HOW MOSES DEALS WITH TECHNICAL ISSUES

by

R. Ray Nachlinger

Phone (713) 975-8146 Fax (713) 975-8179

Contents

I. INTRODUCTION

The purpose of this document is to outline the the way MOSES views things from a technical (mathematical) point of view, and to provide a brief description of the numerical algorithms employed. Unfortunately, MOSES is based on several different scientific disciplines and there is no common nomenclature for the basic ideas. This complicates the task of chosing a notation scheme and of presenting the ideas. Regardless of what system is used, it will be foreign to some readers.

Another difficulty is the scope of the subject itself. To give even brief treatment to all aspects of MOSES, we must cover quite a bit of rigid body mechanics, finite element analysis, and hydrodynamics. If we attempted to delve into all of the details of each of these subjects, we would never finish. Also, the essence of the subject would be hidden beneath the details.

As a result, we have chosen a method of presentation starting from a unified point of view which is, of necessity, quite abstract. Thus, we later specialize the general theory by considering parts of the whole problem.

Normally, we will pay little attention to mathematical niceties. We will assume that functions are smooth enough to perform any operation we want.

In general, we will use bold face latin characters to denote vectors and tensors, with lower case being used for vectors and capitals for tensors. Greek characters will normally be used for scalars. An exception here is when a set of scalars are used as a vector. Normally, complex quantities will be denoted by a superposed bar. In other words, $\bar{f}$ will denote a complex quantity f and $\bar{\bar{f}}$ will denote its complex conjugate. We will denote the tensor product of $\boldsymbol{f}$ with $\boldsymbol{n}$ by $\boldsymbol{f}\otimes\boldsymbol{n}$ and the vector product of $\boldsymbol{f}$ with $\boldsymbol{n}$ by $\boldsymbol{f}\times\boldsymbol{n}$. Finally, a $\cdot$ will denote the inner vector product, a $\mathbf{1}$ will denote the identity transformation, and the boundary of a region S will be denoted by ∂S.

II. BASIC ALGORITHMS

Normally, numerical algorithms will be discussed when the need for them arises. However, some algorithms are used so often that a general discussion of them is appropriate. This section is devoted to that discussion.

II.A Solving Equations

Almost everything done in MOSES involves solving a system of linear equations, i.e. equations of the form

$$\boldsymbol{K}\boldsymbol{q} = \boldsymbol{f} \ . \tag{II.1}$$

The precise method employed depends on the circumstances. For small matrices, or for any system which does not have special properties, the basic Gausian Elimination method with row pivoting is used. For ill-conditioned systems such as constraint equations, column pivoting is also used.

When performing a stress analysis, however, simple elimination is not suitable. While there is nothing wrong with elimination itself, these systems have special properties which lend themselves to special treatment. In particular, they are sparse and symmetric. Thus, if one makes the effort, much less space is required to store the matrix $\boldsymbol{K}$ and many of the computations can be omitted. The solver used in MOSES is basically the one presented in [1] . Before effecting the solution, MOSES reorders the matrix to minimize the profile using the algorithm developed by [2] . We will now discuss two methods employed to solve a system of nonlinear equations

$$\boldsymbol{g}(\boldsymbol{q'}) = 0 \ . \tag{II.2}$$

The first method was originally invented by Newton, but with minor variations has been attributed to numerous others. This scheme can be described as follows: Suppose that one has an estimate (or starting point), $\boldsymbol{q}_0$, of the solution, and that $\boldsymbol{q'}$ is a new estimate of the solution. We can write the error, $\boldsymbol{e}$, as

$$\boldsymbol{e} = \boldsymbol{g}(\boldsymbol{q'}) \ . \tag{II.3}$$

Now, if the function $\boldsymbol{g}$ is expanded in Taylor's series about $\boldsymbol{q}_0$, we find

$$\boldsymbol{e}(\boldsymbol{q'}) = \boldsymbol{g}(\boldsymbol{q}_0) + \frac{\partial \boldsymbol{g}(\boldsymbol{q}_0)}{\partial \boldsymbol{q}}(\boldsymbol{q'} - \boldsymbol{q}_0) + \boldsymbol{t} \ . \tag{II.4}$$

Here, $\boldsymbol{t}$ is a term of second order in $(\boldsymbol{q'} - \boldsymbol{q}_0)$. Since the objective is to make $\boldsymbol{e}$ zero, we can formally write

$$\boldsymbol{q'} - \boldsymbol{q}_0 = \boldsymbol{K}^{-1}(\boldsymbol{g}(\boldsymbol{q}_0) + \boldsymbol{t}) \tag{II.5}$$

where $\boldsymbol{K}$, often called the stiffness, is given by

$$\boldsymbol{K} = -\frac{\partial \boldsymbol{g}(\boldsymbol{q}_0)}{\partial \boldsymbol{q}} \ . \tag{II.6}$$

Notice that since $\boldsymbol{t}$ also contains $\boldsymbol{q'}$ this expansion is really no better than the previous one. If, however, $\boldsymbol{t}$ is neglected, (II.5) provides a new estimate which, hopefully, will have a smaller error. This process can be repeated until some "closure" tolerance is

achieved, or until some number of iterations has been made. In other words, one uses the current estimate, $\boldsymbol{q}_0$ and solves

$$\boldsymbol{q}' = \boldsymbol{q}_0 + \boldsymbol{K}^{-1}(\boldsymbol{q}_0)\boldsymbol{g}(\boldsymbol{q}_0) \tag{II.7}$$

to obtain a new estimate. There are two important features of this scheme. First, the convergence (or divergence) will be quadratic. Second, it is only applicable to systems where $\boldsymbol{K}$ is non-singular.

Let us consider a solution of the system, $\boldsymbol{q}$, and let the region around the solution where $\boldsymbol{K}$ is non-singular be R. It can be shown that if $\boldsymbol{K}$ is positive definite at $\boldsymbol{q}$ then the algorithm will converge whenever the initial estimate is within R, and the convergence will be quadratic. Therefore, when using this algorithm, it is very important to start with a "good guess". Often, the difficulties in finding a solution arise when $\boldsymbol{K}$ is "close to singular". Here, (II.7) will predict too large a change in solution. Problems of this nature can be ameliorated by limiting the change in solution for a single step to some maximum. More difficult problems occur when $\boldsymbol{K}$ is, in fact, singular. In this case there may be no solution to the problem at all.

The other algorithm of interest is the Picard method, often called the method of successive approximations. To proceed, we define a new function, $\boldsymbol{f}$ as

$$\boldsymbol{f}(\boldsymbol{q}') = \boldsymbol{g}(\boldsymbol{q}') + \boldsymbol{q}' \ . \tag{II.8}$$

Now using this definition in (II.2) , we get

$$\boldsymbol{q}' = \boldsymbol{f}(\boldsymbol{q}') \ . \tag{II.9}$$

The algorithm is simply based on using this equation to produce a new estimate of an existing one as

$$\boldsymbol{q}' = \boldsymbol{f}(\boldsymbol{q}_0) \ . \tag{II.10}$$

The convergence here is slower than with a Newton method, and the conditions for convergence are more restrictive, but for particular classes of $\boldsymbol{g}$, it will never fail! In particular, if $\boldsymbol{K}$ has a norm less than 1, then a Picard method will converge uniformly.

II.B Time Series, Spectra, and Extremes

Often, we need to represent a function of time, and do not know all of the details of the function. In this case, we will approximate the function with a series expression which is similar to a Fourier series:

$$f(t) \sim \Re\left(\sum \bar{c}_j e_j\right) \tag{II.11}$$

where

$$e_j = e^{i(\omega_j t + \beta_j)} \ ,$$

i is $\sqrt{-1}$, $\bar{c}_j$ are complex constants, ω_j and β_j are real constants, and $\Re$ signifies that only the real part of the sum is to be considered. Without loss of generality, we will assume that all ω_j are positive and ordered so that

$$\omega_j < \omega_k \text{ when } j < k$$

We will reserve the zero subscript for cases when $\omega_o = 0$.

The constants in the series are determined by a "least squares fit". In other words, we will consider the error in the expansion (II.11)

$$r = f(t) - \sum \bar{c}_j e_j \tag{II.12}$$

and we will chose the coefficients so that the mean square error,

$$\mathcal{C} = \mathcal{M}\left(r\tilde{r}\right) \ ,$$

will be a minimum. Here, $\tilde{r}$ is the complex conjugate of r, and the mean operator is defined as

$$\mathcal{M}\left(f\right) = \lim_{T\to\infty} \frac{1}{T}\int_0^T f \, dt. \tag{II.13}$$

To minimize the error, we take the derivative of $\mathcal{C}$ with respect to each of the coefficients, $\bar{c}_k$, and set the result to zero. This yields the set of n equations

$$\mathcal{M}\left(\left(f(t) - \sum \bar{c}_j e_j\right)\tilde{e}_k\right) = 0$$

where $\tilde{e}$ is the complex conjugate of e. Now, rearranging the above slightly, we get

$$\mathcal{M}\left(f(t)\tilde{e}_k\right) = \sum \bar{c}_j \mathcal{M}\left(e_j \tilde{e}_k\right) \ .$$

As is easily established

$$\mathcal{M}\left(e_j\tilde{e}_k\right) = \lim_{T\to\infty} \frac{1}{T}\int_0^T e_j(t)\tilde{e}_k(t)\, dt = \begin{cases} 1, & \text{if } k = j; \\ 0, & \text{otherwise.} \end{cases} \tag{II.14}$$

Thus, we have an easy formula for the coefficients

$$\bar{c}_k = \mathcal{M}\left(f\tilde{e}_k\right) = \lim_{T\to\infty}\frac{1}{T}\int_0^T f(t)\tilde{e}_k(t)\,dt \ . \tag{II.15}$$

Even though we can now find the coefficients, the constants ω_j and β_j remain arbitrary. Therefore, it is best to interpret (II.15) is: if the constants have been specified, then the best mean fit is obtained by computing the coeffifients according to (II.15) .

To shed a bit more light on the constants, let us consider means of the function itself. First consider the mean which is easily computed as

$$\mathcal{M}\left(f\right) = \Re\left(\mathcal{M}\left(\sum \bar{c}_j e_j\right)\right)$$

or

$$\mathcal{M}\left(f\right) = \Re\left(\bar{c}_o\right) \ . \tag{II.16}$$

In other words, the function has a zero mean if a zero frequency is not included in the representation.

Now, consider the auto-correlation:

$$\mathcal{R}\left(f,t_o\right) = \mathcal{M}\left(f(t)f(t+t_o)\right)$$

which can be written as

$$\mathcal{R}\left(f,t_o\right) = \frac{1}{4}\sum\sum \lim_{T\to\infty}\frac{1}{T}\int_0^T \left[I_{1kj}+I_{2kj}+I_{3kj}+I_{4kj}\right]dt$$

where

$$\begin{aligned} I_{1kj} &= \bar{c}_j\bar{c}_k e_j(t)e_k(t+t_o)\ , \\ I_{2kj} &= \tilde{\bar{c}}_j\tilde{\bar{c}}_k\tilde{e}_j(t)\tilde{e}_k(t+t_o)\ , \\ I_{3kj} &= \bar{c}_j\tilde{\bar{c}}_k e_j(t)\tilde{e}_k(t+t_o)\ , \text{ and} \\ I_{4kj} &= \bar{c}_j\tilde{\bar{c}}_k e_j(t+t_o)\tilde{e}_k(t)\ . \end{aligned} \tag{II.17}$$

If we restrict ourselves to zero mean, so that we have only positive frequencies to deal with, it is easy to see that the first two terms of the integral vanish for all k and j. If we now use (II.14) on the remaining two terms, we arrive at

$$\mathcal{R}\left(f,t_o\right) = \frac{1}{4}\sum \bar{c}_k\tilde{\bar{c}}_k\left[e_k(t_o)+\tilde{e}_k(t_o)\right]$$

or

$$\mathcal{R}\left(f,t_o\right) = \frac{1}{2}\sum \bar{c}_k\tilde{\bar{c}}_k\Re\left(\tilde{e}_k(t_o)\right) \ .$$

To relate this result to something more useful, we will define the spectral density function (also called the spectrum), S, of f by

$$S(\omega_k)\delta\omega_k = \frac{1}{2}\bar{c}_k\tilde{\bar{c}}_k \ , \tag{II.18}$$

where

$$\delta\omega_k = \frac{1}{2}(\omega_{k+1} - \omega_{k-1})$$

so that we can finally write

$$\mathcal{R}\left(f, t_o\right) = \sum S(\omega_k)\delta\omega_k \Re\left(\tilde{e}_k(t_o)\right) \ .$$

Several important things can be said about this result. First, (II.18) is a formula for computing the coefficients if the spectrum of the function is specified. More importantly, however, it provides a guide for how many and which frequencies should be used in the series representation. The frequencies should be chosen so that the major features of the spectrum can be reproduced. If the spectrum is specified, we have a way of chosing both the frequencies and computing the coefficients. The phases, or β_j, still remain arbitrary. Another way of saying this is that for a given set of frequencies there are many different series which will have the same spectrum.

Another important implication of (II.18) is revealed when we consider the auto-correlation at zero. Here, we have

$$\mathcal{R}\left(f, 0\right) = \sum S(\omega_k)\delta\omega_k = \frac{1}{2}\sum \bar{c}_k\tilde{\bar{c}}_k = \mathcal{M}\left(f^2\right) \tag{II.19}$$

so that we can conclude that the area under the spectrum is equal the mean of the square, and also half the sum of the squares of the coefficients.

In the definition given above, the spectrum is defined only for the frequencies in the series. It is natural to extend this to include all frequencies and to consider the above as a consequence of the discrete nature of the series. With the extended definition, we have

$$m_o = \int_0^\infty S(\omega)\, d\omega = \mathcal{M}\left(f^2\right) \tag{II.20}$$

where m_o is called the zeroth moment of the spectrum. Likewise, we can define the nth moment of the spectrum as

$$m_n = \int_0^\infty S(\omega)\omega^n\, d\omega \ . \tag{II.21}$$

A final bit of nomenclature is that for functions with zero mean, we can define

$$\sigma^2 = \mathcal{M}\left(f^2\right)$$

where σ is called the variance of the function f.

Normally, spectra are defined in terms of a "significant" value. Do not be misled. Significant has a precise mathematical meaning, it is not really a synonym for important. The definition of significant is intertwined with the concept of the Rayleigh probability distribution

$$P(x) = \int_x^\infty p(y)\, dy$$

where

$$p(y) = 2\alpha y e^{-\alpha y^2}$$

and

$$\alpha = \frac{1}{2m_o} \ .$$

Here, $P(x)$ is the probability of an extreme being greater than x. We now ask the question: What value has a probability of 1/n of being exceeded? In other words, what is the value x^* which satisfies

$$\frac{1}{n} = \int_{x^*}^\infty p(y)\, dy \ .$$

If we evaluate the integral and solve, we find

$$x^* = \sqrt{m_o}\sqrt{2Ln(n)} \ .$$

With this information, we can evaluate:

$$A_{1/n} = \frac{\int_{x^*}^\infty y p(y)\, dy}{\int_{x^*}^\infty p(y)\, dy}$$

which yields $A_{1/n}$, the average of the $1/n$th highest extremes. Evaluating these integrals yields the final result

$$A_{1/n} = \sqrt{2m_o}\left\{\sqrt{Ln(n)} + \frac{n\sqrt{\pi}}{2}\left[1 - erf(\sqrt{Ln(n)})\right]\right\}. \qquad \text{(II.22)}$$

Evaluating the above for different values of n, we find

$$\begin{aligned} A_1 &= 1.25\sqrt{m_o} \ , \\ A_{1/3} &= 2.00\sqrt{m_o} \ , \\ A_{1/10} &= 2.54\sqrt{m_o} \ , \\ A_{1/100} &= 3.03\sqrt{m_o} \ , \text{ and} \\ A_{1/1000} &= 3.72\sqrt{m_o} \ . \end{aligned} \qquad \text{(II.23)}$$

Finally, notice that for large n, (II.22) becomes

$$A_{1/n} \simeq \sqrt{m_o}\sqrt{2Ln(n)} \ .$$

A Tale of Twos - A Question of Taste: The results above may not be what one is accustomed to seeing. They are based on the amplitude of the function, and many developments deal in height. This causes a great deal of confusion. In particular, one must be very careful when comparing spectral ordinates between two different sources. The can easily differ by a factor of 2, $\sqrt{2}$, or 4, and both be correct within their own domain. MOSES deals with this problem by always asking for a significant height when you define a spectrum. It then scales the ordinates so that the area under the curve is the proper value; i.e. Using the definition above, we have

$$A_{1/3} = 2.00\sqrt{m_o} \tag{II.24}$$

so that

$$H_{1/3} = 4.00\sqrt{m_o} \tag{II.25}$$

or finally

$${H_s}^2/16 = m_o \ . \tag{II.26}$$

All spectra in MOSES have an area of the significant height squared over 16.

We would really like to use max to predict the "maximum" response over some time. To do so requires an estimate of the number of cycles during this time. There are many way to define a period; e.g. the ratio

$$\begin{aligned} w_z &= \sqrt{\frac{\mathcal{M}\left(\dot{f}^2\right)}{\mathcal{M}\left(f^2\right)}} \text{ or} \\ w_z &= \sqrt{\frac{m_2}{m_0}} \end{aligned} \tag{II.27}$$

is, in fact, the frequency for a single component and can be viewed as an average period. This is not the only ratio which yields an average frequency:

$$\begin{aligned} T_m &= 2\pi\frac{m_0}{m_1} \text{ and} \\ T_z &= 2\pi\sqrt{\frac{m_0}{m_2}} \ . \end{aligned} \tag{II.28}$$

are both commonly used. Here, T_m is called the Mean, Average, or Observed period, and T_z is called the "Zero Up Crossing" period.

MOSES uses T_z to give a "maximum" over a specified time; i.e. using max and the above definition for T_z,

$$A_{max}(T) = \sqrt{m_o}\sqrt{2Ln\left(\frac{T}{T_z}\right)} \ . \tag{II.29}$$

II.C Least Squares Distribution

Often we are presented with a situation where we wish to accomplish a given task and yet have a set of constraints satisfied. Mathematically, this can be expressed as:

$$\boldsymbol{g} = \sum_{i=1}^{n} g_i \boldsymbol{e}_i \ . \tag{II.30}$$

Now, if n is equal the length of the $\boldsymbol{e}$ vector, the above can simply be solved for the components g_i. Often, however, n is larger and we have an under-determined system.

As an example, suppose that we are distributing the load from a load group to a set of nodes. Then (II.30) is a statement that the sum of the forces applied at each node must be equivalent to the force system produced by the load group. Alternatively, if we were repositioning a vessel, it is as a statement that the the the sum of the forces in the mooring system be equal to the environmental forces. In either of these cases there are normally more unknown force components than 6.

To resolve this ambiguity, let us define the "cost" function

$$C = \sum g_i^2 \tag{II.31}$$

and require in addition to (II.30) that $\mathcal{C}$ be a minimum. Thus, we form

$$C = \sum g_i^2 + 2\boldsymbol{\lambda}\cdot\left(\boldsymbol{g} - \sum g_i \boldsymbol{e}_i\right) \tag{II.32}$$

Taking the partial derivative of (II.32) with respect to g_k we get the original constraint (II.30) and the new conditions:

$$g_k = \boldsymbol{\lambda}\cdot\boldsymbol{e}_k \tag{II.33}$$

This particular system can be solved quite simply. First multiply (II.33) by $\boldsymbol{e}_k$ and then sum over k to get

$$\sum g_k \boldsymbol{e}_k = \left[\sum \boldsymbol{e}_k \otimes \boldsymbol{e}_k\right] \boldsymbol{\lambda} \tag{II.34}$$

The left hand side is, however, simply the right hand side of (II.30) . Thus summarizing we have

$$g_k = \boldsymbol{\lambda}\cdot\boldsymbol{e}_k \tag{II.35}$$

where $\boldsymbol{\lambda}$ is the solution of

$$\boldsymbol{g} = \left[\sum \boldsymbol{e}_k \otimes \boldsymbol{e}_k\right] \boldsymbol{\lambda} \tag{II.36}$$

It should be mentioned that this algorithm produces force that are sl not in the direction of the applied force $\boldsymbol{g}$. In fact, there is another way to view this algorithm.

Suppose that the points at which the g_i are applied are point on a rigid structure. Also, suppose that each of these points are connected to ground by a spring of stiffness k in the direction $\boldsymbol{e}_i$. The the energy in the system when a force is applied is

$$E = \frac{2}{k} \sum {\delta_i}^2 + \delta_o \boldsymbol{g} \cdot \boldsymbol{e} \tag{II.37}$$

Now, since the structure is rigid, we have have the kinematic constraint

$$\delta_i = \delta_o \boldsymbol{e}_i \cdot \boldsymbol{e} \tag{II.38}$$

Putting this constraint into the energy expression and minimizing yields

$$\boldsymbol{g} = \left[\sum \boldsymbol{e}_i \otimes \boldsymbol{e}_i \right] k \boldsymbol{\delta}_o \tag{II.39}$$

This is exactly the same equation as (II.36) except that the La Grange multiplier $\boldsymbol{\lambda}$ is replaced by the spring constant times the deflection.

III. PROCESSES

Basically, the purpose of MOSES is to simulate a physical process, predict the configurations, and then assess the integrity of the parts of the system. Before getting deeply involved, we had better define some of these things more precisely. First, the universe to MOSES is: a set of bodies, a set of connectors, an environment, and the "ground".

A body is a collection of "particles" and it has a measure, "the mass", associated with it. For the moment, it suffices to say that there is a mapping of the particles of the body into the points in space-time. We call both the values of the mapping and the mapping itself "the generalized coordinates of the body". The particles which make up the body are a primitive. As the need arises, we will impose many restrictions on the smoothness of the mapping. Although it is not necessary, it is best to think of the body as being mapped into a "nice" region of space. In other words, at any instant, the region of space occupied by the body: is connected, has a boundary continuous enough to apply the divergence theorem, and the mass is a continuous function of the space coordinates. A part of the body is simply a subset of the body with "body like" properties.

Connectors are entities which create forces between two bodies, or between a body and the ground. Normally, connectors have no mass, and no forces act on them other than the connection forces themselves.

The environment and the ground are basically used to "fill up" the remainder of space. The ground is the complement of bodies and connectors in space. The concept of environment is that it produces actions on the system due to everything other than the connectors; i.e. the environment is the wind, sea, temperature, etc.

A configuration is simply the set of generalized coordinates of the system at an instant. A process is a one parameter family of configurations. MOSES considers basically three types of processes: static, time domain, and frequency domain; i.e. MOSES condsiders three classes of the parameter: integers, time, or frequency.

III.A The Lagrange Equations

We now turn to the mathematics which govern processes; i.e. we wish to produce a system of equations which can be solved for the quantities of interest, the generalized coordinates, which will be denoted by $\boldsymbol{q}$. These coordinates suffice to determine the location and velocity of each particle in the system by a set of transformations

$$\boldsymbol{x} = \hat{\boldsymbol{x}}(x, \boldsymbol{q}) \tag{III.1}$$

and

$$\dot{\boldsymbol{x}} = \boldsymbol{V}(x, \boldsymbol{q})\dot{\boldsymbol{q}} \tag{III.2}$$

where

$$\boldsymbol{V} = \frac{\partial \hat{\boldsymbol{x}}}{\partial \boldsymbol{q}}(x, \boldsymbol{q}) \ . \tag{III.3}$$

The equations which govern the evolution of the generalized coordinates are the Lagrange equations

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{\boldsymbol{q}}} - \frac{\partial T}{\partial \boldsymbol{q}} = \boldsymbol{g} \tag{III.4}$$

where T is the kinetic energy of the system and $\boldsymbol{g}$ is the generalized force acting on the system. The kinetic energy is defined as

$$T = \frac{1}{2}\int_S \dot{\boldsymbol{x}} \cdot \boldsymbol{B}\dot{\boldsymbol{x}}\, dm \tag{III.5}$$

where S is the system under consideration, $\boldsymbol{B}$ is a symmetric matrix, and the integration is with respect to the mass.

The generalized force is defined by

$$\boldsymbol{g} \cdot \dot{\boldsymbol{q}} = \int_{\partial S} \dot{\boldsymbol{q}} \cdot \boldsymbol{f}\, ds + \int_S \dot{\boldsymbol{q}} \cdot \boldsymbol{b}\, dm \ . \tag{III.6}$$

Here, $\boldsymbol{f}$ is the force applied to the boundary of the system, and $\boldsymbol{b}$ is the force applied to the interior of the system. Notice that both the kinetic energy and the generalized force are defined by integrals over the system, and thus, can be considered to be the sum of the quantities over disjoint parts of the system. In other words, if the the union of B_i is S, and the intersection of B_i is empty, then

$$T = \sum_i T(B_i) \tag{III.7}$$

and

$$\boldsymbol{g} = \sum_i \boldsymbol{g}(B_i) \ . \tag{III.8}$$

We will now use the transformation (III.3) to obtain a more appealing representation of both the kinetic energy and the generalized force. First, notice that we can write (III.5) as

$$T = \frac{1}{2}\dot{\boldsymbol{q}}\cdot\boldsymbol{I}\dot{\boldsymbol{q}} \tag{III.9}$$

where

$$\boldsymbol{I} = \int_S \boldsymbol{V}^T\boldsymbol{B}\boldsymbol{V}\, dm\ . \tag{III.10}$$

Here, $\boldsymbol{I}$ is called the inertia tensor of the body, and it depends only on the properties of the body, the generalized coordinates, and perhaps the time. If we turn our attention to the generalized force, we obtain

$$\boldsymbol{g} = \int_{\partial B} \boldsymbol{V}^T\boldsymbol{f}\, ds + \int_B \boldsymbol{V}^T\boldsymbol{b}\, dm\ . \tag{III.11}$$

Using these results, we can express the equations of motion as

$$\frac{d}{dt}(\boldsymbol{I}\dot{\boldsymbol{q}}) = \boldsymbol{g} + \frac{\partial T}{\partial \boldsymbol{q}}\ . \tag{III.12}$$

As is well known, the system (III.12) along with a set of initial generalized coordinates and their derivatives will yield a unique solution provided $\boldsymbol{I}$ is positive definite and mild regularity restrictions are placed on $\boldsymbol{g}$ and $\frac{\partial T}{\partial \boldsymbol{q}}$. For the remainder, we will assume that these restrictions are satisfied.

III.B Generalized Degrees of Freedom

Our purpose here is to define the generalized coordinates discussed above. While the ideas are simple, the discussion is difficult because we want to be precise enough so that we can use the results, but we also want to retain flexibility to change the coordinates to suit later purposes. Toward that end, suppose that the global location of each point on the body can be expressed as:

$$\boldsymbol{x}'(x) = \boldsymbol{x}'_o + \boldsymbol{R}\boldsymbol{y}(x) \tag{III.13}$$

where $\boldsymbol{x}'$ is the global location of the particle x, $\boldsymbol{x}'_o$ is the global location of a reference point on the body, $\boldsymbol{y}$ is a vector from the reference point to particle x expressed in local body coordinates, and

$$\boldsymbol{R}^T\boldsymbol{R} = \boldsymbol{1} \ . \tag{III.14}$$

The condition (III.14) states that the transformation from body coordinates to global ones is orthogonal. Let us compute the velocity

$$\dot{\boldsymbol{x}}' = \dot{\boldsymbol{x}}'_o + \dot{\boldsymbol{R}}\boldsymbol{y} + \boldsymbol{R}\dot{\boldsymbol{y}} \ . \tag{III.15}$$

Now, differentiating (III.14) with respect to time yields

$$\boldsymbol{R}^T\dot{\boldsymbol{R}} = -\dot{\boldsymbol{R}}^T\boldsymbol{R} \tag{III.16}$$

so that $\boldsymbol{R}^T\dot{\boldsymbol{R}}$ is skew. Since a skew transformation has only three independent components, there exists a unique vector, $\boldsymbol{w}$, such that

$$\boldsymbol{R}^T\dot{\boldsymbol{R}}\boldsymbol{y} = \boldsymbol{w}\times\boldsymbol{y} \ . \tag{III.17}$$

The vector $\boldsymbol{w}$ is called the angular velocity of the body in local coordinates. The above can also be represented as

$$\boldsymbol{w}\times\boldsymbol{y} = \boldsymbol{L}(\boldsymbol{y})\boldsymbol{w} \tag{III.18}$$

where $\boldsymbol{L}$ is a skew transformation, i.e.

$$\boldsymbol{L}(\alpha\boldsymbol{y}_1 + \beta\boldsymbol{y}_2) = \alpha\boldsymbol{L}(\boldsymbol{y}_1) + \beta\boldsymbol{L}(\boldsymbol{y}_2)$$

for every α, β, $\boldsymbol{y}_1$, and $\boldsymbol{y}_2$, and

$$\boldsymbol{L}^T = -\boldsymbol{L} \ . \tag{III.19}$$

Until now, very little has been said about $\boldsymbol{y}$. Before introducing an assumption on its behavior, notice that if the body is rigid, $\boldsymbol{y}$ is a constant. Thus, any variation in $\boldsymbol{y}$ must be attributable to the deformation of the body. Precisely how $\boldsymbol{y}$ depends on

deformation is the subject of the next section. In the interim let us assume that $\boldsymbol{y}$ depends on a finite set of variables, $\boldsymbol{d}$, so that

$$\dot{\boldsymbol{y}} = \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{d}} \dot{\boldsymbol{d}}$$

or

$$\dot{\boldsymbol{y}} = \boldsymbol{P}\dot{\boldsymbol{d}} \ . \tag{III.20}$$

If we combine this with the above, we get

$$\dot{\boldsymbol{x}}' = \boldsymbol{R}\left[\dot{\boldsymbol{x}}_o + \boldsymbol{L}(\boldsymbol{y})\boldsymbol{w} + \boldsymbol{P}\dot{\boldsymbol{d}}\right] \ . \tag{III.21}$$

We will now define

$$\boldsymbol{V}(\boldsymbol{y}) = [\,\boldsymbol{1} \quad \boldsymbol{L}(\boldsymbol{y}) \quad \boldsymbol{P}\,] \tag{III.22}$$

so that the velocity can be written as

$$\dot{\boldsymbol{x}}' = \boldsymbol{R}\boldsymbol{V}\dot{\boldsymbol{q}} \tag{III.23}$$

where

$$\dot{\boldsymbol{q}} = \begin{bmatrix} \dot{\boldsymbol{x}}_o \\ \boldsymbol{w} \\ \dot{\boldsymbol{d}} \end{bmatrix} \ . \tag{III.24}$$

This is the result we have been seeking. It gives the velocity at a point of the body in terms of a set of generalized velocities, $\dot{\boldsymbol{q}}$. The first component of $\dot{\boldsymbol{q}}$ is simply the velocity of a reference point on the body expressed in local coordinates. The other two components are more difficult to interpret. As stated above, $\boldsymbol{w}$ is the angular velocity of the body in local coordinates. In a moment we will look at this in more detail. The last term in the generalized velocity is $\dot{\boldsymbol{d}}$ which is a measure of the velocity of deformation of the body.

Let us now turn our attention to $\boldsymbol{R}$. The orthogonality conditions (III.14) on $\boldsymbol{R}$ provide six independent constraints on the nine elements and thus only three quantities are required to define the rotation. A convenient choice of these three quantities is three successive rotations defined by "Euler angles". Mathematically, this can be expressed as

$$\boldsymbol{R} = \boldsymbol{R}_1\boldsymbol{R}_2\boldsymbol{R}_3 \tag{III.25}$$

where

$$\boldsymbol{R}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & +\cos\theta_1 & -\sin\theta_1 \\ 0 & +\sin\theta_1 & +\cos\theta_1 \end{bmatrix} ,$$

$$\boldsymbol{R}_2 = \begin{bmatrix} +\cos\theta_2 & 0 & +\sin\theta_2 \\ 0 & 1 & 0 \\ -\sin\theta_2 & 0 & +\cos\theta_2 \end{bmatrix} , \text{ and}$$

$$\boldsymbol{R}_3 \quad = \quad \begin{bmatrix} +\cos\theta_3 & +\sin\theta_3 & 0 \\ -\sin\theta_3 & +\cos\theta_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \tag{III.26}$$

To determine the orientation of the body defined by the Euler angles, begin with the body system and the global system aligned, and rotate the body about the local X_3 axis an amount θ_3. Continue by rotating the body about the new X_2 axis an amount θ_2, and finally, rotate about the new X_1 axis an amount θ_1 .

Consider, for a moment, the behavior of $\boldsymbol{R}$ for small values of the angles. Since

$$\boldsymbol{R}_i \simeq \mathbf{1} + \boldsymbol{L}^T(\theta_i \boldsymbol{e}_i)$$

where $\boldsymbol{e}_i$ is a unit vector in the direction i, we can write

$$\dot{\boldsymbol{R}} \simeq \sum [\boldsymbol{L}^T(\boldsymbol{e}_i \dot{\theta}_i)] \ . \tag{III.27}$$

From above however, we have

$$\dot{\boldsymbol{R}} = \boldsymbol{L}^T(\boldsymbol{w})$$

so that for small angles

$$\boldsymbol{w} \simeq \sum \boldsymbol{e}_i \dot{\theta}_i \ ; \tag{III.28}$$

i.e. the angular velocities are rates of change in the angles about the local axes. In general, the angle changes will not be small. To cope with this situation, consider two rotations: a large one defined by $\boldsymbol{U}$, and a small one defined by $\boldsymbol{S}$, so that

$$\boldsymbol{R} = \boldsymbol{S}\boldsymbol{U} \ . \tag{III.29}$$

This gives us the recipe we need to compute the new direction cosine matrix for small changes in angle: we first find the small changes in angle (integrate the angular velocities), build a direction cosine matrix for the change, and compose it with the previous direction cosine matrix to find the new one. This procedure, however, does not yield a new set of Euler angles. It simply yields a new direction cosine matrix. One can, however, find a set of Euler angles from any given $\boldsymbol{R}$, (these are not unique) so knowledge of $\boldsymbol{R}$ is all that is required.

III.C Interpolation and Approximation

If we were content to consider only rigid bodies, we would have all of the theory we need. Our goal, however, is to solve for the stresses in the bodies during both static and dynamic situations. We therefore adopt the finite element method; i.e. we will suppose that one body is actually composed of several smaller, disjoint sub-bodies called elements, and that there is a a finite set of particles, called nodes, which are special. In particular, we will assume that the elements are small enough so that the location of all points in an element can be found by interpolation from the location of the nodes in the element; i.e. if x is in an element l, then its location $\boldsymbol{y}$ in the body system is given by

$$\boldsymbol{y}(x) = \boldsymbol{y}_o(x) + \sum_k \sum_j \Phi_{lk}(x - x_k)\boldsymbol{e}_j d'_{kj}$$

or

$$\boldsymbol{y}(x) = \boldsymbol{y}_o(x) + \boldsymbol{P'}\boldsymbol{d'} \tag{III.30}$$

where k is summed over the number of nodes in the element, and j is summed from one to 3. Here, Phi_{lk} are local interpolation functions, normally polynomials. This provides a concrete representation for the assertion made in the previous section.

Quite a few things need to be said here. First, in practice, we need continuity conditions for the values of $\boldsymbol{y}$ as we cross element boundaries. Next, for some types of elements we also need to interpolate the second derivative based on the slopes at the nodes. This type of detail is quite important in developing a theory of finite elements, but is not what we are interested in here. Our objective is to integrate finite element theory within our framework, not invent the the details. For a complete discussion of precisely how one builds a theory of finite elements, see [4] . We will, however, require

$$\Phi_{lk}(x_k - x_k) = 1$$

and

$$\Phi_{lk}(x_j - x_k) = 0$$

for $j \neq k$ so that when (III.30) is used with a node, we get

$$\boldsymbol{y}(x) - \boldsymbol{y}_o(x) = \sum_j \boldsymbol{e}_j d'_{kj}$$

which substantiates our assertion previously about $\boldsymbol{d'}$ being the deformation.

One problem we have is that while (III.30) gives a rule for computing the location of points in elements, there may be points in which we are interested that do not belong to any element. Until now, we have not really talked about how a body is modeled, but MOSES has necessary concepts which need points that may not be nodes and

which are not in structural elements. For example, we need to compute the pressure on the "exterior" of the body due to wind and the sea. This exterior is modeled as a set of panels. If panels correspond to structural elements, all is well. If, however, one has a beam element model of the body, what do we do about the panels? The way MOSES deals with this is to associate the closest node with each point that is not a node. In this way, we also have the location of these "strange" points available.

If the above is used in the expression for the equations of motion, we find that we have a system of $6+n$ differential equations which govern the motion of a body. Six of these correspond to the gross motion of the body and n of them govern the deformation. If we need a large number of degrees of freedom to describe the deformation, this leads to a problem with substantial computational effort. The standard approach is to decouple the problem; i.e. one assumes that the deformation inertia is small in comparison to the gross inertia. Then one has a system of 6 differential equations which govern the gross motion. Once these are solved, the deformation problem is solved using the gross inertia as a force. An alternative is to ignore the nonlinearities of the general equations of motion and solve the n differential equations for the deformations by a modal superposition method. While each of these approaches is quite useful and yield excellent results for special cases, each has its drawbacks.

When one uses the rigid body approach, he is not only neglecting deformation inertia, but deformation itself. While not large, there are cases where the deformation can be important. Similarly for "large" structures, the deformation inertia can become important. The most serious deficiency with the modal approach is that the inertia matrix must be constant and the damping matrix must be "nice" in the sense that the equations uncouple. With problems "in air" this is not a serious restriction, but in water it raises serious questions. The free surface condition for floating bodies creates added mass and a damping matrix which are frequency dependent, and the damping matrix is not "nice". Thus, for large bodies in water we are left with a choice between two bad alternatives.

Let us consider an alternative. Suppose that the details of the deformation are not important but some aspects of it are; i.e. suppose that there is a projection of $\boldsymbol{d}'$ into a smaller subspace which does an adequate job of describing the deformation grossly, or that there are $m, m \leq n$ deformation vectors which adequately describe the deformation for gross computation of motion. These vectors are called generalized coordinates. With this approximation, (III.30) becomes

$$\boldsymbol{y}(x) = \boldsymbol{y}_o(x) + \boldsymbol{P}(x)\boldsymbol{d} \ . \tag{III.31}$$

The meaning of $\boldsymbol{P}$ and $\boldsymbol{d}$ is quite different from that of $\boldsymbol{P'}$ and $\boldsymbol{d}'$. The matrix $\boldsymbol{P'}$, in essence, defines the topology of the model. $\boldsymbol{P}$ is simply a set of three vectors, a combination of the deformation of the nodes corresponding to each generalized

coordinate. From another point of view, we are saying that an arbitrary deformation field, $\boldsymbol{d}'$ can be approximated by

$$\boldsymbol{d}' \simeq \boldsymbol{P}\boldsymbol{d} \tag{III.32}$$

where the columns of $\boldsymbol{P}$ are the vectors mentioned above.

Notice what we are proposing is to use an approximation to the deformation only to solve for the motion, not to produce detailed results inside the body. In particular, we propose (III.32) to simplify and solve the equations of motion. Then at points of interest during a simulation, we will perform a detailed stress analysis, using all degrees of freedom for the stiffness and the approximation to the velocities and accelerations in computing the force. What should result is something better than the rigid body assumption because the deformation and its derivatives are accounted for more rationally than to assert that they be zero. This approach should be equivalent to the modal superposition technique if the modes used in the superposition are used for the generalized coordinates and the problem is "nice" in the sense described above. It should be better than modal superposition for problems which are not "nice".

Now, let us be more precise. Suppose that we have a body that is composed of a linear material, that undergoes small deformations, and that we have a finite element model of it. The result is that the forces internal to the body can be given by:

$$\boldsymbol{g}'_d = -\boldsymbol{K}'\boldsymbol{d}' \ . \tag{III.33}$$

Using (III.32) , we can transform this into

$$\boldsymbol{g}_d = -\boldsymbol{K}^r\boldsymbol{d} \tag{III.34}$$

where

$$\boldsymbol{K}^r = \boldsymbol{P}^T\boldsymbol{K}'\boldsymbol{P} \ . \tag{III.35}$$

In other words, we have replaced the dependence of our problem on the deformation with a contribution to the force of (III.34) and a contribution to the stiffness of (III.35) . All of the other contributions to the equations of motion can now be computed based on the model, (III.31) , and (III.32) .

It is hard to minimize the importance of this result. First, notice that if the matrix $\boldsymbol{P}$ is the identity, the only approximations involved are those inherent in the finite element model, so (III.41) in this case are the equations of motion for a linearly elastic body undergoing small deformations that require computation of only the boundary forces and the inertia at each step. We could have employed a reduced inertia matrix so that the above would have involved computing the boundary forces, but in most problems of interest here, the inertia changes with time. As we use fewer and fewer generalized coordinates, the equations become more approximate. For a few generalized coordinates, (III.41) is nothing more than a formalization of

the standard energy estimates one was taught to make in school. The real beauty, however, is that a single formalism and set of algorithms work for approximations of all fidelity. Furthermore, by using the approximate inertia as forces applied to the complete stiffness matrix, we have a mechanism of producing excellent deformations and stresses at any point of a simulation.

In conclusion, let us say a few words about how one may choose the generalized coordinates. The most obvious way is to pick a subset of the modes of the system; i.e. solve for some of the natural frequencies and mode shapes of a body and use these as the generalized coordinates. If this procedure is followed, our scheme is equivalent to modal if the problem is linear. Our scheme, however, allows for the inertia matrix to change with time, the damping to be of any form, and nonlinear boundary conditions. As with any modal analysis scheme, you need to have enough modes to capture the input energy. Also, here, you need to have enough modes so that you can create deformation at points where boundary conditions are applied. For example, suppose that you fix a point in the structure and find a set of modes. During the simulation, you also want to apply a connection at another point which effectively fixes it. If you cannot produce a linear combination of your set of modes that have zero deformation at the second point, your answers will not be very good. In cases where the boundary conditions change during the process, a better choice of generalized coordinates would be deflections of the structure to a set of unit forces applied at a set of points - the points which yield the maximum deflection and those where connections will be applied. This is essentially a statement of the Rayleigh-Ritz scheme: pick a deformation which satisfies the boundary conditions and minimize the energy.

III.D Equations of Motion

Let us now return to the basic equations of motion and use our specific generalized coordinates to present a more specific form. First, consider the term

$$\boldsymbol{h} = \frac{d}{dt}(\boldsymbol{I}\dot{\boldsymbol{q}}) \ . \tag{III.36}$$

The vector $\boldsymbol{h}$ is a vector in the body system and its representation in the global system is given by

$$\boldsymbol{h}_g = \boldsymbol{Q}\boldsymbol{h} \tag{III.37}$$

where

$$\boldsymbol{Q} = \begin{bmatrix} \boldsymbol{R} & 0 & 0 \\ 0 & \boldsymbol{R} & 0 \\ 0 & 0 & 0 \end{bmatrix} \ .$$

Computing the time derivative of (III.37) yields

$$\dot{\boldsymbol{h}}_g = \dot{\boldsymbol{Q}}\boldsymbol{h} + \boldsymbol{Q}\dot{\boldsymbol{h}} \tag{III.38}$$

or

$$\boldsymbol{Q}^T\dot{\boldsymbol{h}}_g = \boldsymbol{Q}^T\dot{\boldsymbol{Q}}\boldsymbol{h} + \boldsymbol{I}\ddot{\boldsymbol{q}} + \dot{\boldsymbol{I}}\dot{\boldsymbol{q}} \ .$$

We learned above, however, that

$$\boldsymbol{R}^T\dot{\boldsymbol{R}}\boldsymbol{y} = \boldsymbol{L}(\boldsymbol{y})\boldsymbol{w} = \boldsymbol{L}^T(\boldsymbol{w})\boldsymbol{y} \ ,$$

so the second term in (III.38) can be written as

$$\dot{\boldsymbol{Q}}\boldsymbol{h} = \boldsymbol{N}(\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$$

where

$$\boldsymbol{N}(\dot{\boldsymbol{q}}) = \begin{bmatrix} \boldsymbol{L}^T(\boldsymbol{w}) & 0 & 0 \\ 0 & \boldsymbol{L}^T(\boldsymbol{w}) & 0 \\ 0 & 0 & 0 \end{bmatrix} \boldsymbol{I} \ .$$

If we now combine all of these results, (III.38) can be written as

$$\boldsymbol{I}\ddot{\boldsymbol{q}} = \boldsymbol{g} - \left(\dot{\boldsymbol{I}} + \boldsymbol{N}(\dot{\boldsymbol{q}})\right)\dot{\boldsymbol{q}} - \boldsymbol{K}^r\boldsymbol{q} \tag{III.39}$$

where we have used the approximation for the deformation forces and the generalized forces here, $\boldsymbol{g}$, are only the boundary forces. Also, the reduced stiffness, $\boldsymbol{K}^r$, differs from that defined above by zeros in the first six degrees of freedom.

Before turning to specific processes, we will present the form of the equations of motion which will be commonly used. Suppose we have a known set of generalized coordinates, $\boldsymbol{q}_0$. If we first define

$$\boldsymbol{C}(\boldsymbol{q}_0, \dot{\boldsymbol{q}}_0) \quad = \quad -\frac{\partial \boldsymbol{g}}{\partial \dot{\boldsymbol{q}}} \ ,$$

$$\begin{aligned} \boldsymbol{K}(\boldsymbol{q}_0, \dot{\boldsymbol{q}}_0) &= -\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{q}} - \boldsymbol{K}^r \text{ , and} \\ \boldsymbol{s}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{q}_0, \dot{\boldsymbol{q}}_0) &= \boldsymbol{g}(\boldsymbol{q}) - [\boldsymbol{C} + \boldsymbol{N} + \dot{\boldsymbol{I}}]\dot{\boldsymbol{q}} - \boldsymbol{K}\boldsymbol{q} \end{aligned} \tag{III.40}$$

and use these definitions in the above, we obtain the final representation

$$\boldsymbol{I}\ddot{\boldsymbol{q}} + \boldsymbol{C}\dot{\boldsymbol{q}} + \boldsymbol{K}\boldsymbol{q} = \boldsymbol{s} \ . \tag{III.41}$$

The matrices $\boldsymbol{C}$ and $\boldsymbol{K}$ are called the tangent damping and stiffness matrices at the state $(\boldsymbol{q}_0, \dot{\boldsymbol{q}}_0)$ respectively. The vector $\boldsymbol{s}$ will also be called the generalized force. It, however, differs from $\boldsymbol{g}$ in that it depends on the generalized coordinates and their time derivative by terms of order greater than one in the difference of the coordinates $\boldsymbol{q} - \boldsymbol{q}_0$ and velocities $\dot{\boldsymbol{q}} - \dot{\boldsymbol{q}}_0$.

Finally, notice that we can rewrite (III.41) as

$$\boldsymbol{K}\boldsymbol{q} = \boldsymbol{s} - \boldsymbol{I}\ddot{\boldsymbol{q}} - \boldsymbol{C}\dot{\boldsymbol{q}} \tag{III.42}$$

so that the equations of motion could be viewed as a static system at some time, provided that the velocities and accelerations were specified at that time. This seems to be a useless result, but as we will see later, it is more useful than appears.

III.E Static Processes

A static process is one in which the time behavior of the generalized forces is sufficiently slow that all time derivatives of body properties can be neglected. In other words, $\dot{\boldsymbol{I}}$, $\dot{\boldsymbol{q}}$, and $\ddot{\boldsymbol{q}}$ are all small enough to be neglected. If we evict these terms from the basic equations of motion, we get

$$\boldsymbol{s}(\boldsymbol{q},t)=0\ . \tag{III.43}$$

This is a deceptively simple equation; the exercise of finding an equilibrium configuration is one of the most difficult things that MOSES does. One of the complicating factors is that, since (III.43) may be nonlinear, there may be more than one equilibrium configuration. When one finds an equilibrium configuration, care should be used, it may not be the one that occurs in real life.

To solve for the configurations which satisfy (III.43) it would be tempting to use the Newton method discussed above, but we have some problems. The main problem is that in many cases the stiffness matrix is singular. For example, consider a freely floating ship. Here, one has stiffness in heave, roll and pitch, but none in surge, sway and yaw. Blindly applying a Newton method here is not effective. Let us consider the Newton method for a moment; i.e. suppose that we have a configuration defined by $\boldsymbol{q}_0$. To find a new configuration which is closer to the solution, we compute

$$\boldsymbol{q'}=\boldsymbol{q}_0+\boldsymbol{K}^{-1}(\boldsymbol{q}_0)\boldsymbol{g}(\boldsymbol{q}_0)\ .$$

To avoid a possible singularity in $\boldsymbol{K}$, suppose we replace it with

$$\boldsymbol{K'}=-\omega^2\boldsymbol{I}+\boldsymbol{K}$$

where ω is a small parameter which we can specify. This should fix the singularity problems because for degrees of freedom where $\boldsymbol{K}$ is not singular, the inertia term will be negligible and for singular degrees of freedom of $\boldsymbol{K}$ it adds a term on the diagonal. Even with this additional term, the method may yield a change in configuration which is "too large"; i.e. the new position may actually be further from the desired one than the old one was. To prevent this, there is an additional parameter, m, which limits step size. Suppose that δ is the change in a coordinate (element of $\boldsymbol{q}$ vector). Then what will actually be added to the old coordinate x_o to get the new one x_n is given by

$$x_n=x_o+\begin{cases}\delta & \text{if } |\delta|\le m\\ \frac{m\delta}{|\delta|} & \text{otherwise.}\end{cases}$$

When using this modified Newton method, there are two reasons that one may not find a configuration with tolerance: the step size MOSES takes may be too small or

it may be too large. When one is far from equilibrium, large steps are needed if one is to get close within the maximum number of iterations. Here, the default value of m may need to be increased so that larger steps can be taken. This may not help if ω is what is limiting the step size. You see, the term we added to minimize the chance of a singular stiffness also reduces the step size. For systems with large inertia and small stiffness, the "small extra" can actually dwarf the stiffness. The fix here is to decrease ω. *Caution* is, however, in order. The defaults are set for reliability. Following the above advice can create the other problem - too large a step.

For very stiff systems or systems with tensions only element, equilibrium may fail because steps which are too large are being taken. Here, MOSES thinks it needs to move "a good bit", but once it gets there it finds it has gone too far. Here, the way to help is to make m smaller.

The advice given here may appear contradictory, but there are two distinct cases. If you have problems, you first need to find out the cause. The best way to eliminate difficulties is to begin with a good guess. This eliminates several problems: you no longer need "big steps" to find equilibrium, and you are much more certain that the configuration you have found is, in fact, the one you want.

III.F Time Domain Processes

Time domain processes are simply a set of configurations which satisfy the basic equations of motion (III.41) . They are quite important since they are the only means of properly accounting for all aspects of a problem. Unfortunately, they are also computationally expensive. The equations of motion are differential equations (actually as we will see later they are integro-differential equations). We effect a solution by converting them into a sequence of algebraic equations which can be solved by a Newton method. There are many techniques which can be used in this process, each of which can claim superiority in certain situations. The one which seems to work well in most circumstances, and is described below, is the Newmark method. For a complete discussion of this technique, see [3] . Remember that the equations of motion can be written as

$$\boldsymbol{I}\ddot{\boldsymbol{q}}(t) + \boldsymbol{C}\dot{\boldsymbol{q}}(t) + \boldsymbol{K}\boldsymbol{q}(t) = \boldsymbol{s} \ . \tag{III.44}$$

Now, suppose that we know the solution at time t_1 and want to estimate the solution at t_2. We can formally write

$$\begin{aligned} \ddot{\boldsymbol{q}}(t_2) &= \ddot{\boldsymbol{q}}(t_1) + \boldsymbol{a} \ , \\ \dot{\boldsymbol{q}}(t_2) &= \dot{\boldsymbol{q}}(t_1) + \delta\ddot{\boldsymbol{q}}(t_1) + \gamma\delta\boldsymbol{a} \ , \text{ and} \\ \boldsymbol{q}(t_2) &= \boldsymbol{q}(t_1) + \delta\dot{\boldsymbol{q}}(t_1) + \frac{1}{2}\delta^2\ddot{\boldsymbol{q}}(t_1) + \beta\delta^2\boldsymbol{a} \end{aligned} \tag{III.45}$$

where $\delta = t_2 - t_1$ and $\boldsymbol{a}$, γ, and β are constants. Eliminating $\boldsymbol{a}$ from (III.45) gives

$$\begin{aligned} \ddot{\boldsymbol{q}}(t_2) &= a\ddot{\boldsymbol{q}}(t_1) + b\dot{\boldsymbol{q}}(t_1) + c\,[\boldsymbol{q}(t_2) - \boldsymbol{q}(t_1)] \\ \dot{\boldsymbol{q}}(t_2) &= d\ddot{\boldsymbol{q}}(t_1) + e\dot{\boldsymbol{q}}(t_1) + f\,[\boldsymbol{q}(t_2) - \boldsymbol{q}(t_1)] \end{aligned} \tag{III.46}$$

where

$$\begin{aligned} a &= 1 - \frac{1}{2\beta} \ , \\ b &= -\left(\frac{1}{\beta\delta}\right) \ , \\ c &= \frac{1}{\beta\delta^2} \ , \\ d &= \delta\left(1 - \frac{\gamma}{2\beta}\right) \ , \\ e &= \left(1 - \frac{\gamma}{\beta}\right) \text{ and} \\ f &= \frac{\gamma}{\beta\delta} \ . \end{aligned} \tag{III.47}$$

Using (III.47) in (III.44) allows us to get an equation which can be solved for the location at t_2.

$$\boldsymbol{S}[\boldsymbol{q}(t_2) - \boldsymbol{q}(t_1)] = \bar{\boldsymbol{s}} \tag{III.48}$$

where

$$\begin{aligned} \boldsymbol{S} &= c\boldsymbol{I} + f\boldsymbol{C} + \boldsymbol{K} \text{ and} \\ \bar{\boldsymbol{s}} &= \boldsymbol{s} - [a\boldsymbol{I} + d\boldsymbol{C}]\,\ddot{\boldsymbol{q}}(t_1) - [b\boldsymbol{I} + e\boldsymbol{C}]\,\dot{\boldsymbol{q}}(t_1)\ . \end{aligned} \tag{III.49}$$

Once $\boldsymbol{q}(t_2)$ is obtained, (III.47) can be used to find $\ddot{\boldsymbol{q}}(t_2)$ and $\dot{\boldsymbol{q}}(t_2)$.

One should notice that (III.47) is nonlinear in that all of the quantities depend on $\boldsymbol{q}(t_2)$. In practice, MOSES iterates this solution evaluating the functions at the last iterant until the change in location is less than a specified tolerance. Also, a few words should be said about the Newmark parameters γ and β. The defaults are .25 and .5. There is almost no numerical damping with these values. In fact, for some problems, the scheme results in small *negative* damping. If these values are changed from the default to .33 and .66, then a small bit of numerical damping is induced. Normally one does not need to worry about this, but for problems such as decay problems in calm seas, the defaults do not work very well. The following figure illustrates this effect for the heave decay of a semisubmersible.

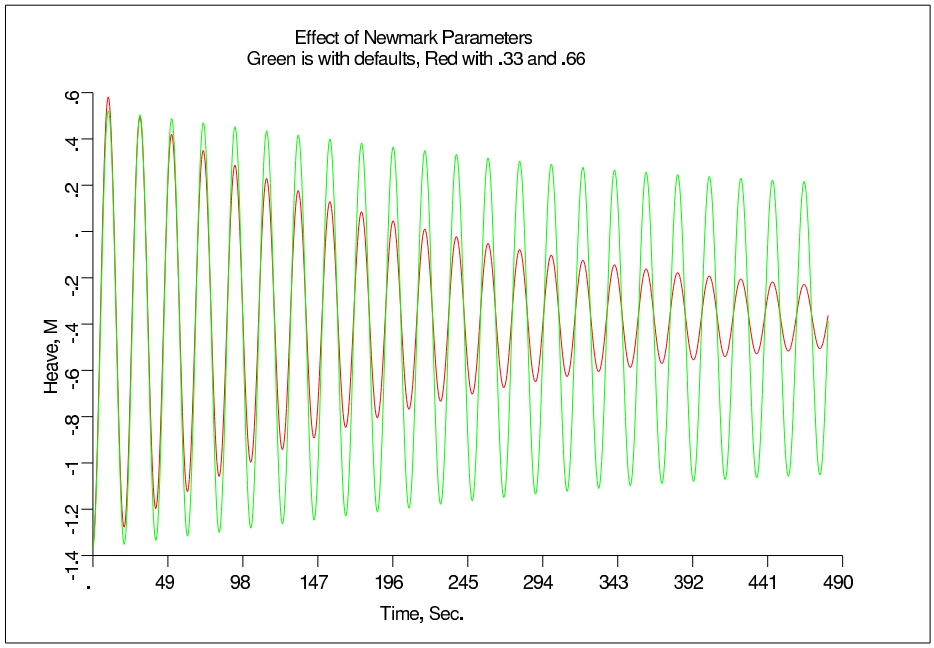

Figure 1: Effect Of Newmark Parameters

III.G Frequency Domain Processes

As mentioned above, a time domain solution offers extreme insight into the behavior of the system for an extreme cost. In many cases an alternative, a frequency domain process, is attractive. Before discussing the relative merits of the two approaches, we must first define precisely what we mean by a frequency domain process. In essence, we will seek an approximation of the equations of motion which is the sum of a static part plus a small deviation. To this end, we approximate the generalized coordinates with a time series as discussed above:

$$\boldsymbol{q} \simeq \Re\left(\sum e_j \bar{\boldsymbol{q}}_j\right) \tag{III.50}$$

where as before,

$$e_j = e^{i(\omega_j t + \beta_j)}$$

and i is $\sqrt{-1}$. Notice that (III.50) can also be written as

$$\boldsymbol{q} \simeq \frac{1}{2}\sum[\tilde{e}_j \bar{\boldsymbol{q}}_j + e_j \tilde{\bar{\boldsymbol{q}}}_j] \tag{III.51}$$

where $\tilde{a}$ denotes the complex conjugate of a. With this approximation, the equations of motions will not necessarily be satisfied, and we can write them as:

$$\boldsymbol{r} = \boldsymbol{I}\ddot{\boldsymbol{q}} + \boldsymbol{C}\dot{\boldsymbol{q}} + \boldsymbol{K}\boldsymbol{q} - \boldsymbol{s} \tag{III.52}$$

where $\boldsymbol{r}$ is the residual resulting from the approximation and the inertia, damping, and stiffness matrices are evaluated at the mean position. We do not know the mean position *a priori*, instead, it will result as the solution for the zeroth frequency. We now use a Galerkin method to find the set of $\bar{\boldsymbol{q}}_j$ which gives zero mean residual. Thus, let us define the operator, F, by

$$F_k(\boldsymbol{x}) = \lim_{T\to\infty} \frac{2}{T}\int_0^T e_k(t)\boldsymbol{x}(t)\, dt \ . \tag{III.53}$$

Before proceeding, let us deduce some of the properties of Fi_k. First, by simple integration, we find that

$$F_k(\tilde{e}_j) = \lim_{T\to\infty} \frac{2}{T}\int_0^T e_k(t)\tilde{e}_j(t)\, dt \begin{cases} 2 & \text{if } k = l \\ 0 & \text{otherwise} \end{cases}$$

and

$$F_k(e_j) = \lim_{T\to\infty} \frac{2}{T}\int_0^T e_k(t)e_j(t)\, dt = 0 \ .$$

Now, let us compute

$$F_k(\boldsymbol{q}) = \frac{1}{2}\sum_j [F_k(\tilde{e}_j)\bar{\boldsymbol{q}}_j + F_k(e_j)\tilde{\bar{\boldsymbol{q}}}_j]$$

or, using the above,

$$F_k(\boldsymbol{q}) = \bar{\boldsymbol{q}}_k \ .$$

The following are also useful:

$$\begin{aligned} F_k(\dot{\boldsymbol{q}}) &= i\omega_k \bar{\boldsymbol{q}}_k \ , \\ F_k(\ddot{\boldsymbol{q}}) &= -\omega_k^2 \bar{\boldsymbol{q}}_k \ , \\ F_k(\Re\,(a e_k)) &= a \ , \\ F_k(\Re\,(a e_l)) &= 0 \ , \text{ and} \\ F_k\left(\int_{-\infty}^{t} \boldsymbol{D}(t-\tau)\dot{\boldsymbol{q}}(\tau)\,d\tau\right) &= F_k(\boldsymbol{D}) F_k(\dot{\boldsymbol{q}}) \end{aligned} \tag{III.54}$$

where a is a constant. Most of these results are simple consequences of those above. The last one, which is a bit more involved, is a simple extension of the convolution theorm from Laplace transforms. Our assumption on the form of $\boldsymbol{q}$ is sufficient for all of these results except the last. Here, we need also that $\boldsymbol{D}$ be bounded and $\boldsymbol{D}(\xi) = 0$ for $\xi < 0$.

If we use this operator on (III.52) , use the properties discussed above, and set the residual to zero, we arrive at a set of $n+1$ equations:

$$F(e_k \boldsymbol{r}) = \sum_j \bar{\boldsymbol{S}}_j(\boldsymbol{q}_0)\bar{\boldsymbol{q}}_j - \bar{\boldsymbol{s}}_k = 0 \tag{III.55}$$

where

$$\begin{aligned} \bar{\boldsymbol{S}}_j(\boldsymbol{q}_0) &= -\omega_j^2 \boldsymbol{I}(\boldsymbol{q}_0) + i\omega_j \boldsymbol{C}(\boldsymbol{q}_0) + \boldsymbol{K}(\boldsymbol{q}_0) \text{ and} \\ \bar{\boldsymbol{s}}_k &= F_k(\boldsymbol{s}) = \lim_{T\to\infty} \frac{2}{T}\int_0^T e_k(t)\boldsymbol{s}(\cdot)\,dt \ . \end{aligned} \tag{III.56}$$

While these equations are valid for all k, it is instructive to notice that for $k = 0$ it is a bit different in form. Here,

$$\boldsymbol{K}(\boldsymbol{q}_0)\boldsymbol{q}_0 = \bar{\boldsymbol{s}}_0$$

which needs to be solved for the mean position before the frequency components can be computed. We will find out later that the mean may depend on the values of $\bar{\boldsymbol{q}}_k$ so that the solution for the mean may be coupled with that for each frequency.

At this point, we have reduced the original system of m differential equation to a system of $n+1$ sets of m algebraic equations. To appreciate the significance of this, let us examine a special case. In particular, let

$$\boldsymbol{s} = \sum_l a_l \Re\,(e_l \boldsymbol{u}_l) - \boldsymbol{A}\ddot{\boldsymbol{q}} - \int_{-\infty}^{t} \boldsymbol{D}(t-\tau)\dot{\boldsymbol{q}}(\tau)\,d\tau$$

which, applying the definition and the properties of F_k, yields

$$\boldsymbol{s}_k = a_k \boldsymbol{u}_k + \omega^2 \boldsymbol{A}\bar{\boldsymbol{q}}_k - i\omega u_l \bar{\boldsymbol{D}}_k \bar{\boldsymbol{q}}_k$$

where

$$\bar{\boldsymbol{D}}_k = F_k(\boldsymbol{D}) \ .$$

If we combine this with our basic equation above, we find that for this type of generalized force, the solution to our problem is given by

$$[-\omega_k^2(\boldsymbol{I}+\boldsymbol{A}) + i\omega_k\bar{\boldsymbol{D}}_k + \boldsymbol{K}]\bar{\boldsymbol{q}}_k = a_k\boldsymbol{u}_k$$

$$\bar{\boldsymbol{S}}^*\bar{\boldsymbol{q}}_k = \boldsymbol{a}_k\boldsymbol{u}_k$$

where

$$\bar{\boldsymbol{S}}^* = [-\omega_k^2(\boldsymbol{I}+\boldsymbol{A}) + i\omega_k\bar{\boldsymbol{D}}_k + \boldsymbol{K}] \ .$$

This is a nice result. For this type of generalized force, we have reduced a system of m differential equations to a system of n algebraic equations of order n; i.e. for this case, the equations uncouple and they are linear. It should be noted that this is the identical result one normally sees for frequency domain motions of a floating body, but it looks different. What is normally called the added mass matrix is not only our $\boldsymbol{A}$, but this plus a portion of the imaginary part of $\bar{\boldsymbol{D}}_k$. The remainder of $\bar{\boldsymbol{D}}_k$ is normally called the radiation damping matrix.

In view of all of the estimates we made in arriving at (III.50) , it would be prudent to compare results from this estimate with those obtained elsewhere. Most such comparisions show excellent agreement *except* near reasonant peaks. In many cases, radiation damping is simply not enough to adequately predict the response at reasonance. As an aside, reasonance is not really a well defined point here. Since the matrix depends on frequency, what matrix do we use in finding the eigenvalues? Here, the term reasonance will be used to describe local peaks in the frequency response.

To augment the radiation damping, we will attempt to add some damping which can be attributed to viscous effects. In particular, Morrison's equation contains a term which, in our notation, can be written as

$$\boldsymbol{s} = \sum_l c_l' \sum_m \boldsymbol{V}_l^T\boldsymbol{u}' \, |\boldsymbol{w}_{lm}| \, [\boldsymbol{w}_{lm}]$$

where

$$\boldsymbol{w}_{lm} = \Re\left(a_m\boldsymbol{v}_{ml}e_m\right) - \boldsymbol{V}_l\dot{\boldsymbol{q}}$$

and the sum over l is over the Morrison elements, the sum over m is over the frequency components in the wave, $\boldsymbol{v}_{ml}$ is the wave particle velocity at the point l due to wave component m, and c_l' is a coefficient which depends on the relative speed between the wave end the point on the body.

If we attempt to compute the frequency components of this force as we did above, we will quickly encounter difficulties. The most troublesome thing here is the absolute value. We have two ways of attacking this difficulty. The first one is to consider the

above equation one frequency component at a time and assume that the coefficient is a linear function of the relative velocity! This may seem unreasonable since this is a nonlinear phenomena, but people have been doing it for years. Before we compute the transform of the above, let us use our basic expression for the velocity in the expression for $\boldsymbol{w}_{lm}$ to obtain

$$\boldsymbol{w}_{lm} = \Re\left(\left(a_m \boldsymbol{v}_{ml} - i\omega_m \boldsymbol{V}_l \bar{\boldsymbol{q}}_m\right) e_m\right)$$

or

$$\boldsymbol{w}_{lm} = \sum_m \boldsymbol{a}_{lm} \cos(\omega_m t + \phi_m) \ .$$

Now, dropping the sum over frequency components we can write

$$\boldsymbol{s}_k = \sum_l c_l' \boldsymbol{V}_l^T \boldsymbol{u}' \lim_{T\to\infty} \frac{2}{T} \int_0^T e_k(t) |w_{lk}| \left[w_{lk}\right] dt$$

or

$$\boldsymbol{s}_k = \sum_l \boldsymbol{V}_l^T \boldsymbol{u}' c_l' \left|\boldsymbol{w}_{lk}\right| \boldsymbol{w}_{lk} \lim_{T\to\infty} \frac{2}{T} \int_0^T e_k(t) \cos(\omega_k t + \phi_k) |\cos(\omega_k t + \phi_k)| \, dt \ .$$

The integral can now be evaluated quite simply since it is a periodic function. One simply picks regions where the cosine is positive and then integrates. Doing this, we find that the value of the integral is $8/3\pi$. Using this result, we obtain the "final" representation for the generalized force due to Morrison's type drag

$$\boldsymbol{s}_k = \sum_l c_l^* \left(a_k \boldsymbol{D'}_l \boldsymbol{v}_{kl}\right) - i\omega_k \bar{\boldsymbol{D}}'(c_l^*) \bar{\boldsymbol{q}}_k \tag{III.57}$$

where

$$\begin{aligned} \boldsymbol{D'}_l &= \boldsymbol{V}_l^T \boldsymbol{u}' \ , \\ \bar{\boldsymbol{D}}'(c_l) &= \sum_l c_l^* \boldsymbol{D'}_l \boldsymbol{V}_l \ , \text{ and} \\ c_l^* &= \frac{8}{3\pi} c_l' \left|a_k \boldsymbol{v}_{kl} - i\omega_k \boldsymbol{V}_l \bar{\boldsymbol{q}}_k\right| \ . \end{aligned} \tag{III.58}$$

For those of you who prefer words to mathematics, this result is called "Equivalent Linearization" and is well known. The approach adopted here is quite different. Normally this is done for only a single degree of freedom and is applied "after the fact"; i.e. the game is to find an equivalent damping coefficient which will dissipate the same energy in a half cycle as does the viscous one. The current approach clearly shows the assumptions involved with obtaining the result and makes one question its validity.

At this point, it is reasonable to look at an alternative to the above exercise. Let us again look at the generalized force due to drag,

$$\boldsymbol{s} = \sum_l \sum_m c_l' \boldsymbol{V}_l^T \boldsymbol{u}' \left|\boldsymbol{w}_{lm}\right| \left[\boldsymbol{w}_{lm}\right]$$

and rewrite it as

$$\boldsymbol{s} = \sum_l \bar{c}_l(\bar{r}) \boldsymbol{D'}_l \sum_m \boldsymbol{w}_{lm}$$

where

$$\bar{c}_l = c_l \left(\alpha \sqrt{\sum_m |\boldsymbol{w}_{lm}|^2} \right)$$

and α is some constant. Computing the transform of this expression is easy, it is linear. Performing the computation we find

$$\boldsymbol{s}_k = \sum_l \bar{c}_l \left(a_k \boldsymbol{D'}_l \boldsymbol{v}_{kl} \right) - i\omega_k \bar{\boldsymbol{D}}'(\bar{c}_l) \bar{\boldsymbol{q}}_k \ . \tag{III.59}$$

This is the same form as with the equivalent linearization, but conceptually quite different. The equivalent linearization uses the results at a single frequency to obtain a damping contribution *at that frequency*. The other scheme (which we will call spectral linearization) uses a fraction of the root mean square velocity due to all frequency components at a point to compute a coefficient which is used for all components. The obvious question is what does one use for α. Here, we can use the statistical linearization as a bit of a guide. If we have only a single frequency component and one degree of freedom, then for the two results to be the same $\alpha = 8/3\pi = .8488$. Alternatively, using the RMS is a reasonable approximation in its own right so $\alpha = 1$ seems reasonable. Finally, if one views the problem from a stochastic perspective, $\alpha = \sqrt{2/\pi} = .797$ is found. In summary, α is left to your choice, but values from .8 to 1. seem like a good place to start.

Now, let us look at solving the system of equations when drag is involved. The system to solve becomes

$$\left[\bar{\boldsymbol{S}}^* + i\omega_k \bar{\boldsymbol{D}}'(c_l) \right] \bar{\boldsymbol{q}}_k = \bar{\boldsymbol{s}}_k^*$$

where

$$\bar{\boldsymbol{s}}_k^* = a_k \boldsymbol{u}_k + \sum_l \bar{c}_l \left(a_k \boldsymbol{D'}_l \boldsymbol{v}_{kl} \right) \ .$$

Since $\bar{\boldsymbol{D}}'$ depends on the motion, this is a nonlinear system of equations. To solve this system, a modified Picard method is used. Suppose we have at least two previous iterations. Then the system which will be solved is

$$\left[\bar{\boldsymbol{S}}^* + i\omega_k \bar{\boldsymbol{D}}'_n \right] \bar{\boldsymbol{q}}_k = \bar{\boldsymbol{s}}_k^* \tag{III.60}$$

where

$$\bar{\boldsymbol{D}}'_n = \frac{1}{2} \left(\bar{\boldsymbol{D}}'_{n-2} + \bar{\boldsymbol{D}}'_{n-1} \right) \ ;$$

i.e. we use an average of the damping matrices from the two *previous* iterations. zero is used for the damping matrix, until a value has been computed. We have tried

many schemes over the years and this seems to work better than any other. By using the average, it stops the wild fluctuations which one can get close to reasonance.

The $\bar{\boldsymbol{q}}$ values which result from (III.50) are not true response amplitude operators since the damping coefficient depends on the response. Notice, however, that the right hand side is linear in a_k, so that after we finish finding a solution, we can divide $\bar{\boldsymbol{q}}_k$ by $\boldsymbol{a}_k$ to obtain something which "looks like" response operators. Remember, however, that some assumption for the sea was used to compute these things. If you look at the results due to a sea which is drastically different than that used to compute the response, the results may be open to question. As an example, the following figure shows the roll "response operator" computed for several different values of steepness (wave height used in the linearization). The difference is dramatic.

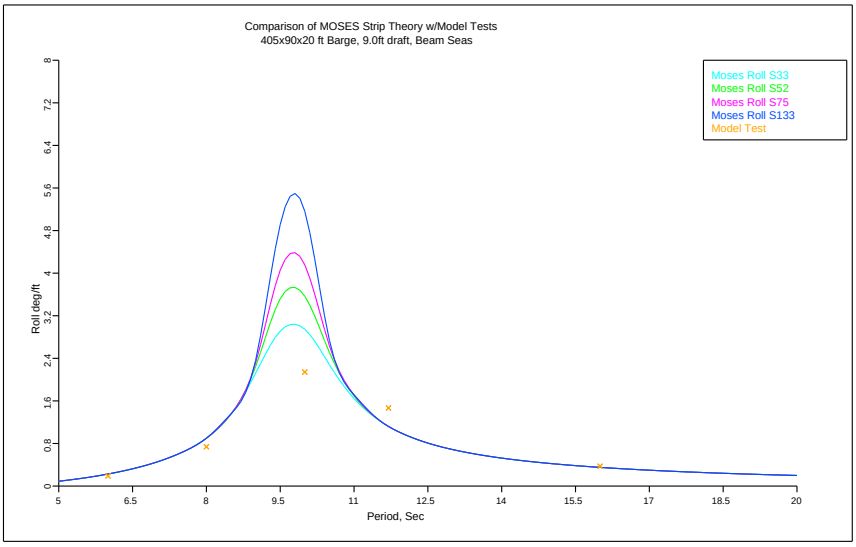

Figure 2: Effect of Nonlinear Damping

One final task remains here - to include some nonlinear forcing. Toward this end, let us look at an addition to the generalized force

$$\boldsymbol{s} = \sum_l \sum_m \Re\left(\boldsymbol{t}^s_{ml} e_{l+m} + \boldsymbol{t}^d_{ml} e_{l-m}\right) \ .$$

Computing the frequency force, we find

$$\boldsymbol{s}_k = \sum_l \sum_m \left(\boldsymbol{t}^s_{ml} h_{k,l+m} + \boldsymbol{t}^d_{ml} h_{k,l-m}\right)$$

where

$$h_{k,l-m} = \begin{cases} 1 & \text{if } \omega_k = \omega_l - \omega_m i \\ 0 & \text{otherwise} \end{cases}$$

and

$$h_{k,l+m} = \begin{cases} 1 & \text{if } \omega_k = \omega_l + \omega_m \\ 0 & \text{otherwise} \end{cases} .$$

This is simply something to add on the right hand side of the equation and does not influence the solution algorithm at all. What is important here is the we get force contributions at different frequencies. In particular, we get contributions to $\boldsymbol{q}_0$ from these forces and we get forces outside the range of periods where the wave forces were generated. Although we did not explicitly consider it, another form of nonlinearity includes combinations of the wave excitation and the body velocity. MOSES does include these terms, but they are, once we have a response, simply more right hand side contributions.

Another thing of interest here is that we can no longer get anything which masquerades as a "response amplitude operator". The response is a nonlinear function of the input Fourier coefficients. Finally, although we have not considered it here, there are many other sources of nonlinearities. The easiest to see is the term in the equations of motion $[\boldsymbol{N} + \dot{\boldsymbol{I}}]\dot{\boldsymbol{q}}$. We have ignored it here because most people do not include it.

IV. CONNECTORS

In actuality, a mooring line is itself a dynamic system which can be treated correctly with "rod elements". As an option to using the computer resources required to solve a system of rod elements, MOSES also allows one to consider mooring lines approximately, i.e. we consider the global force due to a mooring line as being given by

$$\boldsymbol{f} = h(\xi, z)\boldsymbol{e} + v(\xi, z)\boldsymbol{e}^* \tag{IV.1}$$

where ξ is the horizontal distance from the vessel attachment point to the anchor, $\boldsymbol{e}$ is the image in the waterplane of a unit vector from the attachment to the anchor, $\boldsymbol{e}^*$ is a unit vector perpendicular to the waterplane point down, and h and v are the horizontal and vertical components of the force in the line at the attachment. Thus, we are approximating the behavior of a mooring line as that of a series of catenaries - ignoring any dynamic behavior and ignoring any force applied to the line except its weight and buoyancy.

Let us begin by considering the basic equations which follow this assumption and are applicable to a single segment of line, i.e. we have a piece of line which has constant properties along its length and has no additional weights or buoyancies along its length. Thus, we have:

$$\begin{aligned}
\gamma &= \frac{1}{AE} , \\
\ell &= \ell_1 + \ell_2 , \\
{\ell_1}^* &= \ell_1[1 + h\gamma\cos(\alpha)] , \\
{\ell_2}^* &= \ell_2\left[1 + \frac{h\gamma}{\ell_2}\int_o^{\ell_2} g(z')\right] ds , \\
\ell^* &= {\ell_1}^* + {\ell_2}^* , \\
\beta &= \frac{h\ell^*}{w\ell} , \\
z' &= \frac{{\ell_2}^*}{\beta} + z'_o , \\
v &= hz' , \\
t &= hg(z') , \\
z &= {\ell_1}^*\sin(\alpha) + \beta[g(z') - g(z'_o)] , \\
\xi &= {\ell_1}^*\cos(\alpha) + \beta[\sinh^{-1}(z') - \sinh^{-1}(z'_o)] \text{ and} \\
z'_o &\geq \alpha .
\end{aligned} \tag{IV.2}$$

Here, g is a function which is defined as

$$g(x) = \sqrt{1 + x^2} \tag{IV.3}$$

and, w is the weight per foot of the segment, ℓ is the original length of the segment, ${\ell_1}^*$ is the stretched length of line lying on the bottom, ℓ_1 is the original length of line

which lies on the bottom, ${\ell_2}^*$ is the active length of the line (stretched length of line not on the bottom), ℓ_2 is the original length of line which is now not on the bottom, z'_o is the slope of the line at the bottom end, z' is the slope of the line at the top end, z is the vertical distance between the ends of the segment, ξ is the horizontal distance between the ends of the segment, and t is the tension at the top.

In spite of the number of assumptions on the behavior, the above is a complicated system of equations, and is non trivial to solve. Let us look at a special case where γ and α are zero, i.e. we have a rigid line, a bottom with no slope, and there is line remaining on the bottom. In this case, the above becomes:

$$\begin{aligned} \ell &= \ell_1 + \ell_2 \ , \\ \beta &= \frac{h}{w} \ , \\ z' &= \frac{\ell_2}{\beta} \ , \\ v &= hz' \ , \\ t &= hg(z') \ , \\ z &= \beta[g(z') - 1] \ , \text{ and} \\ \xi &= \ell_1 + \beta \sinh^{-1}(z') \ . \end{aligned} \tag{IV.4}$$

Next, let us look at some of the implications of these equations. First, notice that by combining the second, fifth and sixth equations, we have a simple relationship between the tension and the horizontal force:

$$t = h + zw \ , \tag{IV.5}$$

i.e. so long as we have line on the bottom, the tension is simply the horizontal force plus the weight of line per foot times the water depth. Now, also notice that the only part ℓ_1 plays is as an addition to the horizontal distance. This means that so long as there is line on the bottom, it really does not matter how much. Extra line on the bottom is equivalent to simply moving the anchor. Now this observation is not strictly true because we have neglected the bottom friction. Bottom friction will reduce the pull on the anchor based on how much line is on the bottom and thus the line does have an effect. Also, remember that we assumed the line was rigid. Line on the bottom will stretch and so it will also have an effect. Despite these effects, the basic observation is still valid - excess line on the bottom does very little good. We will come back to what we mean by excess later.

To go about solving the simplified equations, we first assume that h, w, ℓ, and z are given, so that we also know β. By rearranging the above, we can get a solution:

$$\begin{aligned} t &= h + wz \ , \\ z' &= \sqrt{(z/\beta + 1)^2 - 1} \ , \end{aligned}$$

$$\begin{aligned} \ell_2 &= \beta z' \ , \\ \ell_1 &= \ell - \ell_2 \ , \\ v &= h z' \ , \\ \xi &= \ell_1 + \beta \sinh^{-1}(z') \ . \end{aligned} \tag{IV.6}$$

In fact, by virtue of the simple relationship between h and t, we can just as easily obtain a solution if t is known instead of h.

Now let us again consider the question of excess line. Suppose that the breaking strength of the line is t_b, so using the above, we get

$$\begin{aligned} h^* &= t_b - wz, \\ \beta^* &= \frac{h^*}{w}, \\ z'^* &= \sqrt{(z/\beta^* + 1)^2 - 1}, \text{ and} \\ \ell_2^* &= \beta^* z'^*. \end{aligned} \tag{IV.7}$$

Here, ℓ_2^* is the maximum length of line which we should be using as a function of the weight per foot, breaking strength, and water depth. In fact, this set of equations gives the amount of line needed to achieve the desired tension, t_b. Unfortunately, when we relax our assumptions, such a simple state of affairs ceases to exist.

Remember that to achieve these simple results, we basically made two assumptions: the slope at the bottom was zero and that the line was rigid. In spite of appearances, relaxing the rigidity assumption is by far the easiest to accomplish. All that is really necessary is to use a Picard method, i.e. we first solve the problem assuming there is no stretch. We then compute the stretch and use the new lengths to solve again, and repeat until the answers do not change much. This process normally converges in one or two iterations.

Now, let us return to the original system of equations (IV.2) , and notice that there are really two things happening. First of all, the second through seventh equations are somewhat self contained. Let us rewrite them as

$$\begin{aligned} \ell &= \ell_1 + \ell_2 \ , \\ {\ell_1}^* &= \ell_1[1 + h\gamma \cos(\alpha)] \ , \\ {\ell_2}^* &= \ell_2 \left[1 + \frac{h\gamma}{\ell_2} \int_o^{\ell_2} g(z') \right] ds \ , \\ \ell^* &= {\ell_1}^* + {\ell_2}^* \ , \\ \beta &= \frac{h\ell^*}{w\ell} \ , \\ z' &= \frac{{\ell_2}^*}{\beta} + z'_o, \ , \text{ and} \\ z'_o &\geq \alpha \ . \end{aligned} \tag{IV.8}$$

Notice that here we have a system of six equations and an inequality in the seven unknowns: ℓ_1, ${\ell_2}^*$ ${\ell_1}^*$, ℓ^*, β, z', and z'_o, provided ℓ_2, h, w, γ, and α are known. The inequality constraint, however, provides the necessary information to solve these. To see this, consider two situations. First, suppose that the inequality holds. In this case there is no line on the bottom and $\ell_2 = \ell$. Conversely, if equality holds then $z'_o = \alpha$. One or the other of these two equations provides what is necessary to solve the above system. Now, consider the remainder of the equations:

$$\begin{aligned} v &= hz' \ , \\ t &= hg(z') \ , \\ z &= {\ell_1}^* \sin(\alpha) + \beta[g(z') - g(z'_o)] \ , \text{ and} \\ \xi &= {\ell_1}^* \cos(\alpha) + \beta[\sinh^{-1}(z') - \sinh^{-1}(z'_o)] \ . \end{aligned} \tag{IV.9}$$

These simply suffice to produce results once the first system has been solved. Formally, this can be written as

$$\begin{aligned} v &= g_1(h, \ell_2, z'_o) \ , \\ t &= g_2(h, \ell_2, z'_o) \ , \\ z &= g_3(h, \ell_2, z'_o) \ , \\ \xi &= g_4(h, \ell_2, z'_o) \ , \end{aligned} \tag{IV.10}$$

and we have suppressed the dependence of the function on all of the assumed properties of the line. In other words, this says that if we know the properties of a line, the active length, the horizontal force, and the slope at one end, we can compute all of the other things of interest. Unfortunately, this is not really what we want. Notice that two of the things that we compute here are z and ξ. We normally want to specify these and compute h and the active length. For the case of a single segment, there are any number of algorithms available to accomplish what we desire, but for a more general case they become quite difficult to implement and consume quite a bit of computer resources.

Here, we propose a simple solution. By building a table of the properties (tension, horizonal force, etc.) of the line as a function of distance between the two ends for a fixed vertical separation, we solve the above system of equations and pick only the solutions for a given value of z. In other words, what we need to do is pick a set of horizontal forces and then solve the equation

$$z^* = g_2(h, \ell_2, z'_o) \tag{IV.11}$$

for each h to obtain the value of either ℓ_2 or z'_o which gives the desired value of z^*. The inequality constraint complicates matters here, so let us look at a special point where $\ell_2 = \ell$ and $z'_o = \alpha$, i.e. the point where all of the line is active, but remains tangent to the sea floor. If the value of z obtained from the above equation for this

point is greater than the desired value, we must have line on the bottom to achieve the desired value and we will be solving the equations

$$\begin{aligned} z^* &= g_2(h, \ell_2, \alpha) \text{ and} \\ \ell_2 &\leq \ell \ . \end{aligned} \tag{IV.12}$$

Otherwise, we will be solving the equations

$$\begin{aligned} z^* &= g_2(h, \ell, z'_o) \text{ and} \\ z'_o &\geq \alpha \ . \end{aligned} \tag{IV.13}$$

In either case we have a single equation and a single unknown. The question remains as to how to solve this equation since g_2 is a formal representation. Notice, however, that in either case the unknown has a bound. If we could find a bound on the other side, then we could use an interval halving algorithm to obtain a solution. Fortunately, the other bound is not hard to get in either case. For the first one, we know that the length of line must be greater than the distance along the bottom to the vessel and then up. The other case is a bit more difficult. Here, we will use the slope of a taut line as an upper bound.

At this point, we have solved the problem for a single segment of line. If the slope at the bottom is zero, this is a well behaved problem. If, however, the slope is non zero, difficulties arise. Perhaps the most obvious is that the problem can easily become badly posed. For example, suppose that the bottom slope is positive, i.e the sea floor decreases in depth as one approaches the vessel. One can input numbers here which make no sense - cases where the ship is below the seafloor. Alternately, suppose that the slope is a large negative number. This corresponds to the case where there is no seafloor at all. An obvious problem here is the bounding scheme for the halving algorithm - the slope is so large that halving will take forever, or the numbers get so large that numerics makes the problem impossible to solve. A more serious problem, however, is that some of the things we wish to solve for do not have unique solutions. For example, consider a hanging line with ends at the same vertical location. We wish to find the length of line which yields a given tension. There are cases where this problem has two solutions, one with a small ℓ and large h, and another with large ℓ and small h. To see this, consider the problem above with zero slope and ask if there are two values of z which yield the same tension? Therefore, be warned that, when you use sloping bottoms and get questionable results, MOSES may not be to blame - you may have a real solution but it is not the one you wanted.

Now, let us consider a line composed of multiple segments. The same equations apply, only this time they are valid for each segment, and we have n segments. In addition, at the interface between segments, we have an additional condition: the slope is discontinuous as

$$z'_{k+1} = \frac{b_k}{h} + z'_k \tag{IV.14}$$

where z'_{k+1} is the slope at the bottom of the $k+1$ segment, z'_k is the slope at the top of the k segment, and b_k is the weight concentrated at the connection between the segments. So long as all of b_k are positive, our original algorithm works as outlined above except that we must first find which segment marks the boundary between line on the bottom and active line. Negative values of b_k cause difficulties because implicit in all of our arguments has been the fact that the slope of the line is a monotone function of distance from the bottom. The lack of monotonicity also occurs for negative bottom slopes and is responsible for the non-uniqueness discussed above. As a result, it is not surprising that spring buoys cause problems. One of the most obvious problems is that one can have "intermediate" grounding - the line from a bouy goes below the surface. These effects are ignored since they occur at tensions where one is not really interested in the solution anyway. Another problem is that without some constraint, the bouy could be above the water surface. This is a real concern because many buoys are designed to operate floating on the surface and to have them flying about would lead to serious errors. Thus, for each spring bouy we need to add a constraint that instead of a fixed force we have an inequality constraint:

$$\begin{aligned} z_{k+1} &\leq 0 \text{ and} \\ z'_{k+1} - z'_k &\leq \frac{b_k}{h} \, . \end{aligned} \tag{IV.15}$$

So long as the connection point is below the water surface, equality holds; otherwise only the fraction of the buoyancy required to hold the connection in the water surface will be applied.

V. MISCELLANEOUS TOPICS

This section contains several discussions which, ideally, should be parts of larger discussions. None are really long enough to stand by themselves and the larger issues have not been completely addressed. Many people, however, have asked about these things so we have included them.

V.A Beam Slenderness

A critical feature of all structural code checks is the Euler critical stress:

$$\sigma_e = \frac{\pi^2 E}{\lambda^2} \tag{V.1}$$

where the slenderness λ is given by

$$\lambda = \frac{kl}{r}$$

and r, the radius of gyration is given by

$$r = \sqrt{\frac{A}{I}} \tag{V.2}$$

and I is the inertial of the section. Now, for a beam which has the same section along its length, this is well defined. If a beam is composed of several segments with different sections, however, things are not clear. Two possibilities come to mind. One could check each section using a slenderness based on the geometrical properties of that section and the total length, or one could compute a slenderness based on the true buckling behavior of the beam. The first alternative is really too conservative to be of value.

To get an equivalent σ_e, let us recast the above as

$$\sigma_e = \frac{P_c}{A}$$

where P_c is the Euler critical load for the beam as a whole and A is the area of the cross section at any point along the beam. To estimate a value for P_c, we will employ a Raleigh quotient:

$$P_c = \frac{\int_0^{kl} EI y''^2 dx}{\int_0^{kl} y^2 dx}$$

where y is the lateral deformation of the beam. If we make the normal assumption that the deformation is a sine curve this can be written as:

$$P_c = \alpha^2 \frac{\int_0^{kl} EI \cos^2(\alpha x) dx}{\int_0^{kl} \sin^2(\alpha x) dx} \tag{V.3}$$

where $\alpha = \pi/kl$. Now, this looks much better if it is written as

$$P_c = (\frac{\pi}{kl})^2 \overline{EI} \tag{V.4}$$

where

$$\overline{EI} = \frac{\int_0^{kl} EI \cos^2(\alpha x) dx}{\int_0^{kl} \sin^2(\alpha x) dx} \tag{V.5}$$

Combining these with the above yields

$$\bar{\lambda} = \frac{kl}{\bar{r}}$$

where

$$\bar{r} = \sqrt{\frac{EA}{\overline{EI}}} \tag{V.6}$$

Now, what we have accomplished is that one can now use the standard formulae, (V.1) , for computing the Euler critical stress for a non-prismatic beam provided that one uses (V.6) instead of (V.2) for the slenderness at each point.

Finally, let us return to (V.5) and use the fact that the beam is a set of sections with constant properties to get

$$\overline{EI} = \frac{2\alpha}{\pi} \sum_i E_i I_i \int_{kx_{i-1}}^{kx_i} \cos^2(\alpha x) dx$$

or carrying out the integration,

$$\overline{EI} = \sum_i E_i I_i \left\{ \frac{(x_{i-1} - x_i)}{l} + \frac{1}{2\pi} [\sin(2\alpha x_i) - \sin(2\alpha x_{i-1})] \right\} \tag{V.7}$$

Notice that if there is only a single segment, (V.7) reduces to the standard result.

V.B Joint Crushing

Joint crushing is an investigation of whether or not a chord will fail under the action of all of the braces and the chord stress itself. In general, this is a complicated problem and in MOSESit will be simplified by reducing it to a two dimensional problem. In particular, the joint will be modeled as a two dimensional ring with distributed loads applied where the braces intersect the chord. The stresses in the ring are computed at thirty six positions around the ring and then they are compared with allowable.

In computing the load applied by the brace to the ring several assumptions are made. The two most important are: that the force in the brace parallel the chord can be neglected and the distance along the chord over which a brace force will be distributed is the same for all braces. Both of these are simply implications of the use of a two dimensional model. The real question is how is the "effective joint length" obtained. Here, we follow API RP2a. If there are no rings then effective length is 2.5 times the chord diameter plus the maximum distance along the chord between two brace/chord intersection points. If rings are present, then the effective length is$\sqrt{Dt}$ where D is the chord outside diameter and t is the chord thickness.

To compute the forces in the ring, two formulae from Roark [5] are used. In particular, Formula 20 and 8 from Chapter 8 Table 17 are added to represent the loads in the ring. Here, angle over which the load is applied is defined by the brace/chord intersection for a tubular brace and by one inch for non-tubular braces.

The two formulae above yield loads (an axial load, a bending moment, and a shear force). To convert these into stresses, more formulae from Roark are employed - Formula 1, 13, or 15 from Chapter 8, Table 16, depending on the type of ring. For no ring, Formula 1 is used, for rings made from flat bar, Formula 13 is used, and for rings that are tees, Formula 15 is used.

Finally to compute an interaction ratio, one must have allowable stresses. Here, $.6F_y$ is used for axial and bending and $.4F_y$ is used for shear. Both of these factors will be altered by any allowable stress modifier before it is used.

VI. REFERENCES

References

[1] Wilson, E. L. and H. H. Dovey "Solution or reduction of equilibrium equations for large complex structural systems: *Advances in Engineering Software* Vol. 1, No. 1 (1978)

[2] Hoit, Marc and E.L. Wilson "AN EQUATION NUMBERING ALGORITHM BASED ON A MINIMUM FRONT CRITERIA", *Computers & Structures* Vol 16, No. 1-4 (1983).

[3] Bathe, Klaus-Jurgen, *FINITE ELEMENT PROCEDURES IN ENGINEERING ANALYSIS*, Prentis-Hall, Inc., ISBN 0-13-317305-4, (1982).

[4] Oden, J. T., *FINITE ELEMENTS OF NONLINEAR CONTINUA* McGraw-Hill, Inc. (1972).

[5] Warren C. Young *Roark's Formulas for Stress and Strain* Sixth edition, ISBN 0-07-072541-1, (1989)

[6] Cummins, W.E., "The Impulse Response Function and Ship Motions", *Schiffstechnik*, Bd. 9, June 1962, Heft 47.

[7] Dalzal, J.F. and C.H. Kim, "An Analysis of the Quadratic Frequency Response for Added Resistance", *Journal of Ship Research*, Vol 23, No. 3 (1979).

[8] Salvesen, N., "Second-Order Steady State Forces and Moments on Surface Ships in Oblique Regular Waves", *International Symposium on Dynamics of Marine Vehicles and Structures in Waves*, Paper 22, London, April (1979).