOK = number of wavelengths
NRV = number of roll angles

(N) SHIP MOTION FILE ERROR
WAVELENGTHS NOT IN DECREASING ORDER

Check format of ship motion file.

(S) SHIP OUTPUT REQUESTED WITH NO SHIP IN THE SYSTEM

Random response request for ship ignored when NSHIPS < 1.

(N) SPECTRUM ERROR, NO FREQUENCIES FOUND WITH SIGNIFICANT WAVE HEIGHTS

Check spectrum parameters and/or frequency range.

(F) TAPE LABEL XXXXXX DOES NOT AGREE WITH XXXXXX

The label check failed on restart. The first six characters on the RESTART title record did not agree with the check word given.

(C) TOO MANY LOAD SETS FOR DRIFT FORCE ITERATION
REFERENCE STATE NOT UPDATED

The previous static analysis used three load sets. This leaves no room to store the drift forces.

(N) UNRECOGNIZED ANALYSIS OPTION = XXXX
TRY TO GET TO NEXT CASE

Usually indicates improper numbers of cards or cards out of sequence.

10.3 Solution Option Execution Messages

(N) DIVERGENCE ON STEP XXX AT XXXXX (load factor or time)

Signals abort of MNR method after KNVRT successive step size reductions.

(C) DIVERGING NONLINEAR ITERATION AT TIME XXXX
STEP XX WITH A TIME INCREMENT OF XXXX
LAST TWO NORMS XXXX XXXX KOUNT = XX

Signals a lack of convergence in the Direct Numerical Integration time domain analysis. The various situations that lead to this message are:

(a) Acceleration more than doubled in one iteration on the component with the largest acceleration.

(b) The displacement norm exceeds 1×10^{12}.

(c) The displacement norm increased in three successive iterations.

(d) The largest displacement increment exceeds a magnitude of 1×10^{10}.

(e) The number of iterations exceeds LMITER on SOLU record. This will be followed by a reduction in time step subject to the limits of KNVRT on the SOLU record.

(C) DIVIDED STEP SIZES XXXX XXXX

A constraint overshoot was detected in the SLI or RFB incremental analyses. The step was divided into two parts as indicated by the message. The first part represents the portion of the original step used to get to the constraint. The remaining portion of the step was then taken with the constraint imposed. A full step size is used after successful completion of the divided step. Repeated divided steps with multiple constraints can cause the solution to fail. In that case small step sizes should be used.

(N) DYNAMIC SOLUTION DOES NOT CONVERGE AT TIME XXXX WITH A TIME INCREMENT OF XXXX
LAST TWO NORMS XXXX XXXX

Signals abort of DIM solution after KNVRT successive step size reductions.

(N,S) EQUATION DECOMPOSITION FAILED ON ROW XXX

The simultaneous equations in the subroutine SLVBAN (DEAD, LIVE, DYN, FREQ) or subroutine COMBAN (FREQ) are singular or sufficiently ill-conditioned to appear singular. The row of the matrix is calculated from (3*(I-1)+J) where I is the node number and J is the direction (1, 2 or 3). Check node I for proper constraint. This also occurs in poorly tensioned (soft) initial configurations. Check for zero tensions, etc. It can also occur from wildly divergent VR iterations due to improperly posed problem or poor choice of parameters (usually damping is too small).

If this occurs with the SLI or RFB methods or in the steady-state response calculations in FREQ SAO, the case is terminated.

If it occurs during a MNR analysis, various attempts are made to remove the singularity and repeat the step. Failing in these, the case is terminated.

(C) EXCESSIVE STRAIN ON ELEMENT XX ON STEP YY, XXXX

The VR iterations produced a strain larger than 1,000.0. The strain value is printed at the end of the message. This is an indication of a wildly divergent iteration resulting from a poorly posed problem and/or a poor choice of analysis parameters. Damping is increased and the iteration retried. Successive errors of this type will lead to an abort.

(N) FAILURE IN VISCOUS RELAXATION SOLUTION

All attempts to get convergent VR iterations have failed. This occurs when repeated norm increases or large strains are encountered.

(N) IMPROPERLY DEFINED DYNAMIC PROBLEM
COMPONENT XX HAS NO MASS

A material property input error has been made or zero length element has been included in the model that is not one of those reserved for payout. The component number is computed from 3*(N-1)+J where N is the node number and J is the direction number (1, 2 or 3).

(C) INCREASING RESIDUAL NORM ON STEP XX
LAST THREE RESIDUAL NORMS XXXX XXXX XXXX

The VR iterations have produced successive increases in the force residual norm. Divergence is indicated. Damping is increased and the iteration restarted.

(C) INCREASING VELOCITY NORM ON STEP XX
NORM VALUES XXXX XXXX XXXX XXXX
REPEAT STEP

Indicates a strongly increasing velocity behavior in the VR iterations. This is assumed to be a signal of divergence if it is occurring repeatedly. Damping is increased and the step is repeated. In some situations this message will occur many times without an abort. This is because it gets intermittent good results rather than successive increases. This usually means the heuristic scheme for adjusting the solution parameters does not work well for the problem. In some situations this can be avoided by selecting a larger damping at the start.

(C) LAST TWO RESIDUAL AND DISPLACEMENT NORMS XXXX XXXX XXXX XXXX
KOUNT = XXXX

The four values printed are the residual and displacement norms for iteration (i-1) and iteration i in the MNR method. This message signals an increasing norm indicative of divergence or a lack of convergence in the number of iterations given by KOUNT. This occurs when norm values are very large (greater than 10,000) or repeated increases are detected. This will be followed by a reduction of step size subject to the limits of KNVRT on the SOLU record.

(C) NEW DAMPING = XXXX

Indicates the action taken.

(C) NEW STEP SIZE = XXXX

Indicates the action taken.

(N) ROW XXX OF STIFFNESS MATRIX HAS NEGATIVE DIAGONAL TERM = XXXX

Message follows previous message on SLI or RFB methods.

(C) SINGULAR EQUATIONS WITH NUMERICAL DAMPING FACTOR OF XXXX

This message is printed from the MNR method. See Section 9.0 for further explanation. This message is printed after the three trials described in that note. This will be followed by a reduction in step size subject to the limits of KNVRT on the SOLU record.

(C) SLOW CONVERGENCE ON STEP XX
LAST FOUR VELOCITY NORMS XXXX XXXX XXXX XXXX
LAST RESIDUAL NORMS XXXX XXXX XXXX

This is a progress report on the VR iterations. Damping is reduced or step size is increased depending on the pattern of residual norm changes.

(C,N) STEP SIZE REDUCED TO XXXX ON INCREMENT XX
RESIDUAL AND DISPLACEMENT NORMS XXXX XXXX

This message follows the reduction of step size in the MNR method. It will follow either of the previous two messages when the number step size reductions is within the limits imposed by KNVRT on the SOLU record.

(C,N) TIME STEP REDUCED TO XXXX ON STEP XXXX
NORM = XXXX

This message follows the reduction of step size in a DIM solution. It will follow the previous message when the number of step size reductions is within the limits imposed by KNVRT on the SOLU record.

11.0 REFERENCES

1. Naval Civil Engineering Laboratory. Contract Report CR-82.019: SEADYN Mathematical Manual, by R. L. Webster, Consulting Engineer. Brigham City, Utah, Apr 1982 (Contract No. N62474-81-C-9391).

2. __________. Contract Report CR-82.018: SEADYN programmer's reference manual, by R. L. Webster, Consulting Engineer. Brigham City, Utah, Apr 1982 (Contract No. N62474-81-C-9391).

3. __________. Contract Report CR-82.014: Test cases for SEADYN verification, by P. E. Nordstrom, and H. Ottsen. Oxnard, Calif., Western Instrument Corporation, Apr 1982 (Contract No. N68305-80-C-0004).

4. __________. Contract Report CR-82.016: SEAPLT: A graphics post-processor for the SEADYN program, by R. L. Webster, Consulting Engineer. Brigham City, Utah, Apr 1982 (Contract No. N62474-81-C-9391).

5. __________. Contract Report CR-82.017: A compendium of cable properties, by J. F. Wadsworth, III. Oxnard, Calif., Western Instrument Corporation, Apr 1982 (Contract No. N68305-80-C-0004).

6. __________. Contract Report CR-82.015: Validation of computer models of cable system dynamics, by David B. Dillon. Rockville, Md., EG&G Washington Analytical Services Center Inc., Apr 1982 (Contract No. N68305-80-C-0020).

7. __________. Technical Note TN: Bibilography of cable dynamics reports. (to be published)

8. R. L. Webster. An application of the finite element method to the determination of nonlinear static and dynamic responses of underwater cable structures," Ph.D. thesis, Cornell University. Ithica, N.Y., Jan 1976. (Also available as General Electric Co. Report TIS-R76EMH12.)

9. __________. "On the static analysis of structures with strong geometric nonlinearities," Computers and Structures Journal, vol 11, no. 1/2, 1980, pp. 137-145.

10. R. D. Cook. Concepts and applications of finite element analysis. John Wiley & Sons, Inc. N.Y., 1974.

11. R. L. Webster and W. R. McCreight. "Analysis of deep sea moor and cable structures," Paper presented at the 11th Annual Offshore Technology Conf., Houston, Tex., May 1979 (OTC paper 3623).

12. General Electric Co., Electronic System Div. Report DSSM-5: Final engineering report, deep sea moor computer summation program," by R. L. Webster. Syracuse, N.Y., (Contract N62477-76-C-0002) Revised Sep 1973.

13. USCG Research and Development Center. Report No. CG-D-39-80: The dynamic behavior of nylon and polyester line, by K. R. Bitting. Groton, Conn., Apr 1980.

14. Naval Research Laboratory. Memorandum Report 3383: SEACON II strumming predictions, by R. A. Skop, O. M. Griffin, R. E. Ramberg. Washington, D.C., Oct 1976.

15. Naval Underwater Systems Center. Report No. 4141: Dynamics of trapezoidal cable arrays, by G. T. Griffin and K. T. Patton. New London, Conn., Mar 1972.

16. Naval Ship Research and Development Center. Report 3376: NSRDC ship-motion and sea-load computer program, by W. G. Meyers, D. J. Sheridan, and N. Salvesen. Bethesda, Md., Feb 1975.

17. Naval Facilities Engineering Command. DM-26: Design manual: Harbor and coastal facilities, Washington, D.C., Jul 1968.

18. G. Hughes. "The air resistance of ships' hulls with various types of distributions of superstructures," Institution of Engineers and Shipbuilders in Scotland, Transactions, vol. 75, pt 5, Mar 1932, pp 302-339 and discussion pp 339-367.

19. H. E. Rossell and L. B. Chapman, editors. "Principles of Naval architecture," Society of Naval Architects and Marine Engineers, New York, 1939.

20. Hydronautics, Inc. Technical Report 7096-1: Forces on ships moored in protected waters, by R. Altman. Jul 1971.

Appendix A

THE SHIP MOTION FILE

The data for the motion equations for a ship driven by harmonic waves are provided to the SEADYN program through the Ship Motion File. This set of data is assumed to be on a sequential binary file on logical unit 08. This Appendix describes the format of that file. Notations from Reference 16 are used.

The equations of motion for a ship moving in waves on a free surface are assumed to have the following form:

$$\sum_{k=1}^{6} (M_{jk} + A_{jk}) \ddot{\eta}_k + B_{jk} \dot{\eta}_k + C_{jk} \eta_k = F_j e^{iW_E t}$$

$$j = 1,\ldots,6 \qquad \text{(A-1)}$$

The terms of this equation are assumed to be provided on the ship motion file in a nondimensional form reflecting the effects of the wavelength of the surface wave and the relative heading between the wave and the ship.

The relationships between the terms of Equation A-1 and the nondimensional terms on the file are given below:

$$M_{jk} = M[L^{\eta(j,k)}][GMU(J,K)] \qquad \text{(A-2)}$$

$$A_{jk} = M[L^{\eta(j,k)}][DA(J,K)] \qquad \text{(A-3)}$$

Units: $FT^2 L^{\eta-1}$

$$B_{jk} = M(g/L)^{0.5} [L^{(j,k)}][DB(J,K)] \qquad \text{(A-4)}$$

Units: $FTL^{\eta-1}$

$$B^*_{44} = M(g/L)^{0.5} (L^2) [B44S(I_{RA})] \qquad \text{(A-5)}$$

Units: FTL

$$C_{jk} = M\,g\,[L^{\eta(j,k)-1}][DC(J,K)] \qquad \text{(A-6)}$$

Units: $FL^{\eta-1}$

$$F_j = M\,g\,[L^{m(j)-1}] \begin{cases} [BOD(J) + iBOD(J+3)] & J = 1,3,5 \\ [BEV(J) + iBEV(J+3)] & J = 2,4,6 \end{cases} \qquad \text{(A-7)}$$

Units: FL^{m-1}

where: M = Ship's mass (TMAS) Units: $FT^2 L^{-1}$

L = Ship's length (ELL) Units: L

g = gravitational acceleration (GRAV) Units: LT^{-2}

I_{RA} = the Ith roll angle index

$\eta(j,k) = m(j) + m(k)$

$m(j)$ = 0 for $j \leq 3$
1 $j \geq 3$

$i = -1$

The coefficients are assumed to be linearized for unit motion amplitude and wave height. Typical units for the Ship Motion File are:

F - long tons (2,240 pounds)
L - feet
T - seconds

Angles and angular responses are assumed to be in radians.

In addition to the ship motion coefficients, the Ship Motion File provides coefficients for estimating the steady-state approximations of the second order, wave-induced drift forces. These dimensionless coefficients are used to estimate the drift forces once the magnitudes of the ship responses, η_j, are obtained. The drift force components are:

Surge (Units - F):

$$F_x = M(g) \sum_{j=1}^{7} \sum_{k=1}^{7} \frac{\bar{\eta}_j}{L^{1-m(j)}} \frac{\bar{\eta}_k^*}{L^{1-m(k)}} TX(J,K) \qquad \text{(A-8)}$$

Sway (Units - F):

$$\bar{F}_y = M(g) \sum_{j=1}^{7} \sum_{k=1}^{7} \frac{\bar{\eta}_j}{L^{1-m(j)}} \frac{\bar{\eta}_k^*}{L^{1-m(k)}} TY(J,K) \tag{A-9}$$

Yaw (Units - FL):

$$\bar{M}_z = M(g)(L) \left[I_m \left\{ \sum_{j=1}^{7} \sum_{k=1}^{7} \frac{\bar{\eta}_j}{L^{1-m(j)}} \frac{\bar{\eta}_k^*}{L^{1-m(k)}} TM(J,K) \right\} \right.$$

$$\left. + \eta_o R_e \left\{ \sum_{j=1}^{7} \sum_{k=1}^{7} \frac{\eta_j}{L^{1-m(j)}} TP(J) \right\} \right] \tag{A-10}$$

where: η_o = wave amplitude (real number)

$$\bar{\eta}_j = \begin{Bmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_6 \\ \eta_0 \end{Bmatrix} \text{ augumented ship response}$$

$m(j)$ = 0 for j = 1,2,3,7
1 for j = 4,5,6

( )* means complex conjugate

This presumes the wave motion is given by $\eta_o e^{iwt}$, where η_o is real and the ship response is $\eta_i e^{iwt}$.

The coordinate system presumed for ship's motions and forces is a righthand cartesian system with its origin at the ship's center of gravity, its X-axis positive aft, its Y-axis positive starboard, and its Z-axis positive upward. The angular convention for the relative heading between the ship and the waves assumes the following:

Wave Heading	Description
0°	Following seas
90°	Beam seas with waves traveling from port to starboard
180°	Head seas
270°	Beam seas with waves traveling from starboard to port

The differences between the wave heading convention and the ship's coordinates should be noted.

The Ship Motion File is organized in logical records. The specific contents of each record will be described below. There are seven distinct record types. The first two records contain data which are independent of wave heading or wavelength. Record types 3 through 7 are dependent on heading and wavelength and are repeated in a nested-loop fashion. The overall form is:

Record 1

Record 2

Loop on Wave Length
- Loop on Wave Heading
  - Record 3
  - Record 4 (presented only on first loop per wavelength in the NCEL file)
  - Record 5
  - Record 6
  - Record 7

The wave headings are assumed to be listed in decreasing order with +180° being the largest allowed. The interpolation routines assume the values given for +180° will be used for -180°, therefore, data for -180° need not be given.

The wavelengths are assumed to be listed in decreasing order (i.e., increasing frequency order).

The individual records of the file are described in terms of the FORTRAN read/write lists associated with each record.

RECORD 1 NAME 1, NAME 2, NAME 3

Three Hollerith variables providing identifying data.

RECORD 2 (TITO(I), I=1, 12), WORD, WORD 2, WORD 3, ELL, BEAM, DRAFT, TVOL, TMAS, TPST, ZG, CBV, NOB, (FN(I), I=1, NOB), NOH, (HDG1(I), I=1, NOH), NOK, (BAM(I), I=1, NOK), VNY, GRAV, NRV, (RANG(I), I=1, NRV), ((GMU(I,J), J=1,6), I=1,6), ((DC(I,J), J=1,6), I=1,6)

TITO = Hollerith title consisting of 12 six-character words
WORD = length unit label (six-character Hollerith)
WORD 2 = force unit label (six-character Hollerith)
WORD 3 = moment unit label (six-character Hollerith)
ELL = ship's length (L)
BEAM = beam (L)
DRAFT = draft (L)
TVOL = ship's volume is obtained from $(ELL/2)^3 \cdot TVOL$
TMAS = ship's mass (FT^2L^{-1})
TPST = longitudinal distance from c.g. to forward most station is obtained from (ELL/2) · TPST
ZG = vertical distance from water line to c.g., (+ up) (L)
CBV = vertical distance from water line to center of buoyancy (+ up) is obtained from ELL · CBV
NOB = number of ship speeds (SEADYN expects only one)

FN(I) = the Froude numbers for each speed (only one expected)
NOH = number of wave headings
HDGI(I) = the wave headings listed in decreasing order starting with 180° and proceeding no further than -179°
NOK = number of wavelengths
BAM(I) = nondimensional wavelength in decreasing order, λ = ELL . BAM(I)
VNY = fluid viscosity (L^2T)
GRAV = gravitational acceleration (LT^{-2})
NRV = number of roll angles
RANG(I) = the values of roll angles (radians) listed in increasing order
GMU(I,J) = the nondimensional mass matrix
DC(I,J) = the non-dimensional hydrostatic restoring matrix

RECORD 3 MM,HDGI(MM), JJ, FN(JJ), LL, BAM(LL)
MM = heading number
JJ = speed number

RECORD 4 ((DA(I,J), J=1,6), I=1,6), ((DB(I,J), J=1,6), I=1,6)
DA(I,J) = the nondimensional added mass matrix for that wavelength
DB(I,J) = the nondimensional wave damping matrix for that wavelength

RECORD 5 (BOD(I), BOD(I+3), BEV(I), BEV(I+3), I=1,3)
BOD, BEV = the nondimensional wave force coefficients

RECORD 6 (B44S(I), I=1, NRV)
B44S(I) = the nonlinear roll damping terms that are added to the linearized damping matrix depending on the size of the roll angle (nondimensional)

RECORD 7 ((TX(I,J), J=1,7), I=1,7), ((TY(I,J), J=1,7), I=1,7), ((TM(I,J), J=1,7), I=1,7), (TP(I), I=1,7)

The nondimensional drift force coefficients.

Appendix B

SHIP LOAD DATA FILE

The purpose of the ship load data file is to provide static loads for ships and other rigid bodies for wind and surface currents at various headings. The file can be constructed and saved as a library of static load functions; Appendix F describes the procedure used in DSSM to scale loads between similar ships using this library of files. Input is provided in a FORTRAN rigid format. Each ship load data set consists of:

SHIP LOAD TITLE RECORD
UNIT LABEL RECORD
SHIP PARAMETERS
WIND RECORD
WIND HEADING RECORD(S)
WIND FORCE RECORD(S)
SURFACE CURRENT RECORD
SURFACE CURRENT HEADING RECORD(S)
SURFACE CURRENT FORCE RECORD(S)

Each of the data records is described below.

The free-field input routine will automatically read these rigid format records when they are placed between the ( ) delimitor records (see Section 5.0). These data can appear anywhere in the input stream after the first title record set. When SEADYN encounters NSFILE>0, these rigid format data are processed into the ship load data file. This is to be done only once in any job deck. It is not possible to stack new problem cases using NEW which intend to define two distinct ship load data files.

SHIP LOAD TITLE RECORD (12A6)*

Columns	Variable Name	Description
1-72	SHPCAP	Any descriptive title

UNIT LABEL RECORD (A6, 4X, A6, 4X, A6, 4X, A6)

Columns	Variable Name	Description
1-6	WLBL	Wind force label (e.g., "TONS," "POUNDS")
11-16	CLBL	Current force label
21-26	LLBL	Length label (e.g., "FEET," "METERS")
31-36	VLBL	Velocity label (e.g., "KNOT," "FT/SEC")

NOTE: These labels are output with the ship data as a reminder of the units used. They are used for no other purpose.

SHIP PARAMETERS (8E10.0)

Columns	Variable Name	Description
1-10	TSLT	Total ship length (L)
11-20	TSAE	End projected wind area (L^2)
21-30	TSAS	Side projected wind area (L^2)
31-40	TSWL	Water line length (L)
41-50	TSB	Beam at midships (L)
51-60	TSD	Draft at midships (L)
61-70	TSDSP	Volume displacement (L^3)
71-80	TSAP	Propeller projected area (L^2)

*FORTRAN format specification.

WIND RECORD (2I5, 6E10.0)

Columns	Variable Name	Description
1-5	NWIND	Number of wind velocity tables (max is 5)
6-10	NTHETW	Number of headings in each table (max is 20)
11-20	SCALE	Test scale factor (A means $1/A^{th}$ scale)
21-30	WNDVEL(1)	First wind velocity (smallest) (LT^{-1})
.	.	
.	.	
.	.	
61-70	WNDVEL(5)	Fifth wind velocity

WIND HEADING RECORD(S) (8E10.0)

(Repeat as required to get NTHETW entries)

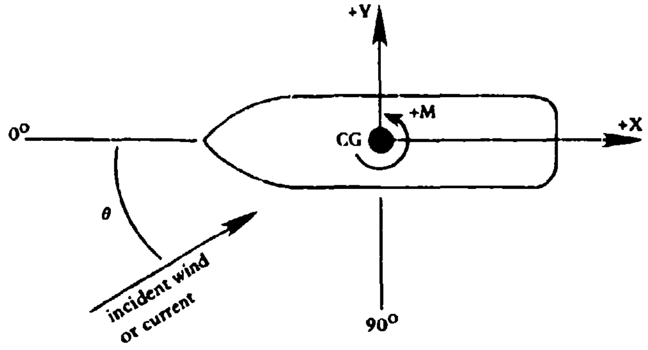
Columns	Variable Name	Description
1-10	WNDHED(1)	First wind heading (degrees)
11-20	etc.	

NOTES

1. Headings should be between 0° and 360°, listed from the smallest to the largest.

2. If the largest value is 180°, the loading functions are assumed to be symmetric about 180° for the end forces and skew-symmetric for the side forces and yaw moments.

3. The angle is measured relative to the ship's local coordinate system, which is illustrated below:



WIND FORCE RECORD(S) (3E10.0)

Columns	Variable Name	Description
1-10	WNDCOE(I,1,J)	End force for I^{th} heading and J^{th} velocity (F)
11-20	WNDCOE(I,2,J)	Side force for I^{th} heading and J^{th} velocity (F)
21-30	WNDCOE(I,3,J)	Moment for I^{th} heading and J^{th} velocity (FL)

(I varies before J)

SURFACE CURRENT RECORD(S) (2I5, 6E10.0)

1-5	NCRNT	Number of current tables
6-10	NTHETC	Number of headings in each current table
11-20	TDEPTH	Test water depth (L)
21-30	CURVEL(1)	First current velocity (smallest) (LT^{-1})
.	.	
.	.	
.	.	
61-70	CURVEL(5)	Fifth current velocity

SURFACE CURRENT HEADING RECORD(S) (8E10.0)

(Repeat as required to get NTHETC entries)

Columns	Variable Name	Description
1-10	CURHED(1)	First Current Heading (Degrees)
11-20	etc.	

(See notes for WIND HEADING RECORD)

SURFACE CURRENT FORCE RECORD(S) (3E10.0)

(One record for each heading repeated for each velocity)

1-10	CURCOE(I,1,J)	End force for I^{th} heading and J^{th} velocity (F)
11-20	CURCOE(I,2,J)	Side force for I^{th} heading and J^{th} velocity (F)
21-30	CURCOE(I,3,J)	Moment for I^{th} heading and J^{th} velocity (FL)

(I varies before J)

The ship load file contains one logical record for each ship catalogued on the file, and it is written in a binary form with the following FORTRAN statement:

```
WRITE(10)NWIND,NTHETW,WNDVEL,WNDHED,WNDCOE,SCALE,NCRNT,NTHETC,
CURVEL,CURHED,CURCOE,TDEPTH,TBLOCK,TSLT,TSAE,TSAS,TSWL,TSB,TSD,
TSDSP,TSAP,SHPCAP,WLBL,CLBL,LLBL,VLBL
```

A number of the items in the list are arrays, and they are written in their entirety using the implied DO-LOOP feature of FORTRAN I-O statements. A description of each item, including the dimensions of the arrays, is given below:

VARIABLE	DESCRIPTION
NWIND	Number of wind velocity tables
NTHETW	Number of headings in each wind table
WNDVEL(5)	Array of wind velocities
WNDHED(20)	Array of wind headings
WNDCOE(20,3,5)	Array of wind load coefficients giving values for up to 20 headings for end force, side force, and yaw moment for up to five wind velocities
SCALE	Scale for wind load tests (A means 1/Ath scale)
NCRNT	Number of current velocity tables
NTHETC	Number of headings in each current table
CURVEL(5)	Array of current velocities
CURHED(20)	Array of current headings
CURCOE(20,3,5)	Array of current load coefficients giving values for up to 20 headings for end force, side force, and yaw moment for up to five current velocities
TDEPTH	Water depth for test
TBLOCK	Ship's block coefficient

TSLT	Total ship length
TSAE	End projected wind area
TSAS	Side projected wind area
TSWL	Water line length
TSB	Beam at midships
TSD	Draft at midships
TSDSP	Volume displacement
TSAP	Propeller projected area
SHPCAP(12)	Title of 12 six-character Hollerith words
WLBL	Wind force label, six-character Hollerith word
CLBL	Current force label, six-character Hollerith word
LLBL	Length label, six-character Hollerith word
VLBL	Velocity label, six-character Hollerith word

Appendix C

BUILT-IN DRAG COEFFICIENTS

In the event that fluid loading is required and the drag coefficients are not given by the subroutine DRAGCO, the program will select default drag coefficients from a set of built-in functions. The default coefficients are used whenever INDRAG = 0 (PROB record, Section 6.3), regardless of the drag coefficient codes on the cable materials and/or the lumped bodies. If INDRAG ≠ 0, the default coefficients will be used for those components that have zero drag coefficient codes.

The default coefficients are:

Spherical Bodies

$$\text{Reynolds number, } R_e = \frac{V\ d}{\upsilon} = \frac{\text{velocity x body diameter}}{\text{kinematic viscosity}} \qquad \text{(C-1)}$$

$$\begin{aligned}
C_D &= 0 \text{ for } R_e \leq 0.1 \\
C_D &= 0.004 = 13.46/(R_e)^{0.5} && \text{for } 0.1 < R_e \leq 1000 \\
C_D &= 0.47 && \text{for } 1000. < R_e \leq 10^5 \\
C_D &= 0.12 && \text{for } R_e > 10^5
\end{aligned} \qquad \text{(C-2)}$$

Cylindrical Bodies and Cable Elements

$$R_e = \frac{V_N\ d}{\upsilon} = \frac{\text{normal velocity x body diameter}}{\text{kinematic viscosity}} \qquad \text{(C-3)}$$

$$R_{eT} = \frac{V_T\ d}{\upsilon} = \frac{\text{tangential velocity x body diameter}}{\text{kinematic viscosity}} \qquad \text{(C-4)}$$

$$
\begin{aligned}
C_N &= 0 && \text{for } R_e \leqq 0.1 \\
C_N &= 0.45 + 5.93/(R_e)^{0.33} && \text{for } 0.1 < R_e \leqq 400 \\
C_N &= 1.27 && \text{for } 400 \leqq R_e \;\; 10^5 \\
C_N &= 0.3 && \text{for } R_e > 10^5
\end{aligned}
\qquad \text{(C-5)}
$$

$$
\begin{aligned}
C_T &= 0 && \text{for } R_{eT} \leqq 0.1 \\
C_T &= 1.88/(R_{eT})^{0.74} && \text{for } 0.1 < R_{eT} \leqq 100.55 \\
C_T &= 0.062 && \text{for } R_{eT} > 100.55
\end{aligned}
\qquad \text{(C-6)}
$$

Appendix D

MOORING COMPONENT INVENTORY

The mooring component inventory contains data tables for the following:

Anchors:

Navy standard stockless
NAVSHIPS lightweight
NAVFAC STATO

Buoys:

Bar riser chain type

Chain:

Steel stud-link

Hawsers:

Samson braids -- 2-in-1® Nylon
2-in-1® Power Braid
2-in-1® Stable Braid
12 Strand Blue Streak®

The inventory lists weights, buoyancies, and strengths in pounds. Lengths and buoy dimensions are in feet. Hawser and chain sizes are in inches. These units may be converted to those needed in the analysis by providing the appropriate conversion factors on the INVE data record (see Section 6.9). The contents of the inventory can be obtained by setting the print flag in Word (4) of the INVE record. The listing of the present inventory is presented below.

C O M P O N E N T   I N V E N T O R Y

ANCHOR TYPE = NAVY STD STOCKLESS

WEIGHT	FED. STOCK NO.	HOLD. POWER (FIRM SAND)
.30000E+04	C2040-516-7758	.21000E+05
.50000E+04	C2040-516-7757	.35000E+05
.60000E+04	C2040-516-7756	.42000E+05
.70000E+04	C2040-516-7755	.49000E+05
.90000E+04	C2040-516-7754	.63000E+05
.10000E+05	C2040-272-2244	.70000E+05
.13000E+05	C2040-272-2245	.91000E+05
.14500E+05	C2040-272-2246	.10150E+06
.18000E+05	C2040-516-7753	.12600E+06
.20000E+05	C2040-272-2247	.14000E+06
.25000E+05	C2040-272-2242	.17500E+06
.30000E+05	C2040-272-2243	.21000E+06
.40000E+05	C2040-277-2423	.28000E+06

ANCHOR TYPE = NAVSHIP(LWT)

WEIGHT	FED. STOCK NO.	HOLD. POWER (FIRM SAND)
.10000E+03	H2040-377-8600	.28374E+04
.15000E+03	H2040-377-8601	.39565E+04
.20000E+03	H2040-377-8602	.50091E+04
.30000E+03	H2040-377-8603	.69848E+04
.50000E+03	H2040-377-8604	.10619E+05
.75000E+03	H2040-377-8605	.14807E+05
.10000E+04	H2040-377-8606	.18746E+05
.15000E+04	H2040-377-8607	.26140E+05
.20000E+04	H2040-377-8608	.33095E+05
.25000E+04	H2040-377-8609	.39740E+05
.30000E+04	H2040-377-8610	.46148E+05
.40000E+04	H2040-377-8611	.58426E+05
.50000E+04	H2040-378-5633	.70157E+05
.60000E+04	H2040-378-5634	.81470E+05
.10000E+05	H2040-377-8612	.12385E+06
.13000E+05	H2040-377-8613	.15358E+06

ANCHOR TYPE = NAVFAC STATO

WEIGHT	FED. STOCK NO.	HOLD. POWER (FIRM SAND)
.20000E+03	2CF2040-800-9659	.40000E+04
.30000E+04	2CF2040-702-7864	.60000E+05
.60000E+04	2CF2040-702-6785	.12000E+06
.90000E+04	2CF2040-702-6786	.18000E+06
.12000E+05	2CF2040-702-6787	.24000E+06
.15000E+05	2CF2040-801-7938	.30000E+06

STEEL STUD-LINK CHAIN

SIZE	STRENGTH	WEIGHT/LENGTH
.7500	48550.	5.5556
.8750	65280.	7.7778
1.0000	84500.	9.4444
1.1250	106080.	12.2222
1.2500	130070.	15.0000
1.3750	156330.	17.7778
1.5000	185060.	21.1111
1.6250	216030.	24.4444
1.7500	249210.	28.3333
1.8750	284540.	32.7778
2.0000	322000.	36.6667
2.1250	361530.	41.1111
2.2500	403100.	46.6667
2.3750	446660.	51.6667
2.5000	492190.	57.7778
2.6250	539620.	63.3333
2.7500	588930.	70.0000
2.8750	640070.	76.6667
3.0000	693000.	83.3333
3.1250	747680.	91.1111
3.2500	804070.	98.3333
3.3750	862130.	106.1111
3.5000	921810.	114.4444
3.6250	983080.	122.7778
3.7500	1045900.	131.1111
3.8750	1110210.	140.0000
4.0000	1176000.	148.8889
4.1250	1234200.	158.8889
4.2500	1311790.	170.0000
4.3750	1381330.	183.8889
4.5000	1452930.	198.3333

B U O Y   D A T A

BUOY TYPE =   BAR RISER CHAIN

O. D.	HEIGHT	WEIGHT	NOM. BUOYANCY	MAX. BUOYANCY	FED. STOCK NO.
6.53125	4.03125	2200.	3562.	4300.	C2050-223-3657
7.03125	5.03125	2500.	5835.	7518.	C2050-264-4497
9.50000	5.00000	7700.	7420.	10445.	C2050-223-3665
10.50000	6.50000	9600.	14414.	20879.	C2050-223-3662
10.50000	7.50000	10100.	17608.	25921.	C2050-264-44981

H A W S E R   D A T A

SIZE	2-IN-1 NYLON STRENGTH	2-IN-1 NYLON WEIGHT/100L	2-IN-1 POWER BRAID STRENGTH	2-IN-1 POWER BRAID WEIGHT/100L	2-IN-1 STABL BRAID STRENGTH	2-IN-1 STABL BRAID WEIGHT/100L	12 ST. BLUESTREAK STRENGTH	12 ST. BLUESTREAK WEIGHT/100L
.5000	8300.	6.60	7500.	5.90	7500.	7.90	6300.	6.40
.7500	18000.	15.00	16000.	13.00	16000.	18.00	13600.	14.00
1.0000	31300.	26.00	28400.	24.00	28400.	32.00	23500.	25.00
1.0625	36500.	31.00	33200.	28.00	33200.	37.00	0.	.10
1.1250	42000.	36.00	38000.	32.00	38000.	43.00	31500.	33.00
1.2500	47700.	41.00	43600.	37.00	43600.	49.00	36000.	38.00
1.3125	54000.	47.00	49000.	42.00	49000.	56.00	40600.	44.00
1.5000	67500.	60.00	61400.	53.00	61400.	71.00	50800.	55.00
1.6250	82600.	74.00	75000.	66.00	75000.	88.00	62200.	68.00
1.7500	99000.	89.00	90000.	80.00	90000.	106.00	74400.	82.00
2.0000	117000.	106.00	106000.	95.00	106000.	126.00	86000.	99.00
2.1250	136000.	124.00	124000.	111.00	124000.	148.00	102000.	115.00
2.2500	156000.	144.00	142000.	129.00	142000.	172.00	117000.	134.00
2.5000	178000.	165.00	162000.	148.00	162000.	197.00	134000.	153.00
2.6250	202000.	188.00	183000.	168.00	183000.	224.00	151000.	175.00
2.7500	226000.	212.00	205000.	190.00	205000.	253.00	170000.	197.00
3.0000	252000.	238.00	229000.	213.00	229000.	284.00	190000.	222.00
3.2500	308000.	294.00	280000.	263.00	280000.	350.00	232000.	274.00
3.6250	369000.	356.00	336000.	318.00	336000.	424.00	278000.	332.00
4.0000	436000.	423.00	396000.	379.00	396000.	504.00	327000.	395.00
4.2500	504000.	497.00	461000.	444.00	461000.	592.00	381000.	463.00
4.6250	586000.	576.00	531000.	515.00	531000.	686.00	439000.	537.00
5.0000	666000.	662.00	606000.	592.00	606000.	788.00	500000.	617.00

Appendix E

RESTART FILE STRUCTURE

The SEADYN program creates up to three restart files (one each for the DEAD, LIVE, and DYN subanalyses). Multiple selections of a subanalysis type simply extends the file unless a rewind is signaled on the SAVE data record. A counter is provided for each of the files to keep track of how many restart records have been written. The FORTRAN file codes used are:

01 - DEAD
02 - LIVE (and TSSS)
03 - DYN

Each time the file is rewound, the counter for that file is set to zero, and a label record is written. The write statement is:

```
WRITE (NFILE) (TITLE (I), I = 1, NHED), NINA, NPRECZ
```

where: TITLE = page heading title card for the run (Each word is assumed to have 10 characters.)
NINA = size of unlabeled common when the file is saved
NPRECZ = precision number for floating point numbers
1 = single precision
2 = double precision
NHED = the number of words in the title = 8

Each restart save operation uses the following write statement:

```
 WRITE(NTAPE) (A(I),I=1,NINA),(B(I),I=1,NINB),(C(I),I=1,NINC),
+(RL(I),I=1,NINRL),(P(I),I=1,NINPO),(T(I),I=1,NINT),(SH(I),I=1,NINS
+HP),(DP(I),I=1,NINDSP),(STM(I),I=1,NINSTM),NFILE
. ,(IAABU(I),I+1,NINBI),(IAACA(I),I=1,NINCI)
. ,(IAACL(I),I=1,NINCLI),(IAADS(I),I=1,NINDSI)
. ,(IAAPO(I),I+1,NINPOI),(IAASHP(I),I=1,NINSHI)
. ,(IAASTM(I),I=1,NINSTI),(IAATIM(I),I=1,NINTMI)
+,DLD,WLD,DYN,CHECKR,NOVEL,NOITER,NOFLUD,NOLOAD,FEEDBK,POUT,
+ REFUP,STEPUP
```

where the arrays are defined by:

```
COMMON/ACOM/ A(1)
COMMON/BUOYS/ B(1)
COMMON/CABLE/ C(1)
COMMON/CONTRL/ RL(1)
COMMON/DSPCON/ DP(1)
COMMON/PAYOUT/ P(1)
COMMON/TIMED/ T(1)
COMMON/SHIPS/ SH(1)
COMMON/STRUM/ STM(1)
COMMON/IBUOYS/ IAABU(1)
COMMON/ICABLE/ IAACA(1)
COMMON/ICNTRL/ IAACL(1)
COMMON/IDSPCN/ IAADS(1)
COMMON/IPAYOT/ IAAPO(1)
COMMON/ISHIPS/ IAASHP(1)
COMMON/ISTRUM/ IAASTM(1)
COMMON/ITIMED/ IAATIM(1)

COMMON/LOGIC/DLD,WLD,DYN,CHECKR,NOVEL,NOITER,NOFLUD,NOLOAD,
1 FEEDBK,POUT,REFUP,STEPUP
```

The sizes of the arrays are given by:

```
COMMON/SIZE/ NINA,NINB,NINRL,NINDSP,NINPO,NINSHP,NINSTM,NINT,
1 NINC,NCOM,IFILE(4),NPRECZ,NINBI,NINCI,NINCLI,NINDSI,
2 NINPOI,NINSHI,NINSTI,NINTMI
```

These sizes are identified in Reference 2, and count the number of single-precision words to be read/written. This count is adjusted for double- or single-precision conditions. The arrays are always to be treated as single precision in the RESTART routine even though they can contain mixed-double precision and fixed-point data in the rest of the program. The actual contents of the labeled common blocks are defined in the calling program with the appropriate word format.

Appendix F

SHIP'S LOADING FUNCTIONS

This Appendix describes the tabular approach used in SEADYN to obtain static loads on a ship subjected to winds and surface currents.

The ship loads are assumed to be applied through the center of gravity of the ship. Three sources of loading are considered: wind loads, surface current loads, and point loads representative of working loads. The point loads are specified through the normal loading options (see Section 7.1.7, LOAD record). It is assumed that the point loads do not change their magnitude or global directions as the ship moves to a new position. Specification of the wind and current loadings is somewhat more complicated. The SEADYN program provides two approaches to defining these loads. The first approach is in the form of loading tables that give loads versus ship's heading relative to the flow. The second approach uses approximate analytical expressions, which are described in Appendix G.

The tabular approach is based on the procedures given in NAVFAC's Design Manual 26 (Ref 17). The DM-26 approach utilizes experimental measurements for the forces and moments for various headings of wind and current for a set of "representative" vessels. Similarity scaling is then applied to get loading values for ships other than the test models.

The DM-26 procedure begins with a set of load measurements obtained from subscale tests on a representative ship's model or any other available source. The measurements give values for the lateral and longitudinal forces and yaw moment versus flow heading and flow velocity. These measurements represent the combined effects of such phenomena as profile and friction drag, lift-induced side forces, and shifts in the center of pressure. Tables of these measurements can be specified as either input to the program or as a special ship loading file previously generated and saved for subsequent referencing by SEADYN. This saved file is assumed to be on logical unit 03 (see Appendix J).

Given the headings of wind and surface current relative to the ship, the loads are obtained by linear interpolation in the tables. In the event that there are tables provided for more than one velocity, the table for the velocity nearest the one specified in the analysis will be used. This is determined by comparing the squares of the velocity ratios.

After the load coefficients are obtained from the tables for the given heading, they must be scaled to account for differences in the conditions modeled in the test and for those being analyzed. The scaling accounts for differences in flow velocity, water depth, and ship geometry. The formulas for adjusting for these effects are given in DM-26 and are restated here for completeness.

WIND

$$F_s = C_f V^2 F_{ms} (A_s/A_{ts}) \qquad \text{(F-1)}$$

$$F_e = C_f V^2 F_{me} (A_e/A_{te}) \qquad \text{(F-2)}$$

$$M_w = C_m V^2 M_m (A_s/A_{ts}) (L/L_t) \qquad \text{(F-3)}$$

where:

- F_s = lateral force on ship
- F_e = longitudinal force on ship
- M_w = yawing moment on ship
- F_{ms} = lateral force on model
- F_{me} = longitudinal force on model
- M_m = yawing moment on model
- V = wind velocity
- A_s = side-projected area above the water line of ship being analyzed
- A_{ts} = side-projected area above the water line of modeled ship
- A_e = end-projected area above the water line of ship being analyzed
- A_{te} = end-projected area above the water line of modeled ship
- L = length of ship being analyzed
- L_t = length of modeled ship

$$C_f = \frac{S^2}{V_T^{\,2}} \qquad \text{(F-4)}$$

$$C_m = \frac{S^3}{V_T^{\,2}} \qquad \text{(F-5)}$$

- S = linear scale of the model (e.g., 50 to 1; $S = 50$)
- V_T = wind velocity used in model test

CURRENT

$$h_2 = h_1 L_{W2}/L_{W1} \tag{F-6}$$

$$V_1 = V_2 (L_{W1}/L_{W2})^{1/2} \tag{F-7}$$

$$F_2 = F_1 \Delta_2/\Delta_1 \tag{F-8}$$

$$M_2 = M_1 (\Delta_2/\Delta_1)(L_{W2}/L_{W1}) \tag{F-9}$$

where: h = depth of water

V = velocity of current

L_W = water line length of vessel

F = lateral or longitudinal resisting force

Δ = displacement

M = yaw resisting moment

Subscript 1 denotes the full-scale vessel for which the model test was made, and subscript 2 denotes the vessel being analyzed.

When the velocity from Equation F-7 does not correspond to one of the tables given for the model test, then the forces and moments must be selected from the tables corresponding to the velocity nearest the value of V_1 in Equation F-7. It will then be necessary to adjust the values by the square of the ratio of the V_1 velocity and the velocity represented in the tables, V_{t1}.

It is quite likely that the depth at the proposed mooring site will not be the same as that obtained for h_2 in Equation F-6. In that event, a correction for depth is required. DM-26 suggests that the correction be made assuming an inverse relationship with the side resistances at the two depths in question. The curves given in Graph 124 (EC-2) of DM-26 are used along with Equation F-6 for this purpose. The data are given in tabular form and the side resistances are obtained by logarithmic interpolation. The resistance for a depth greater than that in the table will be the last value in the table.

The adjustments for current velocity and depth are summarized by the following equations:

$$F'_{s2} = \left[f_h \frac{V_1^2}{V_{t1}^2}\right] F_{s2} \tag{F-10}$$

$$F'_{e2} = \left[f_h \frac{\Delta_2}{\Delta_1} \frac{V_1^{\,2}}{V_{t1}^{\,2}} \right] \left(F_{e1} - \frac{1}{2} \rho\, C_p\, A\, V_{t1}^{\,2} \right) \qquad \text{(F-11)}$$

$$+ \frac{1}{2} \rho\, C_p\, A\, V_2^{\,2}$$

$$M'_2 = \left[f_h \frac{V_1^{\,2}}{V_{t1}^{\,2}} M_2 \right] \qquad \text{(F-12)}$$

where: f_h = the depth scaling factor

V_{t1} = the velocity at which the test data was obtained

A = the propeller projected area

C_p = the propeller drag coefficient

ρ = fluid density

The primes indicate the value has been adjusted to the desired conditions for the mooring site. Equation F-11 reflects the adjustment in the longitudinal force recommended by DM-26 with the assumption that $(1/2)\ \rho\ C_p = 2.88$ (with V in knots). Assuming the specific weight of seawater is 64 lb/ft^3 and the acceleration due to gravity is 32.2 ft/sec^2, then $C_p = 1.00$. The form using $(1/2)\ \rho\ C_p$ rather than 2.88 is required to make the procedure dimensionally independent.

Appendix G

BUILT-IN LOAD FUNCTIONS FOR SHIPS

This Appendix describes the analytical approach for obtaining static loads on a ship subjected to winds and surface currents. The approximate analytical expressions for ship's loading are based primarily on the work of Hughes (Ref 18), standard Naval architectural formulas (Ref 19), and Altman (Ref 20).

The wind loading is given by:

$$F = K \rho_a V^2 (A_s \sin^2 \theta + A_e \cos^2 \theta) \cos (\alpha - \theta) \qquad \text{(G-1)}$$

where: K = constant, 0.6

F = resultant wind force

ρ_a = mass density of air

V = wind velocity

θ = wind heading relative to the bow

α = heading of the resultant wind force relative to the bow

A_s = side projected area of ship above water line

A_e = end projected area of ship above water line

The heading of the resultant wind force, α, is approximated as a function of θ in a 7^{th} order polynomial as follows:

$$\begin{aligned} \alpha = \; & 0.0715608 + 7.954381\,\theta - 0.3254561\,\theta^2 \\ & + 0.0073131\,\theta^3 - 9.3966 \times 10^{-5}\,\theta^4 \\ & + 6.85008 \times 10^{-7}\,\theta^5 - 2.6323 \times 10^{-9}\,\theta^6 \\ & + 4.1453 \times 10^{-12}\,\theta^7 \end{aligned} \qquad \text{(G-2)}$$

In Equation G-2 both θ and α are measured in degrees.

The distance between the ship forward perpendicular and the center of wind pressure, X_{cp}, can be approximated as a polynomial function of the wind direction, θ. This relationship is:

$$\frac{X_{cp}}{L} = 0.2004112 + 0.0048641\,\theta - 4.52442 \times 10^{-5}\,\theta^2 + 5.45736 \times 10^{-7}\,\theta^3 - 3.78789 \times 10^{-9}\,\theta^4 + 1.02881 \times 10^{-11}\,\theta^5 \qquad \text{(G-3)}$$

Here, as above, θ is measured in degrees. Also, L is the overall ship length. The yawing moment due to wind is then approximated by:

$$M_W = F\,L\,\sin\alpha\left(\frac{1}{2} - \frac{X_{cp}}{L}\right) \qquad \text{(G-4)}$$

Analytical expressions for the resistances from current effects utilize the approach presented by Altman in Reference 20. These expressions are summarized below:

$$F_s = F_{s\infty}\left(1. + \frac{10}{(h/H)^2 - 1.}\right) \qquad \text{(G-5)}$$

$$F_{s\infty} = 0.215\,\rho_w\,V^2\,L_w\,H\,\sin\theta \qquad \text{(G-6)}$$

$$F_e = \frac{1}{2}\rho_w\,V^2(S_w\,C_R + A_p\,C_p)\cos\theta \qquad \text{(G-7)}$$

$$M = F_s\,L_{CP} \qquad \text{(G-8)}$$

where: F_s = lateral current force at the specified water depth

$F_{s\infty}$ = lateral current force in deep water

F_e = longitudinal current force

m = yaw current moment

∇ = displaced volume

C_s = wetted surface coefficient, input on SHIP record

C_R = hull resistance coefficient, input on SHIP record or calculated as $C_r + C_f + 0.0005$ (G-9)

C_r = residuary resistance coefficient (see following discussion)

C_f = frictional resistance coefficient,

$$= \frac{0.456}{(\log_{10} R_e)^{2.58}} - \frac{1,700}{R_e} \qquad R_e \geqq 5 \times 10^5 \qquad \text{(G-10)}$$

$$= 0.002 \qquad R_e < 5 \times 10^5$$

$$S_w = C_s \nabla L_W \qquad \text{(G-11)}$$

R_e = Reynolds number for the hull (based on longitudinal component of flow and ship length)

L_{CP} = distance from midships to hull center of pressure

$$= L[\bar{L}_{90} + 0.00226\ (\theta - 90°)] \qquad \text{for } 0° \leqq \theta \leqq 180°$$

$$= L[\bar{L}_{90} + 0.00226\ (\theta - 270°)] \qquad \text{for } 180° \leqq \theta \leqq 360° \qquad \text{(G-12)}$$

$\bar{L}_{90}$ = ratio of distance to center of pressure at $\theta = 90°$ to the distance to the center of hull side area

h = water depth

H = ship draft

ρ_w = mass density of water

L_w = waterline length of ship

A_p = propeller projected area

C_p = propeller drag coefficient

Several of these terms require further discussion. The hull resistance coefficient, C_R, represents the sum of various coefficients for different sources of hull resistance. This coefficient can be input or calculated in the computer program. When no input is given for C_r, it will be calculated as the sum of a residuary resistance coefficient, a frictional resistance coefficient, and a fouling/surface effect coefficient. The fouling/surface effect coefficient is given an arbitrary value of 0.0005. The frictional coefficient is calculated from Equation G-10 and the residuary coefficient is obtained from linear interpolation of a digitized form of Figure 38 of Reference 20. This method of obtaining C_r is limited to low flow velocities since wave-making resistances are ignored.

The longitudinal location of the center of pressure for a hull skewed with respect to the flow is estimated by Equation G-12. This requires an estimate of the ratio of the distance to the center of pressure and the center of area for beam flow, $\bar{L}_{90}$. This factor is estimated by linear interpolation between the values for the ship's DD-692 and EC-2 using the block coefficient as a reference. Reference 20 gives the values for L_{90} for the DD-692 and EC-2 as 0.056 and -0.138, respectively. (Negative means aft of midships.)

It should be emphasized that these analytical expressions are to be viewed as a convenient alternative to the DM-26 experimental curve procedure. It remains to be demonstrated that they are capable of giving reliable approximations of the ship's loading.

Appendix H

USER-SUPPLIED SUBROUTINES

The SEADYN program allows additional modeling flexibility by providing for three user-defined subroutines. They are: DRAGCO, TFNUSR, and CURUSR.

The DRAGCO subroutine provides for definition of line and lumped body drag coefficients. The routine is called each time the coefficient is required. The necessary subroutine parameter definitions are:

```
SUBROUTINE DRAGCO (INDRAG, IBOD, IDR, RE, RET, CN, CT)
```

where: INDRAG = option call number from PROB record

IBOD = 1 - spherical buoy
2 - cylindrical buoy
3 - cable

IDR = drag coefficient number specified by lumped body (BODY record) or cable (MATE record) input data

RE = Reynolds number based on the normal component of the relative fluid velocity

RET = Reynolds number based on the tangential component of the relative fluid velocity (not given for spheres)

CN = return variable for the calculated normal drag coefficient

CT = return variable for the calculated tangential drag coefficient (not used for spheres)

INDRAG, IBOD, and IDR can be used as indices to select the drag coefficients from user-defined catalogs of functions if that is desired.

The TFNUSR provides a single-valued function in time that defines the time variation of loads, currents, motions, payout, etc. The necessary subroutine parameter definitions are:

```
SUBROUTINE TFNUSR (T, F, N, TPARM)
DIMENSION TPARM (1)
```

where: T = current time, $T \geq 0$

F = the returned value of the time function

TFARM = a single-dimensioned array of user-defined input parameters provided for the N^{th} function. The maximum number of parameters is 20.

The CURUSR subroutine is called to define the fluid velocity at all node points in the structure when the FLOW library indicates a user-defined routine. Unless signaled otherwise, the routine is called at every iteration or step of the subanalysis. DYN and TSSS subanalyses can call CURUSR to get the space-dependent flow components and use the TFUN library to define time variation or can require CURUSR to give both time and space variation. (See the FLOW library data.) The necessary subroutine parameter definitions are:

```
SUBROUTINE CURUSR (T, N, NN, X, V, FLPAR)
DIMENSION X(3,1), V(3,1), FLPAR(1)
```

where: T = current time, $T \geq 0.0$

N = flow field number, $N > 0$

NN = number of nodes

X = nodal positions (X, Y, Z position for each node)

V = nodal flow vector (X, Y, Z velocity for each node). These values for all NN nodes are to be returned at each call.

FLPAR = a single-dimensioned array of user-defined input parameters provided for the N^{th} flow field. The maximum number of parameters is 10.

Appendix I

OPTIONAL DEBUG OUTPUT

The SAO data allow for requests of extra output data. In general, this output was intended for use in debugging and checking the program. It has also been found useful in finding difficult input errors and analyzing the efficiency of various choices of analysis methods and parameters. The output is not usually requested, and little effort has been expended in general [illegible] to identify the output. It is assumed that when the higher levels of output detail are requested, the user is familiar enough with the program structure and the analysis methods to be able to utilize the data intelligently. Indiscriminant requests for extra output are strongly discouraged since they greatly add to the output volume and increase run costs.

The FORTRAN variable IBG is the input used to control the extra output. Generally, low levels of output are obtained by small values for IBG. When IBG is less than 7 only a small amount of specific output is usually produced. Values of IBG greater than 6 will give significant amounts of output on iterative analyses. The following is a list of the values of IBG recognized in the subanalysis and a brief description of the output they generate (frequency codes: I = each iteration, S = each step, F = each wave frequency, H = each wave heading):

IBG	FREQUENCY	EXTRA OUTPUT
≧1	S	The number of iterations required to get convergence of iterative solutions (DEAD, LIVE, DYN).
		A message whenever the reference configuration, R_C, is updated.
	I	For VR Method: standard output record.
2	I	A message whenever an element is in compression with no compression stiffness (i.e., slack).
		A message whenever a buoy or anchor changes its status relative to the surface and bottom limits.
		The values of the net vertical forces applied to buoys and anchors.

IBG	FREQUENCY	EXTRA OUTPUT
3	S	The values of nodal displacement, velocity and acceleration in t_C (see Ref 1); the nodal displacement and acceleration at the incremental reference state; the nodal positions and forces in t_C, for each node.
4	I	Payout update data (not implemented yet).
5	I	Ship's static loads in local and global coordinates.
≧6	I	For MNR Method: five values are printed -- residual norm, displacement norm (the one used), and the values for the three alternative displacement norms.
	I	For VR Method: current numerical damping data displacement and velocity norms current control parameters.
	I	For the DIM Dynamic Method: six values are printed -- the iteration number, the displacement norm, the maximum acceleration increment, and the values for three alternative displacement norms.
	F	For the FREQ Subanalysis: the values of the steady-state solution vector, and the response amplitude operator data for displacements and tensions.
≧7		For the MNR Method: A message when the tangential stiffness matrix is calculated. A message when the 1D search is initiated and terminated.
		Element tensions.
		The values of DMU, BETA, and the modified diagonal terms of the tangential stiffness matrix when numerical damping is used.
		A message when alternating estimates are detected and the next "guess" is between them.
		The current positions of the nodes.
	I	For VR Method: current displacements and positions.
	I	For the Dynamic Analyses: A message whenever alternating estimates are detected and the next "guess" is between them.

IBG	FREQUENCY	EXTRA OUTPUT
	S	For the RFB Dynamic Method: A message when tangent stiffness matrix calculated. Element tensions Effective loads Incremental displacements
	I	For the FREQ Subanalysis: the values of the ship's roll angle.
≧8	I	For the MNR Method: current values of the residual norm, the components of the residual, the force vector, the total displacement, and the incremental displacement. IBG is reset to 7 after five steps.
	I	For the DIM Dynamic Method: current values of the force residual, acceleration increment, and displacement increment.
	I	For the FREQ Subanalysis: the values for the wave-induced ship's forces (local coordinates), the force vector (global coordinates), and the steady-state response vector.
	F	Details of dynamic tension calculation.
	S	For the RFB, VR and SLI Static Methods: the incremental or effective forces and the displacement increments.
≧9	Once	For RFB Dynamic and MNR Methods: the terms of the tangential stiffness matrix in compact storage format. IBG set to 8 after one printing.
≧10	H	For FREQ Subanalysis: the dimensionalized form of the ship's mass and restoring matrices (M_S and K_S)
	I	Diagonal terms of the system matrix, $[K-\omega^2M + i\omega C]$. The parameters used in interpolating the ship motion File. The ship's added mass and damping matrices (M_{AS}, C_S) and loading data.

IBG	FREQUENCY	EXTRA OUTPUT
11	I	The values of fluid velocity and fluid induced loads at each node in the system.
		The local and global forms of the ship's static loads (same as IBG = 5)
		The local to global transformation matrix for ships.
≧11	H	The mass matrix in compact form for the mooring system in a FREQ subanalysis.
≧12	Once	The global stiffness matrix in compact form.
≧13	Once	Each element stiffness matrix and local-to-global transformation data.
		The element contributions to the residual. IBG set to 11 after one printing.
14	Once	Same as IBG = 13 except IBG is set to zero after one printing.

Appendix J

PROGRAM SIZE LIMITATIONS AND STORAGE REQUIREMENTS

An attempt has been made to make the SEADYN program flexible in size. There are, however, some specific limitations imposed by FORTRAN dimension statements. These have been chosen large enough to accommodate most models, and can be increased by modifying the specific dimension statements and size variables. This requires program recompilation and assembly, the description of which is beyond the scope of this manual.

The specific size limitations are:

Maximum number of:	bodies in BODY record	50
	body locations given by BLOC record	50
	limit conditions in LIMI record	50
	limit locations given by LLOC record, etc.	50
	lines connecting to a limited node	10
	cable materials in MATE record	10
	entries in any tension/strain table	20
	entries in FLUI record	2*
	catenary lines of nodes generated by LINE record	20
	ship/platform rigid bodies defined by SHIP record	5
	ship/platform rigid bodies in FREQ SAO	1*
	payout/reel-in ends defines in DYN or TSSS SAO	5
	moved nodes defined in DYN or TSSS SAO	5
	lines connecting to a node where an anchor holding power CHEK is made	20
	strum strings defined by STRUM record	30
	elements in any strum string	20
	flow fields defined by FLOW record	10
	parameters associated with any flow field	10
	time functions defined by TFUN record	20
	parameters associated with any time function	20
	load variation sets defined by LOAD/LVAR records	3
	PROB + REST data sets in any run	50
	rigid format data sets in any run	1*

*Program logic limitation

wave headings on ship motion file	30
wave lengths on ship motion file	30
roll angles on ship motion file	8
wind velocities on ship load file	5
wind headings on ship load file	20
current velocities on ship load file	5
current headings on ship load file	20

The size restrictions related to the number of nodes and line elements that can be included in any model are not rigidly defined by dimension statements. A form of variable dimensions is used, which takes a given block of storage (labeled common ACOM) and partitions it according to the problem size and specific needs of each analysis option. The main program for SEADYN is simply a routine that defines the size of common and calls the controlling routine. The minimum storage required for a problem is given by the variable NBASE (printed with PROB data set output):

$$NBASE = (31 * NE) + (66 * NN) \qquad (J-1)$$

where: NE = number of elements

NN = number of nodes

It is possible to reduce the size of NBASE for problems that will not use storage for velocities and accelerations. At present, the logic to do this is not implemented, although a flag to request it is provided in the PROB data set.

Additional storage is required by each of the analysis options. The formulas used to calculate storage needed for each SAO type are:

DEAD, LIVE, TSSS, DYN:

$$NEED = NBASE + NF3*IB \qquad (J-2)$$

MODE:

$$NEED = .5*(3*NF3*NF3 + 5*NF3) \qquad (J-3)$$

FREQ:

$$NEED = NBASE + 2*NF3*IB \qquad (J-4)$$

CHEK:

$$NEED = NBASE \qquad (J-5)$$

where: IB = equation half bandwidth (0 for DYN - DIM solutions)

NF3 = 3*(NN - NSLAVE)

NSLAVE = number of slave nodes

Storage size checks are made at the beginning of each SAO to determine if enough space is available. If not, a message is printed to indicate the space needed, and the run is aborted.

The major users of storage space are simultaneous equation solvers and the Jacobi eigenvalue solver. These analyses are done entirely in core storage assuming sparse and banded matrices that are symmetric. This poses only minor difficulties on virtual memory machines. Real memory machines (even with an extended core) can place severe restrictions on problem size. Use of an extended core (LCM on CDC) requires special modifications that go beyond the scope of this manual. Get a programmer's help! Further relaxation of these restrictions are possible but require significant program modifications.

An example of the size limitations (number of elements and nodes) for a representative real memory machine is given below:

Machine: CYBER 76 (CDC7600) SCOPE 2.1 Operating System

Maximum Available Small Core Memory: $160{,}000_8$

Maximum Program Size (Segmented form - full program w/o ACOM): $126{,}000_8$

Available for ACOM: $32{,}000_8$ or 13312_{10}

Assume half bandwidth of 10 and NE = NN , then

for DEAD, LIVE, TSSS:

$$(31 + 66)\ NN + 30\ NN = 13312$$

$$NN = \frac{13312}{127} = 104$$

for DYN - DIM:

$$NN = \frac{13312}{97} = 137$$

for MODE:

$$27\ NN^2 + 15\ NN - (13312)(2) = 0$$

$$NN = \frac{-15 \pm \quad 225 + (8)(27)(13312)}{54} = 31$$

for FREQ:

$$(31 + 66)\ NN + 60\ NN = 13312$$

$$NN = \frac{13312}{157} = 84$$

SEADYN uses the following files:

FORTRAN File Name	Format	Use
01	Binary	DEAD restart save file
02	Binary	LIVE, TSSS restart save file
03	Binary	DYN restart save file
04	Binary	Alternate restart input file
05	BCD	System input file
06	BCD	System output file
08	Binary	Ship motion file
09	Binary	Scratch file for temporary data storage
10	Binary	Ship load data file
11	Binary	FREQ steady-state response solutions
12	Binary	Storage of FREQ response data (RAOs)
13	Binary	Scratch file for FREQ wave heading data
15	Binary	Deciphered record images of free-form input
16	BCD	Scratch file for rigid format input data
20	Binary	Mode shape output
21	Binary	Scratch file for mode shape calculations