THE RITZ METHOD AND ITS APPLICATION TO STRUCTURAL OPTIMIZATION

A Thesis

Presented to

the Faculty of the Department of Electrical Engineering

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

---

by

Souhail Ali El-Asfouri

July 1974

ACKNOWLEDGEMENTS

I am most indebted to my advisers Dr. B. McInnis and Dr. O. G. Johnson whose help and guidance are deeply appreciated. I would like to thank Dr. G. Dawkins and Dr. F. Kay for serving as committee members and Dr. G. Batten for very helpful discussions. Also I would like to thank Sandi White for her efficiency, patience and care in typing this thesis.

THE RITZ METHOD AND ITS APPLICATION TO

STRUCTURAL OPTIMIZATION

An Abstract of a Thesis

Presented to

the Faculty of the Department of Electrical Engineering

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

---

by

Souhail Ali El-Asfouri

July 1974

ABSTRACT

The Ritz method is presented for minimizing a cost functional of the special form $J(u) = \int_0^L u(x)dx$ subject to differential constraints which are non-linear in the control $u(x)$, and which have the form $q'(x) = A(u,x)q(x)$. Through the use of a suitable space of cubic splines on a mesh of norm h on the interval $[0,L]$, the method is used to minimize $J(u)$ and leads to an approximate solution of the constraints $q'(x)$. The application of this method is demonstrated for two problems in structural optimization. The first example deals with minimizing the weight of a column under a critical load, and the second example shows an interesting case of requiring a geometric constraint on the design variable to arrive at a minimum weight of a beam on transverse vibrations for a specified natural frequency.

TABLE OF CONTENTS

INTRODUCTION

Optimization theory is playing an increasingly important role in the design of systems. In the application of optimization theory to design problems the objective is to minimize a performance measure for the system and satisfy certain constraints that are characteristics of the system in question. Variational techniques are used to derive necessary conditions for optimal design or optimal control. In general the variational approach leads to a non-linear two-point boundary-value problem that can not be easily solved analytically to obtain the optimal solution.

An alternative approach is given by the Ritz method which directly minimizes the performance measure over subspaces of piecewise-polynomial functions by obtaining approximations to the optimal control, the corresponding state and the associated performance measure. The Ritz method transforms the variational problem of finding the minimum of a functional $J[u,x]$ (defined for some admissible controls and states) to solving a system of algebraic equations. Thus, the method by-passes the solution of the two-point boundary-value problem and hence the term "direct methods".

The class of problems considered in this thesis consists of a simple linear performance measure $J(u) = \int_0^L u dx$ and self-adjoint differential constraints of the general form $q'(x) = A(u,x)q(x)$, which are non-linear in the control $u(x)$ with boundary conditions $q(0) = 0$. By introducing the vector of Lagrange multipliers $\lambda(x)$ we formulate the Lagrangian as a functional of $q(x)$, $u(x)$ and $\lambda(x)$ which we finally transform to a functional of $\lambda(x)$ only.

Let S be a suitable space of piece-wise cubic polynomials on a mesh of norm h on the interval [0,L]. Then it is shown that the Ritz method enables us to approximate the solution to the Lagrange multipliers $\lambda(x) = \sum_{i=1}^{M} c_i\omega_i(x)$ as a linear combination of cubic spline polynomials $\omega_i(x)$, where the c_i's are unknowns to be determined. Through this approximation the Lagrangian is formulated as a function of the unknown constants c_i only. In order to obtain an extremum to the Lagrangian we require that $\frac{\partial L}{\partial c_i} = 0$ which results in a system of algebraic equations to be solved for c_i. In Chapter 1 the Ritz method is described for solving optimization problems which may result in linear or non-linear equations. The discussion includes a description of the space of splines of order m which proves to be a useful piece-wise polynomial space possessing certain properties.

The application of the Ritz method is presented in Chapter 2 for a minimum weight design of a cantilever column having a certain length and subjected to an axial load. The problem results in a set of linear algebraic equations and the answers show the method to be promising.

In Chapter 3 the Ritz method is applied to another structural optimization problem, that of minimizing the weight of a beam in transverse vibrations at a certain frequency. The problem is an interesting one mathematically as well as physically. This example results in a set of non-linear algebraic equations thus it serves as a good example in illustrating the application of the method to non-linear problems. Also, it turns out that the problem of finding the minimum weight of a vibrating beam does not possess an optimal solution in the absence of a geometric constraint on the design variable. The constraint is introduced in the problem by perturbing the necessary condition for an optimal solution, thus obtaining a sub-optimal solution to the minimum weight design problem.

In the concluding chapter the results are discussed in terms of the degree of accuracy, and the necessity for using higher dimension of cubic splines to obtain better approximations for certain non-linear problems. The possibility of using some alternative algorithms that are used in the main procedure is also suggested.

Chapter 1

THE RITZ METHOD

In this chapter the Ritz method will be presented for minimizing a functional subject to constraints expressed by means of differential equations which are linear, homogeneous and self-adjoint. The functional or the performance measure is of a simple nature although extension of the method to more complicated functionals is possible.

1.1 Statement of the Problem

The problem treated is defined in the following way. Minimize (over u) a cost functional J(u) defined by

$$J(u) = \int_0^L u(x)dx \tag{1.1}$$

subject to the linear differential homogeneous constraints

$$q'(x) = A(u,x)q(x) \tag{1.2}$$

and the homogeneous boundary conditions[1]

$$q(0) = 0 \tag{1.3}$$

where q(x) is an n-dimensional state vector, u(x) is a scalar design function, A(u,x) an n x n matrix, non-linear in

[1]For the case of non-homogeneous boundary conditions see Bosarge and Johnson [1.1] and [1.2].

$u(x)$, x is the independent variable, and L is the terminal of the independent variable. L is fixed and the control $u(x)$ is unconstrained.[2]

The objective of the problem is to find the optimal control, i. e. the control which minimizes the cost functional (1.1) such that the constraints (1.2) and (1.3) are satisfied. In the case of split boundary conditions or conditions given at the terminal of the independent variable x, the same general procedure holds for problems of the type considered.

We note here the characteristics of the optimization problem defined by (1.1) - (1.3). The cost functional is linear in the design function $u(x)$, hence the derivative of $J(u)$ with respect to $u(x)$ does not yield an equation to be solved for $u(x)$ and no boundary conditions are imposed on the design function. As a result, the existence of a solution hinges entirely on the nature of the constraint equations (1.2) and (1.3) which describe the state variable $q(x)$ as a function of the design variable $u(x)$. The key functional is therefore not $J(u)$ but the constraint equations, which are considered as a functional of $u(x)$.

1.2 Lagrangian Formulation

Introducing the Lagrange multipliers $\lambda(x)$, an n-dimensional vector associated with the state equations,

---

[2] A special case of a constrained control is treated in Chapter 3, section 3.6.

and the constant Lagrange multipliers γ, an n-dimensional vector associated with the boundary conditions, define the Lagrangian $L(u,q,\lambda,\gamma)$ by

$$L(u,q,\lambda,\gamma) \equiv J(u) + \int_0^L \langle\lambda,-q' + Aq\rangle\, dx + \langle\lambda,q(0)\rangle \qquad (1.4)$$

where $\langle .,.\rangle$ indicates the inner product of two vectors. With this new formulation, an alternative way of describing the original problem can be stated by the following definition given by Bosarge and Johnson [1.2]:

Definition 1.1. Given the linear system (1.2) and the cost functional (1.1), find the u^*, q^*, λ^* and γ^* such that the Lagrangian is extremized, that is,

$$L(u^*,q^*,\lambda^*,\gamma^*) = \max_{\substack{\lambda\varepsilon A_\lambda \\ \gamma\varepsilon R_n}} \min_{\substack{u\varepsilon A_u \\ q\varepsilon A_q}} L(u,q,\lambda,\gamma) \qquad (1.5)$$

where u^*, q^*, λ^*, γ^* denote optimal quantities. Here A_u is some set of admissible vector-values functions on $[0,L]$, R_n is real Euclidean n-space, and A_q, A_λ, A_u are interrelated.

As noted by Bosarge and Johnson:

> One of the essential properties of the multipliers $\lambda(x)$ and γ is that, in the process of extremizing L over u, q, λ and γ, the Lagrangian is maximized over the multipliers λ and γ and minimized with respect to u and q. [1.2].

This principle of the Lagrange duality is discussed by Luenberger in [1.3].

1.3 Optimality Conditions

The necessary conditions for an optimal solution are

$$\frac{\partial L}{\partial q} [u,q,\lambda,\gamma] = 0 \tag{1.6a}$$

$$\frac{\partial L}{\partial u} [u,q,\lambda,\gamma] = 0 \tag{1.6b}$$

where the above derivatives are partial Fréchet[3] derivatives of the scalar Lagrangian.

The Lagrangian (1.4) can be expressed as

$$L[u,q,\lambda,\gamma] = \int_0^L u dx + \int_0^L \langle\lambda,-q'\rangle\, dx + \int_0^L \langle\lambda,Aq\rangle\, dx + \langle\gamma,q(0)\rangle. \tag{1.7}$$

Integrating the second term by parts

$$\int_0^L \langle\lambda,-q'\rangle\, dx = \langle\lambda,-q\rangle \Big|_0^L + \int_0^L \langle\lambda',q\rangle\, dx, \tag{1.8}$$

and substituting back in the Lagrangian, we obtain

$$L[u,q,\lambda,\gamma] = \int_0^L u dx + \int_0^L \langle\lambda',q\rangle\, dx + \int_0^L \langle\lambda,Aq\rangle\, dx + \langle\gamma,q(0)\rangle + \langle\lambda,-q\rangle \Big|_0^L . \tag{1.9}$$

Since only homogeneous boundary conditions are considered, all boundary terms drop out from the Lagrangian as we shall see later for self-adjoint systems, and (1.9) becomes

---

[3] See Appendix B for the definition of Fréchet derivative.

$$L[u,q,\lambda,\gamma] =$$

$$\int_0^L u\,dx + \int_0^L \langle\lambda',q\rangle\,dx + \int_0^L \langle\lambda,Aq\rangle\,dx. \tag{1.10}$$

Applying the optimality conditions (1.6a) and (1.6b) to the final form of the Lagrangian (1.10) we obtain from (1.6a)

$$\frac{\partial L}{\partial q} = A^T\lambda + \lambda' = 0 \tag{1.11}$$

and from (1.6b)

$$\frac{\partial L}{\partial u} = 1 + \lambda^T\left(\frac{dA}{du}\right)q = 0. \tag{1.12}$$

The costate equations (1.11) can be written as a first order system of n ordinary differential equations,

$$\lambda' = -A^T\lambda. \tag{1.13}$$

Since the matrix $A(u,x)$ is assumed to be nonlinear in the control u, equation (1.12) will yield the optimal design variable $u(x)$. However, since $u(x)$ does not appear explicitly in (1.12) we will assume that the optimal control is some function

$$u^* = u(\lambda,q). \tag{1.14}$$

Consider the adjoint system (1.13), this system is identical to the original system (1.2), hence there is linear mapping of the Lagrange multipliers λ into the state variables q.[4]

---

[4] This is the property of self-adjointness (see Daniel [1.4]).

In other words, λ can be expressed in terms of q or vice versa, and also the boundary conditions on λ can be deduced this way. Therefore equations (1.11) and (1.12) can be expressed in terms of the Lagrange multipliers only as a result of the self-adjointness property of the system. We also observe that L can be written as a functional of the Lagrange multipliers only. Thus the problem is reduced to maximizing L over λ. Therefore

$$\frac{\partial L}{\partial \lambda} = 0 \tag{1.15}$$

is required, where λ is to be defined.

1.4 Spline Subspaces

At this point we introduce a set of admissible functions to express λ in terms of using certain subspaces of piece-wise polynomials. Let

$$\omega_1(x), \omega_2(x), \ldots., \omega_n(x), \ldots. \tag{1.16}$$

be a sequence of elements in the Hilbert space H_A[5] that satisfy the following two conditions [1.5]:

1. for any n, the elements, $\omega_1, \omega_2, \ldots., \omega_n, \ldots.,$ are linearly independent;

2. sequence (1.16) is such that, for any element $\lambda \varepsilon H_A$ and any $\varepsilon > 0$, there exists a natural number N and constants $c_1, c_2, \ldots., c_n$ such that

---

[5] A Hilbert space is simply a Eucledian space which is infinite dimensional and has an inner product property. For rigorous definition see [1.3], [1.6] and [3.1].

$$\left| \lambda - \sum_{k=1}^{N} c_k \omega_k \right| < \varepsilon. \tag{1.17}$$

The elements (1.16) are called coordinate elements.

The Ritz Method makes it possible to construct the approximate solution λ of a variational problem in the form

$$\lambda = \sum_{k=1}^{n} c_k \omega_k \tag{1.18}$$

where the c_k are constants that are selected so that a functional $L(\lambda)$ is minimal.

An extremely useful piecewise polynomial space possessing the above properties is the space of splines.

According to Bosarge and Johnson,

> Spline subspaces have been used frequently in recent years in the development of practical and efficient numerical algorithms for attacking wide classes of problems. In fact for many practical problems, spline subspaces "deliver" the best results for an equivalent amount of computation, compared with an alternative finite dimension space of piecewise polynomials. [1.2]

We consider now the spline interpolation spaces $Sp^{(m)}(\pi), \pi \geq 1$ first considered by Schoenburg [1.7]. Let $\pi: 0=x_0<x_1<\ldots<x_{N+1}=1$ denote a partition of the unit interval with joints x_i, and m is a positive integer. Then $Sp^{(m)}(\pi)$ is the collection of all real piecewise-polynomial functions $\omega(x)$ defined on $[0,1]$ such that $\omega(x) \varepsilon C^{2m-2}[0,1]$ and such that on each subinterval $[x_i, x_{i+1}]$, $0 \leq i \leq N$, determined by π, $\omega(x)$ is a polynomial of degree 2m-1 [1.8]. The class of functions $C^{2m-2}[0,1]$ denotes the continuity of $\omega(x)$ up to the 2m-2 derivatives at all the joints x_i. The spline functions will be discussed more in Appendix A, and for a comprehensive coverage of the topic of splines, see Ahlberg, Nilson, and

Walsh [1.9] and Schoenburg [1.7].

For practical computations, the choice of a basis for $Sp^{(m)}(\pi)$ is very important. The cardinal functions are a natural choice for a basis for $Sp^{(m)}(\pi)$. However, these functions complicate storage and generally mean slower convergence of iterative techniques on digital computers [1.7]. For the definition of cardinal functions see Appendix A.

In this thesis the patch basis are considered as a choice for $Sp^{(m)}(\pi)$. For the case of m=2 of cubic splines on a uniform mesh of norm h, i.e., $x_i=ih$, $0 \le i \le N+1$ the patch basis are given by [1.1]:

$$
\begin{aligned}
\omega_i(x) &= 0 && x \notin [x_i, x_{i+4}] \\
\omega_i(x) &= (x-x_i)^3 && x \in [x_i, x_{i+1}] \\
\omega_i(x) &= h^3 + 3h^2(x-x_{i+1}) + 3h(x-x_{i+1})^2 && \\
&\quad - 3(x-x_{i+1})^3 && x \in [x_{i+1}, x_{i+2}] \\
\omega_i(x) &= 4h^3 - 6h(x-x_{i+2})^2 + 3(x-x_{i+2})^3 && x \in [x_{i+2}, x_{i+3}] \\
\omega_i(x) &= (x_{i+4}-x)^3 && x \in [x_{i+3}, x_{i+4}]
\end{aligned}
\tag{1.19}
$$

and the graph of $\omega_i(x)$ is

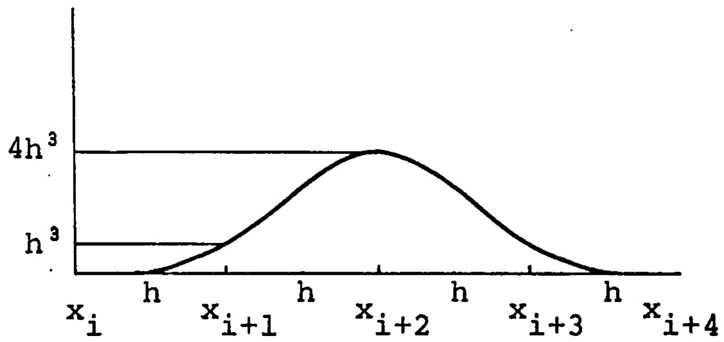

Figure 1.1 $\omega_i(x)$

The patch basis enjoy the very useful properties in computations

$$\int_0^L \omega_i\omega_j dx = 0, \quad |i-j| > 3, \text{ and } \quad \int_0^L \omega'_i\omega'_j dx = 0, \quad |i-j| > 3.$$

Now assume that $\lambda(x) \in Sp^{(m)}(\pi)$, then we can express λ as a linear combination of basis functions $\omega_i, i=1,\ldots,M$. Consequently we write

$$\lambda(x) = \sum_{i=1}^{M} c_i\omega_i(x) \qquad 0 \leq x \leq L \tag{1.20}$$

where each $c_i, i=1,\ldots,M$ is an n-vector[6], n being the dimension of the state variables q.

1.5 Optimizing the Spline Coefficients

We now need an expression for the Lagrangian L in terms of the c_i. As was shown previously L can be expressed as a function of the Lagrange multipliers λ only. Hence substituting for λ from (1.20) we can derive an expression for L in terms of the coefficients of the spline functions c_i, and the Lagrangian will have the form

$$L[c_1,\ldots,c_M] = L[u(c_1,\ldots,c_M),\ q(c_1,\ldots,c_M),\ \lambda(c_1,\ldots,c_M)]. \tag{1.21}$$

Unfortunately, the problem at hand, in its present general form, does not allow expressing L in terms of the c_i explicitly, hence the rest of the procedure will be described

---

[6] M is the dimension of the patch basis, M = N+3 where N is the number of partitions of the interval [0,1]. See Appendix A, for the construction of the patch basis also.

in general terms where necessary and the explicit treatment of the computational procedure will be clarified in the examples to follow.

Extremizing $L[c_1,\ldots,c_M]$ over the c_i, the necessary condition

$$\frac{\partial L}{\partial c_i} = 0, \tag{1.22}$$

which is equivalent to (1.15), must be satisfied.

Before applying equation (1.22), the handling of the boundary conditions will be discussed. Normally the boundary conditions are incorporated in the Lagrangian (see Bosarge and Johnson [1.1]), but since the boundary conditions on λ in this problem deduced from the self-adjointness property are homogeneous, they drop out from the Lagrangian, and they have to be handled separately. A natural way to incorporate these boundary conditions in the procedure is to choose a subset of the admissible spline functions for each Lagrange multiplier λ_i such that this subset satisfies the corresponding boundary condition on λ_i, but this method was observed to induce numerical instability in the computational algorithm. Hence, another way of handling the boundary conditions on λ was used which will be described next.

Corresponding to the homogeneous boundary conditions (1.3) on the state variables, the boundary conditions on the multipliers are of the form

$$\lambda(0) = 0. \tag{1.23}$$

Using (1.20) we can express (1.23) as

$$\lambda(0) = \sum_{i=1}^{M} c_i \omega_i(0) = 0. \tag{1.24}$$

This is a set of n linear homogeneous algebraic equations which can be solved for n spline coefficients, $c_i (i=1,\ldots,n)$, in terms of other coefficients. Substituting for these n coefficients in $L[c_1,\ldots,c_M]$, (1.21) will reduce the number of c_i in the Lagrangian from n x M to n x M - n unknown constants. Now applying the necessary conditions (1.22) for maximizing L (1.21) over c_i will result in a system of (n x M - n) equations which with the system of equations (1.24) can be solved for the n x M constants c_i.

As was mentioned earlier, the Lagrangian can not be written as an explicit function of the coefficients c_i, and we will assume that applying

$$\frac{\partial L}{\partial c_i} = 0 \tag{1.22}$$

will result in a system of algebraic equations of the form

$$Ac + g(c) = 0 \tag{1.25}$$

where the matrix $A = (a_{i,j})$ and the column vector $g(c)$ are given by [1.7],

$$a_{i,j} = \int_0^1 f(\omega_i, \omega_j') \, dx \tag{1.26}$$

and

$$g_i(c) = \int_0^1 f(x, \sum_{j=1}^{M} c_j\omega_j(x))\omega_i(x)dx \qquad (1.29)$$

The system of equations (1.25) may be linear or nonlinear, homogeneous or nonhomogeneous depending upon the nature of the problem itself. The last section of this chapter will be devoted to solving equation (1.25).

Once the optimizing constants are obtained we can get the solution for the multipliers from (1.20) and hence the design variable and the state variables can also be obtained.

1.6 Solving the System of Nonlinear Equations

This section will describe the methods used in solving the system of equations (1.25). If this system of algebraic equations is linear then it can be solved by any of the many techniques available for solving simultaneous linear equations. However, if (1.25) turns out to be nonlinear then care must be taken in choosing a proper method to solve the system without excessive amounts of computation. The method selected here is a modified version of Newton Raphson or quasilinearization method described in detail by Miele [1.10]. Computationally, two problems are involved in Newton's method: the need for solving a linear system at each step, and the need for evaluating the Jacobian of the system at each step, however, this method has the powerful feature of quadratic convergence.

Let us write equation (1.25) in the following form

$$\Psi(x) = 0 \qquad (1.30)$$

where Ψ and x are n-vectors. The modified method of Newton's is based on the property of reducing the cumulative error, P, in the equations by a controlled stepsize α. $\Psi(x)$ is assumed to have first derivatives with respect to the vector x, and that these derivatives are continuous. $\Psi(x)$ is also assumed to have a solution.

Consider a nominal point x and a varied point $\tilde{x}$ such that

$$\tilde{x} = x + \Delta x \tag{1.31}$$

where Δx is the variation in x.

Define the scalar performance index P to be the cumulative error in the functions

$$P = \Psi^T(x)\Psi(x). \tag{1.32}$$

It is clear that P=0 only at a solution of (1.30). As we move from the nominal point to the varied point, the performance index P changes. Variations to first order only are considered. Denote by $\delta(.)$ the first variation, then

$$\delta P = 2\Psi_x^T(x)\delta\Psi(x). \tag{1.33}$$

Now consider the special variations

$$\delta\Psi(x) = -\alpha\Psi(x) \tag{1.34}$$

where α is a scaling factor or stepsize in the range

$$0 \leq \alpha \leq 1. \tag{1.35}$$

For these variations, δP becomes

$$\delta P = -2\alpha \Psi^T(x)\Psi(x) \tag{1.36}$$

or $$\delta P = -2\alpha P. \tag{1.37}$$

Since P is positive and α is positive, then

$$\delta P < 0 \tag{1.38}$$

hence the descent property of the algorithm is satisfied,

i.e., $$\tilde{P} < P. \tag{1.39}$$

To determine the variations Δx, find the first order change in $\Psi(x)$ corresponding to a change Δx,

$$\delta\Psi(x) = \Psi_x^T(x)\Delta x \tag{1.40}$$

where Ψ_x is the n x n Jacobian matrix of the given system (1.30). Equating (1.34) and (1.40) results in

$$\Psi_x^T(x)\Delta x = -\alpha\Psi(x)$$

$$\Psi_x^T(x)\Delta x + \alpha\Psi(x) = 0. \tag{1.41}$$

Note that all quantities are evaluated at the nominal point x, hence we can solve the system (1.41) of n equations that are linear in Δx by any of the methods for solving systems of linear equations. To avoid solving this system for all values of α, a transformation is introduced in the form

$$A = \frac{\Delta x}{\alpha}. \tag{1.42}$$

Then (1.41) becomes

$$\Psi_x^T(x)A + \Psi(x) = 0. \tag{1.43}$$

After solving for A, Δx can be computed for a given $\alpha = \alpha_{ref}$. To start the algorithm, Miele [1.10] chooses the value 1 for α_{ref} and then uses a bisection process on α until

$$P(\alpha) < P(0). \tag{1.44}$$

Once Δx is known the varied point $\tilde{x}$ can be computed from (1.31). Now $\tilde{x}$ becomes the new nominal point, or the new guess at the solution, and the procedure is repeated for this new guess.

The modified quasilinearization algorithm is summarized as follows:

(a) Assume an initial guess at the solution x.

(b) Determine the value of the functions $\Psi(x)$ from (1.30), the Jacobian matrix $\Psi_x(x)$ and the cumulative error $P(0)$ from (1.32), all evaluated at the nominal point x. If $P(0)=0$, then x is the solution.

(c) Determine the vector of variations A by solving equation (1.43) then determine Δx from (1.42).

(d) With Δx known for some value of the stepsize $\alpha = \alpha_{ref}$, compute a varied point $\tilde{x}$ from (1.31) and then evaluate $P(\alpha)$ from (1.32).

(e) For different values of α, iterate on step (d) until the inequality

$$P(\alpha) < P(0)$$

is satisfied.

(f) With the stepsize α known, compute the varied point $\tilde{x}$ from (1.31).

(g) Use $\tilde{x}$ as the new guess at the solution and go to step (b) and iterate the algorithm.

The algorithm terminates when $P \leq \varepsilon$, where ε is some small prescribed value. It must be noted that this modified algorithm is like all Newton algorithm types, in that it guarantees convergence to a solution only when the initial guesses, or the nominal point, is sufficiently close to the solution. In Chapter Three a simple algorithm is given for determining good starting values for the spline coefficients from known physical quantities of the problem at hand.

Chapter 2

APPLICATION OF THE RITZ METHOD TO A MINIMUM WEIGHT COLUMN

As a first example illustrating the main features of the Ritz method we consider the structural problem of determining, for a given load and length, the shape of the column which has the minimum weight or volume. The problem serves as a good example in clarifying the application of the method to a class of homogeneous self-adjoint problems.

2.1 Statement of the Problem

Consider a column of length ℓ and cross-sectional area $A(x)$ which may vary along the length of the column (all cross sections are assumed to be similar). Let $y(x)$ denote the lateral deflection from the straight position caused by an axial load applied at the end of the column. The classical, simple Euler theory states that the bending moment $M(x)$ is

$$M(x) = Py(x). \tag{2.1}$$

The equation of equilibrium of the column in the buckled state is

$$M''(x) = Py''(x). \tag{2.2}$$

Also, the moment $M(x)$ at any cross-section is approximated by

$$M(x) = -EI(x)y''(x) \tag{2.3}$$

where E is the Young's modulus of the column material and I(x) is the moment of inertia of the cross-section about a line through its centroid normal to the plane of the deflected column. Differentiating (2.3) twice with respect to x and equating the result with (2.2) gives

$$\left(EI(x)y''(x)\right)'' + Py''(x) = 0. \tag{2.4}$$

If we denote by $\alpha(x)$ the flexural rigidity of the column,

$$\alpha(x) = EI(x) \tag{2.5}$$

we can write equation (2.4) as [2.1]

$$\left(\alpha(x)y''(x)\right)'' + Py''(x) = 0. \tag{2.6}$$

Equation (2.6) applies to columns that are fixed at one end, say $x = 0$, and clamped (fixed), hinged (pinned) or free at the other end $x = \ell$. In this discussion we will consider the last case, a column fixed at one end and free at the other end.

For a cantilevered column, the boundary conditions are as follows:

At the left end $x = 0$

$$y(0) = 0 \tag{2.7a}$$

$$y'(0) = 0 \tag{2.7b}$$

i. e., the deflection and its slope at $x = 0$ are zero.

At the other end $x = \ell$

$$\alpha(\ell)y''(\ell) = 0 \tag{2.7c}$$

$$(\alpha(\ell)y''(\ell) + Py(\ell))' = 0 \tag{2.7d}$$

i. e., the moment at $x = \ell$ is zero and equation (2.7d) can be obtained from (2.1) and (2.3) at $x = \ell$.

Introducing the new dependent variable $\phi(x)=\alpha(x)y''(x)$, equation (2.6) yields the following second-order differential equation in ϕ [2.2]

$$\phi''(x) + \frac{P}{\alpha}\phi(x) = 0. \tag{2.8}$$

To express the boundary conditions in terms of ϕ, equation (2.8) is integrated once with respect to x, after first replacing $\frac{\phi(x)}{\alpha(x)}$ by $y''(x)$:

$$\int_0^x \phi''(x)dx + P\int_0^x y''(x)dx = 0.$$

Since $y'(0) = 0$, it follows that

$$\phi'(x) - \phi'(0) = -Py'(x).$$

Solving for $y'(x)$, we obtain

$$y'(x) = \frac{1}{P}[\phi'(0) - \phi'(x)]. \tag{2.9}$$

Now equation (2.7c) gives

$$\phi(\ell) = 0, \tag{2.10}$$

and from (2.7d) we have

$$\left(\alpha(\ell)y''(\ell)\right)' = -Py'(\ell)$$

or $$\phi'(\ell) = -Py'(\ell). \tag{2.11}$$

Hence, evaluating (2.9) at $x = \ell$

$$Py'(\ell) = \phi'(0) - \phi'(\ell)$$

and substituting from (2.11), yields

$$\phi'(0) = 0.$$

Therefore the boundary conditions on $\phi(x)$ are

$$\begin{aligned} \phi(\ell) &= 0 \\ \phi'(0) &= 0. \end{aligned} \tag{2.12}$$

We note here that integrating (2.9) with respect to x from $x = 0$ and noting that $y(0) = 0$, yields the equation for the deflection of the column in terms of $\phi(x)$;

$$y(x) = \frac{1}{P}\,[x\phi'(0) - \phi(x) + \phi(0)]. \tag{2.13}$$

Expressing (2.8) as a system of first order differential equations by introducing the state variables q, defined by

$$\begin{aligned} q_1 &= \phi(x) \\ q_2 &= \phi'(x) = q_1', \end{aligned} \tag{2.14}$$

we obtain

$$
\begin{aligned}
q_1' &= q_2 \\
q_2' &= -\frac{P}{\alpha}\, q_1 .
\end{aligned}
\tag{2.15}
$$

or in matrix form

$$q' = Aq \tag{2.16}$$

where

$$A = \begin{bmatrix} 0 & 1 \\ -\frac{P}{\alpha} & 0 \end{bmatrix} \tag{2.17}$$

with the boundary conditions

$$
\begin{aligned}
q_1(\ell) &= 0 \\
q_2(0) &= 0.
\end{aligned}
\tag{2.18}
$$

In order to arrive at a minimum weight design, an assumption must be made relating the mass M and the flexural rigidity of the column. The assumption made by Keller [2.2] is

$$\alpha(x) = k_1 A^2(x) \tag{2.19}$$

where A(x) is the cross-sectional area of the column. A simpler result can be obtained by assuming a linear relation between the flexural rigidity and the mass distribution [2.1]. The linear relation can be shown as follows:

Let $m(x)$ be the mass distribution along the length of the column, and let M be the total mass, then

$$m(x) = \rho A(x)$$

or
$$M = \rho \int_0^{\ell} A(x)\,dx \tag{2.20}$$

where ρ is the density of the material of the column. The flexural rigidity $\alpha(x)$ is given by (2.5)

$$\alpha(x) = EI(x).$$

Because all cross-sections are assumed to be similar, $I(x)$ is related to $A(x)$ by [2.2]

$$I(x) = k_1 A^2(x).$$

Hence, $\alpha(x) = k_1 E\, A^2(x)$

or
$$\int_0^{\ell} \alpha(x)\,dx = k_1 E \int_0^{\ell} A^2(x)\,dx.$$

Dividing the above equation by equation (2.20), we obtain

$$\frac{1}{M} \int_0^{\ell} \alpha(x)\,dx = \frac{k_1 E}{\rho} \frac{\int_0^{\ell} A^2(x)\,dx}{\int_0^{\ell} A(x)\,dx}$$

where the right hand side is a ratio of two definite integrals; hence it can be replaced by a constant multiplied by $\int_0^{\ell} A(x)\,dx$, and the last equation gives

$$\int_0^{\ell} \alpha(x)\ dx = k_2 \int_0^{\ell} A(x)\ dx.$$

Therefore, we can write

$$\alpha(x) = k_2 A(x) = k_3 m(x)$$

where k, k_1, k_2 and k_3 are constants that depend on the cross-sectional shape and the column material.

Thus, the optimum design is characterized by the minimum of

$$M = \frac{1}{k_3} \int_0^{\ell} \alpha(x)\ dx. \tag{2.21}$$

It must be noted that the assumption of linear relation between the mass distribution and the flexural rigidity does simplify the problem greatly. In fact this assumption results in a linear problem rather than a non-linear problem that would result from assumption (2.19).

2.2 Lagrangian Formulation

As a first step toward the Ritz approach, the Lagrangian will be formulated from the final formulation of the cantilevered column:

Minimize

$$M = \frac{1}{k_3} \int_0^{\ell} \alpha(x)\ dx$$

subject to the differential constraints (2.16), and the boundary conditions (2.18). Forming the Lagrangian L by introducing the vector of the Lagrange multipliers $\lambda(x)$ and γ, each of length two, we obtain

$$L = \int_0^\ell \alpha(x)dx + \int_0^\ell \langle\lambda, -q'+Aq\rangle\, dx + \gamma_1 q_1(\ell) + \gamma_2 q_2(0)$$

$$= \int_0^\ell \alpha(x)dx + \int_0^\ell \langle\lambda, -q'\rangle\, dx + \int_0^\ell \langle\lambda, Aq\rangle\, dx$$

$$+ \gamma_1 q_1(\ell) + \gamma_2 q_2(0). \tag{2.25}$$

Integrating the second term by parts, we obtain

$$\int_0^\ell \langle\lambda, -q'\rangle\, dx = -\langle\lambda, q\rangle \Big|_0^\ell + \int_0^\ell \langle\lambda', q\rangle\, dx. \tag{2.26}$$

Hence the Lagrangian can be written as

$$L = \int_0^\ell \alpha(x)dx + \int_0^\ell \langle\lambda', q\rangle\, dx + \int_0^\ell \langle\lambda, Aq\rangle\, dx - \langle\lambda, q\rangle \Big|_0^\ell$$

$$+ \gamma_1 q_1(\ell) + \gamma_2 q_2(0).$$

Expanding the inner product terms we can write

$$L = \int_0^\ell \alpha(x)dx + \int_0^\ell [\lambda_1' \ \lambda_2'] \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} dx + \int_0^\ell [\lambda_1 \ \lambda_2] \begin{bmatrix} q_2 \\ -\frac{P}{\alpha} q_1 \end{bmatrix} dx$$

$$- [\lambda_1 \ \lambda_2] \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \Bigg|_0^\ell + \gamma_1 q_1(\ell) + \gamma_2 q_2(0) \tag{2.27}$$

$$L = \int_0^\ell \alpha(x)dx + \int_0^\ell (\lambda_1' q_1 + \lambda_2' q_2)dx + \int_0^\ell (\lambda_1 q_2 - \lambda_2 \frac{P}{\alpha} q_1)dx$$

$$- \Big[\lambda_1(\ell) q_1(\ell) + \lambda_2(\ell) q_2(\ell) - \lambda_1(0) q_1(0) - \lambda_2(0) q_2(0) \Big]$$

$$+ \gamma_1 q_1(\ell) + \gamma_2 q_2(0). \tag{2.28}$$

2.3 Optimality Conditions

The optimality conditions for minimizing L with respect to q and α are

$$\frac{\partial L}{\partial q} = 0, \qquad \frac{\partial L}{\partial \alpha} = 0 \tag{2.29}$$

where the partial derivatives are Fréchet derivatives:

$$\frac{\partial L}{\partial q_1} = \lambda'_1 - \lambda_2 \frac{P}{\alpha} = 0 \tag{2.30a}$$

$$\frac{\partial L}{\partial q_2} = \lambda'_2 + \lambda_1 = 0 \tag{2.30b}$$

$$\frac{\partial L}{\partial \alpha} = 1 + \lambda_2 \frac{P}{\alpha^2} q_1 = 0. \tag{2.30c}$$

Equation (2.30c) gives the solution for the optimal design variable:

$$\alpha^2(x) = -P\lambda_2(x)q_1(x). \tag{2.31}$$

Consider equations (2.30a) and (2.30b)

$$\lambda'_1 = \lambda_2 \frac{P}{\alpha}$$

$$\lambda'_2 = -\lambda_1. \tag{2.32}$$

This adjoint system is identical to the given system (2.15), hence by inspection

$$\lambda_1 = q_2$$

$$\lambda_2 = -q_1 \tag{2.33}$$

also $\lambda_1(0) = q_2(0) = 0, \quad \lambda_2(\ell) = -q_1(\ell) = 0.$ (2.33a)

From equations (2.31) and (2.33) the solution for the control can be written as

$$\alpha^2 = -\lambda_2 \; P(-\lambda_2)$$

$$= P\lambda_2^2$$

$$\alpha(x) = \sqrt{P}\;\lambda_2(x). \tag{2.34}$$

From equations (2.33) and (2.34) we observe that L can be written as a functional of λ only by substituting for q and α. Thus the problem is of maximizing L over λ. Note that the boundary conditions on q and λ are all homogeneous. Hence all boundary terms drop out from the Lagrangian and (2.28) becomes

$$L = \int_0^{\ell} \alpha(x)\,dx + \int_0^{\ell} (\lambda_1' q_1 + \lambda_2' q_2)\,dx + \int_0^{\ell} (\lambda_1 q_2 - \lambda_2 \frac{P}{\alpha} q_1)\,dx. \tag{2.35}$$

Substitution for q and α from (2.33) and (2.34) yields

$$L = \int_0^{\ell} \sqrt{P}\;\lambda_2 dx + \int_0^{\ell} (-\lambda_1'\lambda_2 + \lambda_2'\lambda_1)\,dx + \int_0^{\ell} (\lambda_1^2 + \sqrt{P}\;\lambda_2)\,dx \tag{2.36}$$

and in its final form the Lagrangian is

$$L = \int_0^{\ell} (\lambda_1^2 + 2\sqrt{P}\;\lambda_2 - \lambda_1'\lambda_2 + \lambda_2'\lambda_1)\,dx. \tag{2.37}$$

2.4 The Ritz Formulation

At this point we are ready to express $\lambda(x)$ as a linear combination of the cubic spline basis elements $\omega_i(x)$. This basis was discussed in Chapter 1 in general form for any number of partitions of the unit interval, but for illustrative purposes, the interval $[0,\ell]$ will be divided into N=4 partitions only and the resulting basis functions will be N+3=7 (see Appendix A).

Let $$\lambda_1(x) = \sum_{i=1}^{7} c_i\omega_i(x) \tag{2.38}$$

and $$\lambda_2(x) = \sum_{i=1}^{7} d_i\omega_i(x) \tag{2.39}$$

where c_i and d_i $(i=1,\ldots,7)$ are the unknown spline coefficients and $\omega_i(x)$ $(i=1,\ldots,7)$ are the cubic spline functions over the interval $[0,\ell]$.

As was discussed in Chapter 1 page 13, the homogeneous boundary conditions on the multipliers λ have to be dealt with separately. The boundary conditions on λ are from (2.33a)

$$\lambda_1(0) = 0$$

$$\lambda_2(\ell) = 0.$$

Substituting for λ_1 and λ_2 from (2.38) and (2.39) respectively we obtain

$$\lambda_1(0) = \sum_{i=1}^{7} c_i\omega_i(0) = 0 \tag{2.40}$$

$$\lambda_2(\ell) = \sum_{i=1}^{7} d_i\omega_i(\ell) = 0. \tag{2.41}$$

From Appendix A equations (A.14) - (A.20) we have

$$\omega_4(0) = \omega_5(0) = \omega_6(0) = \omega_7(0) = \omega_1(\ell) = \omega_2(\ell) = \omega_3(\ell) = \omega_4(\ell) = 0$$

$$\omega_1(0) = \omega_3(0) = \omega_5(\ell) = \omega_7(\ell) = h^3$$

and $\omega_2(0) = \omega_6(\ell) = 4h^3$

where h is the mesh size of the partitions of the interval $[0,\ell]$. Upon substituting these values for ω at the boundaries, equations (2.40) and (2.41) reduce to

$$c_1\omega_1(0) + c_2\omega_2(0) + c_3\omega_3(0) = 0 \qquad (2.40a)$$

$$d_5\omega_5(\ell) + d_6\omega_6(\ell) + d_7\omega_7(\ell) = 0 \qquad (2.41a)$$

or

$$c_1h^3 + c_2(4h^3) + c_3h^3 = 0 \qquad (2.40b)$$

$$d_5h^3 + d_6(4h^3) + d_7h^3 = 0 \qquad (2.41b)$$

which upon dividing by h^3 become

$$c_1 + 4c_2 + c_3 = 0 \qquad (2.40c)$$

$$d_5 + 4d_6 + d_7 = 0. \qquad (2.41c)$$

Any of the constants in (2.40c) and (2.41c) may be solved for in terms of the other constants. We choose here to solve for c_1 and d_7.

$$c_1 = -4c_2 - c_3 \qquad (2.42)$$

$$d_7 = -d_5 - 4d_6. \qquad (2.43)$$

After substitution for c_1 in $\lambda_1(x)$ and d_7 in $\lambda_2(x)$ equations (2.38) and (2.39) become

$$\lambda_1 = (-4c_2-c_3)\omega_1 + c_2\omega_2 + c_3\omega_3 + \sum_{i=4}^{7} c_i\omega_i$$

$$\lambda_2 = \sum_{i=1}^{4} d_i\omega_i + d_5\omega_5 + d_6\omega_6 + (-d_5-4d_6)\omega_7$$

or

$$\lambda_1 = (-4\omega_1+\omega_2)c_2 + (-\omega_1+\omega_3)c_3 + \sum_{i=4}^{7} c_i\omega_i$$

$$\lambda_2 = \sum_{i=1}^{4} d_i\omega_i + (\omega_5-\omega_7)d_5 + (\omega_6-4\omega_7)d_6.$$

Let

$$\begin{aligned} \bar{\omega}_1 &= -4\omega_1 + \omega_2 \\ \bar{\omega}_2 &= -\omega_1 + \omega_3 \\ \bar{\omega}_3 &= -4\omega_7 + \omega_6 \\ \bar{\omega}_4 &= -\omega_7 + \omega_5 \end{aligned} \tag{2.44}$$

then λ_1 and λ_2 can be expressed as

$$\lambda_1 = \bar{\omega}_1 c_2 + \bar{\omega}_2 c_3 + \sum_{i=4}^{7} c_i\omega_i \tag{2.45}$$

$$\lambda_2 = \sum_{i=1}^{4} d_i\omega_i + \bar{\omega}_4 d_5 + \bar{\omega}_3 d_6. \tag{2.46}$$

Now we substitute for λ_1 and λ_2 in their final form (2.45) and (2.46) into (2.37) to get the Lagrangian in terms of the c's and d's

$$L = \int_0^{\ell}\left\{\left[\bar{\omega}_1 c_2 + \bar{\omega}_2 c_3 + \sum_{i=4}^{7} c_i\omega_i\right]^2 + 2\sqrt{P}\left[\sum_{i=1}^{4} d_i\omega_i + \bar{\omega}_4 d_5 + \bar{\omega}_3 d_6\right]\right.$$

$$- \left[\bar{\omega}_1' c_2 + \bar{\omega}_2' c_3 + \sum_{i=4}^{7} c_i\omega_i'\right]\left[\sum_{i=1}^{4} d_i\omega_i + \bar{\omega}_4 d_5 + \bar{\omega}_3 d_6\right] \qquad (2.47)$$

$$\left. + \left[\sum_{i=1}^{4} d_i\omega_i' + \bar{\omega}_4' d_5 + \bar{\omega}_3' d_6\right]\left[\bar{\omega}_1 c_2 + \bar{\omega}_2 c_3 + \sum_{i=4}^{7} c_i\omega_i\right]\right\} dx$$

Maximizing L over c_j $(j=2,\ldots,7)$ and d_k $(k=1,\ldots,6)$ by applying the necessary condition (1.22)

$$\frac{\partial L}{\partial c} = 0$$

we obtain

$$\frac{\partial L}{\partial c_j} = \int_0^{\ell}\left\{2\left[\bar{\omega}_1 c_2 + \bar{\omega}_2 c_3 + \sum_{i=4}^{7} c_i\omega_i\right]\omega_j - \left[\sum_{i=1}^{4} d_i\omega_i + \bar{\omega}_4 d_5 + \bar{\omega}_3 d_6\right]\omega_j'\right.$$

$$\left. + \left[\sum_{i=1}^{4} d_i\omega_i' + \bar{\omega}_4' d_5 + \bar{\omega}_3' d_6\right]\omega_j\right\} dx = 0 \qquad (2.48)$$

$$(j = 2,\ldots,7) \qquad \begin{array}{ll} \omega_j = \bar{\omega}_{j-1} & (j = 2,3) \\ \omega_j = \omega_j & (j = 4,\ldots,7) \end{array}$$

$$\frac{\partial L}{\partial d_k} = \int_0^{\ell}\left\{2\sqrt{P}\omega_k - \left[\bar{\omega}_1' c_2 + \bar{\omega}_2' c_3 + \sum_{i=4}^{7} c_i\omega_i'\right]\omega_k\right.$$

$$\left. + \left[\bar{\omega}_1 c_2 + \bar{\omega}_2 c_3 + \sum_{i=4}^{7} c_i\omega_i\right]\omega_k\right\} dx = 0 \qquad (2.49)$$

$$(k = 1,\ldots,6) \qquad \begin{array}{ll} \omega_k = \omega_k & (k = 1,\ldots,4) \\ \omega_k = \bar{\omega}_{9-k} & (k = 5,6) \end{array}$$

Equations (2.48) and (2.49) constitute twelve equations which in matrix form can be written as

$$Gc = b \tag{2.50}$$

where the vectors c and b are defined as

$$c^T = [c_2\ c_3\ c_4\ c_5\ c_6\ c_7\ d_1\ d_2\ d_3\ d_4\ d_5\ d_6]$$

$$b^T = -2 \int_0^{\ell} \sqrt{P}\ [0\ 0\ 0\ 0\ 0\ 0\ \omega_1\ \omega_2\ \omega_3\ \omega_4\ \overline{\omega}_4\ \overline{\omega}_3]$$

and the 12 x 12 matrix G is defined in partitioned form by

$$G = \left[\begin{array}{c|c} a_{ij} & b_{ij} \\ \hline b_{ji} & 0 \end{array}\right] \tag{2.51}$$

where the elements of the partitions are

$$\begin{aligned} a_{ij} &= 2 \int_0^{\ell} \omega_i \omega_j \, dx \\ b_{ij} &= \int_0^{\ell} (\omega_i \omega_j' - \omega_j \omega_i') dx \end{aligned} \tag{2.52}$$

$$(i = 1,\ldots,6\ ,\ j = 1,\ldots,6) \qquad \begin{aligned} \omega_i &= \overline{\omega}_i && (i = 1,2) \\ \omega_i &= \omega_{i+1} && (i = 3,\ldots,6) \\ \omega_j &= \omega_j && (j = 1,\ldots,4) \\ \omega_j &= \overline{\omega}_{9-j} && (j = 5,6) \end{aligned}$$

The matrix equation (2.50) is a linear homogeneous system of twelve equations which in addition to equations (2.42) and (2.43) can be solved for the spline coefficients c_i and d_i $(i=1,\ldots,7)$. By substituting in (2.38) and (2.39) we can get the Lagrange multipliers $\lambda_1(x)$ and $\lambda_2(x)$ from which we can obtain the design variable $\alpha(x)$ from (2.34).

2.5 Results and Discussion

We note that in this simple example the adjoint system (2.32) with the boundary conditions (2.33a) can be solved analytically to yield the solution

$$\lambda_1(x) = \sqrt{P}\, x \tag{2.53}$$

$$\lambda_2(x) = \sqrt{P}\left(\frac{\ell^2}{2} - \frac{x^2}{2}\right) \tag{2.54}$$

and $\alpha(x)$ from (2.34) has the solution

$$\alpha(x) = P\left(\frac{\ell^2}{2} - \frac{x^2}{2}\right). \tag{2.55}$$

The mass of the column from (2.21)

$$M = \frac{1}{k_3}\int_0^\ell \alpha(x)\,dx = \frac{P}{k_3}\int_0^\ell\left(\frac{\ell^2}{2} - \frac{x^2}{2}\right)dx$$

$$M = \frac{P\ell^3}{3k_3}.$$

The mass of the uniform column is

$$M_u = \frac{1}{k_2}\,\alpha_u\ell$$

where $\alpha_u = \frac{4\ell^3 P}{\pi^2}$, is the flexural rigidity of the uniform column (for example see Shigley [2.3]) and

$$M_u = \frac{4\ell^3 P}{k_2 \pi^2}$$

The saving in volume for the same buckling load P is equal to $1-\frac{M}{M_u} = 1-\frac{\pi^2}{12} = .17754$ or 17.754%.

The results are illustrated in Figure 2.1 for the numerical solution by the Ritz method and for the exact solution from (2.55). Figure 2.1 shows the flexural rigidity for the optimal column subjected to a unit load and having a length of one. In Figure 2.2 the results are shown in terms of the non-dimensional distribution of the flexural rigidity of the optimal column with respect to the uniform column

$$\xi(x) = \frac{\alpha(x)}{\alpha_u} = \frac{\pi^2}{8}\left(1 - \frac{x^2}{\ell^2}\right).$$

For this example where the exact solution can be obtained analytically, the maximum error computed as $|\alpha_{app}-\alpha_{ex}|_{max}$, i.e. the absolute difference between the approximate numerical solution and the exact solution is 0.85520×10^{-6}. The high order of accuracy can be explained by the fact that the solution is contained in the set of cubic spline subspaces.

It should be noted that the interval $[0,\ell]$ was divided into four partitions only, which is somehow a coarse mesh, and this in some problems results in a crude approximation to the exact solution, as we shall see in the next chapter.

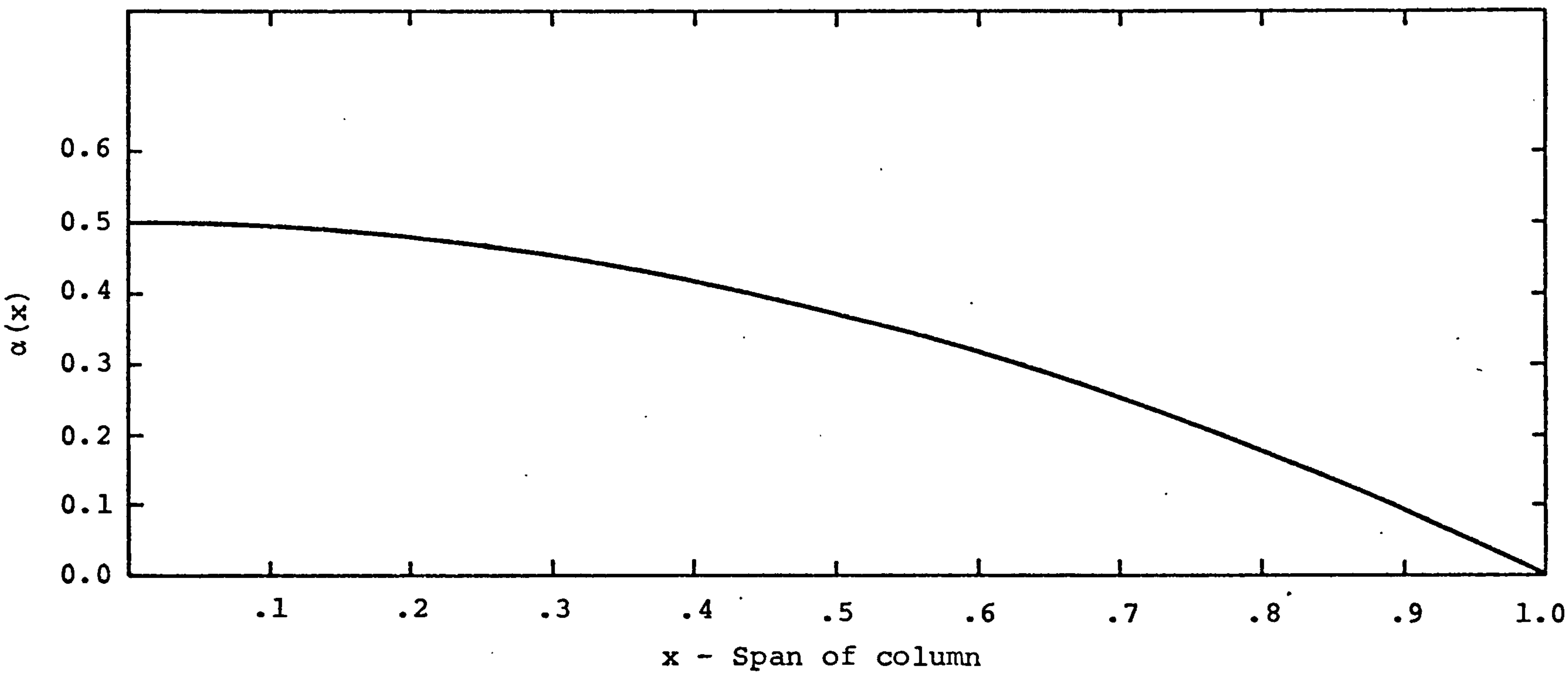

Figure 2.1

Optimum Distribution of Flexural Rigidity in a Clamped-free Column for a Unit Length and Unit Load

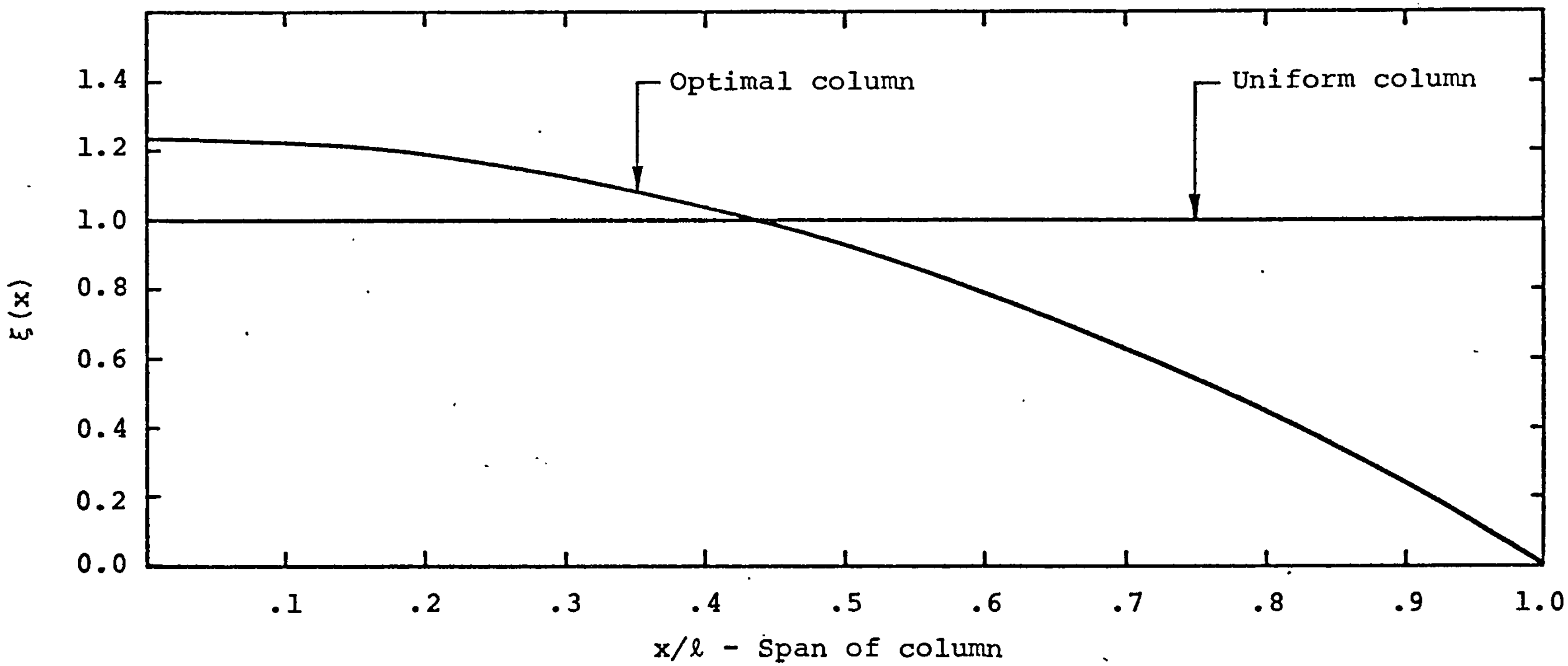

Figure 2.2

Dimensionless Area Distribution in a Clamped-free Column

Chapter 3

APPLICATION OF THE RITZ METHOD TO A MINIMUM WEIGHT VIBRATING BEAM

This chapter deals with the problem of finding the optimal shape of a cantilever beam performing harmonic transverse vibrations which for a given natural frequency would have the lowest possible volume. It turns out that the problem does not possess an optimal solution in the absence of a geometric constraint on the design variable which results in a suboptimal solution. A method for introducing a geometric constraint is presented. Although the cantilever beam has an infinite number of natural frequencies we will be concerned with the lowest frequency. The problem offers an excellent example in illustrating the Ritz method for problems resulting in non-linear algebraic equations.

3.1 Statement of the Problem

For the Bernoulli-Euler cantilever beams, performing small harmonic transverse vibrations under its own weight, let y be the lateral deflection in the plane of bending, $EI(\xi)$ the flexural rigidity, $M(\xi)$ the bending moment at any cross-section (all cross-sections are assumed to be similar) and ξ the coordinate along the axis of the beam, then the differential equation of the deflection curve is [3.1]

$$EI(\xi)\frac{d^2y}{d\xi^2} = -M(\xi). \tag{3.1}$$

Differentiating equation (3.1) twice we obtain

$$\frac{d}{d\xi}\left(EI(\xi)\frac{d^2y}{d\xi^2}\right) = -\frac{dM(\xi)}{d\xi} = -V(\xi) \tag{3.2}$$

$$\frac{d^2}{d\xi^2}\left(EI(\xi)\frac{d^2y}{d\xi^2}\right) = -\frac{d^2M(\xi)}{d\xi^2} = p(\xi) \tag{3.3}$$

where $V(\xi)$ and $p(\xi)$ are the shear and the weight intensity of the beam, respectively. The weight intensity $p(\xi)$ can be obtained by applying Newton's second law and is given by

$$p(\xi) = \rho A(\xi)\frac{\partial^2 y}{\partial t^2} \tag{3.4}$$

where ρ is the density of the material and $A(\xi)$ is the cross-sectional area. The differential equations for the lateral vibrations can thus be written as

$$\frac{d^2}{d\xi^2}\left(EI(\xi)\frac{d^2y}{d\xi^2}\right) = \rho A(\xi)\frac{\partial^2 y}{\partial t^2}. \tag{3.5}$$

When the beam performs a normal mode of vibration the deflection at any location varies harmonically with the time and assuming sustained free vibrations at a frequency ω, the deflection curves can be represented by [3.2]

$$y(\xi,t) = y(\xi)\sin\omega t. \tag{3.6}$$

Differentiating (3.6) twice with respect to time t and substituting in (3.5) results in

$$\frac{d^2}{d\xi^2}\left(EI(\xi)\frac{d^2y}{d\xi^2}\right) - \rho A(\xi)\omega^2 y = 0 \tag{3.7}$$

where y now will be used to represent the maximum amplitude of the motion.

For a cantilever beam we have the following boundary conditions:

at the fixed end $\xi = 0$, the deflection and its slope are zero

$$y\bigg|_{\xi=0} = \frac{dy}{d\xi}\bigg|_{\xi=0} = 0 \tag{3.8a}$$

at the free end $\xi = \ell$, the moment and the shear are zero

$$EI(\xi)\frac{d^2y}{d\xi^2}\bigg|_{\xi=\ell} = \frac{d}{d\xi}\left(EI(\xi)\frac{d^2y}{d\xi^2}\right)\bigg|_{\xi=\ell} = 0. \tag{3.8b}$$

Now, equation (3.7) will be transformed into a dimensionless form by introducing the dimensionless coordinate $x = \xi/\ell$, and the dimensionless area function $\alpha = A\ell/V$, the total volumn of the beam being V [3.3]. Multiplying equation (3.7) by $\frac{\ell^2}{EV^2}$, using a prime (') to indicate differentiation with respect to x, and noting that we introduce a factor $\frac{1}{\ell}$ upon each differentiation with respect to x, we obtain

$$\left(\frac{I\ell^2}{V^2}y''\right)'' - \omega^2\frac{\rho A}{EV}y\ell^6 = 0, \tag{3.9}$$

multiplying again by $\frac{1}{c}$ where $c = I/A^2$, a constant characteristic of the cross-sectional form, equation (3.9) becomes

$$\left(\frac{I\ell^2}{V^2c}y''\right)'' - \omega^2\frac{\rho A}{EcV^2}y\ell^6 = 0, \tag{3.10}$$

or

$$\left(\alpha^2(x)y''\right)'' - \beta\alpha(x)y = 0 \tag{3.11}$$

where

$$\beta = \omega^2\frac{\rho\ell^5}{EcV}. \tag{3.12}$$

Note that β is also a dimensionless constant which upon substituting for ω^2, the natural frequency of vibrations of the uniform beam

$$\omega_n^2 = a_n^2\frac{EI}{\rho A\ell^4}, \tag{3.13}$$

β becomes

$$\beta = a_n^2 \tag{3.14}$$

where a_n is a constant that depends on the mode of vibrations. For the first mode $a_n = 3.515$ [3.4]. The boundary conditions (3.8a) and (3.8b) in this dimensionless form become

$$\begin{aligned} y(0) &= 0 \\ y'(0) &= 0 \\ \alpha^2 y''(1) &= 0 \\ (\alpha^2 y'')'(1) &= 0. \end{aligned} \tag{3.15}$$

It is required to find the distribution of the area function $\alpha(x)$ of a beam vibrating at a certain natural frequency such that the volume of the beam is minimum

$$V = \int_0^1 \alpha(x)\, dx, \tag{3.16}$$

subject to the differential constraint (3.11) and the boundary condition (3.15).

In order to express the equation of motion (3.11) as a first order system of ordinary differential equations, we introduce the vector of state variables q defined by

$$\begin{aligned} q_1 &= y \\ q_2 &= q_1' = y' \\ q_3 &= \alpha^2 q_2' = \alpha^2 y'' \\ q_4 &= q_3' = (\alpha^2 y'')'. \end{aligned} \tag{3.17}$$

Substituting into (3.11), the equation of motion becomes

$$q_4' - \beta\alpha q_1 = 0. \tag{3.18}$$

Hence we have the system of four differential equations

$$\begin{aligned} q_1' &= q_2 \\ q_2' &= \frac{1}{\alpha^2} q_3 \\ q_3' &= q_4 \\ q_4' &= \beta\alpha q_1, \end{aligned} \tag{3.19}$$

or in matrix form

$$q' = Aq \tag{3.20}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{\alpha^2} & 0 \\ 0 & 0 & 0 & 1 \\ \alpha\beta & 0 & 0 & 0 \end{bmatrix}.$$

The boundary conditions (3.15) upon this transformation become

$$\begin{aligned} q_1(0) &= 0, \qquad & q_2(0) &= 0 \\ q_3(1) &= 0, \qquad & q_4(1) &= 0. \end{aligned} \tag{3.21}$$

It must be noted that problem formulation (3.11)-(3.15) is an eigenvalue problem with β being the eigenvalue and y the eigenfunction. Upon multiplying both sides of equation (3.11) by y and integrating between the limits 0 and 1 we obtain the well known formulae of the Rayleigh quotient[1]

$$\beta = \frac{\int_0^1 \alpha^2 (y'')^2 \, dx}{\int_0^1 \alpha y^2 \, dx}. \tag{3.22}$$

[1]The final form of the Rayleigh quotient is obtained by integrating by parts the first term of (3.11) with respect to x twice.

The Rayleigh quotient provides the value of the dimensionless constant β of the optimum beam. This constant is also defined by equation (3.14) and is seen to depend on the mode of vibrations. Consequently, equation (3.22) could be used as a check on the value of β. In other words the value of β for the optimal beam obtained from (3.22) must be the same as the value of β obtained from (3.14). This consequence will be used later in this chapter under discussion of the results.

3.2 Lagrangian Formulation

As a first step toward using the Ritz method, we formulate the Lagrangian for the problem at hand which will be stated in this final form.

Minimize

$$V = \int_0^1 \alpha(x)\, dx$$

subject to the differential constraints (3.20) and the boundary conditions (3.21). Forming the Lagrangian L by introducing the vector of Lagrange multipliers λ of length 4 and γ also of length 4 we can write

$$L = \int_0^1 \alpha(x)\, dx + \int_0^1 \langle \lambda, -q' + Aq \rangle\, dx + \gamma_1 q_1(0) + \gamma_2 q_2(0) + \gamma_3 q_3(1) + \gamma_4 q_4(1). \tag{3.23}$$

Noting that all boundary conditions are homogeneous they drop out from the Lagrangian, and L can be written in the following form

$$L = \int_0^1 \alpha(x)\, dx + \int_0^1 \langle\lambda, -q'\rangle\, dx + \int_0^1 \langle\lambda, Aq\rangle\, dx. \qquad (3.24)$$

Integrating the second term by parts and substituting back into L (see Chapter 1 section 1.3) we obtain

$$L = \int_0^1 \alpha(x)\, dx + \int_0^1 \langle\lambda', q\rangle\, dx + \int_0^1 \langle\lambda, Aq\rangle\, dx + \langle\lambda, -q\rangle\Big|_0^1. \qquad (3.25)$$

3.3 Necessary Conditions for an Optimal Solution

The necessary conditions for minimizing L are (1.6a) and (1.6b)

$$\frac{\partial L}{\partial q} = 0, \qquad \frac{\partial L}{\partial \alpha} = 0$$

where the above derivatives are Fréchet derivatives,

$$\frac{\partial L}{\partial q} = A^T\lambda + \lambda' = 0 \qquad (3.26)$$

$$\frac{\partial L}{\partial \alpha} = 1 + \lambda^T\left(\frac{dA}{d\alpha}\right)q = 0. \qquad (3.27)$$

The costate equations (3.26) can be written as a system of first order differential equations and equation (3.27) yields the necessary conditions λ on the design variable $\alpha(x)$. Thus

$$\lambda' = -A^T\lambda, \qquad (3.28)$$

or

$$\lambda_1' = -\alpha\beta\lambda_4$$

$$\lambda_2' = -\lambda_1$$

$$\lambda_3' = -\frac{1}{\alpha^2}\lambda_2 \tag{3.29}$$

$$\lambda_4' = -\lambda_3$$

and

$$1 + \lambda^T \frac{dA}{d\alpha}q = 0 \tag{3.30}$$

$$1 + \beta\lambda_4 q_1 - 2\alpha^{-3}\lambda_2 q_3 = 0$$

$$\alpha = \left(\frac{2\lambda_2 q_3}{1 + \beta\ \lambda_4 q_1}\right)^{\frac{1}{3}}. \tag{3.31}$$

Consider the adjoint system (3.29). This system is identical to the given system (3.19), and by inspection

$$\lambda_1 = -q_4$$

$$\lambda_2 = q_3$$

$$\lambda_3 = -q_2 \tag{3.32}$$

$$\lambda_4 = q_1$$

The boundary conditions on λ can be obtained from (2.32)

$$\lambda_1(1) = -q_4(1) = 0, \quad \lambda_2(1) = q_3(1) = 0$$

$$\lambda_3(0) = -q_2(0) = 0, \quad \lambda_4(0) = q_1(0) = 0. \tag{3.33}$$

Note that the boundary conditions on λ as well as on q are homogeneous, hence the last term in (3.25) drops out from the Lagrangian which now becomes

$$L = \int_0^1 \alpha(x)\ dx + \int_0^1 \langle\lambda, Aq\rangle\ dx + \int_0^1 \langle\lambda', q\rangle\ dx. \tag{3.34a}$$

Expanding the inner product terms in (3.34) and substituting for q from (3.32), the Lagrangian can be written as

$$L = \int_0^1 \left\{ \alpha + [\lambda_1\ \lambda_2\ \lambda_3\ \lambda_4] \begin{bmatrix} q_2 \\ \frac{1}{\alpha^2} q_3 \\ q_4 \\ \alpha\beta q_1 \end{bmatrix} + [\lambda_1'\ \lambda_2'\ \lambda_3'\ \lambda_4'] \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \right\} dx \tag{3.34b}$$

$$= \int_0^1 \left\{ \alpha + \lambda_1 q_2 + \frac{1}{\alpha^2}\lambda_2 q_3 + \lambda_3 q_4 + \alpha\beta\lambda_4 q_1 + \lambda_1' q_1 + \lambda_2' q_2 + \lambda_3' q_3 + \lambda_4' q_4 \right\} dx \tag{3.34c}$$

$$\int_0^1 \left\{ \alpha - 2\lambda_1\lambda_3 + \frac{1}{\alpha^2}\lambda_2^2 + \alpha\beta\lambda_4^2 + \lambda_1'\lambda_4 - \lambda_2'\lambda_3 + \lambda_2\lambda_3' - \lambda_1\lambda_4' \right\} dx. \tag{3.34d}$$

Substituting for α from (3.31) L becomes, after arranging terms,

$$L = \int_0^1 \left\{ a \left[\lambda_2^{\frac{2}{3}}(1 + \beta\lambda_4^2)^{\frac{2}{3}}\right] - 2\lambda_1\lambda_3 + \lambda_1'\lambda_4 - \lambda_2'\lambda_3 + \lambda_2\lambda_3' - \lambda_1\lambda_4' \right\} dx \tag{3.35}$$

where $a = \frac{3}{2^{\frac{2}{3}}}$.

Hence, the Lagrangian in its final form (3.35) is a function of λ only and now we are ready to express $\lambda(x)$ as a linear combination of the cubic splines basis $\omega_i(x)$. The unit interval [0,1] will be divided into four partitions hence we have the seven basis functions described in Appendix A.

3.4 The Ritz Formulation

Let

$$\lambda_1(x) = \sum_{i=1}^{7} c_i\omega_i(x) \tag{3.36}$$

$$\lambda_2(x) = \sum_{i=1}^{7} d_i\omega_i(x) \tag{3.37}$$

$$\lambda_3(x) = \sum_{i=1}^{7} e_i\omega_i(x) \tag{3.38}$$

and

$$\lambda_4(x) = \sum_{i=1}^{7} f_i\omega_i(x). \tag{3.39}$$

The boundary conditions on the Lagrange multipliers λ from (3.33) can be written in terms of the constants c_i, d_i, e_i and f_i as[2]

$$c_5 + 4c_6 + c_7 = 0 \tag{3.40}$$

$$d_5 + 4d_6 + d_7 = 0 \tag{3.41}$$

$$e_1 + 4e_2 + e_3 = 0 \tag{3.42}$$

$$f_1 + 4f_2 + f_3 = 0. \tag{3.43}$$

---

[2]See Chapter 2, page 30 for complete derivation.

Solving for c_5, d_5, e_1 and f_1, we obtain

$$c_7 = -c_5 - 4c_6 \tag{3.44}$$

$$d_7 = -d_5 - 4d_6 \tag{3.45}$$

$$e_1 = -4e_2 - e_3 \tag{3.46}$$

$$f_1 = -4f_2 - f_3. \tag{3.47}$$

Substituting equations (3.44) - (3.47) into equations (3.36) - (3.39) we get[3]

$$\lambda_1(x) = \sum_{i=1}^{4} c_i\omega_i + c_5\overline{\omega}_4 + c_6\overline{\omega}_3 \tag{3.48}$$

$$\lambda_2(x) = \sum_{i=1}^{4} d_i\omega_i + d_5\overline{\omega}_4 + d_6\overline{\omega}_3 \tag{3.49}$$

$$\lambda_3(x) = e_2\overline{\omega}_1 + e_3\overline{\omega}_2 + \sum_{i=4}^{7} e_i\omega_i \tag{3.50}$$

$$\lambda_4(x) = f_2\overline{\omega}_1 + f_3\overline{\omega}_2 + \sum_{i=4}^{7} f_i\omega_i. \tag{3.51}$$

Equations (3.48) - (3.51) are substituted into L, equation (3.35), to get the Lagrangian in terms of the c's and d's, e's and f's

---

[3]The functions $\overline{\omega}_i(x)$ are defined by (2.44) Chapter 2 and (A.21) in Appendix A.

$$
\begin{aligned}
L = \int_0^1 \Bigg\{ & a\left[\sum_{i=1}^{4} d_i\omega_i + d_5\overline{\omega}_4 + d_6\overline{\omega}_3\right]^{\frac{2}{3}} \left[1 + \beta\left(f_2\overline{\omega}_1 + f_3\overline{\omega}_2 + \sum_{i=4}^{7} f_i\omega_i\right)^2\right]^{\frac{2}{3}} \\
& - 2\left[\sum_{i=1}^{4} c_i\omega_i + c_5\overline{\omega}_4 + c_6\overline{\omega}_3\right]\left[e_2\overline{\omega}_1 + e_3\overline{\omega}_2 + \sum_{i=4}^{7} e_i\omega_i\right] \\
& + \left[\sum_{i=1}^{4} c_i\omega_i' + c_5\overline{\omega}_4' + c_6\overline{\omega}_3'\right]\left[f_2\overline{\omega}_1 + f_3\overline{\omega}_2 + \sum_{i=4}^{7} f_i\omega_i\right] \\
& - \left[\sum_{i=1}^{4} d_i\omega_i' + d_5\overline{\omega}_4' + d_6\overline{\omega}_3'\right]\left[e_2\overline{\omega}_1 + e_3\overline{\omega}_2 + \sum_{i=4}^{7} e_i\omega_i\right] \\
& + \left[e_2\overline{\omega}_1' + e_3\overline{\omega}_2' + \sum_{i=4}^{7} e_i\omega_i'\right]\left[\sum_{i=1}^{4} d_i\omega_i + d_5\overline{\omega}_4 + d_6\overline{\omega}_3\right] \\
& - \left[f_2\overline{\omega}_1' + f_3\overline{\omega}_2' + \sum_{i=4}^{7} f_i\omega_i'\right]\left[\sum_{i=1}^{4} c_i\omega_i + c_5\overline{\omega}_4 + c_6\overline{\omega}_3\right] \Bigg\} dx
\end{aligned}
\tag{3.52}
$$

The Lagrangian is written in this lengthy form as an explicit function of the spline coefficients for illustrative purposes. One could work in the form of equation (3.35) as we will see in the constrained problem page 60, bearing in mind that λ is a function of the constants and hence the derivatives of λ with respect to the corresponding constraints must be considered.

Maximizing L over c_j, d_j ($j = 1,\ldots,6$) and e_k, f_k ($k = 2,\ldots,7$) by applying the necessary conditions (1.22)

$$\frac{\partial L}{\partial c} = 0$$

we obtain

$$\frac{\partial L}{\partial c_j} = \int_0^1 \left\{ -2\left[e_2\bar{\omega}_1 + e_3\bar{\omega}_2 + \sum_{i=4}^{7} e_i\omega_i\right]\omega_j + \omega_j'\left[f_2\bar{\omega}_1 + f_3\bar{\omega}_2 + \sum_{i=4}^{7} f_i\omega_i\right] - \left[f_2\bar{\omega}_1' + f_3\bar{\omega}_2' + \sum_{i=4}^{7} f_i\omega_i'\right]\omega_j \right\} dx = 0 \tag{3.53}$$

$$\frac{\partial L}{\partial d_j} = \int_0^1 \left\{ \frac{2}{3}a\left[\sum_{i=1}^{4} d_i\omega_i + d_5\bar{\omega}_4 + d_6\bar{\omega}_3\right]^{-\frac{1}{3}} \left[1 + \beta\left(f_2\bar{\omega}_1 + f_3\bar{\omega}_2 + \sum_{i=4}^{7} f_i\omega_i\right)^2\right]^{\frac{2}{3}} \omega_j - \left[e_2\bar{\omega}_1 + e_3\bar{\omega}_2 + \sum_{i=4}^{7} e_i\omega_i\right]\omega_j' + \left[e_2\bar{\omega}_1' + e_3\bar{\omega}_2' + \sum_{i=4}^{7} e_i\omega_i'\right]\omega_j \right\} dx = 0$$

$$(j = 1,\ldots 6) \qquad \begin{aligned} \omega_j &= \omega_j \quad (j = 1,\ldots,4) \\ \omega_j &= \bar{\omega}_{9-j} \quad (j = 5,6) \end{aligned} \tag{3.54}$$

$$\frac{\partial L}{\partial e_k} = \int_0^1 \left\{ -2\left[\sum_{i=1}^{4} c_i\omega_i + c_5\bar{\omega}_4 + c_6\bar{\omega}_3\right]\omega_k - \left[\sum_{i=1}^{4} d_i\omega_i' + d_5\bar{\omega}_4' + d_6\bar{\omega}_3'\right]\omega_k + \left[\sum_{i=1}^{4} d_i\omega_i + d_5\bar{\omega}_4 + d_6\bar{\omega}_3\right]\omega_k' \right\} dx = 0 \tag{3.55}$$

$$\frac{\partial L}{\partial f_k} = \int_0^1 \left\{ \frac{4}{3}a\beta\left[f_2\bar{\omega}_1 + f_3\bar{\omega}_2 + \sum_{i=4}^{7} f_i\omega_i\right]\left[1 + \beta\left(f_2\bar{\omega}_1 + f_3\bar{\omega}_2 + \sum_{i=4}^{7} f_i\omega_i\right)^2\right]^{-\frac{1}{3}} \omega_k \left[\sum_{i=1}^{4} d_i\omega_i + d_5\bar{\omega}_4 + d_6\bar{\omega}_3\right]^{\frac{2}{3}} + \left[\sum_{i=1}^{4} c_i\omega_i' + c_5\bar{\omega}_4' + c_6\bar{\omega}_3'\right]\omega_k - \left[\sum_{i=1}^{4} c_i\omega_i + c_5\bar{\omega}_4 + c_6\bar{\omega}_3\right]\omega_k' \right\} dx = 0 \tag{3.56}$$

$$(k = 2,\ldots 7) \qquad \begin{aligned} \omega_k &= \bar{\omega}_{k-1} \quad (k = 2,3) \\ \omega_k &= \omega_k \quad (k = 4,\ldots,7) \end{aligned}$$

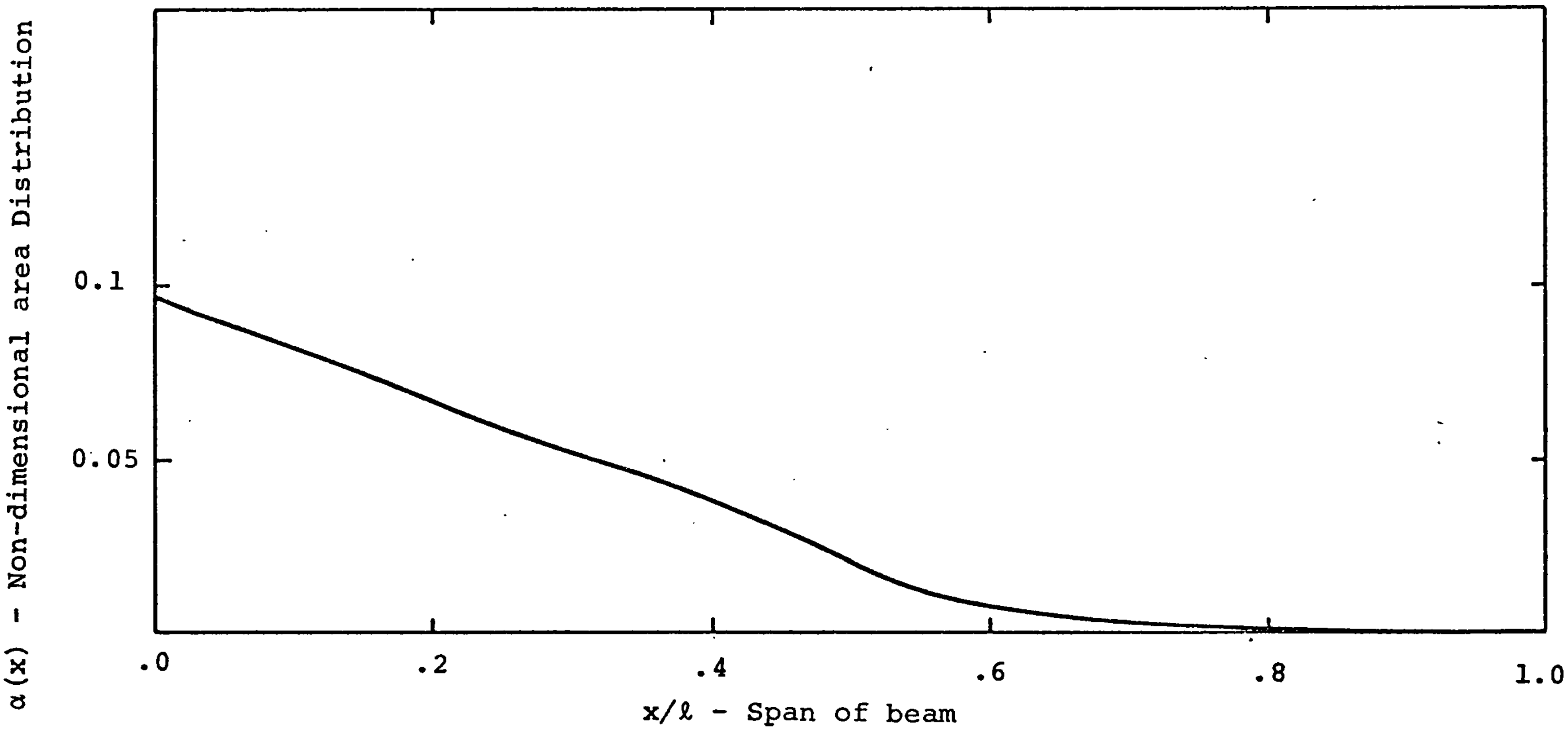

Figure 3.1

Cross-sectional Area Distribution in a Vibrating Beam (Unconstrained)

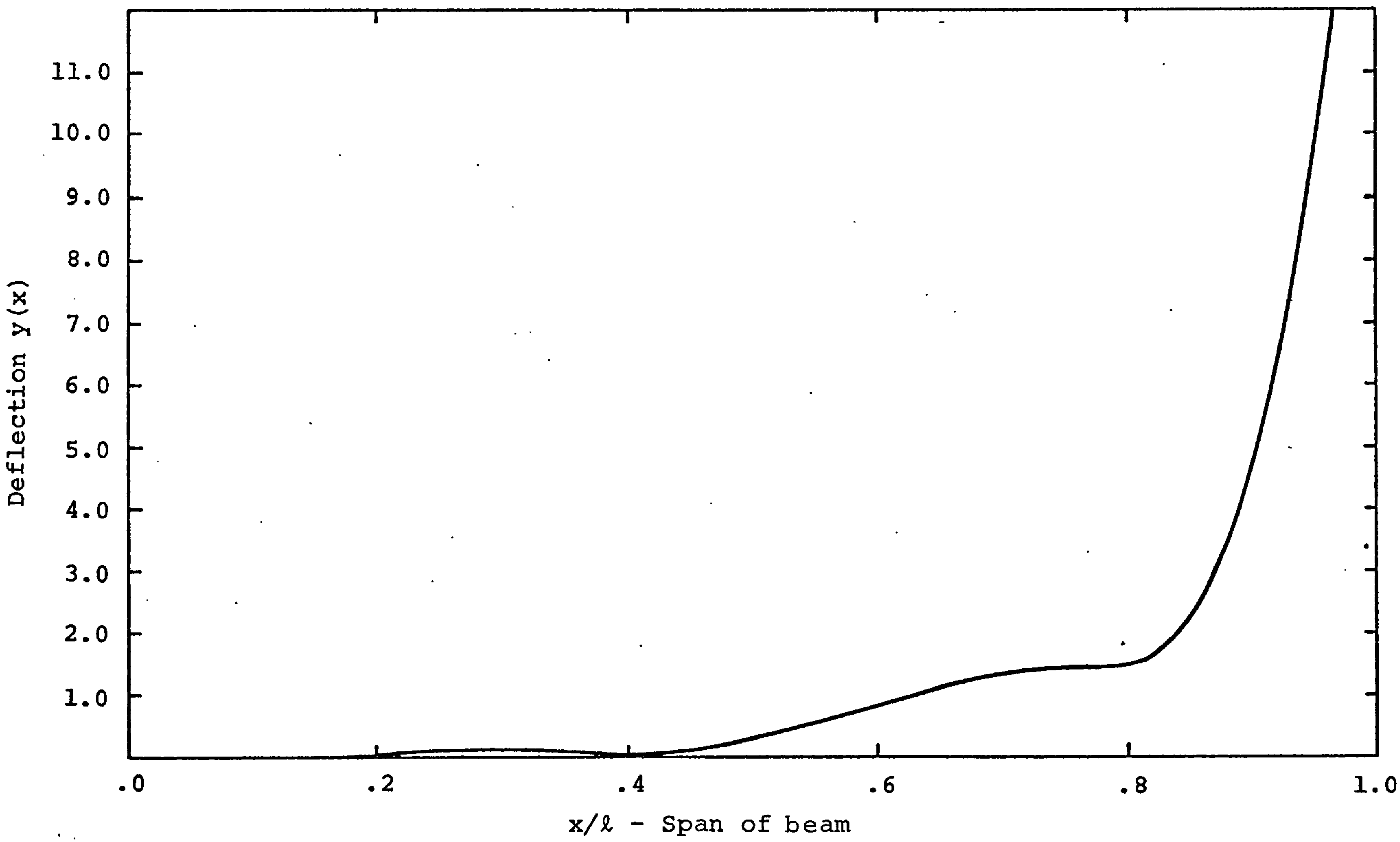

Figure 3.2

Lateral Deflection of a Vibrating Beam at First Fundamental Frequency (Unconstrained)

Equations (3.53) - (3.56) constitute a system of twenty-four equations where twelve of these equations (3.53) and (3.55) are linear in the unknown constants but these are coupled to the other twelve non-linear equations, hence (3.53) - (3.56) must be solved as a system of twenty-four non-linear simultaneous algebraic equations. This system of equations can be solved by the Newton-Raphson method described in Chapter 1 and with equations (3.44) - (3.47) this system gives the spline function coefficients c_i, d_i, e_i, f_i $(i = 1,\ldots,7)$. Once these constants are known we can obtain the Lagrange multipliers (3.48) - (3.51), the state variables (3.32) and the design variable $\alpha(x)$ (3.31). Figures 3.1 and 3.2 show the area function and the deflection of the beam whose behavior will be discussed latter in this chapter.

3.5 Existence of Optimal Solution

At this point we discuss the non-optimality of the solution for the design variable (3.31) obtained from the necessary condition for an optimal solution, equation (1.6b)

$$\frac{\partial L}{\partial \alpha} = 0.$$

Vepa [3.5] shows that there exists no non-trivial solution which satisfies the conditions of optimality for a cantilever beam. To show this let us start with a condition that serves as an additional necessary condition for optimality derived by Pontryagin and his co-workers [3.3]:

If the terminal of the independent variable is fixed and the Hamiltonian does not depend explicitly on the independent variable, then the Hamiltonian must be a constant when evaluated on an extremal trajectory; that is

$$H\left(q^*(x),\ u^*(x),\ \lambda^*(x)\right) = c \quad \text{for } x \in [0,1]$$

where c is a constant, and the Hamiltonian, for the class of problems considered here, is defined by

$$H\left(q(x),\ u(x),\ \lambda(x)\right) = u(x) + \lambda^T(x)A(u,x)q(x).$$

Proof:

Using chain rule for differentiation, we can write the total derivative of the Hamiltonian with respect to x at the optimal solutions as:

$$\frac{d}{dx} H\left(q^*(x),\ u^*(x),\ \lambda^*(x)\right) = \frac{\partial H}{\partial q}\frac{dq}{dx} + \frac{\partial H}{\partial \lambda}\frac{d\lambda}{dx} + \frac{\partial H}{\partial u}\frac{du}{dx} + \frac{\partial H}{\partial x}$$

where $\frac{d}{dx}$ denotes total derivative and $\frac{\partial H}{\partial x}$ denotes partial derivative. From the Hamiltonian consideration for optimization problems, the optimality conditions are (see for example Kirk [3.6])

$$\frac{dq}{dx} = \frac{\partial}{\partial \lambda} H\left(q^*(x),\ u^*(x),\ \lambda^*(x)\right)$$

$$\frac{d\lambda}{dx} = -\frac{\partial}{\partial q} H\left(q^*(x),\ u^*(x),\ \lambda^*(x)\right)$$

$$0 = \frac{\partial}{\partial u} H\left(q^*(x),\ u^*(x),\ \lambda^*(x)\right).$$

Substituting in the expression for $\frac{dH}{dx}$, we obtain

$$\frac{d}{dx} H\left(q^*(x),\ u^*(x),\ \lambda^*(x)\right) = \frac{\partial}{\partial x} H\left(q^*(x),\ u^*(x),\ \lambda^*(x)\right)$$

but since H does not depend explicitly on x, then

$$\frac{\partial}{\partial x} H\left(q^*(x),\ u^*(x),\ \lambda^*(x)\right) = 0.$$

Hence

$$\frac{d}{dx} H\left(q^*(x),\ u^*(x),\ \lambda^*(x)\right) = 0$$

and therefore

$$H\left(q^*(x),\ u^*(x),\ \lambda^*(x)\right) = \text{constant} \qquad x \in [0,1].$$

For the cantilevered beam the Hamiltonian is

$$H = \alpha + \lambda_1 q_2 + \lambda_2 \frac{1}{2\alpha^2} q_3 + \lambda_3 q_4 + \lambda_4 \beta\alpha q_1.$$

At $x = 0$, $q_1(0) = q_2(0) = \lambda_3(0) = \lambda_4(0) = 0$, and

$$H = \alpha + \lambda_2 \frac{1}{2\alpha^2} q_3.$$

From equation (3.32) $q_3 = \lambda_2$, hence

$$H\Big|_{x=0} = \left[\alpha + \frac{q_3^2}{\alpha^2}\right]_{x=0}.$$

From equation (3.31) $\alpha\Big|_{x=0} = (2q_3^2)^{\frac{1}{3}}\Big|_{x=0}$ since $\lambda_4(0) = 0$. Also, $q_3(0)$ is the moment at $x = 0$ which is known to be $\neq 0$ for a cantilevered beam, hence

$$H\Big|_{x=0} \neq 0, \text{ and } H(x) = \text{constant} \neq 0.$$

At $x = 1$, $q_3(1) = q_4(1) = \lambda_1(1) = \lambda_2(1) = 0$, and

$$H\Big|_{x=1} = \Big[\alpha + \lambda_4\beta\alpha q_1\Big]_{x=1} .$$

Inspection of equation (3.31) shows that as a result of the boundary condition $q_3(1) = 0$ (the moment is zero at the free end) α must be zero at $x = \ell$, hence

$$H\Big|_{x=1} = 0,$$

but this contradicts the requirement of the Hamiltonian being constant in $x \in [0,1]$. Therefore, Vepa [3.5] concludes that

> The problem of minimizing the volume or weight of a cantilever beam, keeping the first fundamental frequency in transverse vibration constant, does not possess a solution in the absence of geometric constraints on the design variable.

Obviously, the above result does not state what kind of constraint must be placed on the design variable in order to attain an optimal solution. Vapa proceeds to impose his constraint in the form of an inequality constraint on the linear mass density,[5] $\mu(x)$,

$$\mu \geq \mu_b$$

where μ_b is a lower bound on the linear mass density of

---

[5]Vepa's formulation is in terms of the linear mass density $\mu(x)$ defined by $M = \int_0^\ell \mu(x)\,dx$ where M is the total mass of the beam and ℓ is the length, rather than the area function formulation used here.

the beam. This form of constraint results in a portion of the beam having a constant radius for a distance ℓ_b, where ℓ_b is spanwise location in beam where $\mu = \mu_b$ and also resulting in a corner at $x = \ell_b$. In the next section a sub-optimal solution is presented which yields a smooth shape without the corner.

3.6 Solution with Constraint

It was shown previously that an optimal solution to the cantilevered beam does not exist in the absence of a geometric constraint on the design variable $\alpha(x)$. In this section we will place a constraint on $\alpha(x)$ by perturbing the necessary condition (3.31) and requiring that the area distribution of the beam at the free end to be a finite positive amount rather than zero as would result from (3.27). One way to introduce this restriction and obtain a sub-optimal[6] solution is to add this lower bound on $\alpha(x)$ to the right hand side of (3.31).

Assume that $\alpha(x)$ is desired to be equal to α_b' at the end $x = 1$, then the solution for $\alpha(x)$ becomes

$$\alpha(x) = \left(\frac{2\lambda_2^2}{1 + \beta\ \lambda_4^2} + \alpha_b\right)^{\frac{1}{3}} \tag{3.57}$$

where $\alpha_b = (\alpha_b')^3$.

---

[6]Since we are perturbing the solution obtained from the necessary conditions, the solution obtained with this perturbation is not optimal, hence we will call it a sub-optimal solution.

We note here that since the first term inside the parenthesis in the above equation goes to 0 at x = 1, then $\alpha = \alpha_b^{\frac{1}{3}}$ at the free end. This perturbation on $\alpha(x)$ does not cause any changes either in the form of the equation of motion of the beam (3.11), or in the boundary conditions (3.21). However the Lagrange multipliers will be perturbed and hence the sub-optimal solution (3.57) will not in effect just raise the solution (3.31) by the amount α_b', rather equation (3.57) will have the same shape obtained for the unconstrained case but the volume of the sub-optimal beam will increase as we shall see later in this chapter.

In order to get the Lagrangian as a function of the Lagrange multipliers only, the same procedure as described in the previous sections of this chapter is followed up to equation (3.34d). At that point we substitute for α from (3.57) into the Lagrangian to obtain after arranging terms

$$L = \int_0^1 \Big\{ (1 + \beta\lambda_4^2)^{\frac{2}{3}} \left(2\lambda_2^2 + \alpha_b(1 + \beta\lambda_4^2)\right)^{\frac{1}{3}} + \left(2\lambda_2^2 + \alpha_b(1 + \beta\ \lambda_4^2)\right)^{\frac{2}{3}}\lambda_2^2$$

$$- 2\lambda_1\lambda_3 + \lambda_1'\lambda_4 - \lambda_2'\lambda_3 + \lambda_3'\lambda_2 - \lambda_4'\lambda_1 \Big\}\, dx. \qquad (3.58)$$

with the boundary conditions

$$\begin{aligned} &\lambda_1(1) = 0, \qquad \lambda_2(1) = 0 \\ &\lambda_3(0) = 0, \qquad \lambda_4(0) = 0. \end{aligned} \qquad (3.59)$$

Upon maximizing the Lagrangian over the multipliers λ by taking the partial derivatives of L with respect to the coefficients of the splines functions, c_i and e_i $(i=1,...,7)$ we obtain the same linear equations as in the non-constrained problem equations (3.53) and (3.55). Moreover, the linear terms in the non-linear equations (3.54) and (3.56), obtained by taking the derivatives of L with respect to d_i and f_i $(i=1,...,7)$, will also be the same, but the non-linear terms will be different as we will see.

The derivatives of the Lagrangian with respect to the coefficients of the spline functions after substituting for λ from (3.48) - (3.51) is equivalent to taking the derivatives of L with respect to the Lagrange multipliers λ (see Chapter 1 page 13). Hence maximizing L over the constants will be written in terms of λ for the sake of compactness and to show the generalization of the problem to any number of partitions of the unit interval [0,1], the domain of this problem. The term "+ Linear Terms" refers to adding the linear terms from the corresponding equations of the unconstrained problem (3.54) and (3.56). Taking the derivative of L with respect to the constants gives after simplification

$$\frac{\partial L}{\partial d_j} = \int_0^1 \left\{ (2\lambda_2^j)(1+\beta\lambda_4^2)^{\frac{2}{3}} \left[2\lambda_2^2 + \alpha_b(1+\beta\lambda_4^2) \right]^{-\frac{5}{3}} \left[2\lambda_2^3 + \frac{5}{3}\alpha_b\lambda_2(1+\beta\lambda_4^2) \right] + \text{Linear Terms} \right\} dx = 0 \qquad (3.60)$$

$$(j = 1,...,6)$$

where λ_2^j is the derivative of λ_2 with respect to d_j defined by

$$\begin{aligned} \lambda_2^j &= \omega_j \quad (j = 1,\ldots,4) \\ \lambda_2^j &= \bar{\omega}_{9-j} \quad (j = 5,6). \end{aligned} \tag{3.61}$$

$$\frac{\partial L}{\partial f_k} = \int_0^1 \Big\{ (\tfrac{4}{3}\beta\lambda_4^k)(1 + \beta\lambda_4^2)^{-\frac{1}{3}} \Big[2\lambda_2^2 + \alpha_b(1 + \beta\lambda_4^2) \Big]^{-\frac{2}{3}} \Big[3\lambda_2^2\lambda_4 + \alpha_b(\lambda_4 + \beta\lambda_4^3) \Big]$$

$$+ \Big[\tfrac{2}{3}\alpha_b^2\beta\lambda_4^k \Big] (1 + \beta\lambda_4^2)^{\frac{5}{3}}(\lambda_4) \Big[2\lambda_2^2 + \alpha_b(1 + \beta\lambda_4^2) \Big]^{-\frac{5}{3}}$$

$$+ \text{Linear Terms} \Big\} dx = 0 \tag{3.62}$$

$$(k = 2,\ldots,7)$$

where λ_4^k is the derivative of λ_4 with respect to f_k defined by

$$\begin{aligned} \lambda_4^k &= \bar{\omega}_{k-1} \quad (k = 2,3) \\ \lambda_4^k &= \omega_k \quad (k = 4,\ldots,7). \end{aligned} \tag{3.63}$$

Equations (3.61) and (3.63) in addition to (3.53) and (3.55) will be solved now as a system of twenty-four non-linear simultaneous algebraic equations by the quasilinearization method described in Chapter 1, section 1.6. These equations in addition to equations (3.44) - (3.47) give the coefficients of the spline functions, c_i, d_i, e_i and f_i ($i = 1,\ldots,7$) after which we can solve for the Lagrange multipliers and the constrained design variable (3.57). It must be noted that the quasilinearization algorithm requires evaluating the Jacobian of the system of equations to be solved as discussed in Chapter 1 page 17. The Jacobian was evaluated analytically and the resulting equations are shown in Appendix C.

3.7 Initial Guesses at the Spline Coefficients

It was noted in Chapter 1 section 1.5, that to guarantee convergence of the quasilinearization algorithm for solving a system of non-linear equations, good initial guess at the unknown variables is needed. In this section a simple algorithm is given for determining good estimates for functions whose general shape can be known from physical considerations. For example, the deflection of the beam considered in this chapter can be approximated by a quadratic or cubic general shape (at least on a large portion of the span of the beam), and the area distribution can be considered to have, initially, the shape of the uniform beam, then from equations (3.17) and (3.22), the Lagrange multipliers λ can be approximated by certain shapes. These shapes must be approximated by constant, linear, quadratic or cubic functions because these types of functions are contained in the set of cubic spline functions we are using.

Let some function, λ, be known to have the general shape

$$\lambda(x) = ax^3 + bx^2 + cx + d \qquad 0 \le x \le 1 \qquad (3.64)$$

where a, b, c and d are known coefficients.

Let

$$\lambda(x) = \sum_{i=1}^{7} c_i \omega_i(x) \qquad 0 \le x \le 1 \tag{3.65}$$

where ω_i are cubic spline functions assumed to be in the unit interval [0,1] of a uniform mesh $h = \frac{1}{4}$, and c_i are the coefficients of these spline functions. Then in the first interval [0,h], $\lambda(x)$ can be written as

$$\lambda(x) = c_1\omega_1(x) + c_2\omega_2(x) + c_3\omega_3(x) + c_4\omega_4(x) \tag{3.66}$$

where in this interval (see Appendix A)

$$\begin{aligned}
\omega_1(x) &= (h - x)^3 = h^3 - 3h^2x + 3hx^3 - x^3 \\
\omega_2(x) &= 4h^3 - 6hx^2 + 3x^3 \\
\omega_3(x) &= h^3 + 3h^2x + 3hx^2 - 3x^3 \\
\omega_4(x) &= x^3.
\end{aligned} \tag{3.67}$$

Substituting (3.67) into (3.65) and arranging terms we obtain

$$\begin{aligned}
\lambda(x) = {} & (-c_1 + 3c_2 - 3c_3 + c_4)x^3 + (3c_1h - 6c_2h + 3c_3h)x^2 \\
& + (-3c_1h^2 + 3c_3h^2)x + (c_1h^3 + 4c_2h^3 + c_3h^3).
\end{aligned} \tag{3.68}$$

Comparing equation (3.68) to equation (3.64), we get

$$\begin{aligned} -c_1 + 3c_2 - 3c_3 + c_4 &= a \\ 3hc_1 - 6hc_2 + 3hc_3 &= b \\ -3h^2c_1 + 3h^2c_3 &= c \\ h^3c_1 + 4h^3c_2 + h^3c_3 &= d, \end{aligned} \tag{3.69}$$

and we can solve the linear system of algebraic equations (3.69) to get the coefficients c_1, c_2, c_3 and c_4.

In the second interval [h,2h]

$$\lambda(x) = c_2\omega_2(x) + c_3\omega_3(x) + c_4\omega_4(x) + c_5\omega_5(x) \tag{3.70}$$

and following the same procedure as in the first interval we can find c_2, c_3, c_4 and c_5. Similarly the third interval can be used to find c_3, c_4, c_5 and c_6. However the fourth interval can be used with the first interval to get all the coefficients, and the second and the third intervals can be used as a check.

In the fourth interval [3h,4h]

$$\lambda(x) = c_4\omega_4(x) + c_5\omega_5(x) + c_6\omega_6(x) + c_7\omega_7(x) \tag{3.71}$$

where in this fourth interval (see Appendix A)

$$\begin{aligned} \omega_4(x) &= (4h - x)^3 \\ \omega_5(x) &= 4h^3 - 6h(x - 3h)^2 + 3(x - 3h)^3 \\ \omega_6(x) &= h^3 + 3h^2(x - 3h) + 3h(x - 3h)^2 - 3(x - 3h)^3 \\ \omega_7(x) &= (x - 3h)^3. \end{aligned} \tag{3.72}$$

Substituting (3.72) into (3.71) and arranging terms we obtain

$$\begin{aligned}\lambda_2(x) = {} & (-c_4 + 3c_5 - 3c_6 + c_7)x^3 \\ & + (12hc_4 - 33hc_5 + 30hc_6 - 9hc_7)x^2 \\ & + (-48h^2c_4 + 117h^2c_5 - 96h^2c_6 + 27h^2c_7)x \\ & + (64h^3c_4 - 131h^3c_5 + 100h^3c_6 - 27h^3c_7). \end{aligned} \tag{3.73}$$

Comparing equation (3.73) to equation (3.64), we get

$$\begin{aligned} -c_4 + 3c_5 - 3c_6 + c_7 &= a \\ 12hc_4 - 33hc_5 + 30hc_6 - 9hc_7 &= b \\ -48h^2c_4 + 117h^2c_5 - 96h^2c_6 + 27h^2c_7 &= c \\ 64h^3c_4 - 131h^3c_5 + 100h^3c_6 - 27h^3c_7 &= d. \end{aligned} \tag{3.74}$$

Solving the linear system (3.74) results in the coefficients c_4, c_5, c_6 and c_7. Hence equations (3.69) and (3.74) will yield the initial guess on the spline coefficients c_i $(i = 1,\ldots,7)$.

3.8 Discussion of Results

The behavior of the solution for the unconstrained problem could be attributed to factors that are characteristics of the problem at hand. First, we must note that the value of the cross-sectional area $\alpha(x)$ at the right end of the beam is zero (see equation (3.31) and the boundary condition (3.21)), and since equations (3.19) involve a division by $\alpha(x)$, there is a singularity at this right end. This singularity was handled by carrying the computations up to ninety-nine percent of the length of the beam and further it was checked by the constrained solution with a very small constraint at the right end. In fact the results shown in Figures 3.1 and 3.2 were obtained from the constrained problem with a constraint of 10^{-8} at the tip of the beam. Also, with a zero area at $x = \ell$ the deflection is infinite there, and the range of the deflection is 0 to ∞. Hence the rather pronounced oscillations in the solution. However, it must be remembered that the interval [0,1] was divided into four partitions only which results in a crude approximation to the exact solution.

The results of the constrained problem are shown in Figures 3.3 - 3.5. In Figure 3.3 are shown some representative cross-sectional area distributions for different values of the constraint on $\alpha(x)$. In Figure 3.4 the lateral deflection for the corresponding constrained beams are shown. The oscillations in the solutions, especially in the smaller constraint case might be attributed, partly to the nature of the problem itself, which was discussed previously as not having an optimal solution, thus a small constraint on α makes the beam closer to the unconstrained problem, and partly due to the fact that we are dividing the unit interval [0,1] into a very coarse mesh, four intervals only, resulting in a crude approximation. The oscillations, which were relatively small, in the case of higher constraints are mainly due to the coarse mesh used in the computations. Whatever the case might be, the problem does need more investigation in terms of error analysis and in terms of higher order conditions for the optimal solution. It must also be mentioned that using higher order cubic spline basis or higher dimension spline functions might improve on the results. (For interesting examples see 3.7.)

Now we will discuss the behavior of the natural frequency of vibrations of the beam for different cases. The natural frequency can be evaluated from equation (3.12) after computing β from the Rayleigh quotient, equation (3.22). For the unconstrained problem β has the same value as the one obtained from equation (3.14), but for the constrained problem the value of β was observed to decrease with an increase in the value of the constraint at the right end (see Table 3.1). The ratio V/V_u shown in Table 3.1 is the ratio of the volume of the sub-optimal beam to that of the uniform beam.

Beam	β	V/V_u
Unconstrained	12.3552	0.0321
Constraint = 10^{-3}	12.0403	0.0520
Constraint = 10^{-2}	11.8287	0.0956

Table 3.1

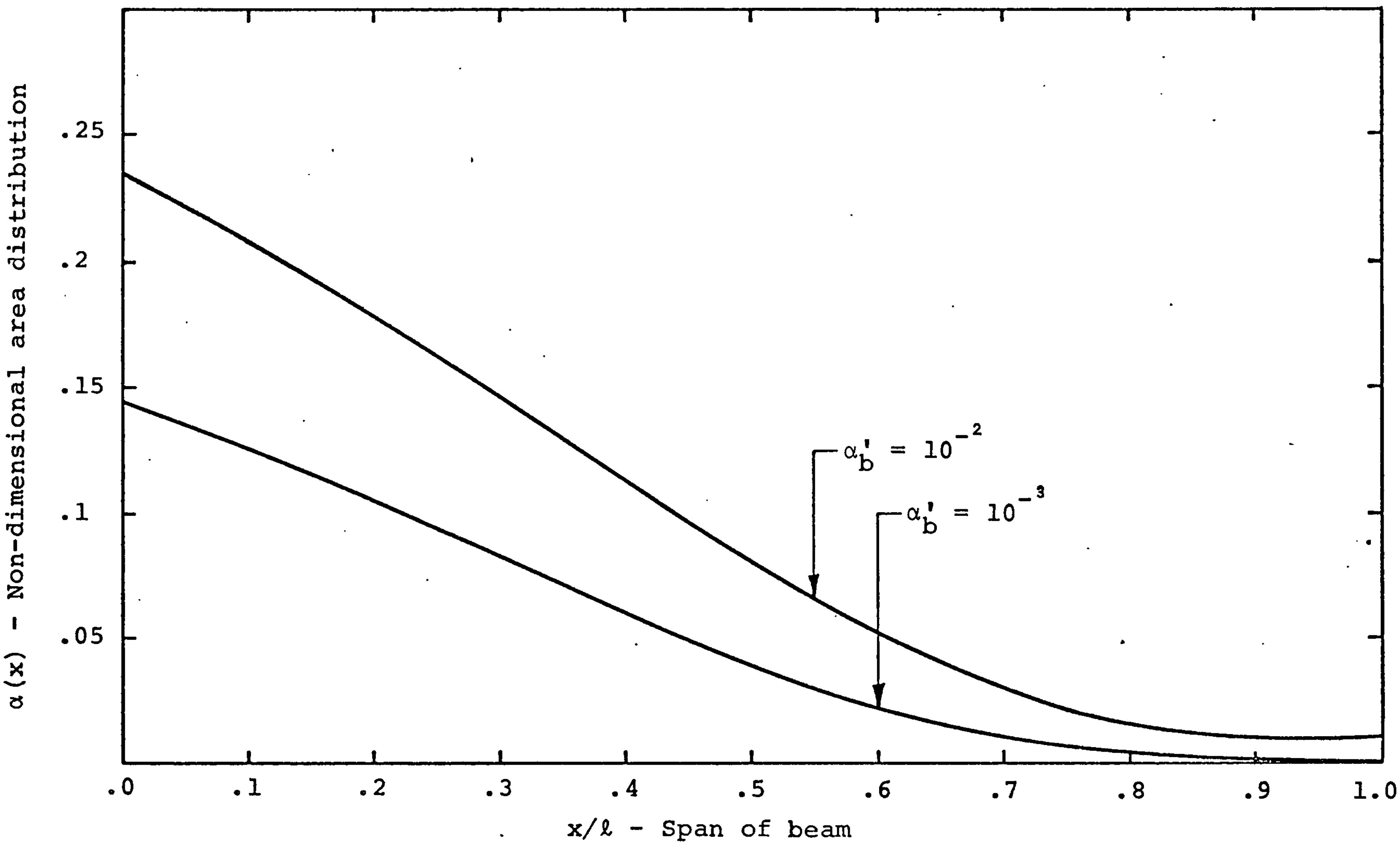

Figure 3.3

Sub-optimal Cross-sectional Area Distribution of
Vibrating Beams (Constrained)

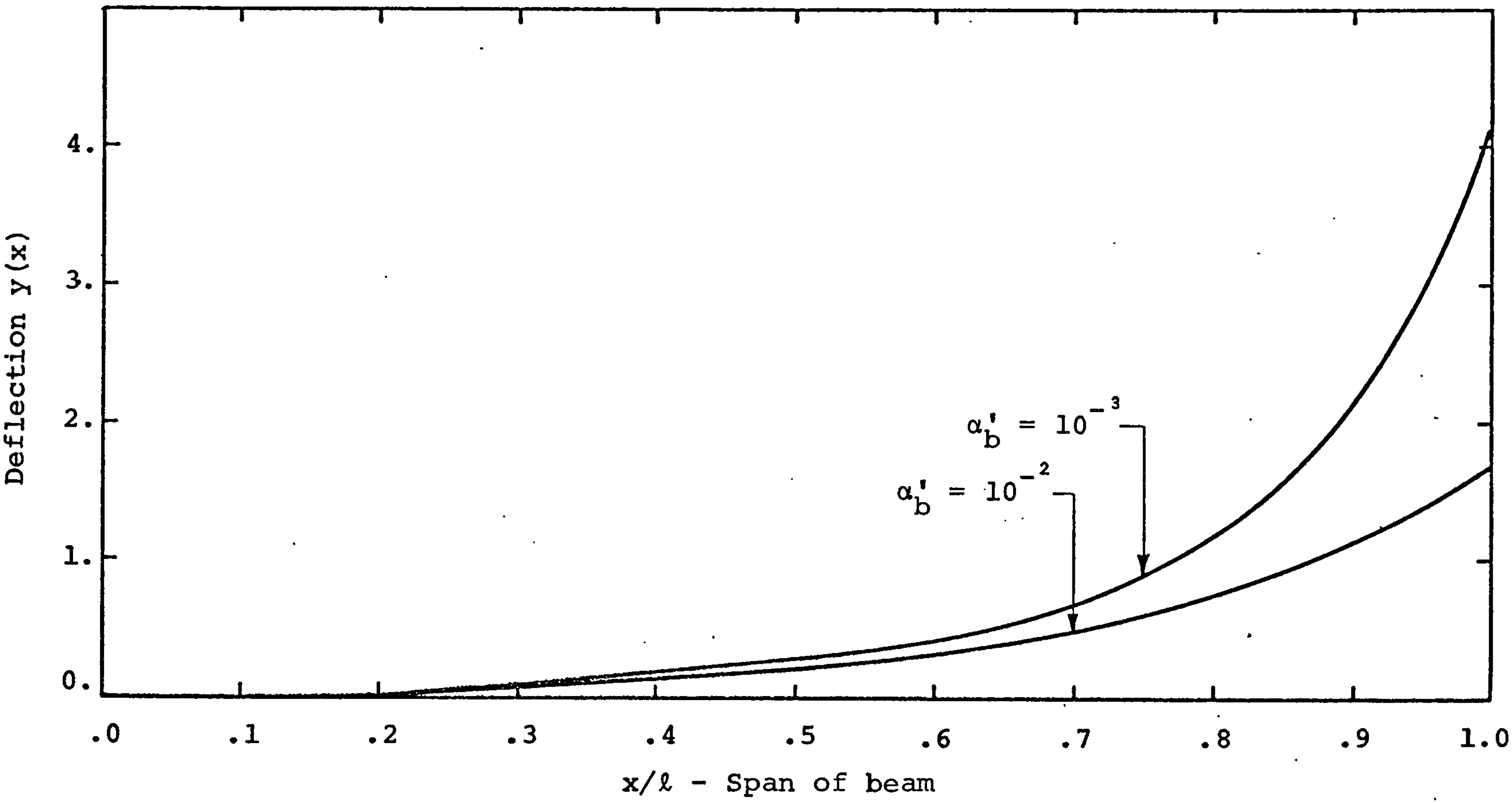

Figure 3.4

Lateral Deflection of Sub-optimal Vibrating Beams at First Fundamental Frequency (Constrained)

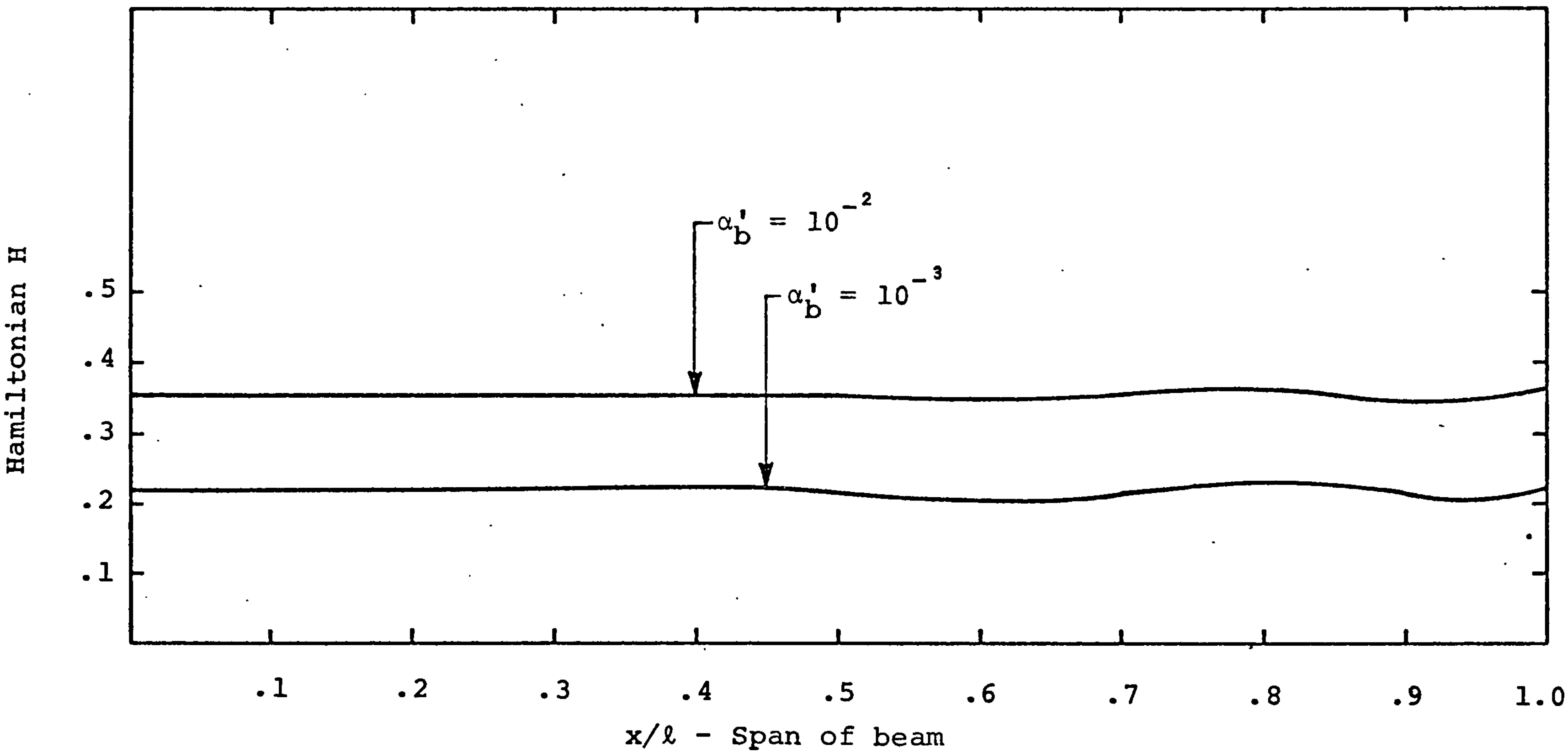

Figure 3.5

Distribution of the Hamiltonian in Sub-optimal Vibrating Beams (Constrained)

The distribution of the Hamiltonian H along the span of the beam for the constrained cases is shown in Figure 3.5. The value of H should be constant for the optimal solution as was shown in this chapter, section 3.5. As shown in Figure 3.5, the value of the Hamiltonian does not drop to zero at the right end as is the case with the unconstrained problem. However, the value of the Hamiltonian fluctuates near the right end. These fluctuations in the constrained cases are due to the fact that the necessary condition for optimality was perturbed as well as the coarse mesh used in the approximation.

CONCLUSIONS

The Ritz method has been successfully applied to optimization problems resulting in linear or non-linear algebraic equations. Approximations of the state and control variables and the associated performance measure through the use of cubic spline sub-spaces have been shown to give excellent results for problems resulting in linear algebraic equations, as well as, at least satisfactory results, for problems that would result in non-linear algebraic equations. Higher dimension cubic splines would improve on the results in the same sense as in using smaller mesh size in the method of finite differences. (For interesting examples of using different dimensions of cubic splines see Ciarlet, Schultz and Varga [1.8] and Neuman and Sen [3.7].)

One undesirable feature of the computational method used in this thesis is the lengthy computations involved in using the quasilinearization algorithm for solving non-linear algebraic equations. This algorithm requires a good deal of hand computations to find the Jacobian of the system of equations to be solved (see Appendix C). Alternative approaches, such as methods that use approximate derivatives or methods that entirely avoid all derivatives and approximations, could be used (see Daniel [1.4]). Gradient methods offer such an alternative. These methods do not require evaluating the Jacobian of the system of equations. However the Gradient methods do not have the powerful feature of quadratic convergence that the Newton-Raphson methods have.

REFERENCES

Chapter 1

1.1 Bosarge, W. E, Jr. and O. G. Johnson, "Numerical Properties of the Ritz-Trefftz Algorithm for Optimal Control," Numerical Mathematics, Communications of the ACM, 14, No. 6 (1971), 402-406.

1.2 Bosarge, W. E., Jr. and O. G. Johnson, "Error Bounds of High Order Accuracy for the State Regulator Problem Via Piecewise Polynomial Approximations," SIAM Journal of Control, 9, No. 1 (1971), 15-27.

1.3 Luenberger, D. G. Optimization by Vector Space Methods. New York: John Wiley Co., 1969.

1.4 Daniel, J. W. The Approximate Minimization of Functionals. Englewood Cliffs, New Jersey: Prentice Hall, 1971.

1.5 Mikhlin, S. G. Approximate Methods for Solutions of Differential and Integral Equations. New York: American Elsevier, 1967.

1.6 Kolomogorov, A. N. and S. V. Formin. Introductory Real Analysis. Englewood Cliffs, New Jersey: Prentice Hall, 1970.

1.7 Schoenburg, I. J., "Contribution to the Problem of Approximation of Equidistant Data by Analytic Functions, Parts A and B" Quarterly of Applied Mathematics, 4 (1946), 45-99 and 112-141.

1.8 Ciarlet, P. G., M. H. Schultz and R. S. Varga, "Numerical Methods of High Order Accuracy for Nonlinear Boundary Value Problems," Numerisch Mathematik, 9 (1967), 394-430.

1.9 Ahlberg, J. H., E. N. Nilson and J. L. Walsh. The Theory of Splines and Their Applications. New York: Academic Press, 1967.

1.10 Miele, A., S. Nagvi and A. V. Levy. "Modified Quasilinearization Method for Solving Nonlinear Equations." Houston: Rice University, Report No. 78, 1970.

Chapter 2

2.1 Vepa, K. "Optimally Stable Structural Forms." Unpublished PhD dissertation, University of Waterloo, 1972.

2.2 Tadjbaksh, I. and J. B. Keller, "Strongest Columns and Isoperimetric Inequalities for Eigenvalues," Journal of Applied Mechanics, 29 (1962), 159-164.

2.3 Shigley, J. E. Mechanical Engineering Design. 2nd ed; New York: McGraw-Hill Co., 1972.

Chapter 3

3.1 Timoshenko, S. Vibration Problems in Engineering. 3rd ed; Princeton: D. Van Nostrand, 1955.

3.2 Den Hartog, J. P. Mechanical Vibrations. 4th ed; New York: McGraw-Hill Co., 1956.

3.3 Niordson, F. I., "The Optimal Design of a Vibrating Beam," Quarterly of Applied Mathematics, 23 (1965), 47-53.

3.4 Thomson, W. T. Vibration Theory and Applications. Englewood Cliffs, New Jersey: Prentice Hall, 1965.

3.5 Vepa, K., "On the Existence of Solutions to Optimization Problems with Eigenvalue Constraints," Quarterly of Applied Mathematics, 31 (1973), 329-341.

3.6 Kirk, D. E. Optimal Control Theory. Englewood Cliffs, New Jersey: Prentice Hall, 1970.

3.7 Neuman, C. P. and A. Sen, "A Suboptimal Control Algorithm for Constrained Problems Using Cubic Splines," Automatica, 9 (1973), 601-613.

Appendix B

B.1 Berberian, S. K. Introduction to Hilbert Space. London, England: Oxford University Press, 1961.

B.2 Hille, E., and R. S. Phillips, Functional Analysis and Semigroups, Vol. 31, American Mathematical Society Colloquim Publications (New York, 1957).

B.3 Zorn, M. A., "Characterization of Analytic Functions in Banach Spaces," Annals of Mathematics, 46 (1945), 585-593.

Appendix A

CUBIC SPLINES AND THEIR BASIS

A.1 Cubic Splines

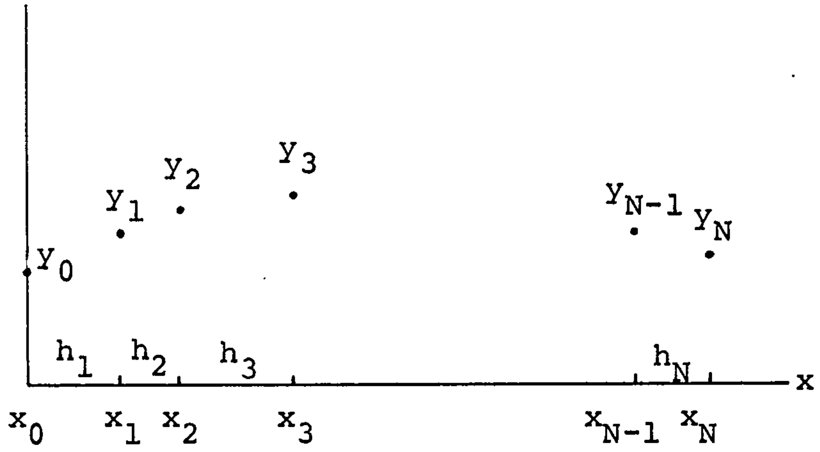

Figure A.1

Given a set of data points (x_0,y_0), $(x_1,y_1),\ldots,$ (x_N,y_N) (Figure A.1), we want to pass a piecewise cubic polynomial through these points such that the value of the functions, their first derivatives, and their second derivatives in two adjacent intervals are equal at the intersecting mesh point or "knot". That is, for two adjacent piecewise polynomials $S_\Delta(x)$, we require that [1.9]

$$\begin{aligned} S_\Delta(x_j-) &= S_\Delta(x_j+) = y_j \\ S'_\Delta(x_j-) &= S'_\Delta(x_j+) \\ S''_\Delta(x_j-) &= S''_\Delta(x_j+). \end{aligned} \tag{A.1}$$

For N intervals we need N polynomials assumed to be of the form

$$S_\Delta(x)_i = a_i x^3 + b_i x^2 + c_i x + d_i, \quad (i = 1,2,\ldots,N) \tag{A.2}$$

each to fit the two end points of each interval. If these polynomials satisfy the above requirements then they are called cubic splines. In order to define these polynomials we need to determine the unknowns a_i, b_i, c_i, d_i $(i=1,2,\ldots,N)$, hence 4N unknowns. The conditions at the knots give 2N equations to be satisfied. The first derivative conditions give N-1 equations. The second derivative conditions give N-1 equations also. Hence, a total of 2N+N-1+N-1 = 4N-2 equations. Two end conditions could be used to have two more equations such as equal moments, or equal slopes, thus 4N equations in 4N unknowns.

The set of splines that satisfy the above requirements are finite and they constitute a set of N+3 linearly independent functions in S [1.9]. If we denote these cubic piecewise polynomials by $\omega_i(x)$, $i=0,1,\ldots,N$, then

$$S(x) = \sum_{i=0}^{N} \alpha_i \omega_i(x) \tag{A.3}$$

is a linear combination of the $\omega_i(x)$, $S(x) \varepsilon S$, and $\omega_i(x)$ is a basis for S. α_i $(i=1,\ldots,N)$ is a set of vectors in R^N.

A.2 Basis of Splines

The definition of two of the spline basis will be given here although only the second type was used in obtaining results. They are the cardinal basis and the patch basis. The later type will be discussed in detail.

Let[1] $\pi: 0 = x_o < x_1 < \ldots < x_{N+1} = 1$ be a partition of $[0,1]$, then $C_i(x)$ $Sp^{(m)}(\pi)$, called the cardinal basis, is defined by (see [1.8] and [1.9])

$$\begin{aligned}
&C_i(x_j) = \delta_{i,j}, \quad 0 \leq j \leq N+1, \quad D^k C_i(0) = D^k C_i(1) = 0, \\
&\qquad\qquad 0 \leq k \leq m-1 \quad \text{for} \quad 1 \leq i \leq N \\[2ex]
&C_{N+\ell}(x_j) = 0, \quad 0 \leq j \leq N+1, \quad D^k C_{N+\ell}(0) = \delta_{\ell,k}, \\
&\qquad D^k C_{N+\ell}(1) = 0, \quad 0 \leq k \leq m-1 \quad \text{for} \quad 1 \leq \ell \leq m \\[2ex]
&C_{N+m+\ell}(x_j) = 0, \quad 0 \leq j \leq N+1, \quad D^k C_{N+m+\ell}(0) = 0, \\
&\qquad D^k C_{N+m+\ell}(1) = \delta_{\ell,k}, \quad 0 \leq k \leq m-1 \quad \text{for} \quad 1 \leq \ell \leq m.
\end{aligned} \tag{A.4}$$

Here the δ is the Kronecher delta, D is the ordinary derivative operator, N is the number of intervals in $[0,1]$ and m is the order of the spline functions, where $2m-1$ is the degree of the polynomials used in constructing the above basis. For cubic splines, $m=2$.

---

[1] We note that N given by the definition of [1.8] is equal to the number of partitions minus one.

A.3 Construction of Patch Basis

A definition of the patch basis was given in Chapter 1, page 11. Here we will give a method for constructing these basis over the interval [a,b]. For simplicity the domain will be partitioned into four intervals only, where, for reasons to become clear, four is the minimum number of partitions we can construct our basis over. We require that the basis be zero outside the boundary. In the first interval and at the left end ($x=x_0=a$) we require that the ordinate, the first derivative and the second derivative be zero. Thus since we are seeking cubic polynomials

$$\omega_i(x) = (x - x_0)^3 \qquad x \in [x_0,x_1]. \tag{A.5}$$

In the last interval we also require that at the right end, ($x=x_4=b$), the ordinate, the first derivative and the second derivative be zero. Hence

$$\omega_i(x) = (x_4 - x)^3 \qquad x \in [x_3,x_4]. \tag{A.6}$$

Now in the two middle intervals, we assume that the spline polynomials have the form

$$\omega_i(x) = a(x - x_1)^3 + b(x - x_1)^2 + c(x - x_1) + d \qquad x \in [x_1,x_2] \tag{A.7}$$

$$\omega_i(x) = e(x - x_2)^3 + f(x - x_2)^2 + g(x - x_2) + h \qquad x \in [x_2,x_3] \tag{A.8}$$

where the constants a, b, c, d, e, f, g and h are to be determined.

From the continuity requirements at x_1 and x_3 we have:

at $x = x_1$, $\omega_i(x) = (x - x_0)^3$, thus

$$\begin{aligned} \omega_i(x_1) &= (x_1 - x_0)^3 = h_1^3 \\ \omega_i'(x_1) &= 3(x_1 - x_0)^2 = 3h_1^2 \\ \omega_i''(x_1) &= 6(x_1 - x_0) = 6h_1 \end{aligned} \tag{A.9}$$

at $x = x_3$, $\omega_i(x) = (x_4 - x)^3$, thus

$$\begin{aligned} \omega_i(x_3) &= (x_4 - x_3)^3 = h_4^3 \\ \omega_i'(x_3) &= 3(x_4 - x_3)^2 = 3h_4^2 \\ \omega_i''(x_3) &= 6(x_4 - x_3) = 6h_4 \end{aligned} \tag{A.10}$$

We have six conditions where the first three conditions apply to (A.7)

$$\omega_i(x) \;, \qquad x \in [x_1, x_2]$$

and the second three conditions apply to (A.8)

$$\omega_i(x) \;, \qquad x \in [x_2, x_3].$$

Substituting these conditions in (A.7) and (A.8) results in six equations in the eight unknowns, hence we need two more equations to determine the eight constants. At $x=x_2$ we also have the continuity requirement, hence equating the ordinates, the first derivative and the second derivative of the two polynomials at the intersecting mesh point x_2 of the two intervals $[x_{i-3},x_{i-2}]$ and $[x_{i-2},x_{i-1}]$ provides three equations which any two of them can be used with the above six equations to determine the eight unknown constants a, b, c, d, e, f, g, h.

After solving the system of eight linear algebraic equations for the coefficients of the polynomials (A.7) and (A.8), we can write the patch basis over four partitions (assumed to be equal with a mesh size h) of the interval [a,b] as

$$
\begin{aligned}
\omega_i(x) &= 0 && x \notin [x_0,x_4] \\
\omega_i(x) &= (x - x_0)^3 && x \in [x_0,x_1] \\
\omega_i(x) &= h^3 + 3h^2(x - x_1) + 3h(x - x_1)^2 && \\
&\quad - 3(x - x_1) && x \in [x_1,x_2] \\
\omega_i(x) &= 4h^3 - 6h(x - x_2)^2 + 3(x - x_2)^3 && x \in [x_2,x_3] \\
\omega_i(x) &= (x_4 - x)^3 && x \in [x_3,x_4]
\end{aligned}
\tag{A.11}
$$

Now we clarify the statement that a minimum number of four intervals is required to build a patch basis. It is clear that for one or two intervals the only function that satisfies all the requirements mentioned at the beginning of this appendix is the function which is zero along the unit interval. For three intervals equations (A.5) and (A.6) are the same at the end intervals and they satisfy the requirements of a cubic spline. The cubic polynomial in the middle interval will have the general form

$$ax^3 + bx^2 + cx + b. \tag{A.12}$$

Applying conditions (A.9) and (A.10), excluding the second derivative requirement results in the quadratic polynomial

$$\omega_i(x) = -3hx^2 + 9h^2x - 5h^3. \tag{A.13}$$

This polynomial violates the second derivative condition and we can conclude that the only function which satisfies all the requirements over three intervals is the function which is zero everywhere inside the interval [a,b]. Hence four is the minimum number of intervals required to build cubic splines of patch basis type.

It was mentioned at the beginning of this Appendix that the set of splines are finite and they constitute a set of N+3 linearly independent functions in S. Thus over our four intervals we can construct only seven linearly independent set of functions that satisfy the definition of cubic splines and the patch basis. Next, we give these seven shapes, Figures A.2 - A.8, over a uniform mesh of four partitions of the interval [a,b]. Then the mesh size becomes $h=\frac{b-a}{4}$, and $x_i=ih$ for $0\leq i\leq N+1$. The equations that appear to the left of each shape is the piecewise analytic expression for that basis function.

$$\omega_1(x) = \begin{cases} (x_{i+1}-x)^3 & x\varepsilon[x_i,x_{i+1}] \\ 0 & x\notin[x_i,x_{i+1}] \end{cases}$$

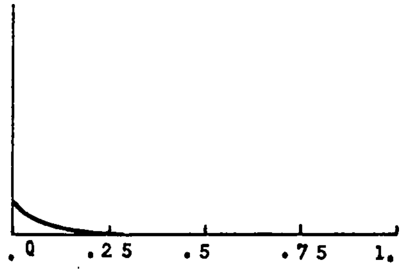

Figure A.2 $\omega_1(x)$

$$\omega_2(x) = \begin{cases} 4h^3-6h(x-x_i)^2+3(x-x_i)^3 & x\varepsilon[x_i,x_{i+1}] \\ (x_{i+2}-x)^3 & x\varepsilon[x_{i+1},x_{i+2}] \\ 0 & x\notin[x_i,x_{i+2}] \end{cases}$$

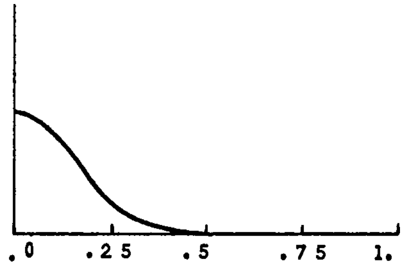

Figure A.3 $\omega_2(x)$

$$\omega_3(x) = \begin{cases} h^3+3h^2(x-x_i)+3h(x-x_i)^2 & \\ \quad -3(x-x_i)^3 & x\varepsilon[x_i,x_{i+1}] \\ 4h^3-6h(x-x_{i+1})^2+3(x-x_{i+1})^3 & x\varepsilon[x_{i+1},x_{i+2}] \\ (x_{i+3}-x)^3 & x\varepsilon[x_{i+2},x_{i+3}] \\ 0 & x\notin[x_i,x_{i+3}] \end{cases}$$

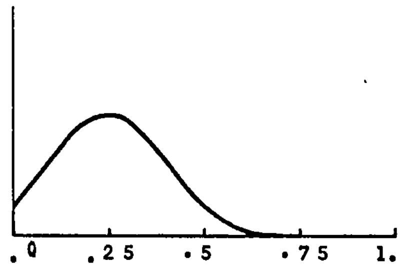

Figure A.4 $\omega_3(x)$

$$\omega_4(x) = \begin{cases} (x-x_i)^3 & x\varepsilon[x_i,x_{i+1}] \\ h^3+3h^2(x-x_{i+1})+3h(x-x_{i+1})^2 & \\ \quad -3(x-x_{i+1})^3 & x\varepsilon[x_{i+1},x_{i+2}] \\ 4h^3-6h(x-x_{i+2})^2+3(x-x_{i+2})^3 & x\varepsilon[x_{i+2},x_{i+3}] \\ (x_{i+4}-x)^3 & x\varepsilon[x_{i+3},x_{i+4}] \\ 0 & x\notin[x_i,x_{i+4}] \end{cases}$$

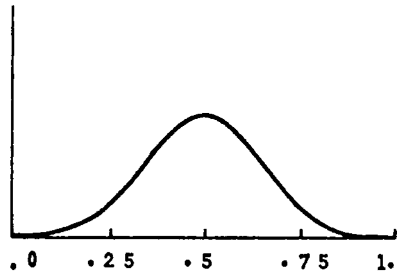

Figure A.5 $\omega_4(x)$

$$\omega_5(x) = \begin{cases} (x-x_{i+1})^3 & x\varepsilon[x_{i+1},x_{i+2}] \\ h^3+3h^2(x-x_{i+2})+3h(x-x_{i+2})^2 & \\ \quad -3(x-x_{i+2})^3 & x\varepsilon[x_{i+2},x_{i+3}] \\ 4h^3-6h(x-x_{i+3})^2+3(x-x_{i+3})^3 & x\varepsilon[x_{i+3},x_{i+4}] \\ 0 & x\notin[x_{i+1},x_{i+4}] \end{cases}$$

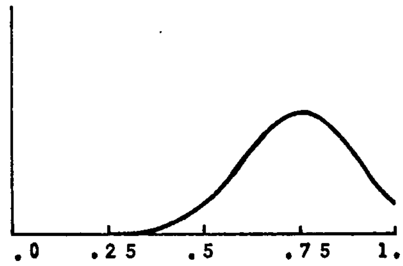

Figure A.6 $\omega_5(x)$

$$\omega_6(x) = \begin{cases} (x-x_{i+2})^3 & x\varepsilon[x_{i+2},x_{i+3}] \\ h^3+3h^2(x-x_{i+3})+3h(x-x_{i+3})^2 & \\ \quad -3(x-x_{i+3})^3 & x\varepsilon[x_{i+3},x_{i+4}] \\ 0 & x\notin[x_{i+2},x_{i+4}] \end{cases}$$

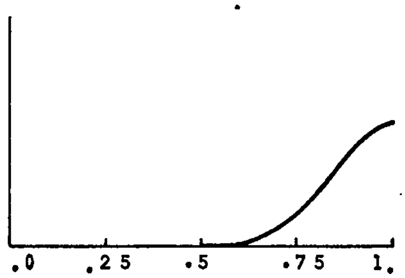

Figure A.7 $\omega_6(x)$

$$\omega_7(x) = \begin{cases} (x-x_{i+3})^3 & x\varepsilon[x_{i+3},x_{i+4}] \\ 0 & x\notin[x_{i+3},x_{i+4}] \end{cases}$$

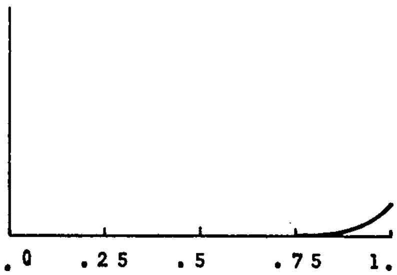

Figure A. 8 $\omega_7(x)$

The following functions, obtained as linear combinations of the above spline basis elements arise in handling boundary conditions and prove to be necessary to obtain a convenient basis for $Sp^{(2)}(\pi)$ for computations [1.9]. Their main feature is that they have non zero slope at one of their end points. These functions are defined by

$$\begin{aligned}
\bar{\omega}_1(x) &= -4\omega_1(x) + \omega_2(x) \\
\bar{\omega}_2(x) &= -\omega_1(x) + \omega_3(x) \\
\bar{\omega}_3(x) &= -4\omega_7(x) + \omega_6(x) \\
\bar{\omega}_4(x) &= -\omega_7(x) + \omega_5(x)
\end{aligned} \tag{A.21}$$

where $\omega(x)$ is as defined by (A.11), and their shapes are given in Figures A.9 - A.12.

$$\overline{\omega}_1(x) = -4\omega_1(x) + \omega_2(x)$$

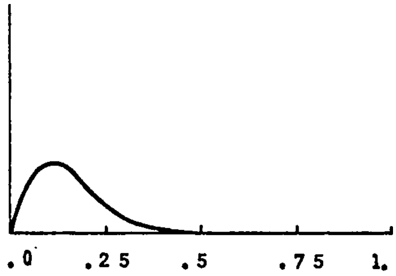

Figure A.9 $\overline{\omega}_1(x)$

$$\overline{\omega}_2(x) = -\omega_1(x) + \omega_3(x)$$

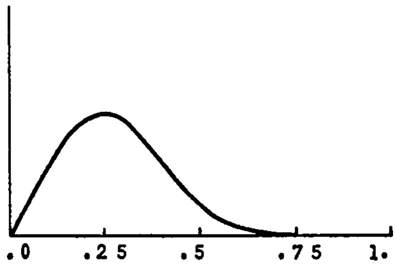

Figure A.10 $\overline{\omega}_2(x)$

$$\overline{\omega}_3(x) = -4\omega_7(x) + \omega_6(x)$$

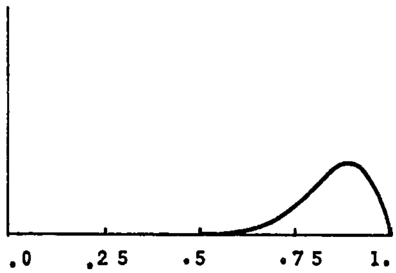

Figure A.11 $\overline{\omega}_3(x)$

$$\overline{\omega}_4(x) = -\omega_7(x) + \omega_5(x)$$

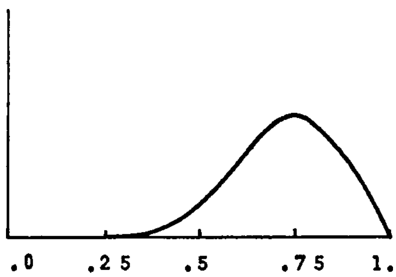

Figure A.12 $\overline{\omega}_4(x)$

Appendix B

FRÉCHET DIFFERENTIALS AND FRÉCHET DERIVATIVES

A definition of Fréchet differentials and Fréchet derivatives requires a generalization of the concept of the differential and the derivative in ordinary calculus. In the following, a definition of the Fréchet differential and the Fréchet derivative is given in terms of vector spaces without explaining all the vector space concepts involved in the definition but rather an example is presented to clarify their application. For a detailed explanation of vector spaces and discussion of Fréchet differentials and Fréchet derivatives see Luenburger [1.3], Berberian [B.1] and Daniel [1.4].

In the following let X be a vector space, Y a normed space, and T a transformation defined on a domain $D \subset X$ and having range $R \subset Y$. Luenberger [1.3] gives his definitions in terms of a general transformation T, but to get a better insight, the definition will be given in terms of some functional $J(x)$, where x is the independent variable. The functional $J(x)$ is a transformation in the sense used by Luenburger.

<u>Definition B.1.</u> If for fixed $x \in D$ and each $\delta x \in X$ there exists (i) $\delta J(x,\delta x) \in Y$ which is linear and continuous with

respect to δx such that[1]

$$\text{(ii)} \lim_{\|\delta x\| \to 0} \frac{\|J(x+\delta x) - J(x) - \delta J(x,\delta x)\|}{\|\delta x\|} = 0 \qquad \text{(B.1)}$$

then J is said to be Fréchet differentiable at x and $\delta J(x,\delta x)$ is said to be the Fréchet differential of J at x with increment δx.

In the above $\delta(.)$ denotes the first variation and $\|.\|$ denotes the norm of a vector. Next, the definition of Fréchet derivative will be given in terms of a Fréchet differentiable functional J(x).

<u>Definition B.2.</u> At a fixed point $x \in D$ the Fréchet differential $\delta J(x,\delta x)$ is, by definition, of the form $\delta J(x,\delta x) = A_x \delta x$, where A_x is a bounded linear operator from X to Y. Thus, as x varies over D, the correspondence $x \to A_x$ defines a transformation from D into the normed linear space B(X,Y); this transformation is called the Fréchet derivative J' of J. Thus we have, by definition, $\delta J(x,\delta x) = J'(x)\delta x$.

As an example for illustrating the above definitions we consider the classical problem in the calculus of variation. Find the function q on the interval $[x_0,x_f]$ such that the integral functional[2] of the form

---

[1]Hille and Phillip [B.2] point out that condition (ii) is redundant and that Zorn [B.3] has shown that condition (ii) is implied by (i).

[2]The functional J will be considered as a functional of q to be consistent with the notation of Chapter 1, where q is a function of x.

$$J(q) = \int_{x_0}^{x_f} f\ [q(x),\ q'(x),\ x]\ dx \qquad (B.2)$$

is minimized.[3] It is assumed that the integrand f has continuous first and second partial derivatives with respect to all of its arguments and that $q(x_0)$ and $q(x_f)$ are fixed. Now the fundamental theorem of the calculus of variation will be stated. [3.5].

If q* is an extremal, the variation of J must vanish on q*; that is,

$$\delta J(q^*,\delta q) = 0 \qquad (B.3)$$

for all admissible q where δJ is the variation of J.

The variation of J can be obtained as follows: Define by $\Delta J(q,\delta q)$ the increment of J(q) due to a perturbation δq, then $\Delta J(q,\delta q)$ can be written as

$$\Delta J(q,\delta q) = \int_{x_0}^{x_f} \left\{ f\ [q + \delta q, q' + \delta q', x] - f\ [q,q',x] \right\} dx. \qquad (B.4)$$

Expanding (B.4) in Taylor series about the point q(x),q'(x) and retaining only linear terms (see Kirk [3.6]) we obtain the variation

$$\delta J(q,\delta q) = \int_{x_0}^{x_f} \left\{ \frac{\partial f}{\partial q}\ [q,q',x]\ \delta q + \frac{\partial f}{\partial q'}\ [q,q',x]\ \delta q' \right\} dx. \qquad (B.5)$$

This is the form obtained in Luenburger [1.3] which he concludes to be Fréchet differentiable.

Now to express $\delta J(q,\delta q)$ entirely in terms containing δq, we integrate by parts the term involving δq' to obtain

---

[3]The prime (') above the q denotes the derivative of q with respect to x, i.e. $\frac{dq}{dx}$.

$$\delta J(q,\delta q) = \left[\frac{\partial f}{\partial q}(q,q',x) \right] \delta q \Big|_{x_0}^{x_f} + \int_{x_0}^{x_f} \left[\frac{\partial f}{\partial q}(q,q',x) \right.$$

$$\left. - \frac{d}{dx} \left(\frac{\partial f}{\partial q'}(q,q',x) \right) \right] \delta q \; dx. \tag{B.6}$$

Since $q(x_0)$ and $q(x_f)$ are specified, then $\delta q(x_0) = 0$, $\delta q(x_f) = 0$ and terms outside the integral vanish. Hence equation (B.6) can be written as

$$\delta J(q,\delta q) = \int_{x_0}^{x_f} \left[\frac{\partial f}{\partial q}(q,q',x) - \frac{d}{dx}\left(\frac{\partial f}{\partial q}(q,q',x) \right) \right] \delta q \; dx. \tag{B.7}$$

At this point we note that equation (B.7) is in the form of definition B.2, that is in the form

$$\delta J(q,\delta q) = J'(q)\delta q \tag{B.8}$$

where $J'(q)$ was defined to be Fréchet derivative. Therefore, by comparing equation (B.7) with (B.8) we can deduce that the Fréchet derivative of the functional $J(q)$ is

$$J'(q) = \int_{x_0}^{x_f} \left[\frac{\partial f}{\partial q}(q,q',x) - \frac{d}{dx}\left(\frac{\partial f}{\partial q}(q,q',x) \right) \right] dx. \tag{B.9}$$

It must be mentioned that carrying the derivation for $\delta J(q,\delta q)$ further we can arrive at Euler-Lagrange equation. Applying the fundamental theorem of the calculus of variation, equation (B.3), we obtain from (B.7)

$$\delta J(q,\delta q) = \int_{x_0}^{x_f} \left[\frac{\partial f}{\partial q}(q,q',x) - \frac{d}{dx}\left(\frac{\partial f}{\partial q}(q,q',x) \right) \right] \delta q \; dx = 0. \tag{B.10}$$

Since the term multiplying δq in equation (B.10) is continuous it follows, from the fundamental lemma of the calculus of variation (Kirk [3.6]) that it must vanish identically on $[x_0, x_f]$. Thus we conclude that the extremal q^* must satisfy the equation

$$\frac{\partial f}{\partial q}(q,q',x) - \frac{d}{dx}\left(\frac{\partial f}{\partial q'}(q,q',x)\right) = 0. \tag{B.11}$$

This is the Euler-Lagrange equation which serves as the necessary condition for q^* to be an extremal.

Now the Fréchet derivatives used in obtaining the optimality conditions (1.6) in Chapter 1 will be discussed. We note that in this appendix the integral functional considered is of the form

$$J(q) = \int_{x_0}^{x_f} f(q,q',x) \quad dx$$

and to arrive at equation (B.6) we integrated by parts equation (B.5) thus expressing the variation of J in terms of δq. But, in Chapter 1, although we start with an integral functional[4] (1.4) of the same form as (B.2), we transform this functional to a form that does not contain q'. Hence by applying the derivation used in this appendix to the problem in Chapter 1, noting that $\frac{\partial L}{\partial q'} = 0$ because L does not contain q' (see equation (1.9)), we arrive at the result

$$\delta L(q,\delta q) = \int_{x_0}^{x_f} \frac{\partial L}{\partial q}\,\delta q\; dx = 0.$$

---

[4]The functional in Chapter 1 is the Lagrangian.

Hence, applying the fundamental lemma of the calculus of variation, we get

$$\frac{\partial L}{\partial q} = 0. \tag{B.12}$$

Similarly, since L is a function of the control u, and not a function of u', it follows that

$$\frac{\partial L}{\partial u} = 0. \tag{B.13}$$

These are the equations that served as necessary conditions to arrive at the optimal solution in the preceding chapters.

Appendix C

THE JACOBIAN

The Jacobian of the system of equations (3.53), (3.55) and (3.60) - (3.62) is given in this appendix without any simplifications because it is this form that was used in the program. It is given for the constrained case because this case is more general than the unconstrained one. The Jacobian will be of dimension (24 x 24) corresponding to the system of twenty-four equations in the twenty-four unknowns. Equations (3.53) will be denoted by g_i (i = 1,...,6), equations (3.55) by g_i (i = 7,...,12) equations (3.62) by g_i (i = 13,...,18), and equations (3.60) by g_i (i = 19,...,24).

$$\frac{\partial g_i}{\partial c_j} = \frac{\partial g_i}{\partial d_j} = 0 \qquad (i = 1,\ldots,6, \quad j = 1,\ldots,6)$$

$$\frac{\partial g_i}{\partial e_j} = \frac{\partial g_i}{\partial f_j} = 0 \qquad (i = 7,\ldots,12, \quad j = 2,\ldots,7)$$

$$\frac{\partial g_i}{\partial c_j} = 0 \qquad (i = 13,\ldots,18, \; j = 2,\ldots,7)$$

$$\frac{\partial g_i}{\partial c_j} = 0 \qquad (i = 19,\ldots,24, \; j = 1,\ldots,6)$$

$$\frac{\partial g_i}{\partial e_j} = 2\int_0^1 \omega_i \omega_j \, dx = a_{ij} \qquad \omega_i = \omega_i \qquad (i = 1,\ldots,4)$$

$$\omega_i = \bar{\omega}_{9-i} \qquad (i = 5,6)$$

$$(i = 1,\ldots,6,\ j = 1,\ldots,6)$$

$$\omega_j = \bar{\omega}_j \qquad (j = 1,2)$$

$$\omega_j = \omega_{j+1} \qquad (j = 3,\ldots,6)$$

$$\frac{\partial g_i}{\partial f_j} = \int_0^1 (\omega_i' \omega_j - \omega_i \omega_j') \, dx = b_{ij}$$

$$(i = 1,\ldots,6,\ j = 1,\ldots,6)$$

where ω_i and ω_j are as defined above and the prime (') denotes a derivative with respect to x.

$$\frac{\partial g_i}{\partial c_j} = a^T_{i,j-6} \qquad (i = 7,\ldots,12,\quad j = 1,\ldots,6)$$

$$\frac{\partial g_i}{\partial d_j} = -b^T_{i-6,j} \qquad (i = 7,\ldots,12,\quad j = 1,\ldots,6)$$

$$\frac{\partial g_i}{\partial c_j} = b^T_{i-12,j} \qquad (i = 13,\ldots,18,\ j = 1,\ldots,6)$$

$$\frac{\partial g_i}{\partial e_j} = -b_{i-18,j} \qquad (i = 19,\ldots,24,\ j = 1,\ldots,6)$$

where (T) denotes the transpose of a matrix.

$$\frac{\partial g_i}{\partial d_j} = \int_0^1 \Big\{ \frac{4}{3}\beta \lambda_4^{i-12} \left(1+\beta \lambda_4^2\right)^{-\frac{1}{3}} \Big[-\frac{2}{3}\left[2\lambda_2^2 + \alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{5}{3}} \left(4\lambda_2 \lambda_2^j\right)$$

$$\left[3\lambda_2^2 \lambda_4 + \alpha_b\left(1+\beta\lambda_4^2\right)\lambda_4\right] + \left[2\lambda_2^2 + \alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{2}{3}}$$

$$\left(6\lambda_2 \lambda_4 \lambda_2^j\right)\Big] + \left(\frac{2}{3}\alpha_b^2 \beta \lambda_4^{i-12}\right)\left(1+\beta\lambda_4^2\right)^{\frac{5}{3}} \left(\lambda_4\right)$$

$$\left[-\frac{5}{3}\left[2\lambda_2^2 + \alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{8}{3}} \left(4\lambda_2 \lambda_2^j\right)\right] \Big\} dx$$

$$(i = 13, \ldots\ldots, 18,\ j = 1, \ldots\ldots, 6)$$

$$\lambda_4^i = \bar{\omega}_{i-12} \quad (i = 13, 14)$$

$$\lambda_4^i = \omega_{i-11} \quad (i = 15, \ldots, 18)$$

$$\lambda_2^j = \omega_j \quad (j = 1, \ldots, 4)$$

$$\lambda_2^j = \bar{\omega}_{9-j} \quad (j = 5, 6)$$

$$\frac{\partial g_i}{\partial f_i} = \int_0^1 \left\{ \frac{4}{3}\beta\lambda_4^i \left[-\frac{1}{3}\left(1+\beta\lambda_4^2\right)^{-\frac{4}{3}} \left(2\beta\lambda_4\lambda_4^j\right) \left[2\lambda_2^2+\alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{2}{3}} \right.\right.$$

$$\left[3\lambda_2^2\lambda_4+\alpha_b\lambda_4\left(1+\beta\lambda_4^2\right)\right] + \left(1+\beta\lambda_4^2\right)^{-\frac{1}{3}}$$

$$\left[-\frac{2}{3}\left[2\lambda_2^2+\alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{5}{3}} \left(2\alpha_b\beta\lambda_4\lambda_4^j\right)\right.$$

$$\left.\left[3\lambda_2^2\lambda_4+\alpha_b\lambda_4\left(1+\beta\lambda_4^2\right)\right]\right] + \left(1+\beta\lambda_4^2\right)^{-\frac{1}{3}}$$

$$\left.\left[2\lambda_2^2+\alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{2}{3}} \left[3\lambda_2^2\lambda_4^j+\alpha_b\lambda_4^j\left(1+3\beta\lambda_4^2\right)\right]\right]$$

$$+\left(\frac{2}{3}\alpha_b^2\beta\lambda_4^i\right)\left[\frac{5}{3}\left(1+\beta\lambda_4^2\right)^{\frac{2}{3}}\left(2\beta\lambda_4\lambda_4^j\right)\left(\lambda_4\right)\right.$$

$$\left[2\lambda_2^2+\alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{5}{3}} + \left(1+\beta\lambda_4^2\right)^{\frac{5}{3}}\left(\lambda_4^j\right)\left[2\lambda_2^2+\alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{5}{3}}$$

$$\left.\left.+\left(1+\beta\lambda_4^2\right)^{\frac{5}{3}}\left(\lambda_4\right)\left[-\frac{5}{3}\left[2\lambda_2^2+\alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{8}{3}}\left[2\alpha_b\beta\lambda_4\lambda_4^j\right]\right]\right] \right\} dx$$

$$\left(i=13,\ldots,18,\ j=1,\ldots,6\right) \qquad \begin{aligned} \lambda_4^i &= \bar{\omega}_{i-12}\ (i=13,14) \\ \lambda_4^i &= \omega_{i-11}\ (i=15,\ldots,18) \\ \lambda_4^j &= \bar{\omega}_j\ \ (j=1,2) \\ \lambda_4^j &= \omega_{j+1}\ (j=3,\ldots,6) \end{aligned}$$

$$\frac{\partial g_i}{\partial \alpha_i} = \int_0^1 2\lambda_2^{i-18}\left(1+\beta\lambda_4^2\right)^{\frac{2}{3}}\left\{-\frac{5}{3}\left[2\lambda_2^2+\alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{8}{3}}\left(4\lambda_2\lambda_2^j\right)\right.$$

$$\left[2\lambda_2^3+\frac{5}{3}\alpha_b\lambda_2\left(1+\beta\lambda_4^2\right)\right]+\left[2\lambda_2^2+\alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{5}{3}}$$

$$\left.\left[6\lambda_2^2\lambda_2^j+\frac{5}{3}\alpha_b\left(1+\beta\lambda_4^2\right)\lambda_2^j\right]\right\}dx$$

$$(i=19,\cdots,24,\ j=1,\cdots,6)\qquad \begin{aligned}&\lambda_2^i=\omega_{i-18}\ (i=19,\cdots,22)\\&\lambda_2^i=\bar{\omega}_{27-i}\ (i=23,24)\\&\lambda_2^j=\omega_j\quad (j=1,\cdots,4)\\&\lambda_2^j=\bar{\omega}_{9-j}\ (j=5,6)\end{aligned}$$

$$\frac{\partial g_i}{\partial \beta_i} = \int_0^1 2\lambda_2^{i-18}\left\{\frac{2}{3}\left(1+\beta\lambda_4^2\right)^{-\frac{1}{3}}\left(2\beta\lambda_4\lambda_4^j\right)\left[\left[2\lambda_2^2+\alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{5}{3}}\right.\right.$$

$$\left.\left[2\lambda_2^3+\frac{5}{3}\alpha_b\lambda_2\left(1+\beta\lambda_4^2\right)\right]\right]+\left(1+\beta\lambda_4^2\right)^{\frac{2}{3}}\left[-\frac{5}{3}\left[2\lambda_2^2+\alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{8}{3}}\right.$$

$$\left[2\alpha_b\beta\lambda_4\lambda_4^j\right]\left[2\lambda_2^3+\frac{5}{3}\alpha_b\lambda_2\left(1+\beta\lambda_4^2\right)\right]+\left[2\lambda_2^2+\alpha_b\left(1+\beta\lambda_4^2\right)\right]^{-\frac{5}{3}}$$

$$\left.\left.\left[\frac{5}{3}\alpha_b\lambda_2\left(2\beta\lambda_4\lambda_4^j\right)\right]\right]\right\}dx$$

$$(i=19,\cdots,24,\ j=1,\cdots,6)\qquad \begin{aligned}&\lambda_2^i=\omega_{i-18}\ (i=19,\cdots,22)\\&\lambda_2^i=\bar{\omega}_{27-i}\ (i=23,24)\\&\lambda_4^j=\bar{\omega}_j\quad (j=1,2)\\&\lambda_4^j=\omega_{j+1}\ (j=3,\cdots,6)\end{aligned}$$

Appendix D

PROGRAMS AND OUTPUTS

In this appendix a list of the programs used in arriving at results is given together with an output of the solutions. The first program is for the solution of the minimum weight column of length ℓ subjected to an axial load P. The second program is for the solution of the minimum weight beam in transverse vibrations at a certain natural frequency. This program listed is for the constrained case but by making the constraint very small in the order of 10^{-8} (but not zero since the algorithm then involves a division by zero) one could reach the solution to the unconstrained problem. The programs have subroutines to construct the spline basis elements over four partitions of the interval $[0,\ell]$ and $[0,1]$. The programs are written in double precision and they both use subroutines from the SSP (Scientific Subroutines Package written by IBM). The SSP subroutines are not listed in this appendix and they are

DQSF - This subroutine computes the vector of integral value for a given equidistant table of function values. The method used is a combination of Simpson rule together with Newton's $\frac{3}{8}$ rule or a combination of these two rules. All integrations were carried out with a stepsize of integration of 0.01 for the unit interval [0,1]

DGELG - This subroutine solves a general system of simultaneous linear equations by means of the Gauss elimination method.

```
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION G(12,12),W(7,101),DW(7,101),WH(4,101),DWH(4,101),
     1X(101),C(12),ALPHA(101),Y(101),Z(101),DLAMD1(101),DLAMD2(101)
      REAL*8 LAMDA1(101),LAMDA2(101),LENGTH,LOAD
    1 READ(5,10,END=1000)NINTV,DELT,LENGTH,LOAD
   10 FORMAT(I2,3F20.15)
      WRITE(6,15)NINTV,DELT
   15 FORMAT('1','     NO. OF SPLINE INTERVALS:',I2,', STEPSIZE DELT=',
     1F8.4/)
      WRITE(6,16)LENGTH,LOAD
   16 FORMAT('      LENGTH OF COLUMN=',F10.4,',LOAD =',F10.4,//)
      X(1)=0.0
      NPOINT=LENGTH/DELT+1
      DO 25 K=2,NPOINT
      X(K)=X(K-1)+DELT
   25 CONTINUE
      NSTEP=NPOINT-1
      LINTV=NSTEP/NINTV
      H=LENGTH/NINTV
      CALL SPLIN(X,H,W,WH,LINTV)
      CALL DSPLIN(X,H,DW,DWH,LINTV)
      DO 30 I=7,12
      DO 30 J=7,12
      G(I,J)=0.D0
   30 CONTINUE
      DO 50 I=1,2
      DO 40 J=I,2
      DO 35 K=1,NPOINT
      Y(K)=WH(I,K)*WH(J,K)
   35 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      G(I,J)=2.D0*Z(NPOINT)
   40 CONTINUE
      DO 50 J=3,6
      DO 45 K=1,NPOINT
      Y(K)=WH(I,K)*W(J+1,K)
   45 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      G(I,J)=2.D0*Z(NPOINT)
   50 CONTINUE
      DO 60 I=3,6
      DO 60 J=I,6
      DO 55 K=1,NPOINT
      Y(K)=W(I+1,K)*W(J+1,K)
   55 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      G(I,J)=2.D0*Z(NPOINT)
   60 CONTINUE
      DO 80 I=1,2
      DO 70 J=7,10
      DO 65 K=1,NPOINT
      Y(K)=WH(I,K)*DW(J-6,K)-DWH(I,K)*W(J-6,K)
   65 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
```

```
      G(I,J)=Z(NPOINT)
   70 CONTINUE
      DO 80 J=11,12
      DO 75 K=1,NPOINT
      Y(K)=WH(I,K)*DWH(15-J,K)-DWH(I,K)*WH(15-J,K)
   75 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      G(I,J)=Z(NPOINT)
   80 CONTINUE
      DO 100 I=3,6
      DO 90 J=7,10
      DO 85 K=1,NPOINT
      Y(K)=W(I+1,K)*DW(J-6,K)-DW(I+1,K)*W(J-6,K)
   85 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      G(I,J)=Z(NPOINT)
   90 CONTINUE
      DO 100 J=11,12
      DO 95 K=1,NPOINT
      Y(K)=W(I+1,K)*DWH(15-J,K)-DW(I+1,K)*WH(15-J,K)
   95 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      G(I,J)=Z(NPOINT)
  100 CONTINUE
      PP=DSQRT(LOAD)
      DO 110 I=7,10
      DO 105 K=1,NPOINT
      Y(K)=W(I-6,K)
  105 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      C(I)=-2.D0*PP*Z(NPOINT)
  110 CONTINUE
      DO 120 I=11,12
      DO 115 K=1,NPOINT
      Y(K)=WH(15-I,K)
  115 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      C(I)=-2.D0*PP*Z(NPOINT)
  120 CONTINUE
      DO 125 I=1,6
      C(I)=0.0
  125 CONTINUE
      DO 130 I=2,6
      DO 130 J=1,I
      G(I,J)=G(J,I)
  130 CONTINUE
      DO 135 I=7,12
      DO 135 J=1,6
      G(I,J)=G(J,I)
  135 CONTINUE
      EPS=1.0D-16
      CALL DGELG(C,G,12,1,EPS,IER)
      C1=-4.D0*C(1)-C(2)
      D7=-C(11)-4.D0*C(12)
```

```
      WRITE(6,140)
  140 FORMAT('    COEFF. OF SPLINES FUNCTIONS C(I)(I=1,....,12) ARE')
      WRITE(6,145)C1,(C(I),I=1,12),D7
  145 FORMAT(' ',7F10.4)
      WRITE(6,150)
  150 FORMAT('0','    SPAN','                              AREA','                          AREA')
      WRITE(6,155)
  155 FORMAT('          OF','                  DISTRIBUTION','          DISTRIBUTION'
     1)
      WRITE(6,160)
  160 FORMAT('    COLUMN','              ( NUMERICAL)','              (ANALYTICAL)'
     1)
      DO 175 K=1,NPOINT
      A1=0.0
      A2=0.0
      AD1=0.0
      AD2=0.0
      DO 170 I=7,12
      A2=A2+C(I)*W(I-6,K)
      A1=A1+C(I-6)*W(I-5,K)
      AD1=AD1+C(I-6)*DW(I-5,K)
      AD2=AD2+C(I)*DW(I-6,K)
  170 CONTINUE
      LAMDA2(K)=A2+D7*W(7,K)
      LAMDA1(K)=A1+C1*W(1,K)
      DLAMD2(K)=AD2+D7*DW(7,K)
      DLAMD1(K)=AD1+C1*DW(1,K)
      ALPHA(K)=PP*LAMDA2(K)
      Y(K)=LOAD*(LENGTH**2-X(K)**2)/2.D0
  175 CONTINUE
      DO 180 K=1,NPOINT,2
      WRITE(6,185)X(K),ALPHA(K),Y(K)
  180 CONTINUE
  185 FORMAT(F10.4,6E20.5)
      DO 190 K=1,NPOINT
      Y(K)=LAMDA1(K)*LAMDA1(K)-DLAMD1(K)*LAMDA2(K)+DLAMD2(K)*LAMDA1(K)
     1+2.D0*PP*LAMDA2(K)
  190 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      ALAG=Z(NPOINT)
      CALL DQSF(DELT,ALPHA,Z,NPOINT)
      BLAG=Z(NPOINT)
      WRITE(6,195)BLAG
  195 FORMAT('    COST FUNCTIONAL=',E20.15)
      WRITE(6,200)ALAG
  200 FORMAT('               LAGRANGIAN=',E20.15)
      GO TO 1
 1000 STOP
      END
```

```
      SUBROUTINE SPLIN(X,H,W,WH,LINTV)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION W(7,101),X(101),WH(4,101)
      DO 5 I=1,7
      DO 5 K=1,101
      W(I,K)=0.0
    5 CONTINUE
      DO 10 I=1,4
      DO 10 K=1,101
      WH(I,K)=0.0
   10 CONTINUE
      NN=LINTV+1
      DO 15 I=1,NN
      W(1,I)=(X(LINTV+1)-X(I))**3
      W(2,I)=4.D0*H**3-6.D0*H*(X(I)-X(1))**2+3.D0*(X(I)-X(1))**3
      W(3,I)=H**3+3.D0*H**2*(X(I)-X(1))+3.D0*H*(X(I)-X(1))**2
     1-3.D0*(X(I)-X(1))**3
      W(4,I)=(X(I)-X(1))**3
      WH(1,I)=-4.D0*W(1,I)+W(2,I)
      WH(2,I)=-W(1,I)+W(3,I)
   15 CONTINUE
      MM=LINTV+2
      MMM=2*LINTV+1
      DO 20 I=MM,MMM
      KK=I-LINTV
      W(2,I)=W(1,KK)
      W(3,I)=W(2,KK)
      W(4,I)=W(3,KK)
      W(5,I)=W(4,KK)
      WH(1,I)=-4.D0*W(1,I)+W(2,I)
      WH(2,I)=-W(1,I)+W(3,I)
      WH(4,I)=-W(7,I)+W(5,I)
   20 CONTINUE
      LL=2*LINTV+2
      LLL=3*LINTV+1
      DO 25 I=LL,LLL
      KK=I-LINTV
      W(3,I)=W(2,KK)
      W(4,I)=W(3,KK)
      W(5,I)=W(4,KK)
      W(6,I)=W(5,KK)
      WH(2,I)=-W(1,I)+W(3,I)
      WH(3,I)=-4.D0*W(7,I)+W(6,I)
      WH(4,I)=-W(7,I)+W(5,I)
   25 CONTINUE
      II=3*LINTV+2
      III=4*LINTV+1
      DO 30 I=II,III
      KK=I-LINTV
      W(4,I)=W(3,KK)
      W(5,I)=W(4,KK)
      W(6,I)=W(5,KK)
      W(7,I)=W(6,KK)
      WH(3,I)=-4.D0*W(7,I)+W(6,I)
```

```
      WH(4,I)=-W(7,I)+W(5,I)
   30 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DSPLIN(X,H,DW,DWH,LINTV)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION DW(7,101),X(101),DWH(4,101)
      DO 5 I=1,7
      DO 5 K=1,101
      DW(I,K)=0.0
    5 CONTINUE
      DO 10 I=1,4
      DO 10 K=1,101
      DWH(I,K)=0.0
   10 CONTINUE
      NN=LINTV+1
      DO 15 I=1,NN
      DW(1,I)=-3.DO*(X(LINTV+1)-X(I))*(X(LINTV+1)-X(I))
      DW(2,I)=-12.DO*H*(X(I)-X(1))+9.DO*(X(I)-X(1))*(X(I)-X(1))
      DW(3,I)=3.DO*H*H+6.DO*H*(X(I)-X(1))-9.DO*(X(I)-X(1))*(X(I)-X(1))
      DW(4,I)=3.DO*(X(I)-X(1))*(X(I)-X(1))
      DWH(1,I)=-4.DO*DW(1,I)+DW(2,I)
      DWH(2,I)=-DW(1,I)+DW(3,I)
   15 CONTINUE
      MM=LINTV+2
      MMM=2*LINTV+1
      DO 20 I=MM,MMM
      KK=I-LINTV
      DW(2,I)=DW(1,KK)
      DW(3,I)=DW(2,KK)
      DW(4,I)=DW(3,KK)
      DW(5,I)=DW(4,KK)
      DWH(1,I)=-4.DO*DW(1,I)+DW(2,I)
      DWH(2,I)=-DW(1,I)+DW(3,I)
      DWH(4,I)=-DW(7,I)+DW(5,I)
   20 CONTINUE
      LL=2*LINTV+2
      LLL=3*LINTV+1
      DO 25 I=LL,LLL
      KK=I-LINTV
      DW(3,I)=DW(2,KK)
      DW(4,I)=DW(3,KK)
      DW(5,I)=DW(4,KK)
      DW(6,I)=DW(5,KK)
      DWH(2,I)=-DW(1,I)+DW(3,I)
      DWH(3,I)=-4.DO*DW(7,I)+DW(6,I)
      DWH(4,I)=-DW(7,I)+DW(5,I)
   25 CONTINUE
      II=3*LINTV+2
      III=4*LINTV+1
      DO 30 I=II,III
      KK=I-LINTV
      DW(4,I)=DW(3,KK)
      DW(5,I)=DW(4,KK)
      DW(6,I)=DW(5,KK)
      DW(7,I)=DW(6,KK)
      DWH(3,I)=-4.DO*DW(7,I)+DW(6,I)
      DWH(4,I)=-DW(7,I)+DW(5,I)
```

```
 30  CONTINUE
     RETURN
     END
```

NO. OF SPLINE INTERVALS: 4, STEPSIZE DELT= 0.0100

LENGTH OF COLUMN= 1.0000, LOAD = 1.0000

COEFF. OF SPLINES FUNCTIONS C(I)(I=1,....,14) ARE

-2.6667	0.0000	2.6667	5.3334	8.0000	10.6667	13.3330
5.1109	5.4445	5.1111	4.1111	2.4444	0.1111	-2.8889

SPAN OF COLUMN	AREA DISTRIBUTION (NUMERICAL)	AREA DISTRIBUTION (ANALYTICAL)
0.0	0.50000D 00	0.50000D 00
0.0200	0.49980D 00	0.49980D 00
0.0400	0.49920D 00	0.49920D 00
0.0600	0.49820D 00	0.49820D 00
0.0800	0.49680D 00	0.49680D 00
0.1000	0.49500D 00	0.49500D 00
0.1200	0.49280D 00	0.49280D 00
0.1400	0.49020D 00	0.49020D 00
0.1600	0.48720D 00	0.48720D 00
0.1800	0.48380D 00	0.48380D 00
0.2000	0.48000D 00	0.48000D 00
0.2200	0.47580D 00	0.47580D 00
0.2400	0.47120D 00	0.47120D 00
0.2600	0.46620D 00	0.46620D 00
0.2800	0.46080D 00	0.46080D 00
0.3000	0.45500D 00	0.45500D 00
0.3200	0.44880D 00	0.44880D 00
0.3400	0.44220D 00	0.44220D 00
0.3600	0.43520D 00	0.43520D 00
0.3800	0.42780D 00	0.42780D 00
0.4000	0.42000D 00	0.42000D 00
0.4200	0.41180D 00	0.41180D 00
0.4400	0.40320D 00	0.40320D 00
0.4600	0.39420D 00	0.39420D 00
0.4800	0.38480D 00	0.38480D 00
0.5000	0.37500D 00	0.37500D 00
0.5200	0.36480D 00	0.36480D 00
0.5400	0.35420D 00	0.35420D 00
0.5600	0.34320D 00	0.34320D 00
0.5800	0.33180D 00	0.33180D 00
0.6000	0.32000D 00	0.32000D 00
0.6200	0.30780D 00	0.30780D 00
0.6400	0.29520D 00	0.29520D 00
0.6600	0.28220D 00	0.28220D 00
0.6800	0.26880D 00	0.26880D 00
0.7000	0.25500D 00	0.25500D 00
0.7200	0.24080D 00	0.24080D 00
0.7400	0.22620D 00	0.22620D 00
0.7600	0.21120D 00	0.21120D 00
0.7800	0.19580D 00	0.19580D 00
0.8000	0.18000D 00	0.18000D 00
0.8200	0.16380D 00	0.16380D 00
0.8400	0.14720D 00	0.14720D 00
0.8600	0.13020D 00	0.13020D 00
0.8800	0.11280D 00	0.11280D 00
0.9000	0.95000D-01	0.95000D-01
0.9200	0.76800D-01	0.76800D-01
0.9400	0.58200D-01	0.58200D-01
0.9600	0.39200D-01	0.39200D-01
0.9800	0.19800D-01	0.19800D-01
1.0000	0.12230D-15	0.47184D-15

COST FUNCTIONAL=.333333333330785D 00

LAGRANGIAN=.333333333330785D 00

```
      IMPLICIT REAL*8(A-H,O-Z)
      DOUBLE PRECISION M,MU,MUB
      DIMENSION X(101),C(24),ALPHA(101),Y(101),Z(101),DLAMD1(101),
     1DLAMD2(101),DLAMD3(101),DLAMD4(101)
      COMMON BETA,AN,AN1,AN2,AN3,AN4,AN5,AN6,AN7,AB,DELT,MU,
     1       M(24,24),W(7,101),DW(7,101),WH(4,101),DWH(4,101),
     2       DD(101),DF(101),DF1(101),DUM(101),DUM1(101),DUM2(101),
     1       DUM3(101),NPOINT
      REAL*8 LAMDA1(101),LAMDA2(101),LAMDA3(101),LAMDA4(101),LENGTH
      NINTV=4
      DELT=0.01D0
      LENGTH=1.0D0
      AN=2.D0
      MUB=1.D-02
      MU=MUB**(AN+1.D0)
      WRITE(6,10)MUB
   10 FORMAT('1','  CONSTRAINT AT RIGHT END OF THE BEAM',E10.3//)
      BETA=3.515D0**2
      WRITE(6,15)NINTV,DELT
   15 FORMAT('    NO. OF SPLINE INTERVALS:',I2,', STEPSIZE DELT=',F10.4/)
      WRITE(6,20)LENGTH,BETA
   20 FORMAT('    LENGTH OF BEAM=',F10.4,', BETA=',F12.5///)
      AN1=AN/(AN+1.D0)
      AN2=(-2.D0*AN-1.D0)/(AN+1.D0)
      AN3=(2.D0*AN+1.D0)/(AN+1.D0)
      AN4=-1.D0/(AN+1.D0)
      AN5=AN+1.D0
      AN6=(-3.D0*AN-2.D0)/(AN+1.D0)
      AN7=(-AN-2.D0)/(AN+1.D0)
      AB=(2.D0*BETA)/(AN+1.D0)
      X(1)=0.0
      NPOINT=LENGTH/DELT+1
      DO 25 K=2,NPOINT
      X(K)=X(K-1)+DELT
   25 CONTINUE
      NSTEP=NPOINT-1
      LINTV=NSTEP/NINTV
      H=LENGTH/NINTV
      CALL SPLIN(X,H,W,WH,LINTV)
      CALL DSPLIN(X,H,DW,DWH,LINTV)
      DO 45 I=13,16
      DO 35J=1,2
      DO 30 K=1,NPOINT
      Y(K)=W(I-12,K)*WH(J,K)
   30 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      M(I,J)=-2.D0*Z(NPOINT)
   35 CONTINUE
      DO 45 J=3,6
      DO 40 K=1,NPOINT
      Y(K)=W(I-12,K)*W(J+1,K)
   40 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      M(I,J)=-2.D0*Z(NPOINT)
```

```
 45   CONTINUE
      DO 65 I=17,18
      DO 55 J=1,2
      DO 50 K=1,NPOINT
      Y(K)=WH(21-I,K)*WH(J,K)
 50   CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      M(I,J)=-2.D0*Z(NPOINT)
 55   CONTINUE
      DO 65 J=3,6
      DO 60 K=1,NPOINT
      Y(K)=WH(21-I,K)*W(J+1,K)
 60   CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      M(I,J)=-2.D0*Z(NPOINT)
 65   CONTINUE
      DO 85 I=1,4
      DO 75 J=1,2
      DO 70 K=1,NPOINT
      Y(K)=W(I,K)*DWH(J,K)-DW(I,K)*WH(J,K)
 70   CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      M(I,J)=Z(NPOINT)
 75   CONTINUE
      DO 85 J=3,6
      DO 80 K=1,NPOINT
      Y(K)=W(I,K)*DW(J+1,K)-DW(I,K)*W(J+1,K)
 80   CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      M(I,J)=Z(NPOINT)
 85   CONTINUE
      DO 105 I=5,6
      DO 95 J=1,2
      DO 90 K=1,NPOINT
      Y(K)=WH(9-I,K)*DWH(J,K)-DWH(9-I,K)*WH(J,K)
 90   CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      M(I,J)=Z(NPOINT)
 95   CONTINUE
      DO 105 J=3,6
      DO 100 K=1,NPOINT
      Y(K)=WH(9-I,K)*DW(J+1,K)-DWH(9-I,K)*W(J+1,K)
100   CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      M(I,J)=Z(NPOINT)
105   CONTINUE
      DO 110 I=13,18
      DO 110 J=7,12
      M(I,J)=-M(I-12,J-6)
110   CONTINUE
      DO 115 I=19,24
      DO 115 J=13,18
      M(I,J)=M(J,I-18)
115   CONTINUE
```

```
      DO 120 I=19,24
      DO 120 J=19,24
      M(I,J)=M(J-18,I-18)
  120 CONTINUE
      DO 125 I=7,12
      DO 125 J=13,18
      M(I,J)=-M(I+12,J+6)
  125 CONTINUE
    1 READ(5,130,END=1000)(C(I),I=1,24)
  130 FORMAT(8F10.5)
      CALL SOLVE(24,C)
      C7=-4.D0*C(6)-C(5)
      D7=-C(11)-4.D0*C(12)
      E1=-4.D0*C(13)-C(14)
      F1=-4.D0*C(19)-C(20)
      WRITE(6,195)
  195 FORMAT(///'   THE COEFFICIENTS OF SPLINE FUNCTIONS C(I)(I=1,...,28
     1) ARE'/)
      WRITE(6,200)(C(I),I=1,6),C7,(C(I),I=7,12),D7,E1,(C(I),I=13,18),F1,
     1(C(I),I=19,24)
  200 FORMAT(' ',7E12.4)
      WRITE(6,135)
  135 FORMAT('1','   SPAN OF','       AREA')
      WRITE(6,140)
  140 FORMAT('      BEAM','     DISTRIBUTION   DEFLECTION'
     1,'      HAMILTONIAN'/)
      DO 150 K=1,NPOINT
      A1=0.0
      A2=0.0
      A3=0.0
      A4=0.0
      DA1=0.0
      DA2=0.0
      DA3=0.0
      DA4=0.0
      DO 145 I=1,6
      A1=A1+C(I)*W(I,K)
      A2=A2+C(I+6)*W(I,K)
      A3=A3+C(I+12)*W(I+1,K)
      A4=A4+C(I+18)*W(I+1,K)
      DA1=DA1+C(I)*DW(I,K)
      DA2=DA2+C(I+6)*DW(I,K)
      DA3=DA3+C(I+12)*DW(I+1,K)
      DA4=DA4+C(I+18)*DW(I+1,K)
  145 CONTINUE
      LAMDA1(K)=A1+C7*W(7,K)
      LAMDA2(K)=A2+D7*W(7,K)
      LAMDA3(K)=A3+E1*W(1,K)
      LAMDA4(K)=A4+F1*W(1,K)
      DLAMD1(K)=DA1+C7*DW(7,K)
      DLAMD2(K)=DA2+D7*DW(7,K)
      DLAMD3(K)=DA3+E1*DW(1,K)
      DLAMD4(K)=DA4+F1*DW(1,K)
      ALPHA(K)=((AN*LAMDA2(K)**2)/(1.D0+BETA*LAMDA4(K)**2)+MU)**(1.D0/
```

```
     1(AN+1.D0))
      Y(K)=ALPHA(K)-2.D0*LAMDA1(K)*LAMDA3(K)+(1.D0/ALPHA(K)**2)
     1      *LAMDA2(K)**2+ALPHA(K)*BETA*LAMDA4(K)**2
  150 CONTINUE
      DO 155 K=1,NPOINT,2
      WRITE(6,190)X(K),ALPHA(K),LAMDA4(K),Y(K)
  190 FORMAT(F10.4,9E15.5)
  155 CONTINUE
      DO 160 K=1,NPOINT
      Y(K)=ALPHA(K)-2.D0*LAMDA1(K)*LAMDA3(K)+(1.D0/(ALPHA(K)**2))*
     1LAMDA2(K)**2+ALPHA(K)*BETA*LAMDA4(K)**2+DLAMD1(K)*LAMDA4(K)-
     1DLAMD2(K)*LAMDA3(K)+DLAMD3(K)*LAMDA2(K)-LAMDA1(K)*DLAMD4(K)
  160 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      ALAG=Z(NPOINT)
      CALL DQSF(DELT,ALPHA,Z,NPOINT)
      BLAG=Z(NPOINT)
      WRITE(6,165)BLAG
  165 FORMAT(/'   COST FUNCTIONAL=',E20.15)
      WRITE(6,185)ALAG
  185 FORMAT('          LAGRANGIAN=',E20.15//)
      DO 170 K=1,NPOINT
      Y(K)=(LAMDA2(K)**2)/(ALPHA(K)**AN)
  170 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      OMEG=Z(NPOINT)
      DO 175 K=1,NPOINT
      Y(K)=ALPHA(K)*LAMDA4(K)**2
  175 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      OMEG=OMEG/Z(NPOINT)
      WRITE(6,180)OMEG
  180 FORMAT('   BETA OBTAINED FROM RAYLIEGH QUOTIENT = ',F10.5)
      GO TC 1
 1000 STOP
      END
```

```
      SUBROUTINE SOLVE(N,X)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(24),FN(24),DFN(24,24),ALPHA(20),PRE(100),PR(20),
     1  R(24),XX(24)
      EPS1=1.D-26
      WRITE(6,10)
   10 FORMAT('   INITIAL GUESSES AT THE SPLINE COEFFICIENTS'/)
      WRITE(6,15)(X(I),I=1,24)
   15 FORMAT('  ',6E12.4)
      DO 75 KK=1,50
      CALL FUNC(N,X,FN)
      CALL DFUNC(N,X,DFN)
      DO 20 I=1,N
      XX(I)=X(I)
      R(I)=-FN(I)
   20 CONTINUE
      CALL DGELG(R,DFN,24,1,1.D-16,IER)
      PRE(KK)=0.0D0
      DO 30 I=1,N
      PRE(KK)=PRE(KK)+FN(I)*FN(I)
   30 CONTINUE
      IF(KK.GT.1) GO TO 40
      WRITE(6,35)PRE(KK)
   35  FORMAT(//'   ERROR IN INITIAL GUESSES=',E15.5/)
   40 ALPHA(1)=1.0D0
      DO 55 J=2,20
      PR(J)=0.0D0
      DO 45 I=1,N
      M=J-1
      X(I)=XX(I)+ALPHA(M)*R(I)
   45 CONTINUE
      CALL FUNC(N,X,FN)
      DO 50 I=1,N
      PR(J)=PR(J)+FN(I)*FN(I)
   50 CONTINUE
      M=J-1
      IF(PR(J).LT.PRE(KK))GO TO 65
      ALPHA(J)=ALPHA(J-1)/2.0D0
   55 CONTINUE
      WRITE(6,60)ALPHA(M),KK
   60 FORMAT(' NONCONVERGENCE BECAUSE OF TOO MANY BISECTIONS OF STEPSIZE
     1,ALPHA=',F10.8,'AFTER',I3,'ITERATIONS')
      GO TO 200
   65 WRITE(6,70)KK,PR(J),ALPHA(M)
   70 FORMAT('   ITERATION NO.',I3,5X,' CUMULATIVE ERROR=',E12.5,', ALPH
     1A=',F10.8//)
      IF(PR(J).LT.EPS1)GO TO 85
      DO 75 I=1,N
      X(I)=XX(I)+ALPHA(M)*R(I)
   75 CONTINUE
      WRITE(6,80)KK
   80 FORMAT(' NO CONVERGENCE AFTER ',I3,' ITERATIONS')
      GO TO 200
   85 WRITE(6,90)PR(J),KK
```

```
   90  FORMAT('   CONVERGENCE. CUMULATIVE ERROR=',E12.5,' AFTER',I3,' ITE
      1RATIONS')
  200  RETURN
       END
```

```
      SUBROUTINE FUNC(N,C,FN)
      IMPLICIT REAL*8(A-H,O-Z)
      DOUBLE PRECISION M,MU
      DIMENSION C(24),FN(24),Y(101),Z(101),Y1(101)
      COMMON BETA,AN,AN1,AN2,AN3,AN4,AN5,AN6,AN7,AB,DELT,MU,
     1       M(24,24),W(7,101),DW(7,101),WH(4,101),DWH(4,101),
     2       DD(101),DF(101),DF1(101),DUM(101),DUM1(101),DUM2(101),
     1       DUM3(101),NPOINT
      DO 10 K=1,NPOINT
      DD(K)=W(1,K)*C(7)+W(2,K)*C(8)+W(3,K)*C(9)
     1      +W(4,K)*C(10)+WH(4,K)*C(11)+WH(3,K)*C(12)
      DF(K)=WH(1,K)*C(19)+WH(2,K)*C(20)+W(4,K)*C(21)
     1      +W(5,K)*C(22)+W(6,K)*C(23)+W(7,K)*C(24)
      DF1(K)=BETA*DF(K)
      DUM(K)=1.D0+DF(K)*DF1(K)
      DUM1(K)=AN*DD(K)**2+MU*DUM(K)
      DUM2(K)=AN*DD(K)**3+AN3*MU*DD(K)*DUM(K)
      DUM3(K)=(AN5*DD(K)**2+MU*DUM(K))*DF(K)
   10 CONTINUE
      DO 20 I=1,12
      J=I+12
      IF(I.GT.6) GO TO 15
      FN(I)=M(J,1)*C(13)+M(J,2)*C(14)+M(J,3)*C(15)
     1      +M(J,4)*C(16)+M(J,5)*C(17)+M(J,6)*C(18)
     1      +M(J,7)*C(19)+M(J,8)*C(20)+M(J,9)*C(21)
     1      +M(J,10)*C(22)+M(J,11)*C(23)+M(J,12)*C(24)
      GO TO 20
   15 FN(I)=M(J,13)*C(1)+M(J,14)*C(2)+M(J,15)*C(3)
     1      +M(J,16)*C(4)+M(J,17)*C(5)+M(J,18)*C(6)
     1      +M(J,19)*C(7)+M(J,20)*C(8)+M(J,21)*C(9)
     1      +M(J,22)*C(10)+M(J,23)*C(11)+M(J,24)*C(12)
   20 CONTINUE
      DO 25 K=1,NPOINT
      Y1(K)=AN*(DUM(K)**AN4)*(DUM1(K)**(-AN1))*DUM3(K)
     1     +MU*MU*(DUM(K)**AN3)*DF(K)*(DUM1(K)**AN2)
   25 CONTINUE
      DO 40 I=1,6
      DO 30 K=1,NPOINT
      IF(I.GE.3) Y(K)=Y1(K)*W(I+1,K)
      IF(I.LT.3) Y(K)=Y1(K)*WH(I,K)
   30 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      FN(I+12)=AB*Z(NPOINT)
   40 CONTINUE
      DO 45 K=1,NPOINT
      Y1(K)=(DUM(K)**AN1)*(DUM1(K)**AN2)*DUM2(K)
   45 CONTINUE
      DO 60 I=1,6
      DO 50 K=1,NPOINT
      IF(I.LE.4) Y(K)=Y1(K)*W(I,K)
      IF(I.GT.4) Y(K)=Y1(K)*WH(9-I,K)
   50 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      FN(I+18)=2.D0*Z(NPOINT)
```

```
   60 CONTINUE
      DO 70 J=13,18
      FN(13)=FN(13)+M(7,J)*C(J-12)
      FN(14)=FN(14)+M(8,J)*C(J-12)
      FN(15)=FN(15)+M(9,J)*C(J-12)
      FN(16)=FN(16)+M(10,J)*C(J-12)
      FN(17)=FN(17)+M(11,J)*C(J-12)
      FN(18)=FN(18)+M(12,J)*C(J-12)
   70 CONTINUE
      DO 80 J=1,6
      FN(19)=FN(19)+M(1,J)*C(J+12)
      FN(20)=FN(20)+M(2,J)*C(J+12)
      FN(21)=FN(21)+M(3,J)*C(J+12)
      FN(22)=FN(22)+M(4,J)*C(J+12)
      FN(23)=FN(23)+M(5,J)*C(J+12)
      FN(24)=FN(24)+M(6,J)*C(J+12)
   80 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DFUNC(N,C,DFN)
      IMPLICIT REAL*8(A-H,O-Z)
      DOUBLE PRECISION M,MU
      DIMENSION C(24),DFN(24,24),Y(101),Z(101),Y1(101)
      COMMON BETA,AN,AN1,AN2,AN3,AN4,AN5,AN6,AN7,AB,DELT,MU,
     1       M(24,24),W(7,101),DW(7,101),WH(4,101),DWH(4,101),
     2       DD(101),DF(101),DF1(101),DUM(101),DUM1(101),DUM2(101),
     1       DUM3(101),NPOINT
C           I=1,.....,6
      DO 20 I=1,6
      II=I+12
      DO 15 J=1,24
      IF(J.LT.13) GO TO 10
      DFN(I,J)=M(II,J-12)
      GO TO 15
   10 DFN(I,J)=0.0D0
   15 CONTINUE
   20 CONTINUE
C           I=7,........,12
      DO 40 I=7,12
      II=I+12
      DO 35 J=1,24
      IF(J.GT.12) GO TO 30
      DFN(I,J)=M(II,J+12)
      GO TO 35
   30 DFN(I,J)=0.0D0
   35 CONTINUE
   40 CONTINUE
C           I=13,.......,18
      DO 60 I=13,18
      DO 55 J=13,18
      DFN(I,J)=0.0D0
   55 CONTINUE
   60 CONTINUE
      DO 70 I=13,18
      II=I-6
      DO 65 J=1,6
      DFN(I,J)=M(II,J+12)
   65 CONTINUE
   70 CONTINUE
C           I=19,........,24
      DO 80 I=19,24
      II=I-18
      DO 75 J=13,18
      DFN(I,J)=M(II,J-12)
   75 CONTINUE
   80 CONTINUE
      DO 90 I=19,24
      II=I-18
      DO 85 J=1,6
      DFN(I,J)=0.0D0
   85 CONTINUE
   90 CONTINUE
      DO 95 K=1,NPOINT
```

```
      Y1(K)=AN*(DUM(K)**AN4)*(-AN1*(DUM1(K)**AN2)*2.D0*AN*DD(K)
     1         *DUM3(K)+(DUM1(K)**(-AN1))*(2.D0*AN5*DD(K)*DF(K)))
     1      +MU*MU*(DUM(K)**AN3)*DF(K)*(AN2*(DUM1(K)**AN6)*2.D0*AN*DD(K))
   95 CONTINUE
      DO 120 I=13,18
      DO 110 J=1,6
      DO 100 K=1,NPOINT
      IF(I.GE.15) Y(K)=Y1(K)*W(I-12+1,K)
      IF(I.LT.15) Y(K)=Y1(K)*WH(I-12,K)
      IF(J.LE.4) Y(K)=Y(K)*W(J,K)
      IF(J.GT.4) Y(K)=Y(K)*WH(9-J,K)
  100 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      DFN(I,J+6)=AB*Z(NPOINT)
  110 CONTINUE
  120 CONTINUE
      DO 130 K=1,NPOINT
      Y1(K)=AN*(AN4*(DUM(K)**AN7)*2.D0*DF1(K)*(DUM1(K)**(-AN1))
     1         *DUM3(K)+(DUM(K)**AN4)*(-AN1)*(DUM1(K)**AN2)
     1         *2.D0*MU*DF1(K)*DUM3(K)+(DUM(K)**AN4)*(DUM1(K)**(-AN1))
     1         *(DUM3(K)/DF(K)+2.D0*MU*DF1(K)*DF(K)))
     1      +MU*MU*(AN3*(DUM(K)**AN1)*2.D0*DF1(K)*DF(K)*(DUM1(K)**AN2)
     1              +(DUM(K)**AN3)*(DUM1(K)**AN2)
     1              +(DUM(K)**AN3)*DF(K)*(AN2*(DUM1(K)**AN6)*2.D0*MU
     1                                                   *DF1(K)))
  130 CONTINUE
      DO 160 I=13,18
      DO 150 J=1,6
      DO 140 K=1,NPOINT
      IF(I.GE.15) Y(K)=Y1(K)*W(I-12+1,K)
      IF(I.LT.15) Y(K)=Y1(K)*WH(I-12,K)
      IF(J.GE.3) Y(K)=Y(K)*W(J+1,K)
      IF(J.LT.3) Y(K)=Y(K)*WH(J,K)
  140 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      DFN(I,J+18)=AB*Z(NPOINT)
  150 CONTINUE
  160 CONTINUE
      DO 165 K=1,NPOINT
      Y1(K)=(DUM(K)**AN1)*(AN2*(DUM1(K)**AN6)*2.D0*AN*DD(K)
     1      *DUM2(K)+(DUM1(K)**AN2)*(3.D0*AN*(DD(K)**2)+AN3
     1      *MU*DUM(K)))
  165 CONTINUE
      DO 190 I=19,24
      DO 180 J=1,6
      DO 170 K=1,NPOINT
      IF(I.LE.22) Y(K)=Y1(K)*W(I-18,K)
      IF(I.GT.22) Y(K)=Y1(K)*WH(27-I,K)
      IF(J.LE.4) Y(K)=Y(K)*W(J,K)
      IF(J.GT.4) Y(K)=Y(K)*WH(9-J,K)
  170 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      DFN(I,J+6)=2.D0*Z(NPOINT)
  180 CONTINUE
```

```
  190 CONTINUE
      DO 195 K=1,NPOINT
      Y1(K)=AN1*(DUM(K)**AN4)*2.D0*DF1(K)*((DUM1(K)**AN2)*DUM2(K))
     1      +(DUM(K)**AN1)*(AN2*(DUM1(K)**AN6)*2.D0*MU*DF1(K)
     1      *DUM2(K)+(DUM1(K)**AN2)*AN3*MU*DD(K)*2.D0*DF1(K))
  195 CONTINUE
      DO 220 I=19,24
      DO 210 J=1,6
      DO 200 K=1,NPOINT
      IF(I.LE.22) Y(K)=Y1(K)*W(I-18,K)
      IF(I.GT.22) Y(K)=Y1(K)*WH(27-I,K)
      IF(J.GE.3) Y(K)=Y(K)*W(J+1,K)
      IF(J.LT.3) Y(K)=Y(K)*WH(J,K)
  200 CONTINUE
      CALL DQSF(DELT,Y,Z,NPOINT)
      DFN(I,J+18)=2.D0*Z(NPOINT)
  210 CONTINUE
  220 CONTINUE
      RETURN
      END
```

```
CONSTRAINT AT RIGHT END OF THE BEAM 0.100D-01

NO. OF SPLINE INTERVALS: 4, STEPSIZE DELT=    0.0100

LENGTH OF BEAM=    1.0000, BETA=    12.35523


INITIAL GUESSES AT THE SPLINE COEFFICIENTS

 0.1440D 02  0.1152D 02  0.8640D 01  0.5760D 01  0.2880D 01  0.0
 0.8880D 01  0.5640D 01  0.3120D 01  0.1320D 01  0.2400D 00 -0.1200D 00
 0.0        -0.1600D 02 -0.3200D 02 -0.4800D 02 -0.6400D 02 -0.8000D 02
-0.6667D 00  0.1333D 01  0.7333D 01  0.1733D 02  0.3133D 02  0.4933D 02

ERROR IN INITIAL GUESSES=    0.69822D 00

ITERATION NO.  1      CUMULATIVE ERROR= 0.95580D-02, ALPHA=1.00000000

ITERATION NO.  2      CUMULATIVE ERROR= 0.26252D-03, ALPHA=1.00000000

ITERATION NO.  3      CUMULATIVE ERROR= 0.13526D-05, ALPHA=1.00000000

ITERATION NO.  4      CUMULATIVE ERROR= 0.28862D-11, ALPHA=1.00000000

ITERATION NO.  5      CUMULATIVE ERROR= 0.23514D-22, ALPHA=1.00000000

ITERATION NO.  6      CUMULATIVE ERROR= 0.21403D-31, ALPHA=1.00000000

CONVERGENCE. CUMULATIVE ERROR= 0.21403D-31 AFTER  6 ITERATIONS


THE COEFFICIENTS OF SPLINE FUNCTIONS C(I)(I=1,...,28) ARE

0.1484D 01 0.1449D 01 0.1444D 01 0.9321D 00 0.3333D 00 0.1182D 00-0.8061D 00
0.1229D 01 0.8622D 00 0.4961D 00 0.1867D 00 0.3762D-01-0.1690D-01 0.2997D-01
0.4089D 01-0.4970D-01-0.3890D 01-0.9778D 01-0.1854D 02-0.7760D 02-0.2870D 02
0.2608D 00-0.1392D 00 0.2962D 00 0.2016D 01 0.5215D 01 0.1569D 02 0.4015D 02
```

SPAN OF BEAM	AREA DISTRIBUTION	DEFLECTION	HAMILTONIAN
0.0	0.23557D 00	-0.52042D-17	0.35335D 00
0.0200	0.23020D 00	0.38708D-03	0.35342D 00
0.0400	0.22478D 00	0.12970D-02	0.35338D 00
0.0600	0.21929D 00	0.27513D-02	0.35326D 00
0.0800	0.21374D 00	0.47715D-02	0.35309D 00
0.1000	0.20814D 00	0.73792D-02	0.35290D 00
0.1200	0.20246D 00	0.10596D-01	0.35272D 00
0.1400	0.19671D 00	0.14443D-01	0.35259D 00
0.1600	0.19089D 00	0.18942D-01	0.35252D 00
0.1800	0.18497D C0	0.24115D-01	0.35253D 00
0.2000	0.17897D 00	0.29983D-01	0.35263D 00
0.2200	0.17286D 00	0.36568D-01	0.35282D 00
0.2400	0.16664D 00	0.43891D-01	0.35309D 00
0.2600	0.16031D 00	0.51974D-01	0.35343D 00
0.2800	0.15387D 00	0.60831D-01	0.35381D 00
0.3000	0.14731D 00	0.70473D-01	0.35421D 00
0.3200	0.14066D 00	0.80909D-01	0.35461D 00
0.3400	0.13391D 00	0.92149D-01	0.35499D 00
0.3600	0.12711D 00	0.10420D 00	0.35532D 00
0.3800	0.12025D 00	0.11707D 00	0.35556D 00
0.4000	0.11339D 00	0.13078D 00	0.35566D 00
0.4200	0.10654D 00	0.14533D 00	0.35559D 00
0.4400	0.99761D-01	0.16072D 00	0.35529D 00
0.4600	0.93079D-01	0.17698D 00	0.35470D 00
0.4800	0.86540D-01	0.19410D 00	0.35377D 00
0.5000	0.80190D-01	0.21210D 00	0.35245D 00
0.5200	0.74062D-01	0.23104D 00	0.35078D 00
0.5400	0.68163D-01	0.25114D 00	0.34905D 00
0.5600	0.62492D-01	0.27268D 00	0.34758D 00
0.5800	0.57055D-01	0.29595D 00	0.34660D 00
0.6000	0.51858D-01	0.32121D 00	0.34624D 00
0.6200	0.46911D-01	0.34876D 00	0.34659D 00
0.6400	0.42227D-01	0.37886D 00	0.34763D 00
0.6600	0.37816D-01	0.41180D 00	0.34931D 00
0.6800	0.33690D-01	0.44784D 00	0.35148D 00
0.7000	0.29860D-01	0.48728D 00	0.35397D 00
0.7200	0.26335D-01	0.53039D 00	0.35657D 00
0.7400	0.23124D-01	0.57745D 00	0.35905D 00
0.7600	0.20236D-01	0.62873D 00	0.36120D 00
0.7800	0.17689D-01	0.68454D 00	0.36239D 00
0.8000	0.15505D-01	0.74520D 00	0.36179D 00
0.8200	0.13709D-01	0.81103D 00	0.35903D 00
0.8400	0.12318D-01	0.88235D 00	0.35448D 00
0.8600	0.11324D-01	0.95948D 00	0.34939D 00
0.8800	0.10681D-01	0.10428D 01	0.34546D 00
0.9000	0.10309D-01	0.11325D 01	0.34391D 00
0.9200	0.10120D-01	0.12290D 01	0.34489D 00
0.9400	0.10037D-01	0.13326D 01	0.34775D 00
0.9600	0.10008D-01	0.14436D 01	0.35176D 00
0.9800	0.10001D-01	0.15624D 01	0.35658D 00
1.0000	0.10000D-01	0.16893D 01	0.36257D 00

COST FUNCTIONAL=.956090618643043D-01
LAGRANGIAN=.954169346256854D-01

BETA OBTAINED FROM RAYLIEGH QUOTIENT = 11.82865